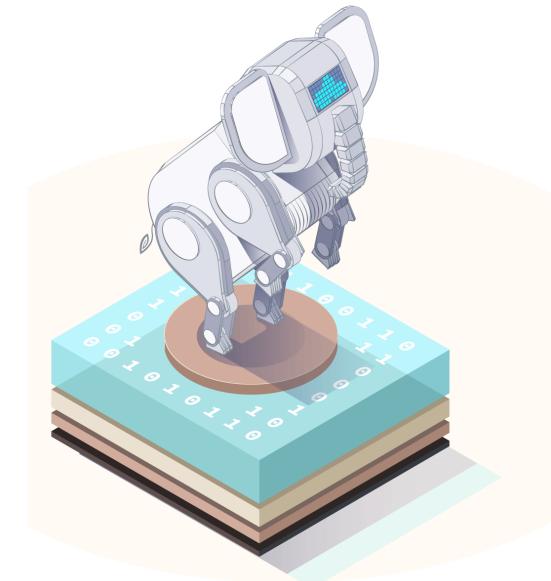


OVH Academy Lab Guide

Discovery scenario

OVH Analytics Data Platform

We will size and deploy a big data Hadoop cluster, then perform a use case allowing us to discover basic features : user and rights, dashboard, ingest data, queries on data.



ALL the credentials, useful links, and code,
We be available on

<https://github.com/baaastijn/workshops>

Discover the ADP product !

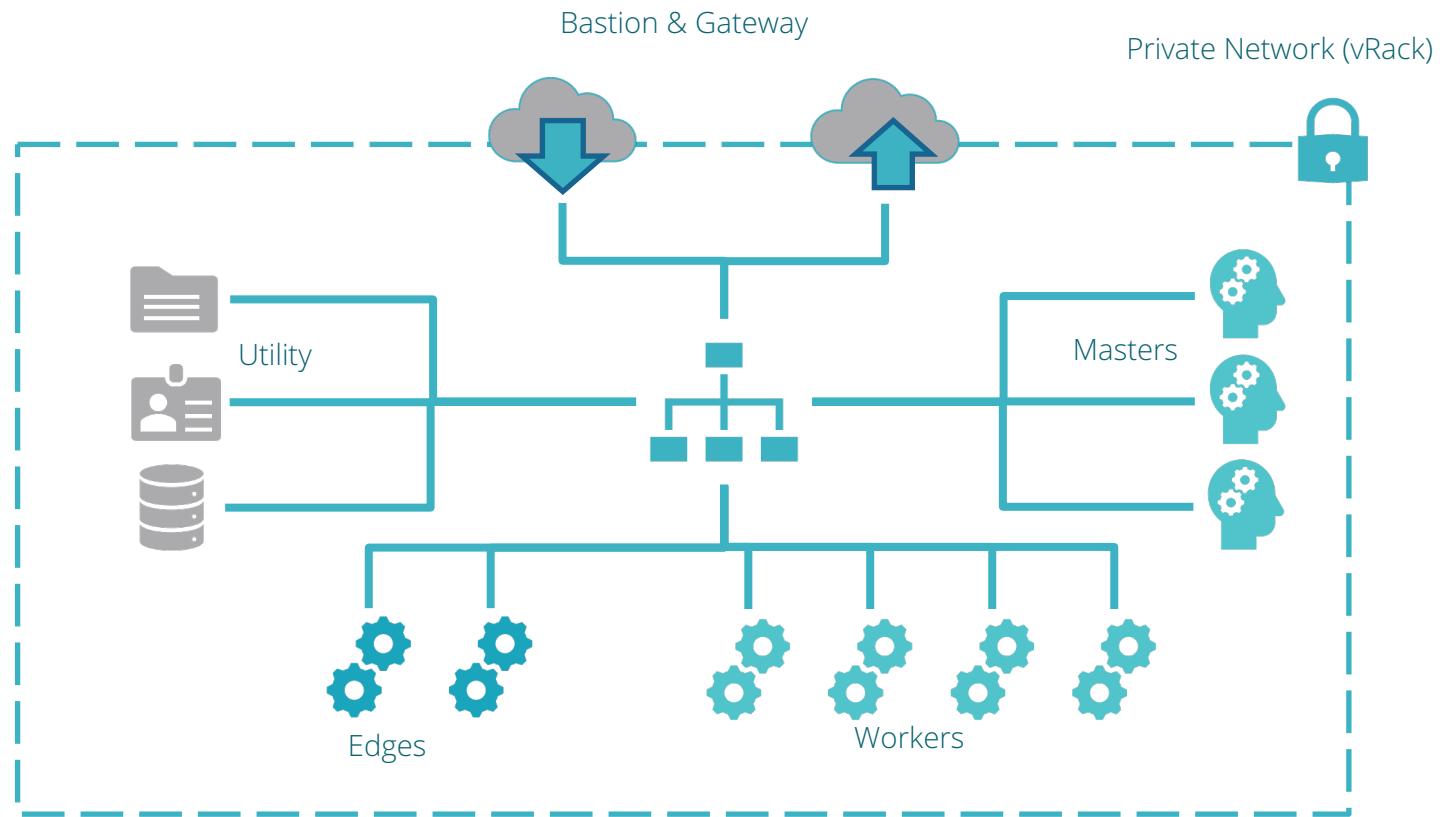


1. Infrastructure
2. Launch a cluster
3. Discover the Ambari dashboard
4. First steps with user management
5. First steps with data security

Step 1: Estimate your needs, size you cluster

We will explain it live :

- ✓ Size the Workers
- ✓ Size the Edges
- ✓ Site the Storage



Step 2 : launch your OVH cluster (live demo)

A cluster deployment takes approximately 40 minutes.

<https://www.ovh.com/fr/public-cloud/big-data-hadoop/>

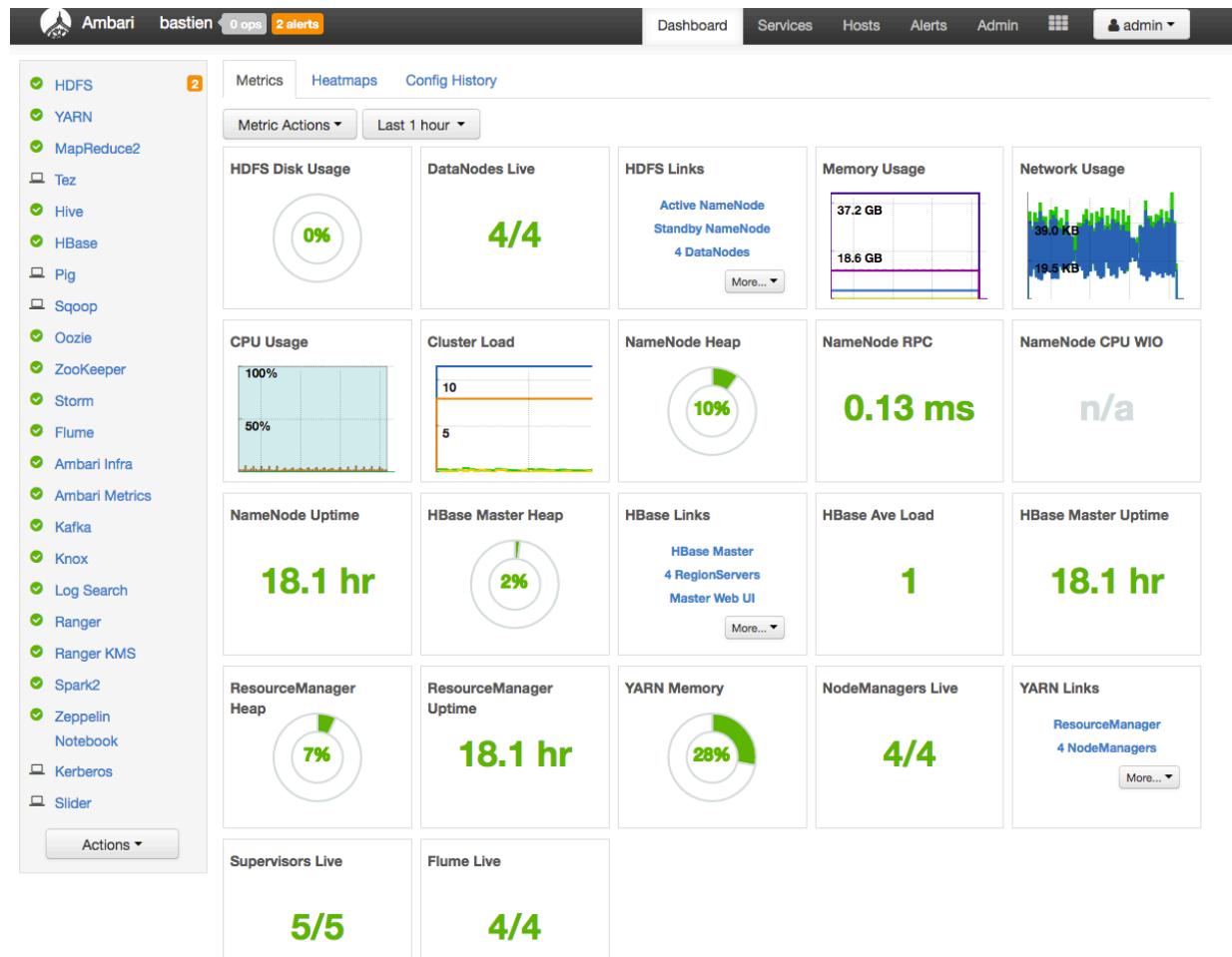
OVH will deploy ADP cluster for you ☺

You will find the credentials on the OVH control panel

The screenshot shows the OVH Public Cloud interface with the 'Public Cloud' tab selected. On the left, a sidebar lists various services: Compute (Instances), Storage (Block Storage, Object Storage, Cloud Archive, Volume Snapshot, Instance Backup), Network (Private Network, Failover IP), Orchestration / Industrialization (Managed Kubernetes Service), Data & Analytics (Analytics Data Platform, currently selected), and Management Interfaces (Horizon). The main content area is titled 'Welcome to the new Public Cloud interface' and shows the path 'pci.projects / [Internal] bigdata india / Analytics Data Platform / Déployer'. It displays the first step of a wizard: 'Deploy an Analytics Data Platform'. The step is described as containing tools for ingest, clean, process, and storage resources. A sub-step '1 General configuration' is shown, with 'Cluster name' set to 'academy'. Other fields include 'Analytic Data Platform software version' (set to 'HDP'), 'Public cloud project' (set to '[Internal] bigdata india'), and a 'Next' button. Steps 2 and 3 are also visible: '2 Security' and '3 Select region and datacenter'.

Step 3 : Connect to you cluster, discover Ambari

1. Browse the Control Panel and find the Ambari cluster URL
2. Connect to Ambari dashboard with your given credentials
3. Discover Ambari dashboard :
 1. Left menu, top menu
 2. HDFS infos and config
 3. Alerts



Step 4 : Discover FreeIPA, the users manager

Ambari is a dashboard to manage your cluster

- To manage the users, please connect to FreeIPA (find the URL in control panel)
- Login/password : same as Ambari

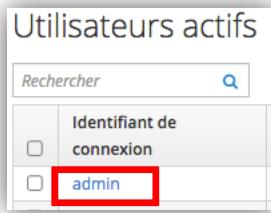
The screenshot shows the FreeIPA web interface. The top navigation bar includes tabs for Identité, Politique, Authentification, Services réseau, and Serveur IPA. Below this, a secondary navigation bar has tabs for Utilisateurs, Hôtes, Services, Groupes, Vues d'identifiants, and Auto-adhésion. The 'Utilisateurs' tab is selected. On the left, a sidebar lists categories: Catégories d'utilisateurs, Utilisateurs actifs (selected), Utilisateurs en attente, and Utilisateurs préservés. The main content area is titled 'Utilisateurs actifs' and contains a search bar labeled 'Rechercher'. A table displays nine active user entries:

	Identifiant de connexion	Prénom	Nom	État	UID	Adresse courrie
<input type="checkbox"/>	admin		Administrator	✓ Activé(e)	241400000	
<input type="checkbox"/>	ambari-qa-bastien	ambari-qa-bastien	ambari-qa-bastien	✓ Activé(e)	241400009	ambari-qa-basti fbc57e5de8ae.d
<input type="checkbox"/>	ambari-server-bastien	ambari-server- bastien	ambari-server- bastien	✓ Activé(e)	241400015	ambari-server-b fbc57e5de8ae.d
<input type="checkbox"/>	hbase-bastien	hbase-bastien	hbase-bastien	✓ Activé(e)	241400011	hbase-bastien@
<input type="checkbox"/>	hdfs-bastien	hdfs-bastien	hdfs-bastien	✓ Activé(e)	241400010	hdfs-bastien@6a
<input type="checkbox"/>	ovhsupport	OVH	Support	— Désactivé	241400003	ovhsupport@6a
<input type="checkbox"/>	spark-bastien	spark-bastien	spark-bastien	✓ Activé(e)	241400014	spark-bastien@6
<input type="checkbox"/>	storm-bastien	storm-bastien	storm-bastien	✓ Activé(e)	241400012	storm-bastien@6
<input type="checkbox"/>	zeppelin-bastien	zeppelin-bastien	zeppelin-bastien	✓ Activé(e)	241400013	zeppelin-bastier

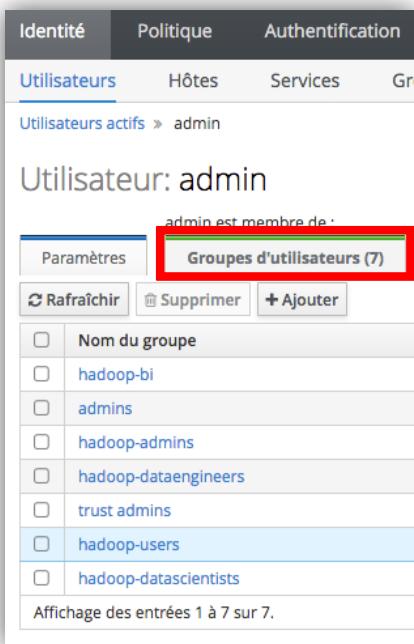
A footer note at the bottom of the table reads 'Affichage des entrées 1 à 9 sur 9.'

Modify your account

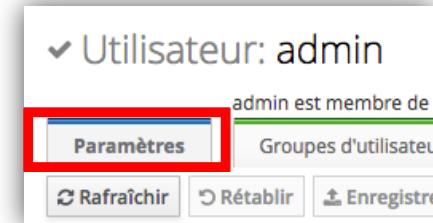
1. Click on your account



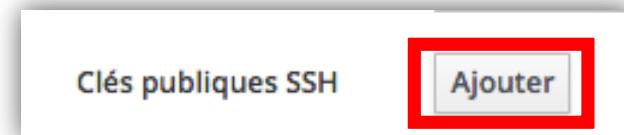
2. Check on which groups you are



3. in the “paramètres” tab



4. Add your SSH public Key if you have one



Step 5 : discover data management

FreeIPA is for managing users, Ranger is for data security

1. To manage the data, please connect to Apache Ranger (find the URL in control panel)
 - Login/password : same as Ambari
2. Then, discover for example HDFS rights → live demo

The screenshot shows the Apache Ranger Service Manager interface. At the top, there is a green navigation bar with the following items: 'Ranger' (selected), 'Access Manager', 'Audit', and 'Settings'. Below the navigation bar, the title 'Service Manager' is displayed. Underneath the title, there is a section for 'HDFS' which includes a '+' button and three small icons. A red box highlights the text 'bastien.hadoop' in a list below this section. At the bottom of the interface, there are two more small icons.

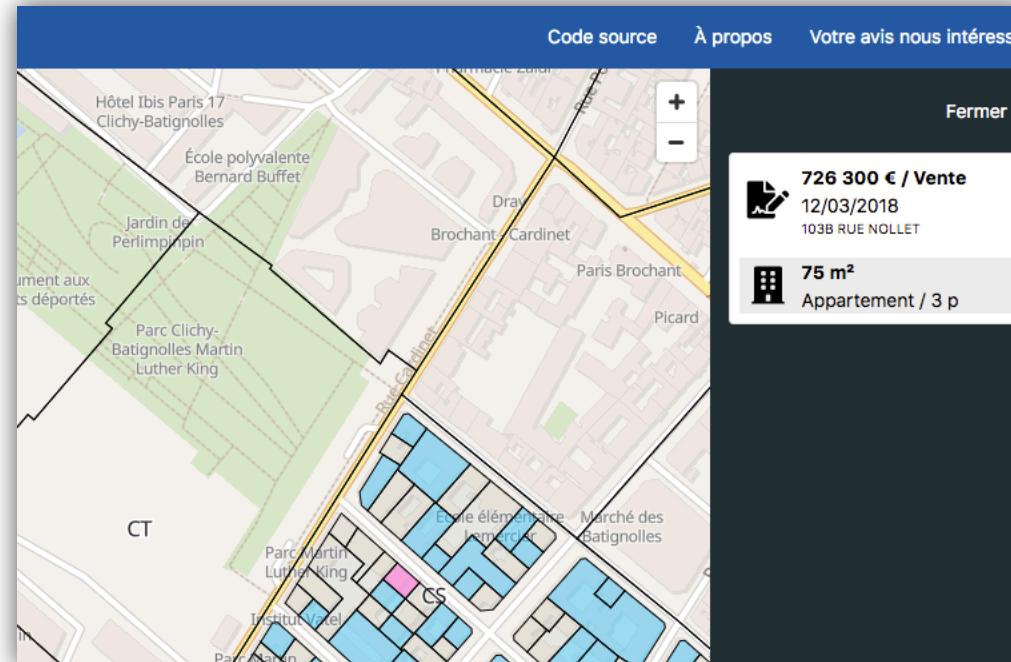
Use case : let's play with real estate market in France !

France government releases a lot of data ("open data")

In 2019, they proposed the DVF datas, the "Demandes de Valeurs Foncières".
= Amount of money for each real estate transaction in France since 5 years.

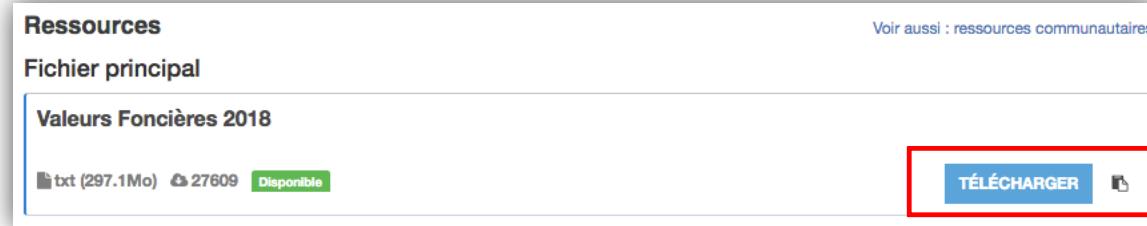
Example : in blue the sales in 2018 in Paris 75017 , with the prices

<https://app.dvf.etalab.gouv.fr/>

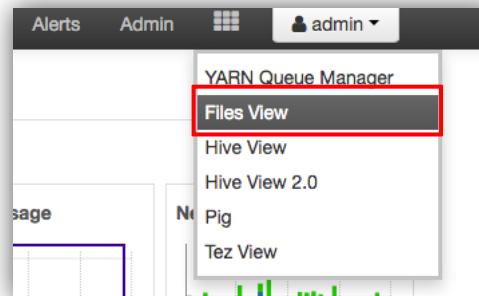


Step 1 (1/2) : ingest static data

1. Go in github link, find the DVF website, download DVF data 2018

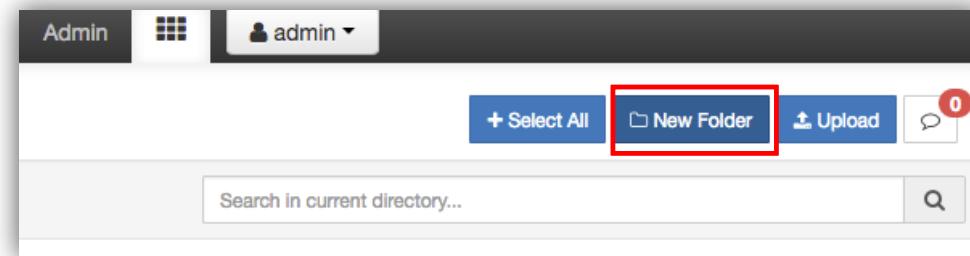


2. Go back in Ambari dashboard
3. Click on Files View :



4. go in /user/<your_login>
5. Create a new directory called "ovh"

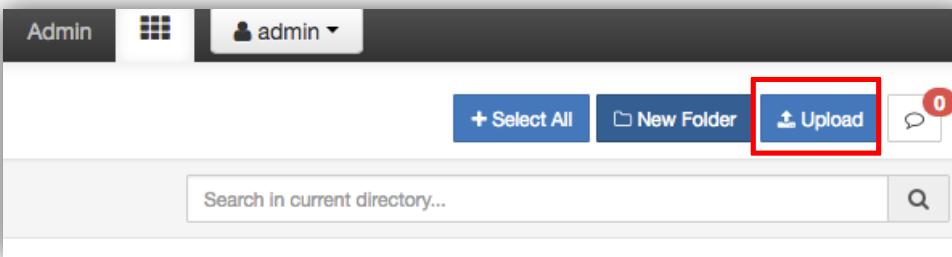
You should have : /user/<your_login>/ovh



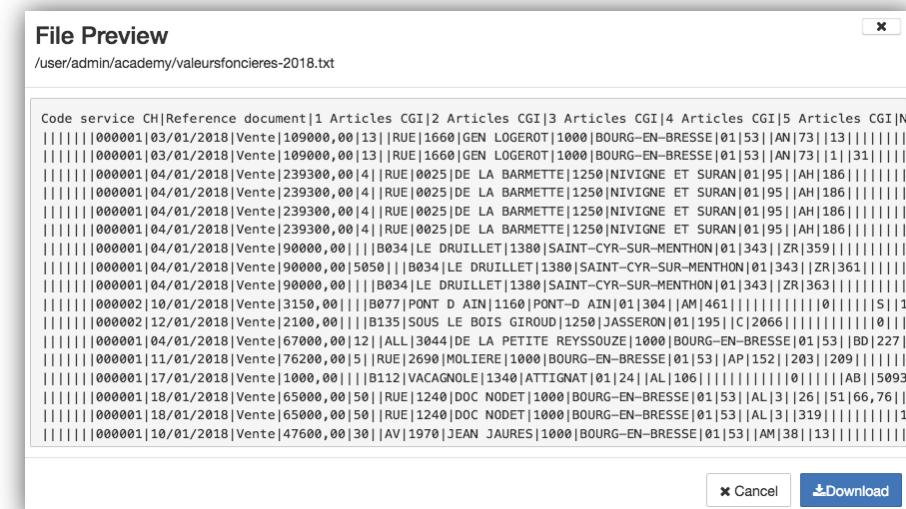
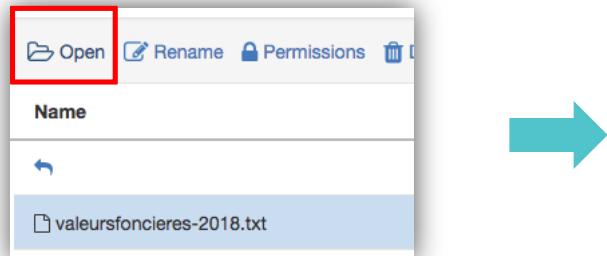
Step 1 (2/2) : ingest static data

Now we will ingest static data, the TXT file previously downloaded.

1. Click on upload and add the TXT file « valeursfoncieres-2018.txt »



2. Once uploaded, you can select your file then click on « Open » to have a file preview

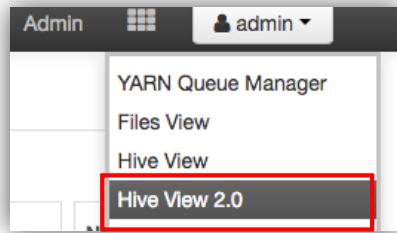


Step 2 : Prepare the data with Hive

Hive allow you to query the data with a language very similar to SQL

We will create a new table in Hive to “load” the TXT file data.

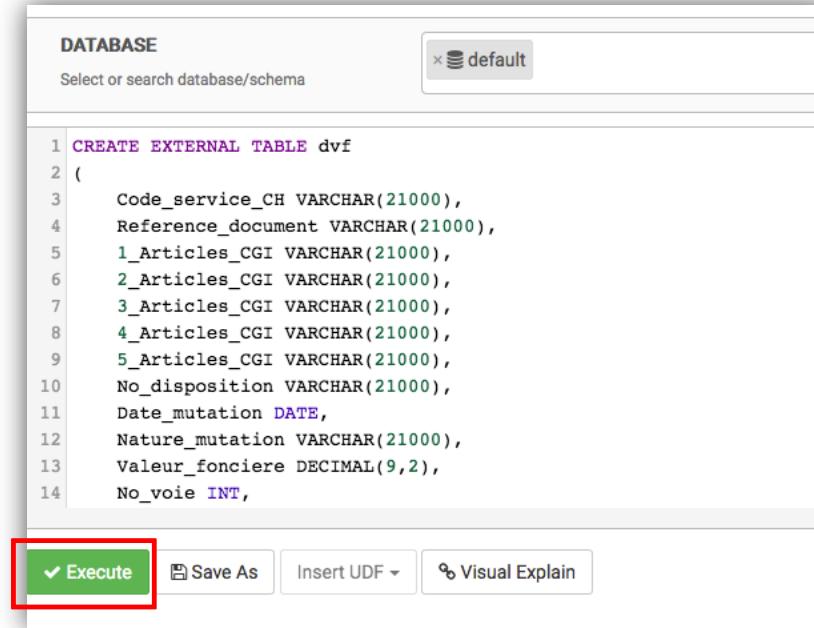
1. Go in Hive view 2.0



2. Copy paste the SQL code to create a table

1. WARNING : change the path with your login

3. Click on “Execute”



```
1 CREATE EXTERNAL TABLE dvt
2 (
3     Code_service_CH VARCHAR(21000),
4     Reference_document VARCHAR(21000),
5     1_Articles_CGI VARCHAR(21000),
6     2_Articles_CGI VARCHAR(21000),
7     3_Articles_CGI VARCHAR(21000),
8     4_Articles_CGI VARCHAR(21000),
9     5_Articles_CGI VARCHAR(21000),
10    No_disposition VARCHAR(21000),
11    Date_mutation DATE,
12    Nature_mutation VARCHAR(21000),
13    Valeur_fonciere DECIMAL(9,2),
14    No_voie INT,
```

Step 3 : simple queries to verify if everything is ok

For each step, enter the query in HIVE window then execute it (see previous step)

1. Let's check the amount of lines in the file

```
SELECT count(*) FROM dvf_<your_login>
```



RESULTS		LOG	VISU
Filter columns		x	
_c0			
2339002			

2. Let's verify the data inside hive

```
SELECT * FROM dvf_<your_login> LIMIT 20
```



e	data.code_voie	data.voie	data.code_postal	data.commune	data.co
1660	GEN	LOGEROT	1000	BOURG-EN-BRESSE	01

3. And more !

```
SELECT * from dvf_<your_login> WHERE code_postal = "75017" AND type_local = "Appartement" LIMIT 100
```

?! Where are the prices for transactions ?

- In the files view, we had the amount of money for each transaction
- In Hive, we don't have it, NULL everywhere

File Preview							
/user/admin/academy/valeursfoncieres-2018.txt							
Code service CH Reference document 1 Articles							
000001 03/01/2018	Vente 109000,00 13						
000001 03/01/2018	Vente 109000,00 13						
000001 04/01/2018	Vente 239300,00 4 F						
000001 04/01/2018	Vente 239300,00 4 F						
000001 04/01/2018	Vente 239300,00 4 F						
000001 04/01/2018	Vente 239300,00 4 F						
000001 04/01/2018	Vente 90000,00 1 B6						

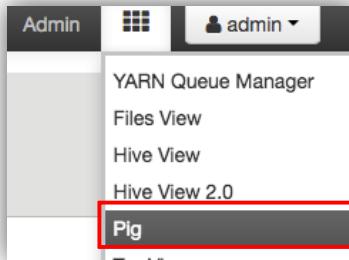
RESULTS	LOG	VISUAL EXPLAIN	TEZ UI
Filter columns <input type="button" value="x"/>			
<input type="button" value="Mutation"/> <input type="button" value="data.valeur_fonciere"/> <input type="button" value="data.no_voie"/> <input type="button" value="data.b_t_q"/> <input type="button" value="data.type_de_voie"/> <input type="button" value="data.code_voie"/> <input type="button" value="data.voie"/> <input type="button" value="data.code_postal"/> <input type="button" value="data.commune"/>			
.mutation		data.valeur_fonciere	
null		13	
null		13	

Issue : in the file; prices are with **commas** (virgules), instead of **periods** (points)

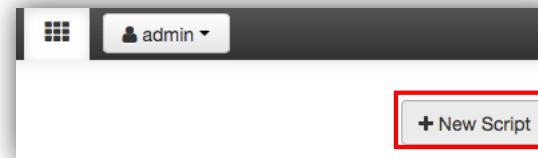
→ let's fix it with Apache Pig !

Step 4 : create script in Apache Pig

1. First, create a new folder in "Files view" in /user/<your_login>/cleaned/
2. Then go in Pig :



3. Create a new script called “commas” :



4. Add 3 lines of code (see github) :

The screenshot shows the 'Script' tab of the Pig interface. The script is named 'commas'. It contains the following code:

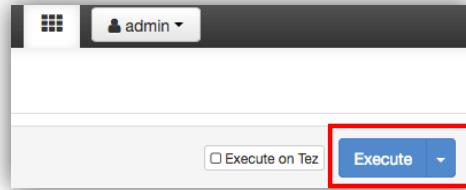
```
1 A = LOAD '/user/<your_login>/ovh/valeursfoncieres-2018.txt' as line;
2 B = FOREACH A GENERATE REPLACE(line,'([,]+)','.');
3 STORE B INTO '/user/<your_login>/cleaned/cleaned_valeursfoncieres-2018.txt';
```

5. Save

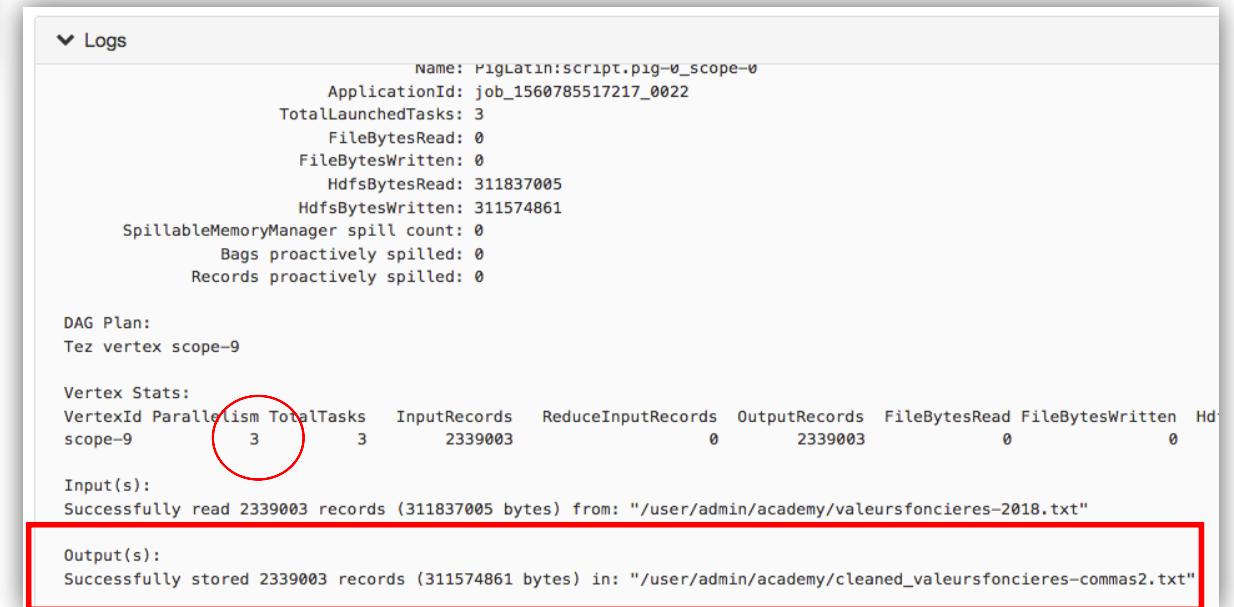
Step 5 : launch your script

The script will load your data, replace "," by "." then store it in another folder

1. Execute the script :



2. Watch the logs, wait for completion



Logs

Name: PIGLATIN:script.pig-a_scope-a
ApplicationId: job_1560785517217_0022
TotalLaunchedTasks: 3
FileBytesRead: 0
FileBytesWritten: 0
HdfsBytesRead: 311837005
HdfsBytesWritten: 311574861
SpillableMemoryManager spill count: 0
Bags proactively spilled: 0
Records proactively spilled: 0

DAG Plan:
Tez vertex scope-9

Vertex Stats:
VertexId Parallelism TotalTasks InputRecords ReduceInputRecords OutputRecords FileBytesRead FileBytesWritten HDFSRead HDFSWritten
scope-9 3 3 2339003 0 2339003 0 0

Input(s):
Successfully read 2339003 records (311837005 bytes) from: "/user/admin/academy/valeursfoncieres-2018.txt"

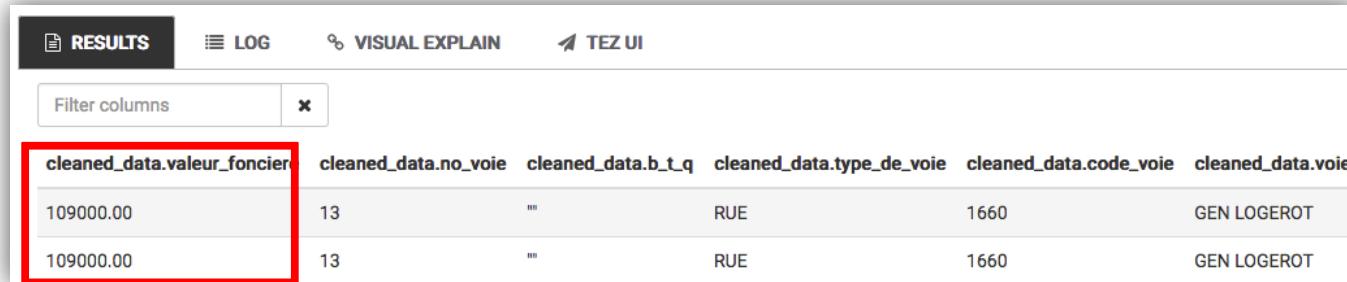
Output(s):
Successfully stored 2339003 records (311574861 bytes) in: "/user/admin/academy/cleaned_valeursfoncieres-commas2.txt"

3. Verify your files in "Files view"

4. Your TXT file is now a folder with 3 sub files containing cleaned data

STEP 6 : Calculate average apartment price

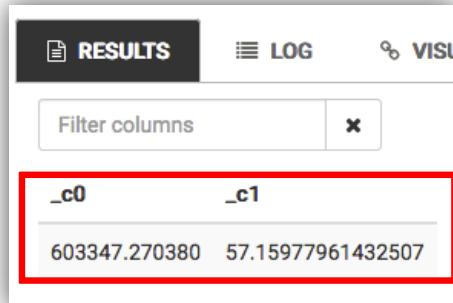
1. Go back in "Hive View 2.0"
2. Create a new table called "cleaned_data" by copy/paste code (in Github)
WARNING : change the path with your login
3. Verify that "Valeur_Fonciere" is now here, with periods (points)



cleaned_data.valeur_fonciere	cleaned_data.no_voie	cleaned_data.b_t_q	cleaned_data.type_de_voie	cleaned_data.code_voie	cleaned_data.voie
109000.00	13	""	RUE	1660	GEN LOGEROT
109000.00	13	""	RUE	1660	GEN LOGEROT

4. Query the data to find the average price of appartement sold in 2018 in Paris 17

```
SELECT AVG(valeur_fonciere), AVG(Surface_reelle_bati) FROM cleaned_data_<your_login> WHERE code_postal = "75017" AND type_local = "Appartement"
```

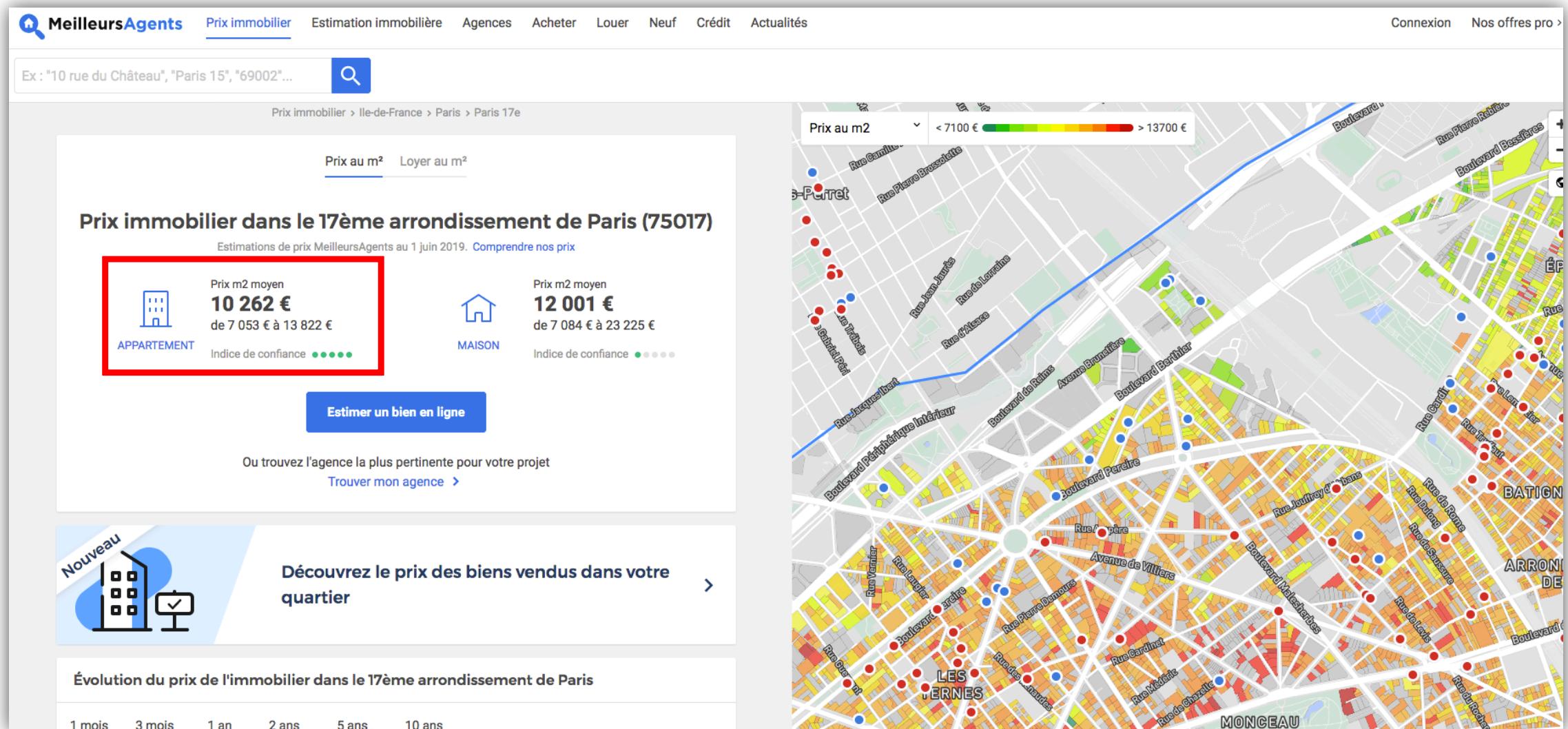


_c0	_c1
603347.270380	57.15977961432507



Average price is 603'347€
Average square meter is 57 m²
= 10'526€ per square meter in 2018

This use case is real for businesses !



(bonus) STEP 7 : connection in SSH to your Edge node

Warning : make sure you added your SSH Public key in your profile in FreeIPA.

Propagation time +- 20mins

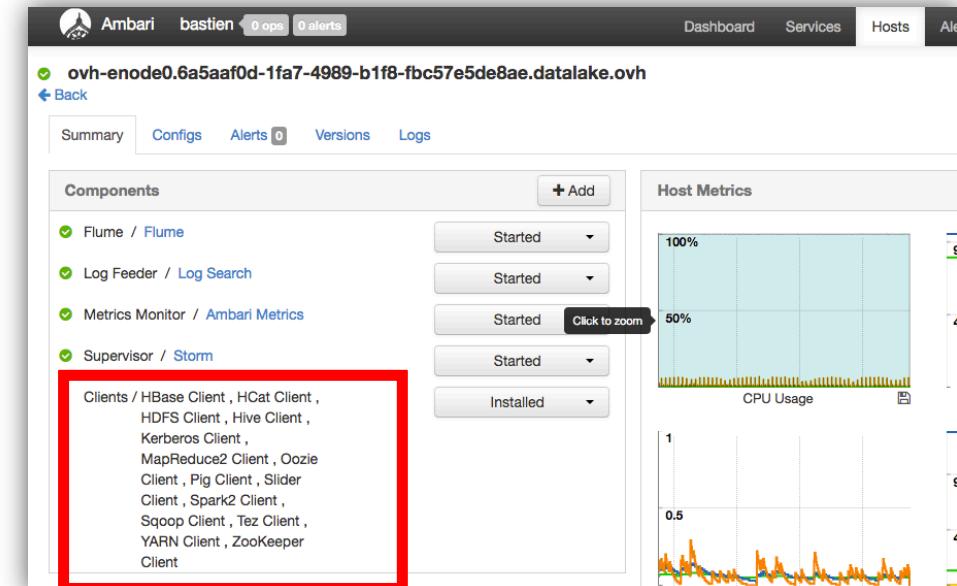
1. Connect to the cluster Bastion

```
ssh -A <user_login>@{bastion_IP}
```

2. Then connect to your Edge node, from your bastion

```
ssh <user_login>@ovh-enode0
```

3. You can now use all the clients available on your Edge node



You can find the list in Ambari / Hosts

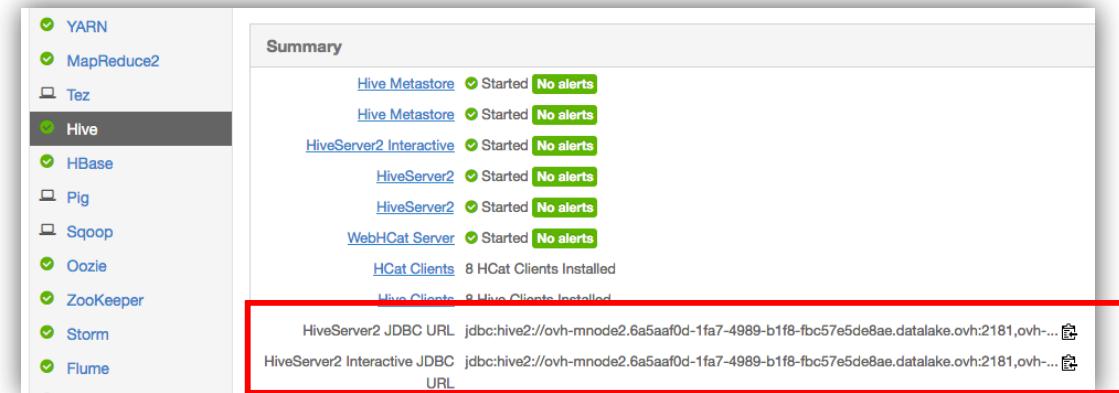
(bonus) Step 8 : submit a Hive job via CLI

- Once you are logged in SSH on your Edge node, launch Apache Beeline

- Then connect to Hive

```
!connect <Hive_server_URL>
```

(You will find your Hive server URL in Ambari)



- Now you can submit queries, exactly like inside the Hive UI

```
0: jdbc:hive2://ovh-mnode2.6a5aaaf0d-1fa7-4989> SELECT * from cleaned_data LIMIT 10;
+-----+-----+-----+-----+
| cleaned_data.code_service_ch | cleaned_data.reference_document | cleaned_data.1_articles_cgi | cleaned_data._mutation |
| cleaned_data.nature_mutation | cleaned_data.valeur_fonciere | cleaned_data.no_voie | cleaned_data.partement |
| cleaned_data.code_communne | cleaned_data.prefixe_de_section | cleaned_data.section | cleaned_data._2eme_lot |
| cleaned_data.3eme_lot | cleaned_data.surface_carrez_du_3eme_lot | cleaned_data.4eme_lot | cleaned_data.local |
| cleaned_data.type_local | cleaned_data.identifiant_local | cleaned_data.surface_reelle_bati | cleaned_data._dependance |
+-----+-----+-----+-----+
| Vente | 109000.00 | 13 | 73 |
| 53 | NULL | AN | NULL |
| Dépendance | 0 | 0 | 0 |
```

Now Imagine the same queries from various softwares...

Big data queries are usually made from third-party softwares

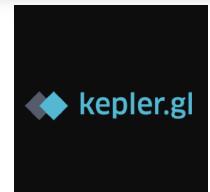
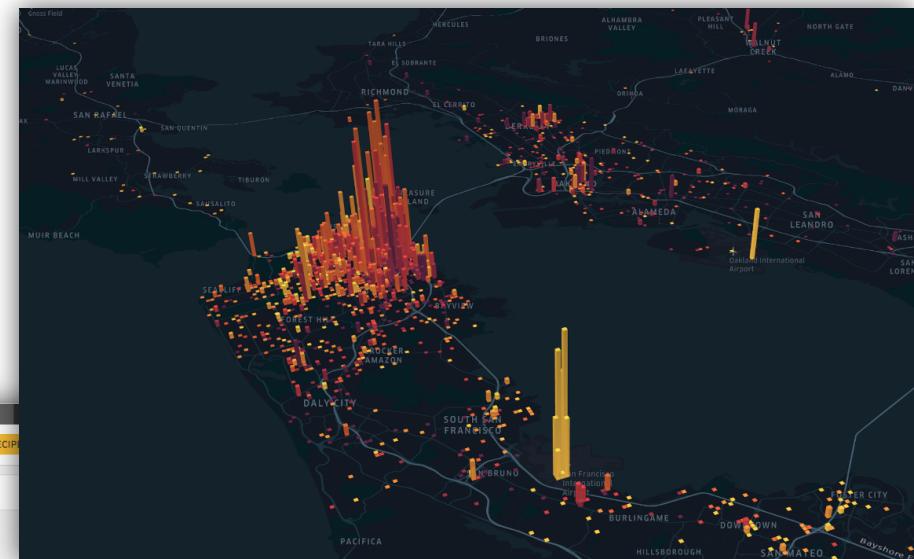
You “connect” them to a big data cluster

The screenshot shows the Tableau interface with the following details:

- Header:** +ab|eau, Production, Content, Users, Groups, Schedules, Tasks, Status, Settings, Martin Rodriguez.
- Breadcrumbs:** Home > Manufacturing Indicators
- Page Title:** Manufacturing Indicators
- Section Headers:** Workbooks (14), Views (18), Data Sources (0), Permissions, Details.
- Search Bar:** + 0 selected
- General Filters:** Owner, Tag, Modified on or after, Modified on or before.
- Sort By:** Views: All (Most-Least)
- Visualizations:**
 - Execution Overview: Scatter plot showing various metrics.
 - Product Defects: Line chart showing defect counts over time.
 - Key Metrics: A dashboard containing multiple small charts including a pie chart and line graphs.
 - Systems Breakdown: Treemap visualization showing system components and their status.
 - Defect Age Tracking: Line chart showing the age distribution of defects.
 - Shipping Rates: Line chart showing shipping rates over time.
 - Cluster Analysis: Scatter plot showing data points grouped into clusters.
 - Base Sales Opportunities: Stacked bar chart showing sales opportunities across different categories.
 - Active Trends: Line chart showing active trends over time.
 - Backend Site Performance: Line chart showing performance metrics for backend sites.
- Bottom Left:** +ab|eau logo with a stylized cross icon.
- Bottom Right:** International_passenger data source preview.

The screenshot shows the Dataiku Learn interface with the following details:

- Top Bar:** DATASETS, Summary, Explore, Charts, Status, History, Settings, PARENT RECORD
- Left Panel:** A sidebar with a tooltip for "International_passenger..." and a "Run In DS" button.
- Chart Area:** A line chart titled "International_passenger... by Month_parsed". The Y-axis ranges from 0 to 600. The X-axis shows dates from 1949-05-27 to 1960-06-28. The chart displays a fluctuating blue line representing the average number of international passengers per month.
- Filter/Grouping:** A modal dialog box is open with the following settings:
 - Y:** International_passenger...
 - X:** Month_parsed (Month)
 - Date ranges:** Month
 - Sorting:** Natural ordering
 - Ticks:** Generate one tick per bin (unchecked)
- Bottom Navigation:** MORE OPTIONS and CHART buttons.



THANK YOU ! LAB DONE ☺



Espace Client (vb44094-ovh) | Webmail | Centre d'aide | OVH Community | OVH Blog |

Serveur

Public Cloud

Web Hosting

Télécom

Programs

À propos

Analytics Data Platform

Présentation Usage Tarifs Espace client

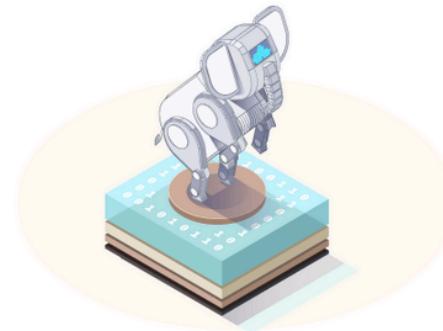
/ Public Cloud / Data and Analytics / Analytics Data Platform

Votre cluster big data Hadoop déployé en quelques clics

Déployer un cluster big data est un processus long et contraignant. OVH Analytics Data Platform vous simplifie votre métier. En moins d'une heure, nous vous livrons une stack Apache Hadoop, préconfigurée et prête à l'usage.

En nous basant sur un standard de distribution Hadoop open source, nous préconfigurons tous les services nécessaires à vos traitements de données et sécurisons vos flux avec le monde externe ainsi que vos utilisateurs.

Déployez OVH Analytics Data Platform pour de nombreux usages : l'analyse des marchés, l'informatique décisionnelle, l'IoT ou encore la maintenance prévisionnelle. À vous de jouer !



Offer page : <https://www.ovh.com/fr/public-cloud/big-data-hadoop/>



Credits : Bastien Verdebout / <https://twitter.com/BastienOvh>