Spark basic and using RDD

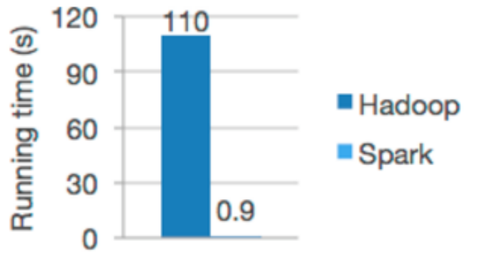
Week 1

What Is Apache Spark?

- 源自Berekely AMP 實驗室的集群式計算平台引擎，運用在**一般目的性的集群式計算**

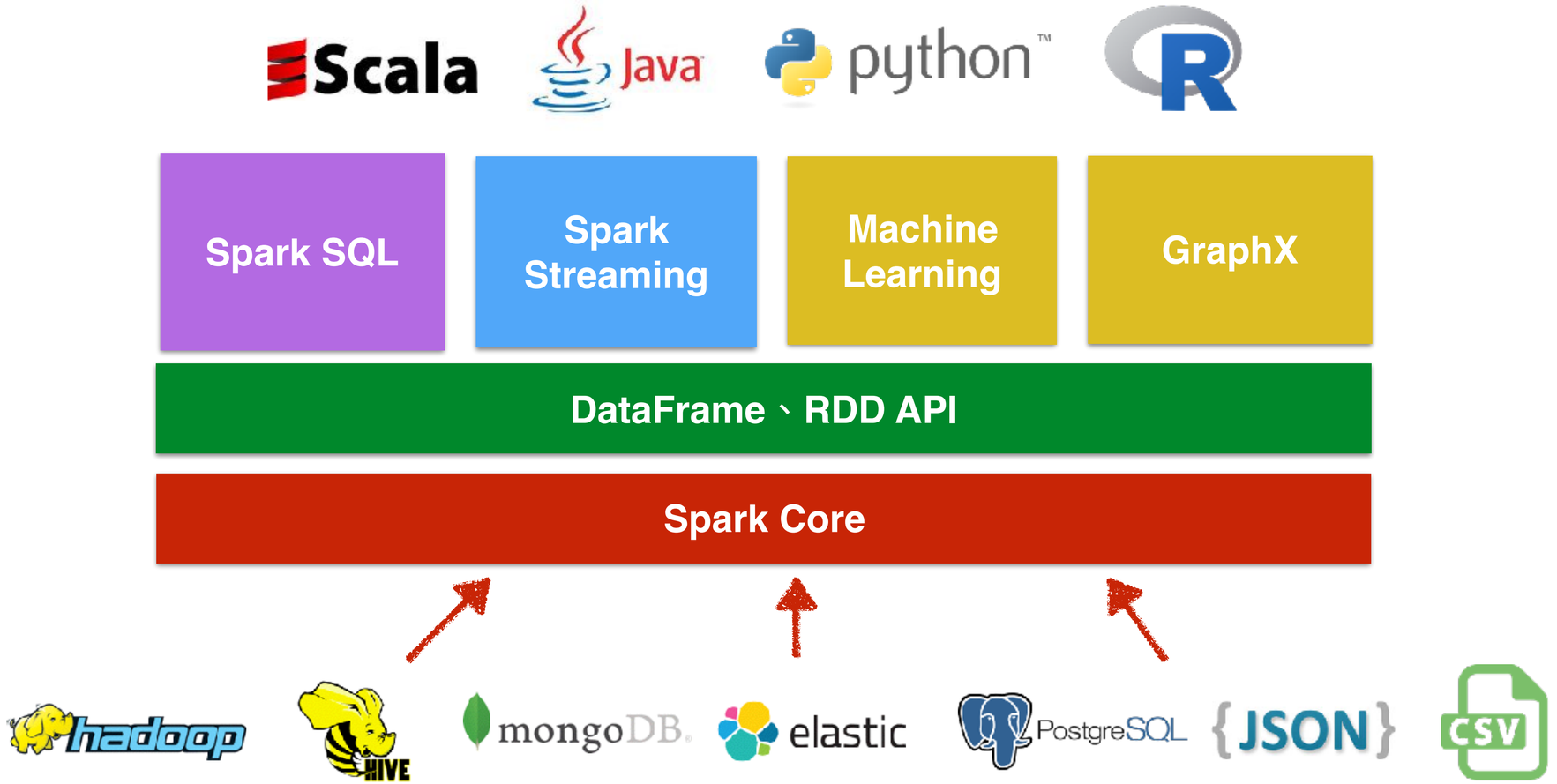
- 用在**大量批次分析**、**交互查詢**、**串流處理**、**反覆迭代**的分佈計算

- 刷新資料排序世界紀錄 - 在190台VM, **30分鐘**內完成**100TB**的資料排序 (每分鐘4TB)

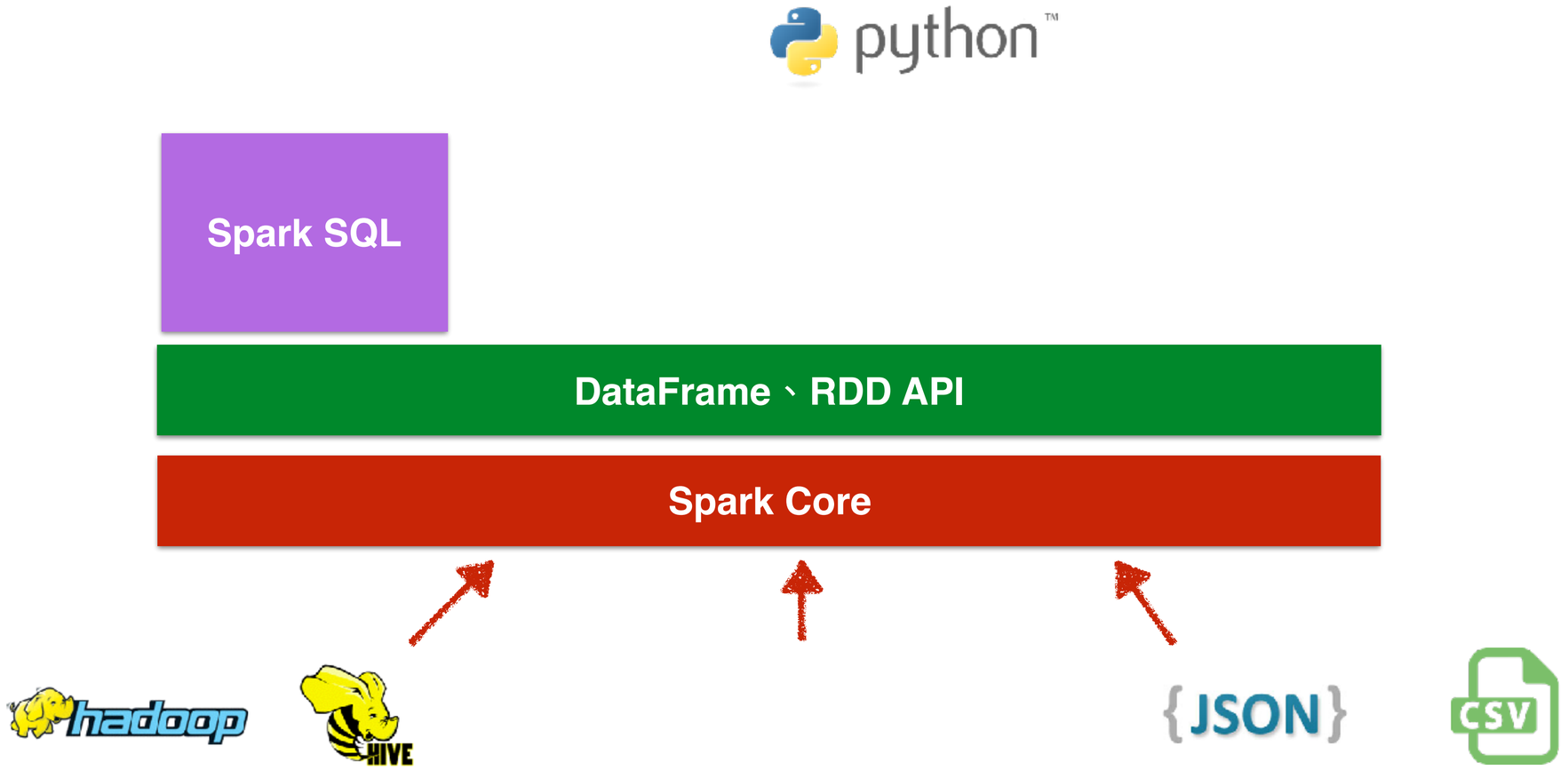


Logistic regression in Hadoop and Spark

Spark征服大數據的全面性架構



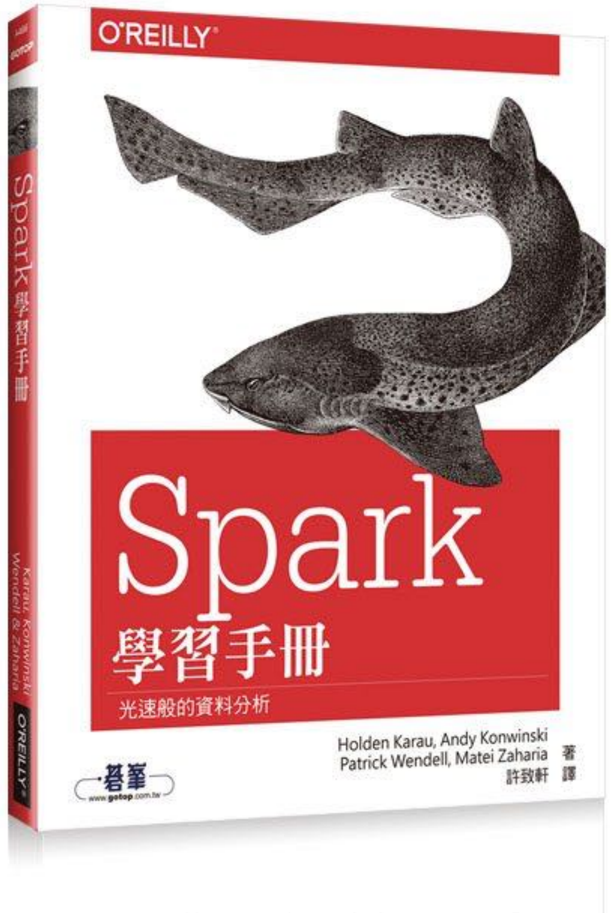
What we will learn?



Reference

主要基本書籍

http://pythonsparkhadoop.blogspot.tw/



困難、細節一點

環境安裝與設定

Download Scala

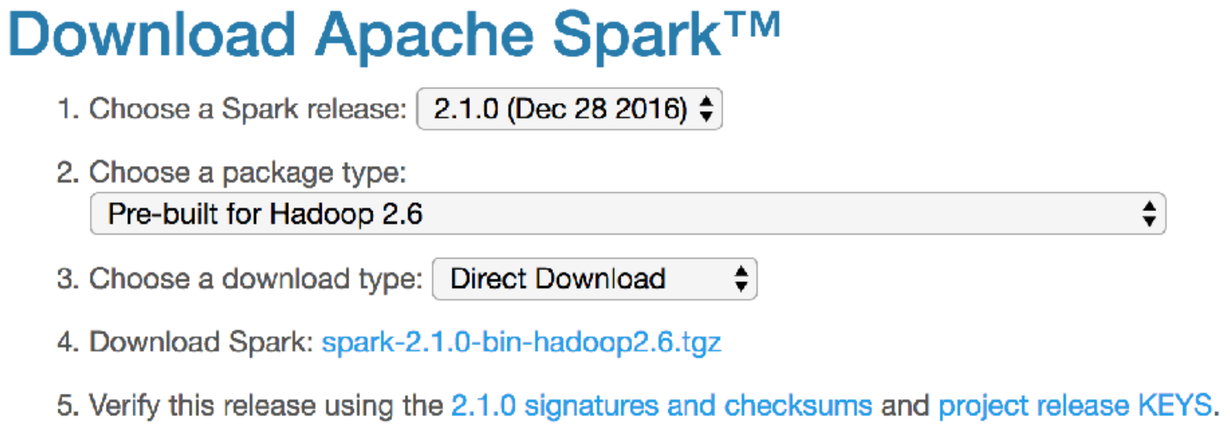
1. 下載 Scala 2.11.8
https://www.scala-lang.org/download/2.11.8.html

2. folder move 到/usr/local/share/scala
(mv scala-2.11.8 /usr/local/share/scala)
並加入到 ~/.bash_profile

```
# Scala
export SCALA_HOME=/usr/local/share/scala
export PATH=$PATH:$SCALA_HOME/bin
```

Download Spark

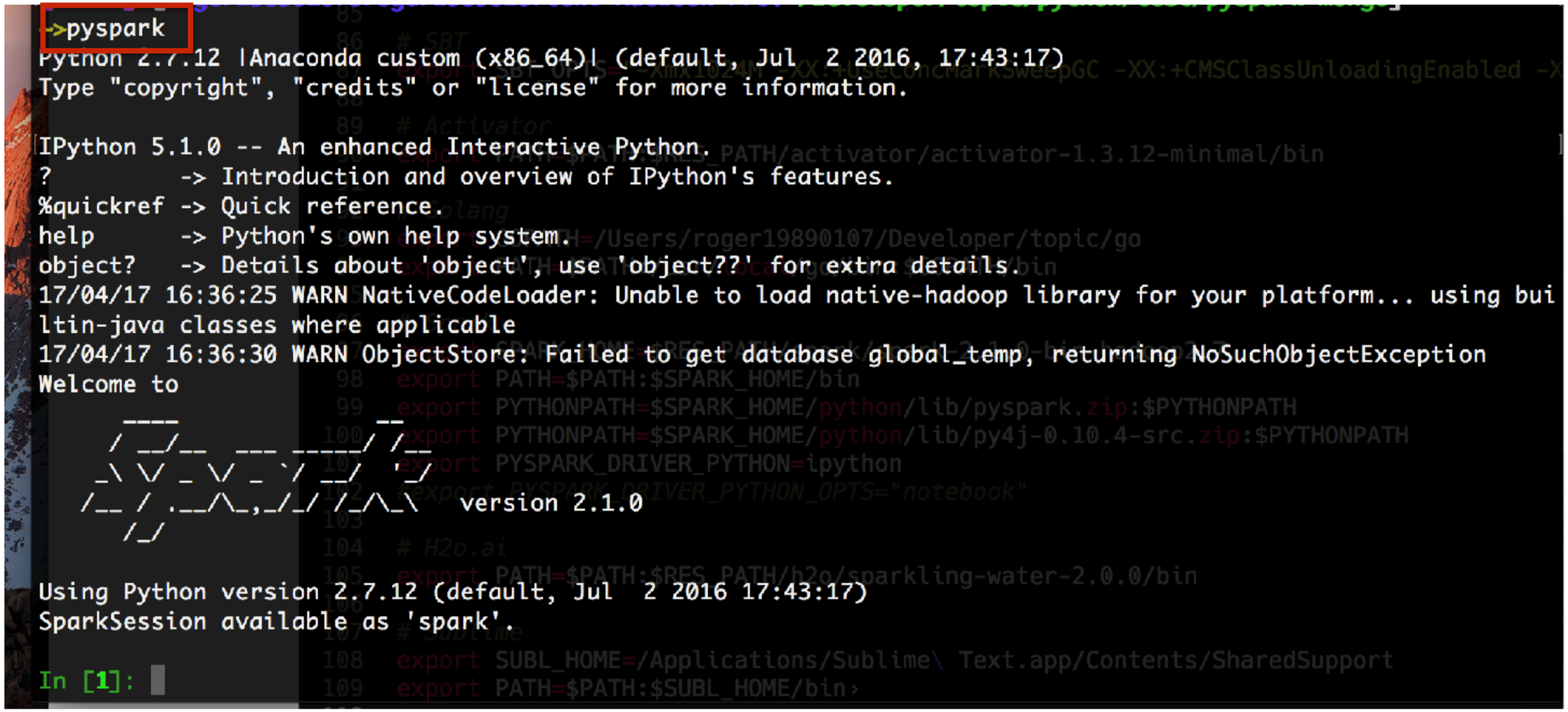
1. 下載 Spark 2.1.0
   http://spark.apache.org/downloads.html



2. 並加入到 ~/.bash_profile

```
# Spark
export SPARK_HOME=$RES_PATH/spark/spark-2.1.0-bin-hadoop2.7
export PATH=$PATH:$SPARK_HOME/bin
export PYTHONPATH=$SPARK_HOME/python/lib/pyspark.zip:$PYTHONPATH
export PYTHONPATH=$SPARK_HOME/python/lib/py4j-0.10.4-src.zip:$PYTHONPATH
export PYSPARK_DRIVER_PYTHON=ipython
```

Check Pyspark



Check pyspark shell

一切從 pyspark shell 開始

Spark 操作進入點

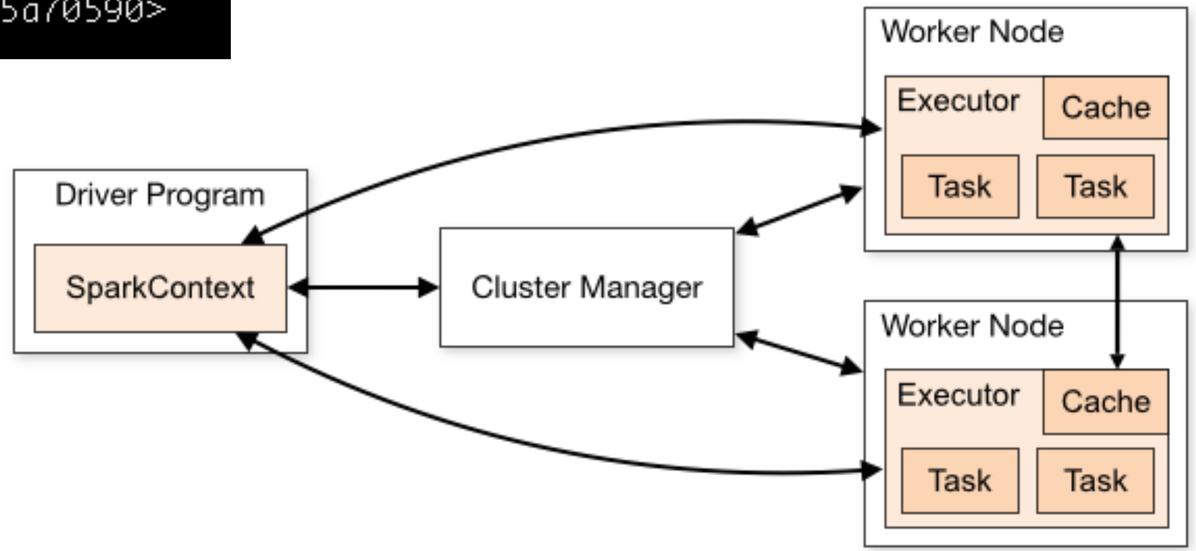
sc: SparkContext => 操作進入點, 操作RDD (spark上處理的分散式資料集)

```
In [16]: sc
Out[16]: <pyspark.context.SparkContext at 0x10588c850>

In [17]: sc.master
Out[17]: u'local[*]'

In [18]: spark
Out[18]: <pyspark.sql.session.SparkSession at 0x105a70590>
```

spark: SparkSession =>
spark2.0加入, 操作SparkSQL、dataframe



簡單的操作

```
In [20]: lines = sc.textFile("news.txt")

In [21]: lines
Out[21]: news.txt MapPartitionsRDD[34] at textFile at NativeMethodAccessorImpl.java:0
```

textFile可以讀檔成RDD

RDD想成是一個分散式的Array

```
In [22]: lines.take(2)
Out[22]:
[u'Google, which not along ago was using artificial intelligence to identify cat pictures, has moved
onto something bigger -- breast cancer.',
 u'Google announced Friday that it has achieved state-of-the-art results in using artificial intellig
ence to identify breast cancer. The findings are a reminder of the rapid advances in artificial intel
ligence, and its potential to improve global health.']
```

textFile出來的RDD會是
每筆資料就是**文字檔的每一行**

take(2) =>
取前兩行load到記憶體

```
In [23]: lines.count()
Out[23]: 12
```

count => 看看總共有幾行

簡單的操作

```
In [26]: lines.map(lambda line: len(line)).collect()
Out[26]: [137, 251, 310, 236, 320, 75, 368, 184, 234, 302, 294, 0]
```

map =>
對每一筆data做事,
這裡是取每行文字長度

```
In [32]: lines.map(lambda line: len(line)).filter(lambda x: x > 100).collect()
Out[32]: [137, 251, 310, 236, 320, 368, 184, 234, 302, 294]
```

filter =>
篩選符合條件的資料,
這裡是選出長度 > 100的資料

```
In [33]: lines.map(lambda line: len(line)).filter(lambda x: x > 100).reduce(lambda x, y: x + y)
Out[33]: 2636
```

reduce => 對每一筆資料做一件事並累積到x

EXAMPLE:
step 1: 137 + 251 = 388
step 2: 388 + 310 = 698
….
step n: 2342 + 294 = 2636

RDD基本操作類型

```
lines = sc.textFile("news.txt")
```

create

RDD
sc.textFile

action → take

```
In [22]: lines.take(2)
Out[22]:
[u'Google, which not along ago was using artificial intelligence to identify cat pictures, has moved onto something bigger -- breast cancer.',
 u'Google announced Friday that it has achieved state-of-the-art results in using artificial intelligence to identify breast cancer. The findings are a reminder of the rapid advances in artificial intelligence, and its potential to improve global health.']
```

action → count

```
In [23]: lines.count()
Out[23]: 12
```

transform → RDD
map(取len(line))

action → collect

全部load到記憶體

```
[137, 251, 310, 236, 320, 368, 184, 234, 302, 294]
```

transform → RDD
filter(x > 100)

action → collect

```
[137, 251, 310, 236, 320, 368, 184, 234, 302, 294]
```

action → reduce(x = x + y)

```
2636
```

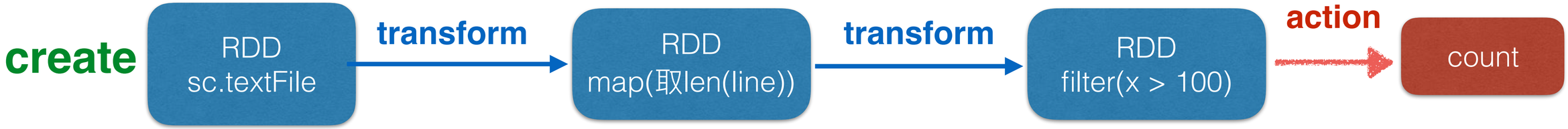
* *create* 和 *transform* 不會執行
除非遇到 *action*

RDD 重複計算問題

```
filter_lines = lines.map(lambda line: len(line)).filter(lambda x: x > 100)
```


```
In [41]: filter_lines.count()
Out[41]: 10
```

action觸發整條RDD流程




```
In [42]: filter_lines.reduce(lambda x, y: x + y)
Out[42]: 2636
```

整條重做！

RDD cache

```
In [42]: filter_lines.reduce(lambda x, y: x + y)
Out[42]: 2636
```

```
filter_lines = lines.map(lambda line: len(line)).filter(lambda x: x > 100)
```

create

RDD
sc.textFile

transform

RDD
map(取len(line))

transform

RDD
filter(x > 100)

cache

RDD
persist

action

count

第一次整條做
記錄紫色狀態

```
filter_lines.persist()
PythonRDD[50] at RDD at PythonRDD.scala:48
```

從紫色開始

action

reduce
(x = x + y)

```
In [41]: filter_lines.count()
Out[41]: 10
```

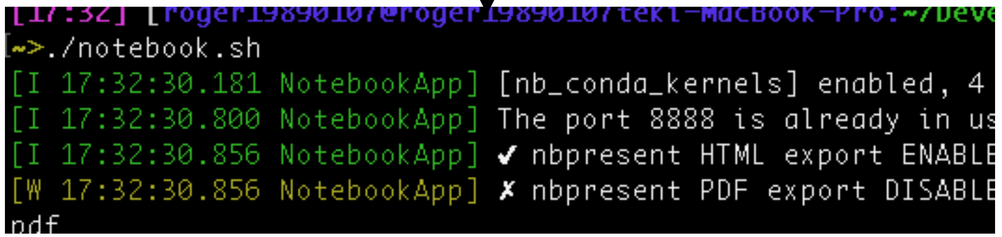
用Jupyter深入學習RDD

Pyspark jupyter ENV

```
export PYSPARK_DRIVER_PYTHON=jupyter
export PYSPARK_DRIVER_PYTHON_OPTS="notebook"
pyspark
```

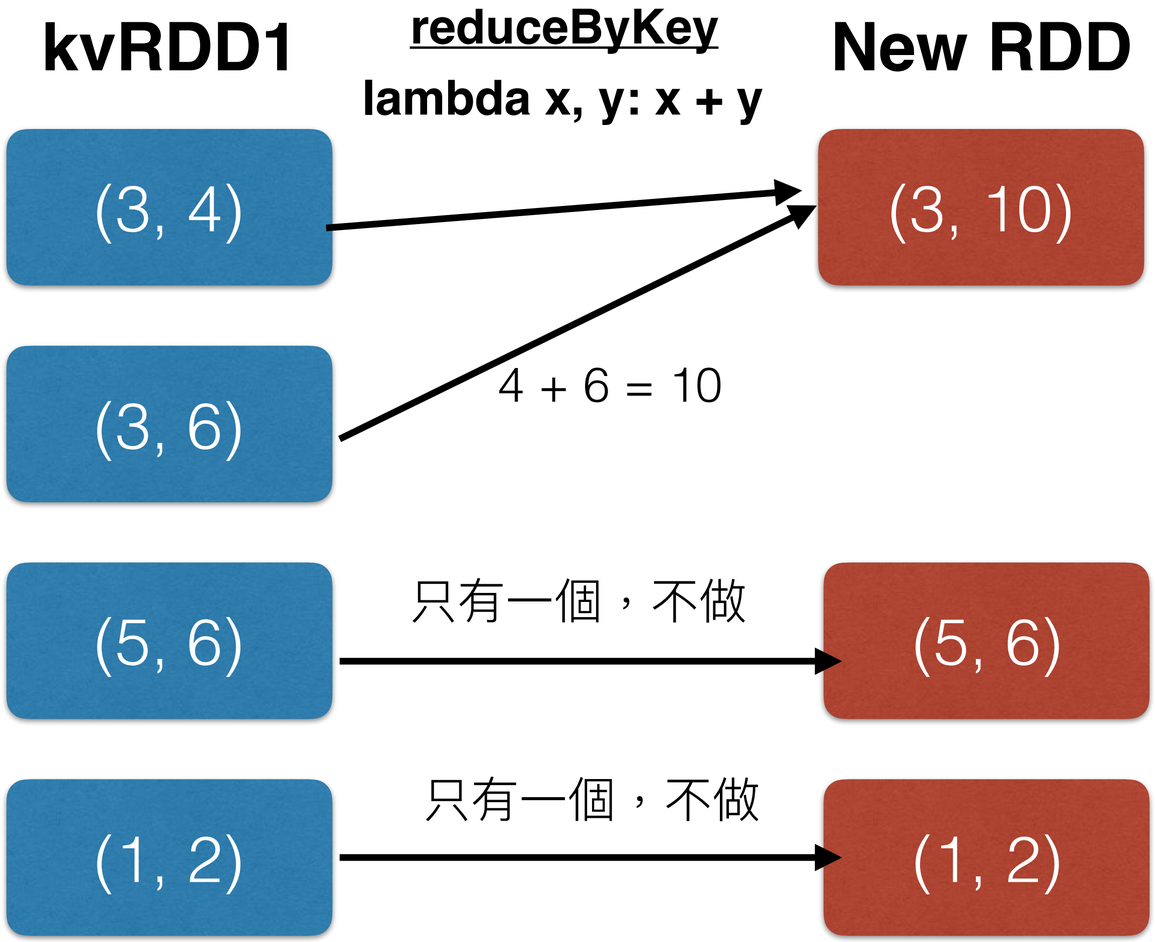
create **notebook.sh**

設定執行file權限

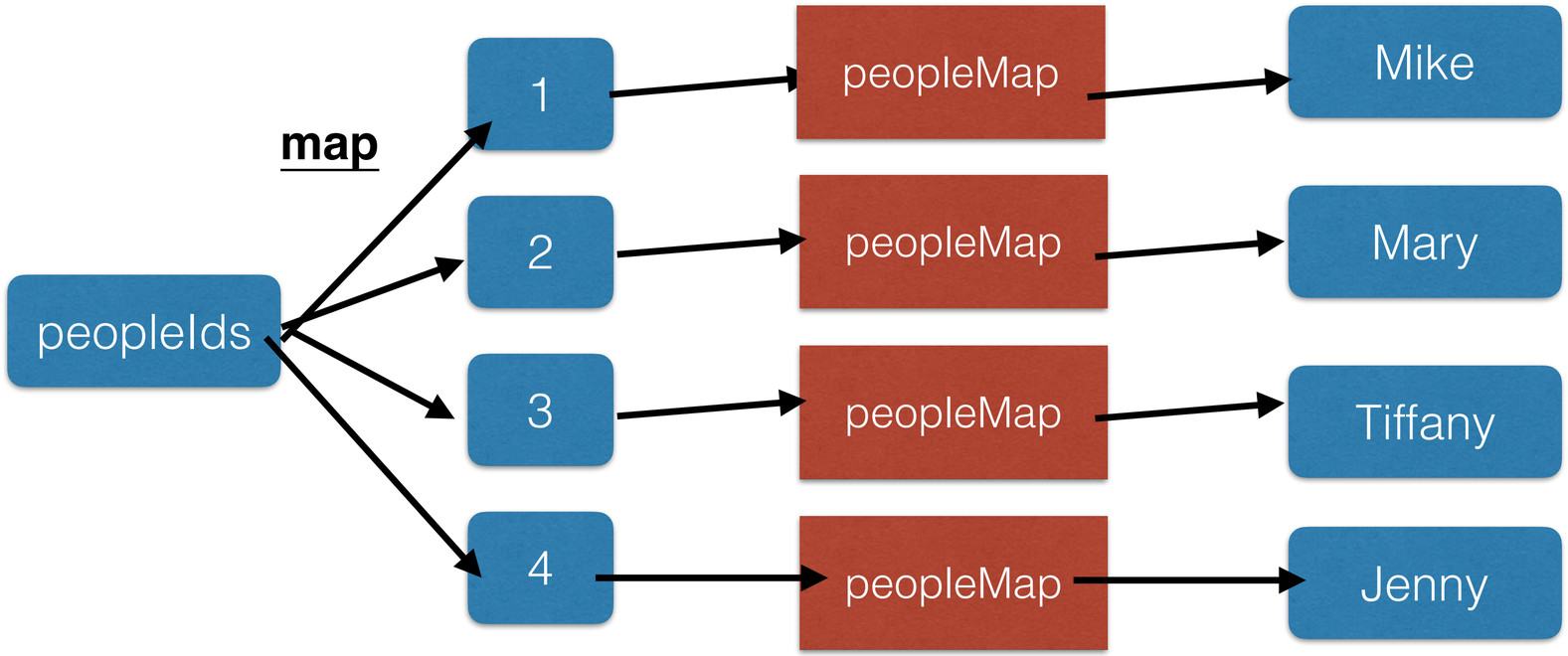


執行notebook.sh

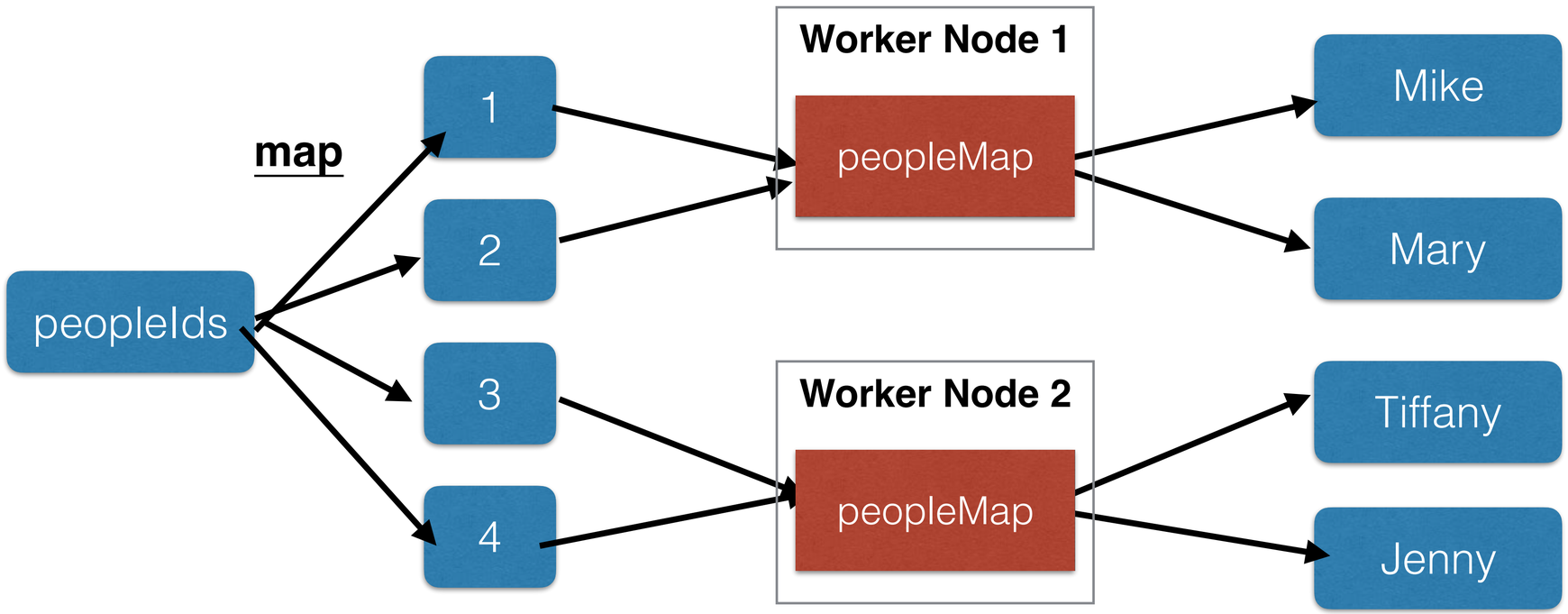
reduceByKey



Before broadcast



After broadcast



submit pyspark file!

```python
from __future__ import print_function
from pyspark import SparkContext, SparkConf

# how to run: spark-submit intro-rdd.py

if __name__ == '__main__':

    # spark context
    conf = SparkConf().setAppName("FirstSpark").setMaster("local[*]")
    sc = SparkContext(conf=conf)

    # log off
    logger = sc._jvm.org.apache.log4j
    logger.LogManager.getLogger("org").setLevel(logger.Level.OFF)
    logger.LogManager.getLogger("akka").setLevel(logger.Level.OFF)

    # process text
    lines = sc.textFile('../dataset/news.txt')
    print(lines.count())

    # save text
    lines.filter(lambda line: 'Google' in line) \
        .saveAsTextFile("../outputs/news2")

    sc.stop()
```

spark-submit intro-rdd.py

Homework

＊隨便找一個文章, 讀取它, 然後算word count

ex: "今天天氣好棒! 跟昨天天氣比起來差很多！"
=> (天氣, 2), (今天, 1) …

Hint: 試試 sc.textFile() + jieba

＊要save text file