

**Institut Universitaire de Technologie,
Aix-Marseille Université**

**RAPPORT DE STAGE
Diplôme Universitaire de Technologie
Spécialité Réseaux et Télécommunications**

Mise en place de plateforme logicielle pour la
gestion interopérable de données
environnementales

Matthew PHILIPPE

OSU PYTHEAS

Responsable entreprise : Maurice Libes
Responsable académique : Anouch Hovsépian

2019

Table des matières

1	Introduction.....	2
2	Sujet du stage.....	3
2.1	Contexte.....	3
2.2	Mission et objectifs.....	4
2.3	Matériel à disposition.....	5
3	Travail réalisé.....	7
3.1	Serveur HTTP Apache Tomcat.....	7
3.1.1	Description.....	7
3.1.2	Mise en place.....	8
3.1.3	Définition d'un utilisateur.....	8
3.1.4	Augmentation de la mémoire Java.....	8
3.1.5	Utilisation de Tomcat sur les ports 80 ou 8080.....	8
3.1.6	Taille de l'archive web déployable.....	9
3.1.7	Passage en HTTPS.....	9
3.1.8	Utilisation de Tomcat.....	10
3.2	Serveur cartographique Geoserver.....	11
3.2.1	Description.....	11
3.2.2	Installation de Geoserver.....	12
3.2.3	Sécurité CORS.....	12
3.2.4	Ajout d'extension.....	12
3.2.5	Mise à jour/Changement de serveur.....	13
3.2.6	Utilisation de Geoserver.....	13
3.2.7	Création de style.....	15
3.2.8	Création d'une couche de données Shapefile.....	16
3.3	Catalogue de données Geonetwork.....	17
3.3.1	Description.....	17
3.3.2	Installation de Geonetwork.....	18
3.3.3	Ajout d'un thésaurus.....	18
3.3.4	Sauvegarde/Mise à jour/Changement de serveur.....	18
3.3.5	Utilisation de Geonetwork.....	19
4	Conclusion.....	21
5	Remerciements.....	23
6	Glossaire.....	25
7	Bibliographie.....	27

1 Introduction

L'Observatoire des Sciences de l'Univers (OSU) Pythéas est un institut qui dépend de l'université d'Aix-Marseille.

L'OSU Pythéas a été créé en 2012 et contient 5 grandes UMRs (Unités Mixtes de Recherche):

- CEREGE, Centre Européen de Recherche et d'Enseignement des Géosciences de l'Environnement,
- IMBE, l'Institut Méditerranéen de Biodiversité et d'Écologie marine et continentale,
- LAM, Laboratoire d'Astrophysique de Marseille,
- LPED, Laboratoire Population Environnement Développement,
- MIO, l'Institut Méditerranéen d'Océanologie.

L'OSU Pythéas est également une école interne de l'Université d'Aix-Marseille. Il est placé sous la tutelle du CNRS (Centre National de la Recherche Scientifique) et de l'IRD (Institut de Recherche pour le Développement), couvrant les grandes thématiques scientifiques des sciences de la Terre, de l'Environnement et de l'Univers : les géosciences de l'environnement, les sciences de la biodiversité et écologie, l'astronomie, l'océanologie, l'interaction sociétés – environnement.

L'OSU a pour mission principale de contribuer à l'enrichissement des connaissances à travers la direction de grands programmes de recherche. L'observation du milieu naturel est également une des grandes missions de l'OSU – Institut Pythéas qui doit assurer et maintenir une activité d'observation permanente des phénomènes naturels et anthropiques sur l'environnement.

Cet organisme a également pour but de participer à la conception d'instrument d'observation, de valoriser ses recherches en développant des liens avec le monde industriel, et de participation à la formation universitaire ainsi que la diffusion de la culture scientifique.

Il y a plus de 1000 personnes qui travaillent pour l'Institut Pythéas sur 11 sites : Marseille – Luminy, Timone, Endoume, St Charles, St Jérôme, Technopole de Château-Gombert – Europôle de l'Arbois, Toulon, La Seyne sur Mer, St Michel l'Observatoire, Avignon ainsi que des équipes dans les DOM-TOM, au Mexique et au Maghreb.

J'ai effectué mon stage dans les locaux du laboratoire MIO sur Luminy, avec Mr Maurice LIBES afin de participer à la constitution d'un portail d'accès aux données environnementales de l'OSU Pythéas.

2 Sujet du stage

2.1 Contexte

Durant mes 10 semaines de stage, j'ai travaillé pour Monsieur Maurice LIBES, Ingénieur au Service d'Observation de l'OSU en Système d'Information Géographiques.

Son travail consiste, avec ses collègues, Mr RAIMBAULT Patrick basé à Luminy et Mr MALLARINO Didier basé à Toulon, à constituer et mettre en place un portail d'accès aux données environnementales collectées sur le terrain par les unités de recherche de l'OSU.

L'objectif de ce portail de données environnementales est de rendre les données de l'OSU «FAIR» c'est-à-dire Facile à trouver, Accessibles, Interopérables, et Réutilisables.

Pour mettre cela en place, il faut des outils qui permettent d'inventorier et cataloguer les jeux de données avec leurs métadonnées attenantes, et en permettre l'accès, la visualisation et la géolocalisation.

Pour cela, nous avons eu besoin de tester et d'installer des outils logiciels permettant de fournir les fonctions ci-dessus. Nous avons donc testé et installé les logiciels suivants :

- ERDDAP (Environmental Research Division Data Access Protocol) : qui permet d'accéder aux données brutes et de les visualiser,
- Geonetwork qui permet de Cataloguer les jeux de données,
- Géoportail et Geoserver : qui permet de géolocaliser les jeux de données.

Le premier outil mis en place est ERDDAP (Figure 1) qui est un serveur de données donnant un moyen simple et cohérent de télécharger des ensembles de données scientifiques sous forme de grilles ou de tableaux. ERDDAP permet un accès aux données brutes et de faire des graphes de ces données. Son objectif est de faciliter l'accès, la disponibilité et l'utilisation des données scientifiques dans les formats interopérables les plus courants.

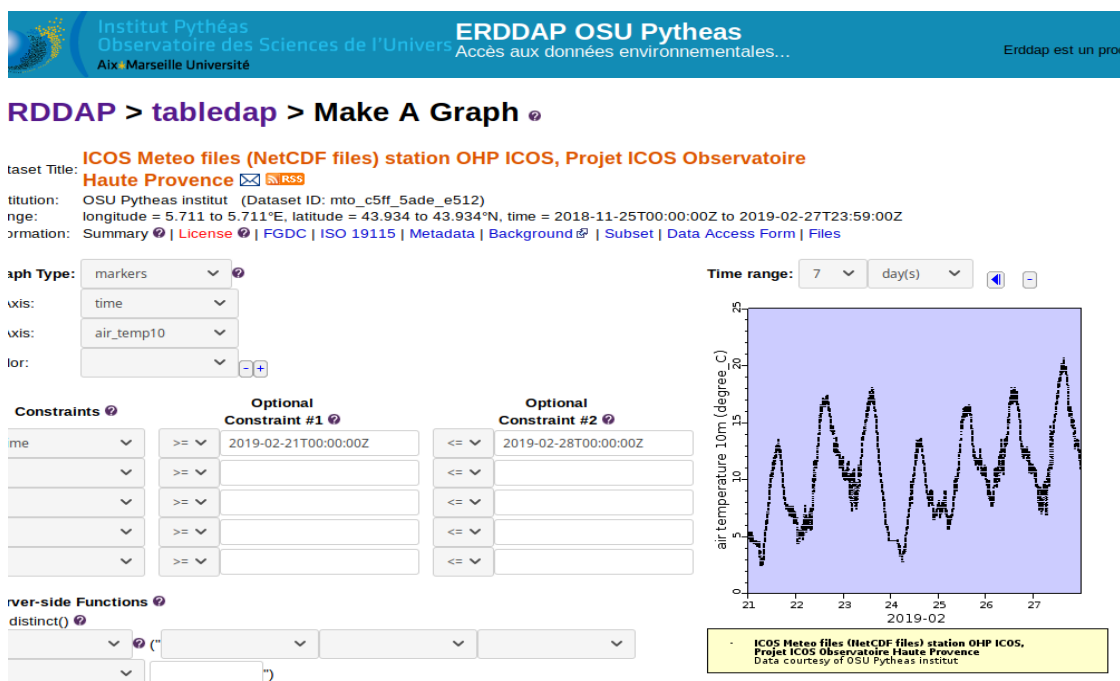


Figure 1 : Interface de ERDDAP permettant de faire un graphe des données brutes
<https://coastwatch.pfeg.noaa.gov/erddap/index.html>

Le second outil (Figure 2) est un développement local effectué en php à l'OSU Pytheas, il a été mis en place par Mr MALLARINO Didier et il permet de géolocaliser les jeux de données sur une carte grâce aux coordonnées Latitude/Longitude présentes dans les fiches de métadonnées de Geonetwork.

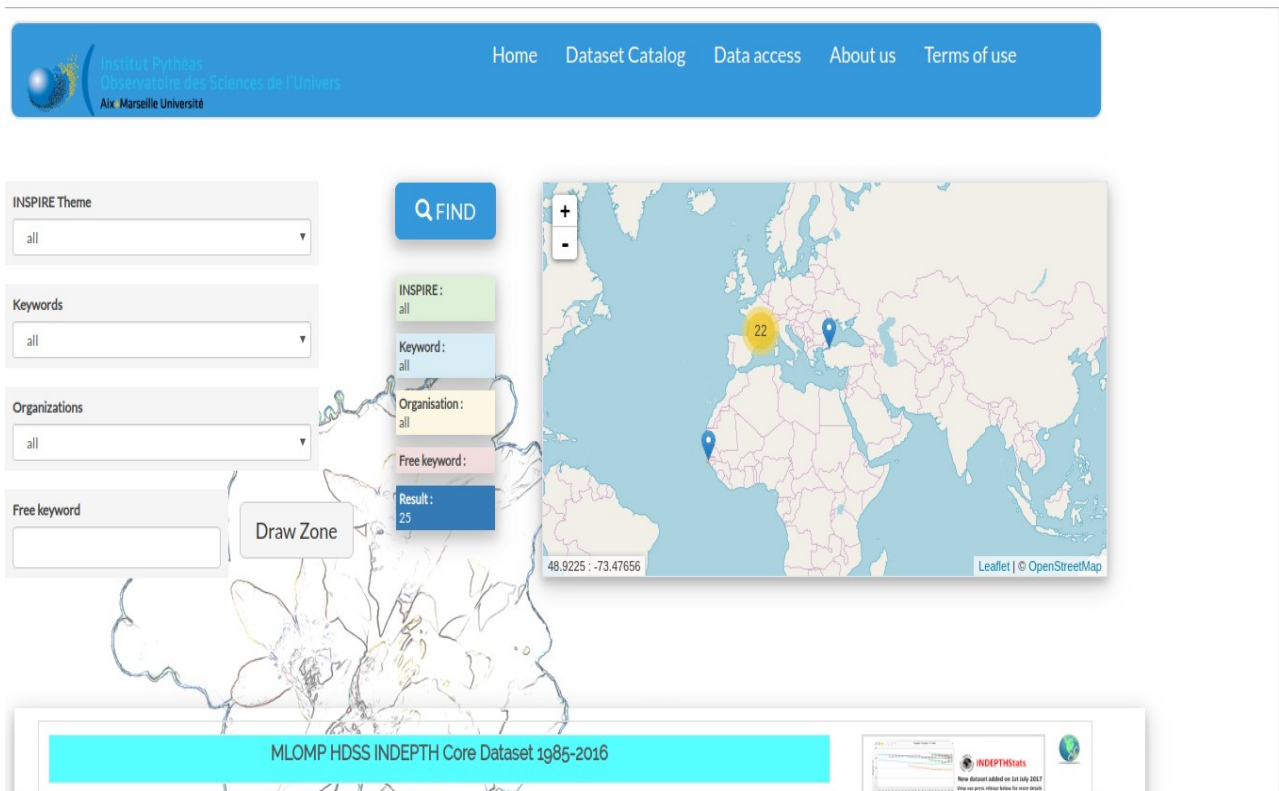


Figure 2 : Interface de Géoportail
<http://geoportail.osupytheas.fr/>

Le troisième outil Geonetwork (Figure 3), permet de cataloguer les jeux de données en fournissant les métadonnées normalisées ISO19139 et de fournir un lien d'accès aux données brutes qui renvoie vers ERDDAP.

Dans le cadre du portail d'accès aux données, Geonetwork a pour objectif d'identifier et inventorier les jeux de données environnementales des UMR de l'OSU Pytheas.



Figure 3 : Logo de Geonetwork
<https://geonetwork-opensource.org/>

Le dernier outil mis en place s'appelle Geoserver (Figure 4), qui est un serveur de données cartographiques qui permet de fournir et partager des données géo spatialisées.

C'est un serveur qui agit de manière interopérable avec Geonetwork au travers d'un protocole standardisé WMS (Web Map Service) de l'OGC (Open Geospatial Consortium).



Figure 4 : Logo de Geoserver
<http://geoserver.org/>

2.2 Mission et objectifs

L'objectif de mon stage à l'OSU Pythéas a été de mettre en place une plateforme de données environnementales « FAIR » pour le Service d'Observation de l'OSU.

Dans ce cadre, Mr LIBES m'a donné pour mission d'installer 2 serveurs : Geoserver et Geonetwork.

Afin d'en comprendre l'utilisation, je suis allé me documenter sur différentes sources. La première source que j'ai utilisé est la documentation présente sur le site officiel. La seconde source a été les divers forums et tutoriels présents sur le web concernant Geoserver et Geonetwork. J'ai pu ensuite tester les logiciels avec des machines virtuelles, fournis par des collègues de Mr LIBES, sous VirtualBox.

Après avoir compris ma mission, et acquis une vue d'ensemble sur ce que proposaient ces deux applications, il m'a fallu passer ces logiciels en version de développement et en version d'exploitation sur le réseau de l'OSU.

Pour réussir ma mission j'ai dû passer par plusieurs étapes. Il m'a fallu savoir installer, configurer et utiliser Geoserver et Geonetwork. Ces 2 applications sont fournies sous forme de servlet (classe Java qui permet de créer dynamiquement des données au sein d'un serveur web), empaquetées sous forme d'archive web (.war) et exécutées via un serveur Tomcat.

Pour exécuter ces servlet .war, il m'a fallu commencer par installer le serveur Apache Tomcat qui est un serveur HTTP (HyperText Transfer Protocol).

2.3 Matériel à disposition

Pour réaliser ma mission, j'ai utilisé du matériel informatique permettant de créer et travailler sur des machines virtuelles.

L'essentiel de mon travail a été effectué sur des containers virtuels. En effet afin de mettre en place les différents outils cités précédemment, j'ai utilisé un outil de virtualisation qui s'appelle Proxmox Virtual Environment (Figure 5).



Figure 5 : Logo de Proxmox Virtual Environnement
<https://www.proxmox.com/en/>

Proxmox est une solution de virtualisation libre basée sur l'hyperviseur Linux KVM (Kernel-based Virtual Machine), et offre aussi une solution de machines virtuelles sous forme de containers avec LXC (Linux Containers).

Proxmox intègre deux technologies de virtualisation que sont KVM et OpenVZ. Les deux solutions sont parfaitement intégrées aux interfaces web et en ligne de commande. KVM est une solution de virtualisation complète (hardware compris), permettant de virtualiser d'autres OS (Operating System), tels que Solaris et Windows par exemple. Tandis que les containers gérés avec LXC (Linux Container) ne permettent de virtualiser que des systèmes Linux.

Les avantages à travailler avec des VM (Machine Virtuelle) sont multiples : Outre les gains financiers, la virtualisation permet une souplesse au niveau de la gestion des serveurs, et une très bonne disponibilité puisqu'on peut facilement déplacer (migrer) et sauvegarder des serveurs facilement et rapidement. Les backups des VM qui sont effectués permettent un moyen de récupération facile en cas de dysfonctionnement de la VM.

Grâce à cet outil, j'ai pu travailler sur plusieurs VM me permettant d'installer et de faire différents essais sur les logiciels que j'ai dû tester.

Dans le cadre de mon stage, j'ai utilisé des containers Linux qui sont efficaces et suffisants pour obtenir des VM Linux utilisés pour les serveurs Geonetwork et Geoserver.

Le gestionnaire de VM Proxmox permet de créer des VM en allouant des ressources physiques (mémoire, processeurs, SWAP, etc, ...) prises sur le serveur physique.

Il est possible de créer autant de VM que l'on veut dans la limite des ressources que propose le serveur physique. De plus l'interface de Proxmox permet d'avoir accès à l'affichage des ressources utilisées par les machines (Figure 6) sur le serveur physique (Figure 7). J'ai pu me rendre compte que les ressources physiques que j'ai attribuées à mes machines virtuelles ont été suffisantes et correctes car elles n'ont pas surchargé l'hôte physique.

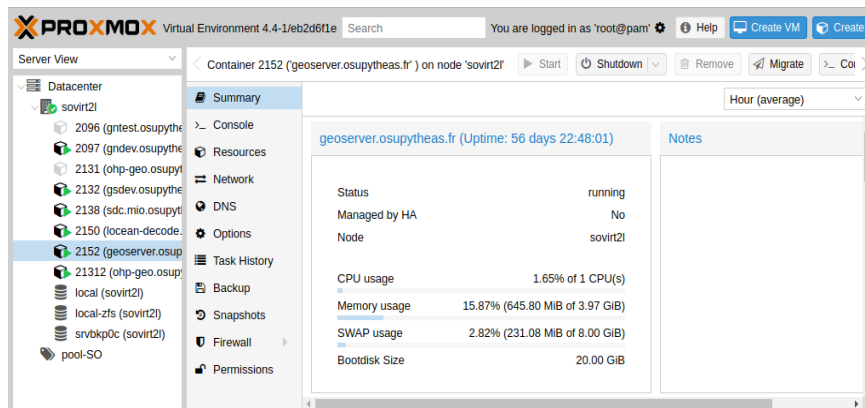


Figure 6 : Usage des ressources de la VM Geoserver

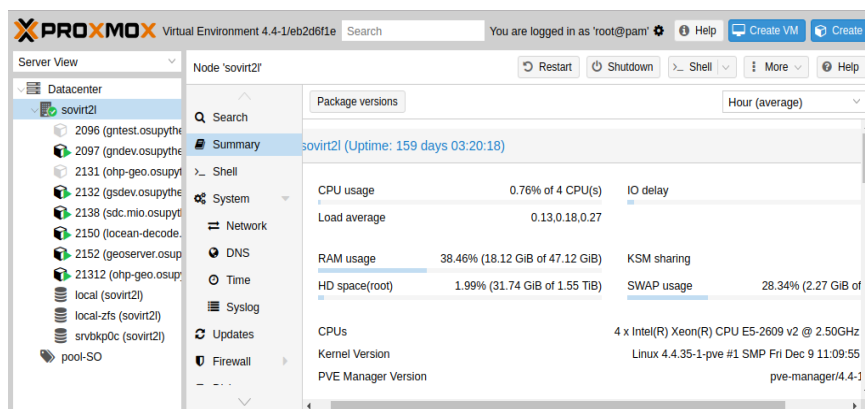


Figure 7 : Usage des ressources du serveur Proxmox

Les caractéristiques techniques que j'ai attribuées (Figure 6) aux VM que j'ai créé sont : 4GB de RAM (Random Access Memory), 8GB de SWAP, 20GB de disque dur tournant sur Debian 9.9. Pour chaque VM j'attribuais également une adresse IP (Internet Protocol) que mon tuteur a ensuite pu référencer dans le DNS (Domain System Name) afin de faire correspondre l'adresse IP au nom de domaine, avec son masque, une passerelle par défaut, 2 serveurs DNS, ainsi qu'un VMID (Virtual Machine Identifier).

3 Travail réalisé

Une fois la machine virtuelle créée, outre l'interface graphique de gestion des VM, on peut également gérer les VM en mode ligne de commandes. Pour ce faire, il faut se connecter au serveur Proxmox où j'avais créé la machine. Ensuite on gère les VM avec la commande « pct » en indiquant l'option et le VMID de la VM.

Exemple :

- pct start <vmid> : pour démarrer une VM,
- pct stop <vmid> : pour stopper une VM,
- pct enter <vmid> : pour rentrer dans le container de la VM en mode ligne de commande.

Préparation préalable

Sur chaque VM, j'ai dû autoriser la connexion en tant qu'utilisateur *root* via SSH (Secure Shell). J'ai fait la mise à jour des paquets via la commande « apt-get update && apt-get upgrade » et l'installation du paquet *ssh* via la commande « apt-get install ssh ».

Une fois l'installation du paquet *ssh* effectuée, il m'a fallu le configurer.

Pour cela, j'ai modifié le fichier */etc/ssh/sshd_config* en ajoutant la ligne suivante :

- PermitRootLogin yes

J'ai également copié ma clé SSH sur les serveurs afin de me connecter plus rapidement à ces derniers.

Pour mettre en place Geoserver et Geonetwork, j'ai créé 4 VM :

- 2 VM de développement qui m'ont permis de faire les tests nécessaires avant de passer en production,
- 2 VM de production.

3.1 Serveur HTTP Apache Tomcat



Figure 8 : Logo de Tomcat
<http://tomcat.apache.org/>

3.1.1 Description

Le logiciel Tomcat est une implémentation open source des technologies Java Servlet, JavaServer Pages, Java Expression Language et Java webSocket qui ont été développées dans le cadre du Java Community Process. Il est développé dans un environnement ouvert et participatif et est distribué sous la licence Apache version 2.

Il est paramétrable par des fichiers XML (Extensible Markup Language) et des propriétés, et inclut des outils pour la configuration et la gestion.

Tomcat est écrit en langage Java et peut être exécuté via la machine virtuelle Java sur n'importe quel système d'exploitation la supportant.

Tomcat m'a permis de déployer des applications sous forme d'archive web *geonetwork.war* et *geoserver.war*.

3.1.2 Mise en place

Pour installer Tomcat sur mes machines, j'ai eu deux options possible :

- Soit par les paquets Debian,
- Soit par la distribution tar.gz.

Chaque mode de distribution a ses avantages et inconvénients. J'ai choisi la solution des paquets Debian étant la plus rapide et la plus commune. De plus le paquet Debian est pré-configuré et prêt à l'emploi et nécessite moins de paramétrages que l'archive *.tar.gz* qui est plus générique.

J'ai donc installé 2 paquets, grâce à la commande « apt-get install », qui sont *tomcat8* et *tomcat8-admin*, avec l'installation de ces 2 paquets, la commande « apt » s'est chargée d'installer les dépendances nécessaires.

3.1.3 Définition d'un utilisateur

La première chose que j'ai dû effectuer a été la mise en place de l'accès à la page de gestion des applications web. Pour cela, j'ai effectué une modification dans le fichier *tomcat-user.xml* afin de définir un utilisateur et un mot de passe ainsi que les rôles attribués à cet utilisateur. Une fois les rôles attribués, j'ai eu accès à la page de gestion de Tomcat.

3.1.4 Augmentation de la mémoire Java

Par la suite, j'ai créé un fichier *setenv.sh* afin de modifier les paramètres d'exécution de Tomcat pour les augmenter et qu'ils répondent à nos besoins.

Il est nécessaire par exemple d'augmenter la quantité de mémoire attribuée à Java, selon l'exemple suivant :

```
#Default Java options
if [ -z "$JAVA_OPTS" ]; then
    java_opts="-Djava.awt.headless=true -Xmx2048M"
fi
```

3.1.5 Utilisation de Tomcat sur les ports 80 ou 8080

Lors de la connexion au serveur via l'URL (Uniform Resource Locator) HTTP : la page était inaccessible. J'ai utilisé diverses commandes afin de me permettre de diagnostiquer le problème :

- « ps ax », elle m'a permis de voir si le processus Tomcat était bien en cour d'exécution,
- Par défaut Tomcat utilise le port 8080, j'ai vérifié que le port était ouvert avec la commande « lsof -i:8080 » qui m'a permis de voir si Tomcat était bien en écoute sur le port 8080 en localhost (sur la machine elle même), ce qui était le cas,
- En demandant de l'aide à l'administrateur réseau, j'ai utilisé la commande « nmap » qui permet de voir les ports ouverts, fermés ou filtrés depuis une machine externe. Ce qui m'a permis de voir que le port 8080 n'était pas accessible depuis une machine externe. Ce port devait donc sans doute être filtré par le routeur du site.

La solution était donc de :

- soit demander à l'administrateur système du site d'ouvrir le port 8080,
- soit de faire écouter Tomcat sur le port 80.

Pour faire écouter Tomcat sur le port 80, 2 solutions se présentent :

- Soit mettre en place une règle *iptables* qui permet de rediriger le trafic TCP (Transmission Control Protocol) du port 80 vers le port 8080,

```
iptables -t nat -A PREROUTING -p tcp --dport 80 -j REDIRECT --to-port 8080
```

- Soit faire écouter nativement Tomcat sur le port 80, ce qui nécessite la configuration du fichier de configuration de Tomcat : `/var/lib/tomcat8/conf/server.xml`.

```
<Connector port="80" protocol="HTTP/1.1"
    enableLookups="false"
    connectionTimeout="20000"
    redirectPort="8443" />
```

Cette modification n'est pas suffisante, car il faut autoriser Tomcat à utiliser des ports bas inférieurs à 1024. Pour cela il faut modifier le fichier `/etc/default/tomcat8`.

```
# If you run Tomcat on port numbers that are all higher than 1023, then you
# do not need authbind. It is used for binding Tomcat to lower port numbers.
# (yes/no, default : no)
AUTHBIND=yes
```

En modifiant la ligne AUTHBIND dans ce fichier de configuration de Tomcat ça permet à Tomcat d'écouter sur des ports bas inférieurs à 1024 car ce sont des ports consacrés aux serveurs. En enlevant cette restriction, j'ai obtenu un Tomcat opérationnel sur le port 80.

3.1.6 Taille de l'archive web déployable

La chose suivante à faire était d'augmenter la taille de l'archive web que je pouvais déployer via la page HTML. Pour cela, il m'a fallu modifier le fichier `/usr/share/tomcat8-admin/manager/WEB-INF/web.xml` et modifier la taille maximal qui était de 50MB que j'ai défini dans un premier temps sur 200MB pour Geoserver. J'ai dû la redéfinir, par la suite, sur 300MB pour Géonetwork.

```
<multipart-config>
  <!-- 300MB max -->
  <max-file-size>314572800</max-file-size>
  <max-request-size>314572800</max-request-size>
  <file-size-threshold>0</file-size-threshold>
</multipart-config>
```

3.1.7 Passage en HTTPS

Un des objectifs du stage était de passer tous les serveurs en HTTPS (HyperText Transfer Protocol Secure).

Pour ce faire, j'ai eu besoin d'obtenir des certificats qui permettent :

- De certifier que le site web accédé est bien celui qu'il prétend être,
- De chiffrer le trafic HTTP entre le client et le serveur web.

Mon tuteur m'a fourni les certificats émis par l'autorité de certification TERENA (Trans-European Research and Education Networking Association), association qui à pour but de promouvoir et participer au développement d'une infrastructure internationale d'information et de la télécommunications au profit de la recherche et de l'éducation et peut délivrer des certificats SSL (Secure Sockets Layer), que j'ai installés dans Tomcat, et qui ont permis de certifier le serveur et de chiffrer le trafic.

Mode opératoire

Pour configurer tomcat en HTTPS il faut deux fichiers du certificat `star_osupytheas.fr.crt` et `star_osupytheas_fr.key` qui vont servir à certifier toute machine dans le domaine `*.osupytheas.fr`.

Il faut utiliser la commande « `openssl` » et « `keytool` » avec lesquelles j'ai créés le fichier. `jks`, qui stocke les certificats et la clé privée. J'ai placé ce dernier dans le répertoire de Tomcat.

La commande « openssl » permet de créer le fichier au format .p12.

- openssl pkcs12 -export -name servercert -in star_osupytheas_fr.crt -inkey star_osupytheas_fr.key -out myp12keystore.p12

La commande « keytool » permet de créer un container .jks à partir du fichier .p12.

- keytool -importkeystore -srckeystore myp12keystore.p12 -destkeystore mykeystore.jks -deststoretype pkcs12 -alias servercert

Ensuite copier le conteneur mykeystore.jks dans le répertoire de Tomcat.

Une fois que le certificat est en place dans le Java Keystore, il faut indiquer à Tomcat d'utiliser le port HTTPS 443. Pour cela j'ai effectué une modification et un ajout au fichier `/var/lib/tomcat8/conf/serveur.xml` de Tomcat afin de rajouter un connecteur correspondant au port 443 qui est le port HTTPS.

<Connector

```
protocol="org.apache.coyote.http11.Http11NioProtocol"
port="443" maxThreads="200"
scheme="https" secure="true" SSLEnabled="true"
acceptCount="10" debug="0"
keystoreFile="/opt/tomcat/conf/mykeystore.jks" keystorePass="mypasswd" >
clientAuth="false" sslProtocol="TLS"
```

/>

Ce connecteur indique le port à utiliser (port), l'activation du SSL (SSLEnable), l'emplacement du conteneur .jks (keystoreFile) et le mot de passe du conteneur (keystorePass).

3.1.8 Utilisation de Tomcat

The screenshot shows the Tomcat Manager web interface. At the top, there is a navigation bar with links for 'Gestionnaire', 'Lister les applications', 'Aide HTML Gestionnaire', 'Aide Gestionnaire', and 'Etat du serveur'. Below this is a table titled 'Applications' with columns for 'Chemin', 'Version', 'Nom d'affichage', 'Fonctionnelle', 'Sessions', and 'Commandes'. The table lists four applications: '/', 'geonetwork', 'host-manager', and 'manager'. Each application row includes buttons for 'Démarrer', 'Arrêter', 'Recharger', and 'Retirer', along with an 'Expirer les sessions' button and a timer for inactive sessions. Below the table is a 'Deployer' section with input fields for 'Chemin de contexte (requis)', 'URL du fichier XML de configuration', and 'URL vers WAR ou répertoire', and a 'Deployer' button. At the bottom, there is a 'Fichier WAR à déployer' section with a 'Choisir un fichier' button and a 'Deployer' button.

Figure 9 : Interface de gestion des applications web Tomcat

Déploiement de l'archive web

Il y a deux possibilités afin de gérer Tomcat :

- Soit par des lignes de commande,
- Soit par l'interface graphique (Figure 9).

Avec l'interface graphique, il suffit d'importer l'archive web .war et de la déployer. L'indication *true* ou *false* permet de savoir si l'application web s'est bien déployée. L'interface graphique de Tomcat permet également le redémarrage des applications web.

En ligne de commande il suffit de copier, à l'aide de la commande « cp », l'archive web dans le répertoire `/var/lib/tomcat8/webapps`.

3.2 Serveur cartographique Geoserver

Dans le but de mettre en place et de tester Geoserver (Figure 10), j'ai créé deux containers virtuels. L'un servant de production et l'autre servant de développement.



Figure 10 : Logo de Geoserver

3.2.1 Description

Geoserver (Figure 10) est un serveur open source, pour le partage de données géo spatialisées, de données cartographiques conçu pour l'interopérabilité, il va publier des données provenant de n'importe quelle source de données spatiales à l'aide de normes ouvertes, aux travers de divers protocoles décrits dans des spécifications maintenues par l'OGC.

L'OGC est un consortium international pour développer et promouvoir des standards ouverts afin de garantir l'interopérabilité des contenus, des services d'échanges dans les domaines de la géomatique et de l'information géographique.

Geoserver intègre plusieurs protocoles dont : le WMS (Web Map Service), Web Coverage Service (WCS), Web Feature Service (WFS).

Le protocole WMS est un protocole de communication standard qui permet d'obtenir des données géoréférencées à partir de différents serveurs de données (Geoserver dans notre cas). Cela permet de mettre en place un réseau de serveurs cartographiques à partir desquels des clients WMS, tels que Geonetwork, puissent afficher les données sur des cartes interactives.

Geoserver gère des couches de données provenant de deux types de modèles :

- Le modèle Raster qui associe une entité descriptive à une localisation. C'est un mode maillé fondé sur un quadrillage régulier du terrain (matrice de point). On associe à chaque point une information, le vide étant également codé.
- Le modèle Vecteur qui associe à une localisation à une entité descriptive. Le terrain est représenté par des entités géographiques (points, lignes et polygones).

Ces modèles raster ou vecteur sont utilisés dans divers formats de fichiers tels que le :

- NetCDF (Network Common Data Form), `.nc` qui est un format de données auto-documenté qui permet la création, l'accès, et le partage de données scientifiques stockées sous la forme de grille ou de tableau,
- Shapefile qui est composé de différents fichiers : `.shp` décrivant la géométrie des objets décrits, `.dbf` qui contient les données attributaires relatives aux objets contenus dans le `.shp`, `.shx` qui stocke l'index de la géométrie.
- Geotiff,
- BD PostGIS,
- BD H2.

Dans notre cas, nous avons utilisé uniquement les formats NetCDF et Shapefile.

3.2.2 Installation de Geoserver

Pour installer Geoserver, il faut télécharger l'archive geoserver.war sur le site officiel (<http://geoserver.org/download/>) et la déployer dans Tomcat.

Pour des raisons de sécurité, il est nécessaire de redéfinir le mot de passe de l'utilisateur administrateur.

3.2.3 Sécurité CORS

Avec le passage en HTTPS de la machine, Geoserver ne parvenait plus à partager ses couches via le protocole WMS.

Le problème venait du mécanisme de sécurité CORS (Cross-Origin Resource Sharing) qui bloque certaines requêtes non autorisées au serveur.

En effet, les sites web restreignent les scripts JavaScript permettant d'accéder à des données provenant d'un autre site web.

Sur la machine Geoserver, j'ai donc mis un filtre, permettant d'autoriser toutes les connexions afin de pouvoir faire passer le protocole WMS, dans le fichier *etc/tomcat8/web.xml*.

```
<filter>
  <filter-name>CorsFilter</filter-name>
  <filter-class>org.apache.catalina.filters.CorsFilter</filter-class>
  <init-param>
    <param-name>cors.allowed.origins</param-name>
    <param-value>*</param-value>
  </init-param>
  <init-param>
    <param-name>cors.allowed.methods</param-name>
    <param-value>GET,POST,HEAD,OPTIONS,PUT</param-value>
  </init-param>
  <init-param>
    <param-name>cors.exposed.headers</param-name>
    <param-value>Access-Control-Allow-Origin,Access-Control-Allow-Methods</param-value>
  </init-param>
</filter>
<filter-mapping>
  <filter-name>CorsFilter</filter-name>
  <url-pattern>/*</url-pattern>
</filter-mapping>
```

J'ai autorisé les méthodes nécessaires au protocole WMS, *GET*, *POST*, *HEAD*, *OPTIONS*, *PUT* et indiqué les en-têtes qui peuvent être affichés *Access-Control-Allow-Origin*, *Access-control-Allow-Methods*. Le paramètre */** autorise toutes les adresses à passer ce filtre.

3.2.4 Ajout d'extension

Dans Geoserver, le format Shapefile est pris en charge par défaut contrairement au NetCDF. J'ai dû installer une extension pour gérer le format NetCDF que j'ai téléchargé au préalable sur le site officiel de Geoserver (<http://geoserver.org/release/stable/>).

Afin d'installer une extension, il faut effectuer la copie de tous les fichiers *.jar*, qui sont présents dans le fichier téléchargé, dans le répertoire */var/lib/tomcat8/webapps/geoserver/WEB-INF/lib* qui contient toutes les extensions de Geoserver.

Il faut par la suite redémarrer Geoserver afin que l'extension soit prise en compte.

3.2.5 Mise à jour/Changement de serveur

Lors de la mise en place d'un serveur, il est important de tester les futures mise à jour/changement de serveur pour ne pas risquer de perdre les données.

Pour cela, il m'a fallu :

- Localiser le répertoire qui contient des données de Geoserver, qui se trouve dans `/var/lib/tomcat8/webapps/geoserver/data`,
- Déplacer ce dossier dans un répertoire extérieur aux applicatifs Tomcat et Geoserver, comme par exemple `/opt` en gardant les droits d'accès. De cette manière si on met à jour le paquet Tomcat ou l'archive Geoserver, les données seront préservées car elles sont mises à l'extérieur des répertoires de l'application,
- Répercuter cette modification dans le fichier `/var/lib/tomcat8/webapps/geoserver/WEB-INF/web.xml`, afin d'indiquer le nouvel emplacement du répertoire des données en remplaçant,

```
<!--  
  <context-param>  
    <param-name>GEOSERVER_DATA_DIR</param-name>  
    <param-value>...</param-value>  
  </context-param>
```

```
-->
```

- par

```
<context-param>  
  <param-name>GEOSERVER_DATA_DIR</param-name>  
  <param-value>/opt/geoserver/data</param-value>  
</context-param>
```
- Effectuer un redémarrage de Geoserver pour que la modification soit prise en compte.

Dans le cas d'un changement de serveur, il a suffit de prendre le dossier contenant les données de Geoserver, le mettre sur le nouveau serveur, et d'indiquer le chemin du dossier comme dans l'explication précédente. Et d'effectuer un redémarrage de Geoserver.

3.2.6 Utilisation de Geoserver

Geoserver met à disposition un grand nombre de possibilités et de réglages afin de publier une couche de données. 3 concepts de base sont présents dans Geoserver :

- Les espaces de travail,
- Les entrepôts,
- Les couches.

Gestion des données de Geoserver

Pour publier une couche de données, il faut :

- Créer un espace de travail qui va permettre de classer les couches de données par projet,
- Créer un entrepôt de données, permet d'importer (uploader) les données à partir des fichiers (shapefile, netcdf et autres, ...), ou de se connecter à une source de données telle qu'une base de données postGIS,
- Publication de la couche avec divers paramètres à configurer.

Outre le nom de la couche, description, etc, il y a deux paramètres à ne pas négliger lors de la configuration de la couche :

- Le SRC (Système de Référence de Coordonnées) (Figure 11), qui est le référentiel dans lequel on peut représenter des éléments dans l'espace et permet de se situer sur l'ensemble du globe terrestre grâce à un couple de coordonnées géographiques,

Systèmes de Référence de Coordonnées (SRC)

SRC natif
 [EPSG:WGS 84...](#)

SRC des données
 [EPSG:WGS 84...](#)

Gestion des SRC

Emprises

Emprise native

Minimum en X	Minimum en Y	Maximum en X	Maximum en Y
164.6068333333	-21.0048333333	164.6586666667	-20.9373333333

[Basées sur les données](#)
[Compute from SRS bounds](#)

Emprise géographique

Minimum en X	Minimum en Y	Maximum en X	Maximum en Y
164.6068333333	-21.0048333333	164.6586666667	-20.9373333333

[Calculées sur les emprises natives](#)

Figure 11 : Configuration du SRC et de l'emprise géographique sur Geoserver

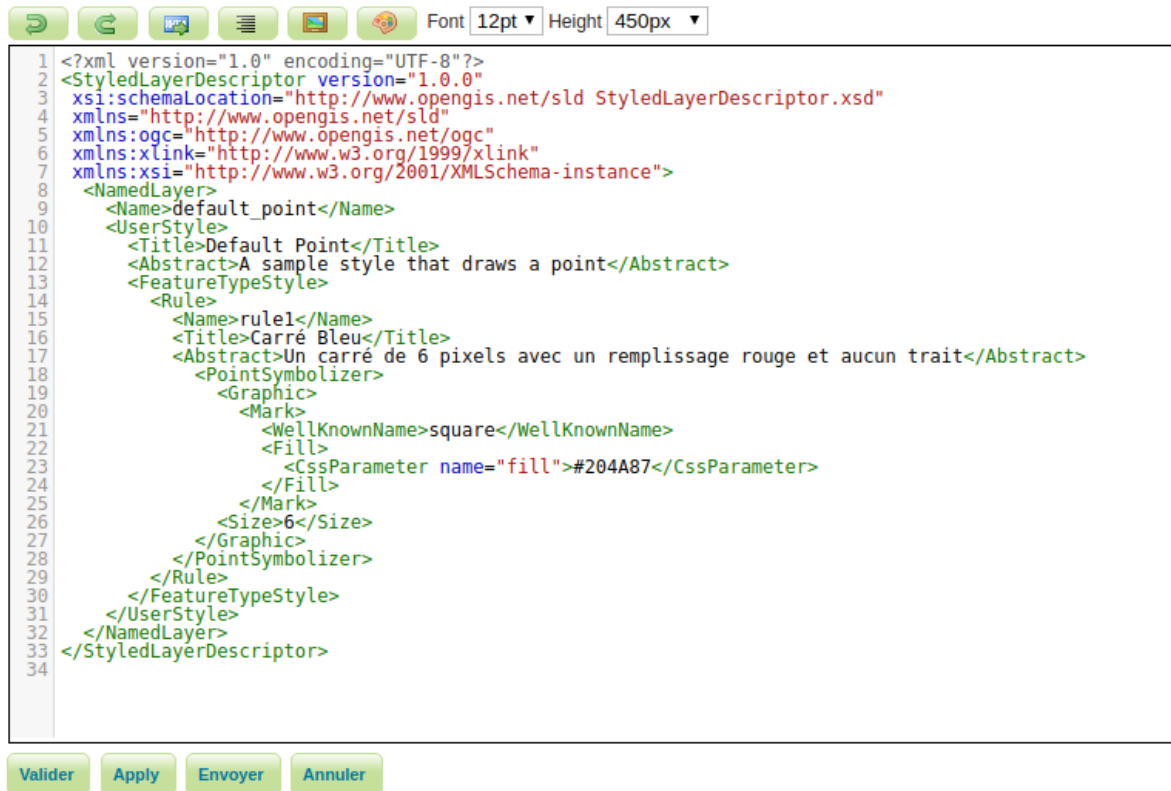
- L'emprise géographique (Figure 11) : c'est la région à afficher encadrée par ces 4 coordonnées LatO/LatE/LonS/LonN,
- Le style d'affichage de la couche est codé au format SLD (Styled Layer Descriptor) qui est un fichier XML utilisé par les systèmes d'information géographique tel que le WMS.

3.2.7 Création de style

Style de base pour shapefile en SLD

Une fois téléchargée dans Geoserver, on peut définir des styles différents pour représenter les points de la couche.

Pour différencier les couches lors de l’affichage, j’ai créé 2 styles (Figure 12) aux formats SLD.



```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <StyledLayerDescriptor version="1.0.0"
3   xsi:schemaLocation="http://www.opengis.net/sld StyledLayerDescriptor.xsd"
4   xmlns="http://www.opengis.net/sld"
5   xmlns:ogc="http://www.opengis.net/ogc"
6   xmlns:xlink="http://www.w3.org/1999/xlink"
7   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
8   <NamedLayer>
9     <Name>default_point</Name>
10    <UserStyle>
11      <Title>Default Point</Title>
12      <Abstract>A sample style that draws a point</Abstract>
13      <FeatureTypeStyle>
14        <Rule>
15          <Name>rule1</Name>
16          <Title>Carré Bleu</Title>
17          <Abstract>Un carré de 6 pixels avec un remplissage rouge et aucun trait</Abstract>
18          <PointSymbolizer>
19            <Graphic>
20              <Mark>
21                <WellKnownName>square</WellKnownName>
22                <Fill>
23                  <CssParameter name="fill">#204A87</CssParameter>
24                </Fill>
25              </Mark>
26              <Size>6</Size>
27            </Graphic>
28          </PointSymbolizer>
29        </Rule>
30      </FeatureTypeStyle>
31    </UserStyle>
32  </NamedLayer>
33 </StyledLayerDescriptor>
34
```

Valider Apply Envoyer Annuler

Figure 12 : Style SLD affichant un carré bleu

Ce style affiche une couche de données avec un carré bleu et mon second style affiche la couche de donnée en carré rouge.

Style de base pour modèle raster en CSS

Dans le cas d’un fichier raster où nous voulons afficher toutes les couleurs de points représentés dans un système de coordonnées Lat/Long, il nous faut définir une plage de couleur entre les minimums et maximums des valeurs des points. Pour ce faire il faut créer un nouveau style.

Format : CSS

```
* {
  raster-channels : auto;
  raster-color-map :
  color-map-entry(#43465C,-1)
  color-map-entry(#176CB9,-0.6)
  color-map-entry(#400913,1);
}
```

Dans cet exemple, la gamme de données s’étale entre -1 et 1 en passant par 0.6 et nous affectons des couleurs à cette gamme de valeur.

La balise “color-map-entry” définit la couleur avec le #RRVVBB. L’argument suivant correspond à l’échelle des couleurs.

Ce CSS définit donc une légende de la couleur #43465C à la couleur #400913 en passant par #176CB9, pour une gamme de variation de données entre -1 et 1

En gros on associe une couleur aux 2 valeurs extrêmes des données.

3.2.8 Création d'une couche de données Shapefile

Afin de publier des couches de données, j'ai créé des fichiers Shapefiles à partir d'un fichier au format *.xlsx* qui est une extension de fichier pour tableur Excel.

J'ai été amené à tester le logiciel Geoserver avec de vrais fichiers de campagne océanographique de divers prélèvements dans un lagon de Nouvelle Calédonie. La campagne s'intitule UECOCOT (Usines Économie Société et Contamination des eaux côtières) fournie par un chercheur du M.I.O. Les données étaient dans des fichiers Excel que j'ai transformés au format CSV puis Shapefile.

Pour ce faire dans QGIS (Figure 13) il faut utiliser la fonctionnalité « *Ajouter une couche de texte délimité* » qui permet de charger un fichier CSV et de le transformer en fichier Shapefile.

Pour cela il m'a fallu :

- Modifier le fichier Excel afin que les cellules ne comportent que les bons caractères (certains caractères ont posé problème lors de l'affichage dans Geonetwork),
- Exporter au format CSV qui est pris en charge par QGIS,
- Importer dans QGIS en indiquant les coordonnées de Longitude/Latitude.

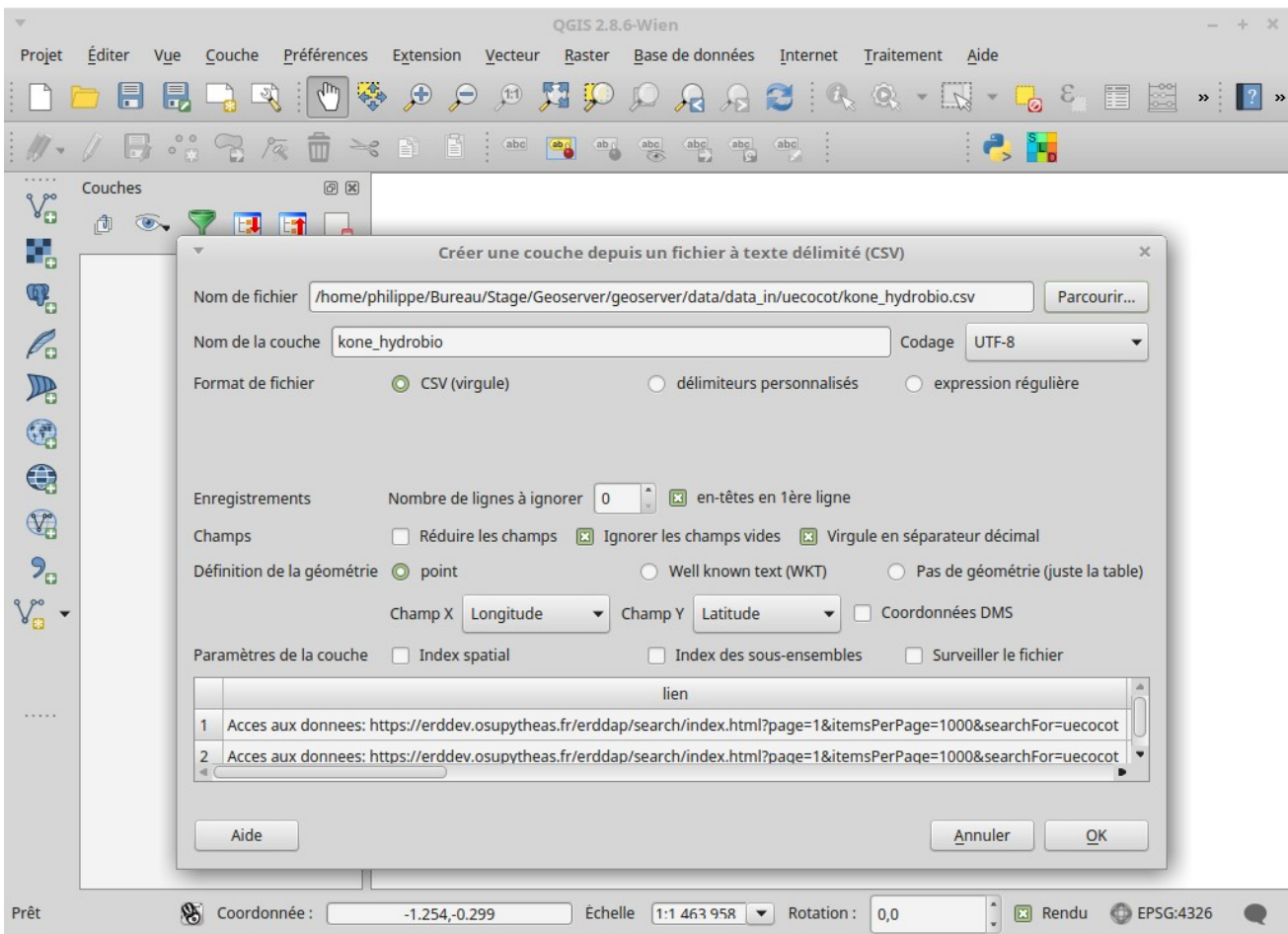


Figure 13 : Interface de QGIS
<https://www.qgis.org/fr/site/>

Une fois que le fichier Shapefile est créé nous pouvons le charger dans Geoserver, tel que cela a été défini plus haut, pour en créer une couche de données, que nous pouvons voir apparaître (Figure 14) et visualiser (Figure 15).

Type	Titre	Nom	Formats usuels	Tous les formats
icos-20181217		icos:OHP_MTO20181217	OpenLayers KML GML	Choisir une couche
icos-20181218		icos:OHP_MTO20181218	OpenLayers KML GML	Choisir une couche
romarin:julio-fev2015		romarin:julio-fev2015	OpenLayers KML GML	Choisir une couche
romarin:julio-jan2015		romarin:julio-jan2015	OpenLayers KML GML	Choisir une couche
Données Hydrobiologie		uecocot:hydrobio	OpenLayers KML GML	Choisir une couche
Données Mesures colonne d'eau		uecocot:mesures_col_eau	OpenLayers KML GML	Choisir une couche
World rectangle		tiger:giant_polygon	OpenLayers KML GML	Choisir une couche

date	lagon	cart	station	Latitude	Longitude	heure_loc	Profondeur	cubitainer
17/11/2008	Veh	3		-20.9668333333	164.6306666667	08:40	14.0	7

Figure 14 et 15 : Onglet «Prévisualisation de la couche» et visualisation de la couche <https://geoserver.osupytheas.fr/geoserver/web/wicket/bookmarkable/org.geoserver.web.demo.MapPreviewPage?1>

Nous voyons ci dessus les points apparaître dans un repère orthonormé, il nous reste à les voir sur un fond de carte, c'est ce qui va être effectué avec le logiciel GeoNetwork.

3.3 Catalogue de données Geonetwork



Figure 15 : Logo de Geonetwork

3.3.1 Description

Geonetwork (Figure 15) permet de cataloguer les jeux de données en fournissant les métadonnées normalisées ISO19139 et des liens d'accès aux données. Il utilise le protocole CSW (Catalog Service for the Web) défini par l'OGC. Il fournit de puissantes fonctions d'édition et de recherche de métadonnées grâce à l'utilisation de thésaurus ainsi qu'un visualiseur de carte web interactif.

La norme ISO19139 permet de définir explicitement et précisément comment les métadonnées d'un jeu de données doivent être implémentées. Elle définit le schéma requis pour décrire des informations géographiques et des services au moyen de métadonnées.

Les métadonnées sont des données décrivant une ressource (jeu de données, carte ou document cartographique, ...).

Il s'agit de toutes les informations que l'on peut recueillir et mettre à disposition pour décrire une ressource :

- Information d'identification : intitulé, description, dates de référence, version, résumé, intervenants, description de l'étendue géographique de la ressource, des aperçus, des contraintes légales et de sécurité attachées à la ressource;
- Information de représentation spatiale : régit la géolocalisation et l'organisation des données;
- Informations de qualité : précision géométrique, temporelle, exhaustivité, description des sources et processus appliqués aux sources;
- Références temporelles;
- Généalogie et méthodologie.

3.3.2 Installation de Geonetwork

Pour installer Geonetwork, il faut télécharger l'archive geonetwork.war sur le site officiel de Geonetwork (<https://www.geonetwork-opensource.org/downloads.html>) et la déployer dans Tomcat tel que vu précédemment pour Geoserver : c'est-à-dire soit via l'interface graphique, soit en déposant le .war dans le répertoire `/var/lib/tomcat8/webapps`.

Pour des raisons de sécurité, j'ai redéfini le mot de passe de l'utilisateur administrateur.

3.3.3 Ajout d'un thésaurus

Un thésaurus est une liste organisée de termes et de mots clés contrôlés et normalisés représentant les concepts d'un domaine de connaissance. C'est un langage contrôlé utilisé pour l'indexation de documents et la recherche de ressources documentaires dans des applications informatiques spécialisées.

J'ai ajouté le thésaurus INSPIRE-theme, qui est le thésaurus établi par la directive Inspire élaborée par la Direction générale de l'environnement de la Commission européenne qui vise à établir en Europe une infrastructure de données géographiques pour assurer l'interopérabilité entre les bases de données et faciliter la diffusion, la disponibilité, l'utilisation et la réutilisation de l'information géographique en Europe.

3.3.4 Sauvegarde/Mise à jour/Changement de serveur

Lors de son premier démarrage, Geonetwork crée une base de données H2. Toutes les données gérées par Geonetwork hormis les thésaurus sont stockées dans cette base de données.

Nous avons voulu nous prémunir de pertes de données lors des futures mises à jour de Geonetwork. Pour cela j'ai fait un script bash qui permet d'effectuer la sauvegarde de ces données.

Pour cela, j'ai dû :

- Localiser la base de donnée H2 qui est composé de 3 fichiers `gn.h2.db`, `gn.trace.db`, `gn.trace.db.old` qui se trouve dans `/var/lib/tomcat8/`,
- Localiser le répertoire contenant les thésaurus qui se trouve dans `/var/lib/tomcat8/webapps/geonetwork/WEB-INF/data/config/codelist/external/thesauri/`.

Le script bash, que j'ai fait, a pour but d'effectuer une sauvegarde journalière de toutes les données de Geonetwork, en effectuant les tâches suivantes :

- Création d'un dossier de sauvegarde à la date du jour s'il n'est pas déjà existant,

```
if [ ! -d "$DEST1" ];then
    mkdir $DEST1
fi
echo " sauvegarde de geonetwork dans $DEST1 "
```

- Copie des données de sauvegarde si elle n'a pas encore été effectuée,


```
date=`date +%Y-%m-%d`
DEST2=$DEST1/SAUV-GN-$date
if [ ! -d "$DEST2" ]; then
    mkdir $DEST2
    cp -a $GN $DEST2
    cp -a -r $THE $DEST2
else
    echo copie deja faite pour $DEST2
fi
```
- Suppression du dossier de sauvegarde ayant plus de 10 jours s'il est existant.


```
dateold=`date --date='10 days ago' +%Y-%m-%d`
echo "Destruction des repertoires de +10jours"
repold="SAUV-GN-$dateold"
if [ -d $repold ]; then
    echo "destruction de $repold"
    rm -fr $repold
else
    echo repertoire $repold non existant
fi
```

J'ai automatisé ce script pour qu'il s'exécute tout les jours à 1h du matin, grâce à l'utilisation de la « crontab » qui est un programme qui permet aux utilisateurs des systèmes Unix d'exécuter automatiquement des scripts, des commandes ou des logiciels à une date et une heure spécifiées ou un cycle défini à l'avance.

3.3.5 Utilisation de Geonetwork

Geonetwork permet de créer des fiches (Figure 16) de métadonnées normalisées.

Création d'une fiche de métadonnées

Lors de la création d'une fiche, Geonetwork demande à renseigner un certains nombres d'informations correspondants aux métadonnées normalisées ISO19139 (résumé, titre, auteur, dates, localisation, etc.) :

- Les métadonnées,
- Des liens d'accès aux données (ERDDAP, Geoserver, Geoportail),
- Des images.

UECOCOT Usines, économie, société et contamination des eaux côtières : lagon de Koné

Le projet interdisciplinaire UECOCOT vise à développer des outils d'aide à la gestion durable des activités minières afin de permettre la meilleure cohabitation possible entre dynamique naturelle des environnements (et leur capacité de résistance aux dommages et de résilience) et les activités humaines (industrielles ou non). L'objectif global du projet est de répondre à la question " comment adapter , à un coût socio-économique acceptable, les activités minières pour que leur impact reste compatible avec la durabilité des écosystèmes (approche mult Dans le Pacific implanté une r Cr, Co, Mn) iss

Téléchargements et liens

<p>hydrobio Données Hydrobiologie Cette donnée est publiée dans le service de visualisation (WMS) disponible à l'adresse https://geoserver.osupytheas.fr/geoserver/uecocot/wms, couche hydrobio.</p>	Visualiser
<p>mesures_col_eau Données Mesures colonne d'eau Cette donnée est publiée dans le service de visualisation (WMS) disponible à l'adresse https://geoserver.osupytheas.fr/geoserver/uecocot/wms, couche mesures_col_eau.</p>	Visualiser

À propos de cette ressource

Thèmes INSPIRE

Catégories

Étendue temporelle
Date de publication
2019-04-01

Période
Mon Jan 01 2018 00:00:00 GMT+0100
05 2018 00:00:00 GMT+0100

Fourni par

Mis à jour :
il y a 2 minutes

Partager

Figure 16 : Fiche Geonetwork du projet UECOCOT

<https://dataset.osupytheas.fr/geonetwork/srv/fre/catalog.search?jsessionid=A3EB382027BAD2F08C18340D7622C23C#/metadata/7443cf2a-f986-4e2d-b532-79a88ca6ae44>

Lien entre Geonetwork et Geoserver

Geonetwork peut se comporter comme un client WMS vis-à-vis du serveur Geoserver.

Le protocole WMS va permettre à Geonetwork d'obtenir des données géoréférencées et de les afficher sur un fond de carte.

Geonetwork va permettre de superposer différentes couches de données de Geoserver sur le fond de carte de Geonetwork. Ceci va permettre de créer une carte interactive en affichant les données sur celle-ci.

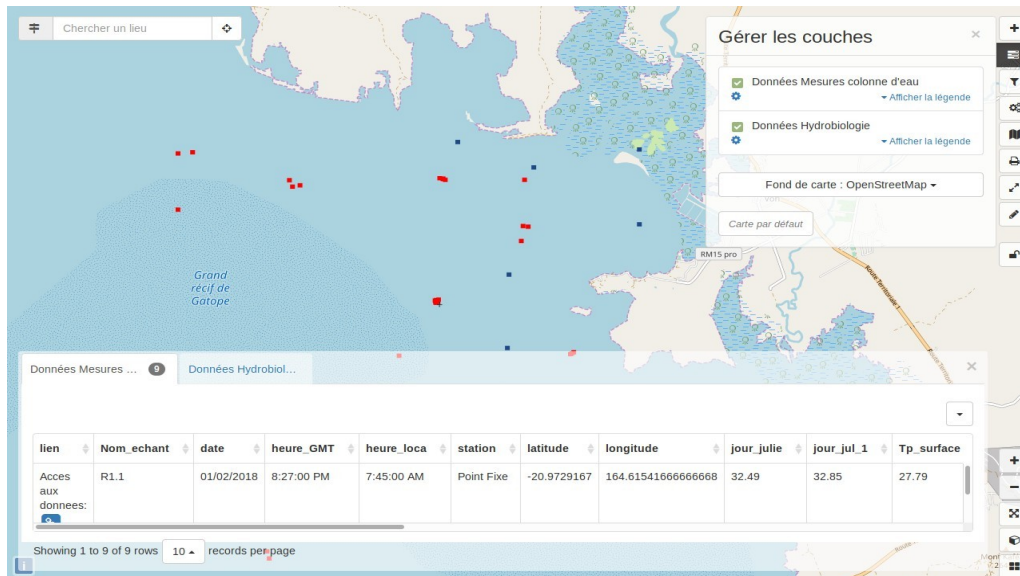


Figure 17 : Affichage d'une couches de données de Geoserver sur un fond de carte Geonetwork

4 Conclusion

L'objectif de mon stage était de mettre en place différents outils logiciels permettant la mise en place d'un portail d'accès aux données environnementales de l'OSU.

Pour cela, j'ai dû me pencher sur l'utilisation d'un nouveau serveur de données cartographiques. J'ai testé et installé un serveur Geoserver qui a permis à l'OSU de pouvoir gérer des données de terrain géo spatialisées.

J'ai dû me pencher sur les divers formats de fichiers gérés par Geoserver (principalement Shapefile et NetCDF) afin de pouvoir créer des couches de visualisation sur de vrais fichiers de données de terrain.

Par la suite j'ai fait le lien entre le serveur de catalogage Geonetwork et le serveur de données cartographiques Geoserver. Grâce au protocole WMS, Geonetwork se comporte comme un client WMS et obtient des couches de données qu'il peut afficher sur un fond de carte. De cette manière, les données de terrain de l'OSU sont disponibles et correctement cataloguées, affichées et référencées (FAIR).

L'installation de ces deux serveurs m'a également permis de découvrir et de savoir utiliser le serveur HTTP Tomcat. J'ai été confronté à de vrais problèmes d'accès aux ports TCP dans un environnement professionnel.

Le stage m'a beaucoup apporté en terme de réflexion et m'a permis de prendre plus de recul sur la façon d'aborder un problème. Il m'a également permis de compléter mes connaissances acquises dans l'enseignement à l'I.U.T.

Ce stage ayant fait appel à des compétences de fonctions systèmes, il entre dans la continuité de mes études, et renforce ma conviction et ma motivation de devenir administrateur systèmes et réseaux.

5 Remerciements

Je tiens à remercier toutes les personnes m'ayant aidé durant mon stage et qui, de ce fait, ont contribué à son succès.

Tout d'abord, j'adresse mes remerciements à mon professeur Monsieur Arnaud FÉVRIER qui m'a aidé à trouver ce stage en me fournissant le contact de Monsieur Maurice LIBES.

Je tiens à remercier mon tuteur de stage, Monsieur Maurice LIBES, Ingénieur en Système d'Information Géographiques au sein du laboratoire MIO, pour son accueil, le temps passé ensemble, le partage de son savoir faire et de ses méthodes de travail au quotidien qui m'ont permis d'approfondir mes connaissances et de les appliquer dans un réel contexte.

Je remercie également Monsieur Didier MALLARINO de m'avoir fourni le matériel nécessaire pour la réussite de mon stage.

6 Glossaire

OSU, Observatoire des Sciences et de l'Univers

UMR, Unité Mixte de Recherche

CEREGE, Centre Européen de Recherche et d'Enseignement des Géosciences de l'Environnement

IMBE, Institut Méditerranéen de Biodiversité et d'Écologie marine et continentale

LAM, Laboratoire d'Astrophysique de Marseille

LPED, Laboratoire Population Environnement Développement

MIO, Institut Méditerranéen d'Océanologie

CNRS, Centre National de la Recherche Scientifique

IRD, Institut de Recherche pour le Développement

FAIR, Facile à trouver Accessible Interopérable Réutilisable

ERDDAP, Environmental Research Division Data Access Protocol

WMS, Web Map Service

OGC, Open Geospatial Consortium

HTTP, Hypertext Transfer Protocol

KVM, Kernel-based Virtual Machine

OS, Operating System

LXC, Linux Container

VM, Machine Virtuelle

VMID, Virtual Machine Identifier

SSH, Secure Shell

XML, Extensible Markup Language

URL, Uniform Resource Locator

TCP, Transport Control Protocol

HTTPS, Hypertext Transfer Protocol Secure

TERENA, Trans-European Research and Education Networking Association

SSL, Secure Socket Layer

NetCDF, Network Common Data Form

CORS, Cross-Origin Resource Sharing

SRC, Systèmes de Référence des Coordonnées

CSW, Catalogue Service for the Web

7 Sitographie

Documentation Ubuntu [En ligne]. Consulté durant tous le stage. Disponible sur <https://doc.ubuntu-fr.org/>

Geoserver [En ligne]. Consulté durant tout le stage. Disponible sur : <http://geoserver.org/>

Geonetwork [En ligne]. Consulté durant tout le stage. Disponible sur : <https://geonetwork-opensource.org/>

It-Connect [En ligne]. Consulté du 13 mai au 18 mai. Disponible sur <https://www.it-connect.fr/configurer-tomcat-avec-ssl/>

OpenClassrooms [En ligne]. Consulté du 27 mai au 31 mai. Disponible sur <https://openclassrooms.com/fr/courses/43538-reprenez-le-controle-a-laide-de-linux/41155-executer-un-programme-a-une-heure-differee>

Proxmox [Réseaux OSU]. Consulté durant tout le stage. Non disponible.

Vidéo Youtube [En ligne]. Consulté du 29 avril au 4 mai. Disponible sur <https://www.youtube.com/watch?v=pfwd8t5tmik>