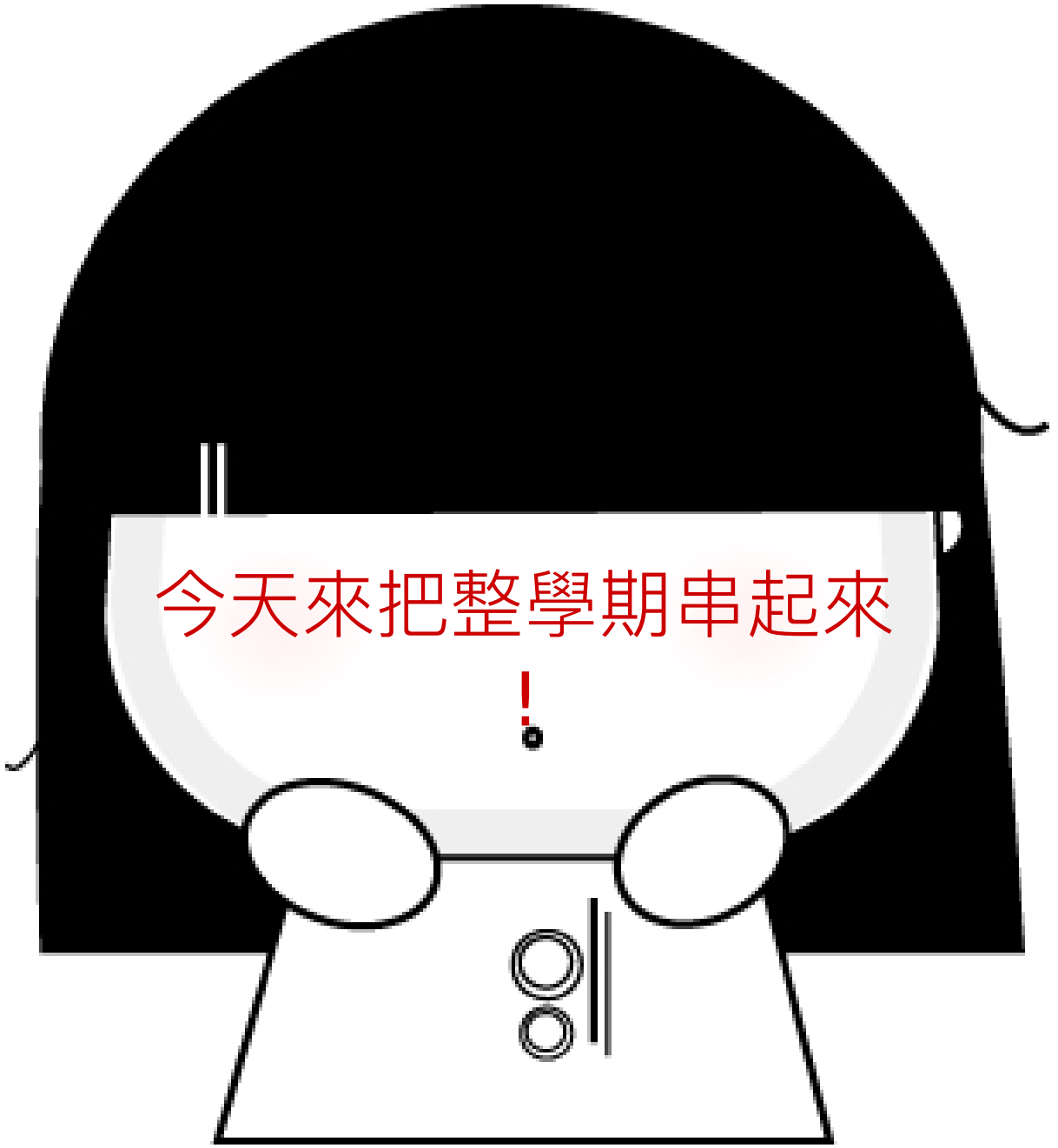
心理與神經資訊學
(Psychoinformatics & Neuroinformatics)

課號：Psy5261　　教室： 博雅 101

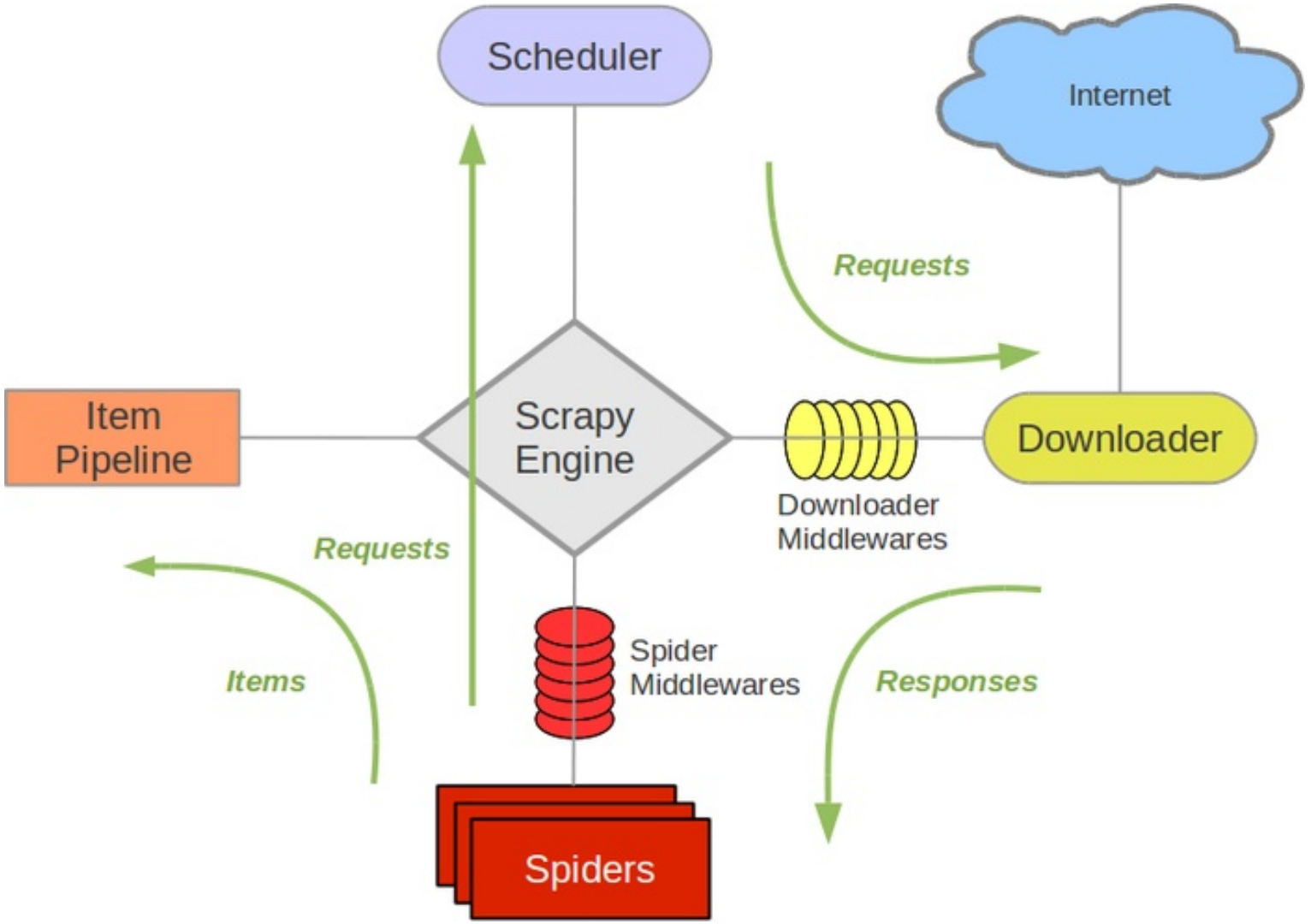
識別碼：227U9340　　時間： 四 234


今天來把整學期串起來
!

總論
(Overview)

觀察法撈大數據

Scrapy 使用 Twisted 的 async I/O

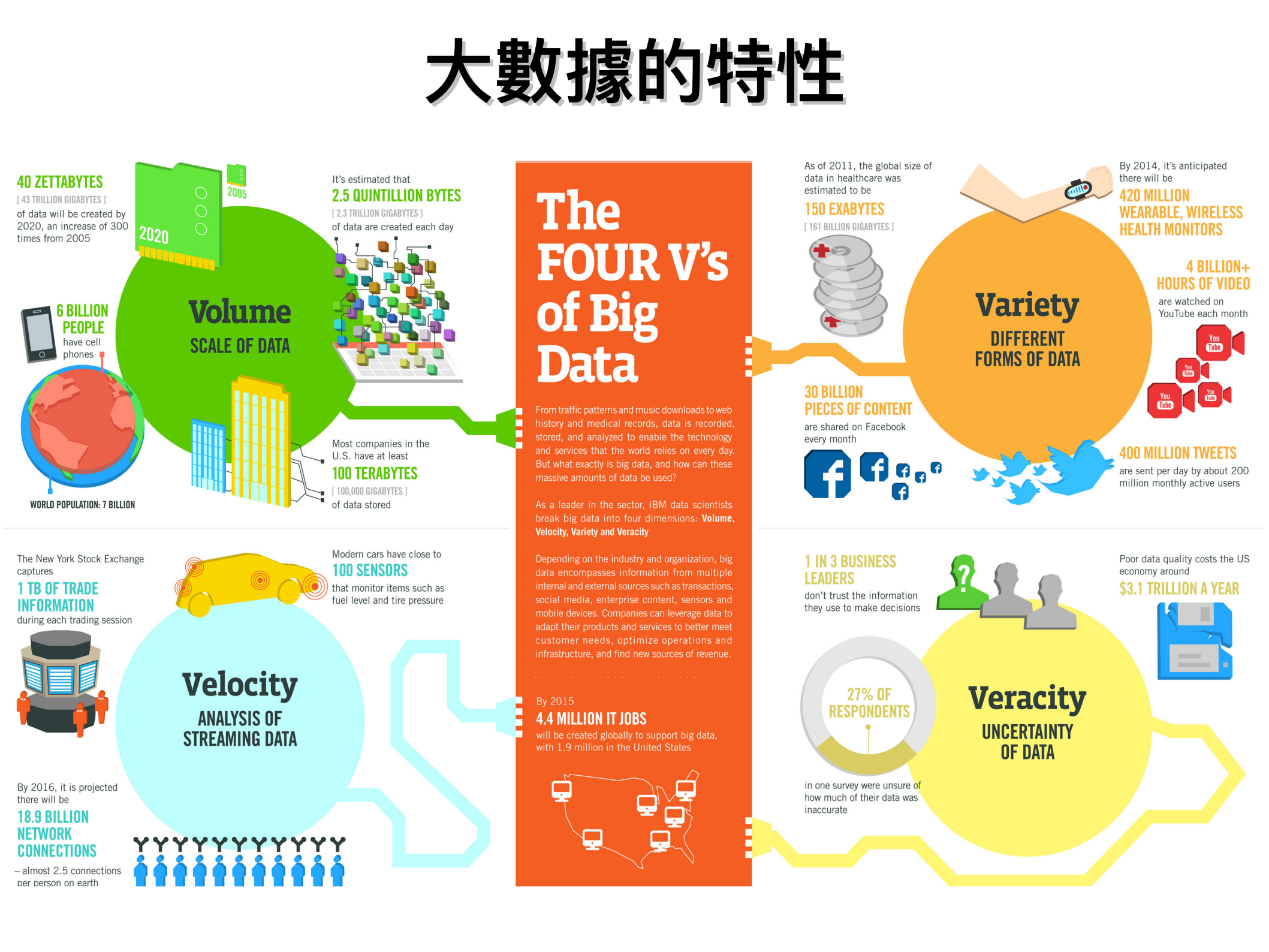


實驗法法收大數據

Javascript & Node.js 使用 async I/O

```
function hell(win) {
 // for listener purpose
 return function() {
   loadLink(win, REMOTE_SRC+'/assets/css/style.css', function() {
     loadLink(win, REMOTE_SRC+'/lib/async.js', function() {
       loadLink(win, REMOTE_SRC+'/lib/easyXDM.js', function() {
         loadLink(win, REMOTE_SRC+'/lib/json2.js', function() {
           loadLink(win, REMOTE_SRC+'/lib/underscode.min.js', function() {
             loadLink(win, REMOTE_SRC+'/lib/backbone.min.js', function() {
               loadLink(win, REMOTE_SRC+'/dev/base_dev.js', function() {
                 loadLink(win, REMOTE_SRC+'/assets/js/deps.js', function() {
                   loadLink(win, REMOTE_SRC+'/src/' + win.loader_path + '/loader.js', function() {
                     async.eachSeries(SCRIPTS, function(src, callback) {
                       loadScript(win, BASE_URL+src, callback);
                     });
                   });
                 });
               });
             });
           });
         });
       });
     });
   });
 };
}
```

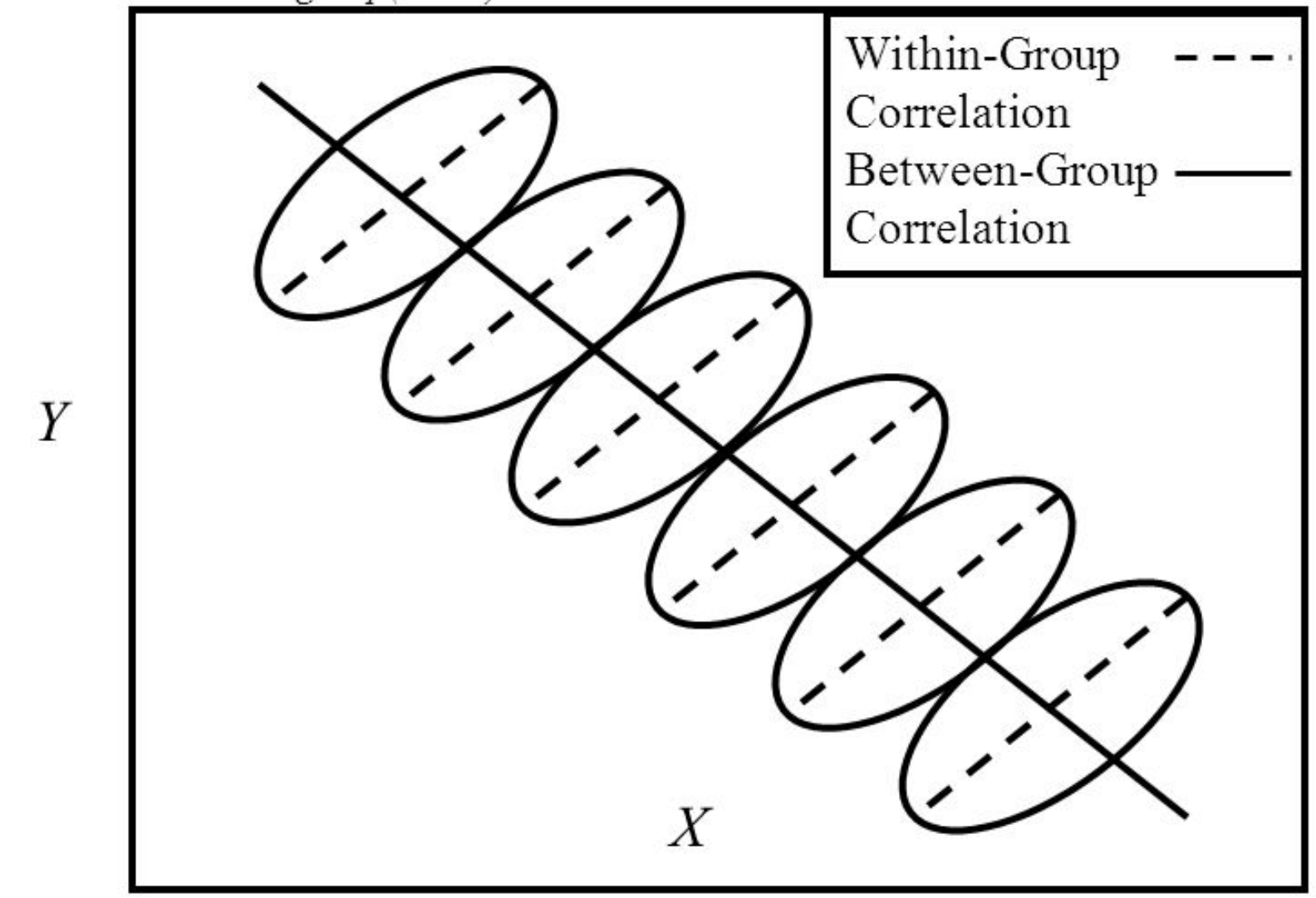
大數據的特性


40 ZETTABYTES
[43 TRILLION GIGABYTES]
of data will be created by 2020, an increase of 300 times from 2005
2005
2020
It's estimated that
2.5 QUINTILLION BYTES
[2.3 TRILLION GIGABYTES]
of data are created each day
6 BILLION PEOPLE
have cell phones
WORLD POPULATION: 7 BILLION
Volume
SCALE OF DATA
Most companies in the U.S. have at least
100 TERABYTES
[100,000 GIGABYTES]
of data stored
The New York Stock Exchange captures
1 TB OF TRADE INFORMATION
during each trading session
Modern cars have close to
100 SENSORS
that monitor items such as fuel level and tire pressure
Velocity
ANALYSIS OF STREAMING DATA
By 2016, it is projected there will be
18.9 BILLION NETWORK CONNECTIONS
– almost 2.5 connections per person on earth
The Four V's of Big Data
From traffic patterns and music downloads to web history and medical records, data is recorded, stored, and analyzed to enable the technology and services that the world relies on every day. But what exactly is big data, and how can these massive amounts of data be used?
As a leader in the sector, IBM data scientists break big data into four dimensions: Volume, Velocity, Variety and Veracity
Depending on the industry and organization, big data encompasses information from multiple internal and external sources such as transactions, social media, enterprise content, sensors and mobile devices. Companies can leverage data to adapt their products and services to better meet customer needs, optimize operations and infrastructure, and find new sources of revenue.
By 2015
4.4 MILLION IT JOBS
will be created globally to support big data, with 1.9 million in the United States
As of 2011, the global size of data in healthcare was estimated to be
150 EXABYTES
[161 BILLION GIGABYTES]
30 BILLION PIECES OF CONTENT
are shared on Facebook every month
Variety
DIFFERENT FORMS OF DATA
By 2014, it's anticipated there will be
420 MILLION WEARABLE, WIRELESS HEALTH MONITORS
4 BILLION+ HOURS OF VIDEO
are watched on YouTube each month
400 MILLION TWEETS
are sent per day by about 200 million monthly active users
1 IN 3 BUSINESS LEADERS
don't trust the information they use to make decisions
27% OF RESPONDENTS
in one survey were unsure of how much of their data was inaccurate
Veracity
UNCERTAINTY OF DATA
Poor data quality costs the US economy around
$3.1 TRILLION A YEAR

大數據幫助看到全貌

例如局部正相關但整體負相關

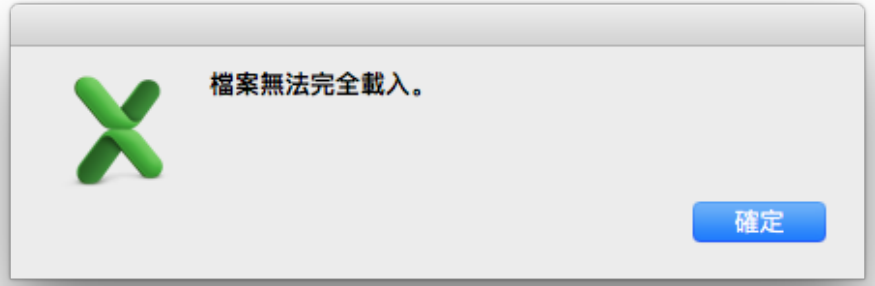
$r_{indiv.} = .12; r_{group(states)} = -.53$



多大叫做大？

試看看 17_Materials.zip

network.txt (5.0 GB): 284,884,514 行
network2.txt (167 MB): 10,000,000 行
network3.txt (32 MB): 2,000,000 行
network4.txt (129 KB): 10,000 行



Excel 最多只能讀 2^{20}=1048,576 行

序列計算 vs. 平行計算

調 調 調 調 調 調 調 調 調 調

0 10 20 30 40 50 60 70 80 90 100

(只有 1 條生產線)

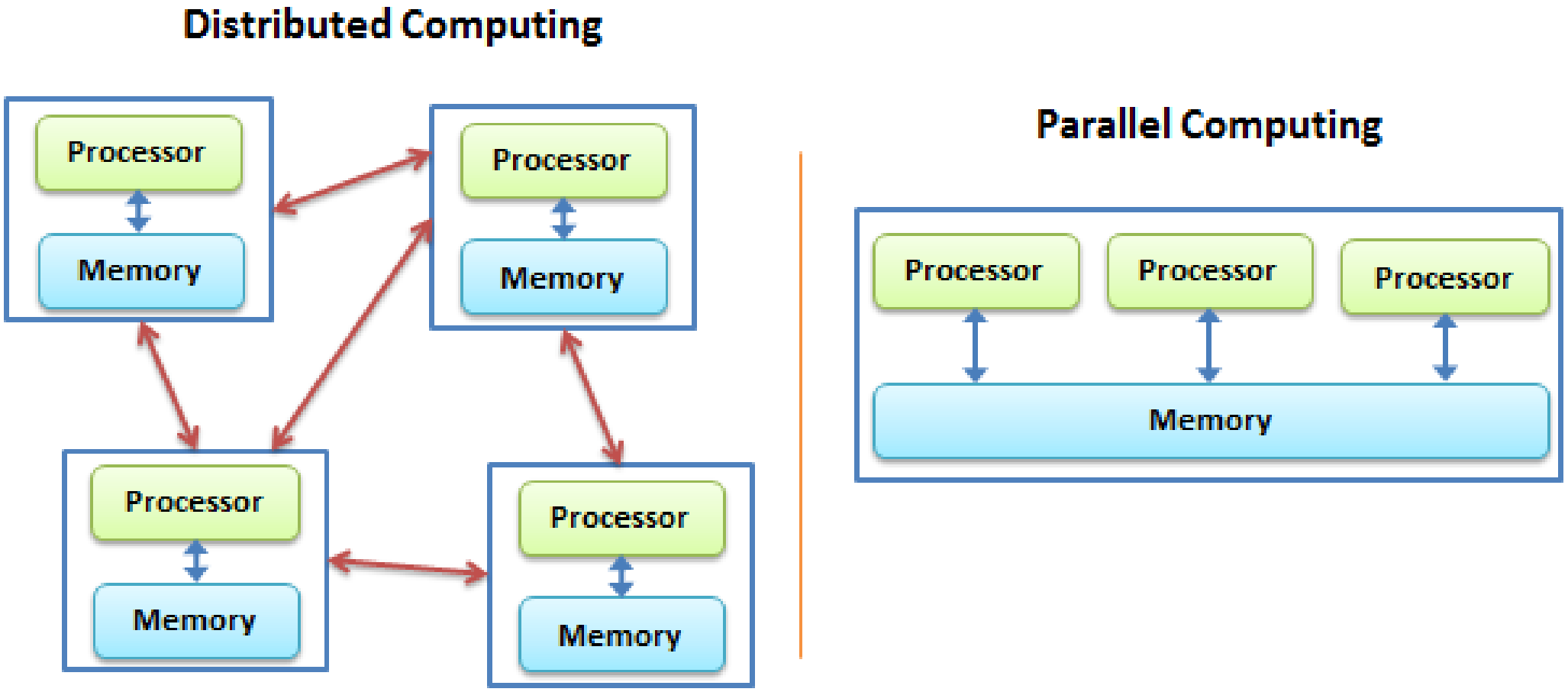
調 調 調 調 調 調 調 調 調 調 調

0 10 20 30 40 50 60 70 80 90 100

(共有 11 條生產線)

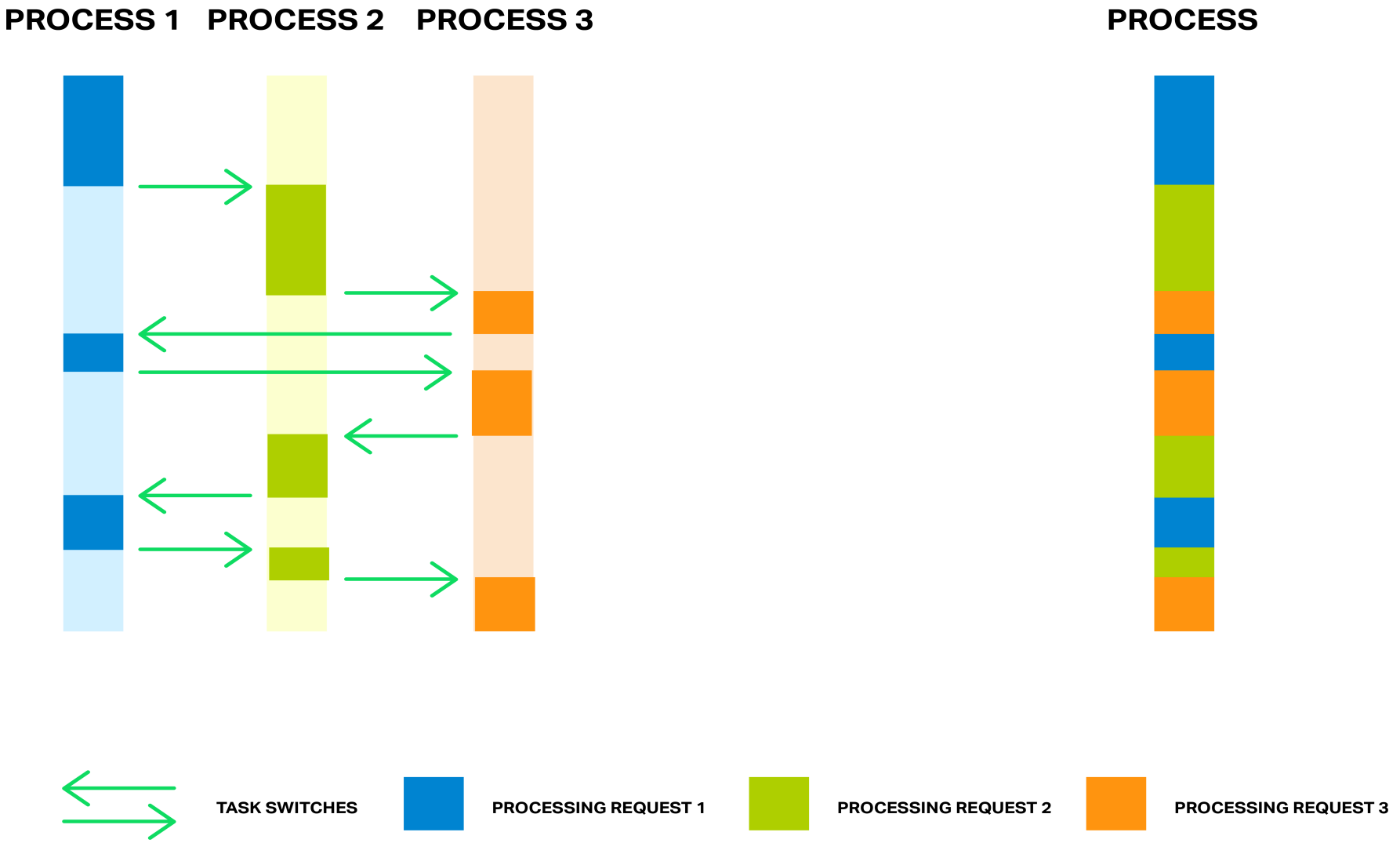
平行計算 vs. 分散式計算

前者用一台後者用多台電腦



平行 / 分散計算 vs. 異步執行

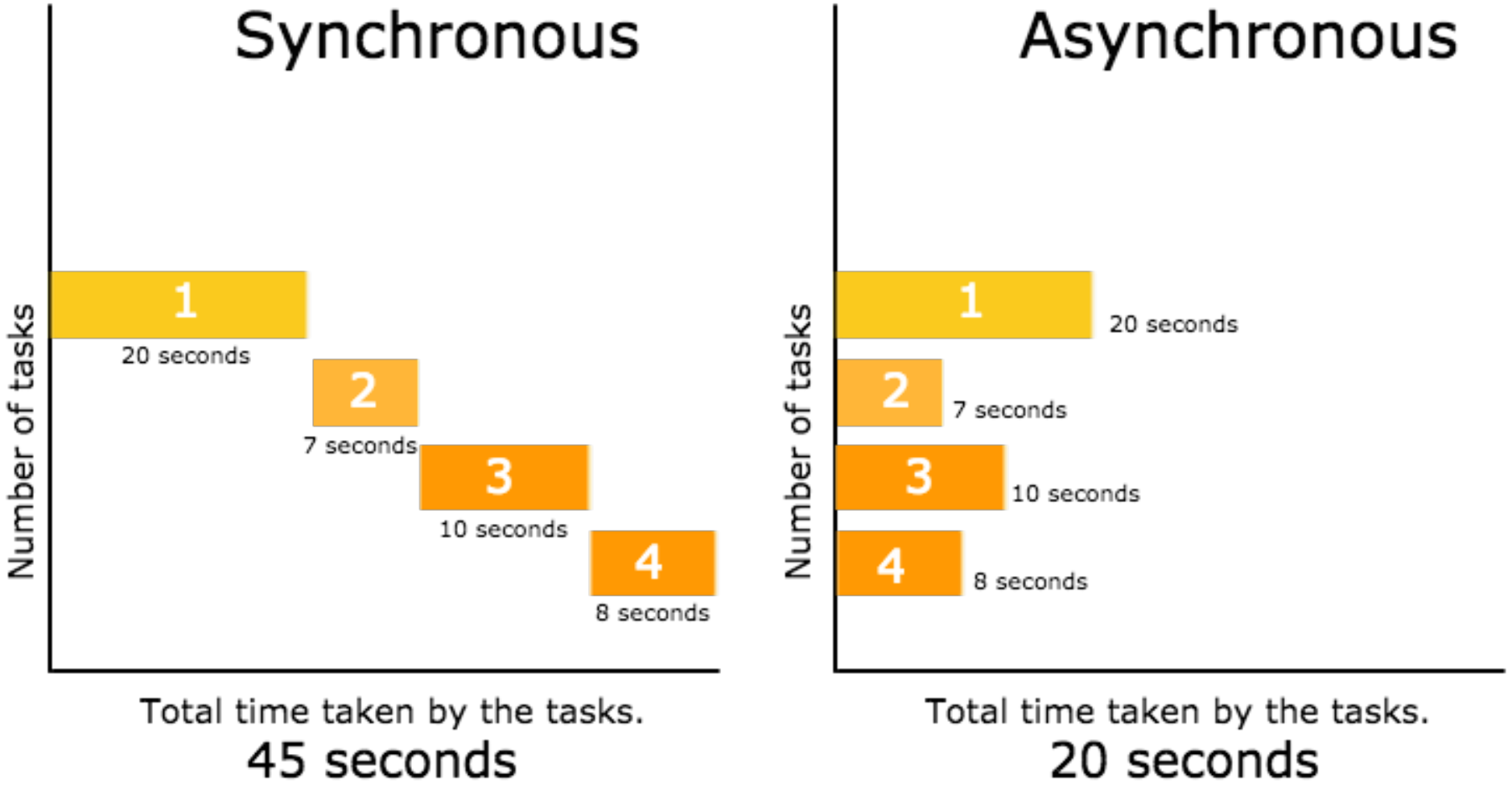
平行 / 分散計算仍會被 I/O 卡住，不如異步執行



異步執行
(Asynchronous Execution)

同步執行 vs. 異步執行 (1/2)

多且慢 I/O 的情況適合使用異步執行



同步執行 vs. 異步執行 (2/2)

Python 中 async I/O 可使用 asyncio 套件

check if done

switch

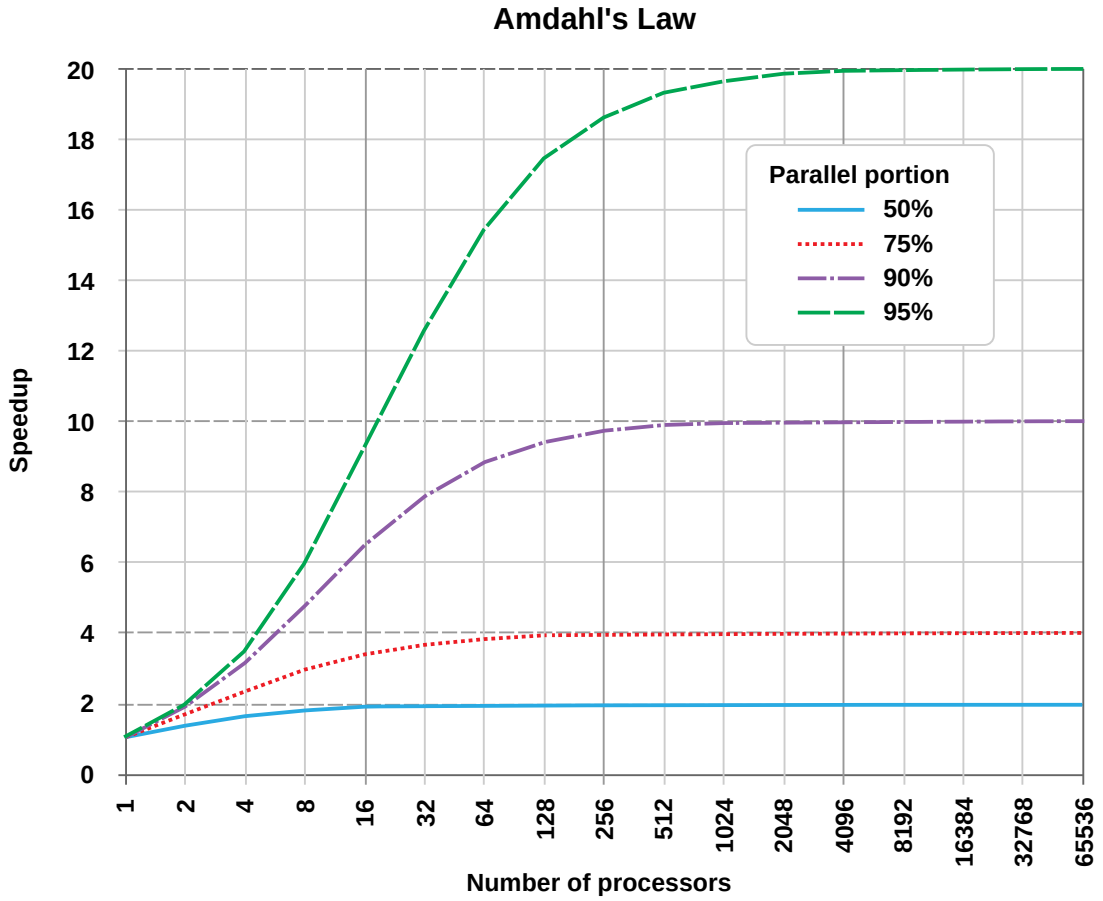
await

純 wait 就是沒有 task switch

平行計算
(Parallel Computing)

阿姆達爾定律

無法平行化的部分使得 N 個 workers 不會 N 倍快

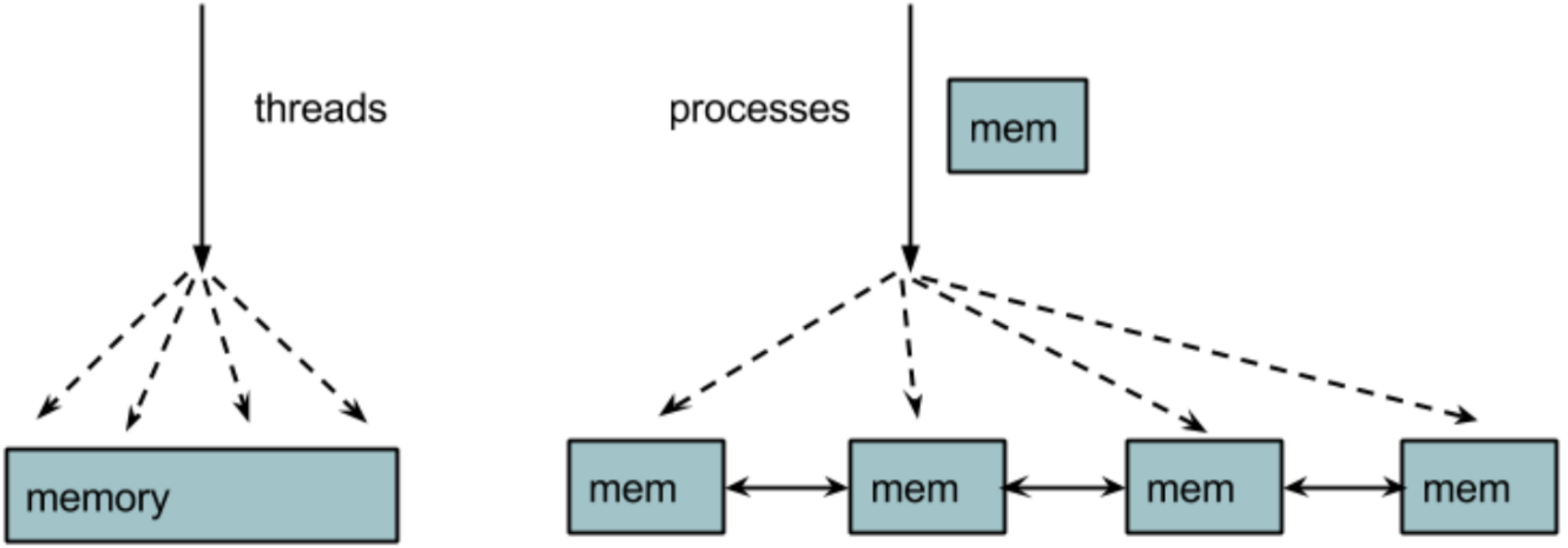


$$T(n) \geq S * T(1) + \frac{P * T(1)}{n}$$

$$T(\infty) \approx S * T(1)$$

Threads vs. Processes

Threads 只有一個 process 並共享記憶體



跑 processes 有更多的創建 / 通訊 overheads
但能更有效率使用 multi-core CPUs

序列計算 vs. 平行計算

調 調 調 調 調 調 調 調 調 調

0 10 20 30 40 50 60 70 80 90 100

(只有 1 條生產線)

調 調 調 調 調 調 調 調 調 調 調

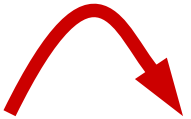
0 10 20 30 40 50 60 70 80 90 100

(共有 11 條生產線)

第二週介紹的內建函數 map

```
import math
def adjust_score(old):
 new=math.sqrt(old)*10
 return new

print(list(map(adjust_score,range(0,101,10))))
```

套上去



multiprocessing 的 map 才是真的平行計算
以後講 Big Data 的時候會再看到類似觀念

Python: concurrent.futures

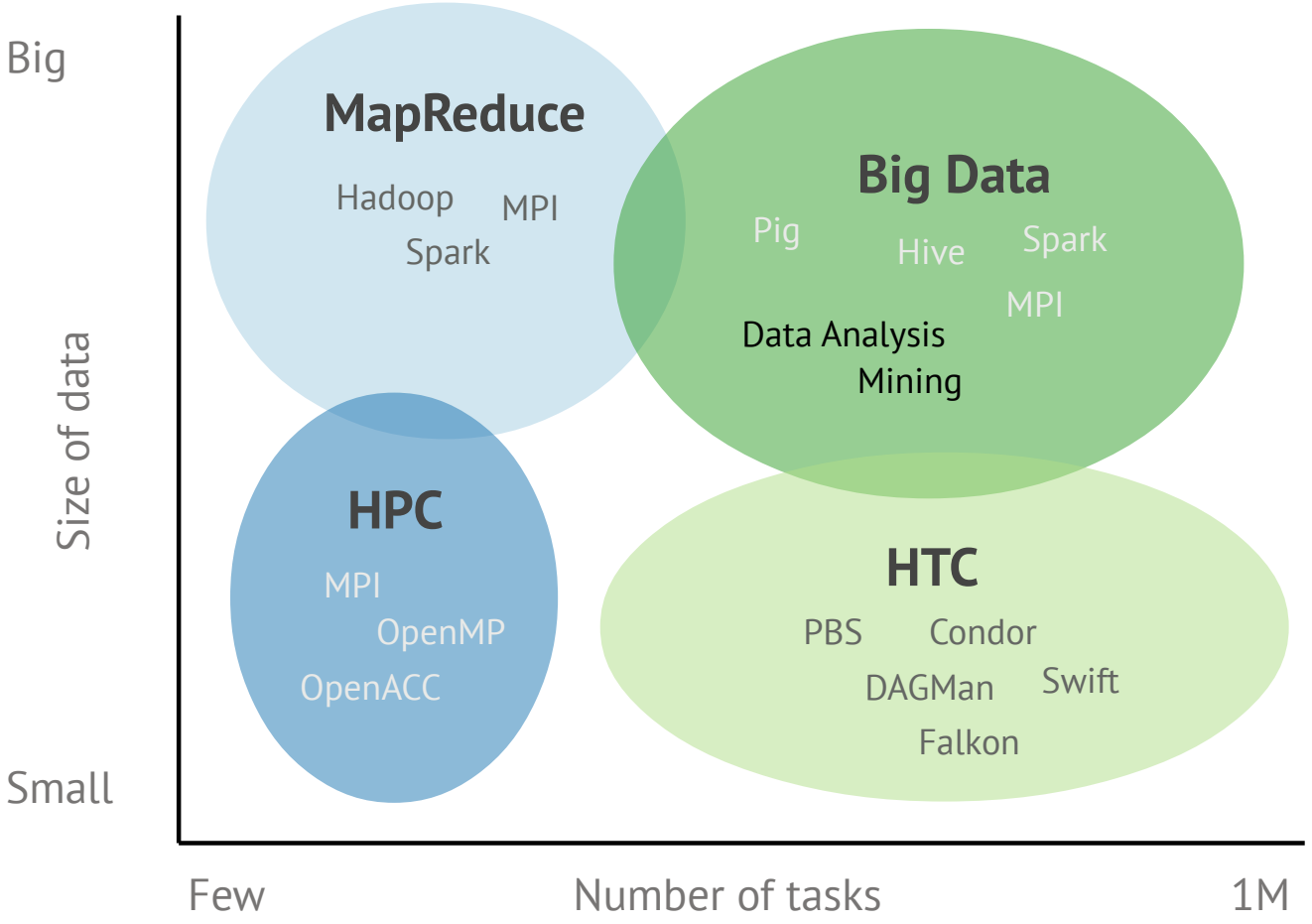
在 Python 3.2 版前是使用 threading 與 multiprocessing

```
import math, concurrent.futures as cf

def adjust_score(old):
 new=math.sqrt(old)*10
 return new

with cf.ThreadPoolExecutor(max_workers=2) as
pool:
 new=pool.map(adjust_score,range(100))

list(new)
```

分散式計算架構 / 環境
(Distributed Computing)

不同領域的資料 / 計算特性

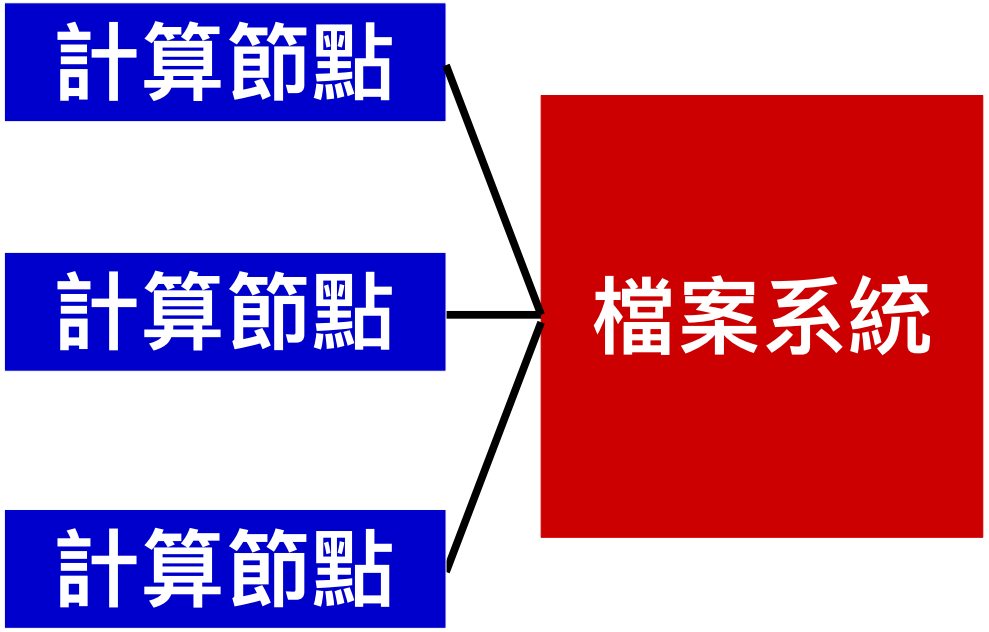
傳統科學資料少計算多；大數據分析資料多計算多



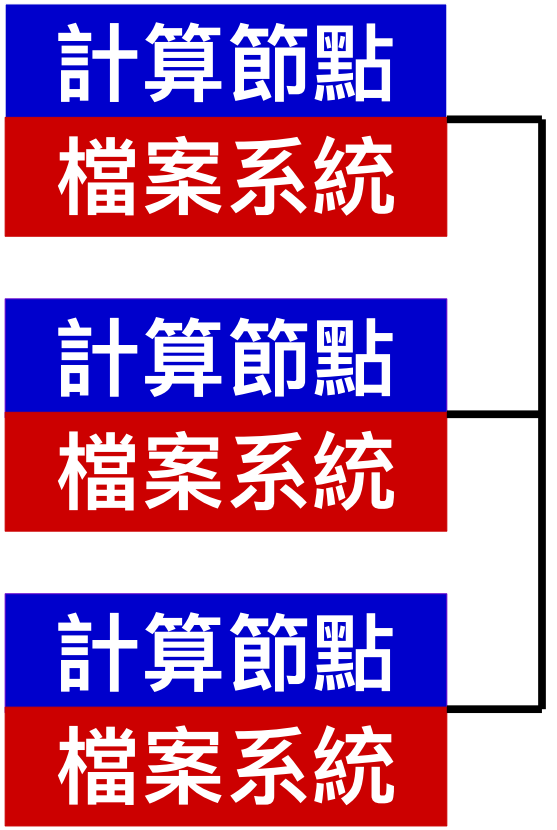
電腦叢級架構

大數據分析：資料在哪裡，就在哪裡分析

傳統的科學計算叢集
（資料集中式儲存）



新式的大資料叢集
（資料分散式儲存）



資料科學家真實案例

某臺大 /MIT 畢業生去 Facebook 應徵時

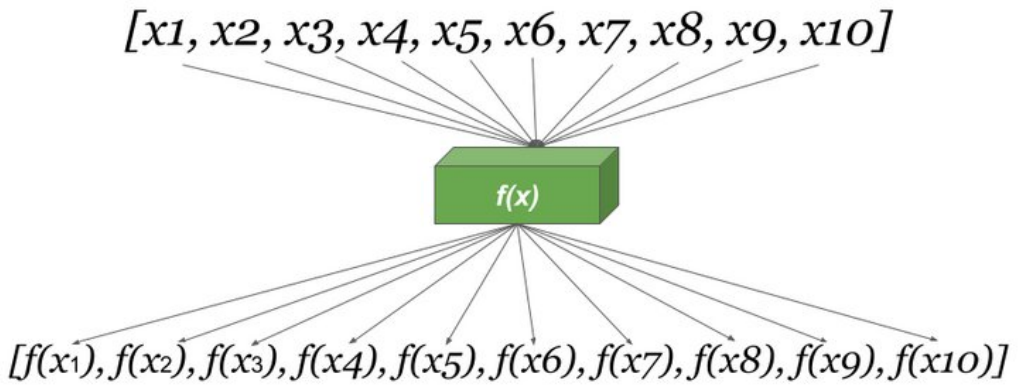
請解釋何謂 MapReduce?

....（默然）

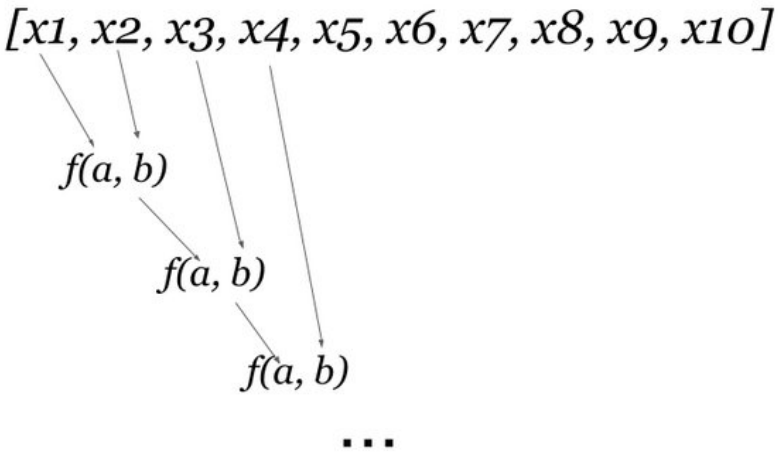
Map+Reduce

就是分而治之之後再彙整所有資訊

Map

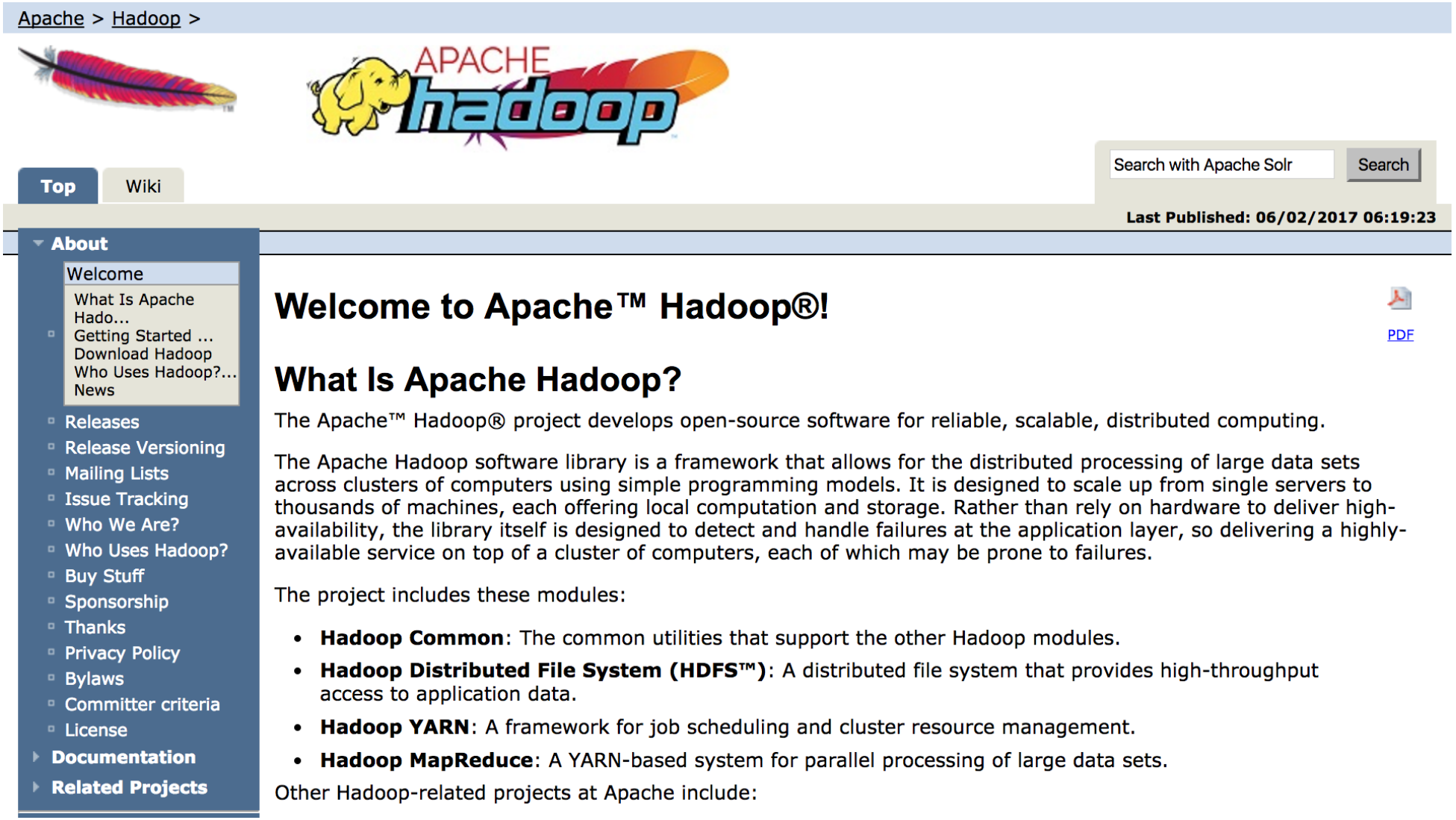


Reduce

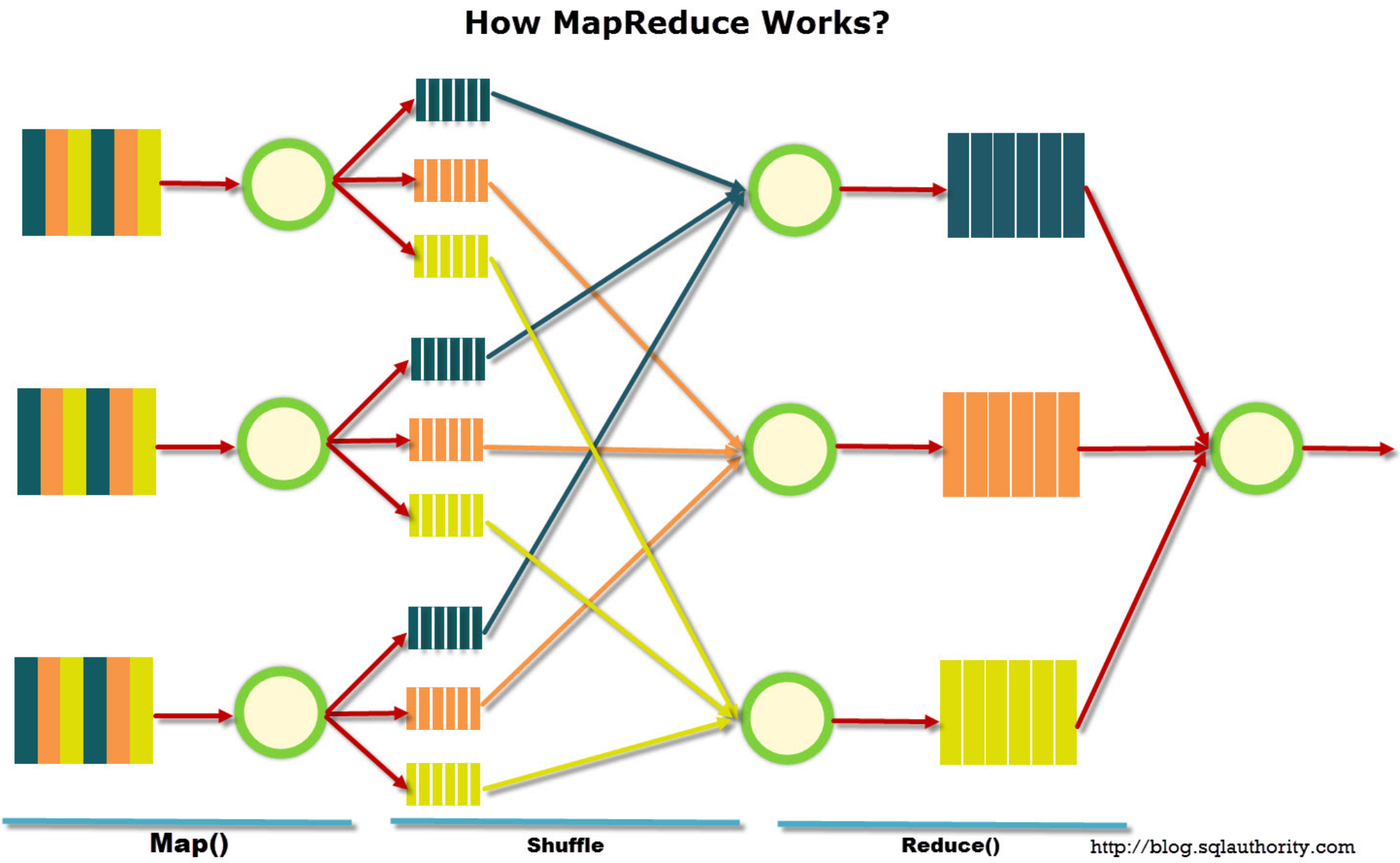


Hadoop: 實做 MapReduce 的平台

Hadoop 實做了很多 Google 發展的技術

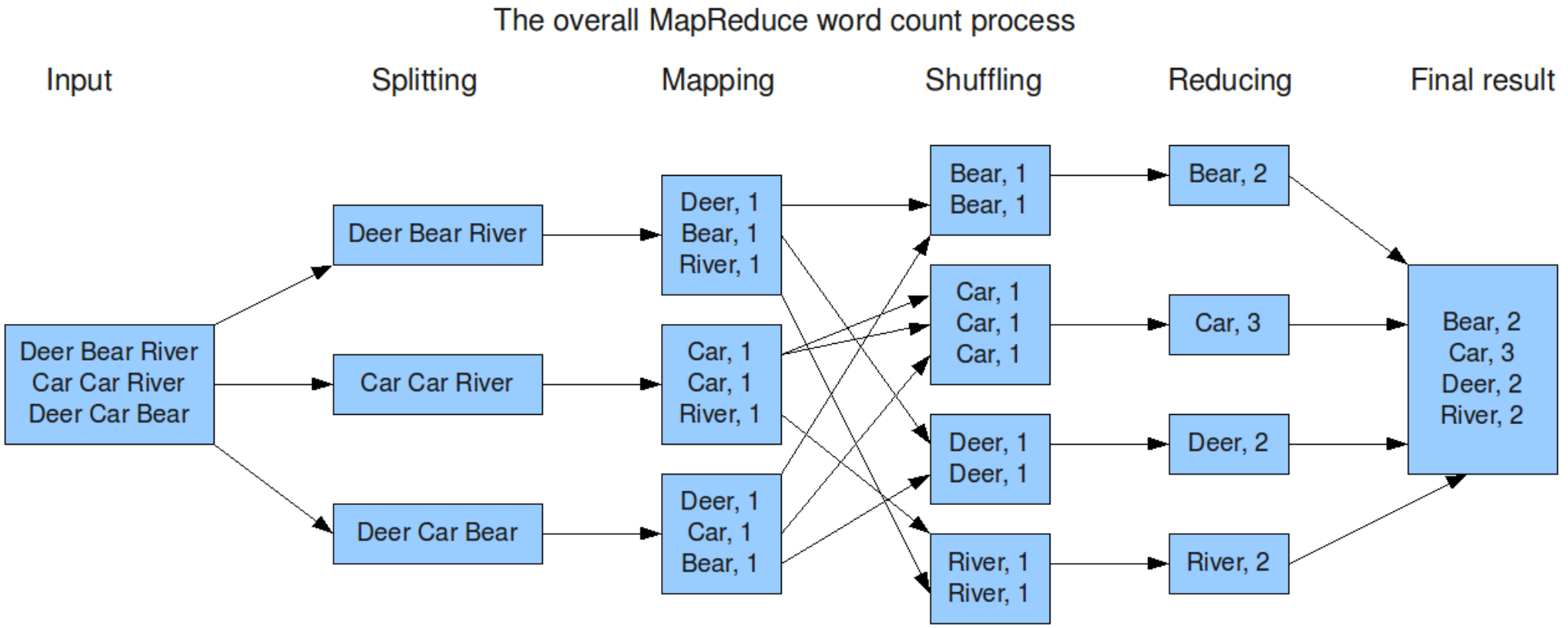


MapReduce原理

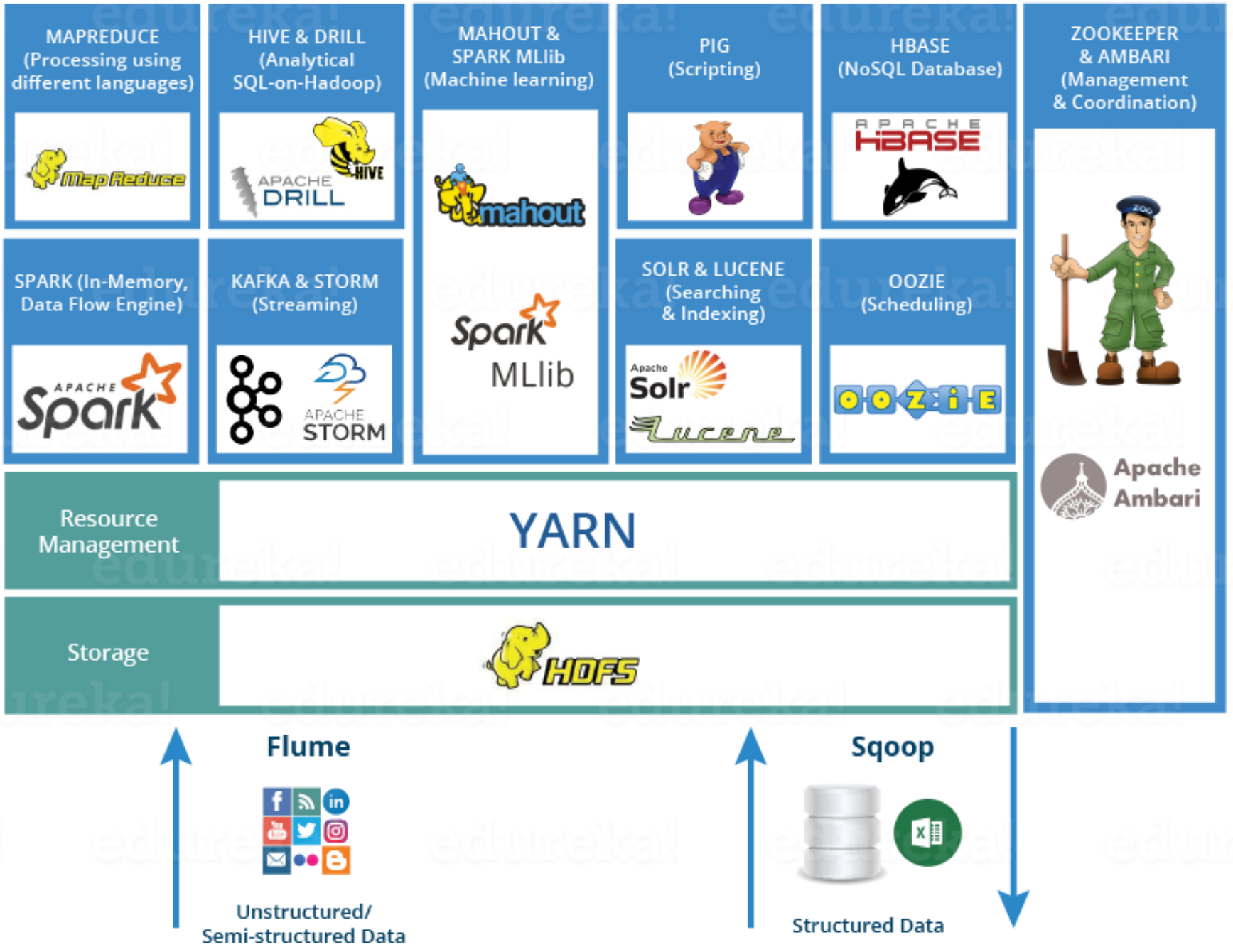


MapReduce範例

光是計算字頻這麼簡單的事都很麻煩



Hadoop 周邊資源 (ecosystem)



異軍突起的 Spark (2009-)

號稱在許多運算上比 Hadoop 快 100 倍

Lightning-fast cluster computing

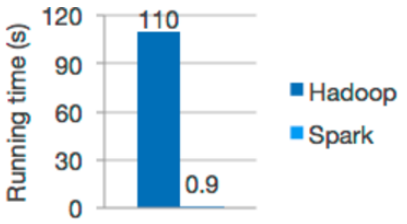
Download | Libraries | Documentation | Examples | Community | Developers | Apache Software Foundation

Apache Spark™ is a fast and general engine for large-scale data processing.

Speed

Run programs up to 100x faster than Hadoop MapReduce in memory, or 10x faster on disk.

Apache Spark has an advanced DAG execution engine that supports acyclic data flow and in-memory computing.



Logistic regression in Hadoop and Spark

Ease of Use

Write applications quickly in Java, Scala, Python, R.

Spark offers over 80 high-level operators that make it easy to build parallel apps. And you can use it *interactively* from the Scala, Python and R shells.

```
text_file = spark.textFile("hdfs://...")

text_file.flatMap(lambda line: line.split())
    .map(lambda word: (word, 1))
    .reduceByKey(lambda a, b: a+b)
```

Word count in Spark's Python API

Latest News

Spark 2.1.1 released (May 02, 2017)

Spark Summit (June 5-7th, 2017, San Francisco) agenda posted (Mar 31, 2017)

Spark Summit East (Feb 7-9th, 2017, Boston) agenda posted (Jan 04, 2017)

Spark 2.1.0 released (Dec 28, 2016)

Archive

Download Spark

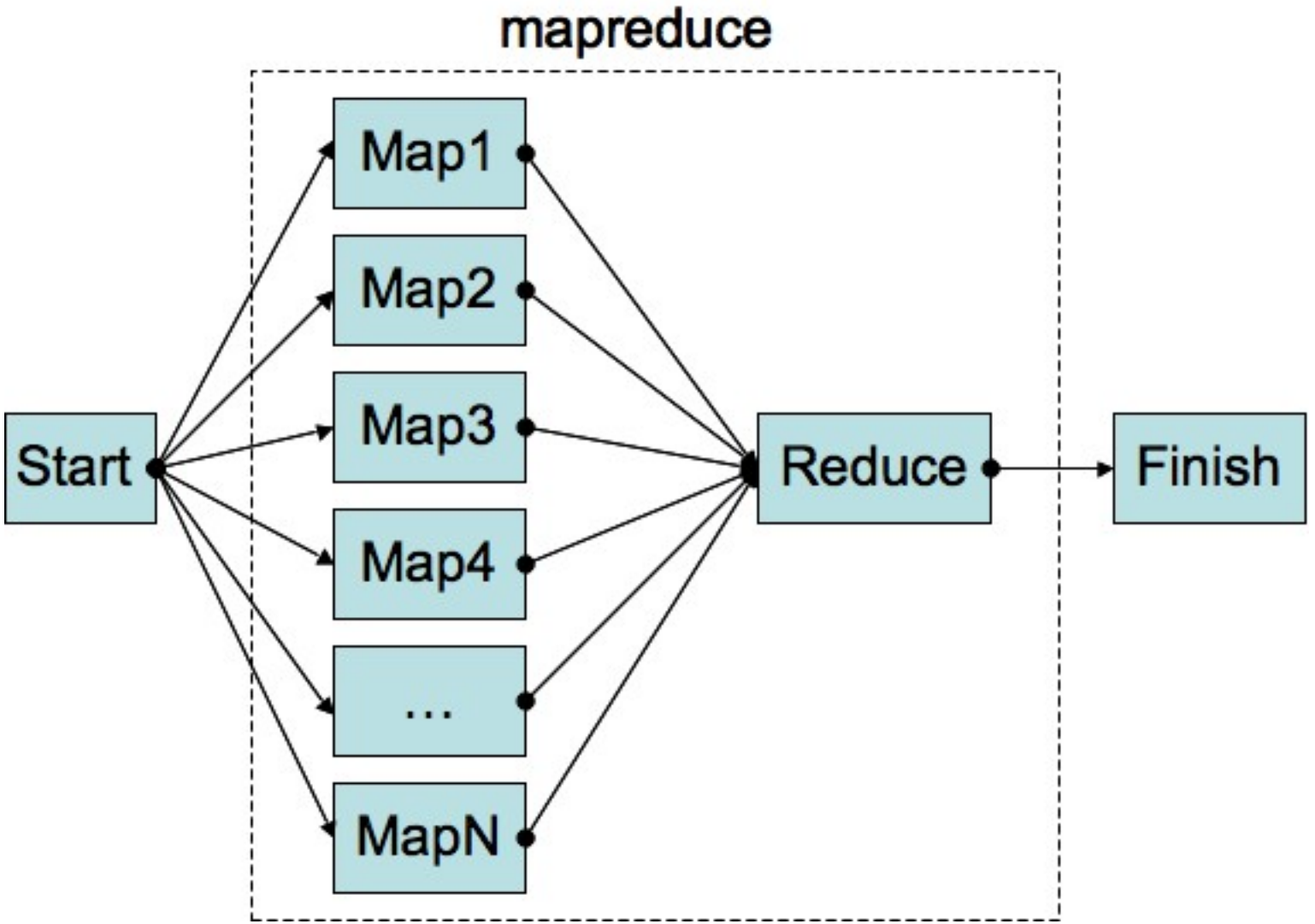
Built-in Libraries:

SQL and DataFrames
Spark Streaming
MLlib (machine learning)
GraphX (graph)

Third-Party Projects

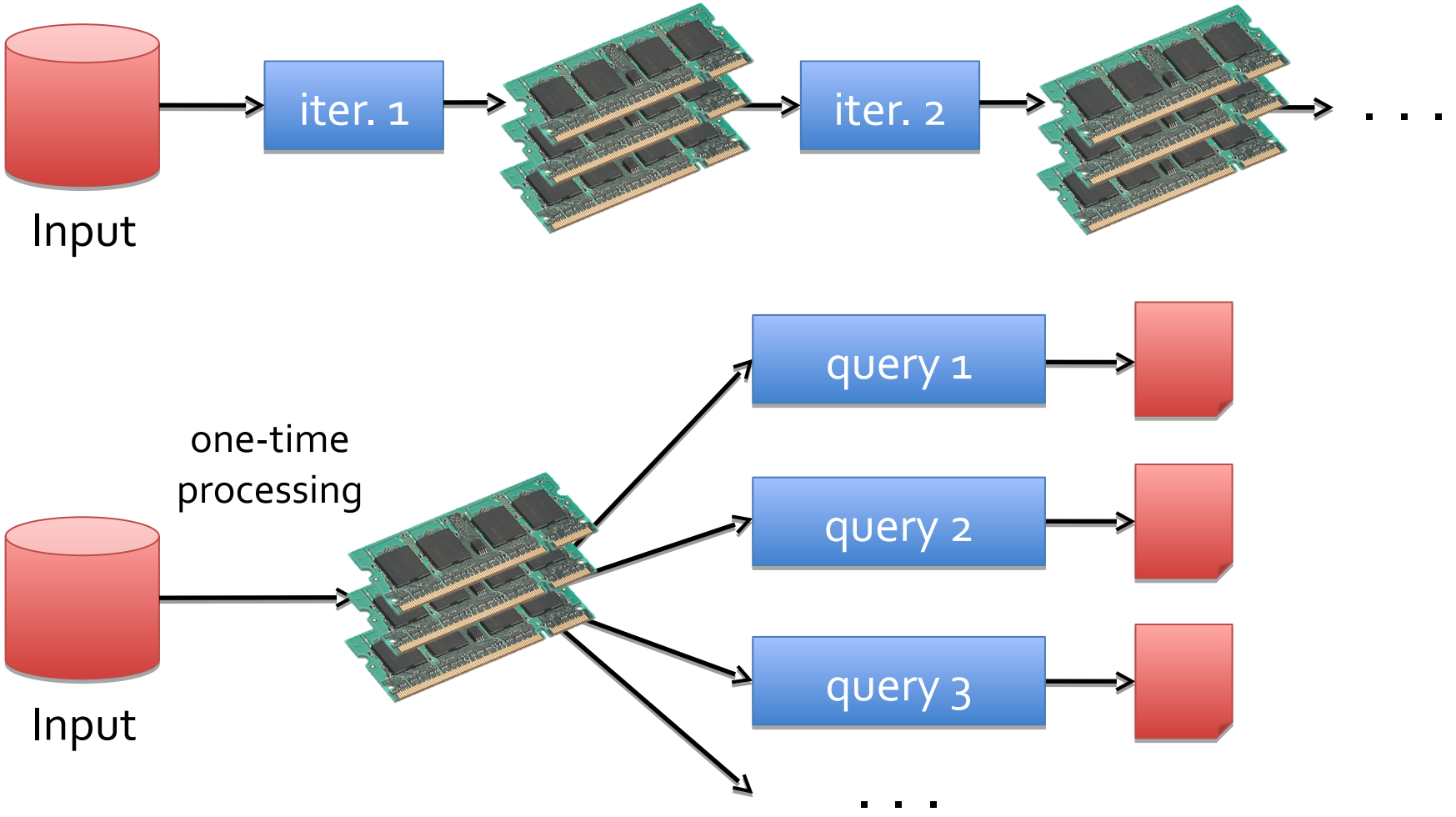
Spark's MapReduce

計算架構和 Hadoop 大致上一樣
(其實所有平行 / 分散式計算架構就是這樣)



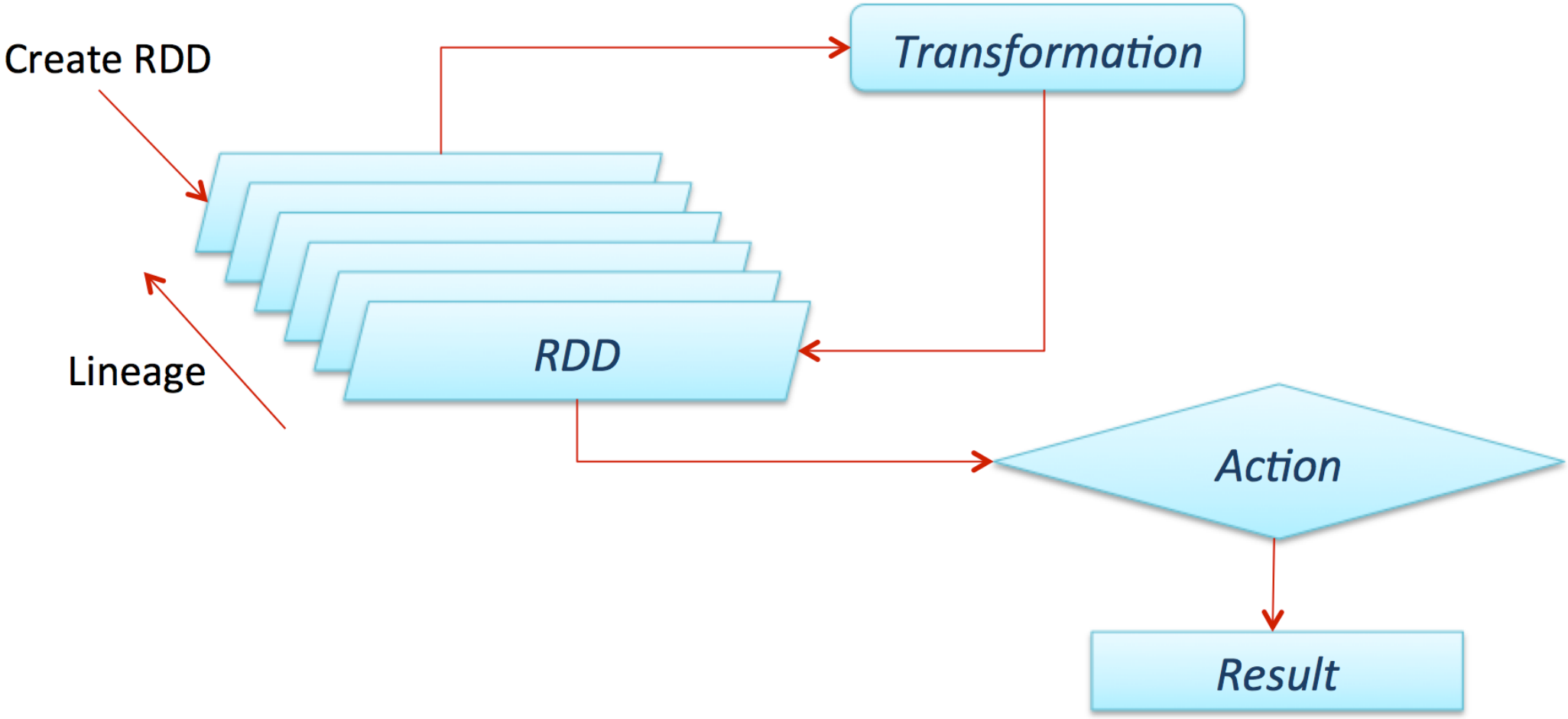
Spark vs. Hadoop

Spark 把資料都丟到記憶體 (RDDs) 減少磁碟操作來加速



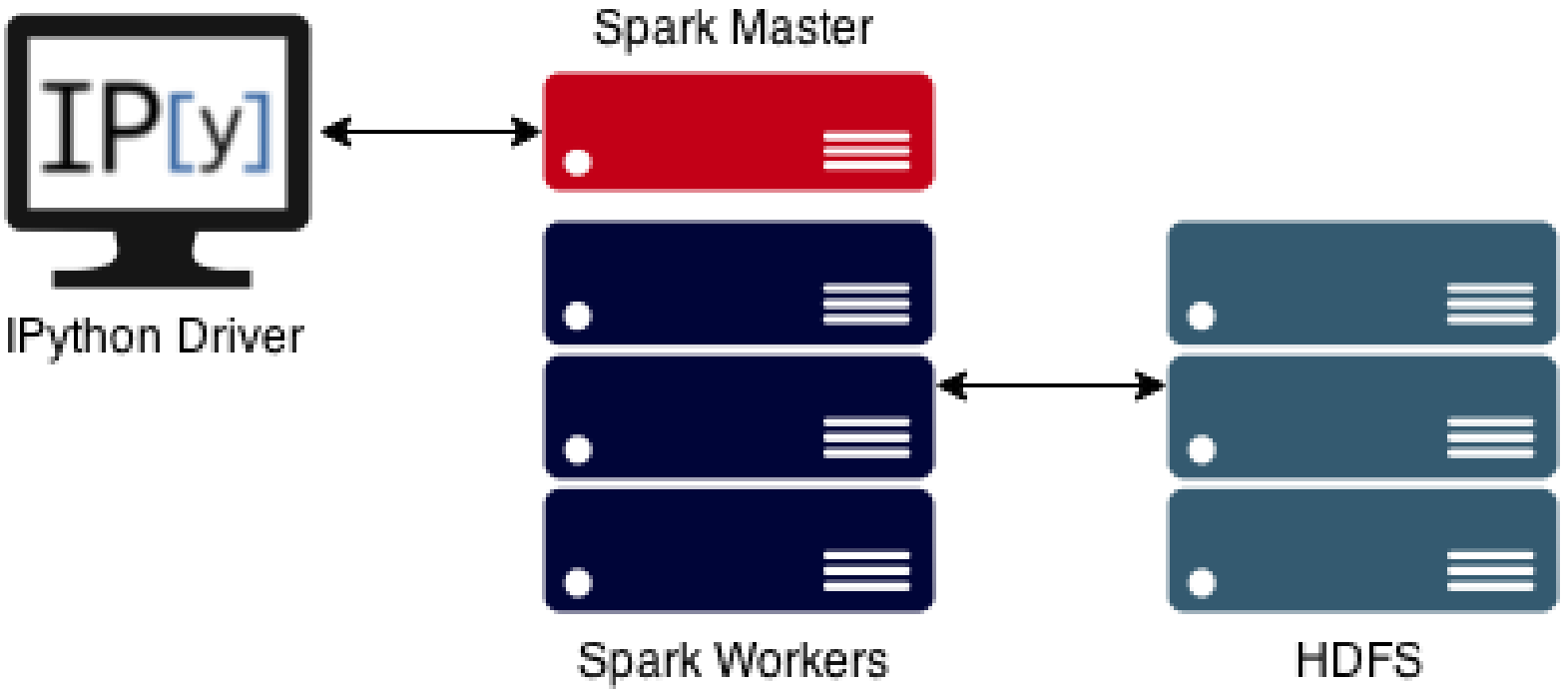
Resilient Distributed Dataset

RDD 是一個資料被 Spark 分散到各記憶體中的變數



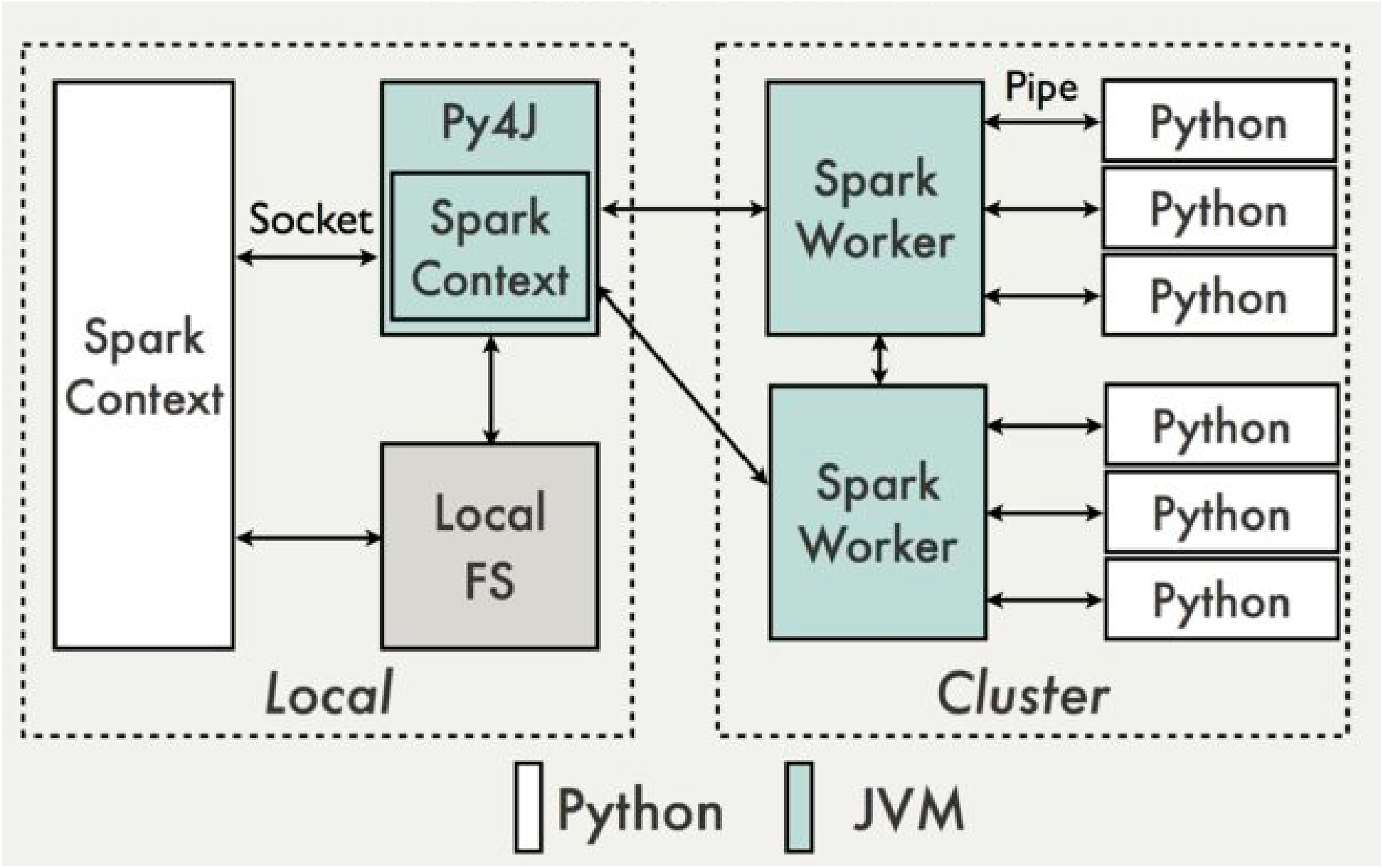
PySpark+Jupyter Notebook

對 Python 使用者來說最好的大數據分析環境



環境成熟，使用人數多

PySpark 硬體架構



PySpark 的學習

照著官方文件的 Examples跑跑看就會了 (!?)

Lightning-fast cluster computing

Download | Libraries | Documentation | Examples | Community | Developers | Apache Software Foundation

Apache Spark Examples

These examples give a quick overview of the Spark API. Spark is built on the concept of *distributed datasets*, which contain arbitrary Java or Python objects. You create a dataset from external data, then apply parallel operations to it. The building block of the Spark API is its RDD API. In the RDD API, there are two types of operations: *transformations*, which define a new dataset based on previous ones, and *actions*, which kick off a job to execute on a cluster. On top of Spark's RDD API, high level APIs are provided, e.g. DataFrame API and Machine Learning API. These high level APIs provide a concise way to conduct certain data operations. In this page, we will show examples using RDD API as well as examples using high level APIs.

RDD API Examples

Word Count

In this example, we use a few transformations to build a dataset of (String, Int) pairs called `counts` and then save it to a file.

Python | Scala | Java

```
text_file = sc.textFile("hdfs://...")
counts = text_file.flatMap(lambda line: line.split(" ")) \
             .map(lambda word: (word, 1)) \
             .reduceByKey(lambda a, b: a + b)
counts.saveAsTextFile("hdfs://...")
```

Latest News

Spark 2.1.1 released (May 02, 2017)

Spark Summit (June 5-7th, 2017, San Francisco) agenda posted (Mar 31, 2017)

Spark Summit East (Feb 7-9th, 2017, Boston) agenda posted (Jan 04, 2017)

Spark 2.1.0 released (Dec 28, 2016)

Archive

Download Spark

Built-in Libraries:

SQL and DataFrames
Spark Streaming
MLlib (machine learning)
GraphX (graph)

Third-Party Projects

Transformation vs. Action

Transformation:Action ~ Map:Reduce

Transformation	Action
map, filter, flatMap, mapPartitions, mapPartitionsWithIndex, sample, union, distinct, intersection, join groupByKey, pipe, ReduceByKey, cogroup, aggregateByKey, sortByKey, cartesian, coalesce, repartition, repartitionAndSortWithinPartitions	reduce, collect, count, first, take, takeSample, takeOrdered, saveAsTextFile, saveAsSequenceFile, saveAsObjectFile, countByKey, foreach, stats,sum,max,min, stdev,mean

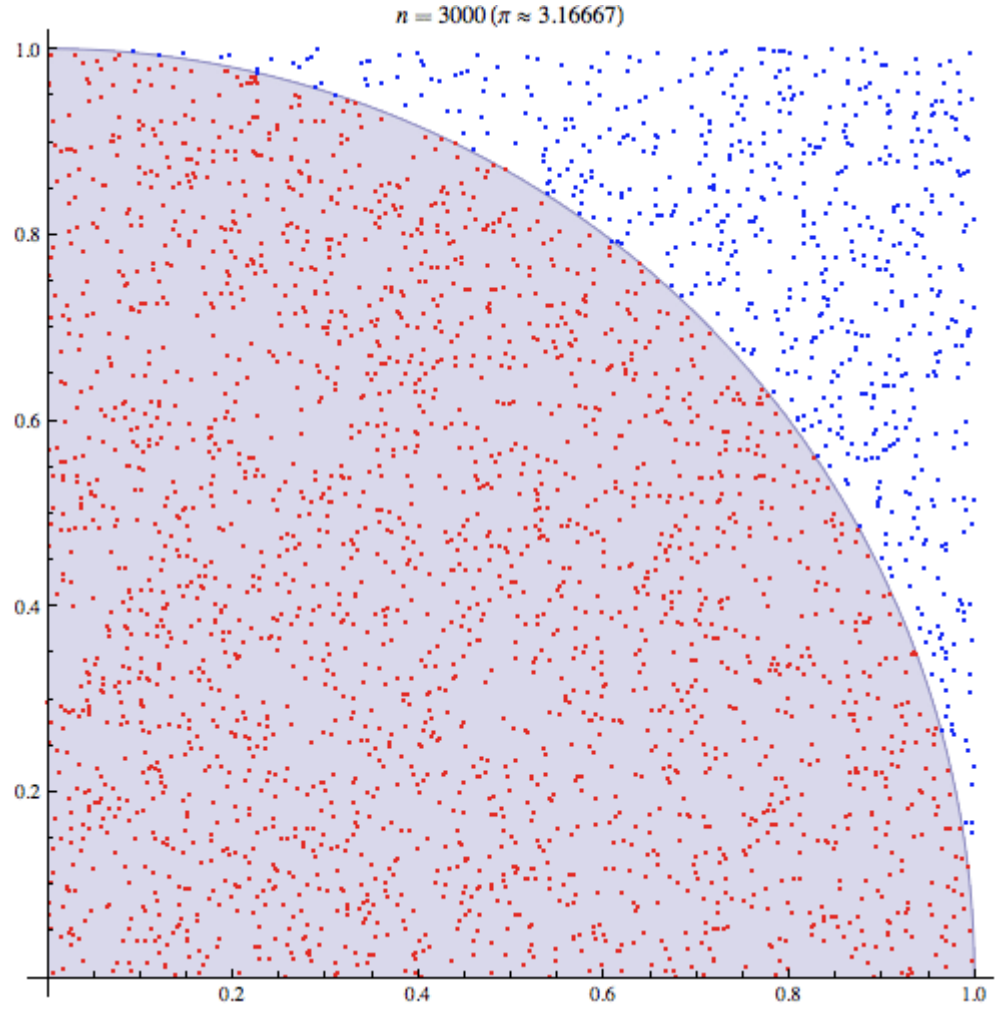
PySpark 的基礎使用

常配合 lambda 來定義一個用完就丟的匿名函數

```
import pyspark
sc=pyspark.SparkContext()

rdd=sc.parallelize(range(10))
rdd.filter(lambda x:x>5).collect()
print(rdd.sum(),rdd.stats())
rdd.reduce(lambda x,y:x+y)

import math
rdd2=rdd.map(lambda x:math.sqrt(x)*10)
rdd2.collect() #collected to local memory
```

Map 範例：蒙地卡羅法算 π

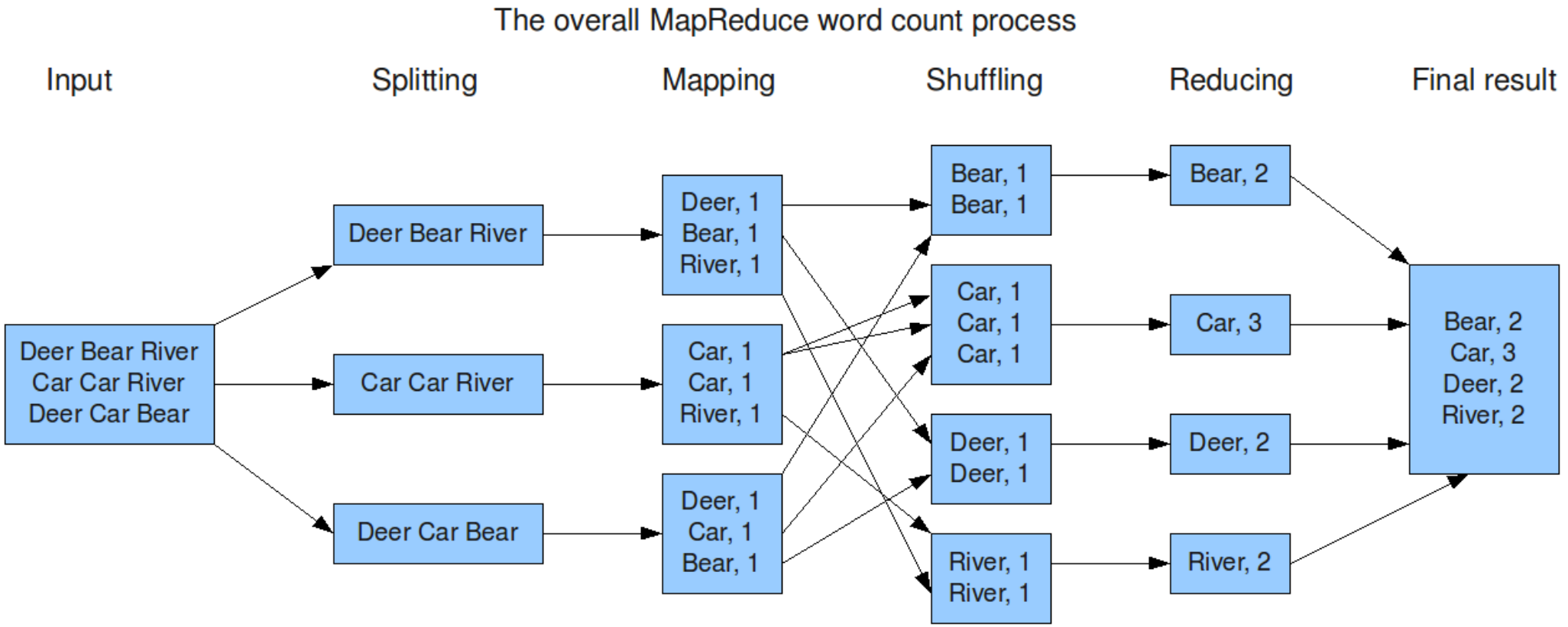
正方形面積與圓缺角面積之比為 1:π/4



根據幾何機率：落在紅色區的點約是隨機點總數 *π/4

MapReduce範例：Word Count

除了 shuffling 外每個階段都有對到一個 Spark 指令



RDD2 RDD3 RDD4

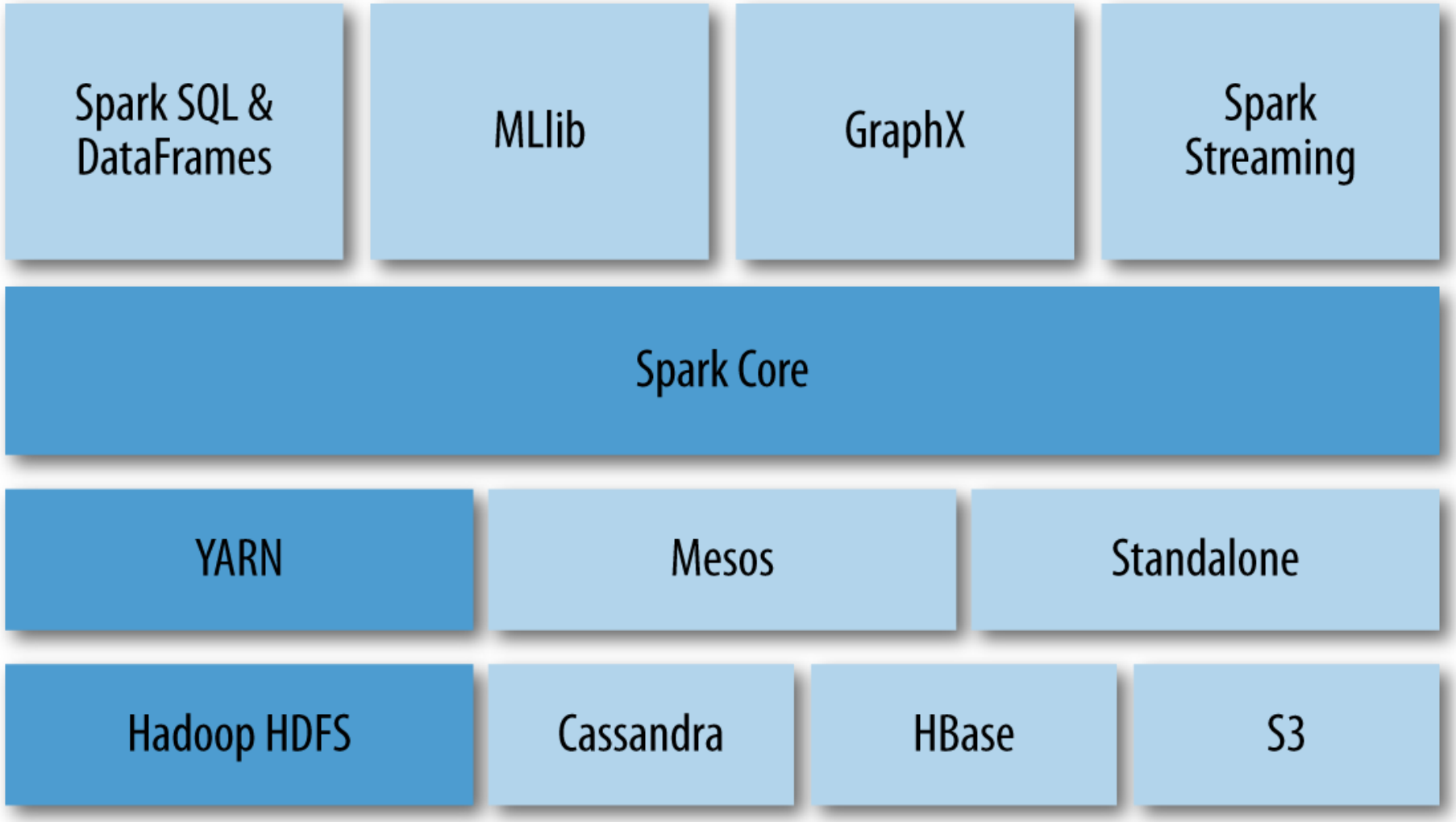
PySpark 範例：Word Count

先下載 CEIBA 上的 alice_in_wonderland.txt

```
#rdd1=sc.textFile('hdfs://alice.txt') # HDFS
rdd1=sc.textFile('alice.txt') # local
#rdd1.count() # lines
rdd2=rdd1.flatMap(lambda line: line.split(' '))
#rdd2.count() # words
rdd3=rdd2.map(lambda word : [word, 1])
rdd4=rdd3.reduceByKey(lambda x,y:x+y)
rdd4.saveAsTextFile('alice_results') # local
```

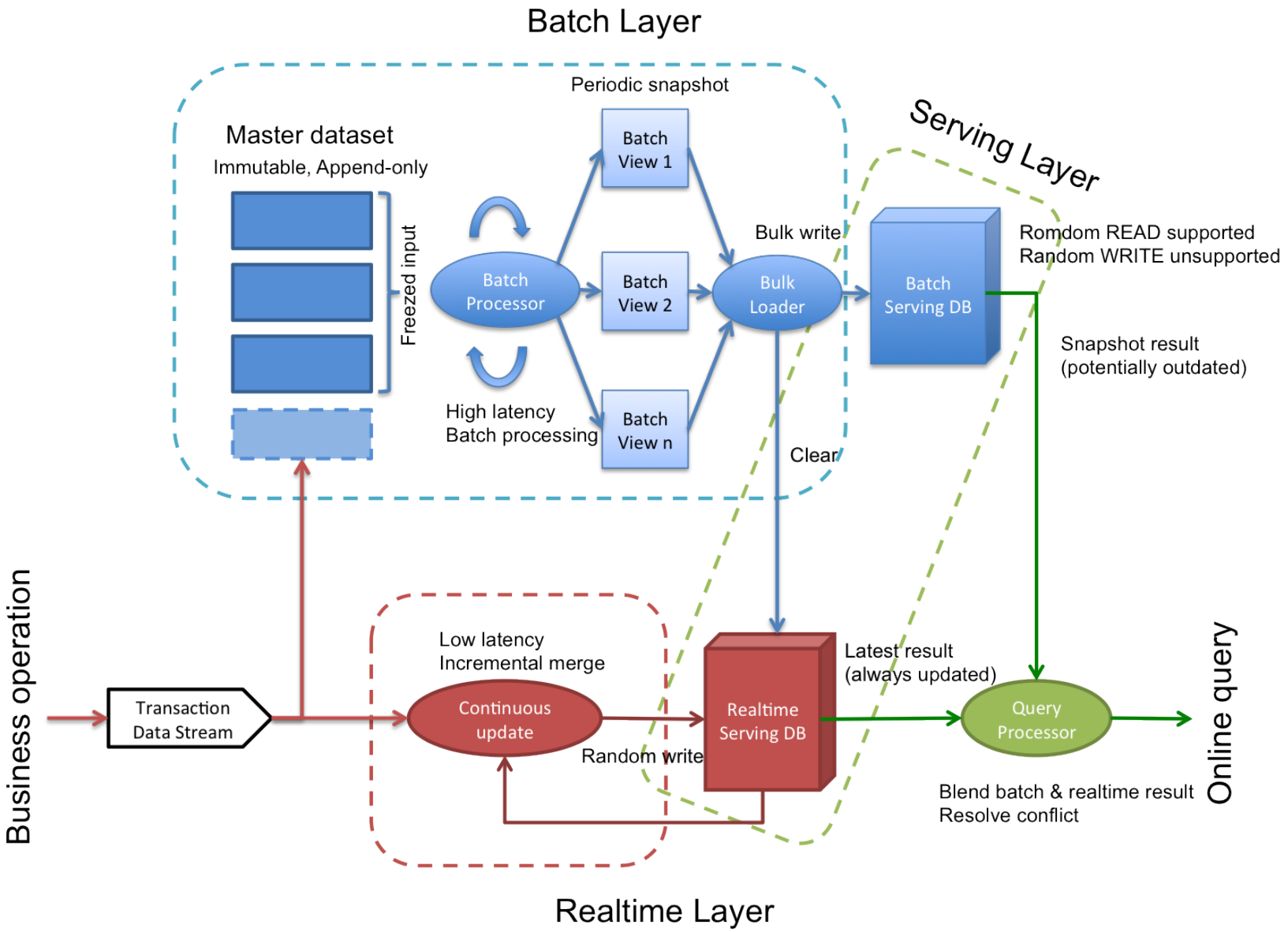
Spark Ecosystem

RDD 太低階，通常使用 Spark DataFrames
再配合 MLlib 跑分散式機器學習



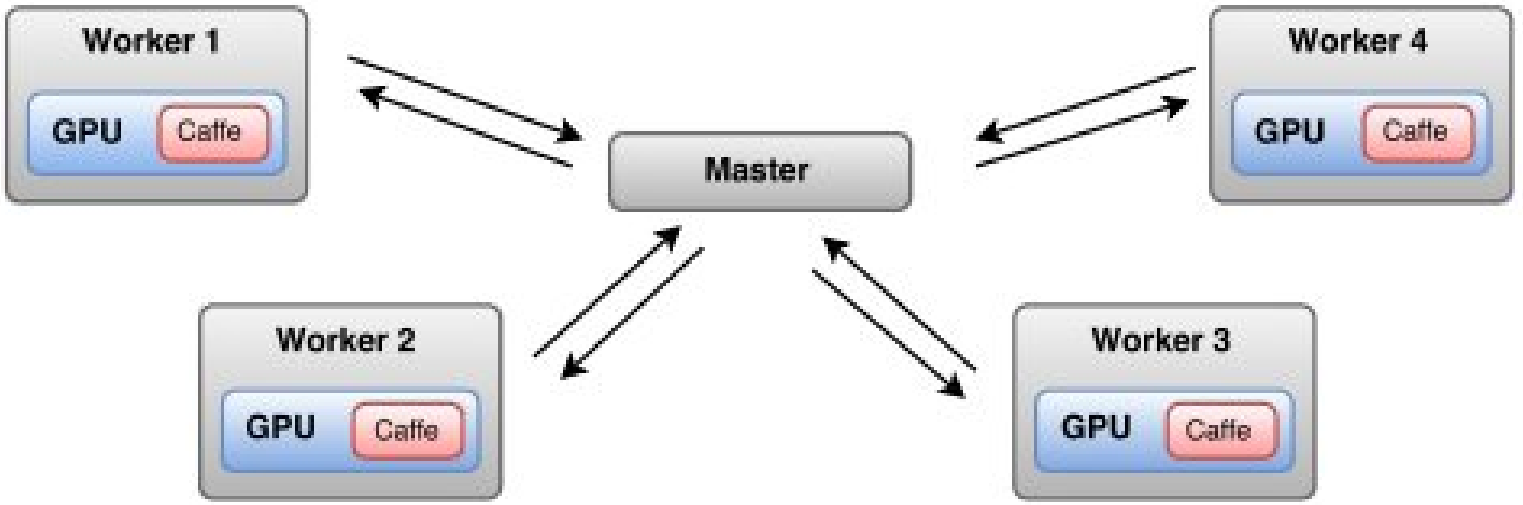
混合架構 (λ architecture)

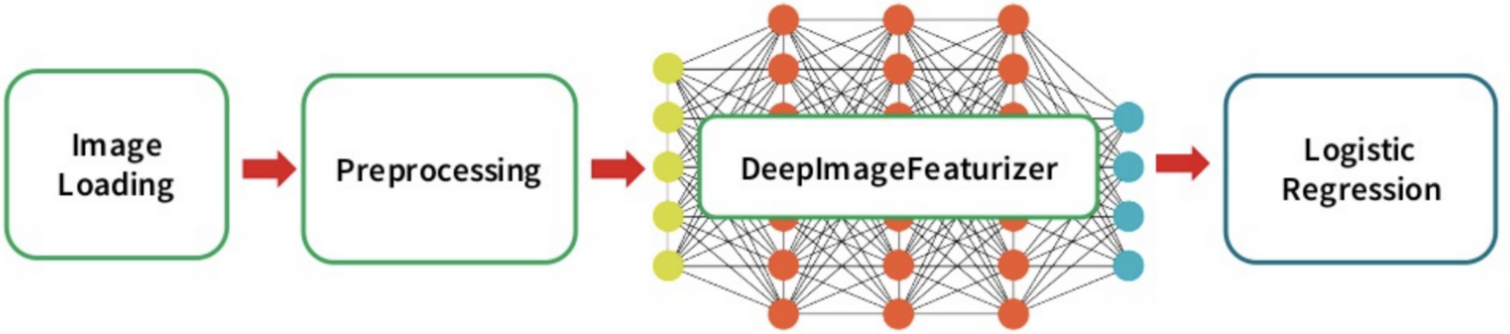
Real-time processing 是商業應用上的核心主題



莫忘 Deep Learning

Spark 上開始嘗試做 Deep Learning



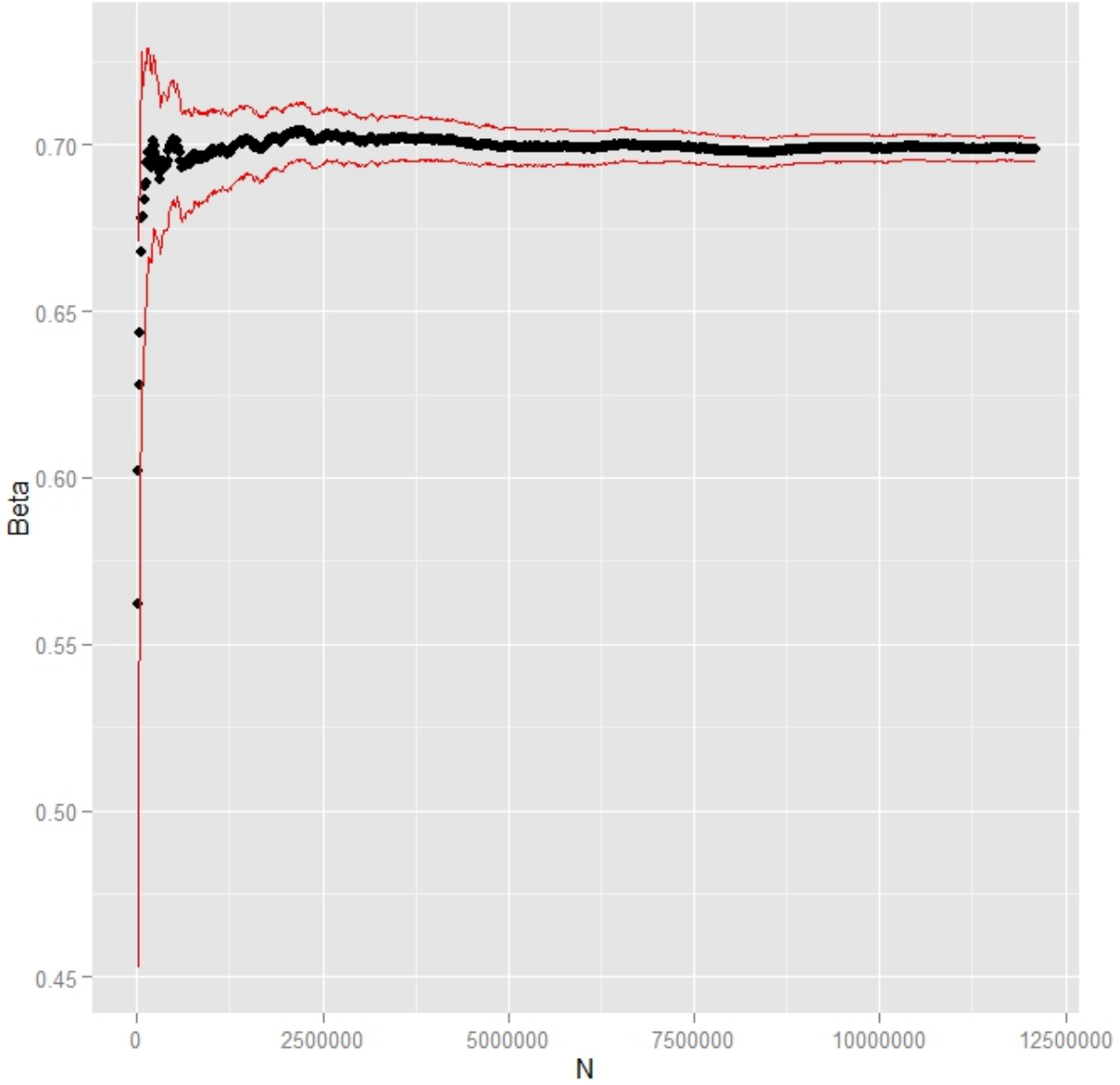


MLlib Pipeline

Big Data 的迷思

更多數據≠更好的結果

加入更多數據不見得會增加 signal to noise ratio



大數據時代可重相關不重因果!?

「知道 what(相關) 就夠了，沒必要知道 why(因果)」

Google 提示相關字

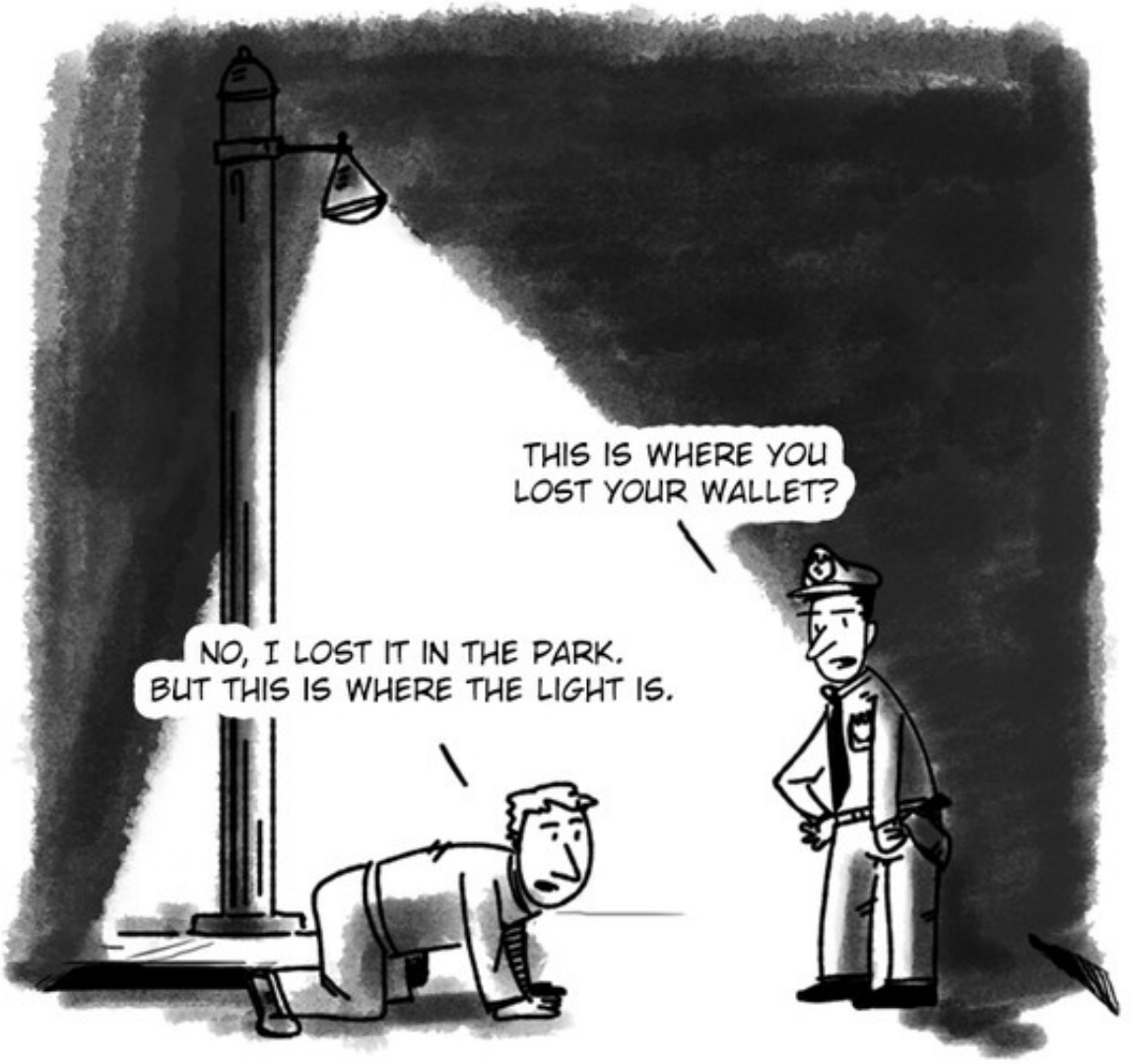
信用與行為的相關

懷孕與購物型態相關

啤酒與尿布常被合購

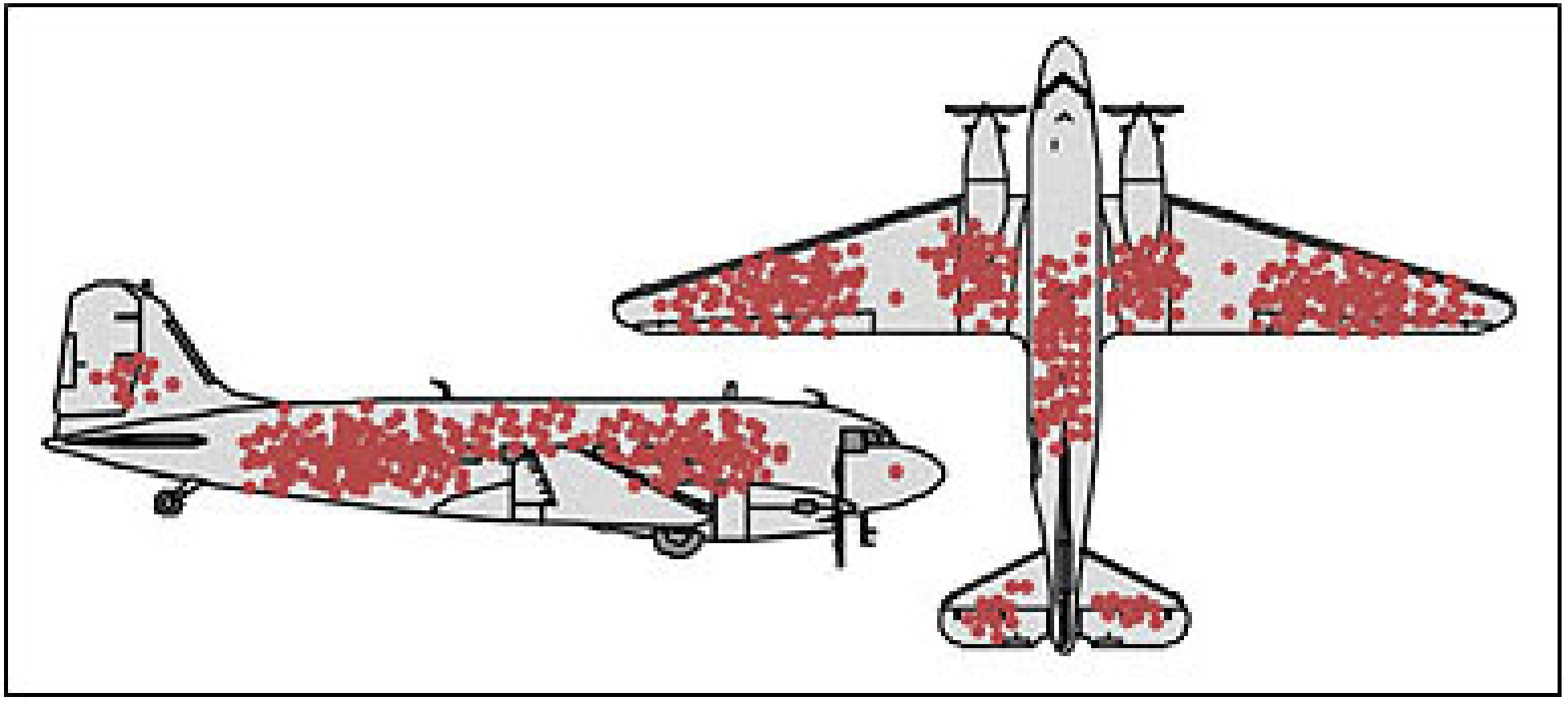
有搜集到關鍵資料嗎？(1/2)

因果關係中的「因」 API 有提供嗎？



有搜集到關鍵資料嗎？(2/2)

應強化機頭，機尾，機翼，或機身來避免失事？



即便資料搜集完整，答案不見得在資料中

That's all for the semester!

Game Over