淺談雲端運算趨勢與關鍵技術

The trend of cloud computing and its core technologies

Jazz Wang

Yao-Tsung Wang

jazz@nchc.org.tw

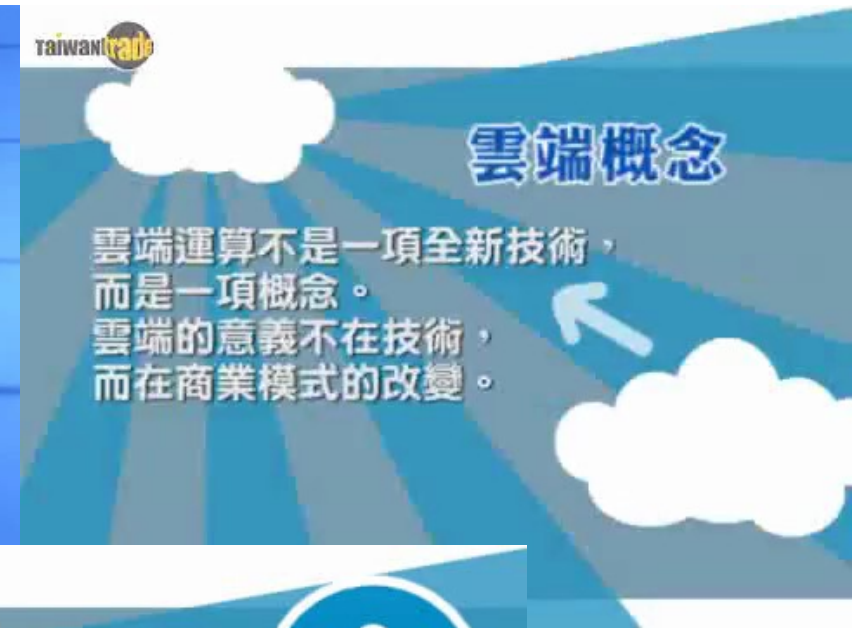
Powered by DRBL

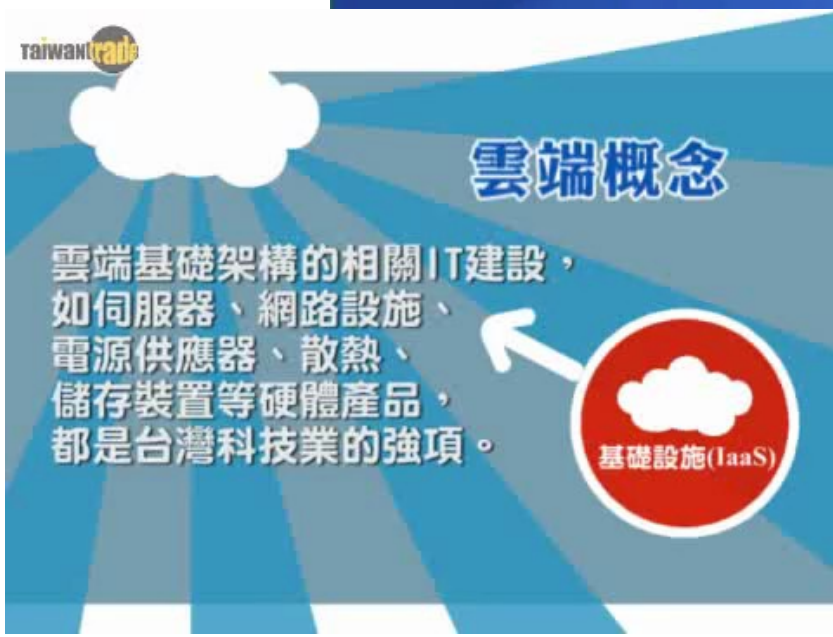
WHAT

Source: http://www.2010taipeiexpo.tw/ct.asp?xItem=17186&CtNode=5952&mp=3

什麼是雲端運算啊？

What is Cloud Computing ?





http://www.youtube.com/watch?v=bJLSAcU6O3U
http://www.youtube.com/watch?v=VIMtd3nfPqc

當紅「雲端運算」 你瞭解了嗎？
雲端產業 8分鐘就上手

什麼是雲端運算啊？可以個簡單的定義嗎？

What is Cloud Computing ?

雲端運算怎麼聽起來要買一些新硬體、新軟體啊？

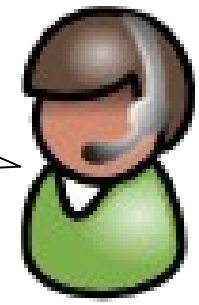
Is it about buying NEW Hardware and Software?

雲端運算可能只是拿來振興經濟的幌子吧？

Is it a trap to another bubble economy ?

我聽你們在那裡講五四三 .....

Cloud Computing is as simple as 5..4..3..2..1...



National Definition of Cloud Computing
美國國家標準局 NIST 給雲端運算所下的定義

5 Characteristics 五大基礎特徵

4 Deployment Models 四個佈署模型

3 Service Models 三個服務模式

1. On-demand self-service.
隨需自助服務

2. Broad network access
隨時隨地用任何網路裝置存取

3. Resource pooling
多人共享資源池

4. Rapid elasticity
快速重新佈署靈活度

5. Measured Service
可被監控與量測的服務

4 Deployment Models of Cloud Computing
雲端運算的四種佈署模型

Public Cloud
公用雲端

Target Market is S.M.B.
主要客户爲中小企業

Dynamic Resource Provisioning between public and private cloud
私有雲端動態根據計算需求調用公用雲端的資源

Hybrid Cloud

以大型企業爲主要客户
Enterprise is key market

私有雲端
Private Cloud

Community Cloud
社群雲端

Academia 學術爲主

3 Service Models of Cloud Computing
雲端運算的三種服務模式（市場區隔）

IaaS
Infrastructure as a Service
架構即服務

PaaS
Platform as a Service
平台即服務

SaaS
Software as a Service
軟體即服務

2 perspectives : Services vs Technologies
您想聽的是「雲端服務」還是「雲端技術」？

Cloud computing hype spurs confusion, Gartner says
http://www.computerworld.com/s/article/print/9115904
淺談雲端運算 **(Cloud Computing)**
http://www.cc.ntu.edu.tw/chinese/epaper/0008/20090320_8008.htm

1 key spirit of Cloud Computing
用一句話說明雲端運算！服務才是王道！

Anytime 隨時

Anywhere 隨地

With Any Devices 使用任何裝置

Accessing Services 存取各種服務

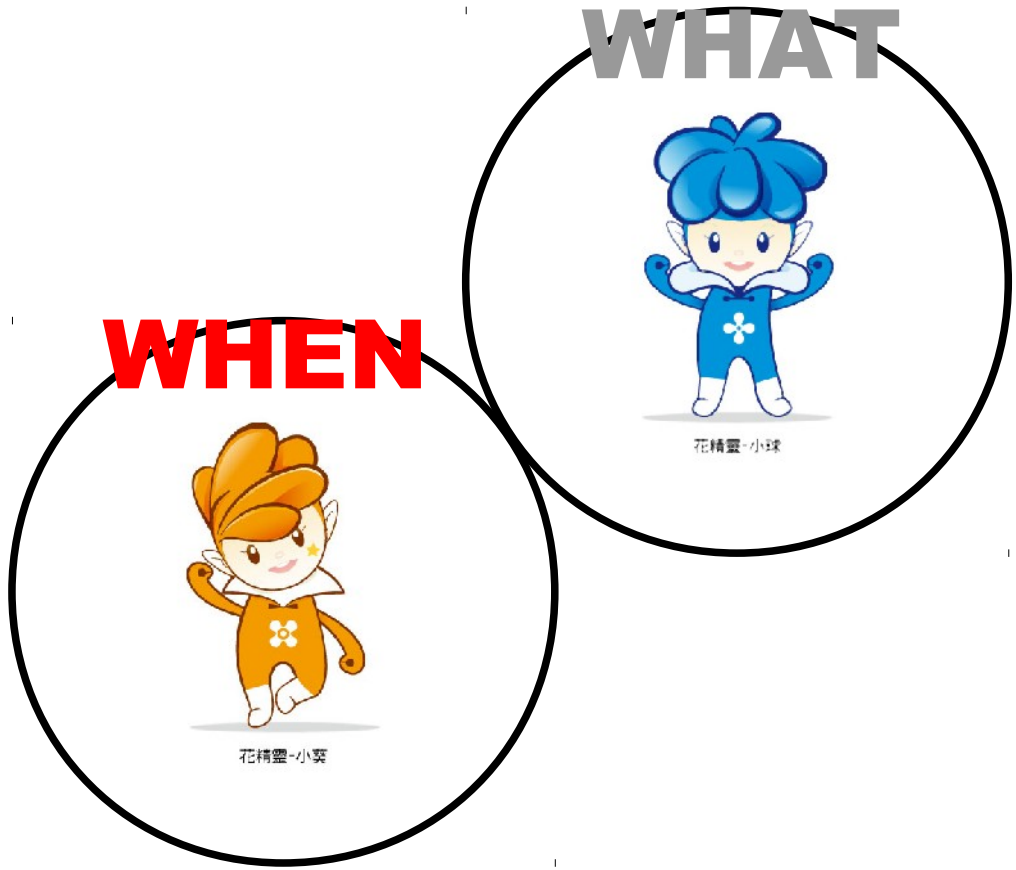
Key spirit of Cloud ~

形成服務才是重點！！

Everything as a Service !!

Cloud Computing =~ Network Computing

雲端運算 =~ 網路運算



Source: http://www.2010taipeiexpo.tw/ct.asp?xItem=17186&CtNode=5952&mp=3

The wisdom of Clouds (Crowds)
雲端序曲：雲端的智慧始終來自於群眾的智慧

2006 年 8 月 9 日

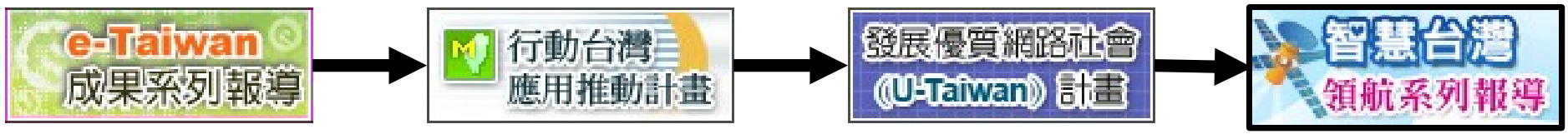
Google 執行長施密特（Eric Schmidt）於SES '06會議中首次使用「雲端運算（Cloud Computing）」來形容無所不在的網路服務

2006 年 8 月 24 日

Amazon 以 Elastic Compute Cloud 命名其虛擬運算資源服務

Source: http://www.cnet.co.uk/i/c/blg/cat/software/cloudcomputing/clouds1.jpg

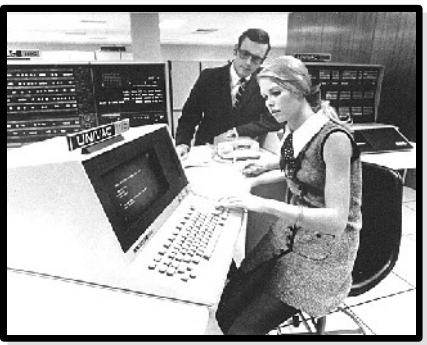
Evolution of Cloud Services
雲端服務只是軟體演化史的必然趨勢



實體 Physical	單機版 個人使用 Personal Software	網路版 多人共享 Share Service Software	行動版 隨時存取 Mobile Cloud Service
信箱 Mailbox	E-Mail	Web Mail	Mobile Mail
電視 TV	電視盒 Setop Box	Web TV Ex. Youtube	Mobile TV
打字機 Typer Writer	Office	Google Docs	M-Office
電話 Telephone	數位電話 PBX	Skype	Flash Wengo
佈告欄 Bullet Borad	電子佈告欄 BBS	部落格 Blog	微網誌 Twitter

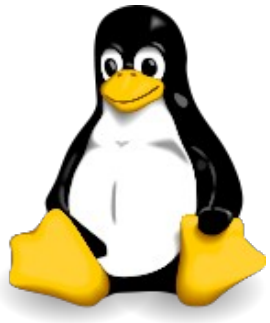
Brief History of Computing
運算技術演進簡史

1960



1977

1991



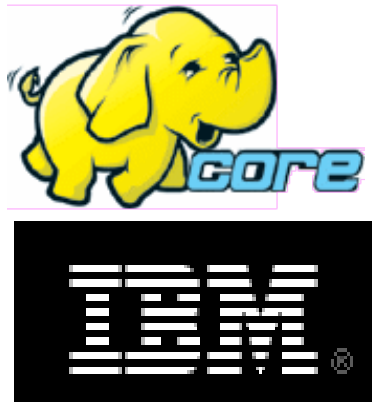
1993

2002

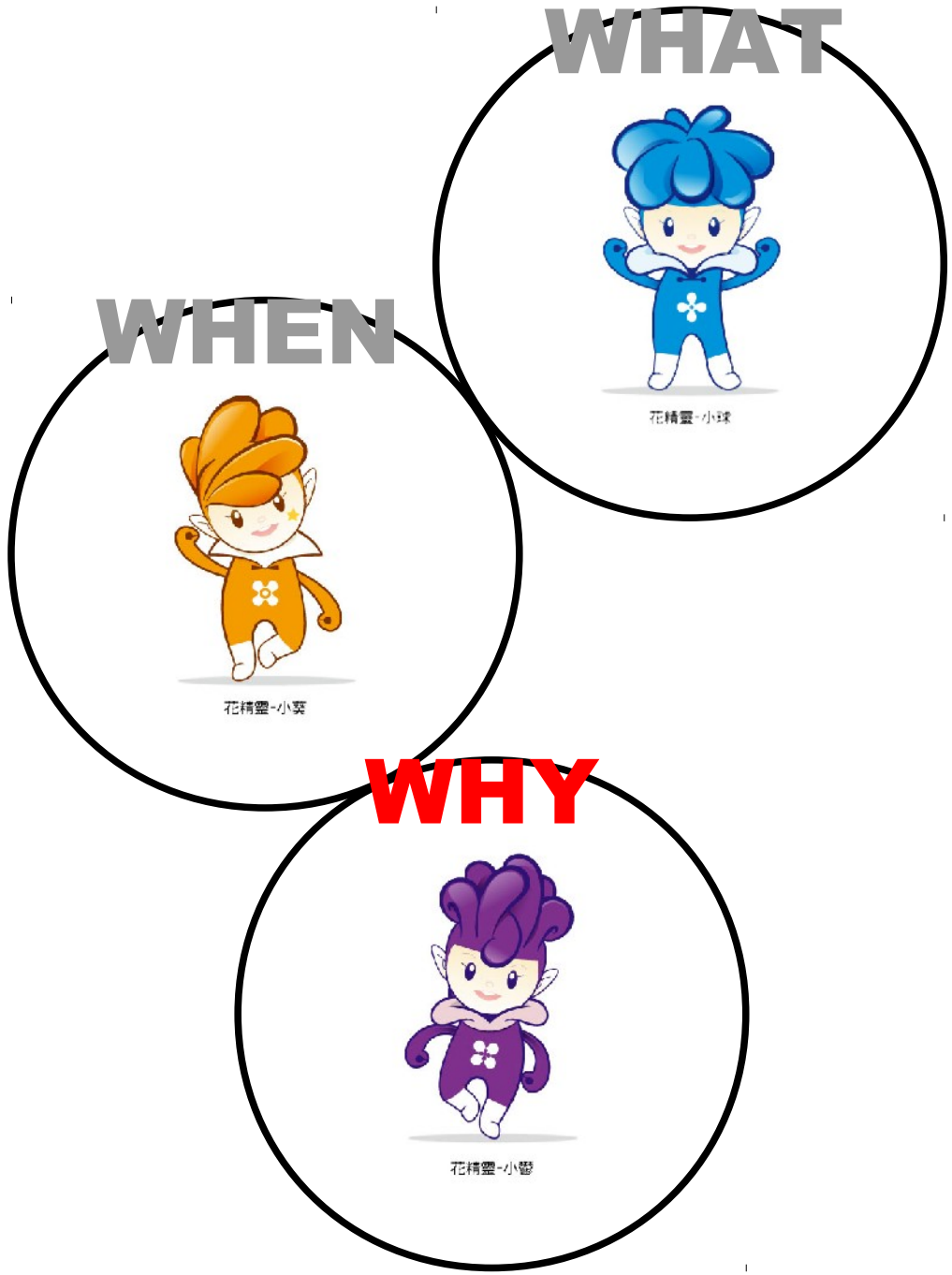
2003

2004

2006

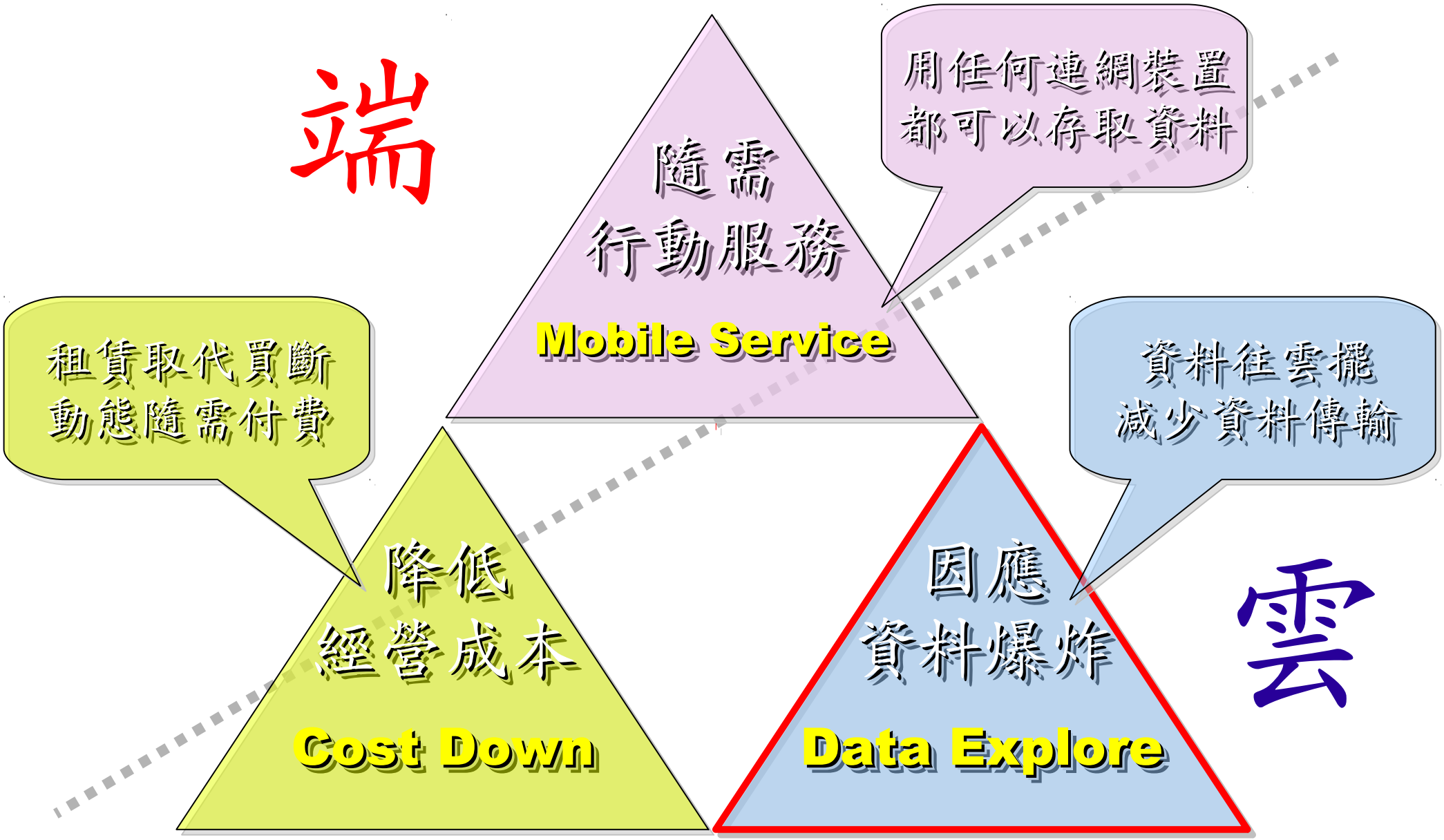


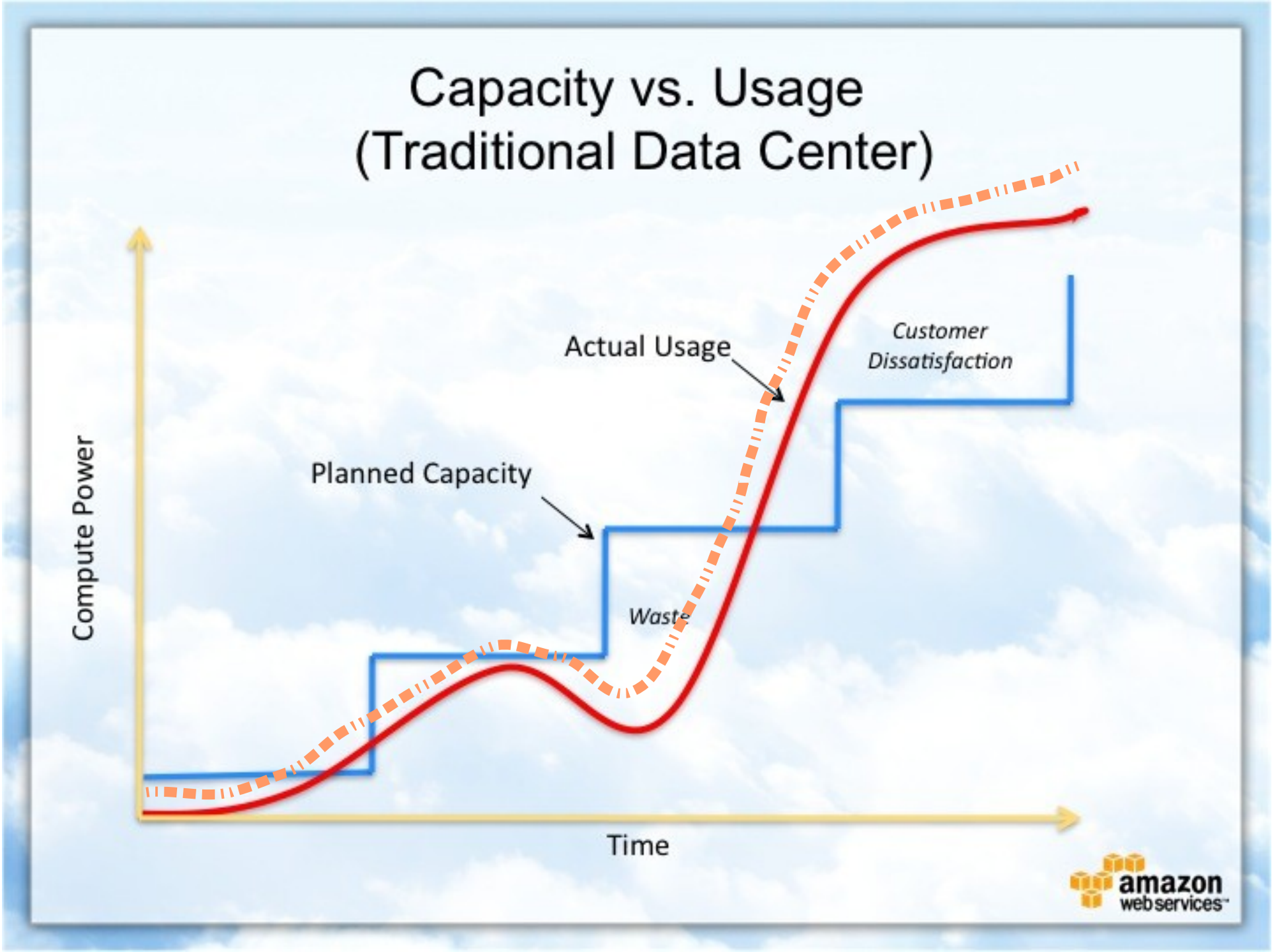
Mainframe Super Computer → **PC / Linux** Cluster Parallel → **Internet** Distributed Computing → **Virtual Org.** Grid Computing → **Data Explode** Cloud Computing



Source: http://www.2010taipeiexpo.tw/ct.asp?xItem=17186&CtNode=5952&mp=3

Key Driving Forces of Cloud Computing
雲端運算的關鍵驅動力





Source : http://awsmedia.s3.amazonaws.com/chart01_traditional_720x540.jpg

Lesson #1: One cluster can't fit all !
教訓一：叢集的單一設定無法滿足所有需求！

Answer #1: Virtual Cluster 新服務：虛擬化叢集

Lesson #2: Grid for Heterogeneous Enterprise !
教訓二：格網運算該用在異業結盟的資源共享！

Answer #2: Peak Usage Time 尖峰用量發生時間點

Lesson #3: Extra cost to move data to Grid !
教訓三：資料搬運的網路與時間成本！

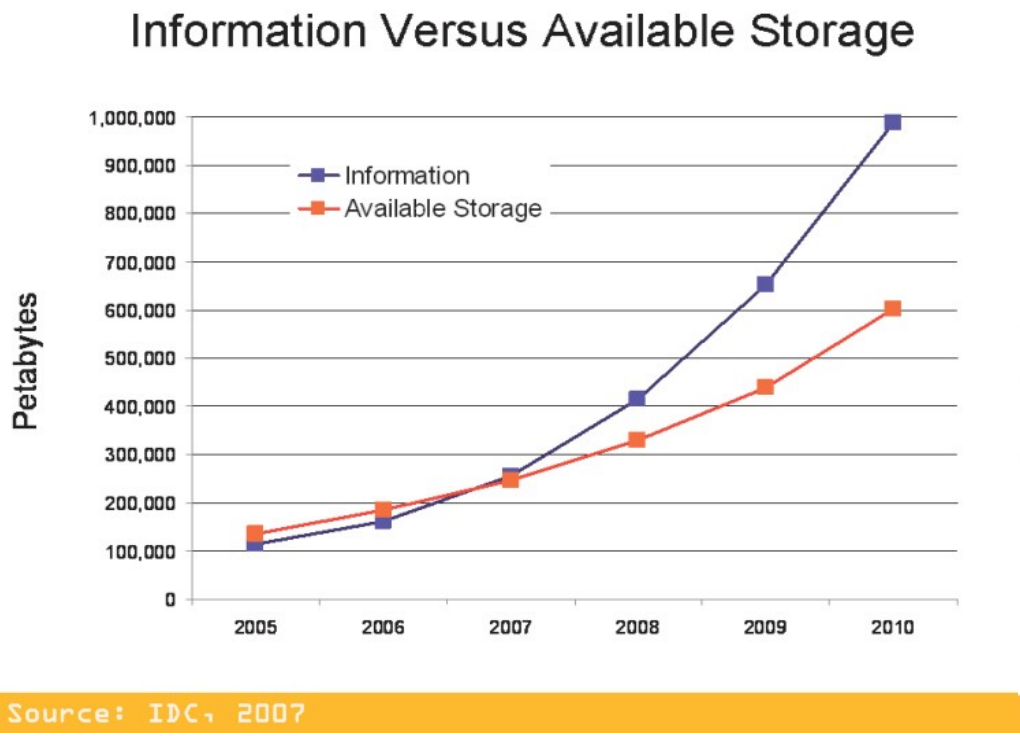
Answer #3: Total Cost of Ownership 總擁有成本

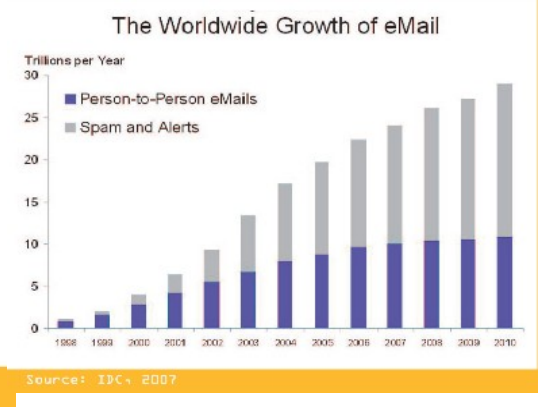
Cost Down is the Key Drive !!

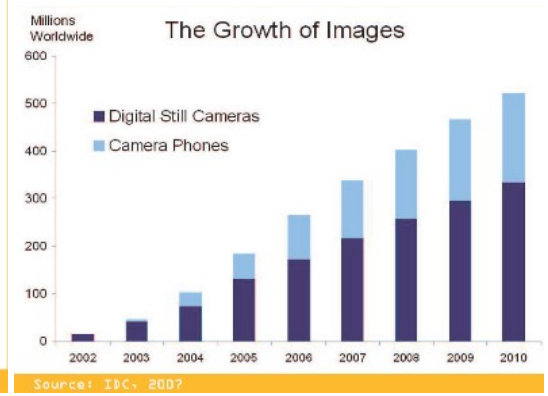
降低營運成本才是企業導入雲端運算的關鍵考量！!

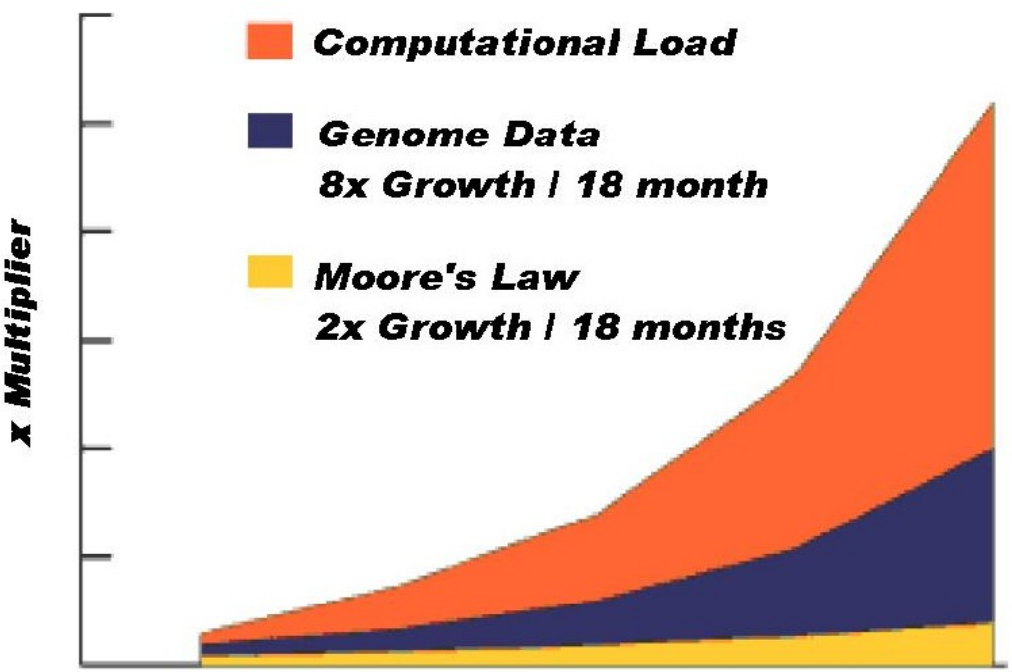
2007 Data Explore

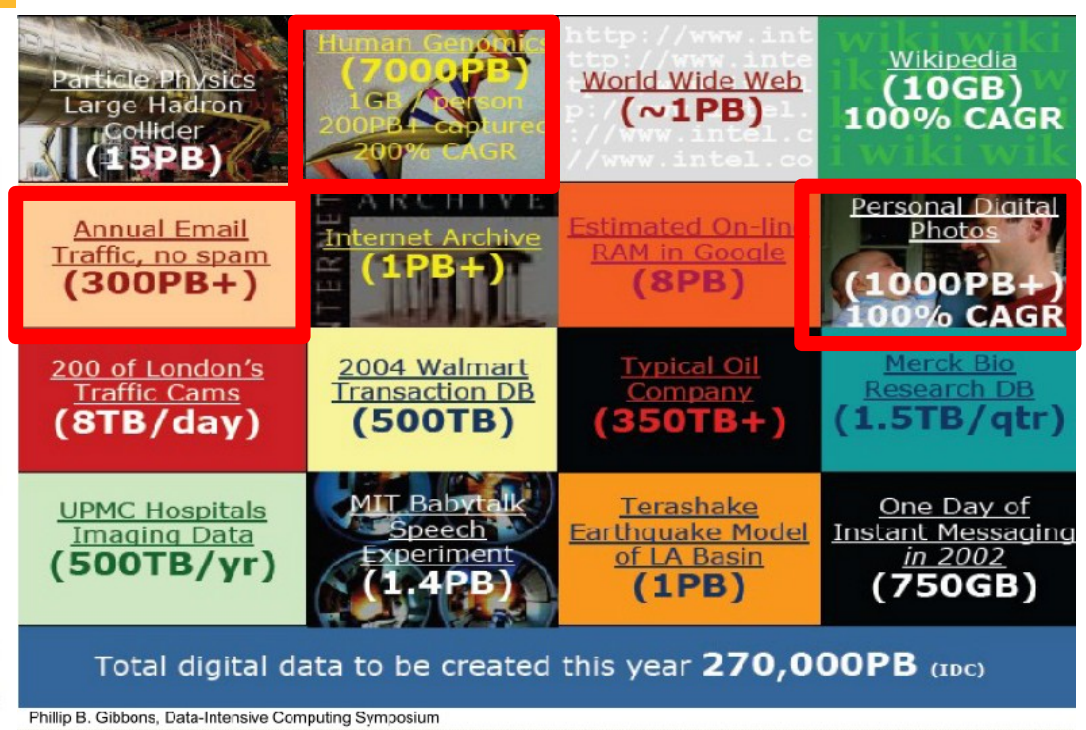
Top 1 : Human Genomics – 7000 PB / Year
Top 2 : Digital Photos – 1000 PB+/ Year
Top 3 : E-mail (no Spam) – 300 PB+ / Year







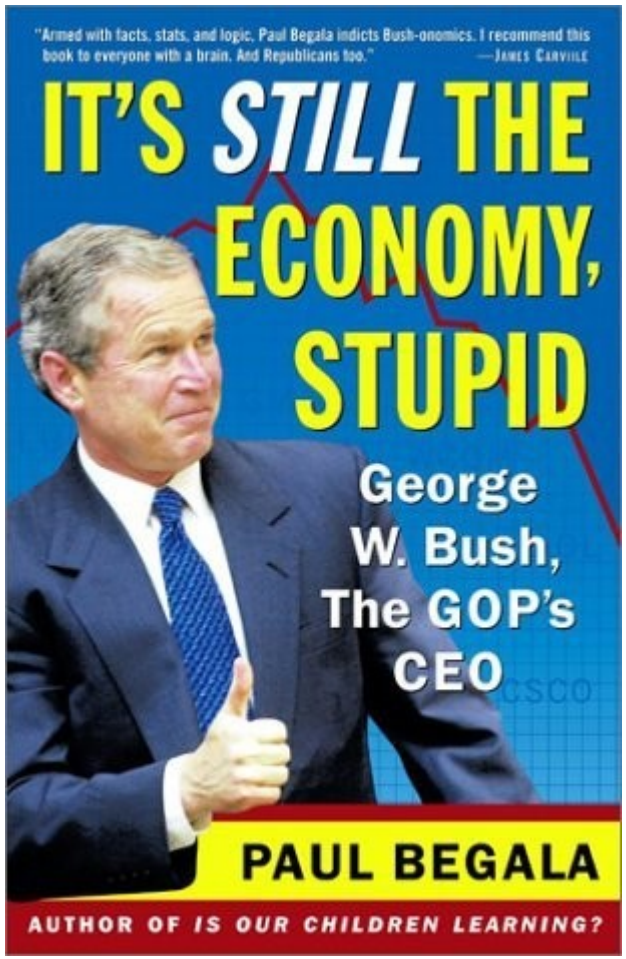




Phillip B. Gibbons, Data-Intensive Computing Symposium

Source: http://www.emc.com/collateral/analyst-reports/expanding-digital-idc-white-paper.pdf
Source: http://lib.stanford.edu/files/see_pasig_dic.pdf

IT'S THE DATA, STUPID!



「笨蛋！重點在經濟」

（**"It's the economy, stupid"**）

卡維爾（**James Carville**）自創這句標語，

促使柯林頓當上美國第 **42** 屆總統。

- 1992 年

「笨蛋！重點還是在經濟」

（**"It's STILL the economy, stupid"**）

卻讓小布希嘲笑是幼稚的總統。

- 2002 年

雲端時代，谷歌會說：「笨蛋！重點在資料」

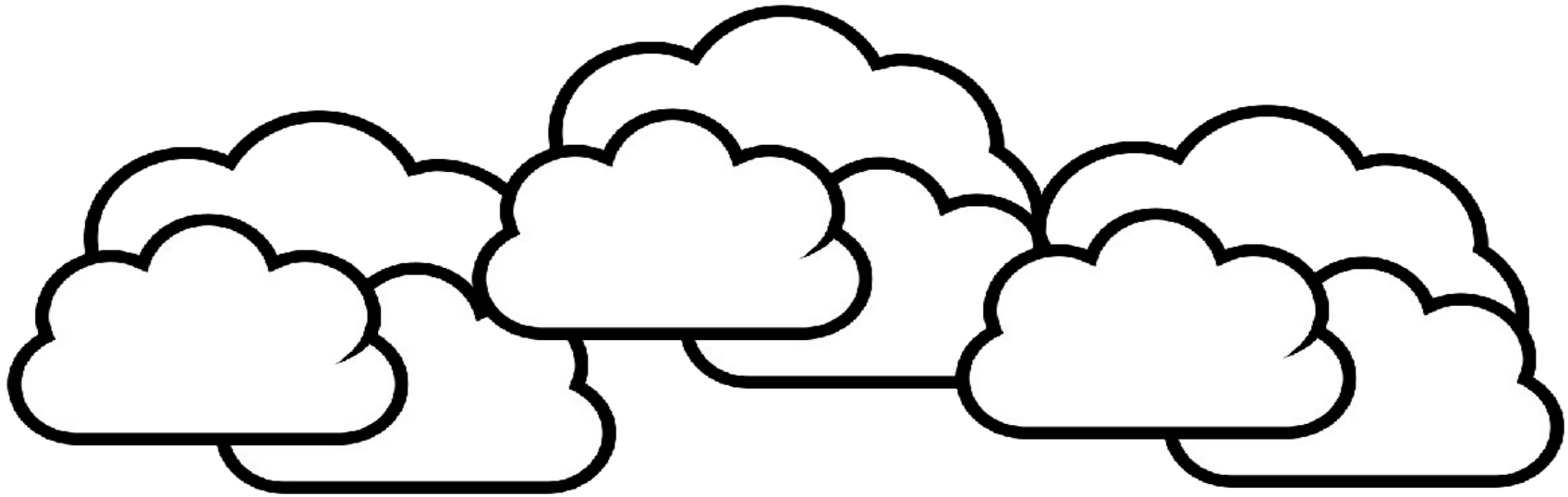
（**"It's the data, stupid"**）

誰掌握了你的資料，就有機會掌握你的荷包

想想看，電腦、手機掉了，您心疼的是甚麼呢？

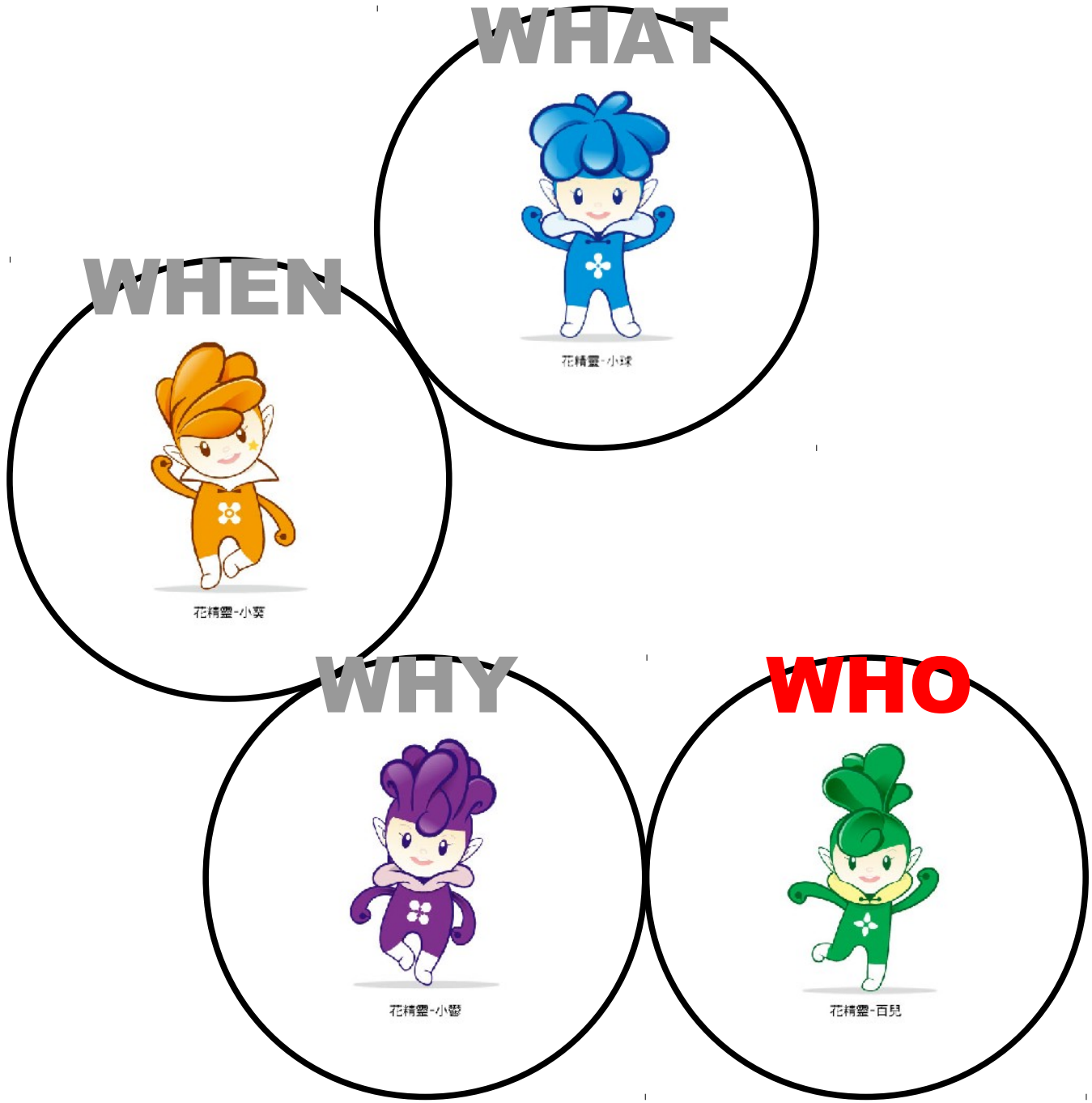
- 2007 年

Data is the source of Wisdom !!
用雲掌握資料，加以分析，形成智能給端用



嵌入式的新思維：未來，端的智能來自於雲

Devices share the wisdom of Cloud



Source: http://www.2010taipeiexpo.tw/ct.asp?xItem=17186&CtNode=5952&mp=3

How can we build Cloud Services ??
觀察雲端關鍵提供者，找尋打造雲端服務的模式

雲端服務參考模型
Reference Model

Public Cloud
公用雲端

Target Market
is **S.M.B.**
主要客户爲
中小企業

Hybrid
Cloud

以大型企業
爲主要客户
Enterprise is
key market

私有雲端
Private Cloud

Community Cloud
社群雲端

Academia 學術爲主

IaaS Key Player: Amazon 亞馬遜網路書店

- Amazon Web Service (AWS)
- 虛擬伺服器： Amazon EC2
  - Small (Default) $0.085 per hour(L) - $0.12 per hour(W)
  - All Data Transfer $0.15 per GB
- 儲存服務： Amazon S3
  - $0.15 per GB – first 50 TB / month of storage used
  - $0.15 per GB – all data transfer in
  - $0.01 per 1,000 PUT, COPY, POST, or LIST requests
- 觀念： Paying for What You Use

參考來源： http://eblog.cisanet.org.tw/post/Cloud-Computing.aspx
http://aws.amazon.com/ec2/pricing/
http://aws.typepad.com/aws/2010/02/aws-data-transfer-prices-reduced.html
http://aws.amazon.com/s3/#pricing

PaaS / SaaS Key Player : Google 公歌

- Google App Engine (GAE)
- 讓開發者可自行建立網路應用程式於 Google 平台之上。
- 提供：
  - 500MB of storage
  - up to 5 million page views a month
  - 10 applications per developer account
- 限制：
  - 程式設計語言只能用 Python 或 Java

計費標準：

- 連出頻寬 $0.12 美元/GB, 連入頻寬 $0.10 美元/GB
- CPU 時間 $0.10 美元/時
- 儲存的資料 $0.15 美元/GB-每月
- 電子郵件收件者 $0.0001 美元/每個收件者

參考來源： http://code.google.com/intl/zh-TW/appengine/
http://code.google.com/intl/zh-TW/appengine/docs/billing.html

PaaS / SaaS Key Player : Google 公歌


Gmail / 電子郵件


Contac / 通訊錄


Postini / 通訊安全


Calendar / 行事曆

Talk / 即時通


Group / 網上論壇


Doc / 文件


Video / 影音


Sites/ 協作平台


Mobile / 行動使用Apps

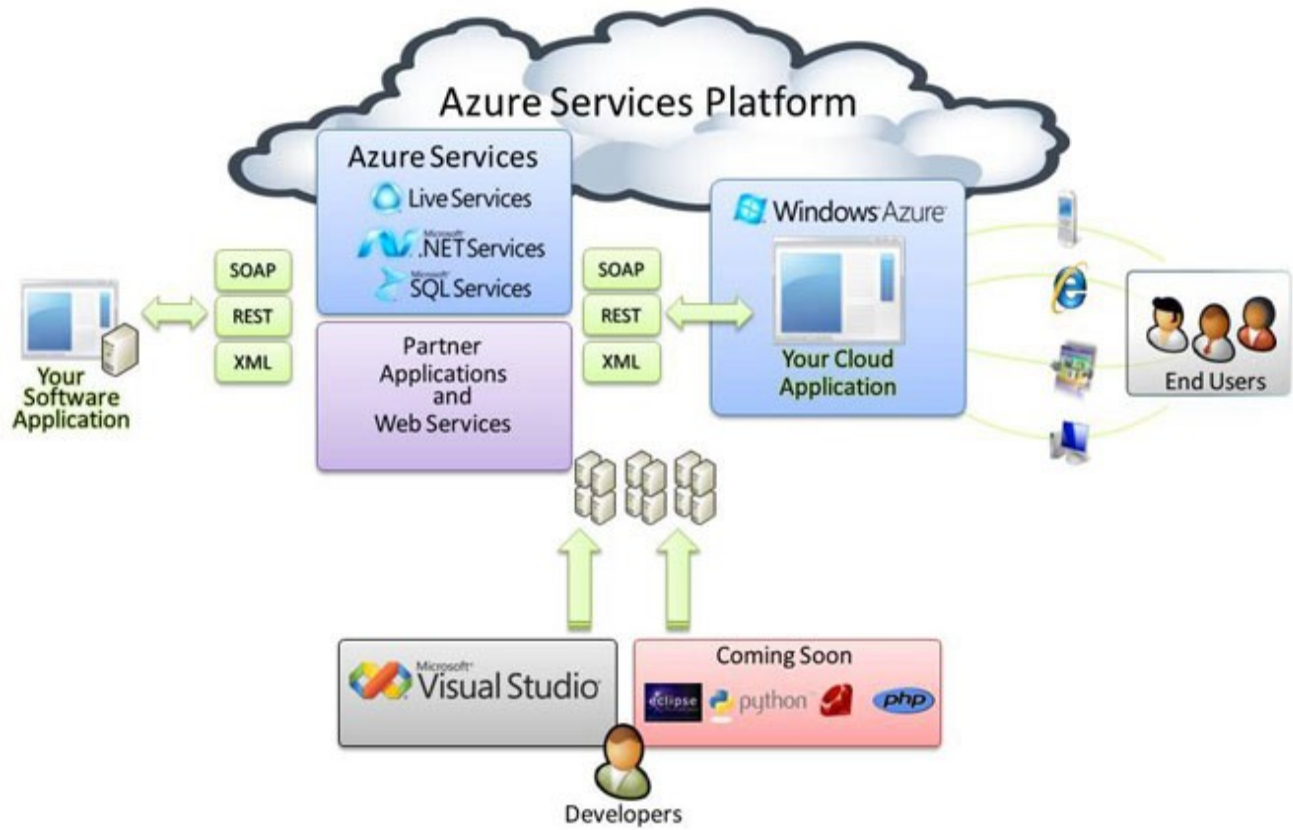

Admin / 管理後台

- **Google Apps**
- **Google Apps for Government**
- **Google Apps for ISPs**
- **Google Apps for Business**
- **Google Apps for Non-profits**

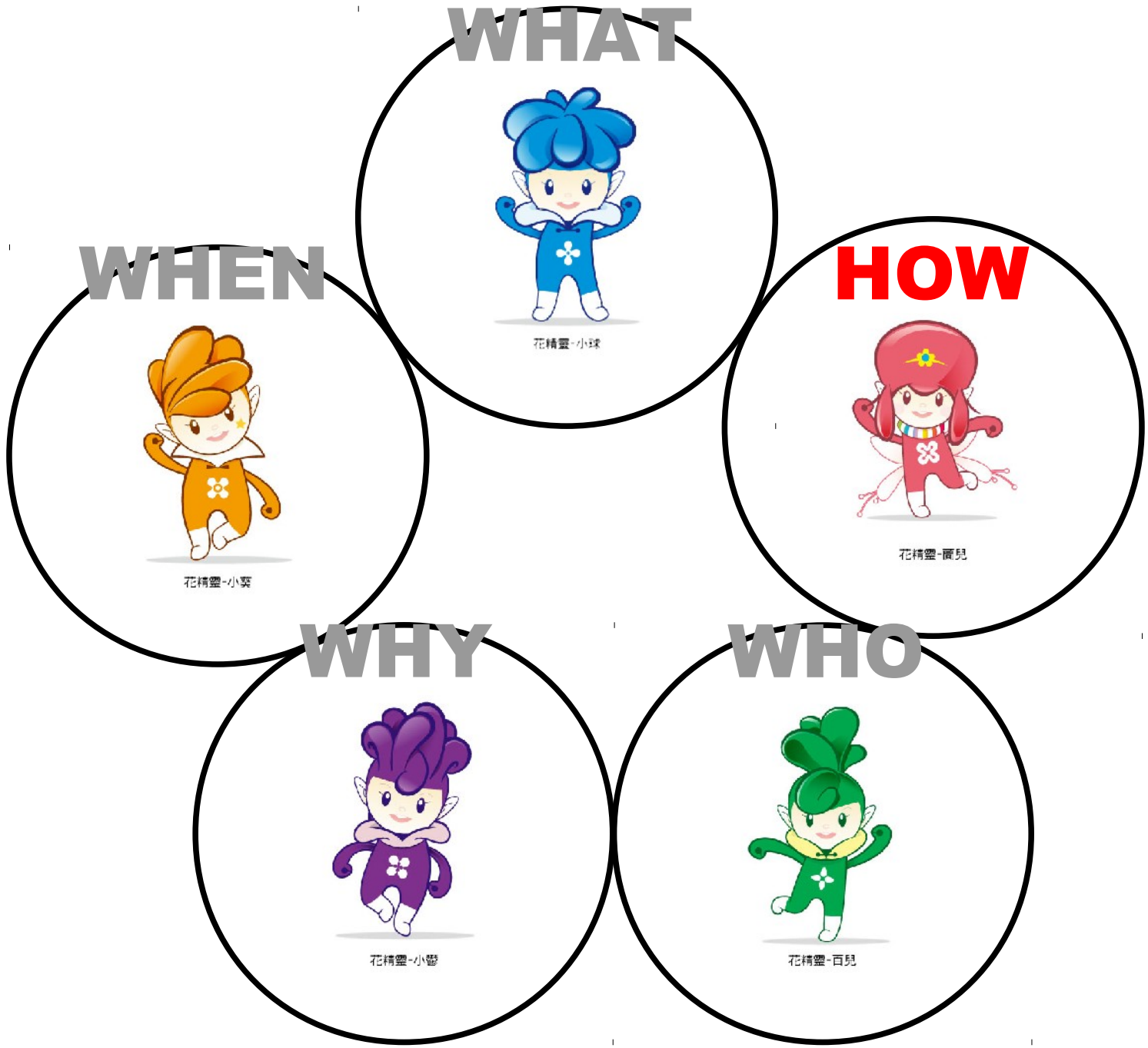
如果無法掌握雲端技術，至少該學會使用雲端服務！

XaaS Key Player : Microsoft 微軟

- Microsoft Azure 是一套雲端服務作業系統。
- 作爲 Azure 服務平台的開發、服務代管及服務管理環境。
- 服務種類：
  - .Net services
  - SQL services
  - Live services

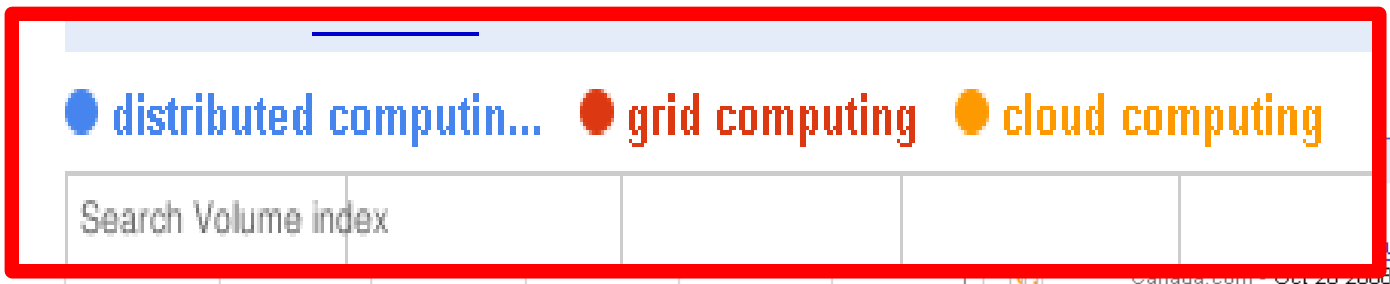


參考來源： http://tech.cipper.com/index.php/archives/332



Source: http://www.2010taipeiexpo.tw/ct.asp?xItem=17186&CtNode=5952&mp=3

What are the trend of next 10 years ?
什麼是下個十年的熱門工作技能？



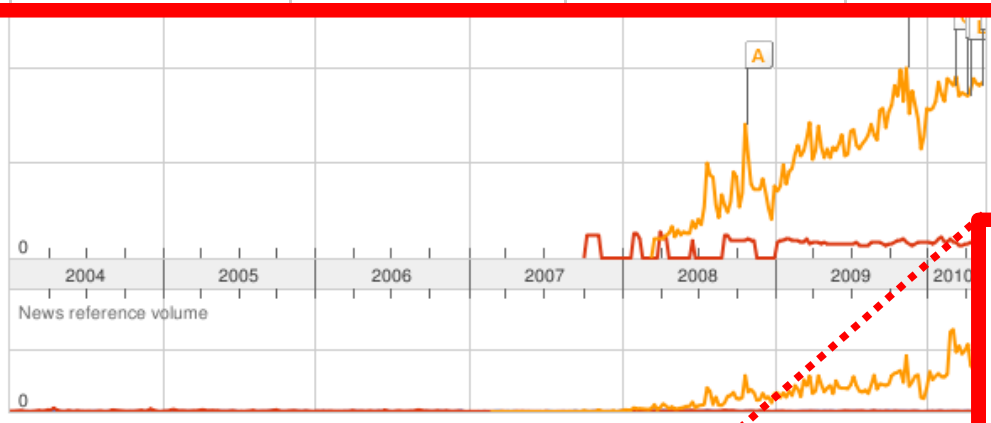
Sign in to see and export additional Tren

All regions | All years

ud" computing space

B Microsoft's cloud computing system is growing up
Philadelphia Inquirer - Nov 17 2009

C Google looks to be 'cloud-computing' rainmaker for other online business services
Winnipeg Free Press - Mar 10 2010



Rank by cloud computing

Regions

1. India
2. Singapore
3. South Korea
4. Hong Kong
5. Taiwan
6. Ireland

Cities

1. Bangalore, India
2. Mahape, India
3. Mumbai, India
4. Chennai, India
5. San Jose, CA, USA
6. Delhi, India

Regions

1. India
2. Singapore
3. South Korea
4. Hong Kong
5. Taiwan
6. Ireland

似乎亞洲國家特別熱愛雲端?! Too Hot in Asia ?!

Are the trends telling the truth ?

你確定沒有被圖表晃點嗎？

善用雲端架構 打造企業人才庫

對於雲端的運用，多半仍停留在創造新商機的層次，然而善用雲端運算，可以替組織創造更多業務、行銷和人才培訓的機會。

作者：麥肯錫　出處：天下雜誌

過去五年，麥肯錫觀察重要科技發展，其中雲端、大量資訊（big data）、智慧裝置（smart assets）三項，以超乎想像的速度發展。這三大技術，帶來五大趨勢，可被應用在企業營運及組織運作。先分別來看這三項技術：

第一、雲端運算。「雲端」在台灣已被一般民眾熟知。但我認為大家多半仍停留在雲端運算如何能創造新商機，卻很少好好思索，該怎麼運用雲端運算來替組織創造更多機會。特別是服務提供者，譬如電信業者、有線電視業者等，都應更有效應用雲端運算，為業務帶來更多機會。

第二、大量資訊。目前，絕大多數台灣企業，分析大量龐雜資料，仍使用類似微軟工具如excel等來整理。事實上，大量資訊經過快速運算分析，能更省時、省費用、有效的進行行銷活動。

第三、智慧裝置。如何善用監控器、智慧電表這類智慧裝置，來更優化公司營運。

雲端運算

大量資訊分析

智慧裝置

參考來源：善用雲端架構 打造企業人才庫，作者：麥肯錫，出處：天下雜誌 455 期（2010/09）
http://www.cw.com.tw/article/print.jsp?id=41776

New Data Science : Social Network + Realtime Search
當「社交網路」遇上「即時搜尋」 ＝ 即時市場行銷分析

創意行銷／臉書行銷 每天400萬顧客在線上

【經濟日報/潘俊琳】 2010.10.11 02:20 am

社交網站臉書Facebook的興起，重新定義了網路行銷的概念，大量的人潮讓業者彷彿看到滾滾錢潮，但臉書「開放平台」的模式，讓習慣有規則可循的行銷業者，必須開始學習全新的社群行銷，試著擁抱這項利器並串連消費者。

根據美國comScore的統計，美國網友8月分共花了1,140萬分鐘在臉書上，首次超越停留在Google旗下網站的時間，而臉書全球已經有超過5億的使用者，其中有35%的人每天登入。

快速分享 即時知道顧客反應

聖洋科技執行長邱繼弘表示，台灣臉書每個月約有700萬的累計使用人次，以60%每天上臉書的人口來算，就有420 萬人天天上線。

邱繼弘指出，臉書最大的行銷價值在於「開放平台」，只要符合它的基本規範，任何人、任何公司都可以在上面「免費」發揮自己的行銷創意。過去想要利用網路行銷，企業必須自己架站，林林總總的後台建設非常繁瑣，有多少人會來也是個問號？

但臉書幫企業解決了後台建設以及人潮，不論是企業或個人，只要成立自己的「粉絲專頁」，然後發揮行銷創意，回收可能比自己架站還更豐碩。因為臉書玩家只要在粉絲專頁按「讚」，就成為「粉絲團」的一員，往後企業發布在粉絲專頁的訊息，所有粉絲團成員都會收到，如果粉絲團的成員覺得某個行銷訊息不錯，只要按「分享」這個訊息就會出現在粉絲個人的臉書上，他所有的朋友就會看到這則行銷訊息，這是目前最高明的病毒式行銷。

社交網路

即時搜尋

評價排行榜

參考來源：創意行銷／臉書行銷 每天 400 萬顧客在線上 【經濟日報／潘俊琳】
http://udn.com/NEWS/FINANCE/FIN11/5901891.shtml

2011 年 10 大策略科技

科技	影響
雲端運算	大型企業將會在 2012 年成立動態採購小組，專門負責雲端運算相關的決定以及管理。
媒體平板以及行動應用	2010 年將會有 12 億人使用具備上網能力的手機。隨著行動上網裝置以及應用程式日趨普及，與地點(location)、動作(motion)相關的應用軟體，可望進一步推動裝置的銷售。
社交溝通以及協作(collaboration)	多數的公司在 2016 年已經把社交科技整合至多數的企業應用中，整合的範圍包含內部社交 CRM、溝通及協作以及外部社交網站。
影片	2013 年每位工作者看到的內容中，將有 25%都是照片、影音。
次世代分析	隨著電腦、行動裝置運算能力、連結能力更強，影響企業如何決策，SAS 是長期領導廠商，IBM 以及甲骨文(Oracle)事後起之秀。
社交分析	衡量人、主題以及想法的關係，範圍不限於社交網路，IBM 預計在 2011 年成爲該領域的主要廠商之一。
情境感知運算(context-aware computing)	較人工智慧更爲寬廣，預計在 2013 年時 Fortune 500 大企業中超過半數會有相關採用方案。
儲存等級記憶體(storage class memory)	快閃記憶體在消費性裝置、娛樂設備中的使用更多。
無所不在的運算(ubiquitous computing)	儘管 Gartner 已經提及這個概念許多年，但隨著手機、射頻晶片更爲普及，越多的物件可以連上網路。
架構化(fabric-based)的基礎建設以及電腦	運算能力模組化，系統可以透過不同的模組來建構，可望提升效能。

資料來源：DIGITIMES 整理，2010/10　　製表：雷佳宜、李盈瑩

雲端運算

平板行動應用

社交溝通協作

多媒體內容

次世代分析

社交分析

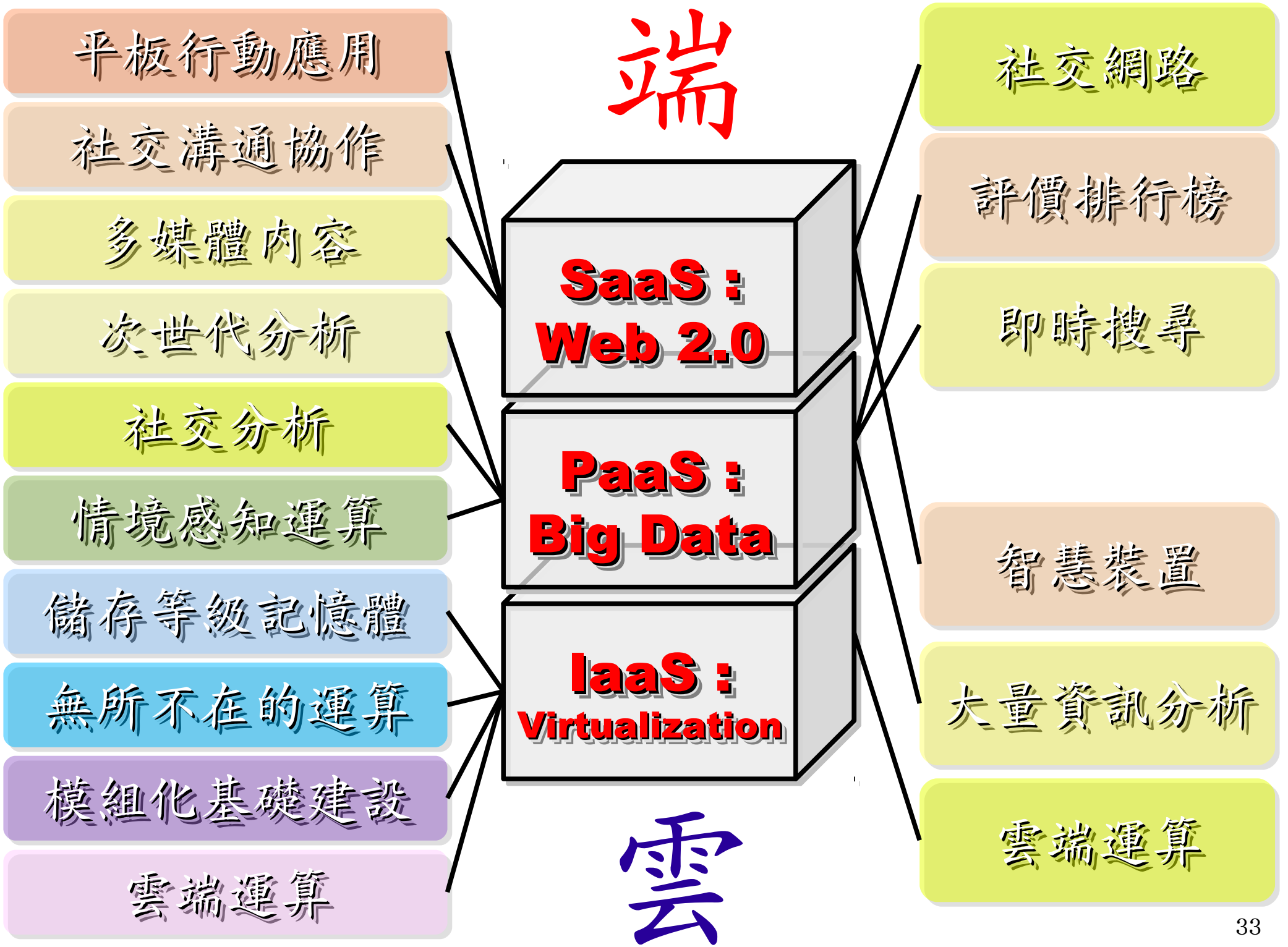
情境感知運算

儲存等級記憶體

無所不在的運算

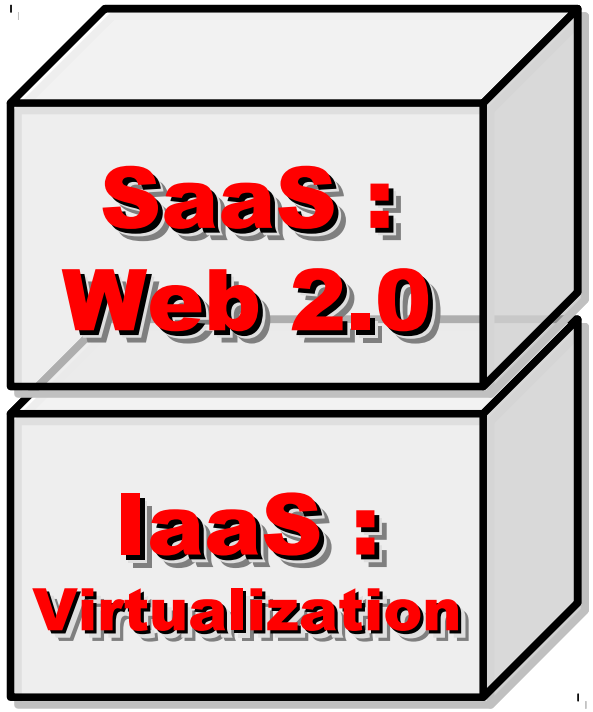
模組化基礎建設

Source ： http://www.gartner.com/it/page.jsp?id=1454221
Source ： http://www.digitimes.com.tw/tw/dt/n/shwnws.asp?Cnlid=4&cat=400&cat1=20&id=0000205798_CUZ63ZS3LCRY7E7UBK6V8


端
平板行動應用
社交溝通協作
多媒體內容
次世代分析
社交分析
情境感知運算
儲存等級記憶體
無所不在的運算
模組化基礎建設
雲端運算
SaaS :
Web 2.0
PaaS :
Big Data
IaaS :
Virtualization
社交網路
評價排行榜
即時搜尋
智慧裝置
大量資訊分析
雲端運算
雲

Two Type of Cloud Architecture ?

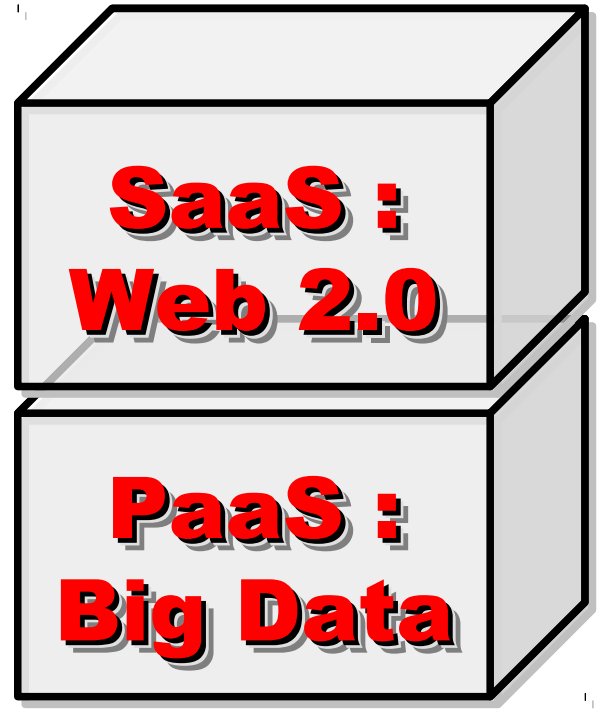
雲端架構的兩大陣營？



amazon web services™

想盡辦法誘你用計算跟網路

Computing Intensive



Google™

想盡辦法誘你提供資料作分析

Data Intensive

Reference Cloud Architecture
雲端運算的參考架構

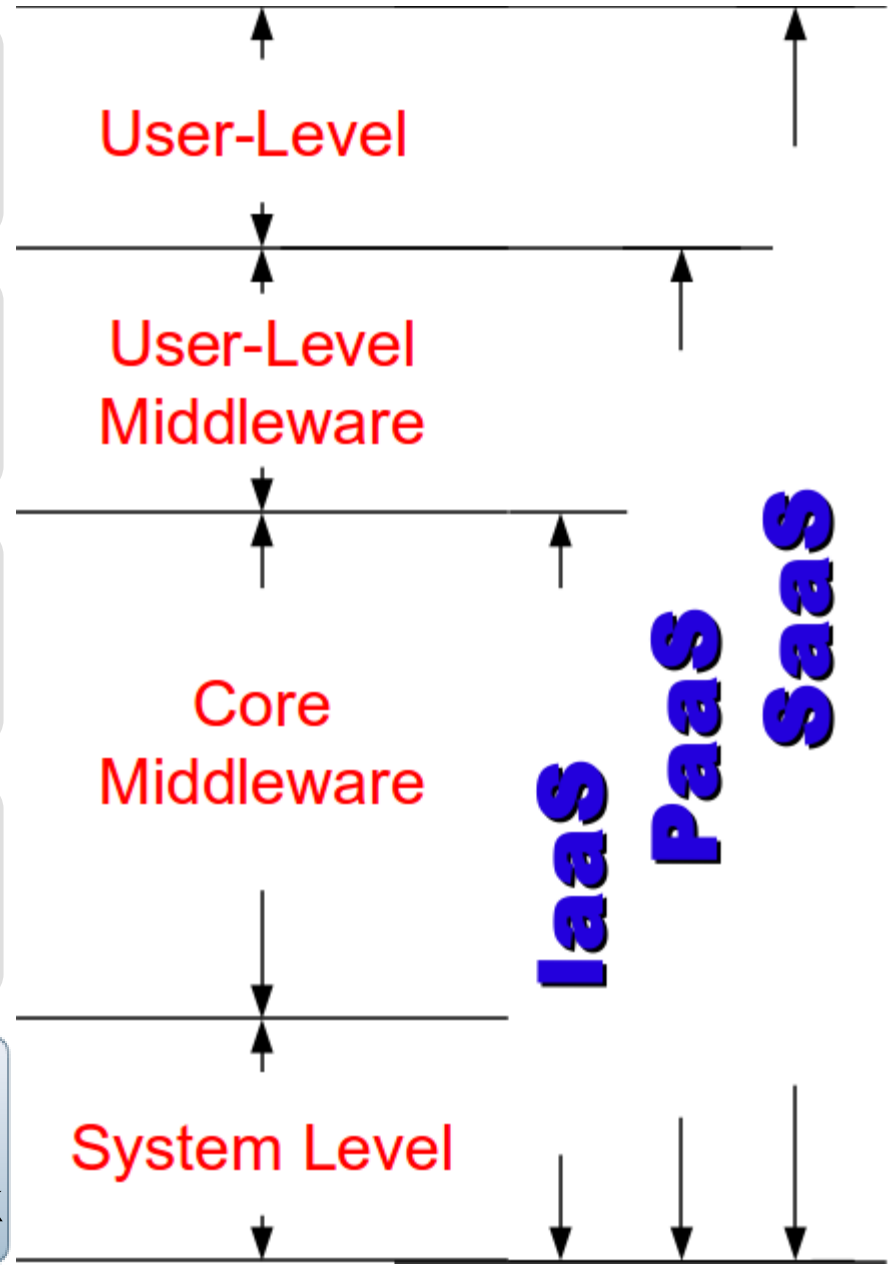
應用軟體 Application
Social Computing, Enterprise, ISV,...

程式語言 Programming
Web 2.0 介面, Mashups, Workflows, ...

控制管理 Control
Qos Neqotiation, Ddmission Control, Pricing, SLA Management, Metering...

虛擬化 Virtualization
VM, VM management and Deployment

硬體設施 Hardware
Infrastructure: Computer, Storage, Network



Open Source to build Cloud Service
建構雲端服務的自由軟體

應用軟體 Application
Social Computing, Enterprise, ISV,...

eyeOS, Nutch, ICAS, X-RIME, ...

程式語言 Programming
Web 2.0 介面 , Mashups, Workflows, ...

Hadoop (MapReduce), Sector/Sphere, AppScale

控制管理 Control
Qos Neqotiation, Ddmission Control, Pricing, SLA Management, Metering...

OpenNebula, Enomaly, Eucalyptus , OpenQRM, ...

虛擬化 Virtualization
VM, VM management and Deployment

Xen, KVM, VirtualBox, QEMU, OpenVZ, ...

Building IaaS with Open Source
用自由軟體打造 IaaS 服務

應用軟體 Application Social Computing, Enterprise, ISV,...	eyeOS, Nutch, ICAS, X-RIME, ...
程式語言 Programming Web 2.0 介面, Mashups, Workflows, ...	Hadoop (MapReduce), Sector/Sphere, AppScale
控制管理 Control Qos Neqotiation, Ddmission Control, Pricing, SLA Management, Metering...	OpenNebula, Enomaly, Eucalyptus , OpenQRM, ...
虛擬化 Virtualization VM, VM management and Deployment	Xen, KVM, VirtualBox, QEMU, OpenVZ, ...
硬體設施 Hardware Infrastructure: Computer, Storage, Network	

What is Virtualization ??
虛擬化技術有哪些呢 ??

- Application Virtualization 應用程式虛擬化
- Desktop Virtualization / Client Virtualization 桌面虛擬化
- Presentation Virtualization 顯示虛擬化
- OS-level Virtualization 作業系統虛擬化
- Network Virtualization 網路虛擬化
- Storage Virtualization 儲存虛擬化

Database Virtualization 資料庫虛擬化

Data Virtualization 資料虛擬化

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

Open Source for Virtualization
虛擬化技術對應的自由軟體

Application Virtualization 應用程式虛擬化	Ex. VMWare ThinApp
Desktop Virtualization 桌面虛擬化	Redhat SPICE
Presentation Virtualization 顯示虛擬化	VNC, FreeNX
OS-level Virtualization 作業系統虛擬化	Xen, KVM, OpenVZ
Network Virtualization 網路虛擬化	OpenFlow vSwitch
Storage Virtualization 儲存虛擬化	Lessfs, SDFS

NIST Mapping of Cloud Technologies
美國國家標準局的定義主要鎖定虛擬化技術

5. Measured Service
可被監控與量測的服務

1. On-demand self-service.
隨需自助服務

2. Broad network access
隨時隨地用任何網路裝置存取

3. Resource pooling
多人共享資源池

4. Rapid elasticity
快速重新佈署靈活度

Monitoring / AAA
狀態監控與認證收費機制

VM Management Tool
虛擬機器管理平台

OS-level Virtualization
作業系統虛擬化

Network Virtualization
網路虛擬化

Storage Virtualization
儲存虛擬化

NIST Mapping of Open Source Cloud
美國國家標準局的定義對應自由軟體技術

Monitoring / AAA 狀態監控與認證收費機制	Ganglia, Nagios / OpenID, SAML
VM Management Tool 虛擬機器管理平台	OpenNebula, Eucalyptus
OS-level Virtualization 作業系統虛擬化	Xen, KVM, OpenVZ
Network Virtualization 網路虛擬化	OpenFlow vSwitch
Storage Virtualization 儲存虛擬化	Lessfs, SDFS, ZFS

VMM Tool #1: Eucalyptus

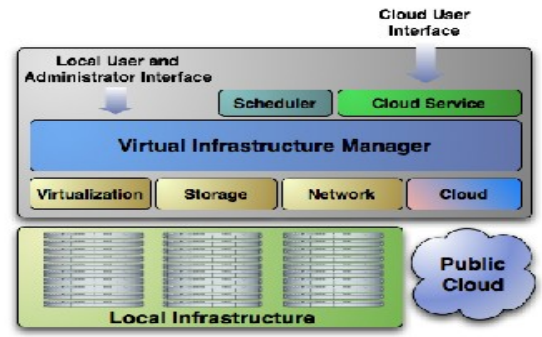
- 原是加州大學聖塔芭芭拉分校 (UCSB) 的研究專案
- **It was a research project of UCSB, USA**
- 目前已轉由 Eucalyptus System 這間公司負責維護
- **Now Eucalyptus System provide technical supports.**
- 創立目的是讓使用者可以**打造自己的 EC2**
- **It designed to help user to build their own Amazon EC2**
- 特色是相容於 Amazon EC2 既有的用户端介面
- **Its feature is compatible with existing EC2 client.**
- 優勢是 Ubuntu 9.04 已經收錄 Eucalyptus 的套件
- **Ubuntu Enterprise Cloud powered by Eucalyptus in 9.04**
- 目前有提供 Eucalyptus 的官方測試平台供註冊帳號
- **You can register trail account at http://open.eucalyptus.com/**
- 缺點：目前仍有部分操作需透過指令模式
- **Cons ： you might need to type commands in some case**

關於 Eucalyptus 的更多資訊，請參考
http://trac.nchc.org.tw/grid/wiki/Eucalyptus

VMM Tool #2: OpenNebula

- http://www.opennebula.org
- 由歐洲研究學會 (European Union FP7) 贊助
- **Sponsor by European Union FP7**
- 將實體叢集轉換成具管理彈性的虛擬基礎設備
- Turn Physical Cluster into Virtual Cluster
- 可管理虛擬叢集的狀態、排程、遷徙 (migration)
- manage status, scheduling and migration of virtual cluster
- Ubuntu 9.04 provide package of opennebula
- 缺點：需下指令來進行虛擬機器的遷徙 (migration) 。
- Cons ： You need to type commands to check or migration

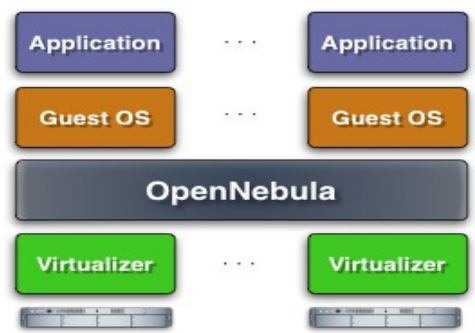
OpenNebula.org



SEVENTH FRAMEWORK PROGRAMME

RESERVOIR

關於 OpenNebula 的更多資訊，請參考 http://trac.nchc.org.tw/grid/wiki/OpenNEbula



Building PaaS with Open Source
用自由軟體打造 PaaS 雲端服務

應用軟體 Application Social Computing, Enterprise, ISV,....	eyeOS, Nutch, ICAS, X-RIME, ...
程式語言 Programming Web 2.0 介面, Mashups, Workflows, ...	Hadoop (MapReduce), Sector/Sphere, AppScale
控制管理 Control Qos Negotiation, Admission Control, Pricing, SLA Management, Metering,...	OpenNebula, Enomaly, Eucalyptus , OpenQRM, ...
虛擬化 Virtualization VM, VM management and Deployment	Xen, KVM, VirtualBox, QEMU, OpenVZ, ...

Three Core Technologies of Google ....
Google 的三大關鍵技術 ....

- 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

Open Source Mapping of Google Core Technologies
Google 三大關鍵技術對應的自由軟體

BigTable A huge key-value datastore	HBase, Hypertable Cassandra, ...
MapReduce To parallel process data	Hadoop MapReduce API Sphere MapReduce API, ...
Google File System To store petabytes of data	Hadoop Distributed File System (HDFS) Sector Distributed File System

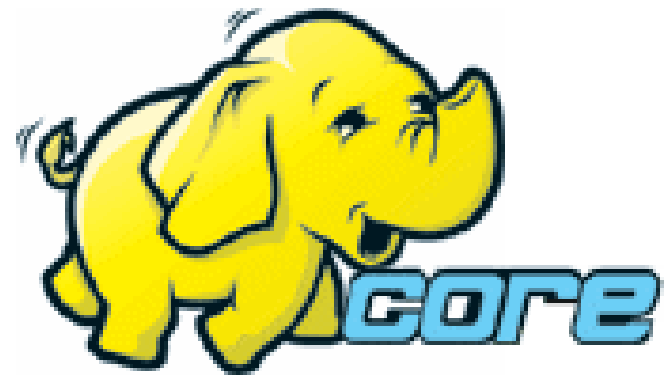
更多不同語言的 MapReduce API 實作：
http://trac.nchc.org.tw/grid/intertrac/wiki%3Ajazz/09-04-14%23MapReduce

其他值得觀察的分散式檔案系統：

- IBM GPFS - http://www-03.ibm.com/systems/software/gpfs/
- Lustre - http://www.lustre.org/
- Ceph - http://ceph.newdream.net/

Hadoop

- http://hadoop.apache.org
- Hadoop 是 Apache Top Level 開發專案
- **Hadoop is Apache Top Level Project**
- 目前主要由 Yahoo! 資助、開發與運用
- **Major sponsor is Yahoo!**
- 創始者是 Doug Cutting ，參考 Google Filesystem
- **Developed by Doug Cutting, Reference from Google Filesystem**
- 以 Java 開發，提供 HDFS 與 MapReduce API 。
- **Written by Java, it provides HDFS and MapReduce API**
- 2006 年使用在 Yahoo 內部服務中
- **Used in Yahoo since year 2006**
- 已佈署於上千個節點。
- **It had been deploy to 4000+ nodes in Yahoo**
- 處理 Petabyte 等級資料量。
- **Design to process dataset in Petabyte**



Facebook、**Last.fm**、**Joost** are also powered by Hadoop

Sector / Sphere

- http://sector.sourceforge.net/
- 由美國資料探勘中心研發的自由軟體專案。
- **Developed by National Center for Data Mining, USA**
- 採用 C/C++ 語言撰寫，因此效能較 Hadoop 更好。
- **Written by C/C++, so performance is better than Hadoop**
- 提供「類似」 Google File System 與 MapReduce 的機制
- **Provide file system similar to Google File System and MapReduce API**
- 基於UDT高效率網路協定來加速資料傳輸效率
- **Based on UDT which enhance the network performance**
- Open Cloud Testbed有提供測試環境，並開發MalStone效能評比軟體
- **Open Cloud Consortium provide Open Cloud Testbed and develop MalStone toolkit for benchmark**

National Center for Data Mining
University of Illinois at Chicago

Open Data Group
http://www.opendatagroup.com/

What we learn today ?

WHAT	隨時隨地用任何裝置存取各種服務!! Accessing services with any device anytime anywhere!!
WHO	亞馬遜、谷歌、微軟等! 什麼都可以是服務 ~ Amazon, Google, Microsoft and more! Everything as a Service!
WHEN	雲端運算是2006年繼格網運算之後的新趨勢!! Cloud Computing become new trend since year 2007 !!
WHY	資料爆炸、節省成本、行動應用 Data-intensive, Cost-Efficency, Mobile Applications
HOW	採用自由軟體也能打造私有雲端 Hadoop, Sector/Sphere, Eucalyptus, and more ....

Questions?

Slides - http://trac.nchc.org.tw/cloud

Jazz Wang

Yao-Tsung Wang

jazz@nchc.org.tw

Powered by DRBL

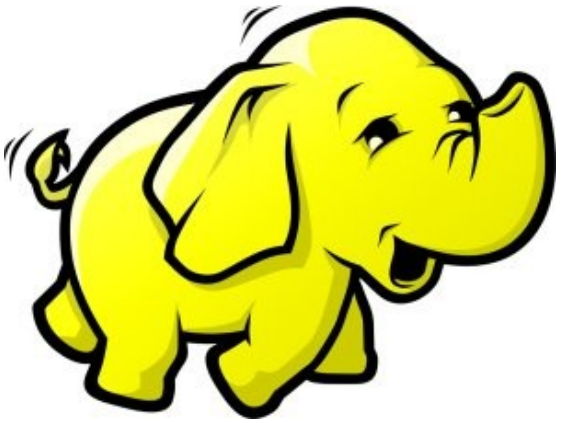
Hadoop 簡介：源起與術語

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

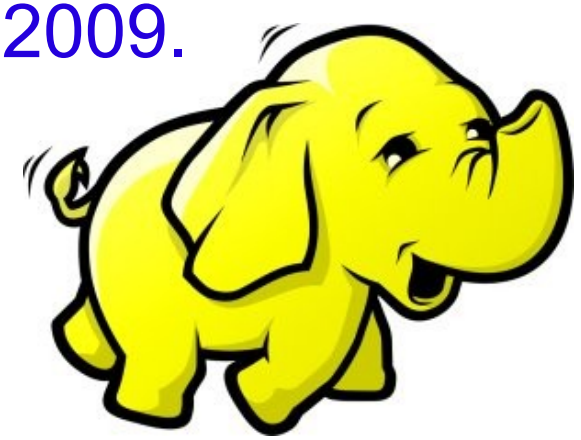
History of Hadoop … 2004 ~ Now

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

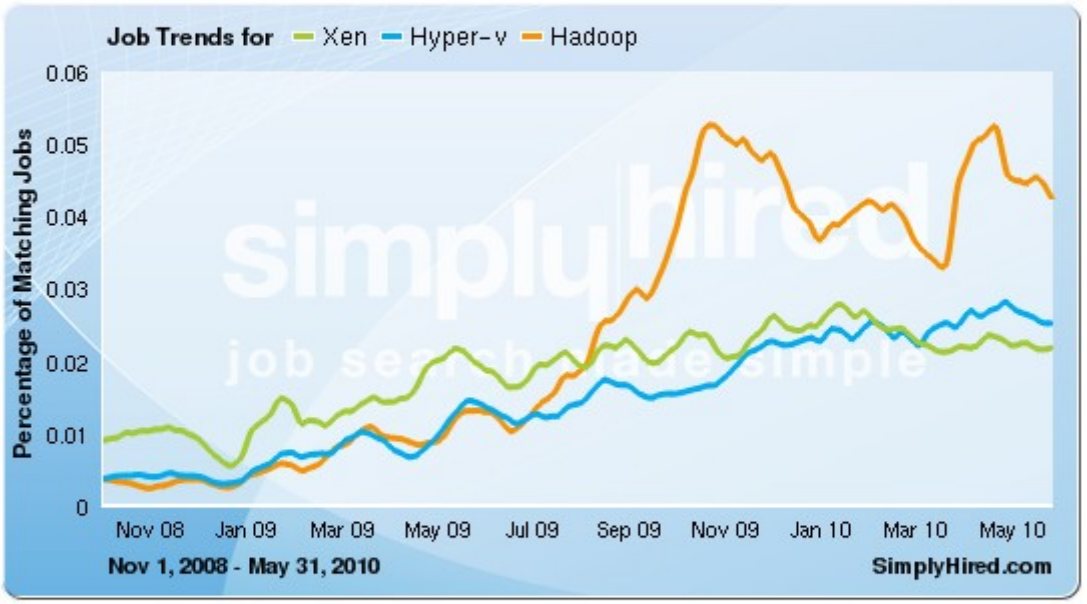
simplyhired®
job search made simple

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

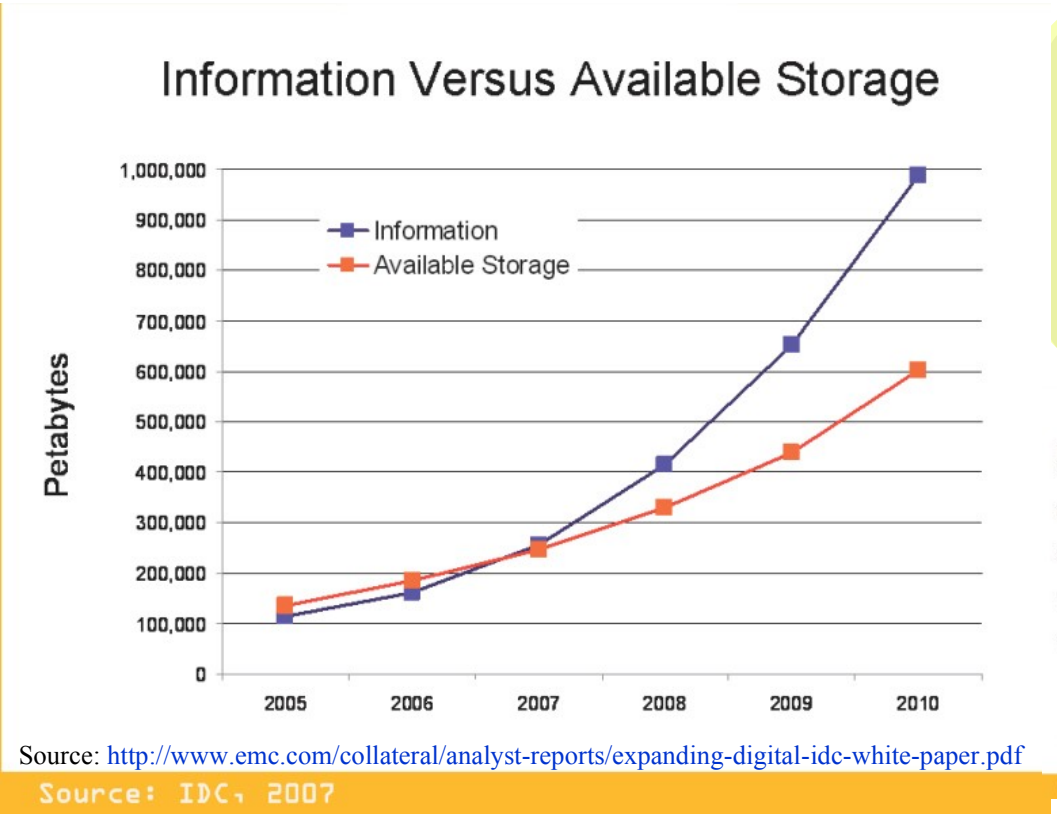
This graph displays the percentage of jobs with your search terms anywhere in the job listing. Since November 2008, the following has occurred:

- 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

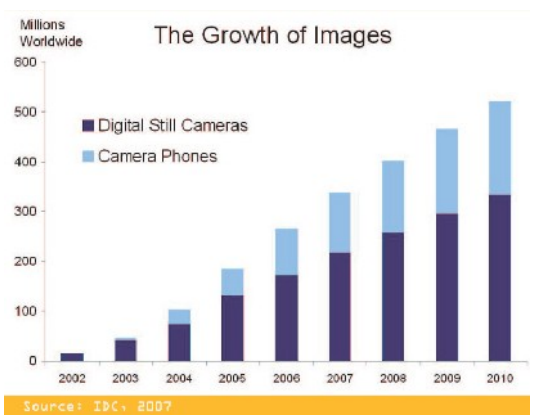
3. Looking for Jobs
好找工作 !!

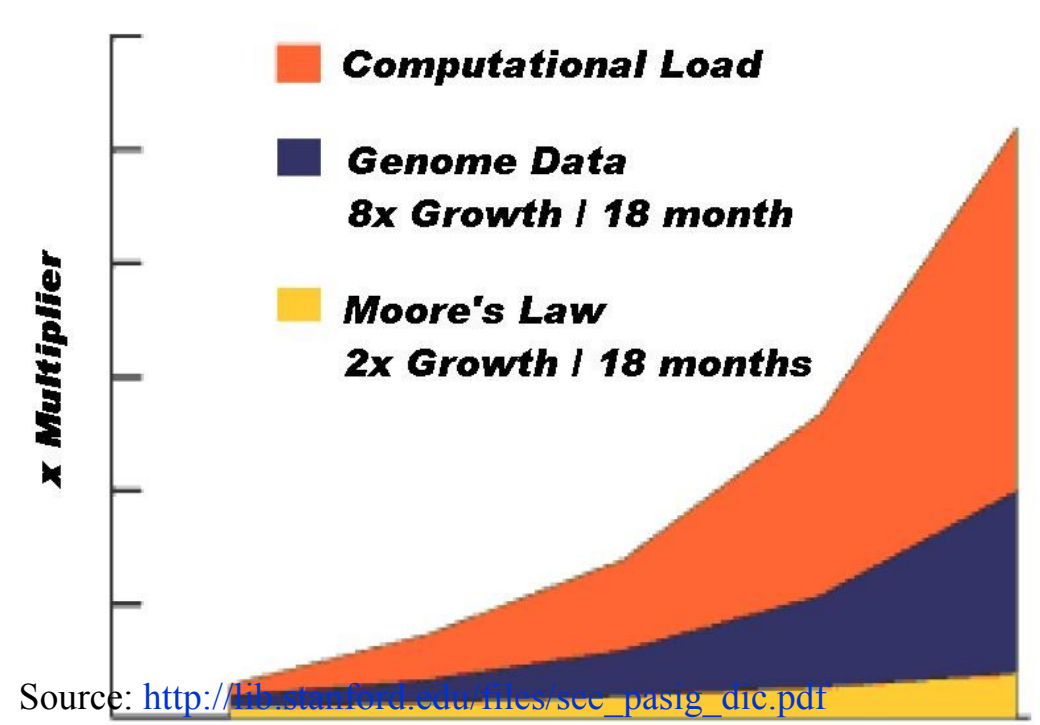


Source: http://www.emc.com/collateral/analyst-reports/expanding-digital-idc-white-paper.pdf

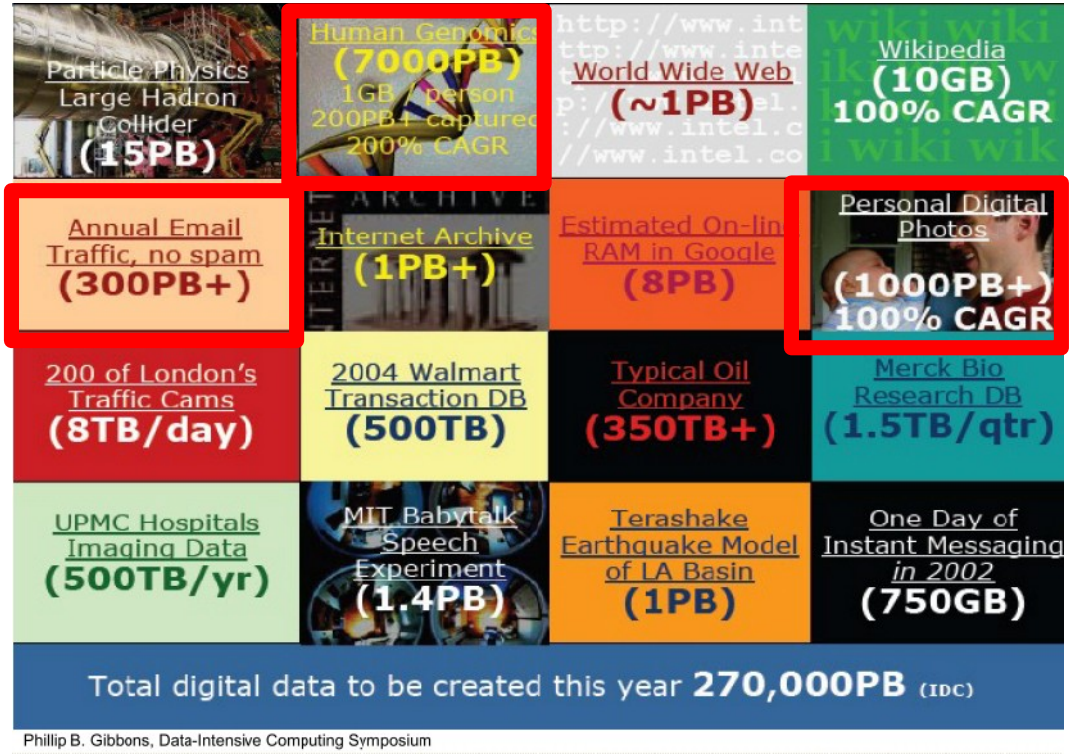
2007 Data Explore

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



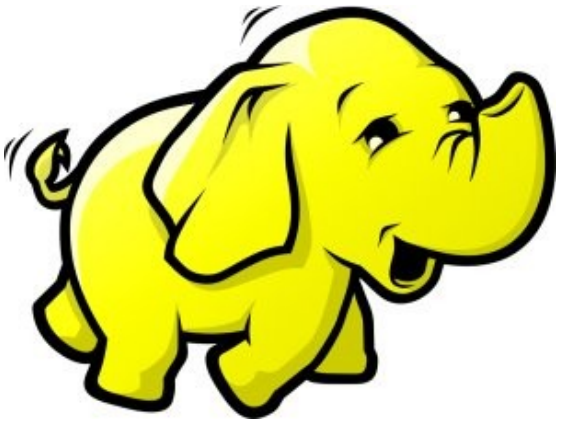
Hadoop 專業術語

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

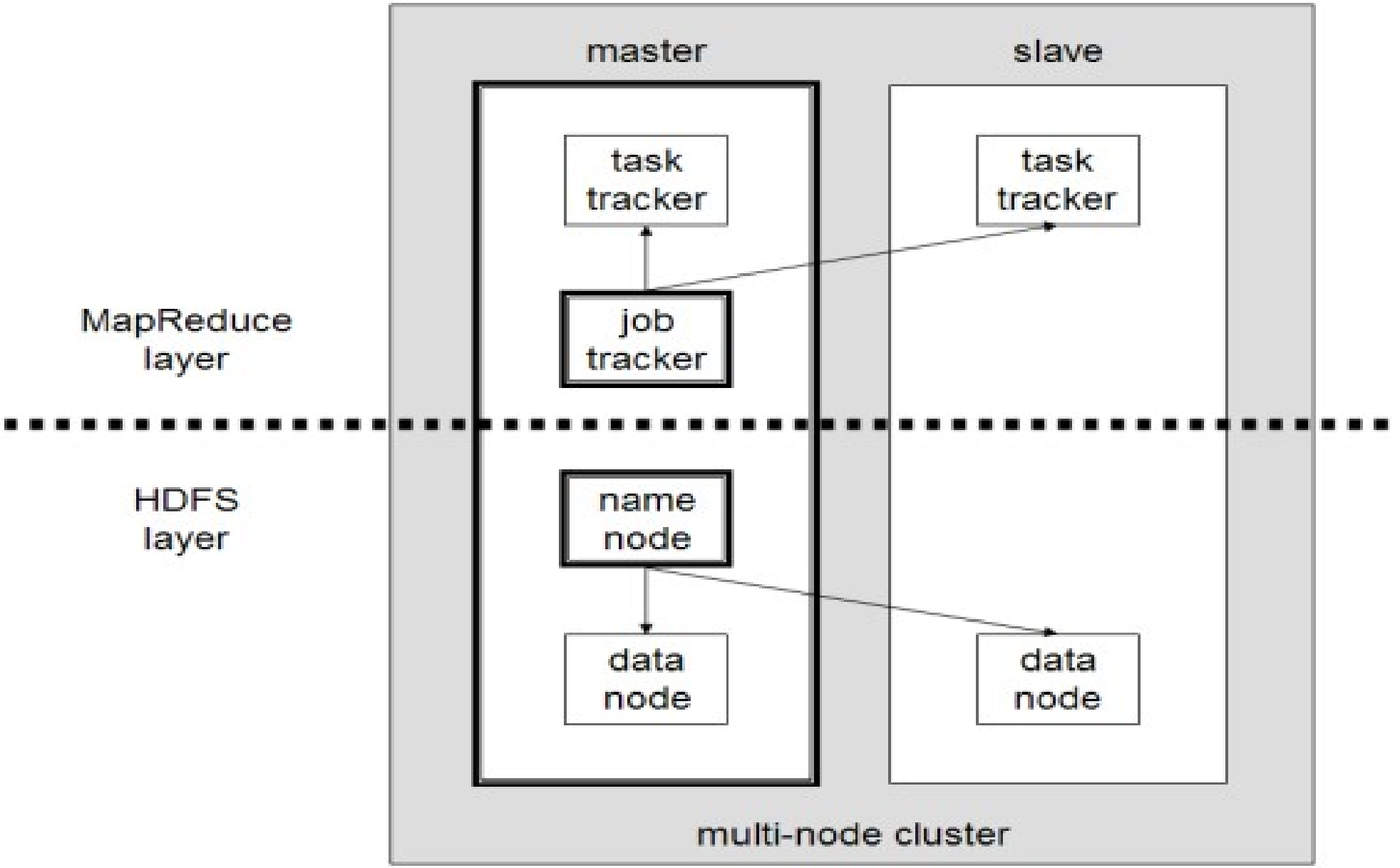
JobTracker

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

TaskTracker

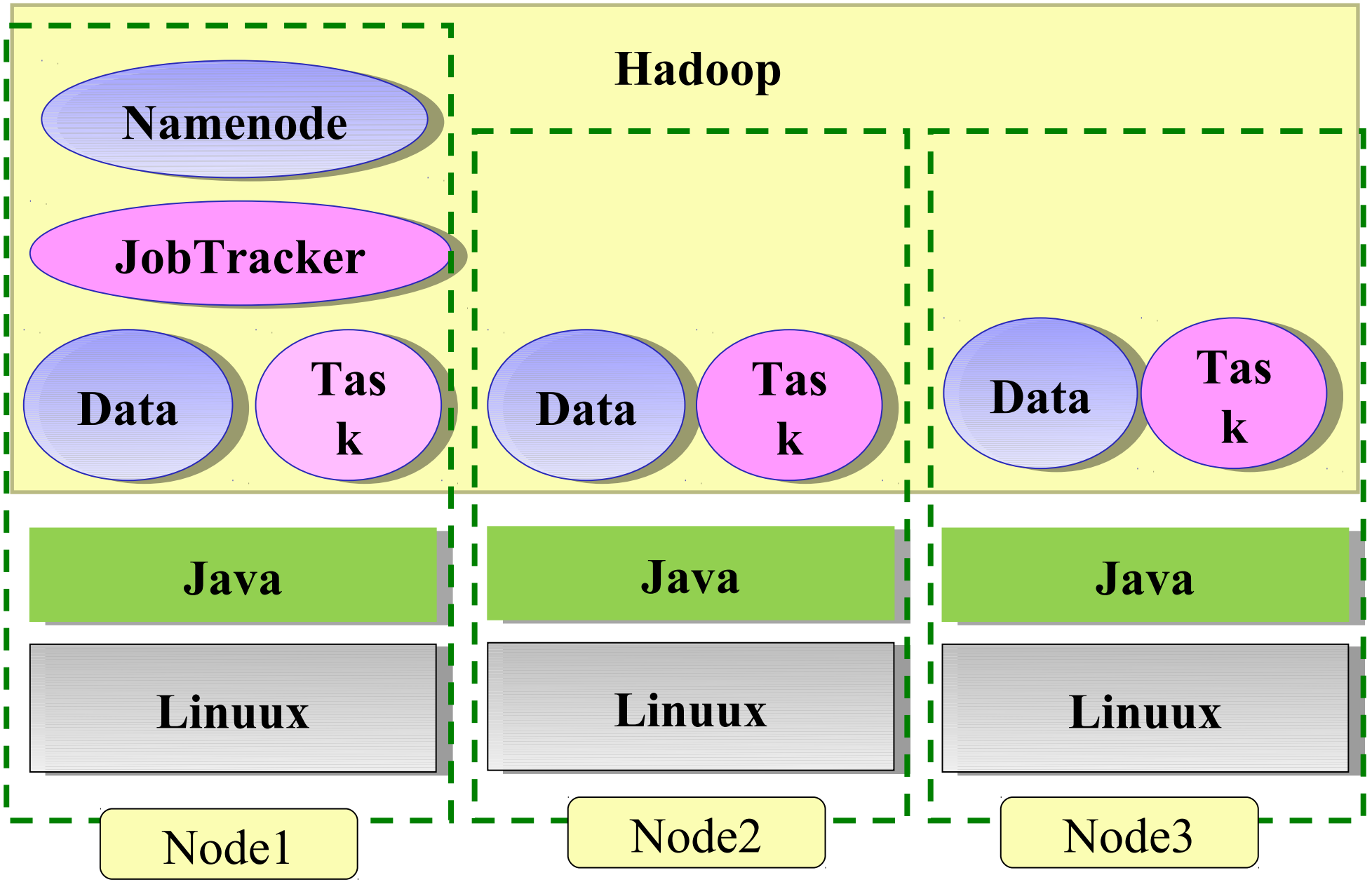
- **Worker Nodes**
- **Excute Mapper and Reducer Tasks**
- **Save Results and report task status**
- **Multiple Nodes**

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

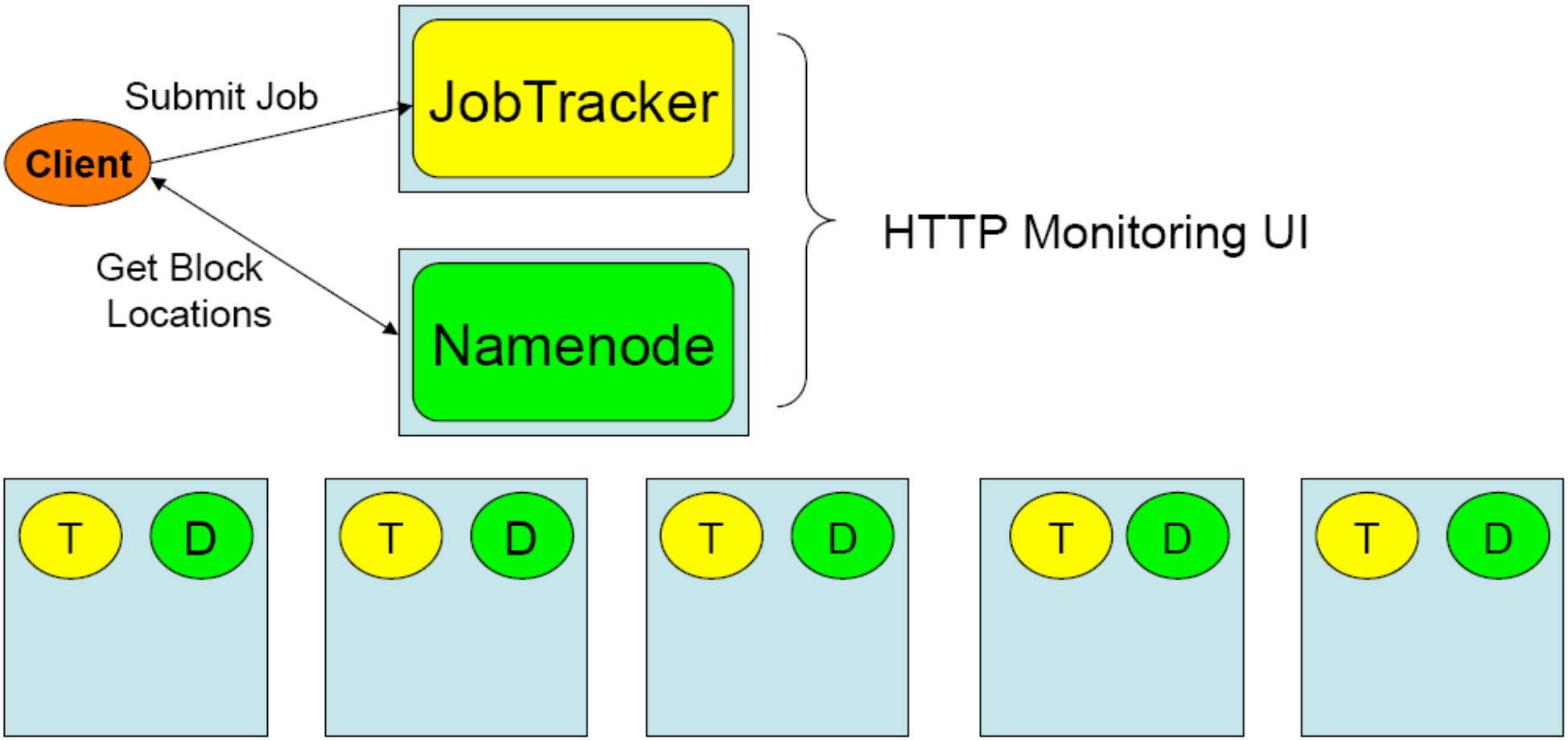


Distributed Operating System of Hadoop

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 !!

Questions?

Slides - http://trac.nchc.org.tw/cloud

Jazz Wang
Yao-Tsung Wang
jazz@nchc.org.tw

Powered by DRBL

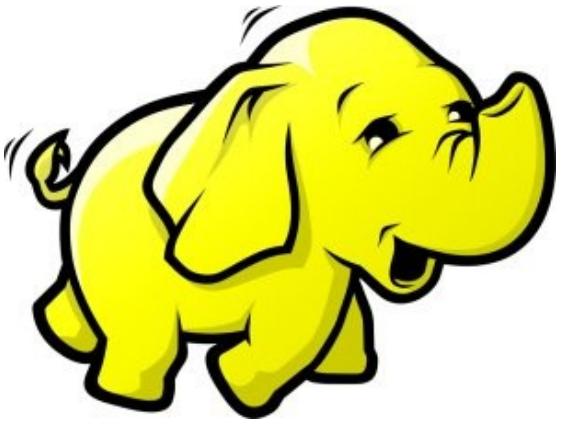
HDFS 簡介

Introduction to Hadoop Distributed File System

Jazz Wang

Yao-Tsung Wang

jazz@nchc.org.tw



What is HDFS ??
什麼是 HDFS ??

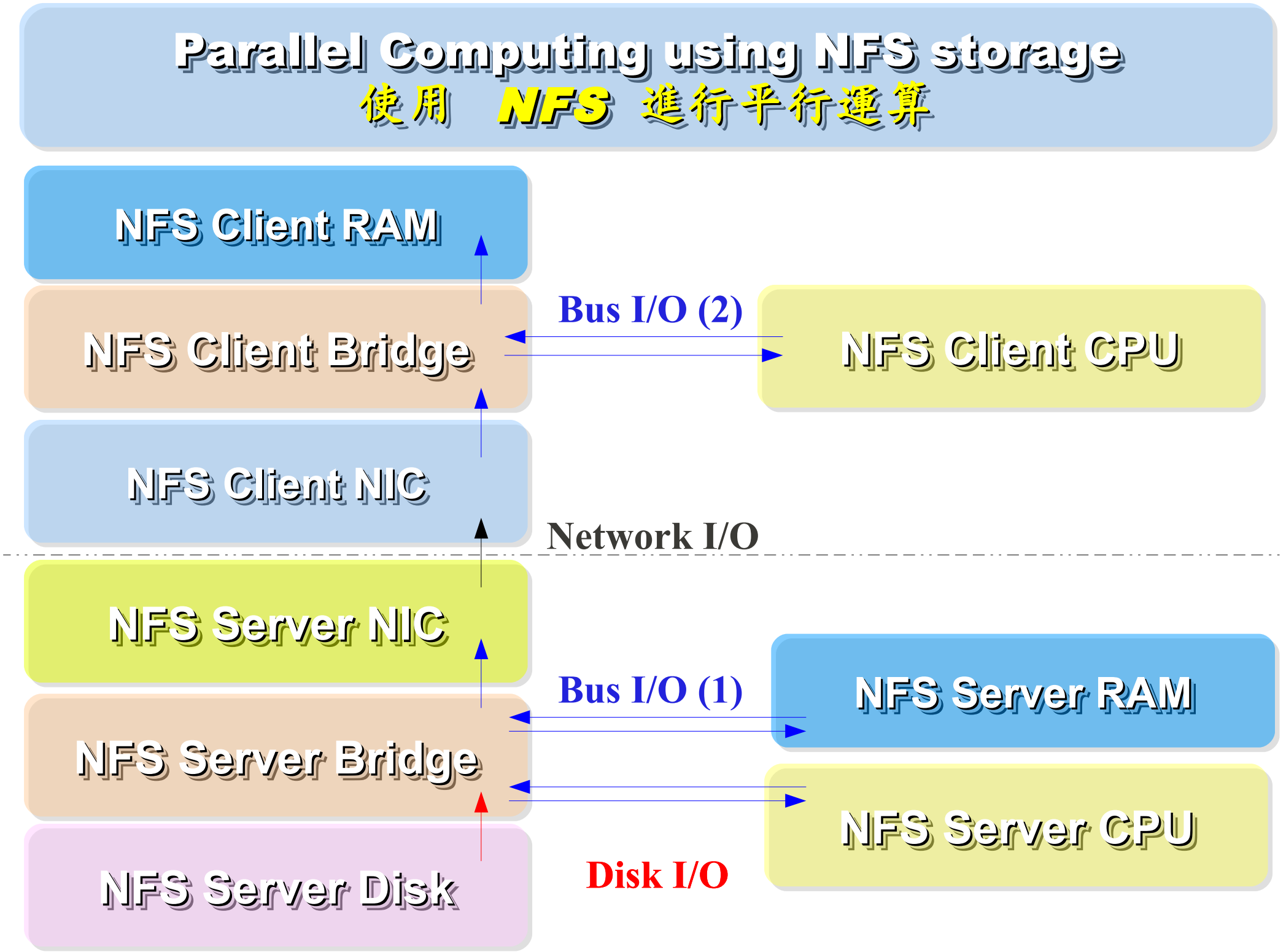
- **Hadoop Distributed File System**
  - 實現類似 Google File System 分散式檔案系統
  - Reference from Google File System.
  - 一個易於擴充的分散式檔案系統，目的爲對大量資料進行分析
  - A scalable distributed file system for large data analysis .
  - 運作於廉價的普通硬體上，又可以提供容錯功能
  - based on commodity hardware with high fault-tolerant.
  - 給大量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 ...

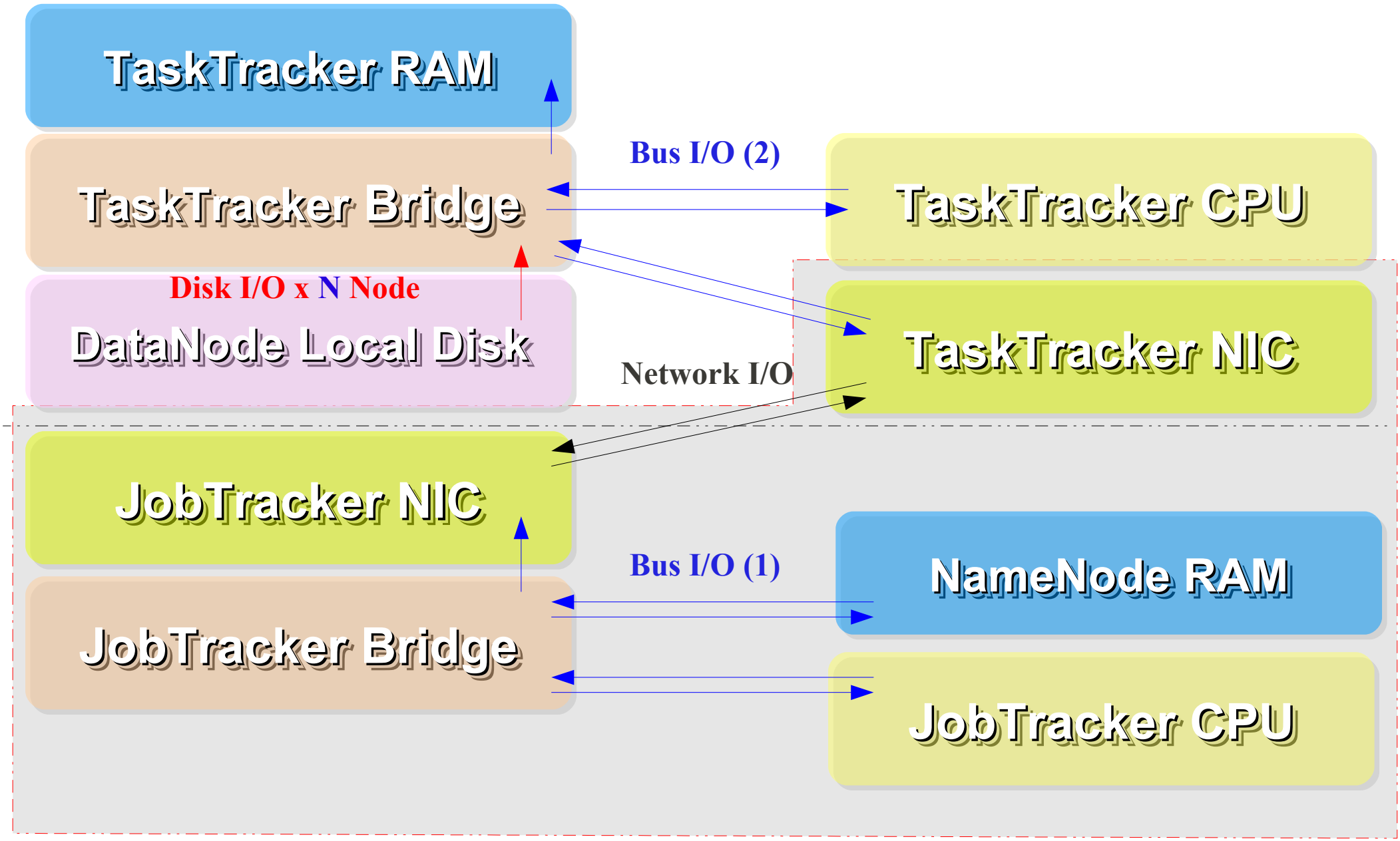
- **硬體錯誤容忍能力 Fault Tolerance**
  - 硬體錯誤是正常而非異常
  - Failure is the norm rather than exception
  - 自動恢復或故障排除
  - automatic recovery or report failure
- **串流式的資料存取 Streaming data access**
  - 批次處理多於用户交互處理
  - Batch processing rather than interactive user access.
  - 高 Throughput 而非低 Latency
  - High aggregate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

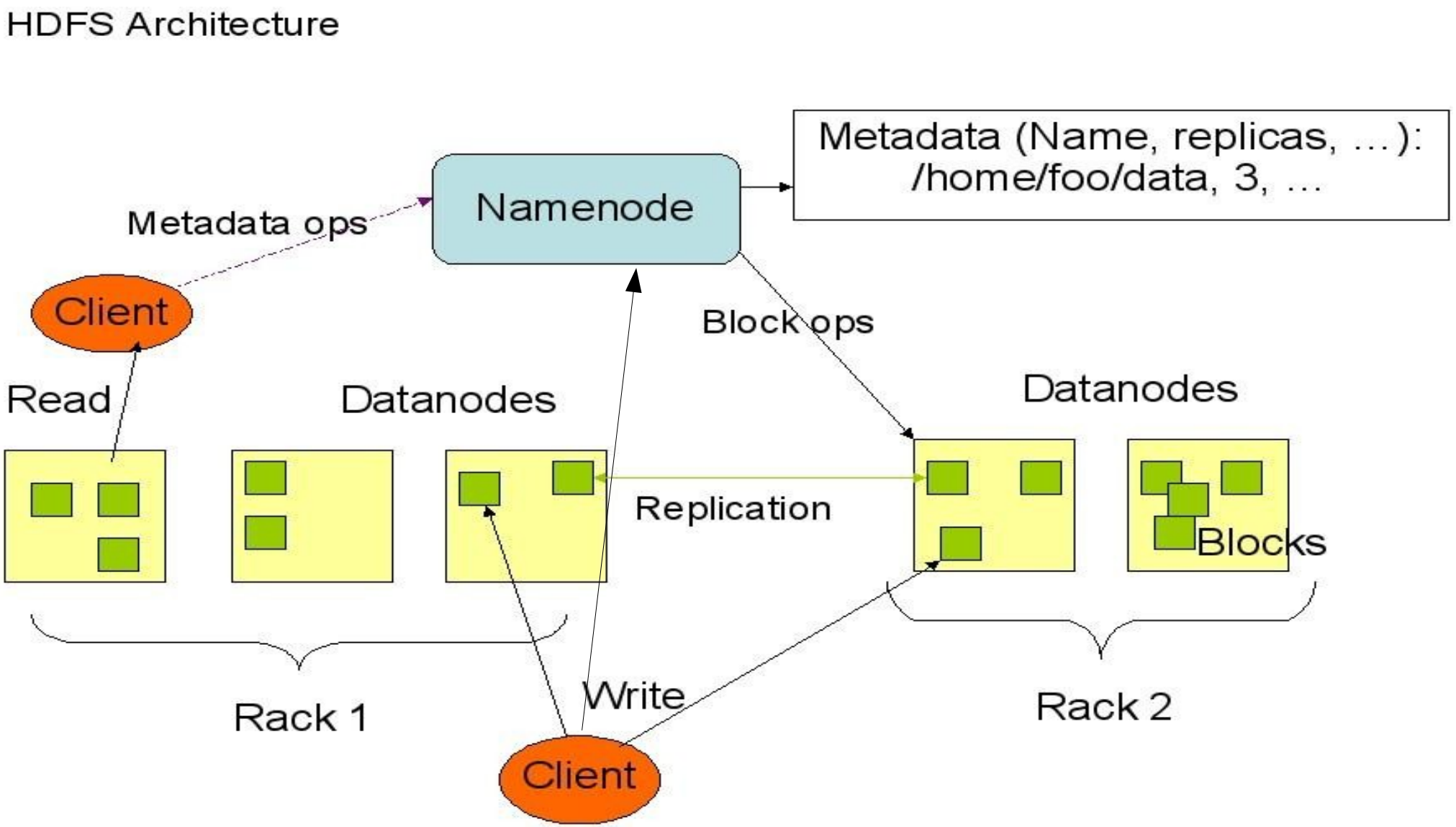

Parallel Computing using NFS storage
使用 NFS 進行平行運算
NFS Client RAM
NFS Client Bridge
NFS Client NIC
Bus I/O (2)
NFS Client CPU
Network I/O
NFS Server NIC
Bus I/O (1)
NFS Server RAM
NFS Server Bridge
NFS Server CPU
NFS Server Disk
Disk I/O

Parallel Computing using HDFS
使用 HDFS 進行平行運算



How HDFS manage data ...

HDFS 如何管理資料 ...



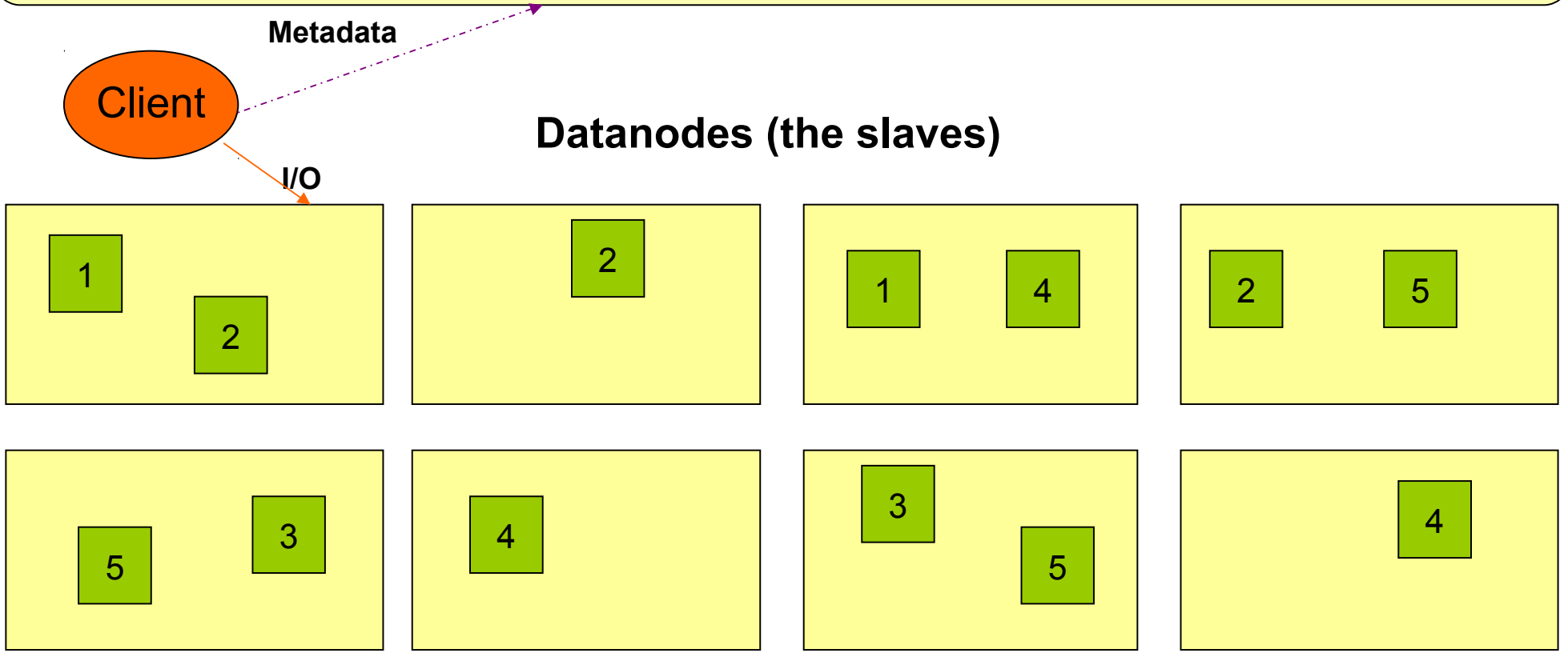
How does HDFS work ...

HDFS 如何運作 ...

Namenode (the master)

Path and Filename – Replication , blocks

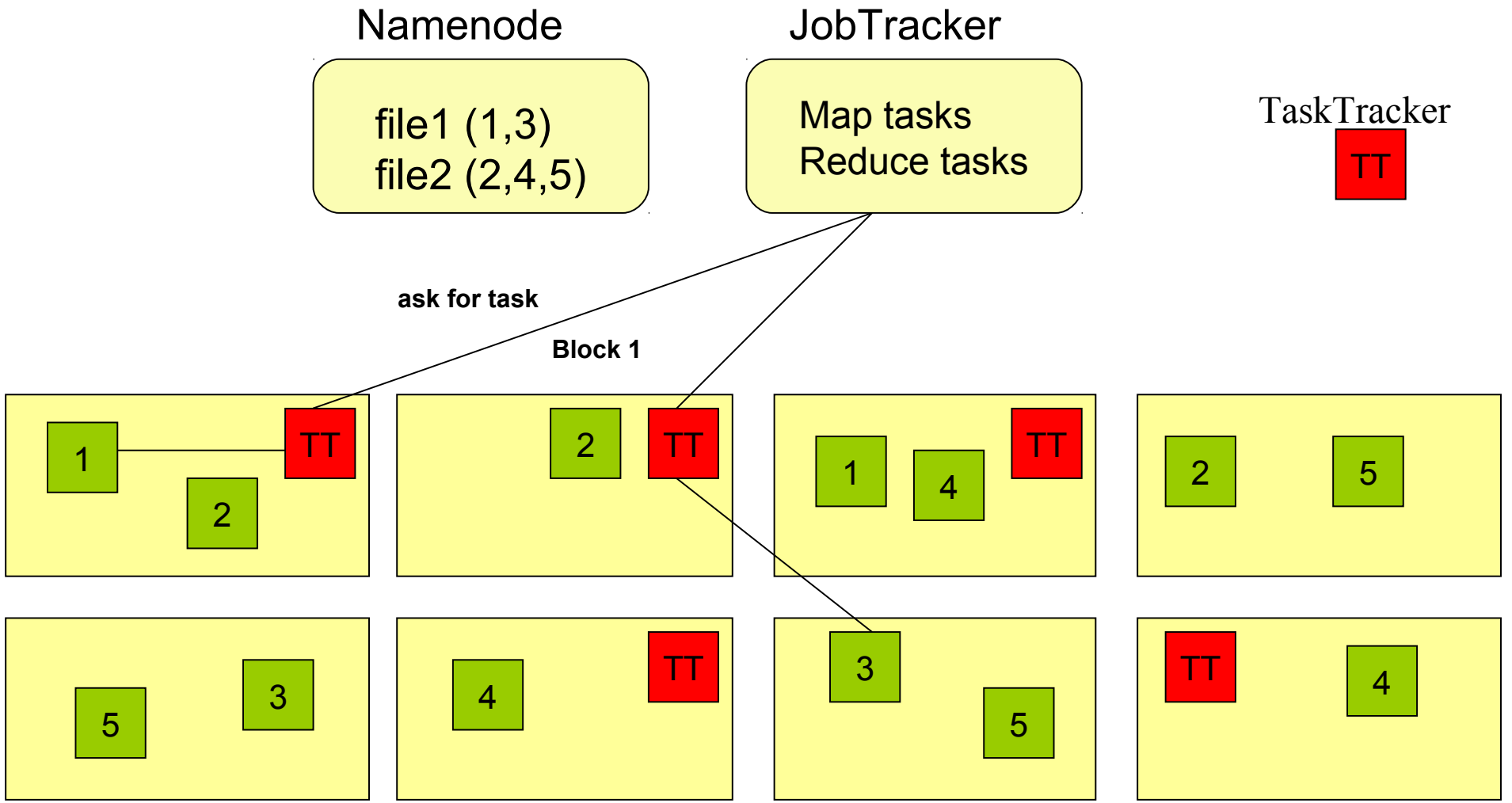
name:/users/joeYahoo/myFile - copies:2, blocks:{1,3}
name:/users/bobYahoo/someData.gzip, copies:3, blocks:{2,4,5}



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]]
```

Questions?

Slides - http://trac.nchc.org.tw/cloud

Jazz Wang
Yao-Tsung Wang
jazz@nchc.org.tw

Powered by DRBL

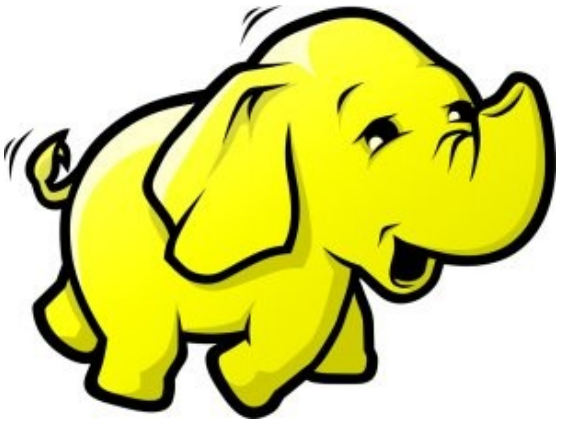
MapReduce 簡介

Introduction to MapReduce

Jazz Wang

Yao-Tsung Wang

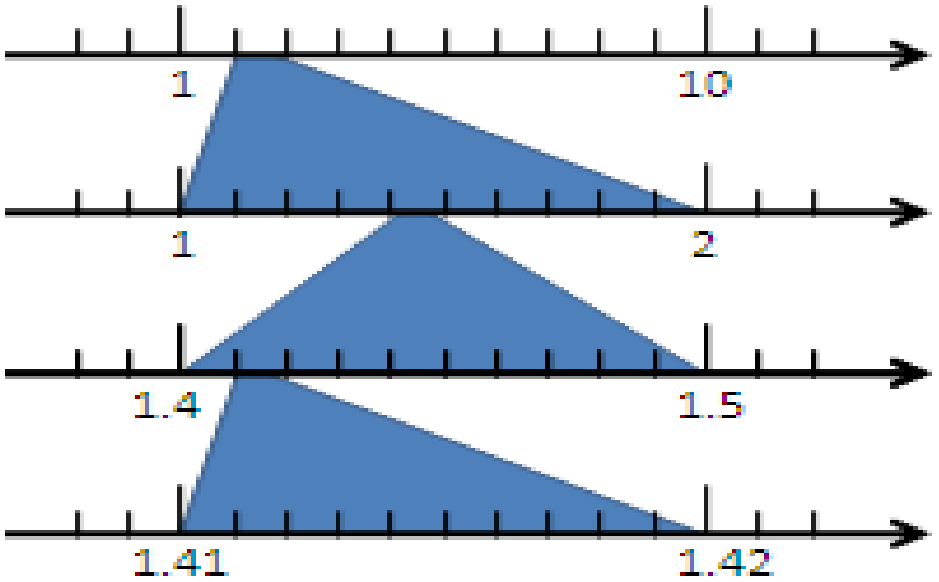
jazz@nchc.org.tw



Divide and Conquer Algorithms

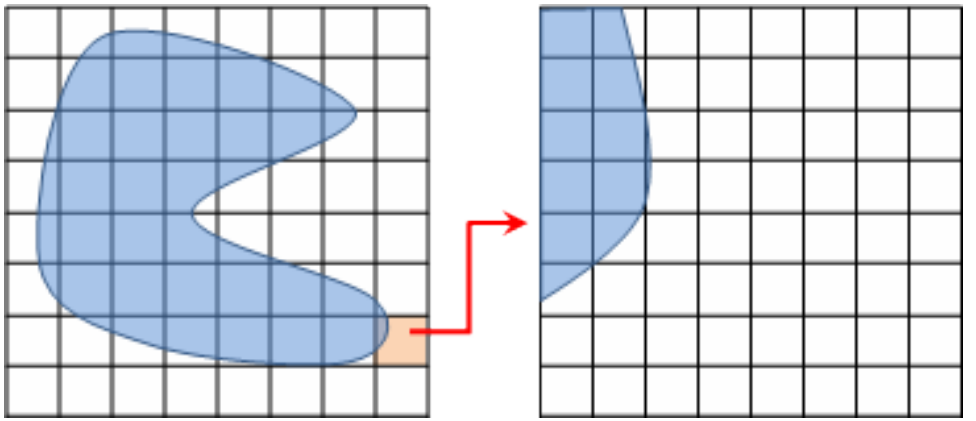
分而治之演算法

Example 1:

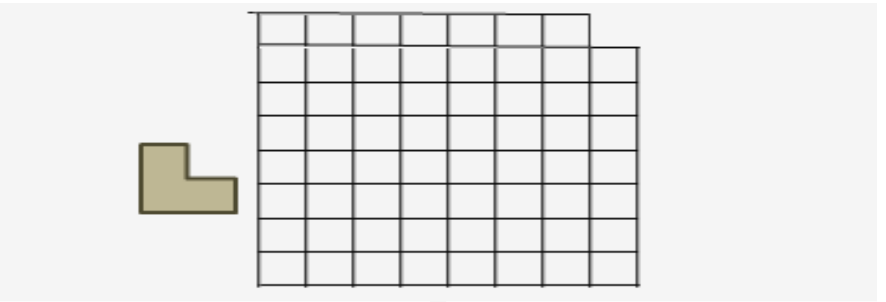


sqrt(2)

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)

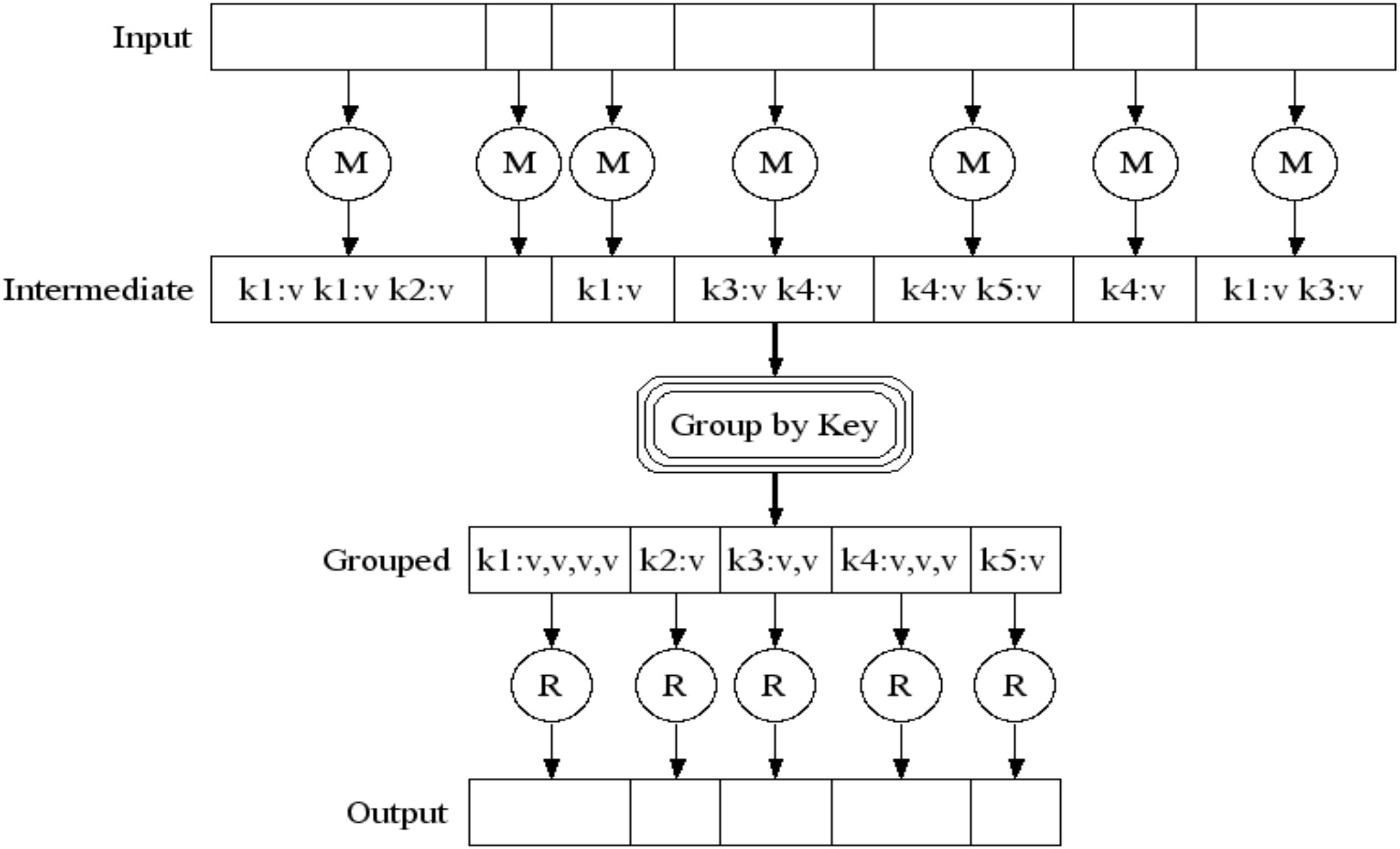
What is MapReduce ??
什麼是 MapReduce ??

- 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

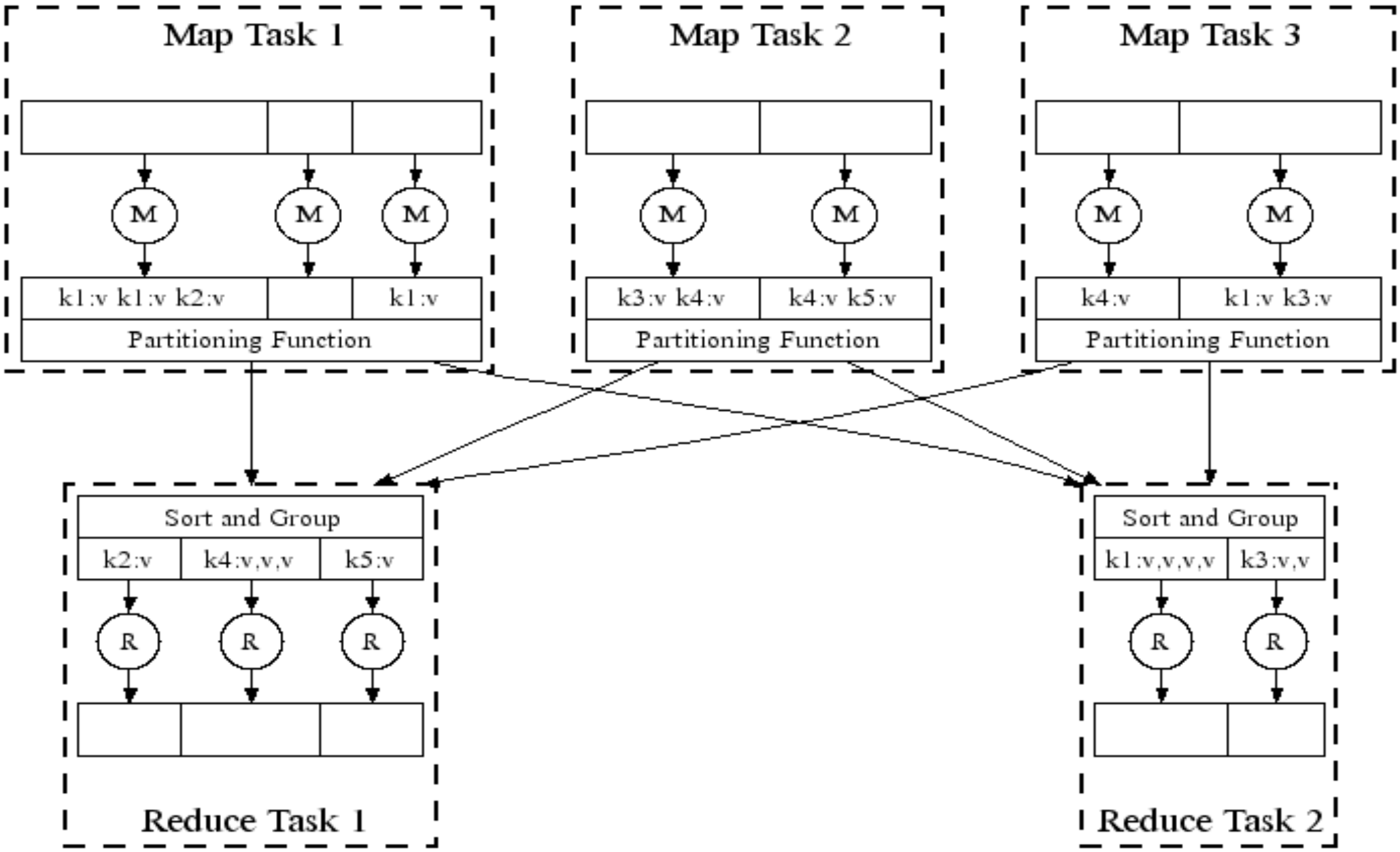
Google's MapReduce Diagram

Google 的 MapReduce 圖解



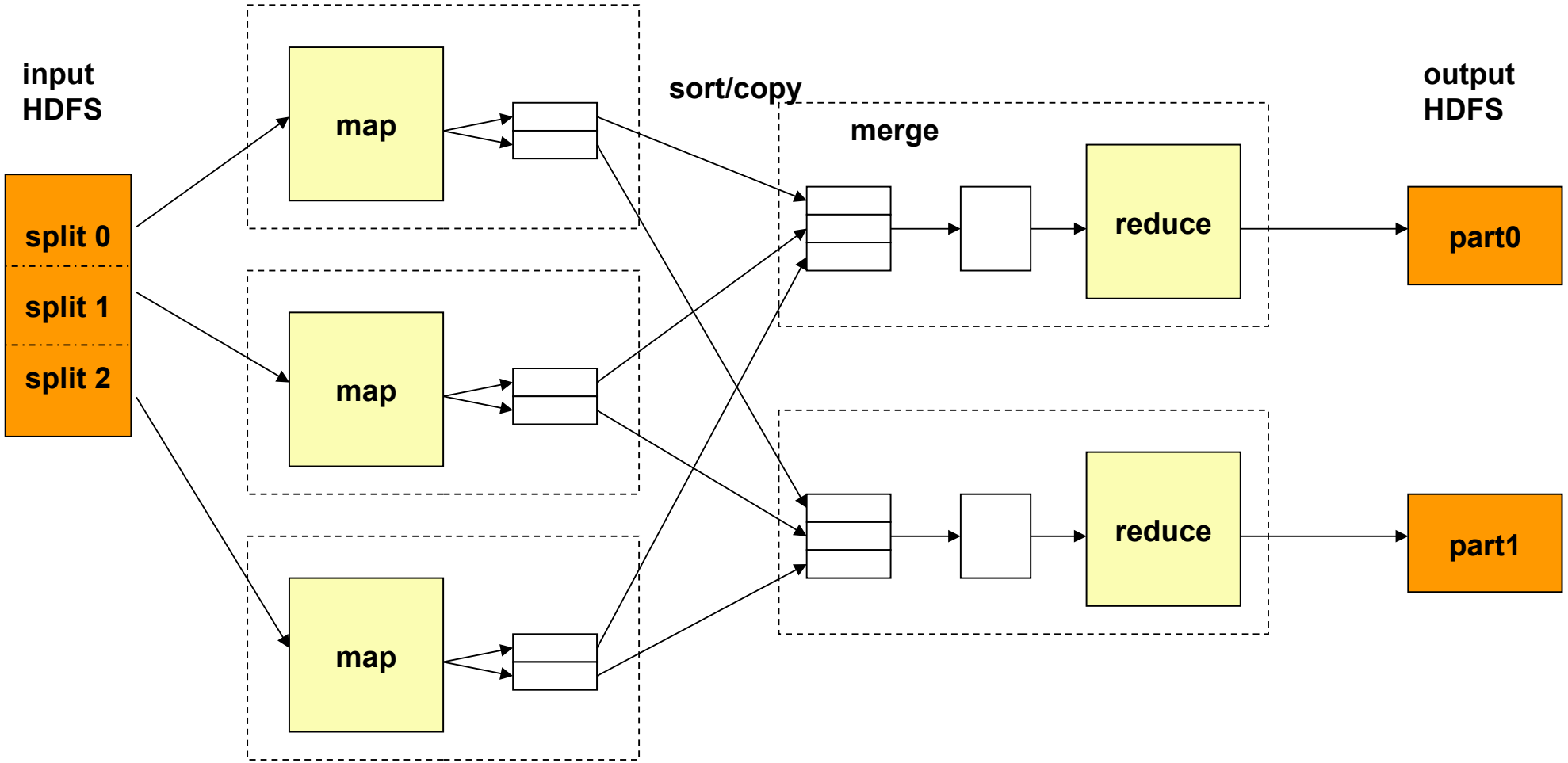
Google's MapReduce in Parallel

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

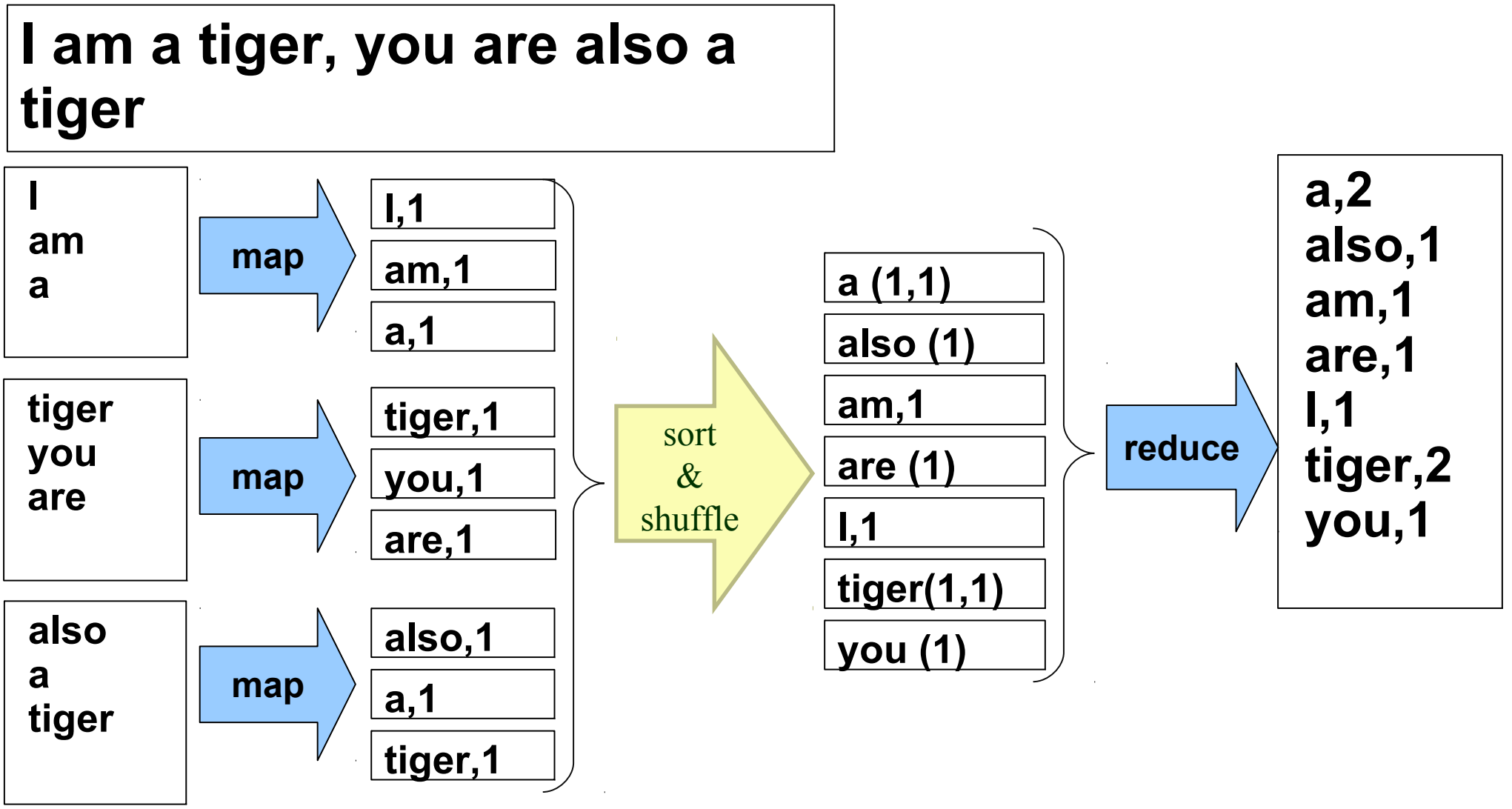
JobTracker 將中間檔案整合排序後，複製到需要的 TaskTracker 去

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

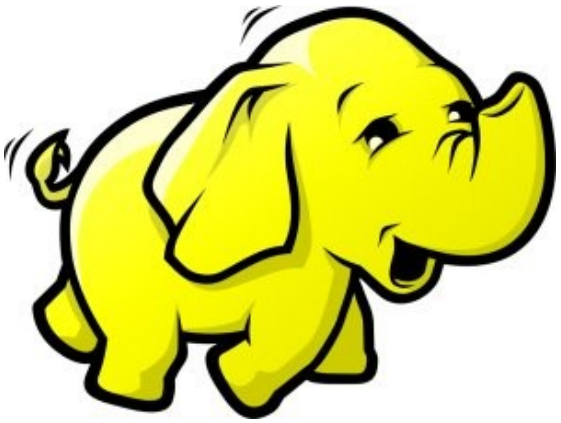
MapReduce 程式設計入門

MapReduce Programing 101

Jazz Wang

Yao-Tsung Wang

jazz@nchc.org.tw



Program Prototype (v 0.20)

Map 區

Reduce 區

設定區

```
Class MR{
    static public Class Mapper …{
    }
    static public Class Reducer …{
    }
    main(){
        Configuration conf = new Configuration();
        Job job = new Job(conf, “job name");
        job.setJarByClass(thisMainClass.class);
        job.setMapperClass(Mapper.class);
        job.setReduceClass(Reducer.class);
        FileInputFormat.addInputPaths(job, new Path(args[0]));
        FileOutputFormat.setOutputPath(job, new Path(args[1]));
        job.waitForCompletion(true);
}}
```

Mapper 程式碼

Reducer 程式碼

Other configure 其他的設定參數程式碼

Program Prototype (v 0.18)

Map 區

Reduce 區

設定區

```
Class MR{
    static public Class Mapper ...{
    }
    static public Class Reducer ...{
    }
    main(){
        JobConf conf = new JobConf( MR.class );
        conf.setMapperClass(Mapper.class);
        conf.setReduceClass(Reducer.class);
        FileInputFormat.setInputPaths(conf, new Path(args[0]));
        FileOutputFormat.setOutputPath(conf, new Path(args[1]));
        JobClient.runJob(conf);
}}
```

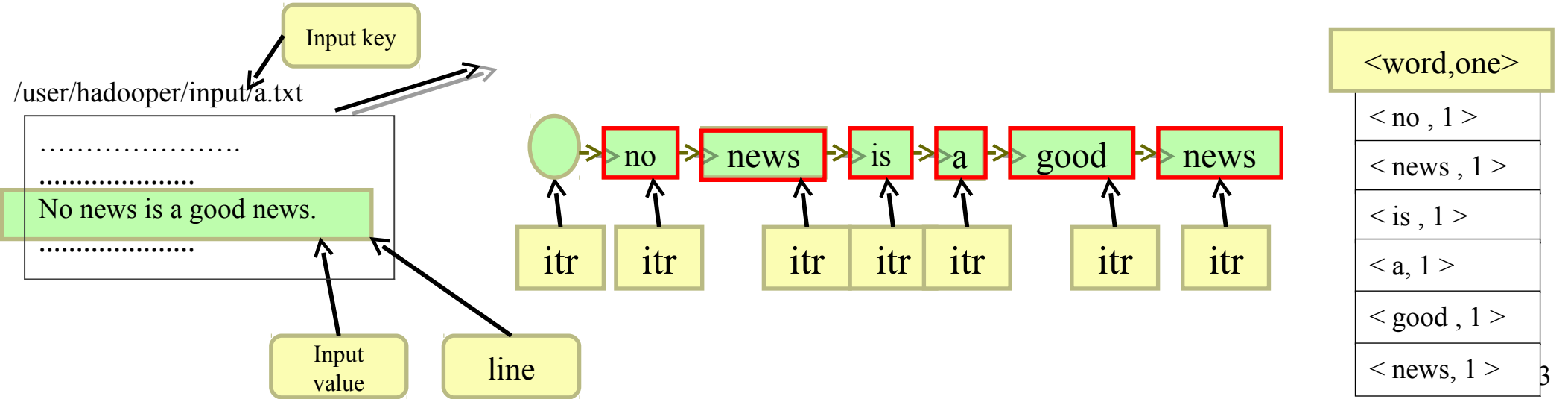
Map 程式碼

Reduce 程式碼

Other configure 其他的設定參數程式碼

Word Count - mapper

```
1  class MyMapper extends Mapper<LongWritable, Text, Text, IntWritable> {
       private final static IntWritable one = new IntWritable(1);
2      private Text word = new Text();
3      public void map( LongWritable key, Text value, Context context)
4      throws IOException , InterruptedException {
           String line = ((Text) value).toString();
5          StringTokenizer itr = new StringTokenizer(line);
6          while (itr.hasMoreTokens()) {
7              word.set(itr.nextToken());
8              context.write(word, one);
9  }}}
```



Word Count - reducer

```
1  class MyReducer extends Reducer< Text, IntWritable, Text, IntWritable> {
2          IntWritable result = new IntWritable();
3          public void reduce( Text key, Iterable <IntWritable> values, Context context)
           throws IOException, InterruptedException {
4                  int sum = 0;
5                  for ( IntWritable val : values )
6                  sum += val.get();
7                  result.set(sum);
8                  context.write ( key, result);
   }}
```

```
for ( int i ; i < values.length ; i ++ ){
  sum += values[i].get()
}
```

<word,one>

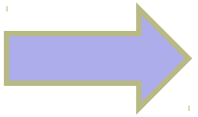
< a, 1 >

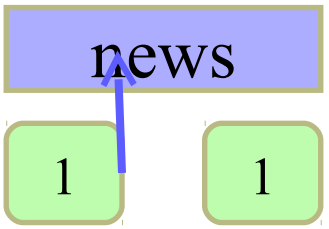
< good, 1 >

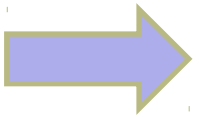
< is, 1 >

< news, 1→1 >

< no, 1 >







<key,SunValue>

< news , 2 >

Word Count – main program

```
Class WordCount{
    main()
        Configuration conf = new Configuration();
        Job job = new Job(conf, “job name” );
        job.setJarByClass(thisMainClass.class);
        job.setMapperClass(MyMapper.class);
        job.setReduceClass(MyReducer.class);
        FileInputFormat.addInputPaths(job, new Path(args[0]));
        FileOutputFormat.setOutputPath(job, new Path(args[1]));
        job.waitForCompletion(true);
}}
```

Questions?

Slides - http://trac.nchc.org.tw/cloud

Jazz Wang
Yao-Tsung Wang
jazz@nchc.org.tw

Powered by DRBL

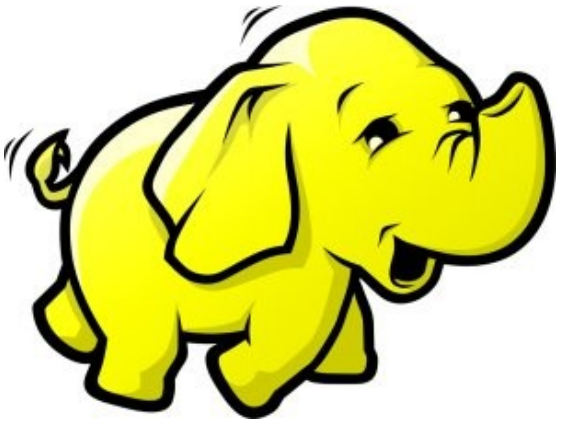
Hadoop 叢集設定解說

Setup Hadoop Fully Distributed Mode

Jazz Wang

Yao-Tsung Wang

jazz@nchc.org.tw



Yahoo's Hadoop Cluster
雅虎的大象軍團

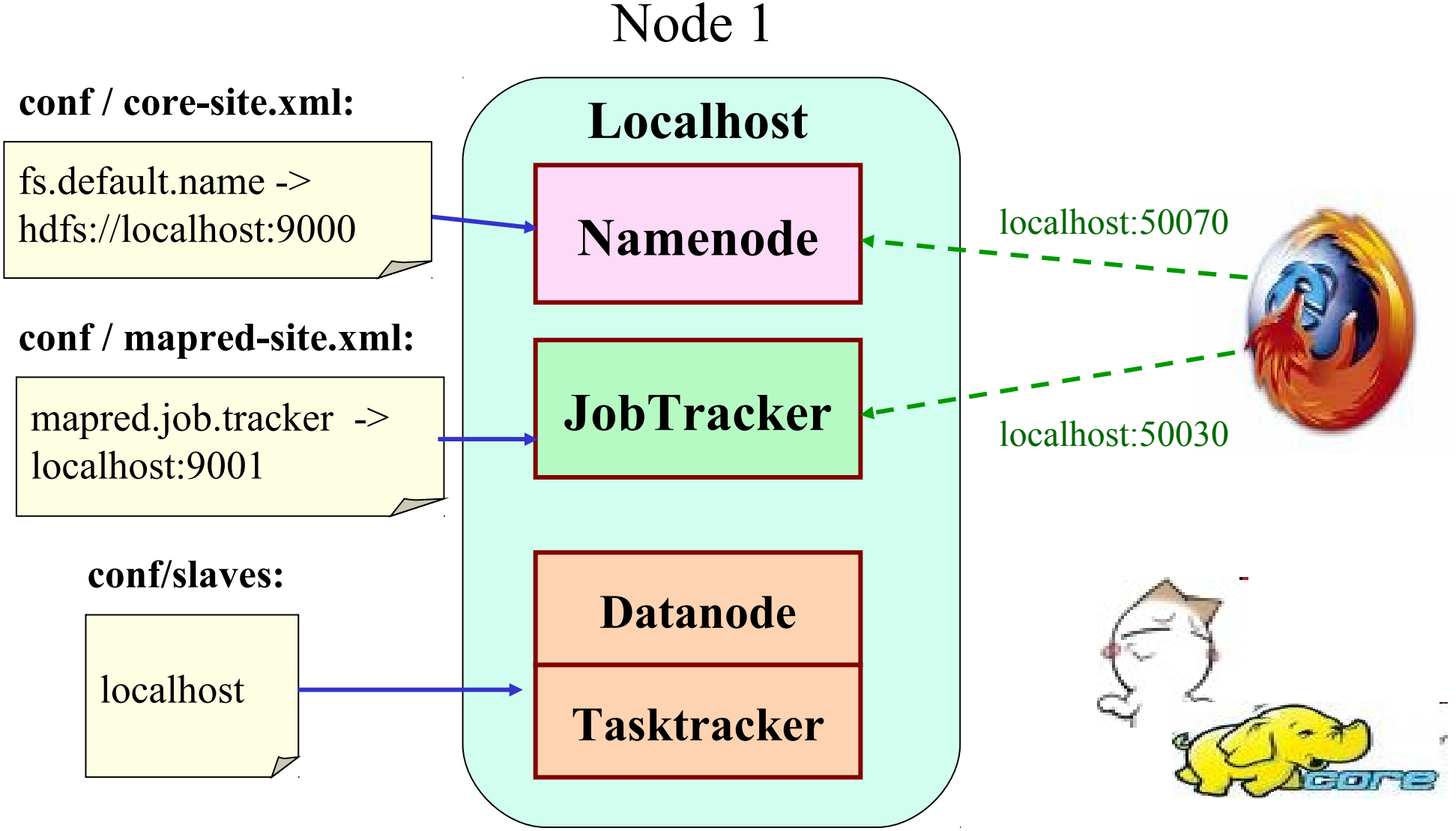
- ~10,000 machines running Hadoop in US
- The largest cluster is currently 2000 nodes
- Nearly 1 petabyte of user data (compressed, unreplicated)
- Running roughly 10,000 research jobs / week

Hadoop Pseudo-Distributed Mode
我們已經實作過單機模式

- Step 1: Setup SSH key exchange
- Step 2. Install Java
- Step 3: Download Hadoop Source Package
- Step 4: Configure hadoop-env.sh
  - export JAVA_HOME=/usr/lib/jvm/java-6-sun
- Step 5: Configure *-site.xml
  - Set Namenode to hdfs://localhost:9000
  - Set Jobtracker to localhost:9001
  - bin/hadoop namenode -format
- Step 6: Format HDFS
- Step 7: Start Hadoop
  - bin/start-all.sh
- Step 8: Complete!! Let's check the status of Hadoop
  - Job admin http://localhost:50030/ HDFS http://localhost:50070/

Diagram of Pseudo-Distributed Mode

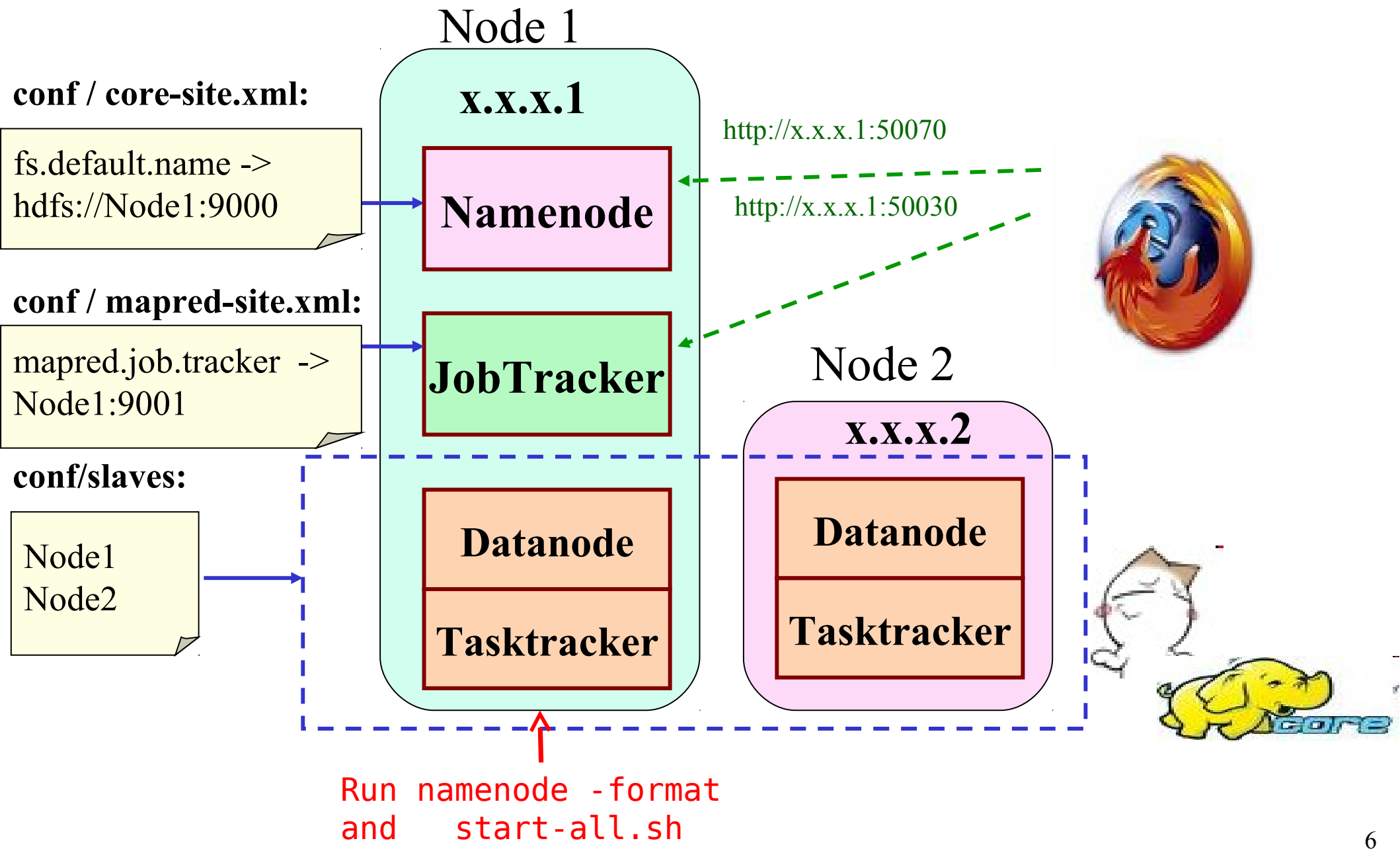
Hadoop 單機環境示意圖



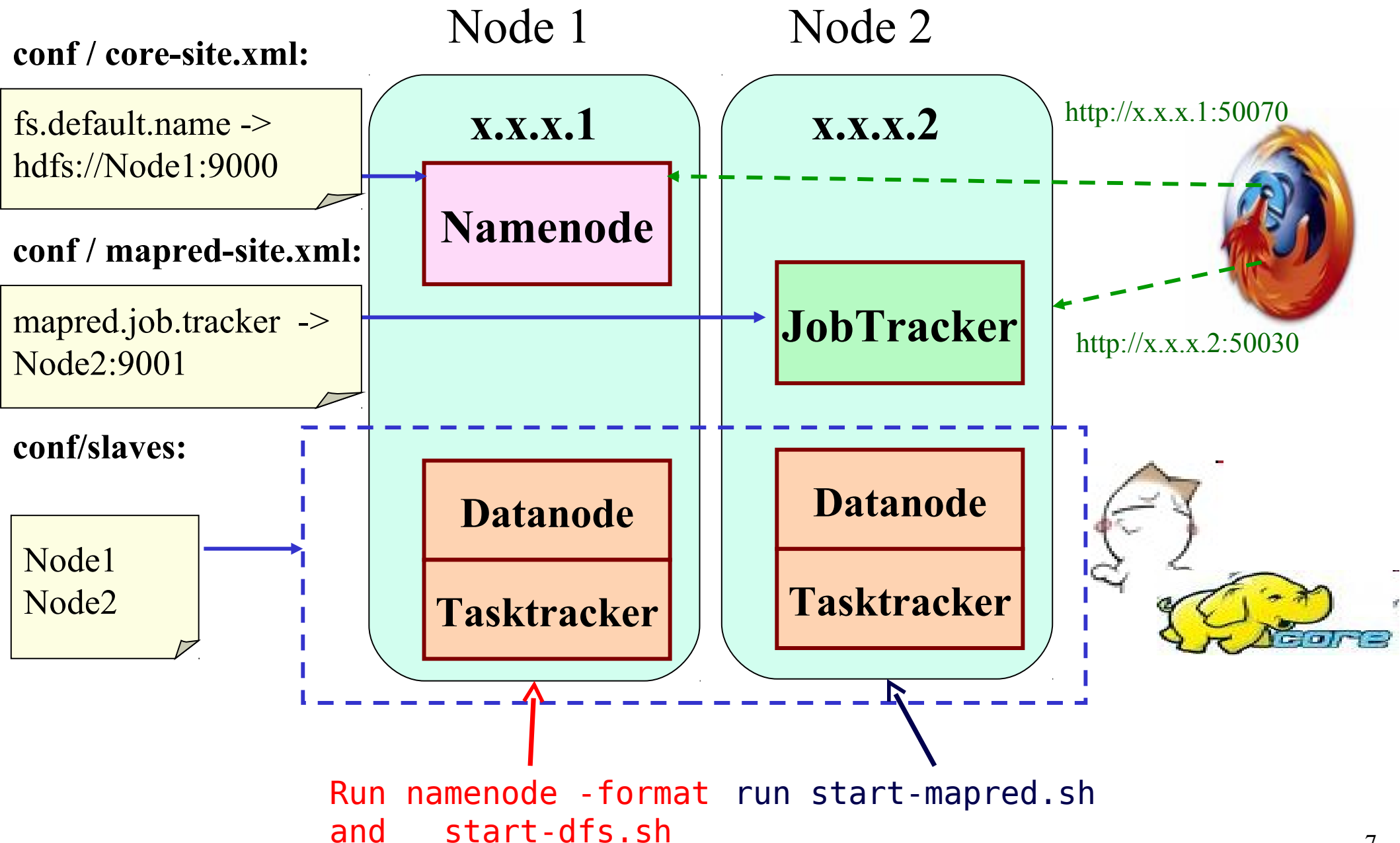
Hadoop Fully-Distributed Mode
我們接著要用兩台電腦實作叢集模式

- Step 1: Setup SSH key exchange
- Step 2. Install Java
- Step 3: Download Hadoop Source Package
- Step 4: Configure hadoop-env.sh
  - export JAVA_HOME=/usr/lib/jvm/java-6-sun
- Step 5: Configure *-site.xml
  - Set Namenode to hdfs://x.x.x.1:9000
  - Set Jobtracker to x.x.x.2:9001
- Step 6: Configure Slaves
- Step 7: Syncronization of all slaves
- Step 8: Format HDFS
  - bin/hadoop namenode -format
- Step 9: Start Hadoop
  - On NameNode ： bin/start-dfs.sh
  - On JobTracker ： bin/start-mapred.sh
- Step 10: Complete!! Let's check the status of Hadoop
  - Job admin http://x.x.x.2:50030/ HDFS http://x.x.x.1:50070/

Use case #1
設定情境一



Use case #2
設定情境二



Use case #3
設定情境三

conf / core-site.xml:

fs.default.name ->
hdfs://Node1:9000

conf / mapred-site.xml:

mapred.job.tracker ->
Node1:9001

conf/slaves:

Node2
…..
NodeN

Node 1

x.x.x.1

Namenode

JobTracker

http://x.x.x.1:50070

http://x.x.x.1:50030

Node 2

x.x.x.2

Datanode

Tasktracker

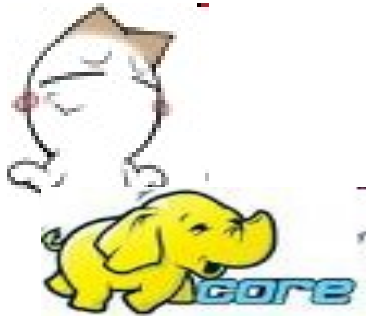
…

Node N

x.x.x.n

Datanode

Tasktracker



Use case #4
設定情境四

conf / core-site.xml:

fs.default.name ->
hdfs://Node1:9000

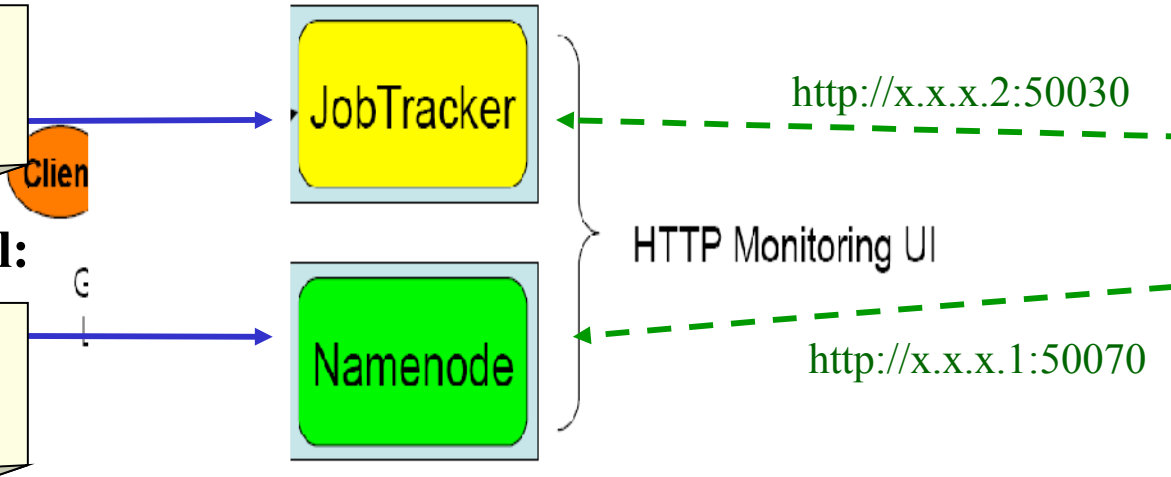
Clien

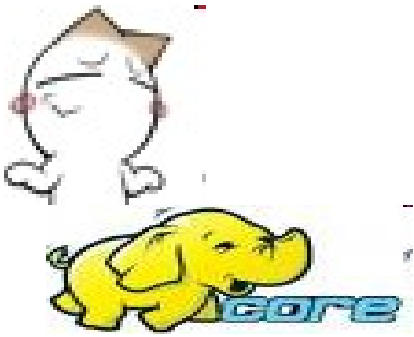
conf / mapred-site.xml:

mapred.job.tracker ->
Node2:9001

conf/slaves:

Node3
…..
NodeN





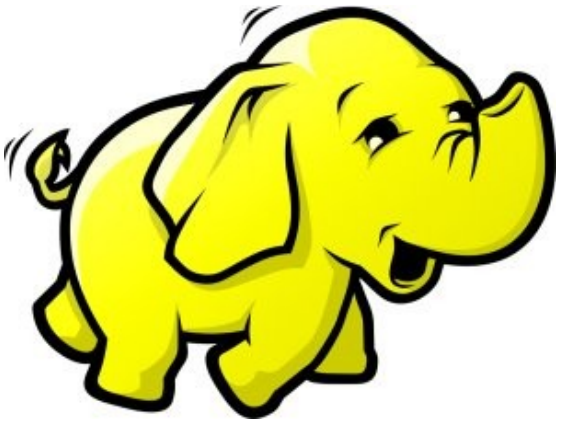
Hadoop 叢集佈署工具

Hadoop Deployment Tool : SmartFog and DRBL

Jazz Wang

Yao-Tsung Wang

jazz@nchc.org.tw



Programmer v.s. System Admin.

Source:http://www.funnyjunksite.com/wp-content/uploads/2007/08/programmer.jpg

Source:
http://www.sysadminday.com/images/people/136-3697.JPG

PART 1 :

PC Cluster 101

Jazz Wang
Yao-Tsung Wang
jazz@nchc.org.tw

At First, We have “ 4 + 1 ” PC Cluster

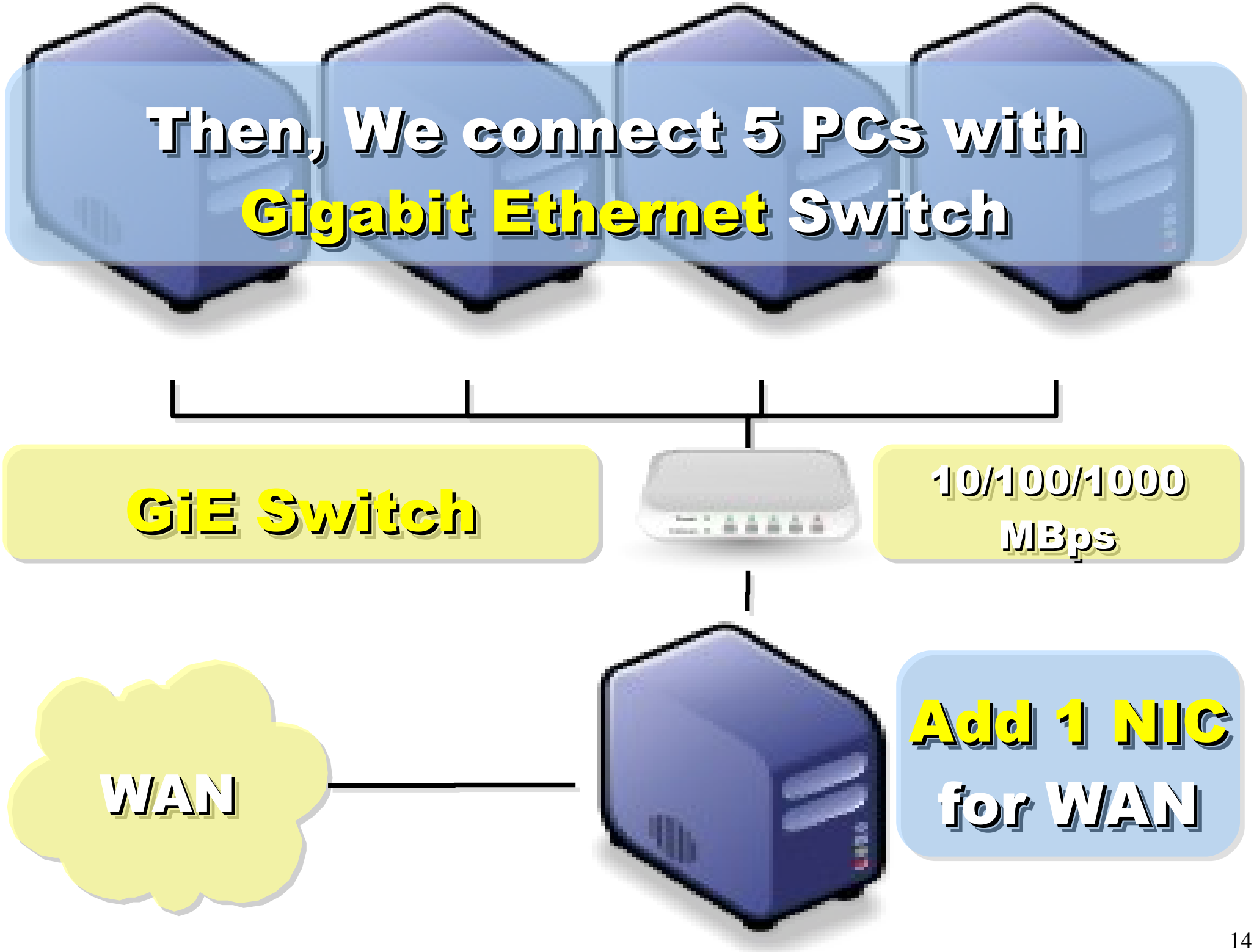
It’d better be

2^n

Manage

Scheduler

Then, We connect 5 PCs with Gigabit Ethernet Switch

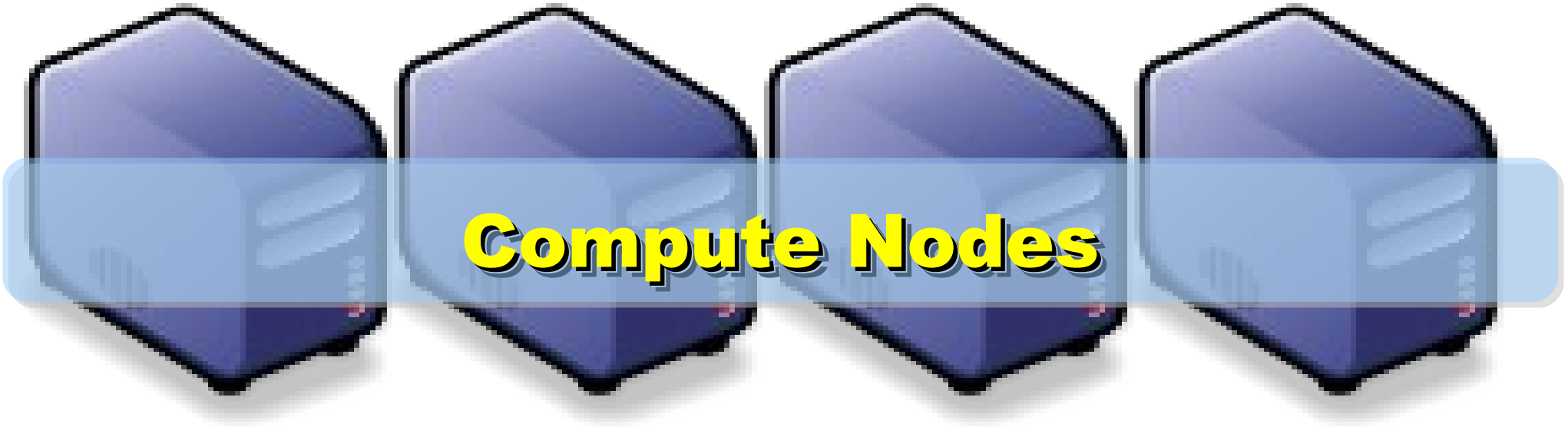


Compute Nodes

4 Compute Nodes will communicate via LAN Switch. Only Manage Node have Internet Access for Security!

WAN

Manage Node



Basic System Setup for Cluster

Messaging

Account Mgnt.

MPICH

SSHD

NIS

YP

GCC

GNU Libc

Bash

Perl

Kernel Module

Linux Kernel

Boot Loader

On Manage Node,
We need to install Scheduler and Network File System for sharing Files with Compute Node

Job Mgnt.

OpenPBS

File Sharing

NFS

Extra

Messaging

MPICH

Account Mgnt.

SSHD

NIS

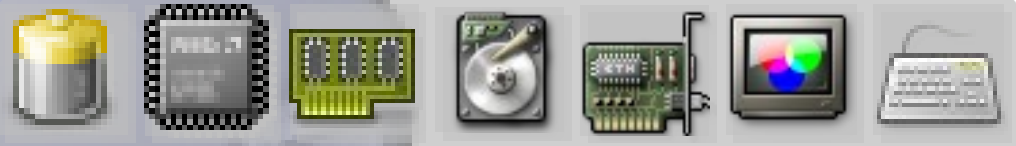
YP

GCC

GNU Libc

Bash

Perl



Kernel Module

Linux Kernel

Boot Loader

Challenges of Cluster Computing

- **Hardware**
  - Ethernet Speed / PC Density
  - Power / Cooling / Heat
  - Network and Storage Architecture
- **Software**
  - Job Scheduler (Cluster level)
  - Account Management
  - File Sharing / Package Management
- **Limitation**
  - Shared Memory
  - Global Memory Management

Common Method to deploy Cluster

1. Setup one **Template** machine

2. Cloning to multiple machine

3. Configure Settings

↓

4. Install **Job Scheduler**

↓

5. Running **Benchmark**

Challenges of Common Method

Add New User Account ?

Upgrade Software ?

How to share user data ?

Configuration Syncronization

How to deploy 4000+ Nodes ????

資料標題：Scaling Hadoop to 4000 nodes at Yahoo!

資料日期：September 30, 2008

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**

Advanced Methods to deploy Cluster

- **SSI (Single System Image)**
  - **Multiple PCs as Single Computing Resources**
  - **Image-based**
    - **homogeneous**
    - **ex. SystemImager, OSCAR, Kadeploy**
  - **Package-based**
    - **heterogeneous**
    - **easy update and modify packages**
    - **ex. FAI, DRBL**
- **Other deploy tools**
  - **Rocks : RPM only**
  - **cfengine : configuration engine**

Comparison of Cluster Deploy Tools

	Distribution	Support Diskless/ Sysmless	Type	Node configuration tools	Cluster management tools	Database installation
System Imager	ALL	Yes	Image	Yes	No	No
OSCAR	RPM-based	Yes	Image	Yes	Yes	No
Kadeploy	ALL	No	Image	Yes	Yes	Yes
DRBL	ALL	Yes	Package	Yes	Yes	No
FAI	Debian-Based	Yes	Package	Yes	No	No

PART 2-1 :

Hadoop Deployment Tool

Jazz Wang
Yao-Tsung Wang
jazz@nchc.org.tw

- Make Hadoop deployment *agile*
- Integrate with dynamic cluster deployments

Source: Deploying hadoop with smartfrog
http://people.apache.org/~stevel/slides/deploying_hadoop_with_smartfrog.pdf

SmartFrog - HPLabs' CM tool

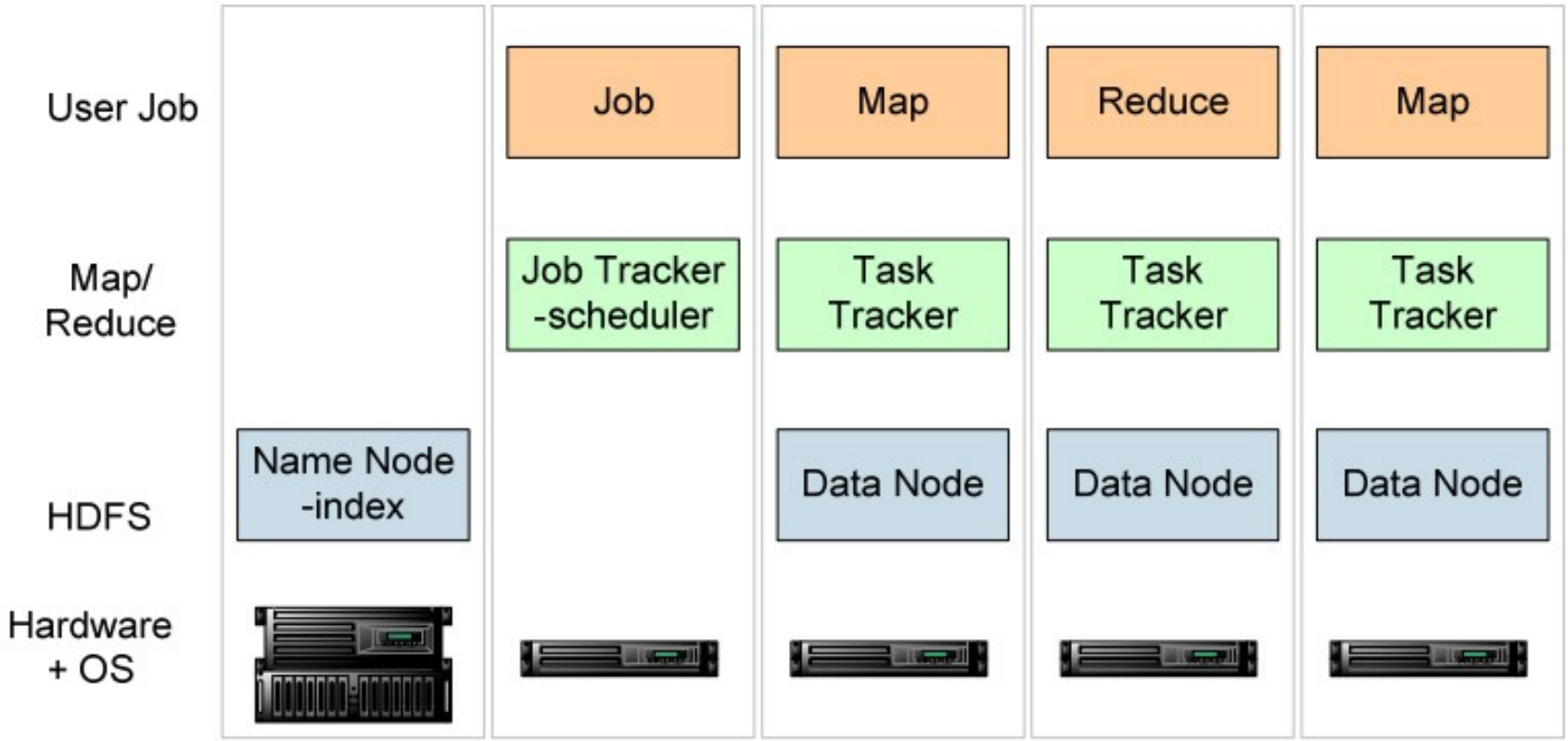
- Language for describing systems to deploy —everything from datacentres to test cases
- Runtime to create *components* from the model
- Components have a lifecycle
- LGPL Licensed, Java 5+

http://smartfrog.org/

Source: Deploying hadoop with smartfrog
http://people.apache.org/~stevel/slides/deploying_hadoop_with_smartfrog.pdf

LABS hp

Basic problem: deploying Hadoop



one namenode, 1+ Job Tracker, many data nodes and task trackers

Source: Deploying hadoop with smartfrog
http://people.apache.org/~stevel/slides/deploying_hadoop_with_smartfrog.pdf

The hand-managed cluster

- Manual install onto machines
- SCP/FTP in Hadoop zip
- copy out hadoop-site.xml and other files
- edit /etc/hosts, /etc/rc5.d, SSH keys …
- Installation scales O(N)
- Maintenance, debugging scales worse

Source: Deploying hadoop with smartfrog
http://people.apache.org/~stevel/slides/deploying_hadoop_with_smartfrog.pdf

The locked-down cluster

- PXE Preboot of OS images
- RedHat Kickstart to serve up (see instalinux.com)
- Maybe: LDAP to manage state, or custom RPMs

Requires:

uniform images, central LDAP service, good ops team, stable configurations, home-rolled RPMs

Source: Deploying hadoop with smartfrog
http://people.apache.org/~stevel/slides/deploying_hadoop_with_smartfrog.pdf

CM-tool managed cluster

Configuration Management tools

- State Driven: observe system state, push it back into the desired state
- Workflow: apply a sequence of operations to change a machine's state
- Centralized: central DB in charge
- Decentralized: machines look after themselves

CM tools are the only way to manage big clusters

Source: Deploying hadoop with smartfrog
http://people.apache.org/~stevel/slides/deploying_hadoop_with_smartfrog.pdf

Model the system in the SmartFrog language

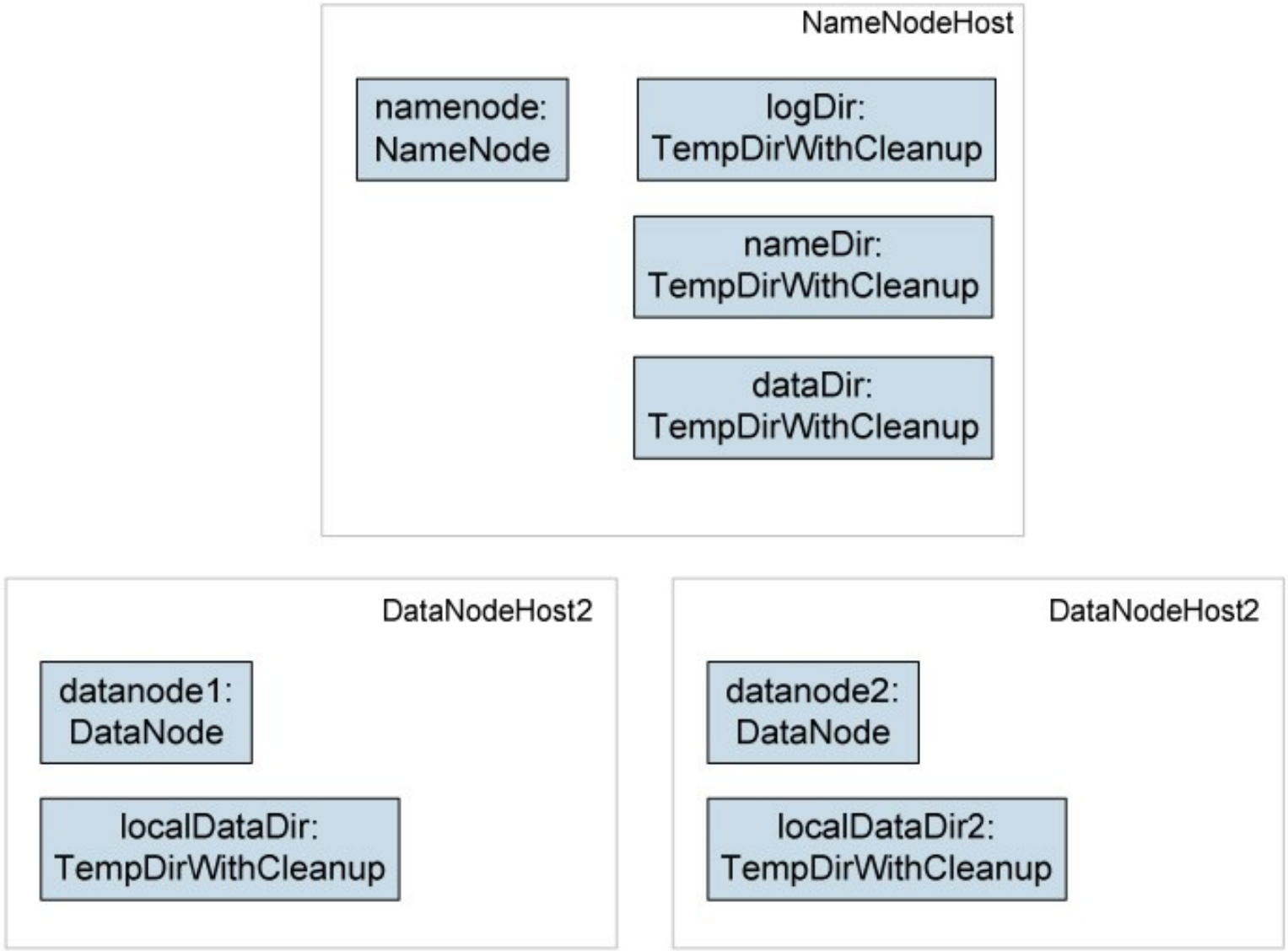
```
TwoNodeHDFS extends OneNodeHDFS {

  localDataDir2 extends TempDirWithCleanup {

  }

  datanode2 extends datanode {
    dataDirectories [LAZY localDataDir2];
    dfs.datanode.https.address "https://localhost:0";
  }
}
```

Inheritance, cross-referencing, templating

Source: Deploying hadoop with smartfrog
http://people.apache.org/~stevel/slides/deploying_hadoop_with_smartfrog.pdf

LABS hp

The runtime deploys the model



Source: Deploying hadoop with smartfrog
http://people.apache.org/~stevel/slides/deploying_hadoop_with_smartfrog.pdf

Steps to deployability

1. Configure Hadoop from an SmartFrog description
2. Write components for the Hadoop nodes
3. Write the functional tests
4. Add *workflow* components to work with the filesystem; submit jobs
5. Get the tests to pass

Source: Deploying hadoop with smartfrog
http://people.apache.org/~stevel/slides/deploying_hadoop_with_smartfrog.pdf

PART 2-2 :

Introduction to DRBL

Jazz Wang
Yao-Tsung Wang
jazz@nchc.org.tw

What is DRBL ??

- **Diskless Remote Boot in Linux**
- Network is cheap, and our time is expansive
- In simple words, DRBL is .....
  - Replace IDE/SATA cable with network cable
  - 40+ student PCs connected to one DRBL server

Diskfull PC

=

+

+

Diskless PC

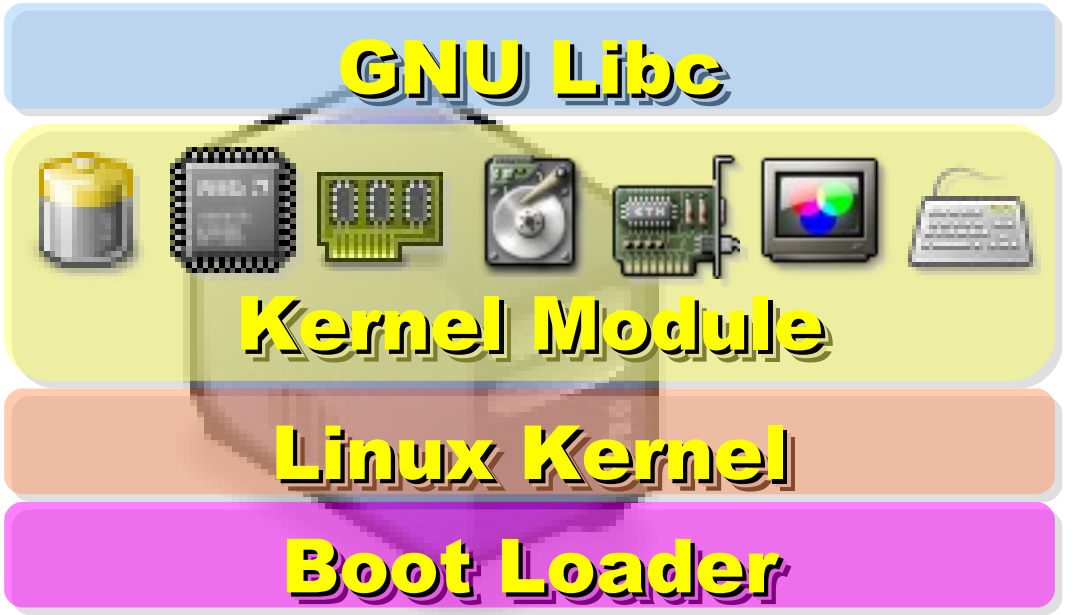
Server

source: http://www.mren.com.tw

1st, We install Base System of GNU/Linux on Management Node.

You can choose:

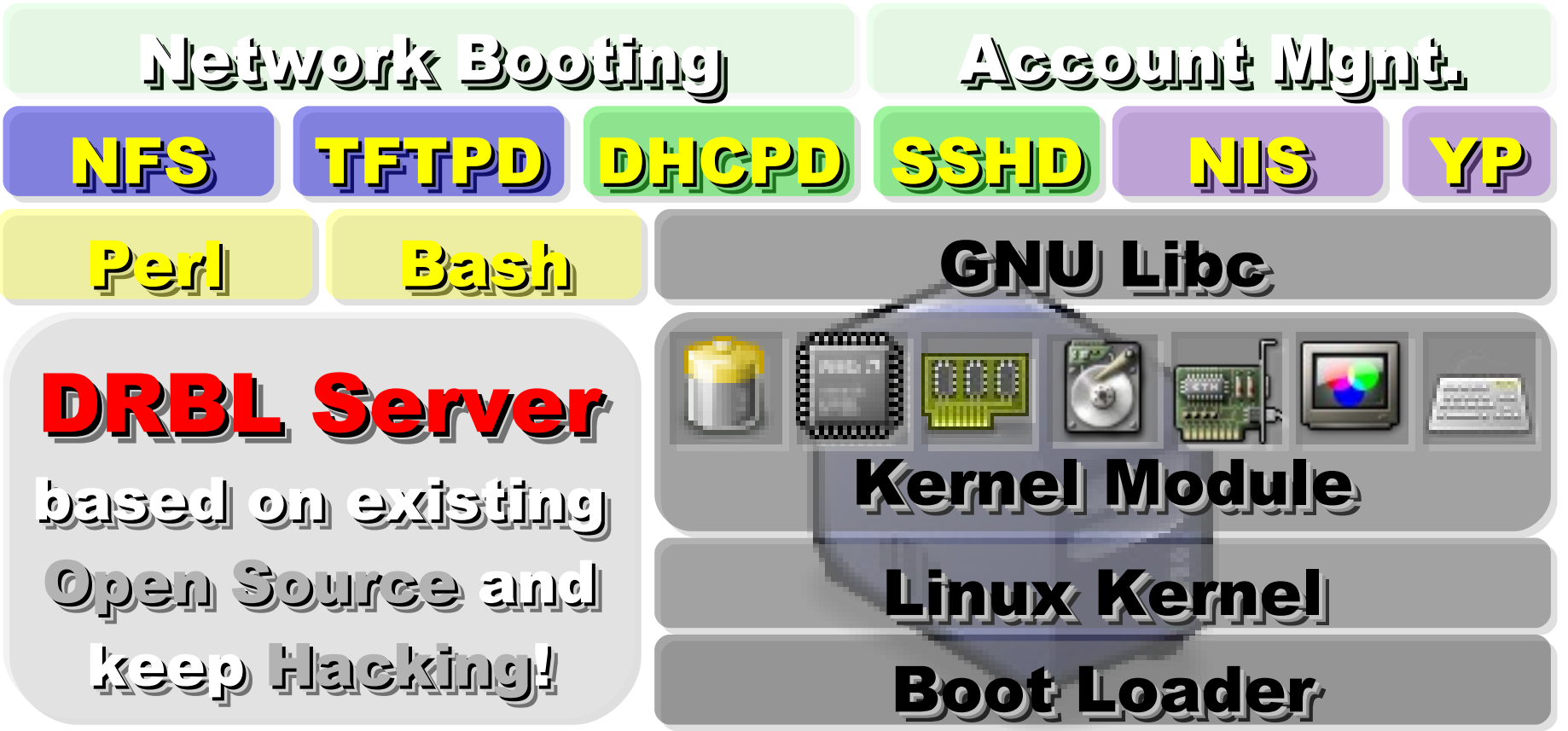
Redhat, Fedora, CentOS, Mandriva, Ubuntu, Debian, ...



2nd, We install DRBL package and configure it as DRBL Server.

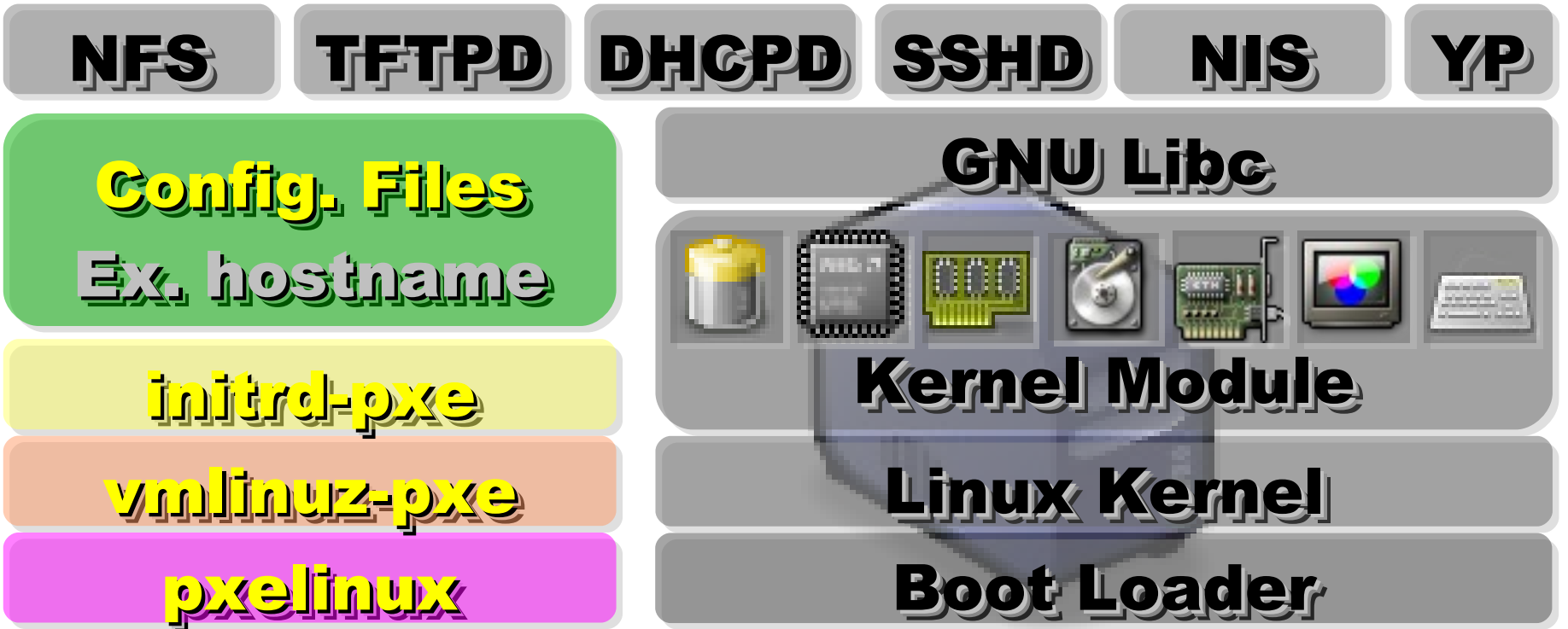
There are lots of service needed:

SSHD, DHCPD, TFTPD, NFS Server,
NIS Server, YP Server ...



DRBL Server
based on existing Open Source and keep Hacking!

After running "drblsrv -i" & "drblpush -i", there will be pxelinux, vmlinux-pex, initrd-pxe in TFTPROOT, and different configuration files for each Compute Node in NFSROOT



3nd, We enable PXE function in BIOS configuration.

BIOS PXE | BIOS PXE | BIOS PXE | BIOS PXE

NFS | TFTPD | DHCPD | SSHD | NIS | YP

Config. Files
Ex. hostname

initrd-pxe

vmlinuz-pxe

pxelinux

GNU Libc

Kernel Module

Linux Kernel

Boot Loader

While Booting, PXE will query IP address from DHCPD.

BIOS PXE | BIOS PXE | BIOS PXE | BIOS PXE

NFS | TFTPD | DHCPD | SSHD | NIS | YP

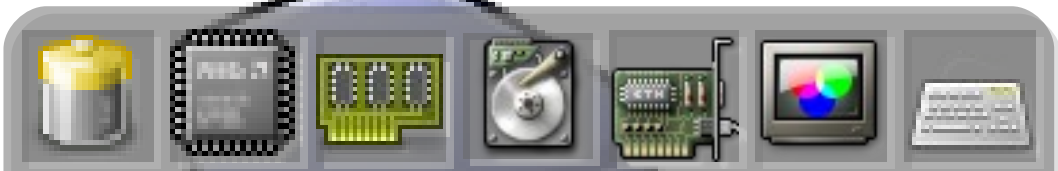
Config. Files
Ex. hostname

initrd-pxe

vmlinuz-pxe

pxelinux

GNU Libc

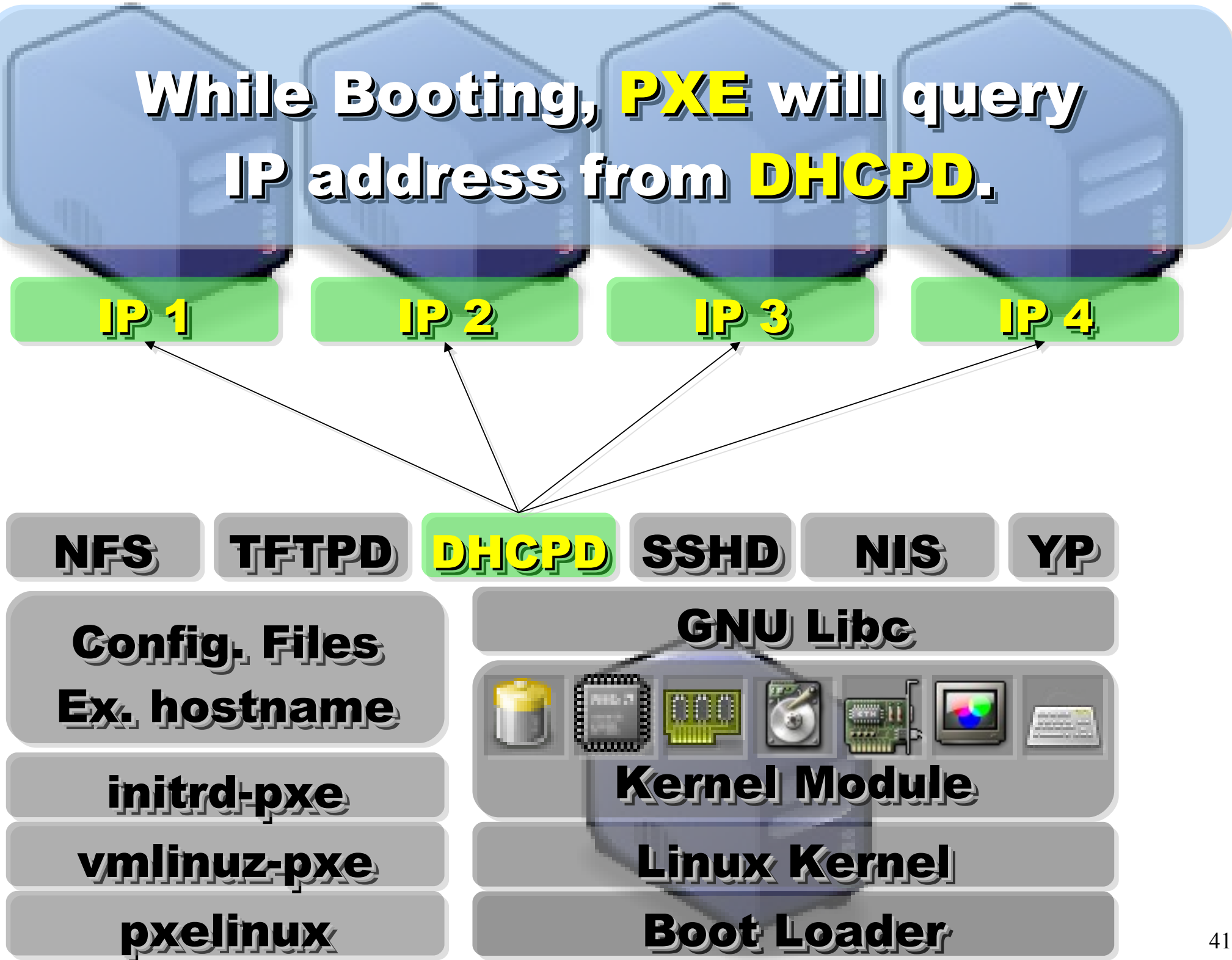


Kernel Module

Linux Kernel

Boot Loader

While Booting, PXE will query IP address from DHCPD.



After PXE get its IP address, it will download booting files from TFTPD.

IP 1 | IP 2 | IP 3 | IP 4

NFS | TFTPD | DHCPD | SSHD | NIS | YP

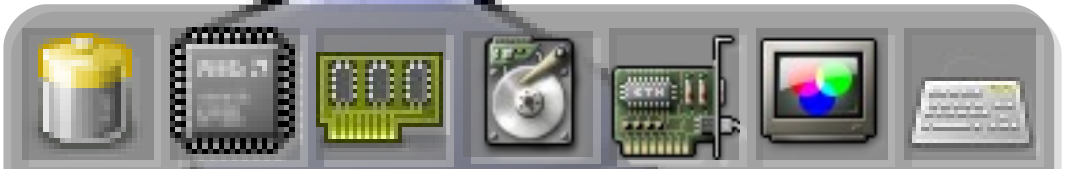
Config. Files
Ex. hostname

initrd-pxe

vmlinuz-pxe

pxelinux

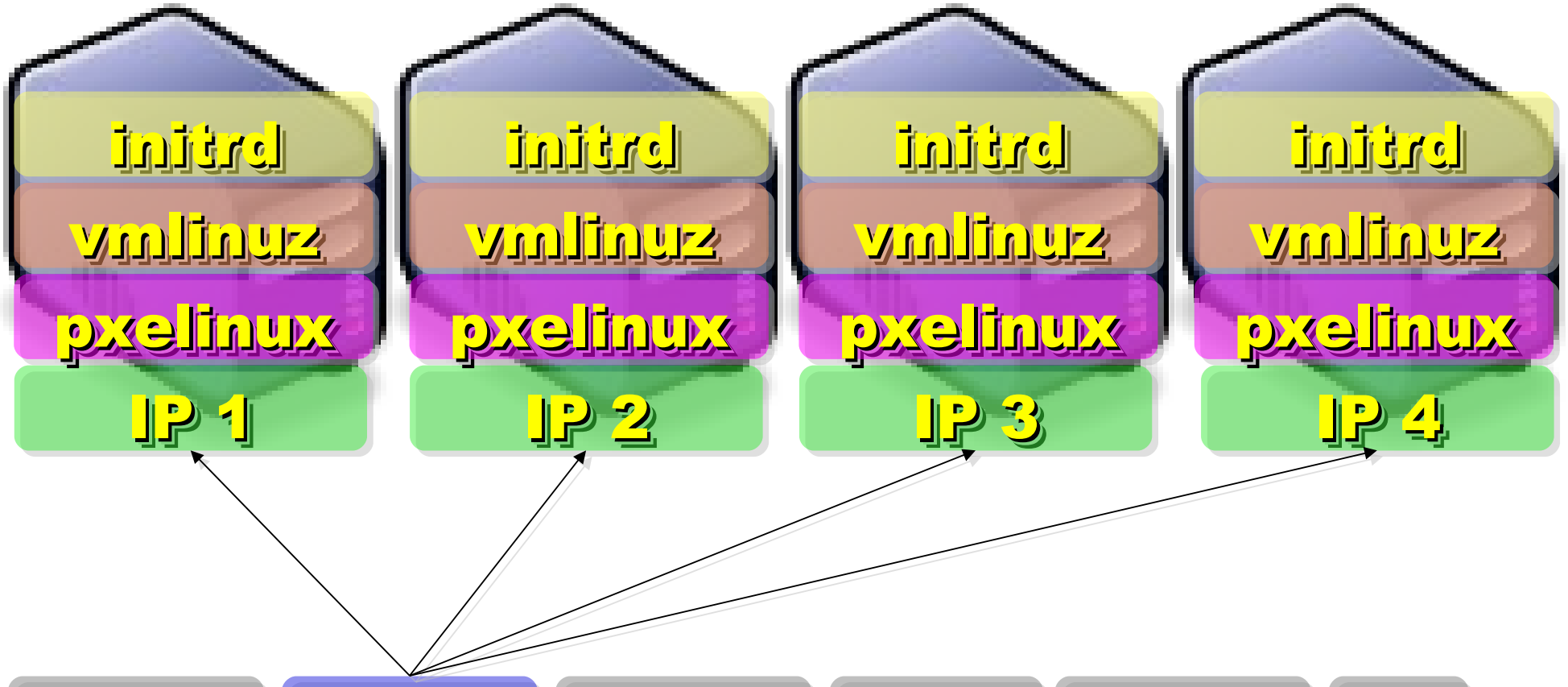
GNU Libc



Kernel Module

Linux Kernel

Boot Loader



NFS | TFTPD | DHCPD | SSHD | NIS | YP

Config. Files
Ex. hostname

initrd-pxe

vmlinuz-pxe

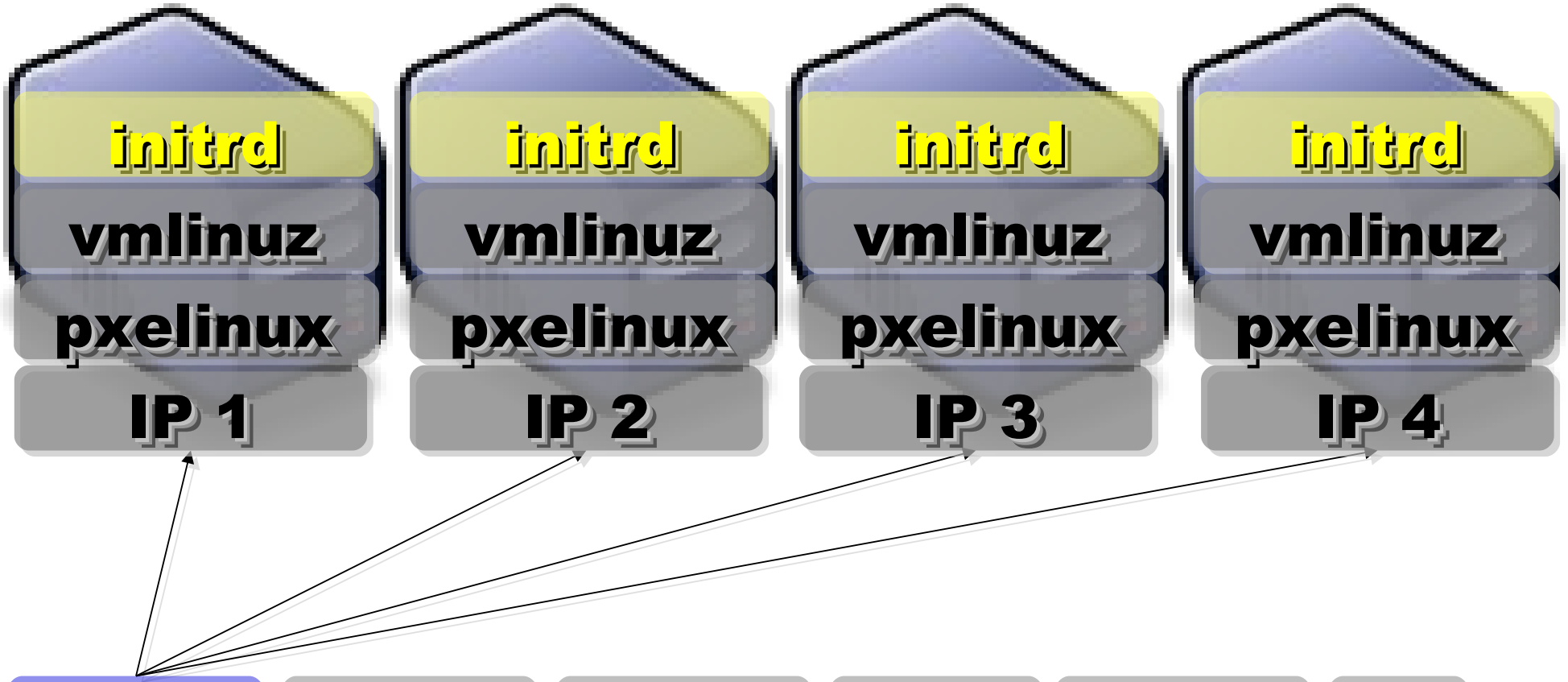
pxelinux

GNU Libc

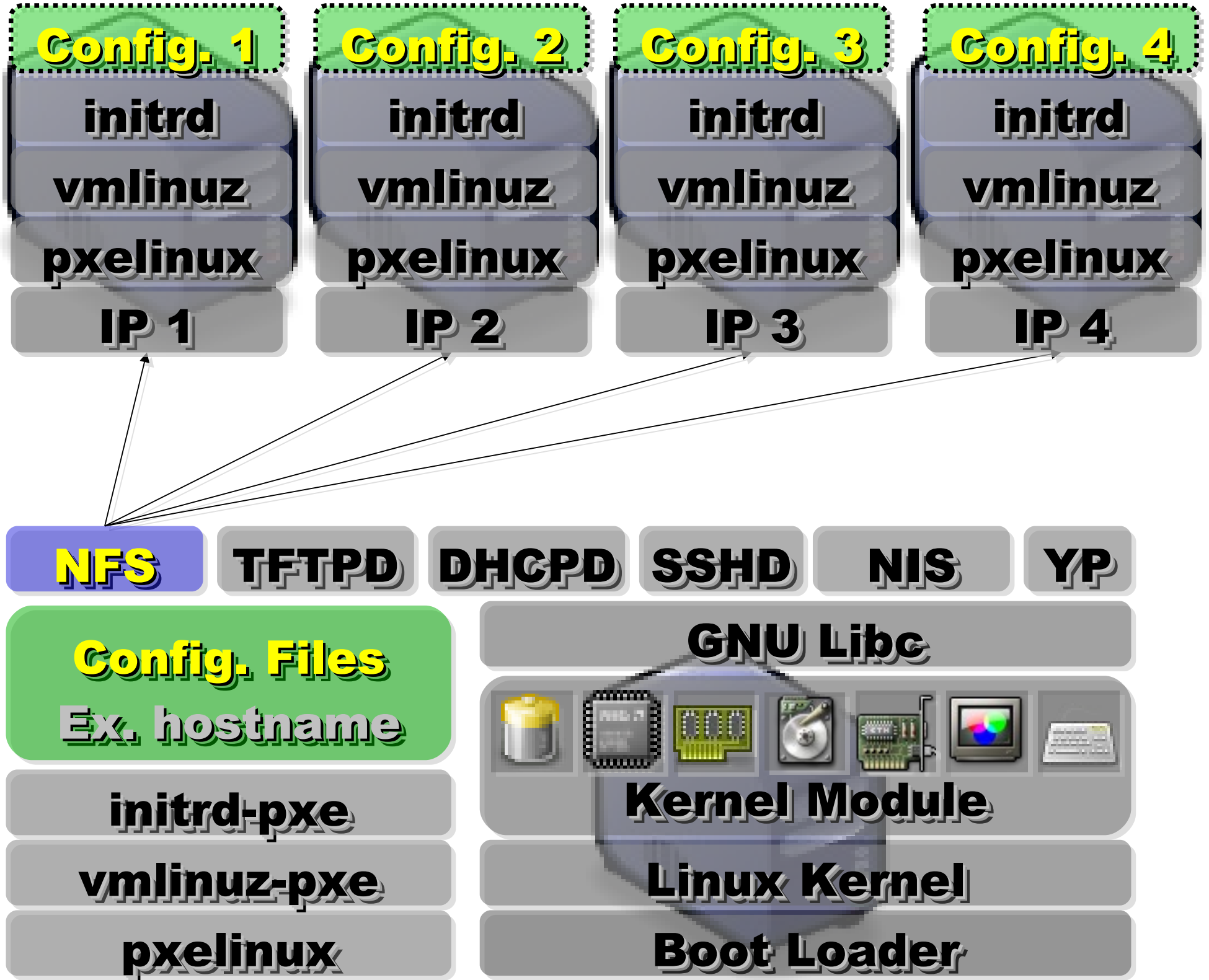
Kernel Module

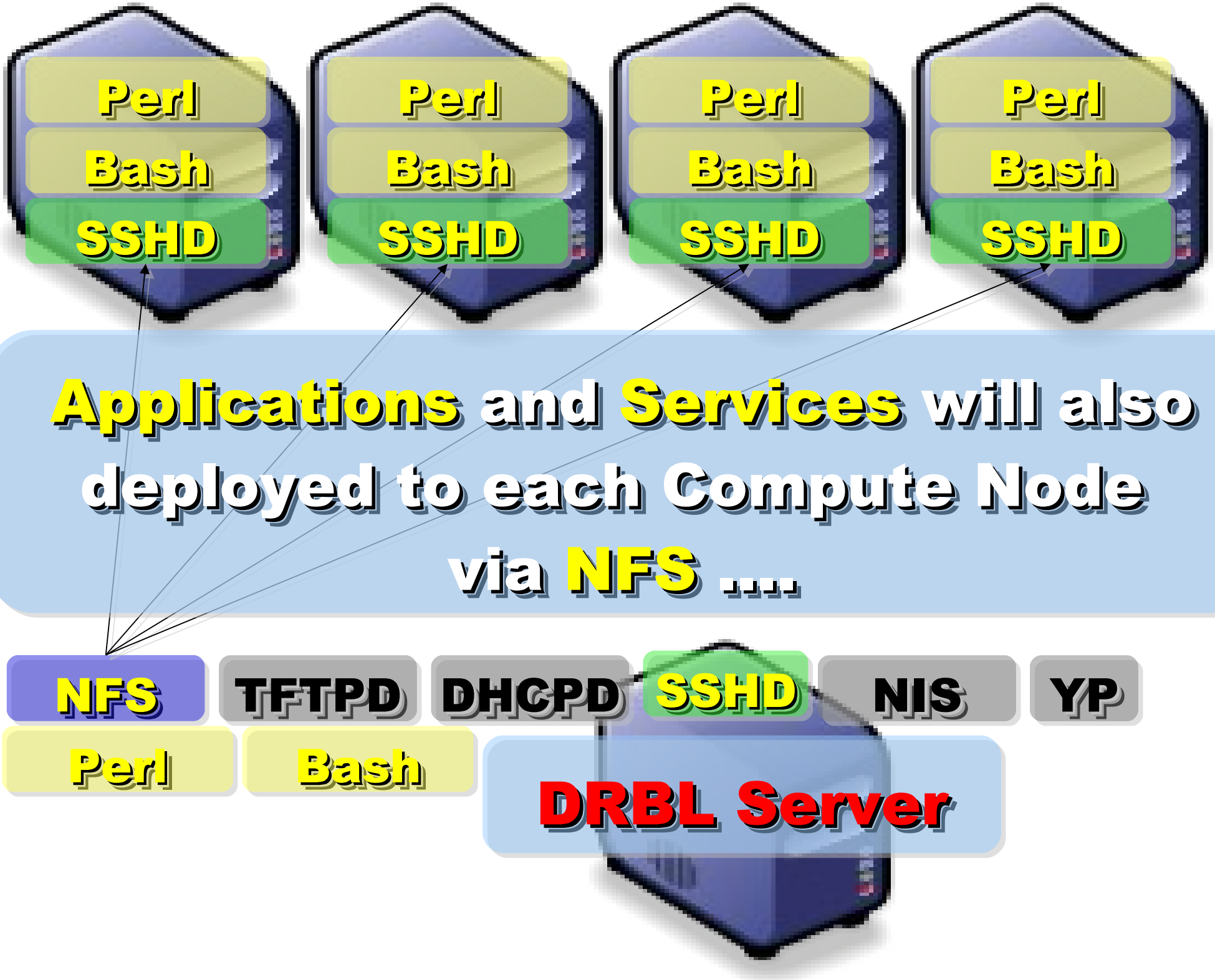
Linux Kernel

Boot Loader



After downloading booting files, scripts in **initrd-pxe** will config **NFSROOT** for each Compute Node.


Config. 1
Config. 2
Config. 3
Config. 4
initrd
vmlinuz
pxelinux
IP 1
IP 2
IP 3
IP 4
NFS
TFTPD
DHCPD
SSHD
NIS
YP
Config. Files
Ex. hostname
initrd-pxe
vmlinuz-pxe
pxelinux
GNU Libc
Kernel Module
Linux Kernel
Boot Loader


Perl
Bash
SSHD
Perl
Bash
SSHD
Perl
Bash
SSHD
Perl
Bash
SSHD
Applications and Services will also deployed to each Compute Node via NFS ....
NFS
TFTPD
DHCPD
SSHD
NIS
YP
Perl
Bash
DRBL Server

With the help of NIS and YP,
You can login each Compute Node
with the Same ID / PASSWORD
stored in DRBL Server!

SSH Client

NFS TFTPD DHCPD SSHD NIS YP

DRBL Server

Questions?

Slides - http://trac.nchc.org.tw/cloud

Jazz Wang
Yao-Tsung Wang
jazz@nchc.org.tw

Powered by DRBL

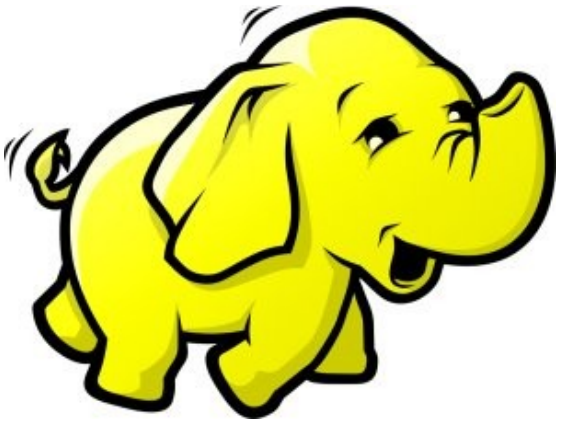
Hadoop 相關計畫

Hadoop Ecosystem

Jazz Wang

Yao-Tsung Wang

jazz@nchc.org.tw

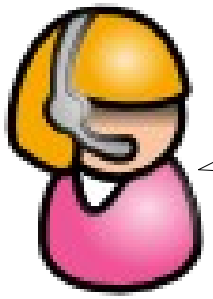


Hadoop 只支援用 **Java** 開發嘛？

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
```

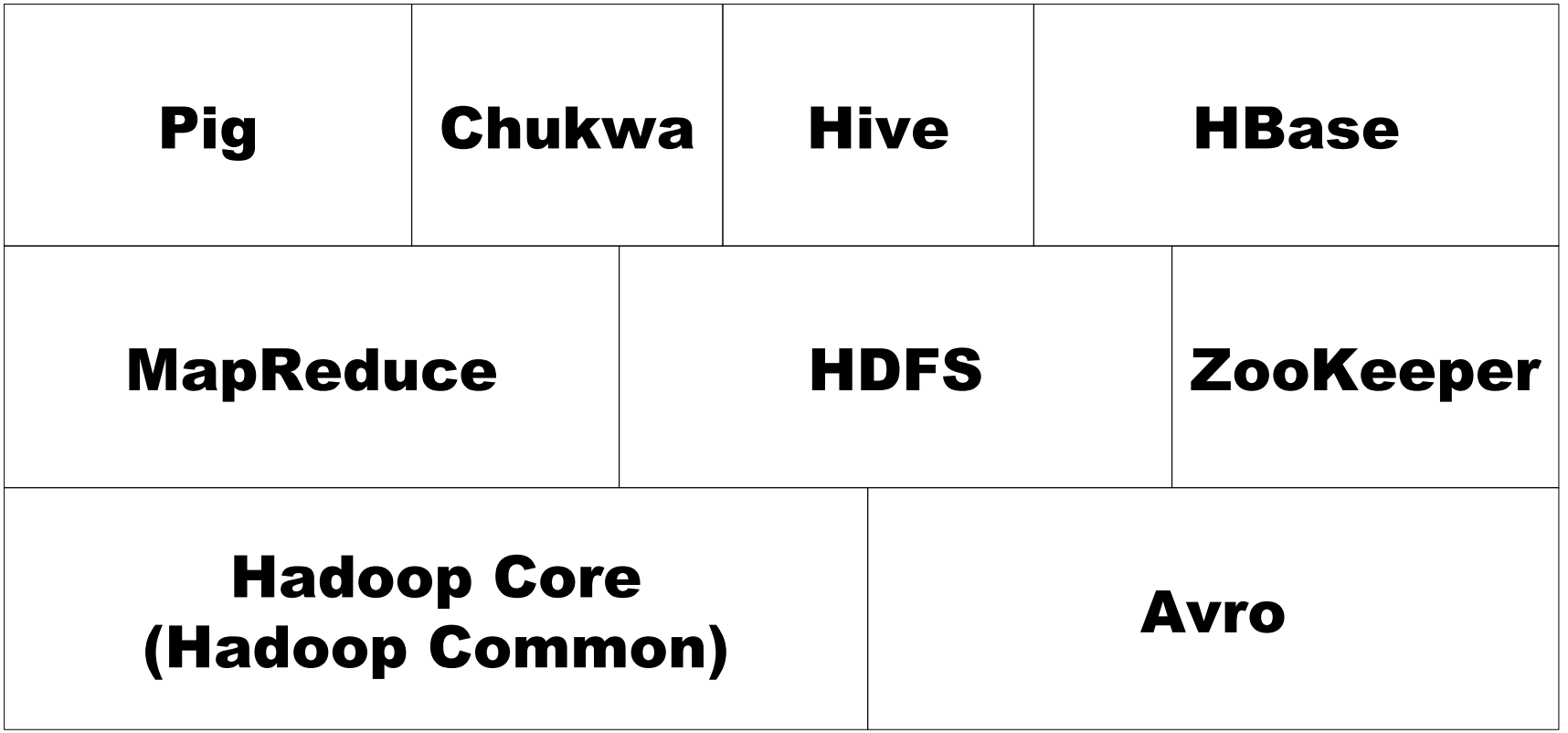
There are serveral Hadoop subprojects

- 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 ...

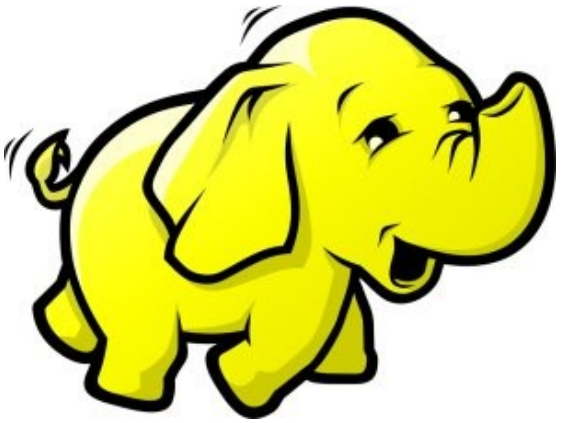
HBase 雲端資料庫

Introduction to HBase

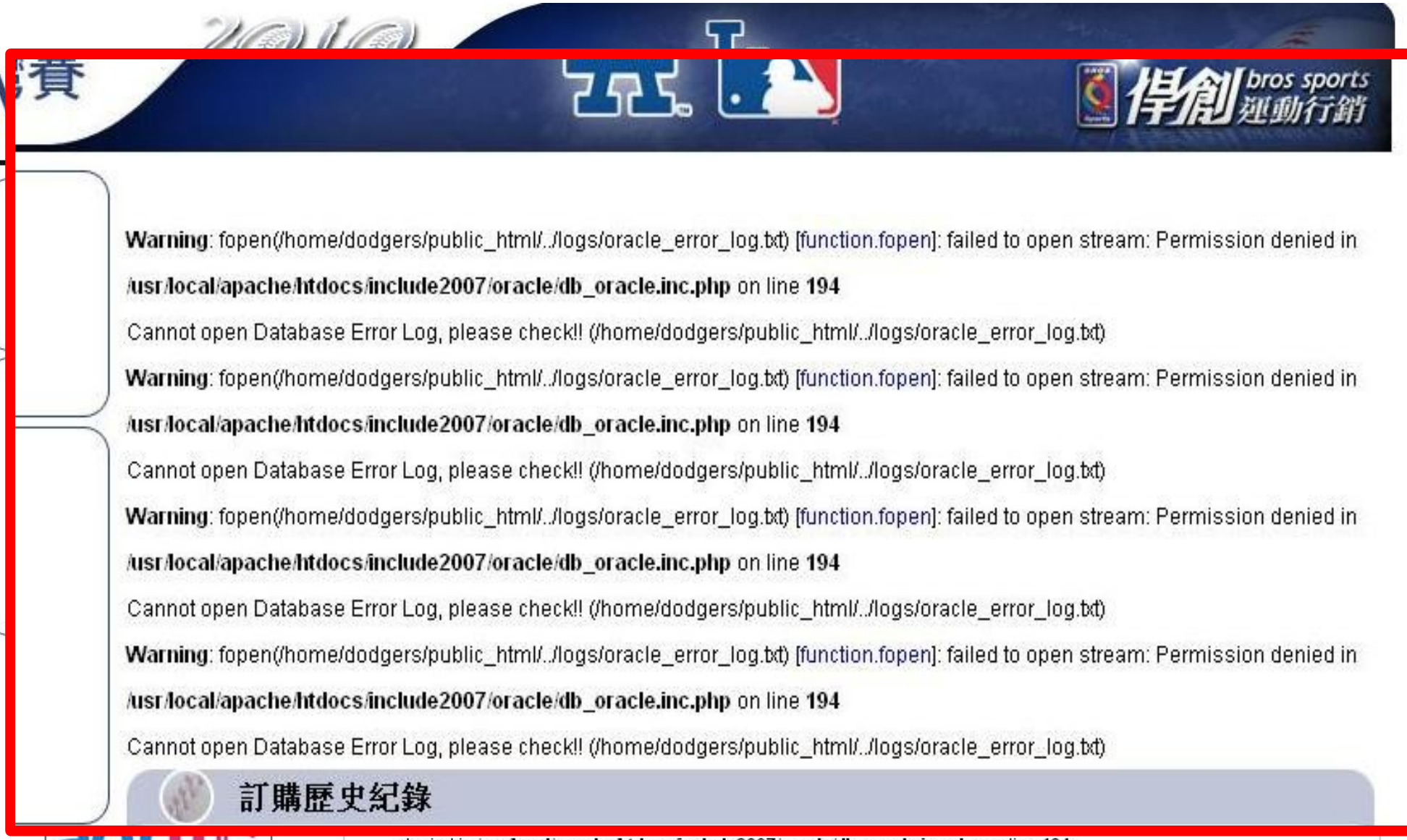
Jazz Wang

Yao-Tsung Wang

jazz@nchc.org.tw



It’s all about SCALE!!

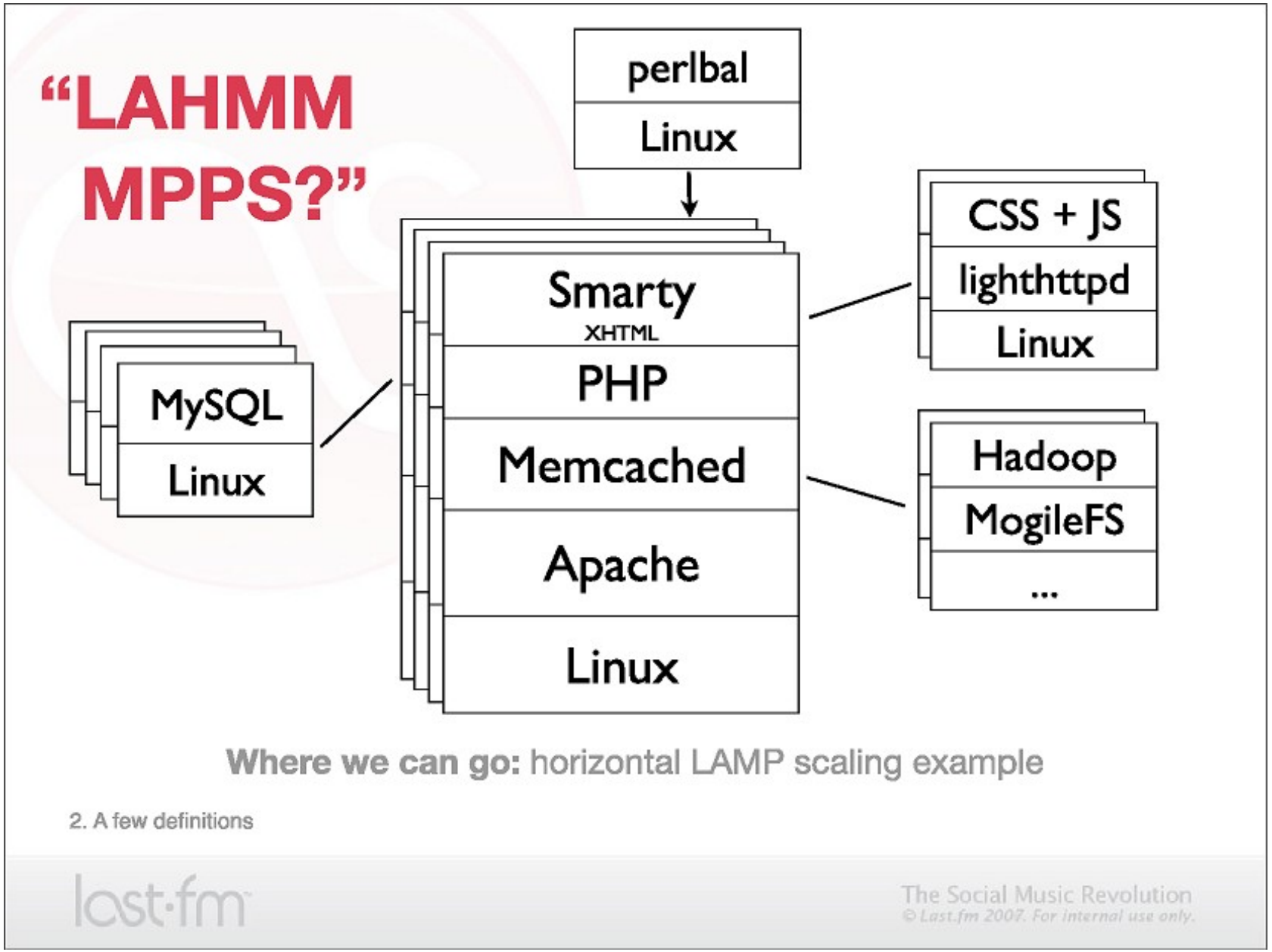


```
Warning: fopen(/home/dodgers/public_html/../logs/oracle_error_log.txt) [function.fopen]: failed to open stream: Permission denied in
/usr/local/apache/htdocs/include2007/oracle/db_oracle.inc.php on line 194
Cannot open Database Error Log, please check!! (/home/dodgers/public_html/../logs/oracle_error_log.txt)
Warning: fopen(/home/dodgers/public_html/../logs/oracle_error_log.txt) [function.fopen]: failed to open stream: Permission denied in
/usr/local/apache/htdocs/include2007/oracle/db_oracle.inc.php on line 194
Cannot open Database Error Log, please check!! (/home/dodgers/public_html/../logs/oracle_error_log.txt)
Warning: fopen(/home/dodgers/public_html/../logs/oracle_error_log.txt) [function.fopen]: failed to open stream: Permission denied in
/usr/local/apache/htdocs/include2007/oracle/db_oracle.inc.php on line 194
Cannot open Database Error Log, please check!! (/home/dodgers/public_html/../logs/oracle_error_log.txt)
Warning: fopen(/home/dodgers/public_html/../logs/oracle_error_log.txt) [function.fopen]: failed to open stream: Permission denied in
/usr/local/apache/htdocs/include2007/oracle/db_oracle.inc.php on line 194
Cannot open Database Error Log, please check!! (/home/dodgers/public_html/../logs/oracle_error_log.txt)
```

訂購歷史紀錄

```
denied in /usr/local/apache/htdocs/include2007/oracle/db_oracle.inc.php on line 194
Cannot open Database Error Log, please check!! (/home/dodgers/public_html/../logs/oracle_error_log.txt)
Warning: fopen(/home/dodgers/public_html/../logs/oracle_error_log.txt) [function.fopen]: failed to open stream: Permission
```

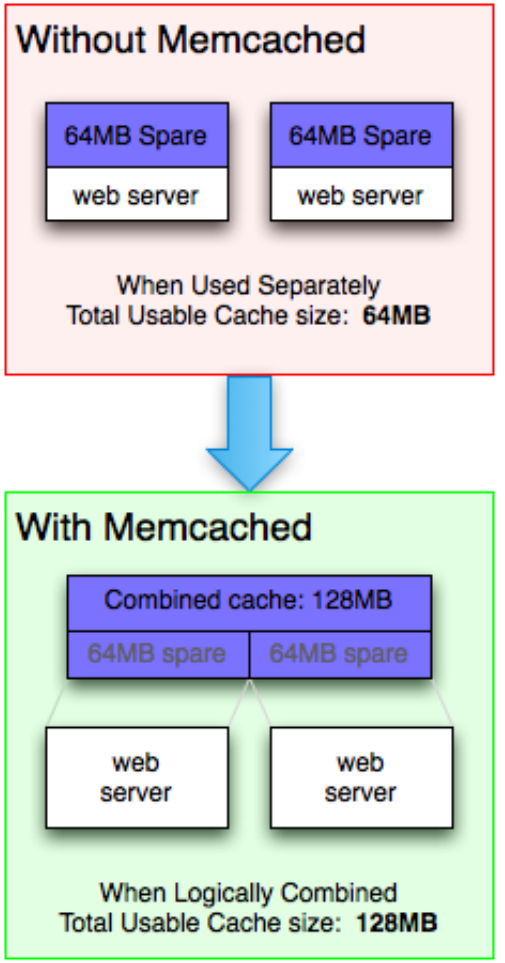
How to scale up web service in the past ?



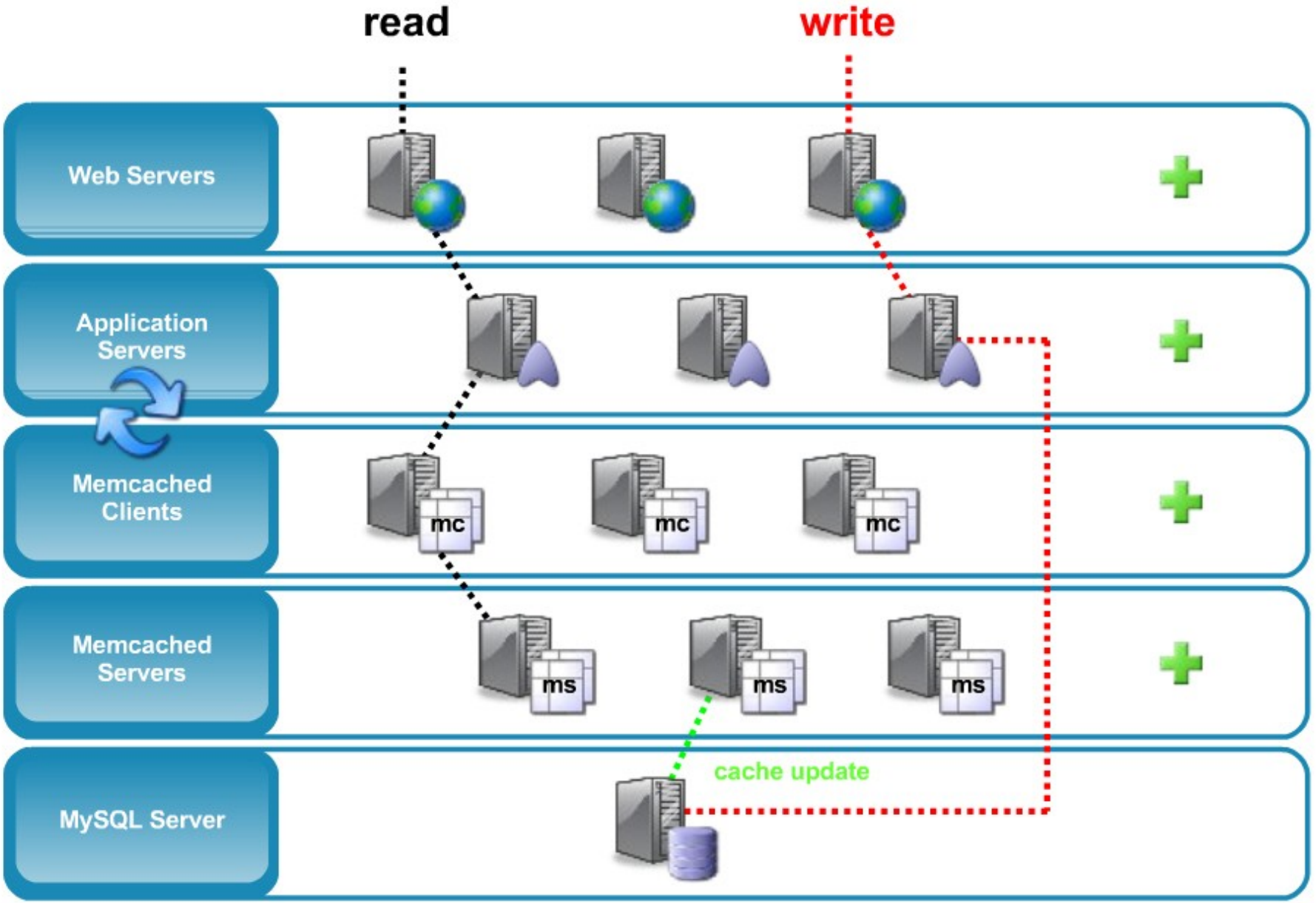
Source: http://www.slideshare.net/mallipeddi/scalable-lamp-development-for-growing-web-apps

Tools used by large scale websites

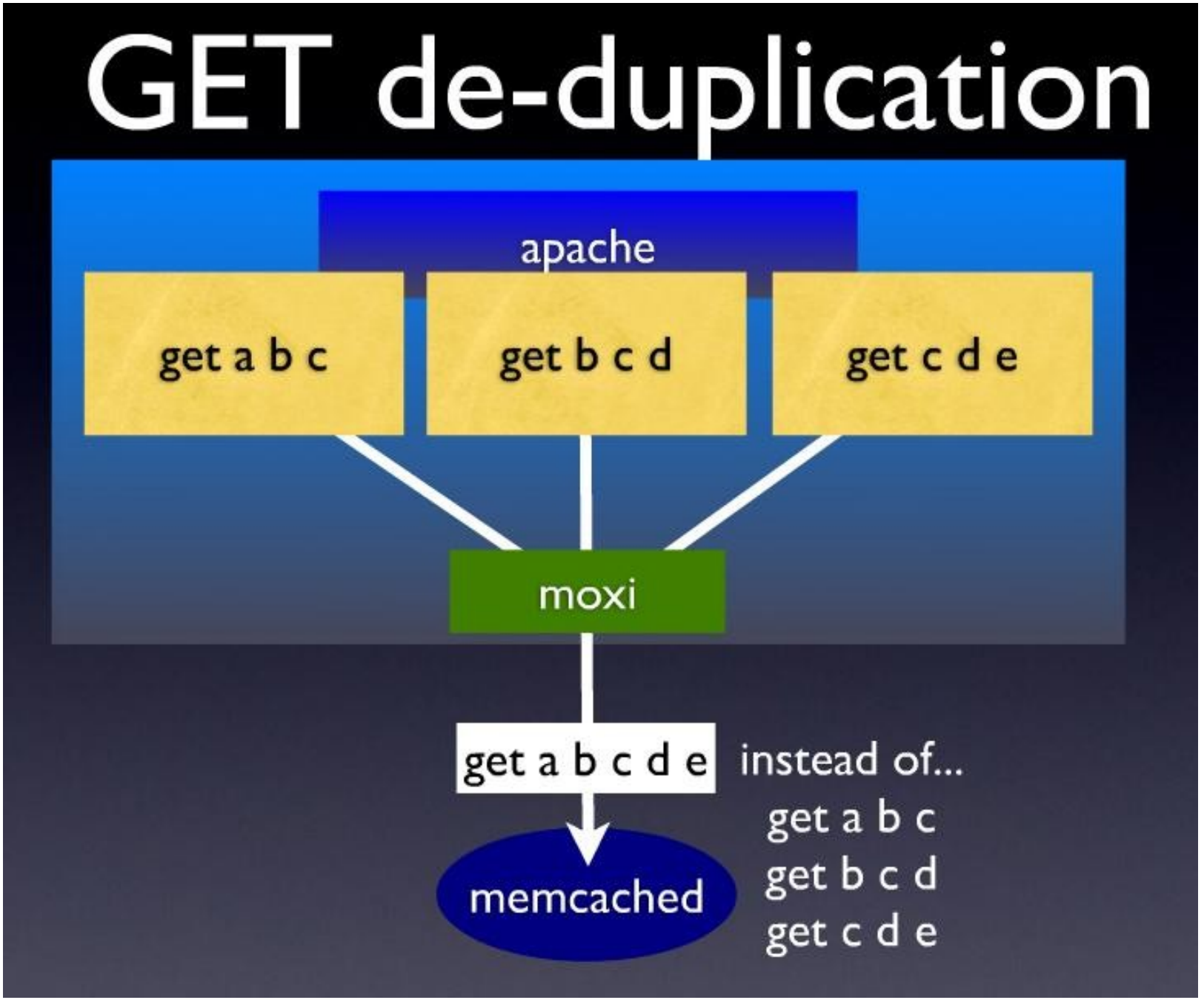
- Perlbal - http://www.danga.com/perlbal/
  - 多個網頁伺服器的負載平衡
  - Load balancer
- MogileFS - http://www.danga.com/mogilefs/
  - 分散式檔案系統
  - Distributed File System fo small files
  - 有公司認爲 MogileFS 比起 Hadoop 適合拿來處理小檔案
- memcached - http://memcached.org/
  - 共享記憶體 ??
  - Share Memory
  - 把資料庫或經常讀取的部分，
    用記憶體快取 (Cache) 方式存放
- Moxi - http://code.google.com/p/moxi/
  - Memcache 的 PROXY
- More Resource:
  - http://code.google.com/p/memcached/wiki/HowToLearnMoreScalability
  - http://www.slideshare.net/techdude/scalable-web-architectures-common-patterns-and-approaches



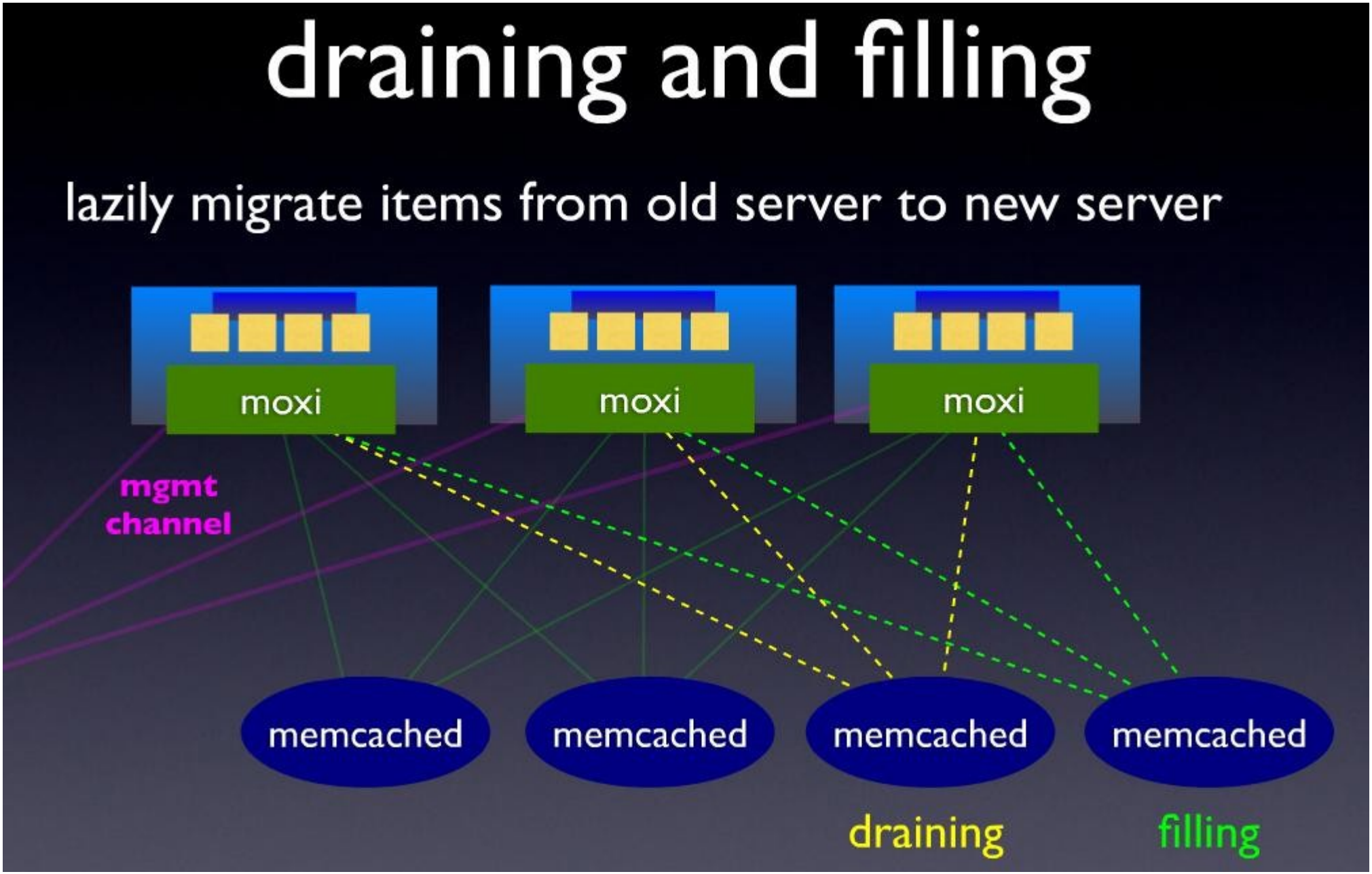
Memcached & MySQL



Source: http://mashraqi.com/2008/07/memcached-for-mysql-advanced-use-cases_09.html



Source: http://www.slideshare.net/northscale/moxi-memcached-proxy



Source: http://www.slideshare.net/northscale/moxi-memcached-proxy

HBase is ..

- HBase is a distributed column-oriented database built on top of HDFS.
- A distributed data store that can scale horizontally to 1,000s of commodity servers and petabytes of indexed storage.
- Designed to operate on top of the Hadoop distributed file system (HDFS) or Kosmos File System (KFS, aka Cloudstore) for scalability, fault tolerance, and high availability.
- Integrated into the Hadoop map-reduce platform and paradigm.

Benefits

- Distributed storage
- Table-like in data structure
  - multi-dimensional map
- High scalability
- High availability
- High performance

Who use HBase

- Adobe
  - 內部使用 (Structure data)
- Kalooga
  - 圖片搜尋引擎 http://www.kalooga.com/
- Meetup
  - 社群聚會網站 http://www.meetup.com/
- Streamy
  - Migrate from MySQL to Hbase http://www.streamy.com/
- Trend Micro
  - 雲端掃毒架構 http://trendmicro.com/
- Yahoo!
  - 儲存文件 fingerprint 避免重複 http://www.yahoo.com/
- More - http://wiki.apache.org/hadoop/Hbase/PoweredBy

Backdrop

- Started toward by Chad Walters and Jim
- 2006.11
  - Google releases paper on BigTable
- 2007.2
  - Initial HBase prototype created as Hadoop contrib.
- 2007.10
  - First useable HBase
- 2008.1
  - Hadoop become Apache top-level project and HBase becomes subproject
- 2008.10~
  - HBase 0.18, 0.19 released

HBase Is Not …

- Tables have one primary index, the *row key*.
- No join operators.
- Scans and queries can select a subset of available columns, perhaps by using a wildcard.
- There are three types of lookups:
  - Fast lookup using row key and optional timestamp.
  - Full table scan
  - Range scan from region start to end.

HBase Is Not …(2)

- Limited atomicity and transaction support.
  - HBase supports multiple batched mutations of single rows only.
  - Data is unstructured and untyped.
- No accessed or manipulated via SQL.
  - Programmatic access via Java, REST, or Thrift APIs.
  - Scripting via JRuby.

Why Bigtable?

- Performance of RDBMS system is good for transaction processing but for very large scale analytic processing, the solutions are commercial, expensive, and specialized.
- Very large scale analytic processing
  - Big queries – typically range or table scans.
  - Big databases (100s of TB)

Why Bigtable? (2)

- Map reduce on Bigtable with optionally Cascading on top to support some relational algebras may be a cost effective solution.
- Sharding is not a solution to scale open source RDBMS platforms
  - Application specific
  - Labor intensive (re)partitionaing

Why HBase ?

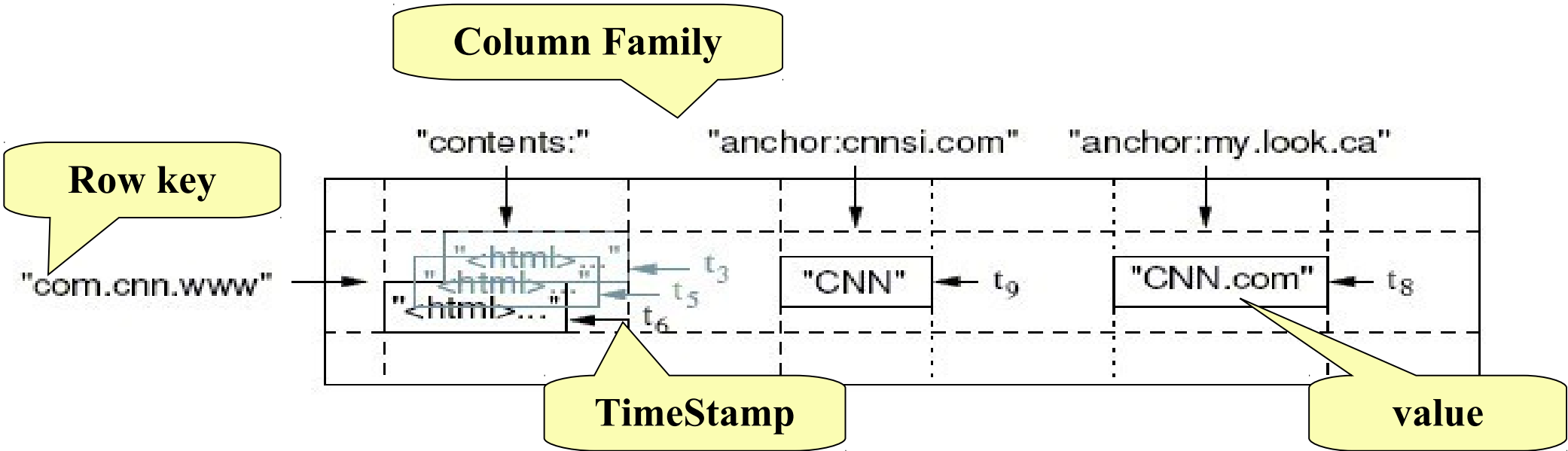
- HBase is a Bigtable clone.
- It is open source
- It has a good community and promise for the future
- It is developed on top of and has good integration for the Hadoop platform, if you are using Hadoop already.
- It has a Cascading connector.

HBase benefits than RDBMS

- *No real indexes*
- *Automatic partitioning*
- *Scale linearly and automatically with new nodes*
- *Commodity hardware*
- *Fault tolerance*
- *Batch processing*

Data Model

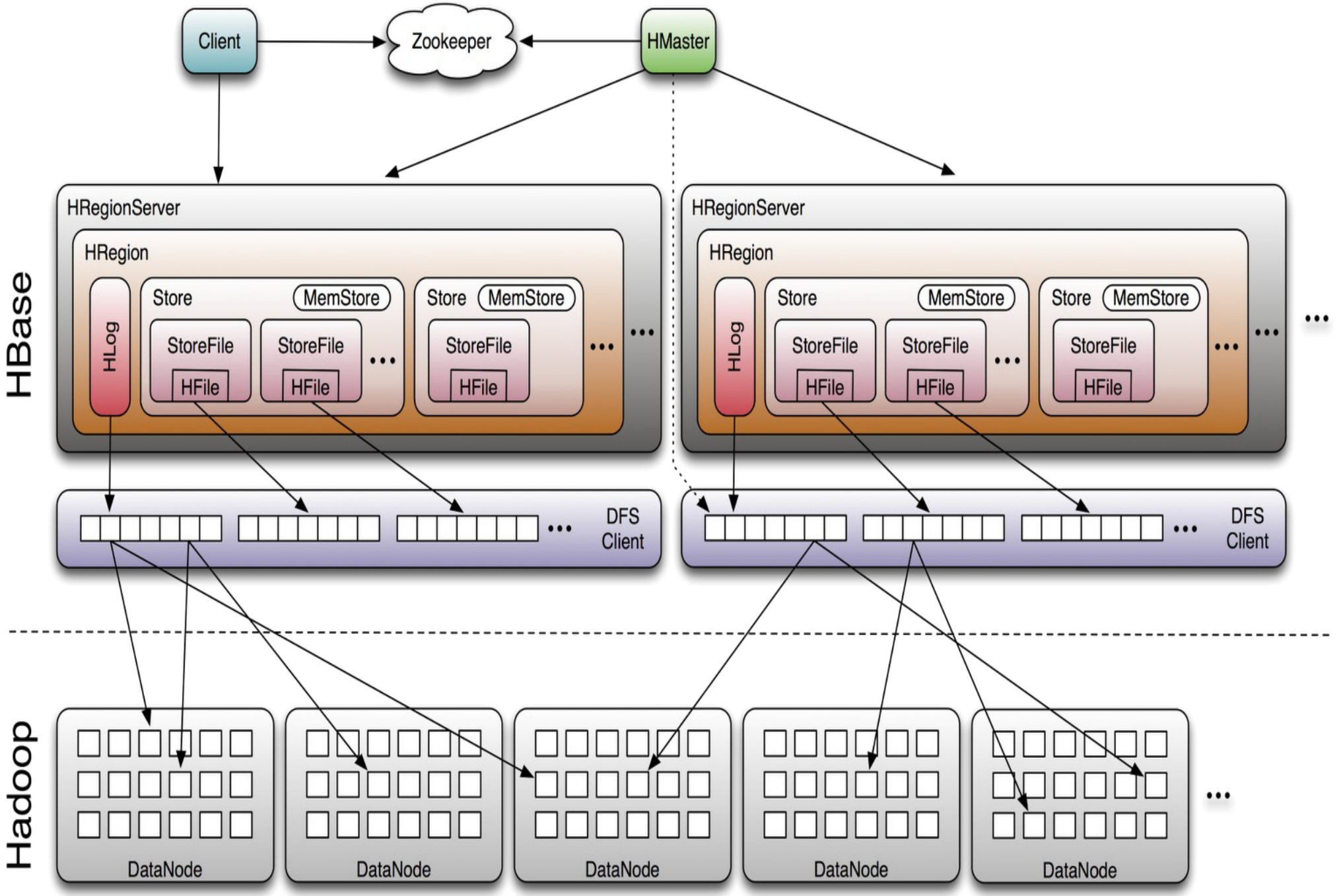
- Tables are sorted by Row
- Table schema only define it's *column families* .
  - Each family consists of any number of columns
  - Each column consists of any number of versions
  - Columns only exist when inserted, NULLs are free.
  - Columns within a family are sorted and stored together
- Everything except table names are byte[]
- (Row, Family: Column, Timestamp) → Value



Members

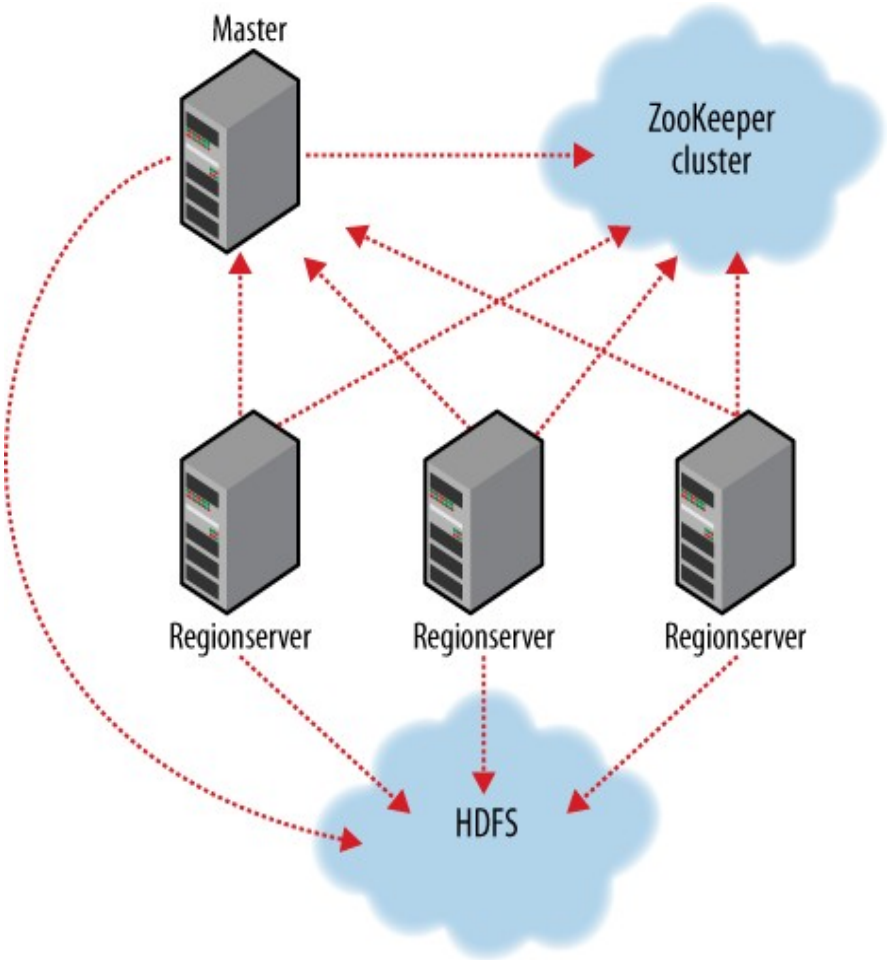
- *Master*
  - Responsible for monitoring region servers
  - Load balancing for regions
  - Redirect client to correct region servers
  - The current SPOF
- *regionserver* slaves
  - Serving requests(Write/Read/Scan) of Client
  - Send HeartBeat to Master
  - Throughput and Region numbers are scalable by region servers

Architecture

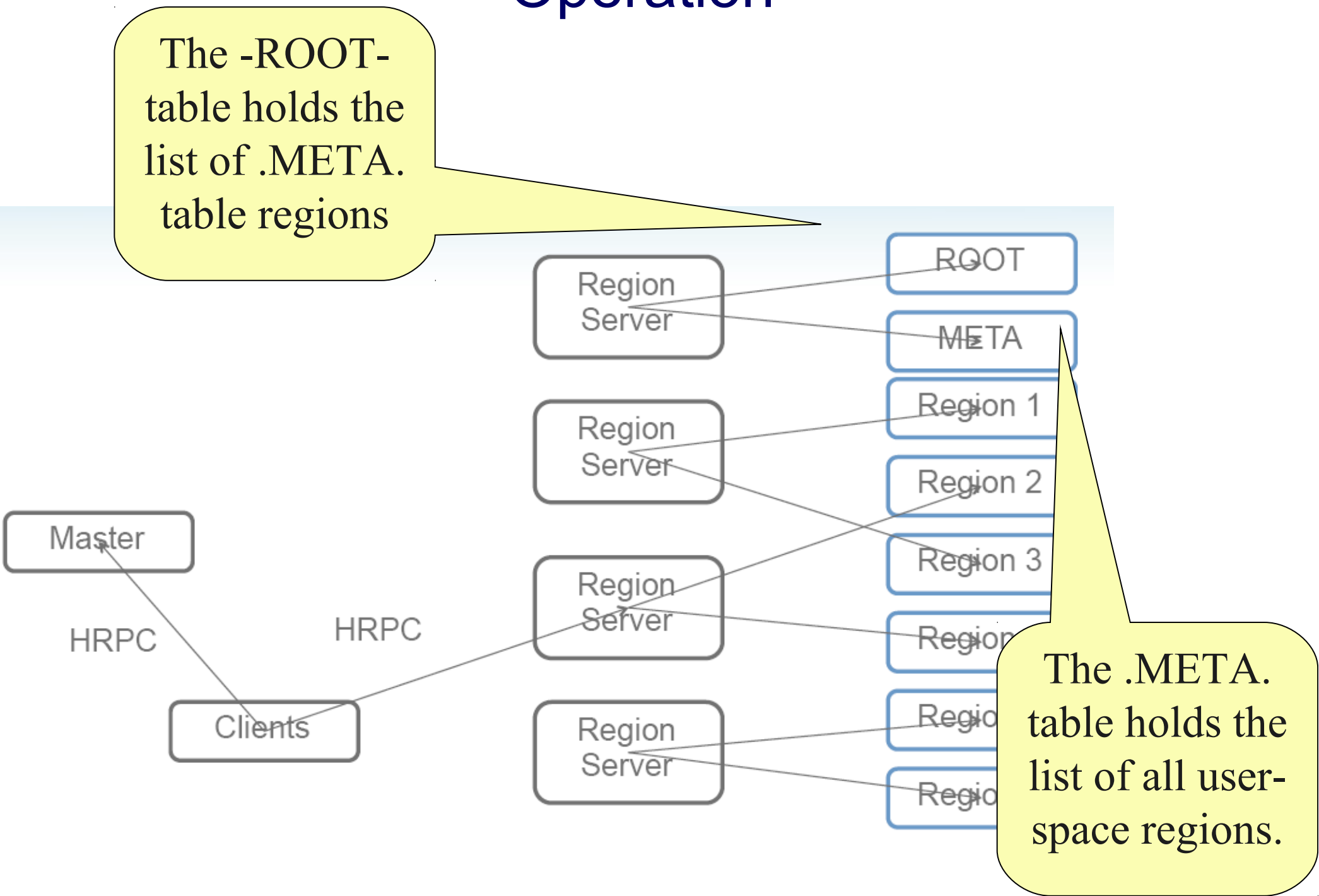


ZooKeeper

- HBase depends on ZooKeeper (Chapter 13) and by default it manages a ZooKeeper instance as the authority on cluster state



Operation


The -ROOT- table holds the list of .META. table regions
Region Server
Region Server
Region Server
Region Server
ROOT
META
Region 1
Region 2
Region 3
Region
Region
Region
Master
HRPC
HRPC
Clients
The .META. table holds the list of all user-space regions.

Questions?

Slides - http://trac.nchc.org.tw/cloud

Jazz Wang
Yao-Tsung Wang
jazz@nchc.org.tw

Powered by DRBL

Introduction to Pig programming

Yahoo Search Engineering

陳奕瑋 (**Yiwei Chen**)

任務！

殺很大、插很大(+瑤瑤寫真性感精選54P) @ osaki's Blog :: Xuite日誌

殺不用錢~~殺online瑤瑤性感變裝照+精選性感寫真童顏巨乳的娃娃音美少女瑤瑤 本名：郭書瑤 暱稱：瑤瑤身高：155cm 體重：42kg 三圍：33E/23/33 生日：1990/7/18 ...

blog.xuite.net/osaki99/blog/21865265 - 頁庫存檔 - 類似內容

電玩美少女瑤瑤精選影音(ヤオヤオ童顏Fカップ爆乳美少女映画videos ...

2008年8月31日 ... 18歲電玩少女瑤瑤半工半讀扛家計 (內有瑤瑤男友) http://blog.xuite.net /kaiger/daily/23136438 ... 20080913 我猜嗲嗲美少女第二段2號Kiki 3號瑤瑤 ...

blog.xuite.net/kaiger/daily/19128818 - 頁庫存檔 - 類似內容

顯示更多來自 blog.xuite.net 的結果

jays1943 分享正妹:NO.24 無名瑤瑤- 樂多日誌

瑤瑤也沒有哪裡得罪你們押你們為審麼這樣罵他說害女生生氣我看你們長的很醜吧不要自以為是喔死網友還罵人く死勒你要不要臉瑤瑤可是我的偶像你們最好是向一點 ...

blog.roodo.com/jays1943/archives/6850053.html - 頁庫存檔 - 類似內容

任務！

瑤瑤航空 - Powered by Discuz!

瑤瑤航空 - Discuz! Board ... 歡迎VIP旅客-魏如昀加入瑤瑤航空(2008-4-7) 歡迎VIP旅客-賴鋐偉加入瑤瑤航空(2008-3-13) ... 瑤瑤家族 瑤瑤在飛虎的第一家族 瑤瑤天空部落格 林佩瑤在天空的部落格 林佩瑤 無名網誌 瑤瑤的新照片都在無名啦! 無不癡齋 ...

www.yaoyaofly.com - 庫存頁面 - 更多此站結果

瑤瑤喵小屋~ - 無名小站

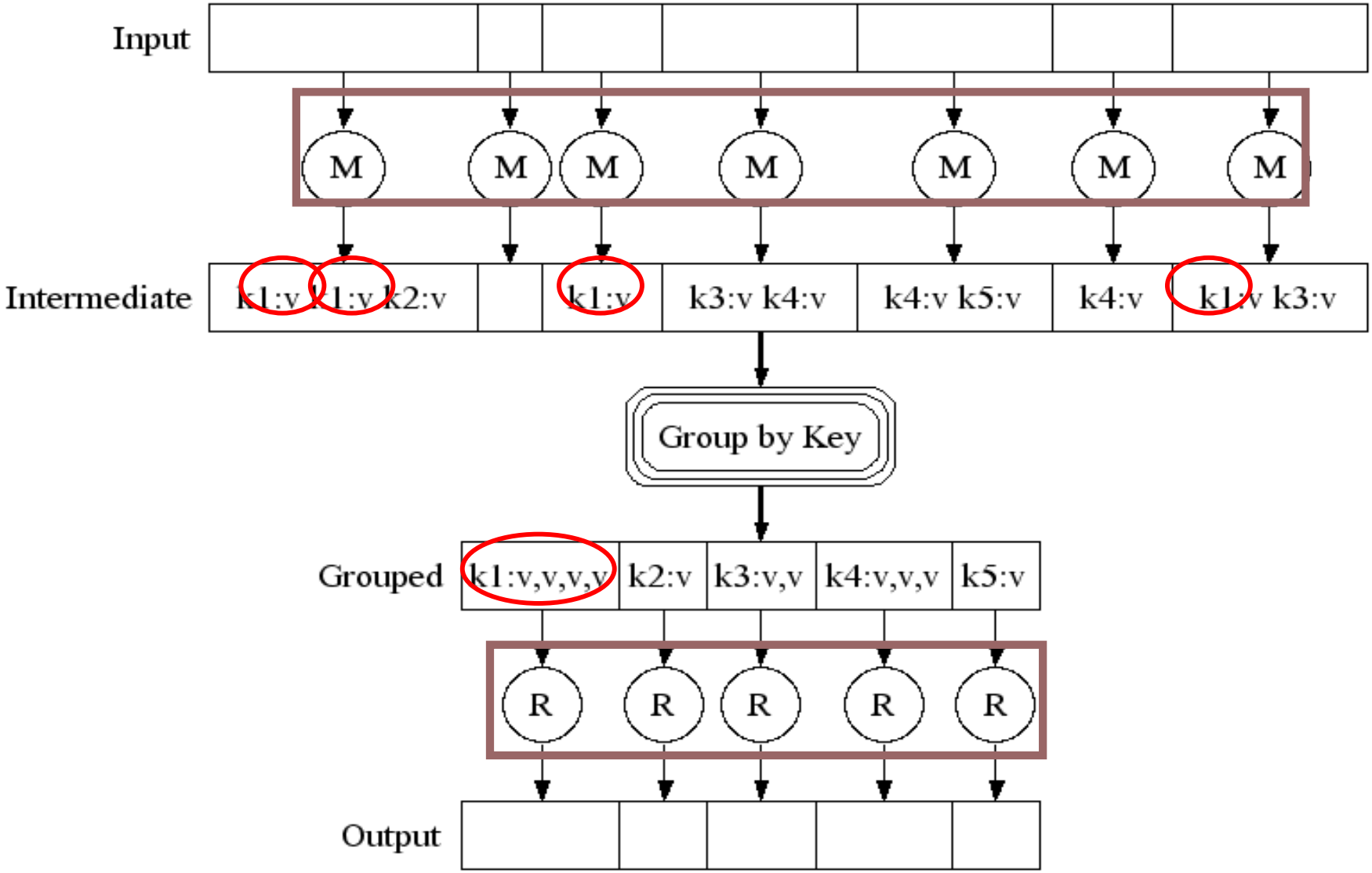
瑤瑤喵小屋~ - 無名小站 Blog Album... 最近好煩煩煩. 我覺得我的腦容量變小了... 好多事情消化不良 好多念頭讓我無法抉擇 (More.) goukigouki at 無名小站 at 02:39 PM post | Reply(27) | Trackback(0) | prosecute ...

www.wretch.cc/blog/goukigouki - 74k - 庫存頁面 - 更多此站結果

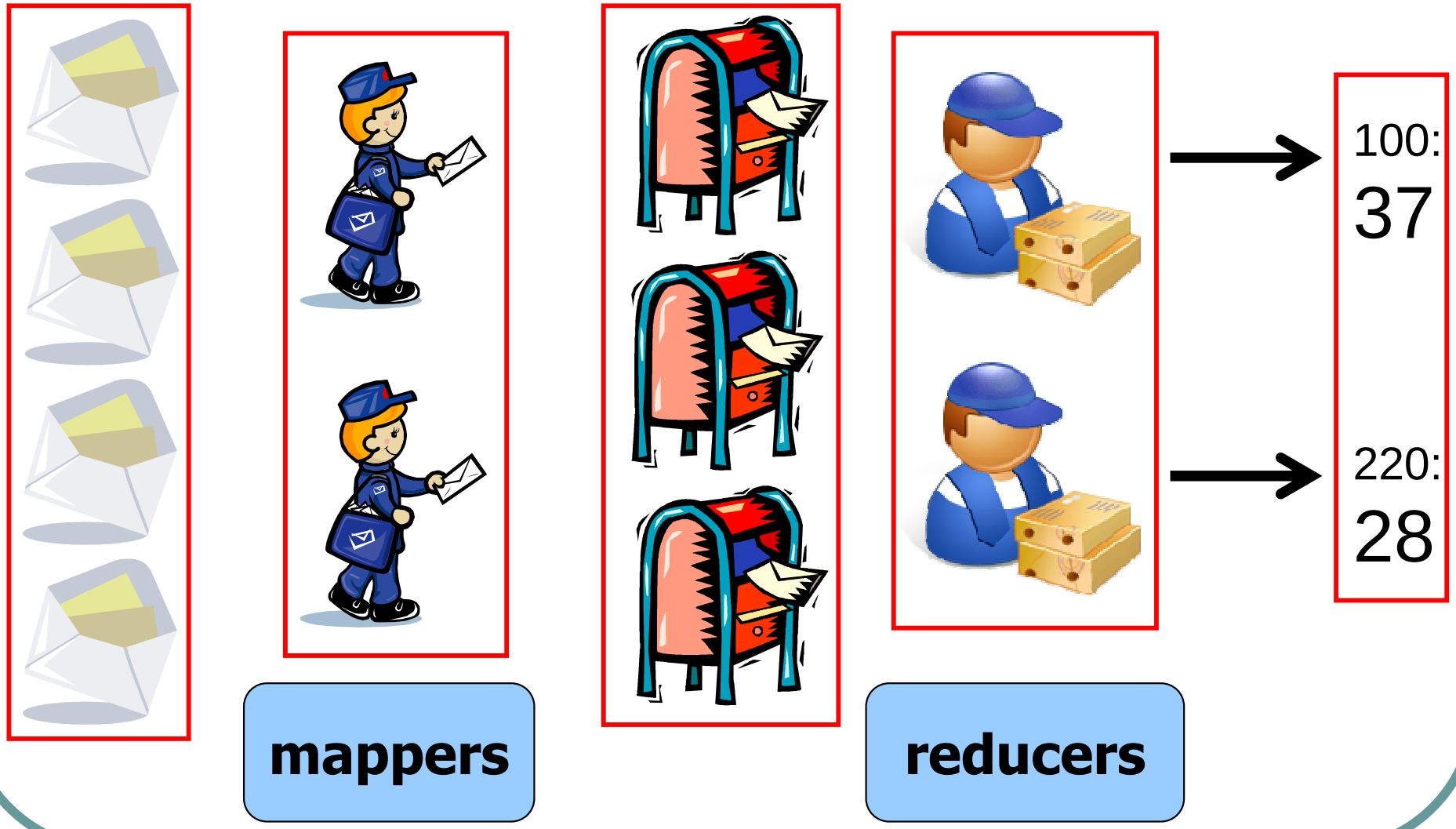
任務！

- 你怎麼知道我們放的網頁比較好？
- 你怎麼知道第一筆結果應該要多熱門？

Hadoop Programming – Map/Reduce



Map / Reduce



Map-Reduce

- 全新想法
- 須分別撰寫 mappers & reducers
- 會有超級無敵霹靂多的 mapper/reducer 要維護！

We usually do …

- 大部份時候:
  - filtering, projecting
  - grouping, aggregation, joining
- 今天有多少人搜尋「美國生」

Pig (Latin)

- Procedural dataflow language (Pig Latin) for Map-Reduce
  - 很像 SQL
    - group, join, filter, sort …
  - 人人都會 SQL

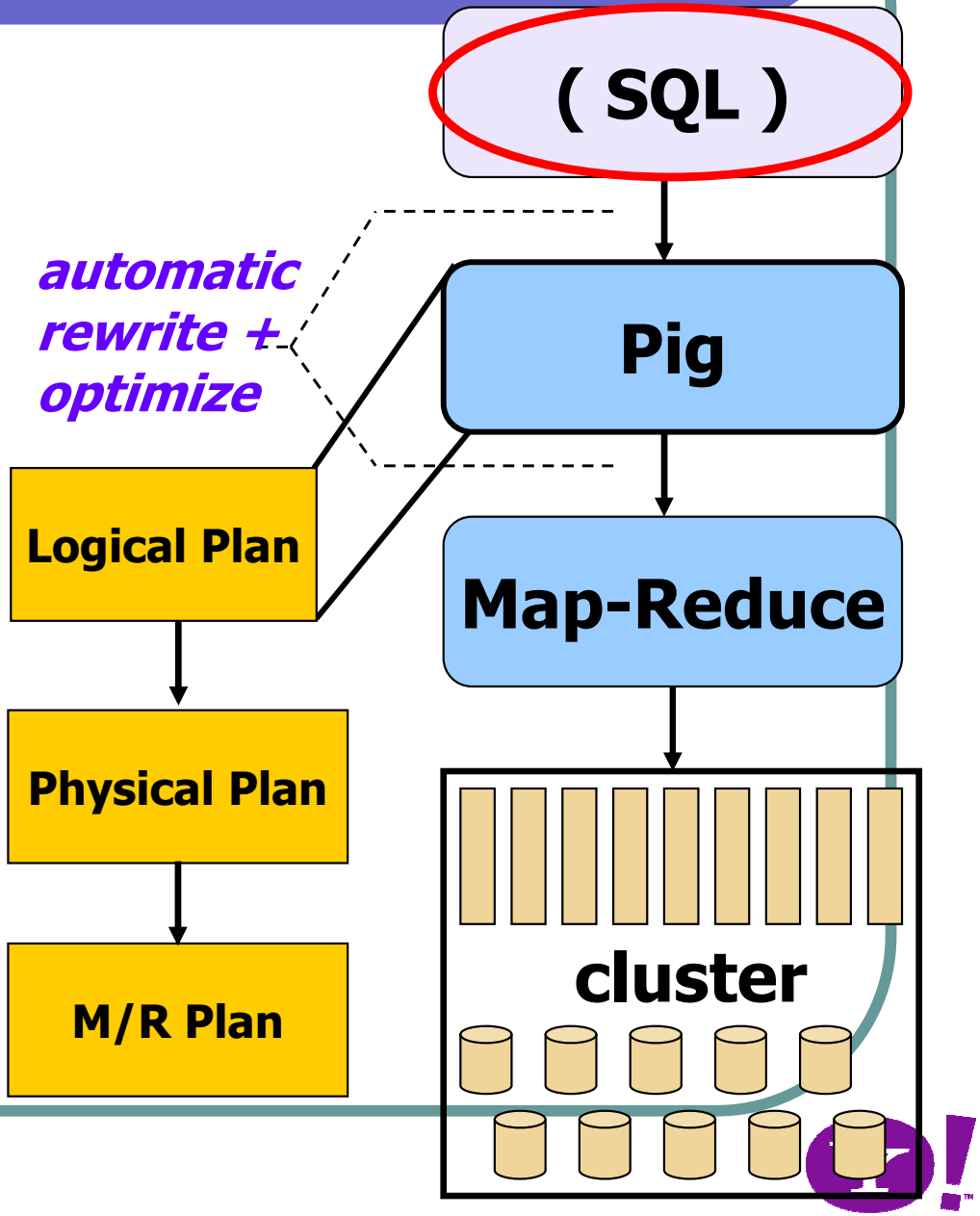
Pig Script Example

- 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



Why Pig?

- 容易學
- 開發快
- 一目瞭然

Why Pig?

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

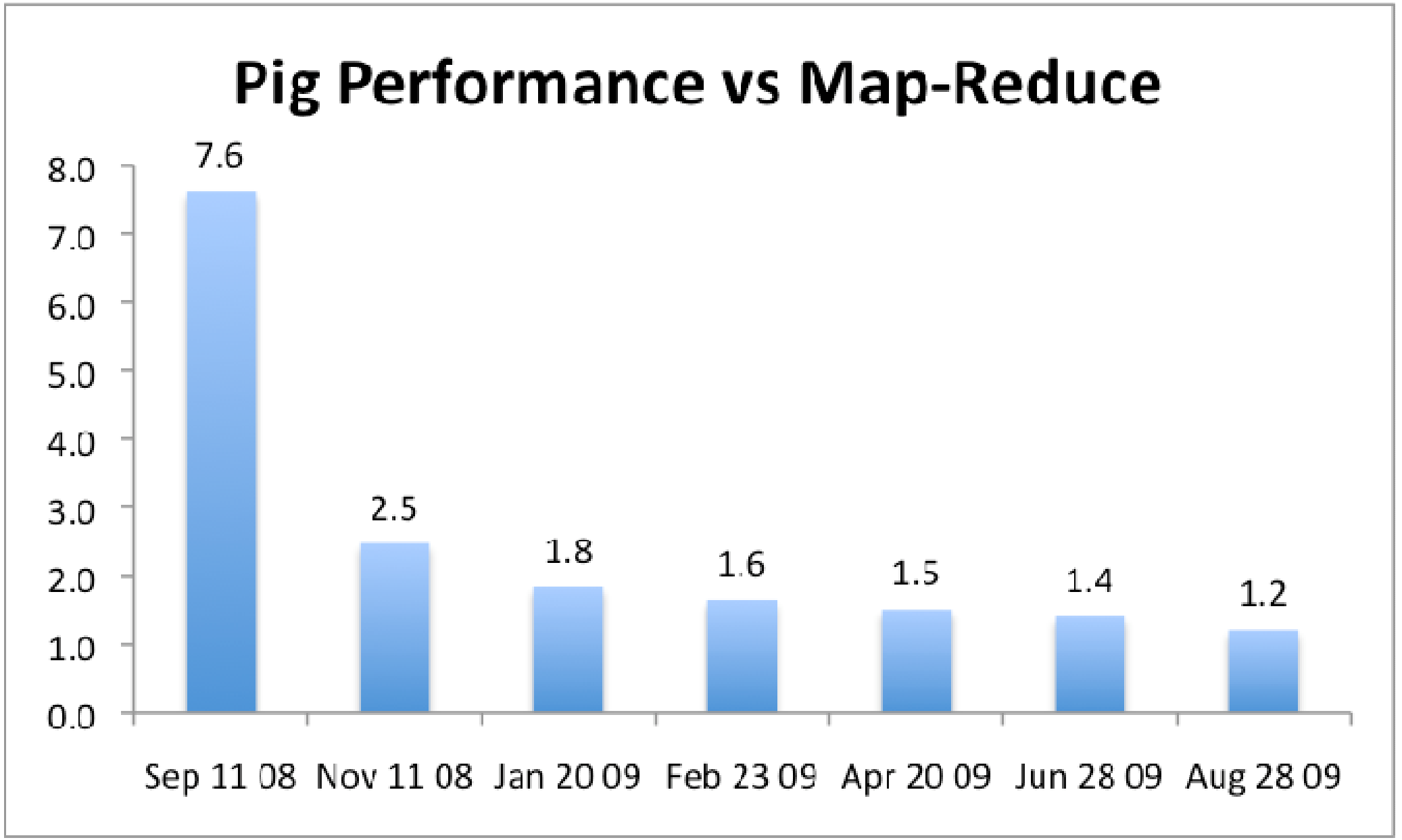
```
import java.io.IOException;
import java.util.ArrayList;
import java.util.Iterator;
import java.util.List;

import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.io.WritableComparable;
import org.apache.hadoop.mapred.FileInputFormat;
import org.apache.hadoop.mapred.FileOutputFormat;
import org.apache.hadoop.mapred.JobConf;
import org.apache.hadoop.mapred.KeyValueTextInputFormat;
import org.apache.hadoop.mapred.Mapper;
import org.apache.hadoop.mapred.MapReduceBase;
import org.apache.hadoop.mapred.OutputCollector;
import org.apache.hadoop.mapred.RecordReader;
import org.apache.hadoop.mapred.Reducer;
import org.apache.hadoop.mapred.Reporter;
import org.apache.hadoop.mapred.SequenceFileInputFormat;
import org.apache.hadoop.mapred.SequenceFileOutputFormat;
import org.apache.hadoop.mapred.TextInputFormat;
import org.apache.hadoop.mapred.jobcontrol.Job;
import org.apache.hadoop.mapred.jobcontrol.JobControl;
import org.apache.hadoop.mapred.lib.IdentityMapper;

public class MRExample {
    public static class LoadPages extends MapReduceBase
        implements Mapper<LongWritable, Text, Text, Text> {

        public void map(LongWritable k, Text val,
                OutputCollector<Text, Text> oc,
                Reporter reporter) throws IOException {
            // Pull the key out
            String line = val.toString();
            int firstComma = line.indexOf(',');
            String key = line.substring(0, firstComma);
            String value = line.substring(firstComma + 1);
            Text outKey = new Text(key);
            // Prepend an index to the value so we know which file
            // it came from.
            Text outVal = new Text("1" + value);
            oc.collect(outKey, outVal);
        }
    }
    public static class LoadAndFilterUsers extends MapReduceBase
        implements Mapper<LongWritable, Text, Text, Text> {

        public void map(LongWritable k, Text val,
                OutputCollector<Text, Text> oc,
                Reporter reporter) throws IOException {
            // Pull the key out
            String line = val.toString();
            int firstComma = line.indexOf(',');
            String value = line.substring(firstComma + 1);
            int age = Integer.parseInt(value);
            if (age < 18 || age > 25) return;
            String key = line.substring(0, firstComma);
            Text outKey = new Text(key);
            // Prepend an index to the value so we know which file
            // it came from.
            Text outVal = new Text("2" + value);
            oc.collect(outKey, outVal);
        }
    }
    public static class Join extends MapReduceBase
        implements Reducer<Text, Text, Text, Text> {

        public void reduce(Text key,
                Iterator<Text> iter,
                OutputCollector<Text, Text> oc,
                Reporter reporter) throws IOException {
            // For each value, figure out which file it's from and
store it
            // accordingly.
            List<String> first = new ArrayList<String>();
            List<String> second = new ArrayList<String>();

            while (iter.hasNext()) {
                Text t = iter.next();
                String value = t.toString();
                if (value.charAt(0) == '1')
first.add(value.substring(1));
                else second.add(value.substring(1));
                reporter.setStatus("OK");
            }

            // Do the cross product and collect the values
            for (String s1 : first) {
                for (String s2 : second) {
                    String outval = key + "," + s1 + "," + s2;
                    oc.collect(null, new Text(outval));
                    reporter.setStatus("OK");
                }
            }
        }
    }
    public static class LoadJoined extends MapReduceBase
        implements Mapper<Text, Text, Text, LongWritable> {

        public void map(
                Text key,
                Text val,
                OutputCollector<Text, LongWritable> oc,
                Reporter reporter) throws IOException {
            // Find the url
            String line = val.toString();
            int firstComma = line.indexOf(',');
            int secondComma = line.indexOf(',', firstComma);
            String key = line.substring(firstComma, secondComma);
            // drop the rest of the record, don't need it anymore,
            // just pass a 1 for the combiner/reducer to sum instead.
            Text outKey = new Text(key);
            oc.collect(outKey, new LongWritable(1L));
        }
    }
    public static class ReduceUrls extends MapReduceBase
        implements Reducer<Text, LongWritable, WritableComparable,
Writable> {

        public void reduce(
                Text key,
                Iterator<LongWritable> iter,
                OutputCollector<WritableComparable, Writable> oc,
                Reporter reporter) throws IOException {
            // Add up all the values we see

            long sum = 0;
            while (iter.hasNext()) {
                sum += iter.next().get();
                reporter.setStatus("OK");
            }

            oc.collect(key, new LongWritable(sum));
        }
    }
    public static class LoadClicks extends MapReduceBase
        implements Mapper<WritableComparable, Writable, LongWritable,
Text> {

        public void map(
                WritableComparable key,
                Writable val,
                OutputCollector<LongWritable, Text> oc,
                Reporter reporter) throws IOException {
            oc.collect((LongWritable)val, (Text)key);
        }
    }
    public static class LimitClicks extends MapReduceBase
        implements Reducer<LongWritable, Text, LongWritable, Text> {

        int count = 0;
        public void reduce(
            LongWritable key,
            Iterator<Text> iter,
            OutputCollector<LongWritable, Text> oc,
            Reporter reporter) throws IOException {

            // Only output the first 100 records
            while (count < 100 && iter.hasNext()) {
                oc.collect(key, iter.next());
                count++;
            }
        }
    }
    public static void main(String[] a
        JobConf lp = new JobConf(MRExa
        lp.setJobName("Load Pages");
        lp.setInputFormat(TextInputFor
        lp.setOutputKeyClass(Text.clas
        lp.setOutputValueClass(Text.cl
        lp.setMapperClass(LoadPages.cl
        FileInputFormat.addInputPath(l
Path("/user/gates/pages"));
        FileOutputFormat.setOutputPath
            new Path("/user/gates/tmp/
        lp.setNumReduceTasks(0);
        Job loadPages = new Job(lp);

        JobConf lfu = new JobConf(MRExa
        lfu.setJobName("Load and Filte
        lfu.setInputFormat(TextInputFo
        lfu.setOutputKeyClass(Text.cla
        lfu.setOutputValueClass(Text.c
        lfu.setMapperClass(LoadAndFilt
        FileInputFormat.addInputPath(l
Path("/user/gates/users"));
        FileOutputFormat.setOutputPath
            new Path("/user/gates/tmp/
        lfu.setNumReduceTasks(0);
        Job loadUsers = new Job(lfu);

        JobConf join = new JobConf(MRE
        join.setJobName("Join Users an
        join.setInputFormat(KeyValueTe
        join.setOutputKeyClass(Text.cl
        join.setOutputValueClass(Text.
        join.setMapperClass(IdentityMa
        join.setReducerClass(Join.clas
        FileInputFormat.addInputPath(j
Path("/user/gates/tmp/indexed_pages"))
        FileInputFormat.addInputPath(
Path("/user/gates/tmp/filtered_users"
        FileOutputFormat.setOutputPath
Path("/user/gates/tmp/joined"));
        join.setNumReduceTasks(50);
        Job joinJob = new Job(join);
        joinJob.addDependingJob(loadPa
        joinJob.addDependingJob(loadUs

        JobConf group = new JobConf(MR
        group.setJobName("Group URLs")
        group.setInputFormat(KeyValueT
        group.setOutputKeyClass(Text.c
        group.setOutputValueClass(Long
        group.setOutputFormat(Sequence
        group.setMapperClass(LoadJoine
        group.setCombinerClass(ReduceU
        group.setReducerClass(ReduceUr
        FileInputFormat.addInputPath(g
Path("/user/gates/tmp/joined"));
        FileOutputFormat.setOutputPath
Path("/user/gates/tmp/grouped"));
        group.setNumReduceTasks(50);
        Job groupJob = new Job(group);
        groupJob.addDependingJob(joinJ

        JobConf top100 = new JobConf(MR
        top100.setJobName("Top 100 sit
        top100.setInputFormat(Sequence
        top100.setOutputKeyClass(LongW
        top100.setOutputValueClass(Tex
        top100.setOutputFormat(Sequenc
        top100.setMapperClass(LoadClic
        top100.setCombinerClass(LimitC
        top100.setReducerClass(LimitCl
        FileInputFormat.addInputPath(t
Path("/user/gates/tmp/grouped"));
        FileOutputFormat.setOutputPath
Path("/user/gates/top100sitesforusers1
        top100.setNumReduceTasks(1);
        Job limit = new Job(top100);
        limit.addDependingJob(groupJob

        JobControl jc = new JobControl
18 to 25");
        jc.addJob(loadPages);
```

Why (NOT) Pig?

- 不是史上究極霹靂大無敵武器
  - Focus: aggregation, filter, join,…
- 另一種做分散運算工作的方式

Sweet spot between SQL – M/R

	SQL	Pig	Map-Reduce
Programming style	Large blocks of declarative constraints	➔	"Plug together pipes"
Built-in data manipulations	Group-by, Sort, Join, Filter, Aggregate, Top-k, etc...	➔	Group-by, Sort
Execution model	Fancy; trust the query optimizer	➔	Simple, transparent
Opportunities for automatic optimization	Many	➔	Few (logic buried in map() and reduce())
Data Schema	Must be known at table creation	➔	Not required, may be defined at runtime


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

Execution and Syntax

Pig Example

- Show users aged 18-25

```
Users = LOAD 'users.txt'
        USING PigStorage(',') AS (name, age);
Fltrd = FILTER Users
        BY age >= 18 AND age <= 25;
Names = FOREACH Fltrd GENERATE name;

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

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

Features

Parameter Substitution

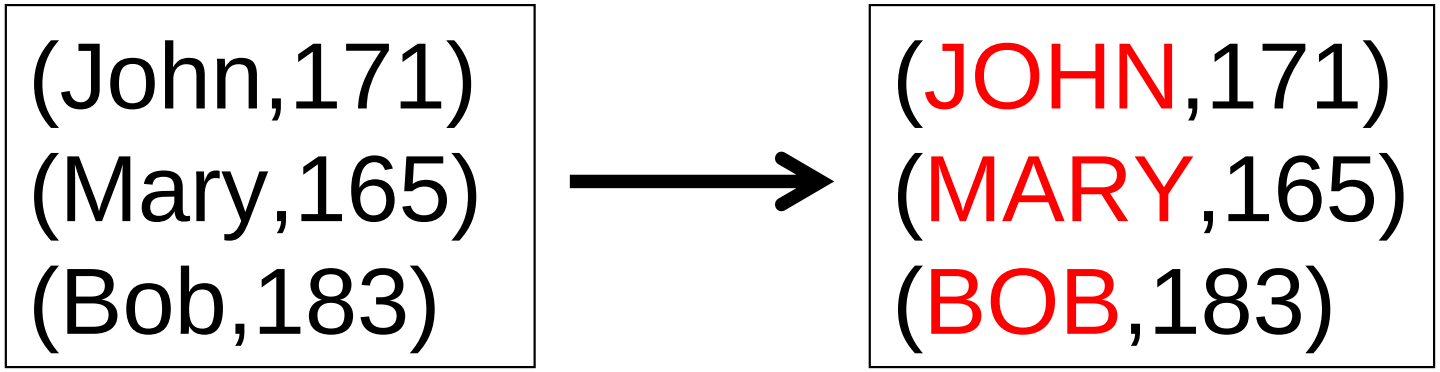
```
%default TYPE 'view'
%declare ID '18987'
A = load '/data/$DATE/$ID/$TYPE‘
```

- $ pig a.pig
- $ pig -param DATE=20091009 a.pig
- $ pig -param DATE=20091009 –param TYPE=click a.pig

RegEx Comparison

- `itsyou = FILTER urls by ($0 MATCHES 'http://.*\\.yahoo\\.com.*')`
- MATCHES matches ‘whole’ string
  - `‘aaaa’ MATCHES ‘aaa.*’` is true
  - `‘bbaaaa’ MATCHES ‘aaa.*’` is false
- pattern syntax: java.util.regex.Pattern

User-defined Function (UDF)



UDF – user function part

```
package myudf;
import java.io.IOException;
import org.apache.pig.EvalFunc;
import org.apache.pig.data.Tuple;

public class UPPER extends EvalFunc<String>
{
    public String exec(Tuple in) throws IOException {
        if (in == null || in.size() == 0) return null;
        String str = (String)in.get(0);
        return str.toUpperCase();
    }
}
```

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

Embedded in Java

```
/* create a pig server in the main class*/
{
  PigServer pigserver = new PigServer(args[0]);
   runMyQuery(pigServer, "/user/viraj/mydata.txt")
}

/* submit in function runMyQuery */

runMyQuery(PigServer pigServer, String inputFile) throws
IOException {
        pigServer.registerQuery("A = load '" + inputFile +
"' as (f1,f2,f3);");
        pigServer.registerQuery("B = group A by f1;");
        pigServer.registerQuery("C = foreach B generate
flatten(group);");
        pigServer.store("C", "/user/viraj/myoutput");
}
```

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

Questions?

Backup slides

Parameter Substitution

```
$ pig -param myparam=val foo.pig
```

```
B = filter A by ($0 eq '$myparam');
```

- `pig -dryrun` produces processed script

```
B = filter A by ($0 eq 'val');
```

Parameter Substitution

- Params in file instead of command line
- `$ pig -param_file myparams.txt a.pig`

```
#myparams.txt
DATE=20081009
TYPE=clicks
```

UDF – build user function

- javac
  -cp $PIG_HOME/lib/pig.jar
  -sourcepath src
  -d classes
  src/myudf/UPPER.java
- jar cf myudf.jar -C classes myudf/UPPER.class

UDF – pig latin part

```
register myudf.jar;
B =
foreach A generate
  myudf.UPPER(name),height;
```

SQL vs. Pig Latin

SQL	Pig	Example
From table	Load file(s)	**SQL**: from X; **Pig**: A = load 'mydata' using PigStorage('\t') as (col1, col2, col3);
Select	Foreach … generate	**SQL**: select col1 + col2, col3 … **Pig**: B = foreach A generate col1 + col2, col3;
Where	Filter	**SQL**: select col1 + col2, col3 from X where col2>2; **Pig**: C = filter B by col2 > '2';

(adapted from Viraj's slide)

SQL vs. Pig Latin

SQL	Pig	Example
Group by	Group + foreach … generate	**SQL:** select col1, col2, sum(col3) from X group by col1, col2; **Pig**: D = group A by (col1, col2); E = foreach D generate flatten(group), SUM(A.col3);
Having	Filter	**SQL**: select col1, sum(col2) from X group by col1 having sum(col2) > 5; **Pig**: F = filter E by $1 > '5';
Order By	Order … By	**SQL**: select col1, sum(col2) from X group by col1 order by col1; **Pig**: H = ORDER E by $0;

(adapted from Viraj's slide)

SQL vs. Pig Latin

SQL	Pig	Example
Distinct	Distinct	**SQL:** select distinct col1 from X; **Pig:** I = foreach A generate col1; J = distinct I;
Distinct Agg	Distinct in foreach	**SQL:** select col1, count (distinct col2) from X group by col1; **Pig:** K = foreach D { L = distinct A.col2; generate flatten(group), SUM(L); }

(adapted from Viraj's slide)

SQL vs. Pig Latin

SQL	Pig	Example
Join	**Cogroup + flatten** **(also shortcut: JOIN)**	**SQL:** select A.col1, B.col3 from A join B using (col1); **Pig:** A = load 'data1' using PigStorage('\t') as (col1, col2); B = load 'data2' using PigStorage('\t') as (col1, col3); C = cogroup A by col1 **inner**, B by col1 **inner**; D = foreach C generate flatten(A), flatten(B); E = foreach D generate A.col1, B.col3;

(adapted from Viraj's slide)

Debug Tips

- Use small data and `pig -x local`
- `LIMIT`
  - `A = LOAD 'data' AS (a1,a2,a3)`
  - `B = LIMIT A 3;`
- `DUMP , DESCRIBE`

FAQ

- http://wiki.apache.org/pig/FAQ
  - can assign #reducer
  - support regex
  - can use allocated HOD cluster

pig.vim

- http://www.vim.org/scripts/script.php?script_id=2186

```
A = load 'data.txt' as (f1,f2,f3);
dump A;
B = foreach A generate f1,f3;
dump B;
store B into 'output.txt' using PigStorage('-');
```

Crawlzilla - A Toolkit for Deploying Cluster Search Engine Quickly and Easily

Shun-Fa Yang、Wei-Yu Chen、Wen-Chieh Kuo

Free Software Lab.@ NCHC

INVENSIVE 2011 May 23, 2011

TAIWAN www.nchc.org.tw National Applied Research Laboratories NCHC

Outline

Crawlzilla
- Feature
- System Implement
- Demo

Background and Motivation
- Nutch
- Hadoop
- Search Engine Library

Performance

Introduction

Future Works

Introduction

- The Information Explosion
- Increase Filter Efficiency by Search Engines
- Intranet also need Search Engines
- Build Search Engines isn't very Easy
- Crawlzilla can help You!

Outline

Crawlzilla
- Feature
- System Implement
- Demo

Background and Motivation
- Nutch
- Hadoop
- Search Engine Library

Performance

Introduction

Future Works

Background and Motivation

Search Engine workflow

Related Open Source Projects

Compare with Other Projects

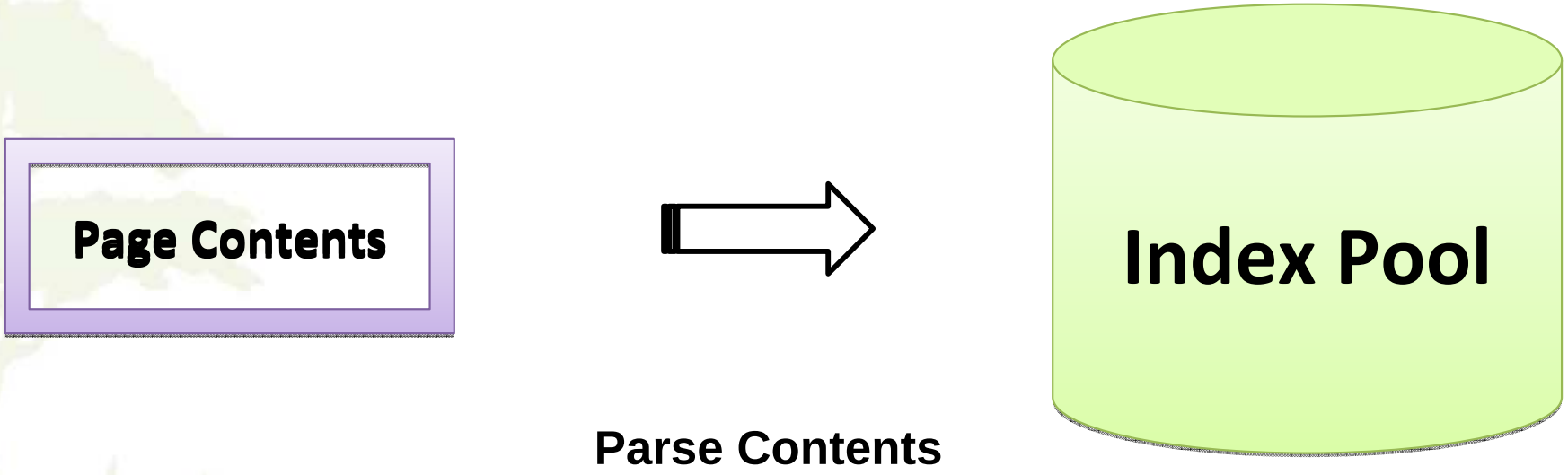
Search Engine workflow – Phase 1

- **Crawling the Web**

Crawler visits the web pages of the links

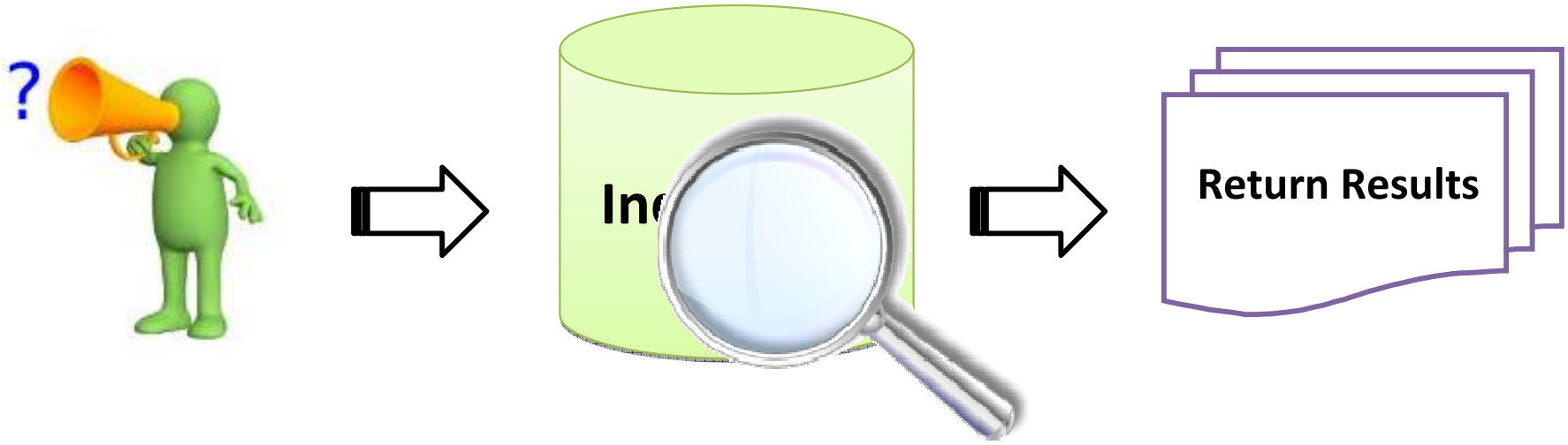
Search Engine workflow – Phase 2

- **Building the Index Pool**



Search Engine workflow – Phase 3

- **Serving Queries**



User Sent a Query **Search from Index Pool**

Background and Motivation

- **Related Open Source Projects**
  - Search Engine - Ntuch
  - Distributed Computing Platform – Hadoop
  - Search Engine Library – Lucene

Background and Motivation

- **If Build Search by Yourself …**
  - Setup Hadoop
  - Deploy System Configure Files
  - Debug Errors…
  - …
  - …
  - …

Compare with Other Projects

	Spidr	Larbin	Jcrawl	Nutch	Crawlzilla
Install	Rube Package Install	Gmake Compiler and Install	Java Compiler and Install	Deploy Configure Files	**Provide Auto Installation**
Crawl website pages	O	O	O	O	**O**
Parser Content	X	X	X	O	**O**
Cluster Computing	X	X	X	O	**O**
Interface	Command	Command	Command	Command	**Web-UI**
Support Chinese Segmentation	X	X	X	X	**O**

Goal

- To Help Users to Build Search Engines Easily!
- To Help Users to Operate System Easily!
- Crawlzilla doesn’t improve the algorithm of Nutch and Hadoop!
- Crawlzilla Provides Friendly Operating Interface and an Easy Way to Deploy Cluster Computing Environment!

Outline

Crawlzilla
- Feature
- System Implement
- Demo

Background and Motivation
- Nutch
- Hadoop
- Search Engine Library

Performance

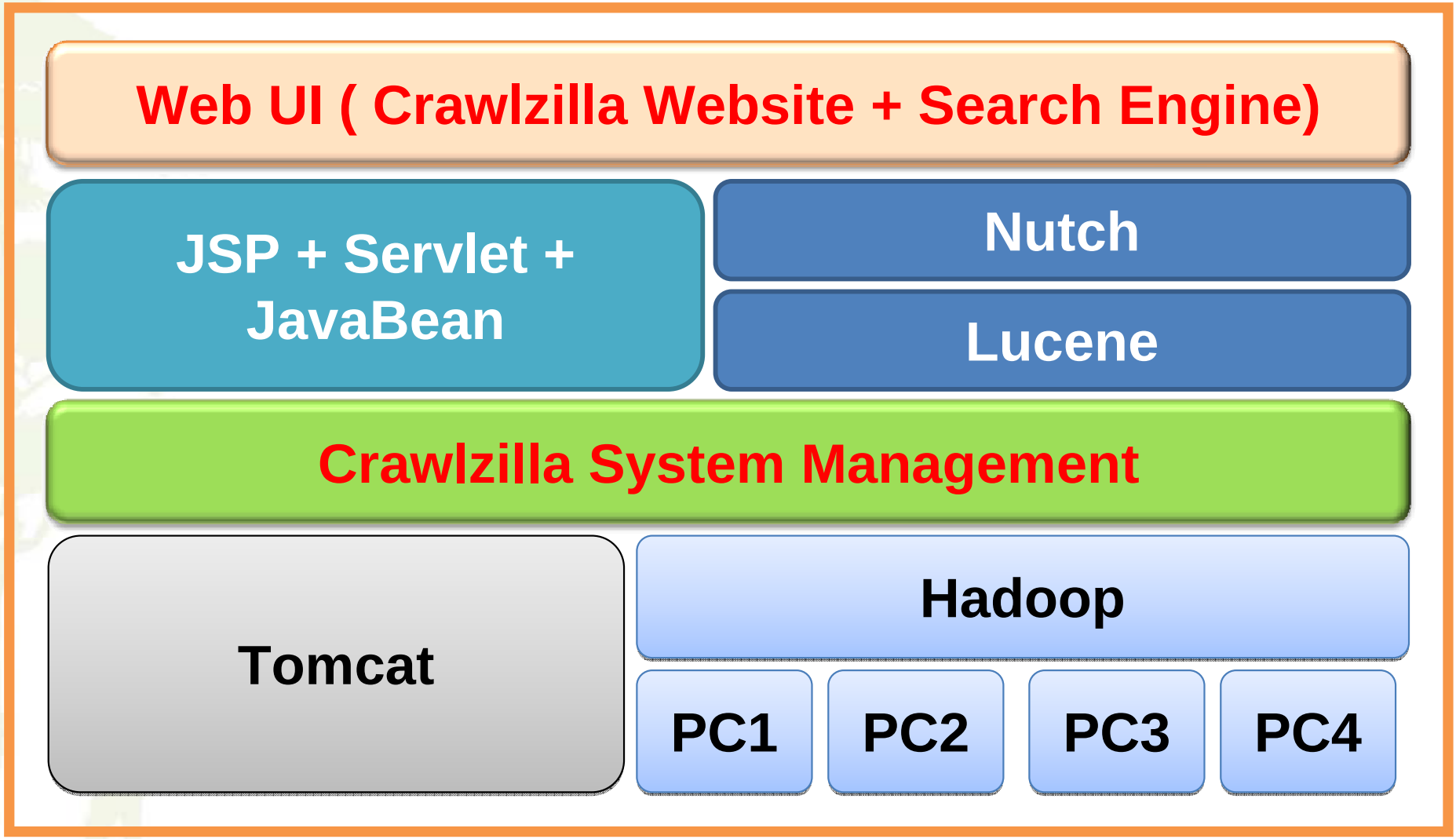
Introduction

Future Works

Crawlzilla Feature

- **Simply Install and Easy to Operate**
  - Customize user interface
- **More Powerful**
  - Support multiple search engines
- **More Search Engine Info.**
- **Developers to focus more**
  - Data mining tools

Crawlzilla Architecture



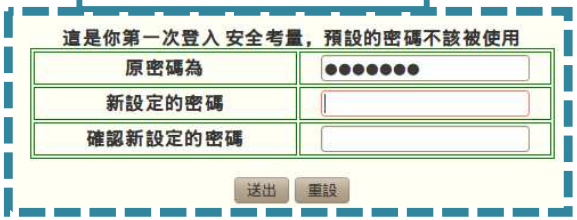
System Implement

	JSP	Shell Script
Function	User UI	Admin and MIS UI
Security	Website Session	Crawler password with RSA Keys
Environment	Browser	Terminal with SSH –Client
Architecture	MVC	Module
Multi Language	i18n	Language parameters
	Default language is depend on O.S. Env.	

Web Management

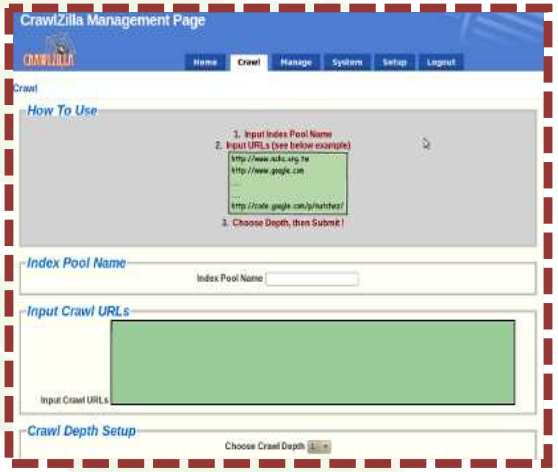
(Model 2)

Setup PW



Capture Lucene index pool

Setup and Drive Crawl Procedure



Servlet

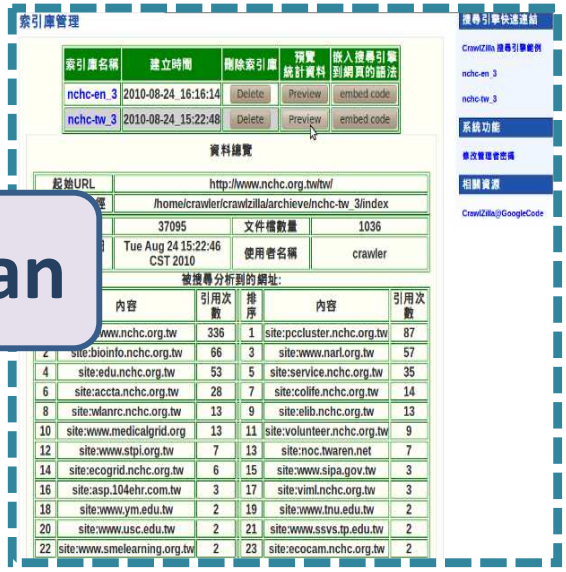
JavaBean

Control

Model

View

JSP

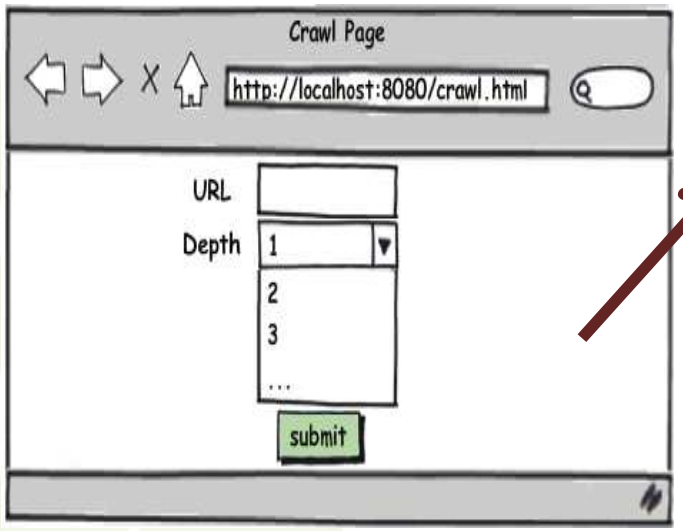


System Status

Session Certification

i18N language setup

System Implement – Web Parser



Servlet

Java Beans

JSP

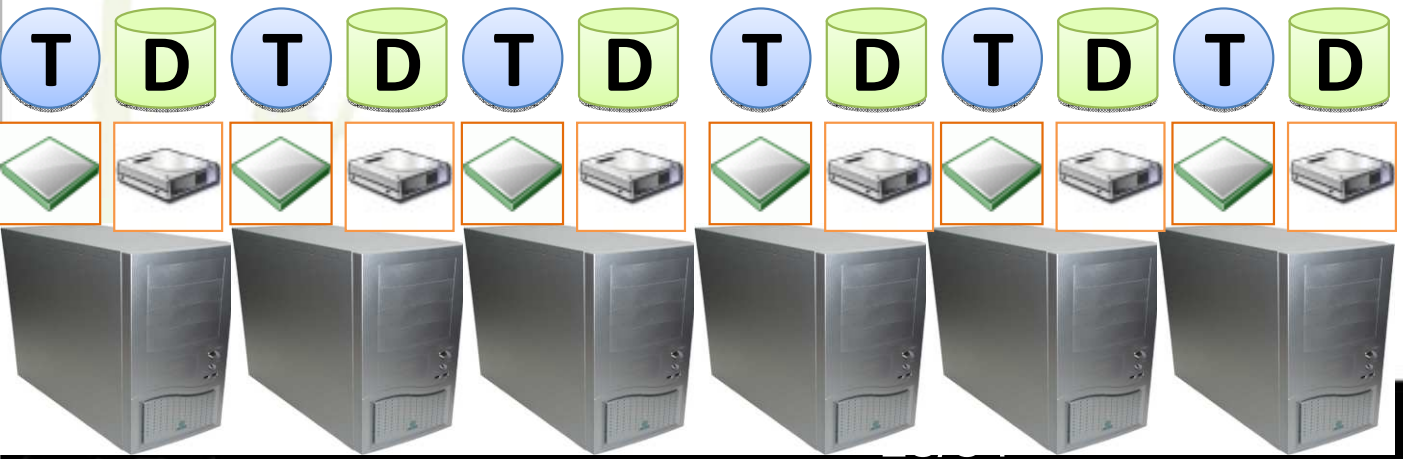
Nutch

Lucene Index DB

Hadoop

JobTracker

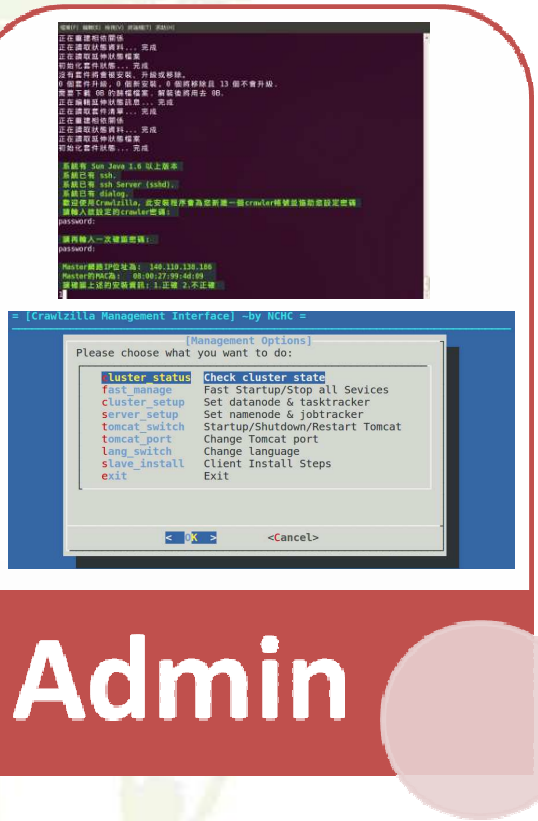
NameNode



T — TaskTracker: Job Executor

D — DataNode: Data Storage Node

Friendly Interface!



Admin

MIS

USER

Live Demo I

Crawlzilla Install

(1) Master Install
(2) Cluster Slave Install

~~Live~~ Video Demo

- **Master Install (Demo Video also @ YouTube)**

~~Live~~ Video Demo

- **Slave Install (Demo Video also @ YouTube)**

Live Demo II

Dialog Management

Live Demo III

Web Management

(1) Crawl Setup

(2)Search Engine Index Pool

(3)Search it!

Outline

Crawlzilla
- Feature
- System Implement
- Demo

Background and Motivation
- Nutch
- Hadoop
- Search Engine Library

Performance

Introduction

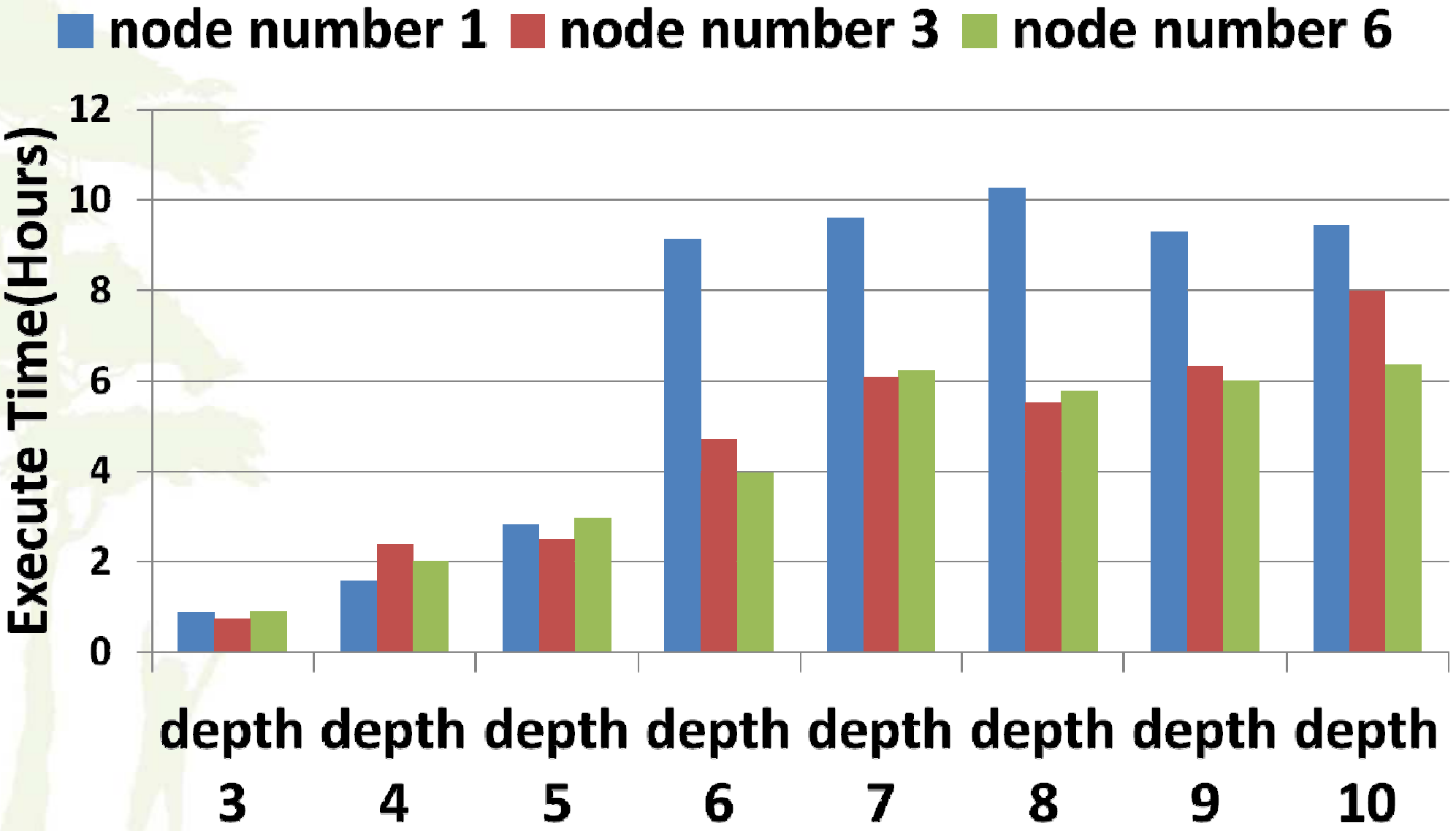
Future Works

Performance

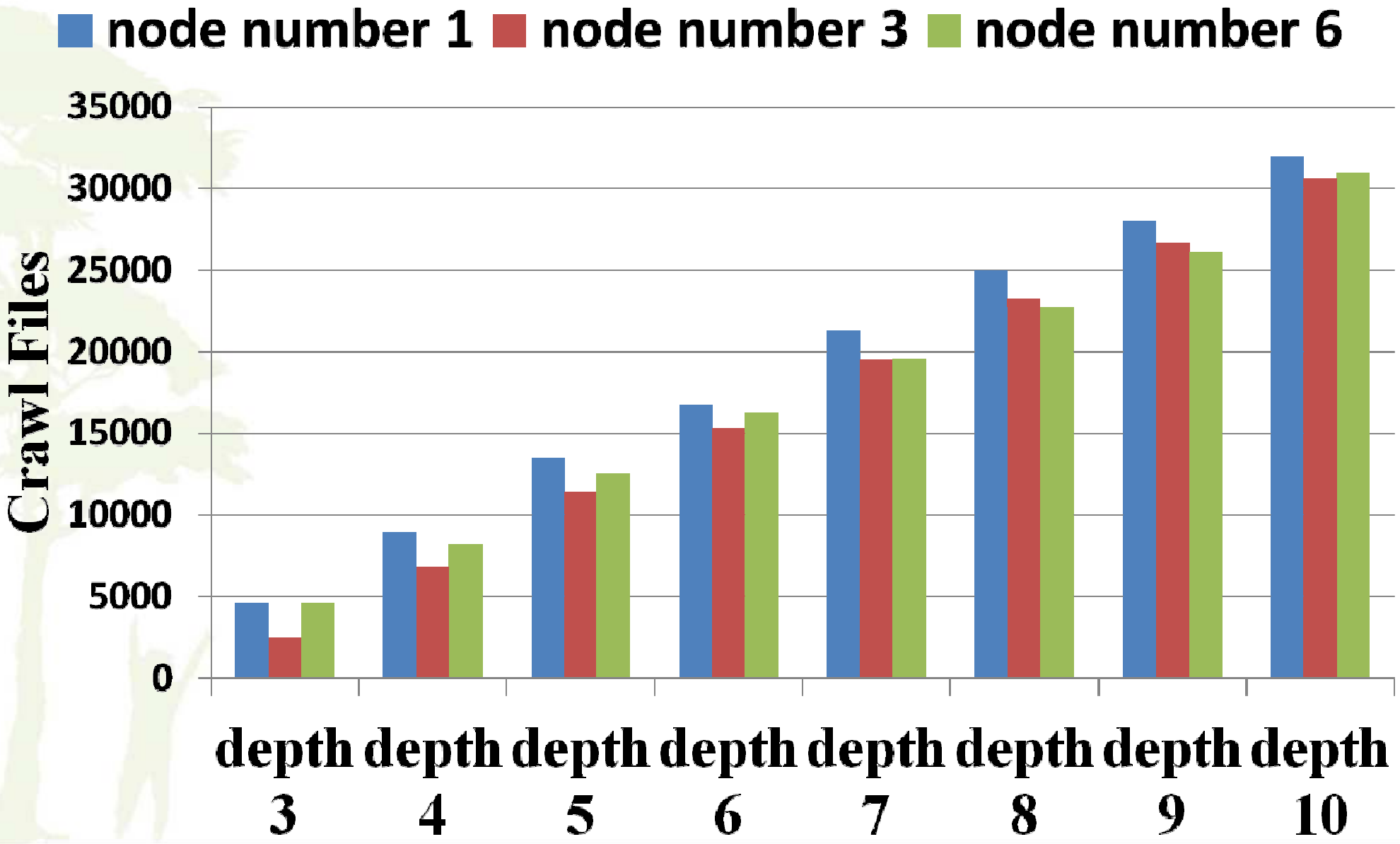
Experiment Environment

- CPU
  - Intel(R) Core(TM)2 Quad CPU Q9550 2.83GHz
- Memroy
  - 8 GigaBytes
- Operation System
  - Ubuntu 10.04 Lucid(x86)
- Crawlzilla Version
  - 0.3.0-101116

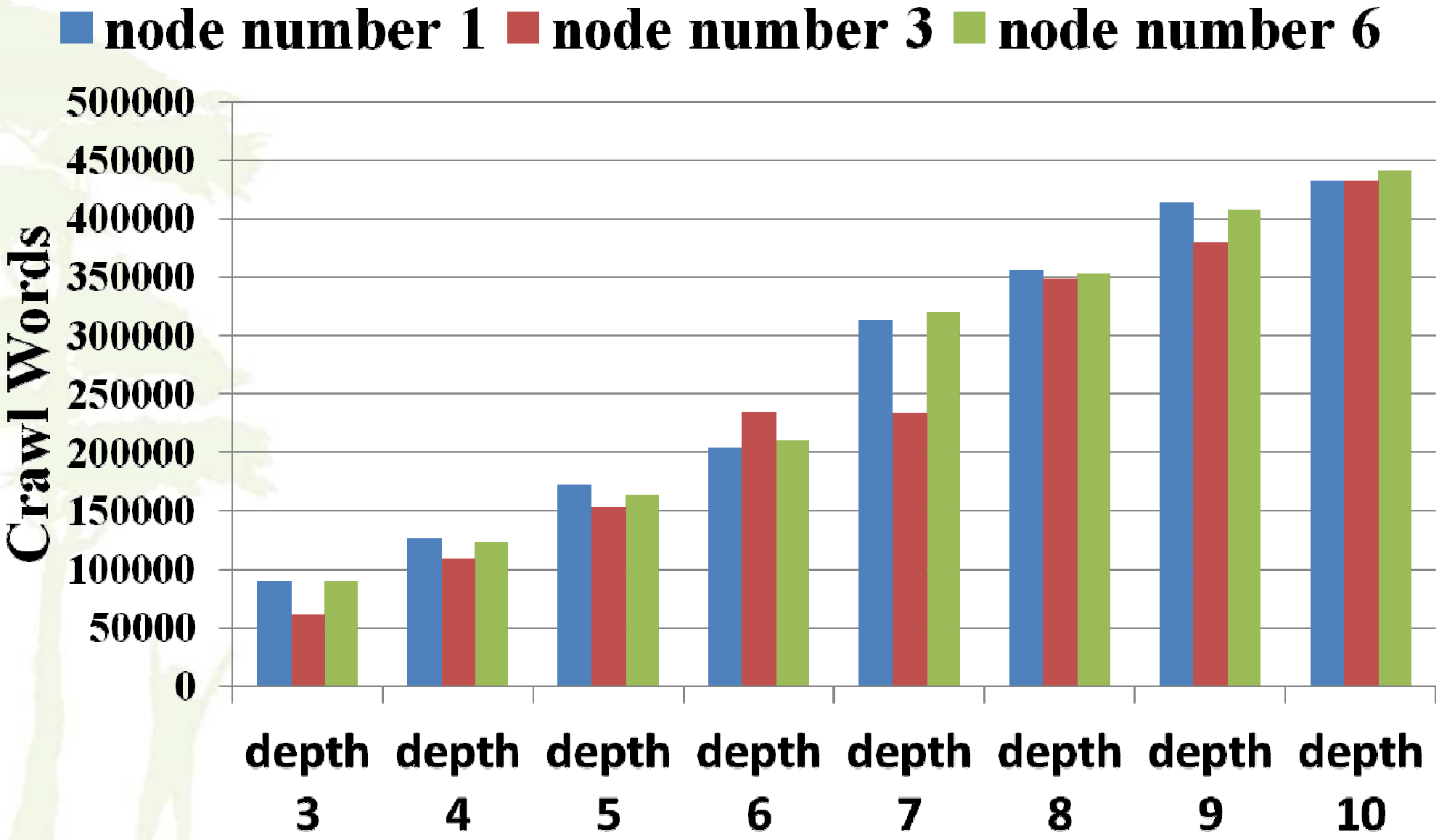
Execute Time



Crawl Files



Crawl Words



Outline

Crawlzilla
- Feature
- System Implement
- Demo

Background and Motivation
- Nutch
- Hadoop
- Search Engine Library

Performance

Introduction

Future Works

Future Works

- **New Version**
  - Support Multi User
  - Support Schedule
  - Update the Kernel
  - More Easily to deploy Slave Computing Nodes
  - Now is testing!
  - Release Day See http://crawlzilla.info

Reference

- J. Dean and S. Ghemawat, MapReduce: Simplified Data Processing on Large Clusters, In Proceedings of the 6th Conference on Symposium on Opearting Systems Design & Implementation - Volume 6, San Francisco, CA, December 06 - 08, 2004.
- S. Ghemawat, H. Gobioff and S. T. Leung, The Google File System, 19th ACM Symposium on Operating Systems Principles, Lake George, NY, October, 2003.
- The Apache Software Foundation, Nutch, available at: http://nutch.apache.org/ , accessed 5 June 2010.
- The Apache Software Foundation, Hadoop, available at: http://hadoop.apache.org/ , accessed 5 June 2010.
- The Apache Software Foundation, Lucene, available at: http://lucene.apache.org/ , accessed 5 June 2010.
- Crawlzilla @ Google Code Project Hosting, available at: http://code.google.com/p/crawlzilla/, accessed 15 Sep 2010.

Enjoy your search engines!!!
Start from Here!

- Crawlzilla @ Google Code Project Hosting (Tutorials in Chinese)
  - http://code.google.com/p/crawlzilla/
- Crawlzilla @ Source Forge (Tutorials in English)
  - http://sourceforge.net/p/crawlzilla/home/
- Crawlzilla User Group @ Google
  - http://groups.google.com/group/crawlzilla-user
- NCHC Cloud Computing Research Group
  - http://trac.nchc.org.tw/cloud

Thank You!
Q & A

運用自由軟體打造資安雲端分析平台

Building Network Security Cloud Analysis Platfrom using Open Source

Yao-Tsung Wang
jazz@nchc.org.tw

Wei-Yu Chen
waue@nchc.org.tw

專家說：雲端每個環節都有安全問題

ZDNet Taiwan - 專家談雲端：每個環節都有安全問題 - 新聞

2010/08/10 19:50:02

專家談雲端：每個環節都有安全問題

ZDNet記者曠文溱／台北報導 雲端的安全問題不是無解，只是不管是雲端服務供應商或者想要建立私有雲的企業用戶，都必須考量到每個環節。

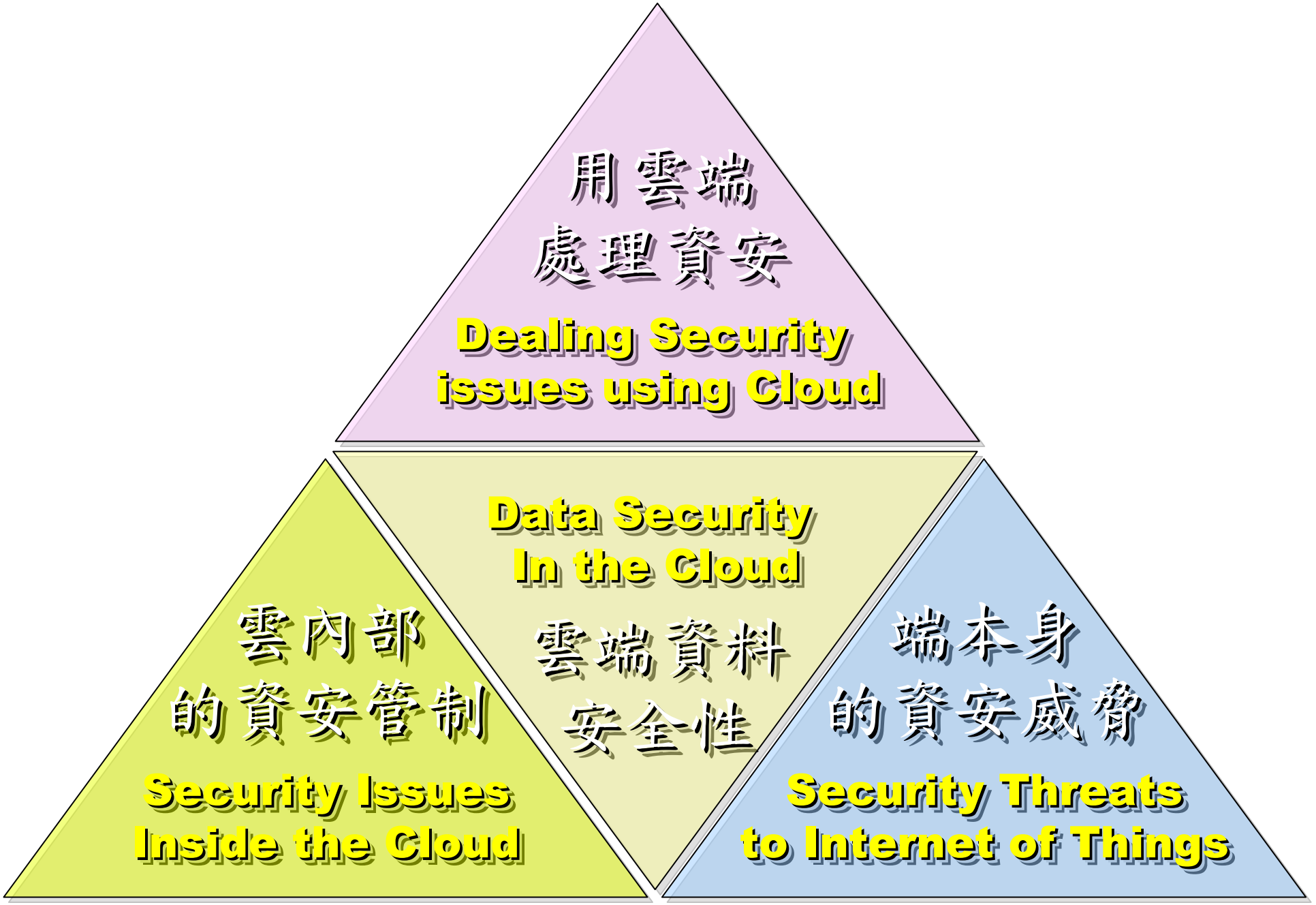
微軟亞太區全球技術支援中心專案經理、同時也是ZDNet專欄作家林宏嘉今（10）日在ZDNeT舉行的IT Priorities圓桌論壇中表示，雲端的安全議題涉及了IaaS、PaaS乃至於SaaS的每個層面。當然有些問題是原本就存在：例如在討論到IaaS時，就涉及到了機房的管理和硬體設備的可用性等；但是講到PaaS時，企業用戶倘若要選擇開原碼的作業系統，必須考量到後續的安全維護；在SaaS的層次，企業用戶必須確保每一個分區（partition）的安全更新和資料安全。

目前正如火如荼建立台灣第一個校園私有雲的台大計算機及資訊網路中心主任孫雅麗則呼應道，Amazon的雲端服務證實了在Hypervisor層有駭客入侵，也就是意味著過去大家在討論如何防範虛擬機器的資料安全，但是威脅已經深化到了更下一層。這些問題都有待解決。

「有些問題甚至是來自於內部，舉例而言，MIS可能會把存在記憶體裡的資料倒出來，或者在Hypervisor層就植入了可以蒐集資料的程式，」孫雅麗說。

安全議題是目前台灣企業對雲端持保留態度的最大主因，這也是何以台灣的大型企業對於雲端的想法，還是偏好建立私有雲，畢竟對用戶而言，要把資料放在別人家，還和其他企業一起共用一家，甚至是競爭對手「共處一室」，

雲端資安的範疇


用雲端
處理資安
Dealing Security
issues using Cloud
Data Security
In the Cloud
雲端資料
安全性
雲內部
的資安管制
Security Issues
Inside the Cloud
端本身
的資安威脅
Security Threats
to Internet of Things

兩大研究方向：你該選「雲」還是「端」？

集中，大廠

Centerized，

Enterprise

端

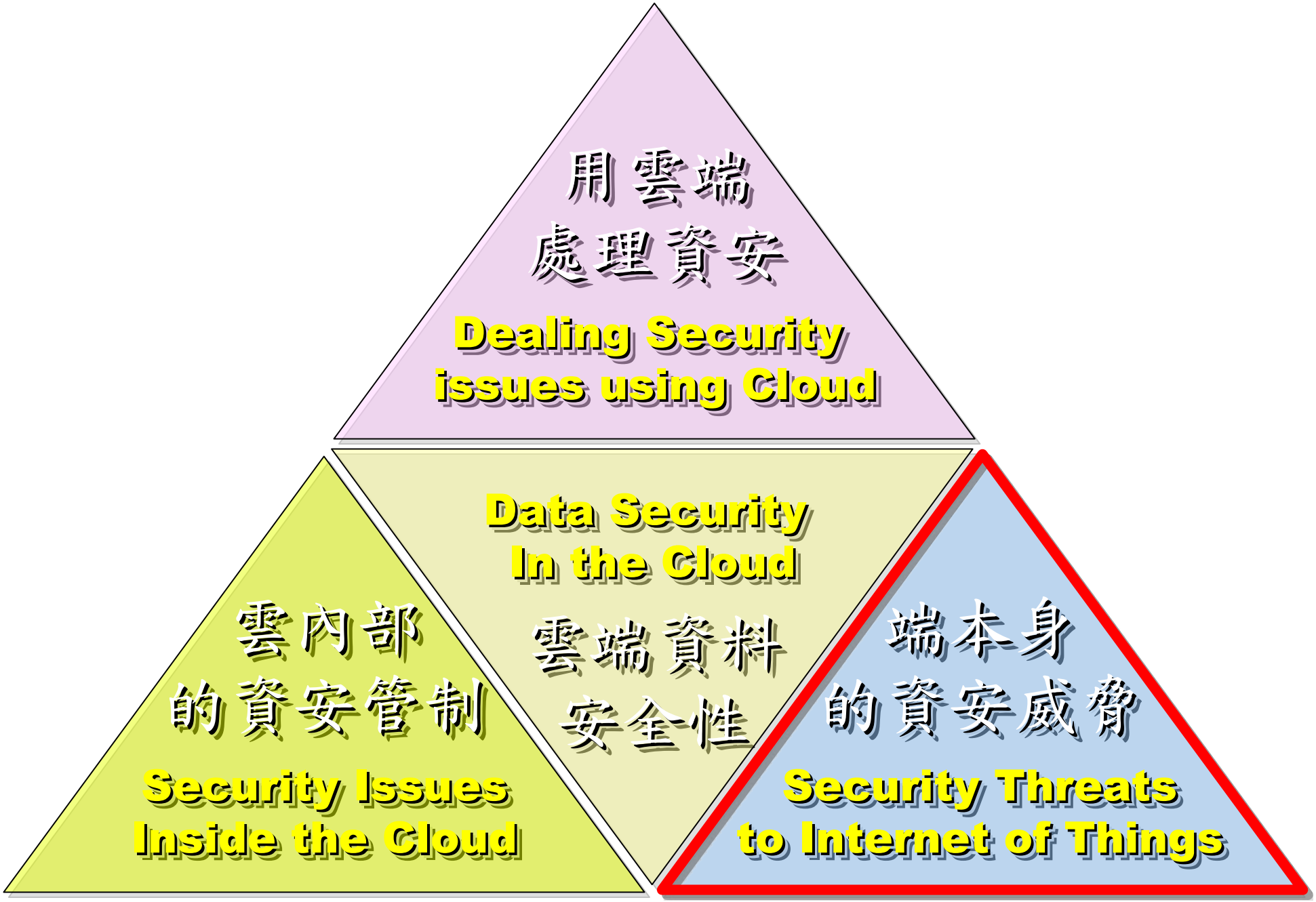
symbian
OS

多元，中小廠

Diversify，

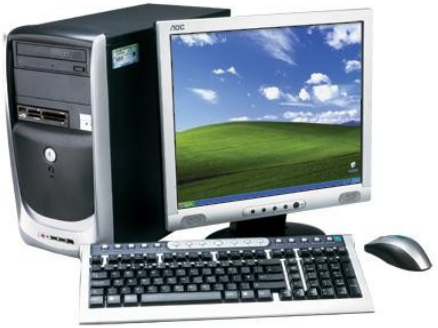
SMB

先來談談「端的安全」


用雲端
處理資安
Dealing Security
issues using Cloud
Data Security
In the Cloud
雲端資料
安全性
雲內部
的資安管制
Security Issues
Inside the Cloud
端本身
的資安威脅
Security Threats
to Internet of Things

以前你只有電腦需要防毒，現在.....

MOD 中華電信



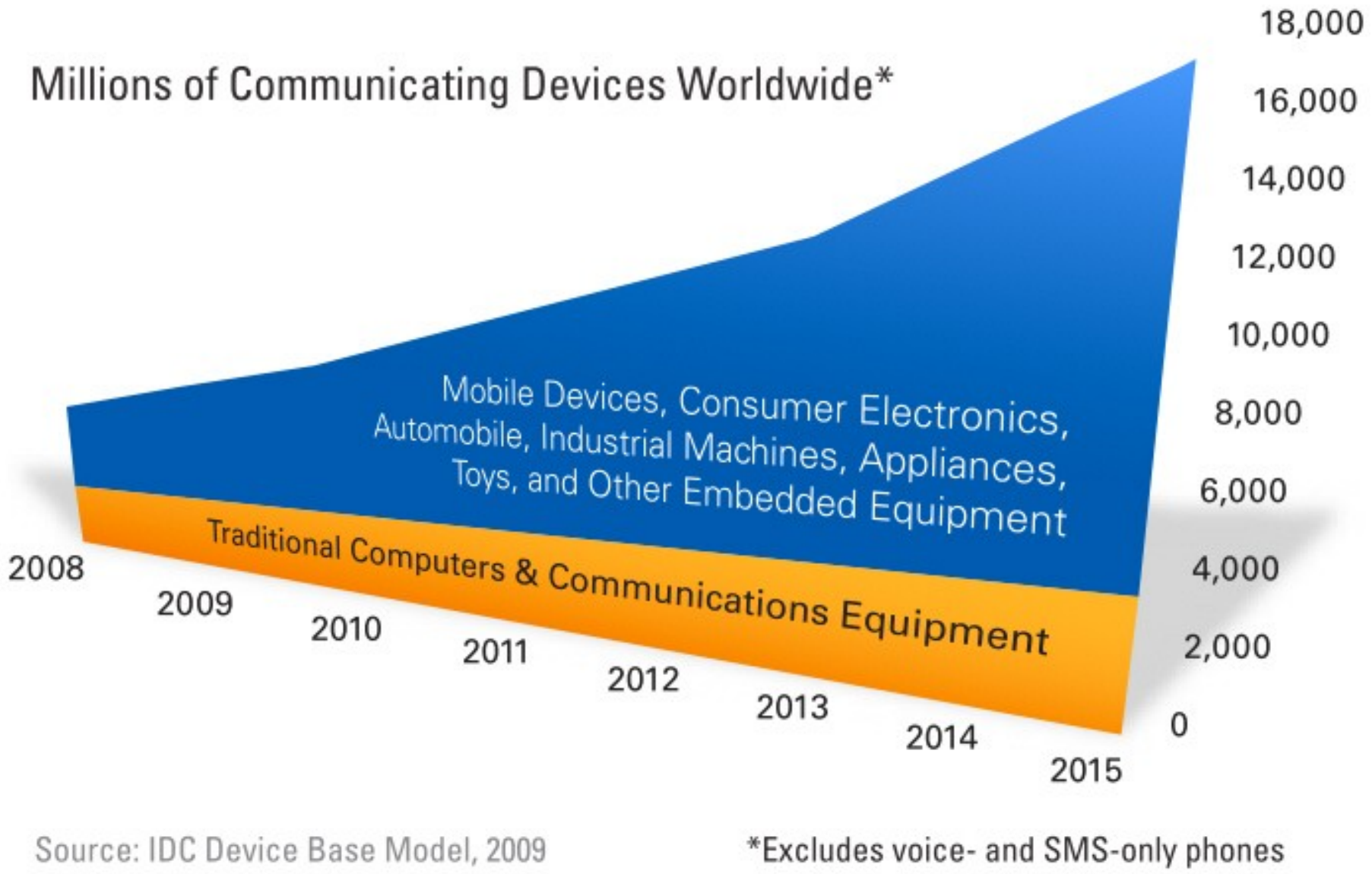
端

symbian OS

多元，中小廠

Diversify，SMB

全球連網裝置急速成長中



圖片來源： Attacks on Mobile and Embedded Systems: Current Trends by Mocana

物聯網的時代來臨

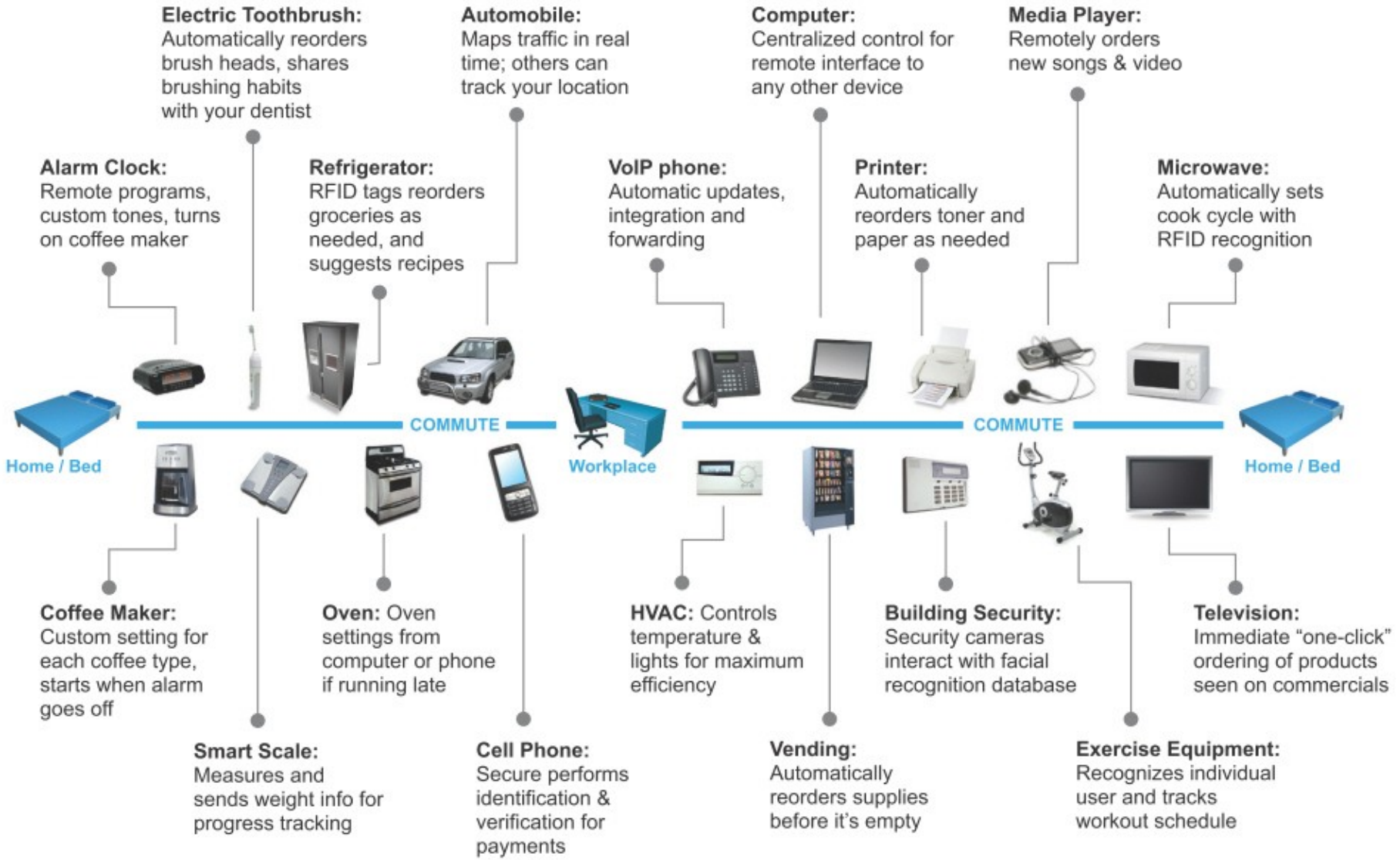
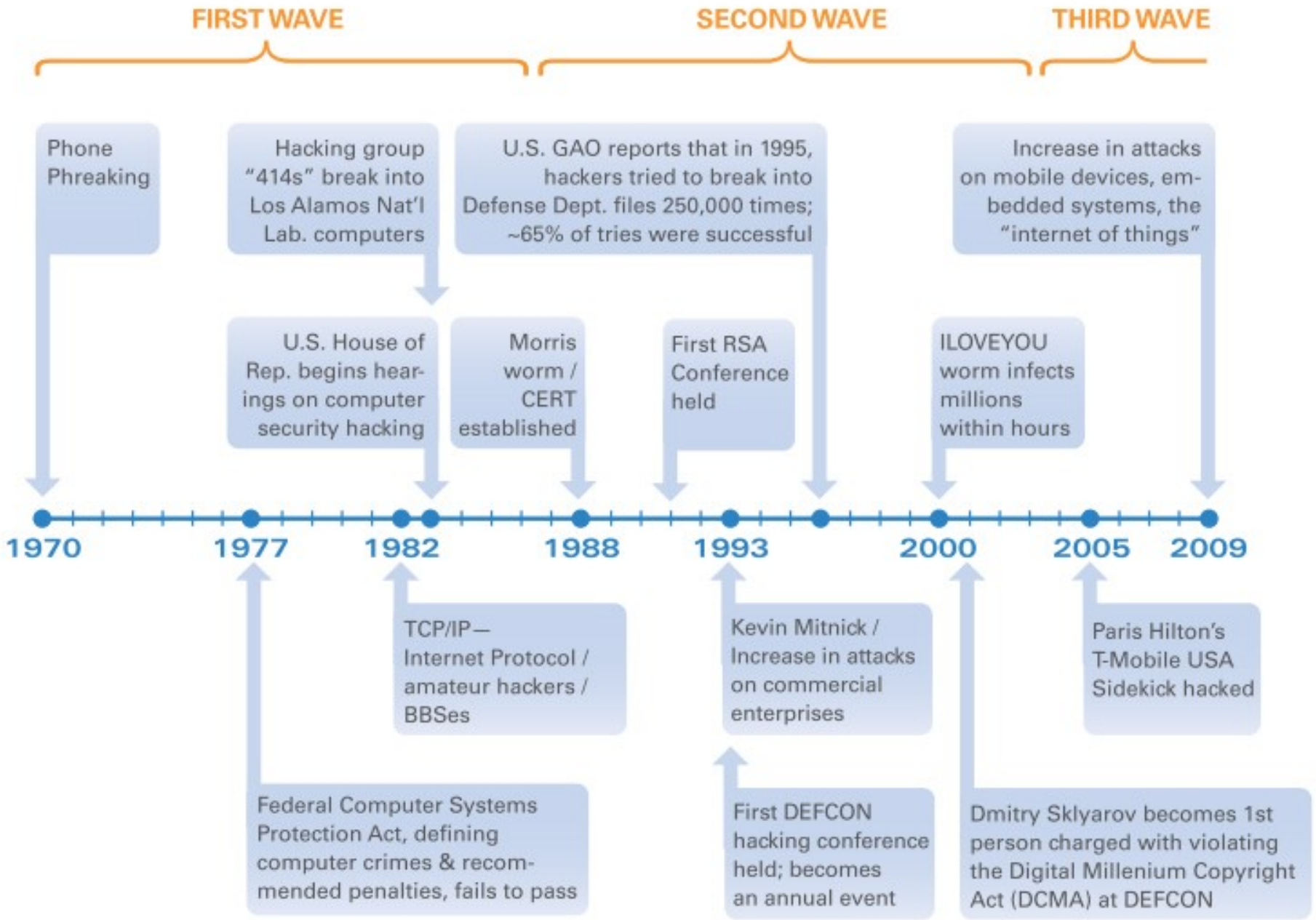


Figure 3. The Internet of Things

圖片來源： Attacks on Mobile and Embedded Systems: Current Trends by Mocana

第三波網路入侵對象將鎖定在『物聯網』



圖片來源： Attacks on Mobile and Embedded Systems: Current Trends by Mocana

針對行動裝置的各種資安問題與經驗

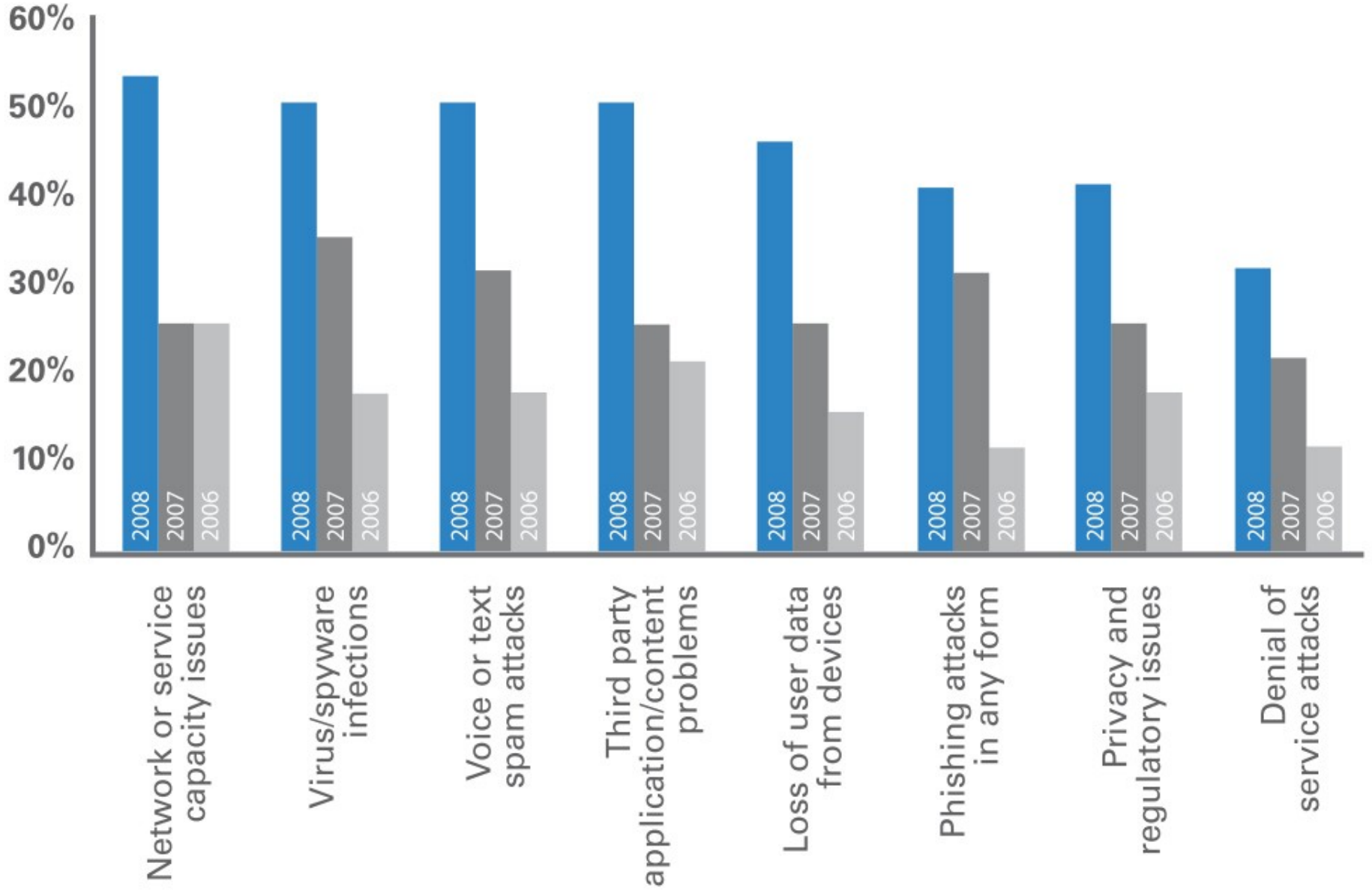
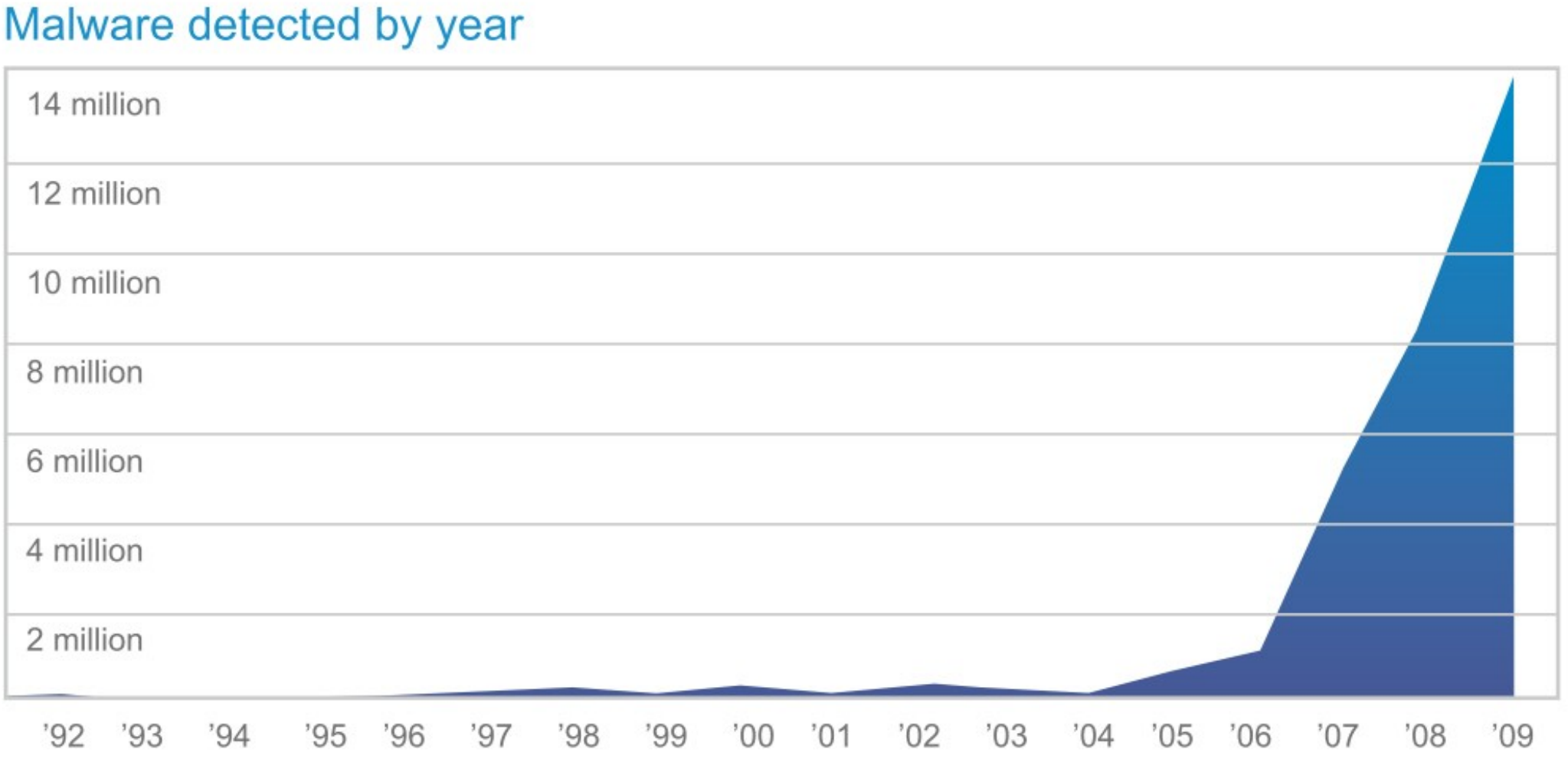


Figure 6. The increase in security issues experienced by mobile device users from 2006 to 2008; % of respondents. McAfee *Mobile Security Report 2009*

圖片來源： Attacks on Mobile and Embedded Systems: Current Trends by Mocana

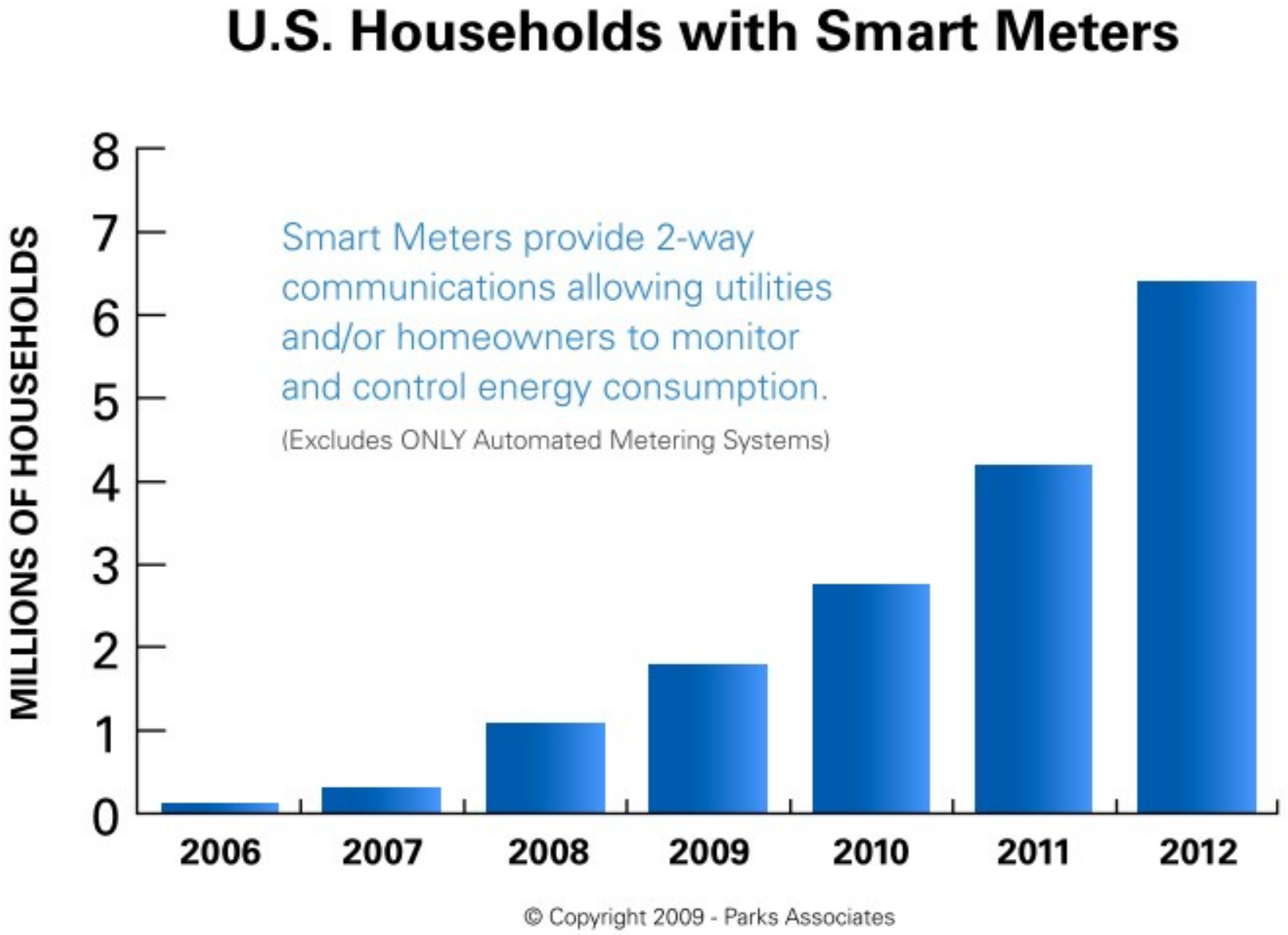
網路惡意程式 (Malware) 逐年激增



Over 3,000 new "species" of PC malware are released onto the Internet every hour. Now that malware is setting its sights on Device platforms.
Source: AV LABS

圖片來源：
U.S. Unprepared for Internet Device Flood: Unaddressed Security Problems & Talent Drought Threaten Long-Term Commercial, Government Interests
By: Kurt Stammberger, CISSP, Adrian Turner and Mat Small, Mocana With: Rich Nass, Sarah Friar, Goldman Sachs

如果你家的智慧電錶被入侵會怎樣？

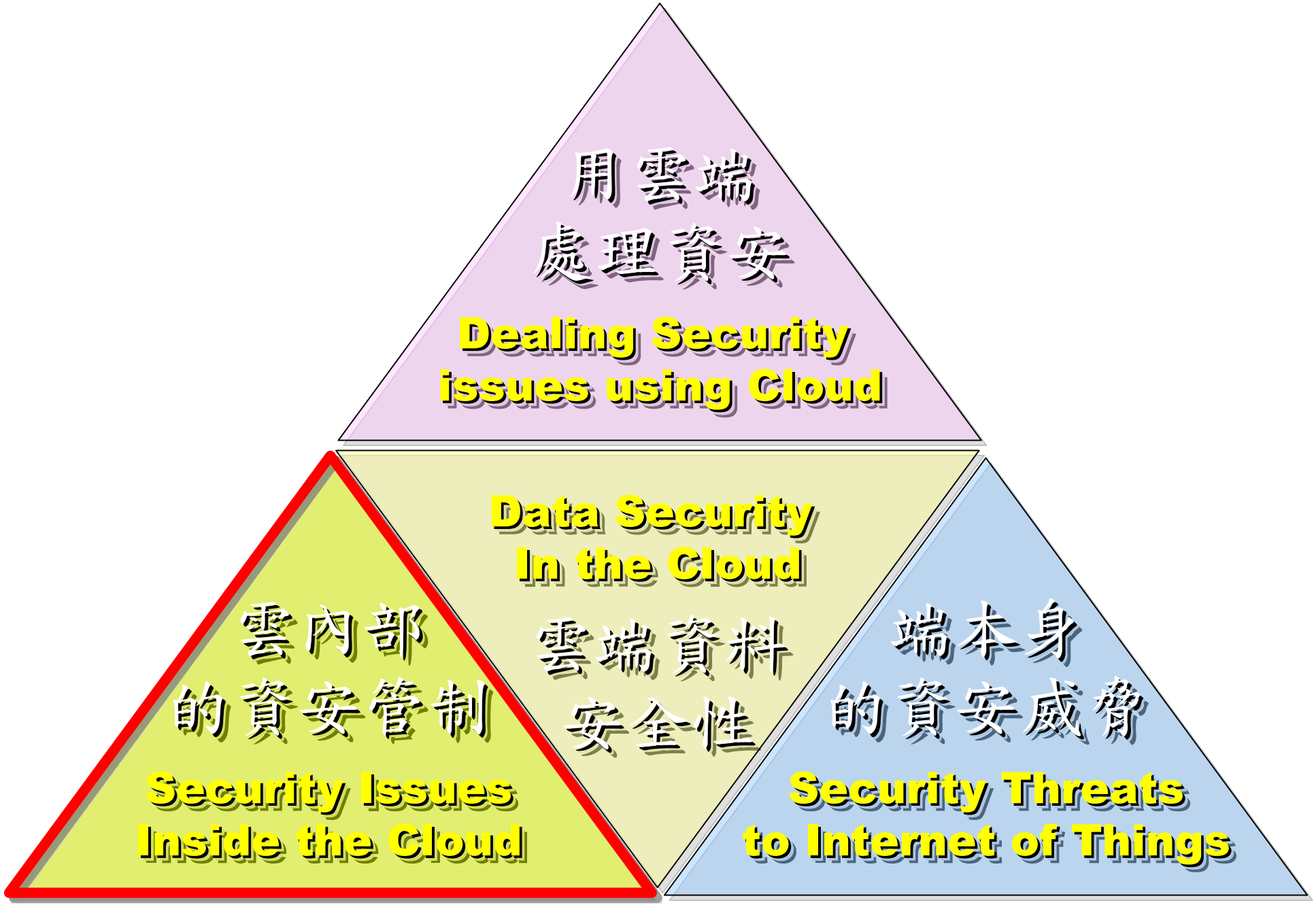


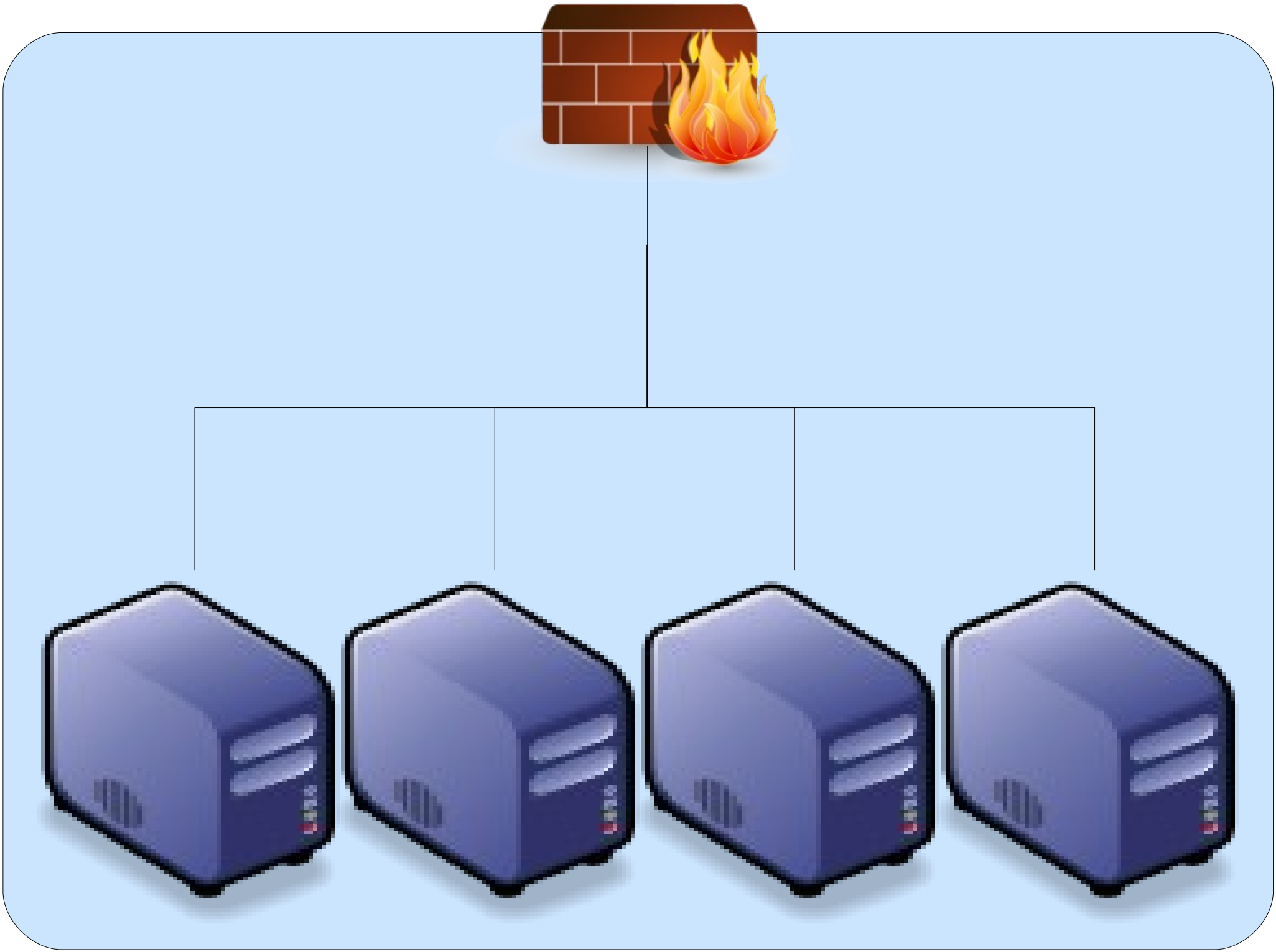
圖片來源：

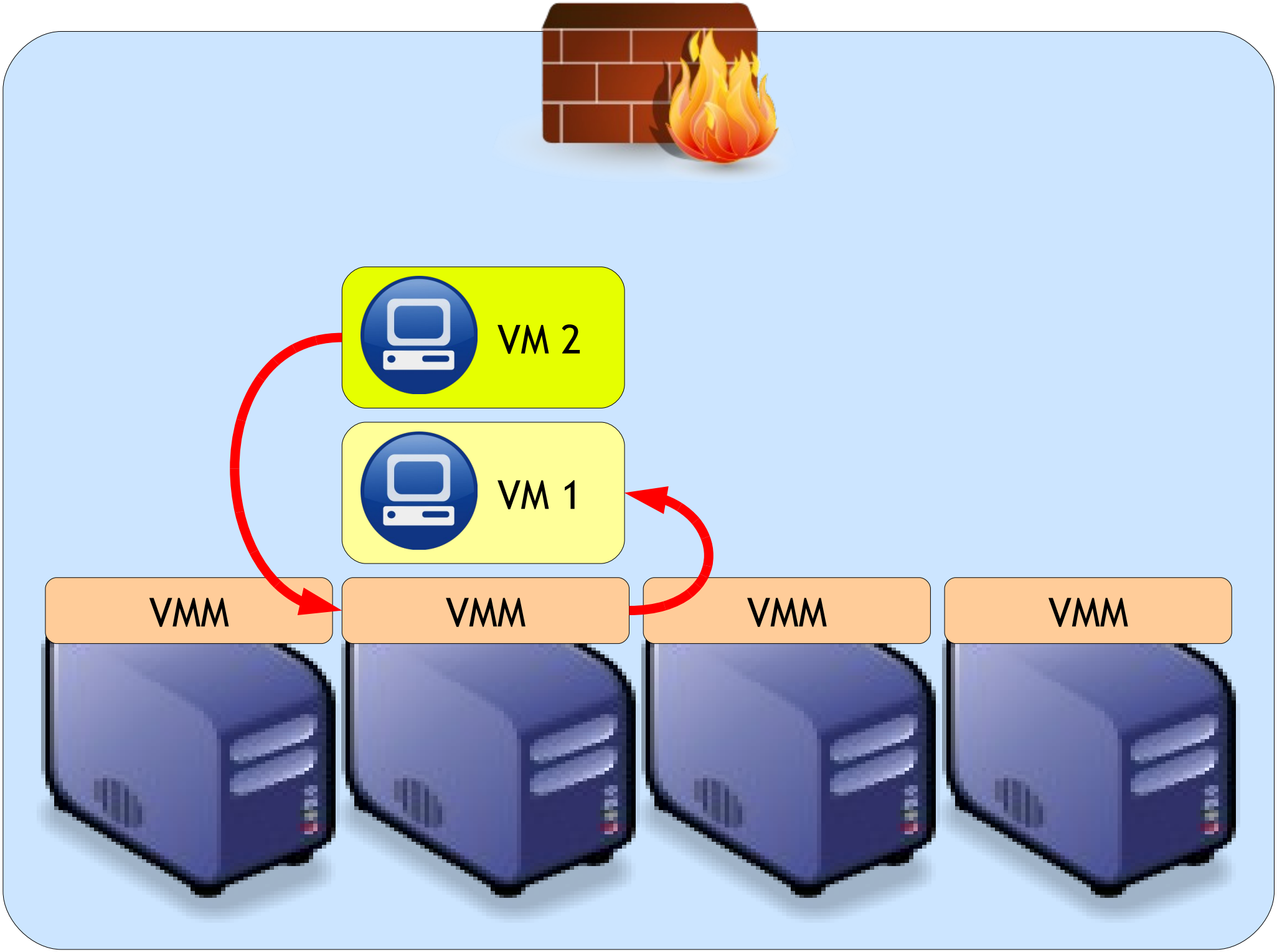
U.S. Unprepared for Internet Device Flood: Unaddressed Security Problems & Talent Drought Threaten Long-Term Commercial, Government Interests

By: Kurt Stammberger, CISSP, Adrian Turner and Mat Small, Mocana With: Rich Nass, Sarah Friar, Goldman Sachs

再來談談「雲的安全」

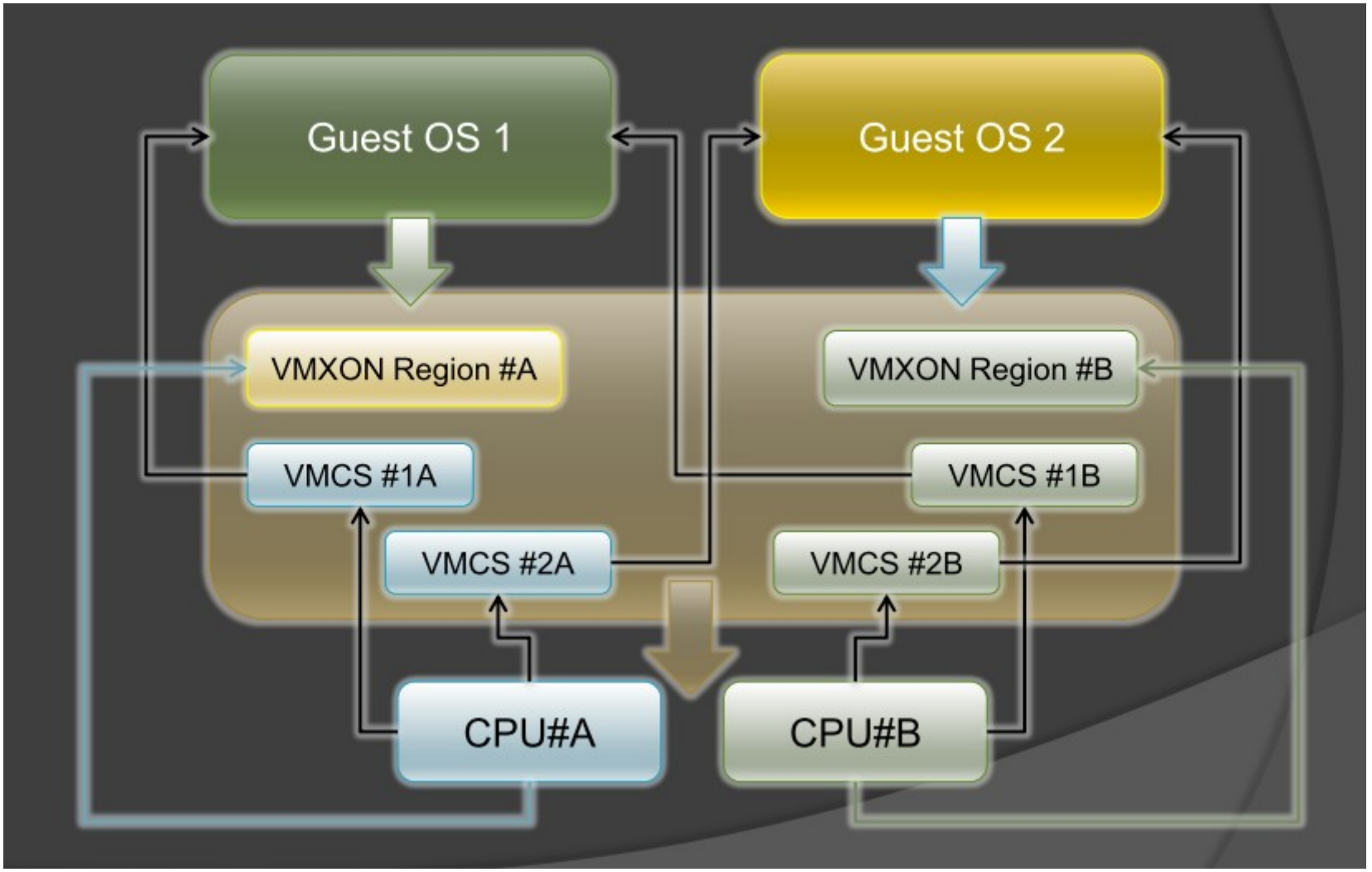

用雲端
處理資安
Dealing Security
issues using Cloud
Data Security
In the Cloud
雲端資料
安全性
雲內部
的資安管制
Security Issues
Inside the Cloud
端本身
的資安威脅
Security Threats
to Internet of Things



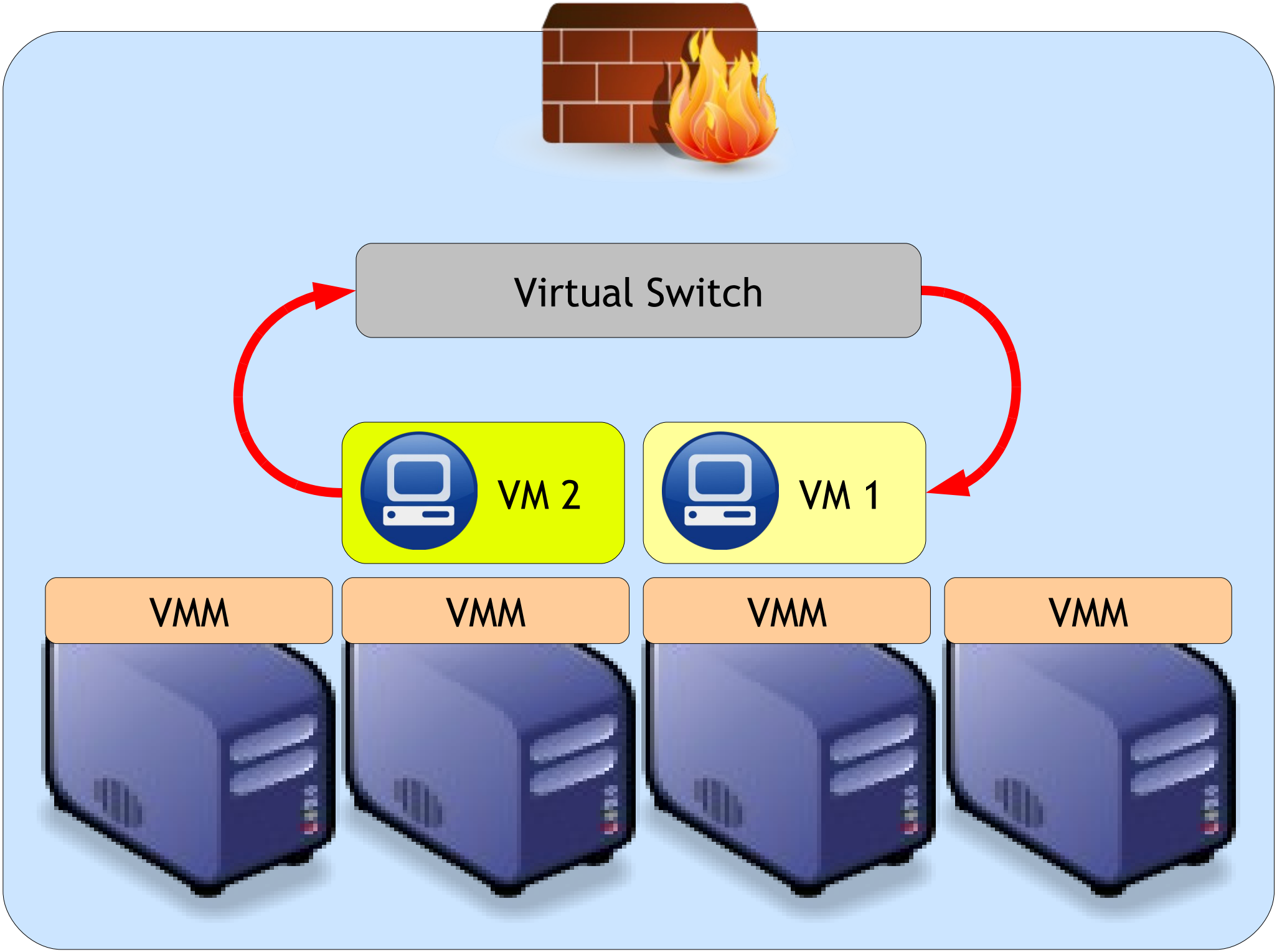

VM 2
VM 1
VMM
VMM
VMM
VMM

虛擬化衍生的新興資安問題

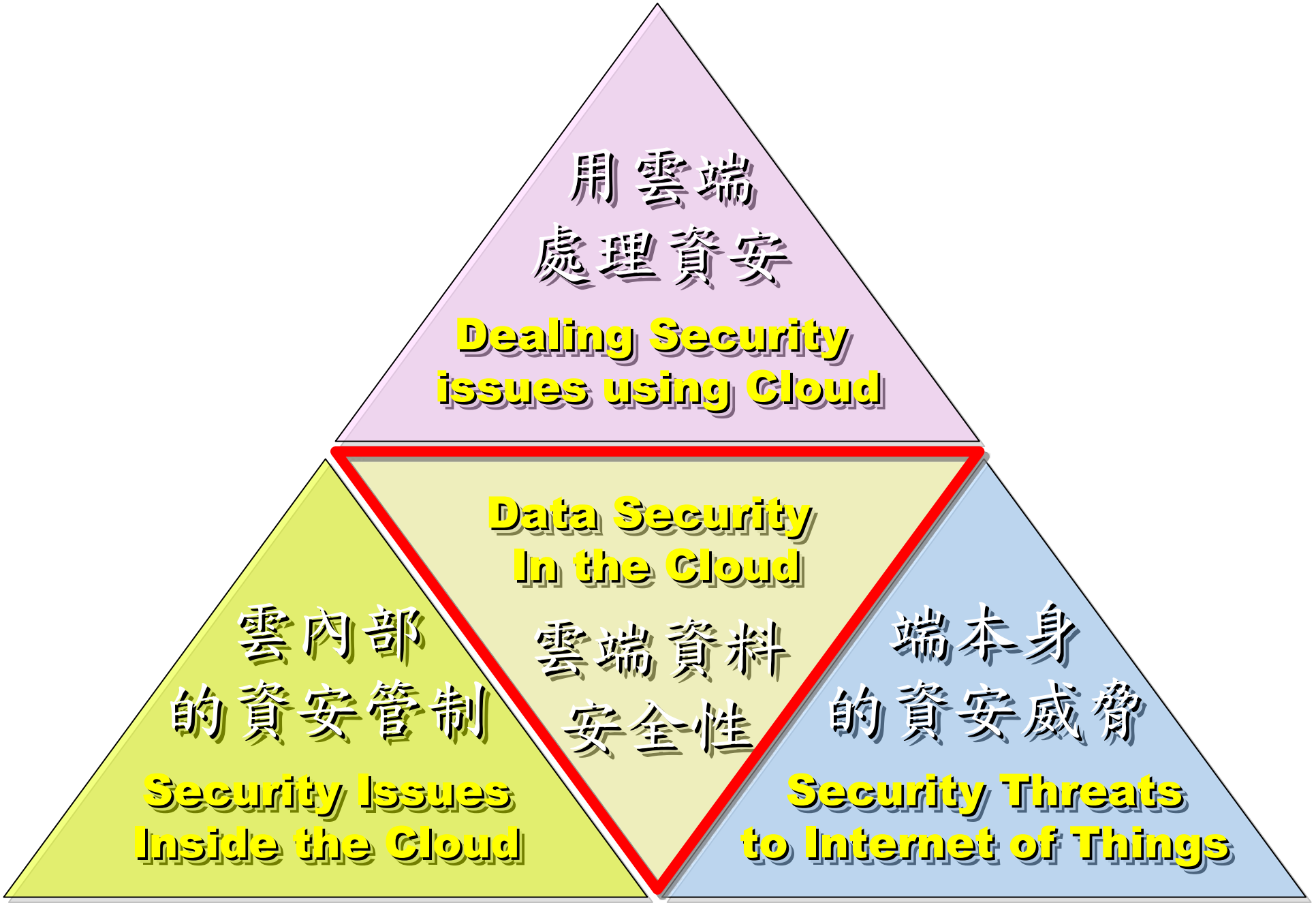
透過虛擬機器，竊取鍵盤輸入、植入後門.....



圖片來源： Hacks in Taiwan Conference 2010
http://www.hitcon.org/hit2010/download/6_New%20Battlefield%20For%20Malware%20Game.pdf


Virtual Switch
VM 2
VM 1
VMM
VMM
VMM
VMM

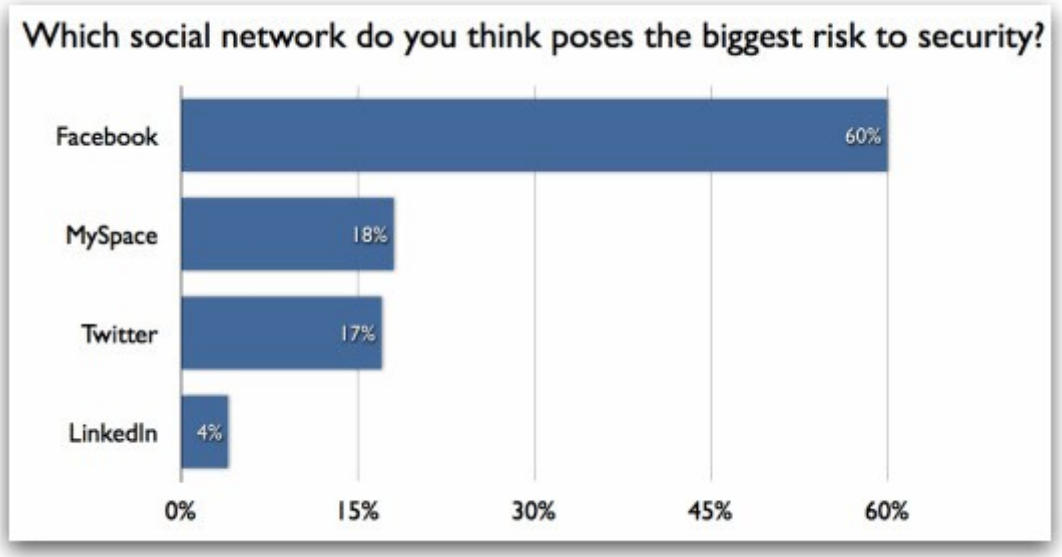
三談「資料安全」


用雲端
處理資安
Dealing Security
issues using Cloud
Data Security
In the Cloud
雲端資料
安全性
雲內部
的資安管制
Security Issues
Inside the Cloud
端本身
的資安威脅
Security Threats
to Internet of Things

Ex. 無名照片外流、臉書個資外洩

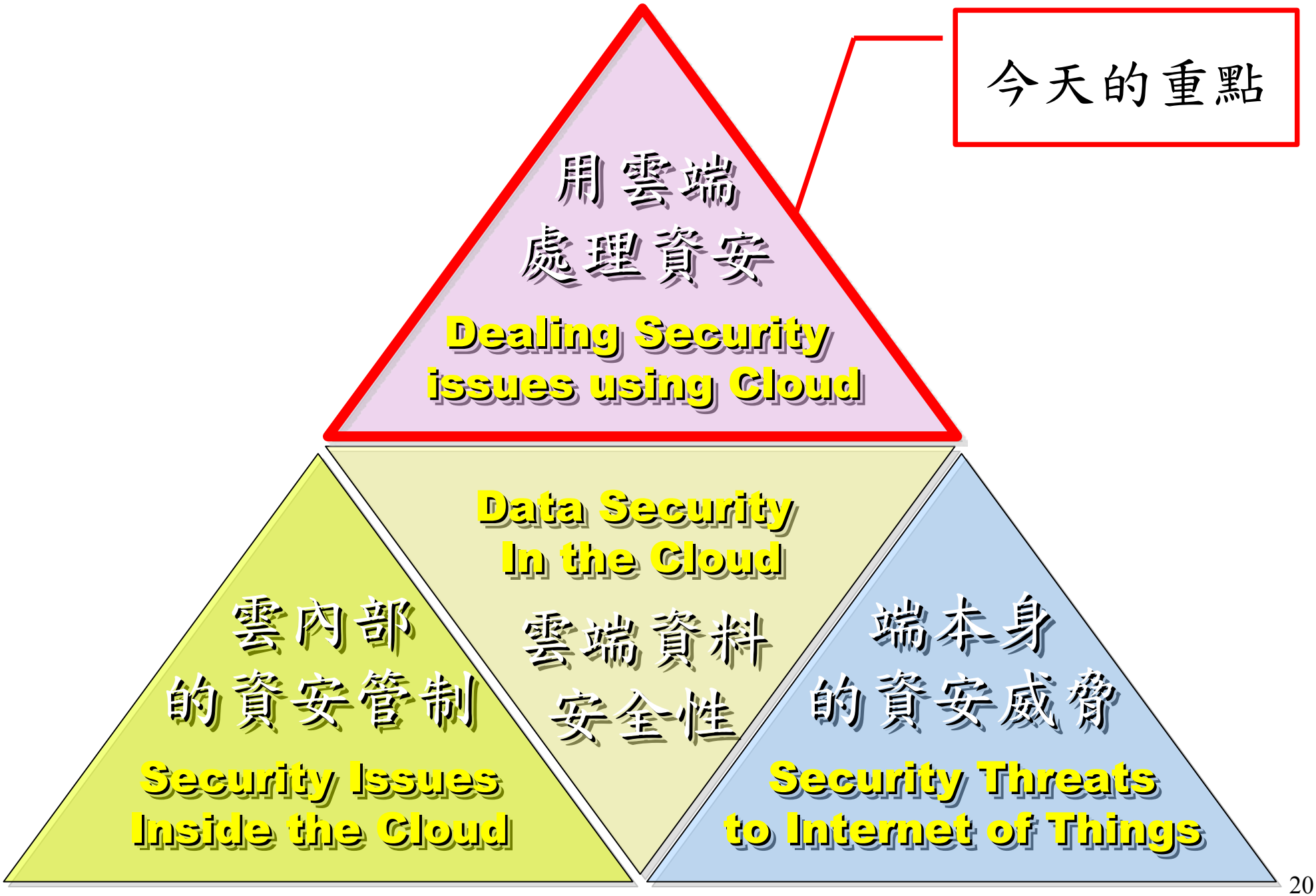
轟動一時黑澀會妹妹容瑄親密自拍照片外流

圖片來源：
Wikileaks and Facebook Privacy / Security: Do we care?

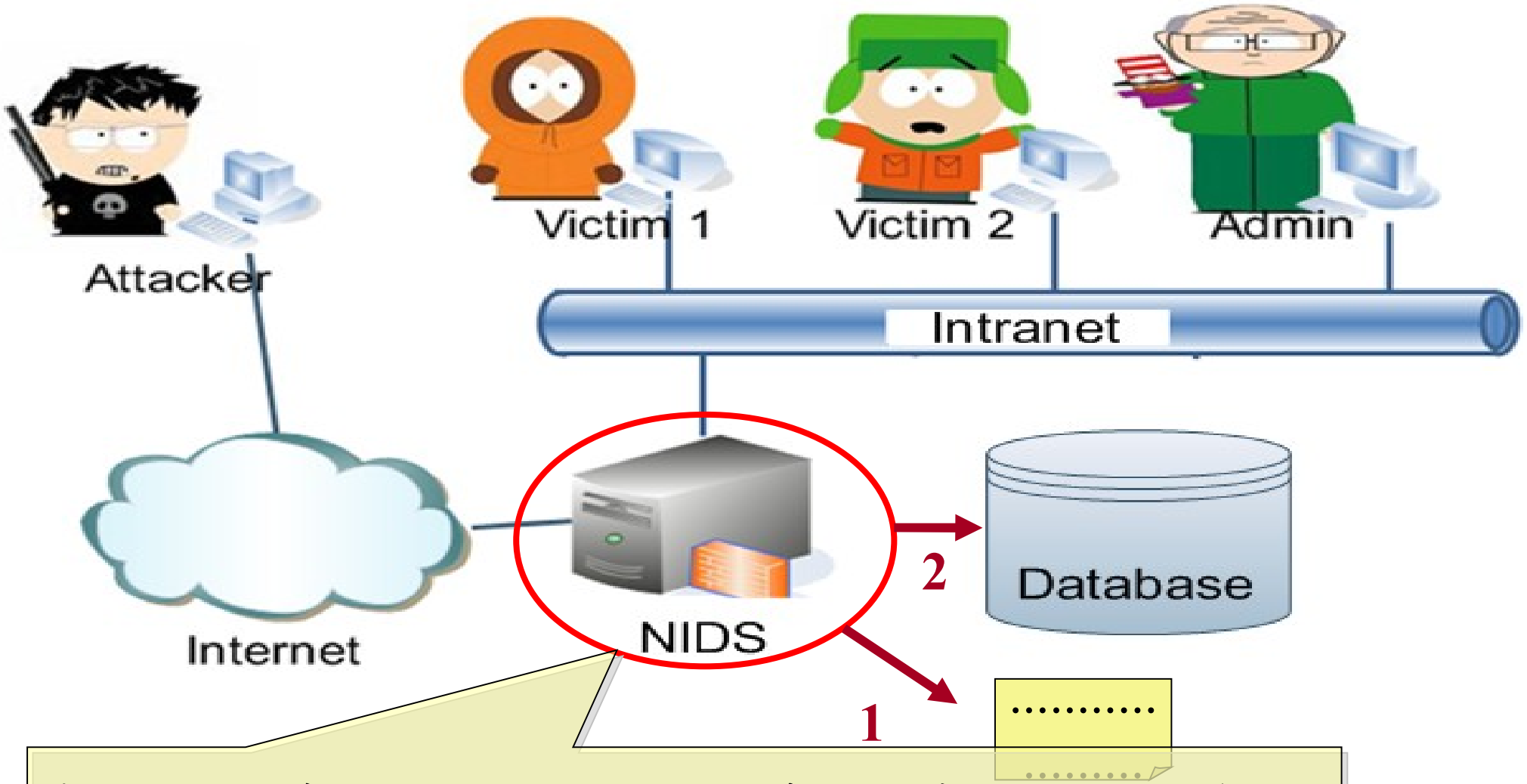


圖片來源：
Report Ranks Facebook As Greatest Corporate Security Risk
http://www.allfacebook.com/facebook-corporate-risk-2010-02

進入今天的主題：用雲端處理傳統資安問題



使用入侵偵測系統 (NIDS) 來找出入侵訊息



當入侵偵測系統偵測到網路上有異常封包時，就會產生警訊以告知有攻擊發生。警訊通常有兩種形式：

1. 紀錄成 log 檔　2. 紀錄到資料庫

傳統 NIDS 的警訊型態 (1) 紀錄在日誌檔內

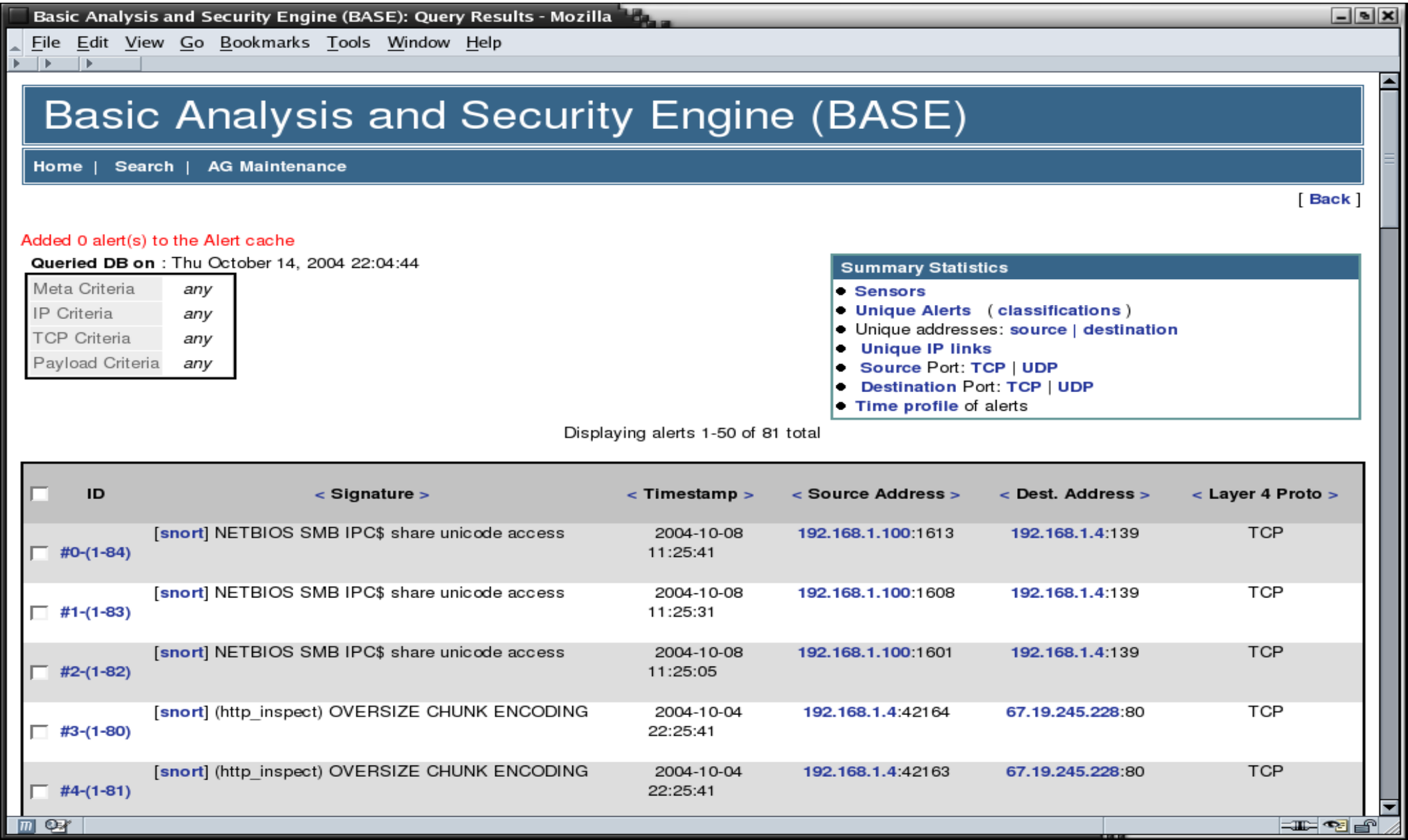
入侵偵測系統所產生警訊日誌檔內一小段內容

```
[**] [1:538:15] NETBIOS SMB IPC$ unicode share access [**]
[Classification: Generic Protocol Command Decode] [Priority: 3]
09/04-17:53:56.363811 168.150.177.165:1051 -> 168.150.177.166:139
TCP TTL:128 TOS:0x0 ID:4000 IpLen:20 DgmLen:138 DF
***AP*** Seq: 0x2E589B8  Ack: 0x642D47F9  Win: 0x4241  TcpLen: 20

[**] [1:1917:6] SCAN UPnP service discover attempt [**]
[Classification: Detection of a Network Scan] [Priority: 3]
09/04-17:53:56.385573 168.150.177.164:1032 -> 239.255.255.250:1900
UDP TTL:1 TOS:0x0 ID:80 IpLen:20 DgmLen:161
Len: 133

[**] [1:1917:6] SCAN UPnP service discover attempt [**]
[Classification: Detection of a Network Scan] [Priority: 3]
09/04-17:53:56.386910 168.150.177.164:1032 -> 239.255.255.250:1900
UDP TTL:1 TOS:0x0 ID:82 IpLen:20 DgmLen:161
Len: 133

[**] [1:1917:6] SCAN UPnP service discover attempt [**]
[Classification: Detection of a Network Scan] [Priority: 3]
09/04-17:53:56.388244 168.150.177.164:1032 -> 239.255.255.250:1900
UDP TTL:1 TOS:0x0 ID:84 IpLen:20 DgmLen:161
Len: 133

[**] [1:538:15] NETBIOS SMB IPC$ unicode share access [**]
[Classification: Generic Protocol Command Decode] [Priority: 3]
09/04-17:53:56.405923 168.150.177.164:1035 -> 168.150.177.166:139
TCP TTL:128 TOS:0x0 ID:94 IpLen:20 DgmLen:138 DF
***AP*** Seq: 0x82073DFF  Ack: 0x2468EB82  Win: 0x4241  TcpLen: 20
```

```
[**] [1:1917:6] SCAN UPnP service discover attempt [**]
[Classification: Detection of a Network Scan] [Priority: 3]
09/04-17:53:56.417045 168.150.177.164:45461 -> 168.150.177.1:1900
UDP TTL:1 TOS:0x0 ID:105 IpLen:20 DgmLen:161
Len: 133

[**] [1:1917:6] SCAN UPnP service discover attempt [**]
[Classification: Detection of a Network Scan] [Priority: 3]
09/04-17:53:56.420759 168.150.177.164:45461 -> 168.150.177.1:1900
UDP TTL:1 TOS:0x0 ID:117 IpLen:20 DgmLen:160
Len: 132

[**] [1:1917:6] SCAN UPnP service discover attempt [**]
[Classification: Detection of a Network Scan] [Priority: 3]
09/04-17:53:56.422095 168.150.177.164:45461 -> 168.150.177.1:1900
UDP TTL:1 TOS:0x0 ID:118 IpLen:20 DgmLen:161
Len: 133

[**] [1:2351:10] NETBIOS DCERPC ISystemActivator path overflow attempt little endian
unicode [**]
[Classification: Attempted Administrator Privilege Gain] [Priority: 1]
09/04-17:53:56.442445 198.8.16.1:10179 -> 168.150.177.164:135
TCP TTL:105 TOS:0x0 ID:49809 IpLen:20 DgmLen:1420 DF
***A**** Seq: 0xF9589BBF  Ack: 0x82CCF5B7  Win: 0xFFFF  TcpLen: 20
[Xref => http://www.microsoft.com/technet/security/bulletin/MS03-026.mspx][Xref =>
http://cgi.nessus.org/plugins/dump.php3?id=11808][Xref => http://cve.mitre.org/cgi-
bin/cvename.cgi?name=2003-0352][Xref => http://www.securityfocus.com/bid/8205]

[**] [122:3:0] (portscan) TCP Portsweep [**]
[Priority: 3]
09/04-17:53:56.499016 198.8.16.1 -> 168.150.177.166
PROTO:255 TTL:0 TOS:0x0 ID:1750 IpLen:20 DgmLen:168
```

傳統 NIDS 的警訊型態 (2) 紀錄在資料庫內

以下為利用瀏覽器透過網頁方式呈現警訊資料庫的內容



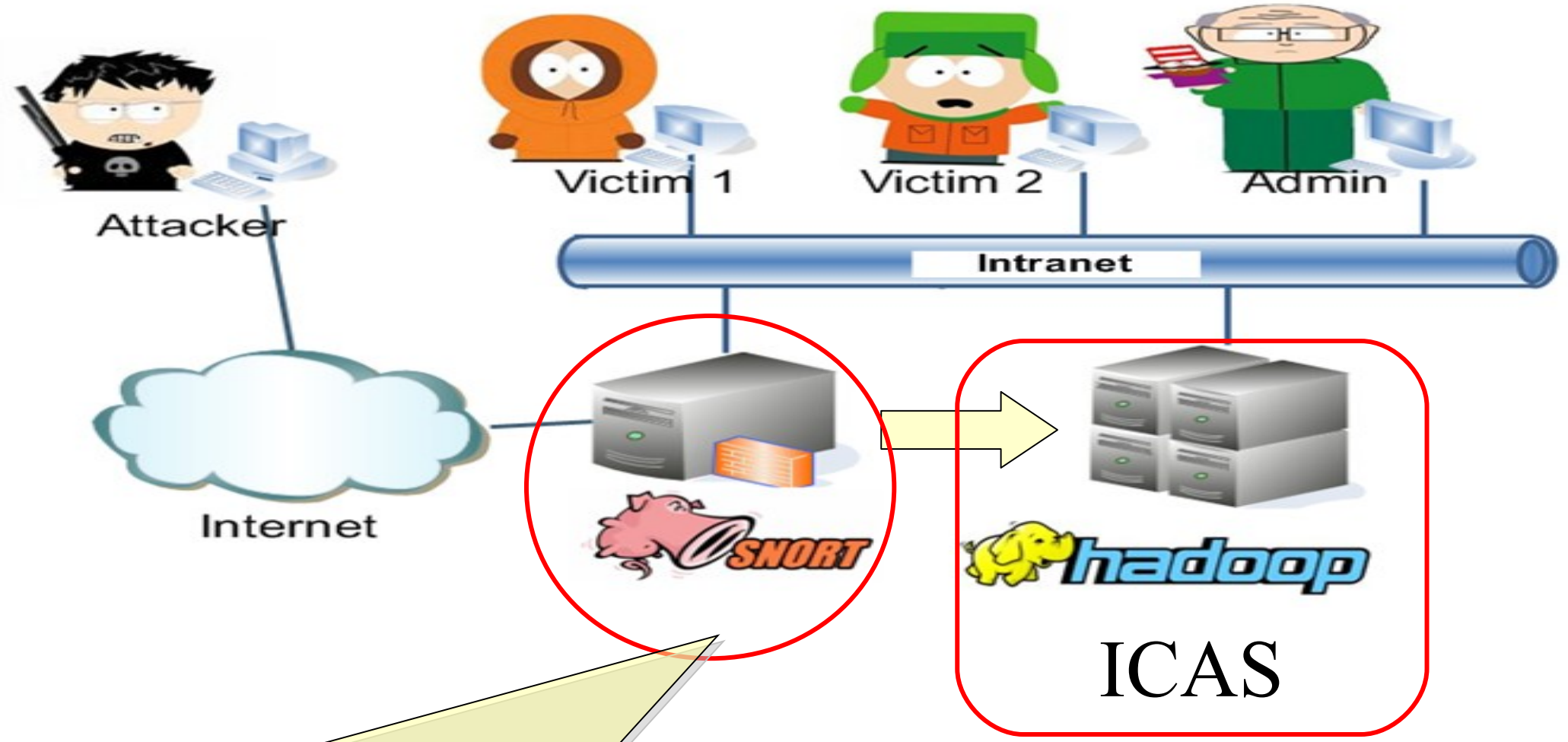
以上作法的缺點

- 警訊僅被『忠實』地被記錄下來，無法顯示彼此間的關聯性，因此系統管理者難以瞭解全部攻擊情形
- 過多的警訊，使得容易忽略重要內容
- 完全依賴單一台資料庫，當資料量一大，該台主機的讀寫效率將成為瓶頸

使用雲端運算的解決方案：ICAS

- ICAS, *IDS Cloud Analysis System*
- 利用雲端運算的特性提供以下好處
  - 對大量資料有高效率
  - 一般主機的叢集
  - 有錯誤容忍
- 分析演算法
  - 整合
  - 關聯

透過 ICAS 協助分析 IDS 的警訊

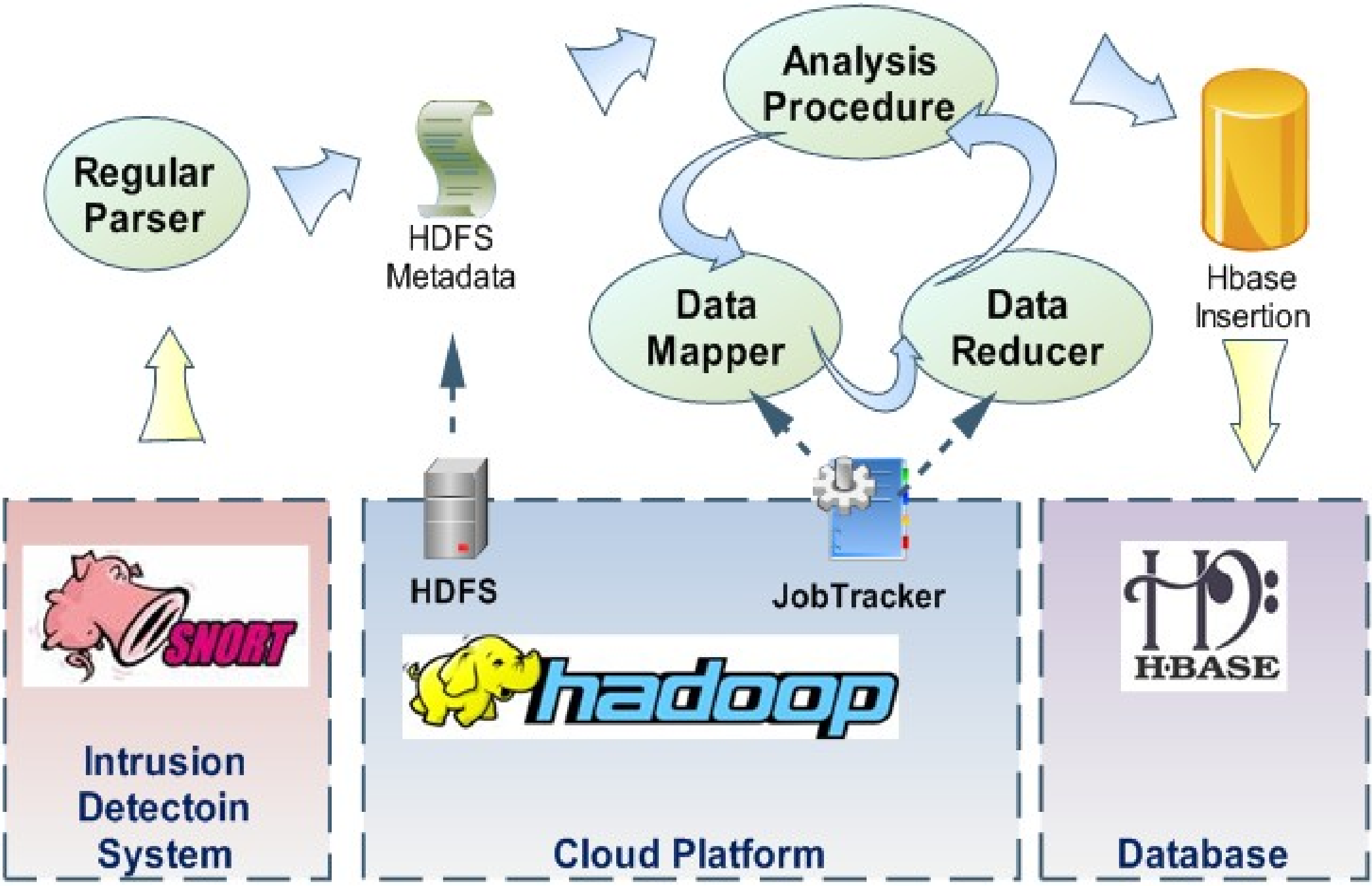


可多個 NIDS 共同產生警訊後，傳送至 ICAS ，分析演算法目前有 ICAS-I 及 ICAS-II

ICAS-I

- 將任意個原始警訊檔上傳到運行 ICAS-I 演算法的 Hadoop 檔案系統空間（ HDFS ）
- 利用 Hadoop 的 MapReduce 平台架構所設計的演算法來分析資料
- 分析完後的資料塞入分散式資料庫 HBase 內

ICAS-I 流程圖


Regular Parser
HDFS Metadata
Analysis Procedure
Data Mapper
Data Reducer
Hbase Insertion
HDFS
JobTracker
SNORT
hadoop
HBASE
Intrusion Detectoin System
Cloud Platform
Database

ICAS-I 整合後的警訊結果

Destination IP	Attack Signature	Source IP	Destination Port	Source Port	Packet Protocol	Timestamp
Host_1	Trojan	Sip1	80	4077	tcp	T1
Host_1	Trojan	Sip2	80	4077	tcp	T2
Host_1	Trojan	Sip1	443	5002	tcp	T3
Host_2	Trojan	Sip1	443	5002	tcp	T4
Host_3	D.D.O.S	Sip3	53	6007	udp	T5
Host_3	D.D.O.S	Sip4	53	6008	tcp	T5
Host_3	D.D.O.S	Sip5	53	6007	udp	T5
Destination IP	Attack Signature	Source IP	Destination Port	Source Port	Packet Protocol	Timestamp

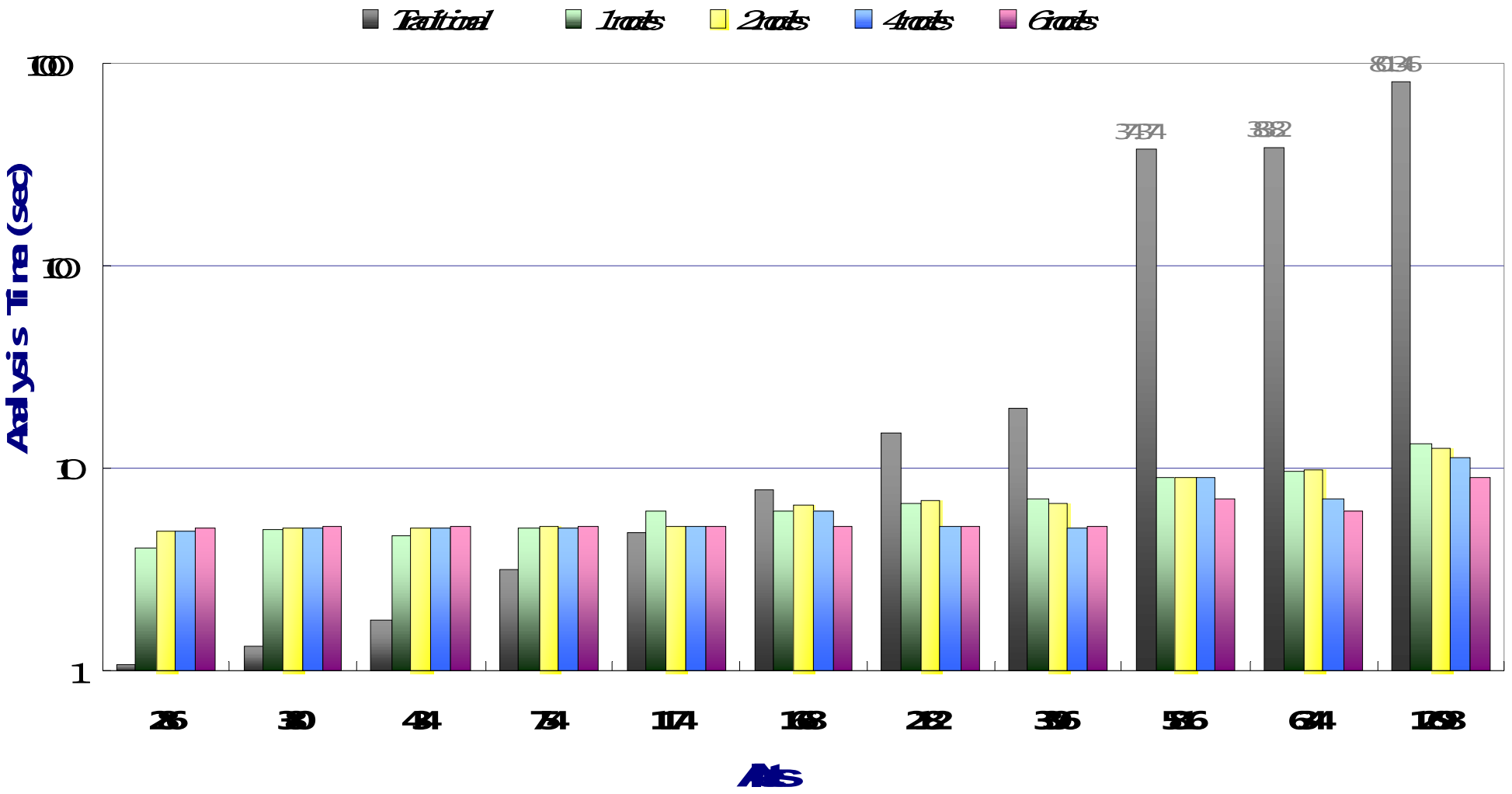
Key		Values				
Host_1	Trojan	Sip1,Sip2	80,443	4077,5002	tcp	T1,T2,T3
Host_2	Trojan	Sip1	443	5002	tcp	T4
Key		Values				

ICAS-I 效能數據的環境

- Machine:
  - CPU : Intel quad-core, Memory : 2 GB,
- OS : Linux : Ubuntu 8.04 server
- Software : version
  - Hadoop : 0.16.4
  - Hbase : 0.1.3
  - Java : 6
- Alerts Data Sets
  - MIT Lincoln Laboratory, Lincoln Lab Data Sets
  - Computer Security group at UCDavis, tcpdump file

ICAS-I 效能分析時間圖

The Consuming Time of Each Number of Data Sets



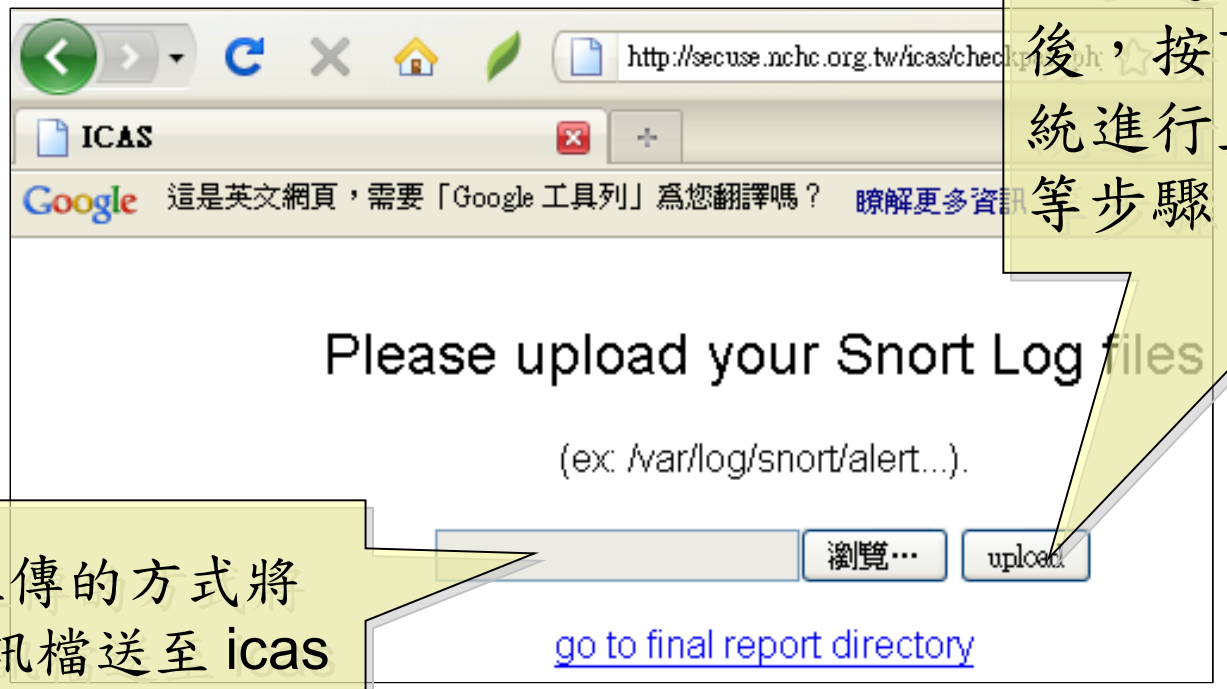
ICAS-I 效能數據表

Throughput Data Overall

Origianl Alerts	Analysis Time (sec)					Results	Reduction Rate
	Traditional	1 nodes	2 nodes	4 nodes	6 nodes		
286	1.068	4.087	4.869	4.864	5.077	30	89.51%
380	1.333	4.94	5.069	5.067	5.097	11	97.11%
434	1.76	4.61	5.066	5.068	5.09	9	97.93%
754	3.145	5.066	5.079	5.038	5.096	16	97.88%
1174	4.73	6.066	5.093	5.089	5.097	33	97.19%
1668	7.909	6.07	6.56	6.071	5.082	16	99.04%
2182	14.949	6.671	6.95	5.166	5.088	16	99.27%
3396	19.901	7.053	6.654	5.076	5.091	68	98.00%
5816	374.374	9.081	9.076	9.07	7.076	66	98.87%
6344	383.82	9.68	9.872	7.069	6.069	72	98.87%
12698	801.346	13.096	12.367	11.367	9.083	36	99.72%

ICAS-II

- ICAS-I 僅將資料塞入資料庫，然而還是文字的敘述
- ICAS-II 將輸入的任意多個警訊整合成一張警訊關聯圖
- 資料的來源可以透過以下兩種方式上傳到分析平台
  - 系統自動設定以 SCP 傳送到 ICAS 工作目錄
  - 管理者透過 ICAS 網頁上傳

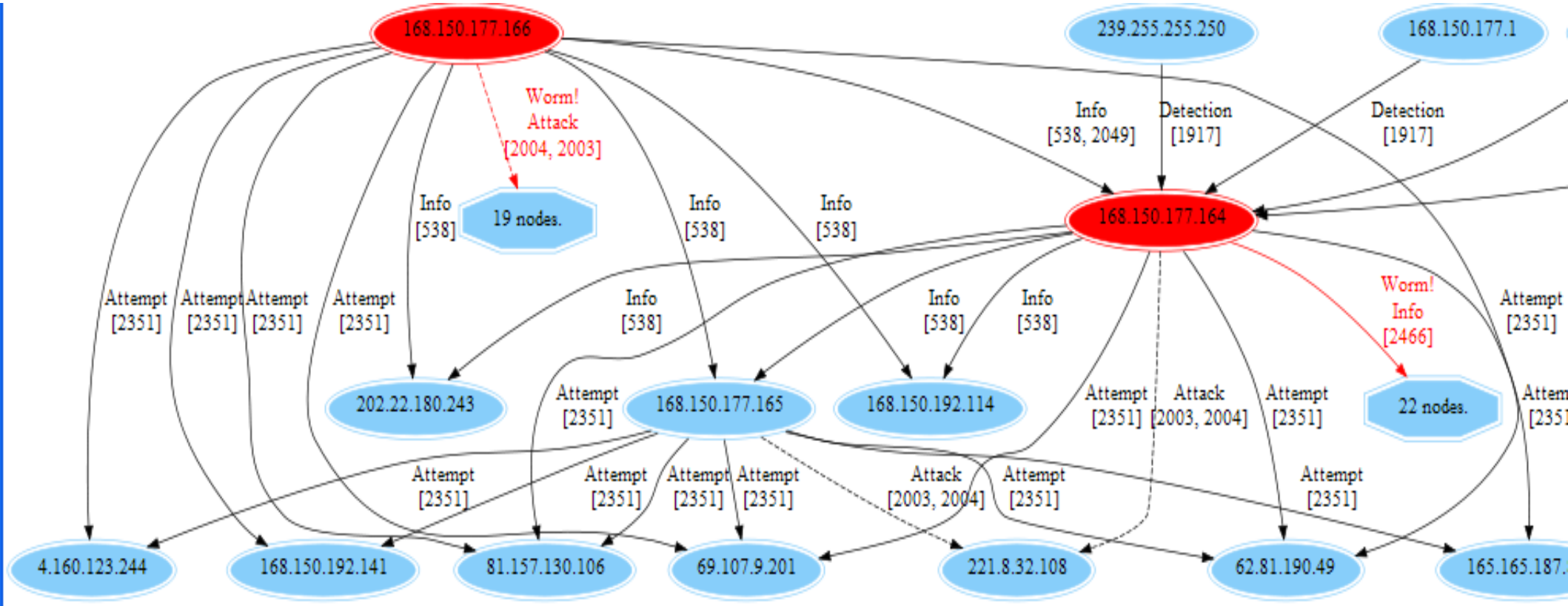


一旦選定需分析的日誌檔後，按下 upload 鈕，系統進行上傳→分析→繪圖等步驟

透過網頁上傳的方式將 snort 的警訊檔送至 icas 分析

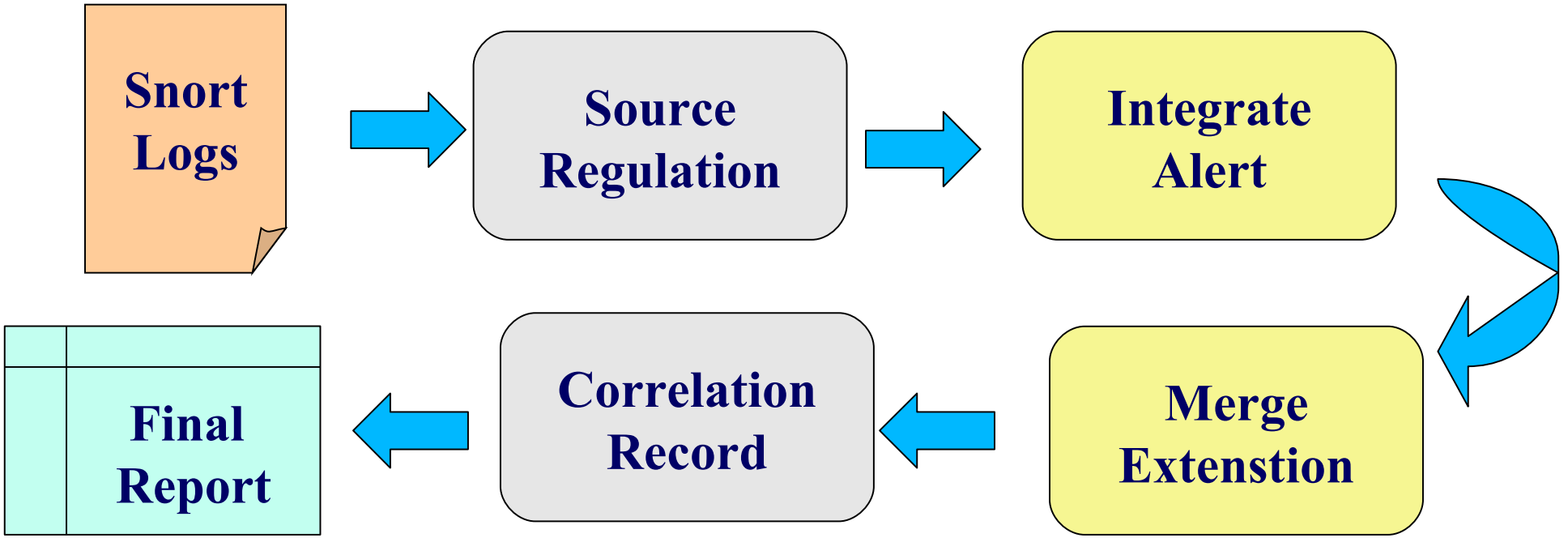
ICAS-II 所產生的報表：警訊關聯圖

- 經過 ICAS-II 分析後，可以得到此警訊關聯圖。
- 圖中橢圓形代表節點，箭頭及線上文字代表攻擊方向與攻擊方法。
- 標為紅色則是經過系統分析之後，被判定有攻擊行為的節點與方法。
- 此圖說明 IP 168.150.177.166 與 168.150.177.164 有進行蠕蟲的攻擊行為



ICAS-II 的分析流程

- Hadoop v 0.20



ICAS-II 結論

- ICAS-II 可經過警訊的來源、目的、攻擊事件綜合分析
  - 提供巨觀攻擊關聯圖來瞭解攻擊事件的始末
  - 自動透過標記顏色的方法將較高危險的事件呈現出來。
- ICAS-II 尚在整合關聯式資料庫，因此還未進行數據量測

ICAS 總結

- 雲端運算處理資料格式相似且資料量大的情況下，能展現其效益
- 提供高容錯率、低獨占系統資源、多工作同時執行等能力
- 可搭配其他軟體作即時的警訊資料呈現，ICAS 可補充分析後資料的部份
- 未來工作
  - 整合多種資料來源平台
  - 產生更詳細與人性化的分析資料

Questions?

Slides - http://trac.nchc.org.tw/cloud

Jazz Wang
Yao-Tsung Wang
jazz@nchc.org.tw