巨量資料的趨勢、挑戰與因應對策
Big Data : Trends, Challenges and Solutions

Who AM I

- 王耀聰 Jazz Yao-Tsung Wang
- Hadoop.TW 共同創辦人
- Hadoop The Definitive Guide 譯者
- Hadoop Operations 譯者
- 自由軟體愛好者 / 推廣者 / 開發者
- http://about.me/jazzwang - slideshare, github, etc.
- http://trac.3du.me/cloud - 原 http://trac.nchc.org.tw/cloud

http://www.cns11643.gov.tw/AIDB/fm_view.do?font=cns_0000&id=77771
http://www.cns11643.gov.tw/AIDB/fm_view.do?font=cns_0000&id=76913

鄉野調查(1)

巨量資料?!

cloud computing **big data** internet of things

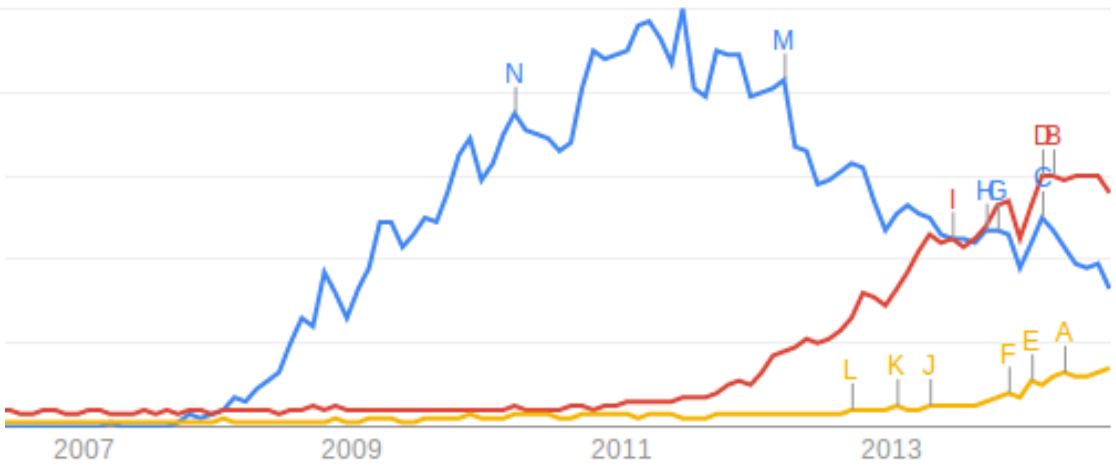
主題

Cloud computing	Big data	internet of things
搜尋字詞	搜尋字詞	搜尋字詞

查詢 | 熱門 | 上升

data analytics	100
big data analytics	100
hadoop big data	75
hadoop	75
google big data	40
big data cloud	30
big data ibm	30

期間熱門度變化



cloud computing **big data** internet of things

區域 | 城市

印度	100
新加坡	67
南韓	64
台灣	48
香港	48
美國	38
南非	28

3 Buzzwords in 2013
三大年度熱門關鍵字

物聯網

Internet of Things

雲端運算

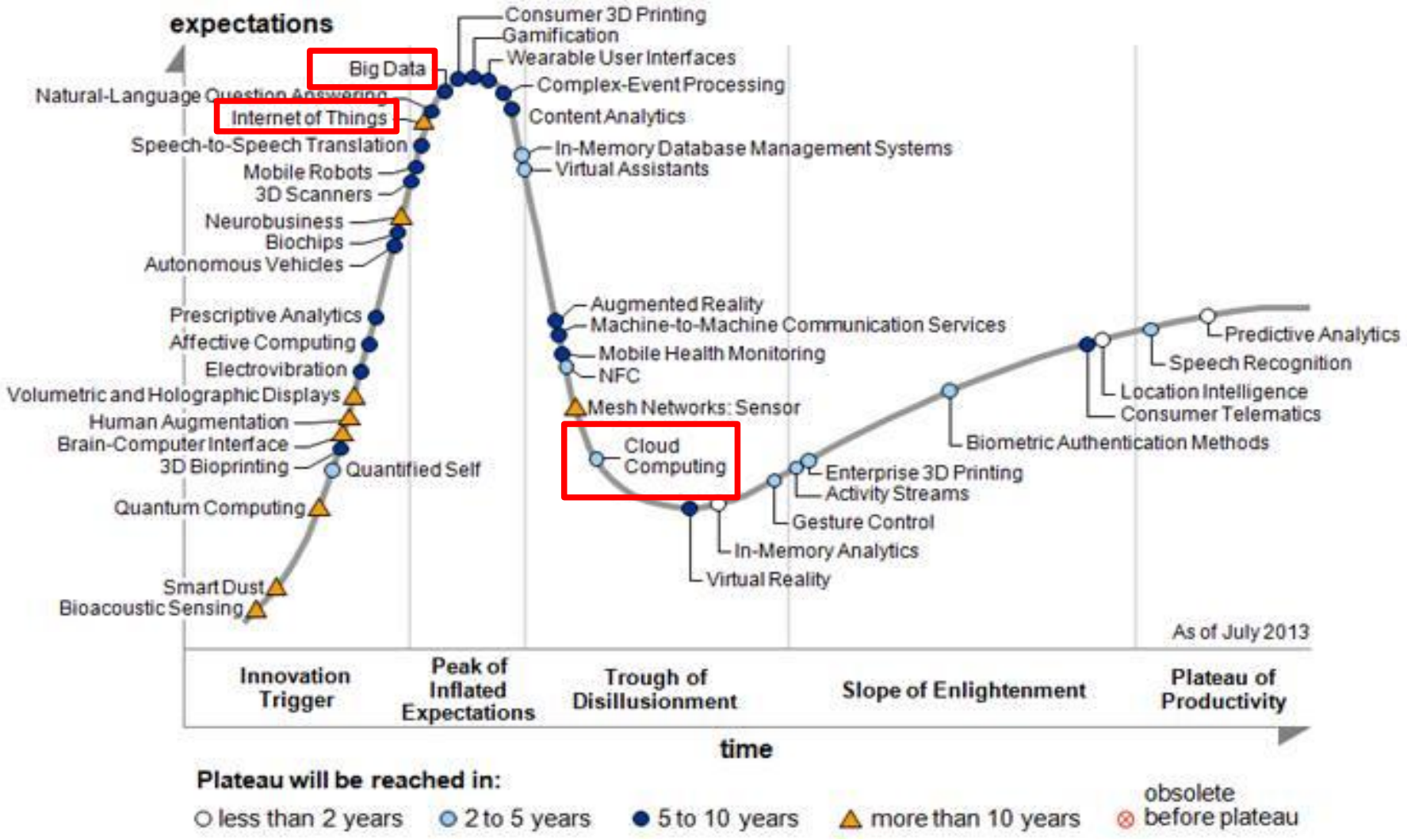
Cloud Computing

巨量資料

Big Data

市場現況：**Gartner Hype Cycle 2013**

萌芽期　夢幻期　幻滅期　平原期　高原期

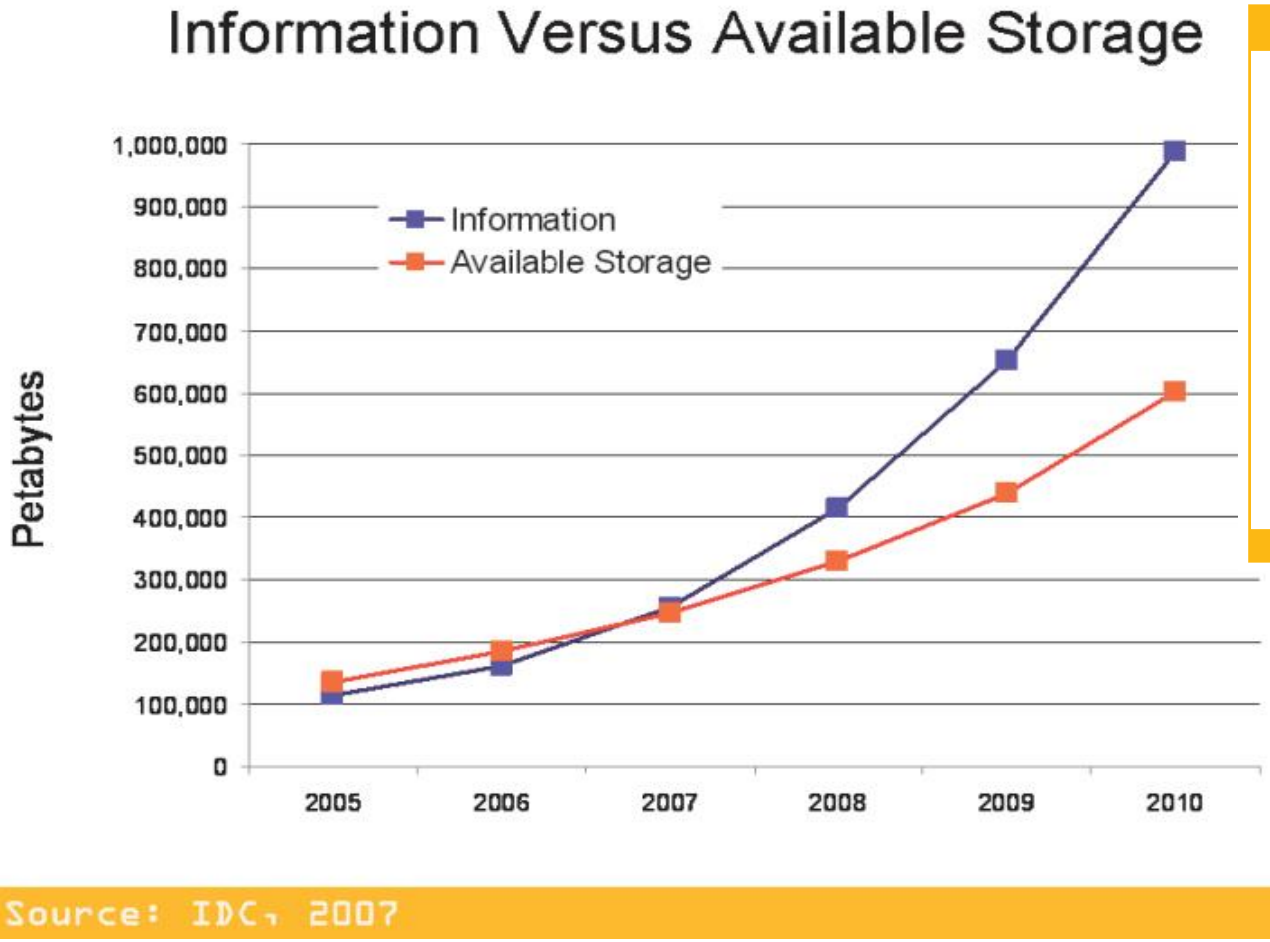


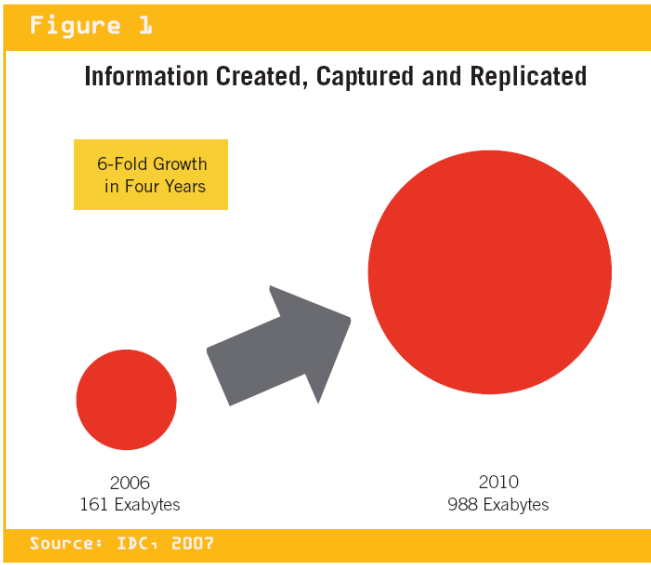
巨量資料的現況 ...
Current Status of Big Data .....

" **Big data** is like **teenage sex**:
everyone **talks about** it,
nobody really knows **how to do** it,
everyone **thinks** everyone else is doing it,
so everyone **claims** they are doing it .. "

– Dan Ariely, Professor at Duke University
and Professor at Center for Advanced Hindsight

始於2007的「資料大爆炸」時代 Data Explosion!!





2007年，IDC預估
2010年會成長六倍！
（相較2006年）

2006 161 EB
2010 988 EB (預測)

出處：The Expanding Digital Universe,
A Forecast of Worldwide Information Growth Through 2010,
March 2007, An IDC White Paper - sponsored by EMC
http://www.emc.com/collateral/analyst-reports/expanding-digital-idc-white-paper.pdf

數位宇宙以每年 1.5 倍速度成長

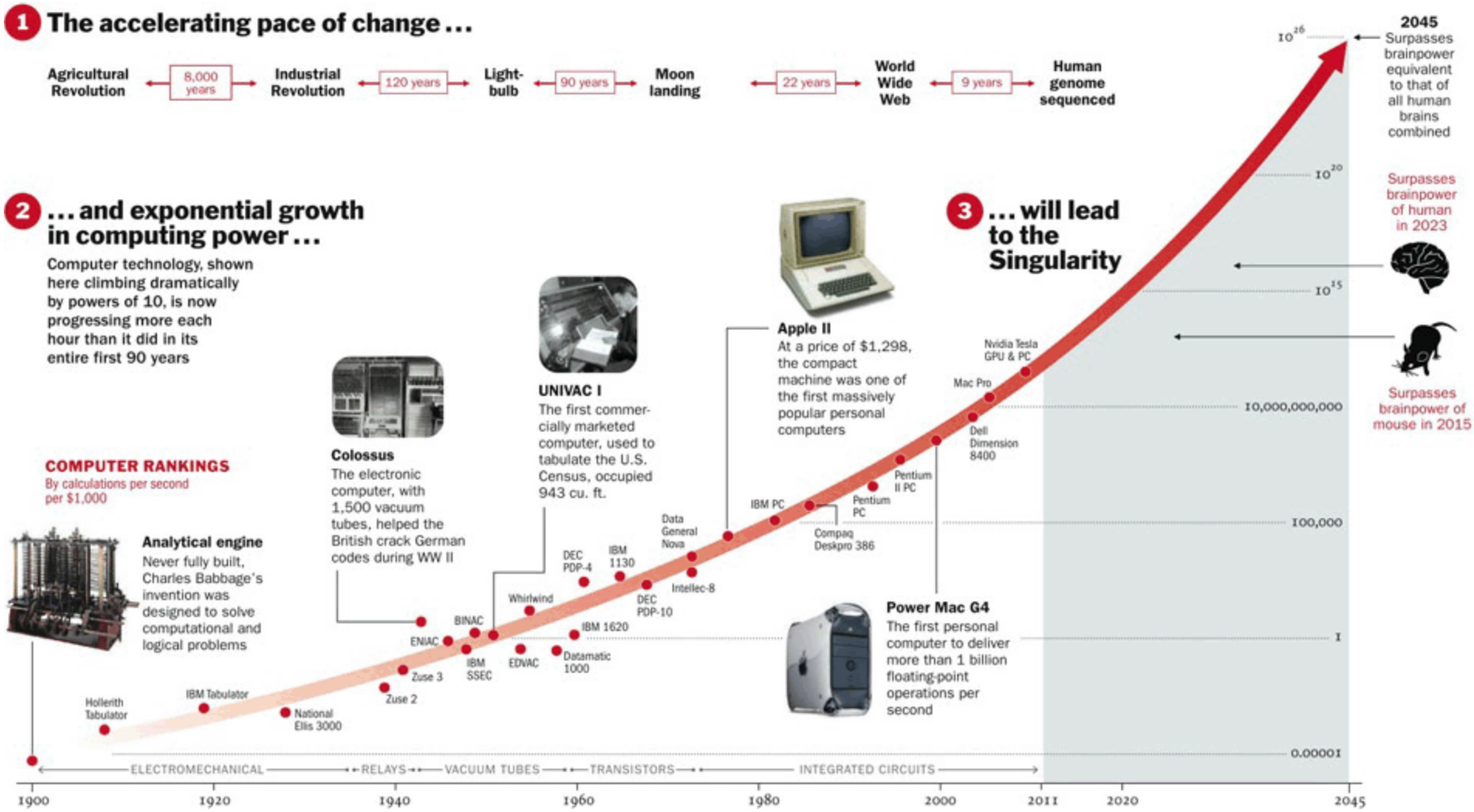
追蹤歷年的IDC數據：

2006 161 EB
2007 281 EB
2008 487 EB
2009 800 EB (0.8 ZB)
2010 988 EB (預測)
2010 1227 EB (1.2 ZB)
2011 1773 EB (預測)
2011 1800 EB (1.8 ZB)
2012 2837 EB (2.8 ZB)
2013 4400 EB (4.4 ZB)

景氣差而成長趨緩？
或受新技術抑制？

出處：The Digital Universe of Opportunities
http://www.emc.com/infographics/digital-universe-2014.htm

典範轉移的時間間距愈來愈短



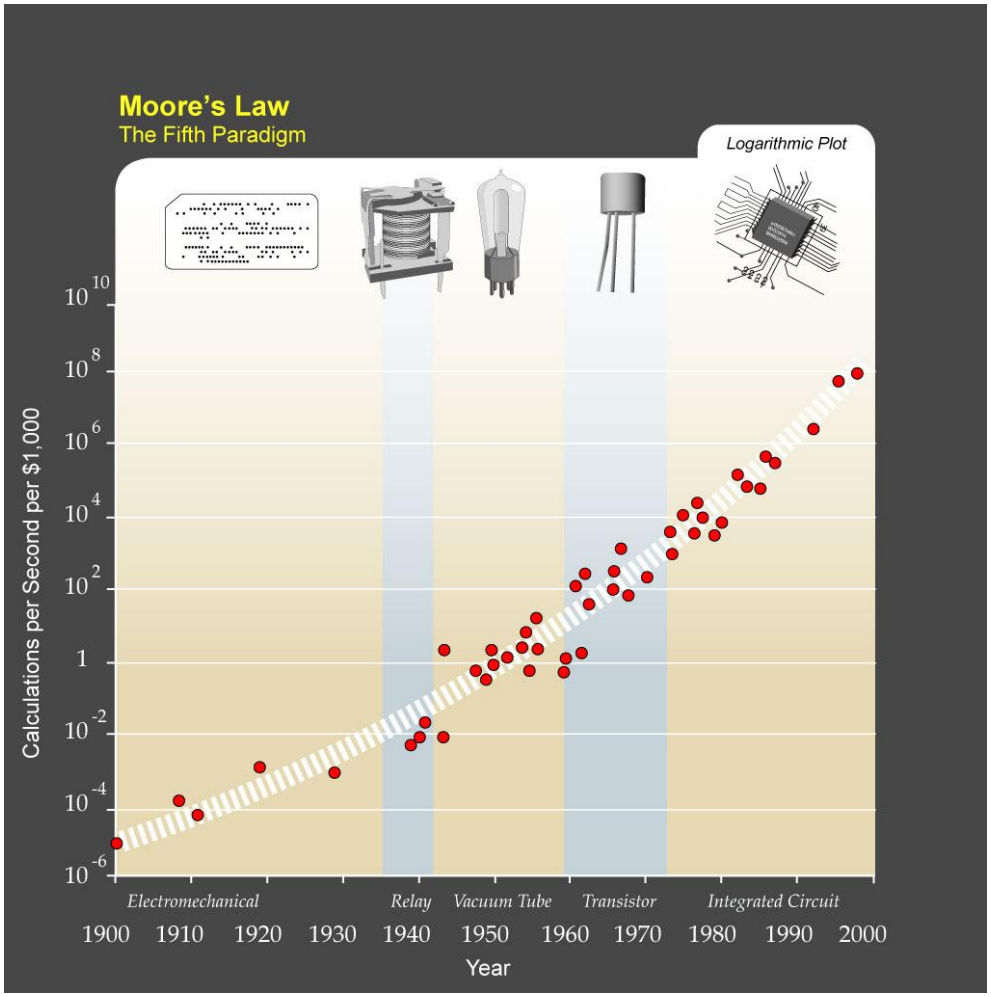
Source : TIME Magazine, "2045: The Year Man Becomes Immortal", Feb. 10, 2011
http://content.time.com/time/magazine/article/0,9171,2048299,00.html
Image Source : http://trickvilla.com/wp-content/uploads/Moores-law-graph.gif

Trend of Computing
– Moore's Law

摩爾定律是1965年由英特爾創始人之一戈登·摩爾提出來的。

在積體電路上可容納的電晶體數目，約每隔24個月便會增加一倍。

英特爾執行長David House所說：每隔18個月晶片的效能提高一倍。

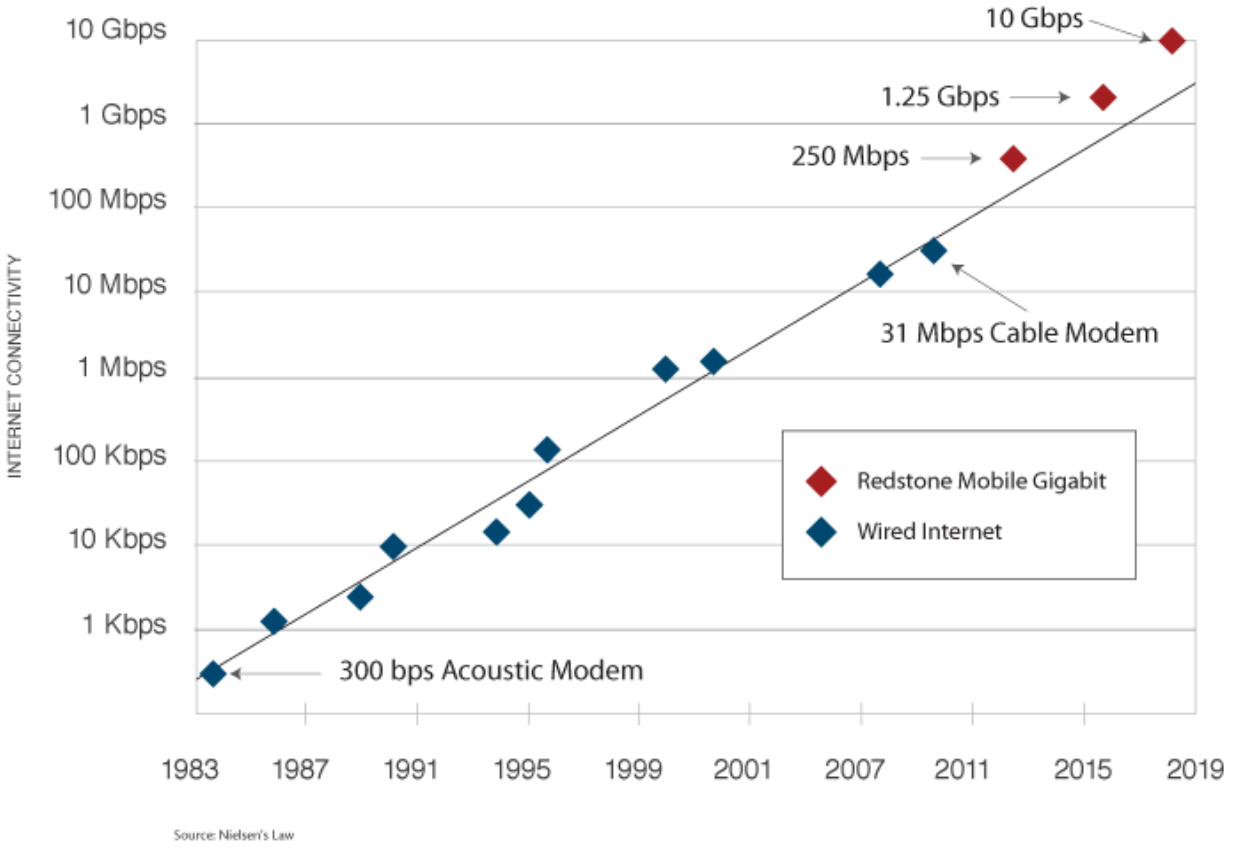


Source: Moore's Law, Wikipedia
http://upload.wikimedia.org/wikipedia/commons/c/c5/PPTMooresLawai.jpg

Trend of Network
– Nielsen's Law of Internet Bandwidth

尼爾森定律是1998年由Jakob Nielsen提出。

每隔20個月，網際網路頻寬會增加一倍。

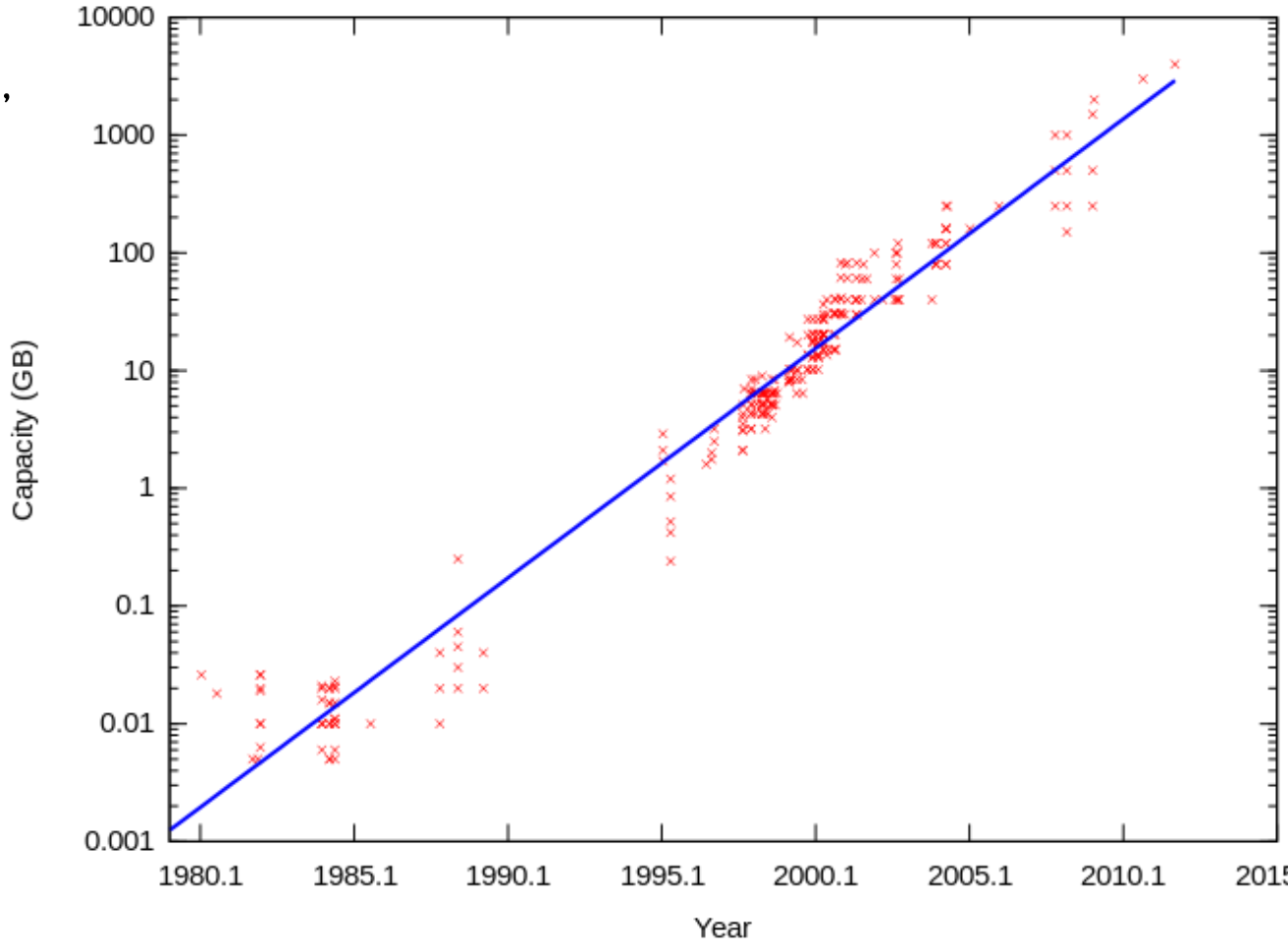


Source: "Nielsen's Law of Internet Bandwidth", April 5, 1998
http://www.nngroup.com/articles/law-of-bandwidth/
Image Source: "Nielsen's Law", May 31, 2013
http://redstone.us.com/2013/05/31/subject-nielsens-law/

Trend of Storage
– Kryder's Law

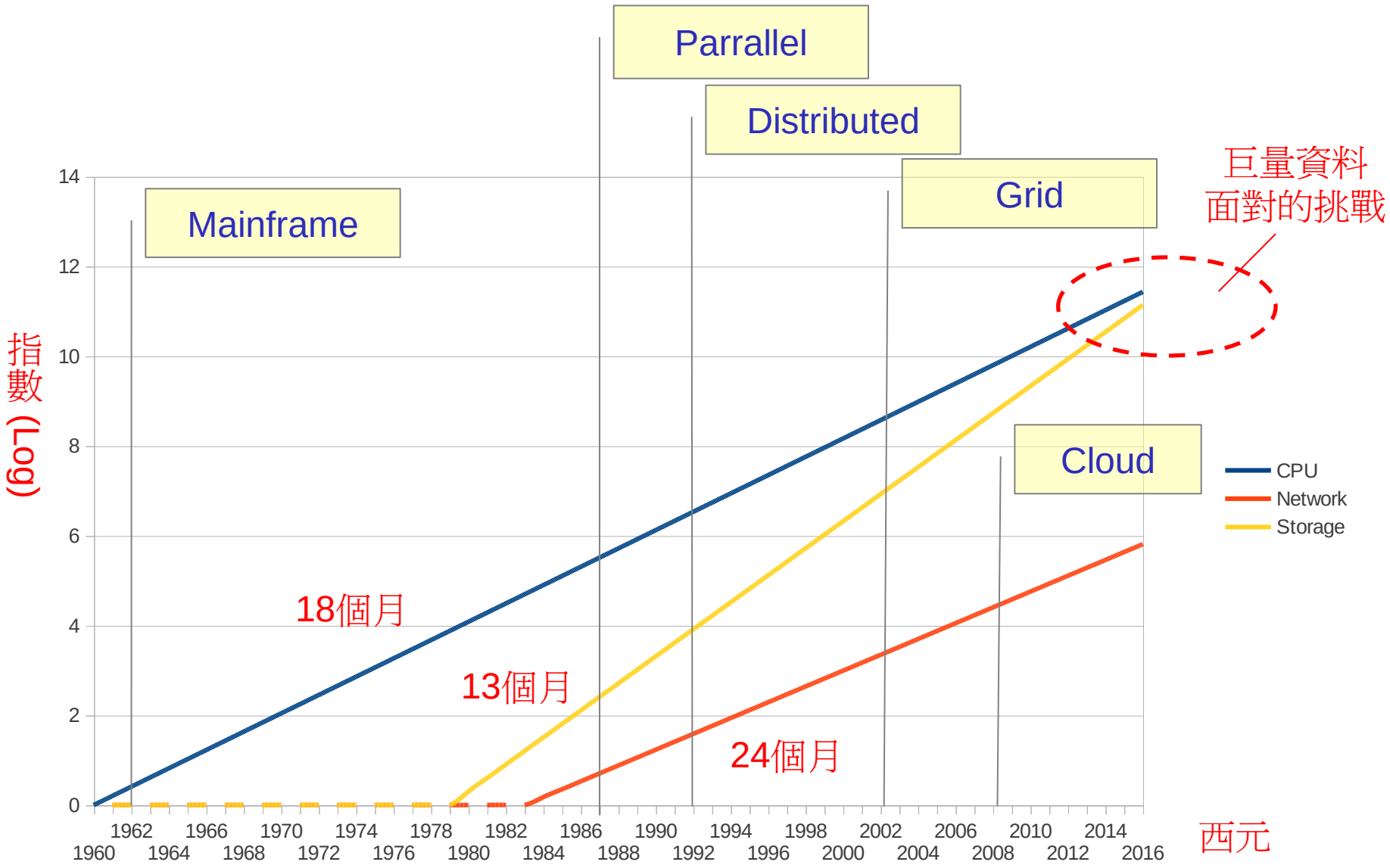
奎德定律是2005年，由希捷資深研發副總馬克奎德提出的。

每隔13個月相同價格的儲存容量就會增加一倍。



Source: 奎德定律，科學人雜誌，2005年9月號，第43期
http://203.68.243.199/saweb/read.asp?docsn=2005092489
Image Source: “Hard drive capacity over time following Kryder's Law (1980-2011)”, Wikepedia
http://en.wikipedia.org/wiki/File:Hard_drive_capacity_over_time.svg

Moore's Law , Nielsen's Law , Kryder's Law



amplab UC BERKELEY

ABOUT | PEOPLE | PUBLICATIONS | PROJECTS | SOFTWARE | BLOG: AMP BLAB | SPONSORS

For Big Data, Moore's Law Means Better Decisions

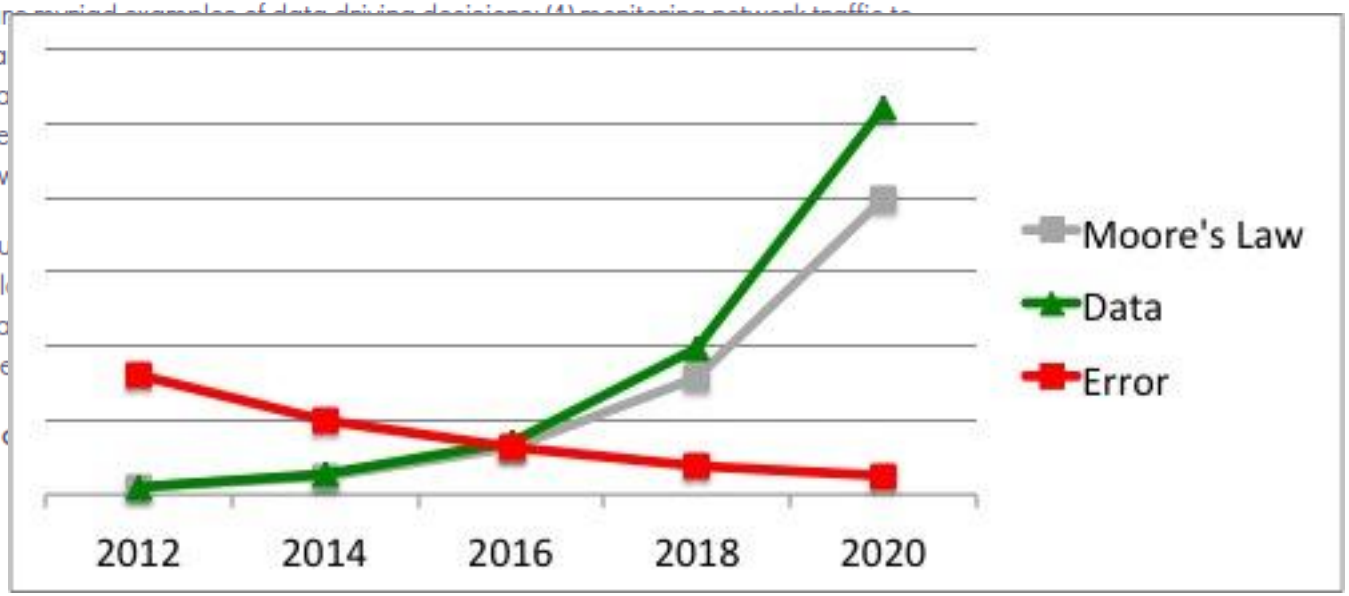
Posted on February 7, 2013 by Ion

Data Drives Decisions

Today, more and more organizations collect more and more data, and they do so with one goal in mind: extracting value. In most cases, this value comes in the form of decisions. There ar
detect a
persona
optimize
decide w

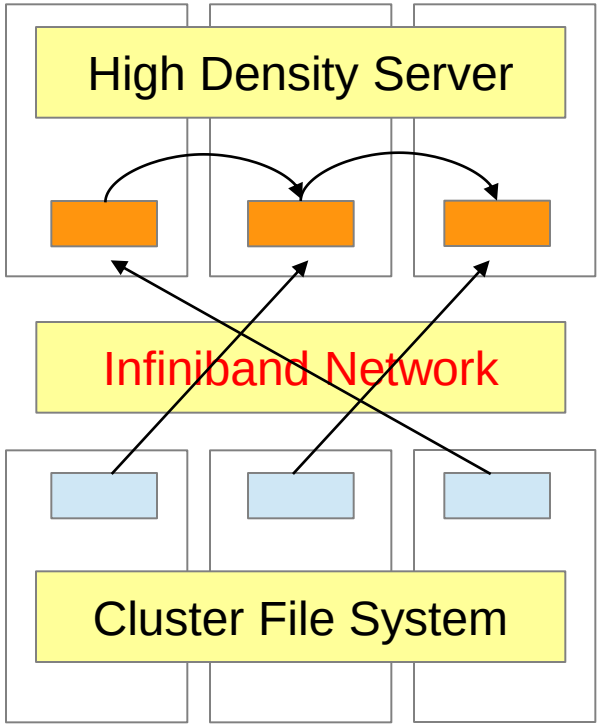
While making decisions based on this hu
the data grows faster than the Moore's l
some categories of data, such as the da
This means that, in the future, we will nee

Approximate Answers, Sampling, a



https://amplab.cs.berkeley.edu/2013/02/07/for-big-data-moores-law-means-better-decisions/

Paradigm Shift in Architecture from Computing Center to Data Center



減少資料搬運

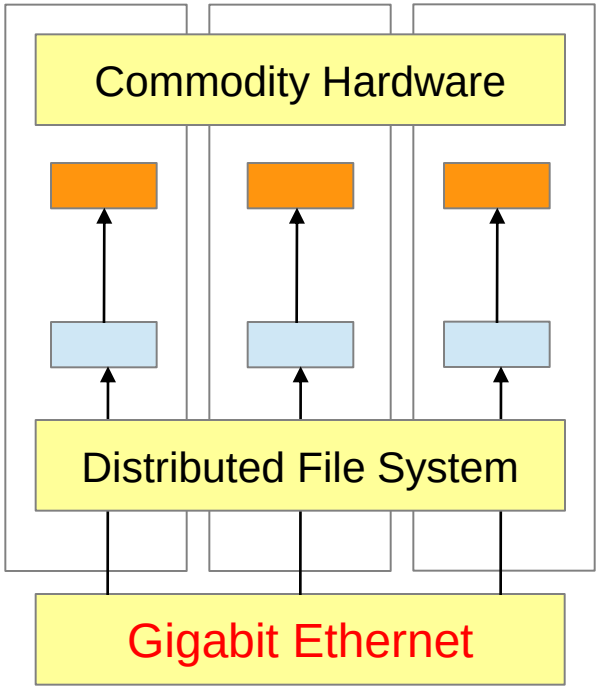
Reduce Data Transfer

強調能源效率

Energy-Efficiency

易於橫向擴充

High-Scalability



Computing Center
Move Data
To Compute
Message Passing

Data Center
Move Compute
To Data
Share Noting

巨量資料的標準定義 What is Big Data?!

海量資料泛指資料大小已無法用一般軟體擷取、管理與處理；
單一資料集大小介於數十TB至數PB的資料。

'Big Data' = few dozen TeraBytes to PetaBytes in single data set.

Definition [edit]

Big data is a term applied to data sets whose size is beyond the ability of commonly used software tools to capture, manage, and process the data within a tolerable elapsed time. Big data sizes are a constantly moving target currently ranging from a few dozen terabytes to many petabytes of data in a single data set.

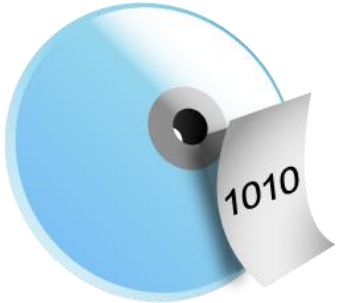
In a 2001 research report[14] and related conference presentations, then META Group (now Gartner) analyst, Doug Laney, defined data growth challenges (and opportunities) as being three-dimensional, i.e. increasing volume (amount of data), velocity (speed of data in/out), and variety (range of data types, sources). Gartner continues to use this model for describing big data.[15]

出處：http://en.wikipedia.org/wiki/Big_data

多個檔案，容量100TB

一個資料庫，容量100TB

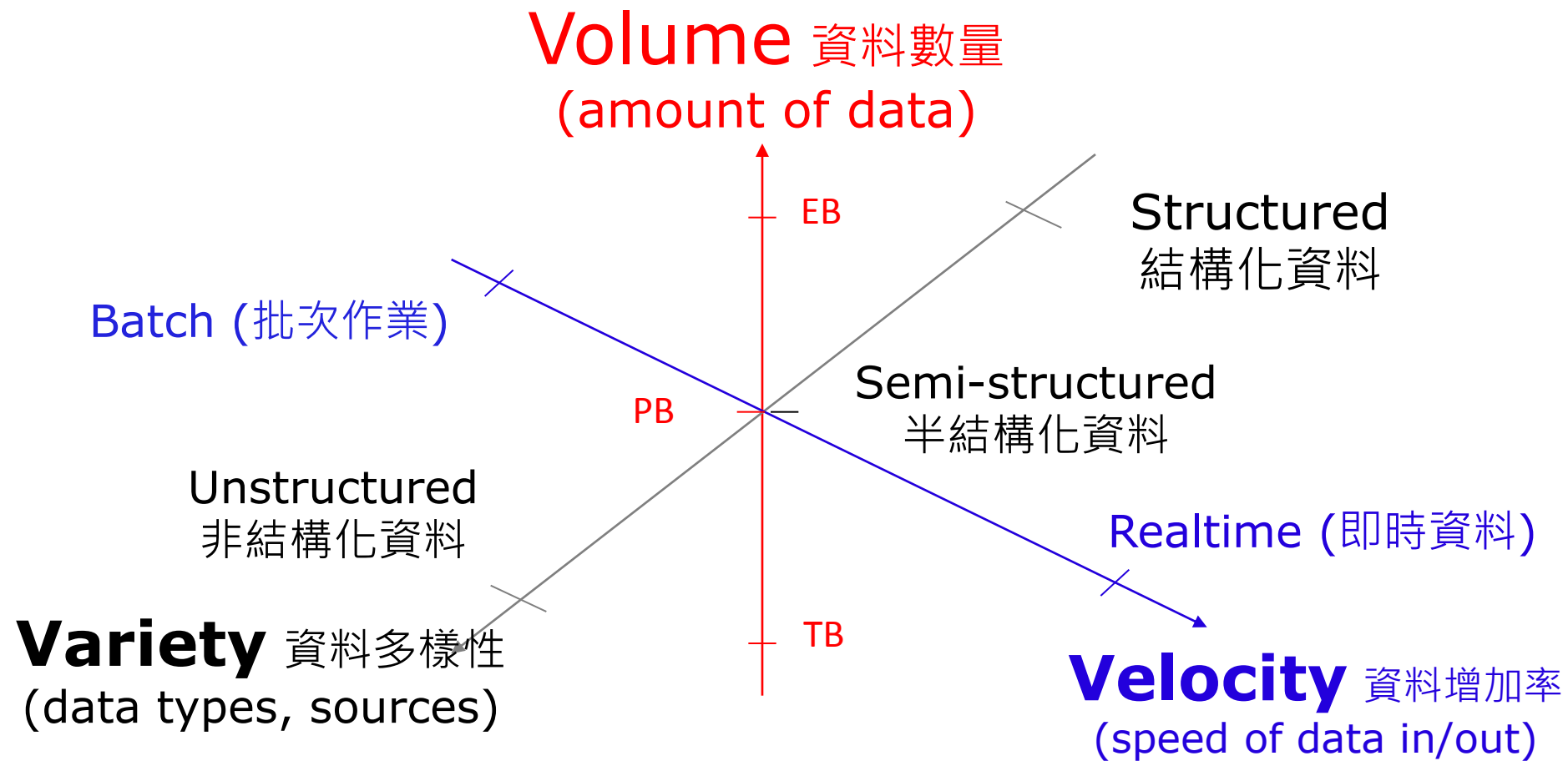
一個檔案，容量100TB



巨量資料的三大挑戰 Challenges - 3 Vs of Big Data

巨量資料的挑戰在於如何管理
「數量」、「增加率」與「多樣性」

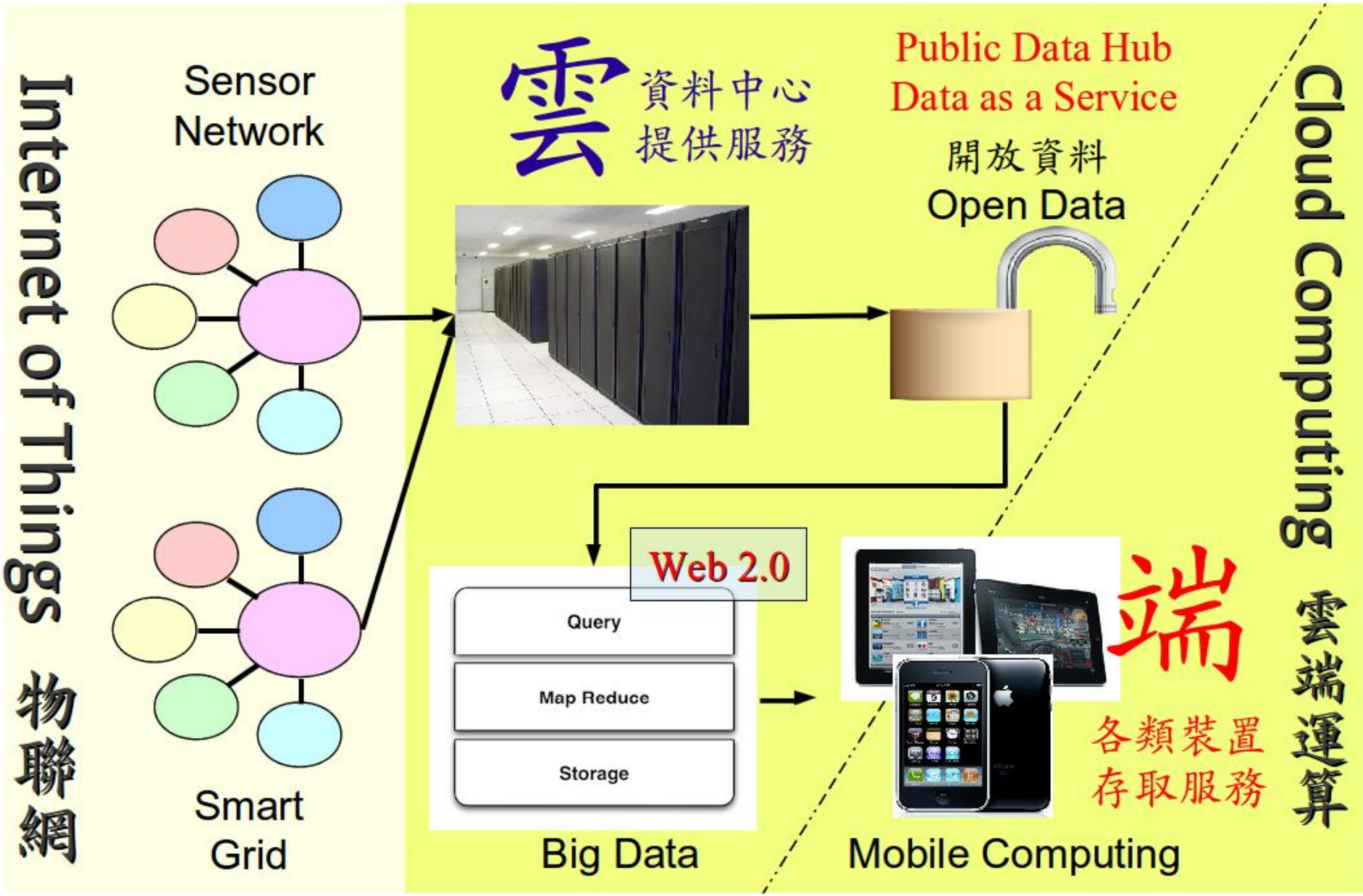
參考來源：
[1] Laney, Douglas. "3D Data Management: Controlling Data Volume, Velocity and Variety" (6 February 2001)
[2] Gartner Says Solving 'Big Data' Challenge Involves More Than Just Managing Volumes of Data, June 2011



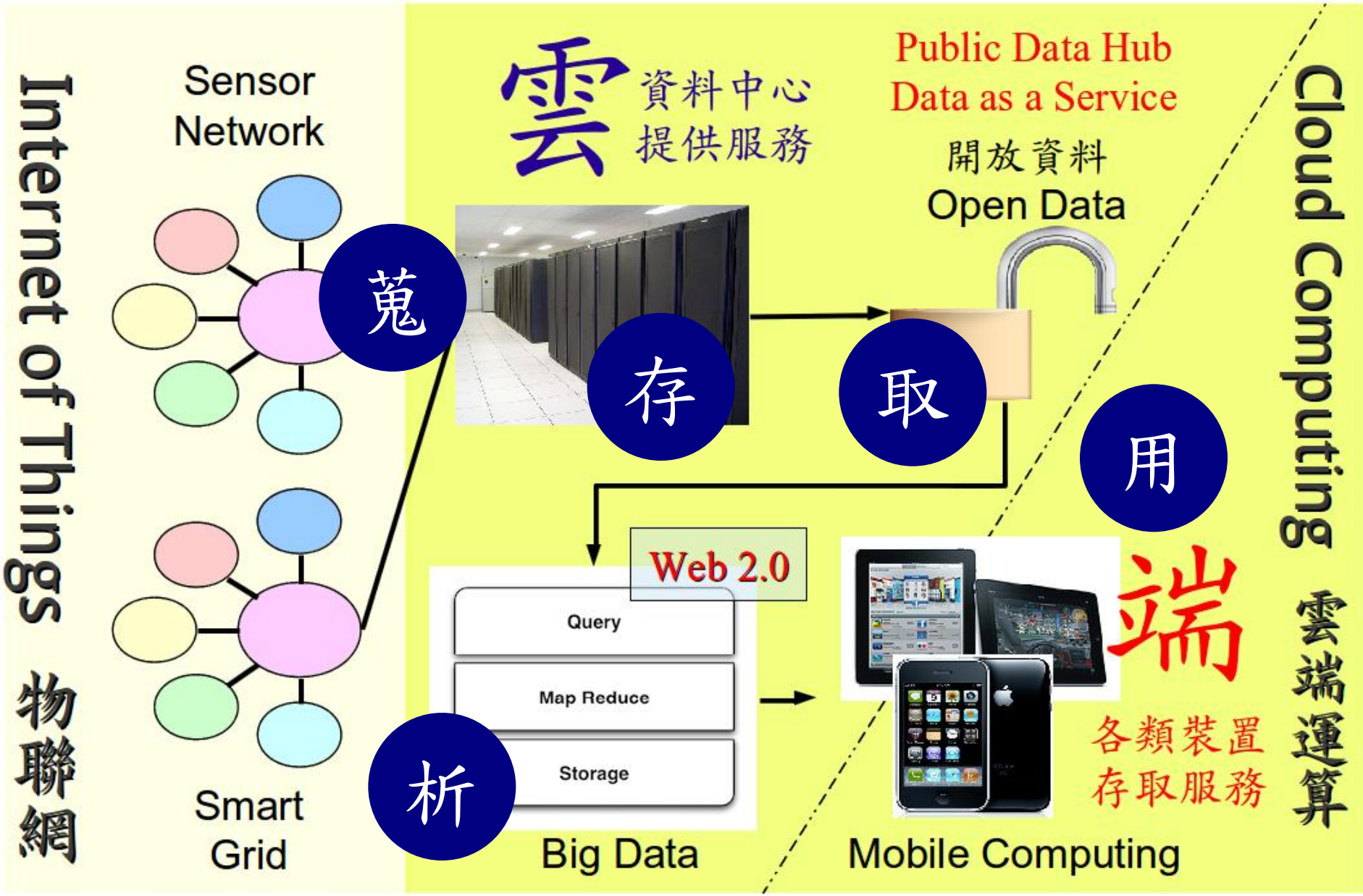
觀點：巨量資料應用的本質

出處碑帖：王羲之 / 臨鍾繇千字文
http://www.cns11643.gov.tw/AIDB/fm_view.do?font=cns_0000&id=38552

巨量資料的奇幻漂流 Life of Big Data

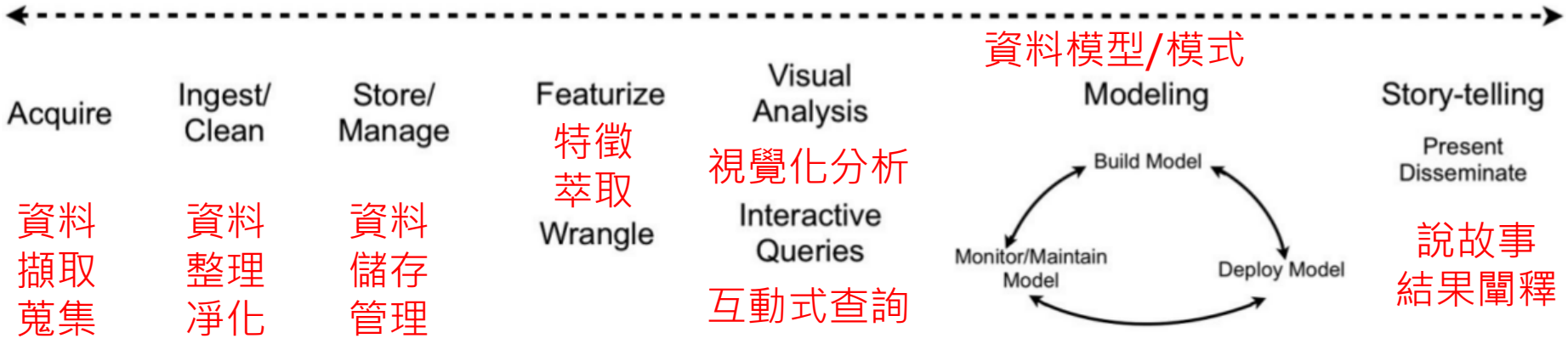


巨量資料的五大生命週期 5 Stage of Big Data Life Cycle



資料科學工作流程 Data Science Workflow

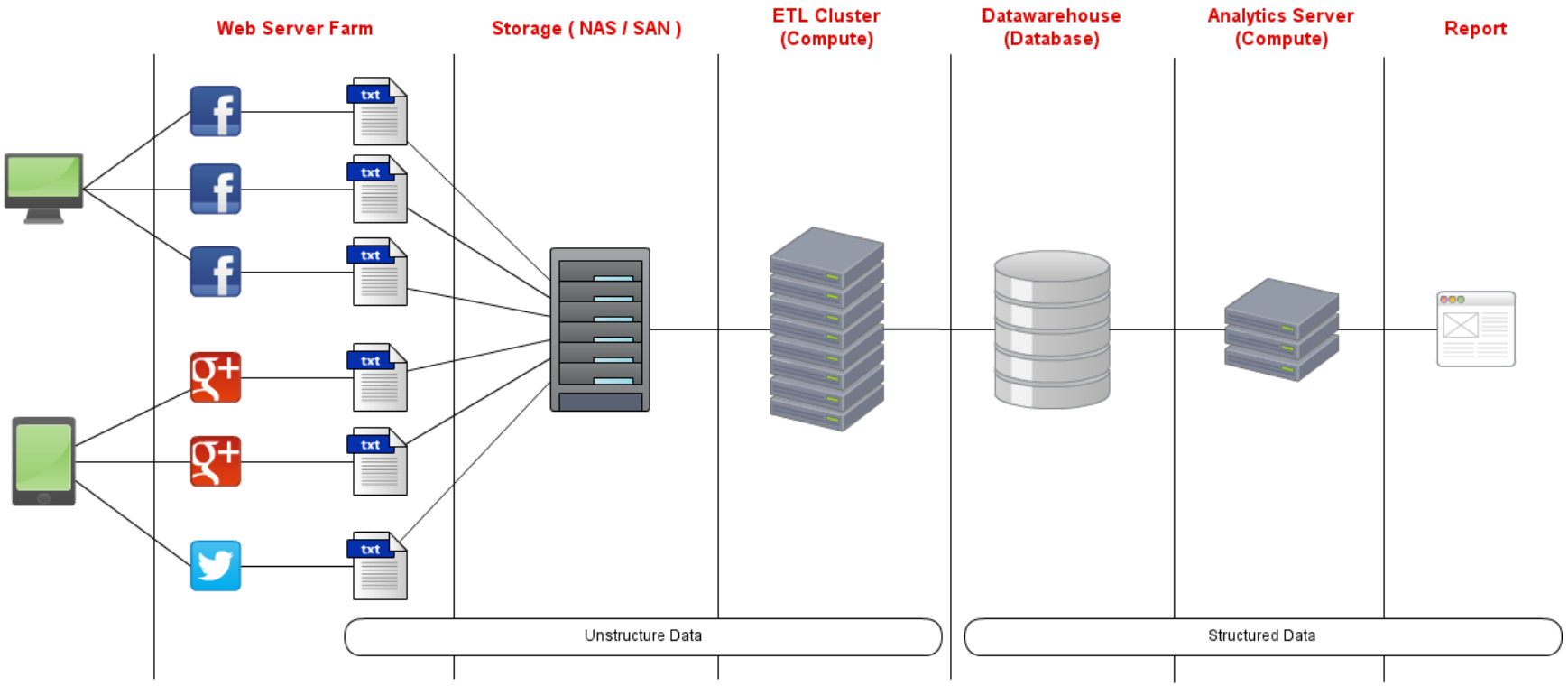
ent tasks. Data scientists tend to use a variety of tools, often across different programming languages. Workflows that involve many different tools require a lot of context-switching, which affects productivity and impedes reproducability.



"Data Analysis: Just One Component of the Data Science Workflow",
By Ben Lorica, 'Big Data Now 2013', OR'eilly
http://www.oreilly.com/data/free/files/bigdatanow2013.pdf

Ex. Data Flow of Log Analysis

生 流 存 算 用 看



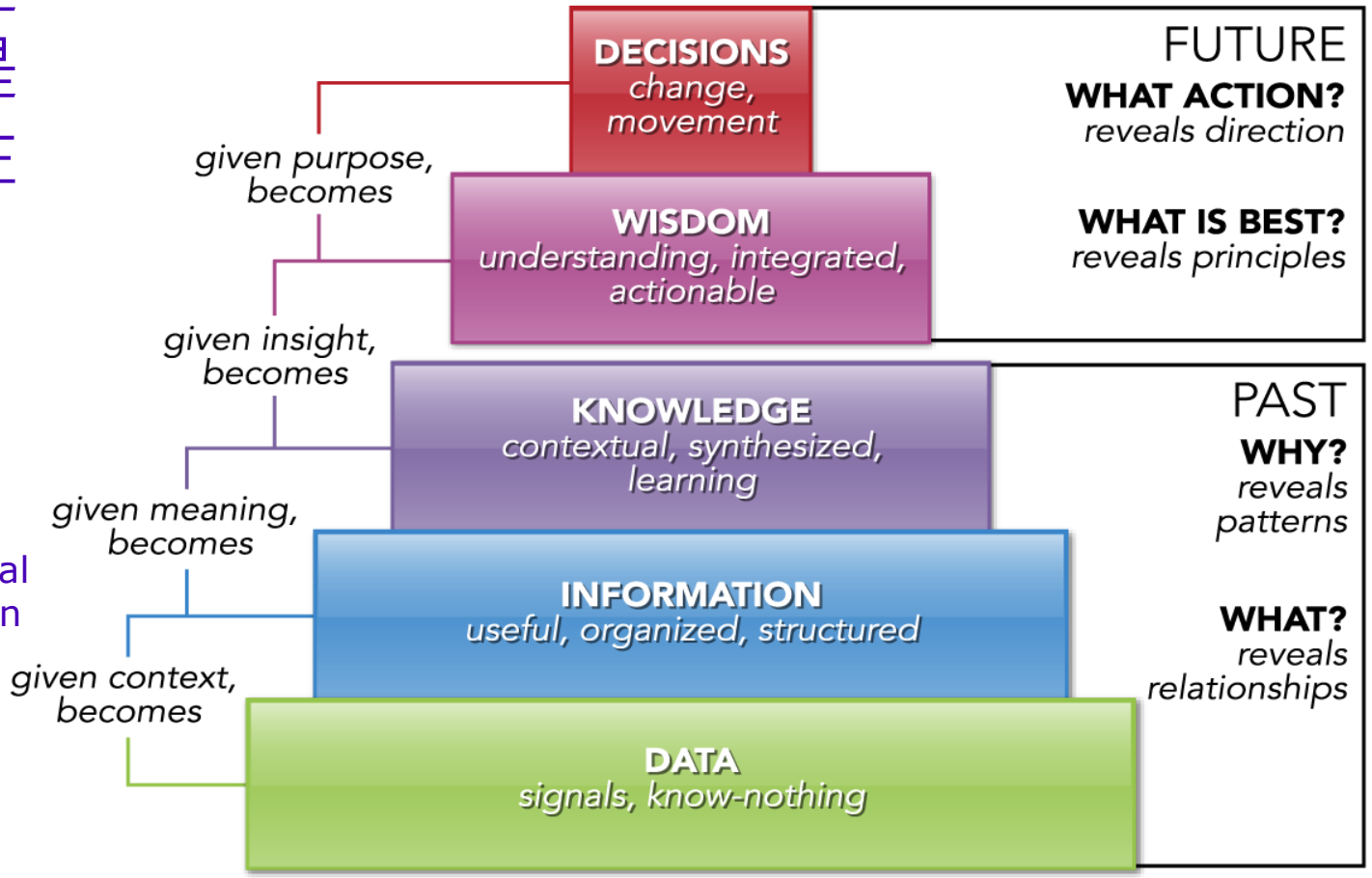
企業導入巨量資料技術的挑戰

知識源自彙整過去，智慧在能預測未來

Knowledge is from the PAST, Wisdom is for the FUTURE.

資料多寡不是重點，重點是我們想要產生什麼價值呢？時效合理嘛？成本合理嘛？

It does not matter how big is your data. The goal is to create VALUE within reasonable time period and total cost of ownership.



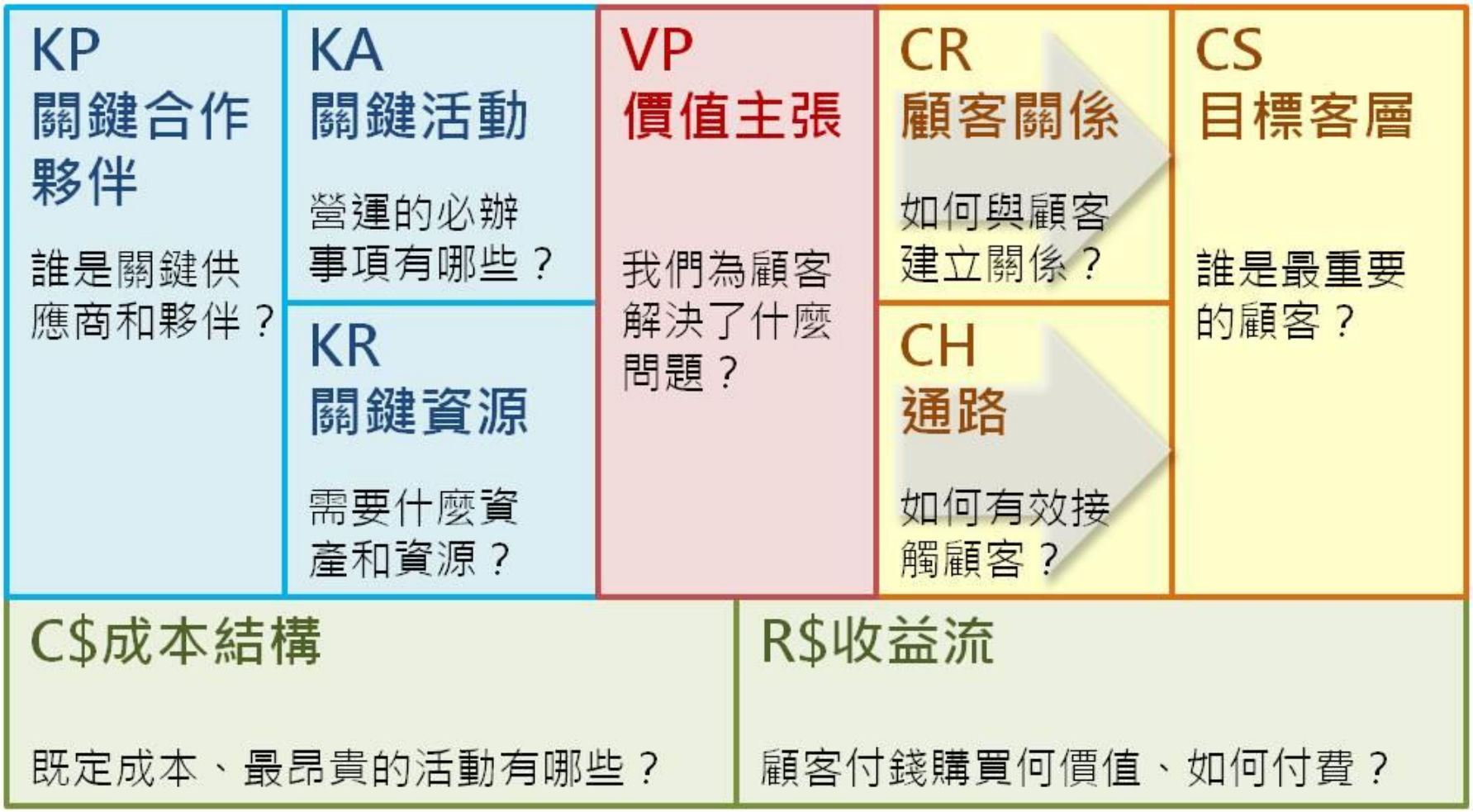
http://www.pursuantgroup.com/blog/tag/dikw-model/

大家都說「資料是金礦」，

那就讓我們拿採礦當類比吧！

國際金價	**提供給客戶的價值**	產品通路
開採成本	總擁有成本	軟硬體投資
提煉廠	分析平台與工具軟體	SMAQ
含金度	資料鑑價？	**商業模式**
開採權	分析資料的合法性	個資法
金礦	資料集	Open Data

從創新到創業，最難的是『創造價值』！



獲利世代：自己動手，畫出你的商業模式
Business Model Generation
http://www.books.com.tw/products/0010567254

雲端運算的商業模式
Business Model of Cloud Computing

規模經濟 (Economies of Scale)

眾人共用資料中心的軟硬體資源，降低總持有成本

資料即服務 (Data as a Service)

綁架你的資料，當資料越來越大，網路傳不動，你就付錢吧！

網路即通路 (Network as a Channel)

一雲多螢，雲端的成功關鍵在於網路頻寬普及率

運算即價值 (Compute as a Value)

當資料集中，連結愈多，愈能透過運算的手段，
找出群眾的智慧，就是提供給客戶最好的價值！

巨量資料處理技術：過去、現在、未來
Big Data Technologies: PAST, NOW and FUTURE

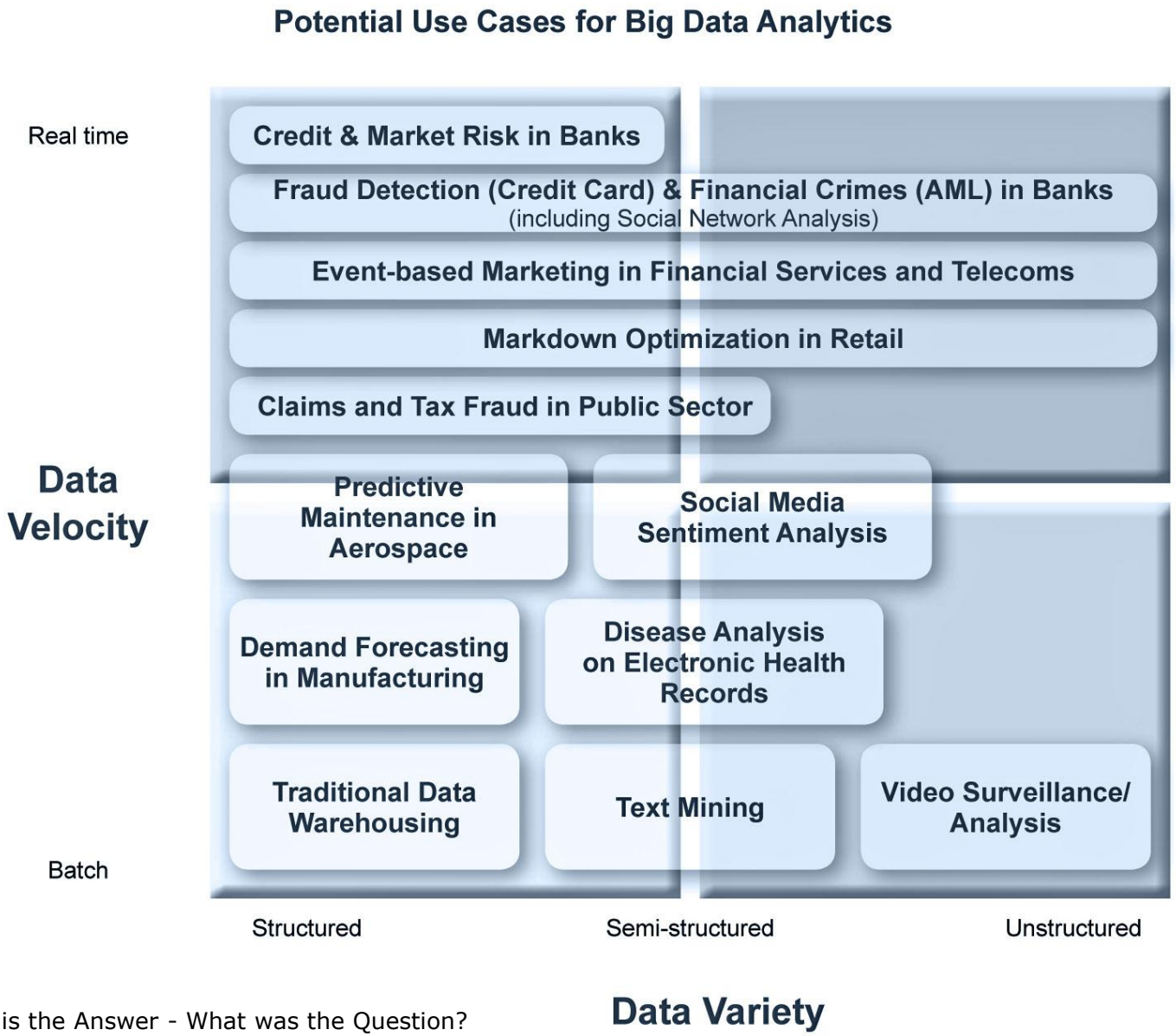
巨量資料處理的資訊架構
The SMAQ stack for big data

LAMP is for Web Services.

未來處理海量資料的人必需知道
You will need a big data stack called SMAQ（Storage, MapReduce and Query）

參考來源：The SMAQ stack for big data，Edd Dumbill，22 September 2010，
http://radar.oreilly.com/2010/09/the-smaq-stack-for-big-data.html
圖片來源：http://smashingweb.ge6.org/wp-content/uploads/2011/10/apache-php-mysql-ubuntu.png

不同的應用特性，落在不同的象限



參考來源：Big Data is the Answer - What was the Question?
http://www.saama.com/blog/bid/76211/Big-Data-is-the-Answer-What-was-the-Question

PAST: Big Data at Rest

Can gigabytes predict the next Lady Gaga?

By Stacey Higginbotham

Want to know how playing on Jimmy Kimmel Live will boost the sales of an artist's album? Or how about figuring out where fans go to find artists after they hit the evening news? What about the effect Whitney Houston's death had on her YouTube and Vevo plays? They shot up 4,525 percent, by the way.

http://nextbigsound.com/

fakebook®

NETFLIX

How big data can curb the world's energy consumption

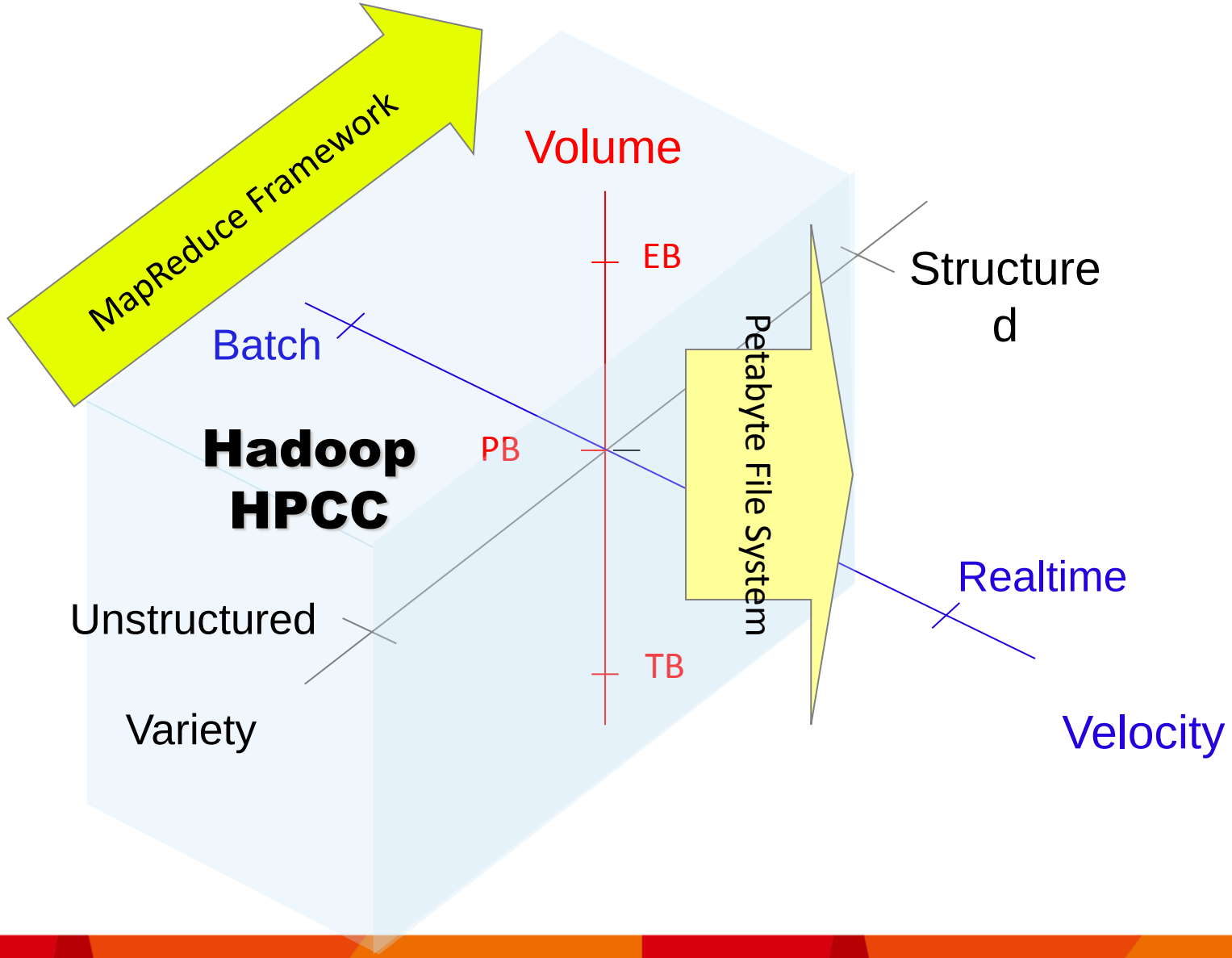
http://www.openpdc.com/

One hospital's embrace of big data

Source: 10 ways big data changes everything, http://gigaom.com/2012/03/11/10-ways-big-data-is-changing-everything

處理巨量資料的三類技術(1)

Data at Rest – MapReduce Framework



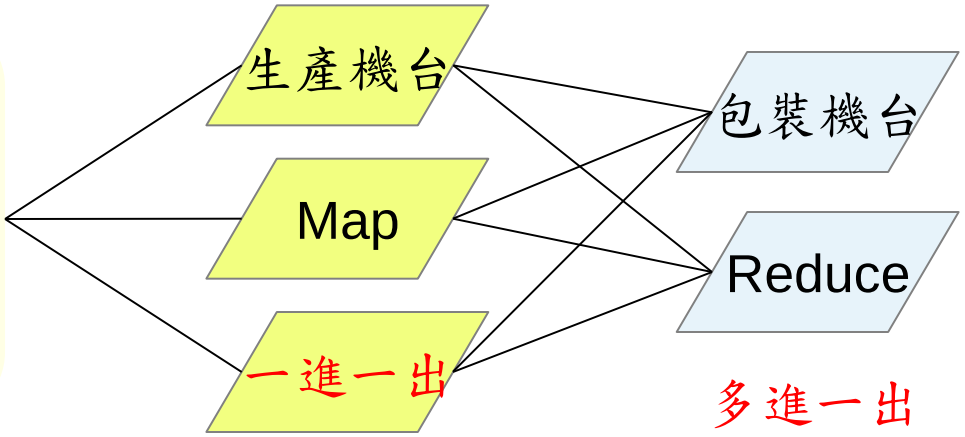
高資料通量處理平台 Hadoop
Key Concept : Data Locality

Hadoop 是一個讓使用者簡易撰寫並執行處理海量資料應用程式的軟體平台。

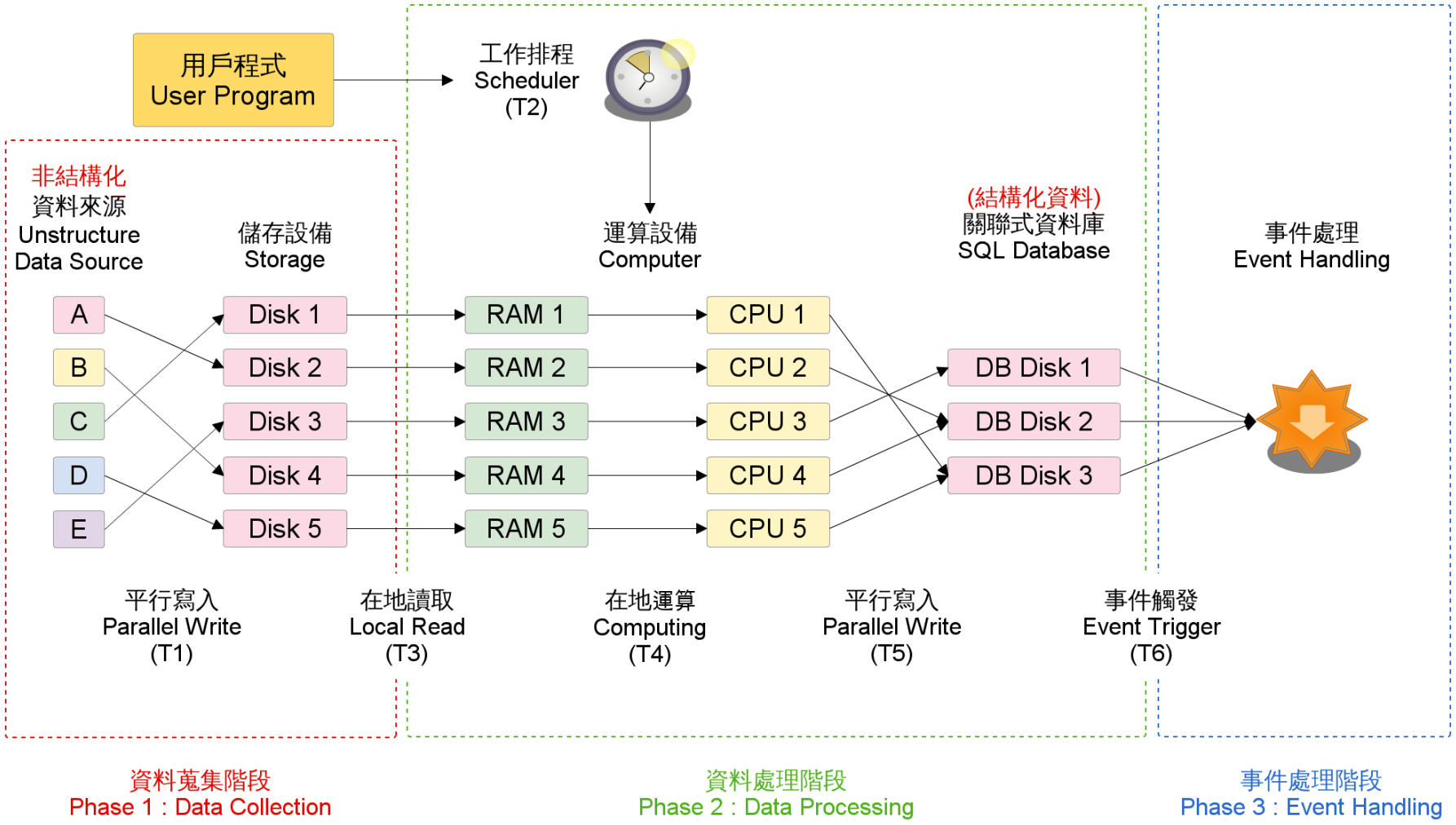
亦可以想像成一個處理海量資料的生產線，只須學會定義 **map** 跟 **reduce** 工作站該做哪些事情。

就像工廠的倉庫
存放生產原料跟待售貨物

HDFS存放
待處理的非結構化資料
與處理後的結構化資料



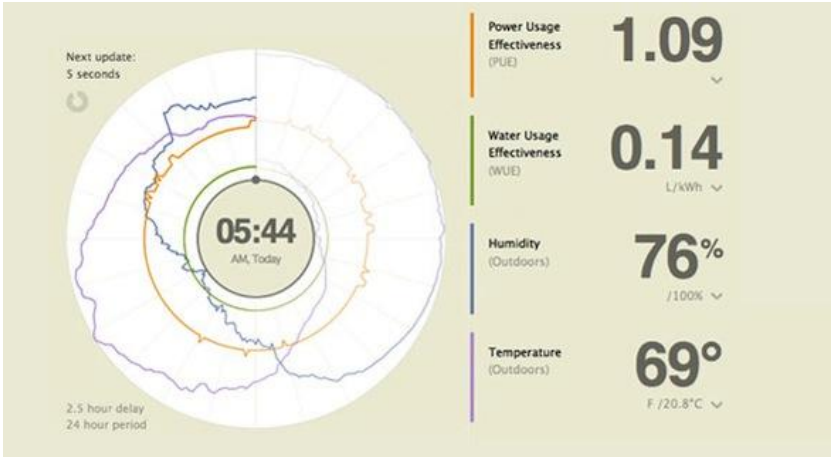
批次作業的運算時間
Processing Time of Batch Jobs



NOW: Big Data in Motion

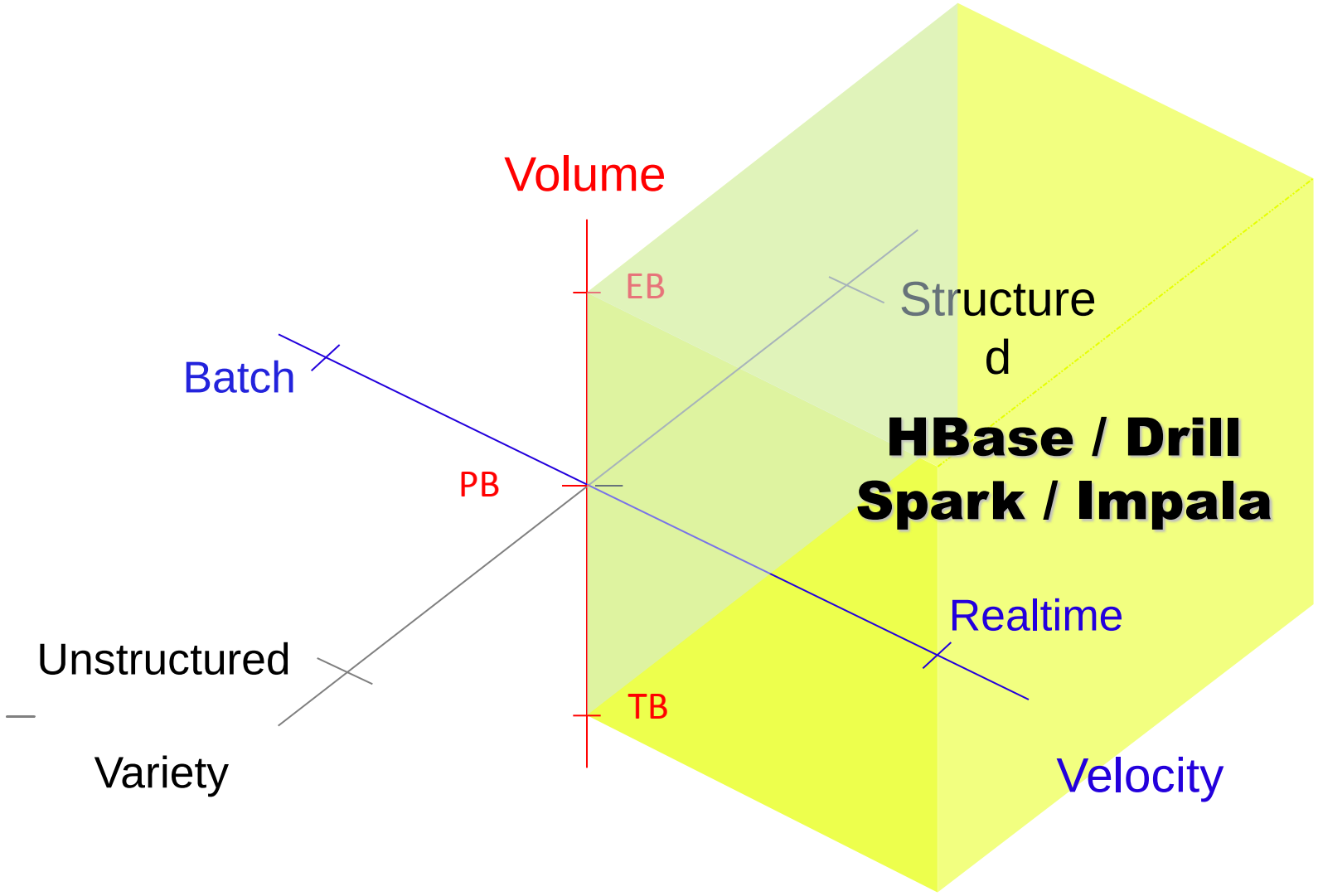
[金融] **Trading Robot**

[災防] 海嘯、土石流
Disaster Prevention
Tsunami Forecast



[資訊] 機房即時用電資訊監控、警訊
Realtime Data Center Power Usage and related notifications
http://www.newmobilelife.com/2013/04/21/facebook-pue-real-time-charts/

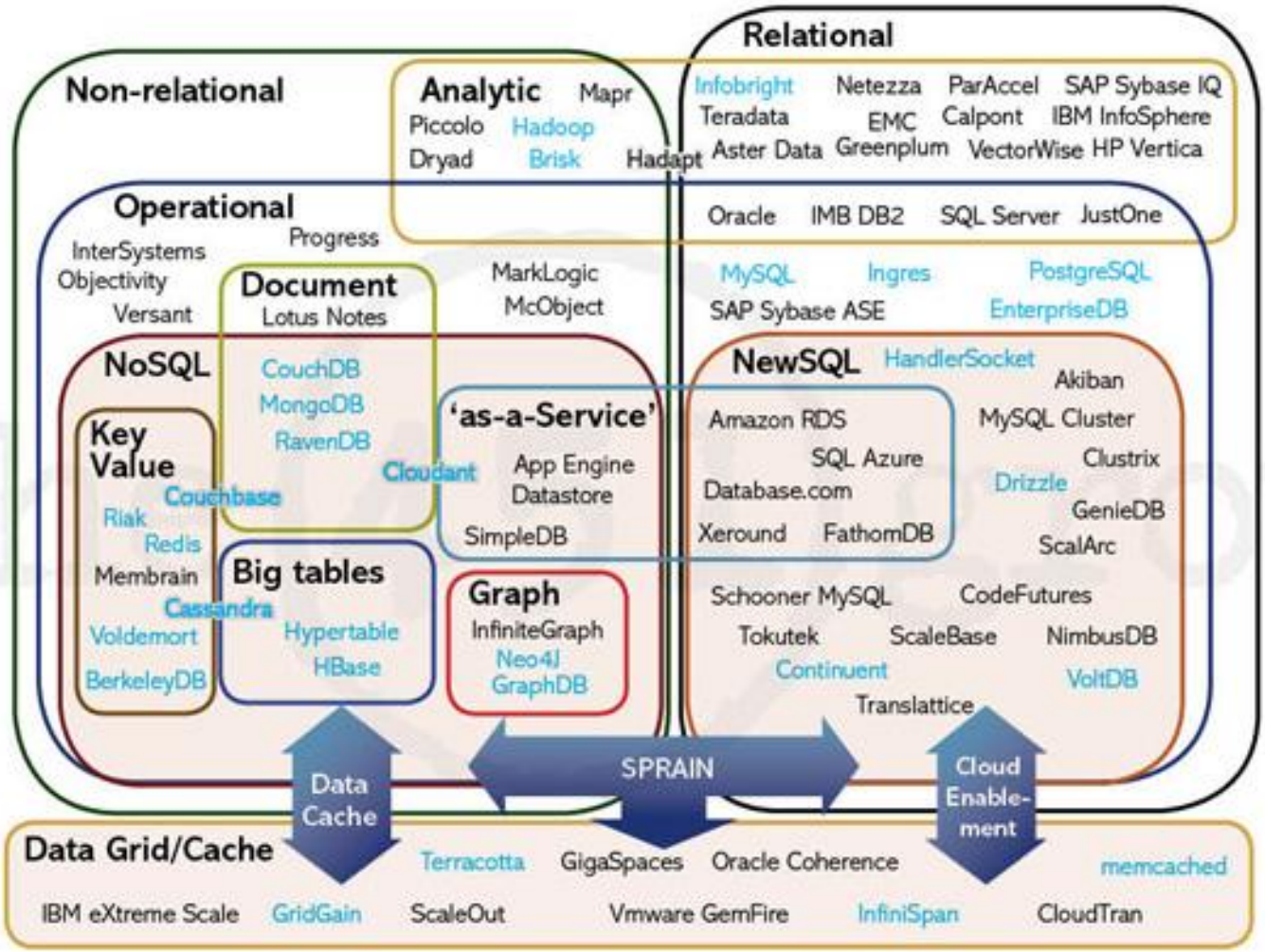
處理巨量資料的三類技術(2)
Data in Motion – In-Memory Processing



Google的技術演進 vs Apache 專案

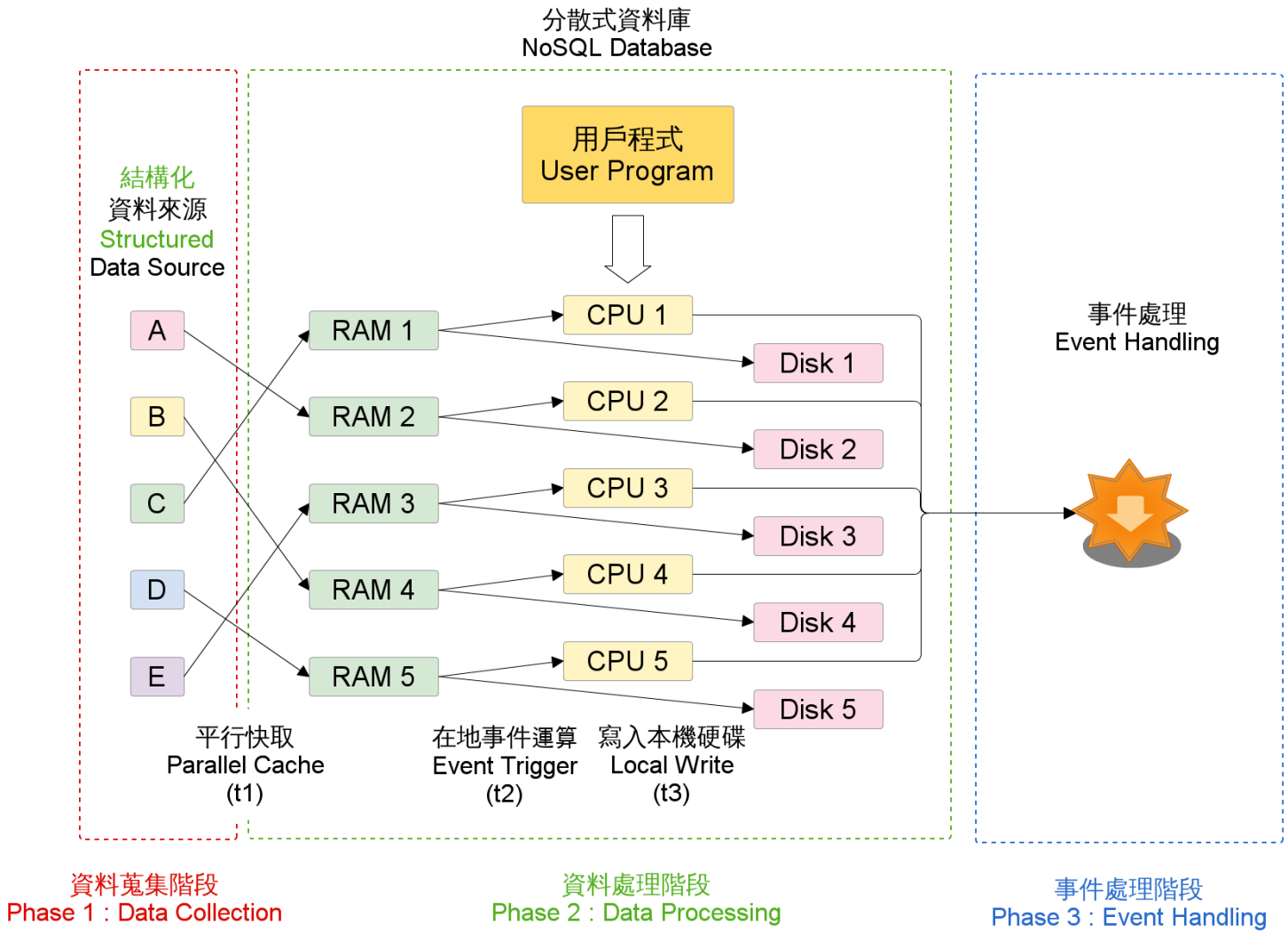
Big Query (JSON, SQL-like)	Dremel (2010)	Apache Drill (2012)
Incremental Index Update (Caffeine)	Percolator (2010)	
Graph Database	Pregel (2009)	Apache Giraph (2011)
Query	BigTable (2006)	Apache HBase (2007)
Map Reduce	MapReduce (2004)	Hadoop MapReduce (2006)
Storage	Google File System (2003)	HDFS (2006)

令人眼花撩亂的多樣化資料庫選擇
NoSQL vs NewSQL

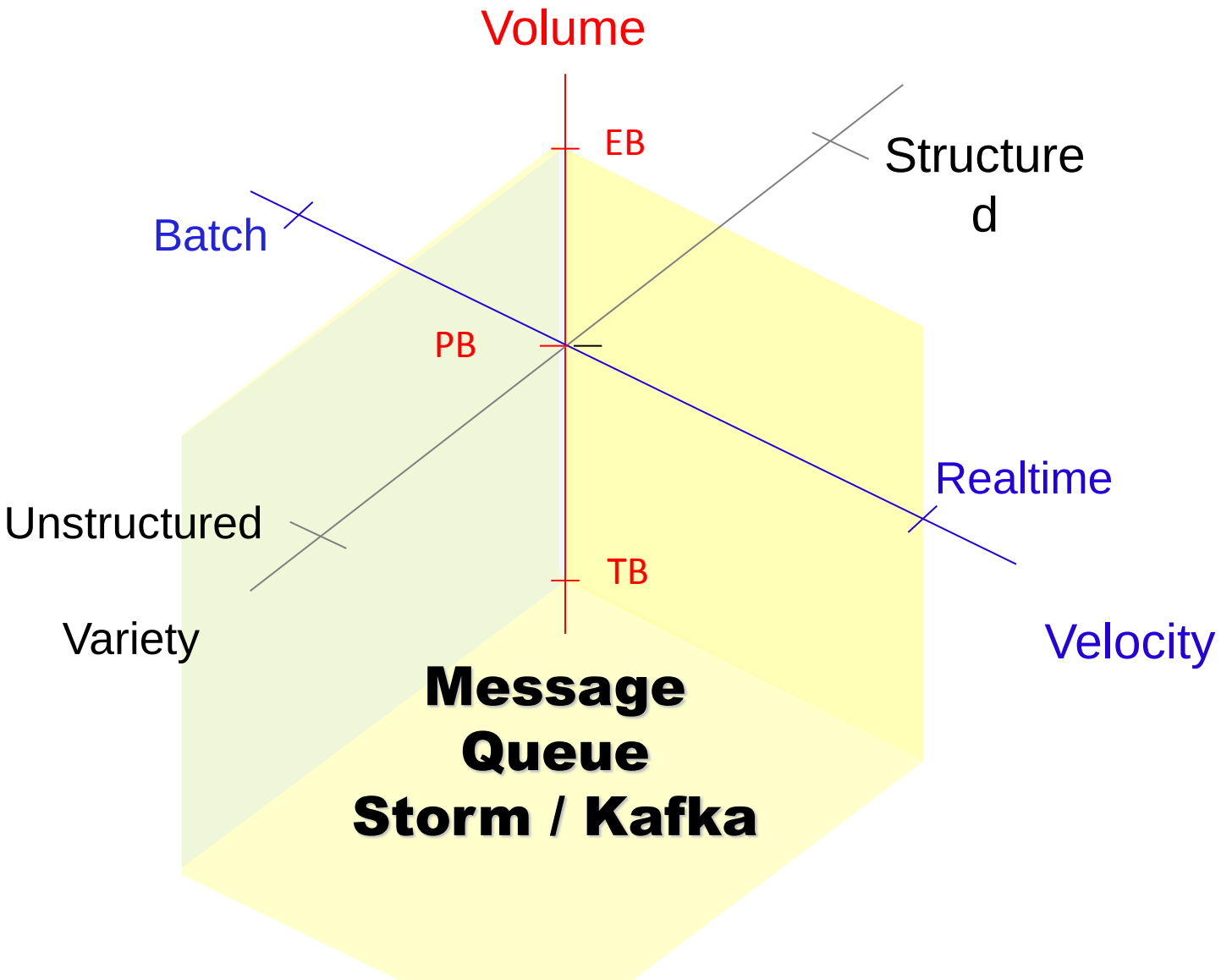


http://www.infoq.com/news/2011/04/newsql

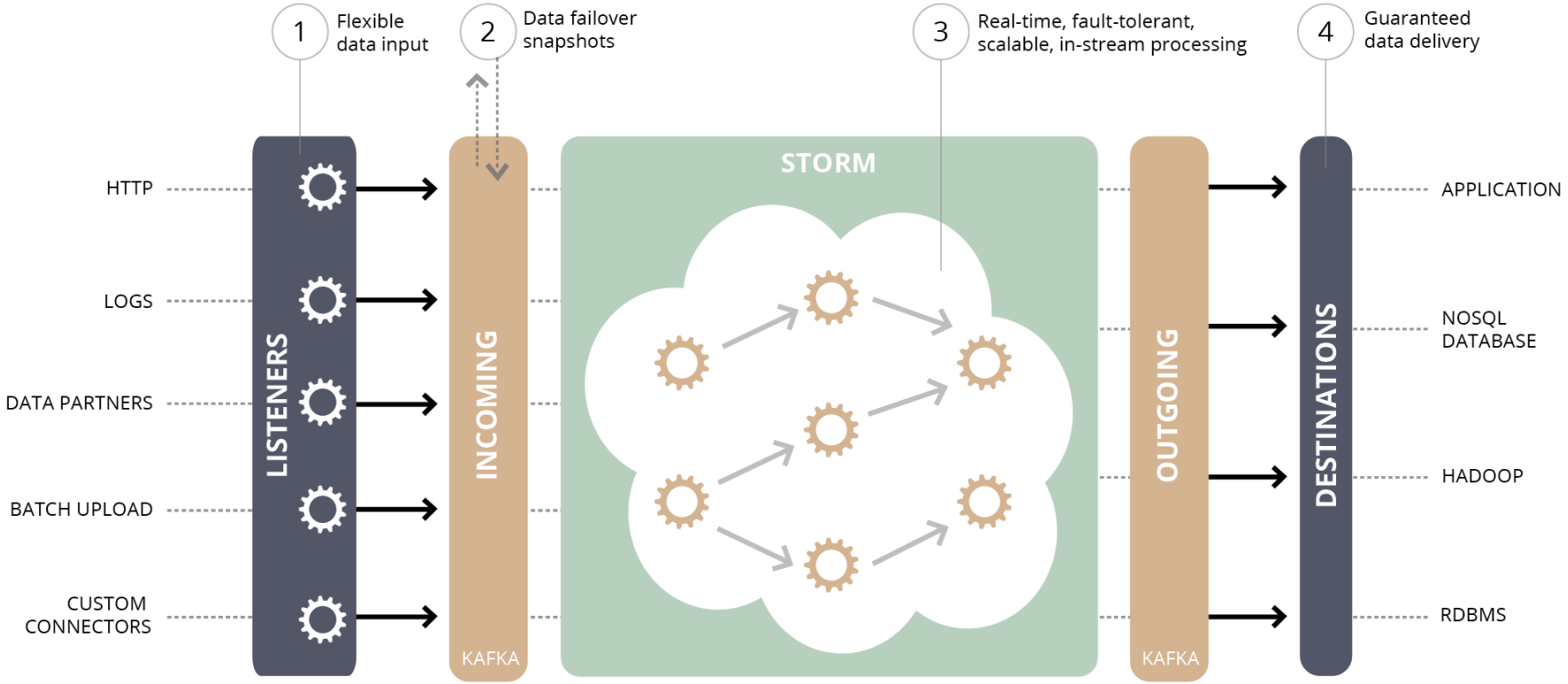
In-Memory Processing的運算時間
以HBase為例



處理巨量資料的三類技術(3)
Streaming Data Collection , Data Cleaning



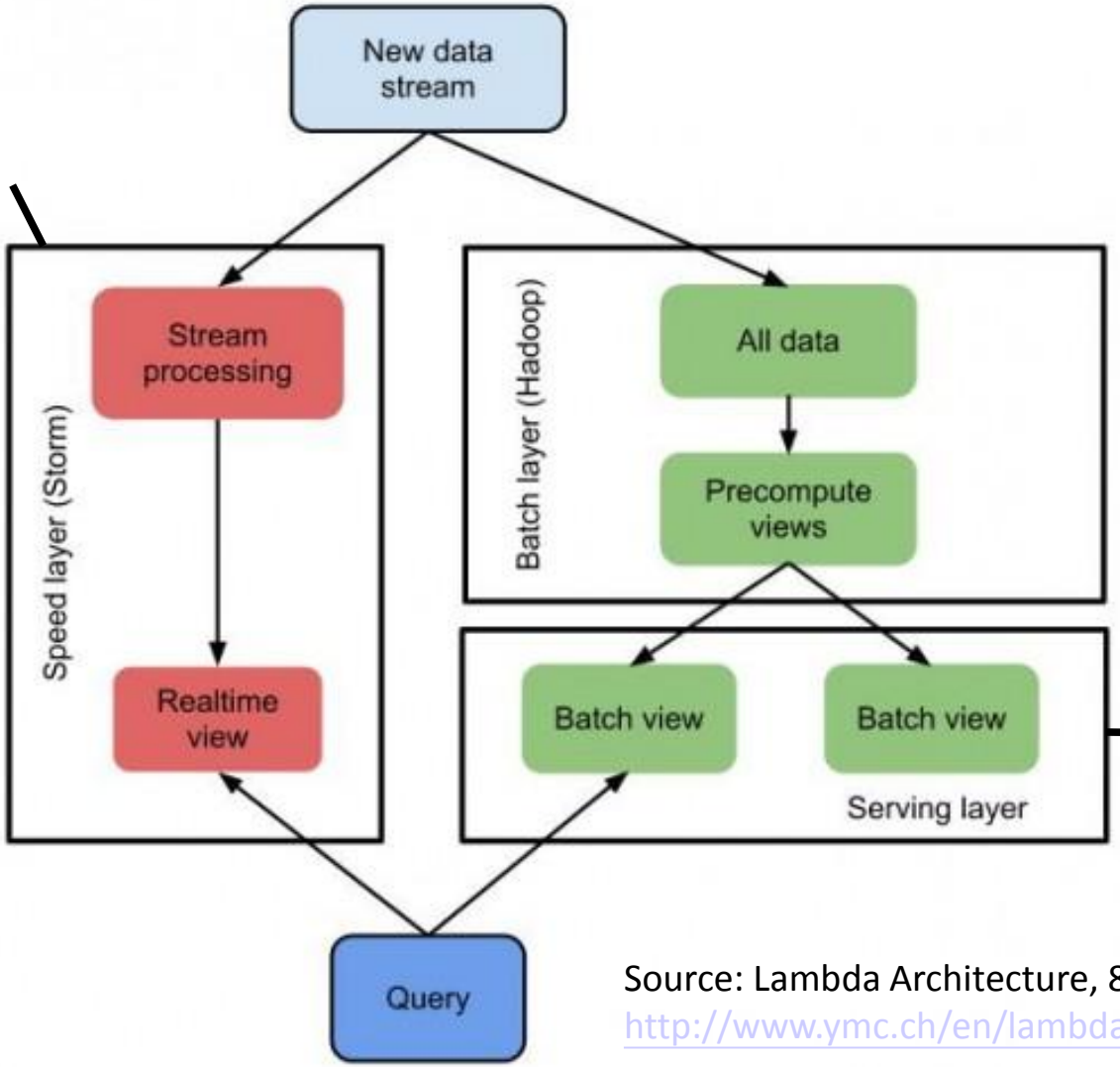
Twitter Storm
+ Apache Kafka



http://blog.infochimps.com/2012/10/30/next-gen-real-time-streaming-storm-kafka-integration/

混合模式的巨量資料處理架構
Lambda Architecture for Big Data after 2013

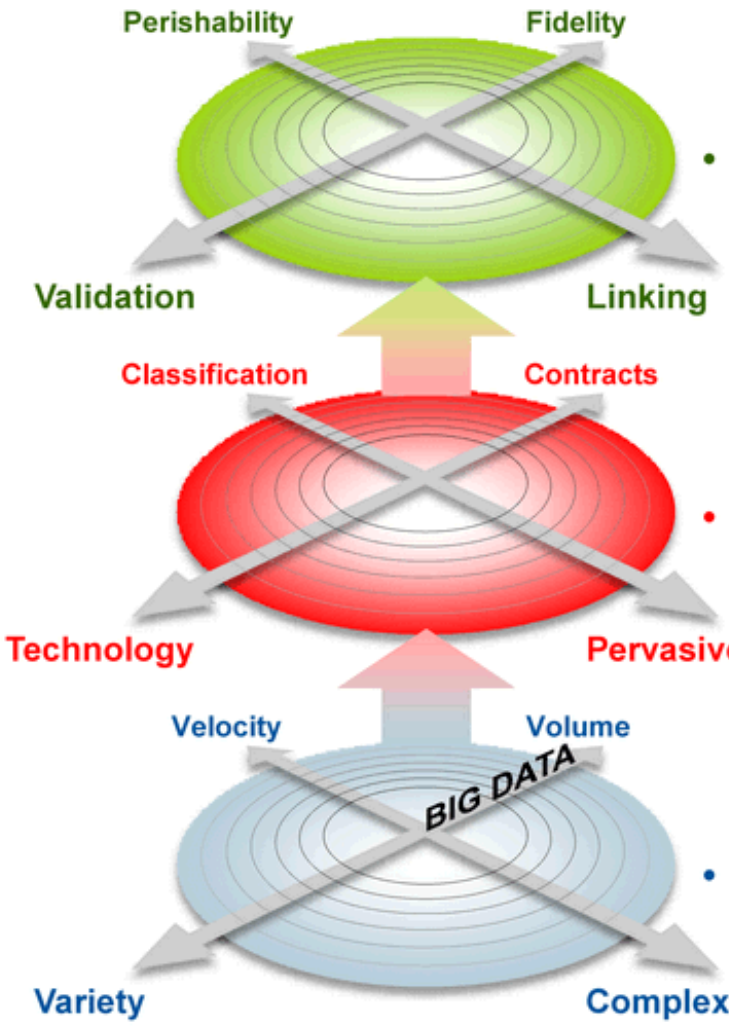
HBase
Storm



ElephantDB
Or
Voldemort

Source: Lambda Architecture, 8. March 2013
http://www.ymc.ch/en/lambda-architecture-part-1

Future: Big Data Security



品質管控

- Qualification and Assurance

權限管控

- Access Enablement and Control

數量管控

- Quantification

當我們緊密相連.....

世界政經：歐盟想分**Tweeter**
找出經濟、政治的脈動

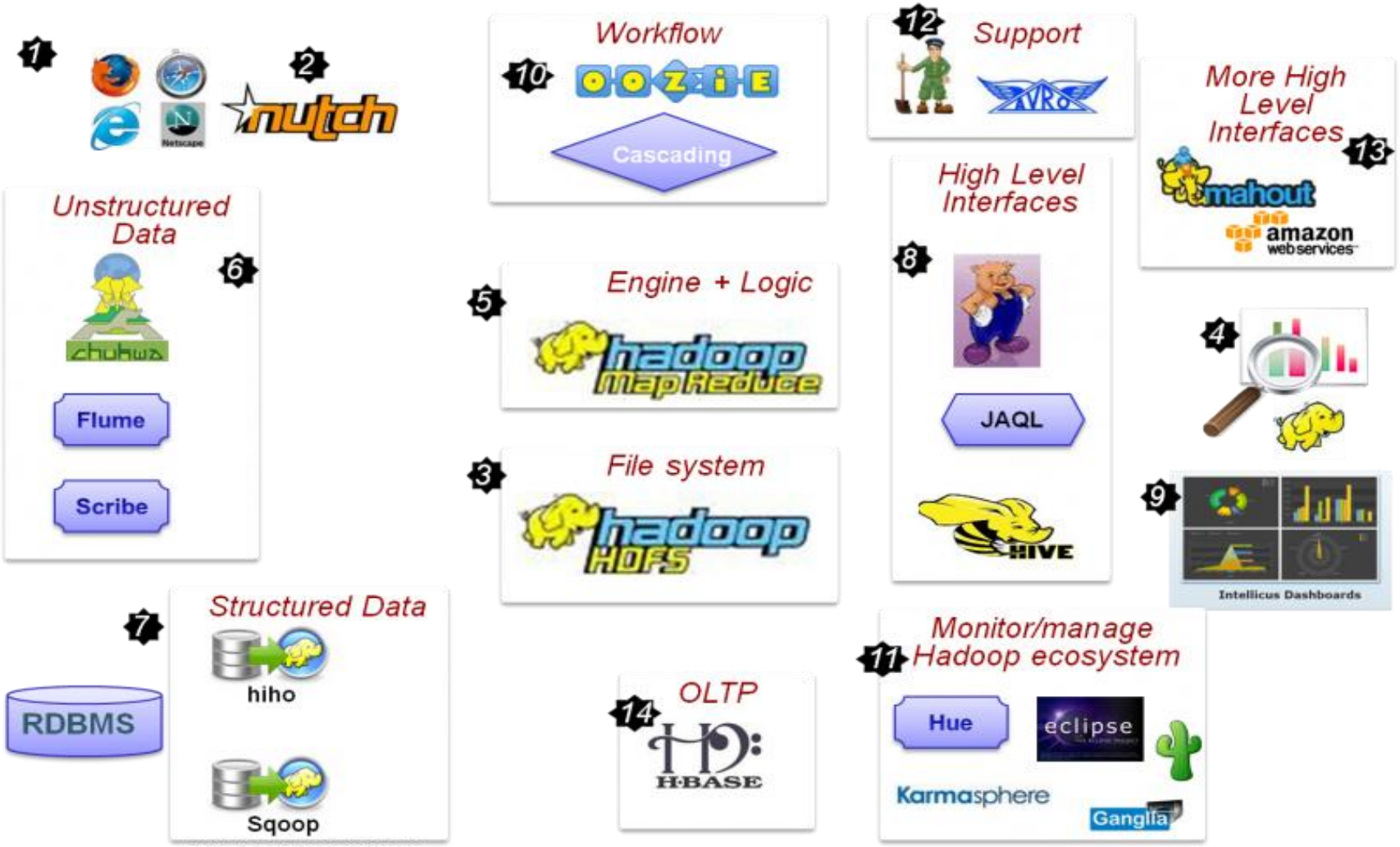
國家安全：美國**PRISM**計劃
(網軍**!**終極警探**4.0)**

組織如何因應**APT ?**
Big Data 平台本身的安全性**?**

有太多安全的問題等待解決！

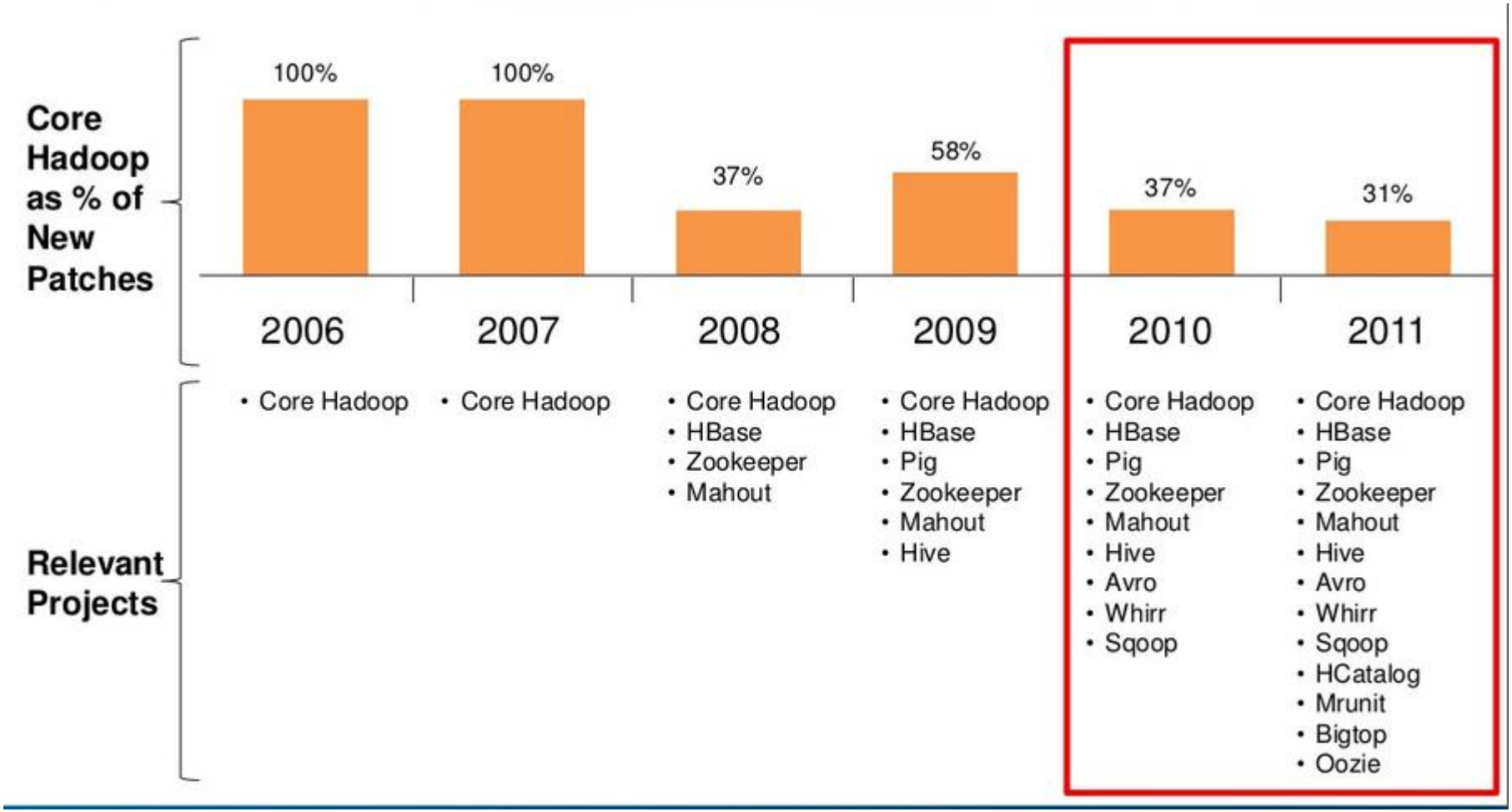
Source: Gartner (March 2011), 'Big Data' Is Only the Beginning of Extreme Information Management, April 2011, http://www.gartner.com/id=1622715

Future: Better Hadoop Ecosystem



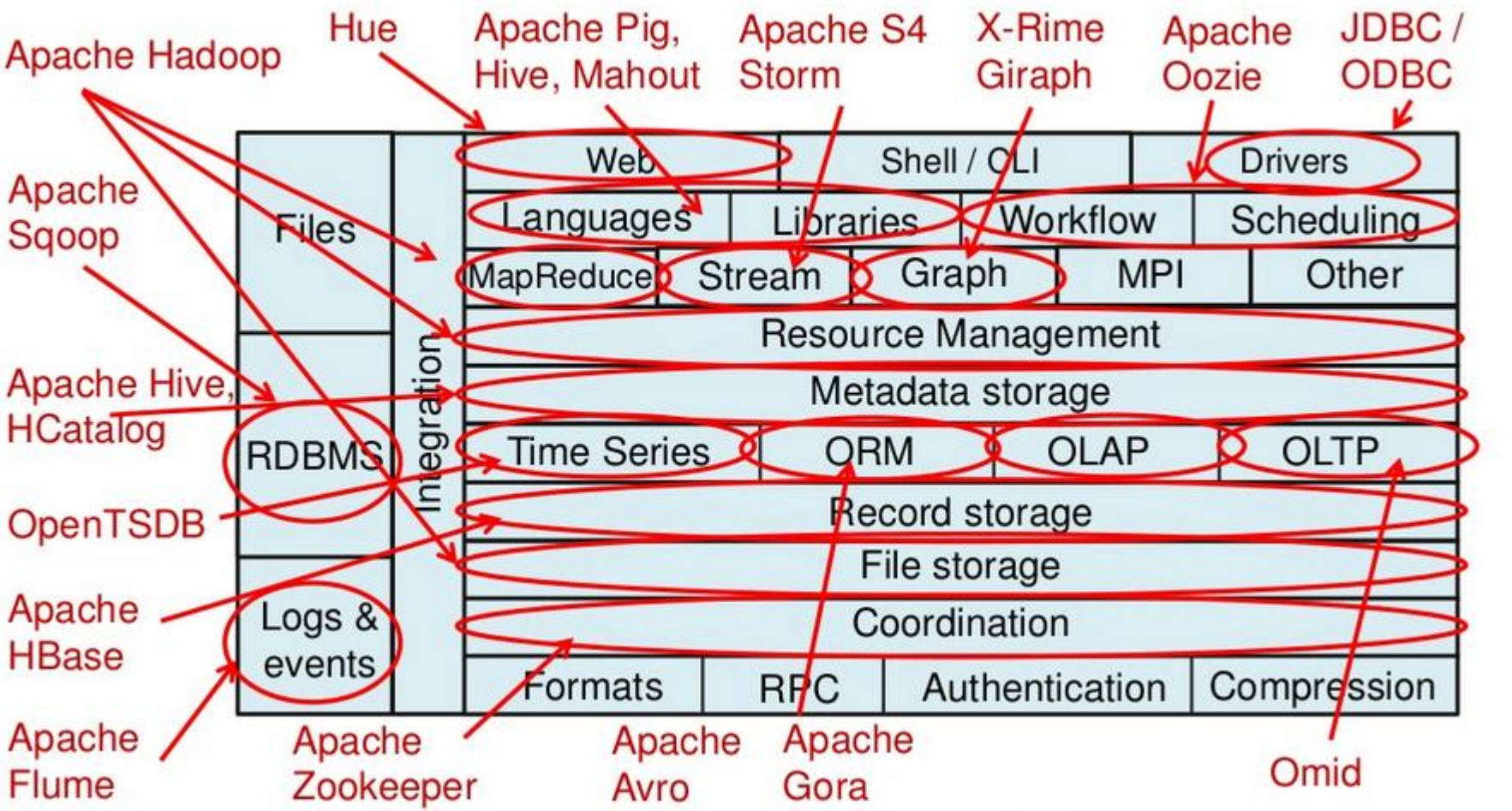
http://rationalintelligence.com/wp_log/?p=104

Evolution of Apache Hadoop Ecosystem



Hadoop World 2011: The Hadoop Stack - Then, Now and in the Future
http://www.slideshare.net/slideshow/embed_code/10110006

Complexity of Apache Big Data Stack



Hadoop World 2011: The Hadoop Stack - Then, Now and in the Future
http://www.slideshare.net/slideshow/embed_code/10110006

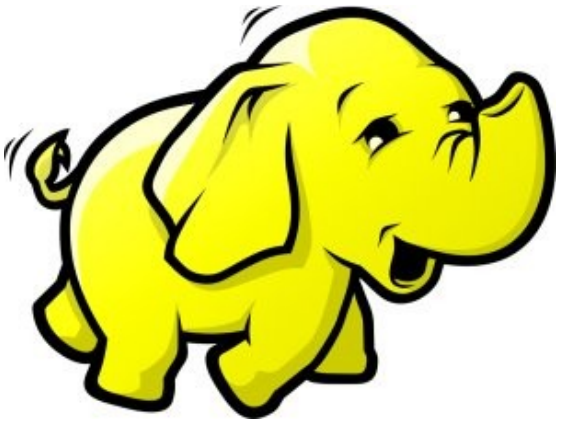
Hadoop 簡介：源起與術語

Introduction to Hadoop : History and Terminology

Jazz Wang

Yao-Tsung Wang

jazz@nchc.org.tw



What is Hadoop ?

用一句話解釋 *Hadoop* 是什麼 ??

Hadoop is a software platform that lets one easily write and run applications that process vast amounts of data.

Hadoop 是一個讓使用者簡易撰寫並執行處理海量資料應用程式的軟體平台。

亦可以想像成一個處理海量資料的生產線，只須學會定義 ***map*** 跟 ***reduce*** 工作站該做哪些事情。

Features of Hadoop …
Hadoop 這套軟體的特色是 …

- **海量 Vast Amounts of Data**
  - 擁有儲存與處理大量資料的能力
  - Capability to STORE and PROCESS vast amounts of data.
- **經濟 Cost Efficiency**
  - 可以用在由一般 PC 所架設的叢集環境內
  - Based on large clusters built of commodity hardware.
- **效率 Parallel Performance**
  - 透過分散式檔案系統的幫助，以致得到快速的回應
  - With the help of HDFS, Hadoop have better performance.
- **可靠 Robustness**
  - 當某節點發生錯誤，能即時自動取得備份資料及佈署運算資源
  - Robustness to add and remove computing and storage resource without shutdown entire system.

Founder of Hadoop – Doug Cutting

Hadoop 這套軟體的創辦人 Doug Cutting

Doug Cutting Talks About The Founding Of Hadoop

http://www.youtube.com/watch?v=qxC4urJOchs

History of Hadoop ... 2002~2004

Hadoop 這套軟體的歷史源起 ... 2002~2004

- Lucene
  - http://lucene.apache.org/
  - 用Java 設計的高效能文件索引引擎API
  - a high-performance, full-featured **text search engine library** written entirely in **Java**.
  - 索引文件中的每一字，讓搜尋的效率比傳統逐字比較還要高的多
  - Lucene create an inverse index of every word in different documents. It enhance performance of text searching.

History of Hadoop ... 2002~2004

Hadoop 這套軟體的歷史源起 ... 2002~2004

- Nutch
  - http://nutch.apache.org/
  - Nutch 是基於開放原始碼所開發的網站搜尋引擎
  - Nutch is open source web-search software.
  - 利用Lucene 函式庫開發
  - It builds on Lucene and Solr, adding web-specifics, such as a crawler, a link-graph database, parsers for HTML and other document formats, etc.

Three Gifts from Google ....

來自 Google 的三個禮物 ....

- Nutch 後來遇到儲存大量網站資料的瓶頸
- Nutch encounter storage issue
- Google 在一些會議分享他們的三大關鍵技術
- Google shared their design of web-search engine
  - SOSP 2003 : “The Google File System”
  - http://labs.google.com/papers/gfs.html
  - OSDI 2004 : “MapReduce : Simplifed Data Processing on Large Cluster”
  - http://labs.google.com/papers/mapreduce.html
  - OSDI 2006 : “Bigtable: A Distributed Storage System for Structured Data”
  - http://labs.google.com/papers/bigtable-osdi06.pdf

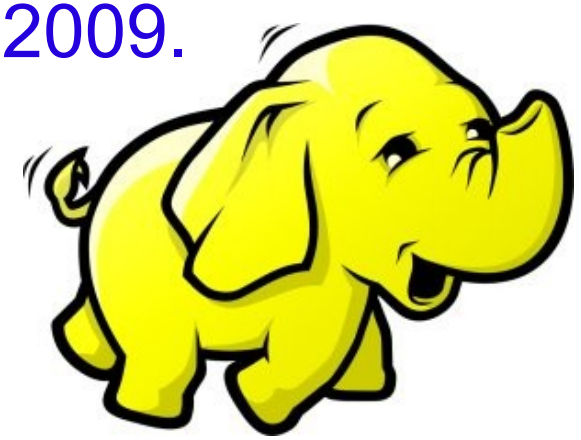
History of Hadoop … 2004 ~ Now

Hadoop 這套軟體的歷史源起 ... 2004 ~ Now

- Dong Cutting reference from Google's publication
- Added DFS & MapReduce implement to Nutch
- According to user feedback on the mail list of Nutch ....
- Hadoop became separated project since Nutch 0.8
- Nutch DFS → Hadoop Distributed File System (HDFS)
- Yahoo hire Dong Cutting to build a team of web search engine at year 2006.
  - Only 14 team members (engineers, clusters, users, etc.)
- Doung Cutting joined Cloudera at year 2009.

YAHOO!

cloudera



Who Use Hadoop ??

有哪些公司在用 Hadoop 這套軟體 ??

- Yahoo is the key contributor currently.
- IBM and Google teach Hadoop in universities …
- http://www.google.com/intl/en/press/pressrel/20071008_ibm_univ.html
- The New York Times used 100 Amazon EC2 instances and a Hadoop application to process 4TB of raw image TIFF data (stored in S3) into 11 million finished PDFs in the space of 24 hours at a computation cost of about $240 (not including bandwidth)

  – from http://en.wikipedia.org/wiki/Hadoop
- http://wiki.apache.org/hadoop/AmazonEC2
- http://wiki.apache.org/hadoop/PoweredBy

- A9.com
- ADSDAQ by Contextweb
- EHarmony
- Facebook
- Fox Interactive Media
- IBM
- ImageShack
- ISI
- Joost
- Last.fm
- Powerset
- The New York Times
- Rackspace
- Veoh
- Metaweb

Hadoop in production run ....
商業運轉中的 Hadoop 應用 ....

- February 19, 2008
- Yahoo! Launches World's Largest Hadoop Production Application
- http://developer.yahoo.net/blogs/hadoop/2008/02/yahoo-worlds-largest-production-hadoop.html

Number of links between pages in the index	roughly 1 trillion links
Size of output	over 300 TB, compressed!
Number of cores used to run single Map-Reduce job	over 10,000
Raw disk used in the production cluster	over 5 Petabytes

Hadoop in production run ....
商業運轉中的 Hadoop 應用 ....

- September 30, 2008
- Scaling Hadoop to 4000 nodes at Yahoo!
- http://developer.yahoo.net/blogs/hadoop/2008/09/scaling_hadoop_to_4000_nodes_a.html

Total Nodes	**4000**
Total cores	**30000**
Data	**16PB**

	500-node cluster		**4000-node cluster**	
	write	**read**	**write**	**read**
number of files	990	990	14,000	14,000
file size (MB)	320	320	360	360
total MB processes	316,800	316,800	5,040,000	5,040,000
tasks per node	2	2	4	4
avg. throughput (MB/s)	**5.8**	**18**	**40**	**66**

Comparison between Google and Hadoop
Google 與 Hadoop 的比較表

Develop Group	Google	Apache
Sponsor	Google	Yahoo, Amazon
Algorithm Method	MapReduce	MapReduce
Resource	open document	open source
File System (MapReduce)	GFS	HDFS
Storage System (for structure data)	big-table	HBase
Search Engine	Google	Nutch
OS	Linux	Linux / GPL

Why should we learn Hadoop ?
為何需要學習 Hadoop ??

Search Jobs | Browse Jobs | Local Jobs | Salaries | Employment Trends

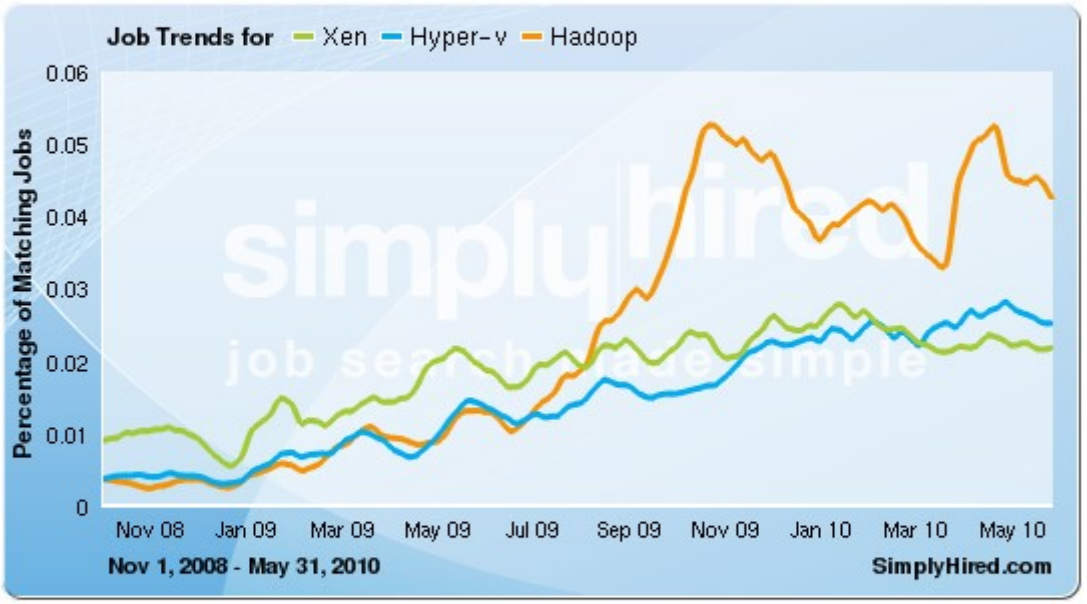
simplyhired®
job search made simple

Employment Trends

Xen, Hyper-V, Hadoop

Tip: You can compare trends by separating them with commas.

Xen, Hyper-v, Hadoop Trends



Xen, Hyper-v, Hadoop Job Trends

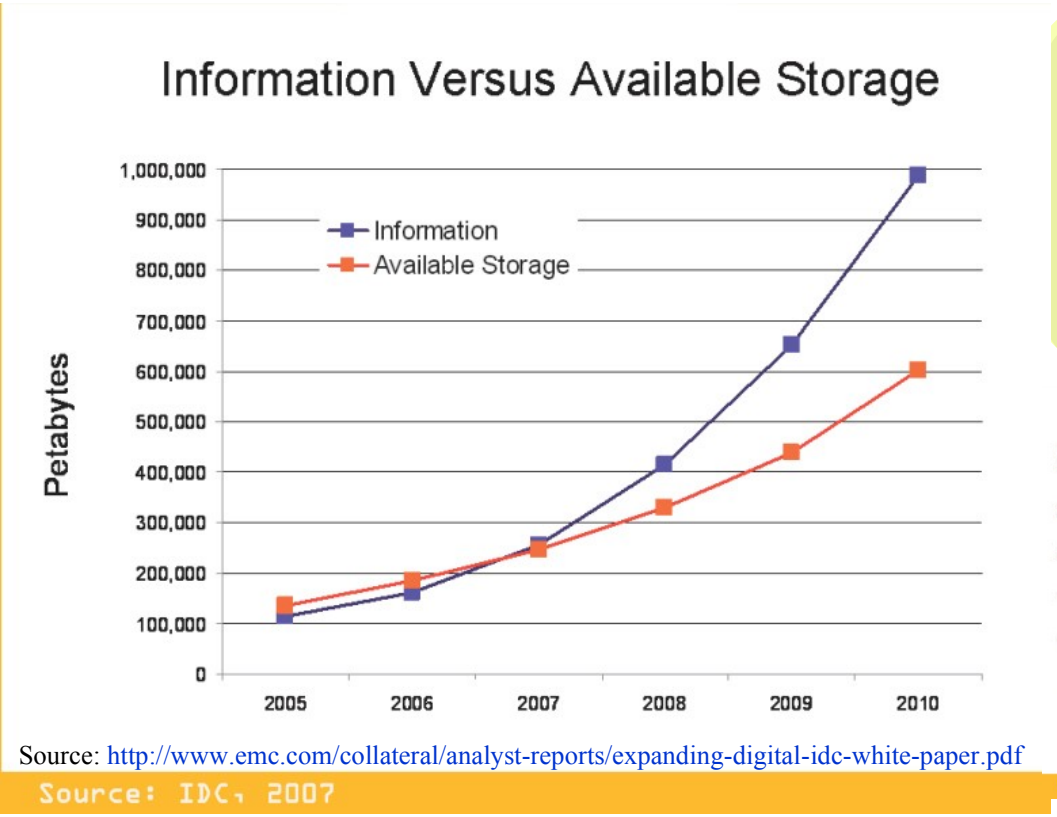
This graph displays the percentage of jobs with your search terms anywhere in the job listing. Since November 2008, the following has occurred:

- Xen jobs increased 141%
- Hyper-v jobs increased 551%
- Hadoop jobs did not change or there is no data available

1. Data Explore
資訊大爆炸

2. Data Mining Tool
方便作資料探勘的工作

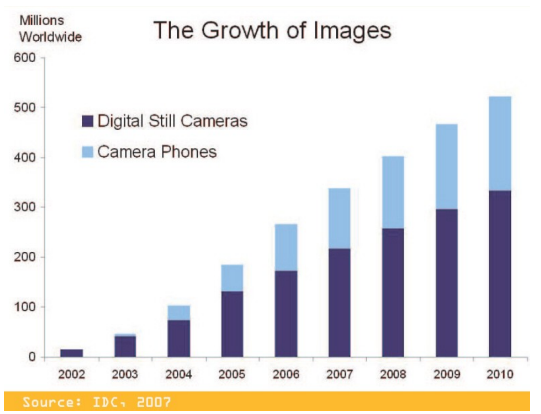
3. Looking for Jobs
好找工作 !!

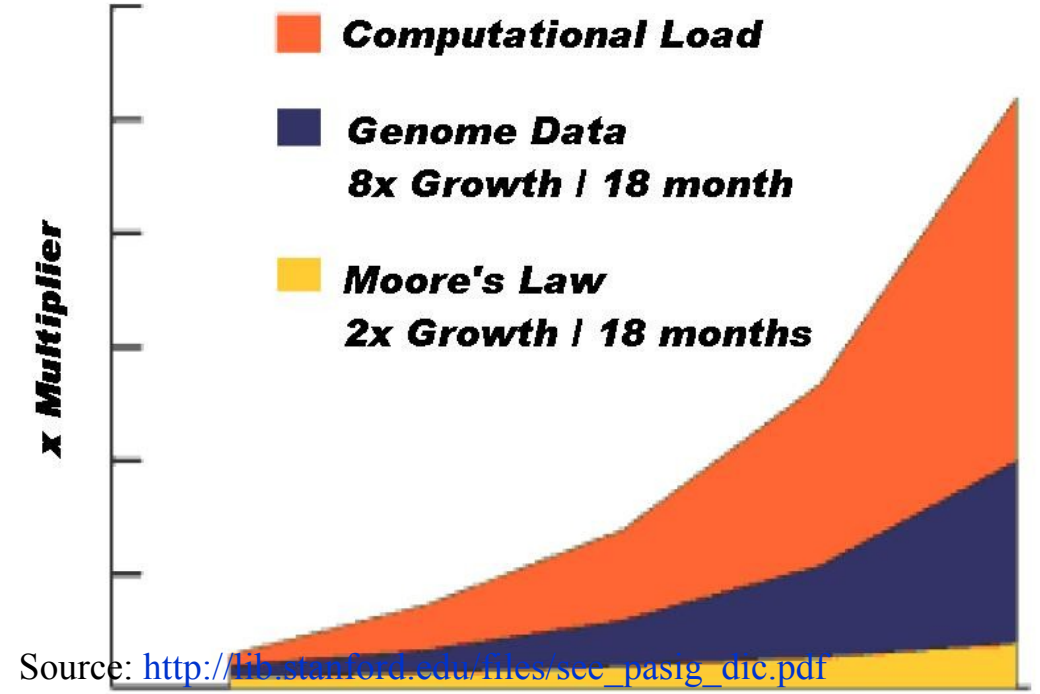


Source: http://www.emc.com/collateral/analyst-reports/expanding-digital-idc-white-paper.pdf

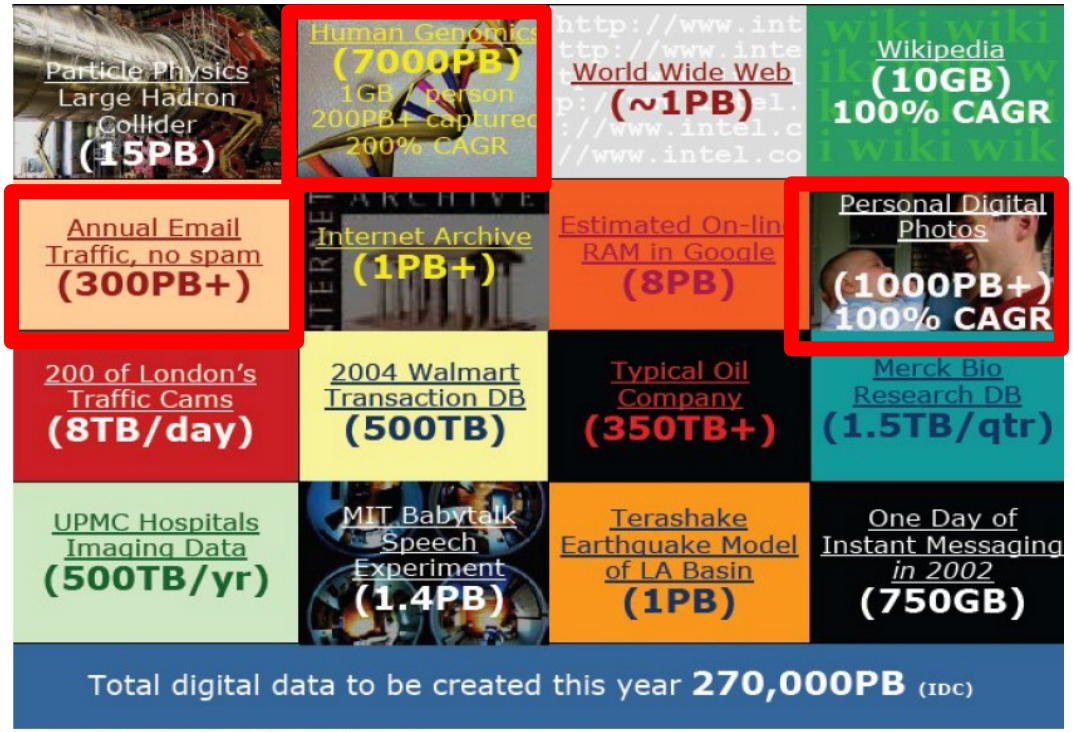
2007 Data Explore

Top 1 : Human Genomics – 7000 PB / Year
Top 2 : Digital Photos – 1000 PB+/ Year
Top 3 : E-mail (no Spam) – 300 PB+ / Year





Source: http://dlib.stanford.edu/files/see_pasig_dic.pdf



Phillip B. Gibbons, Data-Intensive Computing Symposium

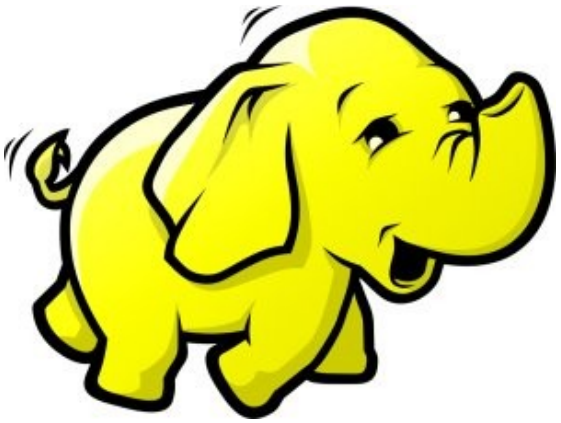
Hadoop 專業術語

Introduction to Hadoop Terminology

Jazz Wang

Yao-Tsung Wang

jazz@nchc.org.tw



Two Key Elements of Operating System
作業系統兩大關鍵組成元素

Scheduler
程序排程

File System
檔案系統

Terminologies of Hadoop

Hadoop 文件中的專業術語

- Job
  - 任務
- Task
  - 小工作
- JobTracker
  - 任務分派者
- TaskTracker
  - 小工作的執行者
- Client
  - 發起任務的客戶端
- Map
  - 應對
- Reduce
  - 總和

- Namenode
  - 名稱節點
- Datanode
  - 資料節點
- Namespace
  - 名稱空間
- Replication
  - 副本
- Blocks
  - 檔案區塊 (64M)
- Metadata
  - 屬性資料

Two Key Roles of HDFS
HDFS 軟體架構的兩種關鍵角色

名稱節點 NameNode

- Master Node
- Manage NameSpace of HDFS
- Control Permission of Read and Write
- Define the policy of Replication
- Audit and Record the NameSpace
- Single Point of Failure

資料節點 DataNode

- Worker Nodes
- Perform operation of Read and Write
- Execute the request of Replication
- Multiple Nodes

Two Key Roles of Job Scheduler
程序排程的兩種關鍵角色

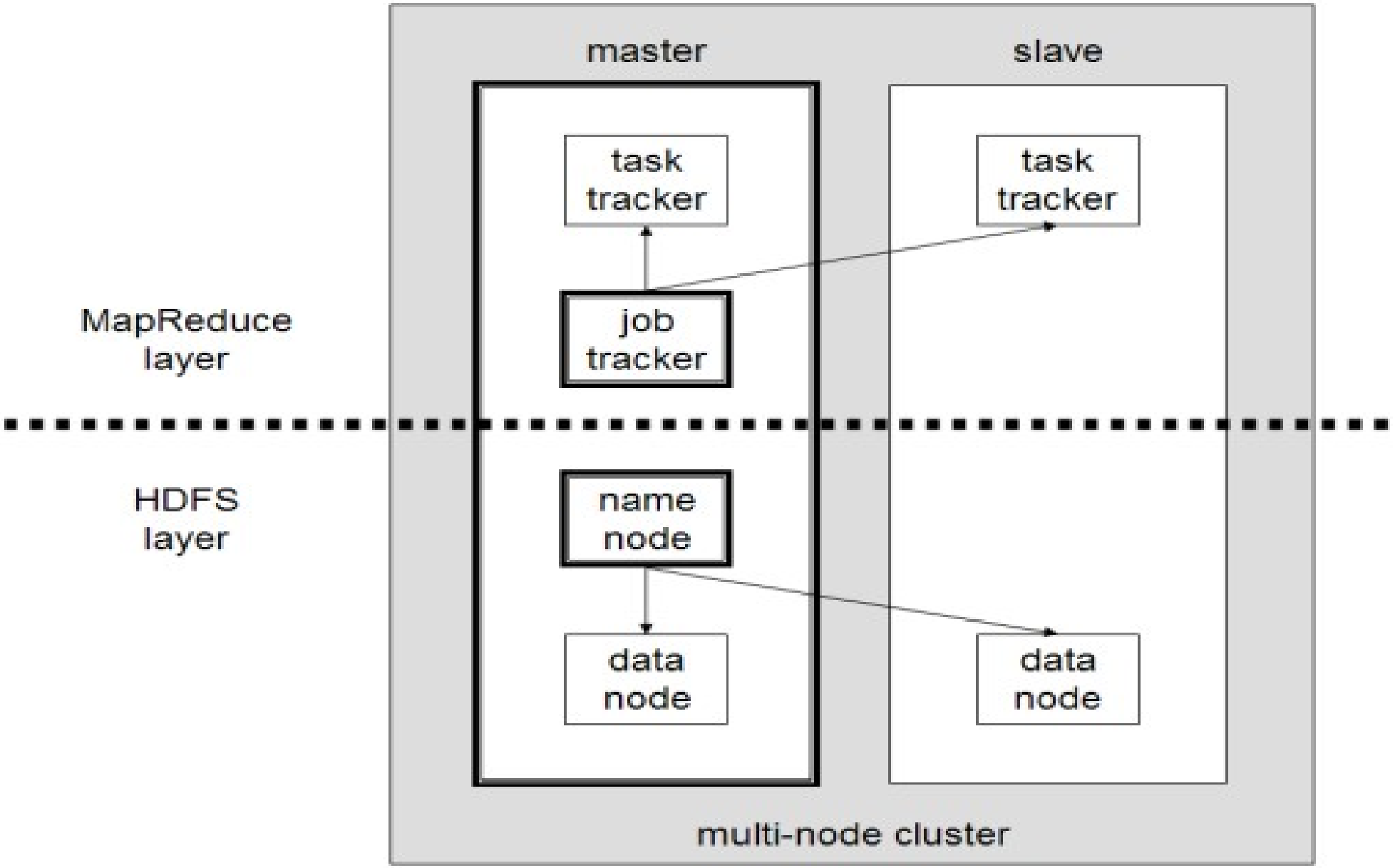
JobTracker

- **Master Node**
- **Receive Jobs from Hadoop Clients**
- **Assigned Tasks to TaskTrackers**
- **Define Job Queuing Policy, Priority and Error Handling**
- **Single Point of Failure**

TaskTracker

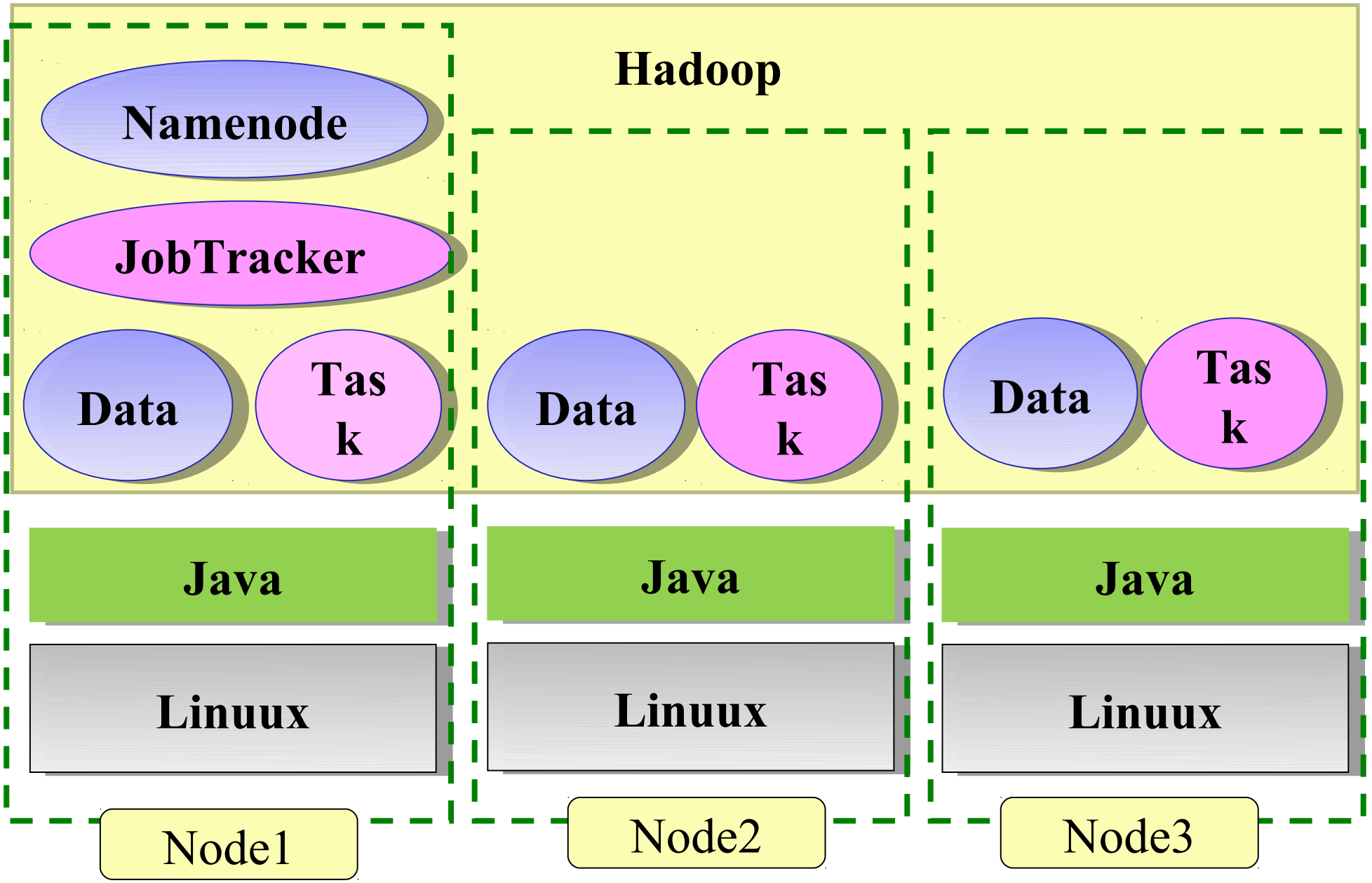
- **Worker Nodes**
- **Excute Mapper and Reducer Tasks**
- **Save Results and report task status**
- **Multiple Nodes**

Different Roles of Hadoop Architecture
Hadoop 軟體架構中的不同角色

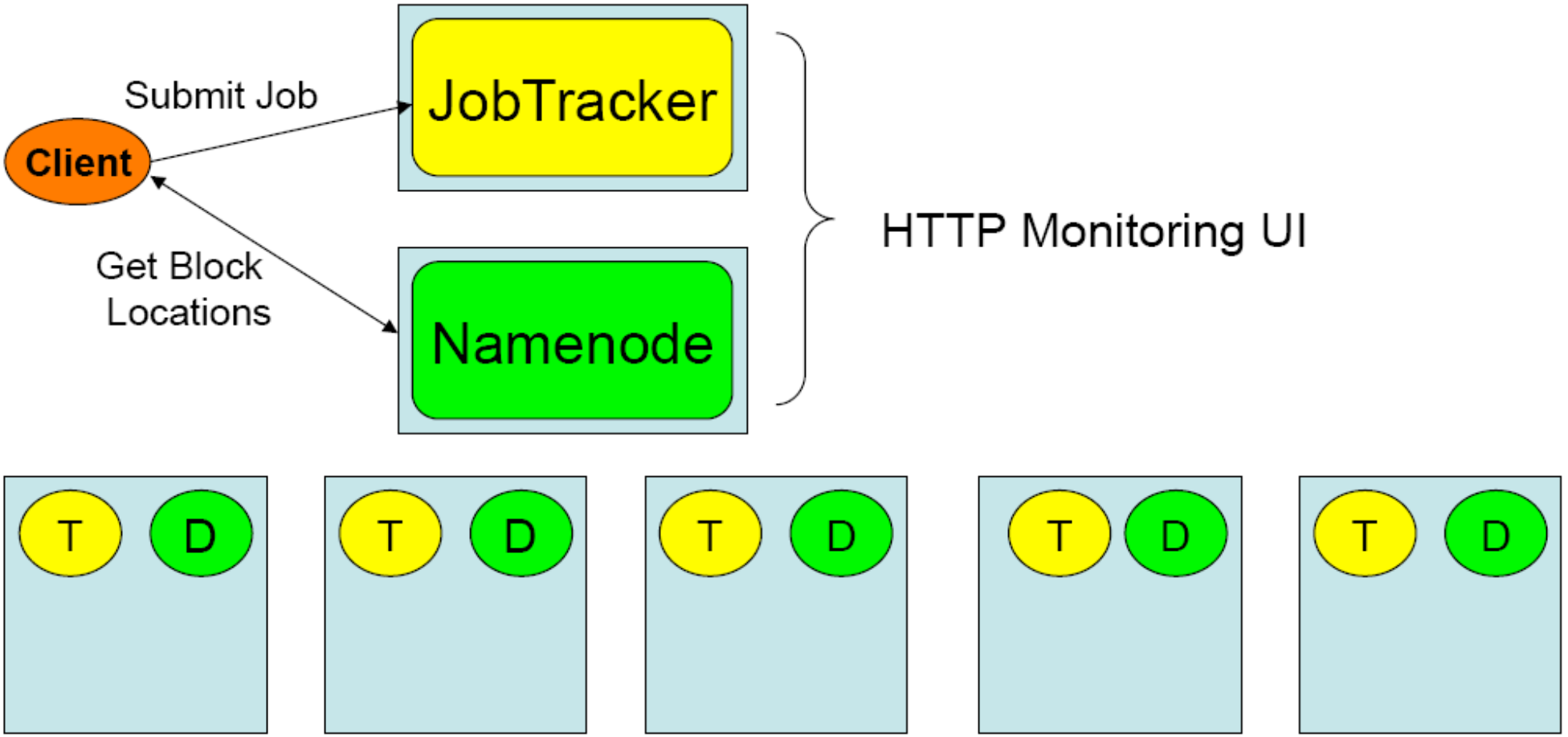


Distributed Operating System of Hadoop

Hadoop 建構成一個分散式作業系統



About Hadoop Client ...
不在雲裡的 Hadoop Client



What we learn today ?

WHAT	Hadoop 是運算海量資料的軟體平台 !! hadoop is a software platform to process vast amount of data!!
WHO	始祖是 Doug Cutting ， Apache 社群支持， Yahoo 贊助 From Doug Cutting to Apache Community, Yahoo and more !
WHEN	Hadoop 是 2004 年從 Nutch 分裂出來的專案 !! Hadoop became separate project since year 2004 !!
WHY	資料大爆炸、資料探勘、找工作 Data Explore, Data Mining, Jobs !!
HOW	建構在大型的個人電腦叢集之上 Install on large clusters built of commodity hardware !!

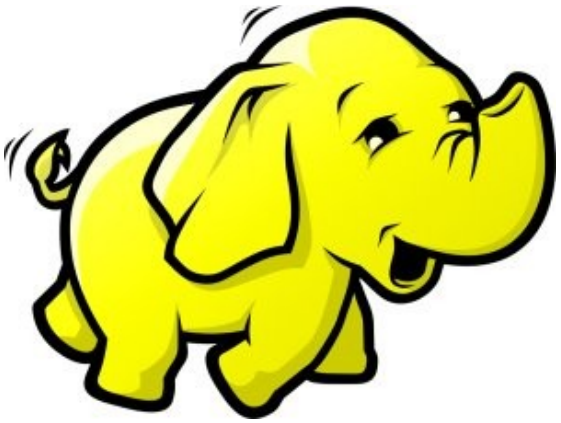
HDFS 簡介

Introduction to Hadoop Distributed File System

Jazz Wang

Yao-Tsung Wang

jazz@nchc.org.tw



What is HDFS ??
什麼是 HDFS ??

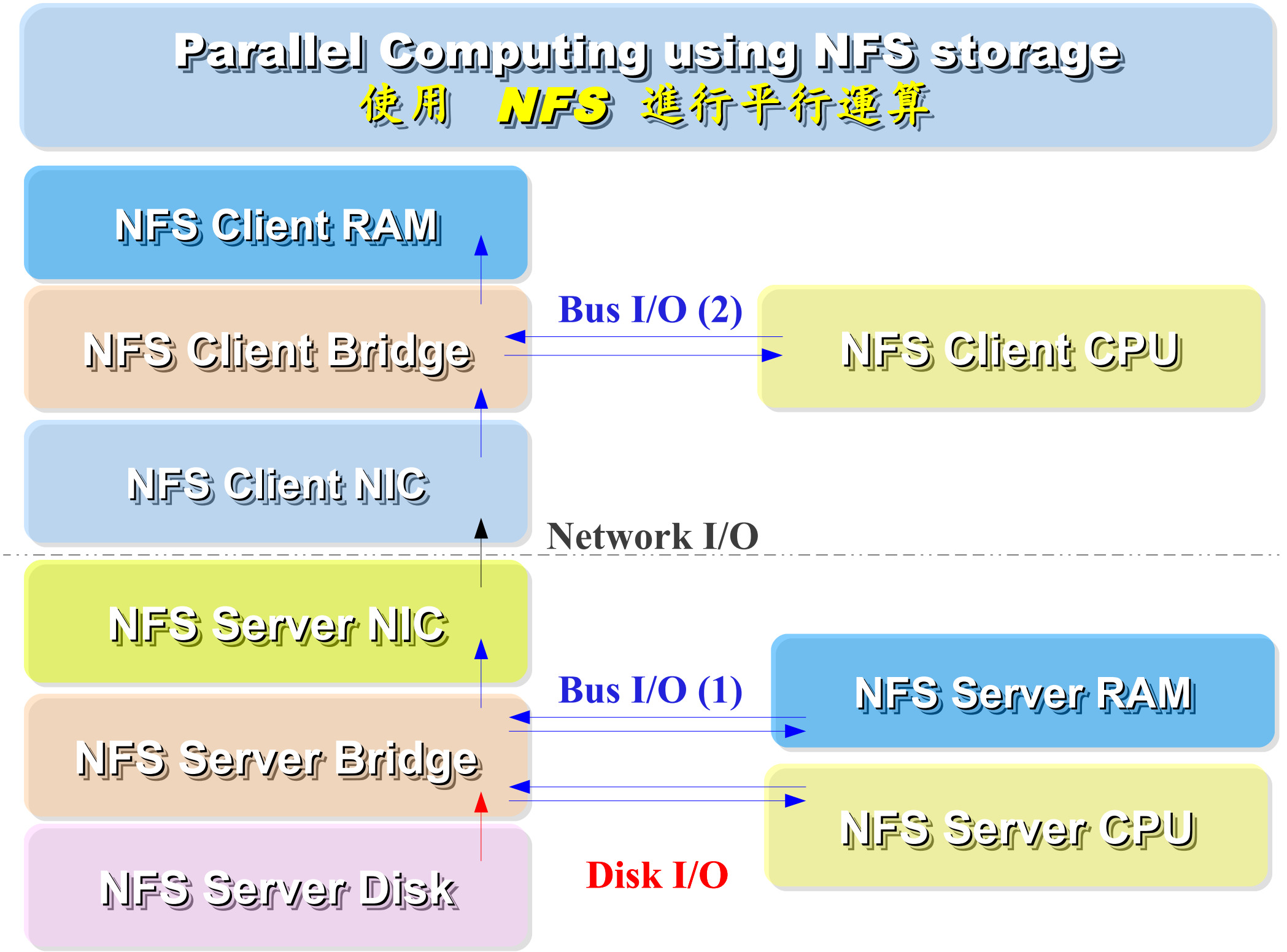
- **Hadoop Distributed File System**
  - 實現類似 Google File System 分散式檔案系統
  - Reference from Google File System.
  - 一個易於擴充的分散式檔案系統，目的爲對大量資料進行分析
  - A scalable distributed file system for large data analysis .
  - 運作於廉價的普通硬體上，又可以提供容錯功能
  - based on commodity hardware with high fault-tolerant.
  - 給大量的用户提供總體性能較高的服務
  - It have better overall performance to serve large amount of users.

Features of HDFS ...
HDFS 的特色是 ...

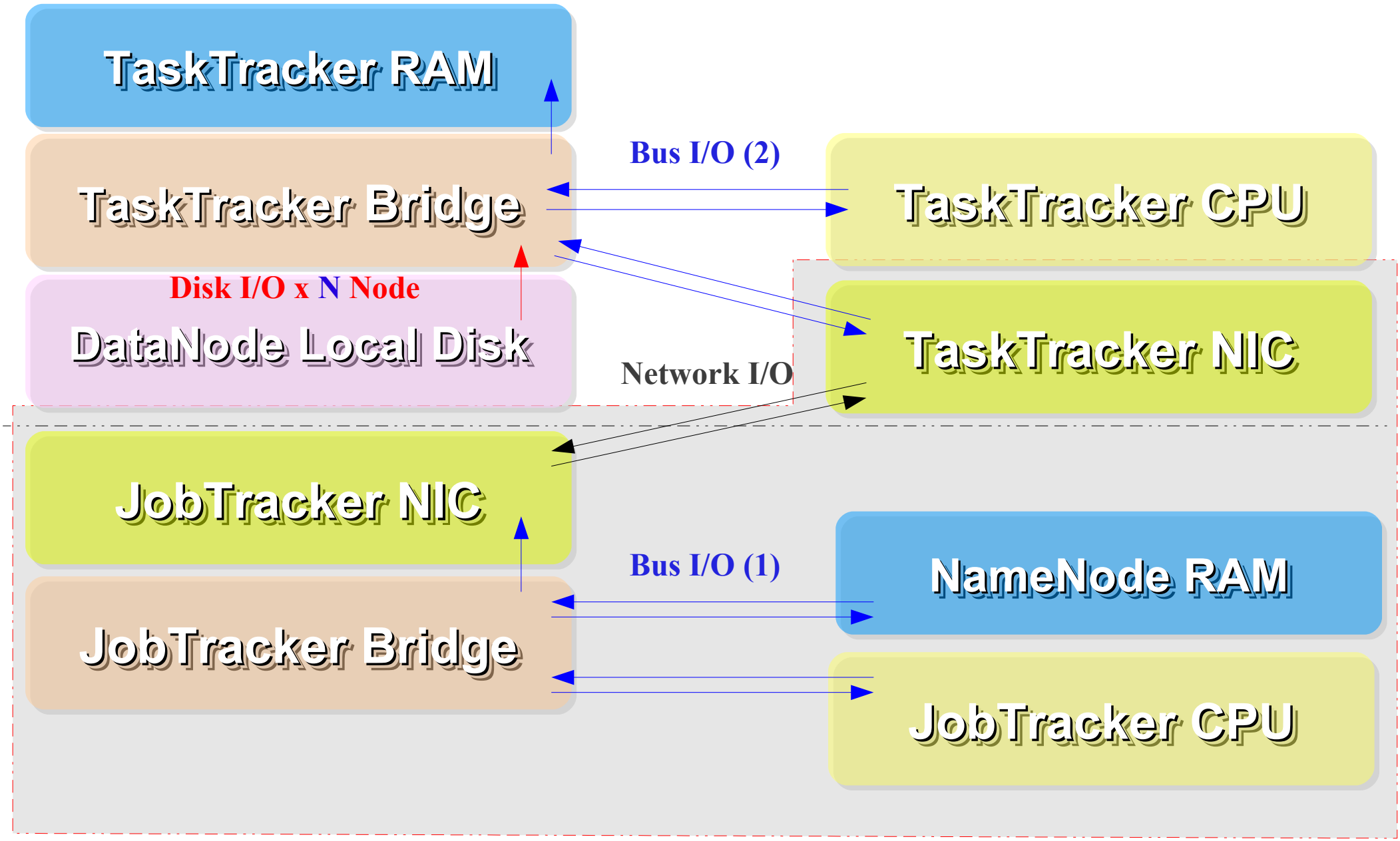
- **硬體錯誤容忍能力 Fault Tolerance**
  - 硬體錯誤是正常而非異常
  - Failure is the norm rather than exception
  - 自動恢復或故障排除
  - automatic recovery or report failure
- **串流式的資料存取 Streaming data access**
  - 批次處理多於用户交互處理
  - Batch processing rather than interactive user access.
  - 高 Throughput 而非低 Latency
  - High aggregate data bandwidth (throughput)

Features of HDFS ...
HDFS 的特色是 ...

- **大規模資料集 Large data sets and files**
  - 支援 Petabytes 等級的磁碟空間
  - Support Petabytes size
- **一致性模型 Coherency Model**
  - 一次寫入，多次存取 Write-once-read-many
  - 簡化一致性處理問題 This assumption simplifies coherency
- **在地運算 Data Locality**
  - 到資料的節點上計算 > 將資料從遠端複製過來計算
  - “move compute to data” > “move data to compute”
- **異質平台移植性 Heterogeneous**
  - 即使硬體不同也可移植、擴充
  - HDFS could be deployed on different hardware

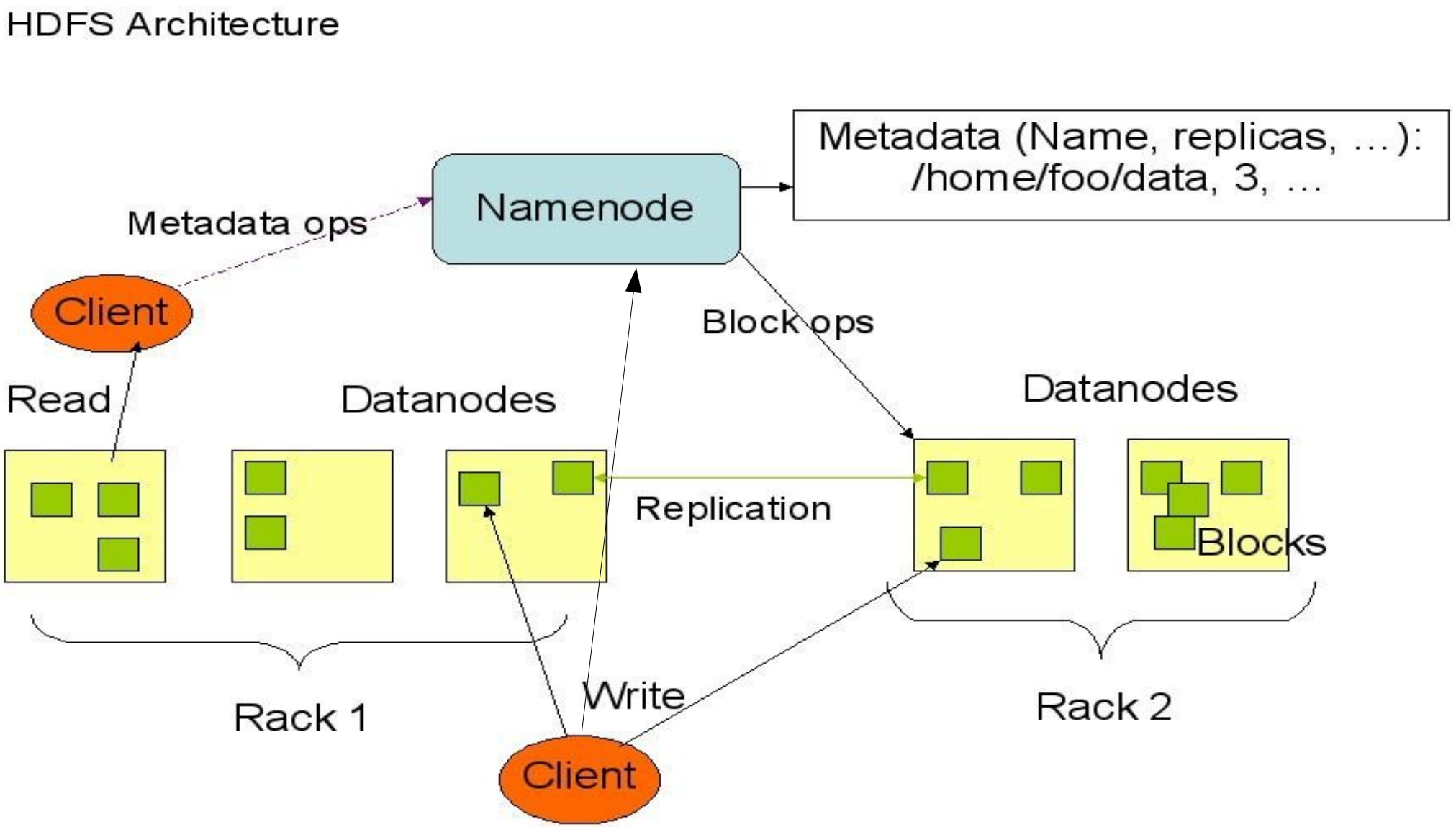

Parallel Computing using NFS storage
使用 NFS 進行平行運算
NFS Client RAM
NFS Client Bridge
NFS Client NIC
Bus I/O (2)
NFS Client CPU
Network I/O
NFS Server NIC
Bus I/O (1)
NFS Server RAM
NFS Server Bridge
NFS Server CPU
NFS Server Disk
Disk I/O

Parallel Computing using HDFS
使用 HDFS 進行平行運算



How HDFS manage data ...

HDFS 如何管理資料 ...



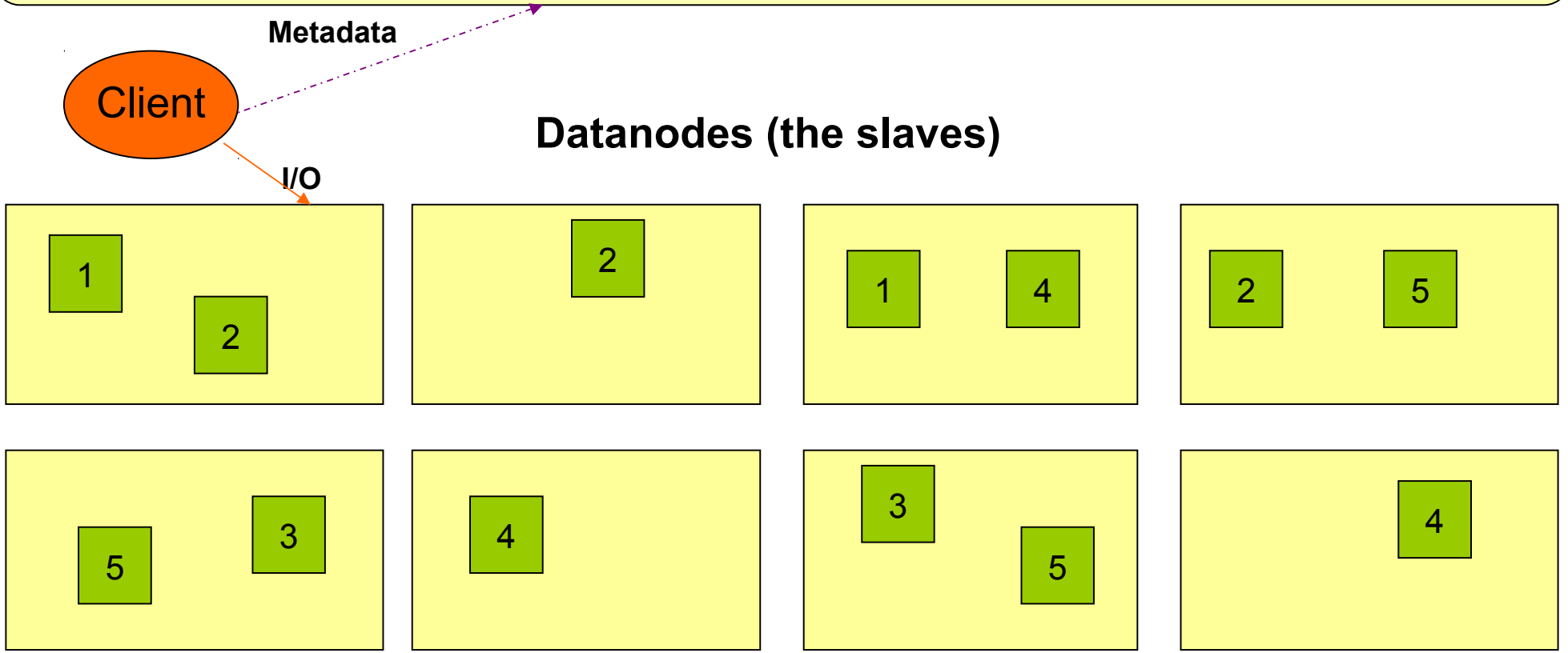
How does HDFS work ...

HDFS 如何運作 ...

Namenode (the master)

Path and Filename – Replication , blocks

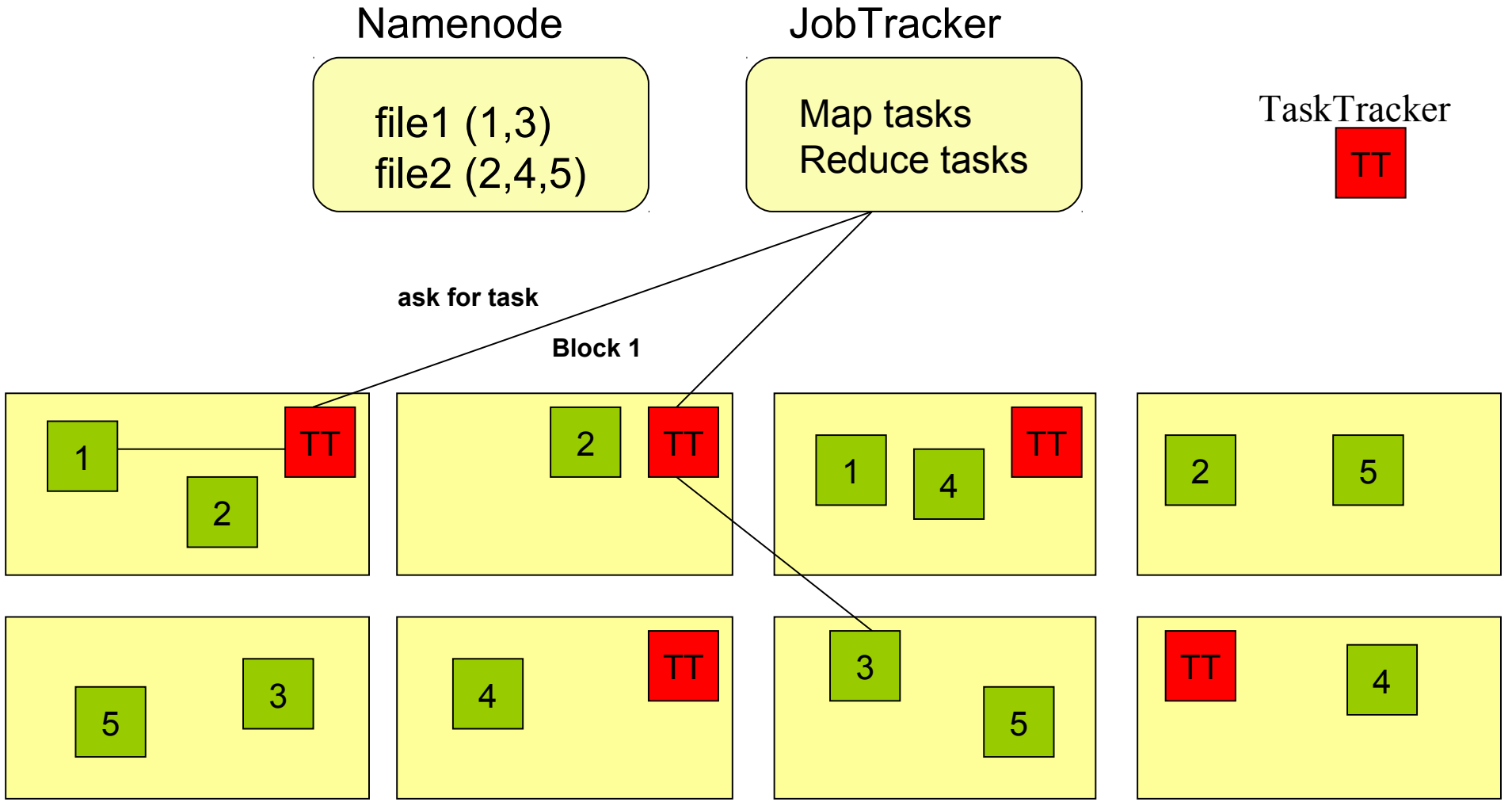
name:/users/joeYahoo/myFile - copies:2, blocks:{1,3}
name:/users/bobYahoo/someData.gzip, copies:3, blocks:{2,4,5}



About Data locality ...

HDFS 如何達成在地運算 ...

- Increase reliability and read bandwidth
  - robustness ： read replication while found any failure
  - High read bandwith ： distribute read （ but increase write bottlenet ）



9

About Fault Tolerance ...

HDFS 如何達成容錯機制 ...

資料崩毀
Data Corrupt

網路或資料節點失效
Network Fault
DataNode Fault

名稱節點錯誤
NameNode Fault

- 資料完整性 Data integrity
  - checked with CRC32
  - 用副本取代出錯資料
  - Replcae corrupt block with replication one
- Heartbeat
  - Datanode send heartbeat to Namenode
- Metadata
  - FSImage、Editlog 爲核心印象檔及日誌檔
  - FSImage – core file system mapping image
  - Editlog – like. SQL transaction log
  - 多份儲存，當名稱節點故障時可以手動復原
  - Multiple backups of FSImage and Editlog
  - Manually recovery while NameNode Fault

10

Coherency Model and Performance of HDFS
HDFS 的一致性機制與效能 ...

- **檔案一致性機制 Coherency model of files**
  - 刪除檔案＼新增寫入檔案＼讀取檔案皆由名稱節點負責
  - NameNode handle the operation of write, read and delete.
- **巨量空間及效能機制 Large Data Set and Performance**
  - 預設每個區塊大小以 64MB 爲單位
  - By default, the block size is 64MB
  - 大區塊可提高存取效率
  - Bigger block size will enhance read performance
  - 檔案有可能大過一顆磁碟
  - Single file stored on HDFS might be larger than single physical disk of DataNode.
  - 區塊均勻散佈各節點以分散讀取流量
  - Fully distributed blocks increase throughput of reading.

POSIX like HDFS commands

與 *POSIX* 相似的操作指令 ...

```
jazz@hadoop:~$ hadoop fs
Usage: java FsShell
           [-ls <path>]
           [-lsr <path>]
           [-du <path>]
           [-dus <path>]
           [-count[-q] <path>]
           [-mv <src> <dst>]
           [-cp <src> <dst>]
           [-rm <path>]
           [-rmr <path>]
           [-expunge]
           [-put <localsrc> ... <dst>]
           [-copyFromLocal <localsrc> ... <dst>]
           [-moveFromLocal <localsrc> ... <dst>]
           [-get [-ignoreCrc] [-crc] <src> <localdst>]
           [-getmerge <src> <localdst> [addnl]]
           [-cat <src>]
           [-text <src>]
           [-copyToLocal [-ignoreCrc] [-crc] <src> <localdst>]
           [-moveToLocal [-crc] <src> <localdst>]
           [-mkdir <path>]
           [-setrep [-R] [-w] <rep> <path/file>]
           [-touchz <path>]
           [-test -[ezd] <path>]
           [-stat [format] <path>]
           [-tail [-f] <file>]
           [-chmod [-R] <MODE[,MODE]... | OCTALMODE> PATH...]
           [-chown [-R] [OWNER][:[GROUP]] PATH...]
           [-chgrp [-R] GROUP PATH...]
           [-help [cmd]]
```

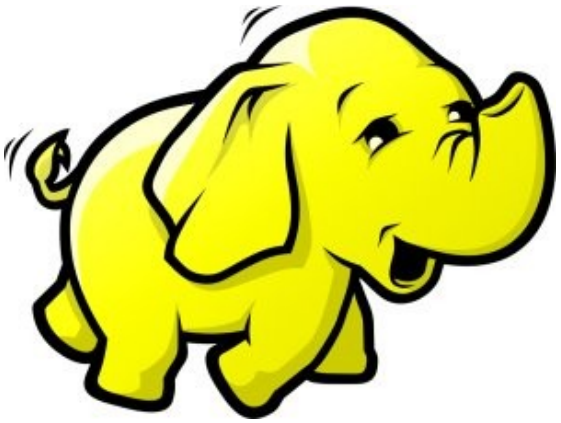
MapReduce 簡介

Introduction to MapReduce

Jazz Wang

Yao-Tsung Wang

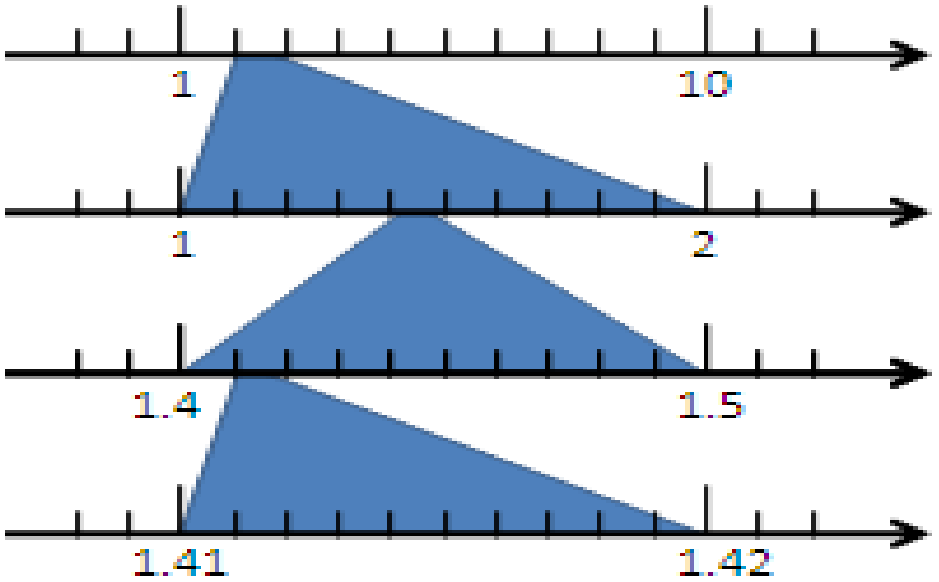
jazz@nchc.org.tw



Divide and Conquer Algorithms

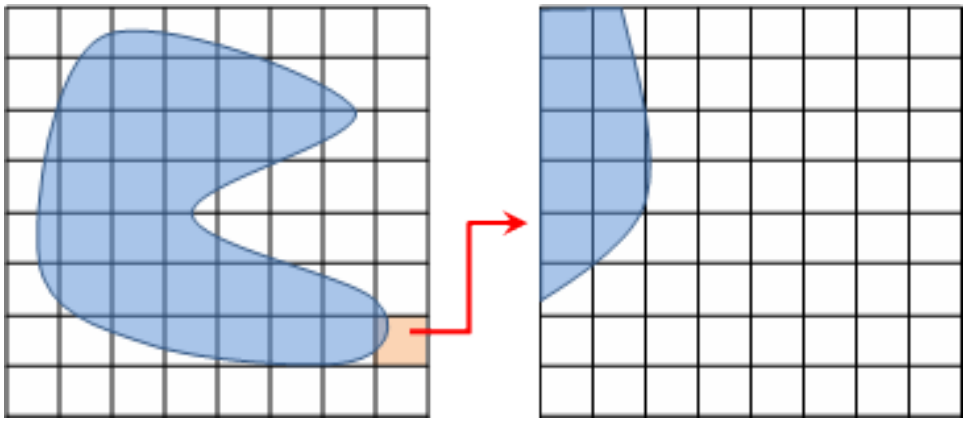
分而治之演算法

Example 1:

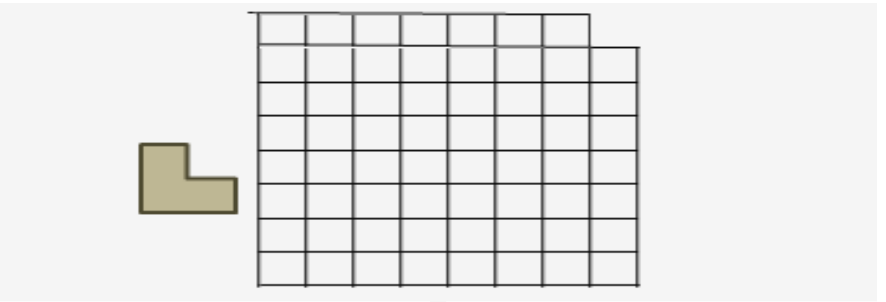


sqrt(2)

Example 2:



Example 3:



Example 4: The way to climb 5 steps stair within 2 steps each time. 眼前有五階樓梯，每次可踏上一階或踏上兩階，那麼爬完五階共有幾種踏法？

Ex : (1,1,1,1,1) or (1,2,1,1)

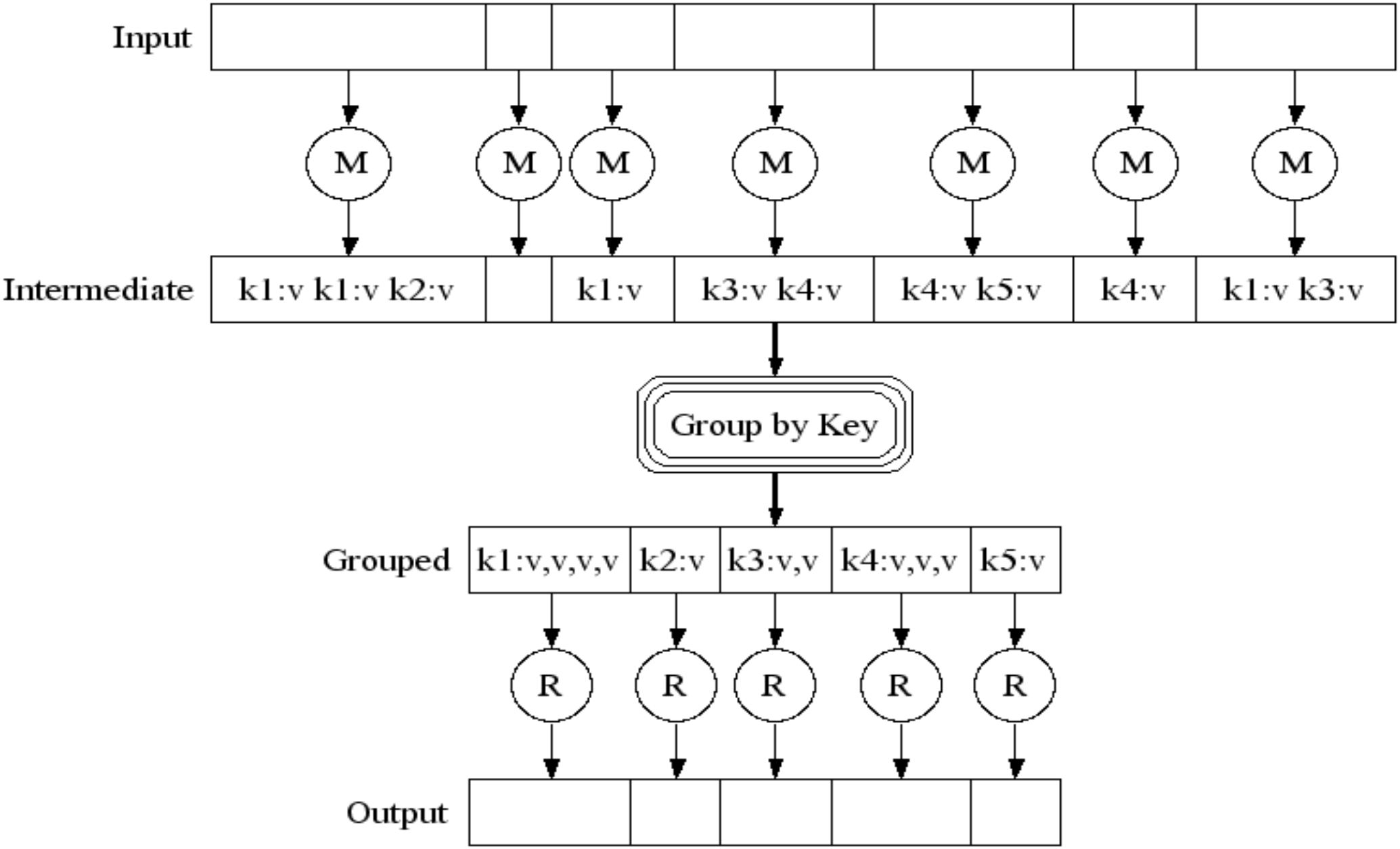
What is MapReduce ??
什麼是 MapReduce ??

- MapReduce 是 Google 申請的軟體專利，主要用來處理大量資料
- **MapReduce is a patented software framework introduced by Google to support distributed computing on large data sets on clusters of computers.**
- 啓發自函數編程中常用的 map 與 reduce 函數。
- **The framework is inspired by map and reduce functions commonly used in functional programming, although their purpose in the MapReduce framework is not the same as their original forms**
  - Map(...) : N → N
    - Ex. [1,2,3,4] – (*2) -> [2,4,6,8]
  - Reduce(...): N → 1
    - [1,2,3,4] - (sum) -> 10
- **Logical view of MapReduce**
    - Map(k1, v1) -> list(k2, v2)
    - Reduce(k2, list (v2)) -> list(k3, v3)

Source: http://en.wikipedia.org/wiki/MapReduce

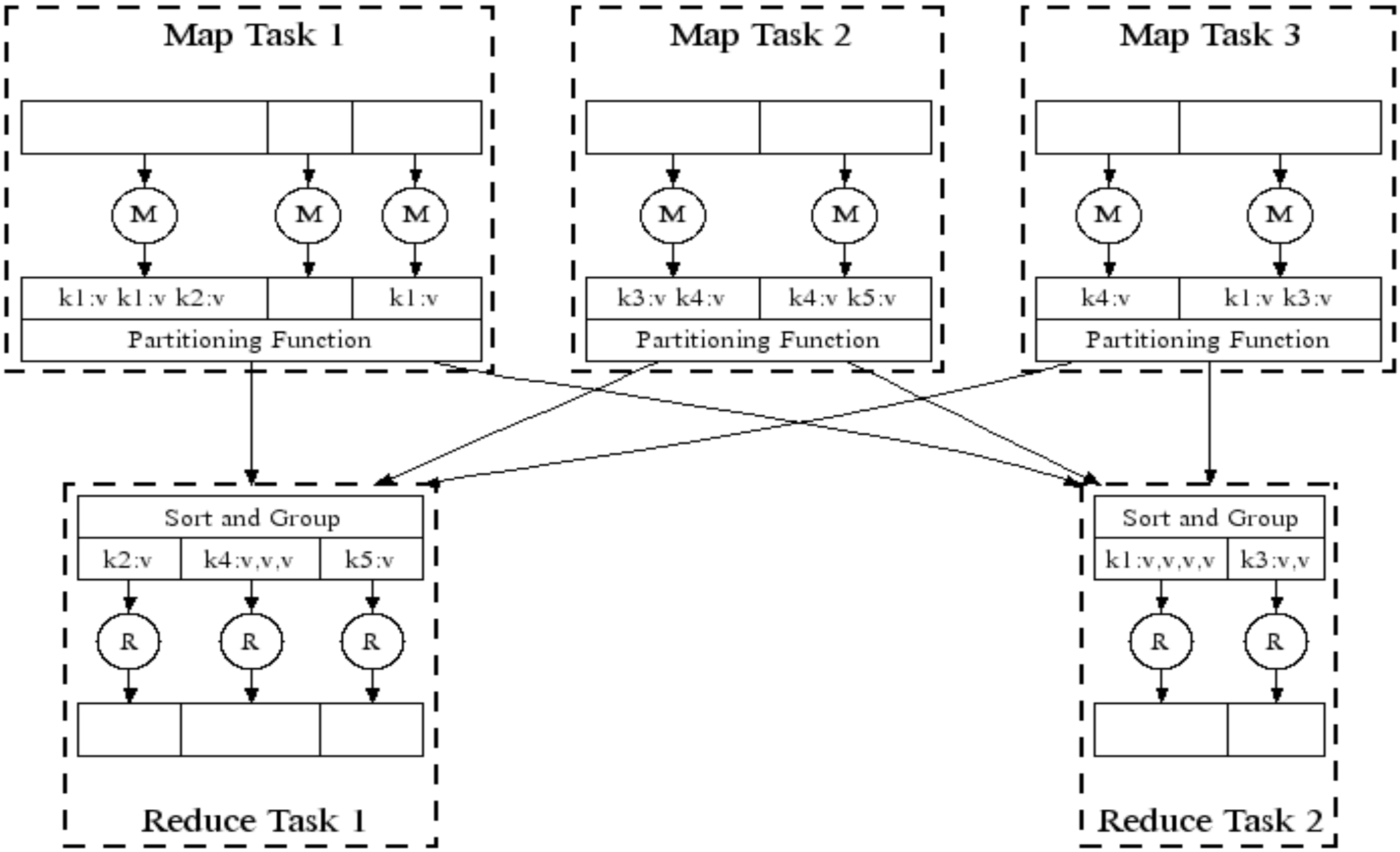
Google's MapReduce Diagram

Google 的 MapReduce 圖解



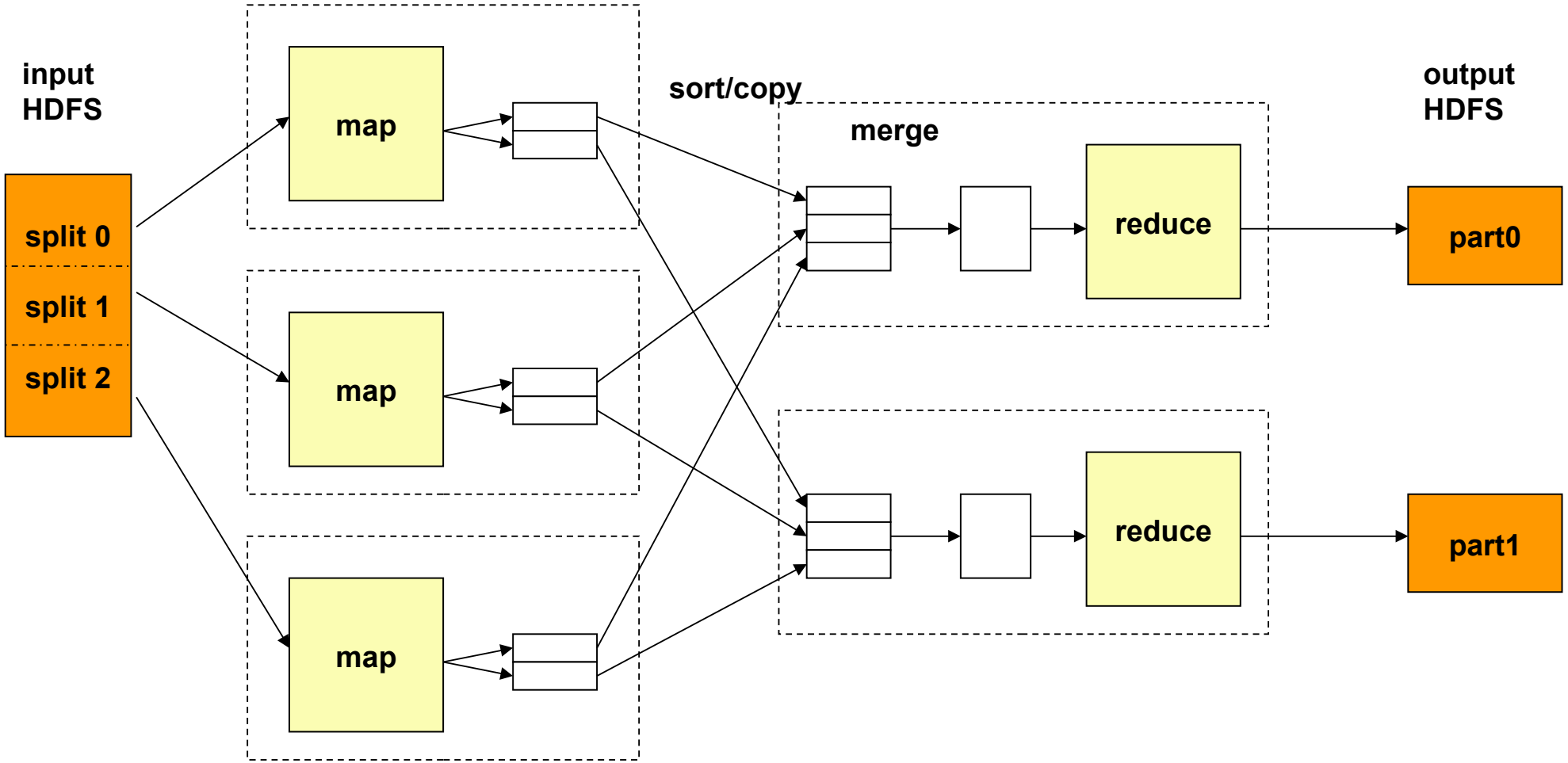
Google's MapReduce in Parallel

Google 的 MapReduce 平行版圖解



How does MapReduce work in Hadoop

Hadoop MapReduce 運作流程



JobTracker 跟 NameNode 取得需要運算的 blocks

JobTracker 選數個 TaskTracker 來作 Map 運算，產生些中間檔案

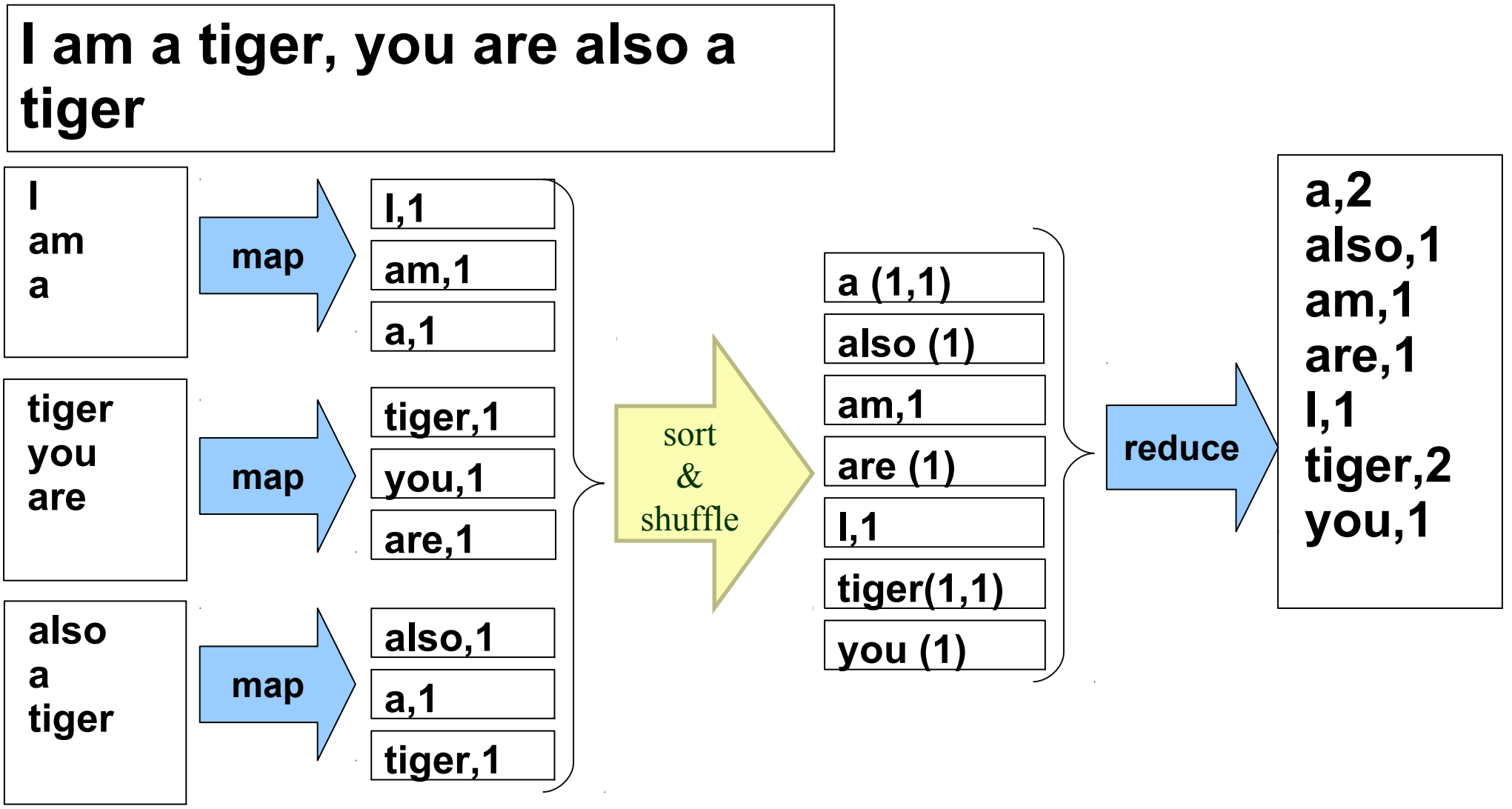
JobTracker 將中間檔案整合排序後，複製到需要的 TaskTracker 去

JobTracker 派遣 TaskTracker 作 reduce

reduce 完後通知 JobTracker 與 Namenode 以產生 output

MapReduce by Example (1)

MapReduce 運作實例 (1)



JobTracker 先選了三個 Tracker 做 map

Map 結束後，hadoop 進行中間資料的重組與排序

JobTracker 再選一個 TaskTracker 作 reduce

MapReduce by Example (2)

MapReduce 運作實例 (2)

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \Rightarrow \begin{bmatrix} \text{sqrt}(a+b) \\ \text{sqrt}(c+d) \end{bmatrix}$$

$$\begin{bmatrix} 1.0 & 0.0 & 3.0 \\ 3.2 & 0.8 & 32.0 \\ 1.0 & 14.0 & 1.0 \end{bmatrix} \Rightarrow ?$$

```
(0,sqrt(1.0 +  0.0 +  3.0))
(1,sqrt(3.2 +  0.8 + 32.0))
(2,sqrt(1.0 + 14.0 +  1.0))
```

Input File

```
0 0 1.0  // A[0][1] = 1.0
0 1 0.0  // A[0][1] = 0.0
0 2 3.0  // A[0][2] = 3.0
1 0 3.2  // A[1][0] = 3.2
1 1 0.8  // A[1][1] = 0.8
```

```
1 2 32.0 // A[1][2] = 32.0
2 0 1.0  // A[2][0] = 1.0
2 1 14.0 // A[2][1] = 14.0
2 2 1.0  // A[2][2] = 1.0
```

map

```
(0,1.0)
(0,0.0)
(0,3.0)
(1,3.2)
(1,0.8)
```

map

```
(1,32.0)
(2,1.0)
(2,14.0)
(2,1.0)
```

sort / merge

```
(0,{1.0,0.0,3.0})
(1,{3.2,0.8,32.0})
(2,{1.0,14.0,1.0})
```

reduce

MapReduce is suitable to ….
MapReduce 合適用於 ….

- 大規模資料集
- **Large Data Set**
- 可拆解
- **Parallelization**

- Text tokenization
- Indexing and Search
- Data mining
- machine learning
- …

- http://www.dbms2.com/2008/08/26/known-applications-of-mapreduce/
- http://wiki.apache.org/hadoop/PoweredBy

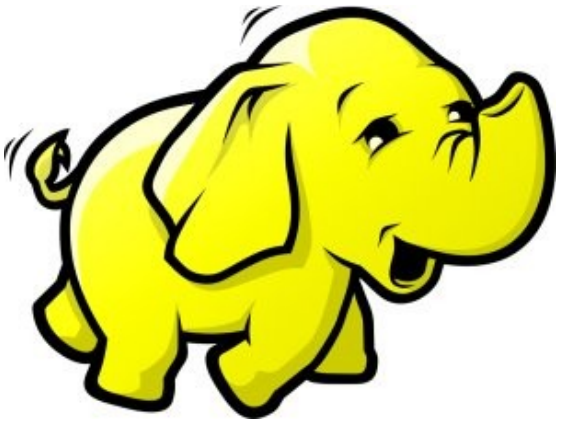
Hadoop 相關計畫

Hadoop Ecosystem

Jazz Wang

Yao-Tsung Wang

jazz@nchc.org.tw

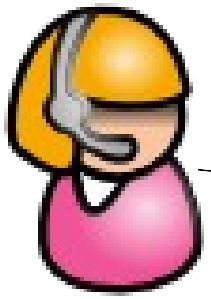


Hadoop 只支援用 **Java** 開發嘛？

Is Hadoop only support Java ?

總不能全部都重新設計吧？如何與舊系統相容？

Can Hadoop work with existing software ?



可以跟資料庫結合嘛？

Can Hadoop work with Databases ?

開發者們有聽到大家的需求.....

Yes, we hear the feedback of developers ...

Is Hadoop only support Java ?

- Although the Hadoop framework is implemented in Java™, Map/Reduce applications need not be written in Java.
- Hadoop Streaming is a utility which allows users to create and run jobs with any executables (e.g. shell utilities) as the mapper and/or the reducer.
- Hadoop Pipes is a SWIG-compatible C++ API to implement Map/Reduce applications (non JNI™ based).

Hadoop Pipes (C++, Python)

- Hadoop Pipes allows C++ code to use Hadoop DFS and map/reduce.
- The C++ interface is "swigable" so that interfaces can be generated for python and other scripting languages.
- For more detail, check the API Document of org.apache.hadoop.mapred.pipes
- You can also find example code at

  hadoop-*/src/examples/pipes
- About the pipes C++ WordCount example code:

  http://wiki.apache.org/hadoop/C++WordCount

Hadoop Streaming

- Hadoop Streaming is a utility which allows users to create and run Map-Reduce jobs with any executables (e.g. Unix shell utilities) as the mapper and/or the reducer.
- It's useful when you need to run existing program written in shell script, perl script or even PHP.
- Note: both the mapper and the reducer are executables that read the input from STDIN (line by line) and emit the output to STDOUT.
- For more detail, check the official document of Hadoop Streaming

Running Hadoop Streaming

```
jazz@hadoop:~$ hadoop jar hadoop-streaming.jar -help
10/08/11 00:20:00 ERROR streaming.StreamJob: Missing required option -input
Usage: $HADOOP_HOME/bin/hadoop [--config dir] jar \
          $HADOOP_HOME/hadoop-streaming.jar [options]
Options:
  -input    <path>     DFS input file(s) for the Map step
  -output   <path>     DFS output directory for the Reduce step
  -mapper   <cmd|JavaClassName>      The streaming command to run
  -combiner <JavaClassName> Combiner has to be a Java class
  -reducer  <cmd|JavaClassName>      The streaming command to run
  -file     <file>     File/dir to be shipped in the Job jar file
  -dfs    <h:p>|local  Optional. Override DFS configuration
  -jt     <h:p>|local  Optional. Override JobTracker configuration
  -additionalconfspec specfile  Optional.
  -inputformat TextInputFormat(default)|SequenceFileAsTextInputFormat|
JavaClassName Optional.
  -outputformat TextOutputFormat(default)|JavaClassName  Optional.
```

… More …

Hadoop Streaming with shell commands (1)

```
hadoop:~$ hadoop fs -rmr input output
hadoop:~$ hadoop fs -put /etc/hadoop/conf input
hadoop:~$ hadoop jar hadoop-streaming.jar -input
input -output output -mapper /bin/cat
-reducer /usr/bin/wc
```

Hadoop Streaming with shell commands (2)

```
hadoop:~$ echo "sed -e \"s/ /\n/g\" | grep ." > streamingMapper.sh
hadoop:~$ echo "uniq -c | awk '{print \$2 \"\t\" \$1}'" > streamingReducer.sh
hadoop:~$ chmod a+x streamingMapper.sh
hadoop:~$ chmod a+x streamingReducer.sh
hadoop:~$ hadoop fs -put /etc/hadoop/conf input
hadoop:~$ hadoop jar hadoop-streaming.jar -input input -output output -mapper streamingMapper.sh -reducer streamingReducer.sh -file streamingMapper.sh -file streamingReducer.sh
```

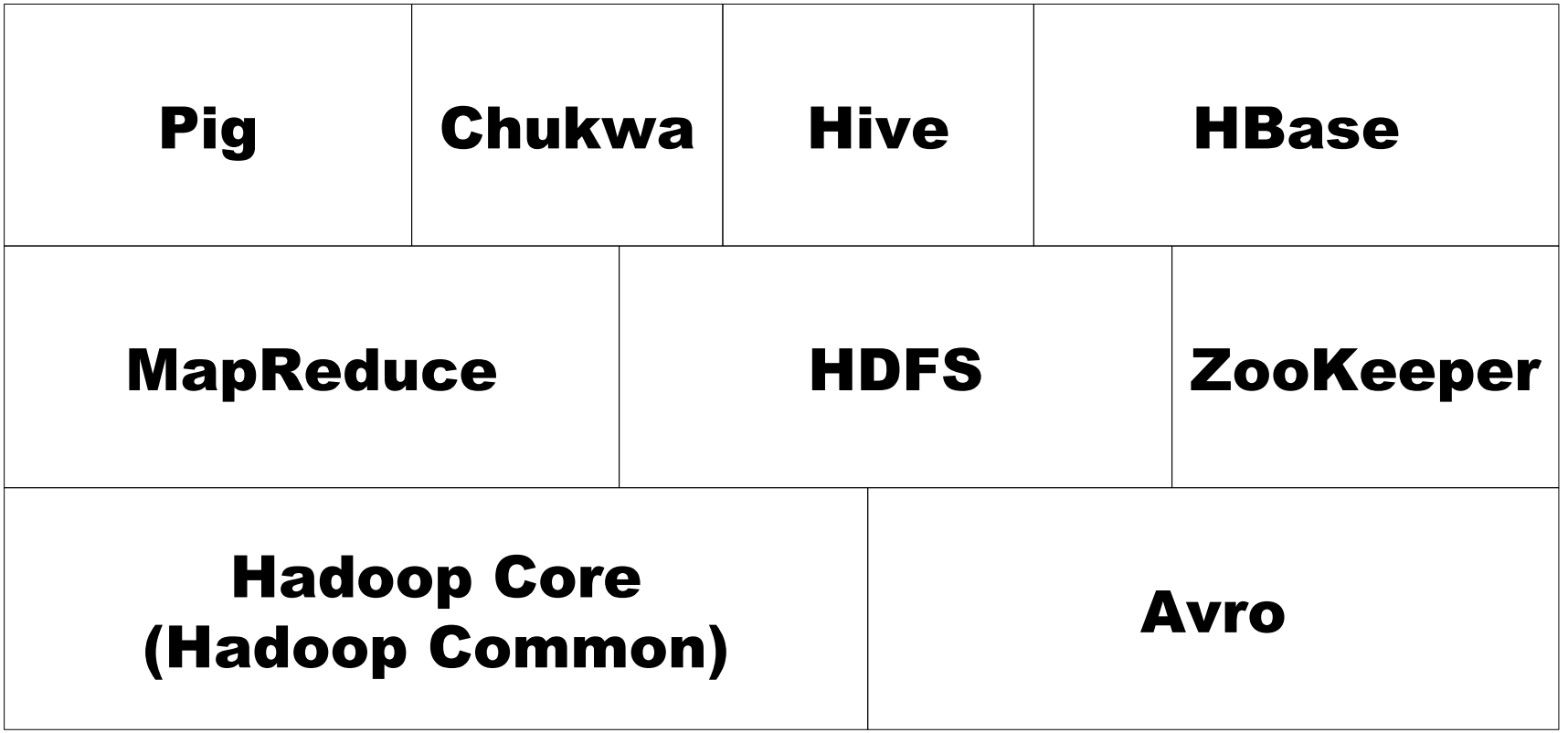
There are serveral Hadoop subprojects

- Hadoop Common: The common utilities that support the other Hadoop subprojects.
- HDFS: A distributed file system that provides high throughput access to application data.
- MapReduce: A software framework for distributed processing of large data sets on compute clusters.

Other Hadoop related projects

- Chukwa: A data collection system for managing large distributed systems.
- HBase: A scalable, distributed database that supports structured data storage for large tables.
- Hive: A data warehouse infrastructure that provides data summarization and ad hoc querying.
- Pig: A high-level data-flow language and execution framework for parallel computation.
- ZooKeeper: A high-performance coordination service for distributed applications.

Hadoop Ecosystem



Source: *Hadoop: The Definitive Guide*

Avro

- Avro is a data serialization system.
- It provides:
  - *Rich data structures.*
  - *A compact, fast, binary data format.*
  - *A container file, to store persistent data.*
  - *Remote procedure call (RPC).*
  - *Simple integration with dynamic languages.*
- Code generation is not required to read or write data files nor to use or implement RPC protocols. Code generation as an optional optimization, only worth implementing for statically typed languages.
- For more detail, please check the official document: http://avro.apache.org/docs/current/

Zoo Keeper

- http://hadoop.apache.org/zookeeper/
- ZooKeeper is a centralized service for maintaining configuration information, naming, providing distributed synchronization, and providing group services. All of these kinds of services are used in some form or another by distributed applications.
- *Each time they are implemented there is a lot of work that goes into fixing the bugs and race conditions that are inevitable. Because of the difficulty of implementing these kinds of services, applications initially usually skimp on them ,which make them brittle in the presence of change and difficult to manage. Even when done correctly, different implementations of these services lead to management complexity when the applications are deployed.*

Pig

- http://hadoop.apache.org/pig/
- Pig is a platform for analyzing large data sets that consists of a high-level language for expressing data analysis programs, coupled with infrastructure for evaluating these programs.
- Pig's infrastructure layer consists of a compiler that produces sequences of Map-Reduce programs
- Pig's language layer currently consists of a textual language called Pig Latin, which has the following key properties:
  - Ease of programming
  - Optimization opportunities
  - Extensibility

Hive

- http://hadoop.apache.org/hive/
- Hive is a data warehouse infrastructure built on top of Hadoop that provides tools to enable easy data summarization, adhoc querying and analysis of large datasets data stored in Hadoop files.
- Hive QL is based on SQL and enables users familiar with SQL to query this data.

Chukwa

- http://hadoop.apache.org/chukwa/
- Chukwa is an open source data collection system for monitoring large distributed systems.
- built on top of HDFS and Map/Reduce framework
- includes a flexible and powerful toolkit for displaying, monitoring and analyzing results to make the best use of the collected data.

Mahout

- http://mahout.apache.org/
- Mahout is a scalable machine learning libraries.
- implemented on top of Apache Hadoop using the map/reduce paradigm.
- Mahout currently has
  - Collaborative Filtering
  - User and Item based recommenders
  - K-Means, Fuzzy K-Means clustering
  - Mean Shift clustering
  - More ...

Introduction to Pig programming

Yahoo Search Engineering

陳奕瑋 (**Yiwei Chen**)

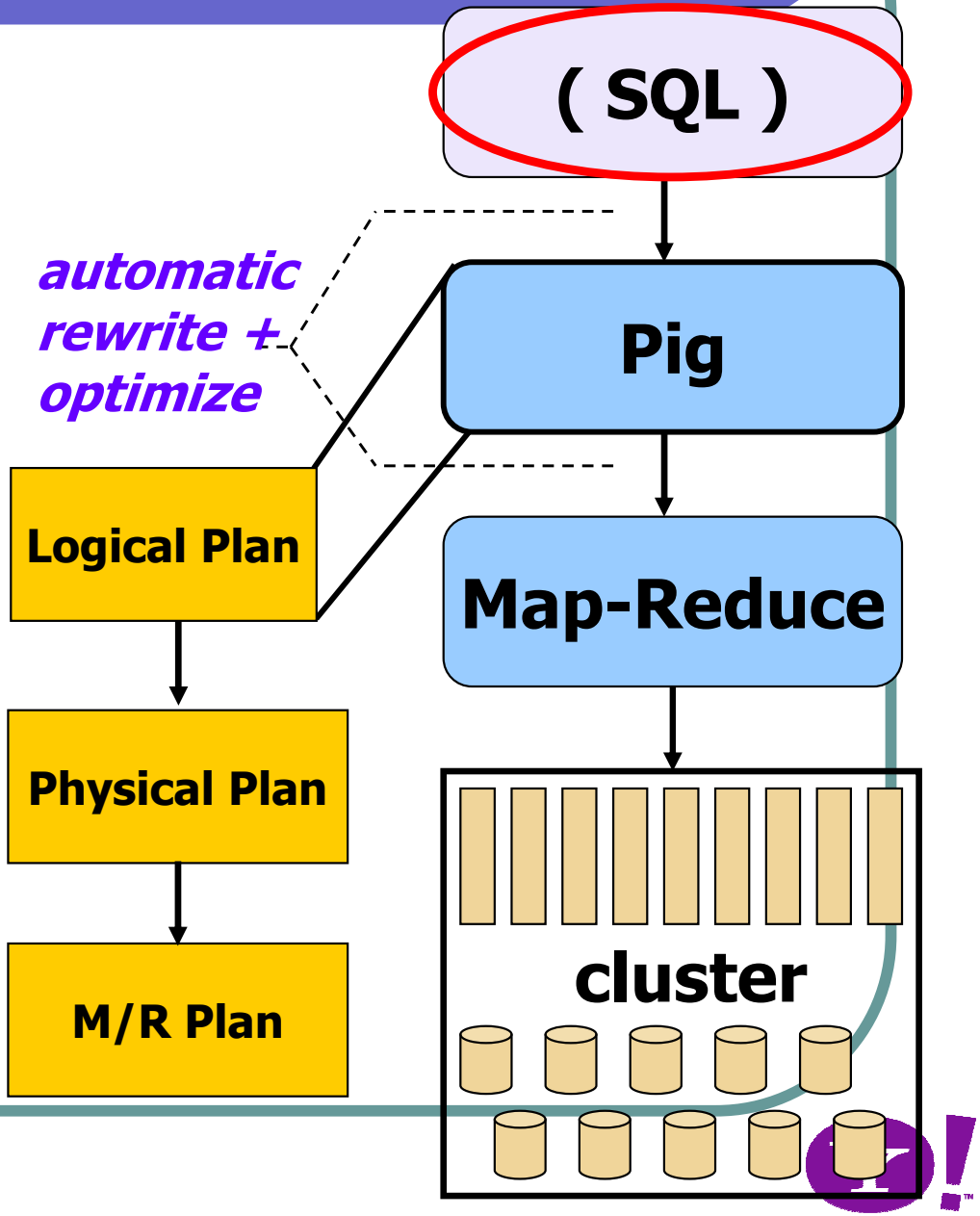
Pig Script Example

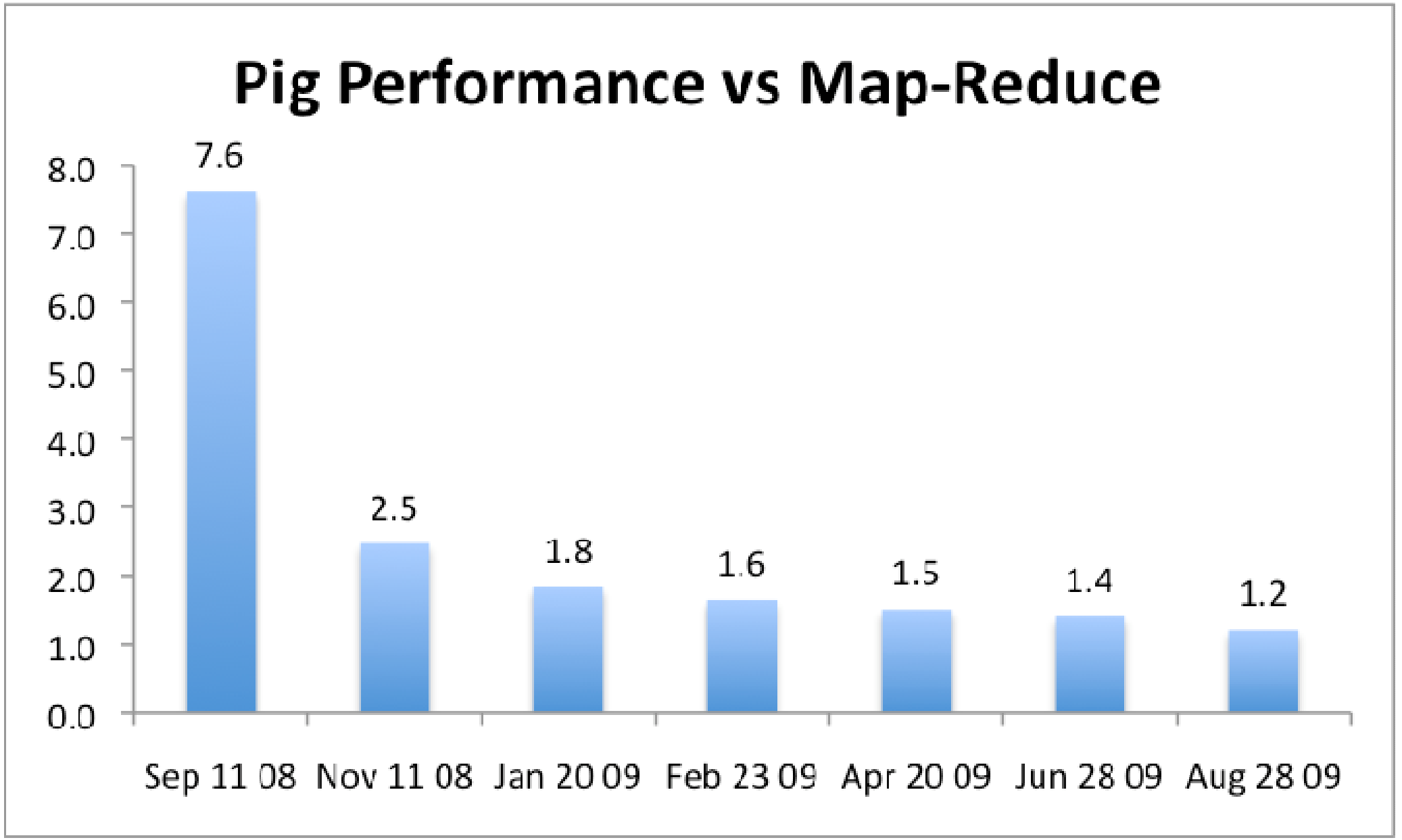
- Top sites visited by users aged 18 to 25

```
Users = LOAD ‘users.in’ AS (name, age);
Fltrd = FILTER Users by age >= 18 and age <= 25;

Pages = LOAD ‘pages.in’ AS (user, url);

Jnd   = JOIN Fltrd BY name, Pages BY user;
Grpd  = GROUP Jnd by url;
Smmd  = FOREACH Grpd GENERATE group, COUNT(Jnd) AS
        clicks;

Srtd  = ORDER Smmd BY clicks;
Top100 = LIMIT Srtd 100;

STORE Top100 INTO ‘top100sites.out’;
```

Pig script → Map/Reduce

- 不需懂底下 Map-Reduce 運作
- Pig 幫忙翻譯




Pig Performance vs Map-Reduce
8.0
7.0
6.0
5.0
4.0
3.0
2.0
1.0
0.0
7.6
2.5
1.8
1.6
1.5
1.4
1.2
Sep 11 08
Nov 11 08
Jan 20 09
Feb 23 09
Apr 20 09
Jun 28 09
Aug 28 09

How to execute

- Local:
  - `pig -x local foo.pig`
- Hadoop (HDFS):
  - `pig foo.pig`
  - `pig -Dmapred.job.queue.name=xxx foo.pig`
    - `hadoop queue -showacls`

How to execute

- Interactive pig shell
  - `$ pig`
  - grunt> _

Load Data

```
Users = LOAD 'users.txt'
        USING PigStorage(',') AS (name, age);
```

- LOAD … AS …
- PigStorage(‘,’) to specify separator

```
John,18
Mary,20
Bob,30
```

name	age
John	18
Mary	20
Bob	30

Filter

```
Fltrd = FILTER Users
        BY age >= 18 AND age <= 25;
```

- FILTER … BY …
  - constraints can be composite

name	age
John	18
Mary	20
Bob	30

name	age
John	18
Mary	20

Generate / Project

```
Names = FOREACH Fltrd GENERATE name;
```

- FOREACH … GENERATE

name	age
John	18
Mary	20

name
John
Mary

Store Data

```
STORE Names INTO 'names.out';
```

- STORE … INTO …
  - PigStorage(‘,’) to specify separator if multiple fields

Command - JOIN

```
Users = LOAD 'users' AS (name, age);
Pages = LOAD 'pages' AS (user, url);
Jnd   = JOIN Users BY name, Pages BY user;
```

name	age
John	18
Mary	20
Bob	30

user	url
John	yaho
Mary	goog
Bob	bing

name	age	user	url
John	18	John	yaho
Mary	20	Mary	goog
Bob	30	Bob	bing

Command - GROUP

```
Grpd  = GROUP Jnd by url;
describe Grpd;
```

name	age	url
John	18	yhoo
Mary	20	goog
Dee	25	yhoo
Kim	40	bing
Bob	30	bing

yhoo	(John, 18, yhoo) (Dee, 25, yhoo)
goog	(Mary, 20, goog)
bing	(Kim, 40, bing) (Bob, 30, bing)

Other Commands

- `PARALLEL` – controls #reducer
- `ORDER` – sort by a field
- `COUNT` – eval: count #elements
- `COGROUP` – structured JOIN
- More at http://hadoop.apache.org/pig/docs/r0.5.0/piglatin_reference.html

UDF

- http://hadoop.apache.org/pig/docs/r0.3.0/udf.html
- http://hadoop.apache.org/pig/javadoc/docs/api/
- PiggyBank
  - Pig users UDF repo
  - http://wiki.apache.org/pig/PiggyBank

References

- FAQ
  - http://wiki.apache.org/pig/FAQ
- Documentation
  - http://hadoop.apache.org/pig/docs/r0.5.0/
- Talks & papers
  - http://wiki.apache.org/pig/PigTalksPapers
  - http://www.cloudera.com/hadoop-training-pig-introduction