

分散式運算基本觀念與BOINC簡介

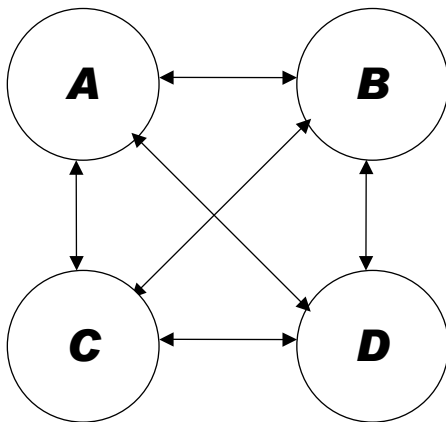
Jazz Wang
Yao-Tsung Wang
jazz@nchc.org.tw



Powered by DRBL

Different Type of Multiple Users Scenario (1)

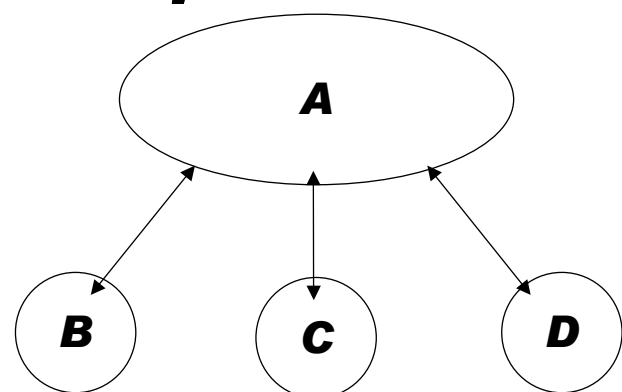
Peer Dispatch Data



**Peer
Receive Data**

TCP/IP Peer-to-Peer

Server Dispatch Data

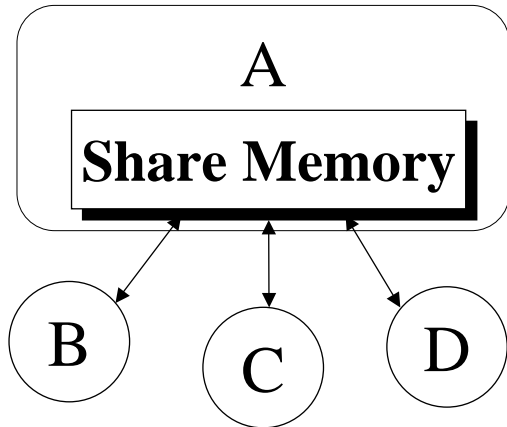


**Client
Receive Data**

TCP/IP Client-Server

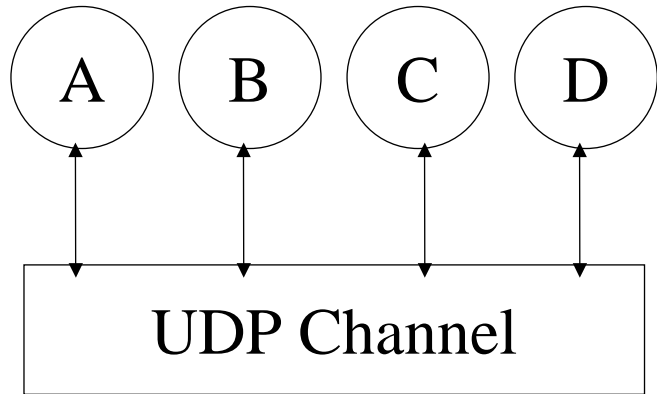
Different Type of Multiple Users Scenario (2)

**Broker
Store Shared Data**



**Peer
Dispatch Data
Receive Data**

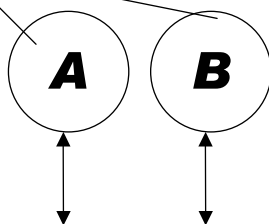
**Peer
Dispatch Data
Receive Data**



UDP Multicast / Broadcast

3 Roles of Distributed Computing

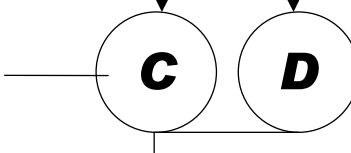
**Service
Provide
Object, Data
or Program**



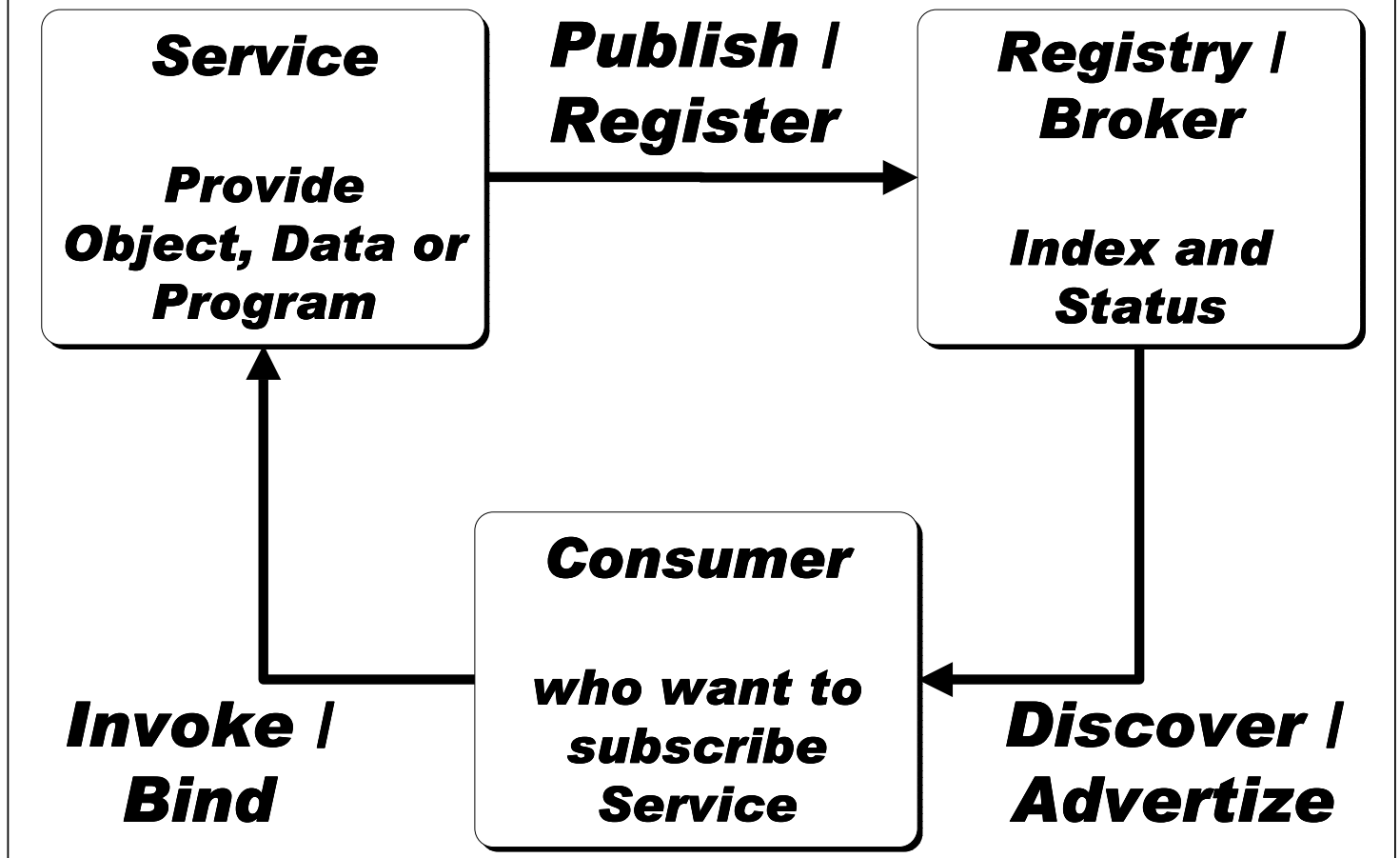
Distributed OS

**Registry /
Broker
Provide
Service Index
and
Real-time
Status**

**Consumer
who want to
subscribe
Service**



3 Basic Actions between 3 Roles



Well-Known Distributed Object Technology and Web Service

CORBA

<http://www.corba.org/>

Java RMI

<http://java.sun.com/javase/technologies/core/basic/rmi/>

DCOM

<http://msdn.microsoft.com/en-us/library/ms809311.aspx>

HLA / IEEE 1516

http://en.wikipedia.org/wiki/IEEE_1516

UDDI

<http://en.wikipedia.org/wiki/UDDI>

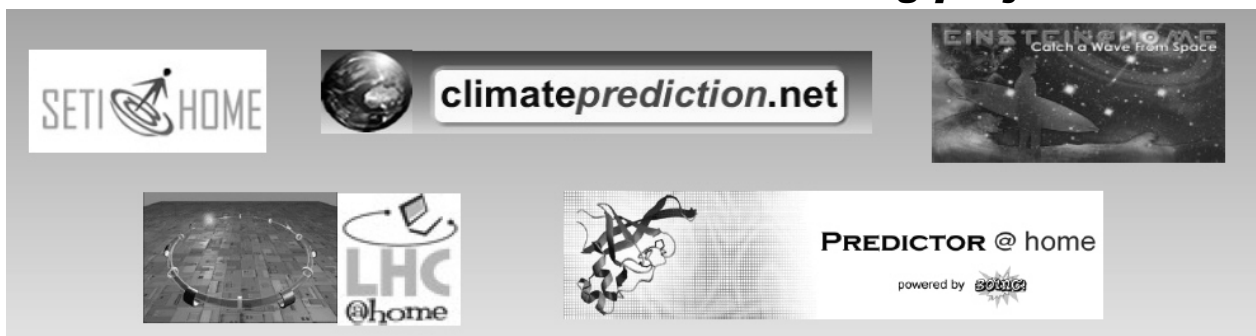
BOINC ??



Source: Linux Magazine Issue 71 October 2006
<http://www.linux-magazine.com>

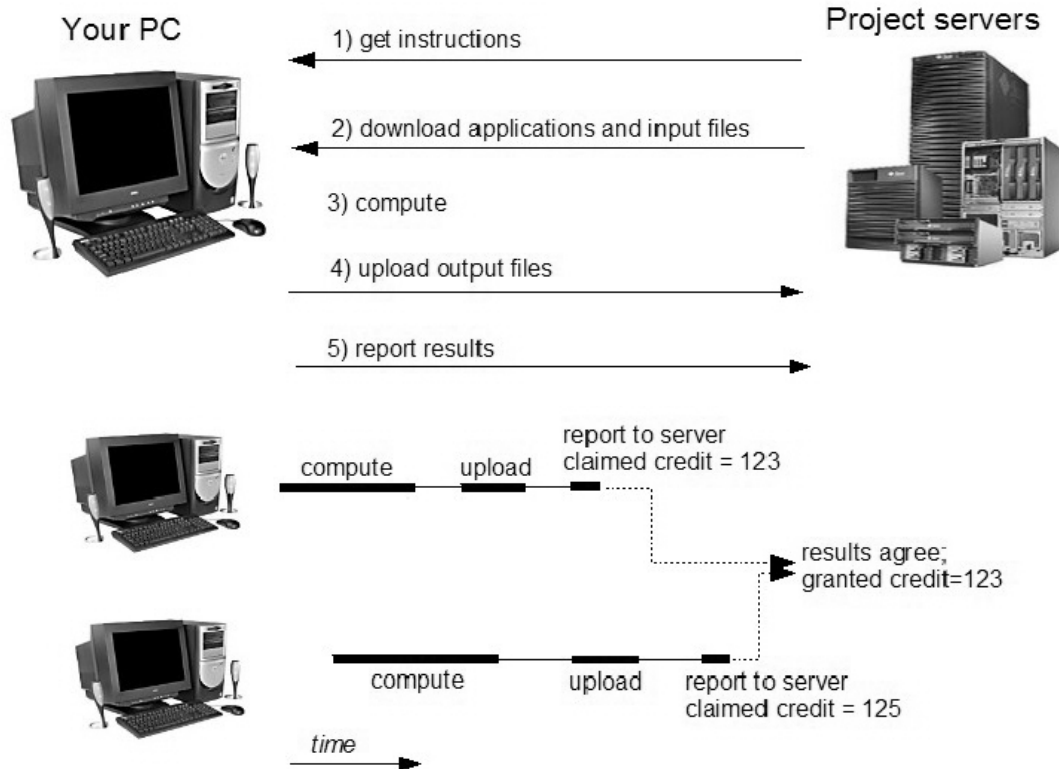
Brief Introduction of BOINC

- **Berkeley Open Infrastructure for Network Computing**
- **<http://boinc.berkeley.edu/>**
- **Started in February 2002**
- **The most well-known BOINC-based project, SETI@home, started 2004/06**
- **BOINC also had been used in following projects:**



Reference: *Architecture and basic principles*, Juan Antonio Lopez Perez, CERN, November, 2005

How BOINC works ?



Reference: *Architecture and basic principles*, Juan Antonio Lopez Perez, CERN, November, 2005

GIVE ME YOUR HAND

**RISE YOUR
HAND**

**WHILE YOU
NEED A HAND**



IT'S THE BEST WAY TO LEARN

首先介紹一下WorldCommunityGrid

The screenshot shows the homepage of World Community Grid. At the top, there is a navigation menu with links for HOME, ABOUT US, RESEARCH, FORUMS, STATISTICS, MY GRID, and a language selection dropdown. A login section is visible with fields for member name (jazzwang) and password, and a 'sign in' button. Below the navigation, a large banner reads 'You can help change the world' with a sub-header 'Join World Community Grid today to contribute to projects that benefit humanity'. The main content area is divided into several sections: 'Who We Are' with a photo of a group of people and a description of the mission; 'How You Can Help' with a photo of hands and a description of donating computer time; 'What We Do' with a list of projects like 'Nutritious Rice for the World'; and 'What's New' with a table of top-performing teams. On the right side, there are buttons for 'SUBMIT RESEARCH', 'BECOME A MEMBER', and 'KIDS WANT TO KNOW THE GRID'. The browser's address bar shows 'http://www.worldcommunitygrid.org/index.jsp?language=en_US'.

有一些跟生物資訊及藥物設計有關的計畫

The screenshot shows the project page for 'Discovering Dengue Drugs - Together'. The page header includes the user's name 'jazzwang', accumulated points (63,929), and current ranking (127,305). The main content area features a large banner for the project with the text 'RESEARCH Discovering Dengue Drugs - Together'. Below the banner, there is a section for 'Project Status and Findings' and a 'Mission' section. A table of 'Significance' is also present. On the left side, there is a sidebar with 'Active Research' and 'Inactive Research' sections. On the right side, there are buttons for 'SUBMIT RESEARCH', 'Visit the Forums', and 'Tell A Friend'. The browser's address bar shows 'http://www.worldcommunitygrid.org/projects_showcase/dddt/viewDddtMain.do'.

開始實作前，請先加入會員

World Community Grid - Home - Mozilla Firefox

http://www.worldcommunitygrid.org/index.jsp?language=en_US

world community grid. technology solving problems

member name: jazzwang password: ***** sign in

> forgot member name? > forgot password? remember me:

HOME ABOUT US RESEARCH FORUMS STATISTICS MY GRID Select Language HELP

You can help change the world
Join World Community Grid today to contribute to projects that benefit humanity

Who We Are

World Community Grid's mission is to create the largest public computing grid benefiting humanity. Our work is built on the belief that technological innovation combined with visionary scientific research and large-scale volunteerism can change our world for the better. Our success depends on individuals - like you - collectively contributing their unused computer time to this not-for-profit endeavor.

- > [Members](#)
- > [Partners](#)
- > [Advisory Board](#)
- > [News & Media](#)

How You Can Help

Donate the time your computer is turned on, but is idle, to projects that benefit humanity! We provide the secure software that does it all for free, and you become part of a community that is helping to change the world. Once you install the software, you will be participating in World Community Grid. No other action must be taken; it's that simple! To learn more and join, click the button below.

- > [Become a Member](#)
- > [Become a Partner](#)
- > [Submit a Proposal](#)
- > [Find a Team](#)
- > [Tell a Friend](#)
- > [Marketing Toolkit](#)

download now

What We Do

Exciting work is now under way on projects that hold tremendous potential to benefit humanity.

- > [Nutritious Rice for the World](#)
- > [Help Conquer Cancer](#)
- > [AfricanClimate@Home](#)
- > [Discovering Dengue Drugs - Together](#)

What's New

World Community Grid would like to spotlight the following 5 teams for contributing the most run time (y:d:h:m:s) yesterday:

> 1. Team 2ch	12:178:10:07:54
> 2. Easynews	10:290:18:49:54
> 3. XtremeSystems	4:285:10:05:01
> 4. IBM	2:124:15:23:50

SUBMIT RESEARCH

World Community Grid's Advisory Board is looking for research projects that can benefit technology positive im humanity.

BECOME A MEMB

The power to change starts here. Join World Community Grid today!

join now

KIDS WANT TO KNOW THE GRID

What is it? Click to find out at [TryScience.org](#)

tryscience home

POWERED BY IBM

完成

填寫欲申請之帳號資料

World Community Grid - Register - Mozilla Firefox

http://www.worldcommunitygrid.org/reg/viewRegister.do

world community grid. technology solving problems

member name: password: sign in

> forgot member name? > forgot password? remember me:

HOME ABOUT US RESEARCH FORUMS STATISTICS MY GRID Select Language HELP

Register

- Create Member Name
- License Agreement

Select Projects

- Select Projects

Download

- Download software
- Install software

Explore

- Explore World Community Grid

It's free and secure! To start, just fill out the form below, and click "continue."

*Member Name: jazzbear Is this name available?

*Enter a Password: Password is valid

*Retype your Password: Passwords match

*E-mail Address: jazz@nchc.org.tw E-mail Address is valid

*Confirm E-mail Address: jazz@nchc.org.tw E-mail Addresses match

Information: World Community Grid may occasionally send out information and/or updates via email. Please uncheck the box if you do not want to receive information updates from World Community Grid.

End User Software License Agreement

* Similar to other software downloads, World Community Grid software requires that you accept an end user license agreement. Please read the [Software License Agreement](#), and check the box to acknowledge that you accept the agreement.

(Fields marked with a * are required)

Already Registered?

Do you want to download the software again? [Click here](#) to visit the Download page.

Visit the Forums

If you need technical assistance beyond what's available in [Help](#), please [visit the forums](#) to post your questions and get answers on how others are using World Community Grid.

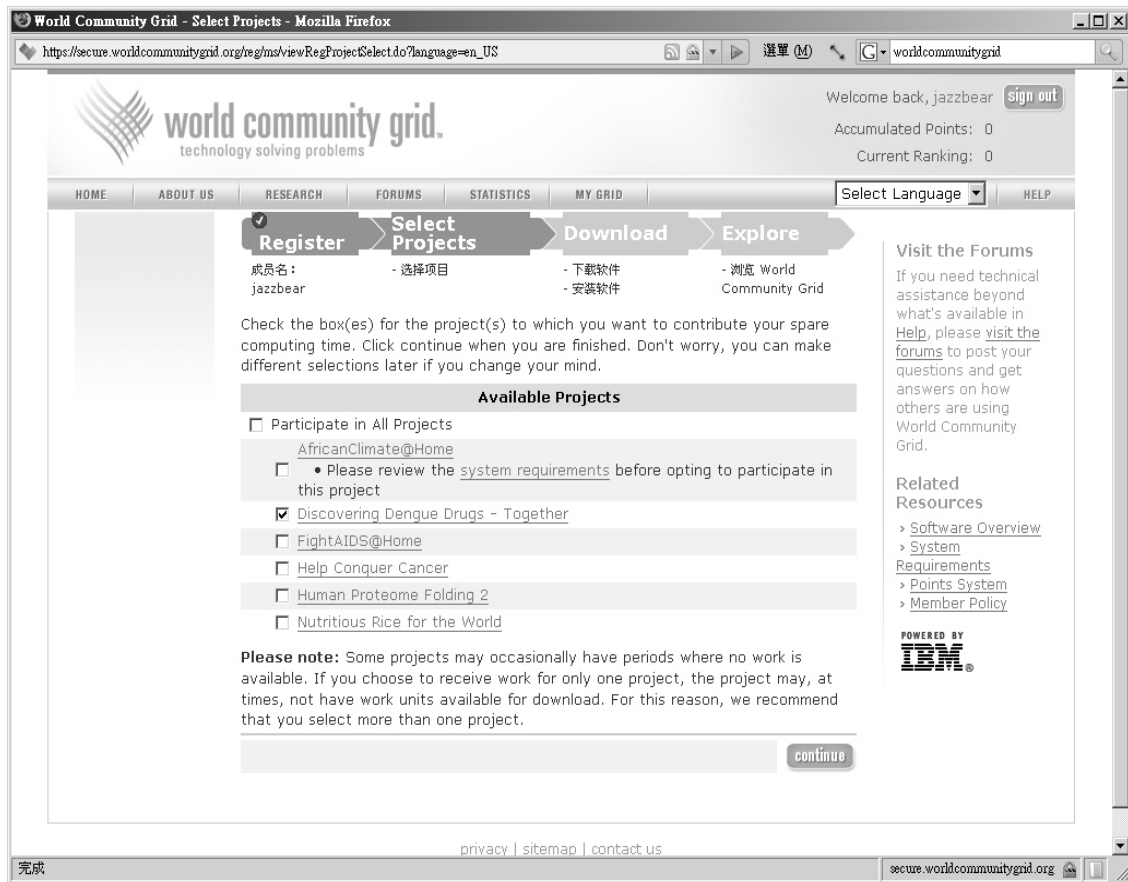
Related Resources

- > [Software Overview](#)
- > [System Requirements](#)
- > [Points System](#)
- > [Member Policy](#)

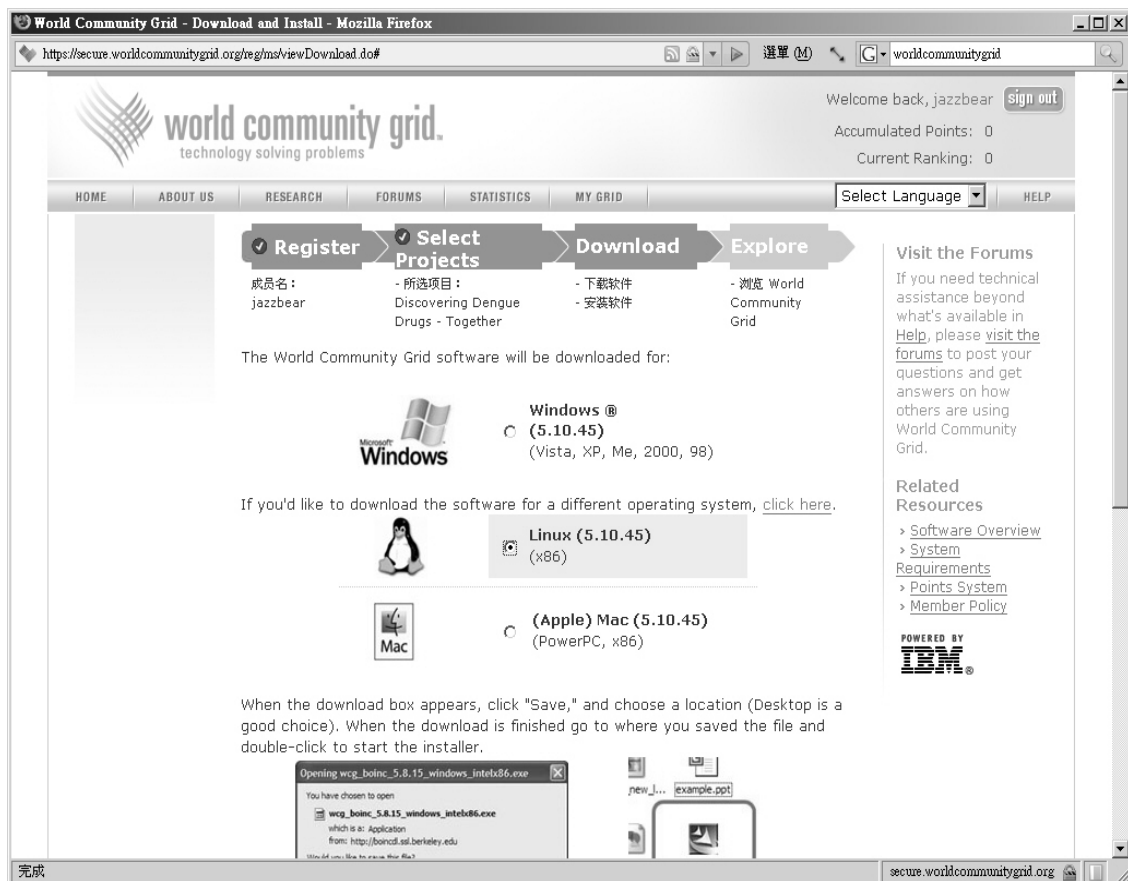
POWERED BY IBM

完成

勾選欲參與之大型分散式運算專案



如果你有其他閒置的機器，可下載軟體



等一下的實作會用到BOINC認證金鑰

The screenshot shows the 'My Profile' page on the World Community Grid website. A callout bubble points to the 'MY GRID' tab in the navigation menu, labeled '1. 點選『MyGrid』'. Another callout bubble points to the 'My Profile' link in the left sidebar, labeled '2. 點選『My Profile』'. The page displays user information for 'jazzbear', including accumulated points, current ranking, and a beta testing status. There are input fields for name, email address, and country, along with a checkbox for receiving updates.

請留著瀏覽器畫面供待會查詢認證金鑰

The screenshot shows the 'My Profile' page with the BOINC account information section highlighted. A callout bubble points to the 'BOINC Account Key' field, which contains the value '767622421ce494418f6d5e67aff9dd50'. The page also shows the BOINC Project URL, BOINC Account Number, BOINC Cross-Project Id, and BOINC Show Hosts checkbox. A 'SAVE' button is visible at the bottom of the form.



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

雲端運算簡介

王耀聰 陳威宇

Jazz@nchc.org.tw

waue@nchc.org.tw

2008. 04 . 27-28

國家高速網路與計算中心(NCHC)

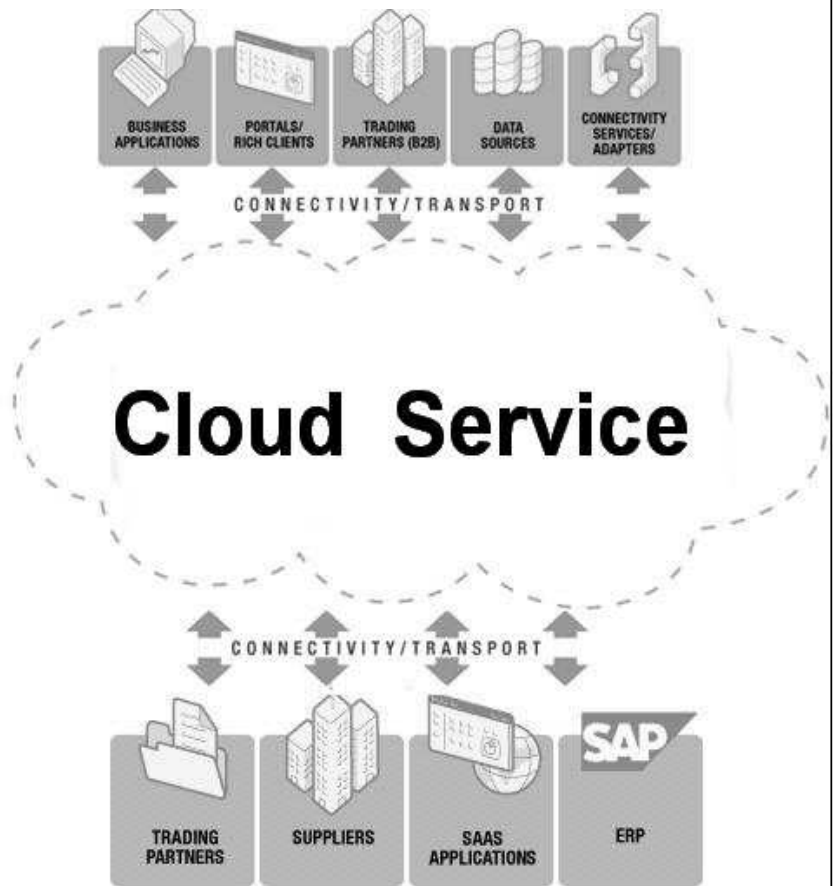
 自由軟體實驗室



雲端運算??

雲端服務

- Web Email
- 線上掃毒
- YouTube
- 線上文件
- 部落格
- ...



雲端運算特色

超大規模

虛擬化

高可靠度

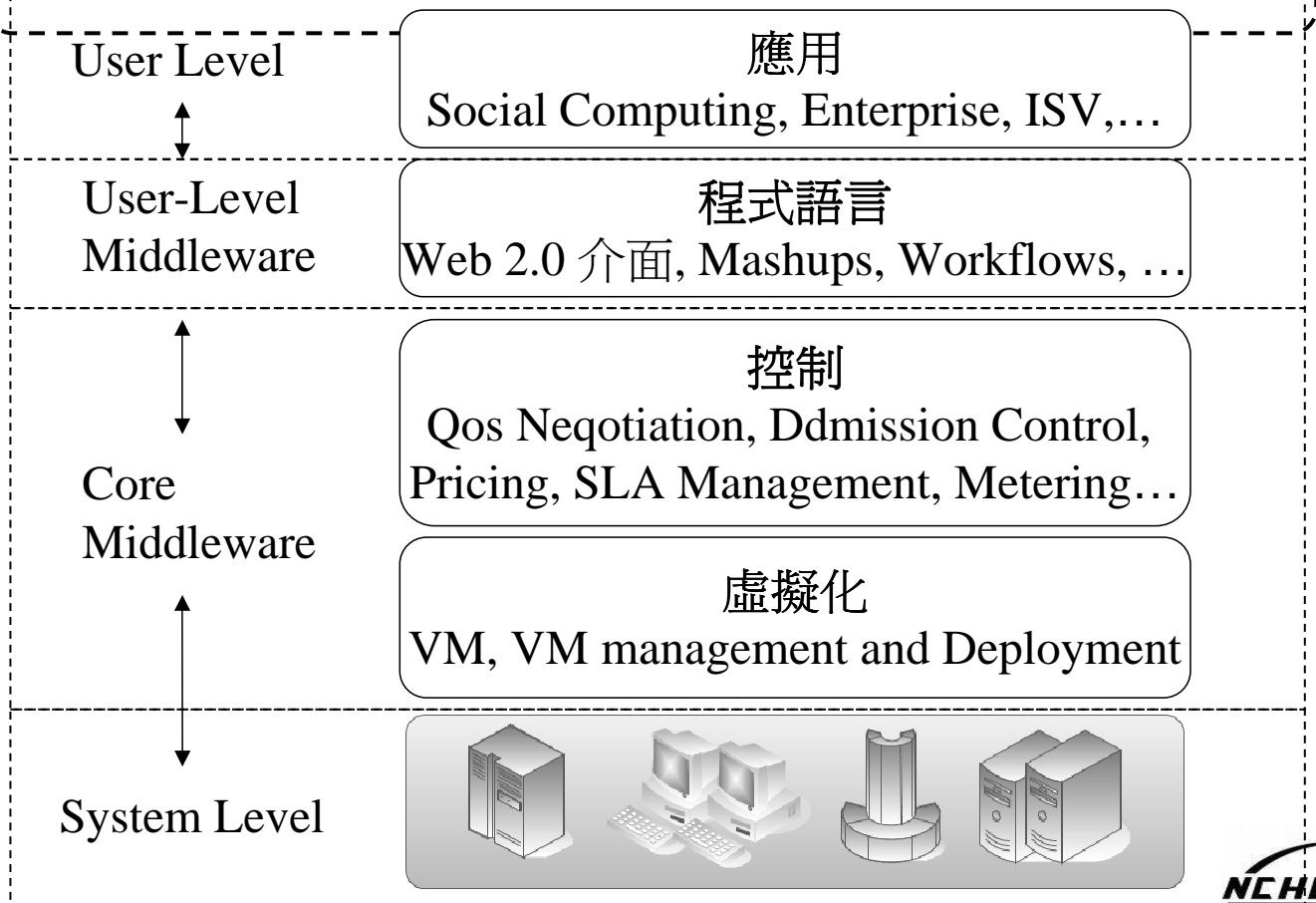
使用者付費

高通用性

成本低

高擴充性

雲端運算的架構



現有的雲端運算服務

- Windows
- Google
- Amazon
- Yahoo
- ...



Amazon : Web Service

- AWS
- 虛擬化的技術：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



<http://eblog.cisnet.org.tw/post/Cloud-Computing.aspx>



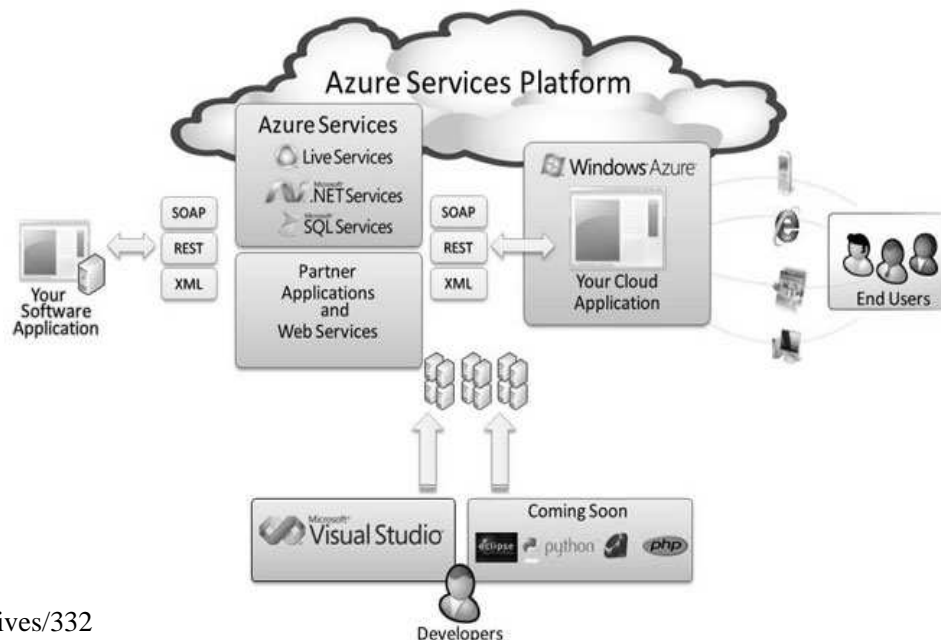
Google : App Engine

- 網路平台，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
- Limit：
 - Language: Python、Java
 - web applications



Windows : Azure

- Windows 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。
- .Net services
- SQL services
- Live services



<http://tech.cipper.com/index.php/archives/332>

Yahoo : Hadoop

- Apache 項目，Yahoo 資助、開發與運用
 - 2006年開始參與開源的雲端運算框架Hadoop，並將其使用在內部服務中。
 - 2008年2：目前最大的Hadoop應用
 - 2千臺伺服器
 - 執行超過1萬個Hadoop虛擬機器
 - 5 Petabytes的網頁內容
 - 分析1兆個網路連結



雲端運算產業類型

SaaS
Software as a Service

PaaS
Platform as a Service

IaaS
Infrastructure as a Service

www.spoutingshite.com/wp-content/uploads/2008/12/saas_and_cloud_computing.ppt



雲端運算產業

架構即服務

- 提供了核心計算資源和網絡架構的服務
- infrastructure stack:
 - Full OS access
 - Firewalls
 - Routers
 - Load balancing

IaaS



雲端運算產業

Examples

- Flexiscale
- AWS: EC2 (Amazon Elastic Compute Cloud)

IaaS



雲端運算產業

平台即服務

- 提供平台給系統管理員和開發人員，以為它構建、測試及部署定製應用程序
- 管理系統的成本昂貴
- Popular services
 - Storage
 - Database
 - Scalability

PaaS

IaaS

雲端運算產業

Examples

- Google App Engine
- AWS: S3 (Simple Storage Service)
- Microsoft Azure

PaaS

IaaS

雲端運算產業

軟體即服務

- 通過Internet提供軟體的模式，用戶向提供商租用基於Web的軟體，來管理企業經營活動，且無需對軟體進行維護，服務提供商會全權管理和維護軟體

SaaS

PaaS

IaaS

雲端運算產業

SaaS

軟體即服務

- 不用管理硬體與軟體
- 操作簡單 (瀏覽器)
- Pay per use
- Instant Scalability
- Security
- Reliability

PaaS

IaaS

雲端運算產業

SaaS

Examples

- Google Docs
- CRM
- Financial Planning
- Human Resources
- Word processing
- Salesforce.com

PaaS

IaaS

比較表

服務 屬性	Amazon EC2	Google App Engine	Microsoft Azure	Yahoo Hadoop
架構	Iaas/Paas	Paas	Paas	Software
服務型態	Compute/ Storage	Web application	Web and non- web	Software
管理技術	OS on Xen hypervisor	Application container	OS through Fabric controller	Map / Reduce Architecture
使用者介面	EC2 Command-line tools	Web-based Administration console	Windows Azure portal	Command line and web
APIs	yes	yes	yes	yes
收費	yes	maybe	yes	no
程式語言	AMI (Amazon Machine Image)	Python	.NET framework	Java,



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

Hadoop 簡介

王耀聰 陳威宇

Jazz@nchc.org.tw

waue@nchc.org.tw

2008. 04 . 27-28

國家高速網路與計算中心(NCHC)

Outline

- 什麼是 Hadoop ?
- 有什麼特色 ?
- 怎麼來的呢 ?
- 有誰在用 ?
- 有實用案例嗎 ?



什麼是
Hadoop

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
 - 藉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

起源：Google論文

- Google File System
 - 可擴充的分散式檔案系統
 - 設計目的在於可以給大量的用戶提供總體性能較高的服務
 - 適用於分散式、對大量資訊進行存取的應用
 - 可運作在一般的普通主機上，且提供錯誤容忍的能力
- “The Google File System “發表於SOSP'03 October，並將設計的概念公開

起源：Google論文

- Google's GFS & MapReduce papers published:
 - SOSP 2003 : “The Google File System”
 - OSDI 2004 : “MapReduce : Simplified Data Processing on Large Cluster”
 - OSDI 2006 : “Bigtable: A Distributed Storage System for Structured Data”
- directly address Nutch's scaling issues

<http://research.google.com/pubs/papers.html>



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.)



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



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.



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

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

Hadoop Distributed File System

王耀聰 陳威宇

Jazz@nchc.org.tw

waue@nchc.org.tw

2008. 04 . 27-28

國家高速網路與計算中心(NCHC)

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

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)
- Rack awareness
 - 用來告知網路拓樸狀況
- Metadata
 - 屬性資料

設計目標 (1)

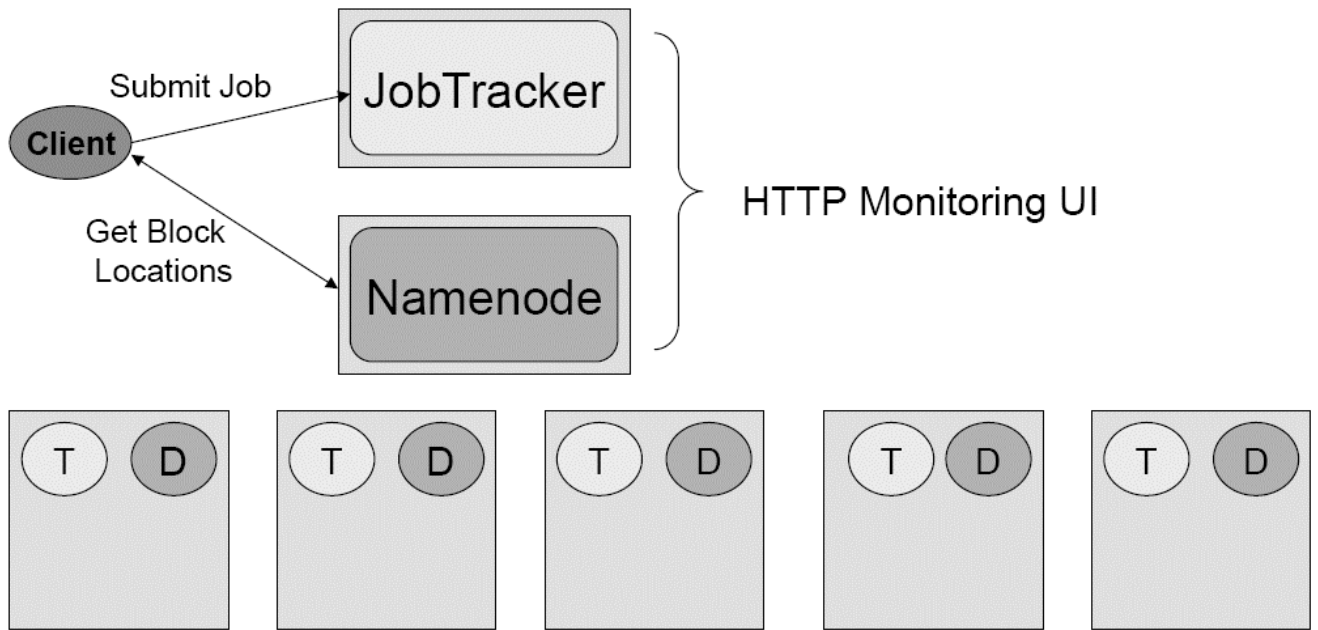
- 硬體錯誤容忍能力
 - 硬體錯誤是正常而非異常
 - 迅速地自動恢復
- 串流式的資料存取
 - 批次處理多於用戶交互處理
 - 高**Throughput** > 低Latency
- 大規模資料集
 - 支援Perabytes等級的磁碟空間

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

HDFS的
架構？

架構



HDFS的
架構？

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

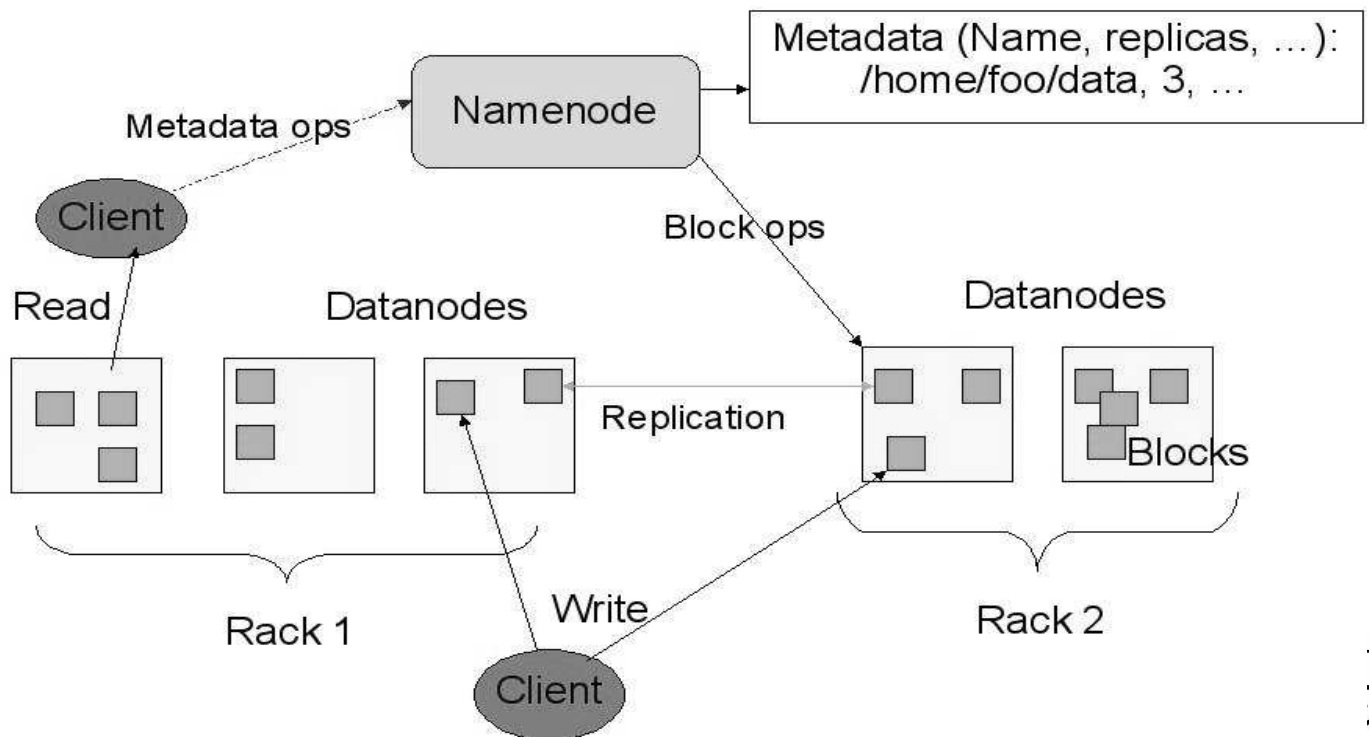
Datanode

- Workers
- 執行讀/寫動作
- 執行Namenode的副本策略

HDFS的
架構？

管理資料

HDFS Architecture



HDFS的
架構？

分派程序

Jobtracker

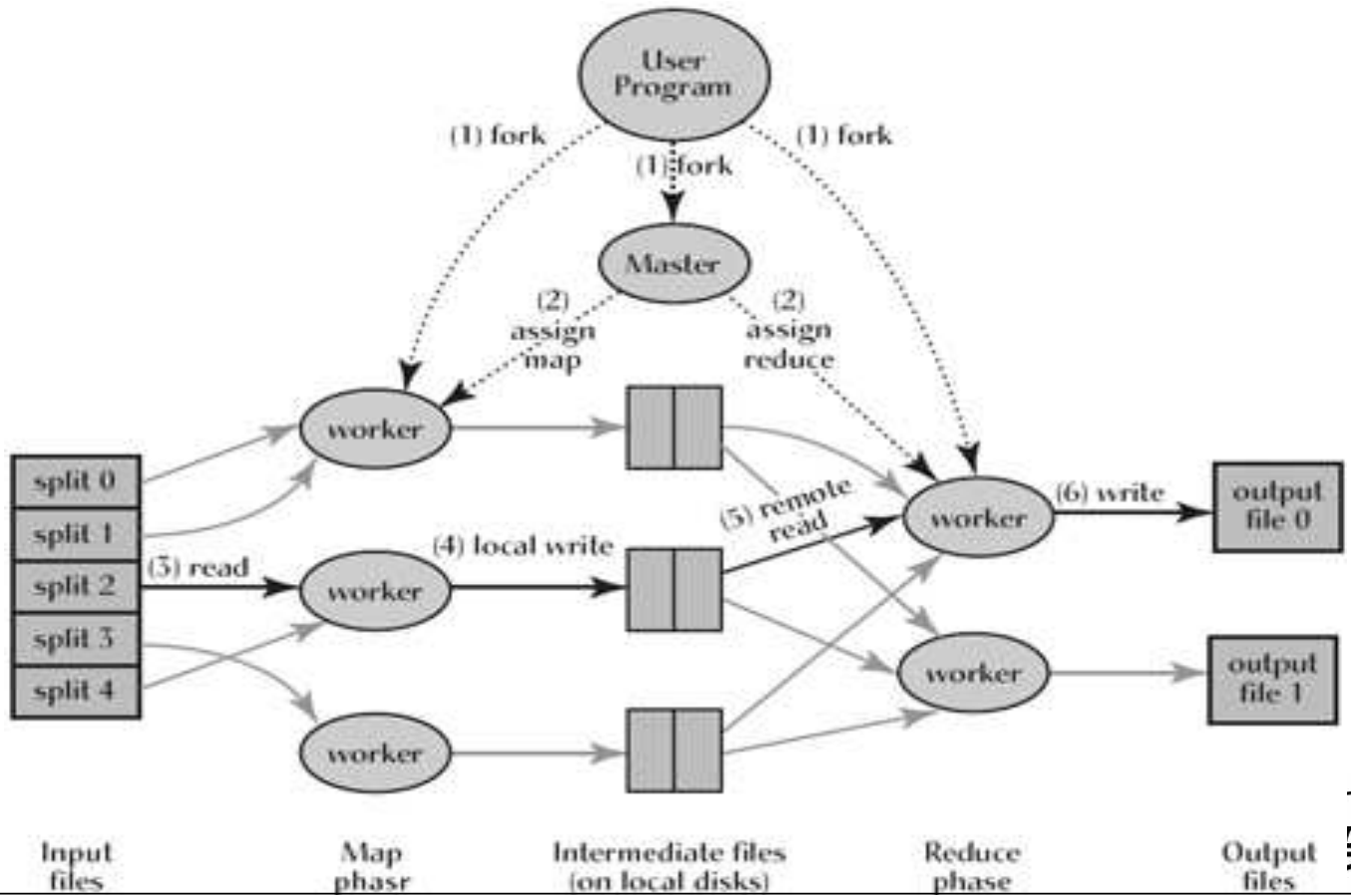
- Master
- 使用者發起工作
- 指派工作給 Tasktrackers
- 排程決策、工作分配、錯誤處理

Tasktrackers

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

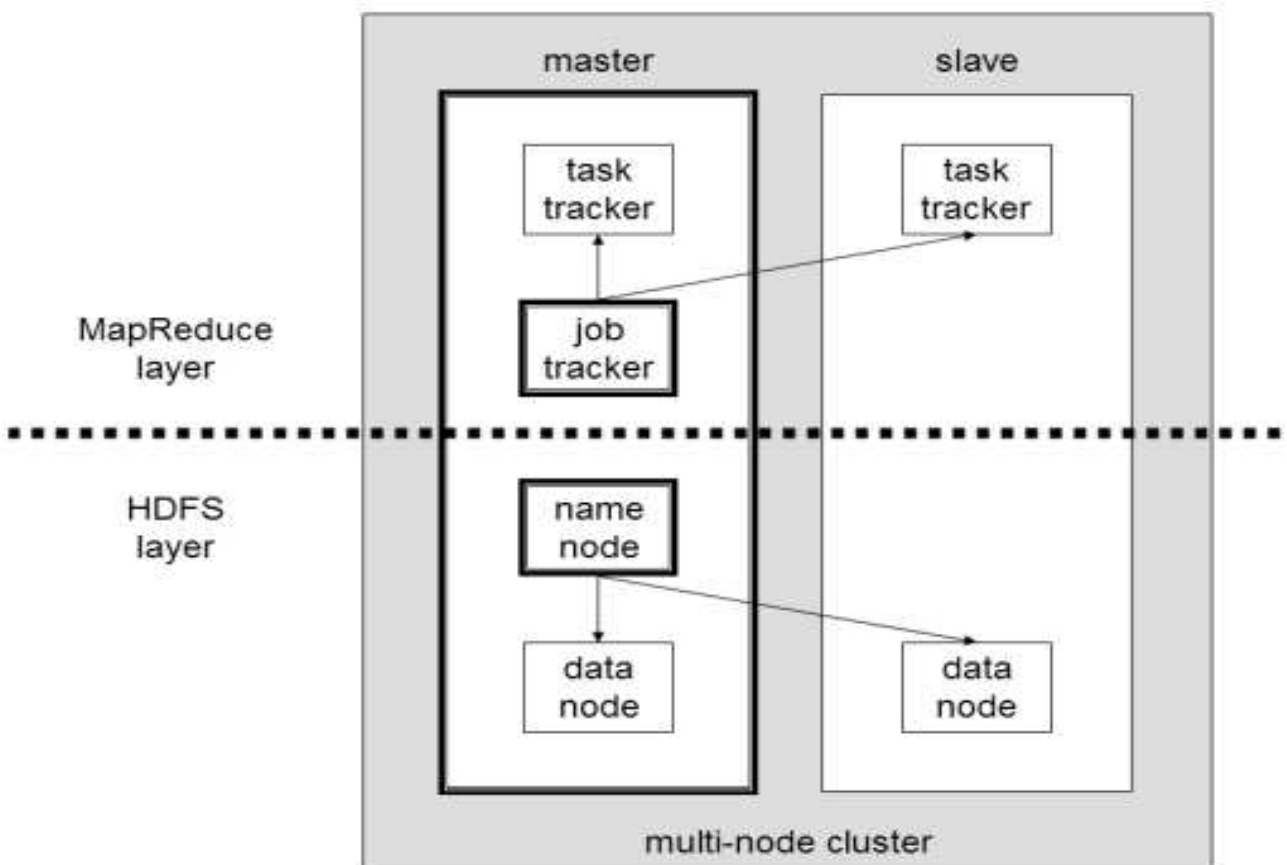
HDFS的
架構？

分派程序



HDFS的
架構？

Overview



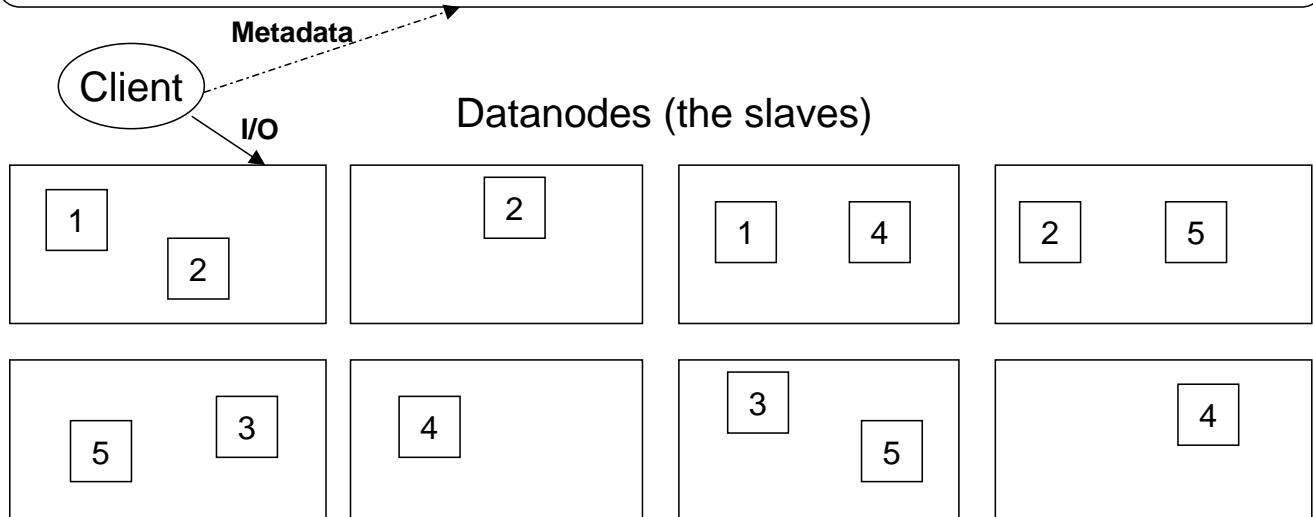
HDFS 運作

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.zip, 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）

Namenode

file1 (1,3)
file2 (2,4,5)

JobTracker

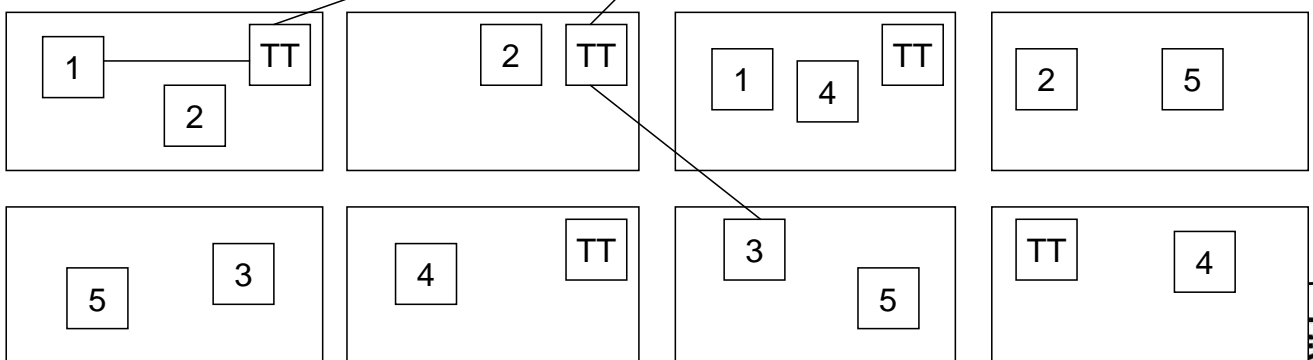
Map tasks
Reduce tasks

TaskTracker

TT

ask for task

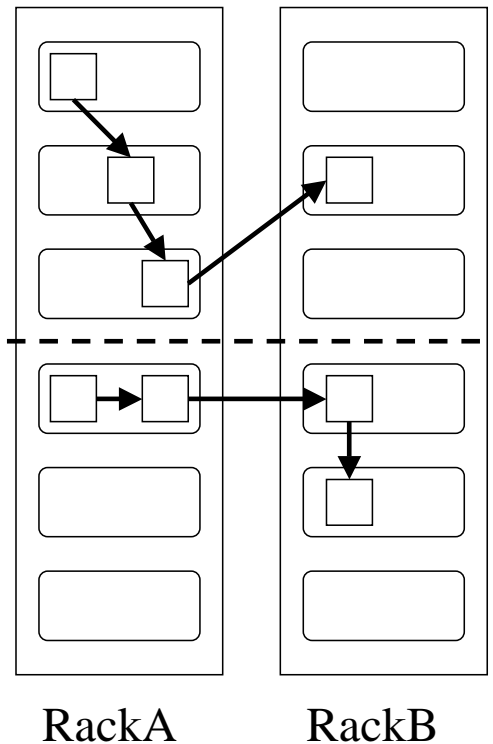
Block 1



HDFS 副本備份機制

• Original ~

- First : 同機架的不同節點
- Second : 同機架的另一節點
- Third : 不同機架另一節點
- More : 隨機挑選



• Hadoop 0.17 ~

- First : 同Client的節點上
- Second : 不同機架中的節點上
- Third : 同第二個副本的機架中的另一個節點上
- More : 隨機挑選

如何達成
其好處？

可靠性機制

常見的
三種
錯誤
狀況

資料崩毀

網路或
資料節點
失效

名稱節點
錯誤

• 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

如何達成
其好處？

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



功能為何？

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]]
```



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

Map Reduce 介紹

王耀聰 陳威宇

Jazz@nchc.org.tw

waue@nchc.org.tw

2008. 04 . 27-28

國家高速網路與計算中心(NCHC)

 自由軟體實驗室

Outline

- Why should we learn this ?
- What is MapReduce ?
- Where does it fit ?
- What is its benefit ?
- How does it work ?
- Must be in Java ?

Why should we learn this ?

目的



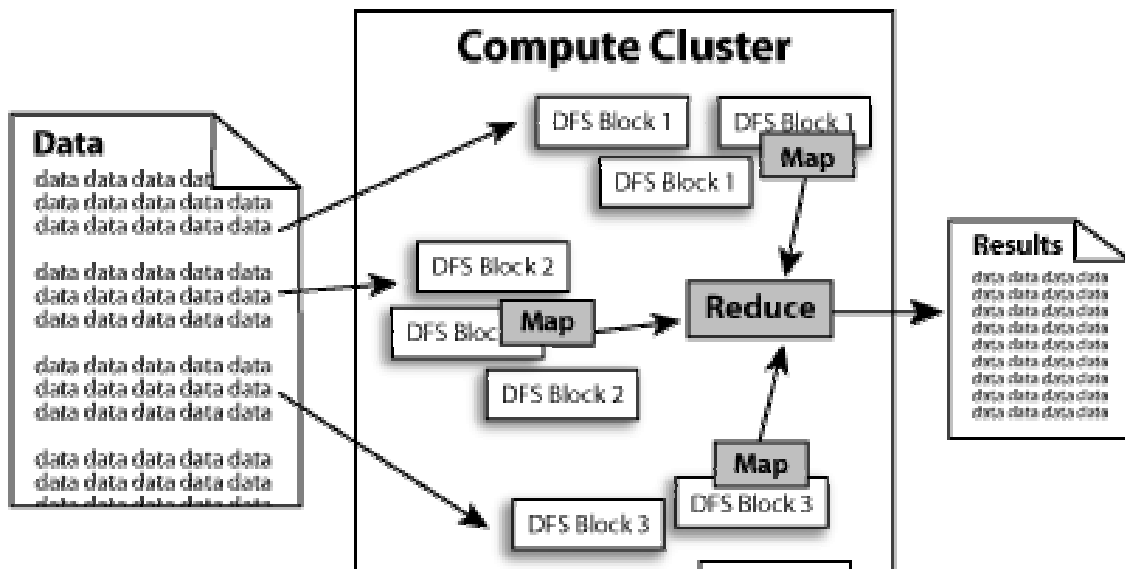
知己知彼百戰百勝



太歲頭上動土

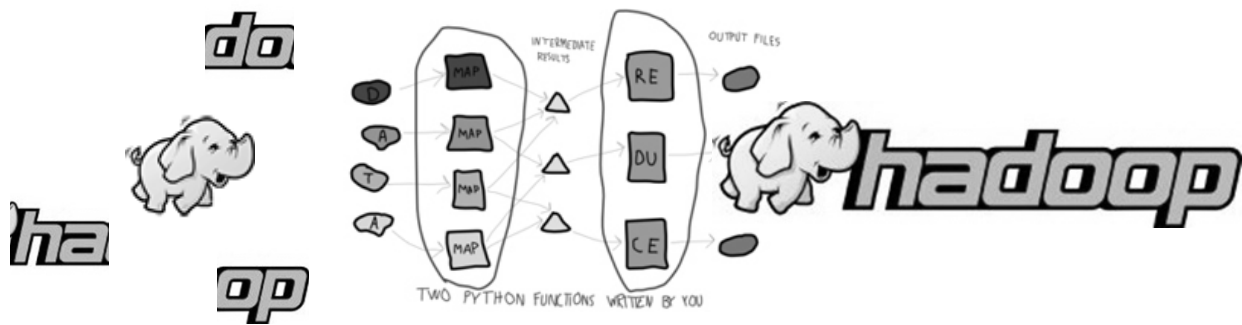
What is MapReduce ?

Google 原生定義



MapReduce is a framework for computing certain kinds of distributable problems using a large number of computers (nodes), collectively referred to as a cluster.

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。

MapReduce 由來

- Functional Programming : Map Reduce
 - map(...):
 - [1,2,3,4] - (*2) -> [2,4,6,8]
 - reduce(...):
 - [1,2,3,4] - (sum) -> 10
 - 對應演算法中的Divide and conquer
 - 將問題分解成很多個小問題之後，再做總和
- 首先被Google引用到程式設計的軟體架構內，使用在大規模數據的運算中

Where does it fix ?

應用範圍

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

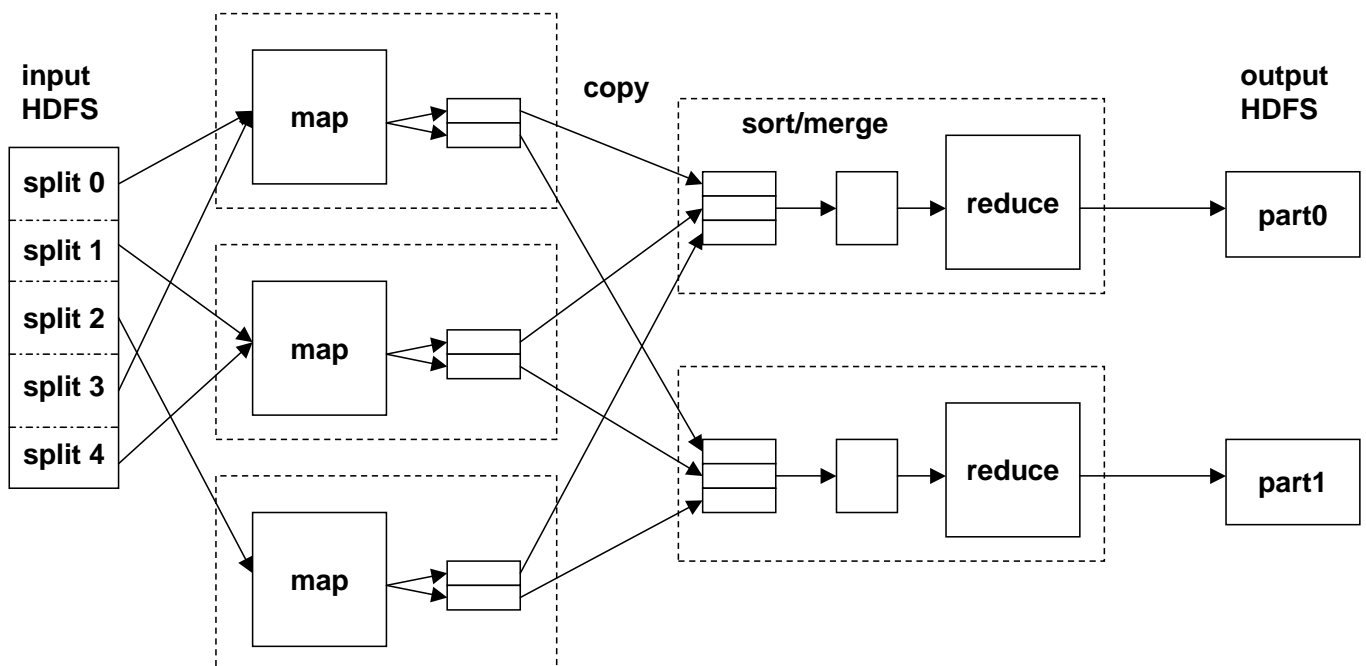


<http://www.dbms2.com/2008/08/26/known-applications-of-mapreduce/>



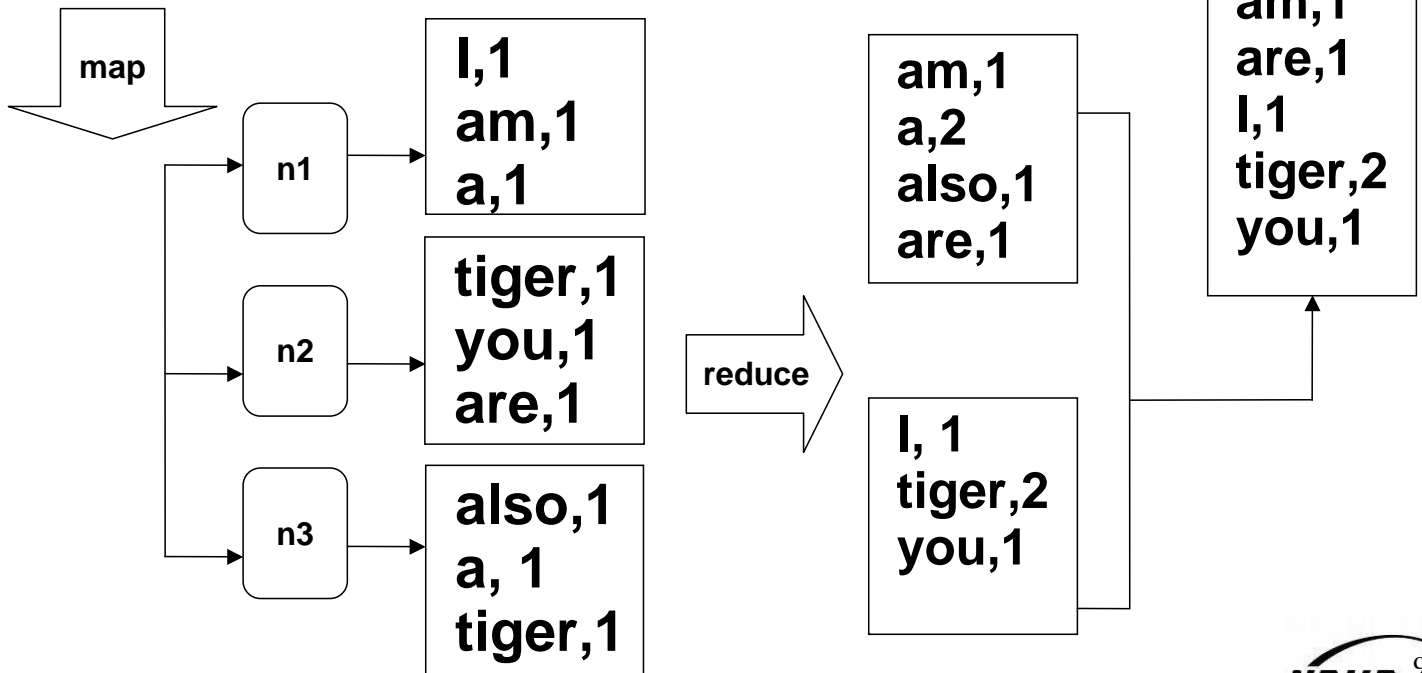
How does it work ?

MapReduce 運作流程



範例

I am a tiger, you are also a tiger



Streaming & Pipes

- 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



Map Reduce Programming

王耀聰 陳威宇

Jazz@nchc.org.tw

waue@nchc.org.tw

2008.04.27-28

國家高速網路與計算中心(NCHC)

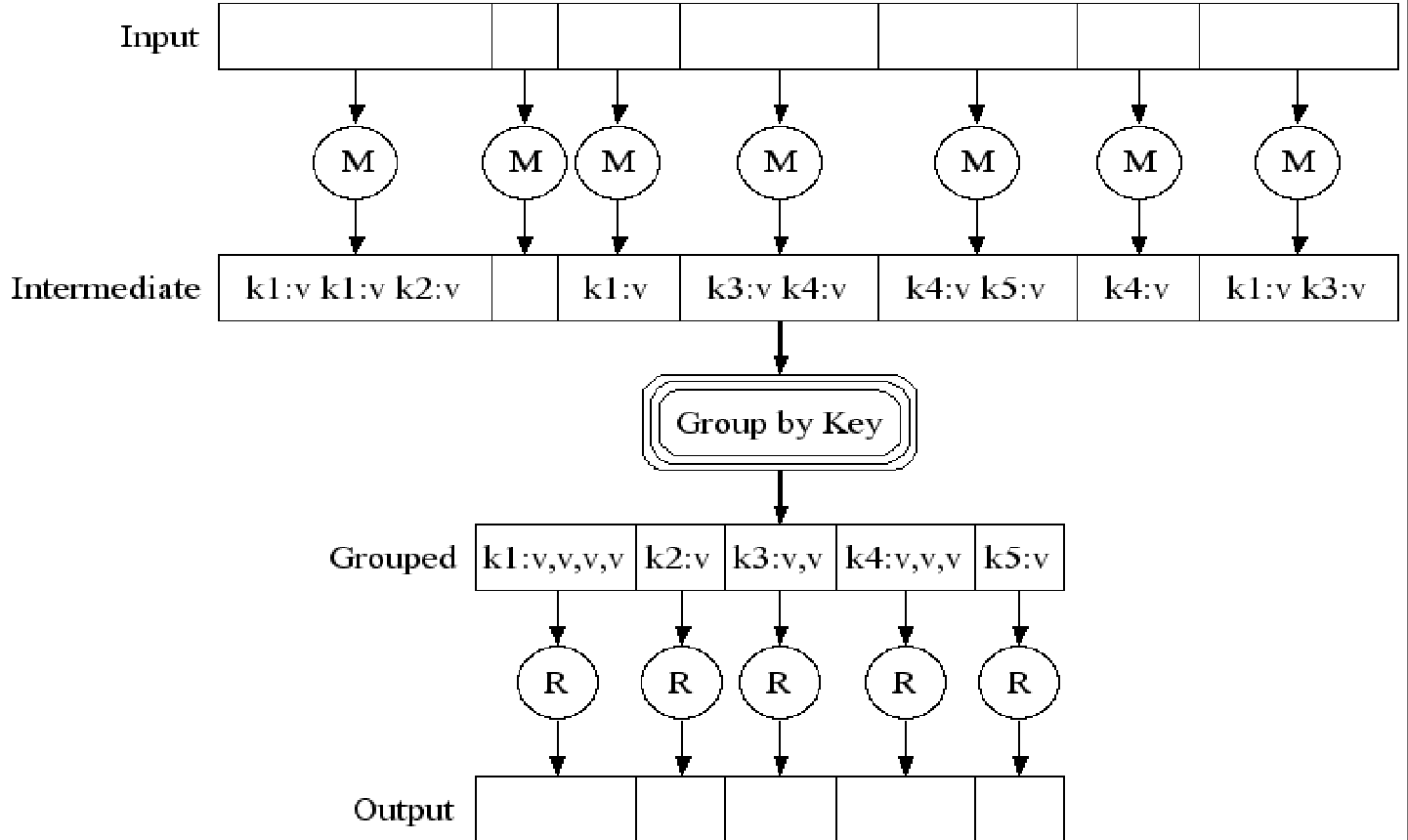
 自由軟體實驗室

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

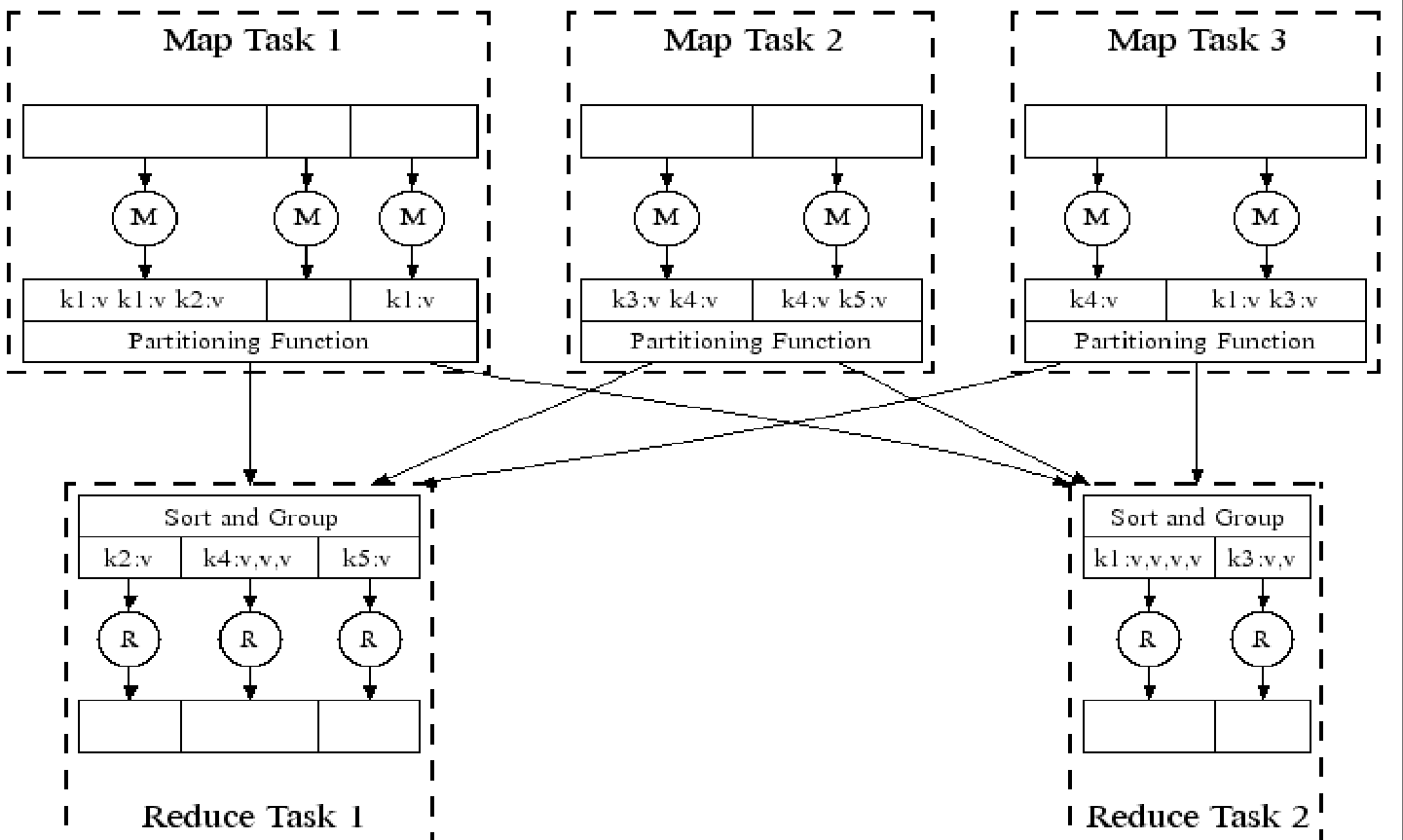
概念

MapReduce 圖解

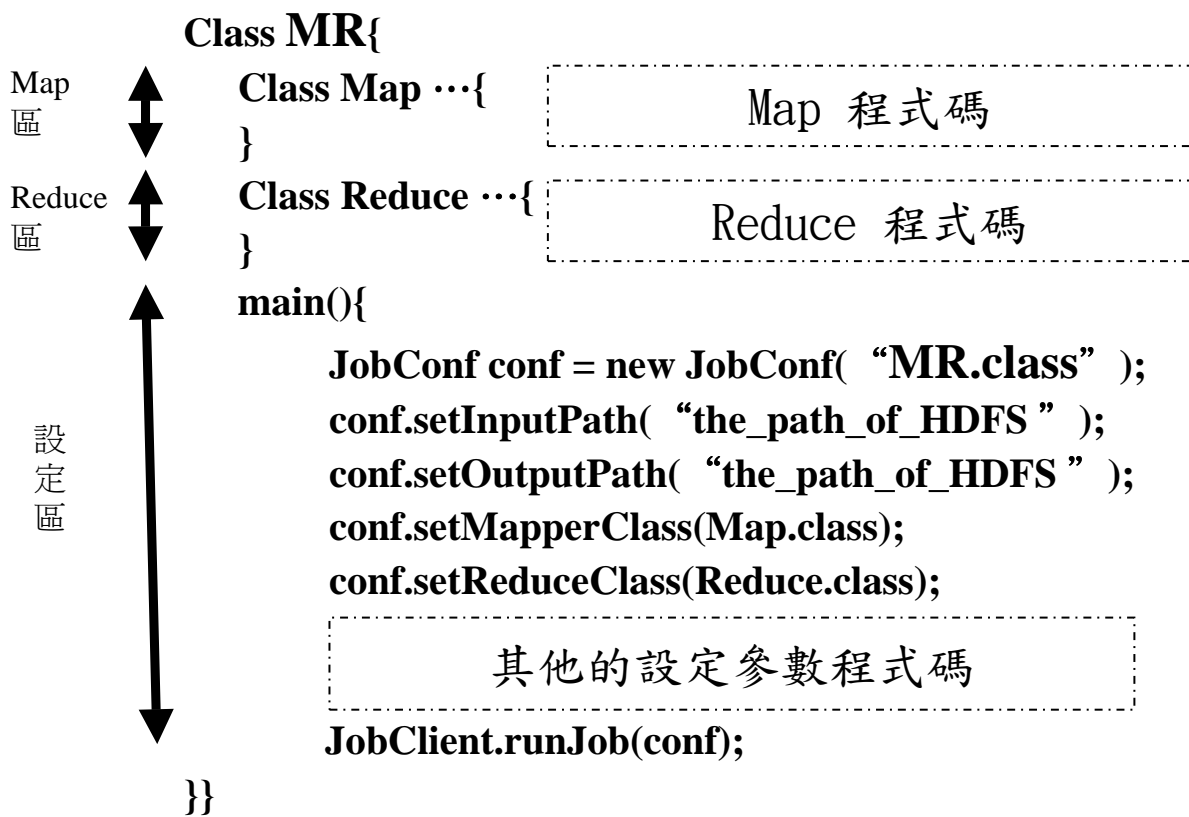


概念

MapReduce in Parallel



Program Prototype



Process Prototype

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
- `My jar. 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

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);  
        }  
    }  
}
```

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);  
    }  
}
```

Word Count Sample (3)

```
Class WordCount{
  main()
    JobConf conf = new JobConf(WordCount.class);
    conf.setJobName("wordcount");
    // set path
    conf.setInputPath(new Path(args[0]));
    conf.setOutputPath(new Path(args[1]));
    // set map reduce
    conf.setOutputKeyClass(Text.class); // set every word as key
    conf.setOutputValueClass(IntWritable.class); // set 1 as value
    conf.setMapperClass(MapClass.class);
    conf.setCombinerClass(Reduce.class);
    conf.setReducerClass(ReduceClass.class);
    onf.setInputFormat(TextInputFormat.class);
    conf.setOutputFormat(TextOutputFormat.class);
    // run
    JobClient.runJob(conf);
}
```

核心 Mapper

- <key/value > 的映射集合
- 設定
 - conf.setMapperClass(MapClass.class);
- 每次map的輸入
 - map (WritableComparable, Writable, OutputCollector, Reporter)
- map完後的輸出
 - OutputCollector.collect (WritableComparable, Writable)

核心Combiner

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

核心Reducer

- 將Map送來的<key/value>，對每個key作value的整合
- 輸入: <key, (list of values)>
 - `Reduce (WritableComparable, Iterator, OutputCollector, Reporter)`
- 輸出
 - `OutputCollector.collect(WritableComparable, Writable)`
- 若沒有Reduce要執行,可以不編寫

配置JobConf

- Hadoop程式架構內主要的執行設定類別
- 指定Mapper、Combiner、Partitioner、Reducer、InputFormat和OutputFormat的類別為何
- 指定輸入文件
 - setInputPaths(JobConf, Path...) / addInputPath(JobConf, Path)
- 指定輸出文件
 - setOutputPath(Path)
- debug script
 - setMapDebugScript(String) / setReduceDebugScript(String)
- 最多的嘗試次數
 - setMaxMapAttempts(int) / setMaxReduceAttempts(int)
- 容許任務失敗的百分比
 - setMaxMapTaskFailuresPercent(int) / setMaxReduceTaskFailuresPercent(int)
-

任務執行

- runJob(JobConf) :
 - 提交作業，僅當作業完成時返回。
- submitJob(JobConf) :
 - 只提交作業，之後需要你輪詢它返回的RunningJob句柄的狀態，並根據情況調度。
- JobConf.setJobEndNotificationURI(String) :
 - 設置一個作業完成通知，可避免輪詢。

WordCount練習 (前置)

1. `cd $HADOOP_HOME`
2. `bin/hadoop dfs -mkdir input`
3. `echo "I like NCHC Cloud Course." > input1`
4. `echo "I like nchc Cloud Course, and we enjoy this crouse." > input2`
5. `bin/hadoop dfs -put input1 input`
6. `bin/hadoop dfs -put input2 input`
7. `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$ █
```
8. 編輯WordCount.java
http://trac.nchc.org.tw/cloud/attachment/wiki/jazz/Hadoop_Lab6/WordCount.java?format=raw
9. `mkdir MyJava`



WordCount練習 (執行)

1. 編譯
 - `javac -classpath hadoop-*-core.jar -d MyJava WordCount.java`
2. 封裝
 - `jar -cvf wordcount.jar -C MyJava .`
3. 執行
 - `bin/hadoop jar wordcount.jar WordCount input/output/`

- 所在的執行目錄為Hadoop_Home
- ./MyJava = 編譯後程式碼目錄
- wordcount.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

WordCount練習 (執行)

```
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 arguments. 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

WordCount練習 (結果)

```
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
```



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 .`



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
```



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
```

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[])

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