財團法人國家實驗研究院
國家高速網路與計算中心
NATIONAL CENTER FOR HIGH-PERFORMANCE COMPUTING

雲端運算基礎課程

王耀聰 陳威宇
Jazz@nchc.org.tw
waue@nchc.org.tw

國家高速網路與計算中心(NCHC)

課程大綱 （1）

第一天	
09:30~10:30	介紹課程 與 雲端運算簡介
10:20~10:30	休息
10:30~12:00	Hadoop 簡介
	實作： Hadoop 單機安裝
12:00~13:00	午餐
13:00~15:00	Hadoop Distributed File System 簡介
	實作： HDFS 指令操作練習
15:00~15:10	休息
15:10~16:30	Map Reduce 介紹
	實作： 執行 MapReduce 基本運算

課程大綱 （2）

第二天	
09:30~10:10	MapReduce 程式設計
10:10~10:20	實作： Hadoop 程式編譯與執行
10:20~10:30	休息
10:30~12:00	Eclipse 與 Hadoop 的邂逅
	Nutch 搜尋引擎實做
12:00~13:00	午餐
13:00~14:00	Hadoop 叢集安裝設定解析
14:00~15:00	實作： Hadoop 叢集環境操作
15:00~15:10	休息
15:10~16:00	Hadoop_DRBL快速佈屬
16:00~16:30	實作：DRBL 快速佈屬 Hdoop 叢集

淺談雲端運算的新趨勢
The Trend of Cloud Computing

Jazz Wang

Yao-Tsung Wang

jazz@nchc.org.tw

Powered by DRBL

NCHC

What is Cloud Computing?
何謂雲端運算?請用一句話說明!

Anytime 隨時

Anywhere 隨地

With Any Devices 使用任何裝置

Accessing Services 存取各種服務

More definition?
其他定義請參考：NIST Notional Definition of Cloud Computing

Cloud Computing =~ Network Computing

雲端運算 =~ 網路運算

NCHC

Evolution of Cloud Services

雲端服務只是軟體演化史的必然趨勢

數位化 → e-Taiwan 成果系列報導 → 行動台灣 應用推動計畫 → 發展優質網路社會 (U-Taiwan) 計畫

實體	單機版 個人使用	網路版 多人共享	行動版 隨時存取
信箱	E-Mail	Web Mail	Mobile Mail
電視	電視盒	Web TV	Mobile TV
打字機	Office	Google Docs	M-Office
電話	數位電話	Skype	Flash Wengo
佈告欄	電子佈告欄	部落格	微網誌

Rome wasn't built in a day !

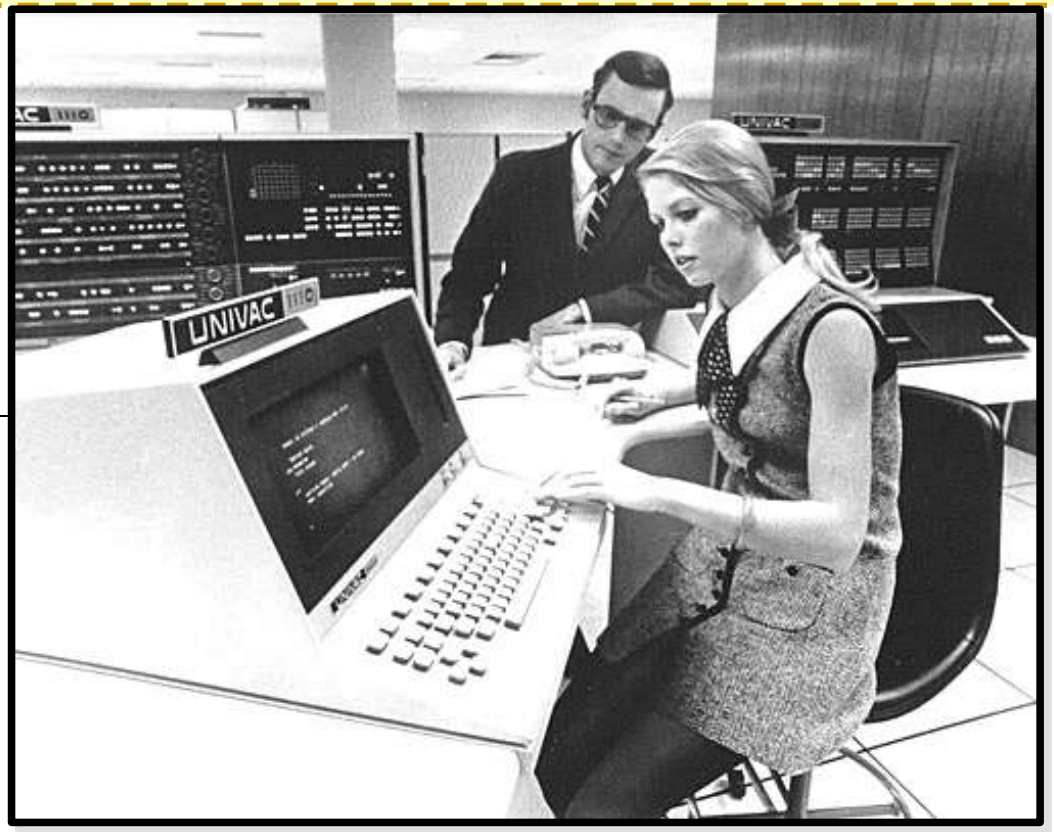
羅馬不是一天造成的!

圖片來源：http://www.mjjq.com/pic/20070822/20070822234234402.jpg

When did the Cloud come ?!

這朵雲幾時飄過來的?!

Brief History of Computing (1/5)



Source: http://pinedakrch.files.wordpress.com/2007/07/

1960 PDP-1

.

.

.

1965 PDP-7

.

.

.

1969 1st Unix

Mainframe

Super Computer

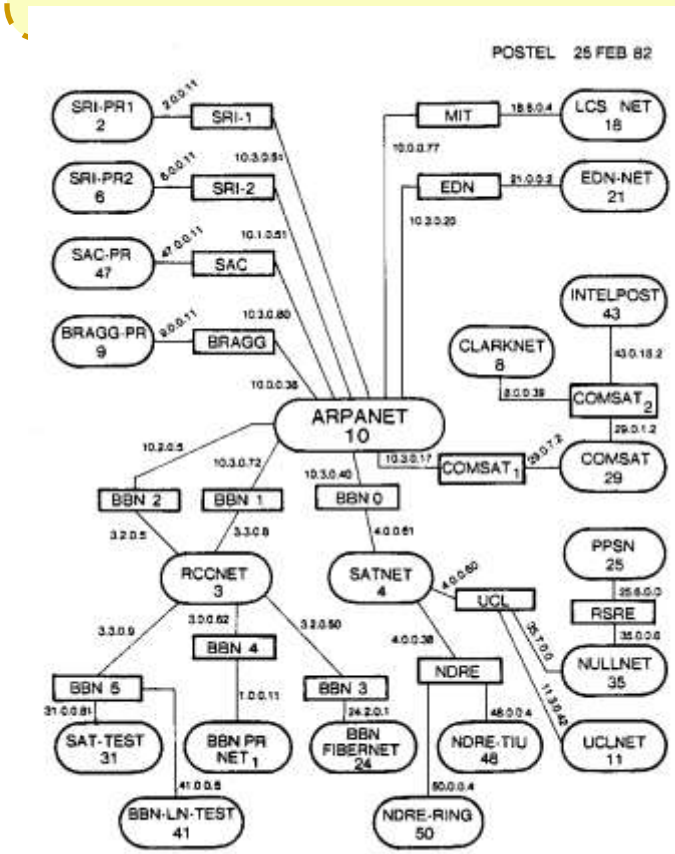
1977 Apple II

1981 IBM 1st PC 5150

Back to Year 1970s ...

1982 TCP/IP

1983 GNU



1991 Linux

Back to Year 1980s ...

Brief History of Computing (2/5)

Source: http://www.nchc.org.tw

Mainframe
Super Computer

PC / Linux
Cluster Parallel

NCHC

1990 World Wide Web by CERN

...

...

1993 Web Browser Mosaic by NCSA

1991 CORBA

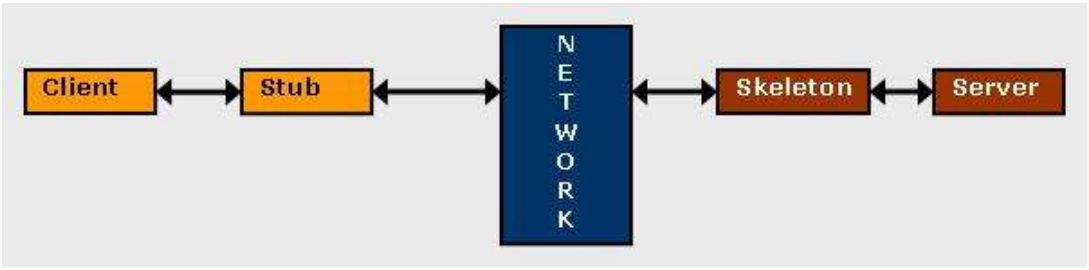
...

Java RMI

Microsoft DCOM

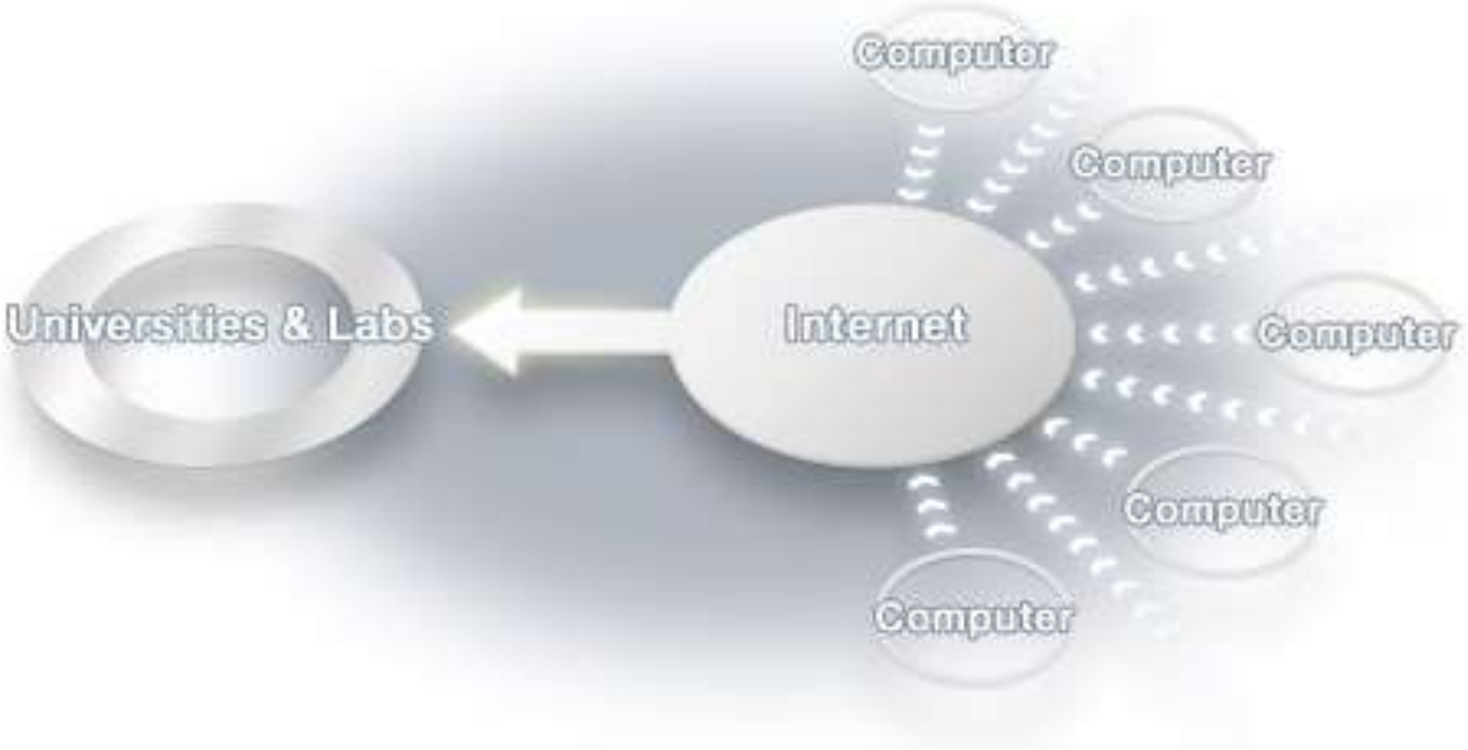
...

Distributed Objects



Back to Year 1990s ...

Brief History of Computing (3/5)



Source: http://www.scei.co.jp/folding/en/dc.html

Mainframe Super Computer → **PC / Linux** Cluster Parallel → **Internet** Distributed Computing

1997 Volunteer Computing

1999 SETI@HOME

2003 Globus Toolkit 2

2002 Berkley BOINC

2004 EGEE gLite

Back to Year 2000s ...

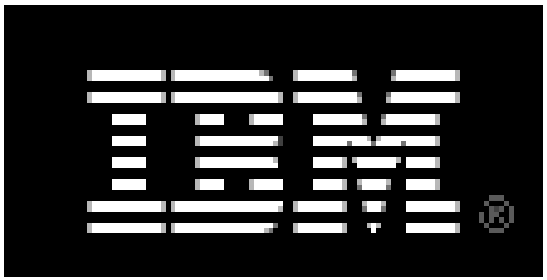
Brief History of Computing (4/5)

Source: http://gridcafe.web.cern.ch/gridcafe/whatisgrid/whatis.html

Mainframe	PC / Linux	Internet	Virtual Org.
Super Computer	Cluster Parallel	Distributed Computing	Grid Computing

NCHC

2001 Autonomic Computing
IBM

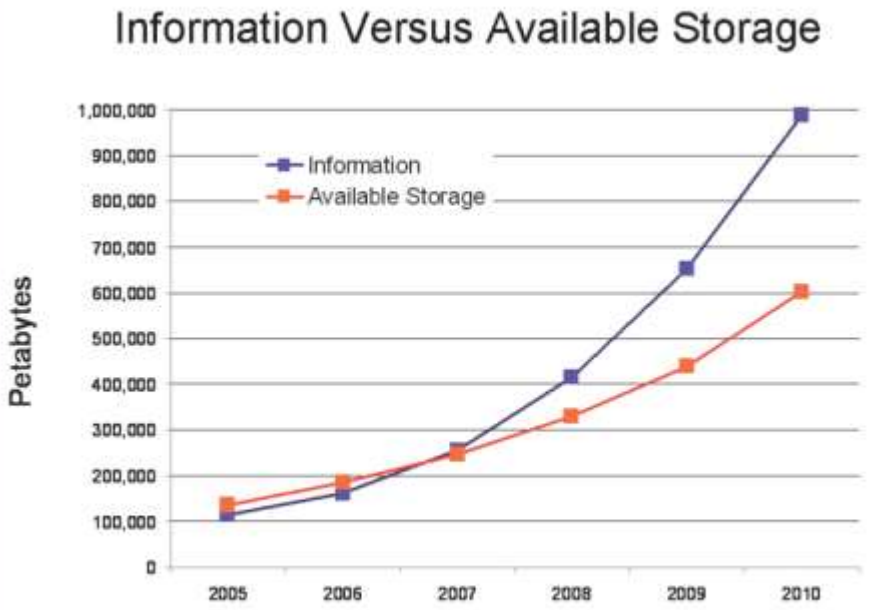


2006 Apache Hadoop

2005 Utility Computing
Amazon EC2 / S3

2007 **Cloud Computing**
Google + IBM

Back to Year 2007 ...

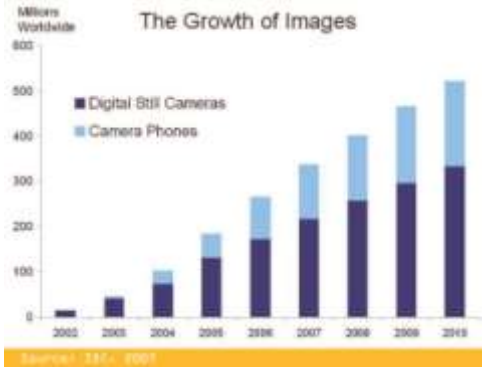


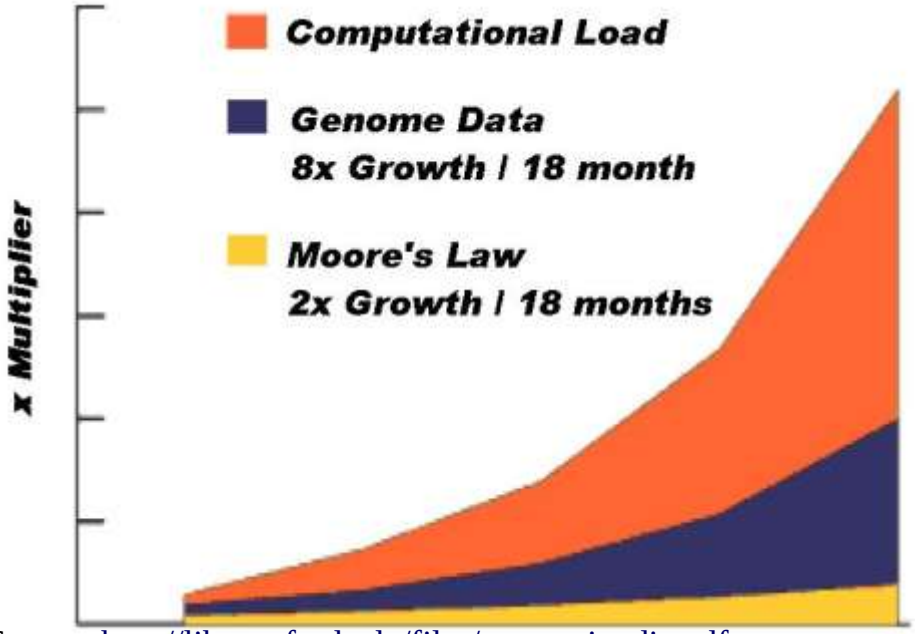
Source: http://www.emc.com/collateral/analyst-reports/expanding-digital-idc-white-paper.pdf

Source: IDC, 2007

2007 Data Explore

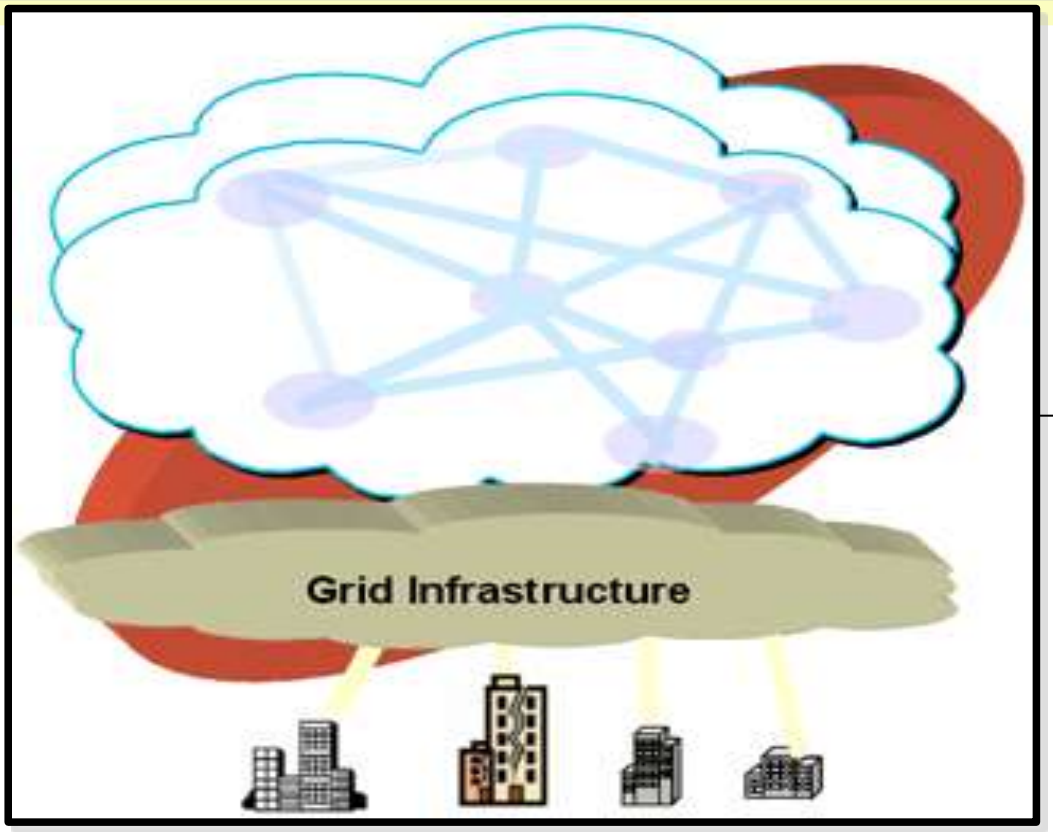
Top 1 : Human Genomics – 7000 PB / Year
Top 2 : Digital Photos – 1000 PB+/ Year
Top 3 : E-mail (no Spam) – 300 PB+ / Year





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

Brief History of Computing (5/5)



Source: http://mmdays.com/2008/02/14/cloud-computing/

nframe
per
puter

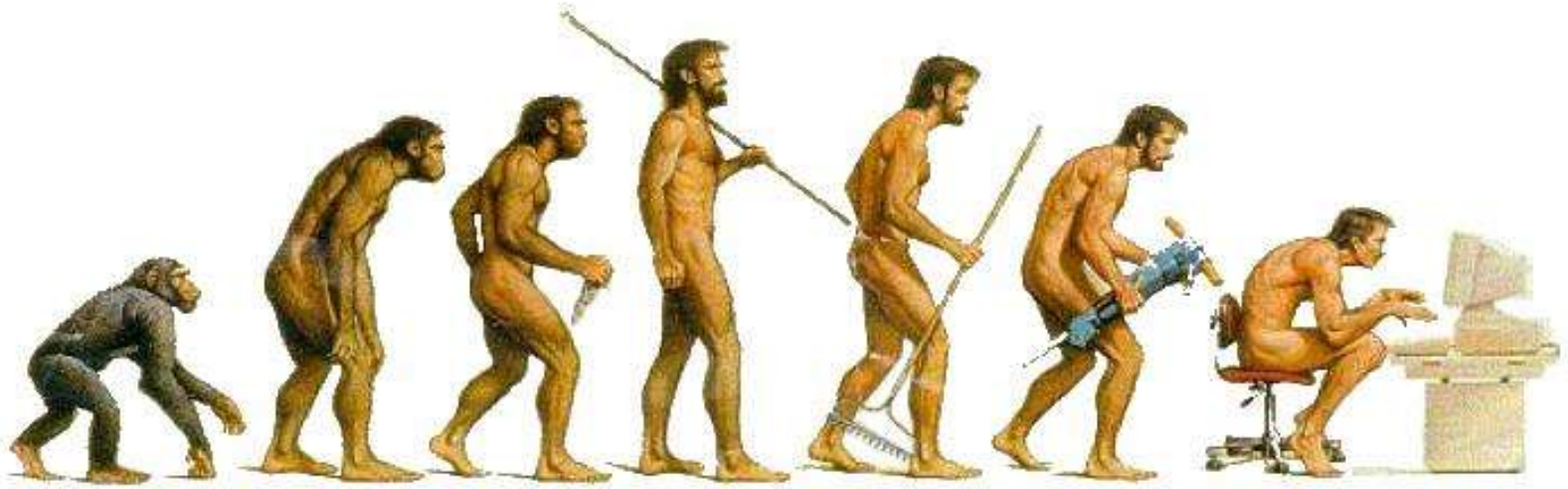
PC / Linux
Cluster
Parallel

Internet
Distributed
Computing

Virtual Org.
Grid
Computing

Data Explode
Cloud
Computing

Evolution



(or is it?)

What can we learn from the past ?!

在這漫長的演化中，我們到底學到些什麼?!

Source: http://cyberpingui.free.fr/humour/evolution-white.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

This is why Cloud Computing matters ?!

這就是為什麼雲端運算變得熱門?!

NCHC

Trend #1: Data are moving to the Cloud
趨勢一：資料開始回歸集中管理

Access data anywhere anytime 為了隨時存取

Reduce the risk of data lost 降低資料遺失風險

Reduce data transfer cost 減少資料傳輸成本

Enhance team collaboration 促進團隊協同合作

How to store huge data ?!

如何儲存大量資料呢?!

Trend #2: Web become default Platform!
趨勢二：網頁變成預設開發平台

Open Standard 網頁是開放標準

Open Implementation 實作不受壟斷

Cross Platform 瀏覽器成為跨平台載具

Web Application 網頁程式設計成為顯學

Browser difference become entry barrier ?!

瀏覽器的差異造成新的技術門檻?!

NCHC

Trend #3: HPC become a new industry
趨勢三：高速計算已悄悄變成新興產業

Parallel Computing 平行運算的技能

Distributed Computing 分散運算的技能

Multi-Core Programming 多核心程式設計

Processing Big Data 處理大資料的技能

Education and Training are needed !!

為了讓這些技能與產業接軌，亟需教育訓練!!

Flying to the Cloud …
or
Falling to the Ground …

Source: http://media.photobucket.com/image/falling%20ground/preeto_f10/falling.jpg

該使用別人打造的雲端，還是自己打造專屬雲端呢?

Types of Cloud Computing
雲端運算的三種型態

Microsoft

Google

Public Cloud

公用雲端

Target Market is S.M.B.

主要客戶為中小企業

Hybrid Cloud

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

以大型企業為主要客戶

Enterprise is key market

IBM

私有雲端

Private Cloud

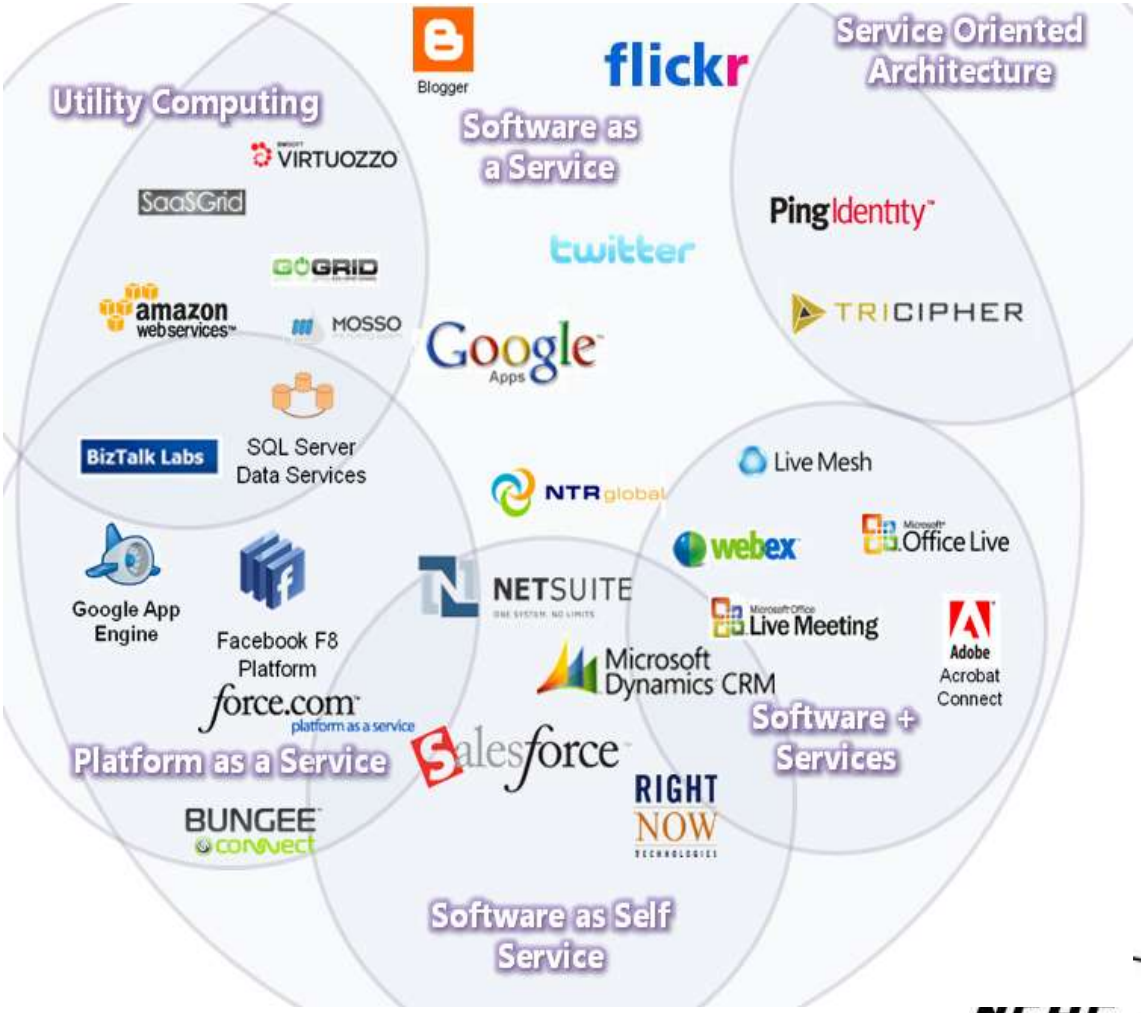
NCHC

Types of Cloud Service Provider
雲端服務的市場區隔

SaaS
Software as a Service
軟體即服務

PaaS
Platform as a Service
平台即服務

IaaS
Infrastructure as a Service
架構即服務



Everything as a Service 啥米鬼都是一種服務

- AaaS Architecture as a Service
- BaaS Business as a Service
- CaaS Computing as a Service
- DaaS Data as a Service
- DBaaS Database as a Service
- EaaS Ethernet as a Service
- FaaS Frameworks as a Service
- GaaS Globalization or Governance as a Service
- HaaS Hardware as a Service
- IMaaS Information as a Service
- **IaaS Infrastructure or Integration as a Service**
- IDaaS Identity as a Service
- LaaS Lending as a Service
- MaaS Mashups as a Service
- OaaS Organization or Operations as a Service
- **SaaS Software or Storage as a Service**
- **PaaS Platform as a Service**
- TaaS Technology or Testing as a Service
- VaaS Voice as a Service

Customer-Oriented

客戶導向

引用自：
https://www.ibm.com/developerworks/mydeveloperworks/blogs/sbose/entry/gathering_clouds_of_xaas

Public Cloud #1: Amazon 亞馬遜網路書店

- Amazon Web Service (AWS)
- 虛擬伺服器：Amazon EC2
  - Small (Default) $0.10 per hour $0.125 per hour
  - All Data Transfer $0.10 per GB
- 儲存服務：Amazon S3
  - $0.150 per GB – first 50 TB / month of storage used
  - $0.100 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

Public Cloud #2: Google 谷歌

- **Google App Engine (GAE)**

•讓開發者可自行建立網路應用程式於Google平台中。

•提供：

- 500MB of storage
- up to 5 million page views a month
- 10 applications per developer account

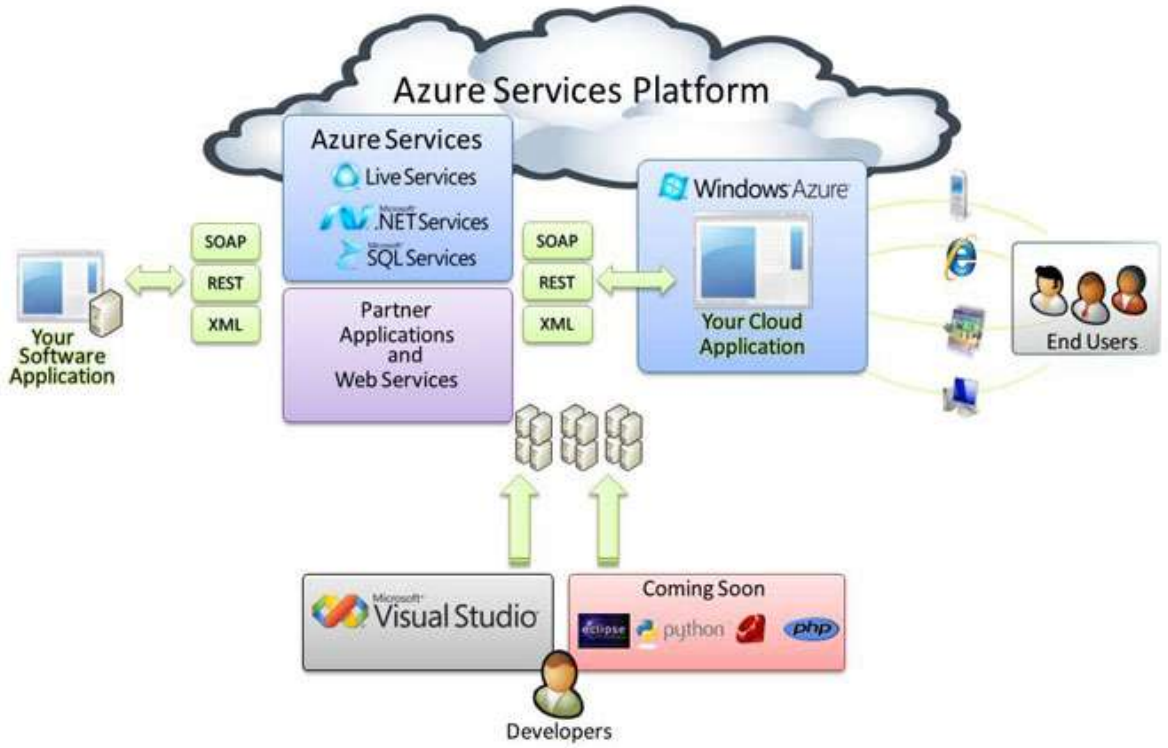
•限制：

- 程式設計語言: Python、Java

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

Public Cloud #3: Microsoft 微軟

- Microsoft Azure 是一套雲端服務作業系統。
- 作為 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

Reference Cloud Architecture

雲端運算的參考架構

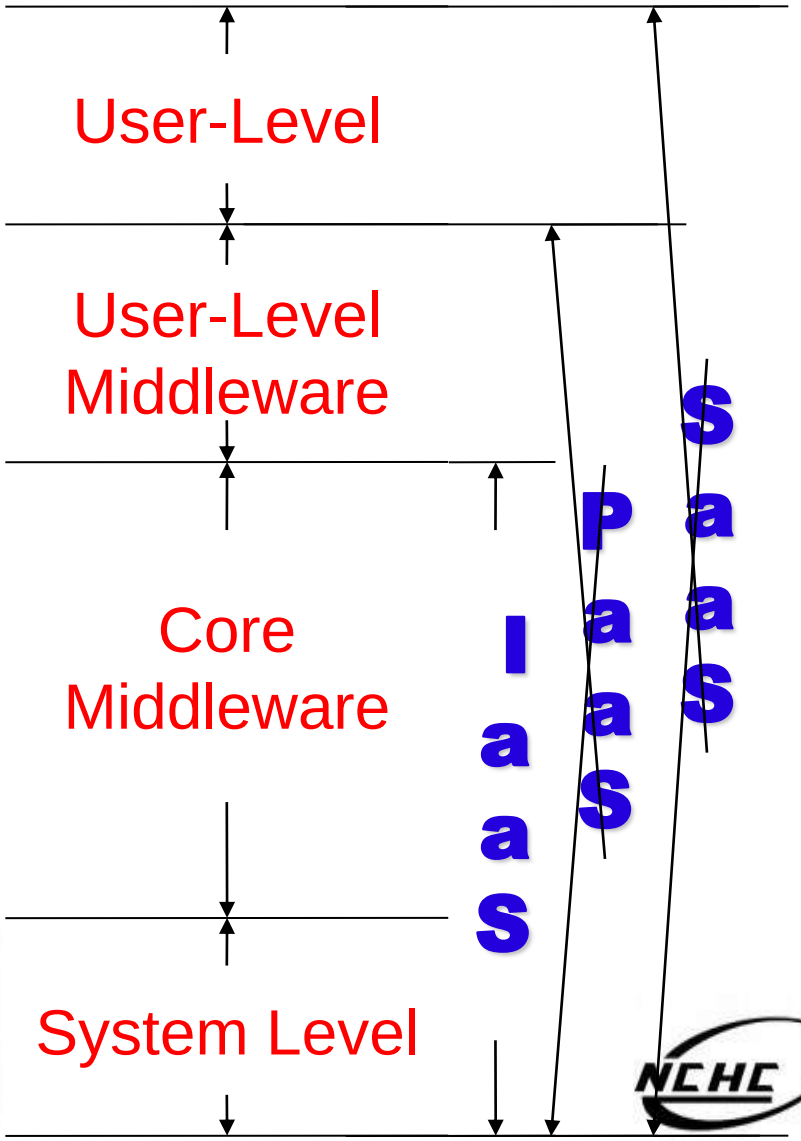
應用
Social Computing, Enterprise, ISV,…

程式語言
Web 2.0 介面, Mashups, Workflows, …

控制
Qos Neqotiation, Ddmission Control, Pricing, SLA Management, Metering…

虛擬化
VM, VM management and Deployment

硬體設施
Infrastructure: Computer, Storage, Network



Open Source for Private Cloud

建構私有雲端運算架構的自由軟體

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

NCHC

Open Cloud #1: Eucalyptus

- http://open.eucalyptus.com/
- 原是加州大學聖塔芭芭拉分校(UCSB)的研究專案
- 目前已轉由Eucalyptus System這間公司負責維護
- 創立目的是讓使用者可以打造自己的EC2
- 特色是相容於 Amazon EC2 既有的用戶端介面
- 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
- 缺點：目前仍有部分操作需透過指令模式

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

NCHC

Open Cloud #2: OpenNebula

- http://www.opennebula.org

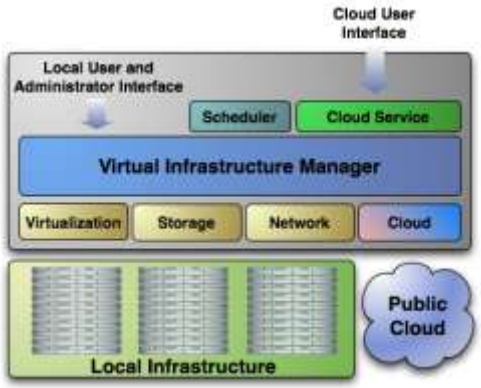
OpenNebula.org

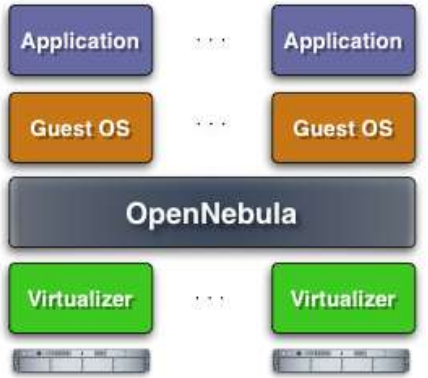
- 由歐洲研究學會(European Union FP7)贊助
- 將實體叢集轉換成具管理彈性的虛擬基礎設備
- 可管理虛擬叢集的狀態、排程、遷徙(migration)
- 優勢是Ubuntu 9.04 已經收錄 OpenNebula 的套件
- 缺點：需下指令來進行虛擬機器的遷徙(migration)。

SEVENTH FRAMEWORK PROGRAMME

RESERVOIR

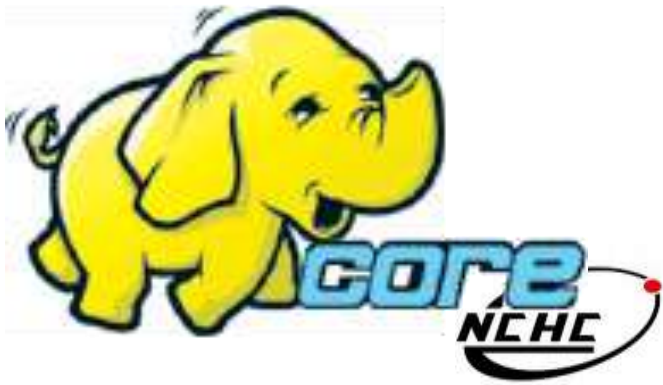
關於 OpenNebula 的更多資訊，請參考
http://trac.nchc.org.tw/grid/wiki/OpenNEbula





Open Cloud #3: Hadoop

- http://hadoop.apache.org
- Hadoop 是 Apache Top Level 開發專案
- 目前主要由 Yahoo! 資助、開發與運用
- 創始者是Doug Cutting，參考Google Filesystem，以Java開發，提供HDFS與MapReduce API。
- 2006年使用在Yahoo內部服務中
- 已佈署於上千個節點。
- 處理Petabyte等級資料量。
- Facebook、Last.fm、Joost … 等
- 著名網路服務均有採用Hadoop。



Open Cloud #4: Sector / Sphere

- http://sector.sourceforge.net/
- 由美國資料探勘中心(National Center for Data Mining)研發的自由軟體專案。
- 採用C/C++語言撰寫，因此效能較 Hadoop 更好。
- 提供「類似」Google File System與MapReduce的機制
- 基於UDT高效率網路協定來加速資料傳輸效率
- Open Cloud Consortium的Open Cloud Testbed，有提供測試環境，並開發了MalStone效能評比軟體。

National Center for Data Mining
University of Illinois at Chicago

UIC

open data

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

NCHC

Questions?

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

Jazz Wang

Yao-Tsung Wang

jazz@nchc.org.tw

Powered by DRBL

NCHC

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2007年繼格網運算之後的新趨勢!! Cloud Computing become new trend since year 2007 !!
WHY	資料集中、虛擬化、異業資源共享 Data-intensive, Virtualization, Heterogeneous
HOW	採用自由軟體也能打造私有雲端 Hadoop, Sectore/Sphere, Eucalyptus, and more ....

NCHC Cloud Computing Research Group

團隊小檔案：國網中心雲端運算研究小組

- 主要研究雲端運算的基礎架構組成元件
- http://trac.nchc.org.tw/cloud, http://trac.nchc.org.tw/grid
- 團隊成員：6名
  - 王耀聰–drbl-xen / drbl-hadoop (~6 Years) 架構
  - 陳威宇–Hadoop / NutchEz / ICAS (~3 Years) 應用
  - 郭文傑–Xen / OpenNebula / Eucalyptus (~3 Years) 元件
  - 涂哲源–Xen GPU / OpenMP / VirtualGL (~3 Years) 元件
  - 鄭宗碩–Google App Engine (~2 Years) 新技術
  - 鄧偉華–AMQP / OpenID (~2 Years) 新技術
- 定位：
  - 研發快速佈建軟體，提供實驗平台服務，開辦訓練課程育才
- 獨特性：
  - 基於企鵝龍(DRBL)，可快速佈署雲端運算的叢集環境

更多相關的開放教材－生物叢集、GAE...

- 陽明生資所**97**年度暑期學分班 格網及平行運算(實驗課程) **http://trac.nchc.org.tw/course/**
- 陽明生資所**98**年度暑期學分班 格網及平行運算(實驗課程) **http://bio.classcloud.org**
- 雲端運算基礎課程(一) **Hadoop**簡介、安裝與範例實作 **http://www.classcloud.org/media/**
- 「**Ruby on Rails** 初學」電子書 **by** 鄭立竺 **http://nchcrails.blogspot.com**
- **Google App Engine** 電子書 **by** 鄭宗碩 **http://nchc-gae.blogspot.com/**
- **More to come .....**

Hadoop 簡介

王耀聰 陳威宇

Jazz@nchc.org.tw

waue@nchc.org.tw

國家高速網路與計算中心(NCHC)

看了這麼多雲端服務

但……

是否有一套能夠

開放給大家使用

的雲端平台呢??

The Other Open Source Projects:

Eucalyptus	University of California, Santa Barbara	http://open.eucalyptus.com/
Sector	The National Center for Data Mining (NCDM)	http://sector.sourceforge.net/
Thrift	Facebook	http://developers.facebook.com/thrift/

Hadoop ?

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

Hadoop

- 以Java開發
- 自由軟體
- 上千個節點
- Petabyte等級的資料量
- 創始者 Doug Cutting
- 為Apache 軟體基金會的 top level project

特色

- 巨量
  - 擁有儲存與處理大量資料的能力
- 經濟
  - 可以用在由一般PC所架設的叢集環境內
- 效率
  - 籍由平行分散檔案的處理以致得到快速的回應
- 可靠
  - 當某節點發生錯誤，系統能即時自動的取得備份資料以及佈署運算資源

起源:2002-2004

- Lucene
  - 用Java設計的高效能文件索引引擎API
  - 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
- Nutch
  - nutch是基於開放原始碼所開發的web search
  - 利用Lucene函式庫開發

怎麼來的

起源：Google論文

- Google File System
  - SOSP 2003 : “The Google File System”
  - OSDI 2004 : “MapReduce : Simplifed Data Processing on Large Cluster”
  - OSDI 2006 : “Bigtable: A Distributed Storage System for Structured Data”
  - 可擴充的分散式檔案系統
  - 大量的用戶提供總體性能較高的服務
  - 對大量資訊進行存取的應用
  - 運作在一般的普通主機上
  - 提供錯誤容忍的能力

怎麼來的

起源:2004~

- Dong Cutting 開始參考論文來實做
- Added DFS & MapReduce implement to Nutch
- Nutch 0.8版之後，Hadoop為獨立項目
- Yahoo 於2006年僱用Dong Cutting 組隊專職開發
  - Team member = 14 （engineers, clusters, users, etc.)
- 2009 年跳槽到Cloudera

有誰在用

誰在用Hadoop

- Yahoo 為最大的贊助商
- IBM 與 Google 在大學開授雲端課程的主要內容
- Hadoop on Amazon Ec2/S3
- More…:

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

實用案例

Hadoop於yahoo的運作資訊

年份	日期	節點數	耗時（小時）
2006	四月	188	47.9
2006	五月	500	42
2006	十一月	20	1.8
2006	十一月	100	3.3
2006	十一月	500	5.2
2006	十一月	900	7.8
2007	七月	20	1.2
2007	七月	100	1.3
2007	七月	500	2
2007	七月	900	2.5

Sort benchmark, every nodes with terabytes data.

實用案例

Hadoop於yahoo的部屬情形

資料標題：Yahoo! Launches World's Largest Hadoop Production Application

資料日期：February 19, 2008

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

實用案例

Hadoop於yahoo的部屬情形

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**

瞭解更多

Hadoop 與google的對應

Develop Group	Google	Apache
Sponsor	Google	Yahoo, Amazon
Algorithm Method	MapReduce	Hadoop
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

動手安裝囉！

Hadoop Overview

王耀聰 陳威宇

Jazz@nchc.org.tw

waue@nchc.org.tw

國家高速網路與計算中心(NCHC)

作業系統的最核心！

儲存空間的資源管理

記憶體空間與
行程分配

名詞

- 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

管理資料

Namenode

- Master
- 管理HDFS的名稱空間
- 控制對檔案的讀/寫
- 配置副本策略
- 對名稱空間作檢查及紀錄
- 只能有一個

Datanode

- Workers
- 執行讀/寫動作
- 執行Namenode的副本策略
- 可多個

分派程序

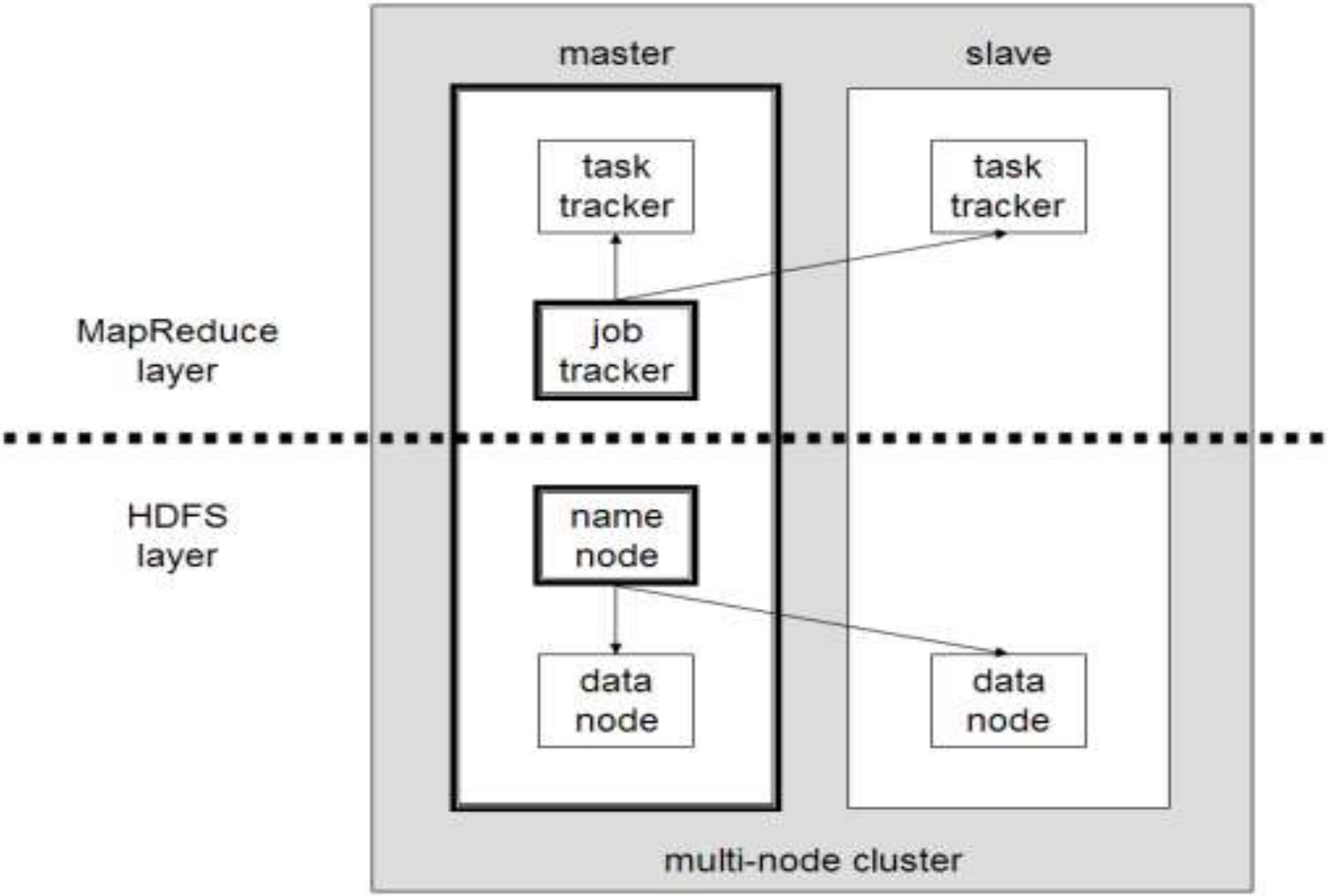
Jobtracker

- Master
- 使用者發起工作
- 指派工作給 Tasktrackers
- 排程決策、工作分配、錯誤處理
- 只能有一個

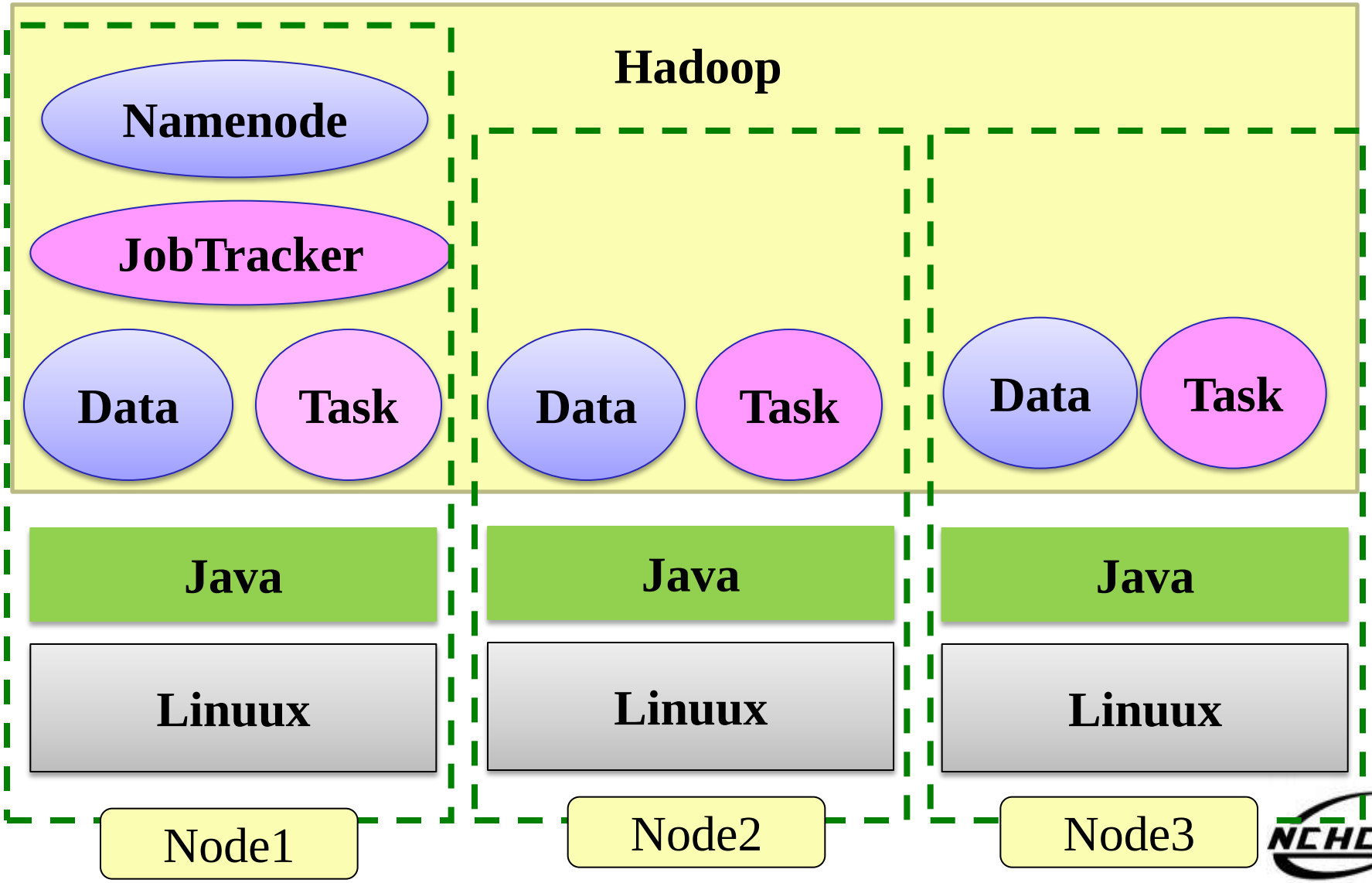
Tasktrackers

- Workers
- 運作Map 與 Reduce 的工作
- 管理儲存、回覆運算結果
- 可多個

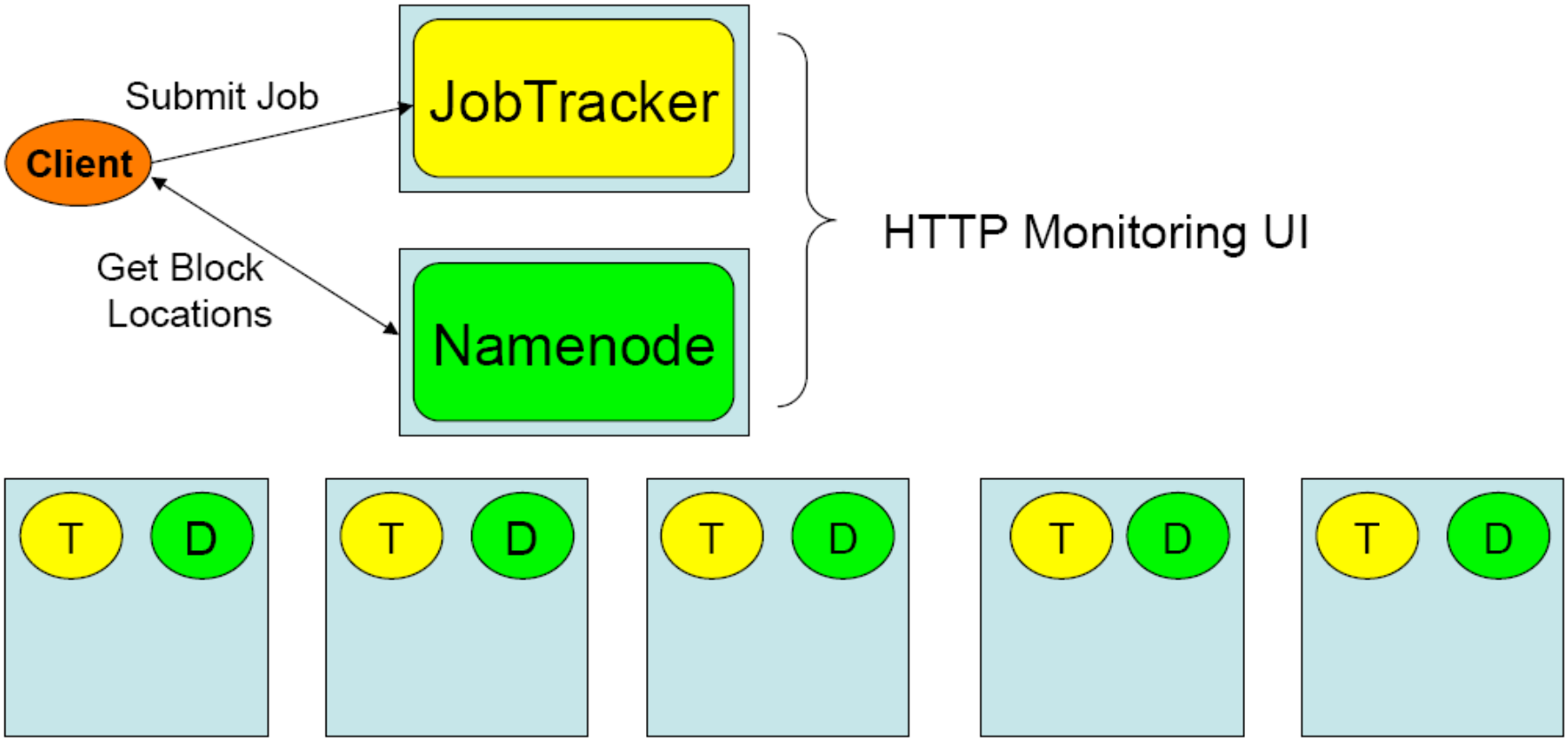
Hadoop的各種身份



Building Hadoop



不在雲裡的 Client



Hadoop Distributed File System

Outline

- HDFS 的定義 ？
- HDFS 的特色？
- HDFS 的架構 ？
- HDFS 運作方式 ？
- HDFS 如何達到其宣稱的好處 ？
- HDFS 功能？

HDFS ?

- Hadoop Distributed File System
  - Hadoop : 自由軟體專案，為實現Google的MapReduce架構
  - HDFS: Hadoop專案中的檔案系統
- 實現類似Google File System
  - GFS是一個易於擴充的分散式檔案系統，目的為對大量資料進行分析
  - 運作於廉價的普通硬體上，又可以提供容錯功能
  - 給大量的用戶提供總體性能較高的服務

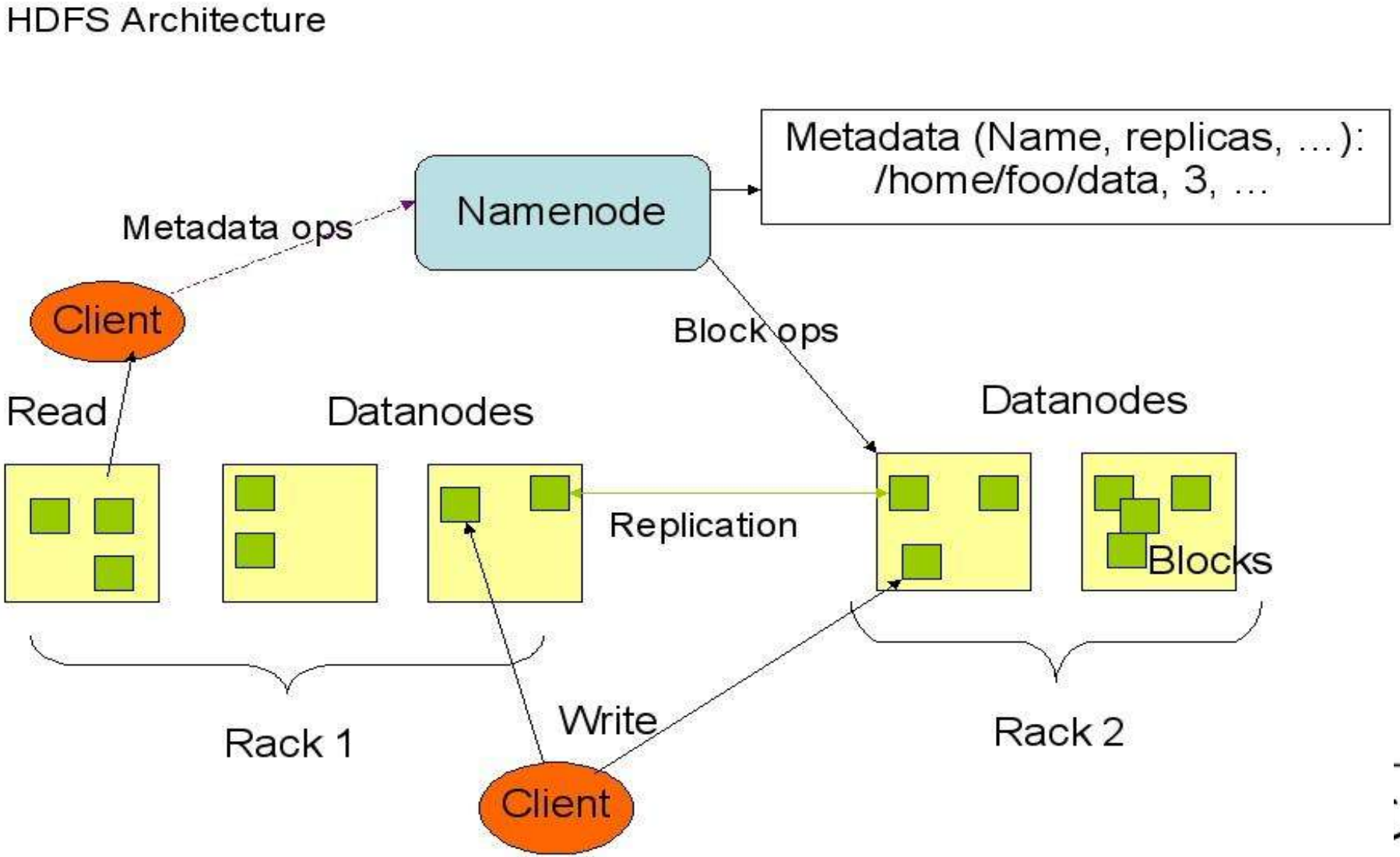
設計目標 (1)

- 硬體錯誤容忍能力
  - 硬體錯誤是正常而非異常
  - 迅速地自動恢復
- 串流式的資料存取
  - 批次處理多於用戶交互處理
  - **高Throughput** 而非低Latency
- 大規模資料集
  - 支援Perabytes等級的磁碟空間

設計目標 (2)

- 一致性模型
  - 一次寫入，多次存取
  - 簡化一致性處理問題
- 在地運算
  - **移動到資料節點計算** > 移動資料過來計算
- 異質平台移植性
  - 即使硬體不同也可移植、擴充

管理資料

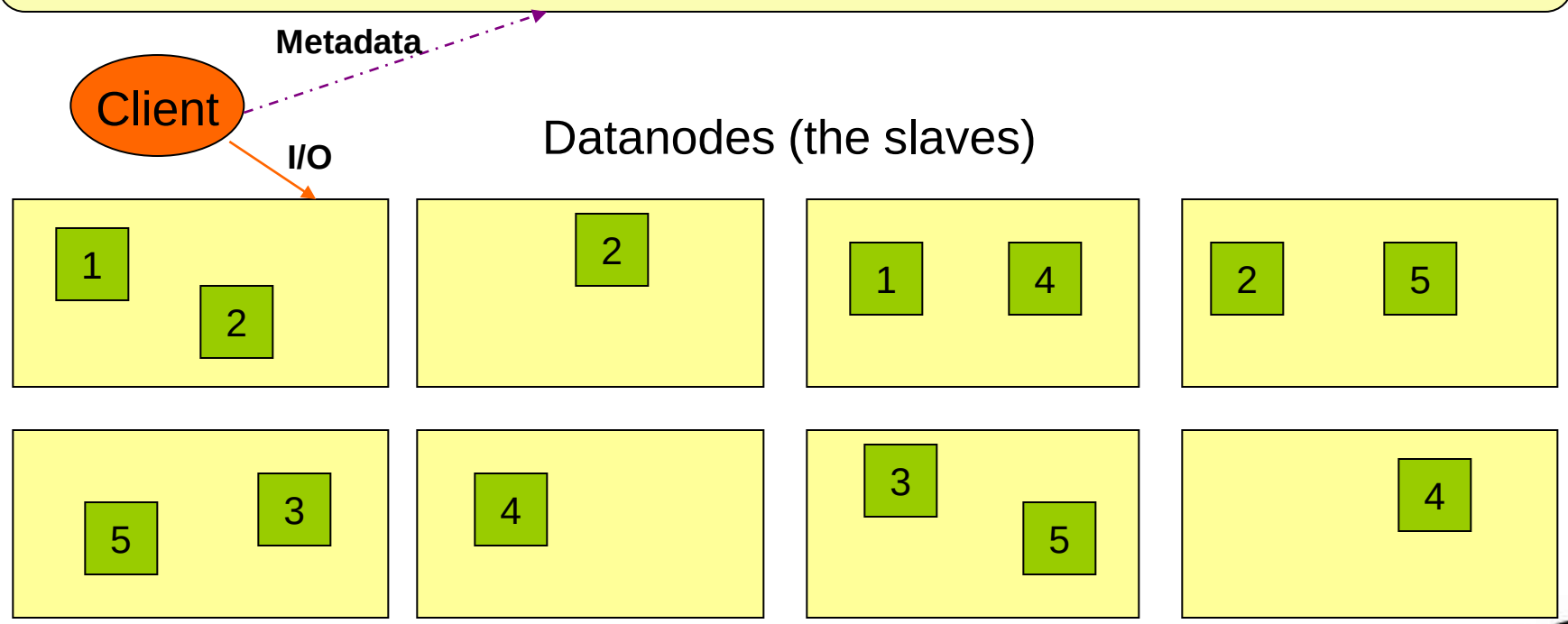

HDFS Architecture
Metadata (Name, replicas, …):
/home/foo/data, 3, …
Namenode
Metadata ops
Client
Block ops
Read
Datanodes
Datanodes
Replication
Blocks
Rack 1
Write
Rack 2
Client

HDFS 運作

Namenode (the master)

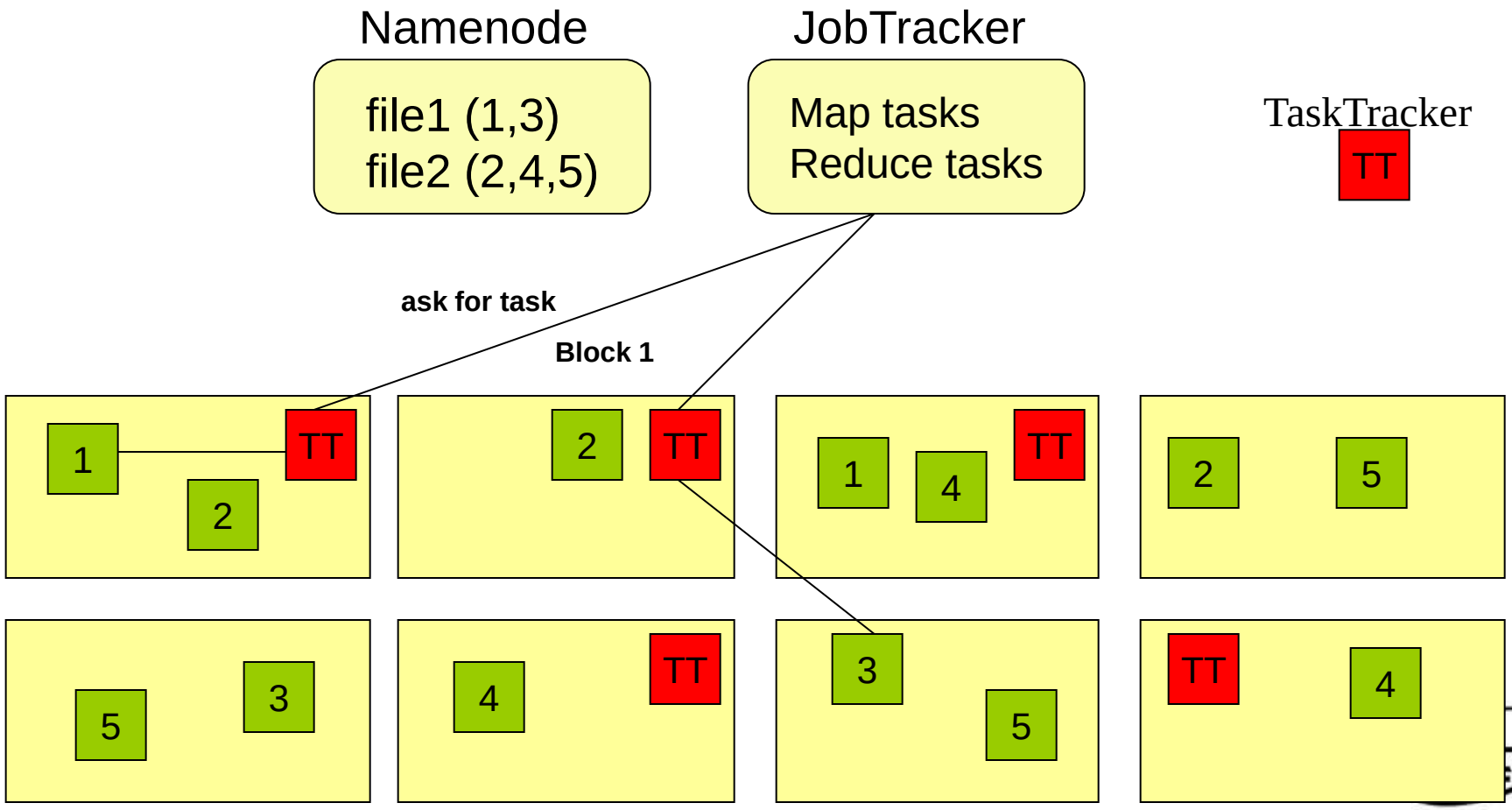
檔案路徑 – 副本數 , 由哪幾個block組成

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

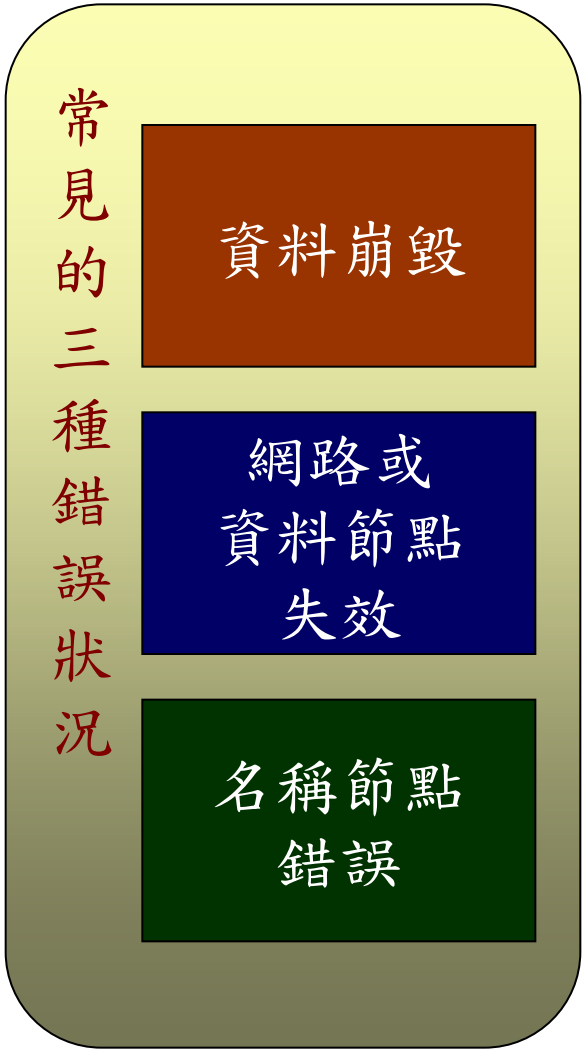


HDFS 運作

- 目的：提高系統的可靠性與讀取的效率
  - 可靠性：節點失效時讀取副本已維持正常運作
  - 讀取效率：分散讀取流量（但增加寫入時效能瓶頸）



可靠性機制



- 資料完整性
  - checked with CRC32
  - 用副本取代出錯資料
- Heartbeat
  - Datanode 定期向Namenode送heartbeat
- Metadata
  - FSImage、Editlog為核心印象檔及日誌檔
  - 多份儲存，當NameNode壞掉可以手動復原

一致性與效能機制

- 檔案一致性機制
  - 刪除檔案＼新增寫入檔案＼讀取檔案皆由Namenode負責
- 巨量空間及效能機制
  - 以Block為單位： 64M為單位
  - 在HDFS上得檔案有可能大過一顆磁碟
  - 大區塊可提高存取效率
  - 區塊均勻散佈各節點以分散讀取流量

HDFS的功能

- 類POXIS指令
- 權限控管
- 超級用戶模式
- Web 瀏覽
- 用戶配額管理
- 分散式複製檔案

POSIX Like

```
hadoop fs [-fs <local | file system URI>] [-conf <configuration file>]
        [-D <property=value>] [-ls <path>] [-lsr <path>] [-du <path>]
        [-dus <path>] [-mv <src> <dst>] [-cp <src> <dst>] [-rm <src>]
        [-rmr <src>] [-put <localsrc> <dst>] [-copyFromLocal <localsrc> <dst>]
        [-moveFromLocal <localsrc> <dst>] [-get <src> <localdst>]
        [-getmerge <src> <localdst> [addnl]] [-cat <src>]
        [-copyToLocal <src><localdst>] [-moveToLocal <src> <localdst>]
        [-mkdir <path>] [-report] [-setrep [-R] [-w] <rep> <path/file>]
        [-touchz <path>] [-test -[ezd] <path>] [-stat [format] <path>]
        [-tail [-f] <path>] [-text <path>]
        [-chmod [-R] <MODE[,MODE]... | OCTALMODE> PATH...]
        [-chown [-R] [OWNER][:[GROUP]] PATH...]
        [-chgrp [-R] GROUP PATH...]
        [-help [cmd]]
```

安裝設定補充說明

王耀聰 陳威宇

jazz@nchc.org.tw

waue@nchc.org.tw

國家高速網路與計算中心(NCHC)

Hadoop Package Topology

資料夾	說明
bin /	各執行檔：如 start-all.sh 、stop-all.sh 、 hadoop
conf /	預設的設定檔目錄：設定環境變數 hadoop-env.sh 、各項參數 hadoop-site.conf 、工作節點 slaves。（可更改路徑）
docs /	Hadoop API 與說明文件 (html & PDF)
contrib /	額外有用的功能套件，如：eclipse的擴充外掛、Streaming 函式庫 。
lib /	開發 hadoop 專案或編譯 hadoop 程式所需要的所有函式庫，如：jetty、kfs。但主要的hadoop函式庫於hadoop_home
src /	Hadoop 的原始碼。
build /	開發Hadoop 編譯後的資料夾。需搭配 ant 程式與build.xml
logs /	預設的日誌檔所在目錄。（可更改路徑）

設定檔：hadoop-env.sh

- 設定Linux系統執行Hadoop的環境參數
  - export xxx=kkk
    - 將kkk這個值匯入到xxx參數中
  - # string…
    - 註解，通常用來描述下一行的動作內容

```
# The java implementation to use.  Required.
export JAVA_HOME=/usr/lib/jvm/java-6-sun
export HADOOP_HOME=/opt/hadoop
export HADOOP_LOG_DIR=$HADOOP_HOME/logs
export HADOOP_SLAVES=$HADOOP_HOME/conf/slaves
…….
```

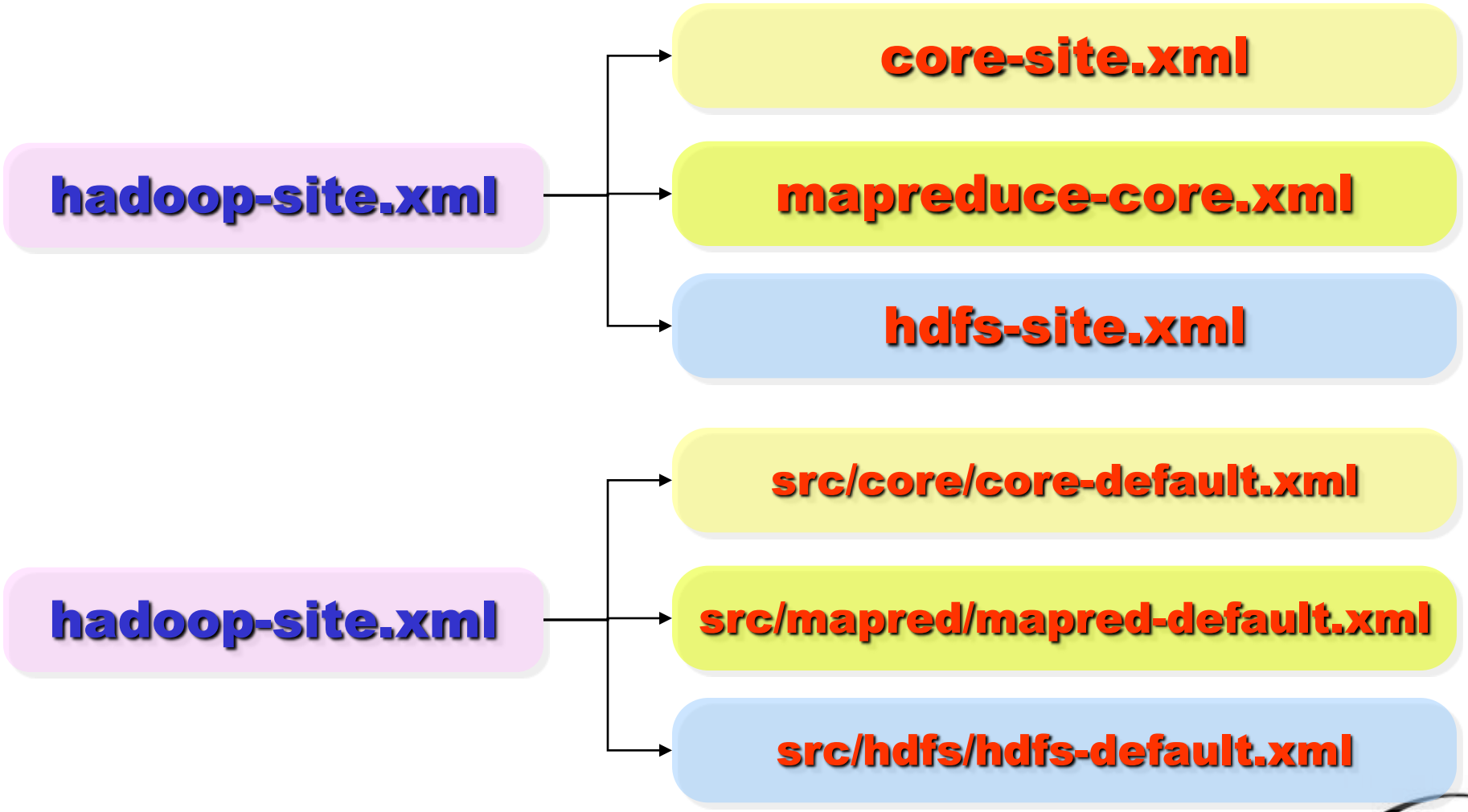
設定檔： hadoop-site.xml (0.18)

```
<configuration>
<property>
  <name> fs.default.name</name>
  <value> hdfs://localhost:9000/</value>
  <description> …  </description>
</property>
<property>
  <name> mapred.job.tracker</name>
  <value> localhost:9001</value>
  <description>…  </description>
</property>
<property>
   <name> hadoop.tmp.dir </name>
   <value> /tmp/hadoop/hadoop-
     ${user.name} </value>
     <description> </description>
</property>
<property>
  <name> mapred.map.tasks</name>
  <value> 1</value>
  <description> define mapred.map tasks to
     be number of slave hosts
     </description>
</property>
<property>
  <name> mapred.reduce.tasks</name>
  <value> 1</value>
  <description> define mapred.reduce tasks
     to be number of slave hosts
     </description>
</property>
<property>
  <name> dfs.replication</name>
  <value> 3</value>
</property>
</configuration>
```

設定檔：hadoop-default.xml (0.18)

- Hadoop預設參數
  - 沒在hadoop.site.xml設定的話就會用此檔案的值
  - 更多的介紹參數：
    http://hadoop.apache.org/core/docs/current/cluster_setup.html#Configuring+the+Hadoop+Daemons

Hadoop 0.18 到 0.20 的轉變



設定檔： core-site.xml (0.20)

```
<configuration>
<property>
  <name> fs.default.name</name>
  <value> hdfs://localhost:9000/</value>
  <description> …   </description>
</property>
<property>
   <name> hadoop.tmp.dir </name>
   <value> /tmp/hadoop/hadoop-
     ${user.name} </value>
     <description> … </description>
</property>
<configuration>
```

詳細 hadoop core 參數，
請參閱 http://hadoop.apache.org/common/docs/current/core-default.html

設定檔： mapreduce-site.xml (0.20)

```
<configuration>
<property>
  <name> mapred.job.tracker</name>
  <value> localhost:9001</value>
  <description>…  </description>
</property>

<property>
  <name> mapred.map.tasks</name>
  <value> 1</value>
  <description> …  </description>
</property>

<property>
  <name> mapred.reduce.tasks</name>
  <value> 1</value>
  <description> … </description>
</property>
</configuration>
```

詳細 hadoop mapreduce 參數，
請參閱 http://hadoop.apache.org/common/docs/current/mapred-default.html

設定檔： hdfs-site.xml (0.20)

```
<configuration>
<property>
  <name> dfs.replication </name>
  <value> 3</value>
  <description>…   </description>
</property>

<property>
  <name> dfs.permissions </name>
  <value> false </value>
  <description> …   </description>
</property>

</configuration>
```

詳細 hadoop hdfs 參數，
請參閱 http://hadoop.apache.org/common/docs/current/hdfs-default.html

設定檔： slaves

- 給 start-all.sh , stop-all.sh 用
- 被此檔紀錄到的節點就會附有兩個身份：datanode & tasktracker
- 一行一個hostname 或 ip

```
192.168.1.1
….
192.168.1.100
Pc101
….
Pc152
….
```

設定檔： masters

- 給 start-*.sh , stop-*.sh 用
- 會被設定成 secondary namenode
- 可多個

192.168.1.1
….
Pc101
….

描述名稱	設定名稱	所在檔案
JAVA安裝目錄	JAVA_HOME	hadoop-env.sh
HADOOP家目錄	HADOOP_HOME	hadoop-env.sh
設定檔目錄	HADOOP_CONF_DIR	hadoop-env.sh
日誌檔產生目錄	HADOOP_LOG_DIR	hadoop-env.sh
HADOOP工作目錄	hadoop.tmp.dir	hadoop-site.xml
JobTracker	mapred.job.tracker	hadoop-site.xml
Namenode	fs.default.name	hadoop-site.xml
TaskTracker	(hostname)	slaves
Datanode	(hostname)	slaves
第二Namenode	(hostname)	masters
其他設定值	詳可見hadoop-default.xml	hadoop-site.xml

控制 Hadoop 的指令

- 格式化
  - $ bin/hadoop ∆ namenode ∆ -format
- 全部開始 (透過 SSH)
  - $ bin/start-all.sh
  - $ bin/start-dfs.sh
  - $ bin/start-mapred.sh
- 全部結束 (透過 SSH)
  - $ bin/stop-all.sh
  - $ bin/stop-dfs.sh
  - $ bin/stop-mapred.sh
- 獨立啟動/關閉(不會透過 SSH)
  - $ bin/hadoop-daemon.sh [start/stop] namenode
  - $ bin/hadoop-daemon.sh [start/stop] secondarynamenode
  - $ bin/hadoop-daemon.sh [start/stop] datanode
  - $ bin/hadoop-daemon.sh [start/stop] jobtracker
  - $ bin/hadoop-daemon.sh [start/stop] tasktracker

Hadoop 的操作與運算指令

- 使用hadoop檔案系統指令
  - $ bin/hadoop △ fs △ −Instruction △ …
- 使用hadoop運算功能
  - $ bin/hadoop △ jar △ XXX.jar △ Main_Function △ ⋯

Hadoop 使用者指令

$ bin/hadoop △**指令**△選項△參數△….

指令	用途	舉例
fs	對檔案系統進行操作	hadoop△**fs**△–put△in△input
jar	啟動運算功能	hadoop△**jar**△example.jar△wc△in△out
archive	封裝hdfs上的資料	hadoop△**archive**△foo.har△/dir △/user/hadoop
distcp	用於叢集間資料傳輸	hadoop△**distcp**△hdfs://nn1:9000/aa △hdfs://nn2:9000/aa
fsck	hdfs系統檢查工具	hadoop△**fsck**△/aa△-files△-blocks △-locations
job	操作正運算中的程序	hadoop△**job** △–kill △jobID
version	顯示版本	hadoop△**version**

Hadoop 管理者指令

$ bin/hadoop Δ**指令** Δ選項 Δ參數 Δ….

指令	用途	舉例
balancer	平衡hdfs覆載量	hadoopΔ**balancer**
dfsadmin	配額、安全模式 等管理員操作	hadoopΔ**dfsadmin**Δ–setQuotaΔ3 Δ/user1/
namenode	名稱節點操作	hadoopΔ**namenode**Δ-format

$ bin/hadoop Δ**指令**

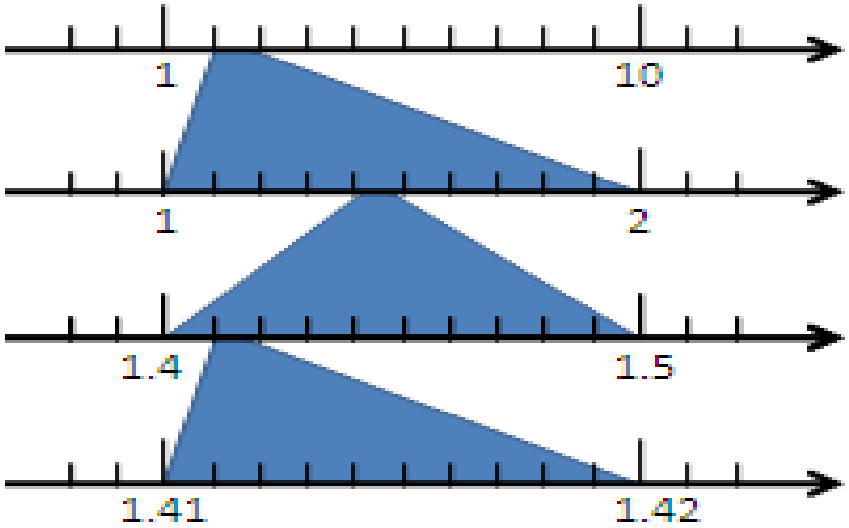
datanode	成為資料節點	hadoopΔdatanode
jobtracker	成為工作分派者	hadoopΔjobtracker
tasktracker	成為工作執行者	hadoopΔtasktracker

Map Reduce 介紹

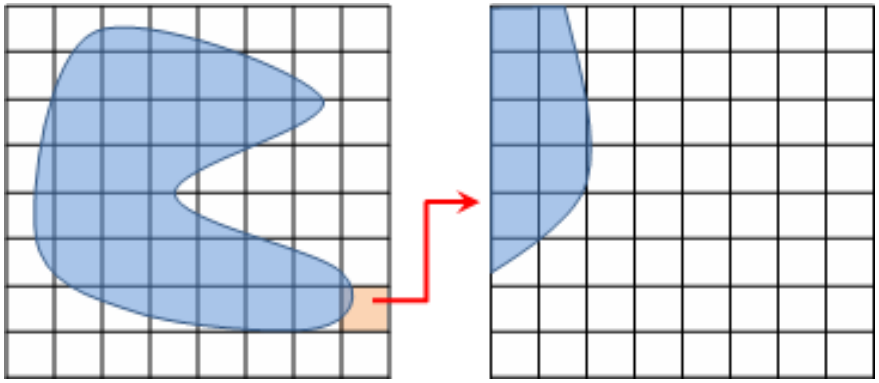
王耀聰 陳威宇
Jazz@nchc.org.tw
waue@nchc.org.tw
國家高速網路與計算中心
(NCHC)

Divide and Conquer

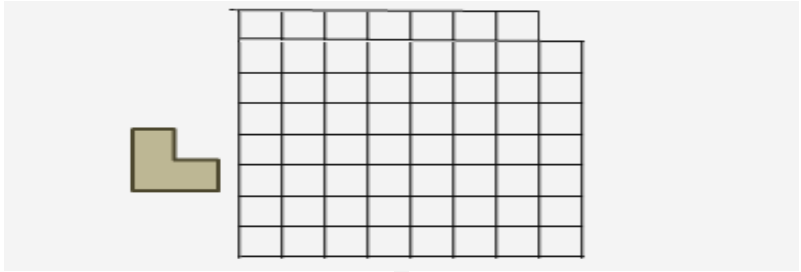
範例一：十分逼近法



範例二：方格法求面積



範例三：鋪滿 L 形磁磚

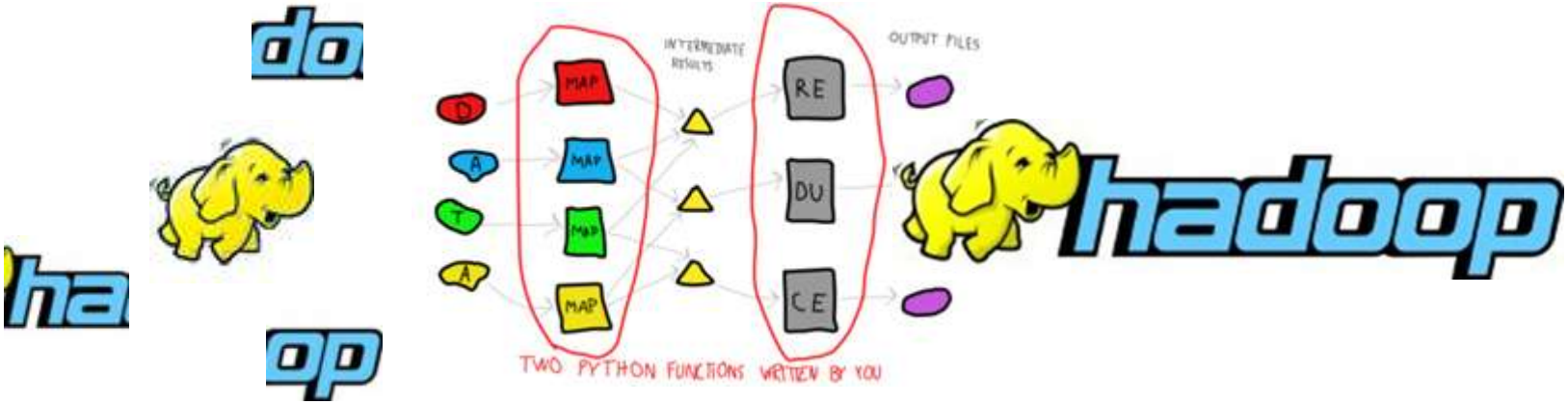


範例四：
眼前有五階樓梯，每次可踏上一階或踏上兩階，那麼爬完五階共有幾種踏法？
Ex：(1,1,1,1,1) or (1,2,1,1)

Map Reduce 起源

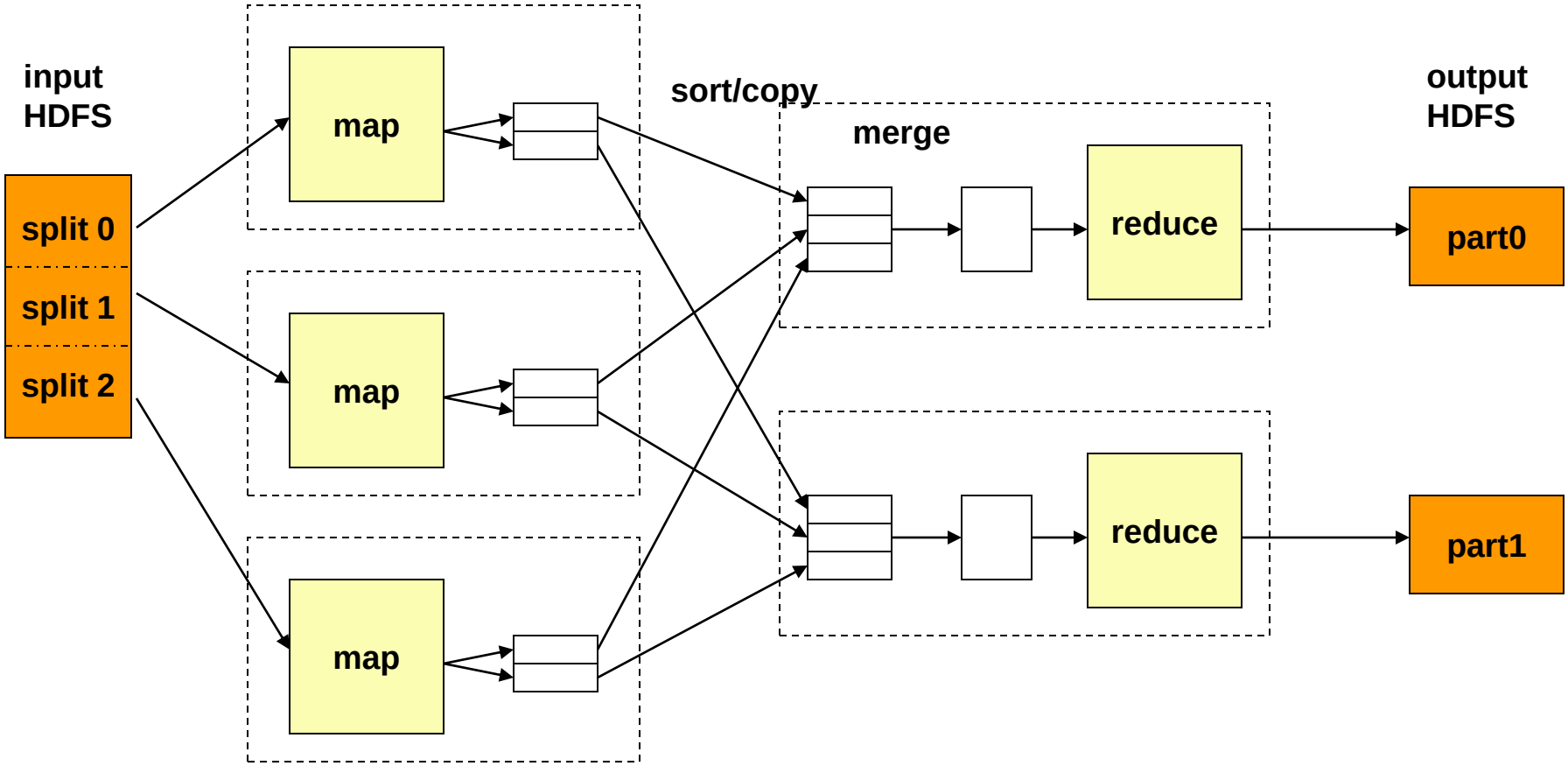
- Functional Programming : Map Reduce
  - map(...) :
    - [1,2,3,4] – (*2) -> [2,4,6,8]
  - reduce(...):
    - [1,2,3,4] - (sum) -> 10
- 演算法（Algorithms）：
  - Divide and Conquer
  - 分而治之
- 在程式設計的軟體架構內，適合使用在大規模數據的運算中

Hadoop MapReduce定義



Hadoop Map/Reduce是一個易於使用的軟體平台，以MapReduce為基礎的應用程序，能夠運作在由上千台PC所組成的大型叢集上，並以一種可靠容錯的方式平行處理上P級別的資料集。

Hadoop-MapReduce 運作流程



JobTracker跟NameNode取得需要運算的blocks

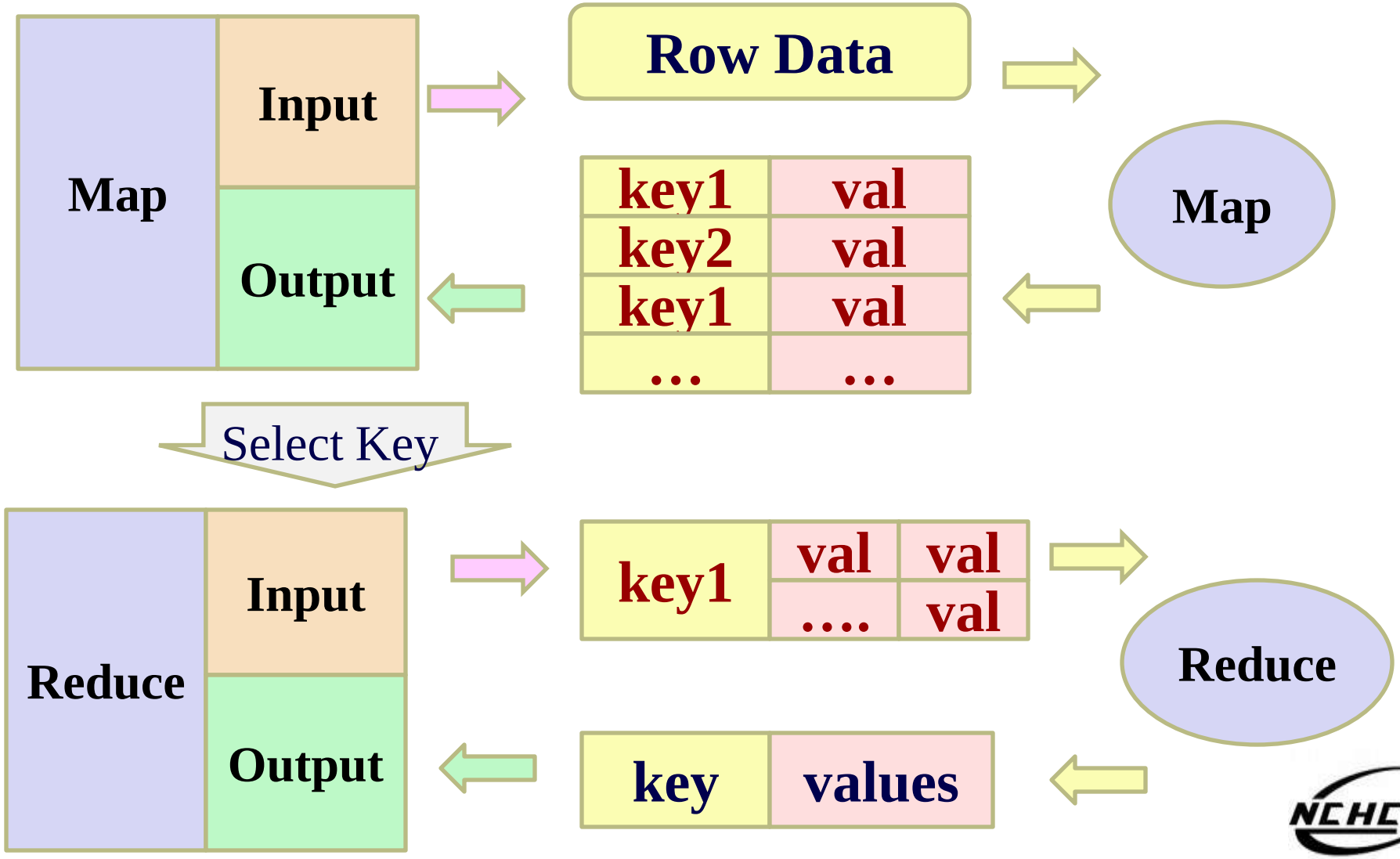
JobTracker選數個TaskTracker來作Map運算，產生些中間檔案

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

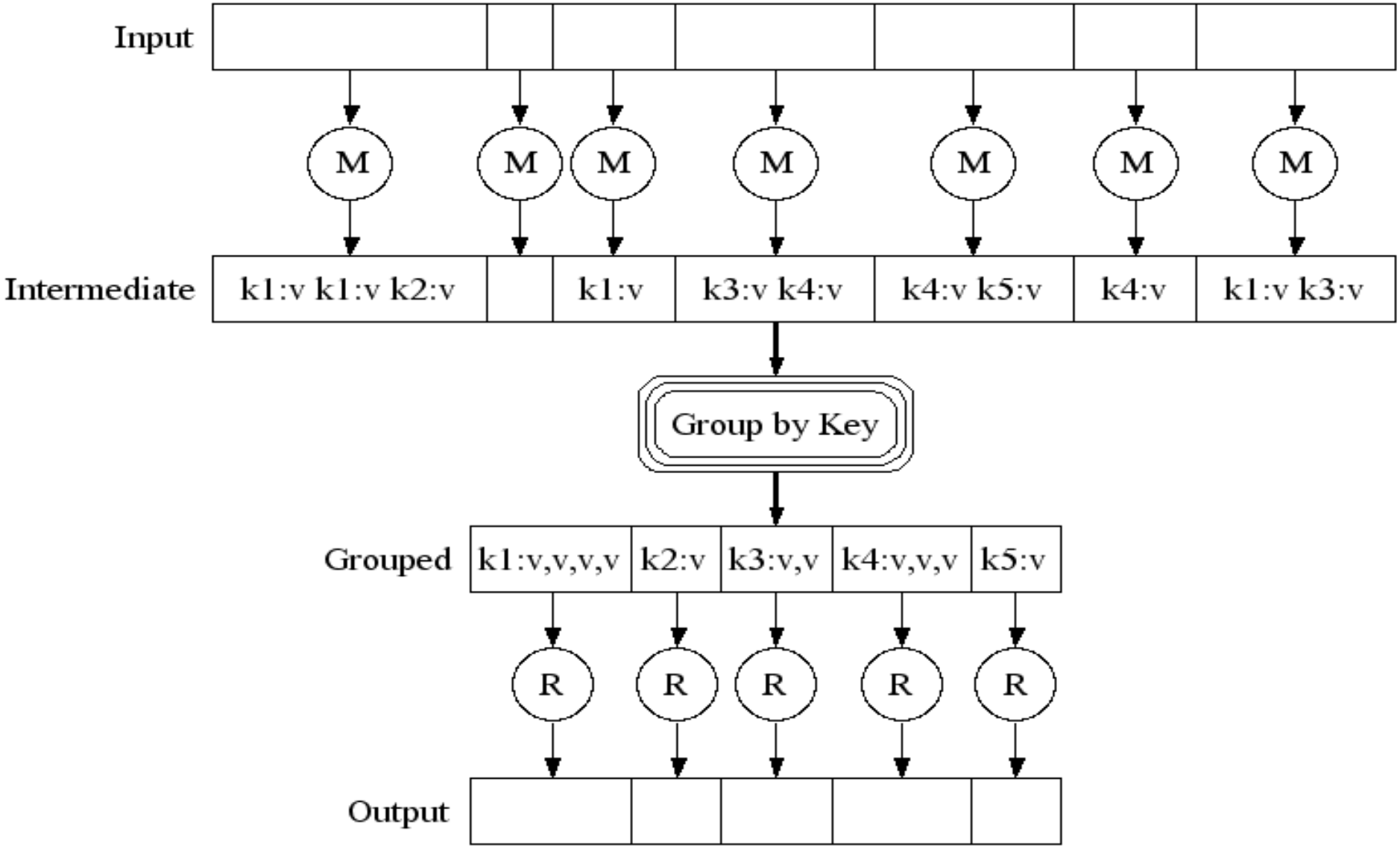
JobTracker派遣TaskTracker作reduce

reduce完後通知JobTracker與Namenode以產生output

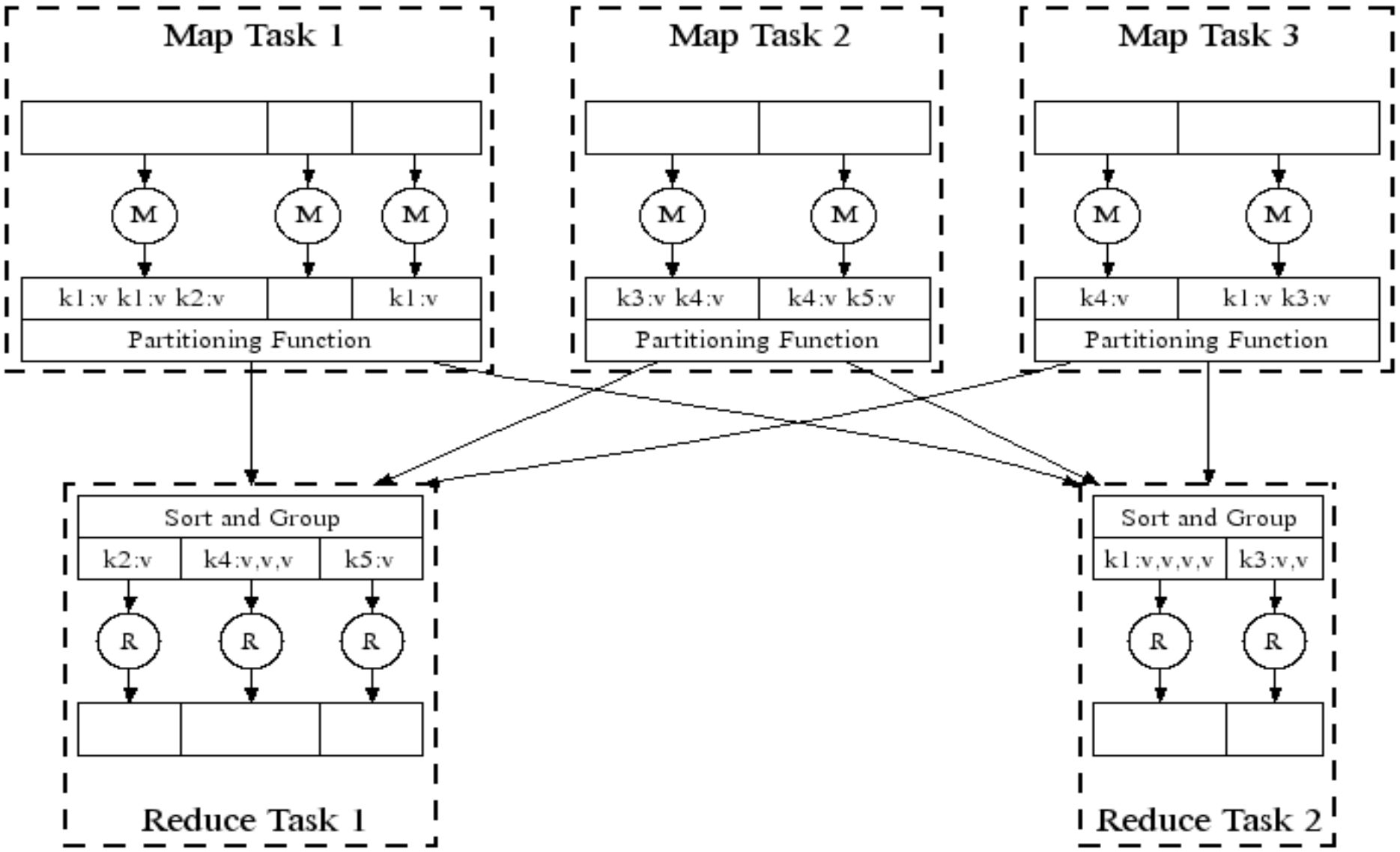
MapReduce 與 <Key, Value>



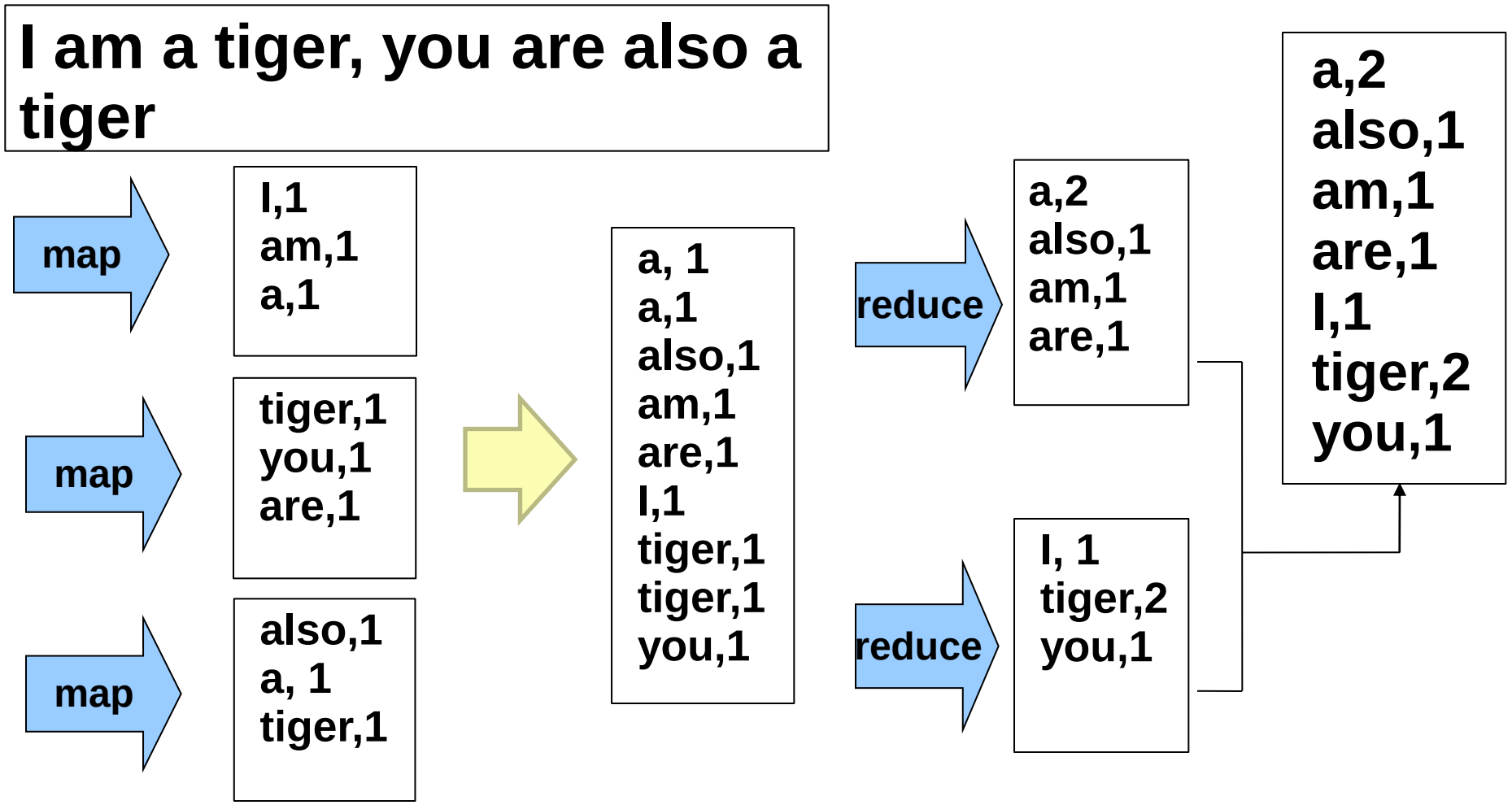
MapReduce 圖解



MapReduce in Parallel



範例



JobTracker先選了三個Tracker做map

Map結束後，hadoop進行中間資料的整理與排序

JobTracker再選兩個TaskTracker作reduce

Hadoop適用於..

- 大規模資料集
- 可拆解
  - Text tokenization
  - Indexing and Search
  - Data mining
  - machine learning
  - …

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

Hadoop Applications (1)

- Adobe
  - use Hadoop and HBase in several areas from social services to structured data storage and processing for internal use.
- Adknowledge - Ad network
  - used to build the recommender system for behavioral targeting, plus other clickstream analytics
- Alibaba
  - processing sorts of business data dumped out of database and joining them together. These data will then be fed into iSearch, our vertical search engine.
- AOL
  - We use hadoop for variety of things ranging from ETL style processing and statistics generation to running advanced algorithms for doing behavioral analysis

Hadoop Applications (2)

- Baidu - the leading Chinese language search engine
  - Hadoop used to analyze the log of search and do some mining work on web page database
- Contextweb - ADSDAQ Ad Excange
  - use Hadoop to store ad serving log and use it as a source for Ad optimizations/Analytics/reporting/machine learning.
- Detikcom - Indonesia's largest news portal
  - use hadoop, pig and hbase to analyze search log, generate Most View News,
  - generate top wordcloud, and analyze all of our logs

Hadoop Applications (3)

- DropFire
  - generate Pig Latin scripts that describe structural and semantic conversions between data contexts
  - use Hadoop to execute these scripts for production-level deployments
- Facebook
  - use Hadoop to store copies of internal log and dimension data sources
  - use it as a source for reporting/analytics and machine learning.
- Freestylers - Image retrieval engine
  - use Hadoop 影像處理
- Hosting Habitat
  - 取得所有clients的軟體資訊
  - 分析並告知clients 未安裝或未更新的軟體

Hadoop Applications (4)

- IBM
  - Blue Cloud Computing Clusters
- ICCS
  - 用 Hadoop and Nutch to crawl Blog posts 並分析之
- IIIT, Hyderabad
  - We use hadoop 資訊檢索與提取
- Journey Dynamics
  - 用 Hadoop MapReduce 分析 billions of lines of GPS data 並產生交通路線資訊.
- Krugle
  - 用 Hadoop and Nutch 建構 原始碼搜尋引擎

Hadoop Applications (5)

- SEDNS - Security Enhanced DNS Group
  - 收集全世界的 DNS 以探索網路分散式內容.
- Technical analysis and Stock Research
  - 分析股票資訊
- University of Maryland
  - 用Hadoop 執行 machine translation, language modeling, bioinformatics, email analysis, and image processing 相關研究
- University of Nebraska Lincoln, Research Computing Facility
  - 用Hadoop跑約200TB的CMS經驗分析
  - **緊湊渺子線圈**（**CMS**，**C**ompact **M**uon **S**olenoid）為瑞士歐洲核子研究組織CERN的大型強子對撞器計劃的兩大通用型粒子偵測器中的一個。

Hadoop Applications (6)

- PARC
  - Used Hadoop to analyze Wikipedia conflicts
- Search Wikia
  - A project to help develop open source social search tools
- Yahoo!
  - Used to support research for Ad Systems and Web Search
  - 使用Hadoop平台來發現發送垃圾郵件的殭屍網絡
- 趨勢科技
  - 過濾像是釣魚網站或惡意連結的網頁內容

Map Reduce Programming

王耀聰 陳威宇

Jazz@nchc.org.tw

waue@nchc.org.tw

國家高速網路與計算中心(NCHC)

Outline

- 概念
- 程式基本框架及執行步驟方法
- 範例一:
  - Hadoop 的 Hello World => Word Count
  - 說明
  - 動手做
- 範例二:
  - 進階版=> Word Count 2
  - 說明
  - 動手做

程式基本框架

Program Prototype (v 0.18)

Map 區

Reduce 區

設定區

```
Class MR{
    Class Mapper …{
    }
    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

其他的設定參數程式碼

Class Mapper

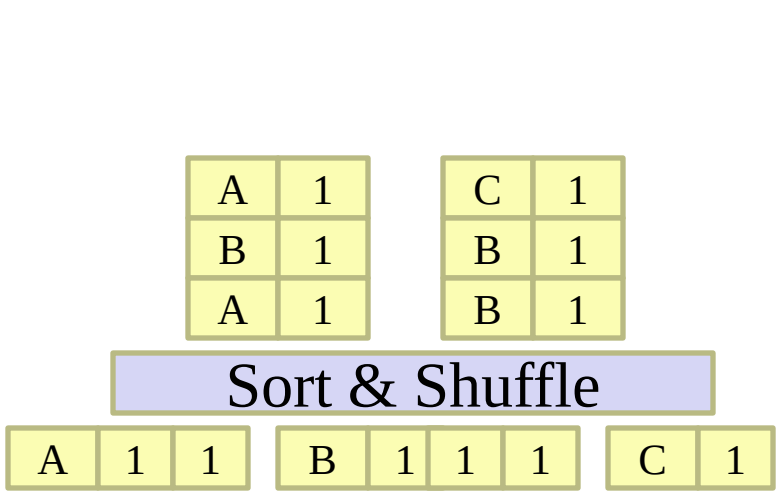
```
1  class MyMap extends MapReduceBase
   implements Mapper < INPUT KEY , INPUT VALUE , OUTPUT KEY , OUTPUT VALUE >
2  {
3  // 全域變數區
4  public void map ( INPUT KEY key, INPUT VALUE value,
           OutputCollector< OUTPUT KEY , OUTPUT VALUE > output,
           Reporter  reporter) throws IOException
5          {
6          // 區域變數與程式邏輯區
7          output.collect( NewKey, NewValue);
8          }
9  }
```

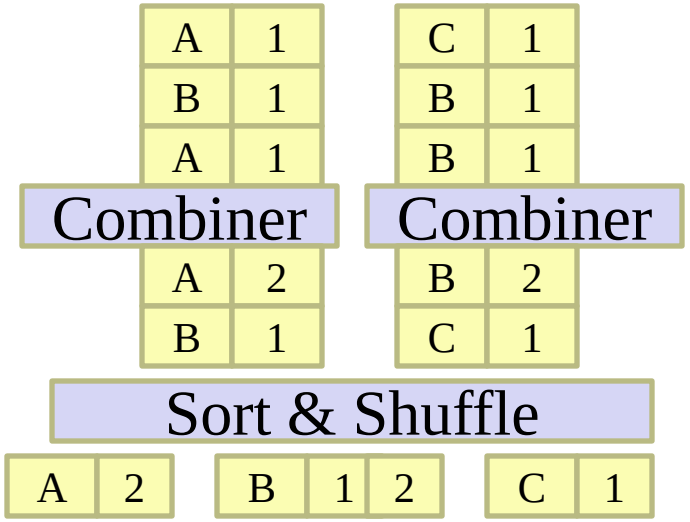
Class Reducer

```
1  class MyRed extends MapReduceBase
   implements Reducer < INPUT KEY , INPUT VALUE , OUTPUT KEY , OUTPUT VALUE >
2  {
3  // 全域變數區
4  public void reduce ( INPUT KEY key, Iterator< INPUT VALUE > values,
           OutputCollector< OUTPUT KEY , OUTPUT VALUE > output,
           Reporter  reporter) throws IOException
5          {
6          // 區域變數與程式邏輯區
7          output.collect( NewKey, NewValue);
8          }
9  }
```

Class Combiner

- 指定一個combiner，它負責對中間過程的輸出進行聚集，這會有助於降低從Mapper 到 Reducer數據傳輸量。
  - 引用Reducer
  - JobConf.setCombinerClass(Class)

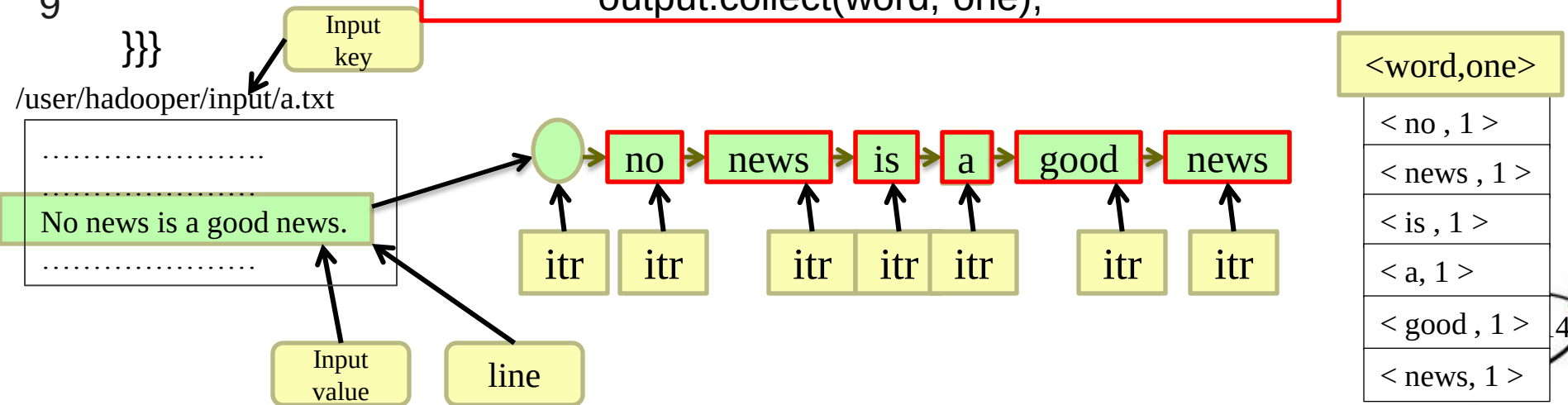




範例程式一

Word Count Sample (1)

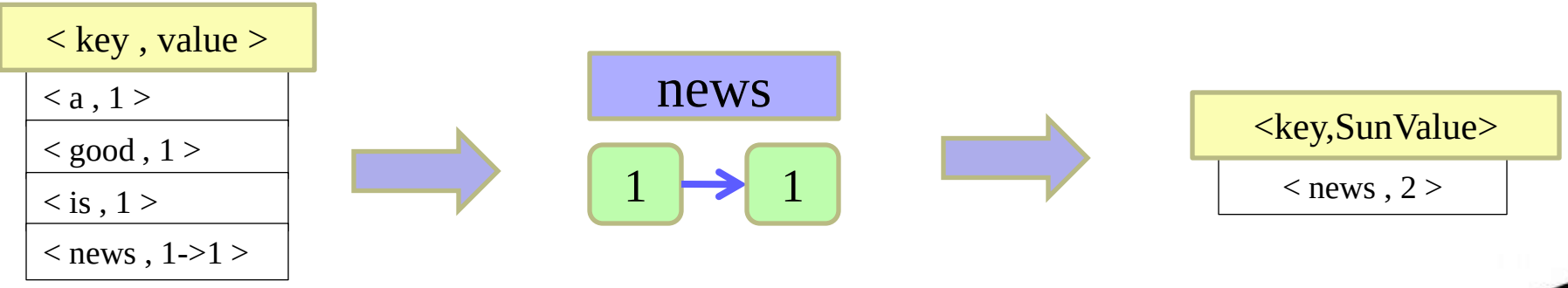
```
1   class MapClass extends MapReduceBase implements
    Mapper<LongWritable, Text, Text, IntWritable> {
2           private final static IntWritable one = new IntWritable(1);
3           private Text word = new Text();
4           public void map( LongWritable key, Text value,
              OutputCollector<Text, IntWritable> output, Reporter
    reporter) throws IOException {
5               String line = ((Text) value).toString();
6               StringTokenizer itr = new StringTokenizer(line);
7               while (itr.hasMoreTokens()) {
8                       word.set(itr.nextToken());
9                       output.collect(word, one);
    }}}
```



範例程式一

Word Count Sample (2)

```
1  class ReduceClass extends MapReduceBase implements Reducer< Text,
   IntWritable, Text, IntWritable> {
2          IntWritable SumValue = new IntWritable();
3          public void reduce( Text key, Iterator<IntWritable> values,
           OutputCollector<Text, IntWritable> output, Reporter reporter)
           throws IOException {
4                  int sum = 0;
5                  while (values.hasNext())
6                          sum += values.next().get();
7                  SumValue.set(sum);
8                  output.collect(key, SumValue);
   }}
```



範例程式一

Word Count Sample (3)

```
Class WordCount{
 main()
        JobConf conf = new JobConf(WordCount.class);
        conf.setJobName("wordcount");
        // set path
        FileInputFormat.setInputPaths(new Path(args[0]));
        FileOutputFormat.setOutputPath(new Path(args[1]));
        // set map reduce
        conf.setMapperClass(MapClass.class);
        conf.setCombinerClass(Reduce.class);
        conf.setReducerClass(ReduceClass.class);
        // run
        JobClient.runJob(conf);
}}
```

執行流程
基本步驟

編譯與執行

1. 編譯
   - javac △ -classpath △ hadoop-*-core.jar △ -d △ MyJava △ MyCode.java
2. 封裝
   - jar △ -cvf △ MyJar.jar △ -C △ MyJava △ .
3. 執行
   - bin/hadoop △ jar △ MyJar.jar △ MyCode △ HDFS_Input/ △ HDFS_Output/

- 所在的執行目錄為Hadoop_Home
- ./MyJava = 編譯後程式碼目錄
- Myjar.jar = 封裝後的編譯檔
- 先放些文件檔到HDFS上的input目錄
- ./input; ./ouput = hdfs的輸入、輸出目錄

範例一
動手做

WordCount1 練習 (I)

1. cd $HADOOP_HOME
2. bin/hadoop dfs -mkdir input
3. echo "I like NCHC Cloud Course." > inputwc/input1
4. echo "I like nchc Cloud Course, and we enjoy this crouse." > inputwc/input2
5. bin/hadoop dfs -put inputwc inputwc
6. bin/hadoop dfs -ls input

```
waue@vPro:/opt/hadoop$ bin/hadoop dfs -ls input
Found 2 items
-rw-r--r--   1 waue supergroup         26 2009-03-22 12:15 /user/waue/input/input1
-rw-r--r--   1 waue supergroup         52 2009-03-22 12:15 /user/waue/input/input2
waue@vPro:/opt/hadoop$
```

範例一
動手做

WordCount1 練習 (II)

1. 編輯WordCount.java
   http://trac.nchc.org.tw/cloud/attachment/wiki/jazz/Hadoop_Lab6/WordCount.java?format=raw
2. mkdir MyJava
3. javac -classpath hadoop-*-core.jar -d MyJava WordCount.java
4. jar -cvf wordcount.jar -C MyJava .
5. bin/hadoop jar wordcount.jar WordCount input/ output/

- 所在的執行目錄為Hadoop_Home（因為hadoop-*-core.jar）
- javac編譯時需要classpath, 但hadoop jar時不用
- wordcount. jar = 封裝後的編譯檔，但執行時需告知class name
- Hadoop進行運算時，只有 input 檔要放到hdfs上，以便hadoop分析運算；執行檔（wordcount. jar）不需上傳，也不需每個node都放，程式的載入交由java處理

範例一
動手做

WordCount1 練習(III)

```
waue@vPro:/opt/hadoop$ mkdir MyJava
waue@vPro:/opt/hadoop$ javac -classpath hadoop-*-core.jar -d MyJava WordCount.java
waue@vPro:/opt/hadoop$ jar -cvf wordcount.jar -C MyJava .
新增 manifest
新增：WordCount.class (讀=1516)(寫=740)(壓縮 51%)
新增：WordCount$Reduce.class (讀=1591)(寫=642)(壓縮 59%)
新增：WordCount$Map.class (讀=1918)(寫=795)(壓縮 58%)
waue@vPro:/opt/hadoop$ bin/hadoop jar wordcount.jar WordCount input/ output/
09/03/22 11:39:01 WARN mapred.JobClient: Use GenericOptionsParser for parsing the argu
ments. Applications should implement Tool for the same.
09/03/22 11:39:01 INFO mapred.FileInputFormat: Total input paths to process : 1
09/03/22 11:39:01 INFO mapred.FileInputFormat: Total input paths to process : 1
09/03/22 11:39:02 INFO mapred.JobClient: Running job: job_200903201526_0007
09/03/22 11:39:03 INFO mapred.JobClient:  map 0% reduce 0%
09/03/22 11:39:08 INFO mapred.JobClient:  map 100% reduce 0%
09/03/22 11:39:15 INFO mapred.JobClient: Job complete: job_200903201526_0007
09/03/22 11:39:15 INFO mapred.JobClient: Counters: 16
09/03/22 11:39:15 INFO mapred.JobClient:   File Systems
09/03/22 11:39:15 INFO mapred.JobClient:     HDFS bytes read=320950
09/03/22 11:39:15 INFO mapred.JobClient:     HDFS bytes written=130568
09/03/22 11:39:15 INFO mapred.JobClient:     Local bytes read=168448
09/03/22 11:39:15 INFO mapred.JobClient:     Local bytes written=336932
09/03/22 11:39:15 INFO mapred.JobClient:   Job Counters
09/03/22 11:39:15 INFO mapred.JobClient:     Launched reduce tasks=1
```

範例一
動手做

WordCount1 練習(IV)

```
waue@vPro:/opt/hadoop$ bin/hadoop dfs -cat output/part-00000
Cloud   2
Course, 1
Course. 1
I       2
NCHC    1
and     1
course. 1
enjoy   1
like    2
nchc    1
this    1
we      1
```

範例二
動手做

WordCount 進階版

- WordCount2

http://trac.nchc.org.tw/cloud/attachment/wiki/jazz/Hadoop_Lab6/WordCount2.java?format=raw

- 功能
  - 不計標點符號
  - 不管大小寫
- 步驟 (接續 WordCount 的環境)
  1. echo "\." >pattern.txt && echo "\," >>pattern.txt
  2. bin/hadoop dfs -put pattern.txt ./
  3. mkdir MyJava2
  4. javac -classpath hadoop-*-core.jar -d MyJava2 WordCount2.java
  5. jar -cvf wordcount2.jar -C MyJava2 .

範例二
動手做

不計標點符號

- 執行
  - bin/hadoop jar wordcount2.jar WordCount2 input output2 -skip pattern.txt dfs -cat output2/part-00000

```
waue@vPro:/opt/hadoop$ bin/hadoop dfs -cat output2/part-00000
Cloud   2
Course  2
I       2
NCHC    1
and     1
course  1
enjoy   1
like    2
nchc    1
this    1
we      1
```

範例二
動手做

不管大小寫

- 執行
  - bin/hadoop jar wordcount2.jar WordCount2 -Dwordcount.case.sensitive=false input output3 -skip pattern.txt

```
waue@vPro:/opt/hadoop$ bin/hadoop dfs -cat output3/part-00000
and     1
cloud   2
course  3
enjoy   1
i       2
like    2
nchc    2
this    1
we      1
```

範例二
補充說明

Tool

- 處理Hadoop命令執行的選項

  -conf <configuration file>

  -D <property=value>

  -fs <local|namenode:port>

  -jt <local|jobtracker:port>

- 透過介面交由程式處理
  - ToolRunner.run(Tool, String[])

範例二
補充說明

DistributedCache

- 設定特定有應用到相關的、超大檔案、或只用來參考卻不加入到分析目錄的檔案
  - 如pattern.txt檔
- DistributedCache.addCacheFile(URI,conf)
  - URI＝hdfs://host:port/FilePath

非JAVA不可？

Options without Java

- 雖然Hadoop框架是用Java實作，但Map/Reduce應用程序則不一定要用 Java來寫
- Hadoop Streaming ：
  - 執行作業的工具，使用者可以用其他語言（如：PHP）套用到Hadoop的mapper和reducer
- Hadoop Pipes：C++ API

Nutch 簡介

王耀聰 陳威宇

Jazz@nchc.org.tw

waue@nchc.org.tw

國家高速網路與計算中心(NCHC)

Outline

- What is Nutch
- Why Nutch
- Nutch’s Details
- Let’s go

What's Nutch

- Nutch是一個open source，以Java來實做的搜索引擎，它提供了架設自己的搜索引擎所需的全部工具。
- 利用Lucene為函式庫
- **架構於Hadoop之上**

Nutch's goals

- 每個月抓取幾十億網頁
- 為這些網頁維護索引
- 對索引文件進行每秒上千次的搜索
- 提供高質量的搜索結果
- 以最小的成本運作

Why Nutch ?

- 透明
  - Opensource，資訊不隱藏
- 擴充
  - 有各種函式庫應用於分析不同檔案
- 隱私
  - 可應用於搜尋專屬資料
- 客製化
  - 可以之為基礎設計自己的data mining 工具

Who use Nutch

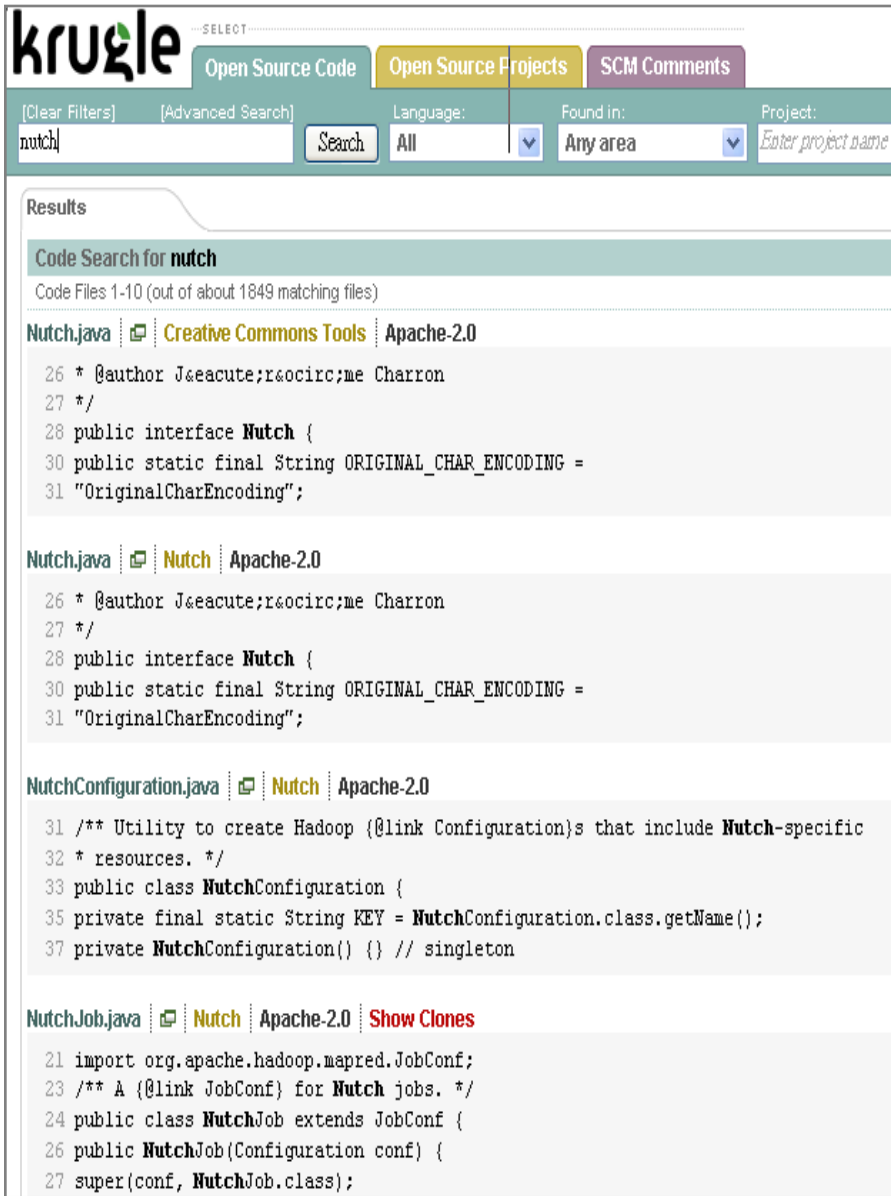
Public search engines using Nutch

Please sort by name alphabetically

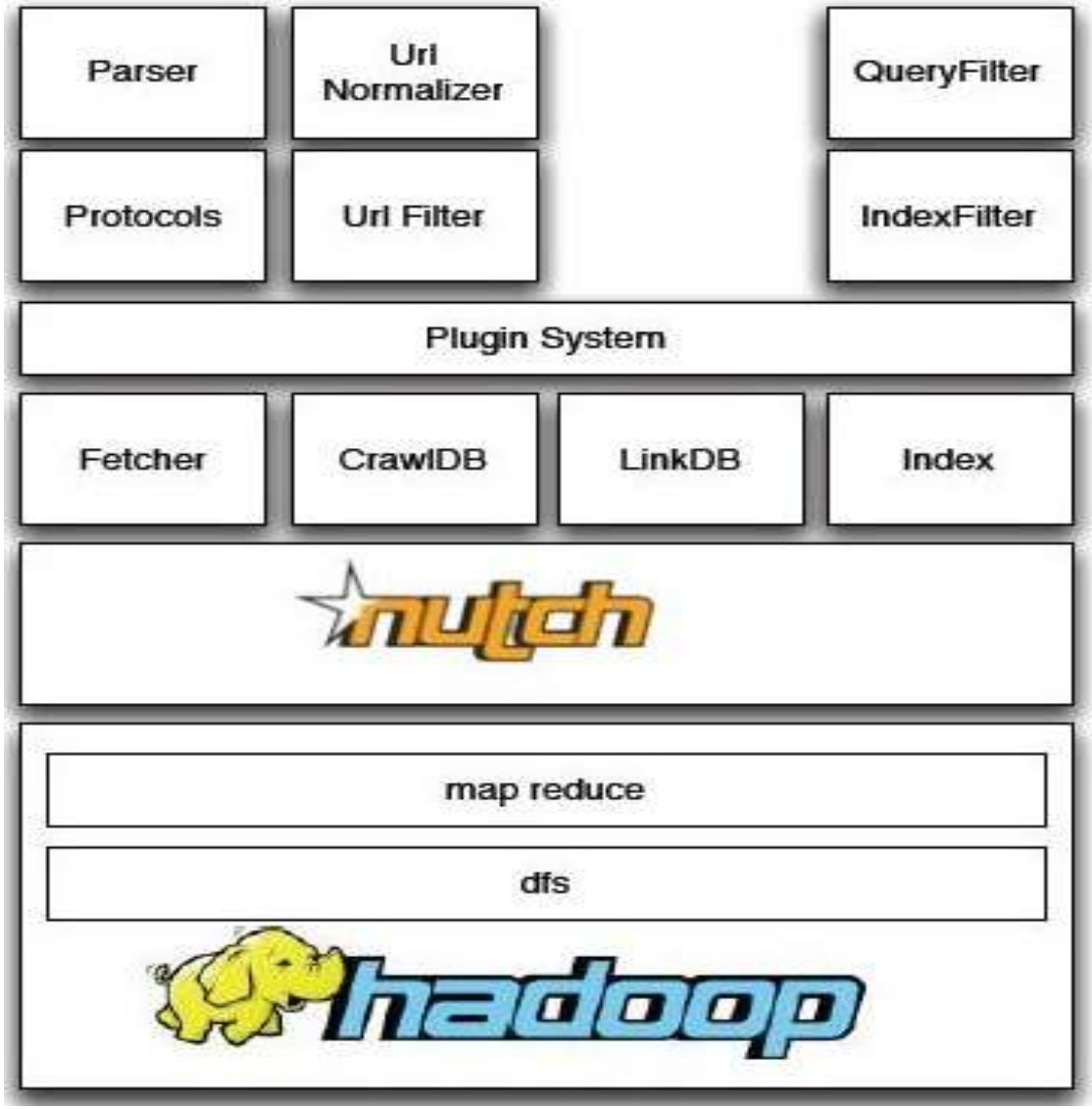
- AskAboutOil is a vertical search portal for the petroleum indust
- Baynote provides free hosted Nutch search for businesses.
- BeThere BeSquare is an Event Search Engine for the San Fra category and get details about events in 4 different views.
- Bigsearch.ca uses nutch open source software to deliver its se
- BusyTonight: Search for any event in the United States, by key from original source Web sites.
- Central Budapest Search is a search engine for English langua events.
- Circuit Scout is a search engine for electrical circuits.
- Comtec Search is a search engine for UK Tour Operator Pack
- Coder-Suche.de searchs for coding stuff like apis, documenta english.
- Cornell University Library is collaborating with the research grc pages based on Nutch. The nutch-based search engine is near tl
- Creative Commons is a search engine for creative commons l
- Dadi360 Usee nutch search engine for providing search of Ch
- Ecolhub Web Search an E. coli specific search engine based thereby reducing the number of spurious hits. Searches can be o More resources getting added.
- Epivista is a search engine of epilepsy related web sites.
- eroscanner is a search engine for german adult stuff. Watching **NSFW**)

…..more

(http://wiki.apache.org/nutch/PublicServers)



架構

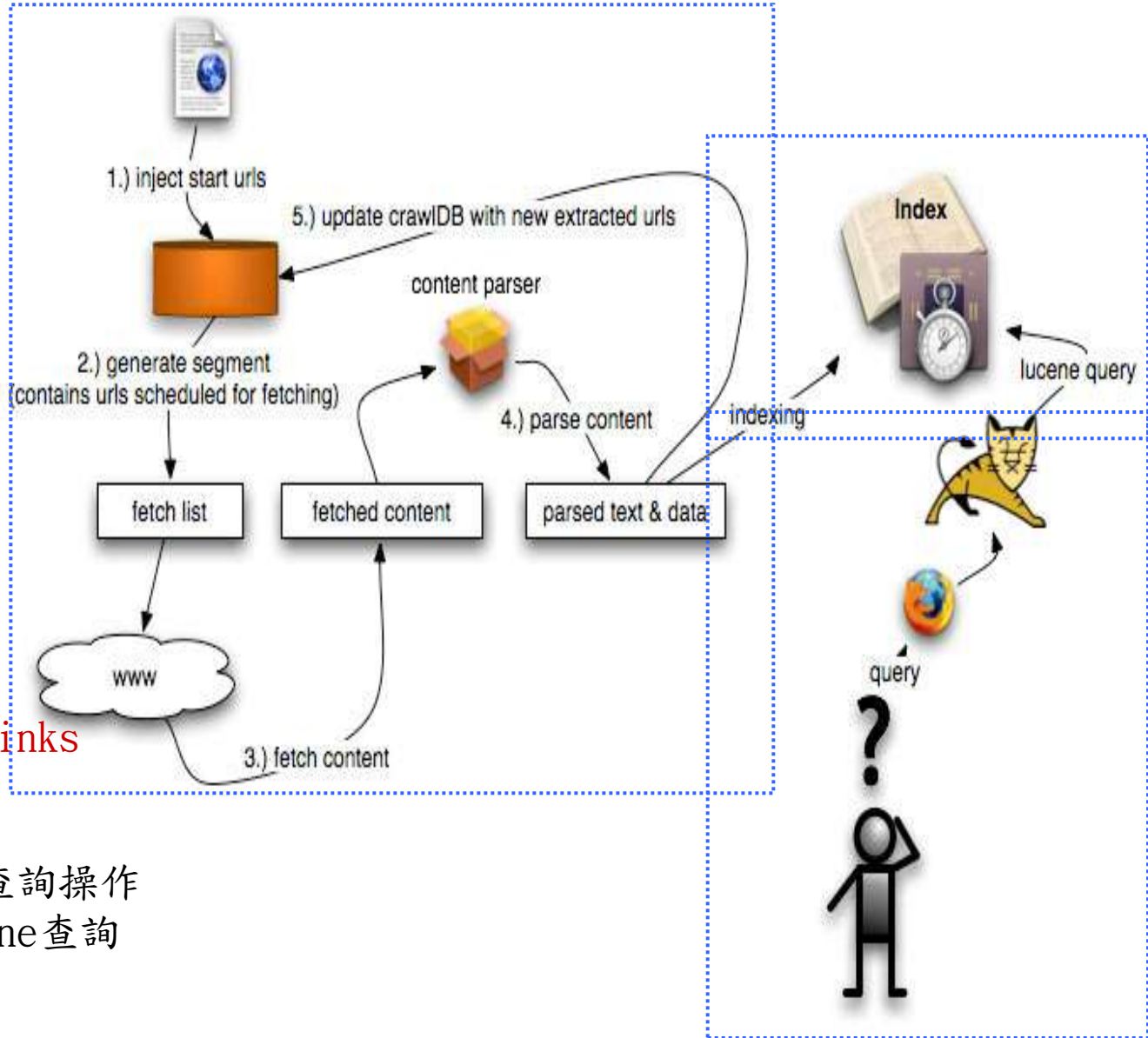

Parser
Url Normalizer
QueryFilter
Protocols
Url Filter
IndexFilter
Plugin System
Fetcher
CrawlDB
LinkDB
Index
nutch
map reduce
dfs
hadoop

運作流程

1）建立初始URL集
2）將URL集注入crawldb---inject
3）根據crawldb建立抓取清單---generate
4）執行抓取，獲取網頁內容---fetch
5)用獲取到的頁面資訊更新crawlDB---updatedb
6）重複進行3～5的步驟，直到預先設定的抓取深度

7）更新linkdb ---invertlinks
8）建立索引---index

9）用戶通過用戶接口進行查詢操作
10）將用戶查詢轉化為lucene查詢
11）返回結果



Plugin

- 修改 conf/nutch-site.xml的plugin.includes屬性
- 在nutch基本功能之上擴充其功能
  - “parse-xx”：加入解析xx檔案類型的能力
  - “protocol -xx”：加入在此協定內的檔案也處理

parse-text
parse-ext
parse-html
parse-js
parse-mp3
parse-zip
parse-rtf
parse-msword
parse-msexcel
parse-pdf
parse-rss
parse-oo
parse-swf
parse-mspowerpoint
protocol-file
protocol-ftp
protocol-http
protocol-httpclient

http://wiki.apache.org/nutch/PluginCentral

International

- 已有多國語言版可選，但若還要客製化…
- the page header
  - src/web/include/*language*/header.xml
- the "about" page
  - src/web/pages/*lang*/about.xml
- the "search" page
  - src/web/pages/*lang*/search.xml
- the "help" page
  - src/web/pages/*lang*/help.xml
- text for search results
  - src/web/locale/org/nutch/jsp/search_*lang*.properties

No！ Nutch

- 告訴網頁機器人是否允許進入爬網
- 將robots.txt放在web上
- robots.txt

```
User-agent: Nutch
Disallow: /
```

Home Page


nutch
About
FAQ
Search
help
nutch
powered by
ca | de | en | es | fi | fr | hu | it | jp | ms | nl | pl | pt | sh | sr | sv | th | zh

Start

- **23 March 2009 - Apache Nutch 1.0 Released**

Let's Go

References..

- Nutch Website
  - http://lucene.apache.org/nutch/
- Nutch wiki
  - http://wiki.apache.org/nutch/
- Nutch API
  - http://lucene.apache.org/nutch/apidocs-1.0/index.html

設置Hadoop環境

王耀聰 陳威宇

Jazz@nchc.org.tw

waue@nchc.org.tw

國家高速網路與計算中心(NCHC)

Yahoo's Hadoop Cluster

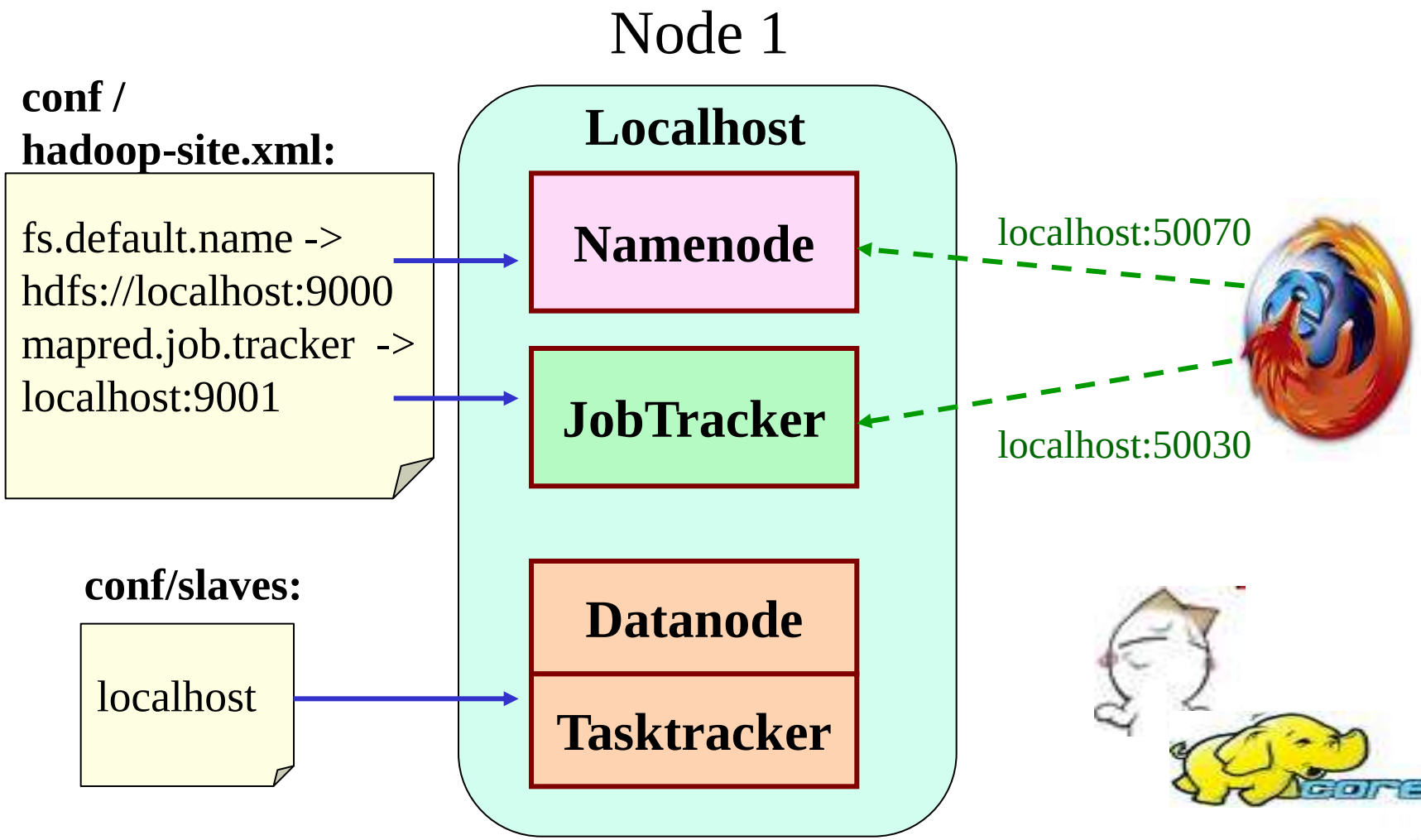
- ~10,000 machines running Hadoop in US
- The largest cluster is currently 2000 nodes
- Nearly 1 petabyte of user data (compressed, unreplicated)
- Running roughly 10,000 research jobs / week

NCHC

Hadoop 單機設定與啟動

- step 1. 設定登入免密碼
- step 2. 安裝java
- step 3. 下載安裝Hadoop
- step 4.1 設定 hadoop-env.sh
  - export JAVA_HOME=/usr/lib/jvm/java-6-sun
- step 4.2 設定 hadoop-site.xml
  - 設定Namenode-> hdfs://localhost:9000
  - 設定Jobtracker -> localhost:9001
- step 5.1 格式化HDFS
  - bin/hadoop namenode -format
- step 5.2 啟動Hadoop
  - bin/start-all.sh
- step 6. 完成！檢查運作狀態
  - Job admin http://localhost:50030/ HDFS http://localhost:50070/

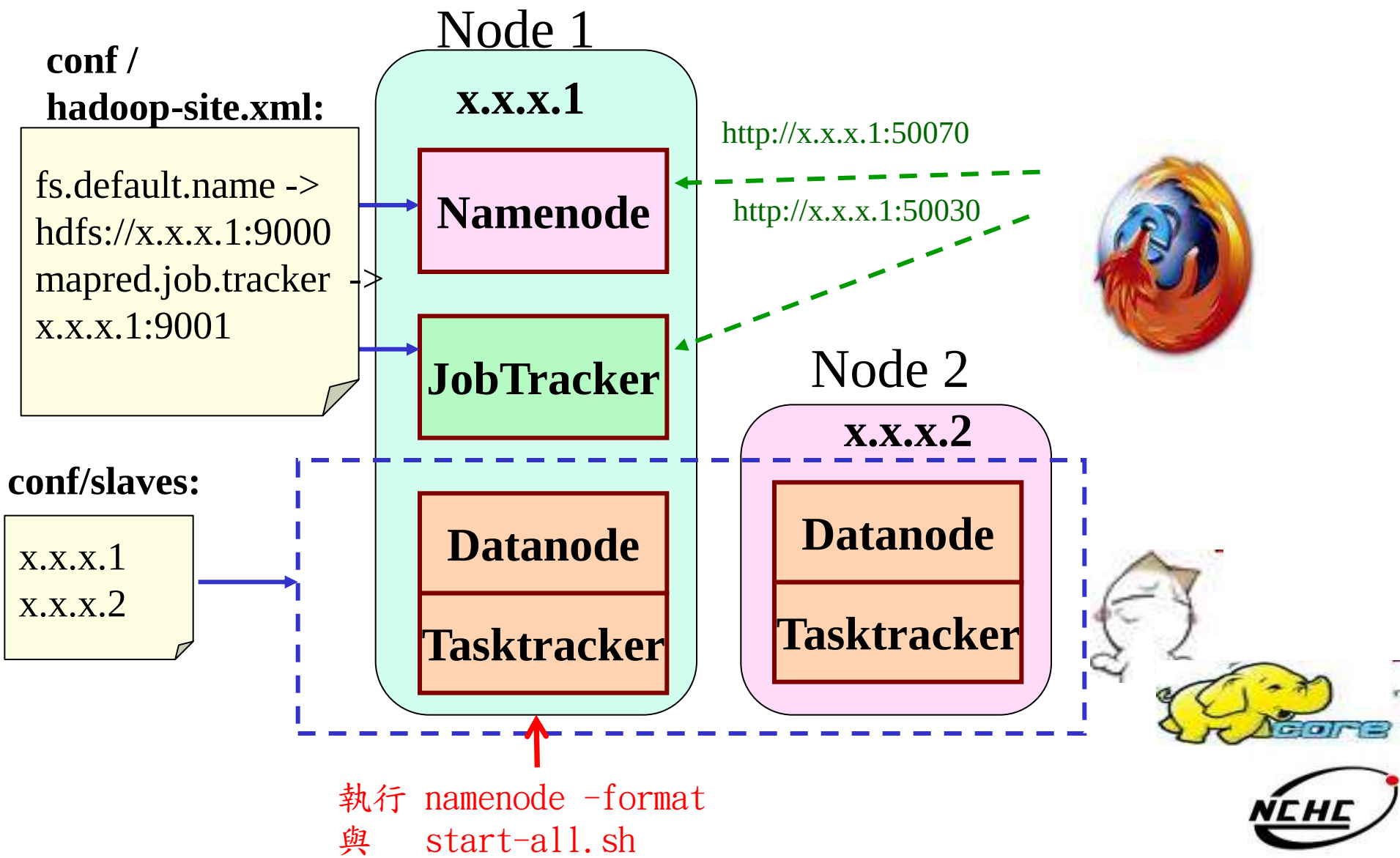
Hadoop 單機環境示意圖



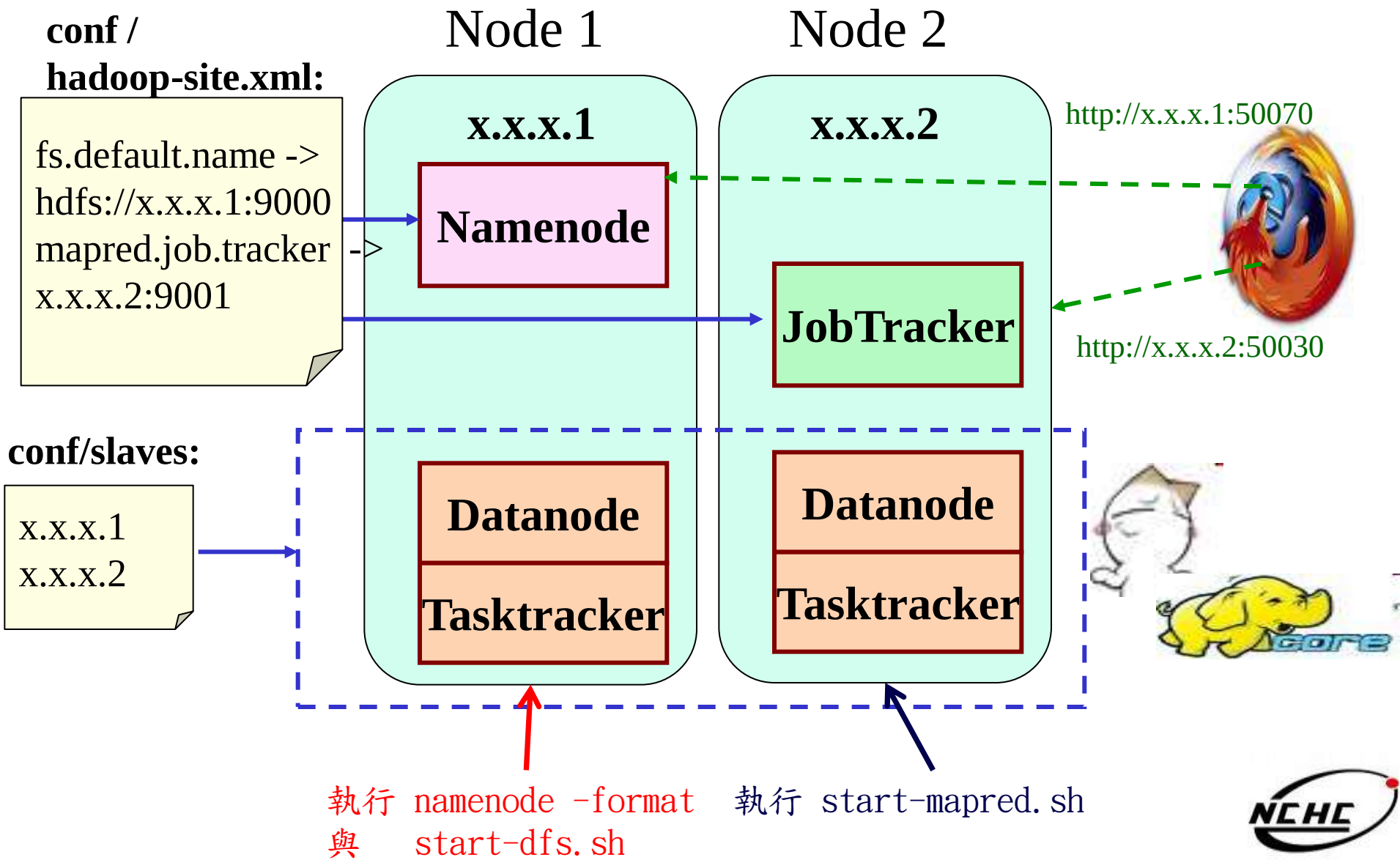
Hadoop 叢集設定與啟動

- step 1. 設定登入免密碼
- step 2. 安裝java
- step 3. 下載安裝Hadoop
- step 4.1 設定 hadoop-env.sh
  - export JAVA_HOME=/usr/lib/jvm/java-6-sun
- step 4.2 設定 hadoop-site.xml
  - 設定Namenode-> hdfs://x.x.x.1:9000
  - 設定Jobtracker -> x.x.x.2:9001
- step 4.3 設定slaves 檔
- step 4.4 將叢集內的電腦Hadoop都做一樣的配置
- step 5.1 格式化HDFS
  - bin/hadoop namenode -format
- step 5.2 啟動Hadoop
  - nodeN執行： bin/start-dfs.sh ； nodeJ執行： bin/start-mapred.sh
- step 6. 完成！檢查運作狀態
  - Job admin http://x.x.x.2:50030/ HDFS http://x.x.x.1:50070/

情況一


Node 1
x.x.x.1
conf / hadoop-site.xml:
fs.default.name -> hdfs://x.x.x.1:9000
mapred.job.tracker -> x.x.x.1:9001
Namenode
JobTracker
http://x.x.x.1:50070
http://x.x.x.1:50030
Node 2
x.x.x.2
conf/slaves:
x.x.x.1
x.x.x.2
Datanode
Tasktracker
Datanode
Tasktracker
執行 namenode -format
與 start-all.sh

情況二


conf / hadoop-site.xml:
fs.default.name -> hdfs://x.x.x.1:9000
mapred.job.tracker -> x.x.x.2:9001
Node 1
Node 2
x.x.x.1
x.x.x.2
Namenode
JobTracker
http://x.x.x.1:50070
http://x.x.x.2:50030
conf/slaves:
x.x.x.1
x.x.x.2
Datanode
Tasktracker
Datanode
Tasktracker
執行 namenode -format 與 start-dfs.sh
執行 start-mapred.sh

情況三

conf / hadoop.site.xml:

fs-default.name -> hdfs://x.x.x.1:9000
mapred.job.tracker -> x.x.x.1:9001

Node 1

x.x.x.1

Namenode

JobTracker

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

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

conf/slaves:

x.x.x.2
…..
x.x.x.n

Node 2

x.x.x.2

Datanode

Tasktracker

…

Node N

x.x.x.n

Datanode

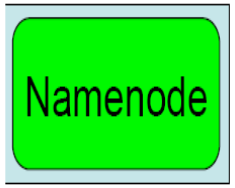
Tasktracker

NCHC

情況四

conf /
hadoop-site.xml:

mapred.job.tracker->
x.x.x.2:9001
fs.default.name ->
hdfs://x.x.x.1:9000



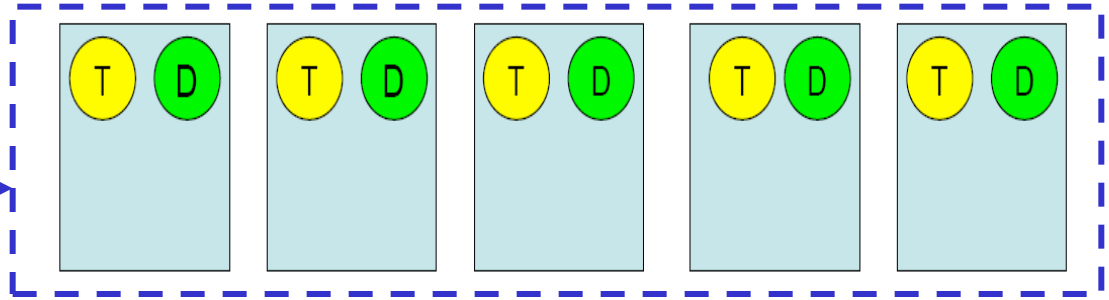
http://x.x.x.2:50030

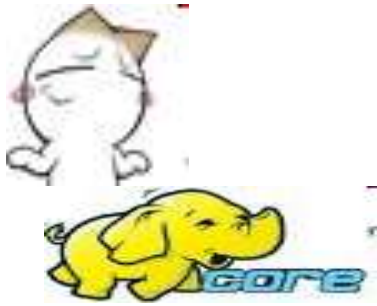
HTTP Monitoring UI

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

conf/slaves:

x.x.x.3
…….
x.x.x.n





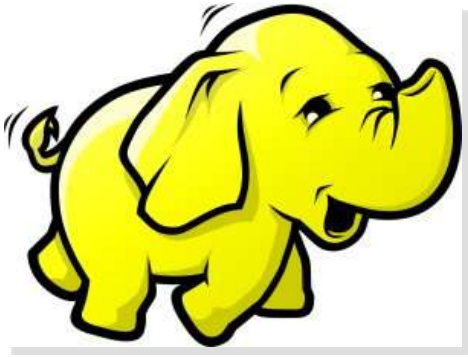
NCHC

當企鵝龍遇上小飛象
DRBL-Hadoop

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/136-3[illegible].JPG

Agenda

PART 1 :

What is Cluster Computing ?

How to deploy PC cluster ?

PART 2 :

What is DRBL and Clonezilla ?

Can DRBL help to deploy Hadoop ?

PART 3 :

Live Demo of DRBL Live and Clonezilla Live

NCHC

PART 1 :

PC Cluster 101

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

Powered by DRBL

NCHC

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

It'd better be

2^n

Manage

Scheduler

NCHC

Then, We connect 5 PCs with Gigabit Ethernet Switch

GiE Switch

10/100/1000 MBps

WAN

Add 1 NIC for WAN

NCHC

Compute Nodes

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

LAN Switch

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

NCHC

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

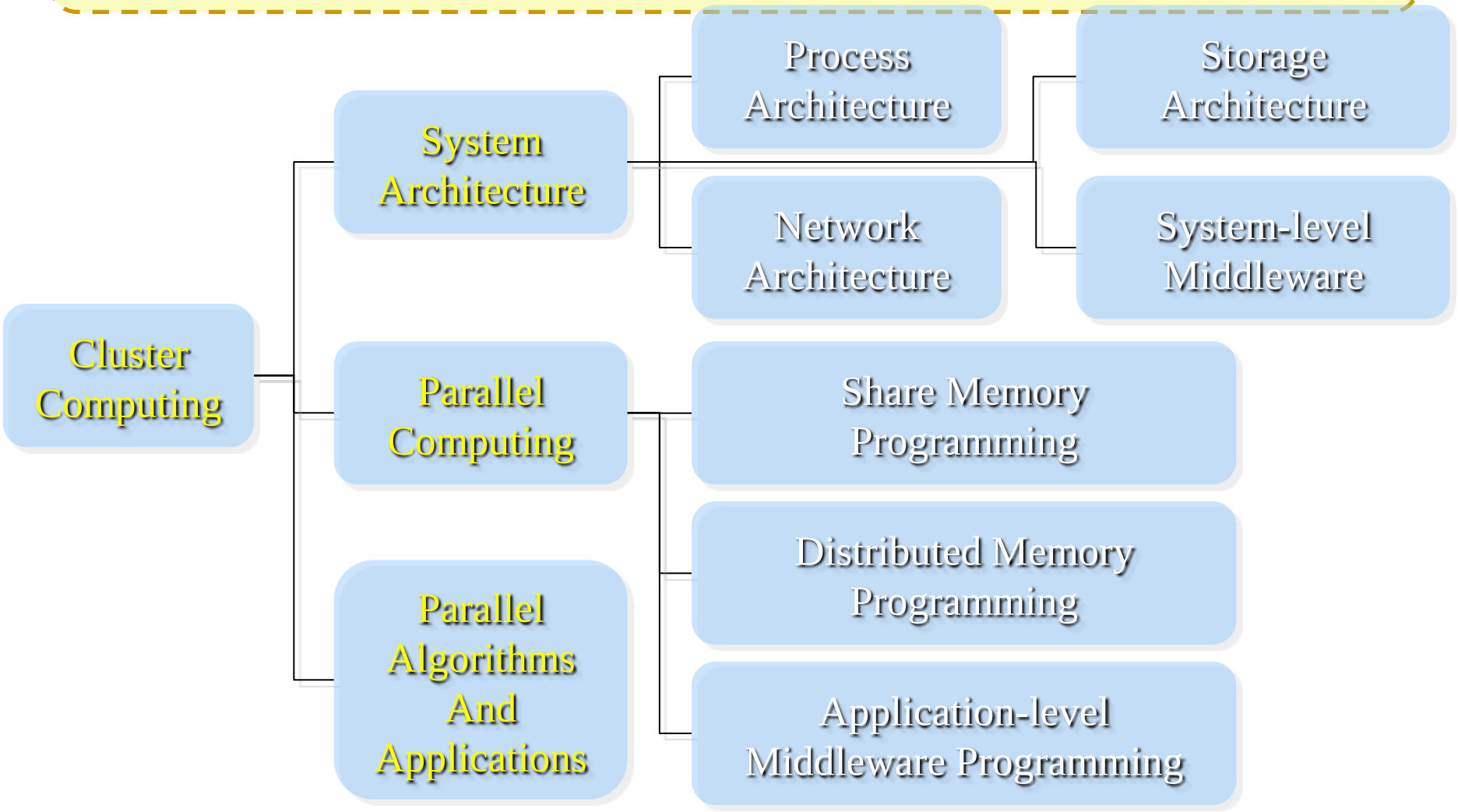
Kernel Module

Linux Kernel

Boot Loader

NCHC

Research topics about PC Cluster



Ref: Cluster Computing in the Classroom: Topics, Guidelines, and Experiences

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**

NCHC

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

NCHC

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

NCHC

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 :

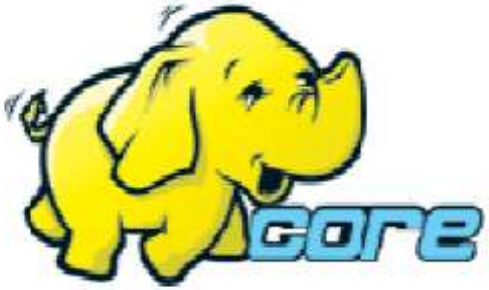
Hadoop Deployment Tool

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

Powered by DRBL

NCHC

SmartFrog



- 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

LABS hp

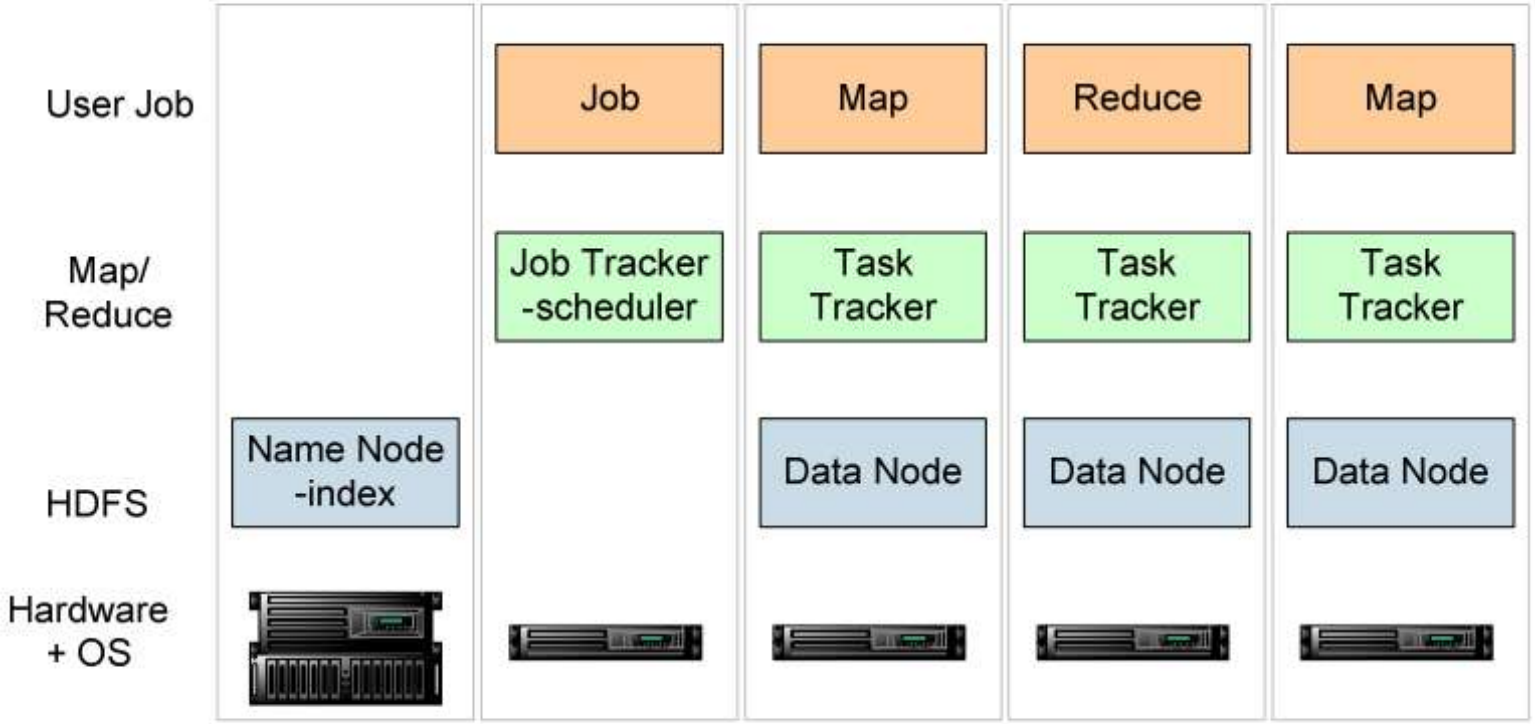
SmartFrog - HPLabs' CM tool

- 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

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

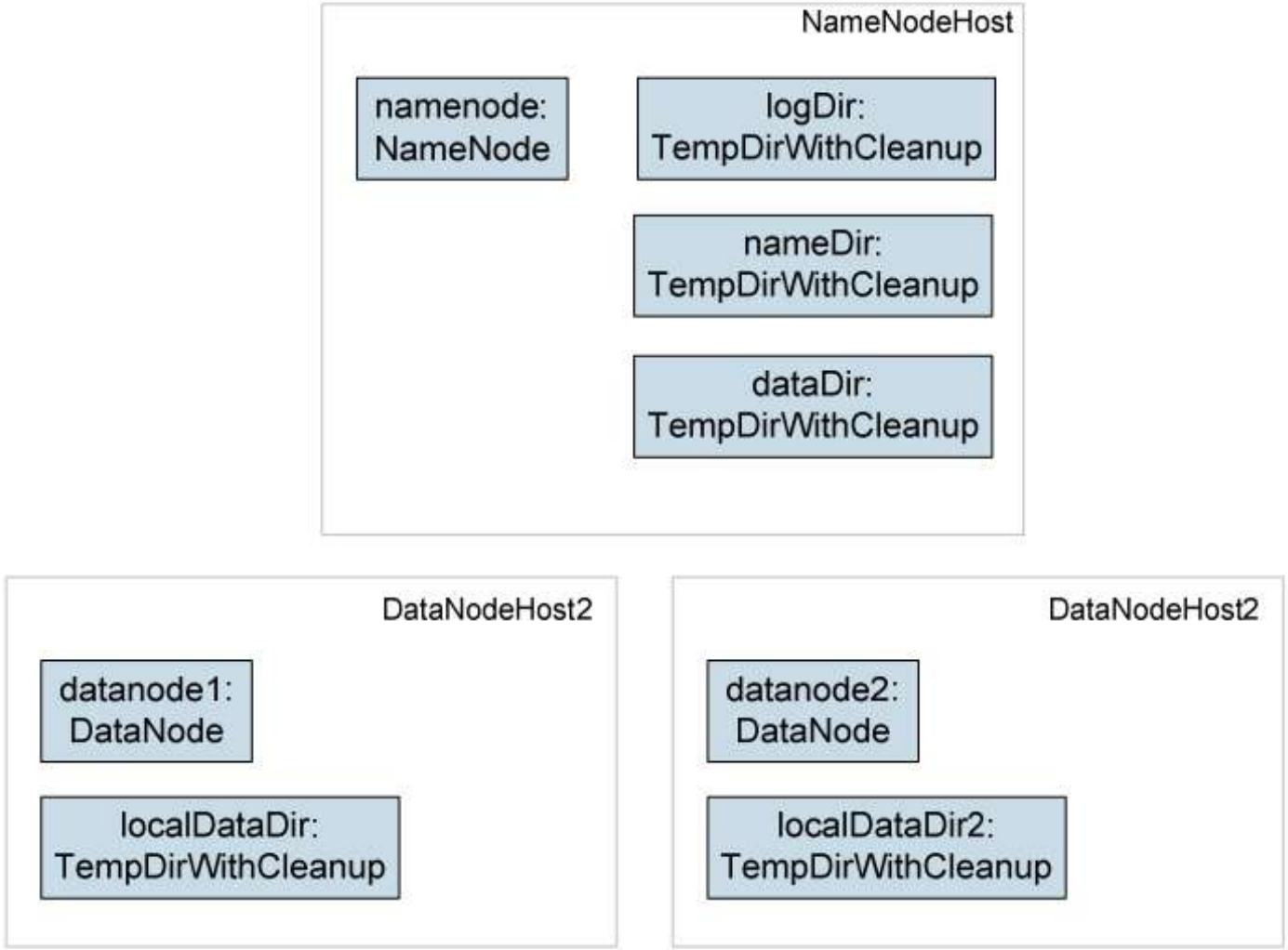
Model the system in the SmartFrog language

```
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

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 :

企鵝龍與再生龍

Jazz Wang

Yao-Tsung Wang

jazz@nchc.org.tw

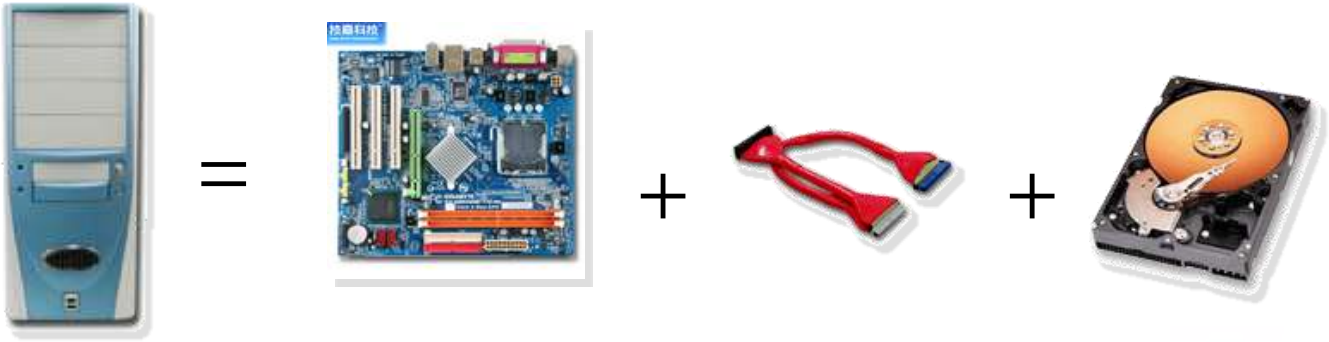
Powered by DRBL

NCHC

何謂企鵝龍DRBL ??

- **Diskless Remote Boot in Linux**
- 網路是便宜的，人的時間才是昂貴的。
- 企鵝龍簡單來說就是.....
  - 用網路線取代硬碟排線
  - 所有學生的電腦都透過網路連接到一台伺服器主機

Diskfull PC



Diskless PC

Server

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

何謂再生龍Clonezilla ??

- **Clone (複製) + zilla = Clonezilla (再生龍)**
- **裸機備分還原工具**
- **Norton Ghost 的自由軟體版替代方案**

Disk to Disk

Disk to Image

Image to N Disks

降低資訊教育管理成本

需要「化繁為簡」的解決方案！

一般國內小學的電腦教室

☑人力、時間成本高

教師1人維護管理多組設備

教學同時分派或收集作業

☑設備維護成本高

需分別處理設定(每班約40台)

如：電腦中毒、環境設定

系統操作問題、開關機、

備份還原等

平衡商業軟體與知識教育

知識和軟體都需要讓孩子「帶著走」！

☑商業軟體授權高成本

在校學習，也需回家複習

學校每台(平均) 2萬

學生家用(平均) 4萬

☑知識與法治的學習

教育知識，也需教育尊重

尊重智財權觀念

國網中心自由軟體開發

多元化資訊教學的新選擇！

以個人叢集電腦(PC Cluster)經驗發展
DRBL&Clonezilla

企鵝龍DRBL

(Diskless Remote Boot in Linux)

適合將整個電腦教室轉換成純自由軟體環境

再生龍Clonezilla

適用完整系統備份、裸機還原或災難復原

是自由！不是免費

分送、修改、存取、使用軟體的自由。
免費是附加價值。

企鵝龍DRBL與再生龍Clonezilla

電腦教室管理的新利器！

■以每班40台電腦為估算單位

DRBL&Clonezilla	未使用	使　用
管理簡化	分別管理40台	管理 1台 伺服器
硬體設備成本	每台都需配備周邊硬體	伺服器控制，節約每台學生機之周邊硬體
軟體授權成本	40台:3000*40=120,000 (MS Windows授權1台電腦之授權費NT$3,000)	軟體授權 NT$0
合法複製、分享	需負擔授權費	複製合法 NT$0
多元化電腦教學	不同系統無法並存	Linux 與M$ Windows可並存

高速計算研究資料儲存備援

教育單位採用DRBL

降低管理維護成本

帶動自由軟體使用

節省龐大軟體授權費

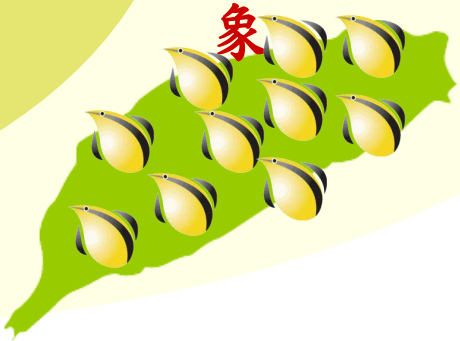
NT. 98,595,000 元

以某商業獨家軟體每機3000元授權費計，
每班35台電腦(3000*35*939)

擴至全國各單位

降低台灣盜版率

提升台灣形象



PART 1-3 :

企鵝龍的開機原理

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

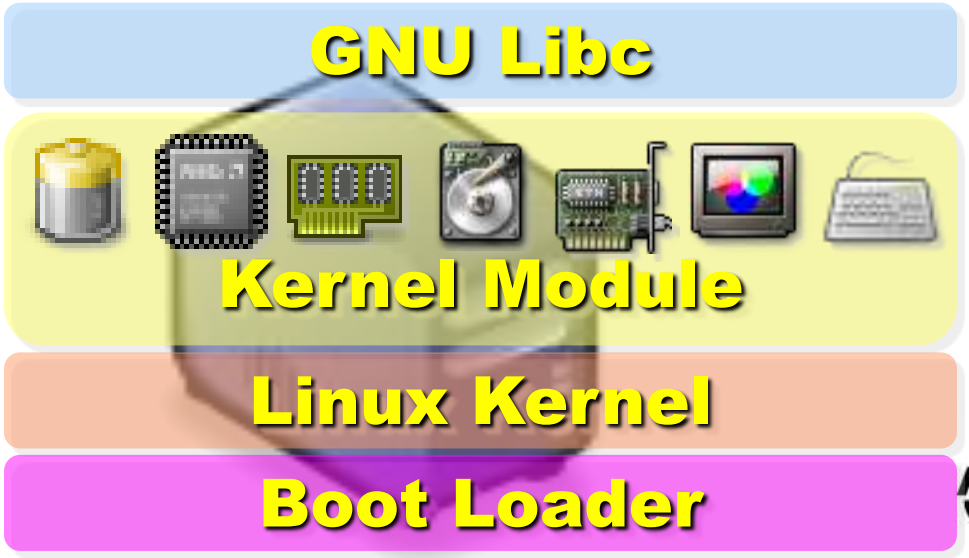
Powered by DRBL

NCHC

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

You can choose:

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

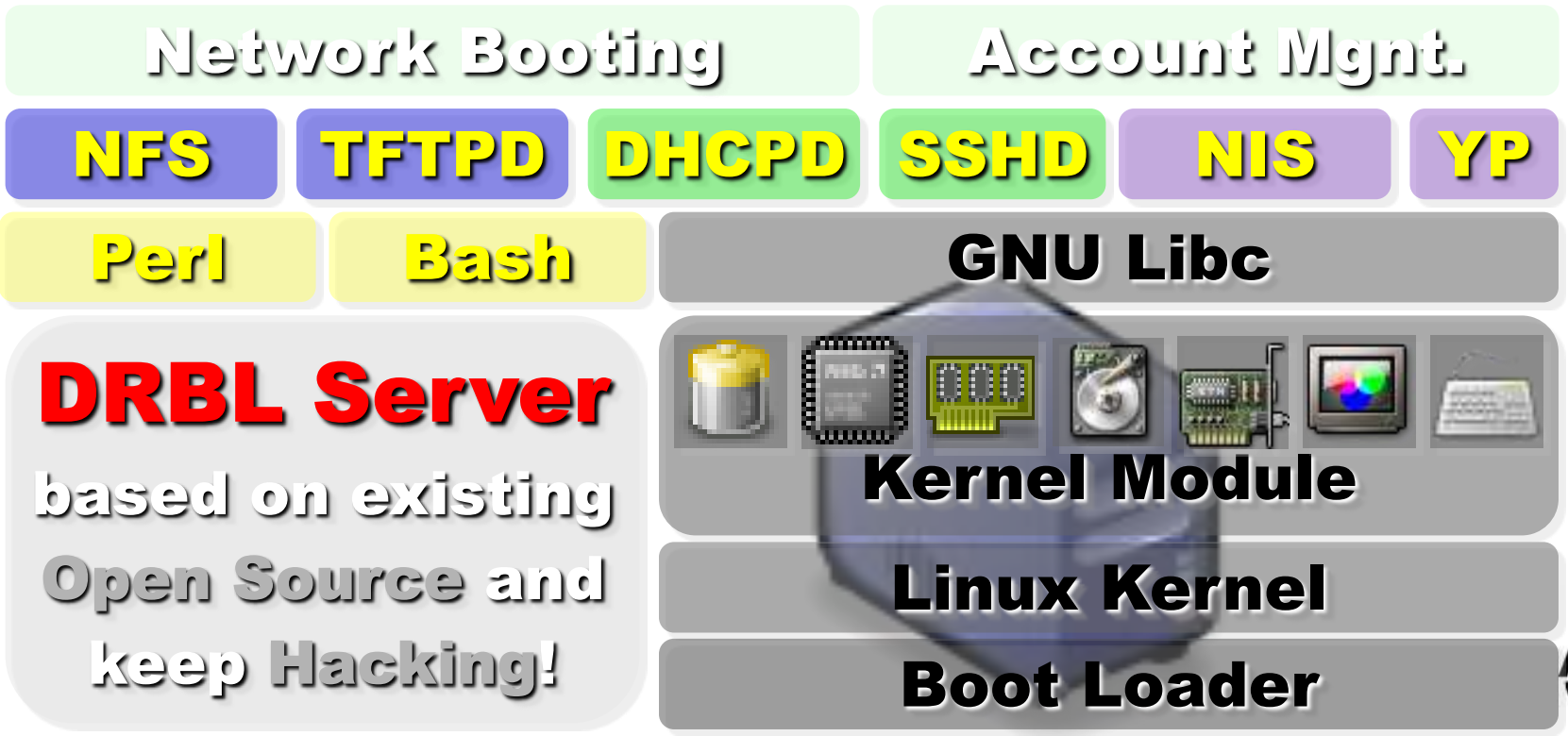


NCHC

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 ...

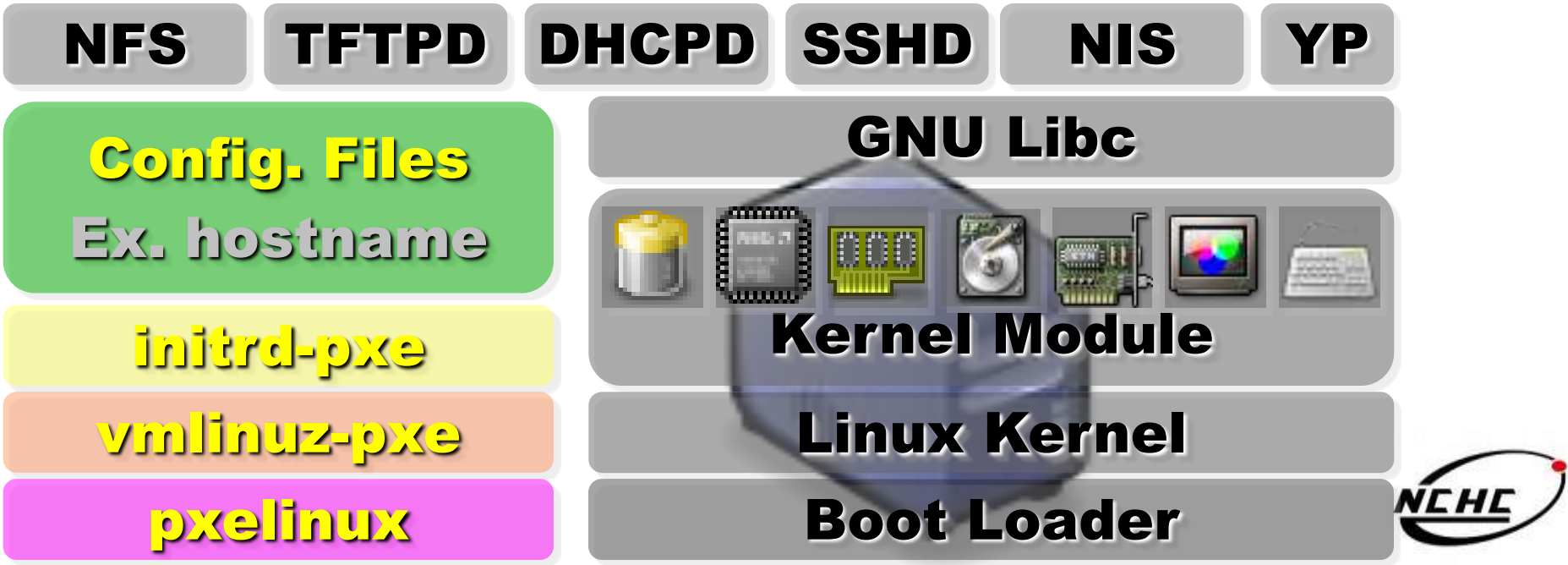


DRBL Server

based on existing Open Source and keep Hacking!

NCHC

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

NCHC

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

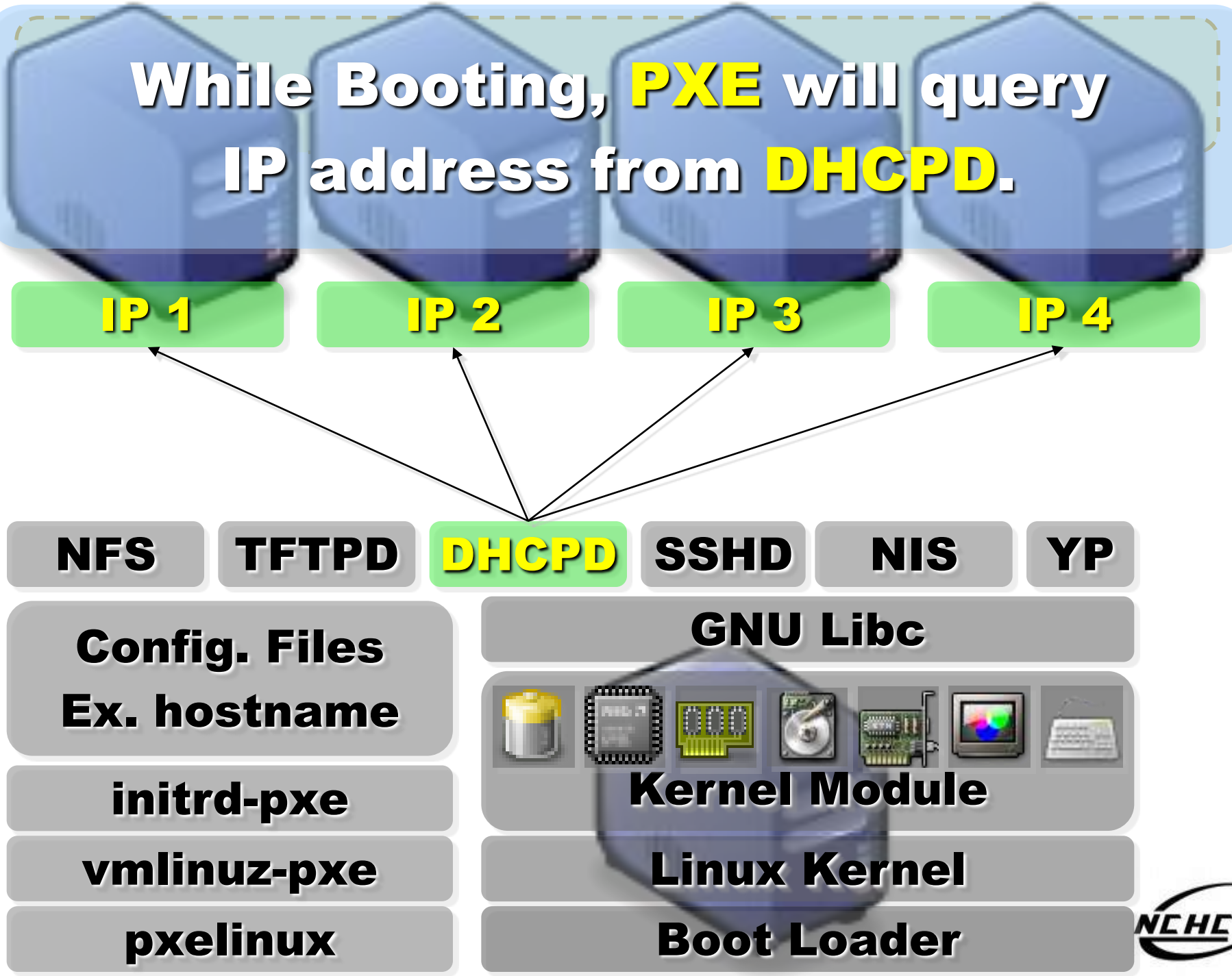
Kernel Module

Linux Kernel

Boot Loader

NCHC

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

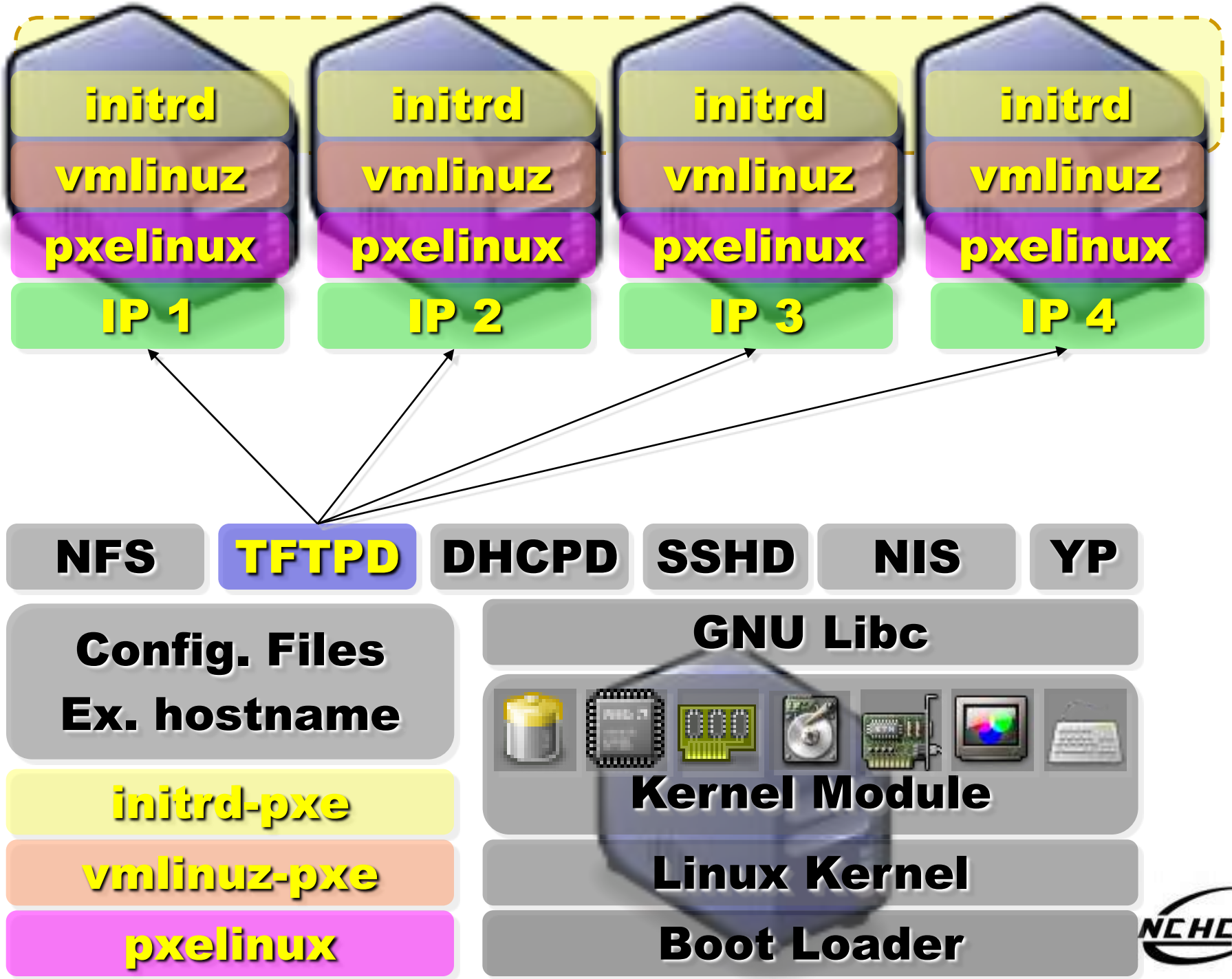
GNU Libc

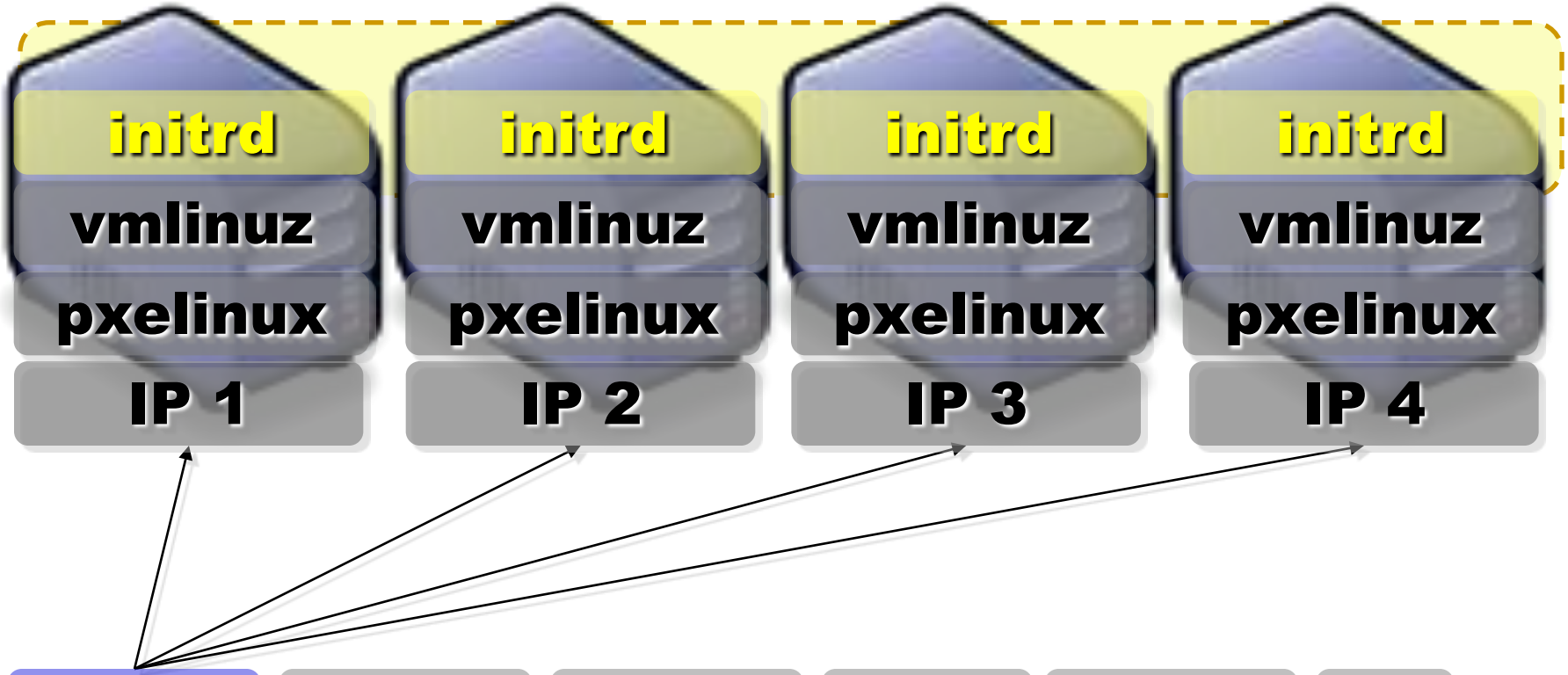
Kernel Module

Linux Kernel

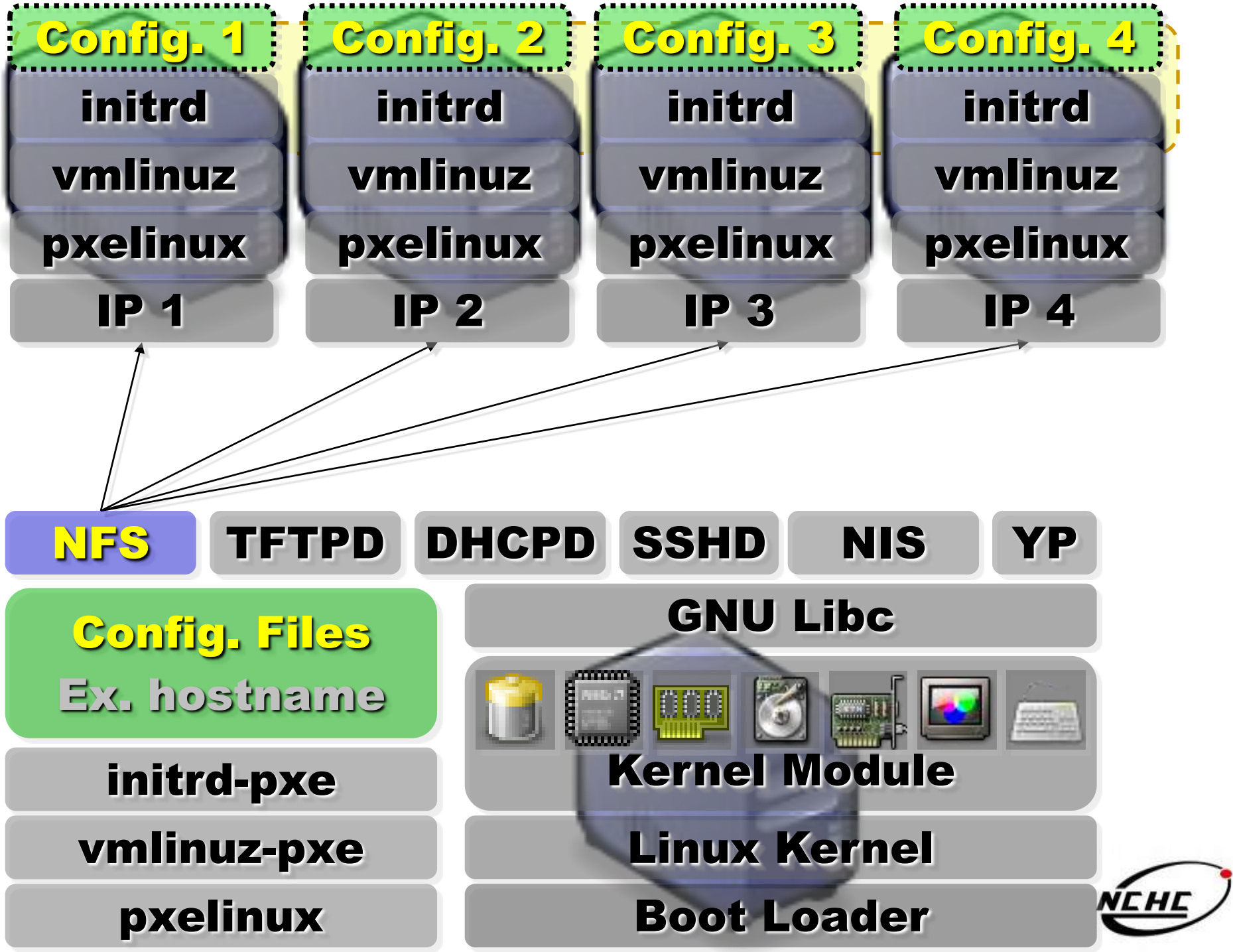
Boot Loader

NCHC


initrd
vmlinuz
pxelinux
IP 1
initrd
vmlinuz
pxelinux
IP 2
initrd
vmlinuz
pxelinux
IP 3
initrd
vmlinuz
pxelinux
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



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

Applications and **Services** will also deployed to each Compute Node via **NFS** ....

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

NCHC

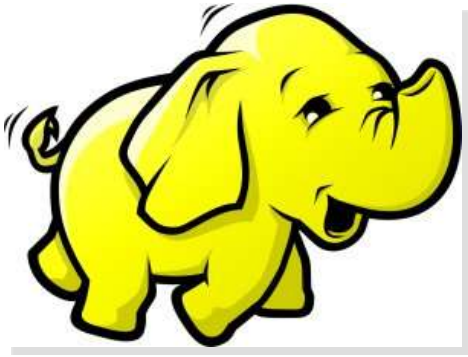
PART 2 -1:

當企鵝龍遇上小飛象

Jazz Wang

Yao-Tsung Wang

jazz@nchc.org.tw



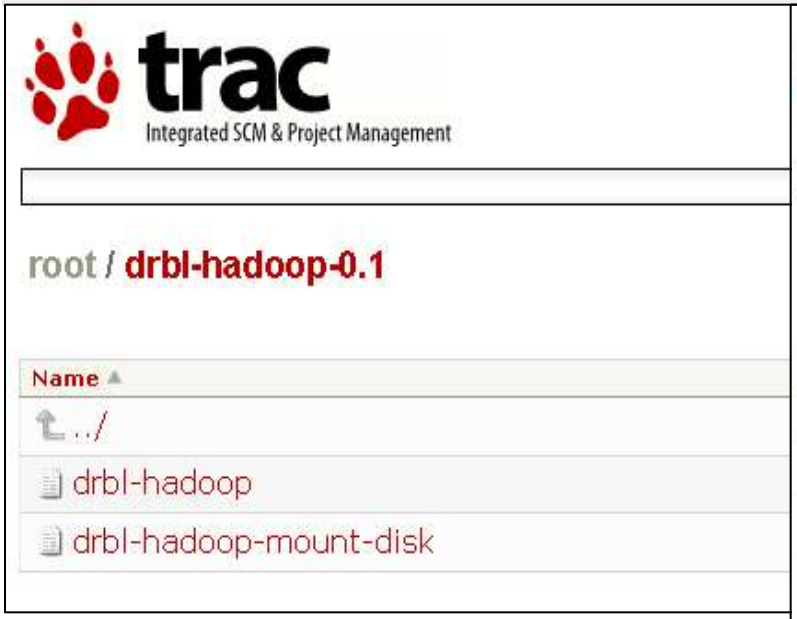
使用DRBL佈署Hadoop

- 仍在開發中，待整理套件
- **drbl-hadoop – 掛載本機硬碟給 HDFS 用**

```
svn co http://trac.nchc.org.tw/pub/grid/drbl-hadoop
```

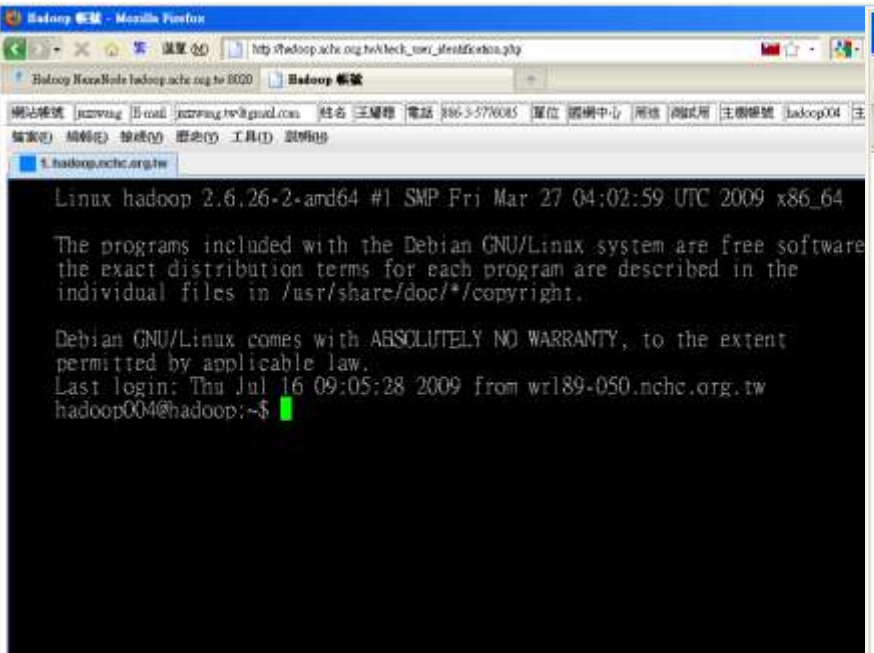
- **hadoop-register – 註冊網站與ssh applet**

```
svn co http://trac.nchc.org.tw/pub/cloud/hadoop-register
```



關於hadoop.nchc.org.tw

- **DRBL Server - 1台(hadoop)，加大/home與/tftpboot空間。**
- **DRBL Client - 19台(hadoop101~hadoop119)**
- **使用Cloudera的Debian套件**
- **使用drbl-hadoop 的設定跟init.d script來協助部署**
- **使用hadoop-register 來提供使用者註冊與ssh applet介面**



Lesson Learn

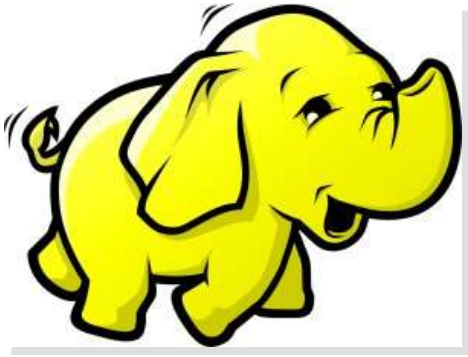
- **Cloudera套件的好處：使用init.d script 來啟動關閉**
  - **name node, data node, job tracker, task tracker**
- 建立大量帳號：
  - 可透過**DRBL**內建指令完成 **/opt/drbl/sbin/drbl-useradd**
- 使用者預設**HDFS**家目錄
  - 跑迴圈切換使用者，下 **hadoop fs -mkdir tmp**
- 設定使用者**HDFS**權限
  - 跑迴圈切換使用者，下 **hadoop dfs -chown $(id) /usr/$(id)**
- **HDFS會使用/var/lib/hadoop/cache/hadoop/dfs**
- **MapReduce會使用/var/lib/hadoop/cache/hadoop/mapred**

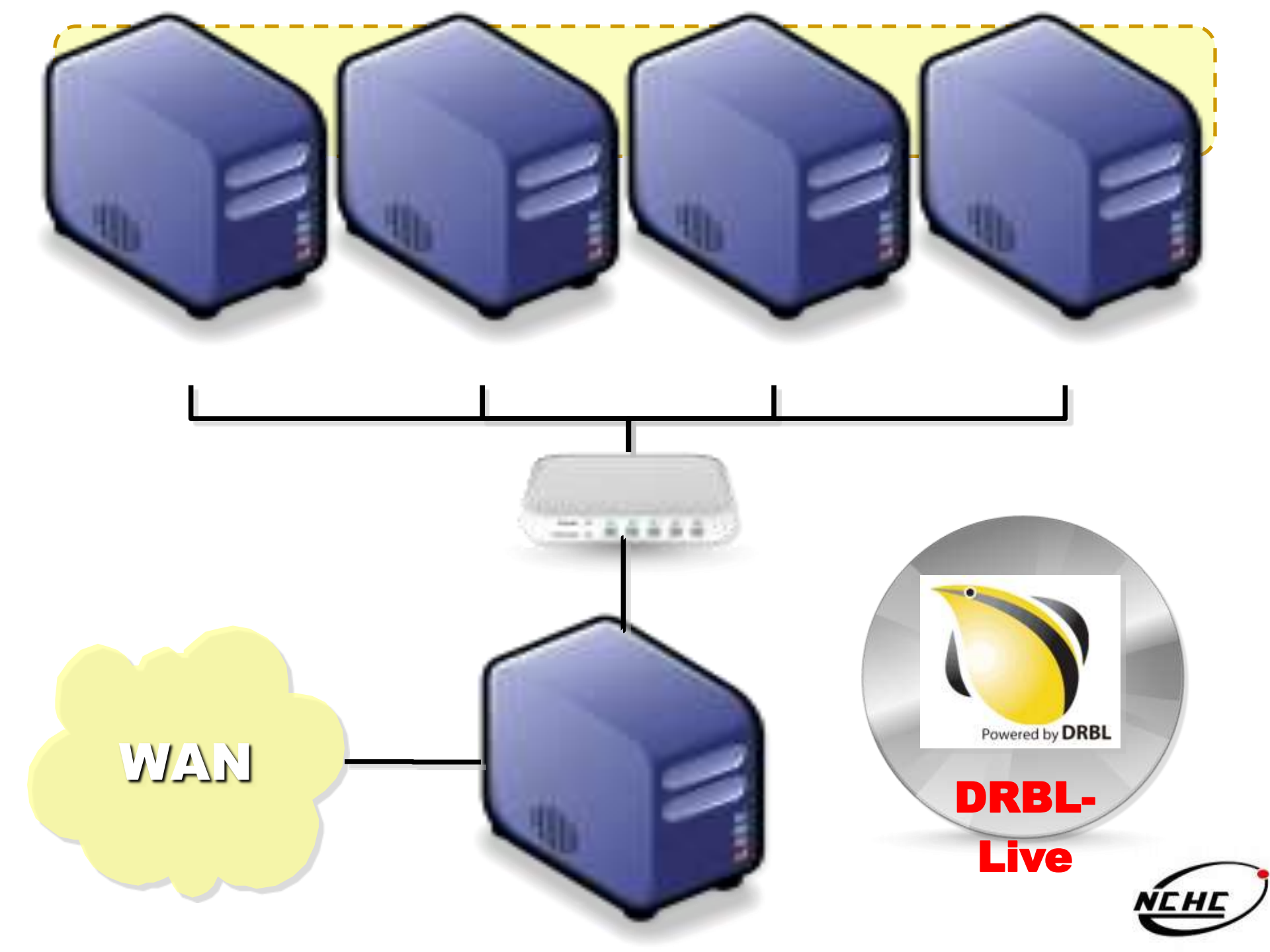
NCHC

PART 2 -2:

Live Demo

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




WAN
Powered by DRBL
DRBL-
Live
NCHC

Demo with DRBL-Live CD

1. Boot Server with DRBL-Live CD

http://free.nchc.org.tw/drbl-live/stable/

2. Download DRBL-Hadoop Script

http://classcloud.org/drbl-hadoop-live.sh

http://classcloud.org/drbl-hadoop-live-run.sh

3. Follow the steps

http://classcloud.org/drbl-hadoop

Questions?

Jazz Wang

Yao-Tsung Wang

jazz@nchc.org.tw

Powered by DRBL

NCHC