Quand DevOps rencontre BigData!

@obazoud - Olivier Bazoud

@vhe74 - Vincent Heuschling

Objectifs

Comprendre ce que les outils Bigdata peuvent apporter dans le traitement des données dans un contexte Devops

DEVOXX™ France

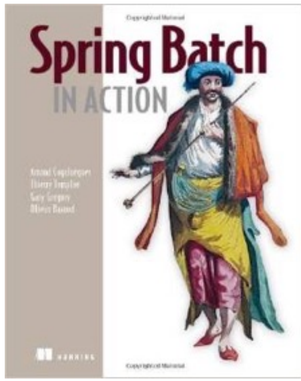
Speakers

Olivier Bazoud - @obazoud

DevOps, Chef/Puppet, Logs, Hadoop

Spring, Node.js, NoSQL

Co-auteur de “Spring Batch in Action”



Vincent Heuschling - @vhe74

Fondateur d’Affini-Tech :“Bigdata Architects”

Bigdata, NoSQL, Hadoop, Spark, Datascience

Co-Animateur du Podcast @Bigdatahebdo

DEVOXX™ France

Agenda

- Présentation du contexte - 15 mn
- Découverte du Hand's on Lab - 5 mn
- A vous de jouer! - 1h30 + 45mn
- Démo du cluster Spark dans le cloud - 10 mn
- Jouer avec le cluster - 15 mn

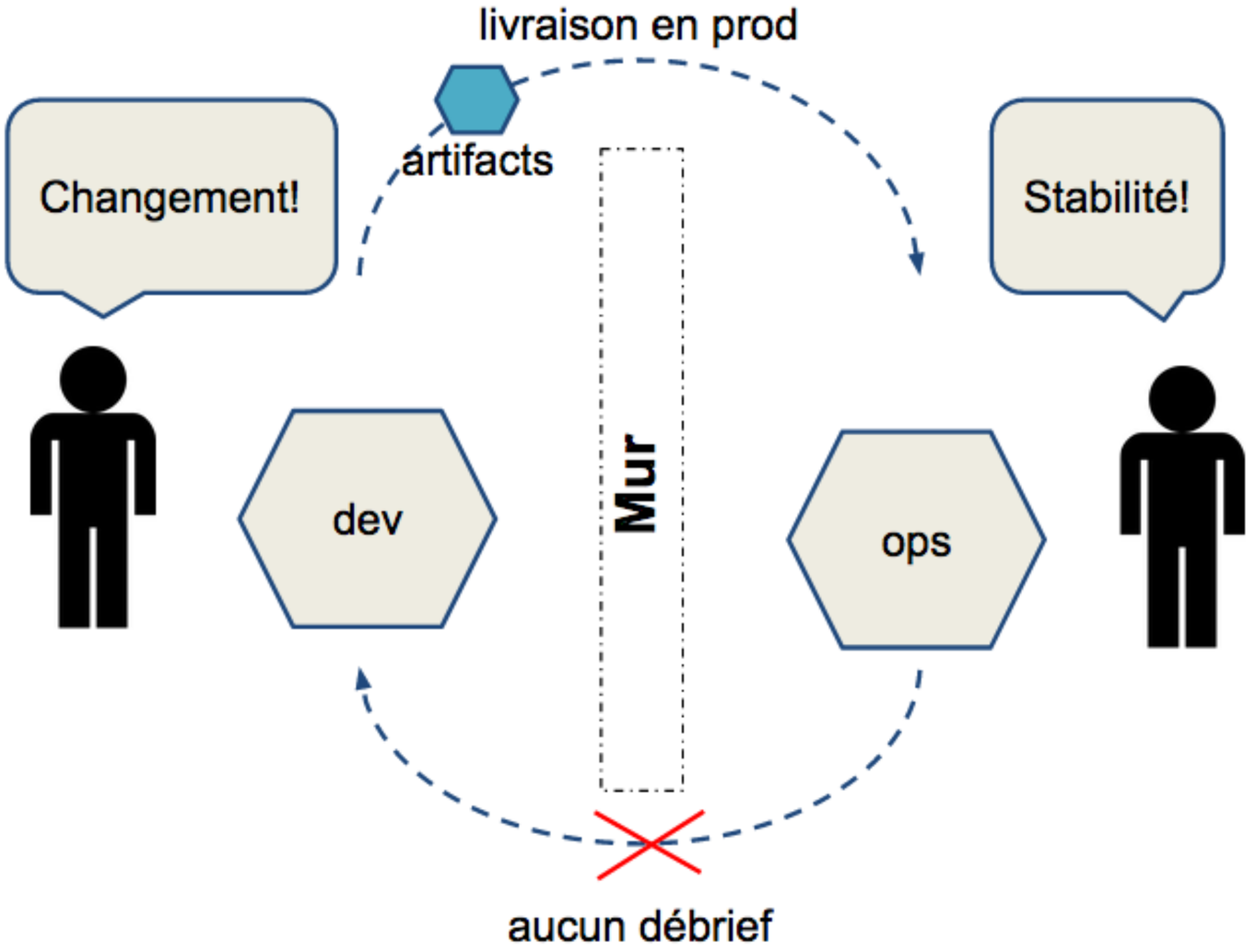
DEVOXX™ France

Contenu du Toolkit

- Spark 1.3
- SBT avec le cache pré-chargé
  & un squelette d'application
- MAVEN avec le cache pré-chargé
  & un squelette d'application
- Sample de données
- https://github.com/obazoud/devoxx-quand-devops-rencontre-bigdata

Contexte

Devops en quelques slides



Devops : Principes CAMS

Culture

Automation

Measurement

Sharing

Devops : Principes CAMS

Casser les silos
L’humain avant les process
Esprit d’équipe

“Infrastructure as code”
Déploiement continue

Monitoring
Gestion centralisée des logs

Dashboards, KPI
L'amélioration continue
“Fast feedbacks”

Analyse de logs

Plus de serveurs, c'est plus de logs à analyser!

Possibilités:

- awk, grep…
- Logstash, Fluentd, Flume, …
- Utiliser les outils de type “BigData”

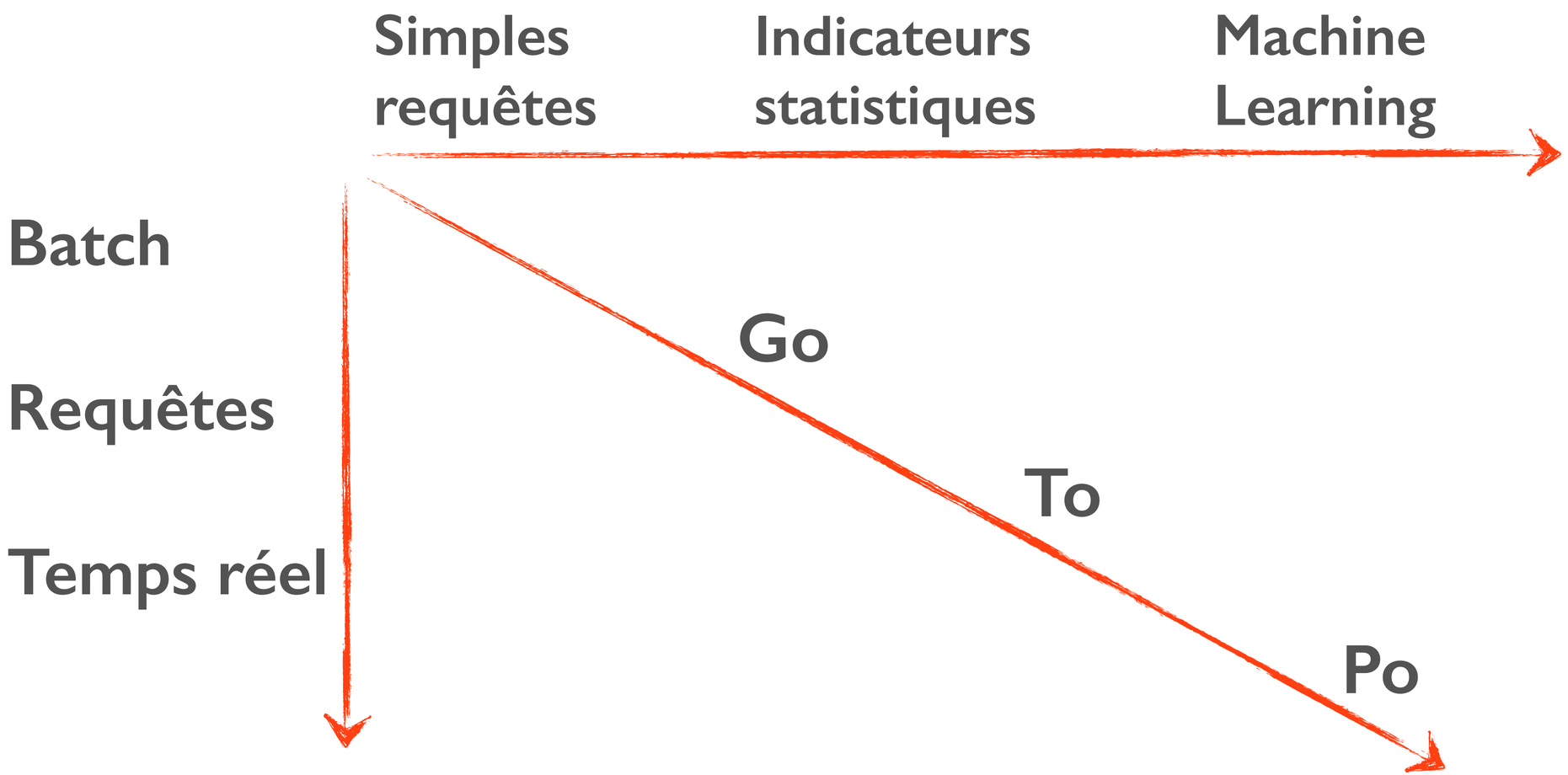
Bigdata

- plus de Data ?
- plus vite ?
- plus d'hétérogénéité dans les Data ?

+

- Rupture Techno
- Open source
- Data-science

Changement de paradigmes

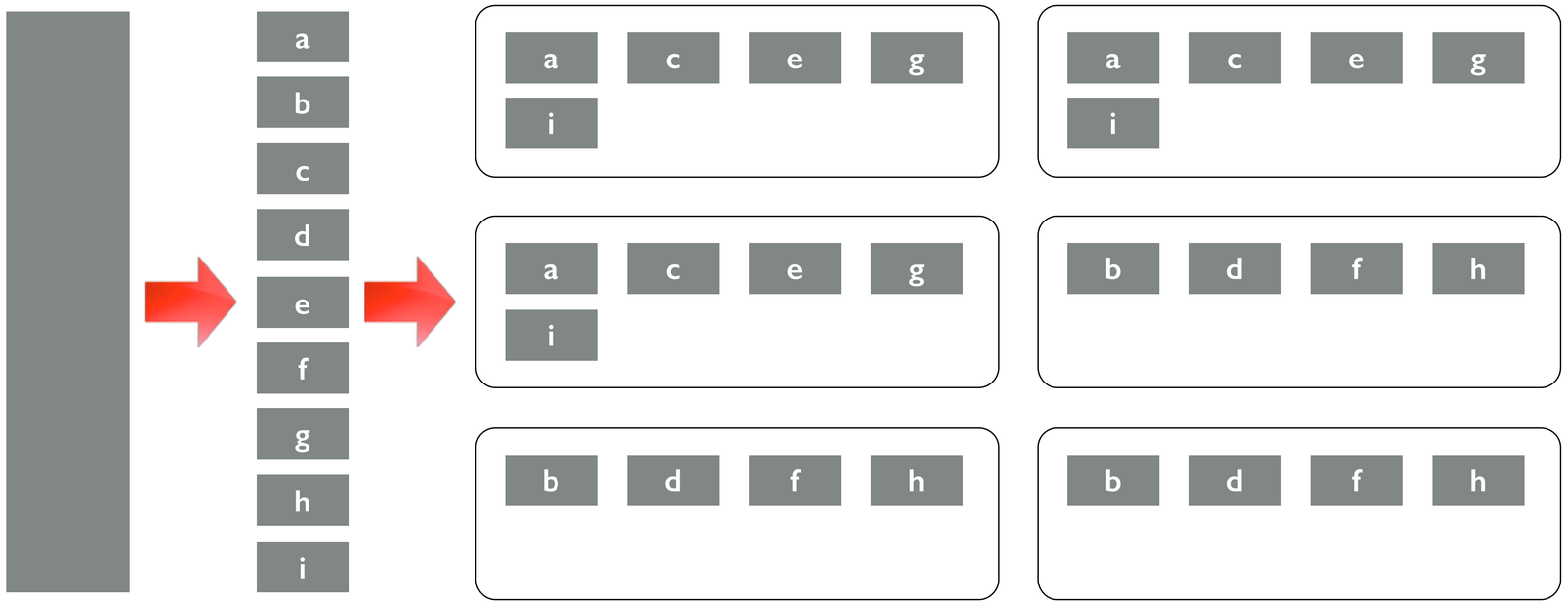


Hadoop ? Spark ?

HDFS
Map-Reduce
Opensource
Google en 06

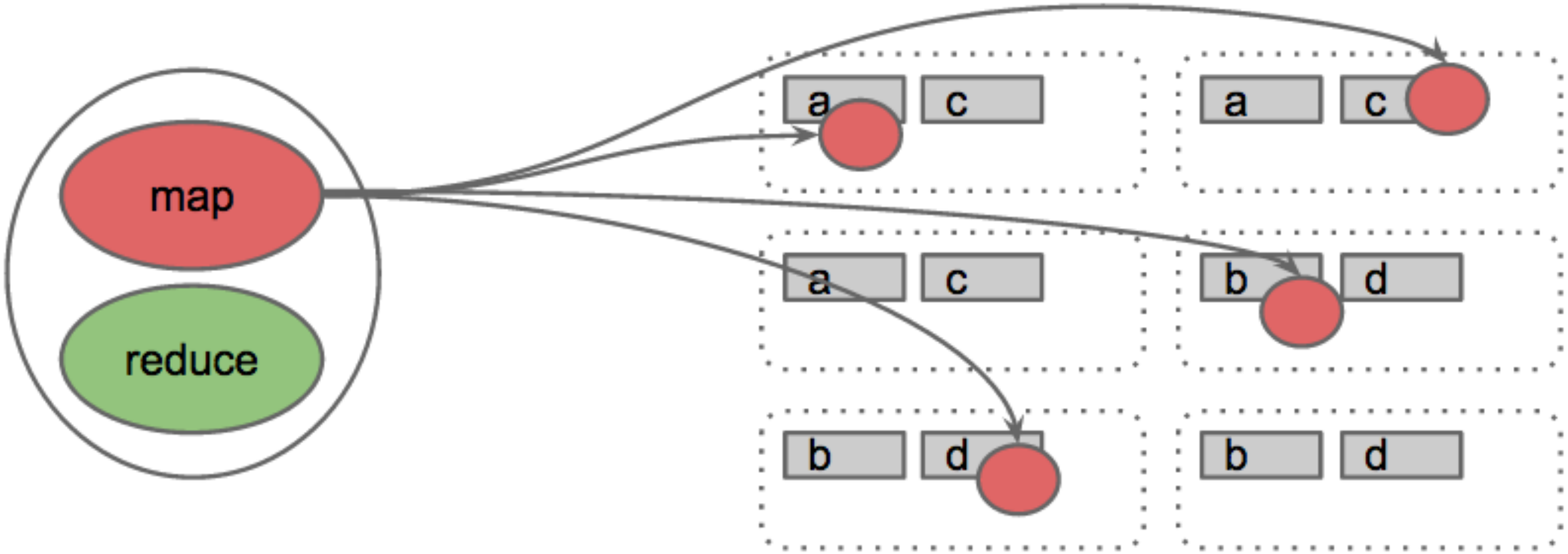
Fonctionnel
In-Memory
Midsize Data
Berkeley AMPLab

Hadoop : HDFS



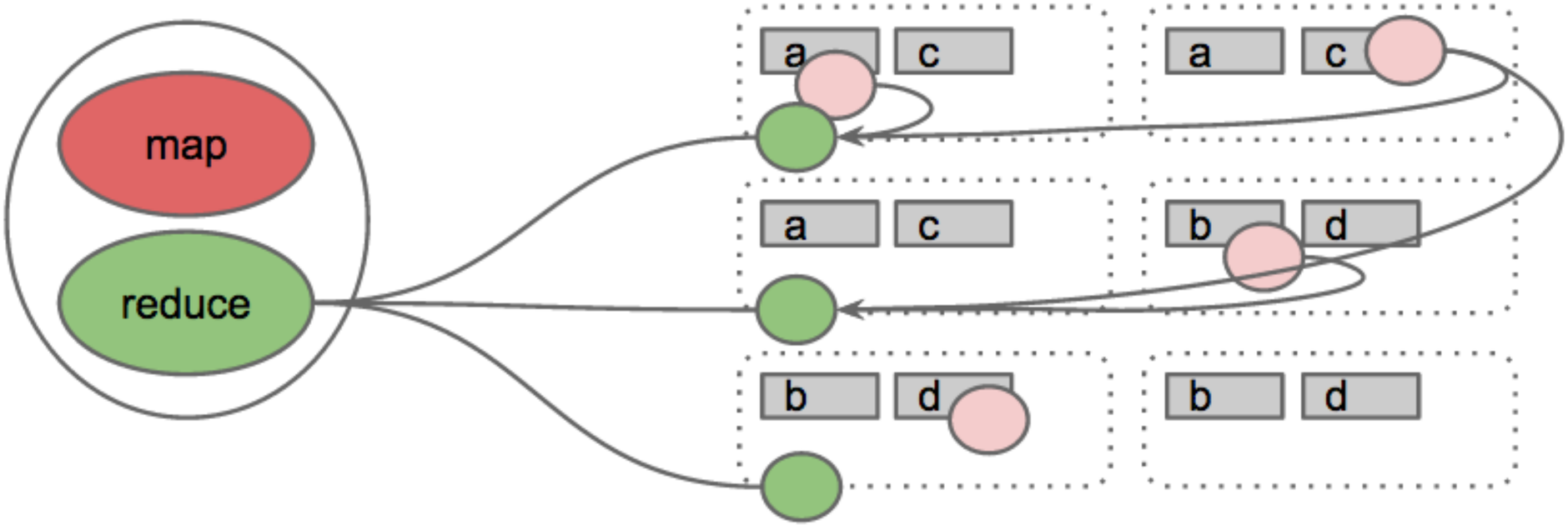
Splitter les fichiers pour pouvoir les traiter en parallèle sans limite de taille

Hadoop : Map-Reduce



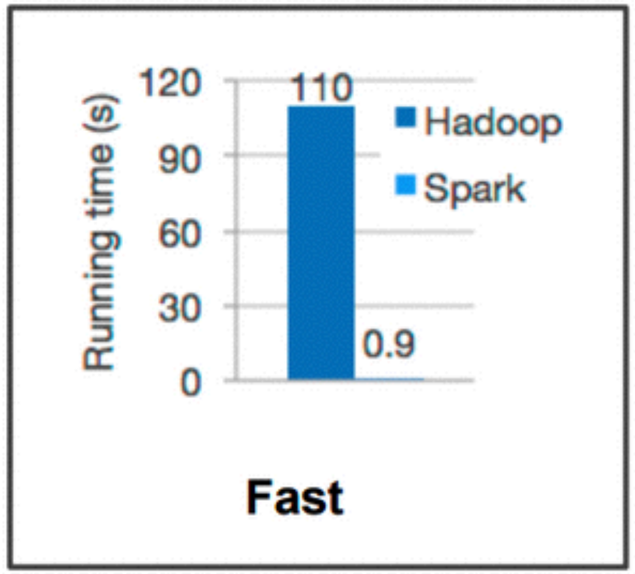
Map : distribuer les traitements sur tous les noeuds du cluster ou des blocs de données sont présents.

Hadoop : Map-Reduce



Reduce : Synthétiser les résultats du Map

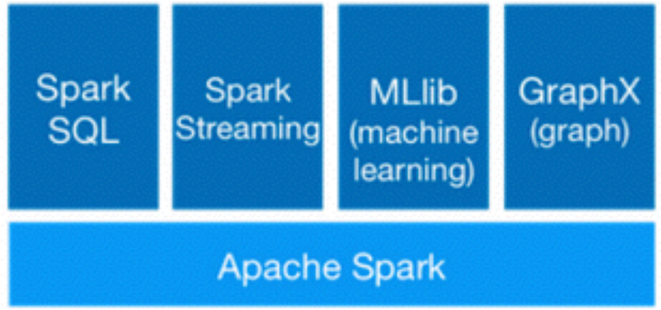
Spark : Mieux car en mémoire



Fast

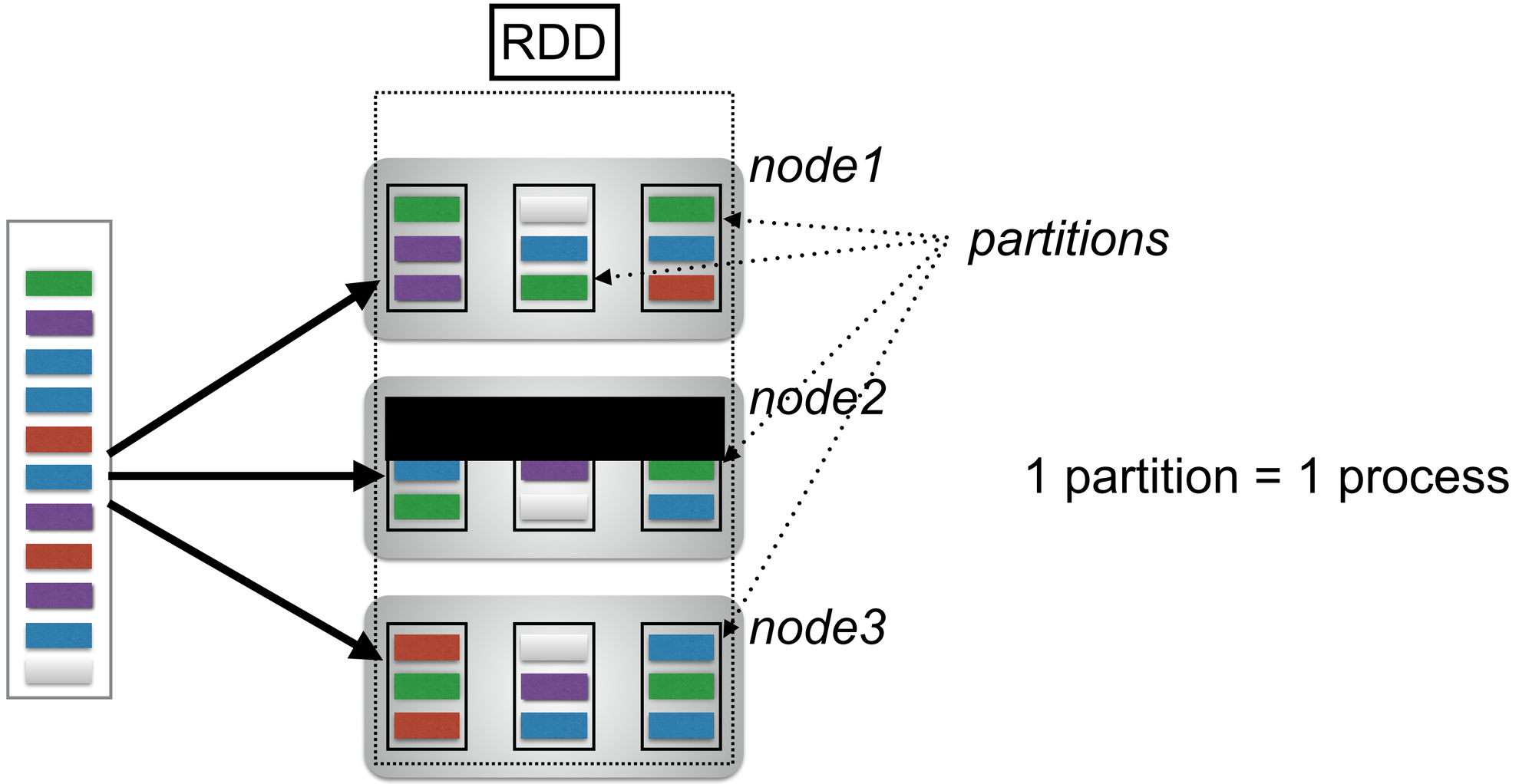
```
val file = spark.textFile("hdfs://...")

val counts = file.flatMap(line => line.
split(" "))
.map(word => (word, 1))
.reduceByKey(_ + _)

counts.saveAsTextFile("hdfs://...")
```

Functional

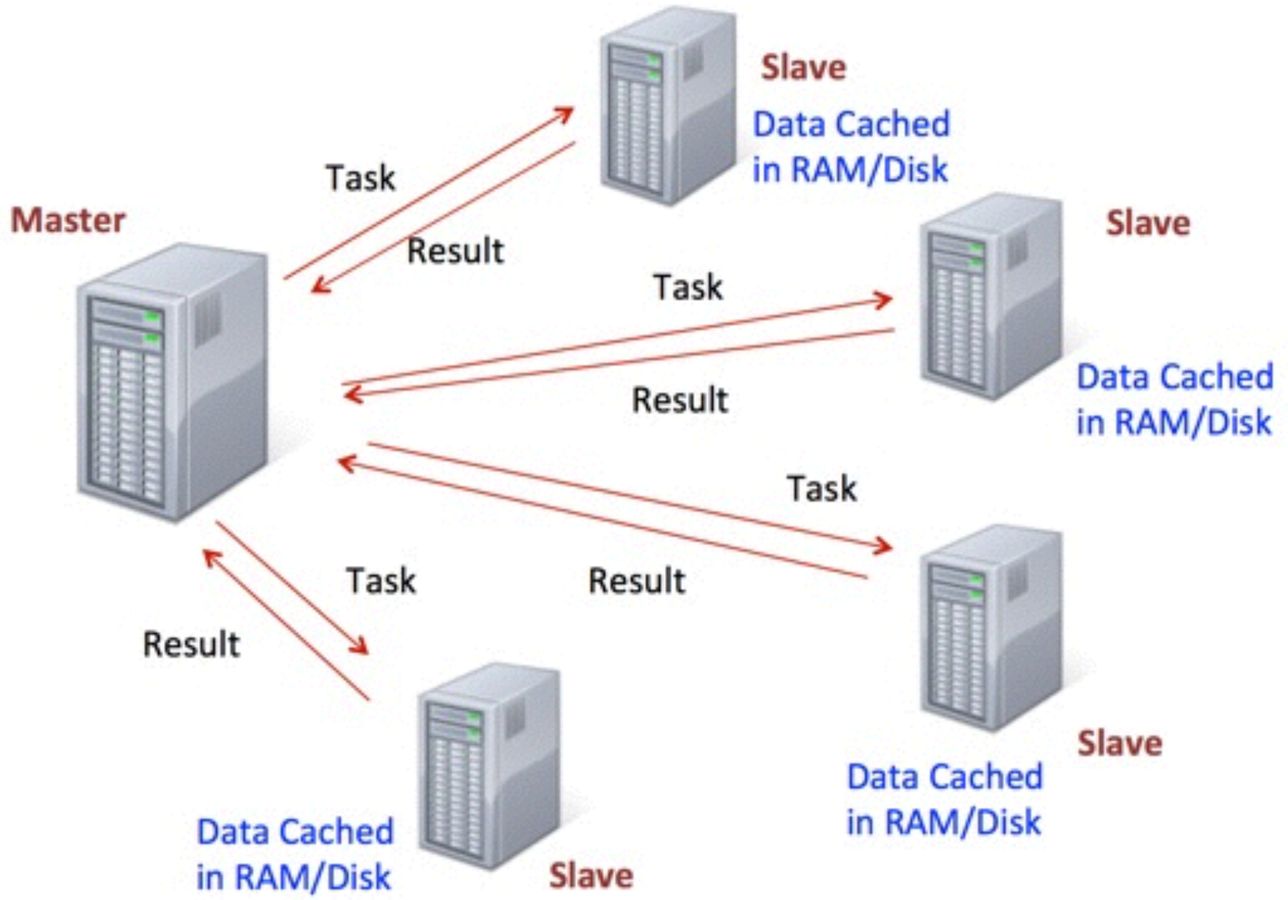


Growing Ecosystem

Spark : RDD



Spark Execution Model



Spark : Fonctionnel

```
val textFile = sc.textFile("hdfs://somefile")

val wordCounts = textFile
                 .flatMap(line => line.split(" "))
                 .map(word => (word, 1))
                 .reduceByKey((a, b) => a + b)
```

Hands on lab

Agenda

1. Manipuler des données avec le REPL
2. Analyser des Apache logs
3. Détecter des anomalies dans des logs réseau
4. Utiliser un cluster

Comptes Google

Pour vous connecter au cluster vous vous authentifierez avec vos comptes Google.

http://bit.ly/devops-bigdata

Repo Github

Updatez le repository founit dans le « package usb »

git clone https://github.com/obazoud/devoxx-quand-devops-rencontre-bigdata.git

Exercice 1

Prise en main de Spark à travers le REPL.

- Spark-shell
- Lire un fichier
- Comptage de lignes
- Wordcount

Exercice 2

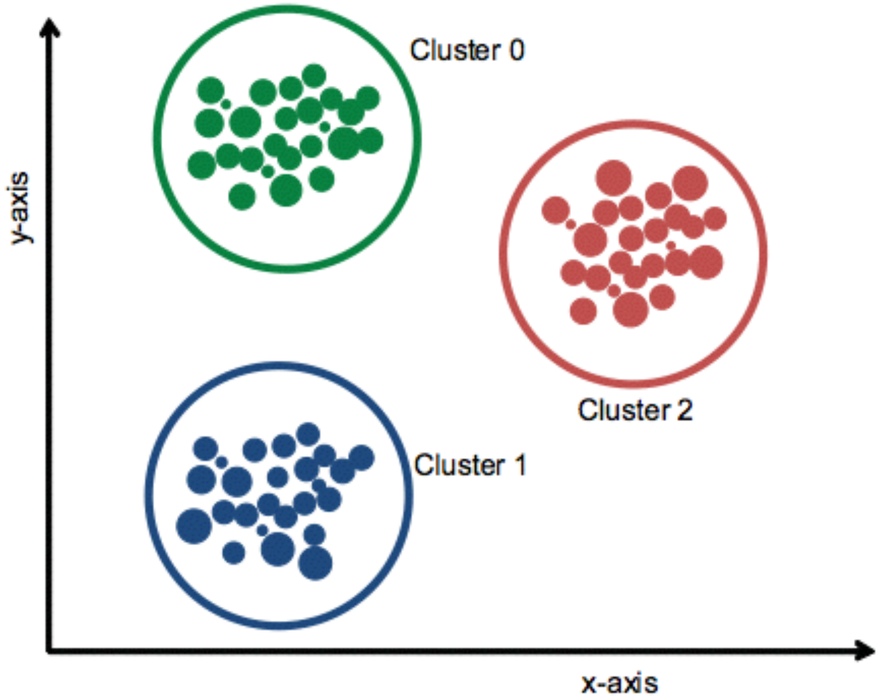
Construction d'une application pour analyser des logs Apache

- En Java
- En Scala

Exercice 3

Regroupement d'événements dans des captures réseau par « Machine Learning non supervisé »

K-Means de Spark ML-Lib



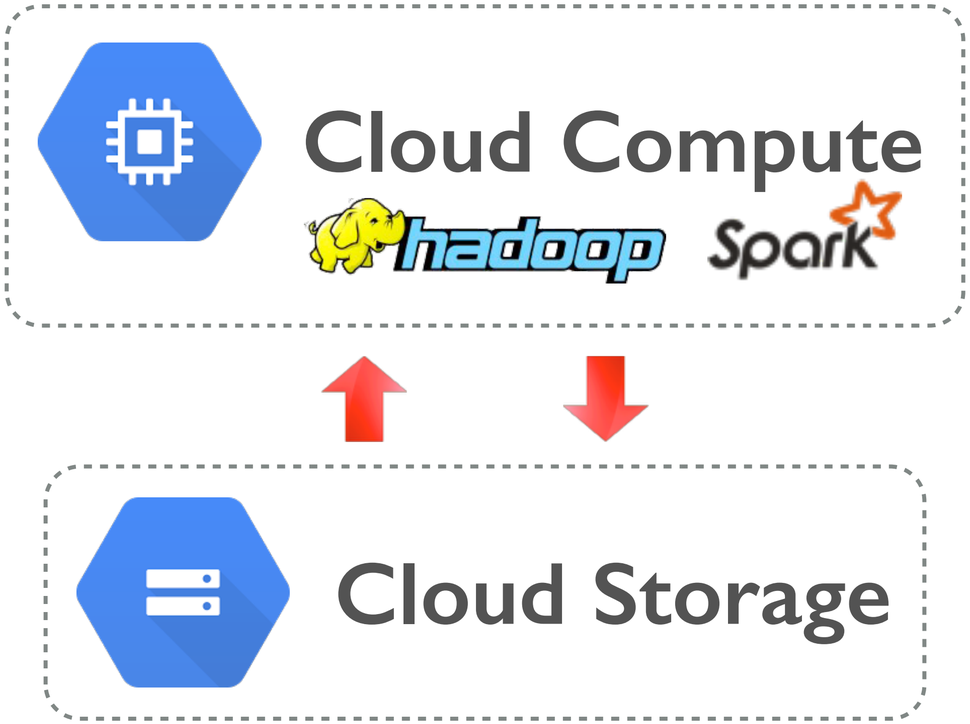
Démo Spark @ Cloud

Spark sur Google Cloud

Emulation Hadoop/HDFS par le Google Cloud storage :
gs://handsondevoxxfr/kddcup.data

Storage browser en mode web :
https://console.developers.google.com/project/blast-machine-201504/storage/browser/handsondevoxxfr/

Spark sur Google Cloud



Votre code sur le Cloud

Utilisation du REPL

Utilisation du REPL Spark depuis le master node du cluster.

Partage des ressources du cluster en limitant le nombre de coeurs et la mémoire alloué à chaque application

Connexion au master

Connexion en pseudo-ssh :

https://cloudssh.developers.google.com/projects/blast-machine-201504/zones/europe-west1-b/instances/spark-m?authuser=0&hl=fr

Check de l'environnement Spark :

http://spark-m:8080/

Lancement du REPL

```
$ MASTER=spark://spark-m:7077 bin/spark-shell \
          --total-executor-cores 8 \
          --executor-memory 2G \
          --name your_nickname

## check on http://spark-m:8080
```

Q & A