

Chapitre 1 :

Introduction a Spark

A. Notes de cours

- 1) Le projet Spark a été lancé pour résoudre les problèmes potentiels dans le cadre Hadoop MapReduce.
- 2) Hadoop MapReduce : framework révolutionnaire pour gérer le traitement de données volumineuses mais une vitesse limitée.
- 3) Spark est donc capable d'effectuer des calculs en mémoire plus rapide que Hadoop MapReduce
- 4) Les trois angles sous lequel les données peuvent être visualisées sont : la manière dont elles sont collectées, stockées et traitées.



Les principaux composants de base de spark sont :

- Stockage : Spark vous permet d'utiliser des bases de données relationnelles traditionnelles ainsi que NoSQL, telles que Cassandra et MongoDB.
- La gestion des ressources : Les deux gestionnaires de ressources les plus utilisés sont YARN et Mesos. Le gestionnaire de ressources comporte deux composants principaux en interne :

- Cluster manager :de gérer les nœuds de travail et de leur attribuer des tâches, en fonction de la disponibilité et de la capacité du nœud de travail
 - Worker
- Moteur et Écosystème : La base de l'architecture Spark est son noyau, qui est construit au dessus des RDD (Resilient Distributed Datasets) et offre plusieurs API pour la construction d'autres bibliothèques et écosystèmes par les contributeurs Spark. Les bibliothèques default de spark sont : Spark SQL, Mllib, Structured Streaming, Graph X.
- API : Spark est disponible en quatre langues tels que scala,python,java et R



Il existe plusieurs façons d'utiliser Spark :

- Configuration locale
- Dockers
- Environnement en nuage (GCP, AWS, Azure)
- Databriks

B. CAS Pratique : Installation de Spark

Pour installer Spark, nous avons premierement installer **Docker Desktop**, puis nous avons ajouter l'image de **jupyter/all-spark-notebook**.

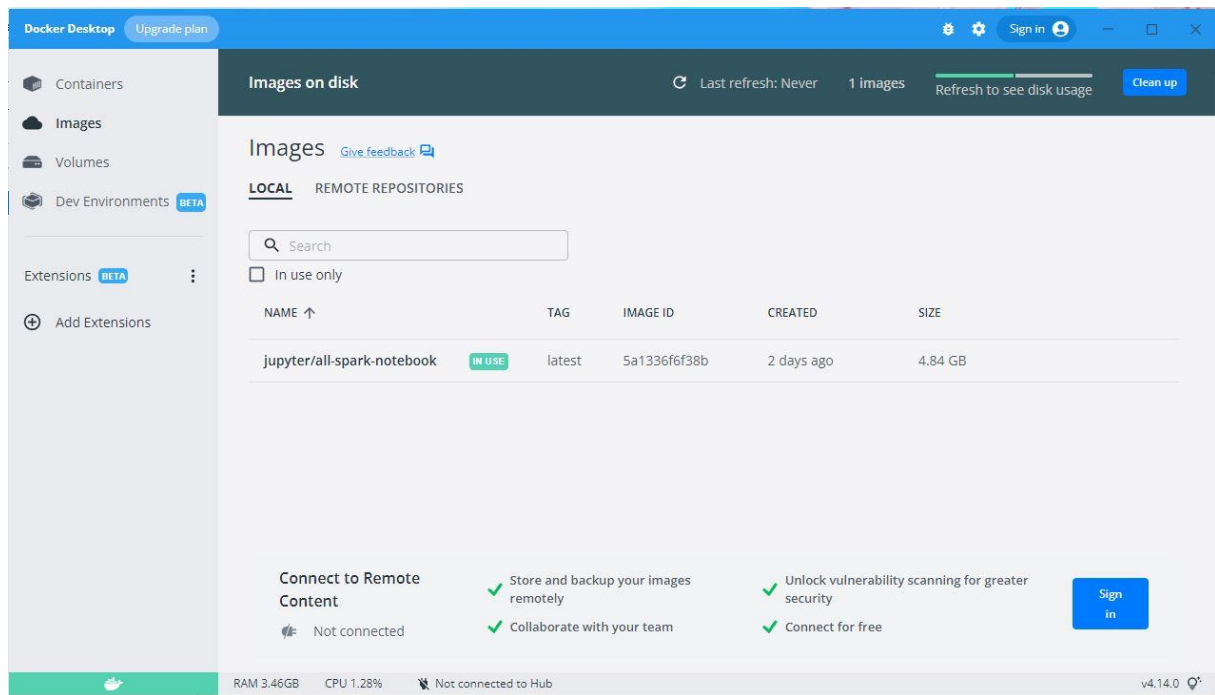


Figure 1: Interface de docker apres avoir ajouter jupyter/all-spark-notebook

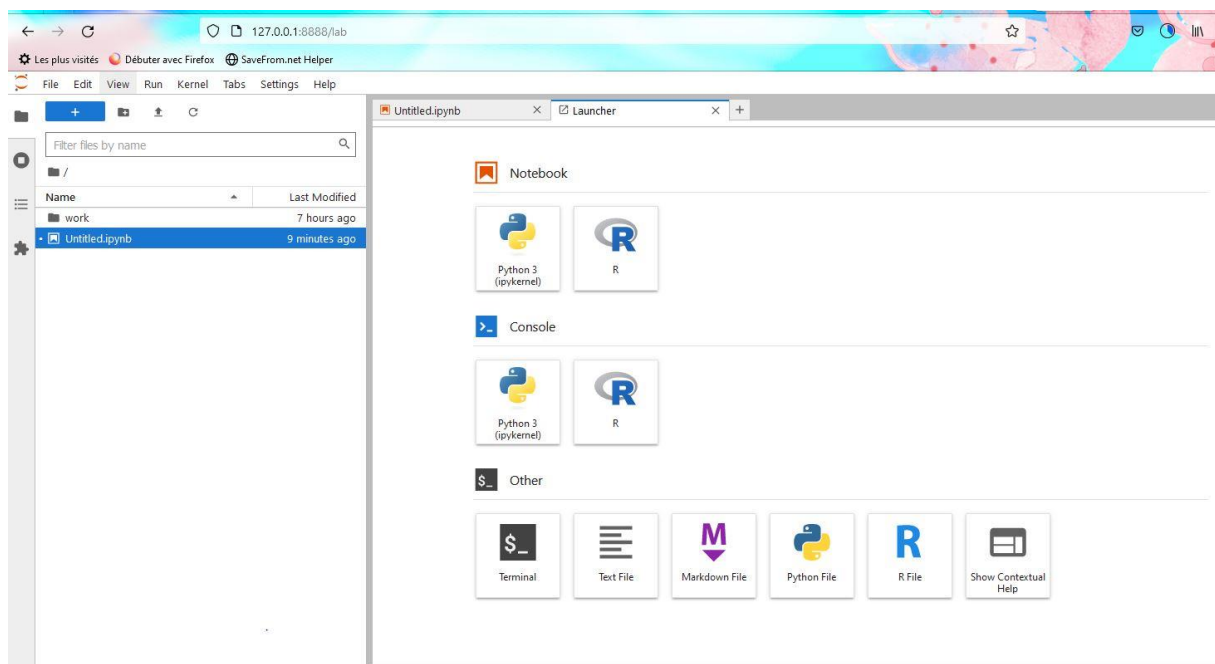


Figure 2: Interface de jupyter