

Rapport du stage

Plan

Présentation de l'entreprise

Contexte du projet et problématique

Architecture de base

Solution proposée

Mise en place des environnements de travail :

- VirtualBox
- Ubuntu
- MongoDB
- Hadoop
- Hive
- Spark
- Power bi desktop
- Facturation
- Orange data mining

Pour chaque environnement de travail on aura :

- La présentation
- Les étapes d'installation
- Le cas d'utilisation

Présentation de l'entreprise

SFM est une entreprise créée en 1995, issue du domaine des télécommunications et des réseaux. Son équipe d'experts et d'ingénieurs de haut niveau réalise des missions d'ingénierie et de conseil pour le compte de régulateurs des télécommunications, d'opérateurs, de Ministères des TIC...etc.

C'est au cours de ses missions que SFM a développé des outils, applications et plateformes pour la digitalisation des processus d'ingénierie, de suivi et de mesures de QoS/QoE.

La société accompagne aujourd'hui les acteurs du secteur public et privé à la maîtrise des données et au développement de solutions IoT (internet d'objets) et du Big Data.

Contexte et problématique

Aujourd'hui, les entreprises ont des informations provenant de différents canaux pour tous leurs aspects métier. L'utilisation correcte de ces données permet de créer la valeur et d'avoir un avantage concurrentiel.

L'entreprise SFM utilise les SGBDR pour stocker et traiter les données structurées. Ces données générées par dropy et voltix sont non seulement volumineuses, ont une vitesse importante, mais aussi diversifiées, ce qui traduit le terme du "big data".

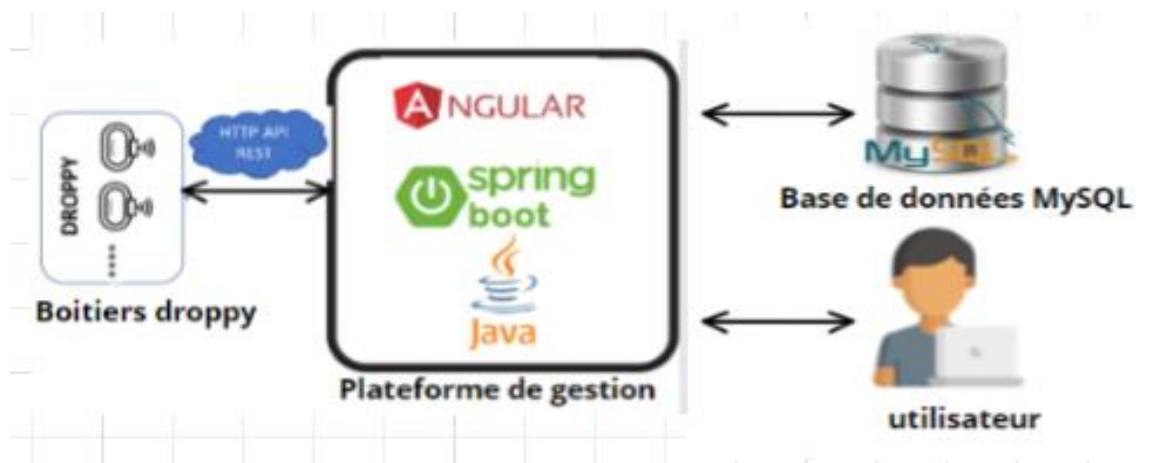
Face aux problèmes de performance, les SGBDR ne sont plus en mesure de stocker des volumes importants des données et de les traiter. Pour pouvoir les prendre en charge et les analyser, il est nécessaire de s'en remettre aux outils analytiques Big Data.

Le Big Data devrait permettre d'analyser et traiter les données générées par dropxy et voltix et ainsi, optimiser l'usage de cette technologie. Ces outils ont la capacité de traiter rapidement et efficacement des larges volumes de données.

Dans une logique de proposer une solution, nous sommes amenés à étudier les limites de l'architecture existante (SGBDR), proposer un nouveau paradigme de stockage, des outils de big data pour le traitement et pour l'analyse des données, des Dashboard pour la visualisation des données afin de prendre une décision et de prévision sur la consommation d'eau et d'électricité.

L'adoption de cette architecture Big Data permettra d'améliorer significativement la gestion du stockage, du traitement et de l'analyse des données pour SFM Connect, ouvrant ainsi de nouvelles opportunités et possibilités pour l'entreprise.

Architecture existante :



SFM Connect est une plateforme de gestion qui consiste à mettre en évidence deux projet IoT de micro-service ayant la même interface graphique.

Les données du boitiers voltix et dropxy sont récupérer via le protocole HTTP API REST par l'envoi de la trame et ils sont enregistré dans MySQL. L'utilisateur accède l'application de SFM Connect via l'interface graphique développé par angular, java et sprintboot.

Les données provenant de dropxy et de voltix sont stockées dans une base de données relationnel (MySQL). Ces données sont tellement volumineuses qu'une simple base de données n'est plus en mesure de stocker et traiter.

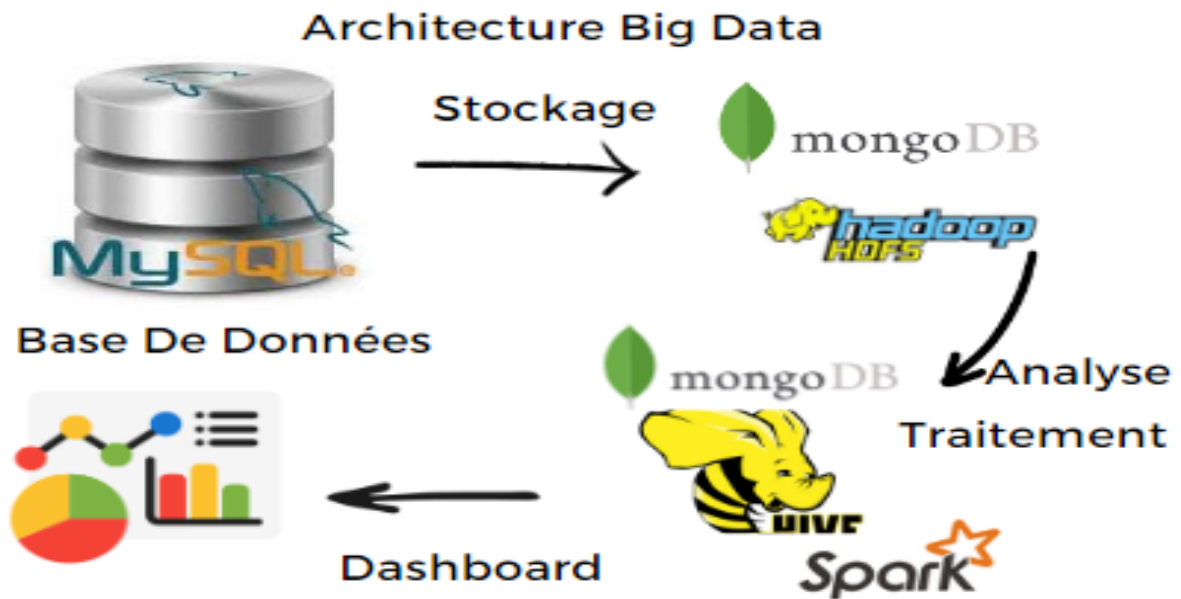
MySQL est un système de gestion de bases de données relationnelles :

- Avec MySQL, on est :
 - ✓ Incapable de gérer de très grands volumes de données.
 - ✓ Incapable de gérer des débits extrêmes.
- MySQL est un peu adaptées au stockage et à l'interrogation de certains types de données
- On ne peut pas faire de traitement parallèle, ni des traitements en temps réel avec MySQL
- C'est un système centralisé, si le serveur tombe en panne, les données seront inaccessibles

Solution proposée

Pour remédier à ces limites et prendre en charge ces données volumineux, nous proposons l'architecture suivante pour le traitement par lots :

- Un nouveau système de stockage (MongoDB, HDFS)
- De traitement et analyse des données (MongoDB, hive et Spark)
- Dashboard avec power bi desktop
- Prédiction avec orange data Mining



Mise en place des environnements de travail :

- ❖ VirtualBox

VirtualBox est le logiciel de virtualisation gratuit, open source et multiplateforme d'Oracle. Celui-ci permet d'héberger une ou plusieurs machines virtuelles, avec des systèmes d'exploitation différents.



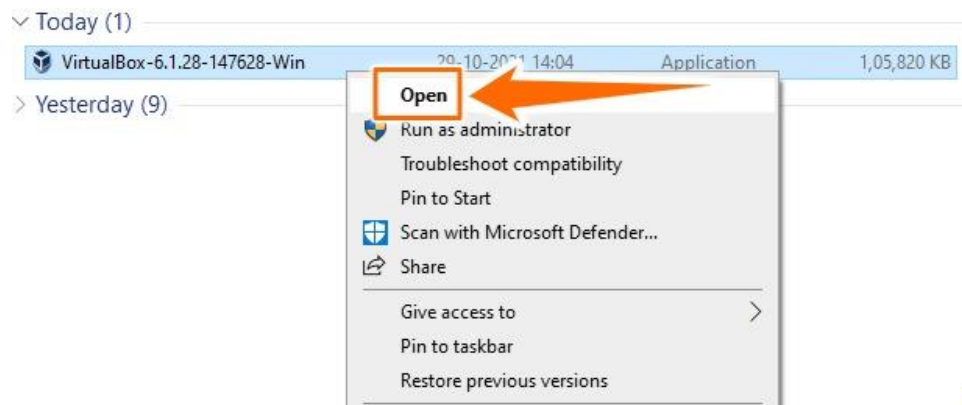
- Les étapes d'installation de VirtualBox

Étape 1 : Allez à la page de téléchargement et cliquez sur ce lien

<https://www.virtualbox.org/wiki/Downloads>.



Étape 2 : Allez maintenant dans le dossier de téléchargement. Cliquez ensuite avec le bouton droit sur le fichier VirtualBox et sélectionnez Ouvrir.

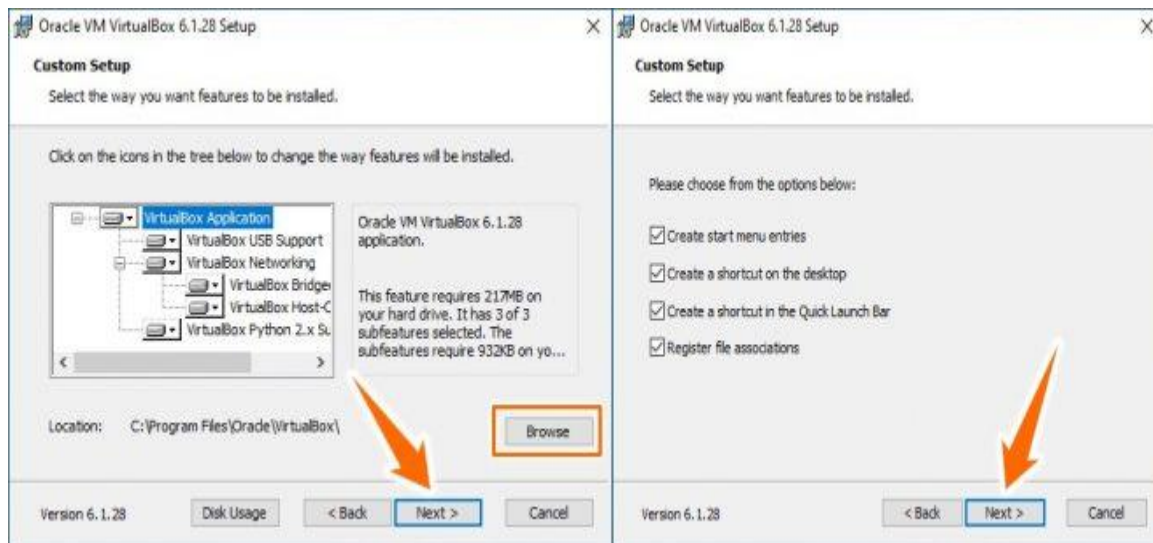


Étape 3 : L'étape précédente ouvrira l'assistant d'installation. Ensuite, cliquez Suivant.

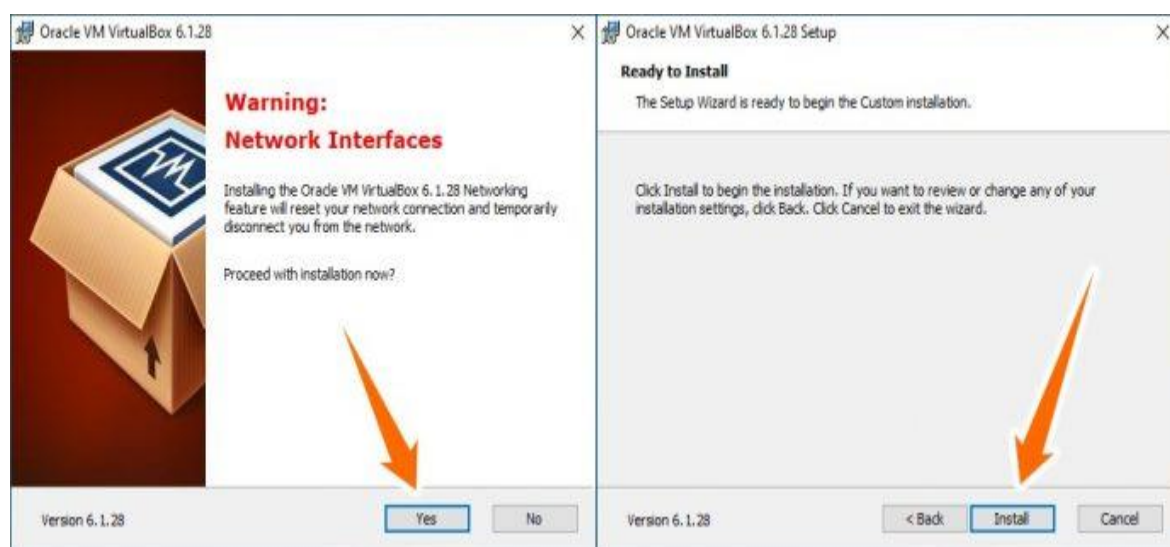


Étape 4 : Dans cette étape, vous pouvez modifier l'emplacement de l'installation en sélectionnant le Explorer languette. Ensuite, cliquez Suivant. L'écran suivant donne la possibilité de créer des

raccourcis. Cela montre également une option d'association de fichier de registre qui lie les fichiers créés par VirtualBox à lui-même.



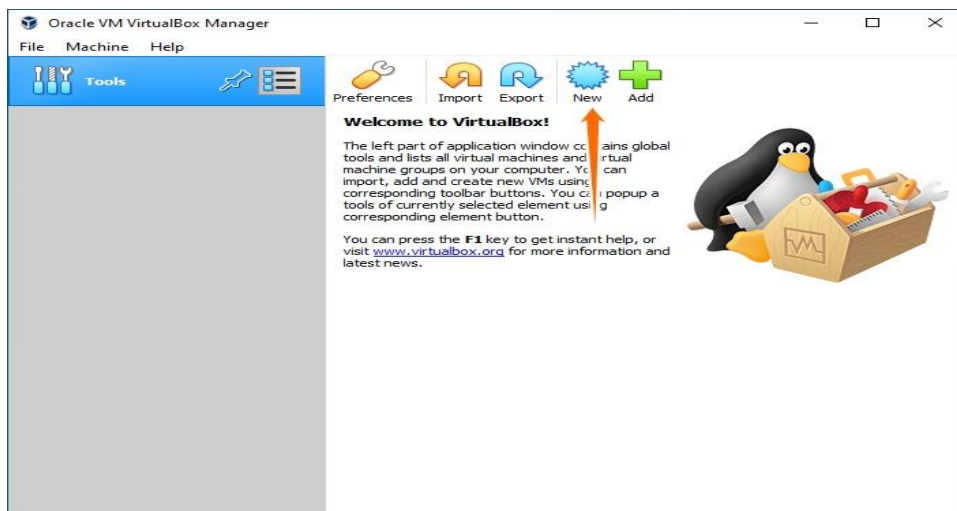
Étape 5 : Passez par la page d'avertissement. Ne vous inquiétez pas ; il va juste déconnecter votre PC pendant l'installation. Presse Oui, Puis cliquez sur Installer sur l'écran suivant pour commencer le processus d'installation.



Étape 6 : Ceci termine l'installation. Après cela, cliquez sur Finition pour lancer l'application.

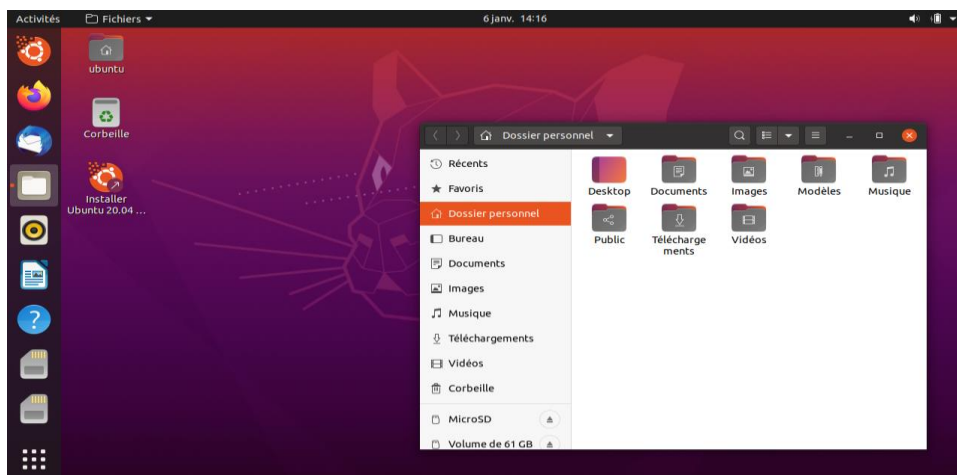


Étape 7 : Maintenant, vous pouvez ajouter n'importe quel système d'exploitation virtuel à partir du Nouveauté option.

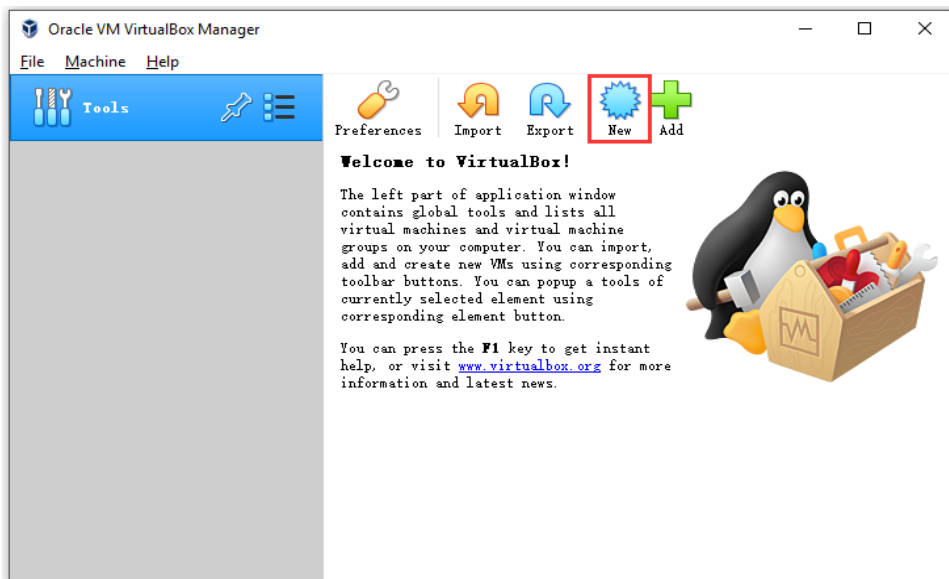


❖ Ubuntu

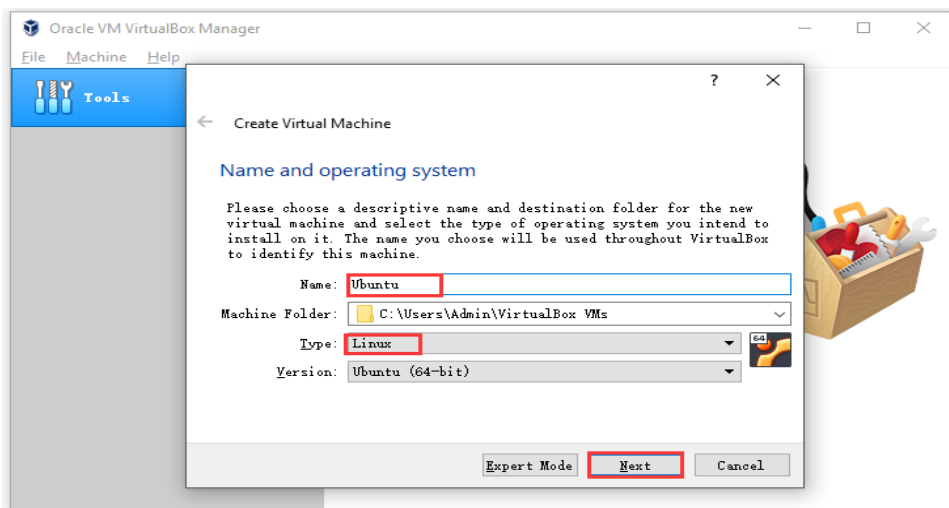
Une fois que le VirtualBox est installé, on installe Ubuntu. Ubuntu est un système d'exploitation GNU/Linux basé sur la distribution Debian. Il est libre, gratuit, et simple d'utilisation.



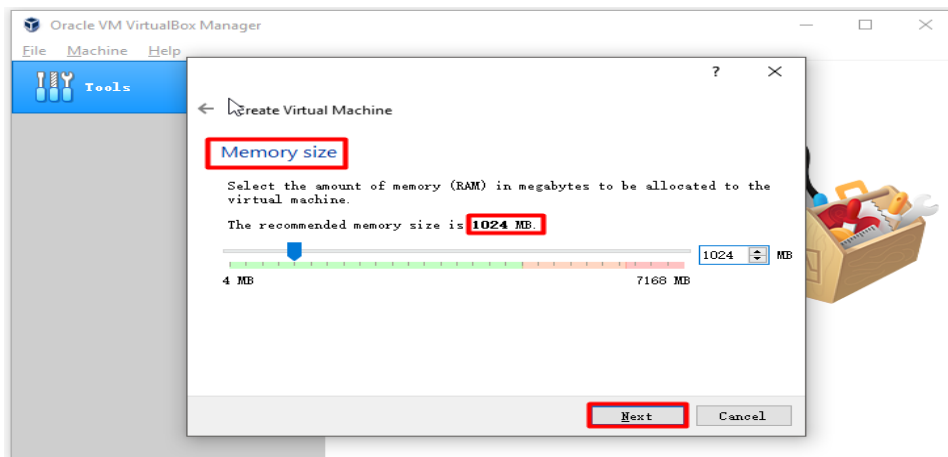
Étape 1 : Cliquez sur l'onglet Nouveau dans la fenêtre du Gestionnaire Oracle VM VirtualBox.



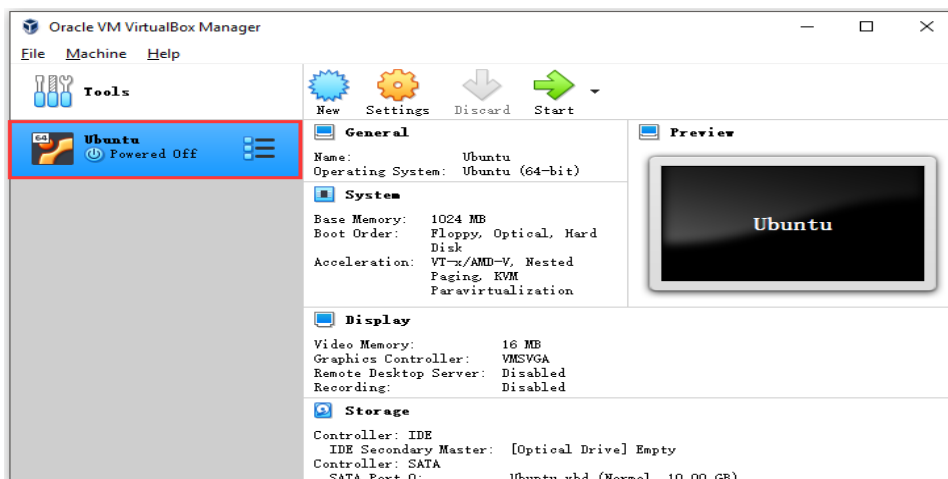
Étape 2 : Dans la fenêtre supérieure, tapez Ubuntu à la section Nom. Choisissez Linux comme Type et Ubuntu (64-bit) comme Version. Ensuite, cliquez sur Suivant.



Étape 3 : Dans cette fenêtre, réglez la taille de la mémoire. En général, celle-ci est réglée sur une valeur idéale. Si ce n'est pas le cas, vous devez la régler vous-même. Pour cela, reportez-vous à la taille recommandée. Ensuite, cliquez sur Suivant.



Étape 4 : Double-cliquez sur l'option Ubuntu dans le volet de gauche. Un menu s'ouvrira alors de lui-même.



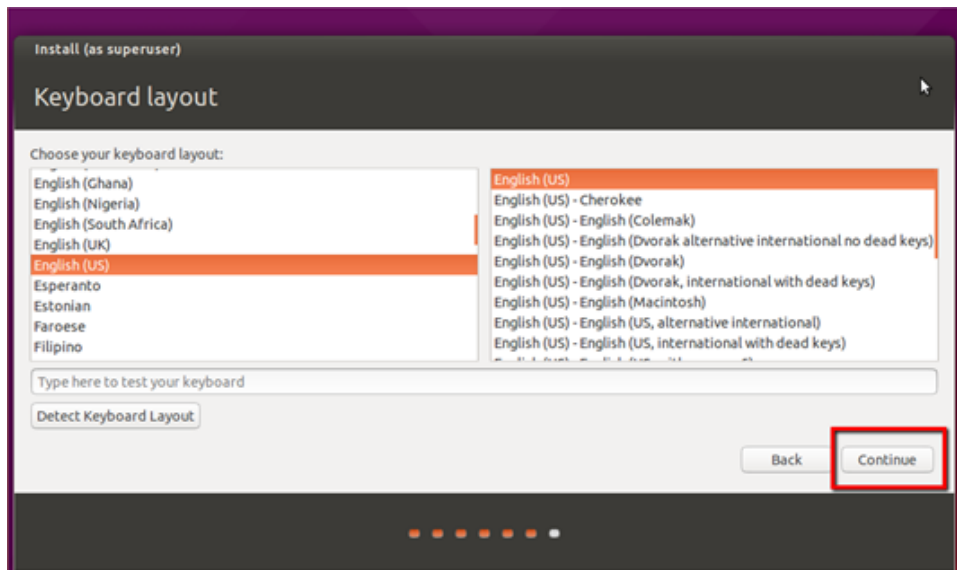
Étape 5 : Dans la fenêtre qui s'affiche, cliquez sur l'icône en bas à droite de la fenêtre. Sélectionnez le fichier ISO que vous avez stocké sur l'ordinateur et cliquez sur le bouton Ouvrir.

Étape 6 : Cliquez ensuite sur le bouton Démarrer pour continuer.

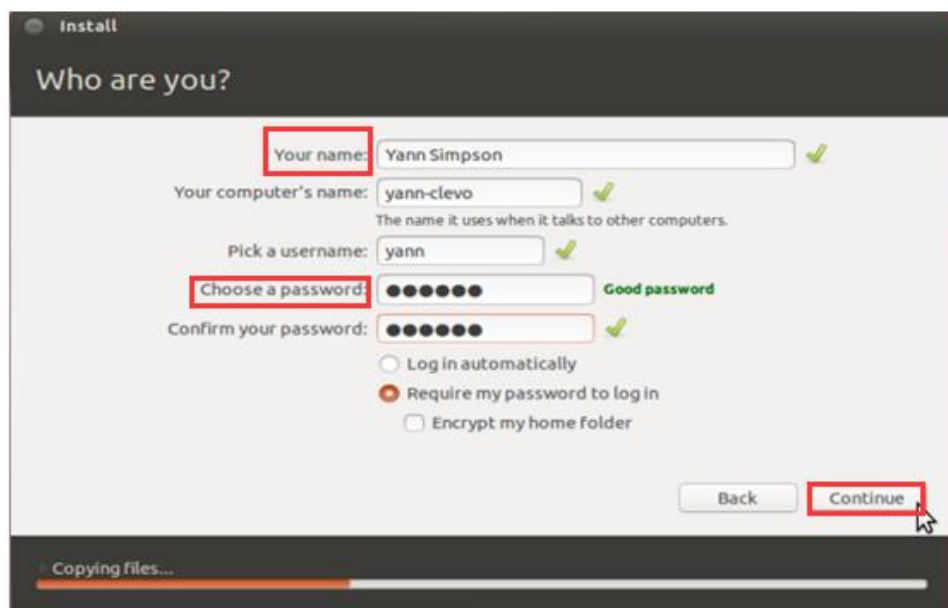
Étape 7 : Dans la fenêtre suivante, cliquez sur Installer Ubuntu.

Étape 8 : Après avoir choisi les deux options répertoriées, cliquez sur Continuer dans la fenêtre Préparation de l'installation d'Ubuntu.

Étape 9 : Une fenêtre s'ouvrira après l'installation d'Ubuntu. Choisissez un fuseau horaire correspondant à votre position actuelle et choisissez la configuration du clavier [ex : English (US)]. Cliquez ensuite sur Continuer pour démarrer le processus.



Étape 10 : Dans la fenêtre suivante, saisissez les informations correspondantes à l'endroit approprié, comme "Votre nom", "Mot de passe", "Nom d'utilisateur", etc. Après cela, cliquez sur Continuer pour poursuivre le processus.



Étape 11 : Maintenant, vous devez attendre patiemment jusqu'à la fin du processus. Ensuite, suivez les instructions à l'écran pour redémarrer la machine virtuelle. Dès que la VM redémarre normalement, vous pouvez l'utiliser. À partir de là, le processus d'installation de VirtualBox Ubuntu prend fin.

❖ MongoDB

MongoDB est un système de gestion de base de données orienté documents, répartissable sur un nombre quelconque d'ordinateurs et ne nécessitant pas de schéma prédéfini des données.



- Les étapes d'installation de MongoDB

Pour installer MongoDB, on clique sur ce lien <https://www.ionos.fr/digitalguide/sites-internet/developpement-web/installer-mongodb-sur-ubuntu/> et on suit les différentes étapes d'installation.

Étape 1 : importer la clé MongoDB

La première étape consiste à importer la clé publique GPG MongoDB. Pour ce faire, ouvrez d'abord le terminal. Ensuite, entrez la commande suivante pour télécharger la clé de la version 5.0 actuelle de MongoDB :

```
wget -qO - https://www.mongodb.org/static/pgp/server-6.0.asc | sudo apt-key add -
```

Après avoir confirmé, le processus d'importation devrait normalement se dérouler sans problème. Il se peut toutefois que le GNU Privacy Guard (GnuPG) ne soit pas encore installé sur votre système. Dans ce cas, un message d'erreur apparaît. Pour y remédier, il suffit d'installer le programme à l'aide de la commande de terminal suivante :

```
sudo apt-get install gnupg
```

Étape 2 : création du fichier de liste

L'étape suivante consiste à créer le fichier de liste correspondant à la version d'Ubuntu de votre appareil. Pour cela, vous pouvez également utiliser le terminal :

```
echo "deb [ arch=amd64,arm64 ] https://repo.mongodb.org/apt/ubuntu focal/mongodb-org/6.0 multiverse" | sudo tee /etc/apt/sources.list.d/mongodb-org-5.0.list
```

Afin que les modifications soient appliquées, il vous faudra ensuite redémarrer votre système. Le référentiel MongoDB sera ainsi ajouté à votre système. Ce processus peut prendre quelques minutes.

```
sudo apt-get update
```

Étape 3 : installer les paquets MongoDB

L'étape suivante consiste à installer les paquets nécessaires à la version de MongoDB que vous souhaitez exécuter. Dans la plupart des cas, il est préférable de choisir la version actuelle de MongoDB. La commande suivante suffit alors pour l'installation :

```
sudo apt-get install -y mongodb-org=5.0.15 mongodb-org-database= 5.0.15 mongodb-org-server= 5.0.15 mongodb-org-shard= 5.0.15 mongodb-org-tools= 5.0.15
```

Après quelques minutes, le processus d'installation se termine et MongoDB est officiellement installé.

Étape 4 : Démarrage de MongoDB

Une fois que vous avez installé avec succès la base de données NoSQL, la commande suivante permet de lancer le système : `sudo systemctl start mongod`

```
soumi@soumi-VirtualBox:~$ systemctl status mongod
● mongod.service - MongoDB Database Server
   Loaded: loaded (/lib/systemd/system/mongod.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2023-05-17 05:40:29 CET; 8h left
     Docs: https://docs.mongodb.org/manual
   Main PID: 843 (mongod)
    Memory: 1.2G
      CPU: 1min 25.144s
    CGroup: /system.slice/mongod.service
            └─843 /usr/bin/mongod --config /etc/mongod.conf

05:40:29 17 مئی soumi-VirtualBox systemd[1]: Started MongoDB Database Server.

soumi@soumi-VirtualBox:~$ mongo
MongoDB shell version v5.0.15
connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("67132559-78b3-40cf-acb5-cd2643016872") }
MongoDB server version: 5.0.15
=====
```

♦ Stockage des données avec mongodb

Création de base de données

Les environnements MongoDB fournissent aux utilisateurs un serveur pour créer des bases de données avec MongoDB. MongoDB stocke les données sous forme d'enregistrements constitués de collections et de documents.

Création de base de données droppy et voltix

```
> use droppy
switched to db droppy
> db
droppy
> show collections
> 
```

- db permet de vérifier avec quelle base de données on y est.
- Show collections permet d'afficher les collections de cette base de données
- Une collection est différente d'une table sql car elle permet de regrouper différents documents.

Création des collection mesure, indicateur et boitier

```

> db.createCollection("mesure")
{ "ok" : 1 }
> db.createCollection("indicateur")
{ "ok" : 1 }
> db.createCollection("boitier")
{ "ok" : 1 }
> show collections
boitier
indicateur
mesure
>

```

Importation des données droppy

```

soumi@soumi-VirtualBox:~$ mongoimport -d droppy -c mesure --type csv --file /home/soumi/droppy/mesure.csv --headerline
2023-05-17T00:16:21.876+0100 connected to: mongodb://localhost/
2023-05-17T00:16:24.877+0100 [.....] droppy.mesure 7.29MB/374MB (1.9%)
2023-05-17T00:16:27.876+0100 [.....] droppy.mesure 14.2MB/374MB (3.8%)
2023-05-17T00:16:30.876+0100 [#.....] droppy.mesure 21.8MB/374MB (5.8%)
2023-05-17T00:18:45.876+0100 [#####.] droppy.mesure 367MB/374MB (98.2%)
2023-05-17T00:18:48.383+0100 [#####] droppy.mesure 374MB/374MB (100.0%)
2023-05-17T00:18:48.383+0100 6441110 document(s) imported successfully. 0 document(s) failed to import.

```

Mongoimport permet d'importer des données dans la base de données droppy, de la collection mesure, le type de fichier(csv) via son emplacement. On constate que 6441110 documents ont été importé avec succès et 0 documents en échec.

♦ Manipulation des données

Dans un document, des champs peuvent être ajoutés, supprimés, modifiés et renommés à tout moment. Contrairement aux bases de données relationnelles, il n'y a pas de schéma prédéfini.

Les documents ou collections de documents MongoDB sont les unités de base des données. Formatés en JSON binaire (Java Script Object Notation), ces documents peuvent stocker différents types de données et être distribués sur plusieurs systèmes.

Étant donné que MongoDB utilise une conception de schéma dynamique, les utilisateurs disposent d'une flexibilité inégalée lors de la création d'enregistrements de données, de l'interrogation de collections de documents via l'agrégation MongoDB et de l'analyse de grandes quantités d'informations.

- Le nombre de document de la collection mesure, indicateur et boitier

```

> db.mesure.count() > db.indicateur.count() > db.boitier.count()
6441110 2 12

```

Pour avoir une lisibilité sur les données, MongoDB offre la possibilité qu'un champ contient plusieurs valeurs du coup on a modifié le champ indicateur et voici un exemple de ce dernier :

```

> db.mesure.update({"indicateur":1},{ $set:{"indicateur":{"id":1,"libelle":"Tension"}},{multi:true})
WriteResult({ "nMatched" : 729990, "nUpserted" : 0, "nModified" : 729990 })
>

```

On affiche la date et la consommation du 30 janvier 2023 en fonction heure, minute et second de la base de données droppey

```
> db.mesure.find({"indicateur.id":2,"date":{"$gte":"2023-01-30 00:00:00","$lte":"2023-01-31 00:00:00"}},
{"date":1,"val1":1})
{ "_id" : ObjectId("64640ec5a5420bc79fb3f1a4"), "date" : "2023-01-30 10:52:55", "val1" : 4 }
{ "_id" : ObjectId("64640ec5a5420bc79fb3f1a6"), "date" : "2023-01-30 11:50:57", "val1" : 4 }
{ "_id" : ObjectId("64640ec5a5420bc79fb3f1a8"), "date" : "2023-01-30 15:53:00", "val1" : 4 }
{ "_id" : ObjectId("64640ec5a5420bc79fb3f1aa"), "date" : "2023-01-30 15:53:06", "val1" : 4 }
{ "_id" : ObjectId("64640ec5a5420bc79fb3f1ac"), "date" : "2023-01-30 15:53:12", "val1" : 4 }
{ "_id" : ObjectId("64640ec5a5420bc79fb3f1ae"), "date" : "2023-01-30 15:53:19", "val1" : 4 }
{ "_id" : ObjectId("64640ec5a5420bc79fb3f1b0"), "date" : "2023-01-30 15:53:26", "val1" : 4 }
{ "_id" : ObjectId("64640ec5a5420bc79fb3f1b2"), "date" : "2023-01-30 15:53:33", "val1" : 4 }
{ "_id" : ObjectId("64640ec5a5420bc79fb3f1b4"), "date" : "2023-01-30 15:53:39", "val1" : 4 }
{ "_id" : ObjectId("64640ec5a5420bc79fb3f1b5"), "date" : "2023-01-30 15:53:47", "val1" : 4 }
{ "_id" : ObjectId("64640ec5a5420bc79fb3f1b7"), "date" : "2023-01-30 15:53:53", "val1" : 4 }
{ "_id" : ObjectId("64640ec5a5420bc79fb3f1b9"), "date" : "2023-01-30 15:54:00", "val1" : 4 }
```

Les documents d'une collection MongoDB peuvent comporter des champs différents.

- ♦ Le champ "_id" est un champ obligatoire, généré et ajouté par MongoDB, c'est un index unique qui permet d'identifier un document.

Cette commande permet d'afficher le 5 prélèvement qui ont eu plus de consommation.

```
> db.mesure.find({"indicateur.id":2}).limit(5).sort({"val1":-1})
{ "_id" : ObjectId("64640f0fa5420bc79fe49844"), "id" : 3188374, "date" : "2023-03-11 01:27:34", "datems" : NumberLong("1678501654369")
, "val1" : 2223, "boitier" : 3, "indicateur" : { "id" : 2, "libelle" : "consommation", "typeBoitier" : "Droppey" } }
{ "_id" : ObjectId("64640f42a5420bc79f0795bf"), "id" : 5481508, "date" : "2023-04-10 06:46:15", "datems" : NumberLong("1681112775366")
, "val1" : 1968, "boitier" : 3, "indicateur" : { "id" : 2, "libelle" : "consommation", "typeBoitier" : "Droppey" } }
{ "_id" : ObjectId("64640f0fa5420bc79fe49835"), "id" : 3188378, "date" : "2023-03-11 01:27:40", "datems" : NumberLong("1678501660608")
, "val1" : 865, "boitier" : 3, "indicateur" : { "id" : 2, "libelle" : "consommation", "typeBoitier" : "Droppey" } }
{ "_id" : ObjectId("64640f0fa5420bc79fe4982d"), "id" : 3188370, "date" : "2023-03-11 01:27:28", "datems" : NumberLong("1678501648159")
, "val1" : 349, "boitier" : 3, "indicateur" : { "id" : 2, "libelle" : "consommation", "typeBoitier" : "Droppey" } }
{ "_id" : ObjectId("64640f0fa5420bc79fe4983d"), "id" : 3188386, "date" : "2023-03-11 01:27:53", "datems" : NumberLong("1678501673146")
, "val1" : 341, "boitier" : 3, "indicateur" : { "id" : 2, "libelle" : "consommation", "typeBoitier" : "Droppey" } }
>
```

Remarque : on peut en déduire qu'il y avait eu une fuite d'eau le 11 mars 2023 à 1h

- Suppression d'un document dont l'id est 6254000

```
> db.mesure.remove({id:6254000})
WriteResult({ "nRemoved" : 1 })
>
```

❖ Hadoop

Hadoop est un Framework libre et open source écrit en Java destiné à faciliter la création d'applications distribuées et échelonnables permettant aux applications de travailler avec des milliers de nœuds et des pétaoctets de données.



- **Les étapes d'installation de Hadoop**

Étape 1 : Avant d'installer Hadoop, vous devez d'abord vous assurer que java8 est installé.

```
sudo add-apt-repository ppa:webupd8team/java
```

```
sudo apt-get update
```

```
sudo apt-get install oracle-java8-installer
```

Vérifiez que Java est correctement installé : `java -version`

Étape 2 : installez le mode Hadoop single node

Ajoutez d'abord un utilisateur Hadoop avec un accès administrateur

```
sudo addgroup hadoop
```

```
sudo adduser --ingroup hadoop hduser
```

```
sudo usermod -a -G sudo hduser
```

Puis connectez-vous avec cet utilisateur

Étape 3 : Installez SSH

```
sudo apt-get install openssh-server
```

Générez des clés SSH, il n'est donc pas nécessaire de saisir un mot de passe à chaque démarrage du processus Hadoop :

```
cat $HOME/.ssh/id_rsa.pub >> $HOME/.ssh/authorized_keys
```

Étape 4 : Téléchargez apache hadoop 3.3.4

cd Download

```
wget https://archive.apache.org/dist/hadoop/core/hadoop-3.3.4/hadoop-3.3.4.tar.gz
```

```
sudo tar -xvzf hadoop-3.3.4.tar.gz
```

```
sudo mv hadoop-3.3.4 /usr/local/hadoop
```

```
sudo chown hduser:hadoop -R /usr/local/hadoop
```

- Créez des répertoires Hadoop temp pour Namenode et Datanode

```
sudo mkdir -p /usr/local/hadoop_tmp/hdfs/namenode
```

```
sudo mkdir -p /usr/local/hadoop_tmp/hdfs/datanode
```

```
sudo chown hduser:hadoop -R /usr/local/hadoop_tmp/
```

Étape 5 : Configuration de Hadoop

- Mettez à jour bashrc: `sudo gedit .bashrc`

```
GNU nano 6.2 .bashrc
if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi

export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
export PATH=$PATH:/usr/lib/jvm/java-8-openjdk-amd64/bin
export HADOOP_HOME=~/hadoop-3.3.4/
export PATH=$PATH:$HADOOP_HOME/bin
export PATH=$PATH:$HADOOP_HOME/sbin
export HADOOP_MAPRED_HOME=$HADOOP_HOME
export YARN_HOME=$HADOOP_HOME
export HADOOP_CONF_DIR=$HADOOP_HOME/etc/hadoop
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib/native"
export HADOOP_STREAMING=$HADOOP_HOME/share/hadoop/tools/lib/hadoop-streaming-3.2.3.jar
export HADOOP_LOG_DIR=$HADOOP_HOME/logs
export PDSH_RCMD_TYPE=ssh

soumi@soumi-VirtualBox:~/hadoop-3.3.4/etc/hadoop$ ls
capacity-scheduler.xml      hadoop-user-functions.sh.example  kms-log4j.properties          ssl-client.xml.example
configuration.xml           hdfs-rbf-site.xml                kms-site.xml                  ssl-server.xml.example
container-executor.cfg      hdfs-site.xml                    log4j.properties             user_ec_policies.xml.template
core-site.xml               https-env.sh                      mapred-env.cmd               workers
hadoop-env.cmd              https-log4j.properties            mapred-env.sh                yarn-env.cmd
hadoop-env.sh               https-site.xml                    mapred-queues.xml.template   yarn-env.sh
hadoop-metrics2.properties  kms-acls.xml                      mapred-site.xml              yarnservice-log4j.properties
hadoop-policy.xml           kms-env.sh                        shellprofile.d                yarn-site.xml

soumi@soumi-VirtualBox:~/hadoop-3.3.4/etc/hadoop$

GNU nano 6.2 hadoop-env.sh
# Many of the options here are built from the perspective that users
# may want to provide OVERWRITING values on the command line.
# For example:
#
# JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
#
# Therefore, the vast majority (BUT NOT ALL!) of these defaults
# are configured for substitution and not append.  If append
# is preferable, modify this file accordingly.
```



```

GNU nano 6.2                                     core-site.xml
limitations under the License. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->

<configuration>
  <property>
    <name>fs.defaultFS</name>
    <value>hdfs://localhost:9000</value>  </property>
  <property>
    <name>hadoop.proxyuser.dataflair.groups</name> <value>*</value>
  </property>
  <property>
    <name>hadoop.proxyuser.dataflair.hosts</name> <value>*</value>
  </property>
  <property>
    <name>hadoop.proxyuser.server.hosts</name> <value>*</value>
  </property>
  <property>
    <name>hadoop.proxyuser.server.groups</name> <value>*</value>
  </property>
</configuration>

```

```

GNU nano 6.2                                     hdfs-site.xml
<!--
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->

<configuration>
  <property>
    <name>dfs.replication</name>
    <value>1</value>
  </property>
</configuration>

```

```

GNU nano 6.2                                     mapred-site.xml
    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->

<configuration>
  <property>
    <name>mapreduce.framework.name</name>  <value>yarn</value>
  </property>
  <property>
    <name>mapreduce.application.classpath</name>
    <value>${HADOOP_MAPRED_HOME}/share/hadoop/mapreduce/*:${HADOOP_MAPRED_HOME}/share/hadoop/mapreduce/lib/*</value>
  </property>
</configuration>

```

```

GNU nano 6.2                                yarn-site.xml
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License. See accompanying LICENSE file.
-->
<configuration>
  <property>
    <name>yarn.nodemanager.aux-services</name>
    <value>mapreduce_shuffle</value>
  </property>
  <property>
    <name>yarn.nodemanager.env-whitelist</name>
    <value>JAVA_HOME,HADOOP_COMMON_HOME,HADOOP_HDFS_HOME,HADOOP_CONF_DIR,CLASSPATH_PREP
  </property>
</configuration>

```

```

hduser@soumi-VirtualBox:~$ jps
28626 ResourceManager
31075 Jps
28102 NameNode
28217 DataNode
28380 SecondaryNameNode
28733 NodeManager
hduser@soumi-VirtualBox:~$

```

♦ Centralisations des données dans hdfs

Premièrement on créer un dossier SFM data dans hdfs pour préparer l'emplacement des données.

```

hduser@soumi-VirtualBox:~$ hdfs dfs -mkdir -p /user/sfmdata
hduser@soumi-VirtualBox:~$ hdfs dfs -mkdir -p /user/sfmdata/droppy
hduser@soumi-VirtualBox:~$
hduser@soumi-VirtualBox:~$ hdfs dfs -mkdir -p /user/sfmdata/voltix

```

Une fois qu'on a créé les dossiers, on charge les données dans le répertoire voltix contenu a hdfs.

```
hduser@soumi-VirtualBox:~$ hdfs dfs -put /home/hduser/voltix/mesure.csv /user/sfmdata/voltix
hduser@soumi-VirtualBox:~$ hdfs dfs -put /home/hduser/voltix/indicateur.csv /user/sfmdata/voltix
hduser@soumi-VirtualBox:~$ hdfs dfs -put /home/hduser/voltix/channel.csv /user/sfmdata/voltix
hduser@soumi-VirtualBox:~$ hdfs dfs -put /home/hduser/voltix/boitier.csv /user/sfmdata/voltix
```

Visualisation des données contenu dans Hdfs via le terminal

```
hduser@soumi-VirtualBox:~$ hdfs dfs -ls /user/sfmdata/voltix
Found 4 items
-rw-r--r--  1 hduser supergroup      1783 2023-05-25 03:14 /user/sfmdata/voltix/boitier.csv
-rw-r--r--  1 hduser supergroup     3086 2023-05-25 03:13 /user/sfmdata/voltix/channel.csv
-rw-r--r--  1 hduser supergroup       435 2023-05-25 03:13 /user/sfmdata/voltix/indicateur.csv
-rw-r--r--  1 hduser supergroup 433884021 2023-05-25 03:12 /user/sfmdata/voltix/mesure.csv
hduser@soumi-VirtualBox:~$
```

❖ Hive

Hive est un système de data warehousing qui permet d'interroger les gros datasets présents dans le HDFS. Hive est un logiciel d'analyse de données permettant d'utiliser Hadoop avec une syntaxe proche du SQL. Hive a été initialement développé par Facebook

Avant Hive, les développeurs étaient confrontés au défi de créer des tâches MapReduce complexes pour interroger les données Hadoop. Hive utilise la langage HQL (Hive Query Language), dont la syntaxe est semblable à celle de SQL. La plupart des développeurs ayant l'habitude des environnements et du langage SQL, ils sont rapidement à l'aise avec Hive.




◆ Les étapes d'installation

Pour configurer Apache Hive, vous devez d'abord télécharger et décompresser Hive. Ensuite, vous devez personnaliser les fichiers et paramètres suivants :

- ✓ Modifier le fichier .bashrc
- ✓ Modifier le fichier hive-config.sh
- ✓ Créer des répertoires Hive dans HDFS
- ✓ Configurer le fichier hive-site.xml
- ✓ Lancer la base de données Derby

Etape 1 : Téléchargement de hive

Pour télécharger hive, on clique sur ce lien <https://hive.apache.org/general/downloads/> .



GENERAL

- Home
- Downloads
- License
- Privacy Policy

DOCUMENTATION

- Language Manual
- Javadoc
- Wiki


COMMUNITY

- Becoming a Committer
- Edit Website
- How to Contribute
- Resources for contributors
- Issue Tracking
- Mailing Lists
- People

DEVELOPMENT

DOWNLOADS

Releases may be downloaded from Apache mirrors:

[Download a release now!](#) 

On the mirror, all recent releases are available, but are not guaranteed to be stable. For stable releases, look in the stable directory.

News

18 April 2020: release 2.3.7 available

This release works with Hadoop 2.x.y You can look at the complete [JIRA change log for this release](#).

26 August 2019: release 3.1.2 available

This release works with Hadoop 3.x.y. You can look at the complete [JIRA change log for this release](#).

Cette page fournit également des instructions utiles sur la façon de valider l'intégrité des fichiers récupérés à partir de sites miroirs.

News About Make a Donation The Apache Way Join Us Downloads



COMMUNITY-LED DEVELOPMENT "THE APACHE WAY"

Projects People Community License Sponsors

We suggest the following mirror site for your download:

<https://downloads.apache.org/hive/>

Other mirror sites are suggested below:

It is essential that you verify the integrity of the downloaded file using the PGP signature (`.asc` file) or a hash (`.md5` or `.sha1` file).

Please only use the backup mirrors to download KEYS, PGP signatures and hashes (SHA* etc) -- or if no other mirrors are working.

HTTP

<https://downloads.apache.org/hive/>

BACKUP SITES

Please only use the backup mirrors to download KEYS, PGP signatures and hashes (SHA* etc) -- or if no other mirrors are working.

<https://downloads.apache.org/hive/>

The full listing of mirror sites is also available.

Hadoop 3.3.4 est déjà installé sur le système Ubuntu présenté dans ce guide. Cette version Hadoop est compatible avec la version **Hive 3.1.2**.

Index of /hive

Name	Last modified	Size	Description
 Parent Directory		-	
 hive-1.2.2/	2018-05-04 20:51	-	
 hive-2.3.7/	2020-04-17 22:03	-	
 hive-3.1.2/ 	2019-08-26 20:21	-	
 hive-standalone-metastore-3.0.0/	2018-06-07 18:12	-	
 hive-storage-2.6.1/	2018-05-11 22:26	-	
 hive-storage-2.7.2/	2020-05-11 16:00	-	
 stable-2/	2020-04-17 22:03	-	
 KEYS	2020-01-17 22:47	82K	

Sélectionnez le fichier **apache-hive-3.1.2-bin.tar.gz** pour commencer le processus de téléchargement.

Index of /hive/hive-3.1.2

Name	Last modified	Size	Description
 Parent Directory		-	
 apache-hive-3.1.2-bin.tar.gz	2019-08-26 20:20	266M	
 apache-hive-3.1.2-bin.tar.gz.asc	2019-08-26 20:20	833	
 apache-hive-3.1.2-bin.tar.gz.sha256	2019-08-26 20:20	95	
 apache-hive-3.1.2-src.tar.gz	2019-08-26 20:20	24M	
 apache-hive-3.1.2-src.tar.gz.asc	2019-08-26 20:20	833	
 apache-hive-3.1.2-src.tar.gz.sha256	2019-08-26 20:20	95	

Une fois le processus de téléchargement terminé, décompressez-le package Hive compressé : `tar xzf apache-hive-3.1.2-bin.tar.gz`

Étape 2 : Configurer les variables d'environnement Hive (bashrc)

\$HIVE_HOME est une variable d'environnement doit permet de diriger le shell client vers le répertoire apache-hive-3.1.2-bin. Modifiez le fichier de configuration du shell .bashrc à l'aide d'un éditeur de texte de votre choix (nous utiliserons nano) :

```
sudo nano .bashrc
```

Ajoutez les variables d'environnement Hive suivantes au fichier .bashrc :

```
export HIVE_HOME="home/hadoop/apache-hive-3.1.2-bin"
export PATH=$PATH:$HIVE_HOME/bin
```

Étape 3 : Modifier le fichier hive-config.sh

Apache Hive doit pouvoir interagir avec le système de fichiers distribué Hadoop. Accédez au fichier hive-config.sh à l'aide de la variable créée précédemment \$HIVE_HOME :

```
sudo nano $HIVE_HOME/bin/hive-config.sh
```

Ajoutez la HADOOP_HOME variable et le chemin d'accès complet à votre répertoire Hadoop :

```
export HADOOP_HOME=/home/hadoop/hadoop-3.3.4
```

Étape 4 : Créer des répertoires Hive dans HDFS

Créer un répertoire d'entrepôt

Créez le répertoire de l'entrepôt dans le répertoire parent /user/hive/ :

```
hdfs dfs -mkdir -p /user/hive/warehouse
```

Ajoutez des autorisations d'écriture et d'exécution aux membres du groupe d'entrepôt :

```
hdfs dfs -chmod g+w /user/hive/warehouse
```

Étape 5 : Configurer le fichier hive-site.xml

```
GNU nano 4.8      hive-site.xml      Modified
<property>
  <name>hive.metastore.db.type</name>
  <value>DERBY</value>
  <description>
    Expects one of [derby, oracle, mysql, mssql, postgres].
    Type of database used by the metastore. Information schema & JDBCSto
  </description>
</property>
<property>
  <name>hive.metastore.warehouse.dir</name>
  <value>/user/hive/warehouse</value>
  <description>location of default database for the warehouse</description>
</property>
<property>
  <name>hive.metastore.warehouse.external.dir</name>
  <value/>
  <description>Default location for external tables created in the warehouse
</description>
</property>
<property>
  <name>hive.metastore.uris</name>
  <value/>
  <description>Thrift URI for the remote metastore. Used by metastore client
</description>
</property>
<property>
  <name>hive.metastore.uri.selection</name>
```

Étape 6 : Lancer la base de données Derby

```
$HIVE_HOME/bin/schematool -dbType derby -initSchema
```

Le processus peut prendre quelques instants.

```
Initialization script completed
schemaTool completed
hdoop@phoenixnap:~/apache-hive-3.1.2-bin/bin$
```

- ◆ Manipulation des données avec hive

Lancement de hive

```
hduser@soumi-VirtualBox:~/Bureau$ hive
Hive Session ID = 635a7b58-0d67-4448-81d1-868ee11f6402

Logging initialized using configuration in jar:file:/home/hduser/ecosystem/apache-hive-3.1.2-bin/lib/hive-common-3.1.2.jar!/hive-log4j2.properties Async: true
Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. spark, tez) or using Hive 1.X releases.
Hive Session ID = c638e26b-0531-43cc-bb7e-15fa75bed1cf
hive>
```

Création de la base de données droppey


```
hive> create database droppy;
OK
Time taken: 0.162 seconds
```

Lorsque la base de données est créée, on peut créer par la suite les tables et chargé les données.

```
hive> use droppy;
OK
Time taken: 0.081 seconds
hive> show tables;
OK
Time taken: 0.145 seconds
```

- Use permet d'accéder à la base de données
- Show tables permet d'afficher les tables de la base de données

Création des tables : création de la table mesure

```
hive> create table mesure
> (
>   mes_id bigint,
>   dates timestamp,
>   datems bigint,
>   val1 varchar(255),
>   boitier bigint,
>   indicateur bigint
> );
OK
Time taken: 0.429 seconds
```

Chargement des données

La commande load permet de charger le fichier mesure.csv contenu en local dans la table mesure.

```
hive> LOAD DATA LOCAL INPATH '/home/hduser/droppy/mesure.csv' OVERWRITE INTO TABLE mesure;
Loading data to table droppy.mesure
OK
Time taken: 25.598 seconds
```

Vérification dans le terminal

On peut vérifier en terminal les fichiers contenus dans hdfs crée par hive.

```
hduser@soumi-VirtualBox:~$ hdfs dfs -ls /user/hive/warehouse/droppy.db
Found 3 items
drwxr-xr-x - hduser supergroup          0 2023-05-22 13:19 /user/hive/warehouse/droppy.db/boitier
drwxr-xr-x - hduser supergroup          0 2023-05-22 13:39 /user/hive/warehouse/droppy.db/indicateur
drwxr-xr-x - hduser supergroup          0 2023-05-22 13:17 /user/hive/warehouse/droppy.db/mesure
hduser@soumi-VirtualBox:~$
```

Vérification au navigateur

On peut vérifier en terminal les fichiers contenus dans hdfs crée par hive.

localhost:9870/explorer.html#/user/hive/warehouse

Hadoop Overview Datanodes Datanode Volume Failures Snapshot Startup Progress Utilities

Browse Directory

/user/hive/warehouse Go!

Show 25 entries Search:

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
drwxr-xr-x	hduser	supergroup	0 B	May 23 14:36	0	0 B	droppy.db
drwxr-xr-x	hduser	supergroup	0 B	May 22 11:09	0	0 B	voltix.db

Showing 1 to 2 of 2 entries Previous 1 Next

Hadoop, 2023.

/user/hive/warehouse/droppy.db Go!

Show 25 entries Search:

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
drwxr-xr-x	hduser	supergroup	0 B	May 24 16:13	0	0 B	boitier
drwxr-xr-x	hduser	supergroup	0 B	May 24 10:55	0	0 B	indicateur
drwxr-xr-x	hduser	supergroup	0 B	May 24 11:52	0	0 B	mesure

Showing 1 to 3 of 3 entries Previous 1 Next

2.6 Manipulation des données

Hive est un outil d'infrastructure d'entrepôt de données permettant de traiter des données structurées dans Hadoop. Il réside au-dessus de Hadoop pour résumer le Big Data et facilite l'interrogation et l'analyse.

Hive transforme les requêtes HiveQL en travaux MapReduce ou Tez qui s'exécutent sur le Framework de planification des travaux distribués d'Apache Hadoop, Yet Another Resource Negotiator (YARN).

Il interroge les données stockées dans une solution de stockage distribué, comme le système de fichiers distribués Hadoop (HDFS). Hive stocke ses métadonnées de base de données et de table dans un métastore, qui est une base de données ou un magasin sauvegardé sur fichier qui permet une abstraction et une découverte faciles des données.

- 5 première consommation de la collection mesure

```
hive> select * from mesure where indicateur=2 Limit 5;
OK
2      2023-01-27 15:13:04      1674832384836      4      1      2
4      2023-01-27 15:13:16      1674832396270      40     1      2
6      2023-01-27 15:13:30      1674832410621      10     1      2
8      2023-01-30 10:52:55      1675075975249      4      1      2
10     2023-01-30 11:50:57      1675079457967      4      1      2
Time taken: 0.324 seconds, Fetched: 5 row(s)
```

- Les cinq premiers enregistrements du 30 janvier 2023

```
hive> select * from mesure where indicateur=2 and dates BETWEEN '2023-01-30 00:00:00' and '2023-01-31 00:00:00' Limit 5;
OK
8      2023-01-30 10:52:55      1675075975249      4      1      2
10     2023-01-30 11:50:57      1675079457967      4      1      2
12     2023-01-30 15:53:00      1675093980001      4      2      2
14     2023-01-30 15:53:06      1675093986392      4      2      2
16     2023-01-30 15:53:12      1675093992648      4      2      2
Time taken: 0.686 seconds, Fetched: 5 row(s)
```

Les différents cas d'utilisation en dropdy

On s'intéresse à calculer :

- ✓ La quantité d'eau consommée par jour.
- ✓ La quantité d'eau consommée par semaine
- ✓ La quantité d'eau consommée mensuelle
- ✓ La quantité d'eau consommée trimestrielle
- ✓ La quantité d'eau consommée en globale

Chaque cas de consommation est illustré par un exemple :

- ◆ La quantité d'eau consommée d'une journée (30 janvier 2023)

```
hive> select sum(val1) from mesure where indicateur=2 and dates BETWEEN '2023-01-30 00:00:00' and '2023-01-31 00:00:00';
Query ID = hduser_20230524121310_d1927487-7a76-4bb1-81d5-d579b176b229
Total jobs = 1
Hadoop job information for Stage-1: number of mappers: 2; number of reducers: 1
2023-05-24 12:14:32,930 Stage-1 map = 0%, reduce = 0%
2023-05-24 12:15:11,544 Stage-1 map = 50%, reduce = 0%, Cumulative CPU 19.05 sec
2023-05-24 12:15:22,686 Stage-1 map = 67%, reduce = 0%, Cumulative CPU 24.02 sec
2023-05-24 12:15:29,700 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 27.52 sec
2023-05-24 12:16:23,253 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 31.77 sec
MapReduce Total cumulative CPU time: 31 seconds 770 msec
Ended Job = job_1684859383422_0002
MapReduce Jobs Launched:
Stage-Stage-1: Map: 2 Reduce: 1 Cumulative CPU: 31.77 sec HDFS Read: 315096834 HDFS Write: 106 SUCCESS
Total MapReduce CPU Time Spent: 31 seconds 770 msec
OK
3756.0
Time taken: 200.773 seconds, Fetched: 1 row(s)
hive>
```

- ◆ La quantité d'eau consommée par semaine (le 06 au 12 février 2023)

```
hive> SELECT SUM(val1) FROM mesure WHERE indicateur=2 and dates BETWEEN '2023-02-06 00:00:00' and '2023-02-13 00:00:00';
Query ID = hduser_20230524122244_74bdc0c0-460b-4d33-89eb-f70693f0806f
Total jobs = 1
Hadoop job information for Stage-1: number of mappers: 2; number of reducers: 1
2023-05-24 12:23:12,295 Stage-1 map = 0%, reduce = 0%
2023-05-24 12:23:39,122 Stage-1 map = 50%, reduce = 0%, Cumulative CPU 17.62 sec
2023-05-24 12:23:44,399 Stage-1 map = 67%, reduce = 0%, Cumulative CPU 22.22 sec
2023-05-24 12:23:52,722 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 26.56 sec
2023-05-24 12:24:08,345 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 30.19 sec
MapReduce Total cumulative CPU time: 30 seconds 190 msec
Ended Job = job_1684859383422_0003
MapReduce Jobs Launched:
Stage-Stage-1: Map: 2 Reduce: 1 Cumulative CPU: 30.19 sec HDFS Read: 315096827 HDFS Write: 107 SUCCESS
Total MapReduce CPU Time Spent: 30 seconds 190 msec
OK
91575.0
Time taken: 87.418 seconds, Fetched: 1 row(s)
hive>
```

- ◆ La quantité d'eau consommée Mensuelle (du 01 au 31 mars 2023)

```
hive> SELECT SUM(val1) FROM mesure WHERE indicateur=2 and dates BETWEEN '2023-03-01 00:00:00' and '2023-04-01 00:00:00';
Query ID = hduser_20230524123254_26f5bfdd-b929-416b-aa42-dc3396120175
Total jobs = 1

Hadoop job information for Stage-1: number of mappers: 2; number of reducers: 1
2023-05-24 12:33:18,793 Stage-1 map = 0%, reduce = 0%
2023-05-24 12:33:41,373 Stage-1 map = 50%, reduce = 0%, Cumulative CPU 18.21 sec
2023-05-24 12:33:46,594 Stage-1 map = 67%, reduce = 0%, Cumulative CPU 22.17 sec
2023-05-24 12:33:51,775 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 26.77 sec
2023-05-24 12:34:07,428 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 30.33 sec
MapReduce Total cumulative CPU time: 30 seconds 330 msec
Ended Job = job_1684859383422_0004
MapReduce Jobs Launched:
Stage-Stage-1: Map: 2 Reduce: 1 Cumulative CPU: 30.33 sec HDFS Read: 315096834 HDFS Write: 108 SUCCESS
Total MapReduce CPU Time Spent: 30 seconds 330 msec
OK
370753.0
Time taken: 75.094 seconds, Fetched: 1 row(s)
hive>
```

- ◆ La quantité d'eau consommée par trimestre

On s'intéresse à savoir la quantité d'eau consommée par trimestre vu que la facture d'eau est délivrée en 3 mois (du 01 février au 01 Mai 2023). Ici, on a la consommation de février, mars et avril.

```
hive> SELECT SUM(val1) FROM mesure WHERE indicateur=2 and dates BETWEEN '2023-02-01 00:00:00' and '2023-05-01 00:00:00';
Query ID = hduser_20230524123919_3c840847-237d-4680-bb7e-6ff200ab17b3
Total jobs = 1
Launching Job 1 out of 1

Hadoop job information for Stage-1: number of mappers: 2; number of reducers: 1
2023-05-24 12:39:35,477 Stage-1 map = 0%, reduce = 0%
2023-05-24 12:39:56,548 Stage-1 map = 50%, reduce = 0%, Cumulative CPU 18.16 sec
2023-05-24 12:40:02,757 Stage-1 map = 67%, reduce = 0%, Cumulative CPU 23.84 sec
2023-05-24 12:40:07,966 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 28.19 sec
2023-05-24 12:40:15,256 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 31.58 sec
MapReduce Total cumulative CPU time: 31 seconds 580 msec
Ended Job = job_1684859383422_0005
MapReduce Jobs Launched:
Stage-Stage-1: Map: 2 Reduce: 1 Cumulative CPU: 31.58 sec HDFS Read: 315096834 HDFS Write: 108 SUCCESS
Total MapReduce CPU Time Spent: 31 seconds 580 msec
OK
937101.0
Time taken: 59.552 seconds, Fetched: 1 row(s)
hive>
```

- ❖ Spark

Spark est un Framework open source de calcul distribué. Il s'agit d'un ensemble d'outils et de composants logiciels structurés selon une architecture définie.



Les outils d'Apache Spark



Spark SQL : Spark SQL permet d'exécuter des requêtes en langage SQL pour charger et transformer des données. Le langage SQL est issu des bases de données relationnelles, mais dans Spark, il peut être utilisé pour traiter n'importe quelles données, quel que soit leur format d'origine.

- **Les étapes d'installation de spark**

Étape 1 : Avant d'installer Spark, vous devez d'abord vous assurer que java 8 est installé :

```
sudo add-apt-repository ppa:webupd8team/java
```

```
sudo apt-get update
```

```
sudo apt-get install oracle-java8-installer
```

Vérifiez que Java est correctement installé :

```
java -version
```

Étape 2 : Vérifiez que vous avez correctement installé hadoop sur votre ordinateur

Étape 3 : Téléchargez Apache Spark

Pour télécharger Spark, on clique sur ce lien <https://spark.apache.org/downloads.html>.

On choisit une version puis on clique sur télécharger

Étape 4 : Terminez le processus d'installation.

Décompresser le fichier une fois que le téléchargement est terminé.

```
tar -xf spark-3.3.2-bin-hadoop3.tgz
```

Créez un lien pour créer une installation :

```
sudo ln -s ~/spark-3.3.2-bin-hadoop2.6 /usr/local/spark
```

Éditez bashrc en utilisant cette ligne de commande :

```
sudo gedit .bashrc
```

Ajoutez ces lignes :

```
# - SPARK ENVIRONMENT VARIABLES START -#  
  
export SPARK_HOME=/usr/local/spark  
  
export PATH=$SPARK_HOME/bin:$PATH  
  
# — SPARK ENVIRONMENT VARIABLES END — #
```

Étape 5 : Testez l'installation

Exécutez cette ligne de commande : *spark-shell*

```
hduser@soumi-VirtualBox:~$ spark-shell  
23/05/25 03:32:39 WARN Utils: Your hostname, soumi-VirtualBox resolves to a loopback address: 127.0  
.1.1; using 10.0.2.15 instead (on interface enp0s3)  
23/05/25 03:32:39 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another address  
Setting default log level to "WARN".  
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).  
23/05/25 03:32:56 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform...  
using builtin-java classes where applicable  
Spark context Web UI available at http://10.0.2.15:4040  
Spark context available as 'sc' (master = local[*], app id = local-1684981981012).  
Spark session available as 'spark'.  
Welcome to  
  
  ____  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  
 / ___|| || | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |  
| |___| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |  
 \___|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|  
                                     version 3.3.2  
  
Using Scala version 2.12.15 (Java HotSpot(TM) 64-Bit Server VM, Java 1.8.0_202)  
Type in expressions to have them evaluated.  
Type :help for more information.  
  
scala> █
```

Une fois que les données sont présente à hdfs, on fait une configuration pour que Spark puisse accéder aux données. Pour cela on ouvre spark-defaults.conf puis on saisit spark.hadoop.fs.defaultFS hdfs://localhost:9000

```
soumi@soumi-VirtualBox:~$ nano spark-defaults.conf  
soumi@soumi-VirtualBox:~$ █  
  
GNU nano 6.2 spark-defaults.conf  
spark.hadoop.fs.defaultFS hdfs://localhost:9000
```

Création des sessions

D'abord on crée une session. SparkSession est le principal point d'entrée pour les fonctionnalités DataFrame et SQL

```
scala> import org.apache.spark.sql.SparkSession
import org.apache.spark.sql.SparkSession

scala> val spark=SparkSession.builder.appName("lecturecsv").getOrCreate()
23/05/25 03:38:34 WARN SparkSession: Using an existing Spark session; only runtime SQL configurations
will take effect.
spark: org.apache.spark.sql.SparkSession = org.apache.spark.sql.SparkSession@23ba044f
```

Importation des données puis on affiche que quatre premiers enregistrements.

```
scala> val chan=spark.read.option("header",true).option("inferSchema",true).csv("hdfs://localhost:9000/user/sfndata/
voltix/channel.csv")
chan: org.apache.spark.sql.DataFrame = [id: string, channel: string ... 1 more field]
```

```
scala> chan.show(4)
+---+-----+-----+
| id|channel|compteur_id|
+---+-----+-----+
|  1|      1|          1|
|  2|      2|          1|
|  3|      3|          1|
|  4|      4|          1|
+---+-----+-----+
only showing top 4 rows
```

Pour utiliser SQL, on crée d'abord une table temporaire sur DataFrame à l'aide de la fonction `createOrReplaceTempView()`. Une fois créée, cette table est accessible tout au long de la SparkSession en utilisant `sql()` et elle sera supprimée avec votre terminaison SparkContext.

```
scala> mes.createOrReplaceTempView("mesure")
```

En d'autres termes, Spark SQL apporte des requêtes SQL RAW natives sur Spark, ce qui signifie que vous pouvez exécuter le SQL ANSI traditionnel sur Spark Dataframe. Dans le didacticiel SQL, vous apprendrez en détail à utiliser SQL select, where, group by, join, union etc.

```
scala> val mesv1=spark.sql("select date,val1 from mesure where indicateur=6")
mesv1: org.apache.spark.sql.DataFrame = [date: string, val1: string]
```

```
scala> mesv1.show(5)
+-----+-----+
|          date|val1|
+-----+-----+
|2022-12-20 10:11:12|0.05|
|2022-12-20 10:12:01|0.05|
|2022-12-20 10:12:39|0.00|
|2022-12-20 10:13:11|0.00|
|2022-12-21 10:56:07|0.08|
+-----+-----+
only showing top 5 rows
```

Dans la base de données voltix, on s'intéresse à calculer :

- ✓ La consommation journalière ou d'une journée par étage donné

- ✓ La consommation Active et réactive d'un mois par un étage donné
- ✓ Faire une étude comparative de la consommation de chaque étage par tous les mois.
- ✓ Faire une étude comparative de la consommation de chaque étage par trimestre.

Consommation d'une journée

Ici, on a la consommation d'électricité du 13 janvier 2023

```
scala> val mesjour=spark.sql("SELECT SUM(m.val1+m.val2+m.val3+m.val4+m.val5+m.val6) as consjour FROM mesure m,channel c WHERE m.channel_id=c.id and c.channel=2 and m.indicateur=6 and m.date BETWEEN '2023-04-24 00:00:00' and '2023-04-25 00:00:00'")
mesjour: org.apache.spark.sql.DataFrame = [consjour: double]
```

```
scala> mesjour.show()
+-----+
|      consjour |
+-----+
| 23.139999999999997 |
+-----+
```

Consommation Mensuelle par un étage donnée

```
scala> val mesmois=spark.sql("SELECT SUM(m.val1+m.val2+m.val3+m.val4+m.val5+m.val6) as consmois FROM mesure m,channel c WHERE m.channel_id=c.id and c.channel=1 and m.indicateur=6 and m.date BETWEEN '2023-02-01 00:00:00' and '2023-03-01 00:00:00'")
mesmois: org.apache.spark.sql.DataFrame = [consmois: double]
```

```
scala> mesmois.show()
+-----+
| consmois |
+-----+
|      647.0 |
+-----+
```

Consommation Mensuelle de tous les étages

```
scala> val mesmtetage=spark.sql("SELECT SUM(m.val1+m.val2+m.val3+m.val4+m.val5+m.val6) as consmtetage FROM mesure m,channel c WHERE m.channel_id=c.id and (c.channel=1 or c.channel=2 or c.channel=3 or c.channel=4) and m.indicateur=6 and m.date BETWEEN '2023-03-01 00:00:00' and '2023-04-01 00:00:00'")
mesmtetage: org.apache.spark.sql.DataFrame = [consmtetage: double]
```

```
scala> mesmtetage.show()
+-----+
| consmtetage |
+-----+
|      2840.42 |
+-----+
```

Consommation par trimestre

```
scala> val mestrim=spark.sql("SELECT SUM(m.val1+m.val2+m.val3+m.val4+m.val5+m.val6) as constrim FROM mesure m,channel c WHERE m.channel_id=c.id and (c.channel=1 or c.channel=2 or c.channel=3 or c.channel=4) and m.indicateur=6 and m.date BETWEEN '2023-01-01 00:00:00' and '2023-03-01 00:00:00'")
mestrim: org.apache.spark.sql.DataFrame = [constrim: double]
```



```
scala> mestime.show()
+-----+
|          constrim|
+-----+
|5753.140000000001|
+-----+
```

❖ Power Bi Desktop

Power BI est une solution d'analyse de données de Microsoft. Il permet de créer des visualisations de données personnalisées et interactives avec une interface suffisamment simple pour que les utilisateurs finaux créent leurs propres rapports et tableaux de bord



Power BI Desktop

- Les étapes d'installation

Pour télécharger Power BI, accédez à votre navigateur et tapez-le

<https://powerbi.microsoft.com/fr-be/desktop/>.

Microsoft | Power BI

Présentation ▾ Produits ▾ Tarification Solutions ▾ Partenaires ▾ Ressources ▾ Recherche 🔍 Connectez-vous Essayez gratuitement

Plus ▾

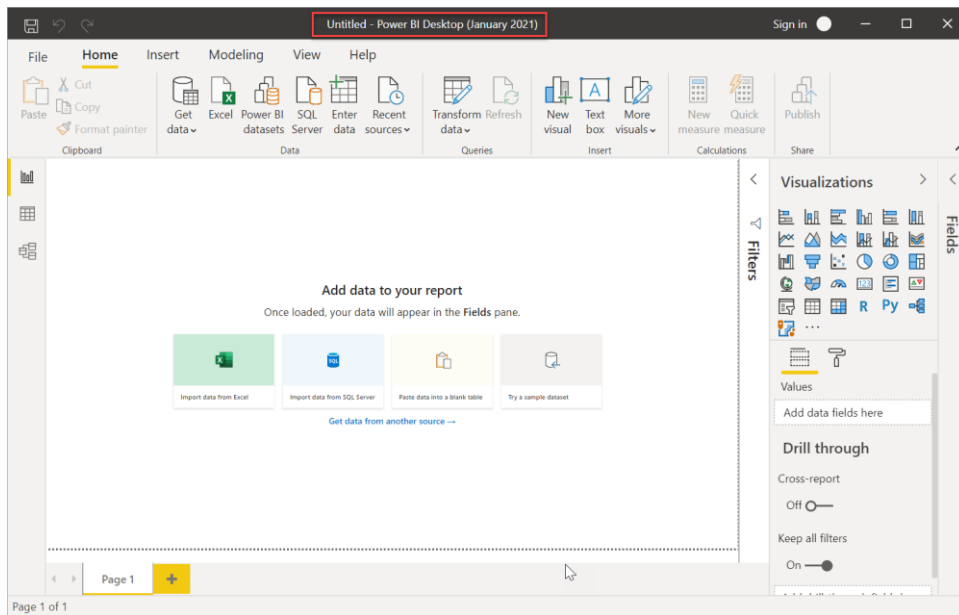
Achetez maintenant

Passez rapidement des données aux insights, puis à l'action avec Power BI Desktop

Créez des rapports interactifs enrichis avec des analyses de données visuelles à portée de main, gratuitement.

Téléchargez gratuitement > [Consultez les options de téléchargement ou de langue >](#)

Une fois que le téléchargement est terminé, taper ouvrir et on peut créer des visuels.



- ◆ Analyse avec power bi

Analyse en droppy

Pour faire des analyses :

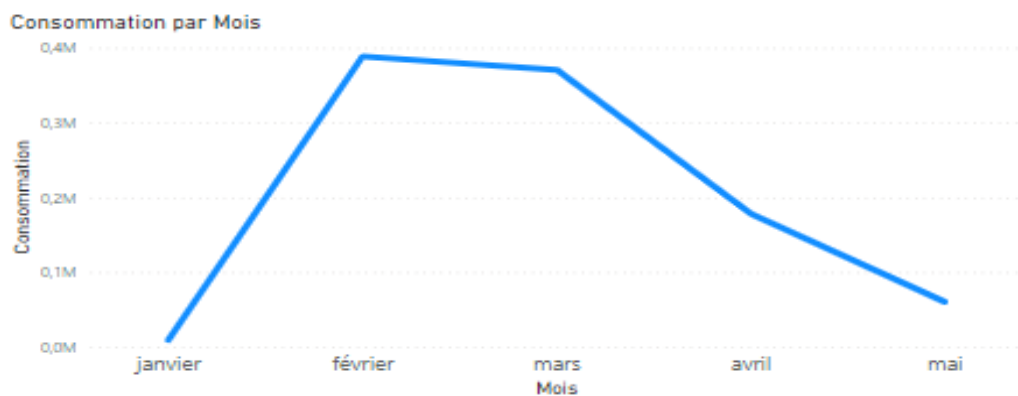
- On a effectué des analyses dans Py Spark
- Une fois qu'on a fait des analyses puis ont converti dataframe en format csv et on enregistre dans le local.

```
>>> mesdv.write.option("header",True).csv("mesdv")
>>>
```

Dans droppy, on s'intéresse :

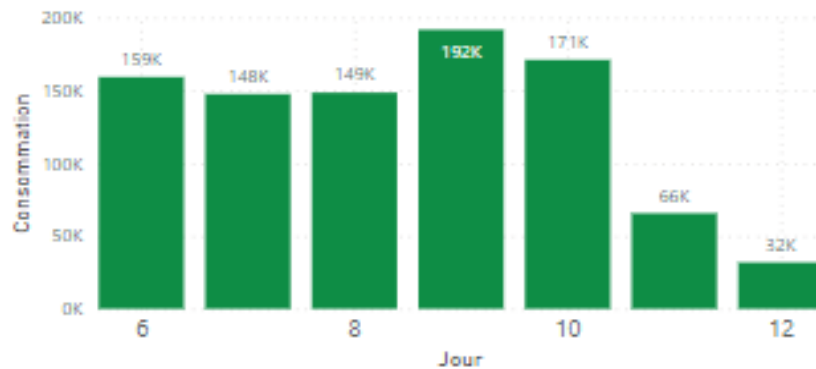
- ✓ La consommation globale d'eau
- ✓ La consommation par semaine
- ✓ La consommation mensuelle
- ✓ La consommation par trimestre

Consommation globale d'eau



Consommation d'eau par semaine du 6 au 12 février 2023

Consommation par Jour



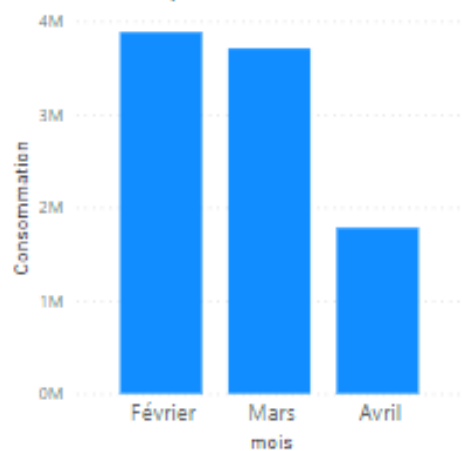
Consommation d'eau au mois de janvier

Consommation par Jour

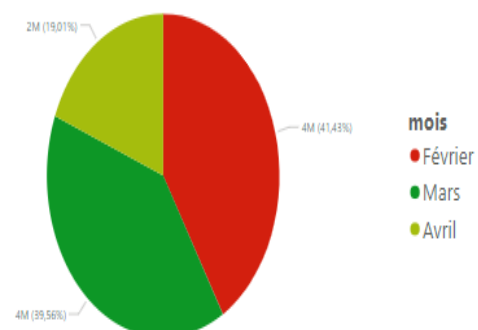


Consommation du février, mars et avril.

Consommation par mois



Consommation par mois

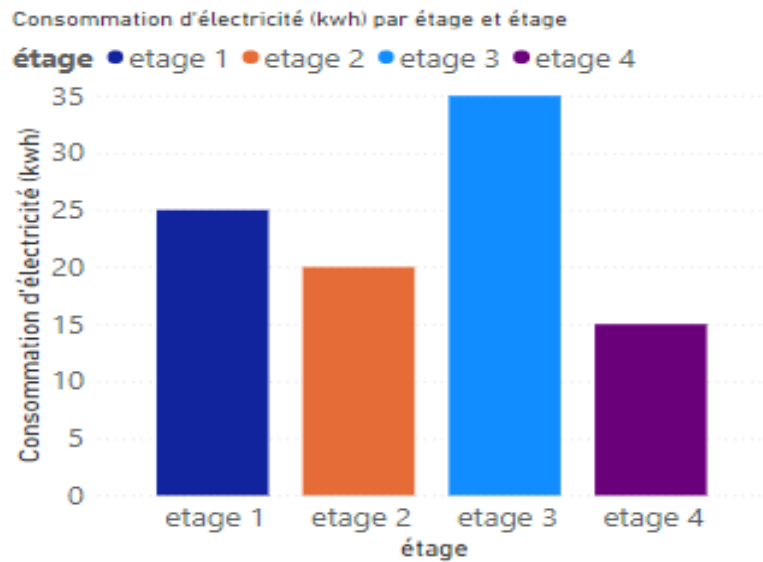


1.4 Analyse voltix

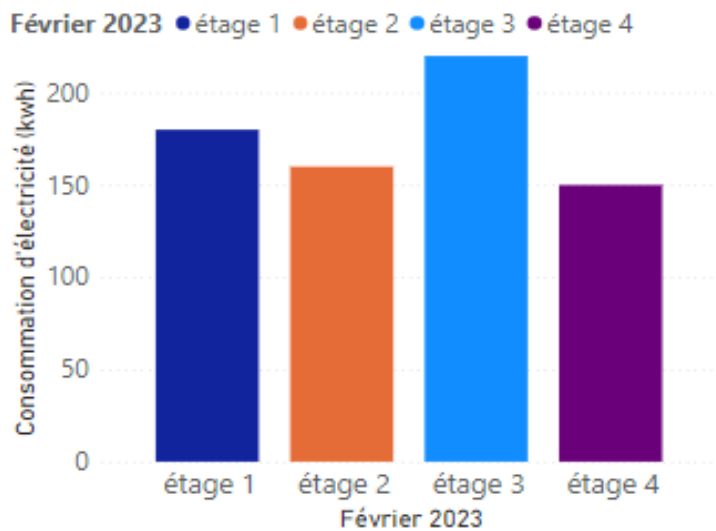
Dans voltix on s'intéresse analyser :

- ✓ La consommation d'une journée par tous les étages
- ✓ La consommation d'un mois par tous les étages
- ✓ La consommation globale de chaque mois pour tous les étages
- ✓ La consommation de deux trimestres

C'est la consommation du 13 janvier 2023.

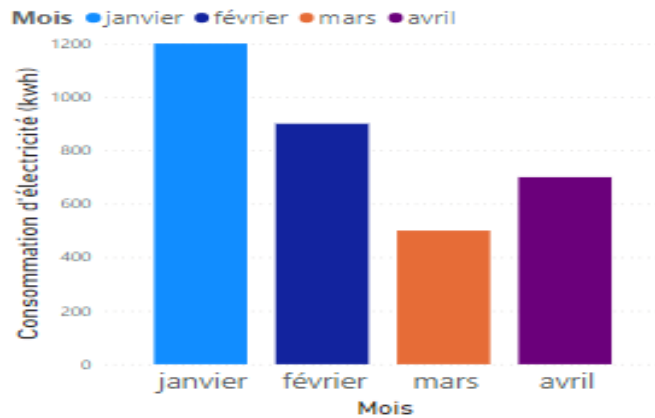


La consommation du février 2023 par tous les étages



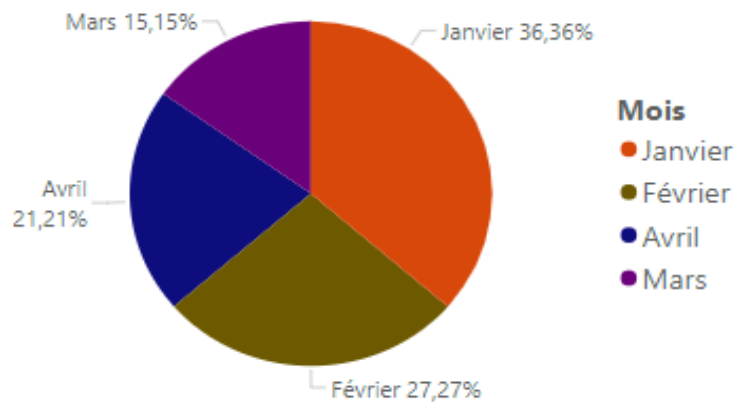
Consommation d'électricité de tous les mois

Consommation d'électricité (kwh) par Mois et Mois



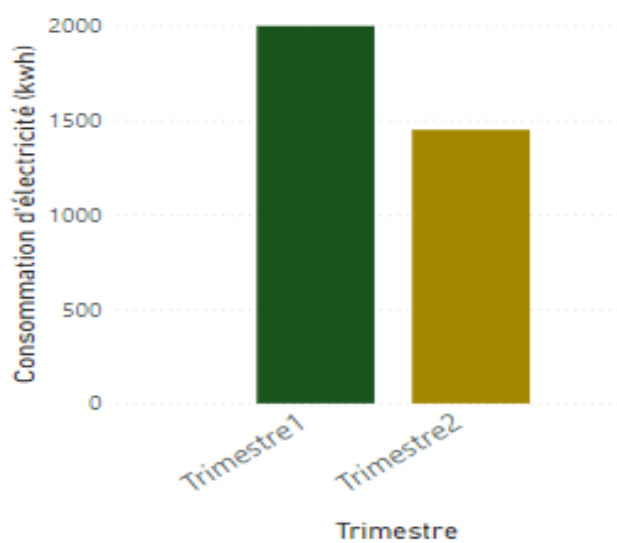
La consommation globale de tous les mois par diagramme circulaire

Consommation d'électricité (kwh) par mois



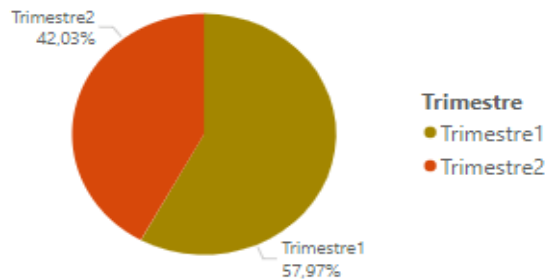
La consommation du deux trimestres. Le trimestre 1 correspond la consommation du janvier et février et le trimestre 2 correspond à la consommation du mars et avril par un diagramme a bar.

Trimestre ● Trimestre1 ● Trimestre2



La consommation d'électricité du deux trimestres. Le trimestre 1 correspond la consommation du janvier et février et le trimestre 2 correspond à la consommation du mars et avril par un diagramme circulaire.

Consommation d'électricité (kwh) par Trimestre



❖ Facturation

◆ Consommations d'eau (droppy)

Une fois qu'on sait la consommation d'eau par m^3 en fonction du jour, mois et semestre, on s'intéresse à savoir la facture. Une facture est établie après la consommation d'une période bien déterminé comme sonede la facture est délivrée après trois mois globalement.

Dans droppy, ont calculé :

- ✓ Facture par mois
- ✓ Facture globale (tous les mois)

C'est la facture estimative au mois du mars 2023



Facturation

Facture estimative d'eau

Période:

Quantité d'eau consommé : 79,55 m^3

La facture estimé en Mars est de 158,3 DT

On sait que la facture de l'eau est délivrée après trois mois. Pour cela on a estimé la facture d'après la consommation d'eau en m^3 .

Facturation

Facture estimative d'eau

Période:

Quantité d'eau consommé : 232,78 m³

La facture estimé par Tous les mois est de 463,2 DT



◆ Voltix

Dans voltix, on a calculé :

- ✓ Facture par mois
- ✓ Facture par trimestre
- ✓ Facture globale (tous les mois)

C'est la facture d'Avril 2023

Facturation

Facture estimative d'électricité

Période:

Quantité d'électricité consommé : 775 Kilowatt

La facture estimé en Avril est de 626,98 DT



C'est la facture du trimestre 1 correspond au janvier et février

Facturation

Facture estimative d'électricité

Période:

Quantité d'électricité consommé : 2100 Kilowatt

La facture estimé en Trimestre1 est de 1650,9 DT



On sait que la facture du STEG est délivrée après quatre mois. De ce fait on a effectué une estimation de la facture d'après la consommation en m³ de tous les mois (janvier, février, mars et avril).

Facturation

Facture estimative d'électricité

Période:

Quantité d'électricité consommé : 3510 Kilowatt

La facture estimé par Tout les mois est de 2791,6 DT



❖ Orange Data Mining

Orange est un logiciel libre d'exploration de données. Il propose des fonctionnalités de modélisation à travers une interface visuelle, une grande variété de modalités de visualisation et des affichages variés dynamiques.



◆ Les étapes d'installation

D'abord, il faut se lancer sur google puis cliquer ce lien : <https://orangedatamining.com/>

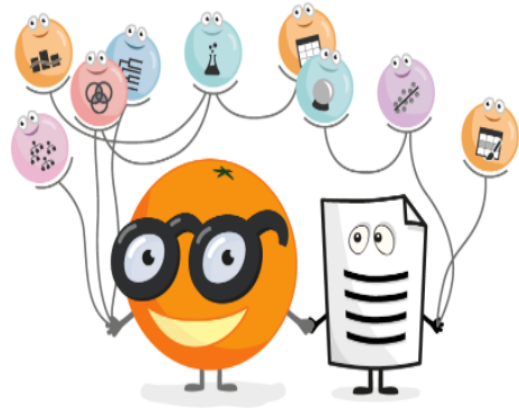
Une fois que vous êtes dans cet interface, cliquer sur télécharger.

Exploration de données Fructueux et amusant

Apprentissage automatique et visualisation de données open source.

Créez visuellement des flux de travail d'analyse de données, avec une boîte à outils vaste et diversifiée.

Télécharger Orange



Pour télécharger orange, c'est en fonction de système d'exploitation que vous désirez travailler.

Si vous utilisez Windows, taper les fenêtres puis télécharger la dernière version.

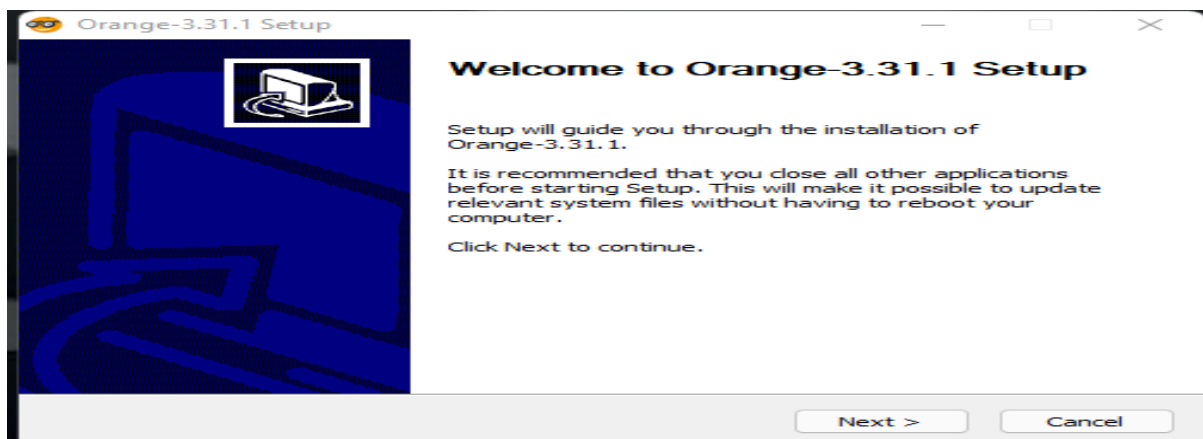


les fenêtres

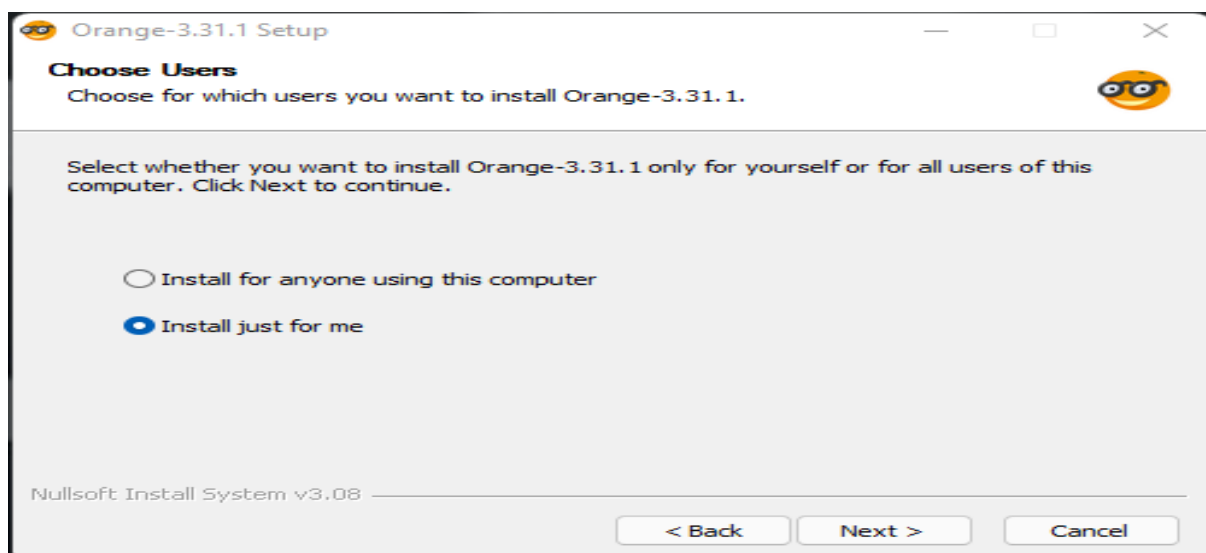
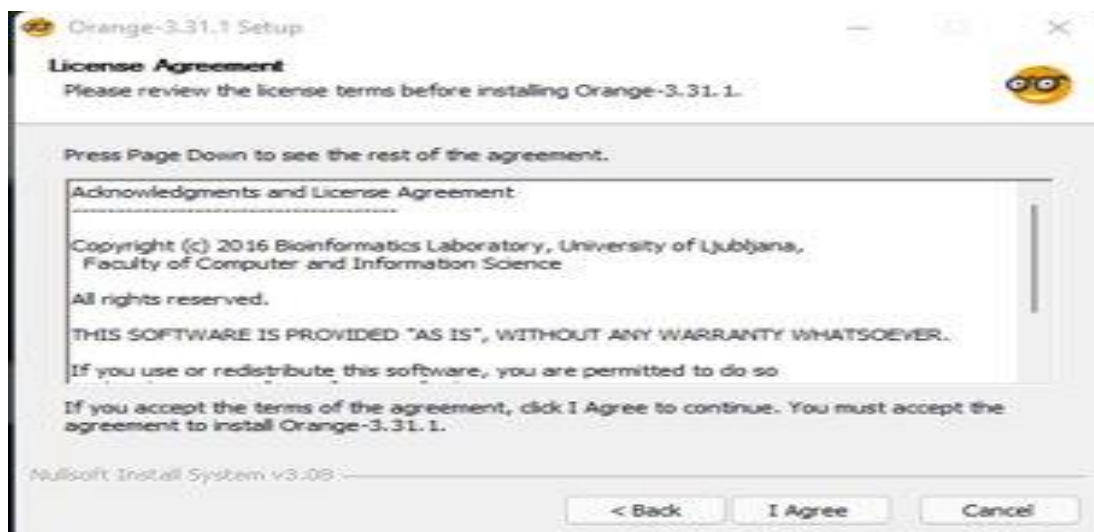
Téléchargez la dernière version pour Windows

Télécharger Orange 3.31.1

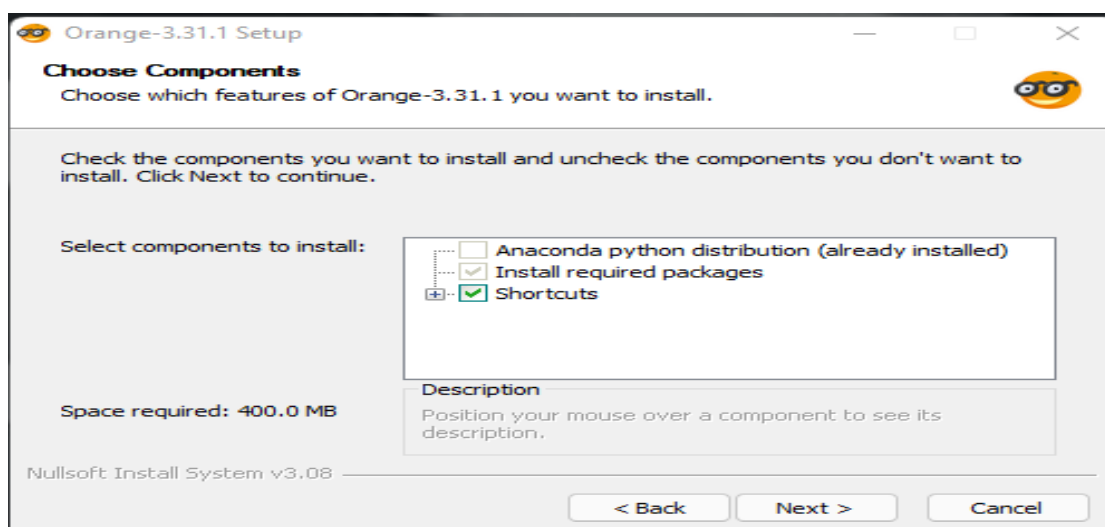
Cliquer sur l'icône .exe, une fois que le chargement se termine, y a cet interface qui s'affiche puis cliquer sur Next.

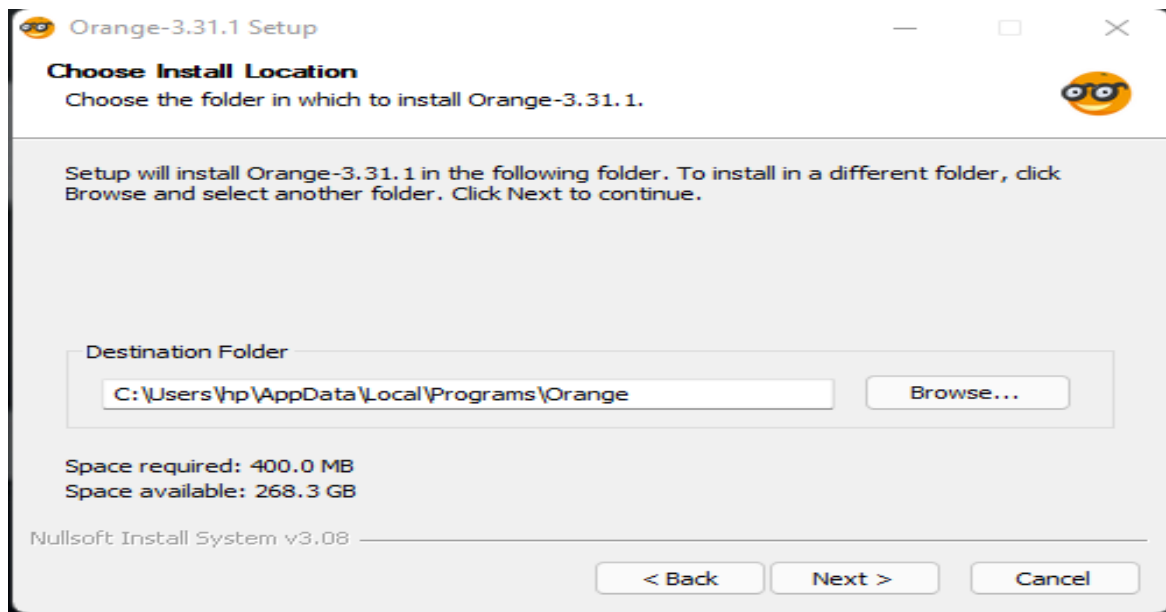


Faut cliquer I Agree puis Next.

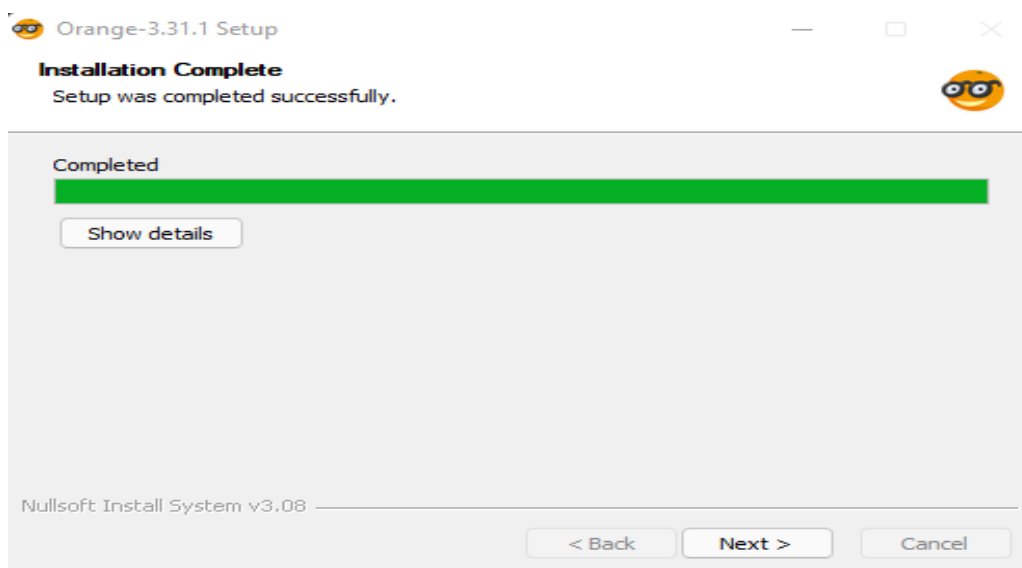
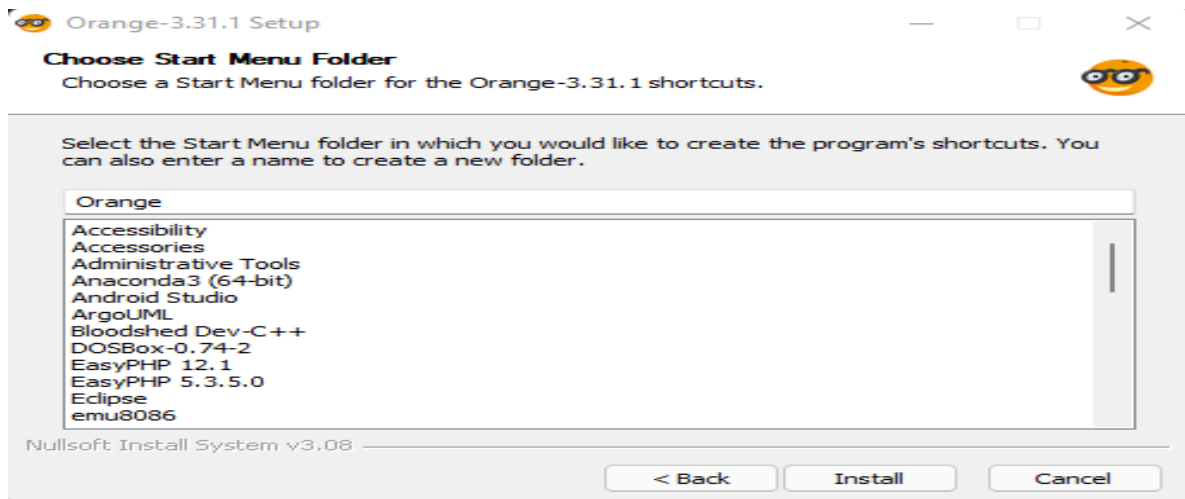


Il faut cliquer sur Next puis Next.

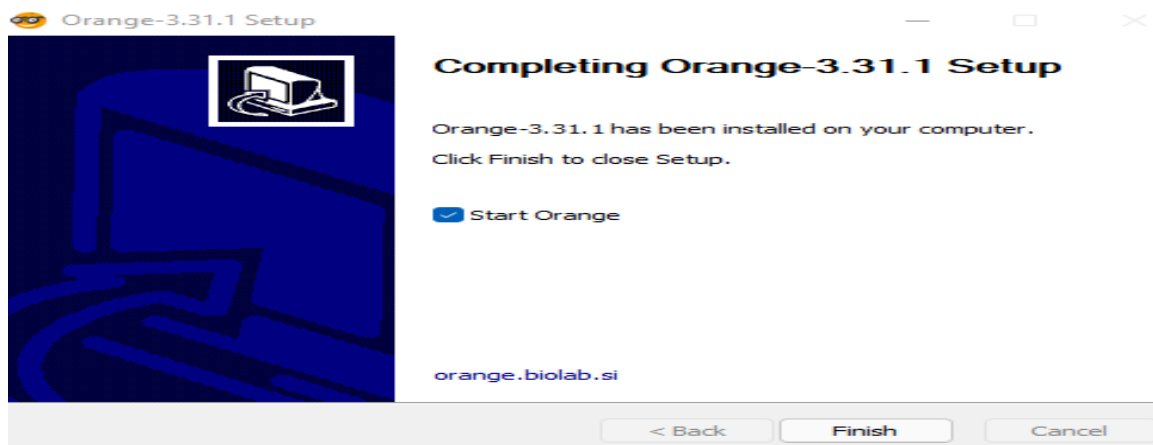




Cliquer sur Install, une fois que le chargement est terminée, il faut cliquer sur Next.



Faut cliquer sur finish, une fois que l'installation est bel et bien terminée, on aura l'ouverture du logiciel.

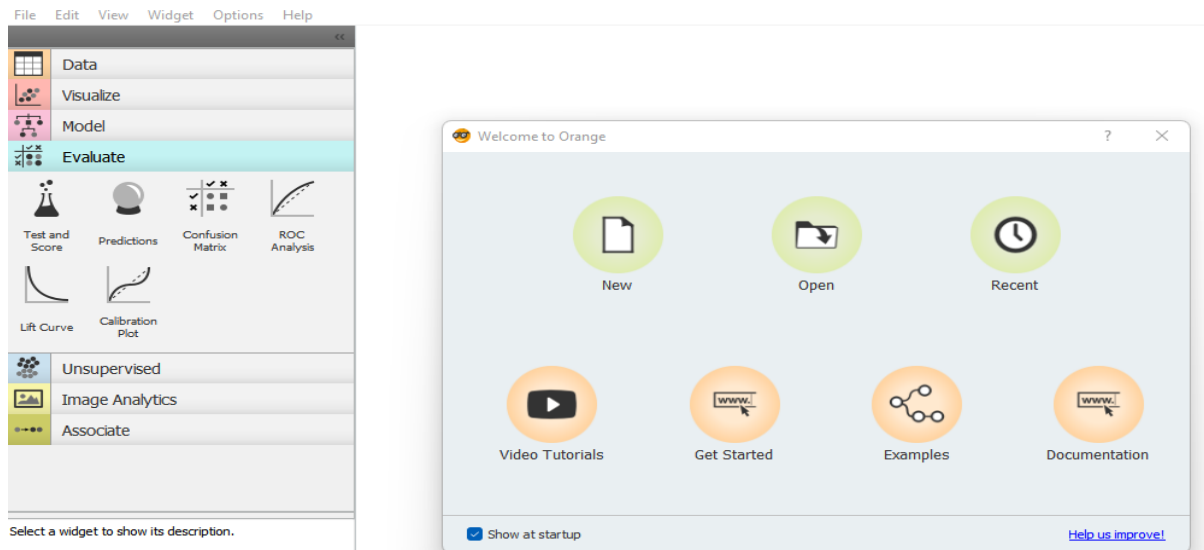


❖ Prédiction de la consommation avec orange

L'environnement d'orange data mining se décompose en deux parties et un entête :

- La palette de widgets (à gauche) : chaque widget va vous permettre d'effectuer des opérations sur les données
- Un canevas (à droite) dans lequel vous allez disposer vos widgets et les enchaîner.

Ouvrez Orange et vous devriez pouvoir voir l'interface utilisateur suivante :



Catégories de widget

La différence est que le premier vous permettra de concevoir graphiquement tandis que le second nécessite des connaissances Python ! Pas de code donc avec Orange. C'est une approche totalement graphique dans laquelle on relie point à point des widgets entres eux.

On trouve plusieurs catégories de widgets dans la palette :

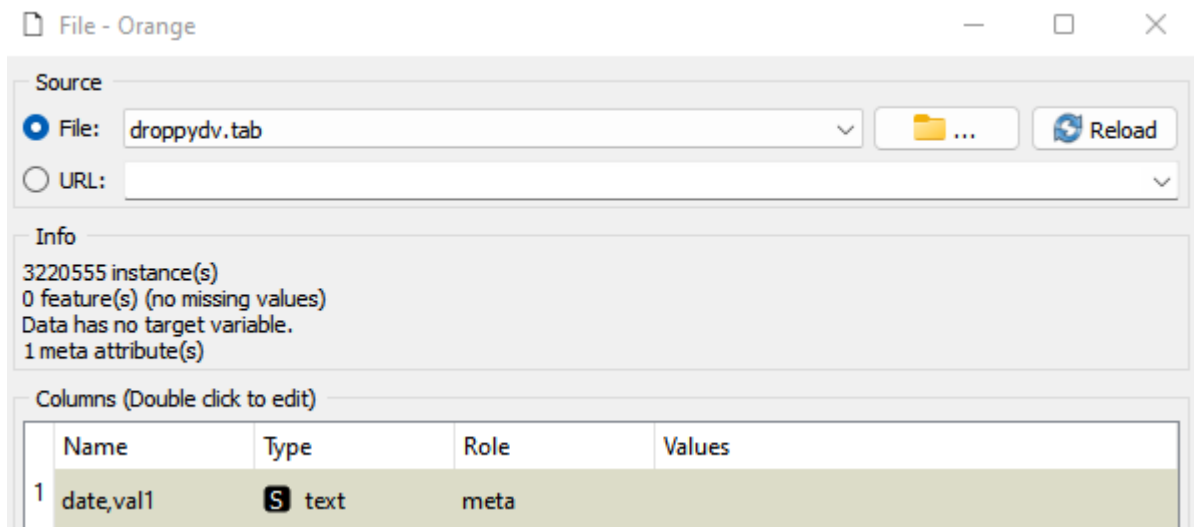
- ◆ **Data** : pour se connecter à des sources de données, analyser et effectuer des opérations sur ces dernières.
- ◆ **Visualize** : pour mieux voir les données au travers de graphiques (bars, scatter plots, etc.)
- ◆ **Model** : Pour modéliser et gérer la persistance de vos modèles.
- ◆ **Evaluate**: Pour prédire et qualifier ses modèles.
- ◆ **Unsupervised** : pour la modélisation non supervisée
- ◆ **Associate** : widgets pour l'extraction d'item sets fréquents et l'apprentissage des règles d'association.
- ◆ **Analyse d'image** : widgets pour travailler avec des images et des intégrations Image Net.

Orange est proposé à la fois aux utilisateurs expérimentés et aux analystes en data mining et machine learning qui souhaitent créer et tester leurs propres algorithmes en réutilisant le maximum de code, et à ceux qui débutent sur le terrain et qui peuvent soit écrire des contenus pythons courts pour les données analyse.

3.5 Prédiction de la consommation

Une fois qu'on sait la consommation d'eau et d'électricité, on s'intéresse à savoir la consommation future. Pour cela on utilisera orange pour la prédiction de la consommation.

D'abord on importe les données de dropdy correspondant à la consommation d'eau.



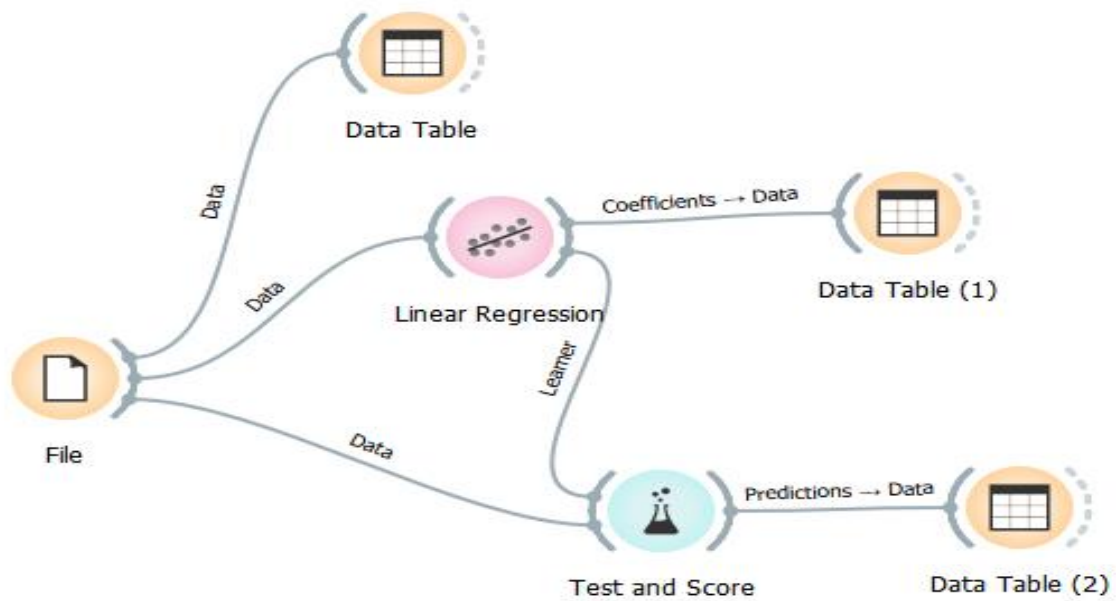
On peut visualiser les données importées via data table.

The screenshot shows the 'Data Table' widget in the Orange Data Mining interface. The 'Info' section displays: 3220555 instances (no missing data), No features, No target variable, and 1 meta attribute. The 'Variables' section has 'Show variable labels (if present)' checked. The 'Selection' section has 'Select full rows' checked. The data table shows the following rows:

	date,val1
1	2023-01-27 15:13:04,4
2	2023-01-27 15:13:16,40
3	2023-01-27 15:13:30,10
4	2023-01-30 10:52:55,4
5	2023-01-30 11:50:57,4
6	2023-01-30 15:53:00,4
7	2023-01-30 15:53:06,4
8	2023-01-30 15:53:12,4
9	2023-01-30 15:53:19,4
10	2023-01-30 15:53:26,4

Une fois qu'on a les données dans orange data mining on a appliqué la régression linéaire. La régression linéaire est une technique d'analyse de données qui prédit la valeur de données inconnues en utilisant une autre valeur de données apparentée et connue.

Dans notre cas, on applique la régression linéaire sur la consommation d'eau afin de prévoir la consommation future.

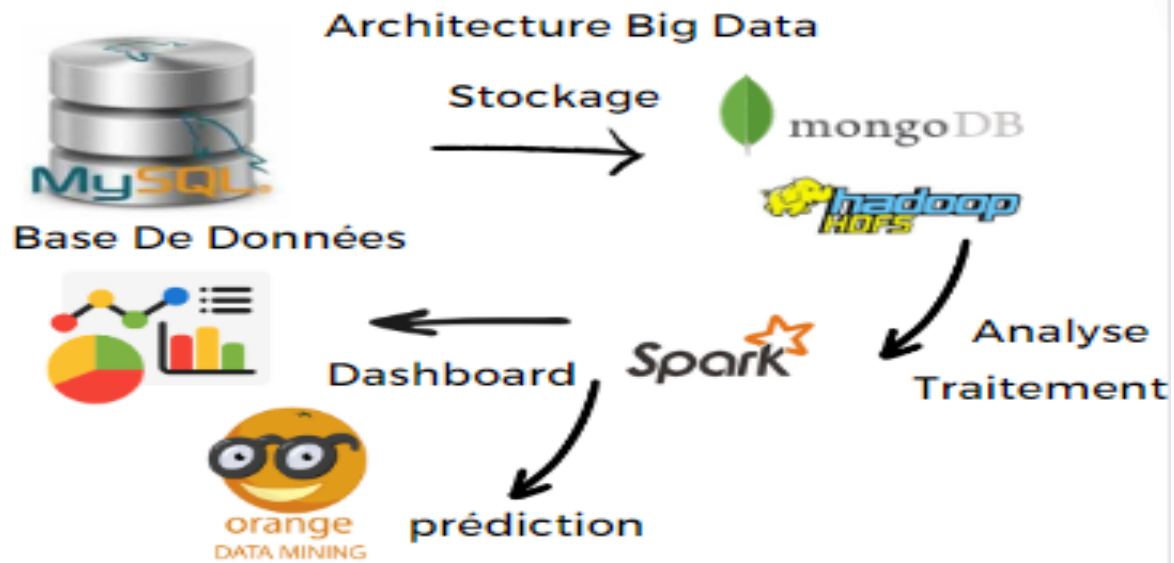


val1 correspond à la consommation actuelle de l'eau alors que linear Regression correspond à la consommation prédite

Data Table (2) - Orange

Info					
27195 instances (no missing data)					
1 feature					
Numeric outcome					
2 meta attributes					
Variables					
<input checked="" type="checkbox"/> Show variable labels (if present)					
<input type="checkbox"/> Visualize numeric values					
<input checked="" type="checkbox"/> Color by instance classes					
Selection					
<input checked="" type="checkbox"/> Select full rows					
	val1	Linear Regression	Fold	date	
9059	34.0	38.2451	2	2023-03-21 10:0...	
9060	55.0	38.2452	2	2023-03-21 10:0...	
9061	11.0	38.2461	2	2023-03-21 10:1...	
9062	30.0	38.2462	2	2023-03-21 10:1...	
9063	33.0	38.2503	2	2023-03-21 11:0...	
9064	33.0	38.2504	2	2023-03-21 11:0...	
9065	33.0	38.2504	2	2023-03-21 11:0...	
9066	33.0	38.2504	2	2023-03-21 11:0...	
9067	33.0	38.2512	2	2023-03-21 11:1...	
9068	48.0	38.2512	2	2023-03-21 11:1...	

Architecture Finale



On a adopté cette architecture a SFM. On avait étudié la limite de l'architecture existante a SFM qui consistait à stocker les données dans une base de données relationnel (MySQL). Pour remédier à ces limites on avait proposé MongoDB pour stocker les données puis centraliser les données dans hdfs. Après la comparaison des différents outils de traitement et d'analyse (MongoDB, Hive, Spark), Spark est la solution idéale pour le traitement et l'analyse de données pour SFM connect. On avait effectué de Dashboard avec power bi desktop pour avoir plus de lisibilité sur la consommation d'eau et d'électricité. Finalement on avait prédit la consommation avec orange data mining.