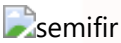
Spark : Fonctionnement



Spark : Fonctionnement

Le principe du travail de spark est de faire du traitement de données massivement et rapidement.

Pour realiser cet objectif, spark utilise la **parralelisation**.

Spark : Parralelisation

La parralelisation permet de **diviser** un problème en plusieurs sous-problèmes.

Puis de les résoudre en meme temps (parallèle) sur différents processeurs.

Le plus souvent repartie sur plusieurs machines : le cluster.

Spark : Parralelisation

Cette **parralelisation** est réalisée par :

- La division des données en Resilient Distributed Dataset (RDD).
- La division du programme par le Direct Acyclic Graph.

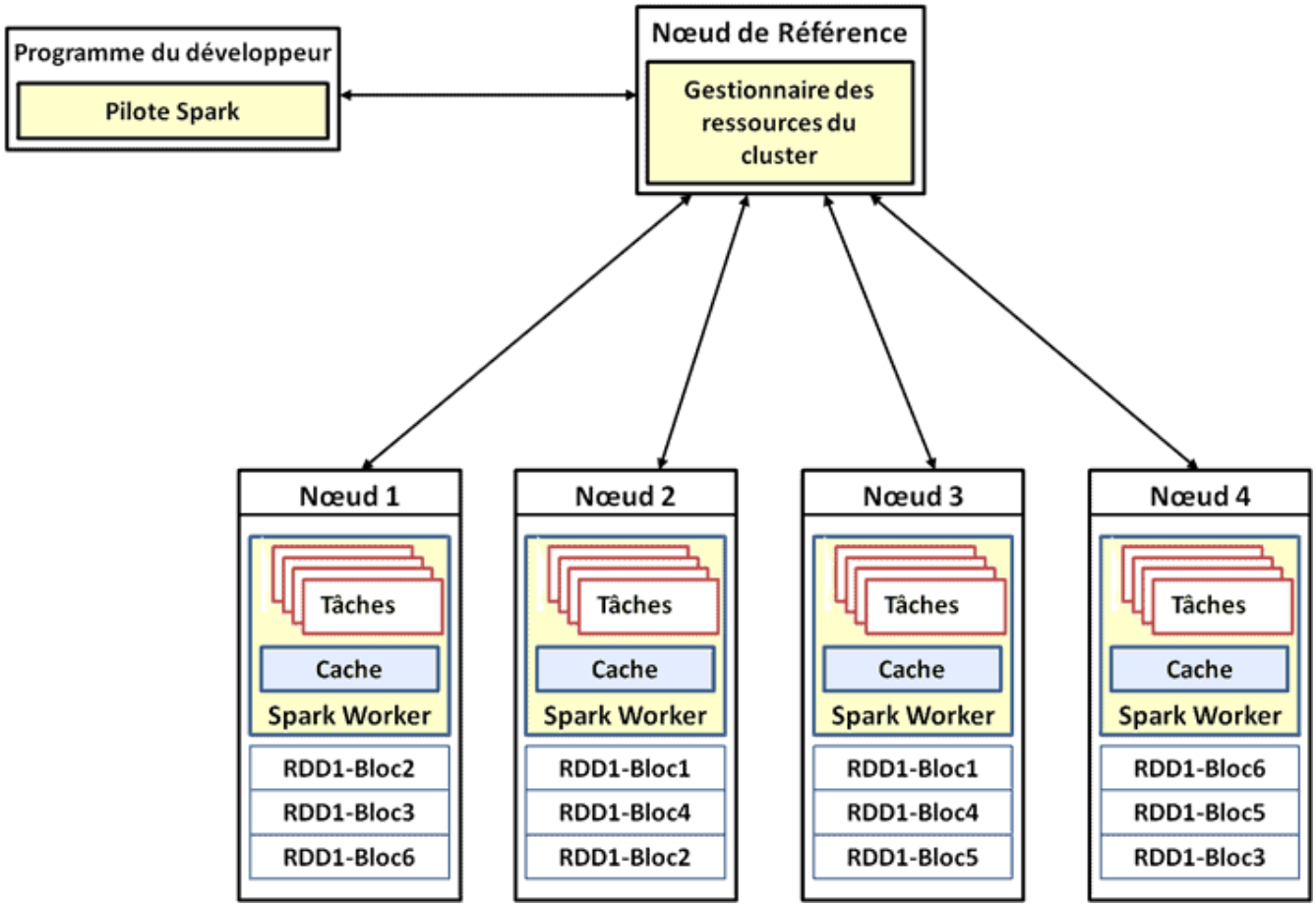
Resilient Distributed Dataset (RDD)

Pour améliorer le traitement des données, spark manipule les informations sous un format divisé en blocs d'informations réparties sur les differents noeuds.

> C'est le Resilient Distributed Dataset (RDD).

Resilient Distributed Dataset (RDD)

Les blocs de données sont appelés **partitions** et sont stockés en plusieurs exemplaire sur les differents noeuds du cluster.



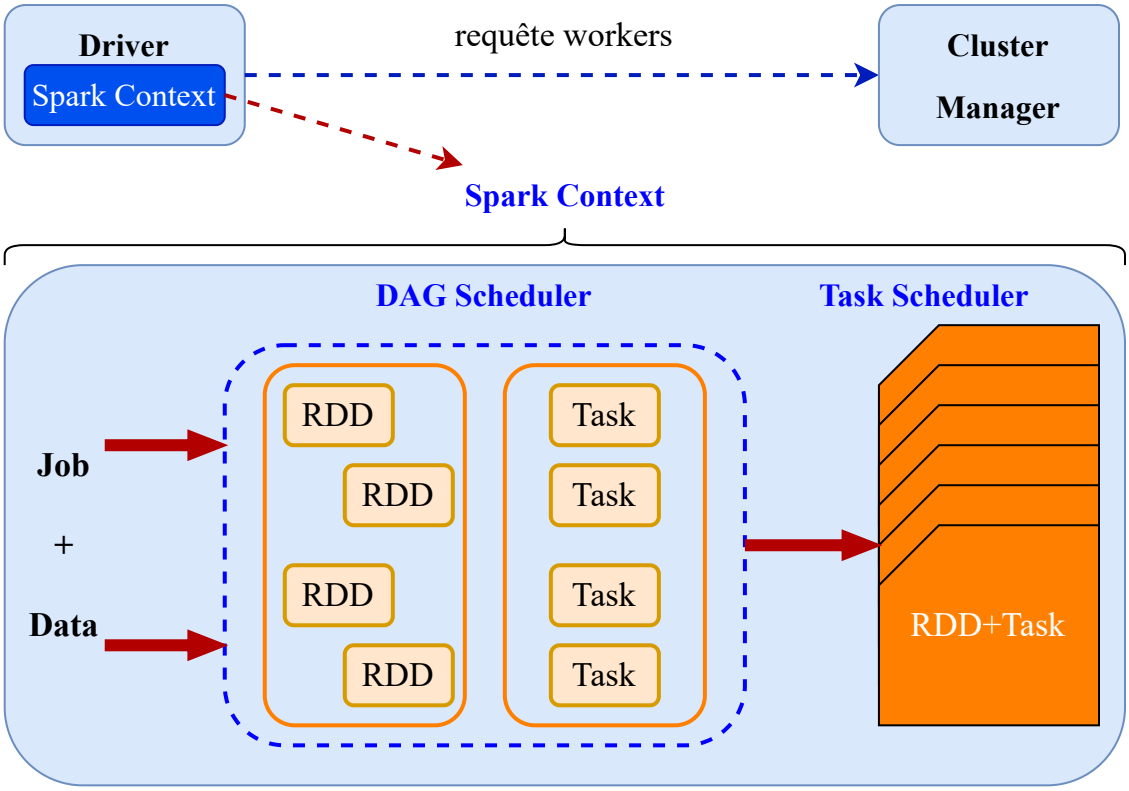
Direct Acyclic Graph (DAG)

Lorsque spark reçoit un programme, celui va etre également transformer.

Le DAG va permettre de **diviser le travail** à accomplir en action à effectuer les unes à la suites des autres.

Une tache est ensuite planifier pour une action sur un bloc de données.

Direct Acyclic Graph (DAG)



Resumé

Spark va diviser les données en partitions et les répartir sur les differents noeuds du cluster.

Puis va diviser le programme en taches à effectuer sur les partitions.

Enfin , il va repartir les taches sur les differents noeuds du cluster.

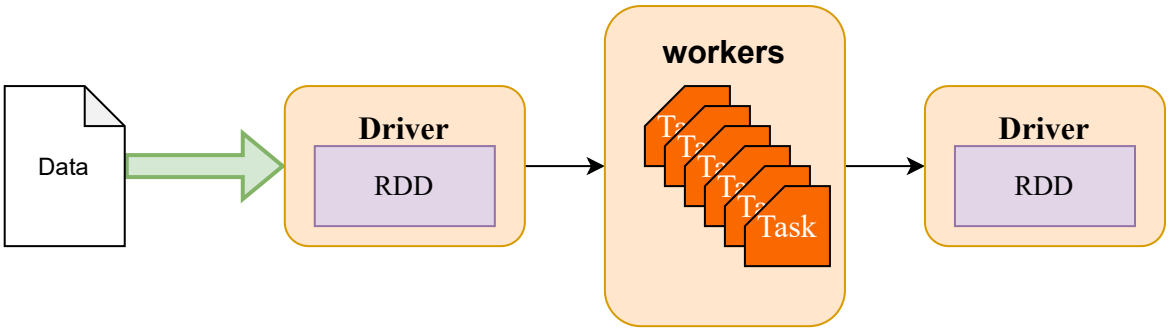
> Le travail peut alors etre effectué en meme temps.

Spark : Flux de données

Le travail de traitement des données par Spark peut se faire selon differentes configurations:

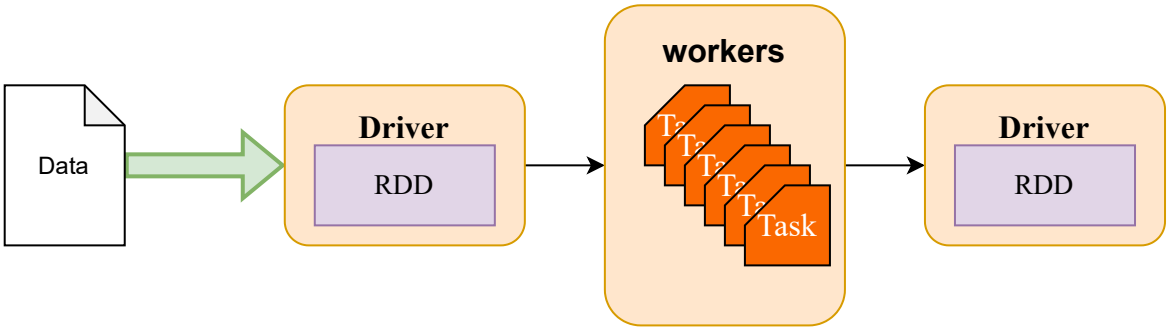
- Traitement par lots (batch)
- Traitement en micro-batch
- Traitement en temps réel (streaming)

Traitement par lots (batch)



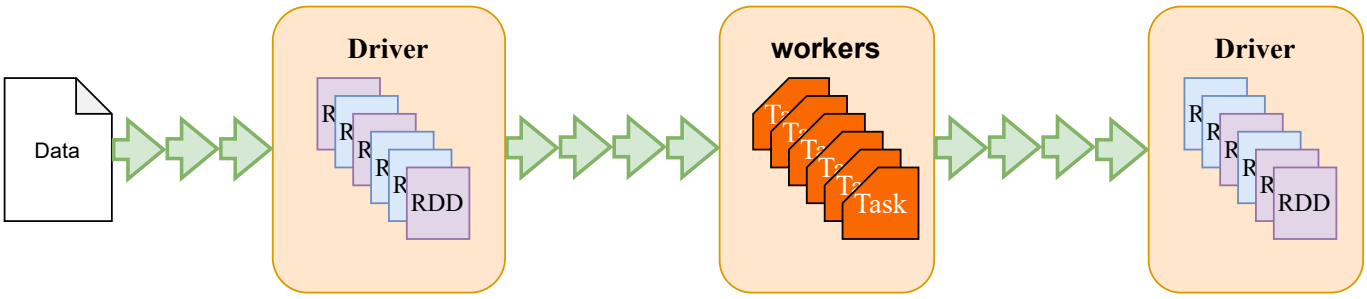
- Consiste à charger en mémoire l'ensemble des données à traiter.
- Un seule RDD sur les differents noeuds du cluster (stocké et traité de maniere distribuée).

Traitement par lots (batch)



- Le traitement s'effectue une fois le chargement complet des données effectuées.
- L'ensemble est ensuite regrouper en un seul RDD.

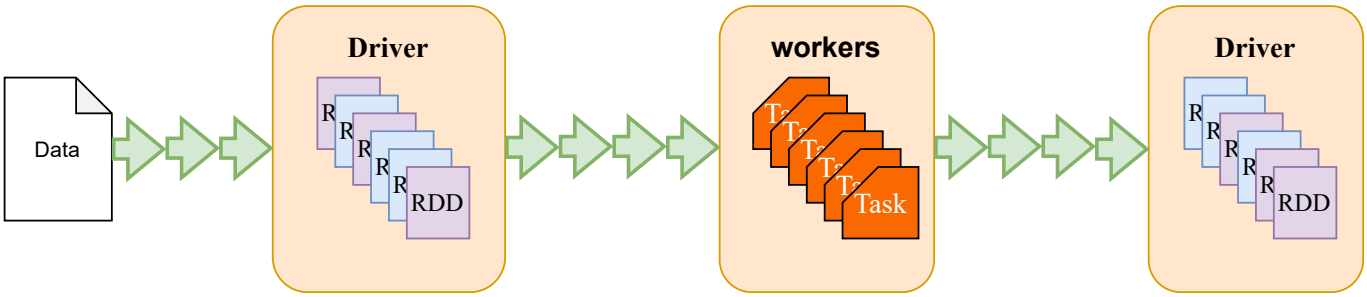
Le traitement en micro-batch



Le traitement en micro-batch consite à charger les données par petits lots appelés **micro-batch** au lieu de tous charger d'un coup.

Interet : traitement au fur et à mesure par petit groupe sans attendre d'avoir tout recuperer.

Traitement en micro-batch



Le traitement en micro-batch est particulierement adaptées aux données tres volumineuses ou au données incrementales.

Traitement en temps réel (streaming)

Le traitement en temps réel consiste à traiter les données au fur et à mesure de leur arrivée.

C'est le traitement le plus rapide car les données sont traitées au fur et à mesure de leur arrivée.

Traitement en temps réel (streaming)

En pratique, le traitement en temps réel est une aggregation des données traitées en micro-batch en continue.

Les modules de Spark