

# Simplifier l'accès aux données d'observations de la Terre via les Jupyter Notebooks



**Dossier Technique**

**Guillaume Lenoir d'Espinasse**

**Janvier 2021**

Tuteur ATOS : **Vincent Saleh**

Tuteur AFPA : **Pascal Dangu**

Entreprise : **ATOS INTEGRATION**

Projet : **Mundi Web Services**

**CONFIDENTIEL INDUSTRIE**

## REMERCIEMENTS

La réalisation de cette alternance et de cette mission n'aurait pas été possible sans le soutien de :

- **Vincent SALEH** (Chef de projet), **Jean Philippe AUDOUIN** (Deputy Program Manager) et **Jean Baptiste LAVEDRINE** (Directeur de projet) qui m'ont accueilli avec bienveillance sur le projet Mundi et dans leur équipe et m'ont accompagné tout au long de la réalisation de mes missions.
- **Marielle DUCROCQ (Responsable d'agence)**, pour son soutien et ses démarches pour me trouver une mission riche et intéressante.
- **Lise ROMANS et Nathalie GRES**, responsables de la POEI chez ATOS, pour leurs présences durant toutes les étapes de ce parcours d'alternance.
- **Patricia GALLAIS, Pascal DANGU** et tous les formateurs AFPA pour la qualité de leurs enseignements pendant la période de formation mais aussi pour leur suivi et soutien durant la période en entreprise.
- Je tiens à remercier **l'ensemble des membres de l'équipe du projet Mundi** pour leur accueil et leur aide à la réalisation de mes missions.
- Enfin j'aurai également un petit mot de remerciements pour **mes collègues de la POEI AFPA/ATOS** pour la bonne humeur, la solidarité et l'esprit d'équipe instaurés durant tout ce parcours aussi bien à l'AFPA qu'au sein de l'entreprise ATOS.

## ABSTRACT

Mundi is a Data Information Access Service (DIAS) belonging to the Copernicus program led by the European Union. Mundi aims to provide with Earth Observation data and services to make these data easy to access. Mundi provides access to Sentinel family satellites (1,2,3,5P), Landsat 8 and different Copernicus services like the Copernicus Atmosphere Monitoring Service (CAMS) which measures the air quality all around the world.

With 12 Terabytes of data produced each day, Mundi is a cloud-based platform offering plenty of services. One of them is the ability to access easily to the Mundi data and processing them by using the Jupyter Notebook which is web interactive development application. Mundi team has developed a lot of programs in Python language allowing Mundi users to access different data source.

My mission was to implement some new programs dedicated to beginners with Earth Observation data and processing who do not have necessarily the coding knowledge. I have then created many Graphic User Interface applications using different Python libraries like ipywidgets, matplotlib, numpy, imageio... In details, I have answered the following needs :

- Access to Sentinel data like Sentinel 2 and 3 satellites
- Access to CAMS data
- Detect Water presence on Sentinel 2 images and analyse data.

Working on this project was a great experience for me. Earth Observation environment is really interesting and challenging and making these data accessible is a great mission took in charge by Mundi. I am proud to have taking part and played a role on this.

## SOMMAIRE

1.	Contexte .....	6
1.1	Présentation de l'entreprise.....	6
1.1.1	Le groupe Atos.....	6
1.1.2	Atos Toulouse .....	7
1.1.3	Agence Aérospatiale.....	8
1.2	Le projet Mundi .....	8
1.2.1	Contexte du projet .....	8
1.2.2	Objectifs du projet.....	10
1.2.3	Architecture du projet.....	11
1.2.4	Organisation de l'équipe et du projet .....	12
1.2.5	Processus de développement et de test pendant le sprint :.....	13
2.	Mes missions : .....	14
2.1	Refonte page offre du site internet :.....	14
2.1.1	Analyse des besoins :.....	14
2.1.2	Conception : .....	15
2.1.3	Tests et validation :.....	17
2.2	Projets de créations de Jupyter notebooks :.....	18
2.2.1	Définition des objectifs et architecture :.....	18
2.2.2	Procédure de développement spécifique au Jupyter Notebook : .....	20
2.2.3	Refonte du notebook CAMS :.....	21
2.2.4	Création du notebook Mundi Easy Search Sentinel 2 image : .....	25
2.2.5	Création du notebook Mundi Sentinel 3 search image :.....	29
2.2.6	Création du notebook de détection des eaux permanentes : .....	30
2.2.7	Définition nouvelle architecture de présentation des notebooks : .....	36
2.2.8	Tests et validations :.....	38
3.	Bilan de mes missions :.....	41
4.	Bilan personnel :.....	42

## INTRODUCTION

Au cours de mes 8 mois de présence dans l'entreprise Atos pour la réalisation de la partie entreprise de mon alternance, j'ai eu l'opportunité de travailler au sein du projet Mundi Web Services.

Mundi Web Services est un Data Information Access Service ou plus simplement appelé DIAS. Un DIAS est une entreprise ou un consortium d'entreprise qui a développé une plateforme sur le cloud permettant d'accéder aux données du programme Copernicus. Les DIAS sont financés par la commission européenne.

Mundi est donc issu d'un consortium dirigé par Atos qui regroupe tout un ensemble de sociétés spécialisées dans les données d'observations de la Terre. La plateforme lancée courant 2018 permet d'accéder à de nombreuses données mais notamment aux données des satellites de la famille Sentinel construits par l'ESA.

Mon rôle au sein de ce projet a consisté à développer une approche orientée client facilitant d'un côté la compréhension de l'offre Mundi et de l'autre l'accès aux données d'observation de la Terre.

J'ai notamment travaillé sur la refonte de la page d'offre du site internet mais également sur le développement de programmes sur les Jupyter Notebooks permettant de rendre l'accès aux données faciles et intuitives pour des non experts en données d'observation de la Terre.

Ce rapport présentera dans un premier temps l'entreprise Atos et l'activité au sein de l'agence toulousaine, le projet Mundi et les processus mis en place, puis abordera l'ensemble des activités auxquelles j'ai participé. Il se terminera enfin par un bilan de mes missions et un bilan plus personnel.

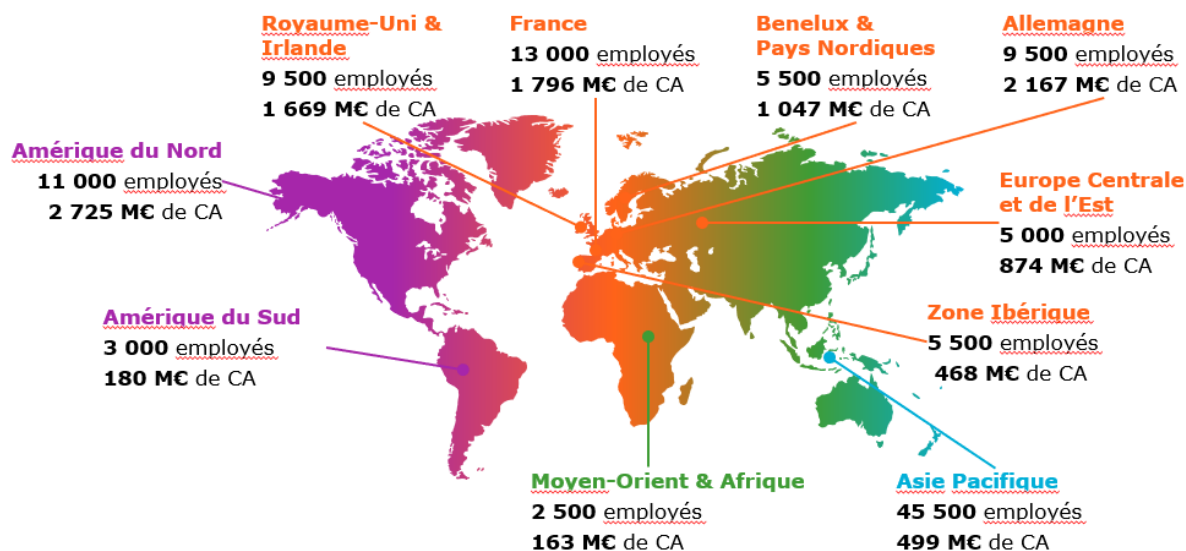
# 1. Contexte

## 1.1 Présentation de l'entreprise

### 1.1.1 Le groupe Atos

Atos est une Société Française spécialisée dans les services numériques, implantée au niveau mondial, avec environ 110 000 employés répartis dans 73 pays différents. Avec un chiffre d'affaire de 12 milliards d'euros en 2019, Atos fait partie des 5 entreprises leaders du numérique dans le monde et occupe la première place en Europe. Elle est dirigée par Elie Girard (Président Directeur Général depuis novembre 2019).

En France, elle est implantée à Aix, Angers, Bordeaux, Caen, Dijon, Grenoble, Lille, Marseille, Metz, Montpellier, Mulhouse, Nantes, Niort, Orléans, Rennes, Rouen, Sophia, Strasbourg, Toulon, Toulouse, Tours, mais aussi dans de nombreuses villes d'Ile de France et dans les DOM TOM.

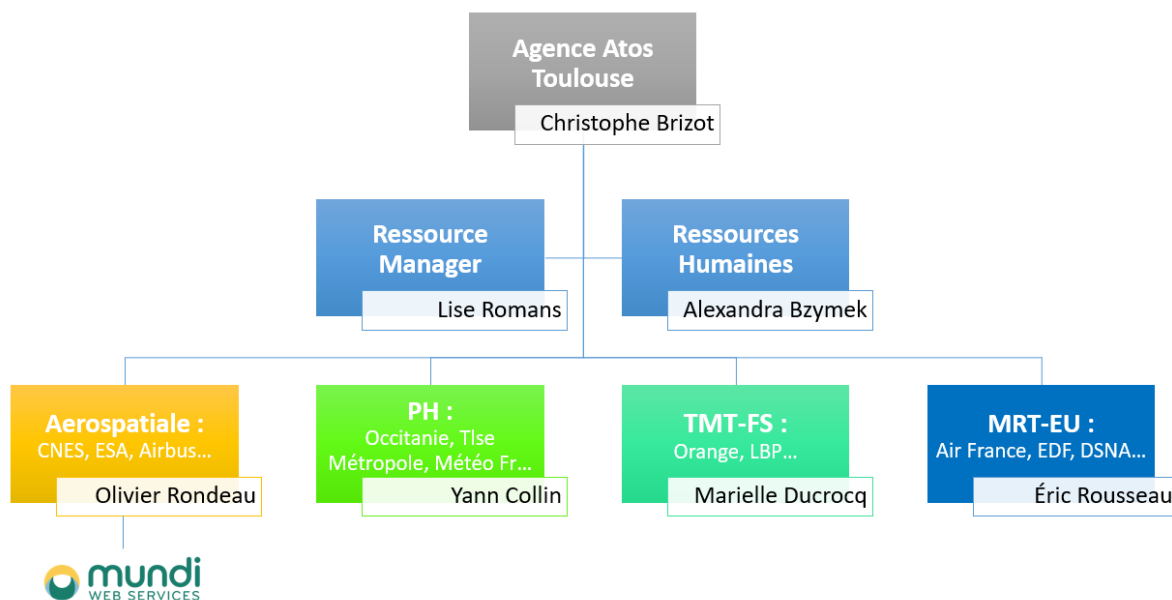


Les entreprises de services du numérique (ESN), aussi appelées sociétés de services en ingénierie informatique (SSII), sont des entreprises spécialisées dans la réalisation, la conception et la livraison de services et de projets informatiques répondant à des besoins venant d'entreprises et de clients externes. Atos est l'une des plus grandes ESN au niveau européen. Ses domaines d'activité de prédilection sont les services transactionnels, le cloud, la cyber sécurité, l'infogérance, le conseil en informatique et l'intégration de systèmes. Elle est depuis 1992 le partenaire numérique des jeux olympiques. En 2013, elle se classe seconde au rang du Palmarès des employeurs français.

### 1.1.2 Atos Toulouse

Le site de Toulouse se situe au nord-ouest de Toulouse dans la Z.I des Ramassiers, et regroupe plus de 600 salariés. Elle est dirigée par Christophe Brizot. L'activité est répartie entre quatre agences distinctes, définies par leurs clients et le type de l'activité qu'on y retrouve :

- Le premier de ces pôles est l'agence Aérospatiale, qui traite principalement avec Airbus et ses sous-traitants (Safran, Latecoere) au niveau aéronautique et avec le CNES, l'ESA, Thales au niveau spatial. Le projet Mundi appartient à ce pôle.
- L'agence PH a pour vocation de s'occuper des activités liées au secteur public, en particulier la santé. Elle traite avec des clients tels que la région Occitanie, Toulouse métropole, ou encore Météo France.
- L'agence MRT (*Manufacture Retail and Transportation*) traite des projets venant du monde industriel comme EDF, Air France, Pierre Fabre, ou la DSNA.
- Enfin l'agence TMT-FS se positionne sur les marchés des télécoms et de la finance. Elle travaille avec des clients tels que Orange, La Banque Postale, la BPCE.





## 1.1.3 Agence Aérospatiale

Atos Aerospace est une équipe de 160 personnes couvrant les métiers de l'aéronautique et du spatial. Les principaux clients de cette agence sont le CNES, l'ESA, Airbus ou encore Thales Alenia Space et les projets (forfaits, Assistance, Etude/R&T) concernent essentiellement les segments sols (centres de contrôles, centres de missions), la valorisation et exploitation de données, l'exploitation des missions, les logiciels de vol ou encore la simulation.

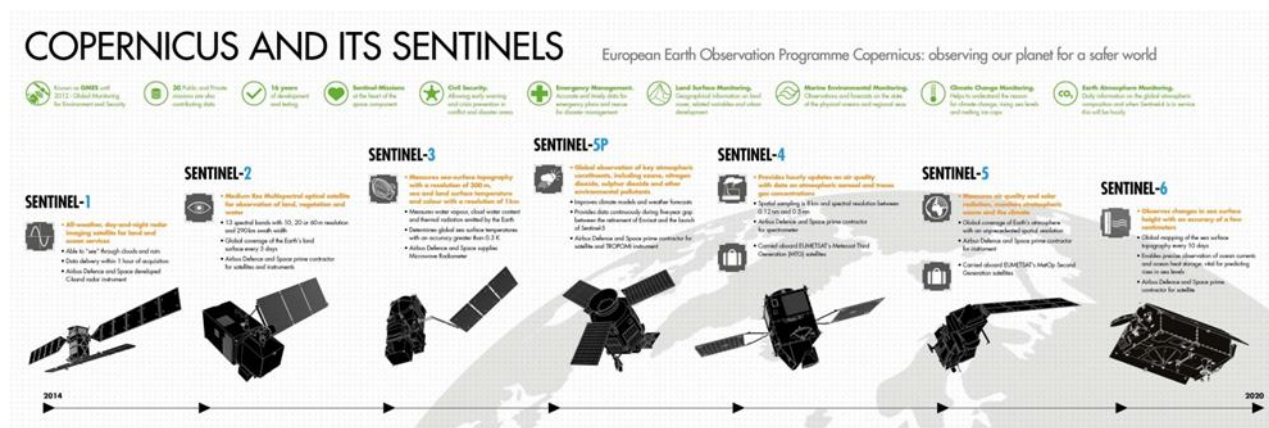
Le projet Mundi est lié à cette agence aérospatiale.

## 1.2 Le projet Mundi

### 1.2.1 Contexte du projet

Afin de répondre à la demande croissante de données d'observation de la terre, l'Agence Spatiale Européenne (ESA) a lancé en 2014 son premier satellite d'observation terrestre de la flotte Sentinel. Cette flotte, qui prévoit le lancement de plus de 20 satellites d'ici 2030, appartient au programme Copernicus soutenu et financé par la commission européenne.

Ces satellites sont répartis en plusieurs catégories (voir ci-dessous)



Ce programme a permis de générer une base de données d'images satellites très importante. La problématique rencontrée par l'ESA est que ces images mises en libre-service, sont des données brutes et donc difficiles à exploiter. Afin de répondre aux besoins des utilisateurs de données d'observation de la terre en données lisibles et facilement accessibles, la Commission Européenne (EC) a lancé une initiative afin de développer des Services d'Accès aux Informations et aux Données Copernicus (DIAS) pour faciliter l'accès aux données et aux



informations du programme pour les services concernés. Ce projet a pour but d'accélérer l'innovation et de permettre la création de nouveaux modèles économiques dans le secteur des données d'observation de la Terre.

Afin de répondre aux exigences de la Commission Européenne et de diversifier l'offre des DIAS, l'ESA a lancé un appel d'offre au niveau européen auquel Atos a répondu et a été retenu ainsi que 4 autres DIAS à travers l'Europe (Sobloo, Creodias, Onda, Wekeo).

Atos travaillait déjà sur un projet similaire depuis 2015 appelé SparkinData. Ce projet existant a donc servi de base pour répondre à l'appel d'offre de l'ESA et a constitué les fondations du projet Mundi qui a démarré en décembre 2017.

Pour répondre à cet appel d'offre, Atos s'est entouré d'entreprises reconnues dans le secteur de l'observation de la terre afin d'apporter savoir-faire et expérience qui lui ont permis d'être sélectionné.

Ce consortium se compose des entreprises suivantes (liste non exhaustive) :

- T-Systems qui fournit l'ensemble des services liés à l'offre Cloud
- Sinergise, GAF AG, EOX qui fournit des services liés à l'accessibilités des données ou à l'imagerie satellite.

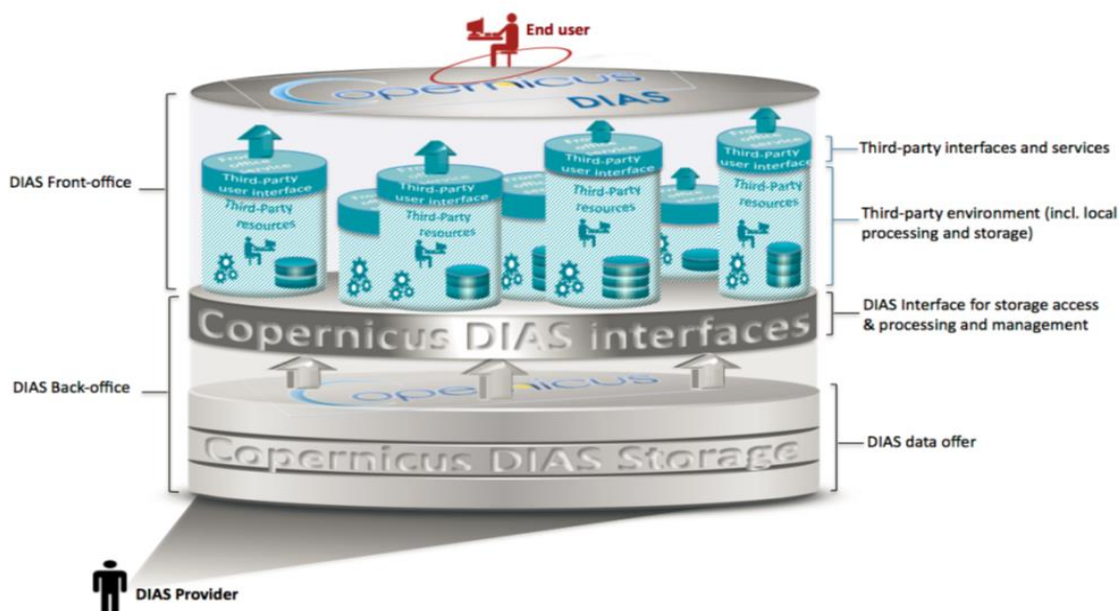
La solution Mundi s'articule autour de 4 piliers qui permettent de proposer une offre complète adressant l'ensemble du cycle de vie du client :



## 1.2.2 Objectifs du projet

Les objectifs du projet Mundi sont de répondre aux spécifications des DIAS imposées par la Commission Européenne et l'ESA :

- Les composants du DIAS sont le Back-Office, le DIAS Interface services et le Front-Office.
- Chaque back-office doit offrir un accès illimité, gratuit, complet et à jour aux données d'observation de la terre, et doit offrir également un environnement de stockage et de calcul évolutif dans des conditions commerciales concurrentielles.
- Chaque DIAS doit posséder une interface afin d'accéder aux données de la plateforme et de les gérer.
- Les fournisseurs du DIAS doivent permettre aux Third Party Users (3PU) d'accéder aux données ingérées sur la plateforme et de leur fournir une assistance. Ce terme 3PU vient de la relation triple entre le fournisseur du DIAS, l'ESA et l'utilisateur.

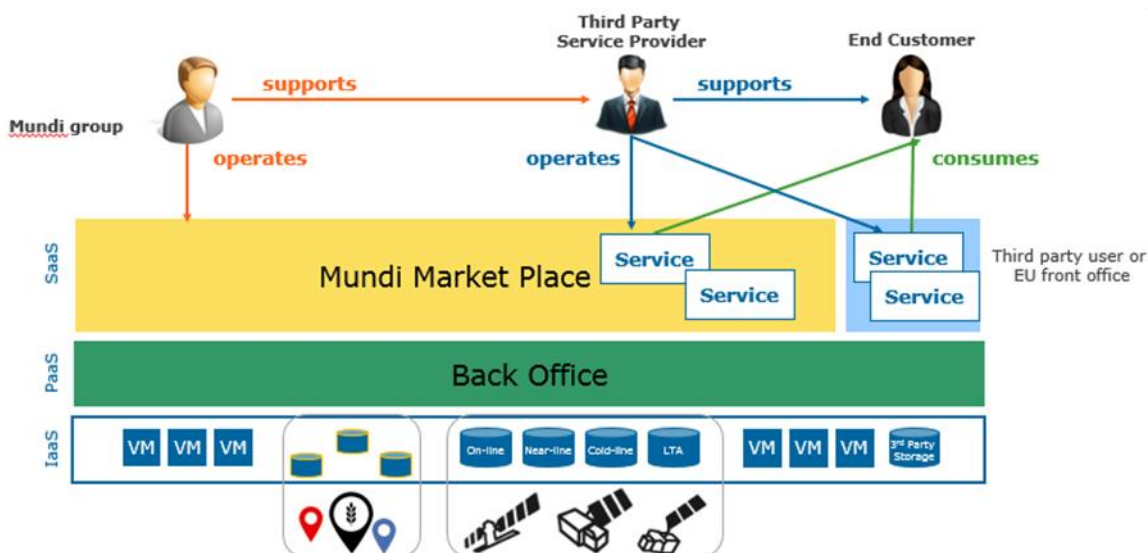


Comme le montre le schéma ci-dessus le projet prévoit que chaque DIAS télécharge les données du programme Copernicus et les intègre à sa plateforme. Cette phase d'ingestion des données doit continuer une fois toutes les données passées ingérées afin que les données quotidiennes soient présentes sur le DIAS et qu'elles soient toujours à jour.

Chaque DIAS doit posséder une interface afin d'accéder aux données de la plateforme et de les gérer. Cela permet de rendre les données disponibles pour les utilisateurs de la plateforme. L'objectif du DIAS est donc d'héberger les applications des 3PU et de leur permettre d'accéder aux données ingérées sur la plateforme.

### 1.2.3 Architecture du projet

Afin de répondre aux spécifications de DIAS identifiés par la commission Européenne, Atos a mis en place une architecture basée sur le Cloud :



**IaaS (Infrastructure as a Service)** est l'une des trois principales catégories de services de *Cloud Computing*, il s'agit de fournir l'infrastructure informatique, les machines virtuelles et d'autres ressources comme les serveurs de fichiers, le stockage des données, les équilibreurs de charge, les adresses IP, les réseaux locaux virtuels, etc...

**PaaS (Platform as a Service)**, comme son nom l'indique, on fournit la plateforme d'accès aux données.

**SaaS, (Software as a Service)** il s'agit de fournir l'accès aux services sur la MarketPlace (regroupement des services en lien avec le projet) de Mundi pour les Third Party Users, ils peuvent utiliser directement les services d'accès aux données satellites sans passer par des installations ou par des configurations.

#### Intégration continue :

Le projet Mundi se base sur le principe d'intégration continue, induite par des méthodes de management itératif/agile, et le besoin de redéployer l'intégralité de la plateforme à la fin de chaque sprint pour répercuter les modifications apportées. Cette intégration continue s'inscrit dans le mouvement **Devops** (développement logiciel et administration des infrastructures), dont le but est l'augmentation de la fréquence des déploiements et des livraisons continues, pour une meilleure atteinte des objectifs économiques des projets. 4 livraisons sont prévues chaque année.

### 1.2.4 Organisation de l'équipe et du projet

L'organisation de l'équipe projet découle de l'architecture du projet.

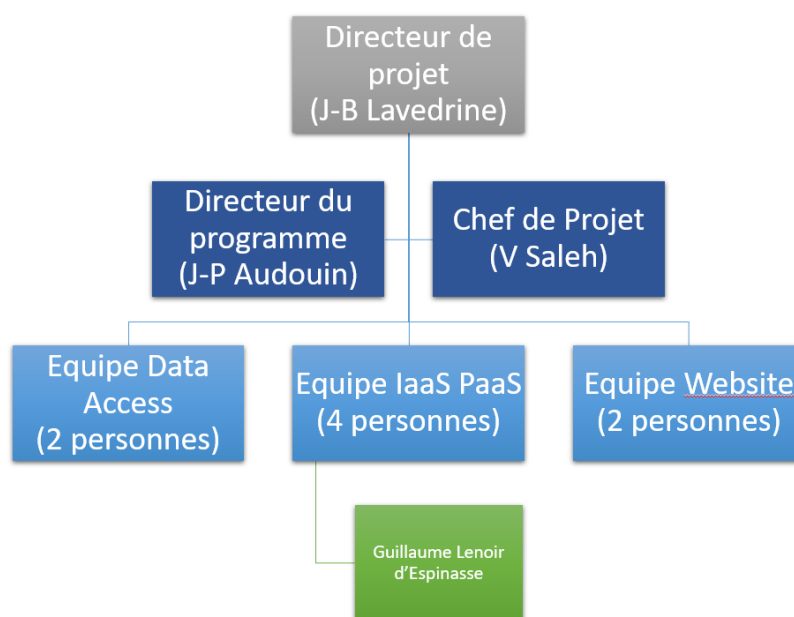
Ainsi on retrouve trois équipes :

- Une équipe dédiée au DAS (Data Access Services) et à l'ingestion des données dans le catalogue Mundi.
- Une équipe au service des couches IaaS, PaaS, SaaS, de la sécurité et du *Monitoring*.
- Une équipe dédiée au site Web ayant le rôle d'interface avec les utilisateurs finaux.

Le management est découpé en trois axes :

- Le management des partenaires et des utilisateurs de la plateforme.
- Le suivi du projet.
- Les aspects liés à la phase opérationnelle du projet.

Le marketing et les aspects commerciaux sont gérés par une équipe dédiée, et la direction du projet est assurée par deux personnes.

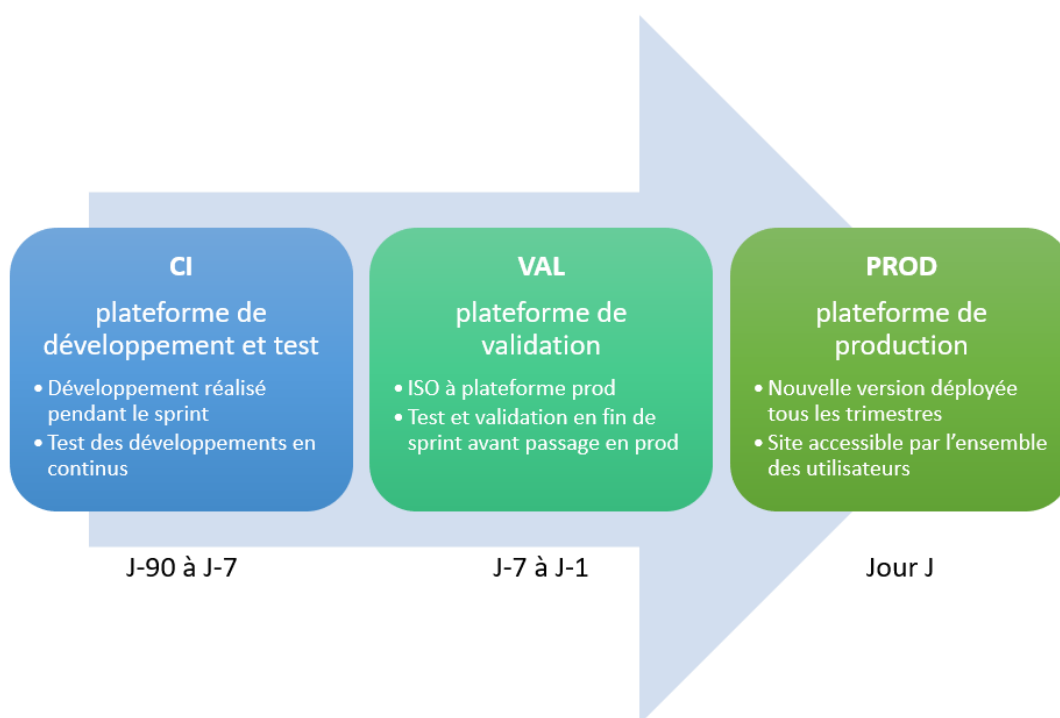


Le projet se base essentiellement sur la méthode agile/itérative dans le but d'une conception incrémentale du produit final et donc de pouvoir livrer progressivement des fonctionnalités du projet. L'avancement est suivi quotidiennement via un « daily meeting ».

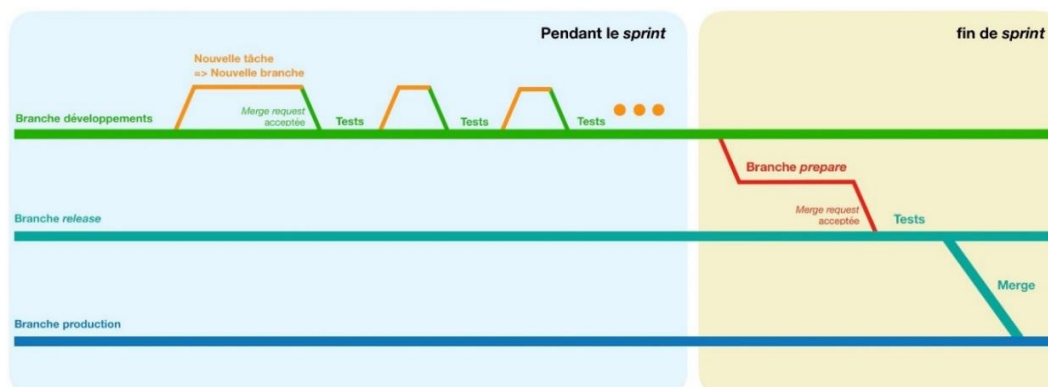
Des sprints d'une durée de 3 mois sont mis en place, en général communs aux 3 équipes et s'inscrivant dans le principe d'intégration continue. Une release est prévue à chaque début de saison.

### 1.2.5 Processus de développement et de test pendant le sprint :

Le processus de développement et de test au cours d'un sprint se déroule à travers 3 plateformes en ligne possédant chacune un rôle très précis comme on peut le voir sur le schéma ci-dessous :



Pour le versioning, l'outil GitLab est utilisé. On peut voir sur le schéma ci-dessous l'organisation des branches et des merges au cours du sprint. La branche développement correspond à la plateforme CI, la branche release à VAL et la branche production à PROD. A chaque merge un pipeline est automatiquement lancé déployant la nouvelle version du projet sur la plateforme correspondante.



## 2. Mes missions :

Au cours de ma période d'alternance au sein du projet Mundi, j'ai réalisé des missions différentes. Le point commun étant de rendre Mundi plus accessible aux nouveaux utilisateurs.

### 2.1 Refonte page offre du site internet :

La première mission qui m'a été confiée visait à proposer une nouvelle présentation de l'offre de Mundi sur le site internet [www.mundiwebservices.com](http://www.mundiwebservices.com) . L'objectif principal étant de rendre cette offre plus compréhensible et simple à lire.

#### 2.1.1 Analyse des besoins :

L'offre de Mundi est assez complexe à comprendre de prime abord étant donné la spécificité du secteur d'activité et de la richesse de la proposition de valeur mélangeant accès aux données, services cloud mais encore divers outils aidant à l'utilisation de ces données.

L'offre ayant été définie au début du projet en 2018, elle avait en partie évoluée, il était donc nécessaire d'apporter un nouveau regard dessus. De plus, nouvel arrivant sur le projet, possédant une longue expérience dans le domaine du marketing, j'ai pu apporter des idées et un regard neuf.

Le constat fut le suivant :

- Les infos n'étaient pas regroupées par thématiques d'offres mais par type d'informations (contenu, prix) ce qui générait des doublons et une difficulté de compréhension
- Il n'y avait quasiment pas de texte expliquant les différents éléments de l'offre.

En réponse à ce constat, j'ai proposé les solutions suivantes :

- Regrouper l'ensemble des informations disponibles par catégorie d'offre : noms, description, prix
- Conservation des éléments graphiques (logos offres)
- Renforcer identité de marque MUNDI en mettant en avant son nom devant chaque solution proposée. (Cf ex autres secteurs)
- Découpage de l'offre dans chaque rubrique entre ce qui est gratuit (FREE) et ce qui est payant (ADVANCED)

Au niveau des contraintes, y compris techniques on retrouvait les éléments suivants :

- Modifier uniquement le contenu de la page et non sa structure (pas de développement structurel) en utilisant le CMS Keystone et la technologie texte riche (HTML/CSS).
- S'inspirer d'une première ébauche réalisée en 2019
- Conserver la charte graphique Mundi et notamment les codes couleurs.

## 2.1.2 Conception :

Pour répondre à l'ensemble de ces éléments, j'ai commencé par définir une maquette permettant de montrer ce qu'on pourrait obtenir. Pour ce faire, j'ai utilisé le logiciel ADOBE XD (voir **annexe 1**).

Une fois cette maquette validée, j'ai commencé à travailler sur la construction de cette page. J'ai ainsi réalisé l'ensemble de mes développements sur la plateforme CI (cf. partie 1.2.5).

J'ai utilisé le CMS Keystone qui est utilisé pour la création de contenu sur le site sans avoir besoin de réaliser des modifications structurelles sur le code du site.

Les pages du site se basent sur des layouts prédéfinis, permettant de construire des briques en utilisant différentes possibilités : 1, 2 ou 3 colonnes en largeur, background color, images...

OFFER - 100 Data Offer - section 1

ID: 5f80078ef142de02ecd02636

Language

English

x

▼

Translates

Select...

▼

layout structure

1 Column

x

▼

title

Mundi Data Offer

Title color

dark green

x

▼

title position

Left

x

▼

subtitle

content

File

Edit

Insert

View

Format

Tools

B

I

U

☰

☰

☰

☰

🔗

🔄

🖼️

⌵

Mundi integrates a large range of Earth Observation and non EO data. The collection is regularly enriched to offer all material to build and operate your service. You can find free and licensed products according to your needs.

p

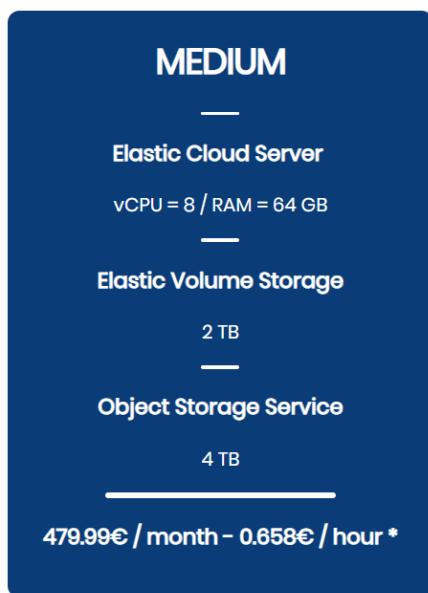


Cependant, l'utilisation du CMS Keystone ne permet pas de tout réaliser. Pour apporter une certaine flexibilité, un éditeur de texte riche a été ajouté. Ceci permet de réaliser des modifications dans le code HTML et CSS sans avoir besoin de toucher au code du site et avec un déploiement immédiat (au rechargement de la page).

J'ai ainsi réalisé plusieurs éléments à partir de ce texte riche notamment au niveau du style.

### 1<sup>er</sup> exemple :

Le premier exemple correspond à l'ajout de barres de soulignement dans les zones de texte comme on peut le voir sur l'exemple ci-dessous :



3 premières barres :

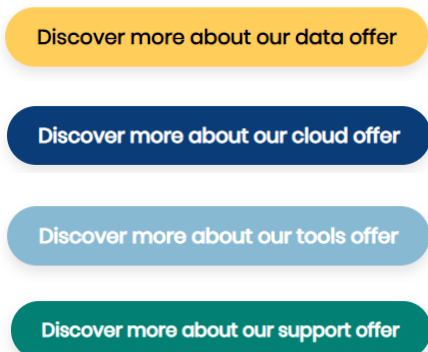
```
<div style="background-color: white; height: 3px; border-radius: 50px; margin-left: 45%; margin-right: 45%;"> </div>
```

Dernière barre:

```
<div style="height: 5px; background-color: white; border-radius: 50px; margin-left: 20%; margin-right: 20%;"> </div>
```

### 2ème exemple :

Le second exemple correspond à la création d'un bouton spécifique à cette page pour rediriger vers les pages correspondantes aux thématiques de l'offre abordée.



Code pour chaque bouton (juste le background color qui change) :

```
<div><a style="background-color: #ffcd5a; border-radius: 50px; padding: 12px 25px; text-decoration: none; color: black; font-weight: bold; box-shadow: 0 0.3125rem 0.625rem 0 rgba(150,150,150,0.25);" href="/data">Discover more about our data offer</a> </div>
```

D'autres éléments ont soit été modifiés ou créés grâce au texte riche. Je pense notamment à l'ajout de lien hypertexte, au travail sur les margin et padding des différentes DIV...

Ci-dessous, on retrouve le résultat final de la construction de la page avec la section dédiée à la partie DATA. Pour voir l'ensemble de la page, on peut se rendre directement sur le site à l'adresse suivante : [www.mundiwebservices.com/offer](http://www.mundiwebservices.com/offer) ou voir en **annexe 2**.

## MUNDI DATA OFFER



Mundi integrates a large range of Earth Observation and non EO data. The collection is regularly enriched to offer all material to build and operate your service. You can find free and licensed products according to your needs.

### FREE

Free access to full Copernicus Sentinel satellites data (1,2,3,5P), and also Landsat satellites :

- > Discover data by filtering time, geographic area and other parameters through GeoData tool or Mundi catalogue API
- > Visualise and download products through :
  - GeoData tool
  - REST API

### ADVANCED

- > As a user, access to specific data collection allowing applications in various domains such as: agriculture monitoring, climate change, land and urban monitoring.
- > As a data provider, add your own data collection to the Mundi offer, and monetize it.
- > Find all data collection in [the Marketplace!](#)

**Prices according to data collection**

[Discover more about our data offer](#)

## 2.1.3 Tests et validation :

Comme évoqué précédemment, l'énorme avantage du CMS + texte riche est que nous pouvons apercevoir le résultat et tester en direct en rechargeant la page. Cela permet d'aller très vite dans la construction de la page et d'avancer étape par étape après test et validation de chaque élément développé.

Par la suite, une validation a été réalisée par l'équipe de managers et quelques allers retours ont été effectués pour aboutir à un résultat convaincant pour tout le monde.

La page a par la suite selon le processus décrit dans la partie 1.2.5 été déployée sur la plateforme VAL, puis après validation en PROD.

Cette mission fut très intéressante et m'a permis à la fois de mieux comprendre l'offre Mundi et d'apporter mon regard neuf mais également de perfectionner mes compétences sur le front notamment en HTML/CSS.

## 2.2 Projets de créations de Jupyter notebooks :

À la suite de cette première mission de refonte de la page offre du site internet, j'ai eu l'opportunité de travailler sur un autre aspect important de la plateforme Mundi. Il s'agit des Jupyter Notebooks. Le Jupyter Notebook est une application web open source permettant de créer et de diffuser des programmes développés et exécutés directement depuis l'application en ligne.

Très utilisé dans le monde scientifique pour le traitement et l'analyse des données il permet de mélanger au sein d'un même document du code, du texte explicatif, des images... Plusieurs langages sont proposés

Les avantages sont nombreux, mais on retient principalement :

- La possibilité d'interagir en direct avec son code, de l'exécuter et le modifier directement si nécessaire.
- De documenter son code de manière très facile et de le rendre interactif
- De s'affranchir des problématiques d'installations des environnements de développement en utilisant une interface en ligne centralisée et où l'environnement de développement (incluant les librairies) est paramétrable et applicable à tous les développeurs d'une même équipe.

Parmi les multiples possibilités offertes par les Jupyter Notebooks, il est possible de réaliser des programmes ayant l'apparence d'un site internet via l'utilisation des balises HTML et de feuille de style CSS mais aussi et surtout par l'utilisation de la librairie Python ipywidgets dédiée au Jupyter Notebook.

### 2.2.1 Définition des objectifs et architecture :

Les Jupyter Notebooks sont souvent utilisés par les clients pour découvrir les possibilités offertes par Mundi. C'est en effet un outil gratuit qui permet d'accéder à un panel de programmes et d'informations idéal au début de l'utilisation de Mundi.

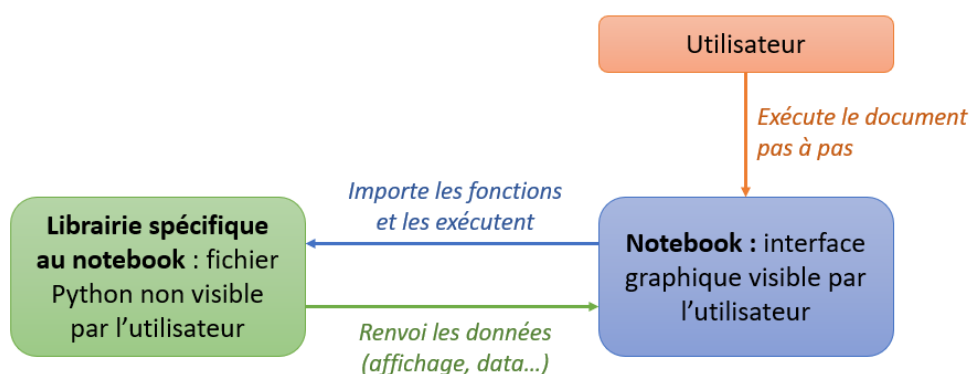
Nous avons constaté que les Jupyter Notebooks étaient utilisés par deux profils d'utilisateurs :

1. **Des profils débutants avec les données d'observations de la terre et non techniques :** ils souhaitent accéder simplement à la donnée sans avoir besoin de taper ou comprendre des lignes de code
2. **Des profils experts avec les données d'observations de la terre et techniques :** ils ont besoin d'information précises sur les possibilités offertes pour l'exploitation des données. Ils ont une expérience du développement et de Python et souhaitent donc pouvoir visualiser et récupérer des programmes pour l'adapter à leurs besoins.

Ma mission s'est concentrée sur la réponse au besoin du 1<sup>er</sup> profil à savoir les débutants. En effet l'offre de notebooks pour la seconde catégorie était déjà bien fournie.

L'objectif était le suivant : permettre aux utilisateurs d'accéder facilement aux données Mundi via les Jupyter Notebooks à travers des interfaces graphiques et sans code apparent.

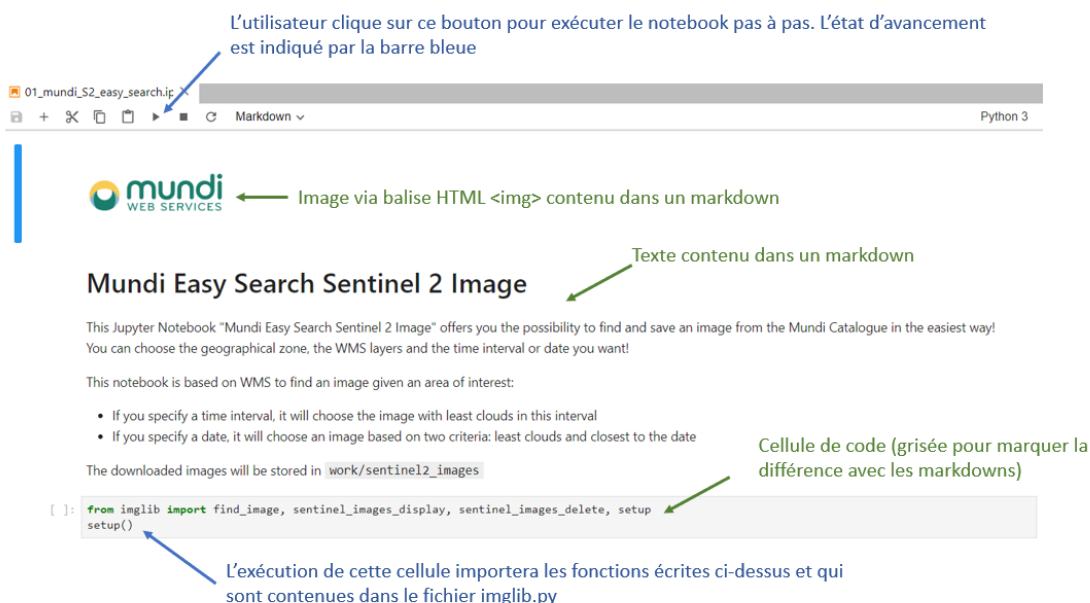
Cette mission pouvait soit consister à retravailler des programmes existants ou alors à en créer d'autres. Pour répondre au besoin de créer des notebooks avec une interface graphique, j'ai dû utiliser l'architecture de fonctionnement suivante :



Au sein du notebook, il existe deux possibilités de création de cellules :

- ✓ **Markdown** : qui contient du texte brut ou des balises HTML.
- ✓ **Code** : qui contient du code Python.

Sur le schéma ci-dessous on peut voir un exemple de notebook décrivant ces diverses possibilités :



L'utilisateur clique sur ce bouton pour exécuter le notebook pas à pas. L'état d'avancement est indiqué par la barre bleue

Image via balise HTML <img> contenu dans un markdown

Texte contenu dans un markdown

Cellule de code (grisée pour marquer la différence avec les markdowns)

L'exécution de cette cellule importera les fonctions écrites ci-dessus et qui sont contenues dans le fichier `imglib.py`

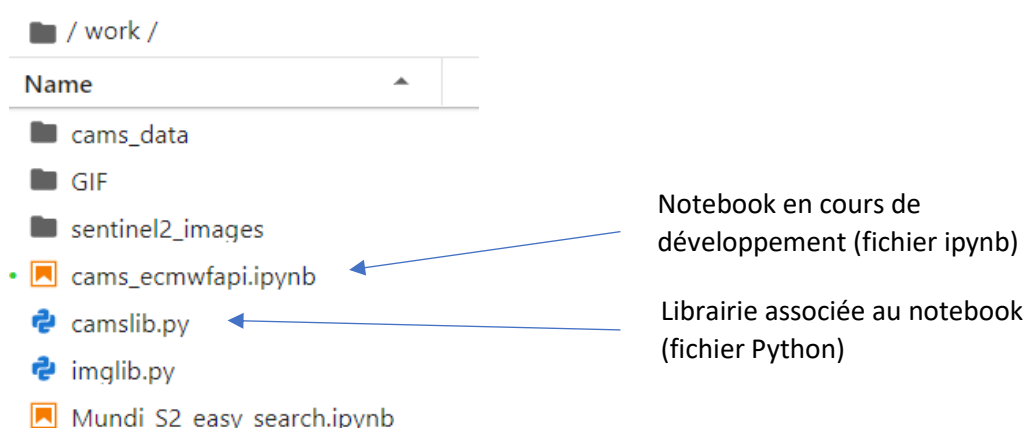
## 2.2.2 Procédure de développement spécifique au Jupyter Notebook :

Les Jupyter Notebook étant une interface de développement en ligne à part entière, la procédure de développement est un peu particulière par rapport aux projets classiques. L'idée étant dans un premier temps de travailler directement sur cette interface afin de développer le programme de manière itérative. Cela facilite également grandement les procédures de tests (cf. partie 2.2.8).

Dans le détail voici la procédure suivie :

### 1. Création du programme dans mon espace WORK :

Chaque utilisateur des Jupyter Notebook, qu'il soit client ou membre de l'équipe Mundi possède un espace Work. Celui-ci permet de stocker divers éléments comme des images, mais également des programmes notebooks et fichiers Pythons. On utilise donc cet espace pour développer les notebooks, comme on peut le voir sur l'image ci-dessous :



### 2. Ajout sur plateforme CI avec Pycharm :

Une fois la première étape réalisée, que le programme est testé et validé, on ajoute les deux fichiers dans le repository du projet sur Pycharm. On les ajoute dans les bons dossiers à savoir :

- Dossier Public pour le notebook
- Dossier Lib pour la librairie Python.

Enfin, il suffit ensuite de réaliser un « commit » des nouveautés et un « push » pour qu'un pipeline se lance et déploie automatiquement ces nouveaux fichiers sur la plateforme CI. Pour toutes modifications ultérieures on peut soit reproduire la même procédure ou pour des modifications légères, modifier directement les fichiers dans Pycharm.

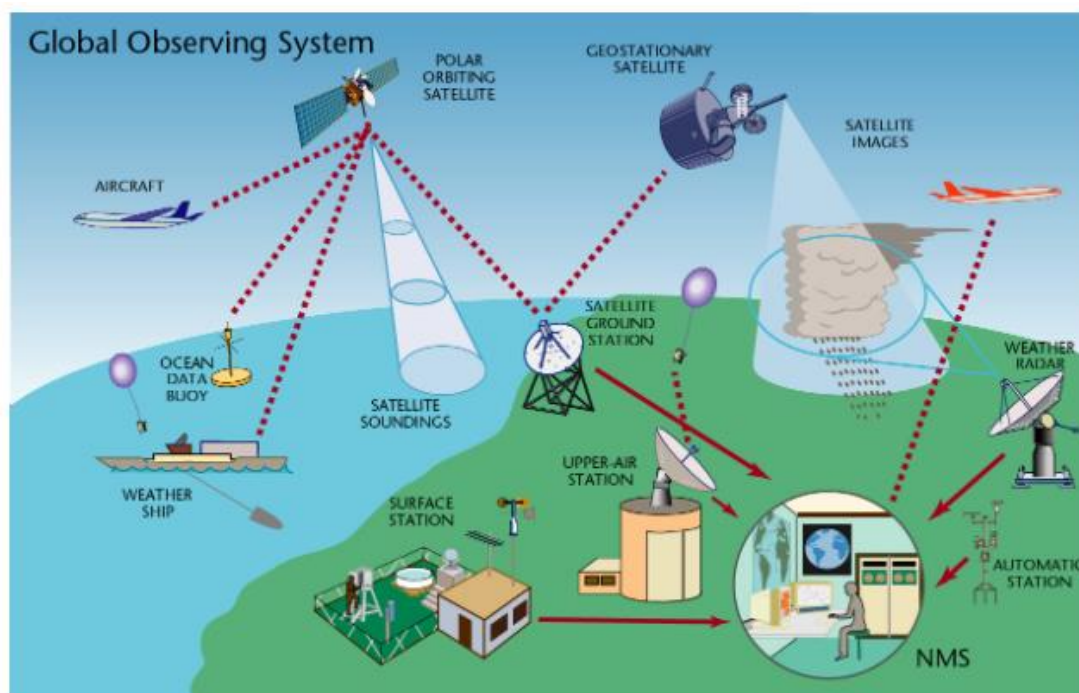
## 2.2.3 Refonte du notebook CAMS :

### a) Contexte et définition du besoin :

Le notebook CAMS permet d'accéder aux données du programme Copernicus Atmosphere Monitoring Service créé par l'organisme ECMWF qui est le centre européen pour les prévisions météorologiques à moyen terme. Celui-ci a créé un service qui permet d'accéder à de multiples données liées à l'analyse de la qualité de l'air partout dans le monde et par l'intermédiaire de plusieurs sources de données :

- Les satellites d'observations de la terre comme Sentinel 5P de l'ESA ou AURA de la NASA
- Les données in-situ présentes sur diverses localisations : stations météo, bateaux, avions...

Le schéma ci-dessous décrit l'ensemble des sources de données :



Pour permettre l'accès à ces données, ECMWF a créé une API qui permet de récupérer facilement les données à partir de requêtes.

Un premier notebook avait été créé au lancement de Mundi mais celui-ci était vraiment compliqué à manipuler et pas du tout adapté au profil débutant. En effet, le code Python était apparent, il manquait du texte explicatif des possibilités du service et de l'explication de son fonctionnement...

L'objectif fut donc de retravailler ce notebook afin de le rendre plus facile à comprendre et à utiliser.



**b) Conception :**

Pour la conception, je suis reparti du notebook existant et notamment des étapes pour la collecte et l'affichage des données.

**✓ 1<sup>ère</sup> étape :** Se connecter à l'API ECMWF :

L'accès à l'API nécessite de créer un compte sur ECMWF et de se connecter ensuite. Via un widget de la librairie ipywidgets, je demande les login/Key et crée la connexion.

```
[2]: configure_ecmwfapi()
```

Your Email

guillaume.lenoirdespinasse@atos

Your Key

b35e8df118b51640e3e896ee0937e37a

Validate

Your ECMWF login email is : guillaume.lenoirdespinasse@atos.net, and your ECMWF Key is : b35e8df118b51640e3e896ee0937e37a

**✓ 2<sup>ème</sup> étape :** Construire la requête :

Il s'agissait de la partie la plus complexe à réaliser. En effet, pour créer cette requête il est nécessaire de passer par une interface web de ECMWF où l'on coche les paramètres que l'on veut et on récupère par la suite la requête au format Python.

Ne souhaitant pas faire transiter l'utilisateur par cette interface (toujours dans un souci de simplicité) et afficher du code Python, j'ai décidé de demander les paramètres via un widget. L'ensemble des paramètres disponibles étant trop important j'ai réduit aux principaux à savoir : type de gaz (5 propositions), date de début, date de fin et nombre d'heures dans une journée :

```
[5]: request_ecmwfapi(output_filename)
```

Data selecti...

Nitrogen dioxide

▼

Starting Date

16 / 11 / 2020

📅

Final date

16 / 12 / 2020

📅

Select one or many hours below :

☐ 06:00:00

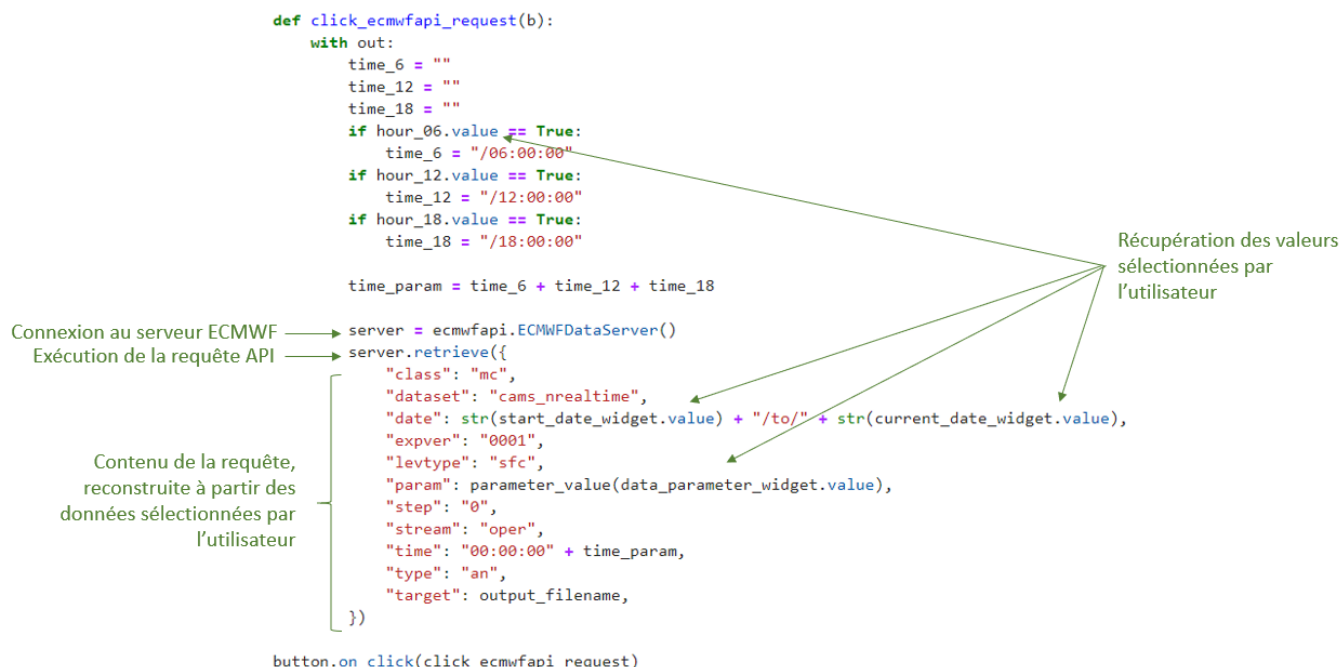
☐ 12:00:00

☐ 18:00:00

Validate



La complexité résidait dans la reconstruction de la requête à partir des informations renseignées par l'utilisateur. Comme on peut le voir sur l'image ci-dessous, il s'agissait de récupérer les valeurs, et de les intégrer à la requête de l'API ECMWF.



Après exécution de cette requête, l'ensemble des données collectées sont stockées dans un fichier output.grib enregistré dans un répertoire de l'espace « work » de l'utilisateur.

### ✓ 3<sup>ème</sup> étape : Définir les caractéristiques de l'affichage des données :

Pour exploiter les données contenues dans le fichier `output.grib` il est nécessaire de définir certains paramètres d'affichage. En effet ce fichier contient l'ensemble des données pour le monde entier, mais il est possible de spécifier la zone d'affichage.

Par l'exécution de la fonction `get_projection_map()`, il est possible de sélectionner un continent où l'on souhaite afficher les données. Pour y parvenir j'ai utilisé la librairie Python **Matplotlib** qui est très réputé, complète et utilisé dans le domaine scientifique. Elle permet de créer des visualisations à partir d'une source de données.

Dans cette librairie, pour l'affichage sur un continent, j'ai utilisé **Basemap** qui permet d'afficher les données en fonction des coordonnées de latitudes et longitudes d'une zone géographique donnée et pour le choix de la palette de couleur j'ai utilisé **Colormap** qui propose une multitude de palettes de couleurs.















Il est également possible de spécifier un titre à apposer sur chaque image téléchargée et un nom pour le fichier GIF qui sera généré à la fin.

Image montrant le widget généré par la fonction `get_projection_map()` avec le choix de la projection, de la palette de couleurs, du titre et nom du fichier :

```
[6]: get_projection_map(output_dir, output_filename)
```

Map Location

**Colormaps sample**

ocean	
gist_earth	
terrain	
gist_stern	
gnuplot	
gnuplot2	
CMRmap	
brg	
hsv	
jet	
nipy_spectral	
gist_ncar	
tab20b	
tab20c	

Map colors

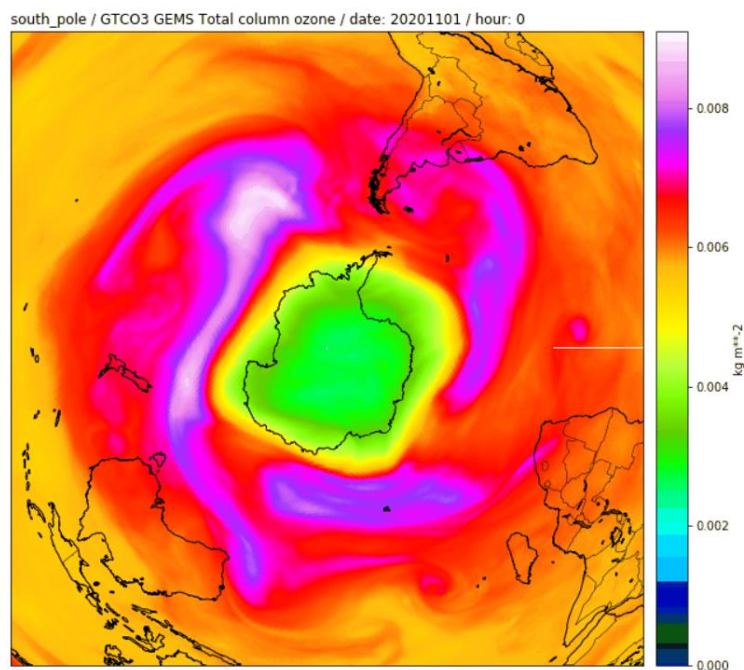
Display dire...

Title

GIF file name

✓ **4<sup>ème</sup> étape** : Télécharger et afficher les données :

Par la suite, un téléchargement d'images se lance. Une image est téléchargée par jour et heure demandé. Puis un GIF est automatiquement généré et stocké dans le répertoire « work » de l'utilisateur. Ci-dessous, un exemple d'une image générée montrant le trou de la couche d'ozone en Antarctique. Un tweet sur le compte Mundi a été publié à partir de ces données.



## 2.2.4 Création du notebook Mundi Easy Search Sentinel 2 image :

### a) Contexte et définition du besoin :

Parmi le programme Copernicus, Sentinel 2 (S2) est le satellite le plus « populaire » et le plus utilisé. Il permet d'obtenir des images avec une résolution de 10 mètres par 10 mètres ce qui est une belle avancée dans le domaine de l'imagerie spatiale. Cela permet d'observer notamment des villes ou des zones assez restreintes géographiquement. Le satellite passe à minima 1x tous les 7 jours au même endroit. Cela permet de suivre certaines évolutions (cf. notebook détection des eaux).

Ce deuxième projet consistait donc à réaliser un Notebook permettant d'accéder d'une manière très simplifiée aux données du satellite Sentinel 2. Pour schématiser, l'objectif était d'être capable de télécharger une image en moins de 1 minute à travers une interface graphique.

### b) Conception :

Pour concevoir ce notebook, il faut déjà comprendre comment récupérer une image satellite avec différents paramètres sachant que plusieurs Téraoctets d'images sont produits chaque jour. Mundi se base sur le protocole WMS pour Web Map Service qui est un protocole de demande d'image à partir de critères précis. Parmi les principaux, on retrouve :

- **Zone géographique précise** : coordonnées min et max en latitudes et longitudes
- **Le type de produit** : S2 possède deux types : L1C avec une réflectance au sommet de l'atmosphère, L2A au fond de l'atmosphère.
- **Le layer** : Un layer est une manière de retoucher l'image brute récupérée depuis le satellite. Ils sont gérés via une interface spécifique et ils sont développés en Javascript.
- **Date de début et de fin** : via le protocole WMS, le service va automatiquement rechercher la meilleure image dans l'intervalle de temps demandé. Meilleure image en imagerie spatiale dans le cadre S2 sous-entend celle où il y a le moins de nuages.
- **Dimension de l'image** : hauteur et largeur en pixel.

L'ensemble de ces critères sont intégrés à une URL qui va interroger le serveur et afficher une image en réponse.

Voici par exemple à quoi ressemble une requête pour les critères suivants : ville de Toulouse, L2C, Natural-color, entre le 01/12/2020 et le 31/12/2020, en 480x720 :

Protocole utilisé

Nom du Layer

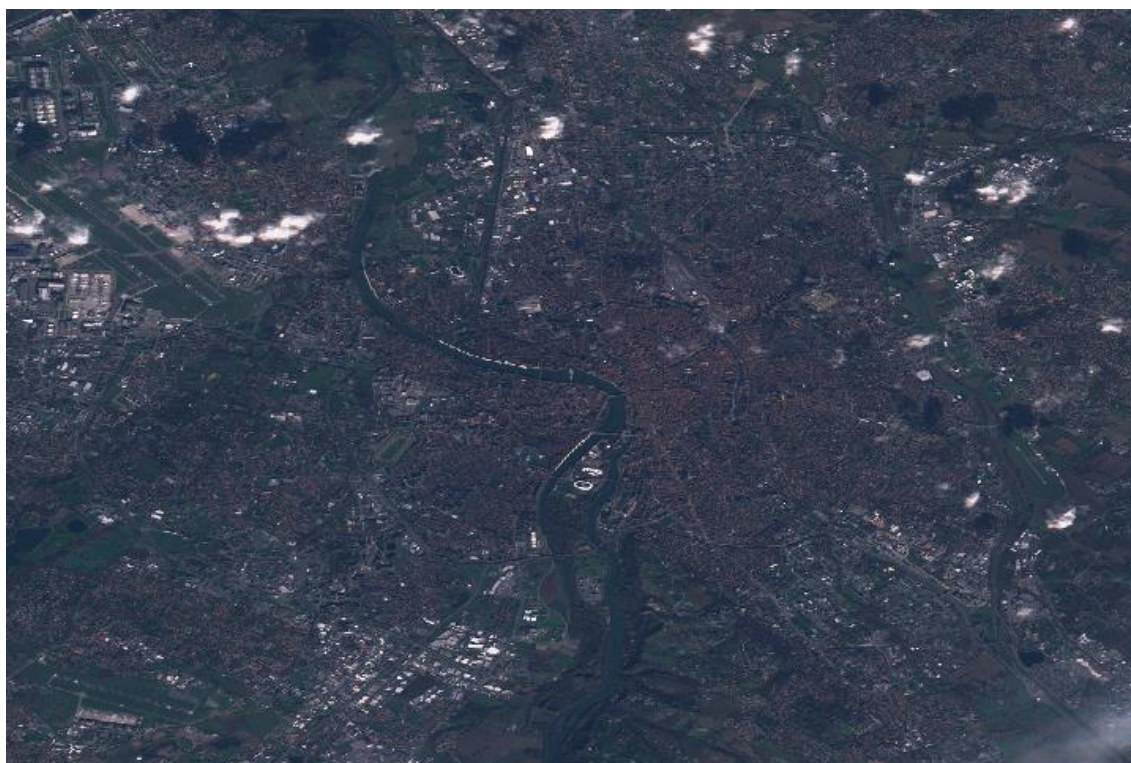
Période de temps

Largeur et Hauteur

Coordonnées géographiques (Bounding Box)

```
http://shservices.mundiwebservices.com/ogc/wms/d275ef59-3f26-4466-9a60-ff837e572144?SERVICE=WMS&REQUEST=GetMap&TRANSPARENT=true&LAYERS=1_NATURAL_COLOR&VERSION=1.1.1&FORMAT=image%2Fpng&STYLES=&showLogo=false&time=2020-12-01%2F2020-12-31&SRS=EPSG%3A4326&WIDTH=720&HEIGHT=480&BBOX=1.3503956%2C43.532654%2C1.5153795%2C43.668708
```

Et voici la réponse obtenue :



On peut remarquer que l'image est assez nette mais que malgré tout il subsiste quelques nuages. Ce qui veut dire que sur l'ensemble du mois de décembre 2020, aucune image prise par le satellite ne possédait pas de nuage. Il est possible de savoir la date exacte de la prise de l'image.

Pour répondre à tous ces besoins, l'équipe Mundi a créé une librairie spécifique à la recherche d'image en se basant sur tous les protocoles existants (WMS, WMTS, WCS, WFS) qui s'appelle mundilib.py et qui permet d'utiliser des méthodes de classe facilitant grandement la recherche d'image et d'informations associées comme la date.



C'est donc à partir de ces méthodes de classe que j'ai créé la librairie spécifique à ce notebook et qui s'appelle imglib.py. Voici les 3 fonctionnalités offertes par ce notebook :

✓ **1<sup>ère</sup> étape** : Rechercher une image :

La fonctionnalité principale de ce notebook est donc de pouvoir rechercher et afficher une image à partir d'une interface graphique.

J'ai donc construit une succession de widgets qui permettent de récolter l'ensemble des paramètres nécessaires à la création de la requête WMS (voir ci-dessous).

[2]: find\_image()

Cities

Bounding Box

Raw Bounding Box

Enter a city:

Validate

Toulouse, Haute-Garonne, Occitania, Metropolitan France, France  
BBOX = (1.5153795,43.668708,1.3503956,43.532654)

Collection

Validate

Layer

Validate

Best img nearest date

Best img in timedelta

Select your date :  
Final date

Title

File name

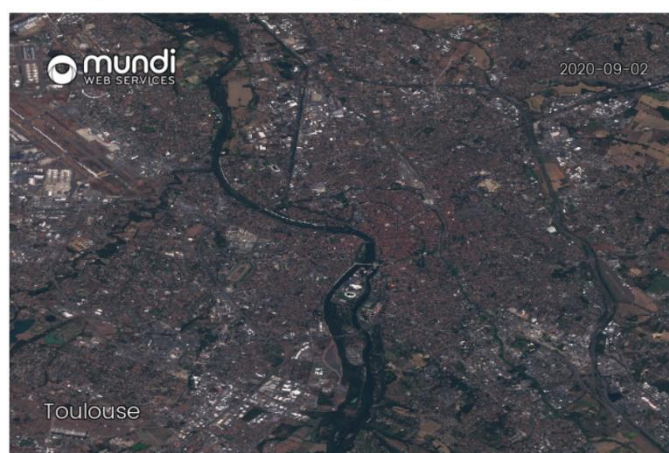
Height

Width

Download Image

Une fois ces paramètres récupérés et l'image récupérée, j'y ajoute des éléments comme le logo, la date, le titre..., puis je l'enregistre dans un répertoire de l'utilisateur et l'affiche sur l'interface.

Your image has been successfully downloaded and saved in the sentinel2\_images folder !



✓ **2<sup>ème</sup> étape** : Afficher une image :

La deuxième fonctionnalité créée sur ce notebook est la possibilité d'afficher une image à partir du dossier contenant l'ensemble des images téléchargées par l'utilisateur.

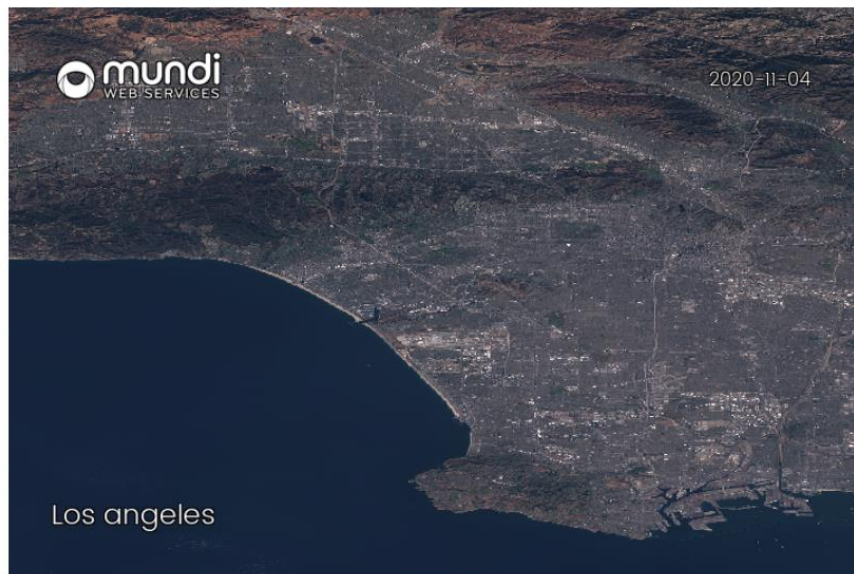
Pour cela je réalise une boucle sur ce dossier pour afficher les noms des fichiers dans le drop-down du widget, puis j'affiche l'image sélectionnée.

```
[3]: sentinel_images_display()
```

Image To Di...

Display this image

Your image has been successfully displayed !



✓ **3<sup>ème</sup> étape** : Effacer une image :

La dernière fonctionnalité consiste à pouvoir effacer une des images téléchargées par l'utilisateur.

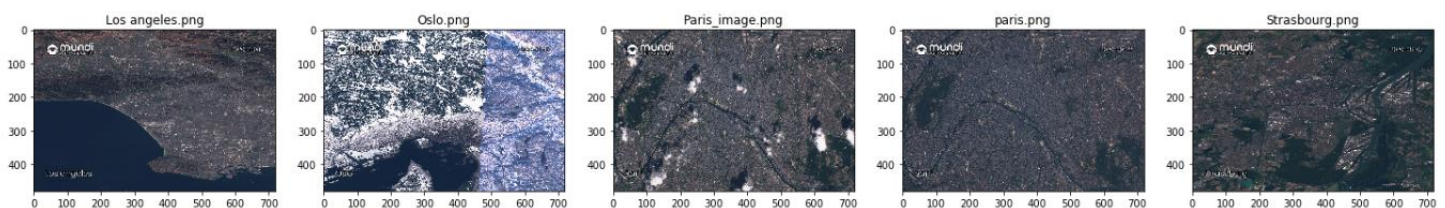
En reprenant une partie de la fonction d'affichage des images, j'y ajoute la visualisation en miniature de ces images pour guider l'utilisateur dans son choix.

```
[6]: sentinel_images_delete()
```

Delete some img

File To Delete

Delete this file



## 2.2.5 Création du notebook Mundi Sentinel 3 search image :

### a) Contexte et définition du besoin :

Le satellite Sentinel 3 (S3) répond à la même logique que Sentinel 2 à savoir l'observation de la terre par l'imagerie mais à une échelle différente. Il permet en effet de mesurer l'évolution de phénomènes de plus grande taille puisque sa résolution est de 300 mètres par 300 mètres. Il va donc être utile pour suivre l'évolution de phénomènes météorologiques comme les ouragans ou de divers phénomènes océaniques. De plus l'un des gros avantages de S3 est d'avoir une fréquence de passage quotidienne, ce qui est très intéressant pour ces suivis de phénomènes.

L'idée de création de ce notebook est d'ailleurs venue d'un besoin de suivre un phénomène particulier à savoir la dérive d'un des plus grands icebergs mesurés dans l'océan Austral afin de communiquer dessus sur le compte Twitter de Mundi. Cet iceberg mesure en effet 150km de long et 50km de large. Les données Sentinel 3 étaient donc parfaitement adaptées à ce suivi. Cependant, il n'existait pas d'outil simple pour récupérer et exploiter ces données. Nous avons décidé de créer un notebook dédié avec pour vocation dans un premier temps de rester interne à l'équipe Mundi.

### b) Conception :

Contrairement aux données Sentinel 2, il n'existe pas de méthodes de classe de la librairie MundiLib permettant de requêter ces données.

J'ai dû donc trouver une alternative et je me suis appuyé sur la construction de l'URL permettant de requêter le service WMS de Sentinel 3. A partir des paramètres (identique à ceux demandés pour Sentinel 2) demandés via l'IHM, je construis l'URL puis j'utilise la méthode get() de la librairie requests pour questionner le service et récupérer l'image en réponse comme on peut le voir sur l'image ci-dessous :

```
xmin = bbox[2]
ymin = bbox[3]
xmax = bbox[0]
ymax = bbox[1]

url = 'https://creodias.sentinel-hub.com/ogc/wms/891d1d73-cdb0-43ba-a453-b34a6e1dc6e8?
SERVICE=WMS&REQUEST=GetMap&TRANSPARENT=true&LAYERS='+layers+'&VERSION=1.1.1&MAXCC=10&FORMAT=image%2Fpng&STYLES=&showLogo=false&time='+str(start_date)+'%2F'+str(stop_date)+'&width='+str(width)+'&height='+str(height)+'&SRS=EPSG%3A4326&bbox='+str(xmin)+'%2C'+str(ymin)+'%2C'+str(xmax)+'%2C'+str(ymax)

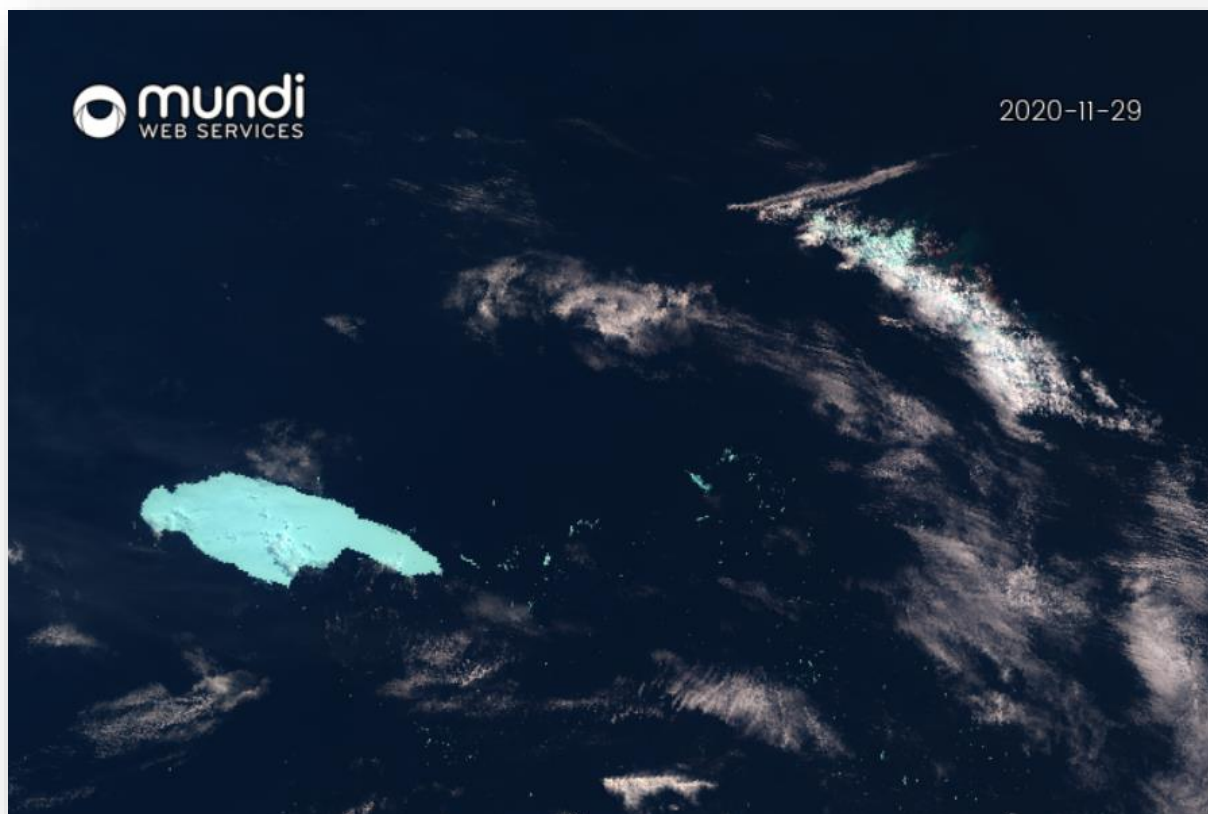
img = requests.get(url)

return img
```

Les autres étapes sont identiques à celles du notebook Mundi Sentinel 2 Easy Search. La différence est la possibilité de générer un fichier GIF à partir du téléchargement de plusieurs images.



Voici ci-dessous un exemple d'image récupérée via ce notebook. Il s'agit de l'iceberg évoqué précédemment :



## 2.2.6 Création du notebook de détection des eaux permanentes :

### a) Contexte et définition du besoin :

L'un des objectifs principaux de Mundi et du programme Copernicus est de proposer des solutions avant-gardistes pour répondre à des problématiques diverses d'entreprises ou d'institutions.

Ces innovations peuvent soit venir d'un besoin clairement exprimé ou soit venir de la réponse à un besoin anticipé. L'idée est ensuite de réaliser un « Proof of concept » (POC) afin de démontrer les capacités de Mundi pour y répondre.

Dans ce cadre précis, l'expression du besoin est venue du département de l'Environnement, de l'Alimentation et des Affaires rurales (Department for Environment, Food and Rural Affairs ou DEFRA) qui a soumis la volonté de pouvoir mesurer dans le temps l'évolution des zones d'eau permanentes au Royaume-Uni.

On qualifie d'eau permanente, la présence d'eau sur un endroit donné tout au long de l'année (lac, rivières, fleuves...). Bien entendu cette présence d'eau peut fluctuer. L'idée de cette mesure est de pouvoir anticiper d'éventuelles crues de ces eaux permanentes et de mesurer l'impact de ces crues sur l'écosystème local.

Pour répondre à ce besoin global, nous avons identifié les besoins fonctionnels suivants :

- 1- Être en capacité de mesurer sur une période de temps précise la présence d'eau sur une zone géographique donnée.
- 2- Être en capacité de comparer cette présence d'eau sur plusieurs périodes de temps.
- 3- Calculer et mettre en avant les données suivantes : superficie en km<sup>2</sup>, proportion d'eau sur la zone géographique, moyenne de la superficie en km<sup>2</sup> sur plusieurs périodes de temps, moyenne de la proportion sur plusieurs périodes de temps.

## **b) Conception :**

### **a. Choix du layer :**

Pour concevoir ce notebook, il fallait déjà se poser la question de quelle source de données utiliser. La réponse fut les données du satellite Sentinel 2, car celui-ci permet via ses caractéristiques de distinguer divers éléments présents sur une image.

En effet, par la création de layer, on est capable entre autres de qualifier la présence de végétation, ou de parcelles d'agriculture, détecter la présence de neige, ou de zones brûlées par les feux de forêt...

Pour la présence d'eau, un layer existait déjà. Il se nomme NDWI pour Normalized Difference Water Index (cf. image ci-dessous).



Cependant, celui-ci n'était pas totalement approprié car il mettait en avant d'autres éléments que l'eau et qu'il n'était pas assez précis sur les zones urbaines.

De ce fait, nous avons décidé de créer un layer spécifique. Pour ce faire, je me suis basé sur un layer existant qui s'appelle Scene Classification (SCL) et qui permet d'attribuer un numéro à chaque pixel de l'image équivalent à un type d'élément (eau, neige, végétation, culture, nuages...). En fonction de ce numéro, on retourne une couleur qui s'affiche sur l'image.

Dans le cadre de la détection des eaux permanentes, le seul élément que l'on veut voir s'afficher est l'eau. J'ai donc créé un layer que j'ai nommé SCL-WATER-ONLY, qui ne prend en compte que cet élément en renvoyant la couleur bleue si détectée, blanche si non détectée, et noire si pas de donnée :

```
var scl = SCL;
if (scl == 0) { // No Data
  return [0, 0, 0]; // black
} else if (scl == 6) { // Water
  return [0, 0, 1]; // blue
} else {
  return [1,1,1]; //white
}
```

#### b. Ajout des données Sentinel 2 layer SCL-WATER-ONLY sur une cartographie :

Suite à la création de ce layer, nous étions en capacité de récupérer des données Sentinel 2 n'affichant que les pixels représentant de l'eau (en bleu).

Cependant, pour rendre les données compréhensibles et interprétables par un utilisateur, il était nécessaire « d'habiller » l'image avec des données de cartographie. Pour y parvenir, j'ai utilisé la librairie Python Folium.

Cette librairie permet de créer une cartographie et de superposer divers éléments y compris des données WMS.

La première étape étant de recueillir les informations de localisation et de temporalité de la part de l'utilisateur via l'interface graphique du notebook :

define\_parameters\_first\_map()

Cities

Bounding Box

Raw Bounding Box

Enter a city in the UK:

Ex: London, Birmingham, Glasgow...

Starting Date

20/11/2020

Stopping D...

21/12/2020

Validate

A partir de ces données, je suis en capacité de créer la folium map et d'y associer les données Sentinel 2 avec le layer SCL-WATER-ONLY. Voici la méthode créée pour y parvenir :

```
def create_folium_map(bbox,start_date,stop_date):
    """This function creates the folium map based on parameters defined in define_parameters_first_map function"""

    bbox_to_save.clear()
    bbox_to_save.append(bbox)

    xmin = bbox[2]
    ymin = bbox[3]
    xmax = bbox[0]
    ymax = bbox[1]
    centerx = (xmax+xmin)/2
    centery = (ymax+ymin)/2

    map=folium.Map(location=[centery, centerx], zoom_start=13,tiles='OpenStreetMap', max_native_zoom=19, max_zoom=100)
    map.add_child(folium.raster_layers.WmsTileLayer(
        url='http://shservices.mundiwebservices.com/ogc/wms/ea23bfb3-2a67-476f-90e3-fe54873ff897?
        time='+str(start_date)+'%2F'+str(stop_date)+'&RESX=10m&RESY=10m',
        layers=['SCL-WATER-ONLY'],
        fmt='image/png',
        version='1.3.0',
        attr='Sentinel2-L2A-SCL-layer',
        name='SCL',
        min_zoom=10,
        max_zoom=20
    ))

    return map
```

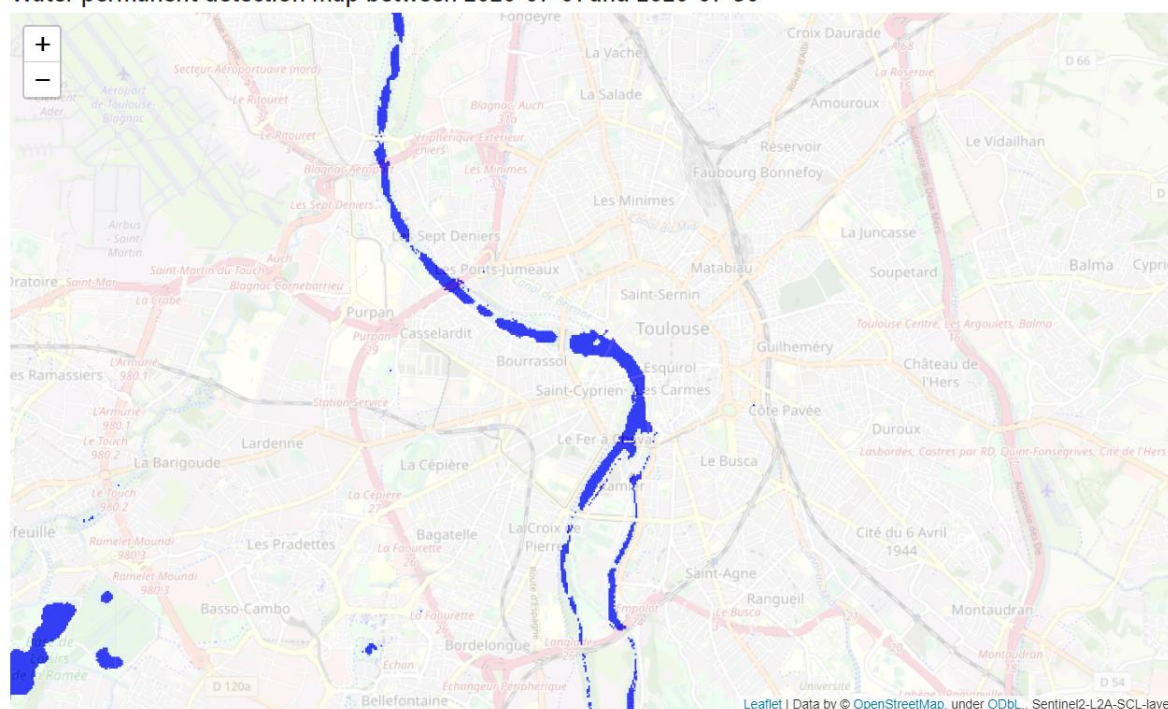
Définition de la zone géographique (à partir de la bounding box)

Création de la folium map avec les données géographiques

Ajout des données Sentinel 2 avec layer SCL-WATER-ONLY

Voici à titre d'exemple, le résultat obtenu pour la ville de Toulouse sur la période allant du 1<sup>er</sup> au 31 juillet 2020. Les données Sentinel 2 qui apparaissent sont celles de la journée de ce mois de Juillet 2020 où il y avait le moins de nuages (caractéristiques du service WMS évoqué précédemment).

Water permanent detection map between 2020-07-01 and 2020-07-30





En complément de la génération de cette première cartographie avec les données Sentinel 2 SCL-WATER-ONLY, j'offre la possibilité de réaliser une comparaison en générant une seconde cartographie (en dessous) qui conserve la même zone géographique mais où on peut modifier la période de temps.

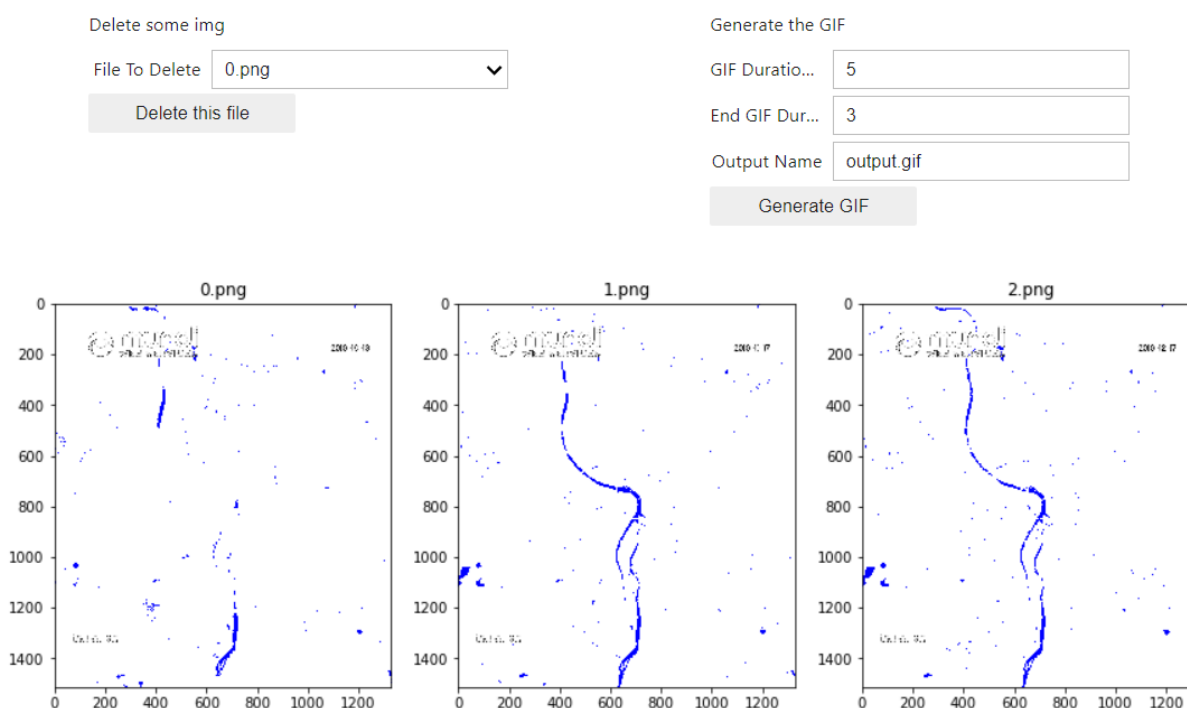
### c. Génération des images et calcul des données :

La deuxième étape de ce notebook est la possibilité de générer des images et de les sauvegarder dans un répertoire dédié sur l'espace « work » de l'utilisateur. L'idée est de télécharger plusieurs images sur une période de temps afin de pouvoir suivre l'évolution de la présence d'eau dans le temps. On demande notamment une date de début et de fin mais également une périodicité.

**Exemple :** si je choisis une date début au 1<sup>er</sup> janvier 2020 et une date de fin au 31 décembre 2020 avec une périodicité de 30 jours, alors le service WMS récupérera une image par tranche de 30 jours, soit 12 images sur cette année.

Une fois ces images téléchargées, il est possible de visualiser ces images en miniature pour décider d'en supprimer une ou plusieurs si elles ne sont pas satisfaisantes (présence trop importante de nuages par exemple). Enfin à partir des images restantes, il est possible de réaliser une animation sous format d'un fichier GIF.

[5]: gif\_creation()



Enfin, l'étape importante à partir de ces images téléchargées est de pouvoir calculer différentes données à partir du nombre de pixels d'eau présents sur l'image. On sait qu'un pixel représente une zone de 10 mètres par 10 mètres soit 100 mètres<sup>2</sup>. On peut donc calculer les données de superficie et la proportion d'eau sur l'image.

Tous ces calculs sont réalisés dans la fonction `display_pixels()`, comme on peut voir ci-dessous :

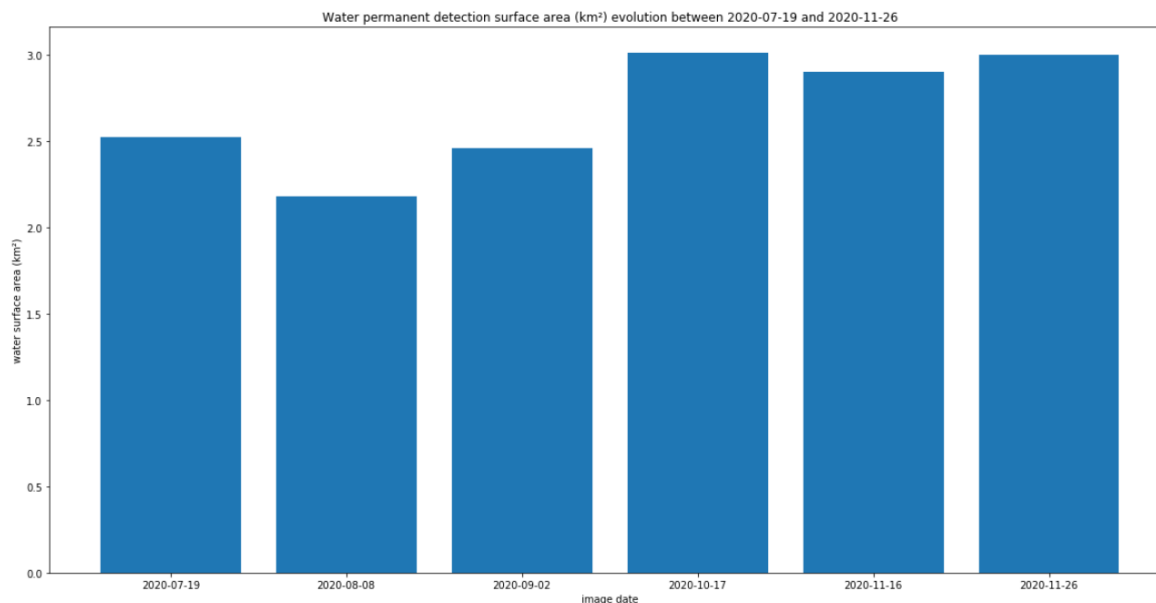
On prend chaque image contenue dans le répertoire dédié	←	<code>for file in final_filenames:</code>
On récupère chaque pixel de l'image avec la méthode <code>getdata()</code>	←	<code>im = Image.open(SAVE_FOLDER_IMAGES+file)</code> <code>blue_pixel = 0</code> <code>total_pixel = 0</code> <code>for pixel in im.getdata():</code>
Si la couleur du pixel est égale à bleue alors on l'ajoute à variable <code>blue_pixel</code>	←	<code>total_pixel += 1</code> <code>if pixel == (0, 0, 255): # if your image is RGB (if RGBA, (0, 0, 0, 255) or so</code> <code>blue_pixel += 1</code>
On calcule la superficie en km <sup>2</sup>	←	<code>blue_pixel_in_km2 = round((blue_pixel/10000),2)</code> <code>list_histo.append(np.array(blue_pixel_in_km2))</code>
On calcule la moyenne de la superficie de l'ensemble des images	←	<code>total_blue = total_blue + blue_pixel_in_km2</code> <code>avg_km2 = round((total_blue / len(list_histo)),2)</code>
On calcule la proportion de pixels d'eau par rapport au total de pixels. On calcule la moyenne de ces proportions sur toutes les images.	←	<code>water_proportion = round(((blue_pixel / total_pixel)*100),2)</code> <code>total_water_proportion = total_water_proportion + water_proportion</code> <code>avg_water_proportion = round((total_water_proportion / len(list_histo)),2)</code>
On affiche l'ensemble des données dans le notebook	←	<code>print('Image name : \033[1m'+str(file)+'\033[0m, Date : \033[1m'+dates_list[cpt]+''\033[0m :')</code> <code>print('The water surface area is : \033[1m\033[94m'+str(blue_pixel_in_km2)+'km²\033[0m. The water proportion is : \033[1m\033[94m'+str(water_proportion)+'%\033[0m')</code> <code>cpt += 1</code>  <code>display(HTML('&lt;h3&gt; The average water surface area is : &lt;b&gt;'+str(avg_km2)+'km²&lt;/b&gt;, and the average water proportion is : &lt;b&gt;'+str(avg_water_proportion)+'%&lt;/b&gt;. &lt;/h3&gt;'))</code> <code>print('')</code>

Le résultat en sortie sur l'interface du notebook est le suivant (exemple réalisé pour Toulouse sur les 6 derniers mois) :

```
Image name : 0.png, Date : 2020-07-19 :
The water surface area is : 2.52km². The water proportion is : 1.25%
Image name : 1.png, Date : 2020-08-08 :
The water surface area is : 2.18km². The water proportion is : 1.08%
Image name : 2.png, Date : 2020-09-02 :
The water surface area is : 2.46km². The water proportion is : 1.22%
Image name : 3.png, Date : 2020-10-17 :
The water surface area is : 3.01km². The water proportion is : 1.5%
Image name : 4.png, Date : 2020-11-16 :
The water surface area is : 2.9km². The water proportion is : 1.44%
Image name : 5.png, Date : 2020-11-26 :
The water surface area is : 3.0km². The water proportion is : 1.49%
```

**The average water surface area is : 2.68km<sup>2</sup>, and the average water proportion is : 1.33%.**

Enfin, grâce à la librairie `pyplot` de `Matplotlib`, il est possible de réaliser un graphique reprenant l'ensemble des données de superficie en km<sup>2</sup>, comme on peut le voir sur l'image ci-dessous :



En conclusion, la réalisation de ce notebook démontre la capacité de Mundi d'entreprendre une démarche innovante répondant à des besoins souvent exprimés par les clients de Mundi. Cela démontre également l'intérêt de l'imagerie satellite pour répondre à des besoins concrets de la vie quotidienne.

## 2.2.7 Définition nouvelle architecture de présentation des notebooks :

### a) Contexte et définition du besoin :

Nous avons évoqué au début de ce chapitre sur les Jupyter Notebooks, le fait que deux profils d'utilisateurs avaient été identifiés : un profil débutant et un profil expert. Nous avons évoqué le fait que certains notebooks s'adressaient au profil expert et d'autres au profil débutant. Enfin, à travers la réalisation de mes diverses missions sur les notebooks, j'ai particulièrement ciblé le profil débutant en proposant des notebooks permettant d'accéder aux données Mundi de manière interactive et simplifiée.

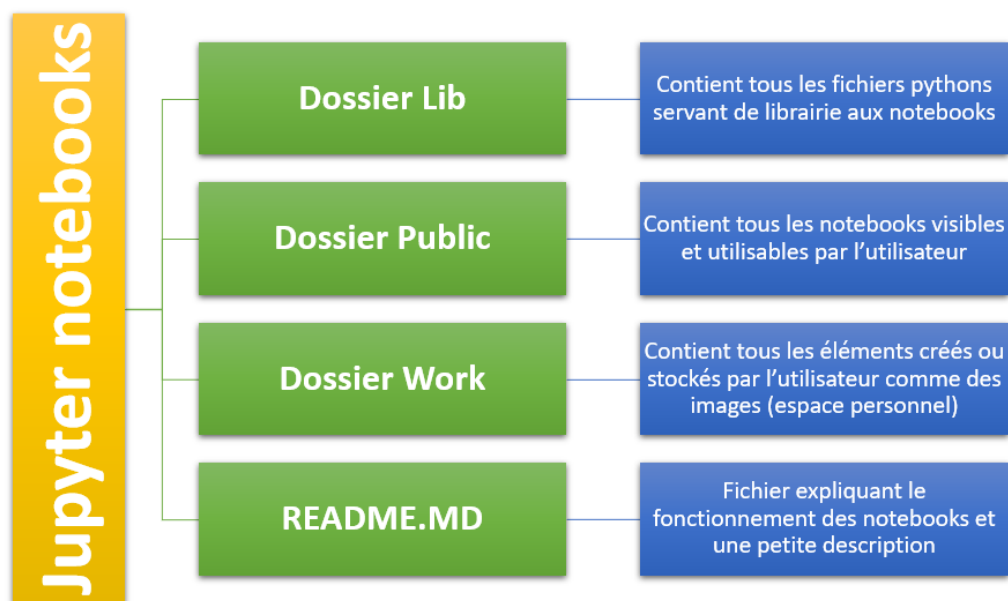
Toutefois, cette catégorisation ne se retrouvait pas dans la présentation de l'offre notebook lorsque l'utilisateur arrivait sur la page d'accueil. De plus, on a constaté qu'il manquait de l'information pour comprendre globalement à quoi servait les Jupyter notebook et comment s'en servir.

Il a été décidé de ce fait de revoir l'architecture de ces notebooks et de créer un notebook d'accueil informationnel.

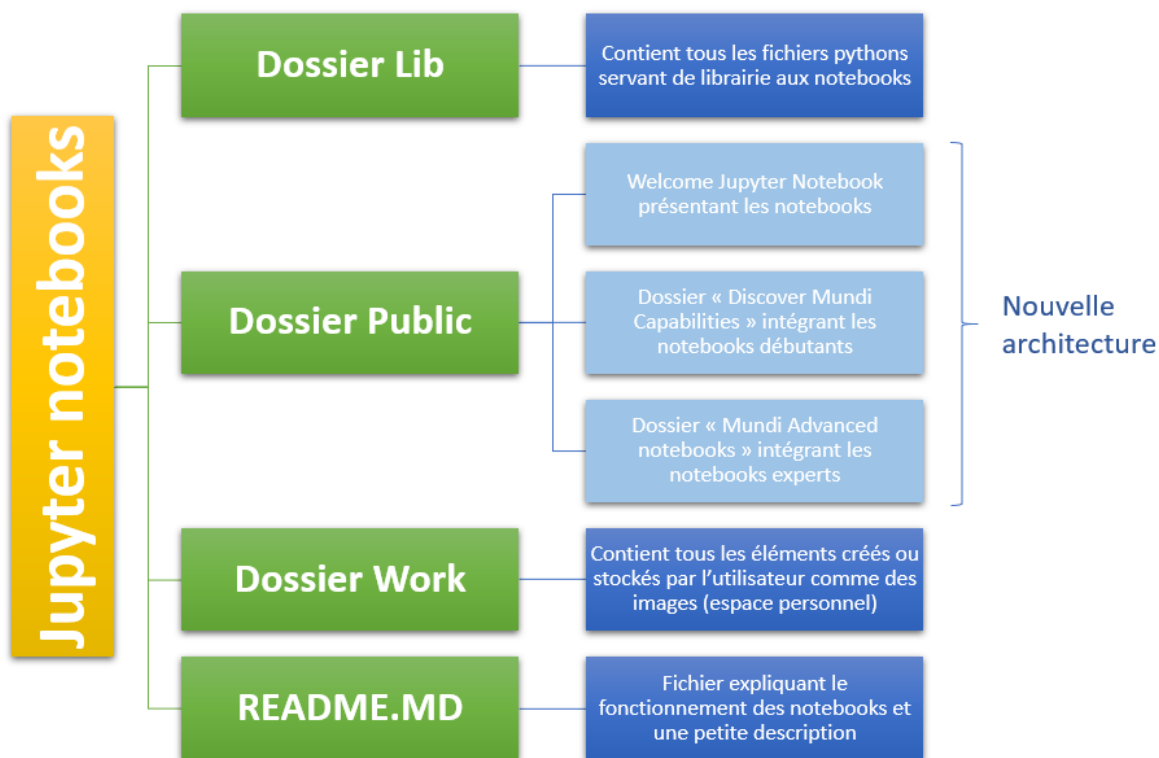


## b) Conception :

L'architecture précédente des Jupyter Notebooks était la suivante :



L'idée fut donc de diviser le dossier Public en deux, afin de distinguer les notebooks qui s'adressent au profil débutant et ceux qui s'adressent au profil expert. De plus un notebook d'accueil a été créé et positionné à la racine de ce dossier Public :



Le notebook Welcome Jupyter Notebooks permet d'expliquer aux nouveaux utilisateurs les éléments suivants :

- ✓ Qu'est-ce que Mundi ?
- ✓ Que sont les Jupyter Notebooks ?
- ✓ Comment fonctionne les Jupyter Notebooks ?
- ✓ Comment sont-ils structurés ?
- ✓ Une description de chaque notebook avec un lien pour l'ouvrir.

Voici à quoi ressemble le Welcome Jupyter Notebook :

### III- How our notebooks are structured ?

In Mundi team, we think that the Jupyter notebooks can be used in two different ways :

- First, it can be used to discover what Mundi can offer, what kind of data you can access and how manipulate them easily. It is addressed to **beginners with Earth Observation data**. All of those notebooks are referenced in the `Discover_mundi_capabilities` folder.
- Secondly, it can be used to go deeper in the Earth Observation data analysis. If you have specific needs, or need help with algorithms problematics, we provide you with different notebooks. It is then addressed to **experts with Earth Observation data**. All of those notebooks are referenced in the `Mundi_advanced_notebooks` folder.

#### 1- Discover Mundi capabilities notebooks :

- `01_mundi_S2_easy_search` :

This notebook is the better way to start with Mundi. In less than 1 minute, you will be able to create your first Sentinel 2 image ! Select the city you want, the sentinel 2 layer, the nearest date and give an image title and that's all !

In few seconds, the image desired will be displayed and stored on your sentinel2\_images folder.

[Click here to access mundi\\_S2\\_easy\\_search notebook](#)

Cette nouvelle approche permet finalement de permettre une meilleure compréhension des Jupyter Notebooks par les nouveaux utilisateurs. L'information est centralisée, claire et doit permettre de répondre à un besoin spécifique.

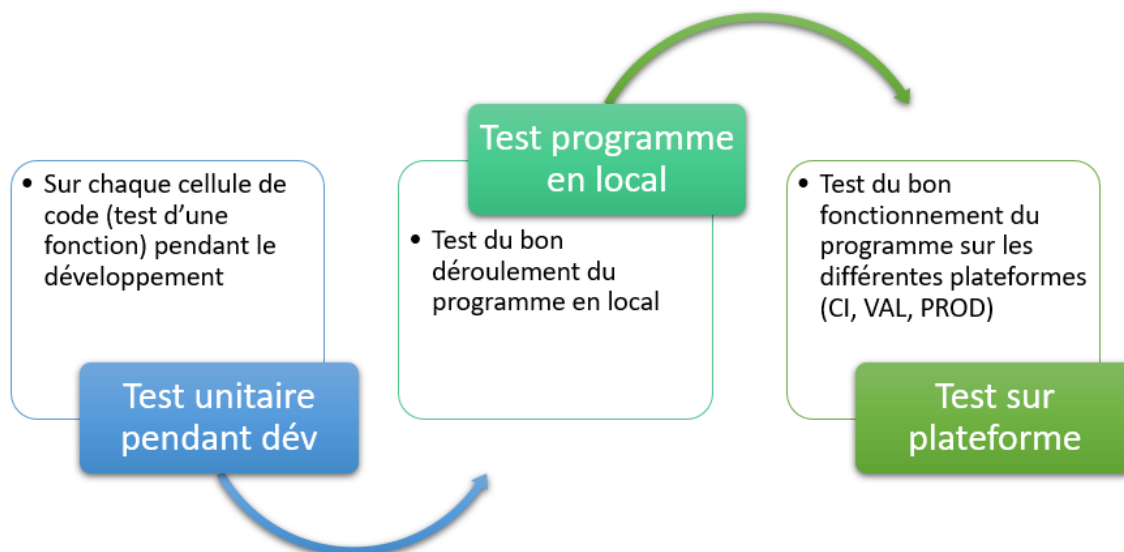
## 2.2.8 Tests et validations :

Nous avons vu au cours du chapitre précédent, comment ont été réalisés les divers programmes relatifs aux Jupyter Notebook. Nous allons voir dans ce chapitre quelle procédure de tests et de validations a été mise en place afin de s'assurer du bon fonctionnement des nouveaux notebooks créés.

Le fonctionnement intrinsèque des Jupyter Notebooks permet de mettre en place une procédure de test unitaire. En effet, le fait que le code soit découpé dans des cellules permet de tester le programme pas à pas et indépendamment des autres parties du programme.

Ainsi à chaque méthode développée, il est possible de l'exécuter et de la tester instantanément. Vient ensuite le test du programme en entier, puis sur les différentes plateformes évoquées précédemment.

La procédure de test peut donc se décrire de la manière suivante :



Cette procédure est étroitement liée à la procédure de développement détaillée en partie 2.2.2. Il est temps maintenant de détailler les différentes étapes d'un test unitaire. Pour cela nous allons prendre l'exemple d'une fonction du Notebook CAMS ECMWF API : `request_ecmwfapi()`.

Dans cette fonction, on construit la requête puis l'exécute. Nous devons donc tester que :

- La requête est bien construite et intègre bien les bons paramètres
- Vérifier le bon fonctionnement de l'appel à l'API et la réception du fichier en réponse.

Pour tester que la requête est bien construite avec les bons paramètres, on ajoute des « `print()` » dans la fonction avec les valeurs des paramètres et la requête en entier.

```
print("parameter : "+data_parameter_widget.value+", start date : "+str(start_date_widget.value)+", stop date : "+str(current_date_widget.value)+", time : "+time_param)
```

request\_ecmwfapi(output\_filename)

Data selecti... Nitrogen dioxide ▼

Starting Date 27/11/2020 📅

Final date 27/12/2020 📅

Select one or many hours below :

☒ 06:00:00

☐ 12:00:00

☐ 18:00:00

Validate

parameter : Nitrogen dioxide, start date : 2020-11-27, stop date : 2020-12-27, time : /06:00:00

On voit bien que les valeurs correspondent. Pour cette fonction, il existe 5 possibilités de paramètres de gaz et 3 possibilités horaires. On teste donc toutes les possibilités.

Ensuite pour tester le bon fonctionnement de la requête il suffit de l'exécuter et de s'assurer qu'elle appelle bien l'API d'ECMWF et que le fichier est bien transféré en réponse.

## ✓ Exécution de la requête API :

```
2021-01-11 20:25:34 ECMWF API python library 1.5.4
2021-01-11 20:25:34 ECMWF API at https://api.ecmwf.int/v1
2021-01-11 20:25:34 Welcome Guillaume Lenoir d'Espinasse
2021-01-11 20:25:34 In case of problems, please check https://confluence.ecmwf.int/display/WEBAPI/Web+API+FAQ or contact servicedesk@ecmwf.int
2021-01-11 20:25:35 Request submitted
2021-01-11 20:25:35 Request id: 5ffcb43f5c9103446e12af5f
2021-01-11 20:25:35 Request is submitted
2021-01-11 20:25:36 Request is active
```

## ✓ Réponse à la requête API :

```
Process '['nice', 'mars', '/tmp/20210111-2020/b7/tmp_marspukSIW.req']' finished
2021-01-11 20:25:47 Request is complete
2021-01-11 20:25:47 Transferring 61.89 Mbytes into /home/jovyan/work/cams_data/cams_ecmwfapi/images_data/output.grib
2021-01-11 20:25:47 From https://stream.ecmwf.int/data/webmars-public-svc-blue-000/data/scratch/20210111-2020/bc/_mars-webmars-public-svc-blue-000-6fe5cac1a363ec1525f54343b6cc9fd8-BKyCg8.grib
2021-01-11 20:25:51 Transfer rate 14.2341 Mbytes/s
```

Ensuite, dans la fonction `get_projection_map()`, on s'assure que les différents paramètres de téléchargement et d'affichage des images soient cohérents avec les valeurs renseignées dans l'IHM. Exemple avec le choix de la projection géographique :

```
get_projection_map(output_dir, output_filename)
```

Map Location

Validate

projection map : WORLD

Enfin on s'assure que la sortie (images) respecte bien l'ensemble des paramètres sélectionnés : choix du gaz, dates, time, projection, palette couleur, titre, nom du fichier GIF :

gist\_ncar

tab20b

tab20c

Map colors

gist\_ncar

Display dire...

Right to left

Title

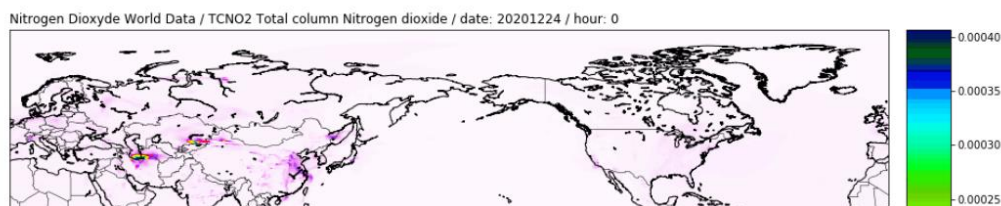
Nitrogen Dioxyde World Data

GIF file name

Test

Validate

GIF File Generated



Une fois que les tests unitaires et les tests de fonctionnement du programme en local sont validés, on peut dès lors « pousser » le programme sur la plateforme de développement (CI).

On y déroule ensuite de nouveaux scénarii de test identiques à ceux que nous avons déroulé en local afin de s'assurer que la migration n'a pas entraîné de problèmes particuliers sur le programme en question mais également qu'elle n'a pas entraîné des effets de bord sur d'autres programmes.

Vient ensuite le passage en VAL puis en PROD où on réalise à nouveau l'ensemble de ces tests de fonctionnement et de validation.

### 3. Bilan de mes missions :

Au cours de cette période en entreprise j'ai eu l'opportunité de travailler sur différents projets et de mettre en applications les différentes compétences acquises durant la formation. J'ai ainsi pu exercer mes compétences en développement FRONT avec la mission de refonte de la page d'offre du site internet mais également mes compétences en développement BACK et FRONT avec les missions de développement de programmes Jupyter Notebook. En résumé, j'ai eu l'opportunité de travailler sur :

- Le développement de 5 programmes
- D'utiliser les langages suivants : HTML, CSS, Python.
- De me perfectionner sur le langage Python et de découvrir plusieurs librairies importantes : Matplotlib, Numpy, Folium, Ipywidgets, Imageio...
- De développer de manière autonome à la fois dans la réflexion, la conception et l'implémentation tout en bénéficiant de soutien si besoin.
- D'apporter des idées novatrices en termes de contenu (page offre), mais aussi en termes d'approche clients (nouvelle architecture des Jupyter Notebooks).
- De proposer et de concevoir des programmes novateurs répondant à des besoins clients spécifiques (différents programmes d'accès aux données Mundi).
- De travailler sur la réalisation d'un Proof Of Concept, innovant, mettant en évidence les capacités des données d'observations de la terre pour répondre à des besoins spécifiques et concrets du quotidien.

L'ensemble des programmes, pages, que j'ai développé sont en production et donc actuellement opérationnels.

## 4. Bilan personnel :

Réaliser mon alternance au sein du projet Mundi a été une très belle opportunité. J'ai eu l'occasion de travailler dans un environnement complexe, innovant, international, où la créativité est une nécessité. J'ai également eu l'opportunité de côtoyer des personnes spécialisées à la fois dans les projets numériques complexes mais également dans le domaine des données d'observation de la Terre.

J'ai de ce fait énormément appris au cours de mes missions. J'ai découvert le monde des données spatiales et plus précisément des données d'observations de la Terre. C'est un univers très riche, intéressant avec un champ des possibilités conséquent. Ce marché n'en est pas à ses prémices mais beaucoup reste à faire pour transformer ces données en solution impactant directement le quotidien des gens. J'ai selon moi à mon humble échelle contribué à rendre ces données plus accessibles en simplifiant leurs accès par la création de différents programmes mais aussi en m'adressant directement à cette cible de clients non experts.

Ce projet a été l'occasion pour moi d'approfondir les compétences acquises durant les 6 mois de formation à l'AFPA. Guidé par la curiosité, j'ai notamment découvert un nouveau langage Python que j'ai particulièrement apprécié. J'ai aussi découvert les tenants et aboutissants d'un projet numérique de grande ampleur, les contraintes que cela pouvait représenter et l'organisation au quotidien qui en découlait.

J'ai apprécié la liberté, la confiance et l'autonomie qui m'a été donnée par l'équipe managériale et notamment par Vincent Saleh, ce qui m'a permis à la fois d'être force de proposition mais aussi de créer par moi-même, avec les essais-erreurs que cela pouvait comporter mais qui sont les clés de l'apprentissage.

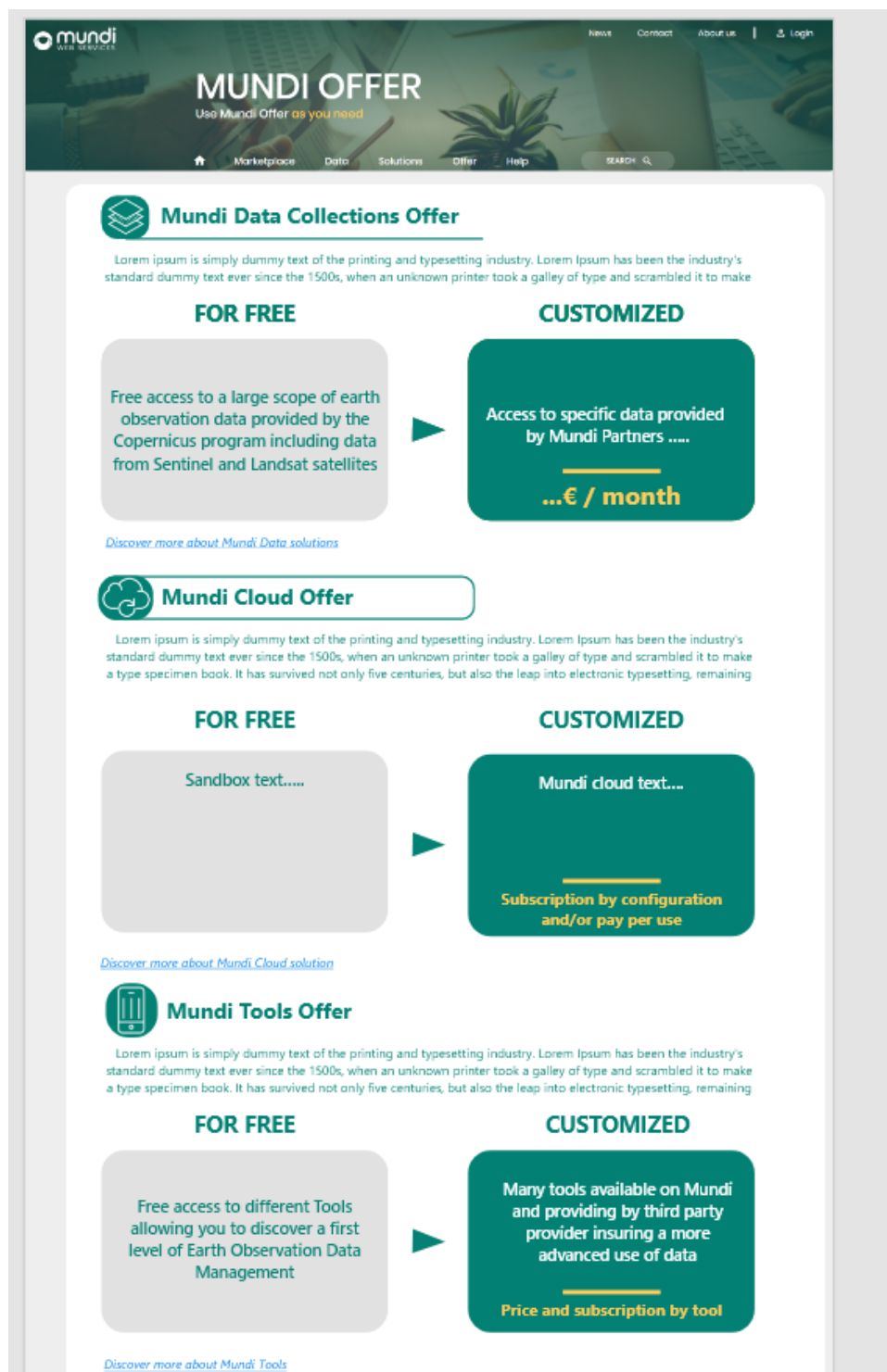
J'ai été particulièrement intéressé par les problématiques fonctionnelles avec l'objectif pour moi d'aller plus loin que la programmation en comprenant les enjeux et les problématiques du projet et en me familiarisant avec tous les contours du projet.

C'est d'ailleurs l'une des raisons (parmi d'autres) qui me poussent pour la suite à m'orienter sur les métiers fonctionnels car j'ai à cœur étant donné mon parcours précédent la formation et ces nouvelles compétences acquises de me positionner au cœur de la compréhension du besoin client. Un projet IT n'a de sens selon moi que s'il répond à des besoins clients et créer ce lien entre le client et l'équipe de développement est un rôle qui a du sens pour moi. C'est pourquoi je rejoindrai à l'issue de ce projet un poste de Product Owner pour les outils de Supervision de la DSNA (Direction des Services de la Navigation Aérienne).



## 5. ANNEXES :

### Annexe 1 : Maquette page offre



## Annexe 2 : Page Offre du site internet Mundi Web Services

## MUNDI DATA OFFER



Mundi integrates a large range of Earth Observation and non EO data. The collection is regularly enriched to offer all material to build and operate your service. You can find free and licensed products according to your needs.

## FREE

Free access to full Copernicus Sentinel satellites data (1,2,3,5P), and also Landsat satellites :

- > Discover data by filtering time, geographic area and other parameters through GeoData tool or Mundi catalogue API
- > Visualise and download products through :
  - GeoData tool
  - REST API

## ADVANCED

- > As a user, access to specific data collection allowing applications in various domains such as: agriculture monitoring, climate change, land and urban monitoring.
- > As a data provider, add your own data collection to the Mundi offer, and monetize it.
- > Find all data collection in [the Marketplace !](#)

**Prices according to data collection**

[Discover more about our data offer](#)

## MUNDI CLOUD OFFER



Mundi has selected Open Telekom Cloud for a High-Level cloud service to host the platform and all your applications. Operated by T-Systems, it is a robust and scalable public cloud platform that insures the performance and the security of your applications.

## FREE SANDBOX ON THE CLOUD

[Mundi Jupyter Notebook](#) is a great tool to discover the power of Mundi. It allows you to process data on Mundi with your own algorithms.

Mundi Jupyter Notebook is a web-based interactive development environment. You can use either tools developed by Mundi team to process the data, or your own programs based on Python or R languages.

## ADVANCED

As each application needs are different, Mundi offers thanks to Open Telekom Cloud the ability to customize the cloud configuration that suits your needs: elastic cloud server, compute optimized server, bare metal server, GPU server... Make your own choices on the [cloud price calculator](#). See below some examples (small, medium, large).

**Price depends on the processing power**

SMALL	MEDIUM	LARGE
<b>Elastic Cloud Server</b> vCPU = 2 / RAM = 16 GB	<b>Elastic Cloud Server</b> vCPU = 8 / RAM = 64 GB	<b>Elastic Cloud Server</b> vCPU = 32 / RAM = 256 GB
<b>Elastic Volume Storage</b> 500 GB	<b>Elastic Volume Storage</b> 2 TB	<b>Elastic Volume Storage</b> 8 TB
<b>Object Storage Service</b> 500 GB	<b>Object Storage Service</b> 4 TB	<b>Object Storage Service</b> 80 TB
<b>115.62€ / month - 0.158€ / hour *</b>	<b>479.99€ / month - 0.658€ / hour *</b>	<b>2915.40€ / month - 3.994€ / hour *</b>

Discover more about our cloud offer

\* From Linux configuration example calculated with the cloud-price calculator, non-contractual price.

## MUNDI SERVICES & TOOLS OFFER



Our tools collection is built on a selection of specialized building-bricks to manage all types of data and usecases. Mundi eases the access to data and processing power and proposes dedicated free or licensed software.

### FREE

Free access to Tools and Services allowing you to discover a first level of Earth Observation data usage.

> Free use of services such as :

- Discovery service (Opensearch, CSW),
- View Service (Trial usage),
- Download through the website,
- Request offline data (up to 300 products/month)
- Access to [archived data](#) (login required)

> Dedicated tools available for free from our [Marketplace](#) :

- Mundi Explore
- Mundi Software Shared Repository with open-source library ESA SNAP, OTB (Orfeo Toolbox) and GDAL

### ADVANCED

Many tools available on Mundi provided by third-party partners insuring a more advanced use of data.

> Advanced services :

- Direct download from S3 Object Storage, Massive download,
- Processing-as-a-Service :
  - Enhanced Cloud detection
  - ARD products for Agriculture
- Access to the full archive of data

> Advanced tools :

- We offer a selection of best-of-breed commercial services, directly available on our [Marketplace](#).

Price and subscription by tool

Discover more about our tools offer

## MUNDI SUPPORT OFFER



Mundi has the skills and the expertise to assist you in a very large range of activities. From technical or functional support to more thematic or business consulting, Mundi team is organized to provide a support tailored to your needs.

### FREE HELPDESK

Access the free Helpdesk to request support for service inquiries, quotations, technical documentation and how-to, about any topic related to the data offer, the public cloud usage and the free tools.

Use either the [contact](#) form for new users, or directly the online [helpdesk](#) tool for registered users.

### ADVANCED SUPPORT

Thanks to the specialized skills of Mundi, we are here to deliver top-level support including IT and IS support, Earth Observation expertise and business support and consultancy.

IT and IS support are provided by Tier-1 companies: [Atos](#) and [T-Systems](#).

#### Mundi Technical Support

- > Technical support for cloud infrastructure setup and move to cloud strategy
- > Technical expertise for application development and integration
- > SLA and operational commitments for application management
- > Support for data providers onboarding on Mundi

**750€ / day**

[Discover more about our support offer](#)

#### Mundi Business Support

- > Increase your awareness and business by promoting your service or application on our Marketplace (**6 months free subscription then 2000€/month**)
- > Benefit from the international footprint and Tier-1 customers contacts to raise your service at a new business level
- > Find top-level startups and specialized partners for the development of your offer in the space business

**Price on demand**

[Discover more about our marketplace](#)

Our General Terms & Conditions :

[General Terms & Conditions V4.0 currently valid](#)