



Université de Toulouse

MASTER 2 GEOMATIQUE

« Science de l'Information Géoréférencée pour la Maîtrise de l'environnement et l'Aménagement des territoires » (SIGMA)

<http://sigma.univ-toulouse.fr>

RAPPORT DE STAGE

Tuilage et Génération de Tile Map Service à l'échelle Nationale

Fatou Sylla



Office National de la Chasse et de la Faune Sauvage

Maître de stage : Onésime Prudhomme
Tuteur-enseignant : Laurent Jégou

Septembre 2018

Résumé

L'Office National de la Chasse et de la Faune Sauvage (ONCFS) est un établissement public à caractère administratif sous la double tutelle du ministère chargé de l'Environnement et du ministère chargé de l'Agriculture. Il est en charge de la connaissance de la faune sauvage et de ses habitats.

L'ONCFS possède une cellule SIG et Développement, qui développe des applications web et mobiles permettant à ses agents de saisir leurs observations faites sur le terrain directement au format numérique et standardisé. Ces applications sont développées à l'aide des générateurs d'applications GEO et Aigle proposé par Business Geografic.

Ces applications géographiques reposent actuellement sur des fonds cartographiques basées sur les produits de l'IGN (Institut National de l'Information Géographique et Forestière). Elles permettent de stocker les fonds de carte sur les appareils mobiles afin de permettre aux agents sur le terrain de pouvoir saisir directement leurs observations même en zone non couverte par l'échange de données mobiles.

C'est dans ce cadre que ce stage s'inscrit afin de générer à l'échelle nationale un tuilage en TMS (Tile Map Service) des principaux fonds cartographiques couramment utilisées dans les applications web et mobiles déconnectés. Le tuilage¹ consiste en un re-découpage optimisé du raster en petites images à des échelles prédéfinies pour être utilisées par des outils web.

La génération des tuiles a été réalisée avec le serveur cartographique mapserver et plus particulièrement mapcache qui implémente la mise en cache de tuiles pour fluidifier la consultation des fonds cartographiques.

Lors de ce stage, les tuiles pour des cartes topographiques (séries Scan de l'IGN) et photos aériennes (BD Ortho de l'IGN) ont pu être générées sur toute la France métropolitaine.

Un document explicatif de toute la procédure de tuilage a été créé pour permettre à l'ONCFS de pouvoir régénérer les tuiles à l'avenir.

¹ <http://blog.martzluff.net/category/tutoriels-sig/page/2/>

Abstract

The ONCFS (National Office of Hunting and Wildlife) is a public administrative institution under the dual authority of the Ministry of the Environment and the Ministry of Agriculture. Its main objective is to have in charge the knowledge of the wildlife and the habitats.

ONCFS has a GIS and Development team, which develops web and mobile applications allowing its agents to digitalize directly their observations made on the field with an standardized format. These solutions are developed with GEO generators and Aigle, which are proposed by Business Geografic.

These geographic applications are based on cartographic databases from the products of IGN (National Geographical Institut). They make possible to storage of maps bases on mobile devices in order to allow the agents on the field to directly enter their observations.

This context sets the general framework of the internship. The goal is to generate a nationwide TMS tileset (Tile Map Service) of the main cartographic databases commonly used in the web and in the mobile applications offline. The tiling² consists of an optimized re-splitting of the raster into small images at predefined scales for use by web tools.

The tile generation has been done with Mapserver cartographic server and more particularly with mapcache who implements the caching of tiles, which facilitates the consultation of cartographic backgrounds.

During this work experience, tiles for topographic maps (IGN Scan series) and aerial photos (IGN BD Ortho) could be generated for the metropolitan area of France.

A document exposing the entire tiling procedure was created to allow the agents of the ONCFS to regenerate the tiles in the future.

² <http://blog.martzluff.net/category/tutoriels-sig/page/2/>

Remerciements

Je tiens à remercier Onésime Prudhomme et Margaux Joulain de m'avoir offert l'opportunité de travailler sur ce projet.

Je tiens également à remercier mon tuteur enseignant Monsieur Laurent Jégou pour son expertise, ses conseils et son accompagnement sans relâche malgré sa charge de travail tout au long de mon stage.

Je remercie aussi tout le personnel de l'ONCFS pour leur accueil chaleureux et leur disponibilité et plus particulièrement à Jérémie Gooding.

Je remercie aussi Yves Jacolin de CamptoCamp qui a participé à la bonne marche de ce stage. Mes remerciements vont aussi à l'endroit de toute l'équipe pédagogique du Master SIGMA pour la qualité de leur enseignement.

Sommaire

| | |
|--|----|
| Résumé | 2 |
| Abstract | 3 |
| Remerciements | 4 |
| Sommaire | 5 |
| Introduction | 7 |
| 1. Contexte du projet | 9 |
| 1.1. Contexte Général..... | 9 |
| 1.2. Présentation de la structure d'accueil..... | 9 |
| 1.3 Les besoins techniques..... | 12 |
| 1.4 Contexte thématique | 13 |
| 2. Le webmapping | 16 |
| 2.1. Webmapping et interopérabilité..... | 16 |
| 2.2. Les web services | 17 |
| 2.3. Les web services géo spatiaux..... | 17 |
| 2.4. Notion de tuilage | 20 |
| 2.5. Le stockage des tuiles en cache..... | 21 |
| 2.6. État de l'art..... | 22 |
| 2.7. Installation et paramétrage de MapServer et MapCache..... | 25 |
| 2.8. Configuration Vhost Apache..... | 26 |
| 2.9. Principales fonctionnalités de mapcache..... | 27 |
| 3. Mise en œuvre des tuiles | 28 |
| 3.1 Acquisition des données..... | 28 |
| 3.3 Serveur de tuile | 30 |
| 3. 4. La création des fichiers mapfile et mapcache.xml | 32 |
| 3. 5. La génération des tuiles pour les scans | 35 |
| 3.6. Optimisation | 39 |

| | |
|--|----|
| 3.7. La génération des tuiles pour les orthophotos | 42 |
| 4. Planning prévisionnel | 45 |
| 5. Difficultés rencontrées | 46 |
| 6. Conclusion et perspectives | 47 |
| Bibliographie et webographie | 49 |
| Table des illustrations..... | 51 |
| Table des tableaux | 52 |

Introduction

Le webmapping (diffusion de données spatiale par internet) a connu des évolutions au cours des années avec de nouvelles méthodes et de nouveaux outils. De la carte statique, c'est à dire que les cartes étaient présentées sous forme d'images plus ou moins interactives sur une page web, il est passé à la carte dynamique et interrogeable pour tout le monde.

Ainsi, les navigateurs de carte avaient un défaut d'ergonomie ce qui fait qu'à chaque fois qu'on modifiait le contenu d'une page web, il fallait réactiver l'image, ce qui causait un problème aux utilisateurs. Mais l'arrivée de Google maps en 2005³ a changé la donne en introduisant le concept de carte glissante ou « slippy map » en coupant court à ce défaut ergonomique. Cette révolution a entraîné l'évolution du webmapping que l'on connaît de nos jours. Ainsi il est devenu plus accessible grâce au développement des outils open source qui sont généralement gratuits et libres d'utilisation.

Le Webmapping aujourd'hui n'est plus aussi monolithique ni aussi lourd et complexe à mettre en place on trouve une diversité beaucoup plus grande de solutions logicielles, qui sont assez fragmentées, mais très complémentaires entre elles. Au final ces solutions permettent une meilleure adaptation aux besoins et l'accès à de nouvelles fonctions comme les services web cartographiques et des interfaces utilisateur plus complètes⁴.

Grâce à cette évolution, se déplacer sur une carte devient de plus en plus facile. Les déplacements latéraux étaient continus, fluides grâce à un mécanisme de pré chargement de la zone jouxtant la partie visible de la carte. Les déplacements verticaux (zoom avant ou arrière) étaient rapides et ne nécessitaient pas de message d'attente car quasi immédiatement des petits bouts de l'image s'affichaient, laissant à entendre que le reste arriverait vite, ce qui était en effet le cas.⁵

Ces déplacements rapides on été rendus possibles grâce a une technologie de mise en cache. Le cache est un emplacement où sont stockées sur le terminal les tuiles sous format d'image qu'on peut visualiser avec ou sans connexion réseau et avec un gain de temps énorme.

³ <https://www.neogeo-online.net/blog/archives/1727/>

⁴ Support de cours M2 Sigma : Module 904_23

⁵ <https://www.neogeo-online.net/blog/archives/1727/>

L'office national de la chasse et de la faune sauvage développe des applications web et mobile afin de permettre à ces agents sur le terrain de saisir directement leurs observations. Ces applications doivent stocker les fonds de carte sur les appareils mobiles en mode déconnecté.

L'Institut national de l'information géographique et forestière (IGN) fournit ces fonds de carte pour les entreprises qui en ont besoin. Mais ils ne sont pas gratuits, ce sont des transactions web payantes qui nécessitent un budget conséquent en fonction du nombre de transactions. Ainsi, leurs coûts sont difficilement planifiables car ils varient selon le nombre de transactions. Par conséquent l'ONCFS a besoin de prévoir beaucoup de transactions non seulement pour ces agents mais aussi pour les structures partenaires avec lesquelles il travaille. Ce qui fait que ces coûts ne rentrent pas dans le budget actuel de l'ONCFS d'où la mise en place de ce projet.

L'objectif de ce stage pour la structure est donc de générer ses propres tuiles pour les fonds cartographiques couramment utilisés à l'échelle de la France métropolitaine pour que les agents qui travaillent en zones reculées et non couvertes en téléphonie mobile puissent utiliser les fonds de carte sur leurs tablettes.

Ce rapport présente les travaux de tuilage pour l'Office National de la Chasse et de la Faune Sauvage avec le serveur cartographique Mapserver/Mapcache. Il se divise en trois parties. La première partie présente le contexte du stage. La seconde partie énonce le webmapping, l'état de l'art des différentes solutions de mise en cache des tuiles et le paramétrage et la configuration de Mapserver/Mapcache. La dernière partie présente la mise en œuvre des tuiles et ses différentes étapes.

1. Contexte du projet

1.1. Contexte Général

Les 6 mois de stage se sont déroulés au sein de la cellule SIG et développement du département Assistant à Maitrise d’Ouvrage (AMOA) de la Direction des Systèmes d’Informations (DSI) à l’Office National de la Chasse et de la Faune Sauvage (ONCFS) de Toulouse.

1.2. Présentation de la structure d’accueil

L’ONCFS⁶ est un établissement public à caractère administratif sous la cotutelle du ministère chargé de l’Ecologie, du Développement Durable et de l’Energie, et du ministère chargé de l’Agriculture, de l’Agroalimentaire et de la Forêt. Il dispose d’un budget de 120 millions d’euros pour remplir ses missions sur l’ensemble du territoire (métropole et DOM) et compte 1000 agents techniques de l’environnement, 350 techniciens de l’environnement, 70 ingénieurs et techniciens, 80 cadres techniques, 156 personnels administratifs, 30 ouvriers et 25 inspecteurs du permis de chasser soit en tout 1700 personnes au service de la biodiversité⁷ :

- **Missions**

L’ONCFS remplit différentes missions :

- surveillance quotidienne des territoires et application de la réglementation de l’environnement par l’intermédiaire de la police de l’environnement et de la chasse exercée par les services départementaux.
- recherches et expertises réalisées sur la faune sauvage et ses habitats, assurées par la Direction de la recherche et de l’expertise (DRE) via ses Unités (Unité Avifaune migratrice, Unité Cervidés – Sanglier, Unité Faune de montagne, Unité Faune sédentaire de plaine, Unité Prédateurs et animaux déprédateurs, Unité sanitaire de la faune).
- appui technique et conseil apportés aux administrations, collectivités territoriales, gestionnaires et aménageurs du territoire (assurés par les Cellules techniques

⁶ <http://www.oncfs.gouv.fr>

⁷ ONCFS

interrégionales qui réalisent les animations scientifiques et techniques de différents programmes de gestion de la faune sauvage).

- gestion de la pratique de la chasse et organisation du permis de chasse.

Il a été créé en 1972, en remplacement du Conseil supérieur de la chasse. Avec la loi relative à la chasse de juillet 2000, il devient Office national de la chasse et de la faune sauvage (ONCFS).

Son fonctionnement est assuré par le Conseil d'administration et le Directeur général de l'établissement.

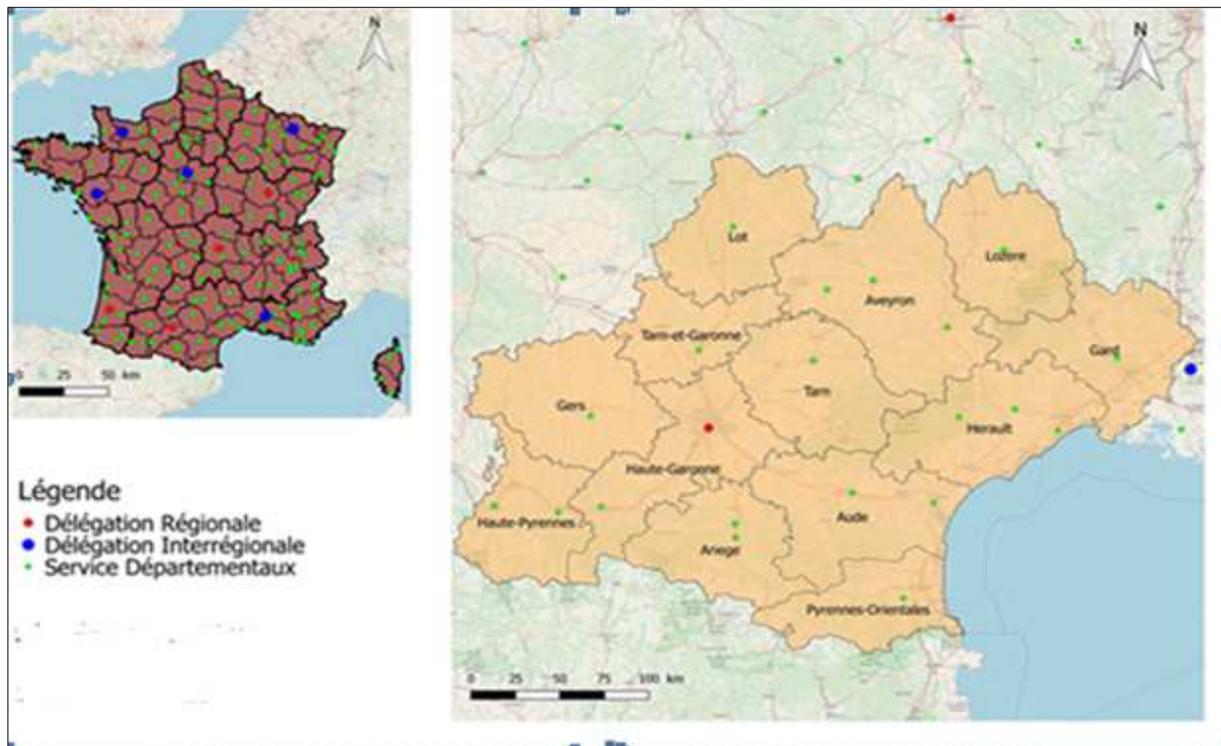


Figure 1: Implantations de l'ONCFS en France métropolitaine, Source : ONCFS

• Organisation

L'ONCFS est découpé en plusieurs directions à savoir :

- La Direction de la police (DP)
- La Direction de la recherche et de l'expertise (DRE)
- La Direction des ressources humaines (DRH)
- La Direction des affaires financières (DAF)
- La Direction des systèmes d'information (DSI)

La DSI a pour rôle de gérer les besoins de l'établissement en matière d'informatique et de système d'informations en prenant en compte les besoins de chaque structure et de respecter les priorités décidées par le Directeur général et proposées par le Comité de Management des systèmes d'information (CMSI).

Dans le cadre de l'urbanisation des données scientifiques et techniques de l'ONCFS, la DSI travaille en collaboration avec la Direction de la Recherche et de l'Expertise (DRE) qui accueille une cellule d'appui en matière d'analyse de données scientifiques. La DSI est organisée comme suit :

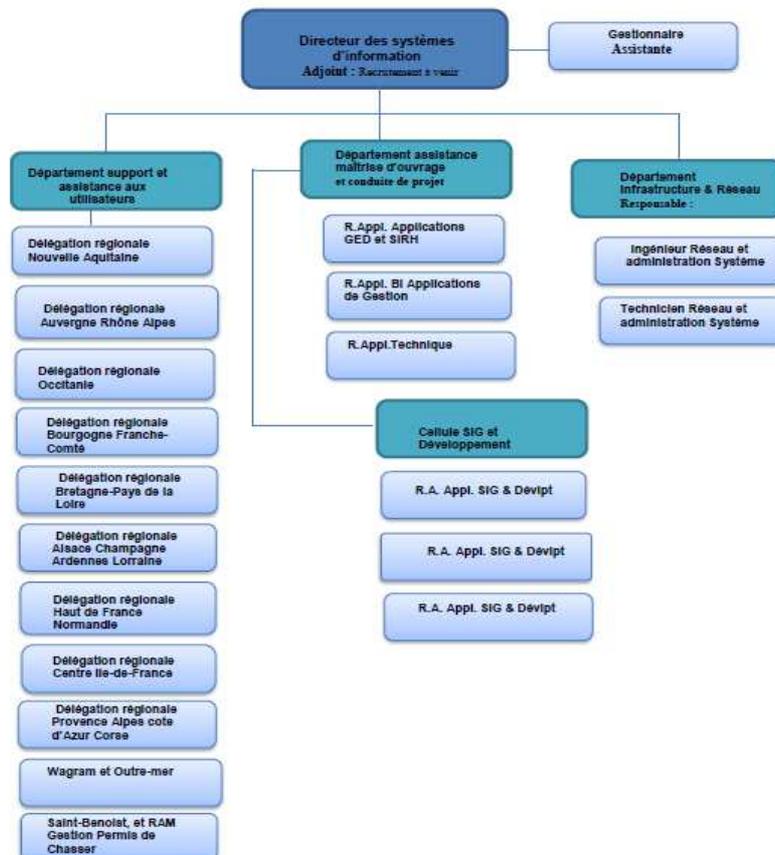


Figure 2: Organigramme Direction des Systèmes d'informations, Source : ONCFS, janvier 2018

La DSI englobe la cellule SIG et développement ou j'ai effectué mon stage. L'équipe est composée de deux ingénieurs en SIG et développement d'applications : Margaux Joulain et Onésime Prudhomme, ainsi qu'une développeuse informatique Clarisse Peyronnin.

Les missions de l'équipe sont :

- Centraliser les besoins en cartographie, SIG au niveau national
- Conseiller dans le domaine logiciel, de l'équipement informatique
- Etre un interlocuteur au niveau national avec les partenaires tels que l'IGN

- Organiser, actualiser, mettre à disposition les ressources géographiques.

Ainsi, les ingénieurs et chercheurs de la structure utilisent les applications SIG bureautiques (qgis, arcgis, R etc.) et les agents de terrain les solutions webmapping à travers des applications métiers internes.

1.3 Les besoins techniques

La cellule SIG et développement développe des applications métier en interne et, en ce moment, plusieurs applications sont en production. Parmi ces applications on peut citer.

- Le portail « ORION », qui permet de relever les infractions dans le domaine cynégétique,
- Le portail « BD BIODIV » qui a pour but de :
 - permettre aux agents de terrain de saisir les observations d'espèces
 - de conserver une trace à long terme des données produites par les agents de l'ONCFS
 - de maîtriser et d'harmoniser les méthodes de collecte et de valoriser la donnée collectée à l'échelle nationale etc.

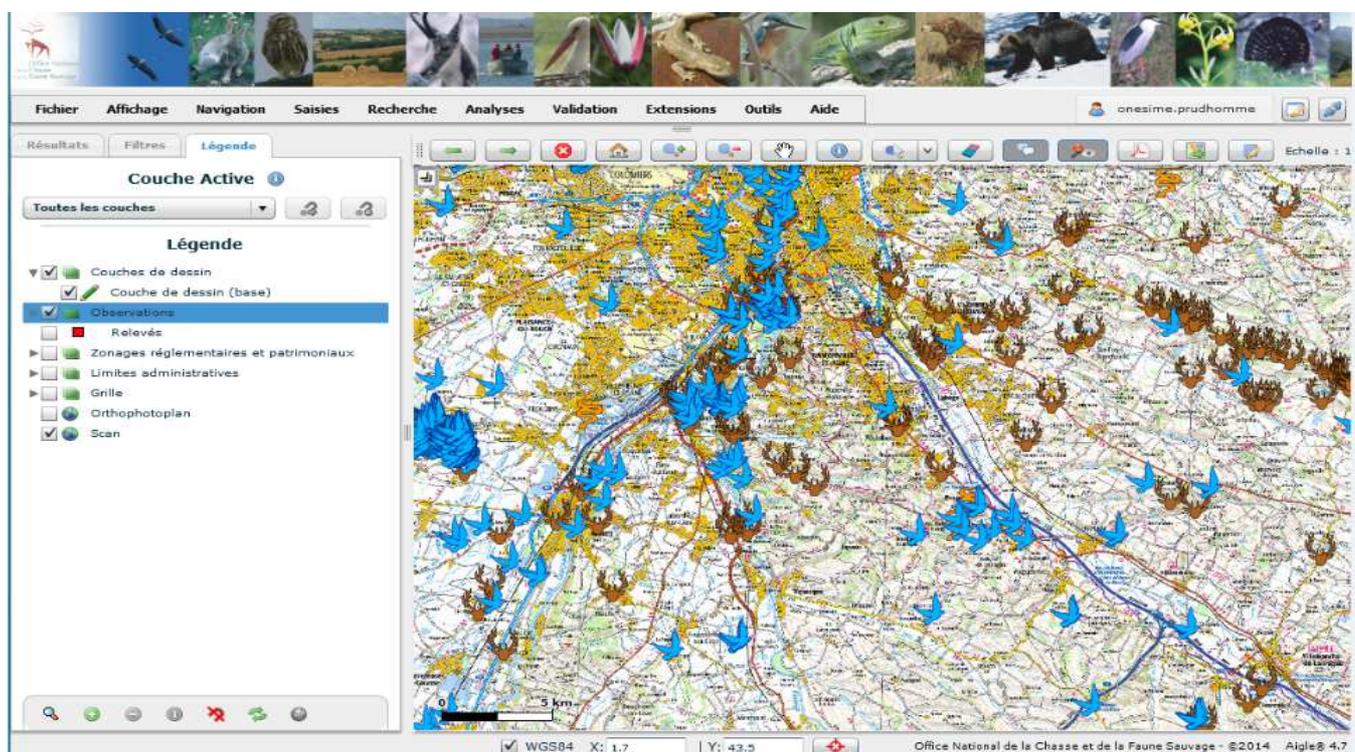


Figure 3: Interface du portail BD BIODIV ; Source : ONCFS

Ce portail est visualisable à cette url <http://bdbiodiv.oncfs.net> mais n'est pas ouvert au public.

Le développement de ces applications passe par Aigle et Géo qui sont des générateurs d'application développés par Business Géografic⁸. Se sont des services cartographiques web interopérables avec tous types de données géographiques et aux formats standardisés.

Aigle est la première version développée par Business Géografic et Géo est une nouvelle avec plus de fonctionnalités et notamment la prise en compte du mode responsive. De ce fait l'ONCFS est entrain de migrer vers Géo dont l'interface se présente comme suit :

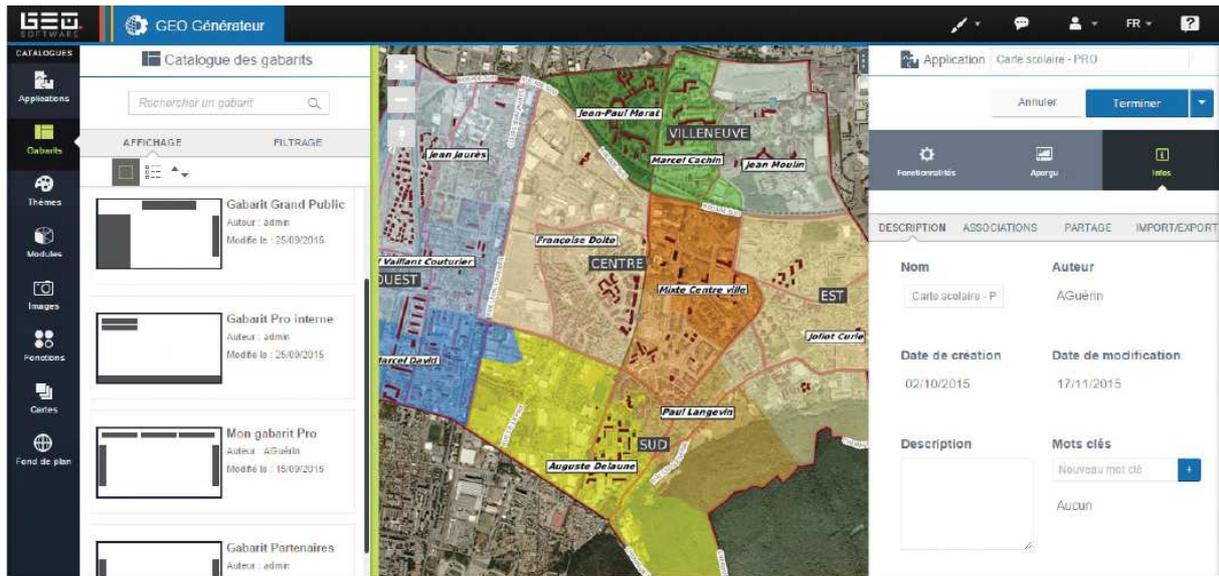


Figure 4: Interface Géo Générateur, Source : ONCFS

Géo générateur permet ainsi à l'ONCFS de développer ses applications métiers à la fois web mais aussi mobiles par l'ajout d'un module dans l'application. Ces applications métiers ont besoin des fonds cartographiques afin de permettre aux agents sur le terrain de saisir directement leurs observations comme illustré dans l'interface de BD BIODIV (figure 3).

1.4 Contexte thématique

Les fonds de carte ont été générés en 2012 par Jérôme Haas un ancien étudiant M2 Sigma, avec Qgis server⁹, mais l'ONCFS a besoin de les actualiser. L'objectif de ce stage consiste à mettre à jour les tuiles en réfléchissant à l'organisation de leurs productions et en l'adaptant aux besoins de l'ONCFS. Il s'agit aussi de choisir la solution la mieux adaptée et la plus performante par rapport à l'environnement (windows) via un état de l'art des différentes

⁸ <https://www.business-geografic.com/fr/>

⁹ https://live.osgeo.org/fr/overview/qgis_mapserver_overview.html

méthodes de tuilage. Enfin une fois le choix fait, d'effectuer des tests de tuilage au niveau départemental puis régional avant d'envisager au niveau national pour le scan (scan25, scan100, et scan1000) et l'orthophotos (5m et 50cm). Et en fonction de l'avancement du stage réfléchir sur le tuilage ou non du cadastre.

Pour faire le tuilage, l'ONCFS utilise les fonds cartographiques de l'IGN. L'Institut national de l'information géographique et forestière (IGN) est l'opérateur de l'État en matière d'information géographique et forestière de référence. Il est incontournable dans le domaine de la cartographie surtout au niveau des cartes topographiques qui représentent avec précision le relief (courbes de niveaux) mais aussi les détails du terrain (route, sentiers arbres, rivières etc.) d'où le choix porté sur ces produits au détriment d'Open Street Map¹⁰ (qui ne présente pas une aussi bonne homogénéité de qualité), de Google Maps¹¹ ou de Bing Maps¹² (qui n'ont pas de carte topo, ce sont principalement des cartes routières ou photo aériennes).

Ainsi, l'IGN fournit ses géoservices gratuitement pour les missions de service public non industrielles et non commerciales. Mais des limites sont rencontrées sur ces offres de l'IGN en consultant le barème public des géoservices IGN (au 1^{er} avril 2017).

Ce barème de transaction concerne la consommation annuelle des ressources par les structures pour leurs outils et sites web. Le tableau ci-dessous (tableau 1) montre le type de ressource avec l'équivalence de transaction. L'IGN se base sur ce tableau pour calculer le nombre de transactions et fixer ainsi le prix.

| Type de Ressources | Équivalence avec une transaction |
|----------------------|----------------------------------|
| WMS | 1 image soit 1 écran |
| WMTS | 16 tuiles soit 1 écran |
| WFS | 1 requête limitée à 1 000 objets |
| Géocodage | 4 géocodages |
| Alticodage | 1 profil altimétrique |
| Calcul d'itinéraires | 2 calculs d'itinéraires |
| Calcul d'isochrones | 1 calcul d'isochrone |

Tableau 1 : Tarification des transactions des Géoservices IGN, Source : IGN, avril 2017)

¹⁰ <https://www.openstreetmap.org>

¹¹ <https://www.google.fr/maps>

¹² <https://www.bing.com/maps>

Pour les services publics l'IGN offre 100 000 transactions par an mais, au-delà de ce nombre, il faut payer la transaction comme illustré dans le tableau ci-dessous.

| Mission de service public non industrielle et non commerciale | |
|--|--------------------|
| Nombre annuel de transactions | Prix (€ HT) |
| 200 000 | 5 000 € |
| 500 000 | 10 000 € |
| 1 000 000 | 17 500 € |
| 2 000 000 | 30 000 € |
| 5 000 000 | 57 500 € |
| 10 000 000 | 80 000 € |
| 20 000 000 | 112 500 € |

Tableau 2: Consommation des ressources, Source : IGN, avril 2017

Ainsi le nombre de transactions annuel détermine le prix à payer comme le montre ce tableau. Mais cette tarification ne concerne que les missions de service public non commercial et non industriel. Ce tarif est beaucoup plus élevé pour les entreprises privées et le grand public. De ce fait, si on tient compte de l'équivalence des transactions et des grilles tarifaires les besoins de transaction pour l'ONCFS lui coûteraient 50 000€ pour environ 4 000 000 de transactions d'après le tableau d'estimation de consommation des tuiles, basé sur les tuiles et flux utilisés actuellement au niveau de la structure (issu du tuilage effectué en 2012) comme le montre le tableau ci-dessous.

| | Nombre de niveaux de zoom | Nombres de tuiles |
|---------------------------|----------------------------------|-------------------------------|
| Scan25 | 11 | 5 842 000 |
| Orthophotos | 13 | 12 709 000 |
| Cadastre | 16 | 43 473 000 |
| Total | | 62 024 000 tuiles |
| Total transactions | | 3 876 500 transactions |

Tableau 3: Tarification des transactions des Géoservices IGN, Source : IGN, avril 2017

Ce tableau donne une estimation du nombre de tuiles qui devront être disponibles sur les serveurs de l'ONCFS pour ensuite être diffusées sur les terminaux mobiles.

De ce fait, dans un contexte de réduction budget de fonctionnement et devant l'incapacité de connaître l'évolution de la consommation de ses ressources, cette option n'est pas envisageable, vu que l'ONCFS a besoin non seulement de ces ressources pour ses propres agents mais aussi pour les agents des structures partenaires.

Par conséquent, c'est à la suite de ces réflexions et analyses que le projet est mis en œuvre par l'ONCFS afin de générer ses propres flux dans ses applications métiers. Ces applications seront proposées à la fois pour un support web et mobile. Vu que les agents travaillent essentiellement sur des zones naturelles, ils sont souvent limités dans l'accès au réseau mobile et ont donc besoin d'avoir des applications déconnectées (fonctionnant sans couverture réseau) et de pouvoir stocker les fonds de carte sur des cartes mémoire SD d'où le besoin de tuiles stockées en cache local (sur la machine).

2. Le webmapping

La définition du webmapping s'avère très vaste mais néanmoins on peut le considérer comme étant la diffusion de données spatiales sur internet. Cette diffusion peut être faite en comprenant un traitement cartographique coté client comme coté serveur.

Chacune de ces solutions ont leurs avantages et défauts mais répondent à des objectifs bien différents. Côté client, l'utilisateur gère les données spatiales avant tous les outils proposés par son SIG. Cependant, il ne peut modifier les données affichées. Côté serveur, l'utilisateur ne dispose que de quelques outils proposés (par le créateur du site)¹³. Cette diffusion des données spatiales nécessite un certain échange de données entre les différents acteurs d'où leur interopérabilité.

2.1. Webmapping et interopérabilité

L'interopérabilité¹⁴ est la capacité que possède un produit ou un système, dont les interfaces sont intégralement connues, à fonctionner avec d'autres produits ou systèmes existants ou futurs et ce sans restriction d'accès ou de mise en œuvre. L'interopérabilité des données géographiques est aujourd'hui une réalité dans les SIG et la thématique d'aménagement du territoire est assez parlante du fait qu'elle peut être traitée qu'en croisant de nombreux critères et de nombreuses sources de données ce qui nécessitent une certaine interopérabilité.

Dans le cadre du webmapping, cet échange ne s'effectue plus entre acteurs mais à travers plusieurs serveurs c'est-à-dire que les données ne sont plus traitées par une seule machine. Mais, pour que ce système soit efficace il faut qu'il soit régi par des normes communes, d'où la mise en place de l'open géospatial consortium (OGC).

¹³ <https://www.unamur.be/sciences/geographie/etudiants/logiciels-sig-libres-et-webmapping>

¹⁴ <https://fr.wikipedia.org/wiki/Interop%C3%A9rabilit>

L'OGC¹⁵ est un consortium international pour développer et promouvoir des standards ouverts, les spécifications Open GIS, afin de garantir l'interopérabilité des contenus, des services et des échanges dans les domaines de la géomatique et de l'information géographique. Il normalise des formats de données libres pour faciliter les échanges entre plateforme.

Ces protocoles sont des web services qui permettent la communication entre applications et systèmes hétérogènes dont le protocole de transport est essentiellement le HTTP.

2.2. Les web services

Les web services sont des protocoles d'interface informatique qui exécutent des tâches et respectent un format spécifique. Ils facilitent l'échange de données en faisant appel à des fonctionnalités à distance.

Ils permettent aussi le dialogue entre application indépendamment de leur plateforme ou de leur langage d'où leurs avantages. Ils constituent donc des composants développés dans des langages différents et peuvent être déployés sur des plateformes différentes.

2.3. Les web services géo spatiaux

La technologie des services Web géo spatiaux est aujourd'hui très utilisée car elle fournit un nouveau moyen d'interopérabilité des données géo spatiales. Ils permettent ainsi de publier des données géographiques sur Internet et d'y accéder à travers des applications web en gérant les interactions avec les utilisateurs.

Ces fournisseurs de services sont normalisés par l'OGC (Open Geospatial Consortium). Parmi les spécifications, les standards de l'OGC on peut citer :

- **WCS** : Web Coverage Service (fournit une interface qui permet de télécharger des données MNT, ortho images etc).
- **GML** : Geography Markup Language (permet d'encoder, manipuler et échanger des données géographiques).
- **KML** : Keyhole Markup Language (gère l'affichage de données géo spatiales dans les logiciels SIG).

¹⁵https://fr.wikipedia.org/wiki/Open_Geospatial_Consortium

- **WMS** : Web Map Service (permet de mettre en place un réseau de serveurs cartographiques qui permettra aux clients de construire des cartes interactives).
- **WMTS** : Web Map Tile Service (permet de mettre à disposition d'utilisateurs distants des images géo référencées).

Dans ce stage nous nous intéressons notamment aux WMS et WMTS. Le Web Map Service est un protocole de communication standard qui permet d'obtenir des cartes de données géo référencées à partir de différents serveurs de données. Il existe différentes requêtes qui sont disponibles dans un service wms. Parmi ces requêtes on peut citer :

- **GetCapabilities** : permet d'obtenir les métadonnées du service.
- **GetMap** : permet d'obtenir une carte avec les paramètres demandés et des informations bien définies,
- **GetLegendGraphic** : cette requête est facultative pour un WMS compatible SLD. Il fournit un mécanisme général pour l'acquisition de symboles de légende, au-delà de la référence LegendURL des fonctionnalités WMS¹⁶.
- **GetFeatureInfo** : permet d'obtenir des informations attributaires sur un objet de la carte.

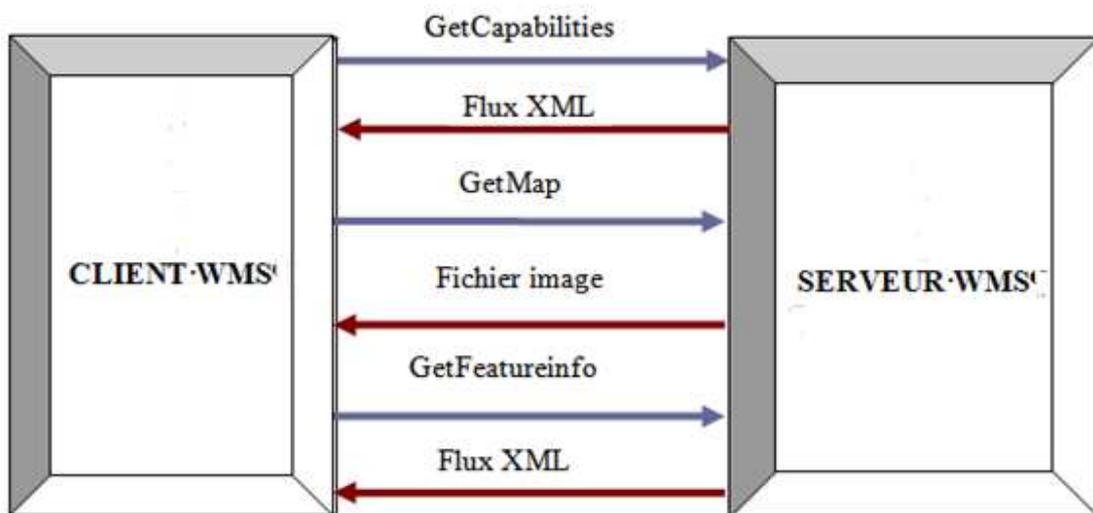


Figure 5:°Principaux opération du WMS, Source : géorezo

¹⁶ http://docs.geoserver.org/stable/en/user/services/wms/get_legend_graphic/

Ce service produit ainsi des cartes sous forme d'image, dans des formats comme jpeg, png, gif etc. La transparence gérée par certains de ses formats permet d'obtenir la superposition de différentes couches.

Quant au WMTS, il s'agit d'un des services d'images tuilées et un complément à WMS, son principal avantage réside dans le fait qu'il offre de meilleures performances dans la diffusion des données cartographiques. Comme le WMS le WMTS supporte aussi différentes requêtes et on peut citer:

- **GetCapabilities** : permet de renseigner sur le contenu du service wmts.
- **GetTile** : cette requête permet à un client wmts de demander une tuile pour une certaine pyramide d'une certaine couche et selon un certain format. Ainsi le service wmts renvoie la tuile demandée.
- **GetFeatureInfo** : c'est une requête optionnelle qui permet à un client wmts de demander une information attributaire sur un objet présent à une certaine position dans une tuile. Il renvoie les informations demandées.

Voici un exemple de requête WMTS à un serveur local et son résultat :

```
http://localhost/cgi-bin/mapserv.exe?map=C:/ms4w/apps/cgi-viewer/htdocs/mapfile.map&SERVICE=WMS&VERSION=1.3.0&REQUEST=GetCapabilities&REQUEST=GetMap&LAYERS=scan25&CRS=EPSG:2154&bbox=99038.39914407907,6046555.798283662,1242435.5404792284,7110523.76340624&WIDTH=5000&HEIGHT=5000&FORMAT=image/png
```



Figure 6: Résultat requête WMTS

Cette requête comporte le type de données (layer), la projection, l'étendue le format etc.

Le Web Map Tile Service ne permet ainsi de requêter des images pré-calculées appelées aussi tuiles.

2.4. Notion de tuilage

Le tuilage est le découpage du raster général en petite images à des échelles prédéfinies pour être utilisé par des outils web¹⁷. Cette technologie de tuilage est devenue aujourd'hui inséparable du webmapping.

Le tuilage a vu le jour grâce à un défaut ergonomique avec l'arrivée de Google Maps qui a rendu le déplacement sur la carte beaucoup plus simple et plus rapide. Tous ces déplacements fluides sur une carte ont pu être réalisés par une technologie de cache. La tuile est donc l'unité de base de stockage des données dans les caches. La figure ci-dessous représente le mode de fonctionnement d'un serveur cartes sans cache et un serveur de tuile avec cache.

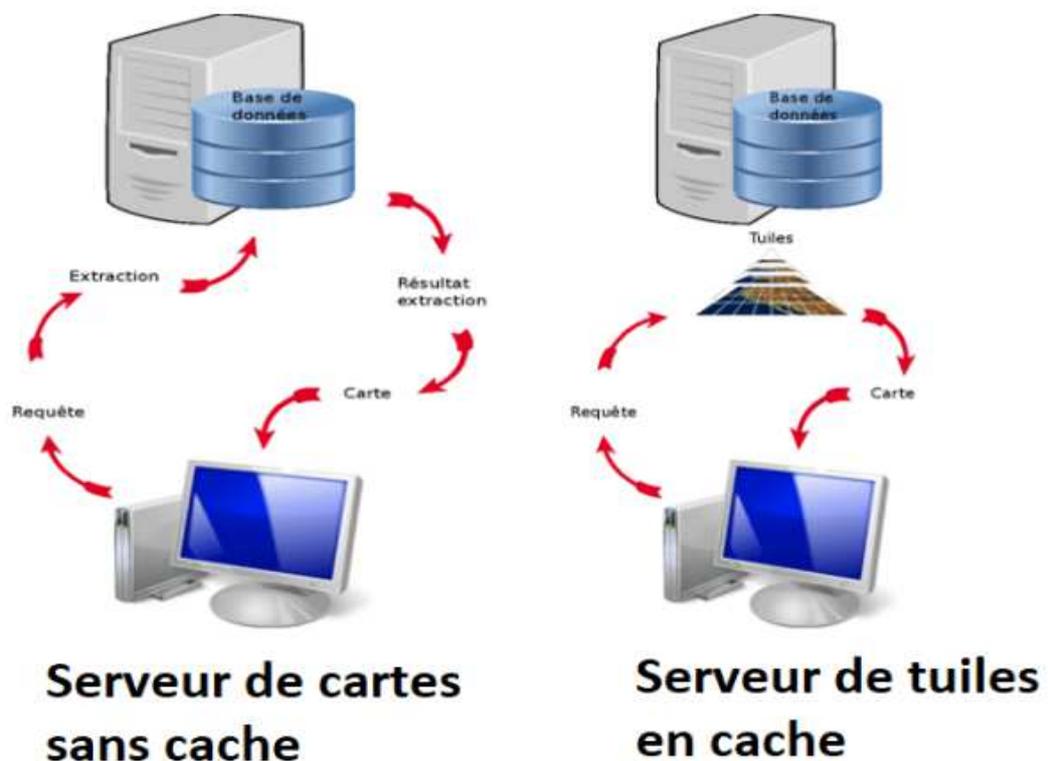


Figure 7: Fonctionnement d'un serveur carte avec ou sans cacheSource : <https://www.sigterritoires.fr/index.php/geoserver-avance-le-tuilage-principes/>

¹⁷ <http://blog.martzluff.net/realiser-un-carte-dimages-aeriennes-comparees-avec-qgis-gdal-et-leaflet/>

Le serveur de tuile demande ainsi un simple calcul pour déterminer les images qui devront être retournées au client avec un gain de temps important par rapport au serveur de carte.

2.5. Le stockage des tuiles en cache

Cette technologie consiste à pré-calculer des images d'une carte à différents niveaux de zoom et à les stocker sur un disque. Son intérêt est de d'accélérer l'affichage en pré-général les tuiles, mais aussi en permettant de les stocker sur l'appareil, la machine locale, pour ne pas avoir à les récupérer de nouveau. L'avantage est double, gain de temps à l'affichage et disponibilité des cartes (photos, etc.) même sans connexion au réseau.

En outre, cette technologie repose sur un maillage rectangulaire régulier de la zone couverte, une image carrée étant associée à chaque maille (appelée aussi tuile).

Pour pouvoir représenter cette zone à différents niveaux d'échelle (ou niveaux de zoom), des maillages de plus en plus fins sont définis pour la zone. Le cache d'images ainsi tuilées apparaît donc comme une pyramide dont chaque niveau de zoom représente un étage¹⁸.

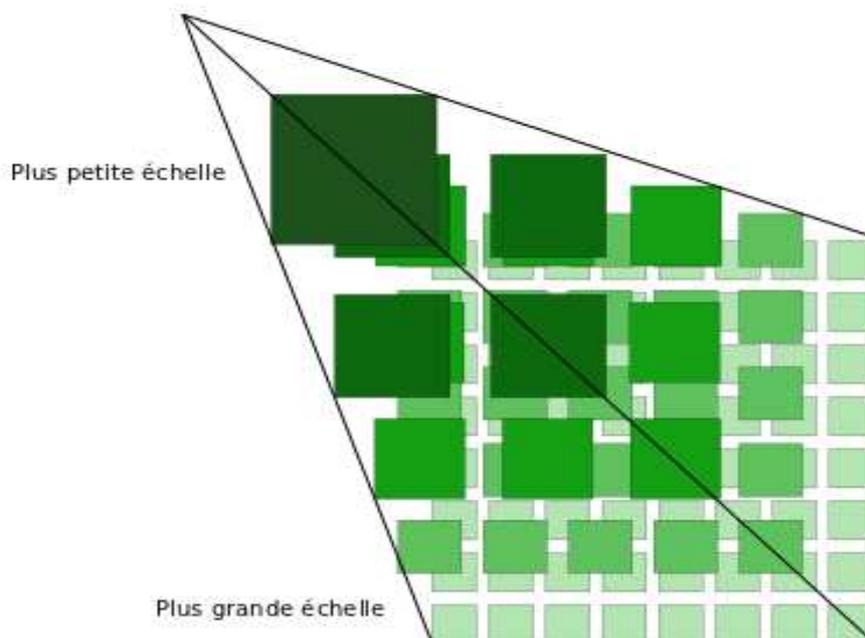


Figure 8: Pyramide de tuile, Source : IGN

¹⁸ <https://geoservices.ign.fr/documentation/geoservices/wmts.html>

Enfin, différentes solutions permettent de mettre en place un service d'image tuilé d'où la nécessité de faire un état de l'art.

2.6. État de l'art

L'établissement d'un état de l'art a été déterminant sur le choix de la technologie à utiliser afin de pouvoir optimiser et générer les tuiles. Mon maître de stage avait déjà commencé cette étape avant le démarrage de mon stage. De ce fait il avait lancé ces recherches à travers les forums notamment GeoRezo afin d'avoir des conseils sur la solution la mieux adaptée aux besoins de l'ONCFS et à l'infrastructure locales. J'ai continué ce travail au démarrage de mon stage en essayant de parcourir les différentes solutions de mise en cache des tuiles et leurs spécificités. Ce travail n'a pas été réalisé à travers des tests de rendu mais en élaborant un comparatif à travers leurs fonctionnalités et spécificités mais aussi des retours d'expérience d'utilisateurs. Parmi ces paramètres, il y a : le langage, l'environnement (les serveurs de l'ONCFS sont sur Windows), le fonctionnel, les performances et le coût notamment. Parmi les solutions on retrouve :

- Mapnik ¹⁹: qui est un moteur de rendu cartographique qui permet de générer des cartes au format raster ou des tuiles vectorielles. Il est écrit en C++ avec quelques scripts en Python. Actuellement, il est plus facile de compiler mapnik depuis les sources sous Linux parce que toutes les dépendances peuvent être facilement récupérées. Cependant, les fichiers binaires sont aussi disponibles pour Windows depuis la version 0.5 mais le processus sera plus compliqué, notamment parce qu'il y aura beaucoup plus d'éléments à installer et que certaines versions en ligne de commande des outils employés ne sont pas disponibles sous forme de binaires. Aussi les produits écrits en python sont moins rapides. Python n'a pas été créé pour être rapide mais simple à utiliser²⁰. Python est un langage extrêmement pratique car il est facile à lire et à écrire, d'un autre côté, à l'exécution il est beaucoup plus lent²¹ En plus Mapnik lit les données OpenStreetMap, via un fichier de configuration en XML mais ne correspond pas à l'environnement (données IGN).

¹⁹ <https://wiki.openstreetmap.org>

²⁰ Yves Jacolin, CampToCamp

²¹ <https://bioinfo-fr.net/cython-votre-programme-python-mais-100x-plus-vite>

- Qgis Server²² : c'est une application FastCGI/CGI (Common Gateway Interface) écrite en C++ qui fonctionne avec un serveur web (tel que Apache, Lighttpd). Il dispose d'une extension Python qui autorise un développement rapide et efficace et un déploiement de nouvelles fonctionnalités. Ainsi, l'ONCFS a besoin de générer des tuiles à la demande. QGIS server est un serveur WMS mais ne propose pas de gestion des tuiles en plus niveau performance il est plutôt lent²³.
- Géowebcache²⁴ : c'est une application Web Java utilisée pour mettre en cache des mosaïques de cartes provenant de diverses sources, telles que WMS (Web Map Service) OGC. Il implémente diverses interfaces de service (telles que WMS-C, WMTS, TMS, Google Maps KML, Virtual Earth) afin d'accélérer et d'optimiser la diffusion des images de cartes. Il peut également recombinaison des tuiles pour travailler avec des clients WMS réguliers. Geowebcache est livré avec l'installation de Géoserver, il est installé et activé par défaut. L'ONCFS a besoin d'une solution performante par rapport au nombre de tuiles à générer.
- Mapproxy²⁵ : c'est un proxy de tuiles géospatiales open source qui prend en charge la reprojection. Initialement développé par Omniscale, il peut lire des données à partir de WMS, de tuiles, de mapserver et de mapnik, et mettre en cache ces données en tant que WMS, WMTS, TMS et KML. Pour Mapproxy, les options de déploiement sur Windows sont limitées, de plus les performances sont meilleures sous Linux ou Mac OS X que sous Windows²⁶.
- MapTiler et ArcGis Online se sont des solutions propriétaires. Le premier est une application bureautique pour la création de tuiles de carte, la publication de carte raster rapides. La géodonnée est transformée en tuiles compatibles avec Google Maps et Earth prêtes pour la publication directe vers n'importe quel serveur web ou stockage dans le "cloud" (comme Amazon S3)²⁷. Le second est une solution d'analyse et de cartographie basée sur le cloud. Il est utilisé pour créer des cartes, pour analyser et

²² <https://docs.qgis.org>

²³ Yves Jacolin, CampToCamp

²⁴ <http://geowebcache.org/>

²⁵ <https://mapproxy.org>

²⁶ <https://lists.osgeo.org/pipermail/mapproxy/2015-December/002276.html>

²⁷ <https://live.osgeo.org>

partager des données²⁸. Il permet aussi de créer un service de carte mis en cache qui pourra être utilisé par des applications client. Ces solutions nécessitent une licence et un coût qui n'est pas prévu dans le budget de l'ONCFS.

- Mapcache²⁹ : c'est un serveur qui implémente la mise en cache de tuiles pour accélérer l'accès aux couches WMS. C'est un module pour le serveur Apache codé en C. Il fait partie de la communauté Mapserver qui était l'un des premiers serveurs cartographiques de services WMS ou WFS mais, petit à petit, il s'est vu augmenté d'autres outils, comme mapcache. C'est un projet particulièrement mature et très performant. Au niveau fonctionnel, il répond aux besoins de l'ONCFS :
 - o pré-générer des tuiles
 - o générer des tuiles à la demande (pour ne pas à avoir à tout générer, cela fait trop de tuile donc trop de temps et d'espace disque)
 - o générer des données mbtiles pour mobile déconnecté

Cette liste est loin d'être exhaustive mais ces quelques solutions nous ont permis de faire le choix sur la solution qui sera adaptée au mieux aux besoins de l'ONCFS dans un environnement windows à savoir Mapcache.

Une formation de 2 jours avec la société Camptocamp a permis de consolider ce choix. CamptoCamp³⁰ est un fournisseur de solutions personnalisées, une entreprise innovante dans l'intégration de logiciels servant à la mise en valeur de données géospatiales (SIG) à la gestion complète d'entreprise (ERP) et à la gestion d'infrastructures informatiques utilisant les meilleures technologies Open Source.

Après le choix de mapcache, place ensuite à l'installation et au paramétrage afin de pouvoir générer les tuiles.

²⁸ <https://doc.arcgis.com/fr/arcgis-online/reference/what-is-ago.htm>

²⁹ <https://mapserver.org/fr/mapcache>

³⁰ <https://www.camptocamp.com/>

2.7. Installation et paramétrage de MapServer et MapCache

MapServer est un logiciel open source permettant le rendu de données géographiques. Ainsi il fonctionne sur plusieurs plateformes à savoir windows, linux, mac etc. Il lit plusieurs formats de données (raster, vecteur) et prend en charge de nombreuses normes OGC à savoir le WMS, WFS etc. MapCache prend en charge la mise en cache des tuiles produites par MapServer.

Dans ce projet l'installation de MapServer a été faite sous windows à travers le groupe d'utilitaires (ou « package ») nommé ms4w³¹, fourni par les auteurs de MapServer. C'est un paquetage qui regroupe les binaires précompilés de l'ensemble des composants nécessaire pour installer un service web basé sur le serveur Apache, le langage PHP ainsi que MapServer. Les serveurs de l'ONCFS sont sur windows ce qui explique le choix porté sur ms4w. Ce package comporte :

- **MapServer 7.0.7**
- **GDAL 2.2.3** (bibliothèque de traducteurs pour les formats de données géospaciales raster et vectorielles)
- **Apache 2.4.29** (serveur HTTP permettant à des clients d'accéder à des pages web)
- **PHP 5.6.34** (ou Hypertext Preprocessor), un langage courant de programmation, libre, principalement utilisé pour produire des pages Web dynamiques via un serveur HTTP)
- **MapCache 1.6.1**

MapServer³² est utilisé sur un serveur Internet pour générer des images dans des pages web, et ainsi permettre l'affichage mais aussi l'interrogation, la modification, d'images cartographiques sur un site Internet. On peut aussi utiliser MapServer en local, sur son ordinateur, pour générer des cartes, effectuer des requêtes et des analyses. MapCache implémente la mise en cache de tuiles pour accélérer l'accès aux couches WMS. Le fonctionnement de mapcache est constitué de la ligne de commande (CLI) qui permet de générer les tuiles puis le mode web.

Pour la configuration du logiciel, il faut connaître les différents blocs nécessaires à la configuration d'un projet, à savoir dimension, tileset, types de cache, source etc. Sa configuration repose sur un fichier XML comprenant plusieurs blocs et structurés comme suit :

La configuration commence par le bloc mapcache qui englobe tous les autres blocs.

³¹ <https://ms4w.com/>

³² <http://www.portailsig.org/faq/mapserver/presentation-0>

- Le bloc grille : une grille précise la zone, la référence spatiale, l'étendue géographique, la résolution et la taille des tuiles.
- Le bloc source : une source est un service qui peut être interrogé pour récupérer des données images. En général c'est un serveur wms accessible par un URL.
- Le bloc cache : définit l'endroit où seront stockées les tuiles.
- Le bloc format : définit dans quel format mapcache va générer les tuiles.
- Le bloc tileset : c'est l'élément de configuration essentiel, il constitue un jeu de tuiles que mapcache va pouvoir proposer aux utilisateurs dans un format donné.
- Le bloc service : les services sont le type de requête auquel mapcache répondra.

```
<mapcache>
<grid>...</grid>
<source>...</source>
<cache>...</cache>
<format>...</format>
<tileset>...</tileset>
<services>...</services>
</mapcache>
```

2.8. Configuration Vhost Apache

Apache est un serveur http qui permet de recevoir et de renvoyer des données. Le client envoie des requêtes à Apache et en retour ce dernier lui fournit des pages à afficher que l'internaute pourra consulter dans un navigateur. Ainsi le terme hôte virtuel (Vhost) fait référence à la pratique consistant à exécuter plusieurs sites Web. L'hébergement³³ intégré des hôtes virtuels du Serveur HTTP Apache permet au serveur de fournir différentes informations en fonction de l'adresse IP, du nom d'hôte ou du port faisant l'objet de la requête sur une seule machine. La configuration du vhost est effectuée grâce à des directives avec des fichiers sous format texte. Le fichier principal est httpd.conf.

Le fichier mapcache.xml sera localisé dans ms4w/apache/httpd. Ensuite il faut activer le fichier apache/conf/httpd.conf pour enfin démarrer apache et pour envoyer des requêtes et recevoir des images.

```
<VirtualHost *:80> # Apache doit écouter sur le port 80

    <Directory C:\ms4w\apps\mapcache\>
        Require all granted
    </Directory>
    MapCacheAlias /mapcache
    "C:\ms4w\apps\mapcache\mapcache.xml" #chemein vers le fichier
    xml
</VirtualHost>
```

³³ <http://web.mit.edu/rhel-doc/4/RH-DOCS/rhel-rg-fr-4/s1-apache-virtualhosts.html>

2.9. Principales fonctionnalités de mapcache

Mapcache offre plusieurs fonctionnalités, parmi lesquelles on peut citer :

La commande pour «*Seeder*» c'est-à-dire générer des tuiles en amont. L'utilitaire d'enclenchement s'appelle *mapcache_seed* avec des options en ligne de commande simples telles que :

- **-t** : la couche qu'on veut générer
- **-z** : permet de gérer les différents niveaux de zoom
- **-e** : c'est l'étendue à Seeder c'est-à-dire une boîte englobante de la zone à tuiler.
- **n** : nombre de thread (lancer 4 processus en parallèle pour dispatcher les requêtes. Mais cette option n'est pas compatible avec l'option **-p** donc on ne peut pas les utiliser en même temps soit l'un soit l'autre.
- **-p** : nombre de processus parallèles à utiliser pour demander des tuiles de la source WMS
- **-C** : précise le cache qu'on veut utiliser.
- **-c** : détermine le chemin vers le dossier de configuration
- **-D** : définit quelle dimension utiliser si le tileset gère les dimensions. Peut être utilisé plusieurs fois pour définir plusieurs dimensions.
- **-g** : la grille à utiliser
- **-M** : la taille de la metatile à utiliser. Une méta-tuile³⁴ est un concept utilisé par divers systèmes de rendu, de mise en cache et de stockage. Les carreaux se dirigent finalement vers le navigateur Web sous la forme de très petites images (généralement 256x256 pixels) couvrant une très petite zone de la carte. Au lieu de traiter ces petites images en interne, nous traitons une plus grande zone de la carte, une "méta-tuile".

Au-delà de sa capacité de seeder, mapcache gère aussi des caches. On peut noter ainsi :

- Le cache en local c'est-à-dire sous la forme de fichiers plats sur le disque, qui est le plus facile à configurer et offre ainsi un accès plus rapide aux tuiles existantes.
- Le cache mbtiles est un raccourci vers le cache dans une table SQLite avec des requêtes SQL pré-remplies correspondant à la spécification mbtiles. Son avantage est d'être très pratique sur mobile du fait qu'il permet de stocker les tuiles hors ligne.

³⁴ https://wiki.openstreetmap.org/wiki/Meta_tiles

Cette liste est loin d'être exhaustive il existe d'autres types de caches qui ne sont pas énumérés ici.

Mapcache gère aussi des **dimensions** c'est-à-dire pour un flux wms, on peut avoir plusieurs dimensions. Il gère également des requêtes **Getfeatureinfo** qui est liée aux services wms etc.

3. Mise en œuvre des tuiles

Cette phase correspond à la mise en œuvre des tuiles avec les produits de l'IGN à savoir les scans (25,100 et 1000) et les ortho photos (5m et 50cm) de la France métropolitaine.

Ces scans³⁵ sont des séries de cartes cartographiques sous la forme d'images numériques géoréférencées, immédiatement utilisables pour localiser et positionner des informations, repérer des interventions, saisir et mettre à jour des données métier, ainsi qu'habiller des documents pour des échelles de visualisation autour du 1:1000 000, 1:100 000 et 1:25 000. Cette base de données image répond à des besoins clairement identifiés en participant à une plus large diffusion de l'information géographique et en permettant à des néophytes d'appréhender rapidement et en toute simplicité les enjeux cartographiques auxquels ils sont confrontés.

Quant à l'orthophotographie³⁶ c'est une image obtenue par traitement d'un cliché aérien numérique ou argentique dont la géométrie a été redressée de sorte que chaque point soit superposable à une carte plane. Plus fréquemment mis à jour, elle constitue l'outil numérique de référence des collectivités et des ministères, pour mettre en valeur le territoire, enrichir la visualisation des données et les projets. Elle³⁷ est devenue le support géographique de nombreuses applications dans les domaines de l'aménagement et de l'urbanisme, de l'environnement ou de l'agriculture, grâce à sa richesse en informations et à sa qualité géométrique.

3.1 Acquisition des données

Depuis le 1^{er} janvier 2016, l'ensemble des Données IGN est utilisable gratuitement pour l'exercice de missions de service public par l'État et ses établissements publics administratifs (EPA). Sur ceux l'ONCFS étant une structure publique a bénéficié de cette gratuité qui lui

³⁵ <http://professionnels.ign.fr/donnees>

³⁶ <http://professionnels.ign.fr/orthoimages>

³⁷ <http://professionnels.ign.fr/bdortho-50cm>

permet de récupérer les sources brutes à travers un compte IGN professionnel. L'ONCFS possédait déjà ces données à mon arrivée mais il a fallu mettre certains à jour en téléchargement depuis le compte de la structure.

Après acquisition des données il a fallu mettre en place un serveur de tuiles. L'ONCFS avait déjà un serveur de tuile qui a été mise en place en 2012. De ce fait, c'est pour actualiser les fonds de carte que l'ONCFS a besoin d'un nouveau serveur de tuile qui fonctionnera en parallèle avec l'ancienne. En plus de ce serveur de tuiles, la structure possède d'autres serveurs mais dédiés à d'autres activités : de formation, de développement, de production, etc. Mais les serveurs qui étaient en place sont en train d'être renouvelés afin de mettre en place une nouvelle architecture beaucoup plus simplifiée.

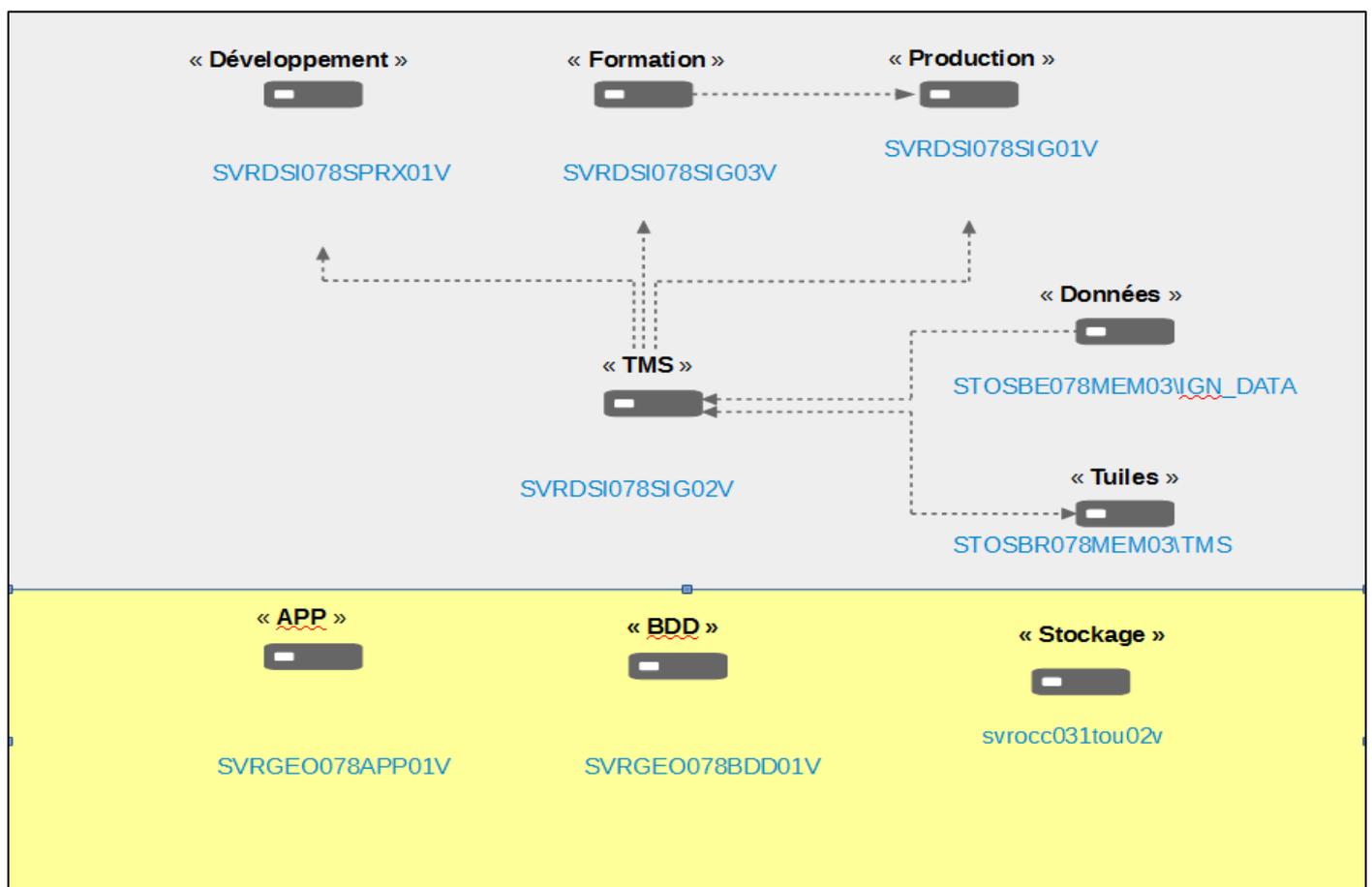


Figure 9: Schéma des serveurs de l'ONCFS, Source : ONCFS

Ainsi, le schéma en jaune est plus simplifié avec un serveur de base de données (BDD) pour tout ce qui concerne les bases de données de l'office, un serveur de stockage pour toutes les données brutes et un serveur d'application (APP).

3.3 Serveur de tuile



Toulouse



Saint benoît (Paris)

Figure 10: Schéma du serveur entre Toulouse et Saint-Benoît

Le serveur de tuile de l'ONCFS est basé en région Parisienne, plus précisément à Saint Benoit dans le département des Yvelines (78), au sein de la direction générale. Il est équipé du système d'exploitation Windows. Mais, au début du stage, le serveur n'était pas encore mise en place. Un serveur test installé à Toulouse a permis de commencer les traitements avant de continuer avec le serveur de Saint-Benoît.

Il a donc fallu comprendre le fonctionnement du serveur de tuile, découvrir comment y installer mapcache et mapserver ainsi que comment le faire fonctionner avec les spécificités d'Apache. Cela a permis par la suite de mettre en place le schéma de fonctionnement des différents paramètres et flux. Le schéma ci-dessous résume tout cela.

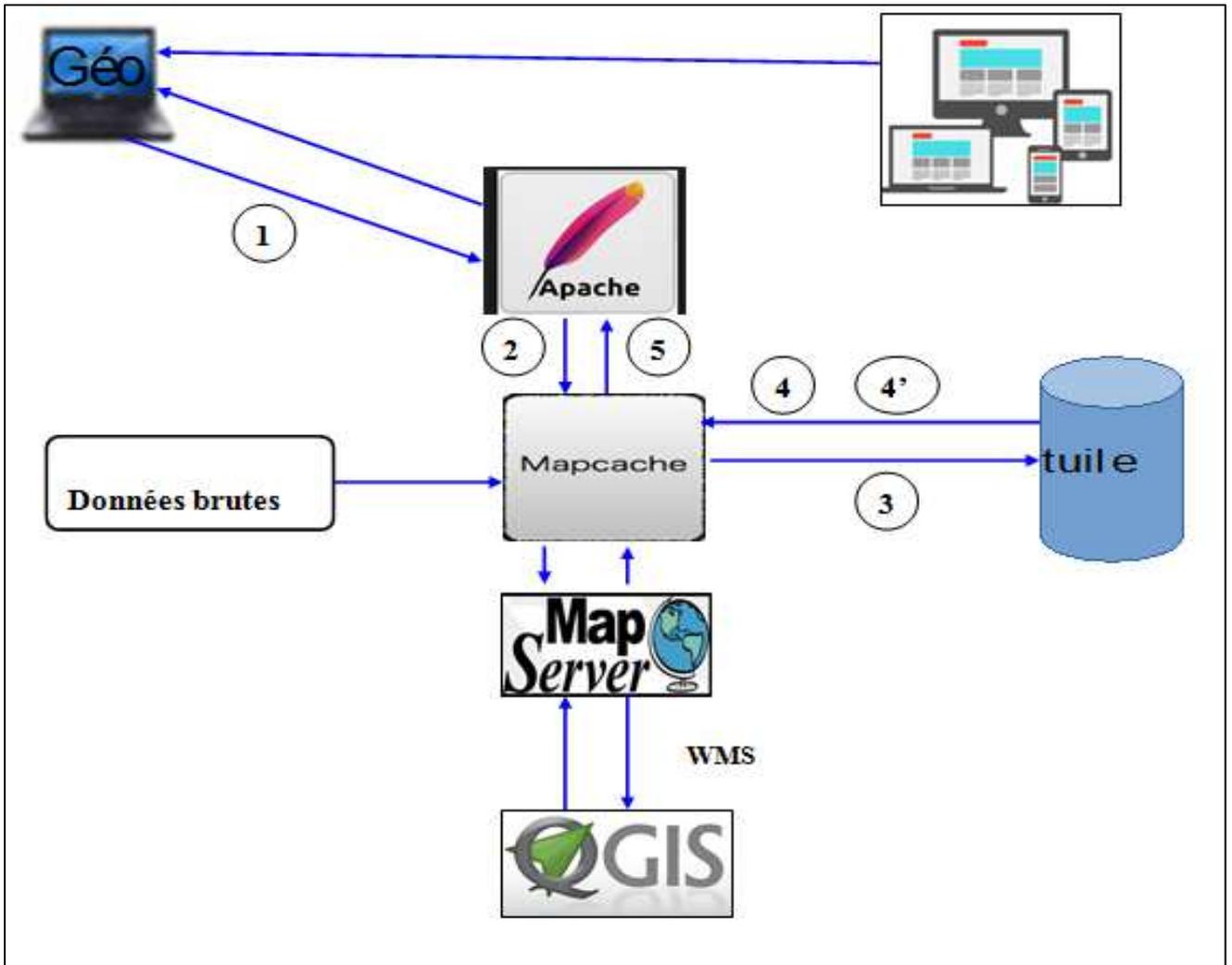


Figure 11: Fonctionnement du serveur de tuilage, Réalisation personnelle

Ce schéma présente le fonctionnement du serveur de tuilage si tuile présente. Ainsi on peut distinguer :

1. L'utilisateur accède à une application web de l'ONCFS, construite avec GEO.
2. GEO demande une tuile à Apache
3. Apache demande une tuile à Mapcache
4. Mapcache vérifie s'il existe déjà des tuiles ou pas
4. Si pas de tuile, demande à MapServer de générer la tuile
5. Remonte la tuile à Apache puis à GEO

Les flux WMTS peuvent être aussi visualisable sur Qgis.

Après avoir mis en place le serveur et défini le mode de fonctionnement qu'il doit avoir, place ensuite à la génération des tuiles elles-mêmes, en *seeding* comme à la demande.

Cette génération nécessite un certain nombre de paramètres avec la création du mapfile et de mapcache.xml.

3. 4. La création des fichiers mapfile et mapcache.xml

Le fichier mapfile (ou map) est le fichier de configuration de base utilisé par MapServer pour afficher des données géospatiales sous forme d'images, de vecteurs ou les diffuser en tant que. Son but est de définir les couches qu'il peut dessiner. C'est un fichier texte composé de différents objets et chaque objet a une variété de paramètres. Ce fichier présente des normes qu'il faut respecter afin de pouvoir générer correctement les informations d'une couche donnée. Il est basé sur des blocs, chaque bloc commence par son identifiant et se termine par le mot-clé « END ».

Chaque bloc est composé de différents paramètres en fonction du besoin d'affichage de la couche. Ainsi, un mapfile commence toujours par l'objet map et contient une liste d'objets *layer* (couches) que mapfile lit et dessine.

Ainsi pour générer les tuiles le mapfile a été configuré comme suit :

```
MAP
  NAME "oncfs WMS"
  EXTENT 99038.39914407907 6046555.798283662 1242435.5404792284
7110523.76340624
  UNITS METERS
  RESOLUTION 96
  DEFRESOLUTION 96
  MAXSIZE 5000
  SHAPEPATH 'D:\tuilage
  IMAGECOLOR 255 255 255
  STATUS ON
  OUTPUTFORMAT
    NAME jpeg
    DRIVER "AGG/JPEG"
    MIMETYPE "image/jpeg"
    IMAGEMODE RGB
    EXTENSION "jpeg"
    FORMATOPTION "QUALITY=75, PROGRESSIVE=TRUE"
  END
  OUTPUTFORMAT
    NAME png
    DRIVER AGG/PNG
    MIMETYPE "image/png"
    IMAGEMODE RGBA
    EXTENSION "png"
    FORMATOPTION "INTERLACE=OFF"
```

```

    FORMATOPTION "QUANTIZE_DITHER=OFF"
    FORMATOPTION "QUANTIZE_FORCE=ON"
    FORMATOPTION "QUANTIZE_COLORS=256"
END
PROJECTION
    "Init=epsg: 2154"
END
WEB
    METADATA
        "wms_title" "ONCFS WMS"
        "wms_abstract" "XXXXXXXXXXXXXXXXXXXXXXXXXXXX"
        #"wms_onlineresource" "http://XXXXXXXXXXXXXXXXXXXXXXXXXXXX"
        "wms_srs" "EPSG: 2154"
        "wms_encoding" "UTF-8"
        "wms_enable_request" "*"
        "wfs_enable_request" "!*"
    END
END
LEGEND
    LABEL
        ENCODING "UTF-8"
        TYPE TRUETYPE
        FONT "Arial"
        SIZE 9
    END
END
LAYER
    NAME "SC1000_JP2"
    GROUP "scans"
    TYPE RASTER
    TILEINDEX "SC1000_JP2/index_scan1000.shp" # Relative au
chemin définie dans SHAPEPATH
    STATUS ON
    MINSCALEDENOM 100000
END

END
LAYER
    NAME "SC100"
    GROUP "scans"
    TYPE RASTER
    TILEINDEX "SC100 /index_SC100.shp" # Relative au chemin
définie dans SHAPEPATH
    STATUS ON
    MAXSCALEDENOM 7559040
END

END
LAYER
    NAME "SC25"
    GROUP "scans"
    TYPE RASTER
    TILEINDEX "SC25 /index_SC25.shp" # Relative au chemin
définie dans SHAPEPATH
    STATUS ON

```

```
    MINSCALEDENOM 7000
  END
```

END

Après avoir configuré le mapfile j'ai ensuite configuré mapcache qui repose sur un fichier XML avec différents blocs comme illustré dans la partie installation et paramétrage de MapServer/MapCache (2.7). Ces différents blocs sont représentés comme suit dans ce travail.

```
<mapcache>
  <grid name="francel93">
    <metadata>
      <title>grille française en lambert 93</title>

    </metadata>
    <extent>99038.39914407907 6046555.798283662
1242435.5404792284 7110523.76340624</extent>
    <srs>EPSG:2154</srs>
    <units>m</units>
    <size>256 256</size>
    <resolutions>4891.9698091 2445.9849042 1222.9924523
611.4962264 305.7481135 152.8740566 76.43702827 38.2185148
19.1092579 9.554628510 4.777314211 2.388657</resolutions>
  </grid>

  <source name="wms_oncfs" type="wms">
    <http>
      <url>http://localhost/cgi-bin/mapserv.exe</url>
    </http>
    <getmap>
      <params>
        <map>C:\mapfile.map</map>
        <layers>scans</layers>
      </params>
    </getmap>
  </source>
  <locker type="disk">
    <directory>C:\ms4w\tmp</directory>
  </locker>
  <cache name="disk_cache_oncfs" type="disk">
    <base>E:\tiles</base>
  <!-- <symlink_blank/> -->
  </cache>

  <format name="myjpeg" type="JPEG">
    <quality>85</quality>
    <photometric>ycbcr</photometric>
  </format>
  <format name="mypng" type="PNG">
    <compression>fast</compression>
    <colors>256</colors>
  </format>
  <format name="mymixed" type="MIXED">
    <opaque>myjpeg</opaque>
    <transparent>mypng</transparent>
  </format>
```

```

<tileset name="scans">
  <source>wms_oncfs</source>
  <cache>disk_cache_oncfs</cache>
  <grid>francel93</grid>
  <grid>g</grid>
  <metatile>8 8</metatile>
  <metadata>
    <title>scans test</title>
    <abstract>couche scans ign_oncfs</abstract>
  </metadata>
  <format>PNG</format>
</tileset>
<service type="wmts" enabled="true" />
<service type="tms" enabled="true" />
<service type="wms" enabled="true" />
<service type="demo" enabled="true" />
<auto_reload>true</auto_reload>
</mapcache>

```

Après ces configurations j'ai lancé la génération des tuiles pour les scans et les orthophotos.

3. 5. La génération des tuiles pour les scans

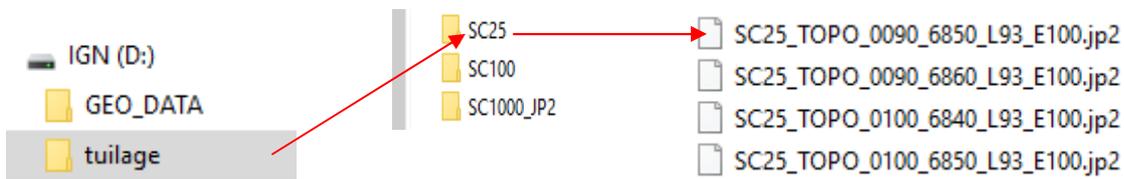
La génération des tuiles s'avère rapidement complexe lorsqu'on traite de gros volumes de données. Nous avons rencontré beaucoup de difficultés pour générer les tuiles surtout pour le scan25, plus important en volume. Par contre, la génération a été rapide pour le scan1000 et le scan100.

L'un des objectifs de ce stage était, une fois la solution choisie, d'effectuer un test au niveau départemental et régional avant de tuiler pour la France métropolitaine. Le test s'est réalisé avec le scan25 au niveau régional pour l'Occitanie. La génération des tuiles nécessite avant tout de générer l'index de tuile. Ce programme construit un shapefile avec un enregistrement pour chaque fichier raster en entrée, un attribut contenant le nom du fichier, et un objet polygone d'extension du raster. Ce fichier est utilisable dans MapServer comme une mosaïque de raster (tileindex)³⁸.

Cet index se produit grâce à l'utilitaire `gdaltindex` distribué avec GDAL, il fait partie du package MS4W présenté dans la partie 2.6. Il permet ainsi une gestion plus rapide d'une grande quantité de données raster. Le script pour générer l'index dans la ligne de commande a besoin du chemin du répertoire où sont stockés les rasters scan25 de la région Occitanie

³⁸ <http://mapserver.org>

(d:\tuilage\SC25) sous format jpeg, de l'utilitaire gdaltindex et de l'emplacement absolue du fichier.



```
d:\tuilage\SC25>gdaltindex -write_absolute_path index_SC25.shp SC25 *.jp2
```

```
Creating new index file...
```

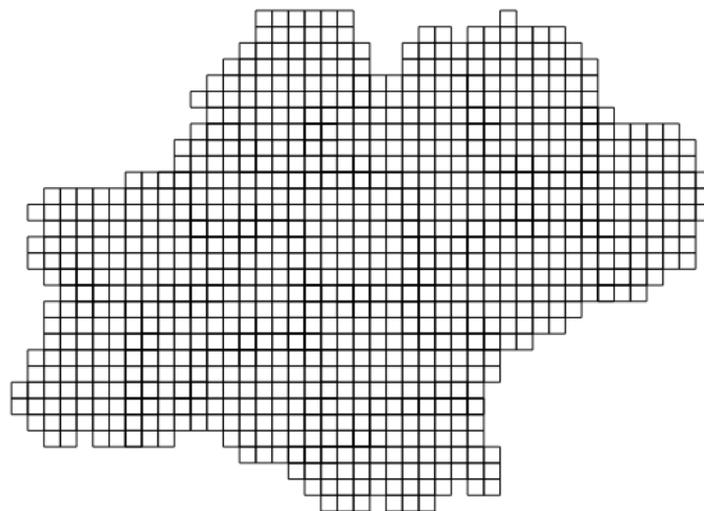


Figure 12: Indexe de tuile de la région Occitanie

Ce n'est qu'après avoir généré l'index de tuile qu'on génère les tuiles. La génération des tuiles s'effectuent en ligne de commande aussi comme pour l'index en spécifiant le chemin du répertoire où sont stockés les rasters. S'en suit l'utilitaire d'amorçage *mapcache_seed*, le *tileset* à générer, le niveau de zoom, la taille de la *metatile* à utiliser, le nombre de processus, le cache à utiliser et le chemin d'accès au fichier de configuration *mapcache.xml*.

```
d:\tuilage\SC25>mapcache_seed.exe -t scans -z 0,11 -M8,8 -n 4 -C disk_cache_oncfs -c C:\ms4w\apps\mapcache\mapcache.xml
```

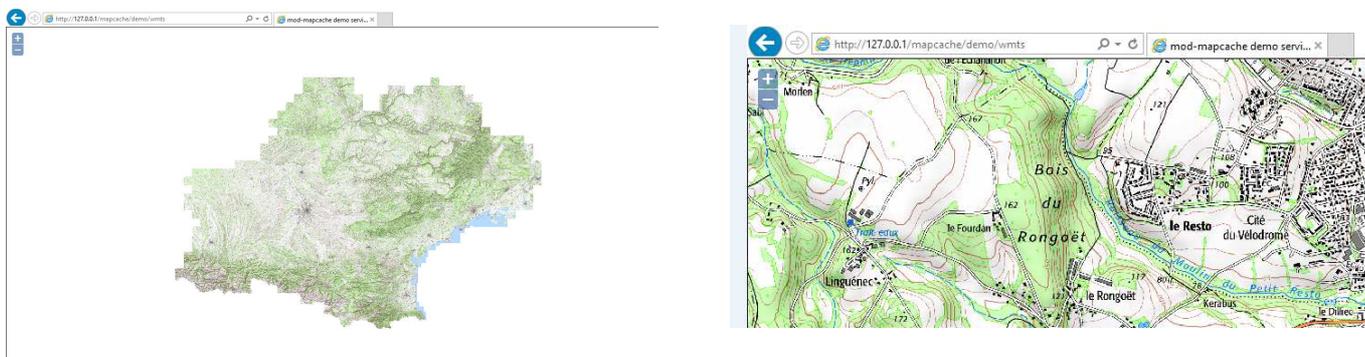


Figure 13 : Génération des tuiles Occitanie

C'est à la suite de ce test concluant que le traitement sur la France métropolitaine s'est mis en place. Il consista à tuiler les scan25, scan100 et scan1000 et avec des niveaux de zoom différents c'est-à-dire pour chaque scan ses niveaux de zoom minimal et maximal, ce qui a conduit à mettre en place le système de pyramidage des tuiles.

| Pyramide | Scan | Niveau de zoom | Echelle d'affichage |
|--------------------|-----------------|----------------|--------------------------------------|
| Pyramide des scans | Scan1000 | 00 - 04 | 1:750 000 1:100 000 |
| | Scan100 | 05 - 08 | 1:150 000 1: 50 000 |
| | Scan25 | 09 -11 | 1:18 000 1: 7000 |

Tableau 4: Échelle du cache de la pyramide des scans

Dans la pyramide, chaque niveau de zoom représente un étage. C'est une pyramide multi échelle allant de 1 :150 000 à 1 :7000 ce qui permet une visualisation continue des scans du zoom 00 au zoom 11 (unité de zoom classique en Webmapping, chaque niveau correspondant à un clic sur la carte).

La génération des tuiles du scan1000 pour la France entière a été très rapide et n'a causé aucun problème de paramétrage comme le montre le tableau ci-dessous.

| Scan | Volume de données | Niveau de zoom | Temps de rendu | Nombre de tuiles par seconde | Total Tuiles |
|----------|-------------------|----------------|----------------|------------------------------|--------------|
| Scan1000 | 108 Mo | 00_04 | 32,8 secondes | 46,9 tuiles/sec | 1536 |

Tableau 5: Temps de rendu des tuiles du scan1000

Ainsi, en 32,8 secondes les tuiles du scan1000 ont été générées du niveau zéro au quatrième niveau de zoom et a permis d'obtenir le résultat suivant :

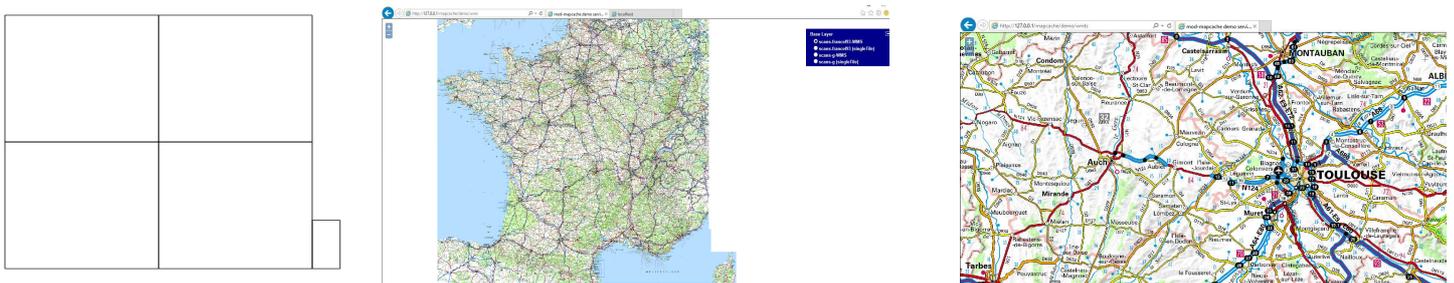


Figure 14: Génération de l'index et des tuiles du scan 1000 du zoom niveau 00 au 04

La génération des tuiles pour le scan100 a été aussi rapide mais moins que le scan1000. Le volume de données est plus important avec 21,9Go mais n'a causé cependant pas de problème lors de la génération des tuiles.

| Scan | Volume de données | Niveau de zoom | Temps de rendu | Nombre de tuiles par seconde | Total Tuiles |
|---------|-------------------|----------------|-----------------|------------------------------|--------------|
| Scan100 | 21,9Go | 05_08 | 1779,3 secondes | 157,4 tuiles/sec | 279.360 |

Tableau 6: Temps de rendu des tuiles du scan100

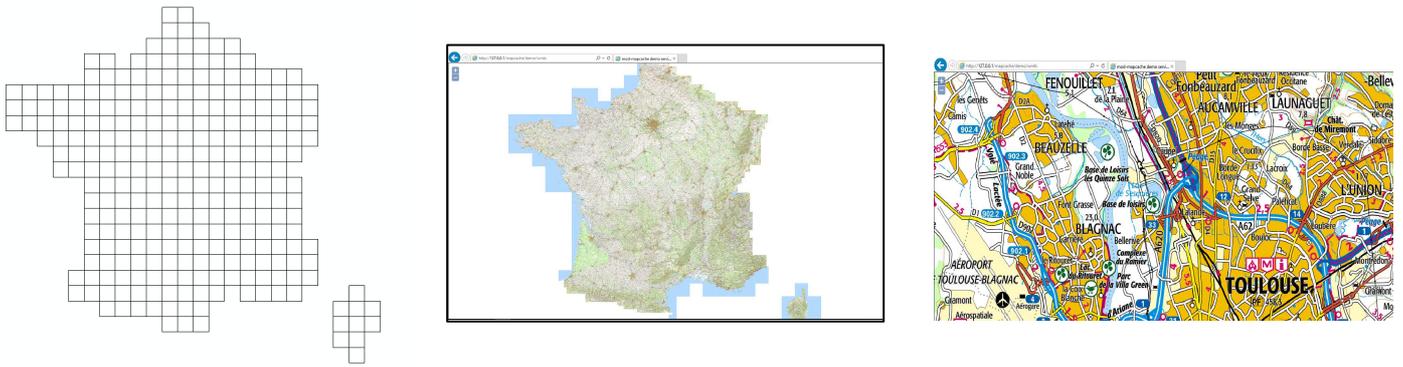


Figure 15: Génération de l'index et des tuiles du scan 100 du zoom niveau 05 au 08

Autant la génération des tuiles pour les scans 100 et 1000 a été simple et rapide, autant il a été difficile de les générer pour le scan25 à cause du volume de données. La construction des tuiles consomme de l'espace disque et des ressources CPU conséquentes. Vu que les machines avec lesquelles j'effectuais ce travail n'étaient pas assez puissantes, des difficultés furent rencontrées.

Processeur : Intel(R) Xeon(R) CPU E5-2650 v4 @ 2.20GHz 2.20 GHz (4 processeurs)
 Mémoire installée (RAM) : 16,0 Go

Figure 16: Configuration du serveur

De ce fait il a fallu mettre en place une procédure d'optimisation pour résoudre le problème.

3.6. Optimisation

Après plusieurs tentatives de générer les tuiles du scan25 sans succès et avec toujours le message d'erreur suivant:

The requested URL returned error: 504 Gateway Timeout aborting seed as 16.7% of the last 1000 requests failed

Après l'optimisation de la configuration de MapServer et augmentation de la puissance du processeur et de la RAM sans succès, j'ai mis en place un processus de mosaïquage. Ce processus permet de réduire le volume de données (145Go scan25). L'index de tuile ci-dessous est assez illustratif.

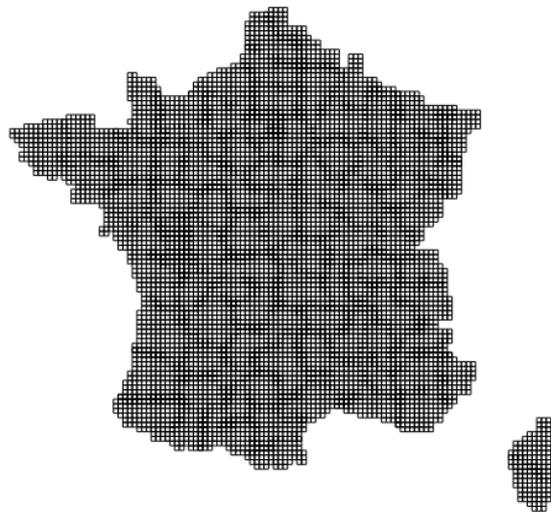


Figure 17: Index de tuile du scan25 avant mosaïque

Il s'agit donc réduire les nombreux rasters en dalles plus grandes à travers le mosaïquage. Mais des difficultés sont aussi rencontrées dans ce processus de mosaïquage. En premier lieu j'ai procédé à une mosaïque par département avec qgis à travers un traitement par lot.

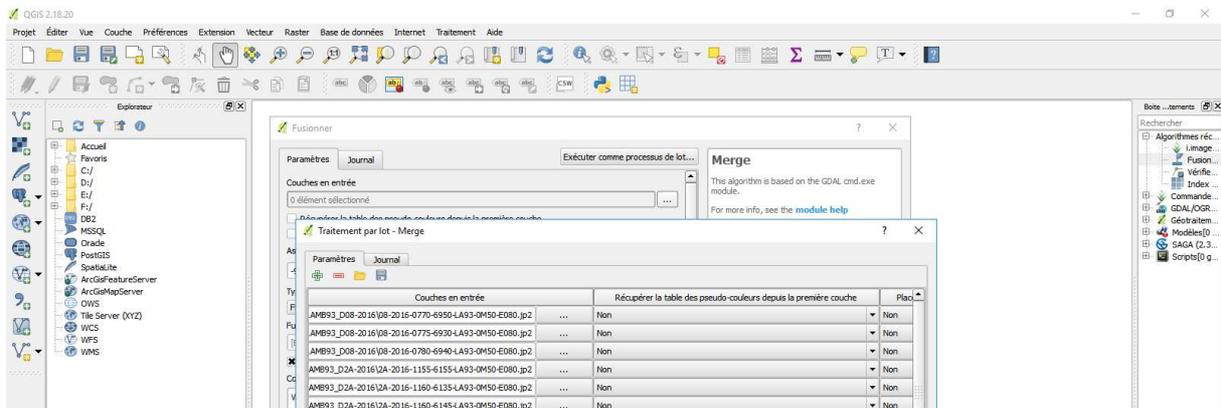


Figure 18: Processus de mosaïque sur Qgis

Mais la mosaïque par département n'a pas résolu le problème. En fait quand tu génère l'index de tuile les départements se chevauchent et mapcache ne génère pas correctement les tuiles.

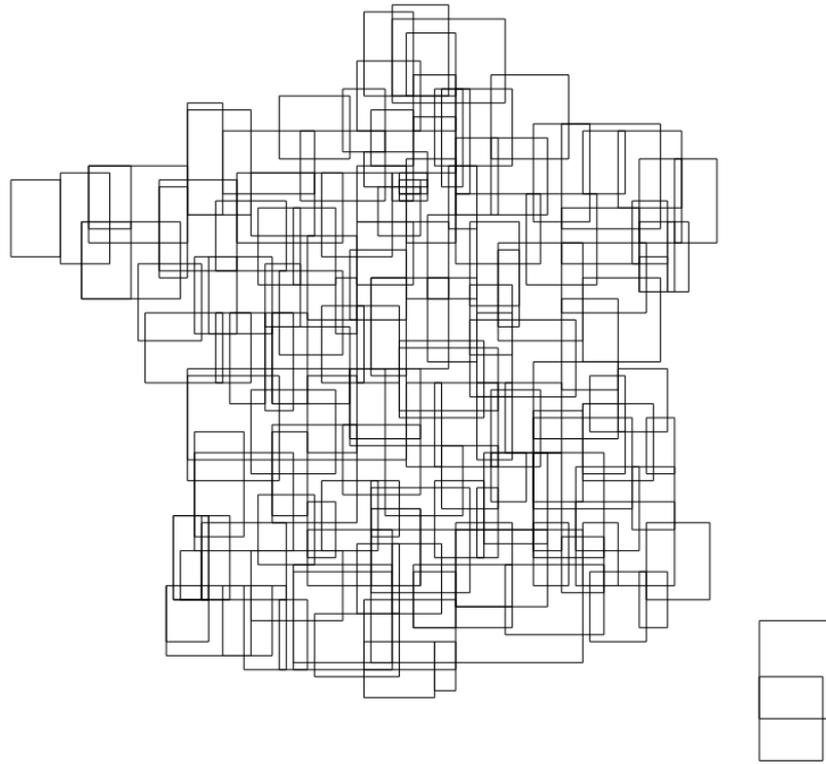


Figure 19: Index de tuile mosaïque par département

Donc il a fallu trouver un moyen de mosaïquer les données sans chevauchement. Le scan25 est découpé en carrés de 10 km sur 10 km en augmentant ces carrés en 100km sur 100km, j'ai dû réduire le volume de données en procédant comme suit :

| | | | |
|---|------------------|---|-----------|
| SC25_TOPO_0220_6860_L93_E100.jp2 | 14/01/2016 16:30 | Fichier JP2 | 22 423 Ko |
| SC25_TOPO_0220_6870_L93_E100.jp2 | 14/01/2016 16:30 | Fichier JP2 | 21 020 Ko |
| SC25_TOPO_0220_6880_L93_E100.jp2 | 14/01/2016 16:30 | Fichier JP2 | 20 470 Ko |
| SC25_TOPO_0220_6890_L93_E100.jp2 | 14/01/2016 16:30 | Fichier JP2 | 1 909 Ko |
| SC25_TOPO_0700_6860_L93_E100.jp2 D:\tuilage\scan25 | | Modifié le: 12/04/2017 14:35 Taille: 14,2 Mo | |
| SC25_TOPO_0960_6860_L93_E100 (2).jp2 D:\tuilage\scan25 | | Modifié le: 12/04/2017 14:35 Taille: 14,1 Mo | |
| SC25_TOPO_0960_6860_L93_E100.jp2 D:\tuilage\scan25 | | Modifié le: 12/04/2017 14:35 Taille: 14,1 Mo | |
| SC25_TOPO_0950_6860_L93_E100 (2).jp2 D:\tuilage\scan25 | | Modifié le: 12/04/2017 14:34 Taille: 14,1 Mo | |

Figure 20: Série de données scan25

Avec le temps de rendu ci-dessous les tuiles du scan25 ont été générées du 09 au 11ème niveau de zoom.

| Scan | Volumes de données | Niveau de zoom | Temps de rendu | Nombre de tuiles par seconde | Total Tuiles |
|--------|--------------------|----------------|------------------|------------------------------|--------------|
| Scan25 | 145Go | 09_11 | -1077,7 secondes | 4046,1 tuiles/sec | 4.360.576 |

Tableau 7: Temps de rendu des tuiles du scan25



Figure 21: Génération de l'index et des tuiles du scan 25 du Zoom niveau 09 au 11

Tous les scans sont regroupés en pyramide ou chaque niveau de zoom correspond à un scan, cela permet une visualisation continue de la carte topographique sur l'amplitude totale d'échelle désirée par l'Office.

3.7. La génération des tuiles pour les orthophotos

Comme pour le scan25, la génération des tuiles pour les orthophotos n'était pas facile à mettre en place vu que ces rasters sont beaucoup plus lourds. Mais, avant la génération des tuiles pour la France métropolitaine, j'ai testé la génération à l'échelle départementale avec le département de la Haute-Garonne. Ce test a été effectué avec l'ortho50cm et n'a causé pas de problème lors de la génération.



Figure 22: Génération de l'index et des tuiles de l'ortho50cm du département de la Haute Garonne

Après ce test sur le département de la Haute Garonne le processus de génération des tuiles pour la France métropolitaine s'est mis en place. Comme pour les scans, la génération des tuiles pour les orthophotos a été faite sous forme de pyramide aussi où chaque étage correspond à un niveau de zoom.

| Pyramide | orthophotos | Niveau de zoom | Echelle d'affichage |
|--------------------------|--------------------|-----------------------|--|
| Pyramide des orthophotos | ortho5m | 00 - 05 | 1:188 976 1:7 559 040 |
| | Ortho50cm | 06 - 13 | 1:94 488 1:1000 |

Tableau 8: Échelle du cache de la pyramide des orthophotos

Ainsi chaque étage correspond à un niveau de zoom ce qui permet une visualisation continue des orthophotos avec 13 niveaux de zoom.

La génération des orthophotos a nécessité aussi une mosaïquage comme le scan25 et le processus reste le même. Par contre le traitement a pris beaucoup plus de temps surtout pour l'ortho50cm avec 390Go. Quant à l'ortho5m il est moins lourd avec 71,4Go.

| Scan | Volumes de données | Niveau de zoom | Temps de rendu | Nombre de tuiles par seconde | Total Tuiles |
|-------------|---------------------------|-----------------------|-----------------------|-------------------------------------|---------------------|
| Ortho5m | 71.4 | 00_05 | 106.5 secondes | 62.2tuiles/sec | 1953 |

Tableau 9: Temps de rendu des tuiles de l'ortho5m



Figure 23: Génération des tuiles de l'ortho5m du zoom niveau 00 au 05

Les tuiles générées sont stockés dans des dossiers nommés scans et orthos. Il contient les caches des scans et des orthos organisés par niveau d'échelle en format png pour le scan et jpeg pour l'orthophotographie. Cette capture d'écran ci-dessous montre l'organisation du cache scans dans l'environnement windows.

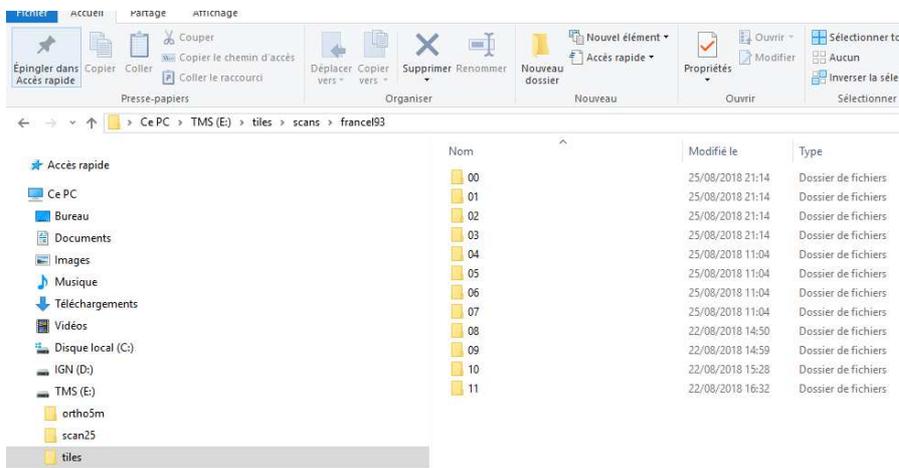


Figure 24: Organisation du cache scans

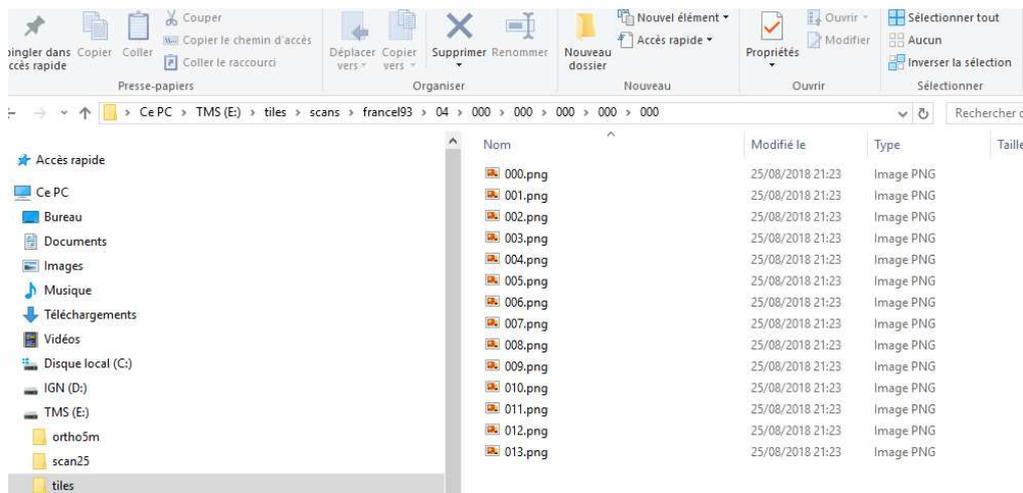


Figure 25: Vision du cache scan

4. Planning prévisionnel

Pour mener à bien les différentes tâches qui me furent confiées, j'ai mis en place un planning prévisionnel. Ce planning m'a permis de suivre la répartition des tâches dans le temps. Le diagramme ci-dessous donne la répartition des tâches des différentes tâches.

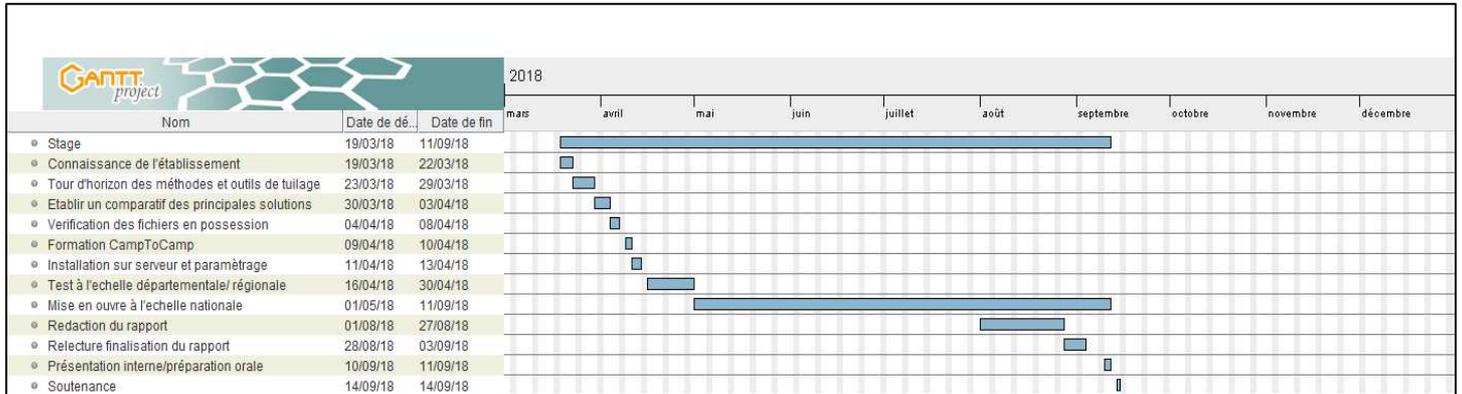


Figure 26: Diagramme de Gantt

La mise en œuvre de toutes ces tâches m'a permis de développer des compétences diverses dans le domaine des SIG mais m'a également permis de me familiariser avec l'outil MapServer, vu rapidement lors de la formation.

5. Difficultés rencontrées

Lors de ce stage, j'ai été confrontée à beaucoup de difficultés. La première réside dans le fait d'être face à une situation nouvelle et il m'a fallu du temps pour pouvoir intégrer le projet.

Ensuite, ce sont posés des problèmes liés aux machines qui n'étaient pas assez puissantes par rapport aux traitements à effectuer. La génération du cache consomme de l'espace disque avec les tuiles mais aussi des ressources CPU pour la génération des images.

En outre, le serveur sur lequel je travaillais était à Paris et à chaque fois que je rencontrais des problèmes avec ce dernier il fallait contacter l'administrateur, cela prenait des fois du temps, ce qui a entraîné des lenteurs dans le traitement des données.

Enfin, les serveurs de l'ONCFS sont des serveurs Windows la plupart des recherches que j'ai effectué ont montré que les serveurs sous Linux sont mieux documentés et plus efficaces pour ce genre de traitement. Toutes ces difficultés rencontrées ont impacté la bonne marche du stage.

6. Conclusion et perspectives

▪ Dans le cadre de ce stage, mon travail consistait à générer des fonds cartographiques pour les scans et la BD Ortho à l'Office National de la Chasse et de la Faune Sauvage.

L'objectif était de tester différentes solutions de mise en cache des tuiles afin de choisir la solution la mieux adaptée aux besoins de l'ONCFS. Les besoins étant, de pré-générer les tuiles, de générer les tuiles à la demande, de générer des données mbtiles pour mobile déconnecté, des tuiles sur carte SD avec une solution performante dans un environnement windows.

Ce choix ne s'est pas fait à travers des tests mais à travers différents critères : l'environnement windows, la performance, le fonctionnel, le langage, le retour d'expérience d'utilisateur, le coût etc.

Parmi tous ces critères, Mapcache apparait comme la solution la mieux adaptée par rapports aux besoins de la structure. Il est non seulement performant mais aussi il est adapté à l'environnement windows, génère les tuiles à la volée, générer des mbtiles etc.

Après l'installation et le paramétrage de Mapcache des tests de génération des tuiles à l'échelle départementale et régionale ont été réalisés avec succès. Mais la génération au niveau national a causé des difficultés d'où l'optimisation de la configuration de Mapcache, l'augmentation du CPU et la RAM. Ce processus n'ayant pas résolu le problème il a fallu mettre en place une procédure de mosaïquage avec des difficultés dans son élaboration mais qu'on a réussi à surmonter après de multiples tentatives. Un important travail de traitement des données a été mise en œuvre pour mener à bien le projet.

Ce travail a permis d'aboutir à l'actualisation des différents fonds de carte qui existaient depuis 2012. Les tuiles mise en cache sont déployées sur Géo-générateur afin d'être visualisées sur les applications, mais aussi déposées sur les mobiles dans des cartes SD pour les usages déconnectés.

Ainsi, les scans 25, 100 et 1000 ainsi que l'ortho 5m ont été générés, l'ortho50cm est en cours de génération. L'ONCFS aurait aimé tuiler le cadastre mais par manque de temps cela n'a pas été possible durant ses 6 mois de stage.

Par conséquent, le tuilage du cadastre reste une perspective à voir pour l'Office National de la Chasse et de la Faune Sauvage.

Mosaiquer les rasters par département qui se chevauchent lors de la génération de l'index de tuile fait perdre du temps et crée des problèmes au niveau du recouvrement des départements. Afin d'éviter cela, il faut se baser sur le maillage utilisé par l'IGN comme illustrer dans la figure 20.

J'ai créé un document explicatif du procédé de tuilage et du paramétrage de Mapserver, Mapcache ainsi que la construction du mapfile que j'ai mis à disposition de l'Office National de la Chasse et de la Faune Sauvage pour que le travail puisse continuer.

D'un point de vue personnel, cette période m'a permis de mettre en œuvre les capacités acquises lors du Master2 SIGMA. Ainsi il m'a permis de me familiariser avec les serveurs cartographiques, leurs fonctionnements mais aussi les outils de tuilage.

Effectuer ce stage à l'ONCFS m'a permis aussi de connaître leurs travaux et le fonctionnement d'une structure de la fonction publique.

Enfin, ce stage m'a permis de renforcer mes compétences en gestion de projet et en travail d'équipe.

Bibliographie et webographie

Jégou, Laurent, 2013, Support de cours M2 Sigma: Module 904_23, Cartographie et SIG interactifs en ligne.

<http://blog.martzluff.net/category/tutoriels-sig/page/2/> Géomatique libre et autres astuces cartographiques

<https://www.neogeo-online.net/blog/archives/1727/> Petite histoire de tuilage

<http://www.oncfs.gouv.fr> Office National de la Chasse et de la Faune Sauvage

<https://www.business-geografic.com/fr/> Business Geografic

<https://www.openstreetmap.org> OpenStreetMap

<https://www.google.fr/maps> Google Maps

<https://www.bing.com/maps> Bing Maps

<https://www.unamur.be/sciences/geographie/etudiants/logiciels-sig-libres-et-webmapping>

Logiciels SIG libres et Webmapping

<https://fr.wikipedia.org/wiki/Interop%C3%A9rabilit%C3%A9> définition intéropérabilité

https://fr.wikipedia.org/wiki/Open_Geospatial_Consortium définition Open Geospatial Consortium

http://docs.geoserver.org/stable/en/user/services/wms/get_legend_graphic/ GeoServer

<http://blog.martzluff.net/realiser-un-carte-dimages-aeriennes-comparees-avec-qgis-gdal-et-leaflet/> Construction des tuiles

<https://geoservices.ign.fr/documentation/geoservices/wmts.html> Utilisation avancée de couches en WMTS

<https://wiki.openstreetmap.org> OSM Wiki

<https://docs.qgis.org> QGIS Documentation

<http://geowebcache.org/> GeoWebCache

<https://mapproxy.org> Caractéristiques de MapProxy

<https://lists.osgeo.org/pipermail/mapproxy/2015-December/002276.html> Mapproxy

<https://live.osgeo.org> OSGéo

<https://doc.arcgis.com/fr/arcgis-online/reference/what-is-ago.html> Présentation d'ArcGIS Online

<https://mapserver.org/fr/mapcache> Présentation de MapCache

<https://www.camptocamp.com/> Des solutions innovantes par des experts Open source

<https://ms4w.com/> MapServer for Windows

<http://www.portailsig.org/faq/mapserver/presentation-0> Qu'est-ce que MapServer

<http://web.mit.edu/rhel-doc/4/RH-DOCS/rhel-rg-fr-4/s1-apache-virtualhosts.html> Hôtes
virtuels

https://wiki.openstreetmap.org/wiki/Meta_tiles Wiki Meta tiles

<http://professionnels.ign.fr/donnees> Institut National de l'Information Géographique et
Forestière

<http://professionnels.ign.fr/orthoimages> Institut National de l'Information Géographique et
Forestière, Ortho-images

<http://professionnels.ign.fr/bdortho-50cm> Institut National de l'Information Géographique et
Forestière, Bd Ortho

https://live.osgeo.org/fr/overview/qgis_mapserver_overview.html QGIS Server

<https://bioinfo-fr.net/cython-votre-programme-python-mais-100x-plus-vite>

Table des illustrations

| | |
|--|----|
| Figure 1: Implantations de l'ONCFS en France métropolitaine, Source : ONCFS..... | 10 |
| Figure 2: Organigramme Direction des Systèmes d'informations, Source : ONCFS, janvier 2018..... | 11 |
| Figure 3: Interface du portail BD BIODIV ; Source : ONCFS | 12 |
| Figure 4: Interface Géo Générateur, Source : ONCFS..... | 13 |
| Figure 5: °Principaux opération du WMS, Source : géorezo | 18 |
| Figure 6: Résultat requête WMTS | 19 |
| Figure 7: Fonctionnement d'un serveur carte avec ou sans cacheSource : https://www.sigterritoires.fr/index.php/geoserver-avance-le-tuilage-principes/ | 20 |
| Figure 8: Pyramide de tuile, Source : IGN..... | 21 |
| Figure 9: Schéma des serveurs de l'ONCFS, Source : ONCFS..... | 29 |
| Figure 10: Schéma du serveur entre Toulouse et Saint-Benoît..... | 30 |
| Figure 11: Fonctionnement du serveur de tuilage, Réalisation personnelle | 31 |
| Figure 12: Index de tuile de la région Occitanie | 36 |
| Figure 13 : Génération des tuiles Occitanie | 37 |
| Figure 14: Génération de l'index et des tuiles du scan 1000 du zoom niveau 00 au 04 | 38 |
| Figure 15: Génération de l'index et des tuiles du scan 100 du zoom niveau 05 au 08 | 39 |
| Figure 16: Configuration du serveur | 39 |
| Figure 17: Index de tuile du scan25 avant mosaïque | 40 |
| Figure 18: Processus de mosaïque sur Qgis | 40 |
| Figure 19: Index de tuile mosaïque par département | 41 |
| Figure 20: Série de données scan25 | 41 |
| Figure 21: Génération de l'index et des tuiles du scan 25 du Zoom niveau 09 au 11..... | 42 |
| Figure 22: Génération de l'index et des tuiles de l'ortho50cm du département de la Haute Garonne | 43 |
| Figure 23: Génération des tuiles de l'ortho5m du zoom niveau 00 au 05 | 44 |
| Figure 24: Organisation du cache scans | 44 |
| Figure 25: Vision du cache scan | 44 |
| Figure 26: Diagramme de Gantt..... | 45 |

Table des tableaux

| | |
|--|----|
| Tableau 1 : Consommation des ressources, Source : IGN, avril 2017) | 14 |
| Tableau 2: Consommation des ressources, Source : IGN, avril 2017 | 15 |
| Tableau 3: Tarification des transactions des Géoservices IGN, Source : IGN, avril 2017)..... | 15 |
| Tableau 4: Échelle du cache de la pyramide des scans | 37 |
| Tableau 5: Temps de rendu des tuiles du scan1000 | 38 |
| Tableau 6: Temps de rendu des tuiles du scan100 | 38 |
| Tableau 7: Temps de rendu des tuiles du scan25 | 42 |
| Tableau 8: Échelle du cache de la pyramide des orthophotos..... | 43 |
| Tableau 9: Temps de rendu des tuiles de l'ortho5m | 43 |