



# 深入解析雲端大量資料分析技術

Part 3 : Deep Dive into Data Science Technologies

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



Powered by DRBL

# Open Source Mapping of Google Core Technologies

## Google 三大關鍵技術對應的自由軟體

### BigTable

A huge key-value datastore

HBase, Hypertable  
Cassandra, ....

### MapReduce

To parallel process data

Hadoop MapReduce API  
Sphere MapReduce API, ...

### Google File System

To store petabytes of data

Hadoop Distributed File System (HDFS)  
Sector Distributed File System

更多不同語言的 MapReduce API 實作：

<http://trac.nchc.org.tw/grid/intertrac/wiki%3Ajazz/09-04-14%23MapReduce>

其他值得觀察的分散式檔案系統：

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

# Building PaaS with Open Source

## 用自由軟體打造 PaaS 雲端服務

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

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

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

Hadoop (MapReduce),  
Sector/Sphere, AppScale

控制管理 Control  
Qos Negotiation, Admission Control,  
Pricing, SLA Management, Metering...

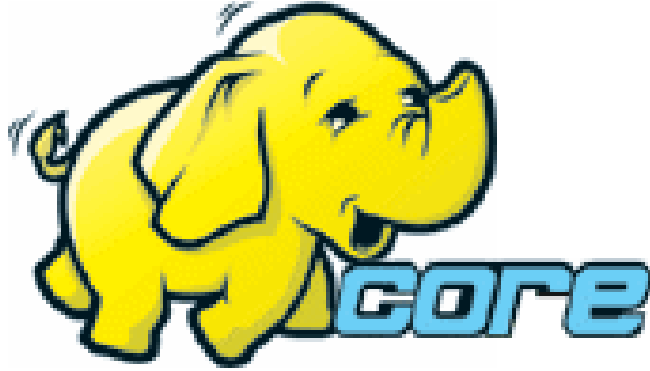
OpenNebula, Enomaly,  
Eucalyptus, OpenQRM, ...

虛擬化 Virtualization  
VM, VM management and Deployment

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

硬體設施 Hardware  
Infrastructure: Computer, Storage, Network

# Hadoop

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

# Sector / Sphere

- <http://sector.sourceforge.net/>
- 由美國資料探勘中心研發的自由軟體專案。
- **Developed by National Center for Data Mining, USA**
- 採用 C/C++ 語言撰寫，因此效能較 Hadoop 更好。
- **Written by C/C++, so performance is better than Hadoop**
- 提供「類似」Google File System 與 MapReduce 的機制
- **Provide file system similar to Google File System and MapReduce API**
- 基於UDT高效率網路協定來加速資料傳輸效率
- **Based on UDT which enhance the network performance**
- Open Cloud Testbed有提供測試環境，並開發Ma1Stone效能評比軟體
- **Open Cloud Consortium provide Open Cloud Testbed and develop Ma1Stone toolkit for benchmark**

**Sector-Sphere**

National Center for Data Mining  
University of Illinois at Chicago



Open Data Group

<http://www.opendatagroup.com/>

# Why should we learn Hadoop ?

## 為何需要學習 **Hadoop ??**

[Search Jobs](#) [Browse Jobs](#) [Local Jobs](#) [Salaries](#) [Employment Trends](#)

**simply**hired<sup>®</sup>

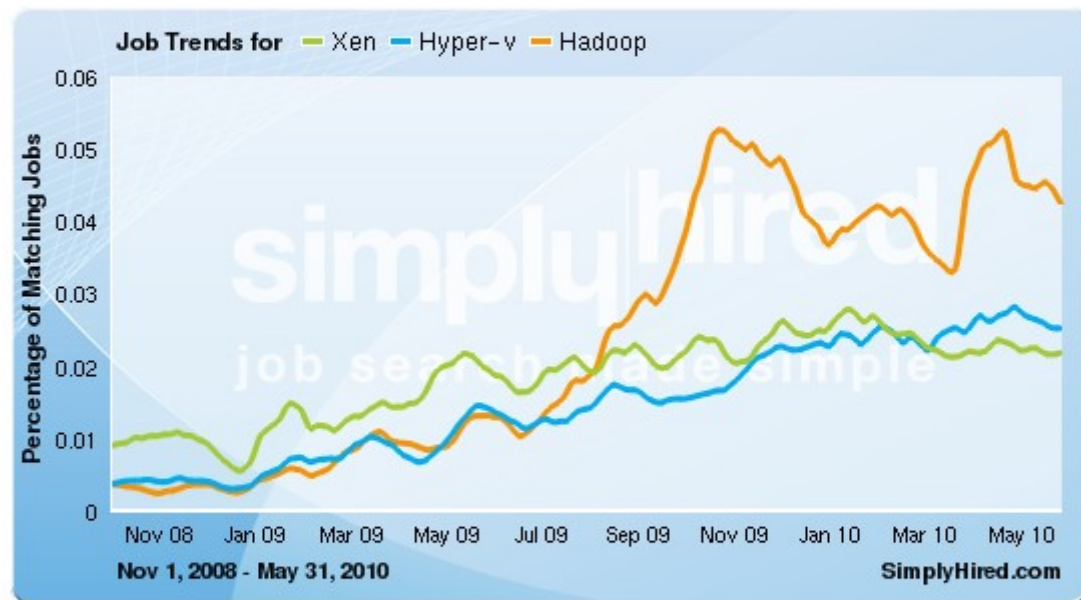
job search made simple

Employment Trends

Xen, Hyper-V, Hadoop

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

Xen, Hyper-v, Hadoop Trends



### Xen, Hyper-v, Hadoop Job Trends

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

- [Xen jobs](#) increased 141%
- [Hyper-v jobs](#) increased 551%
- [Hadoop jobs](#) did not change or there is no data available

## 1. Data Explore

資訊大爆炸

## 2. Data Mining Tool

方便作資料探勘的工作

## 3. Looking for Jobs

好找工作 !!

# Comparison between Google and Hadoop

## Google 與 Hadoop 的比較表

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

# What is Hadoop ?

用一句話解釋 **Hadoop** 是什麼??

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

*Hadoop* 是一個讓使用者簡易撰寫並執行**處理海量資料**應用程式的**軟體平台**。

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



# *Features of Hadoop ...*

## **Hadoop 這套軟體的特色是 ...**

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

# *Founder of Hadoop – Doug Cutting*

**Hadoop** 這套軟體的創辦人 **Doug Cutting**

Doug Cutting Talks About The Founding Of Hadoop

clouderahadoop

9 部影片

編輯訂閱項目

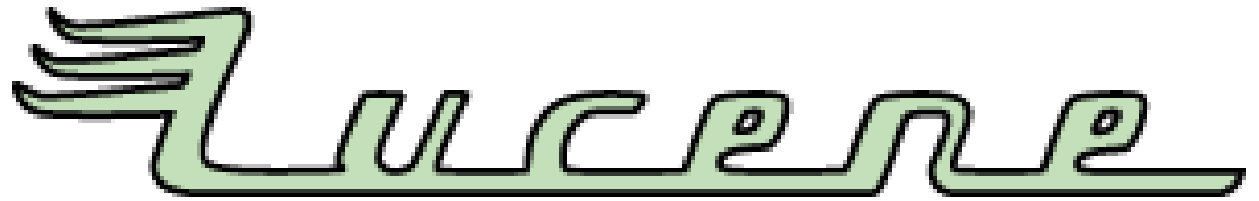


Doug Cutting Talks About The Founding Of Hadoop

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

# History of Hadoop ... 2002~2004

## Hadoop 這套軟體的歷史源起 ... 2002~2004



- Lucene

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

# History of Hadoop ... 2002~2004

## Hadoop 這套軟體的歷史源起 ... 2002~2004

- Nutch



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



# Three Gifts from Google ....

## 來自 Google 的三個禮物 ....

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



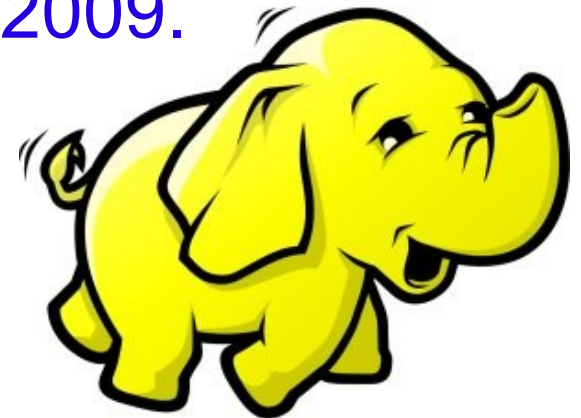
# ***History of Hadoop ... 2004 ~ Now***

## ***Hadoop 這套軟體的歷史源起 ... 2004 ~ Now***

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

**YAHOO!**

 **cloudera**



# Who Use Hadoop ??

## 有哪些公司在用 **Hadoop** 這套軟體 ??

- **Yahoo** is the key contributor currently.
- **IBM** and **Google** teach Hadoop in universities ...
- [http://www.google.com/intl/en/press/pressrel/20071008\\_ibm\\_univ.html](http://www.google.com/intl/en/press/pressrel/20071008_ibm_univ.html)
- **The New York Times** used **100 Amazon EC2 instances** and a Hadoop application to process **4TB of raw image TIFF data** (stored in S3) into **11 million finished PDFs** in the space of **24 hours** at a computation cost of about **\$240** (not including bandwidth)
  - from <http://en.wikipedia.org/wiki/Hadoop>
- <http://wiki.apache.org/hadoop/AmazonEC2>
- <http://wiki.apache.org/hadoop/PoweredBy>
  - A9.com
  - ADSDAQ by Contextweb
  - EHarmony
  - Facebook
  - Fox Interactive Media
  - IBM
  - ImageShack
  - ISI
  - Joost
  - Last.fm
  - Powerset
  - The New York Times
  - Rackspace
  - Veoh
  - Metaweb

# Performance improvement of Hadoop

## Hadoop 過去幾年的效能改進 (from Yahoo)

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 in production run ....*

## 商業運轉中的 **Hadoop** 應用 ....

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

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

# *Hadoop in production run ....*

## 商業運轉中的 **Hadoop** 應用 ....

- **September 30, 2008**
- **Scaling Hadoop to 4000 nodes at Yahoo!**
- [http://developer.yahoo.net/blogs/hadoop/2008/09/scaling\\_hadoop\\_to\\_4000\\_nodes\\_a.html](http://developer.yahoo.net/blogs/hadoop/2008/09/scaling_hadoop_to_4000_nodes_a.html)

<b>Total Nodes</b>	<b>4000</b>
<b>Total cores</b>	<b>30000</b>
<b>Data</b>	<b>16PB</b>

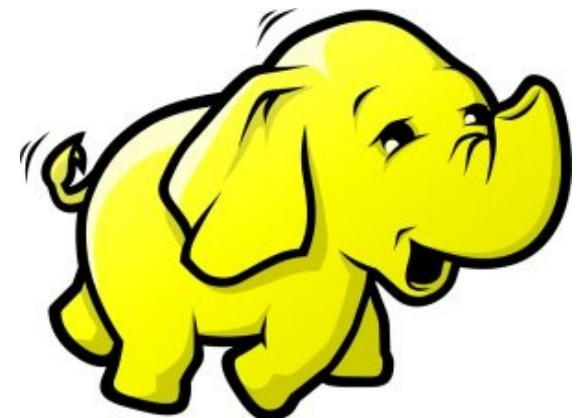
	<b>500-node cluster</b>		<b>4000-node cluster</b>	
	<b>write</b>	<b>read</b>	<b>write</b>	<b>read</b>
<b>number of files</b>	990	990	14,000	14,000
<b>file size (MB)</b>	320	320	360	360
<b>total MB processes</b>	316,800	316,800	5,040,000	5,040,000
<b>tasks per node</b>	2	2	4	4
<b>avg. throughput (MB/s)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



# Hadoop 專業術語

*Introduction to Hadoop Terminology*

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



# Two Key Elements of Operating System

## 作業系統兩大關鍵組成元素

**Scheduler**  
程序排程



**File System**  
檔案系統



# Terminologies of Hadoop

## Hadoop 文件中的專業術語

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



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



# *Two Key Roles of HDFS*

## **HDFS** 軟體架構的兩種關鍵角色

### 名稱節點 **NameNode**

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

### 資料節點 **DataNode**

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

# *Two Key Roles of Job Scheduler*

## *程序排程的兩種關鍵角色*

### *JobTracker*

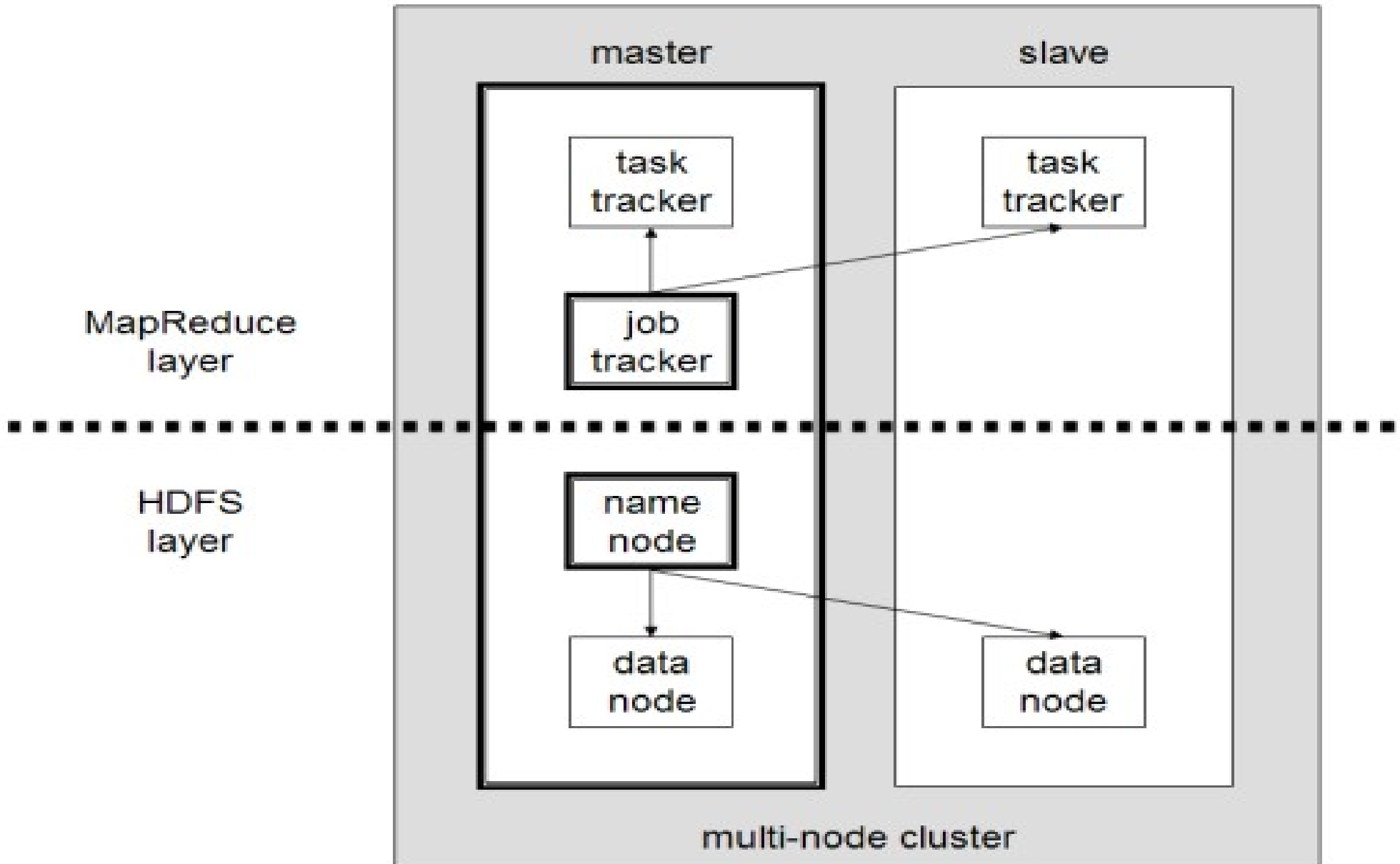
- **Master Node**
- **Receive Jobs from Hadoop Clients**
- **Assigned Tasks to TaskTrackers**
- **Define Job Queuing Policy, Priority and Error Handling**
- **Single Point of Failure**

### *TaskTracker*

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

# *Different Roles of Hadoop Architecture*

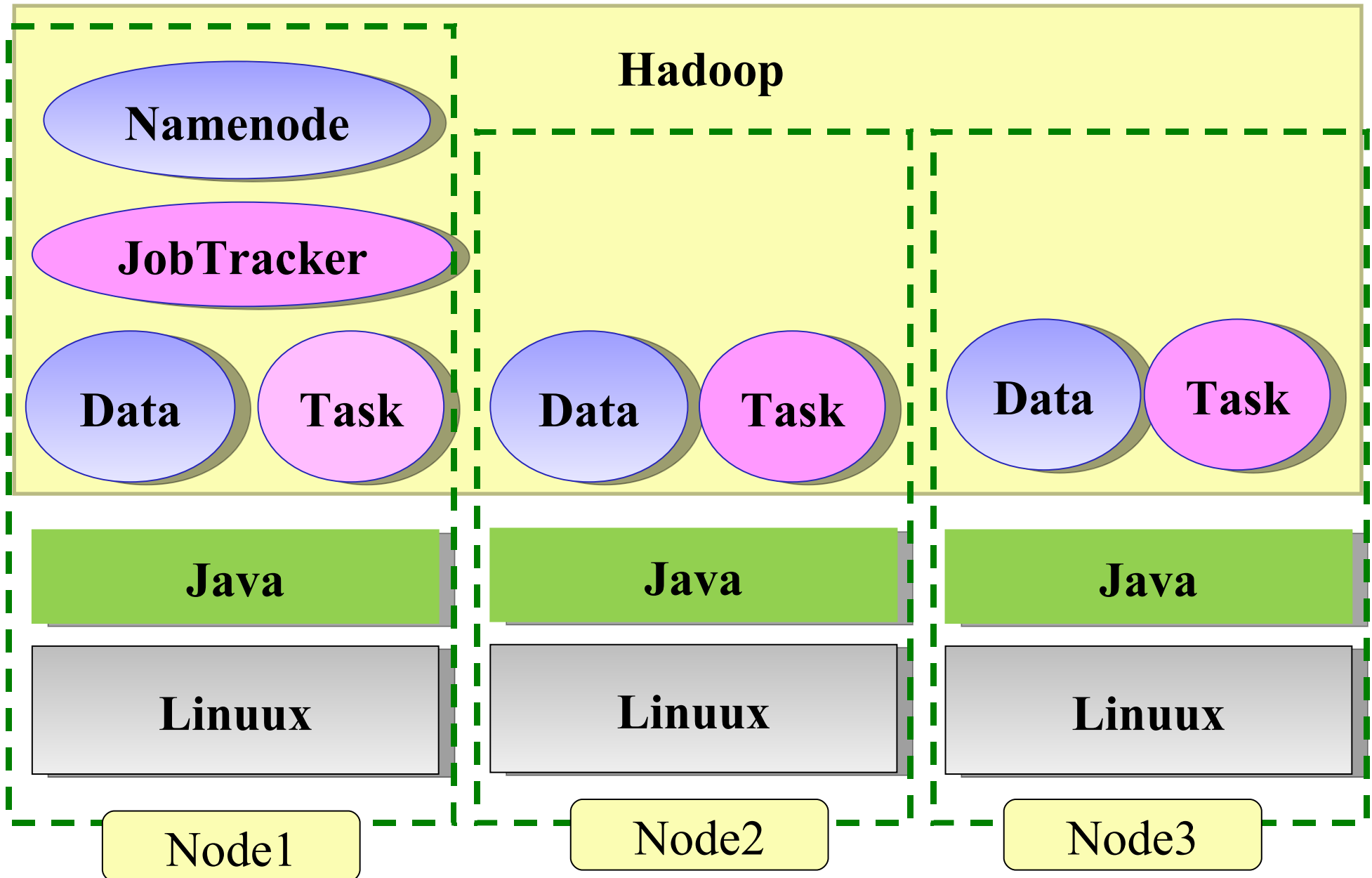
## **Hadoop** 軟體架構中的不同角色





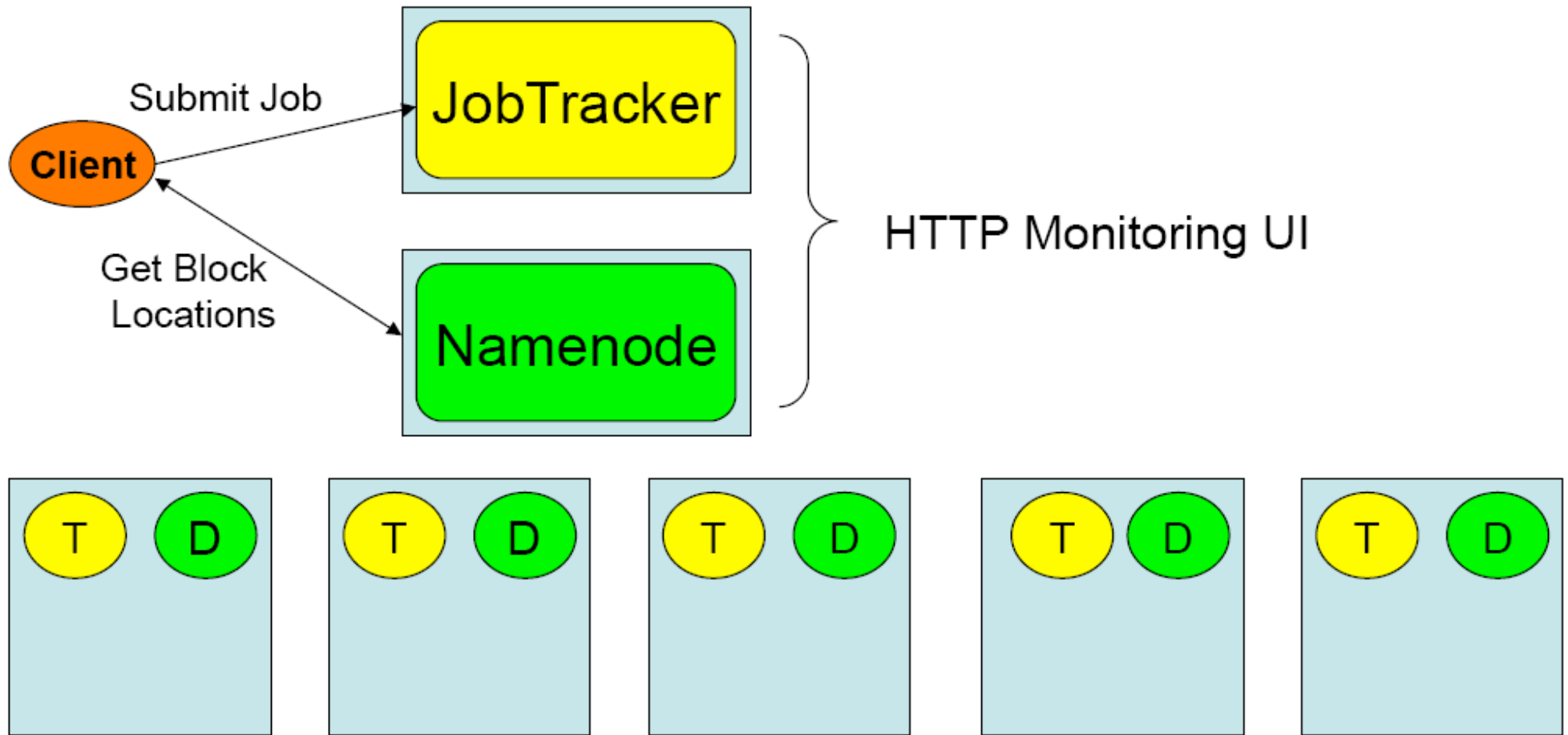
# Distributed Operating System of Hadoop

**Hadoop** 建構成一個分散式作業系統



# About Hadoop Client ...

## 不在雲裡的 **Hadoop Client**

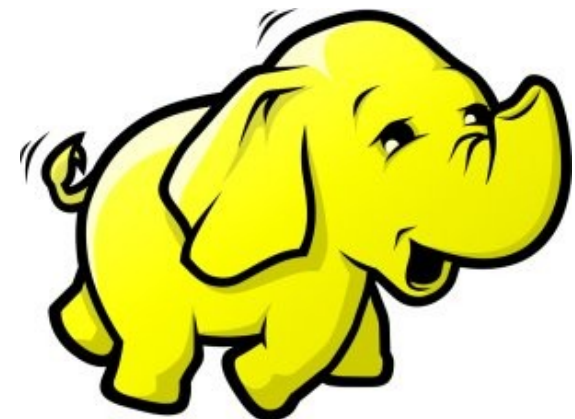




# HDFS 簡介

*Introduction to Hadoop Distributed File System*

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



# ***What is HDFS ??***

## ***什麼是 HDFS ??***

- **Hadoop Distributed File System**

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

# Features of HDFS ...

## HDFS 的特色是 ...

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

# Features of HDFS ...

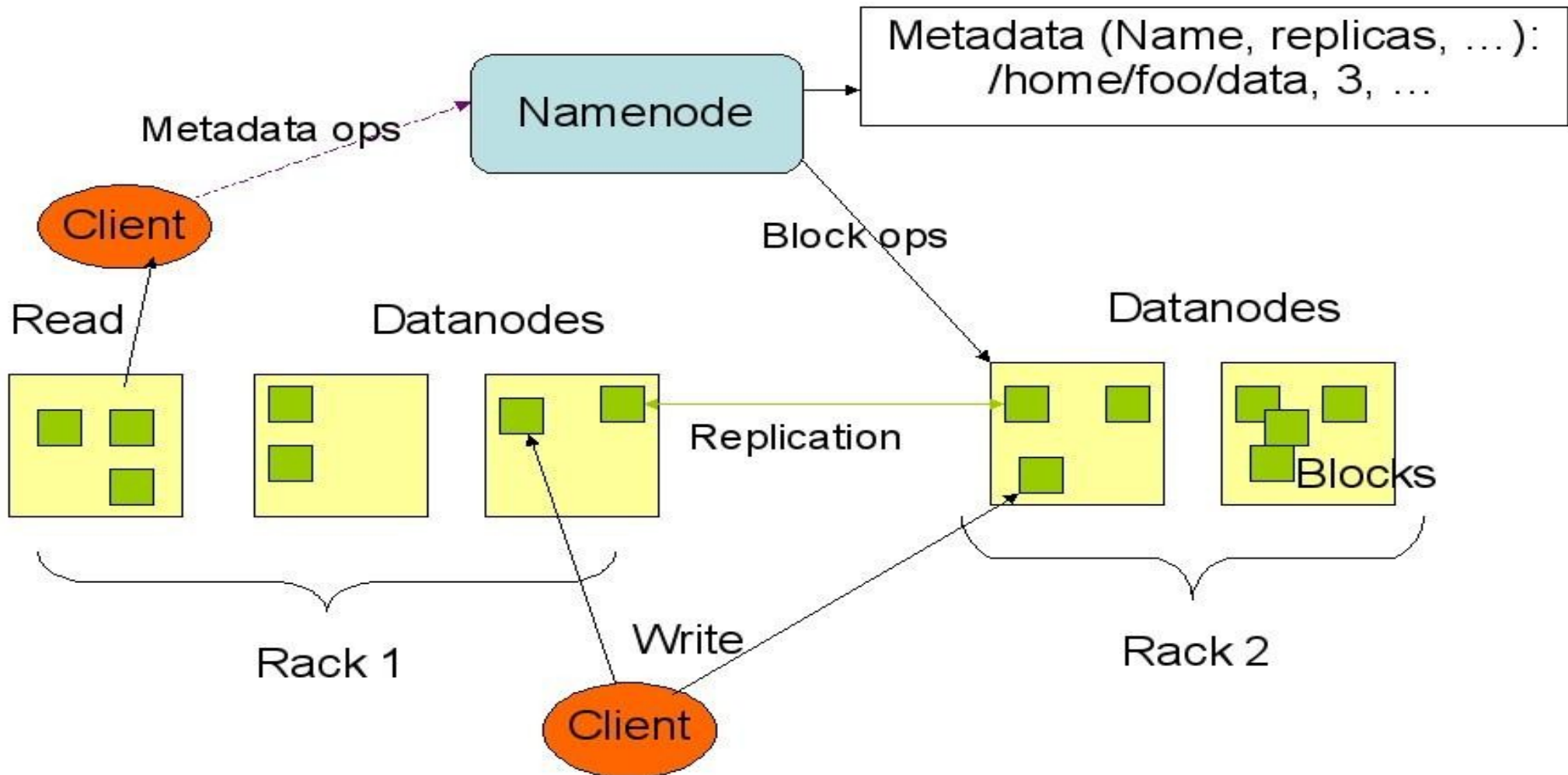
## HDFS 的特色是 ...

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

# How HDFS manage data ...

## HDFS 如何管理資料 ...

HDFS Architecture



# How does HDFS work ...

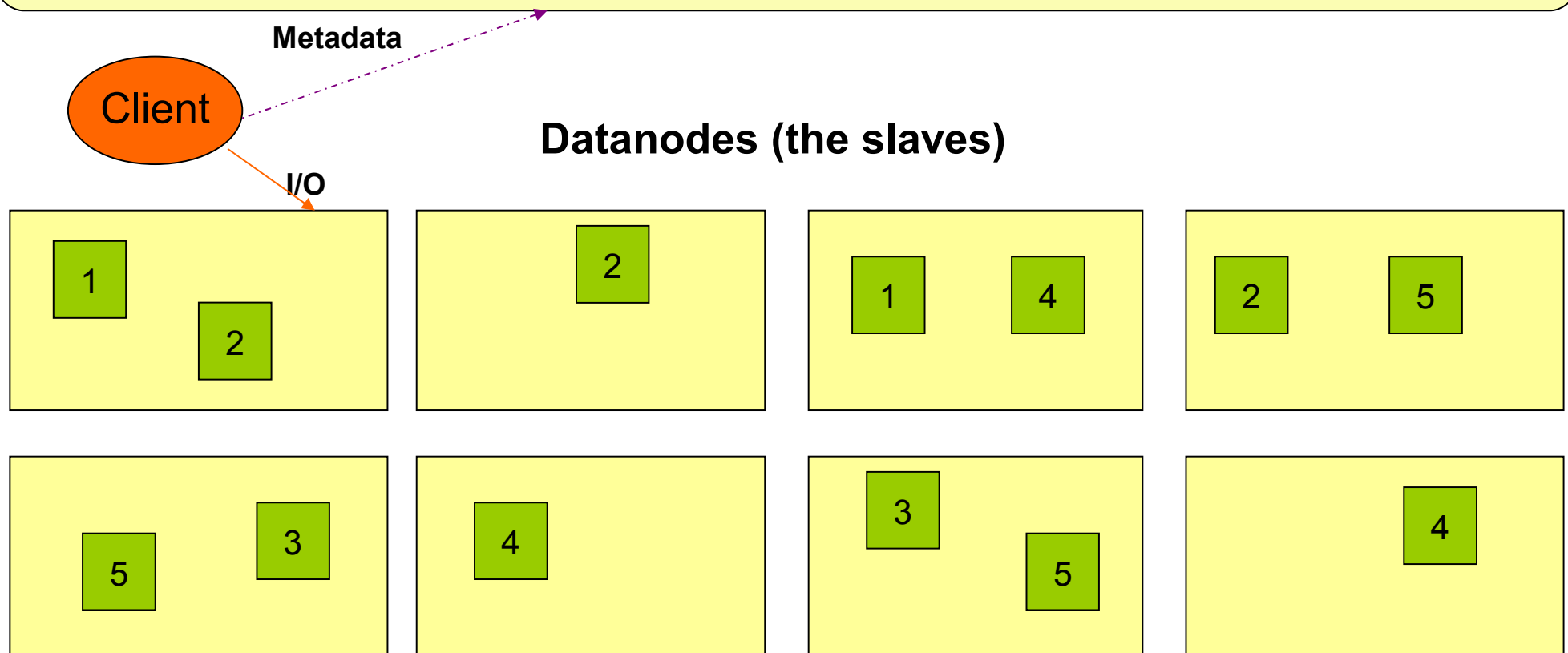
## HDFS 如何運作 ...

Namenode (the master)

Path and Filename – Replication , blocks

name:/users/joeYahoo/myFile - copies:2, blocks:{1,3}

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

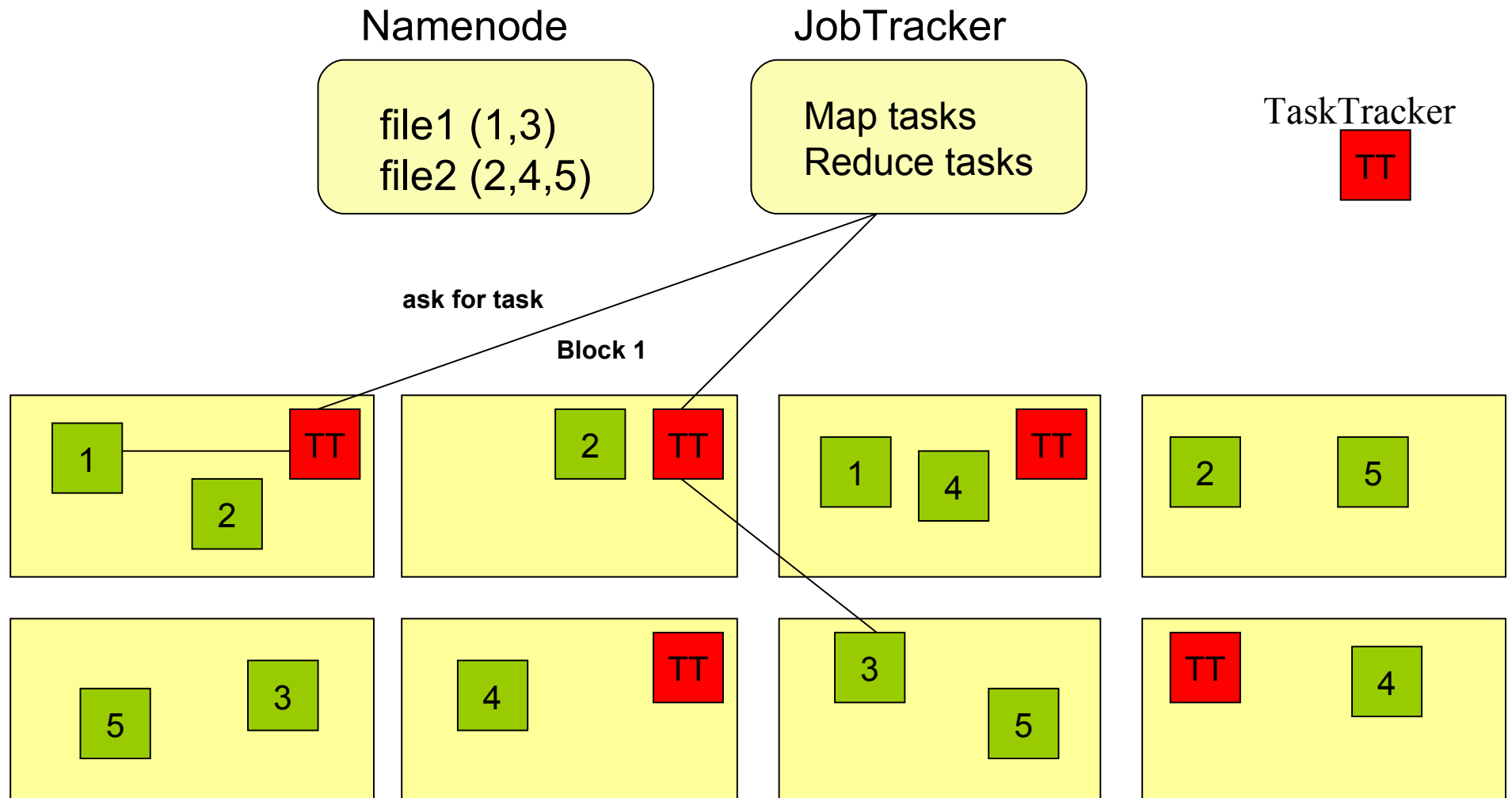




# About Data Locality ...

## HDFS 如何達成在地運算 ...

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



# About Fault Tolerance ...

## HDFS 如何達成容錯機制 ...

資料崩毀  
Data Corrupt

網路或資料  
節點失效  
Network Fault  
DataNode Fault

名稱節點錯誤  
NameNode Fault

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

# Coherency Model and Performance of HDFS

## HDFS 的一致性機制與效能 ...

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

# POSIX like HDFS commands

## 與 **POSIX** 相似的操作指令 ...

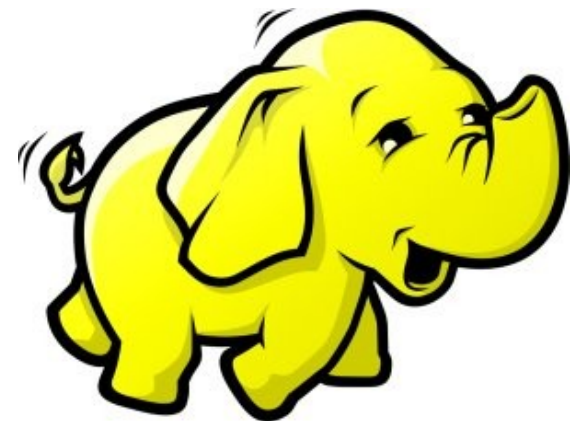
```
jazz@hadoop:~$ hadoop fs
Usage: java FsShell
    [-ls <path>]
    [-lsr <path>]
    [-du <path>]
    [-dus <path>]
    [-count[-q] <path>]
    [-mv <src> <dst>]
    [-cp <src> <dst>]
    [-rm <path>]
    [-rmr <path>]
    [-expunge]
    [-put <localsrc> ... <dst>]
    [-copyFromLocal <localsrc> ...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



# MapReduce 簡介

*Introduction to MapReduce*

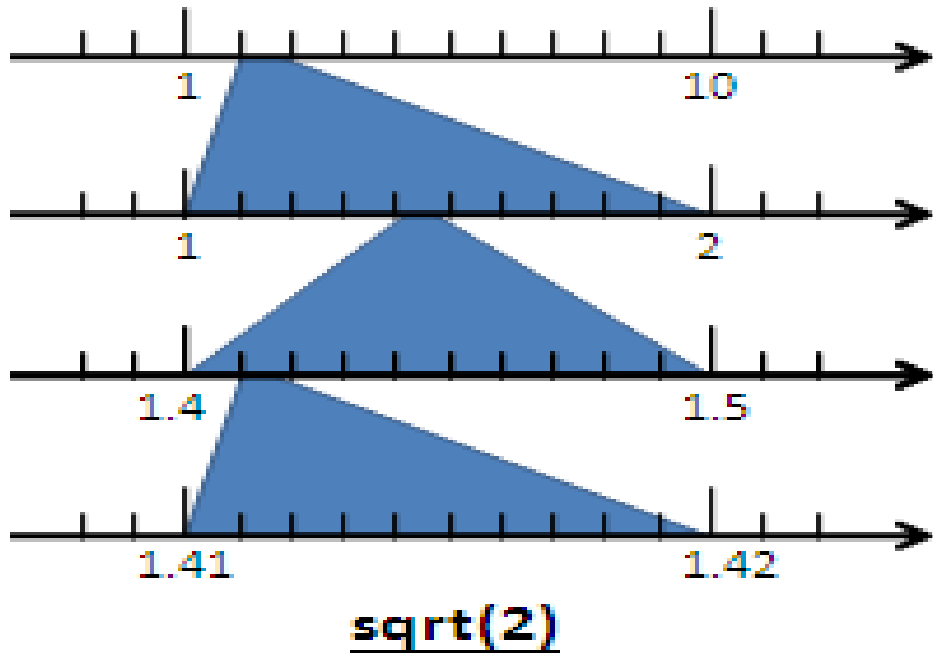
***Jazz Wang***  
***Yao-Tsung Wang***  
***jazz@nchc.org.tw***



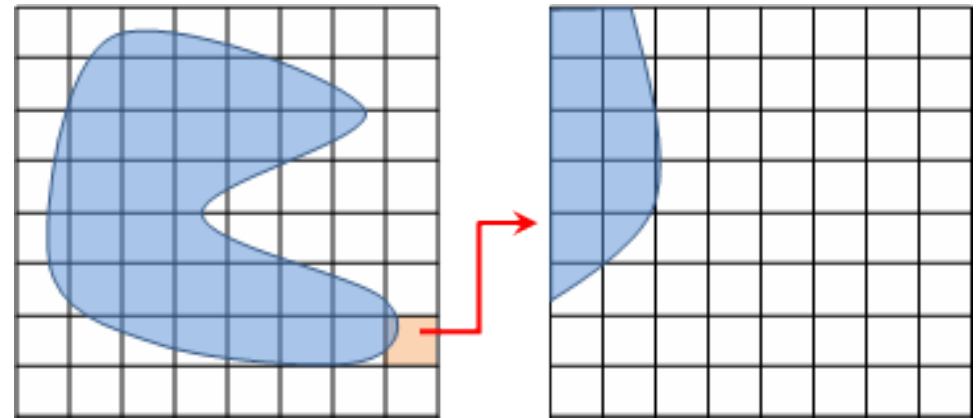
# Divide and Conquer Algorithms

## 分而治之演算法

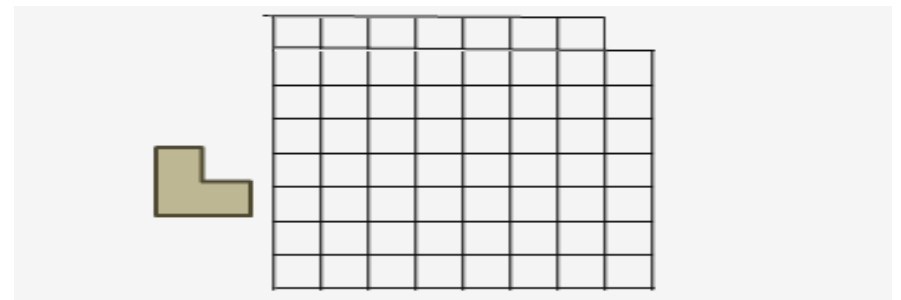
Example 1:



Example 2:



Example 3:



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

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

# What is MapReduce ??

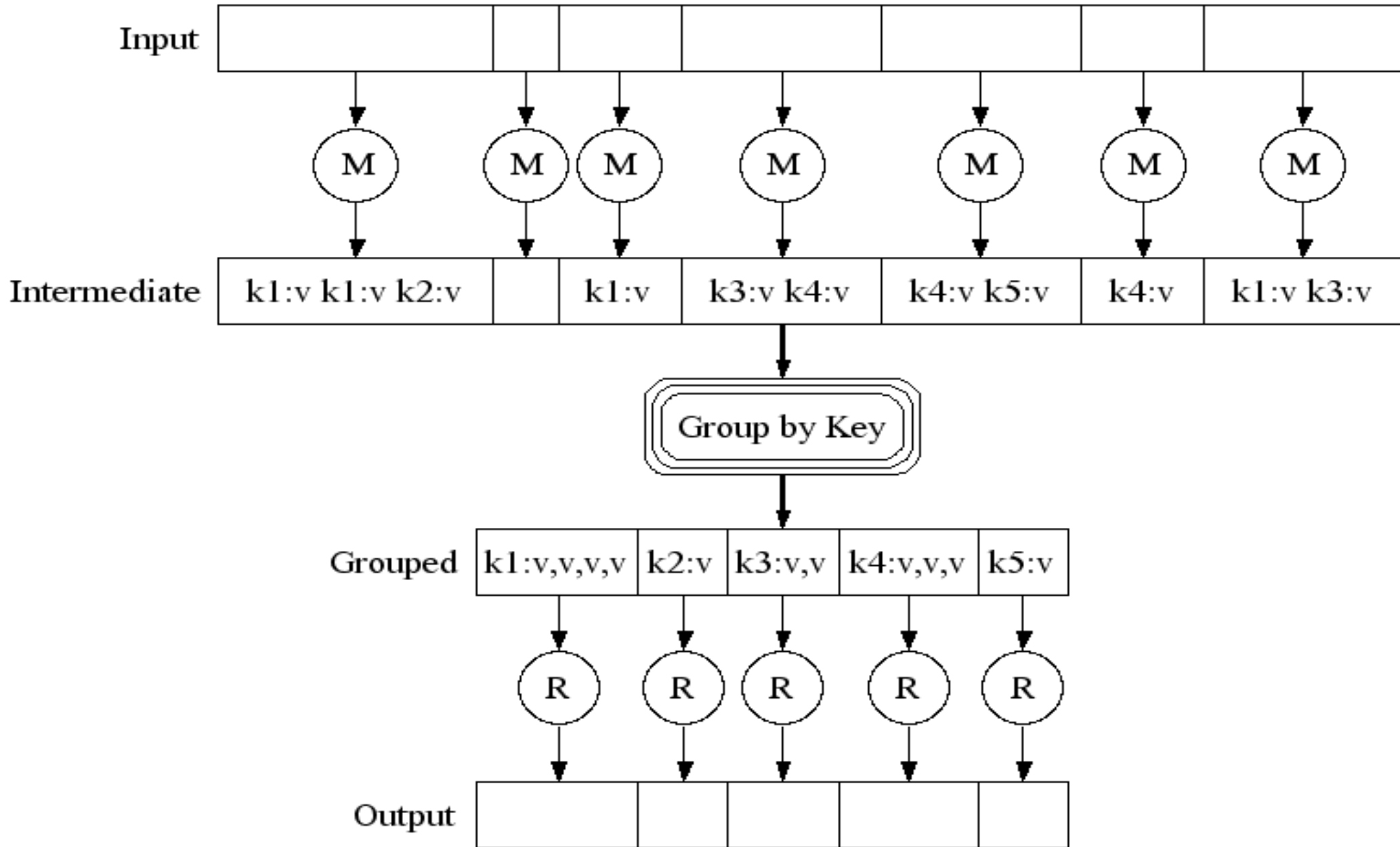
## 什麼是 MapReduce ??

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

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

# Google's MapReduce Diagram

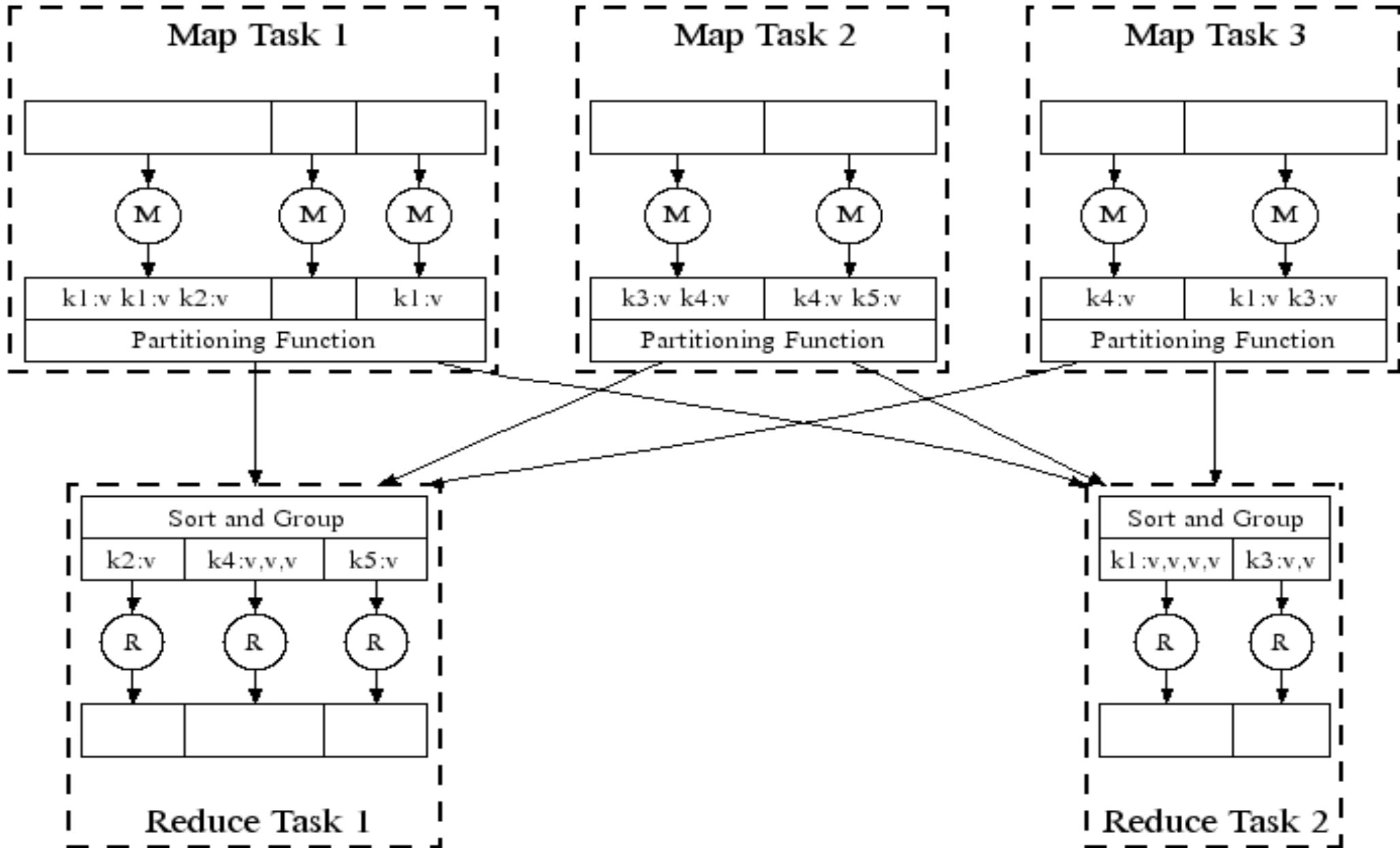
## Google 的 MapReduce 圖解





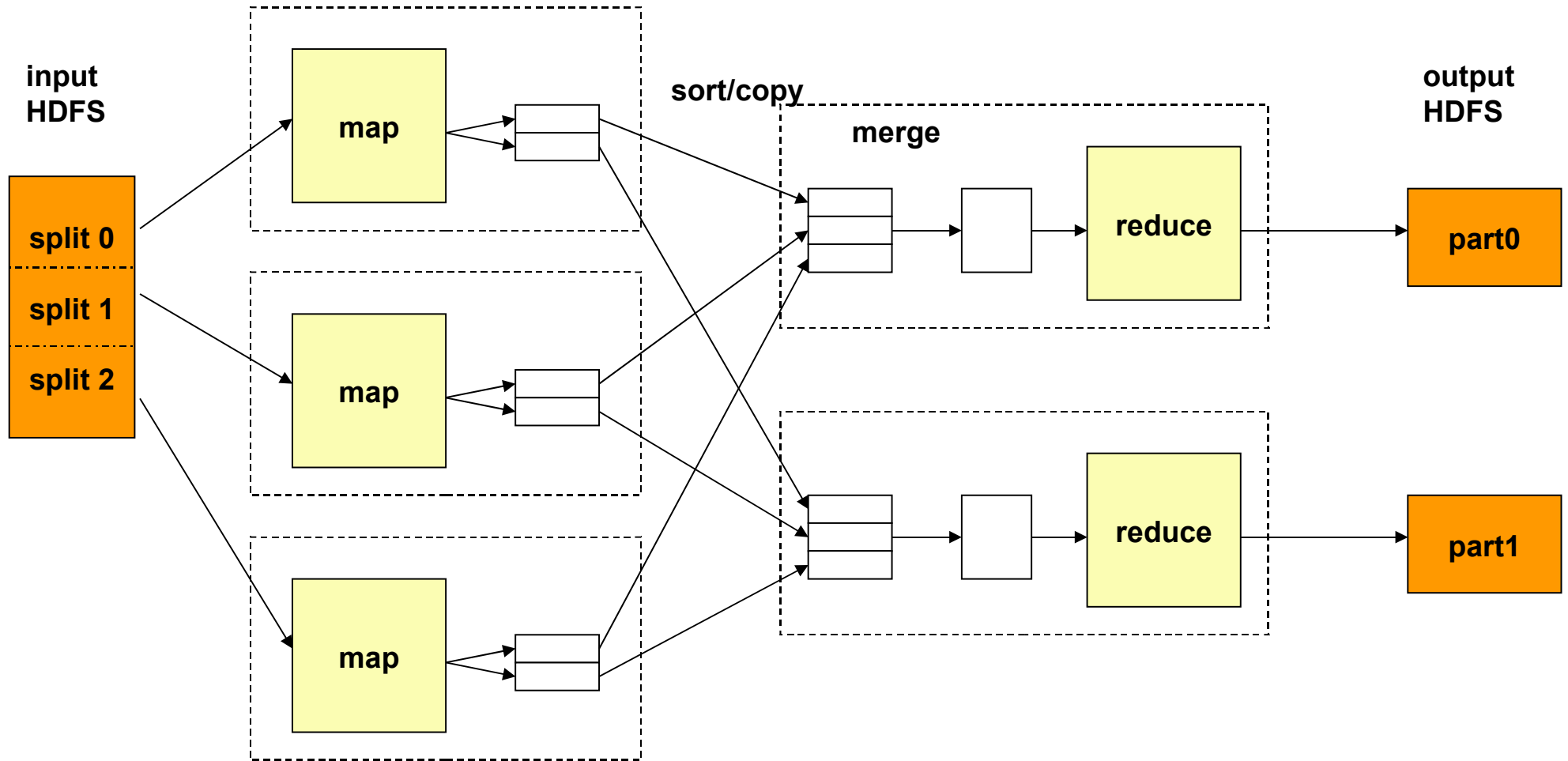
# Google's MapReduce in Parallel

## Google 的 MapReduce 平行版圖解



# How does MapReduce work in Hadoop

## Hadoop MapReduce 運作流程



JobTracker 跟 NameNode 取得需要運算的 blocks

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

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

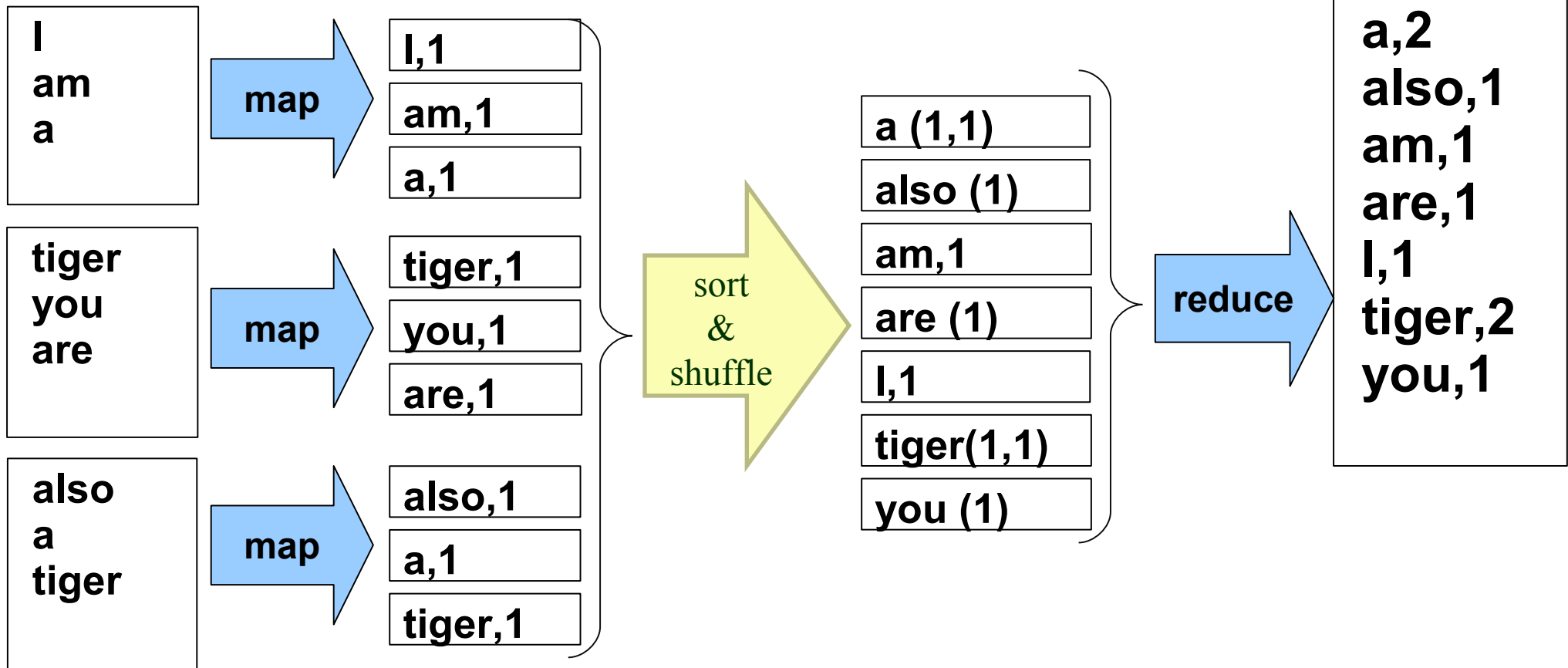
JobTracker 派遣 TaskTracker 作 reduce

reduce 完後通知 JobTracker 與 Namenode 以產生 output

# MapReduce by Example (1)

## MapReduce 運作實例 (1)

I am a tiger, you are also a tiger



JobTracker 先選了三個 Tracker 做 map

Map 結束後，hadoop 進行中間資料的重組與排序

JobTracker 再選一個 TaskTracker 作 reduce

# MapReduce by Example (2)

## MapReduce 運作實例 (2)

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

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

Input File

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

map

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

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

map

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

sort /  
merge

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

reduce

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

# *MapReduce is suitable to ....*

## **MapReduce** 合適用於 ....

- 大規模資料集
- **Large Data Set**
- 可拆解
- **Parallelization**
- Text tokenization
- Indexing and Search
- Data mining
- machine learning
- ...

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

# What we learn today ?

## WHAT

**Hadoop 是運算海量資料的軟體平台 !!**

hadoop is a software platform to process vast amount of data!!

## WHO

始祖是 Doug Cutting，Apache 社群支持，Yahoo 贊助

From Doug Cutting to Apache Community, Yahoo and more !

## WHEN

**Hadoop 是 2004 年從 Nutch 分裂出來的專案 !!**

Hadoop became separate project since year 2004 !!

## WHY

資料大爆炸、資料探勘、找工作

***Data Explore, Data Mining, Jobs !!***

## HOW

採用自由軟體也能打造私有雲端

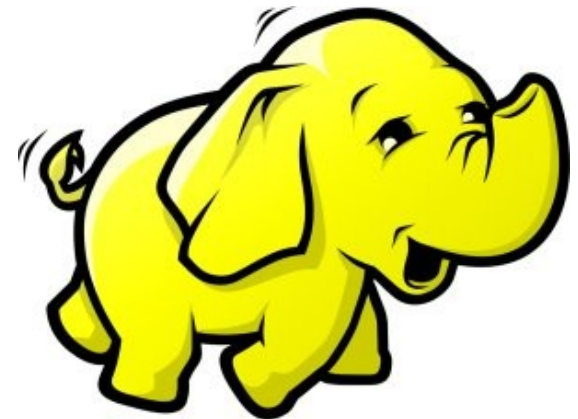
Install on large clusters built of commodity hardware !!



# Hadoop 相關計畫

*Hadoop Ecosystem*

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





**Hadoop** 只支援用 **Java** 開發嘛？  
***Is Hadoop only support Java ?***

總不能全部都重新設計吧？如何與舊系統相容？

***Can Hadoop work with existing software ?***



可以跟資料庫結合嘛？

***Can Hadoop work with Databases ?***

開發者們有聽到大家的需求.....

***Yes, we hear the feedback of developers ...***





# Is Hadoop only support Java ?

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

# Hadoop Pipes (C++, Python)

- Hadoop Pipes allows **C++** code to use Hadoop DFS and map/reduce.
- The C++ interface is "swigable" so that interfaces can be generated for **python** and other scripting languages.
- For more detail, check the API Document of [org.apache.hadoop.mapred.pipes](http://org.apache.hadoop.mapred.pipes)
- You can also find example code at [hadoop-\\*/src/examples/pipes](http://hadoop-*/src/examples/pipes)
- About the pipes C++ WordCount example code: <http://wiki.apache.org/hadoop/C++WordCount>

# Hadoop Streaming

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

# Running Hadoop Streaming

```
jazz@hadoop:~$ hadoop jar hadoop-streaming.jar -help
10/08/11 00:20:00 ERROR streaming.StreamJob: Missing required option -input
Usage: $HADOOP_HOME/bin/hadoop [--config dir] jar \
    $HADOOP_HOME/hadoop-streaming.jar [options]
```

Options:

```
-input      <path>          DFS input file(s) for the Map step
-output     <path>          DFS output directory for the Reduce step
-mapper     <cmd|JavaClassName> The streaming command to run
-combiner   <JavaClassName> Combiner has to be a Java class
-reducer    <cmd|JavaClassName> The streaming command to run
-file       <file>         File/dir to be shipped in the Job jar file
-dfs        <h:p>|local    Optional. Override DFS configuration
-jt         <h:p>|local    Optional. Override JobTracker configuration
-additionalconfspec specfile Optional.
-inputformat TextInputFormat(default) |SequenceFileAsTextInputFormat |
JavaClassName Optional.
-outputformat TextOutputFormat(default) |JavaClassName Optional.
```

... More ...

# Hadoop Streaming with shell commands (1)

```
hadoop:~$ hadoop fs -rmr input output
```

```
hadoop:~$ hadoop fs -put /etc/hadoop/conf input
```

```
hadoop:~$ hadoop jar hadoop-streaming.jar -input  
input -output output -mapper /bin/cat -reducer /  
usr/bin/wc
```

# Hadoop Streaming with shell commands (2)

```
hadoop:~$ echo "sed -e \"s/ /\n/g\" | grep ." >  
streamingMapper.sh
```

```
hadoop:~$ echo "uniq -c | awk '{print \$2 \"\t\"  
\$1}'" > streamingReducer.sh
```

```
hadoop:~$ chmod a+x streamingMapper.sh
```

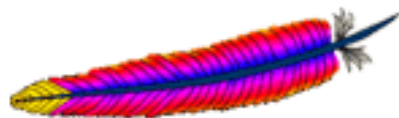
```
hadoop:~$ chmod a+x streamingReducer.sh
```

```
hadoop:~$ hadoop fs -put /etc/hadoop/conf input
```

```
hadoop:~$ hadoop jar hadoop-streaming.jar -input  
input -output output -mapper streamingMapper.sh  
-reducer streamingReducer.sh -file  
streamingMapper.sh -file streamingReducer.sh
```

# There are several Hadoop subprojects

Apache > Hadoop >



Top

Common

Chukwa

HBase

HDFS

Hive

MapReduce

Pig

ZooKeeper

▼ About

▫ Welcome

▫ Who We Are?

▫ Mailing Lists

## Welcome to Apache Hadoop!

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

## Other Hadoop related projects

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



# Hadoop Ecosystem

<b><i>Pig</i></b>	<b><i>Chukwa</i></b>	<b><i>Hive</i></b>	<b><i>HBase</i></b>
<b><i>MapReduce</i></b>		<b><i>HDFS</i></b>	<b><i>ZooKeeper</i></b>
<b><i>Hadoop Core (Hadoop Common)</i></b>		<b><i>Avro</i></b>	

Source: *Hadoop: The Definitive Guide*

# Avro

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



# Zoo Keeper



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

# Pig

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



# Hive

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



# Chukwa

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



# Mahout

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

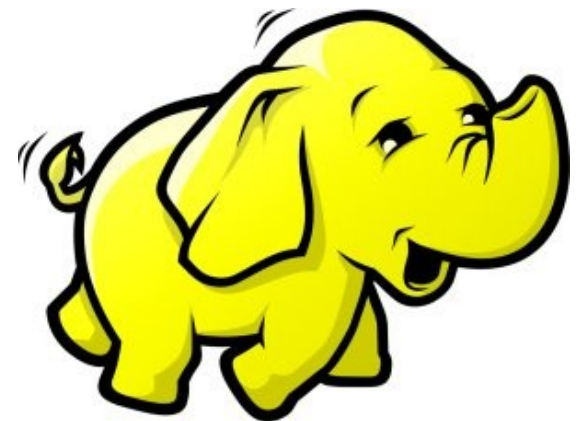




# HBase 雲端資料庫

*Introduction to HBase*

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





# It's all about SCALE!!



**Warning:** fopen(/home/dodgers/public\_html/./logs/oracle\_error\_log.txt) [function.fopen]: failed to open stream: Permission denied in /usr/local/apache/htdocs/include2007/oracle/db\_oracle.inc.php on line 194

Cannot open Database Error Log, please check!! (/home/dodgers/public\_html/./logs/oracle\_error\_log.txt)

**Warning:** fopen(/home/dodgers/public\_html/./logs/oracle\_error\_log.txt) [function.fopen]: failed to open stream: Permission denied in /usr/local/apache/htdocs/include2007/oracle/db\_oracle.inc.php on line 194

Cannot open Database Error Log, please check!! (/home/dodgers/public\_html/./logs/oracle\_error\_log.txt)

**Warning:** fopen(/home/dodgers/public\_html/./logs/oracle\_error\_log.txt) [function.fopen]: failed to open stream: Permission denied in /usr/local/apache/htdocs/include2007/oracle/db\_oracle.inc.php on line 194

Cannot open Database Error Log, please check!! (/home/dodgers/public\_html/./logs/oracle\_error\_log.txt)

**Warning:** fopen(/home/dodgers/public\_html/./logs/oracle\_error\_log.txt) [function.fopen]: failed to open stream: Permission denied in /usr/local/apache/htdocs/include2007/oracle/db\_oracle.inc.php on line 194

Cannot open Database Error Log, please check!! (/home/dodgers/public\_html/./logs/oracle\_error\_log.txt)



訂購歷史紀錄

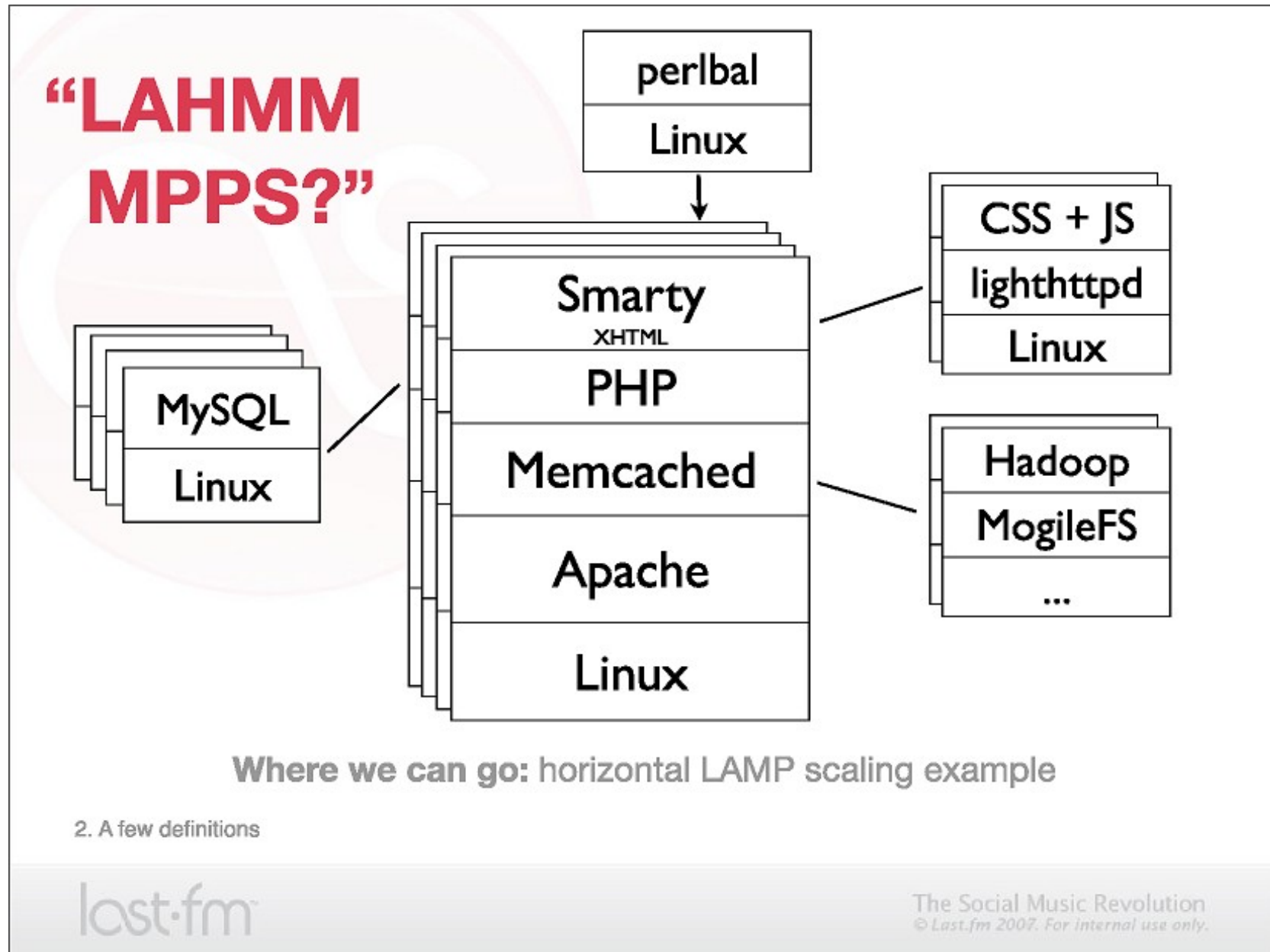


denied in /usr/local/apache/htdocs/include2007/oracle/db\_oracle.inc.php on line 194

Cannot open Database Error Log, please check!! (/home/dodgers/public\_html/./logs/oracle\_error\_log.txt)

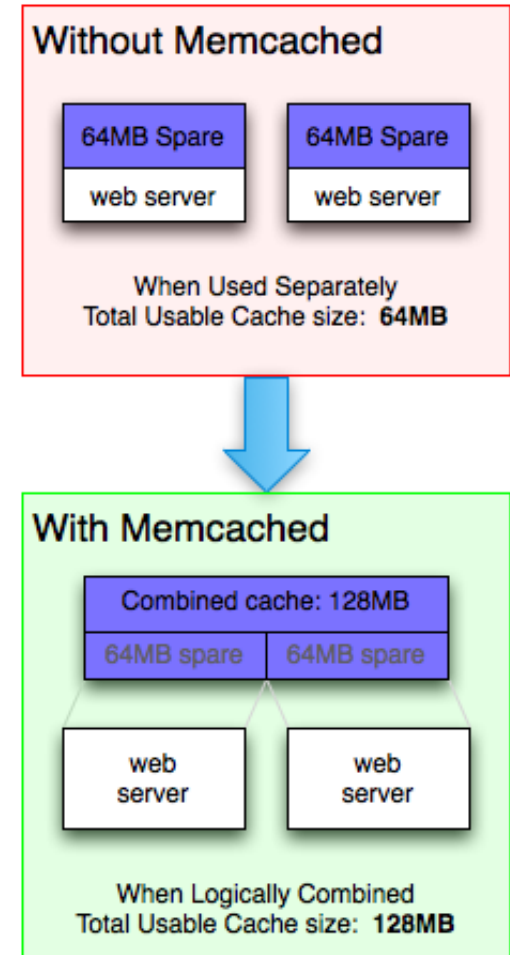
**Warning:** fopen(/home/dodgers/public\_html/./logs/oracle\_error\_log.txt) [function.fopen]: failed to open stream: Permission

# How to scale up web service in the past ?



# Tools used by large scale websites

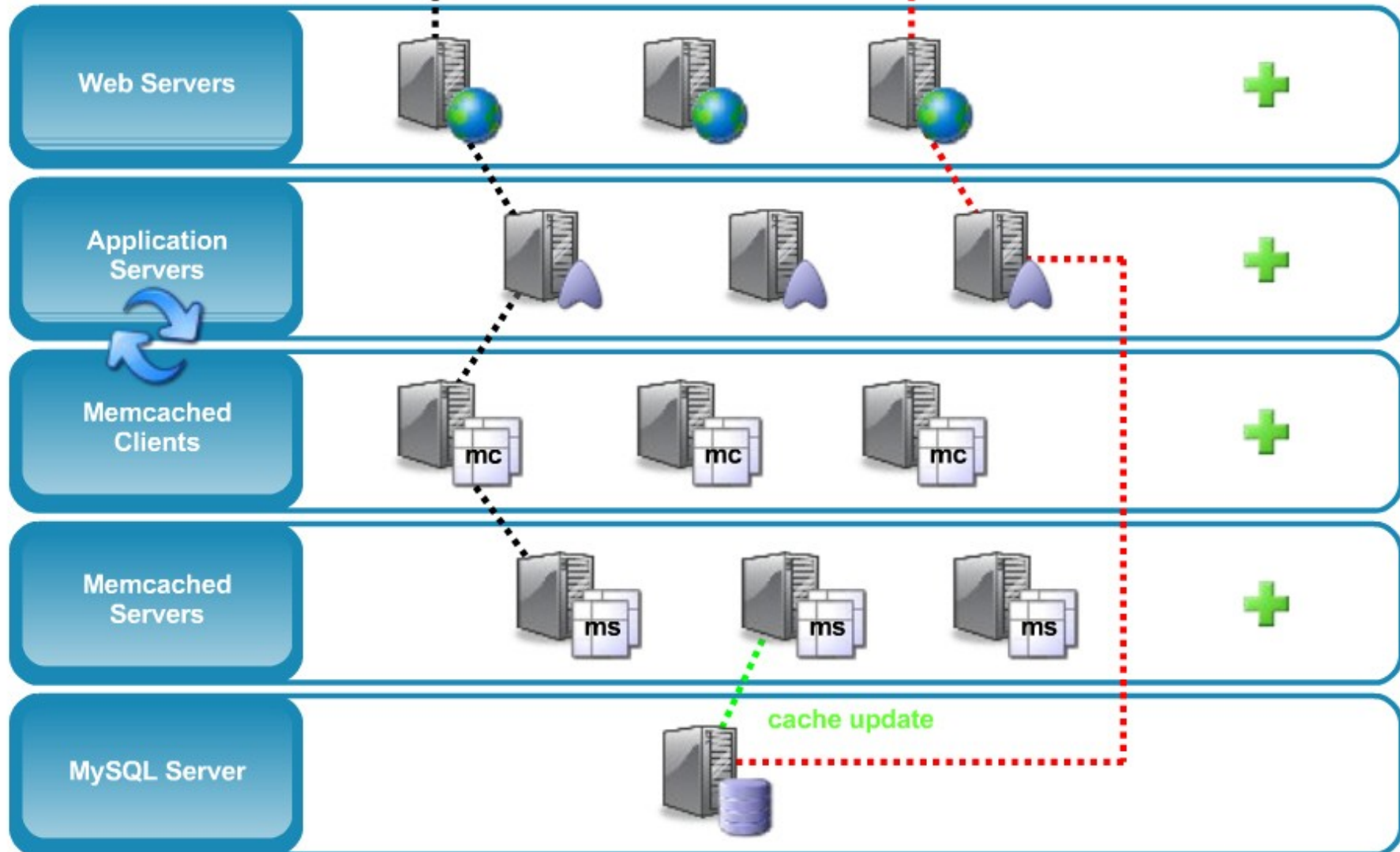
- Perlbal - <http://www.danga.com/perlbal/>
  - ◆ 多個網頁伺服器的負載平衡
  - ◆ Load balancer
- MogileFS - <http://www.danga.com/mogilefs/>
  - ◆ 分散式檔案系統
  - ◆ Distributed File System for small files
  - ◆ 有公司認為 MogileFS 比起 Hadoop 適合拿來處理小檔案
- memcached - <http://memcached.org/>
  - ◆ 共享記憶體??
  - ◆ Share Memory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



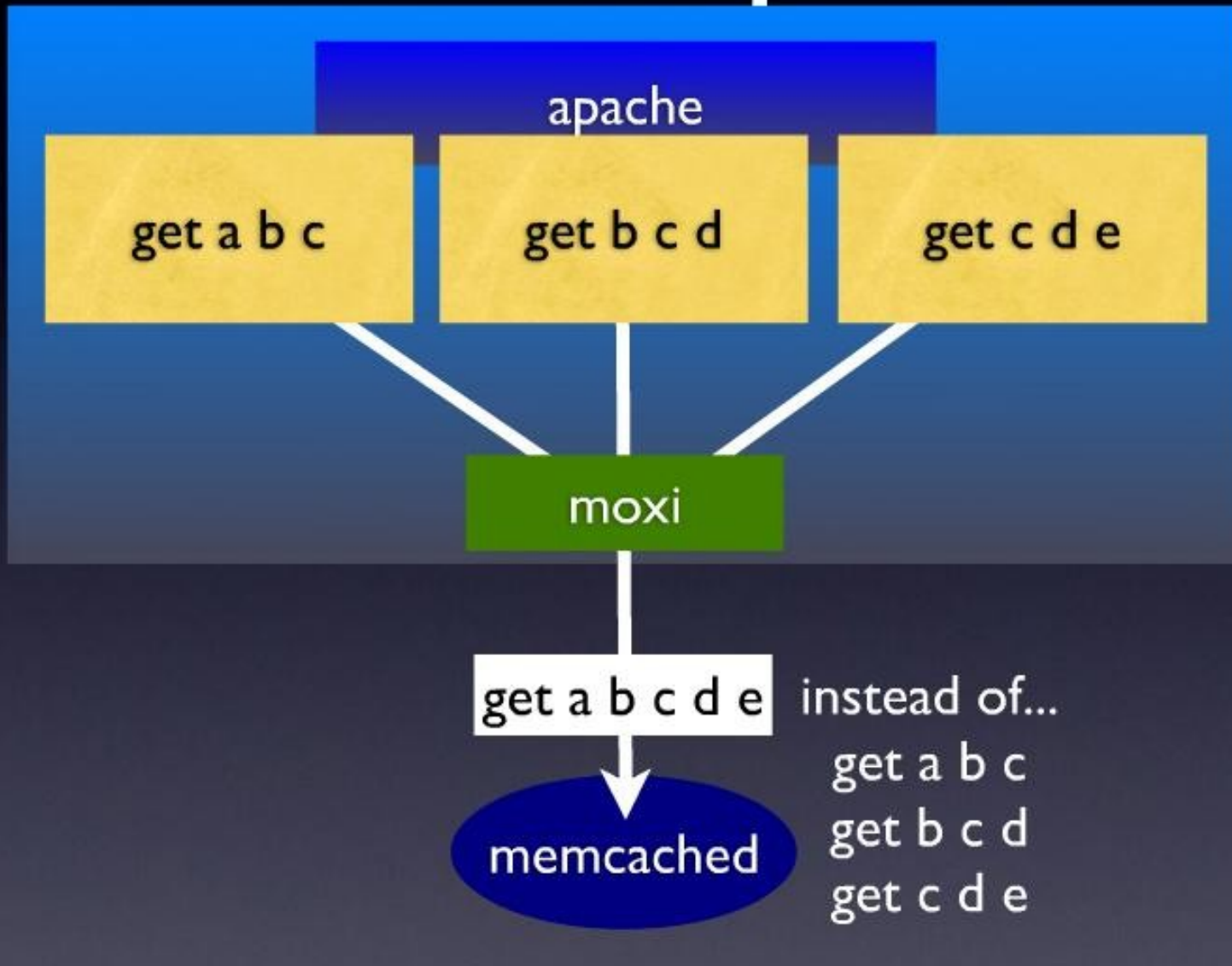
# Memcached & MySQL

read

write

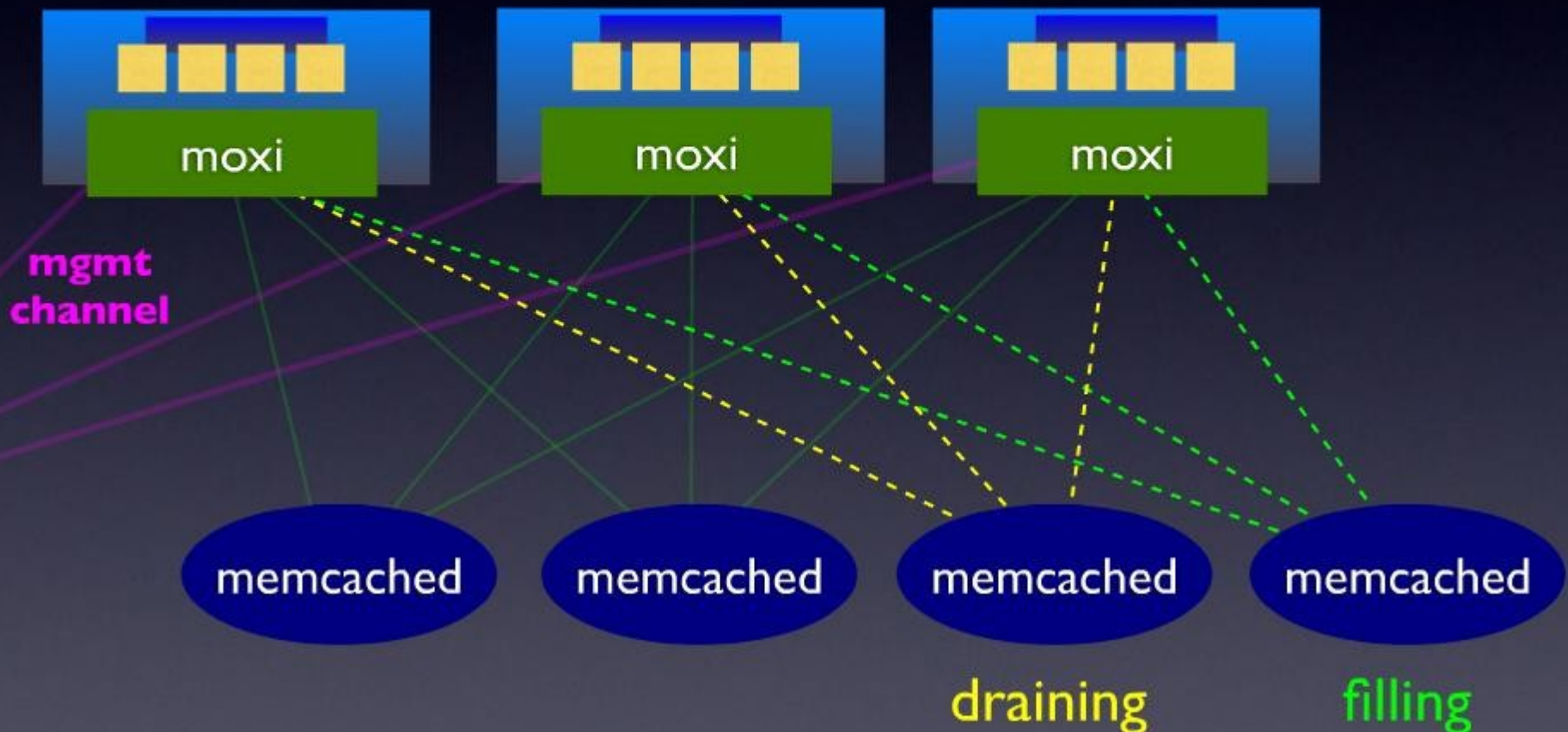


# GET de-duplication



# draining and filling

lazily migrate items from old server to new server



# HBase is ..

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

# Benefits

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



# Who use HBase

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

# Backdrop

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

# HBase Is Not ...

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

## HBase Is Not ... (2)

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

# Why Bigtable?

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

## Why Bigtable? (2)

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

# Why HBase ?

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

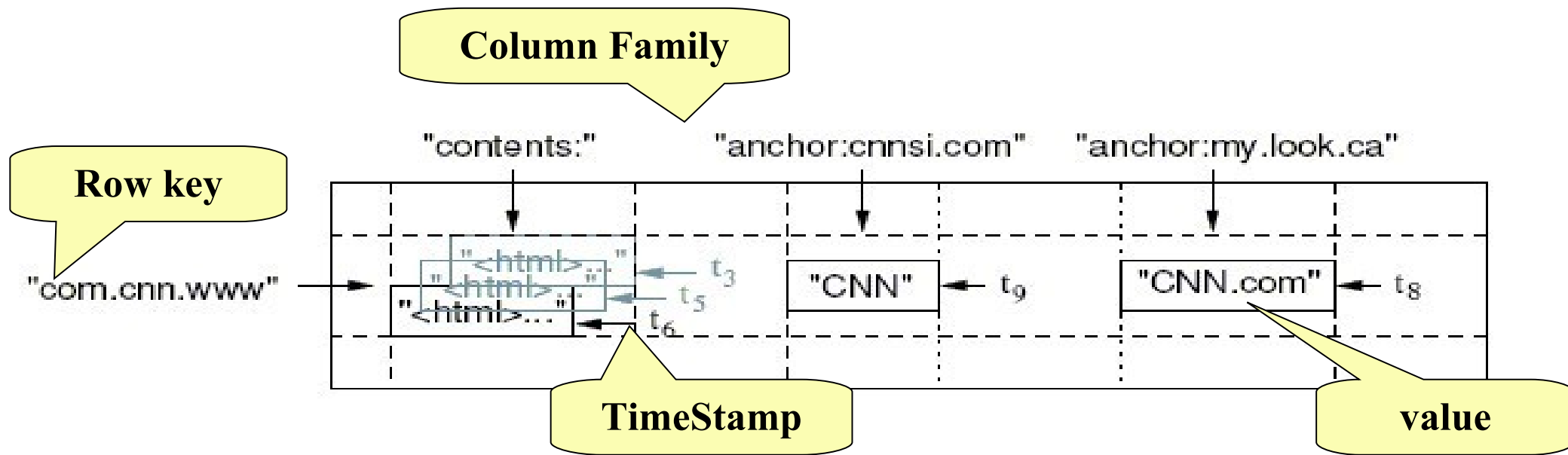
# HBase benefits than RDBMS

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



# Data Model

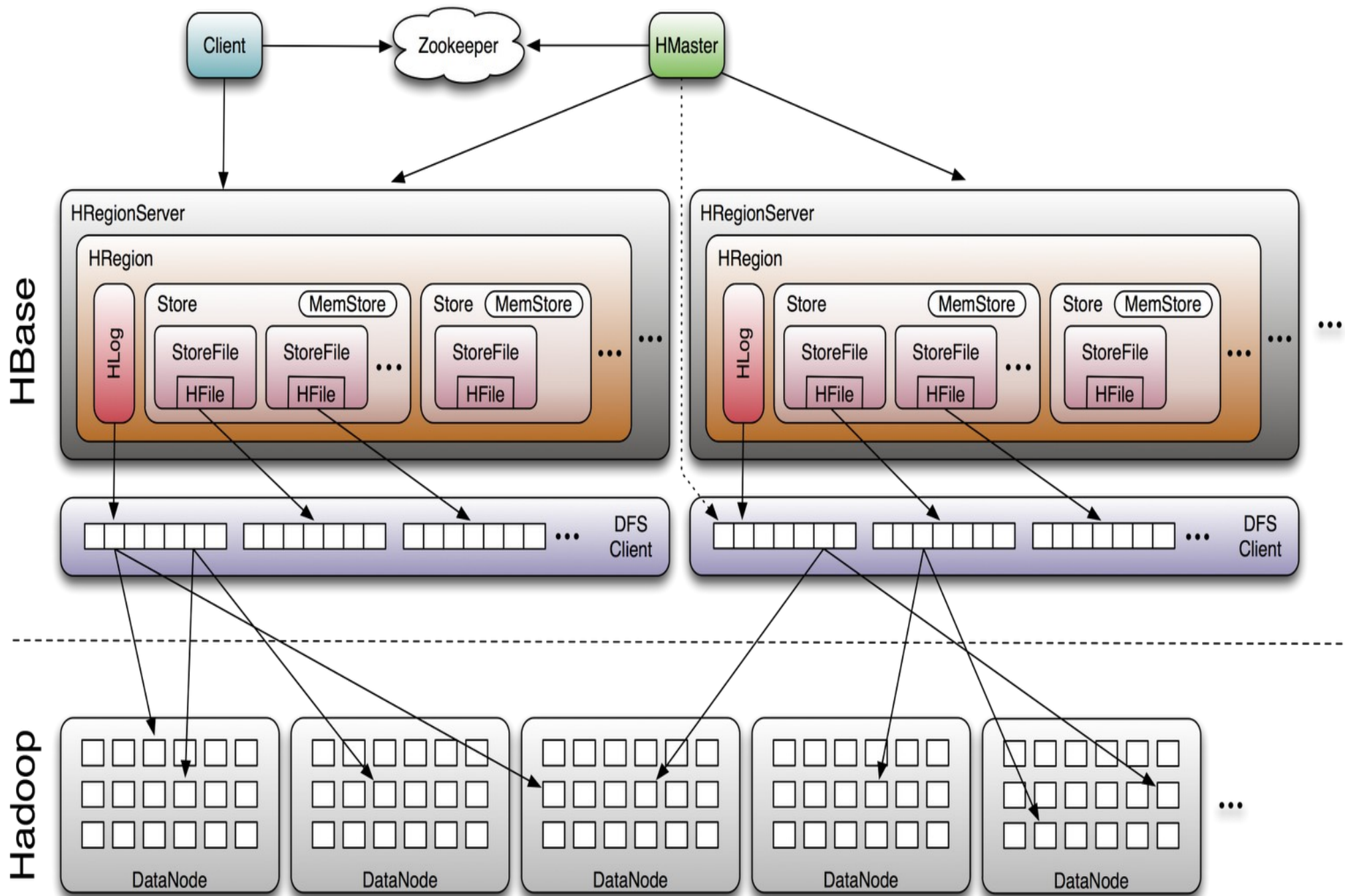
- Tables are sorted by **Row**
- Table schema only define it's *column families* .
  - Each family consists of any number of columns
  - Each column consists of any number of versions
  - Columns only exist when inserted, NULLs are free.
  - Columns within a family are sorted and stored together
- Everything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



# Members

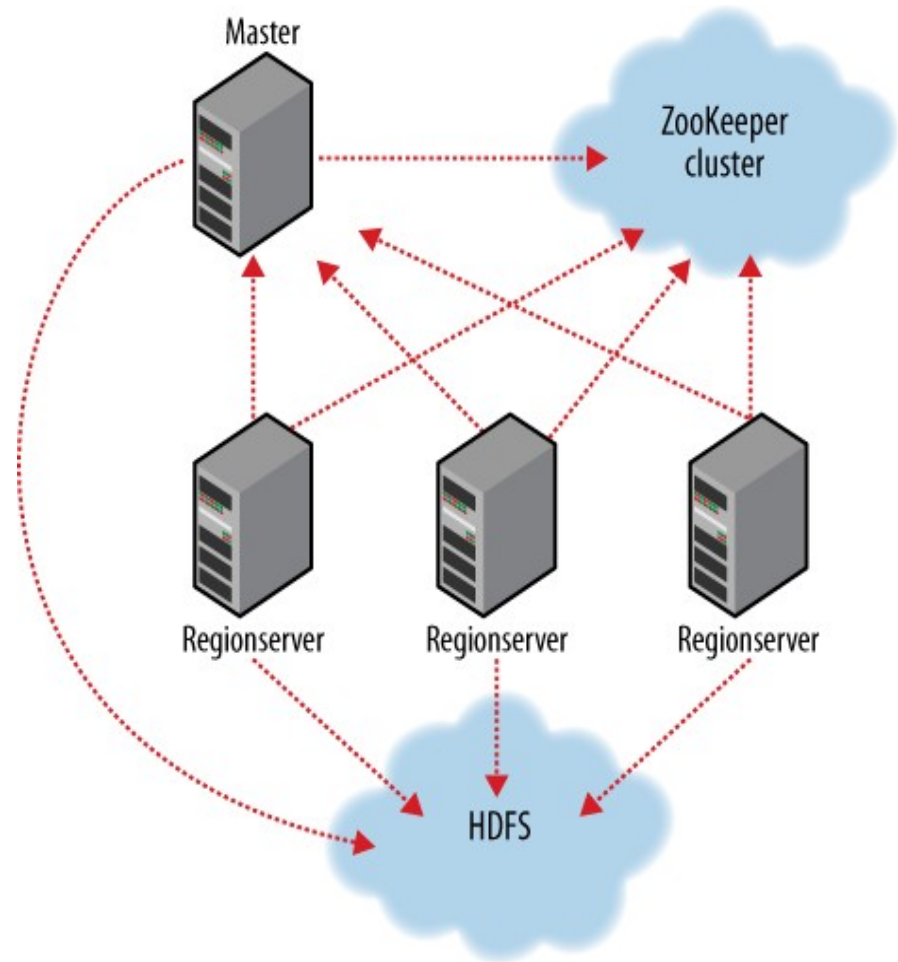
- *Master*
  - Responsible for monitoring region servers
  - Load balancing for regions
  - Redirect client to correct region servers
  - The current SPOF
- *regionserver slaves*
  - Serving requests(Write/Read/Scan) of Client
  - Send HeartBeat to Master
  - Throughput and Region numbers are scalable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

# Architecture



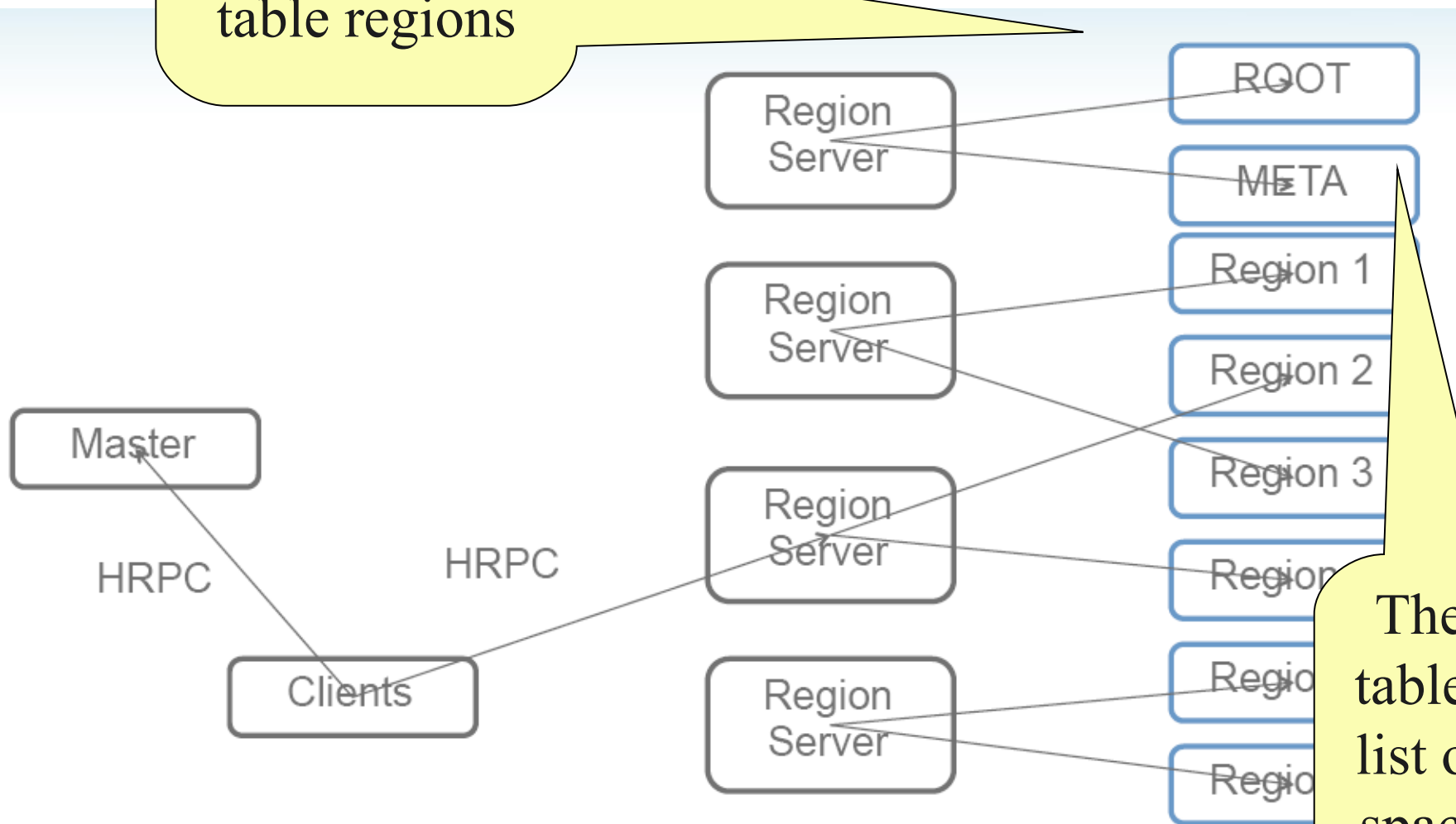
# ZooKeeper

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



# Operation

The `-ROOT-` table holds the list of `.META.` table regions



The `.META.` table holds the list of all user-space regions.



# 運用自由軟體打造資安雲端分析平台

Building Network Security Cloud Analysis Platform using Open Source

**Yao-Tsung Wang**

[jazz@nchc.org.tw](mailto:jazz@nchc.org.tw)

**Wei-Yu Chen**

[waue@nchc.org.tw](mailto:waue@nchc.org.tw)



# 專家說：雲端每個環節都有安全問題

ZDNet Taiwan - 專家談雲端：每個環節都有安全問題 - 新聞

2010/08/10 19:50:02

專家談雲端：每個環節都有安全問題

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

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

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

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

安全議題是目前台灣企業對雲端持保留態度的最大主因，這也是何以台灣的大型企業對於雲端的想法，還是

# 雲端資安的範疇

用雲端  
處理資安

**Dealing Security  
issues using Cloud**

**Data Security  
In the Cloud**

雲內部  
的資安管制

**Security Issues  
Inside the Cloud**

雲端資料  
安全性

端本身  
的資安威脅

**Security Threats  
to Internet of Things**



# 兩大研究方向：你該選「雲」還是「端」？



# 先來談談「端的安全」

用雲端  
處理資安

**Dealing Security  
issues using Cloud**

**Data Security  
In the Cloud**

雲內部  
的資安管制

**Security Issues  
Inside the Cloud**

雲端資料  
安全性

端本身  
的資安威脅

**Security Threats  
to Internet of Things**

以前你只有電腦需要防毒，現在 .....



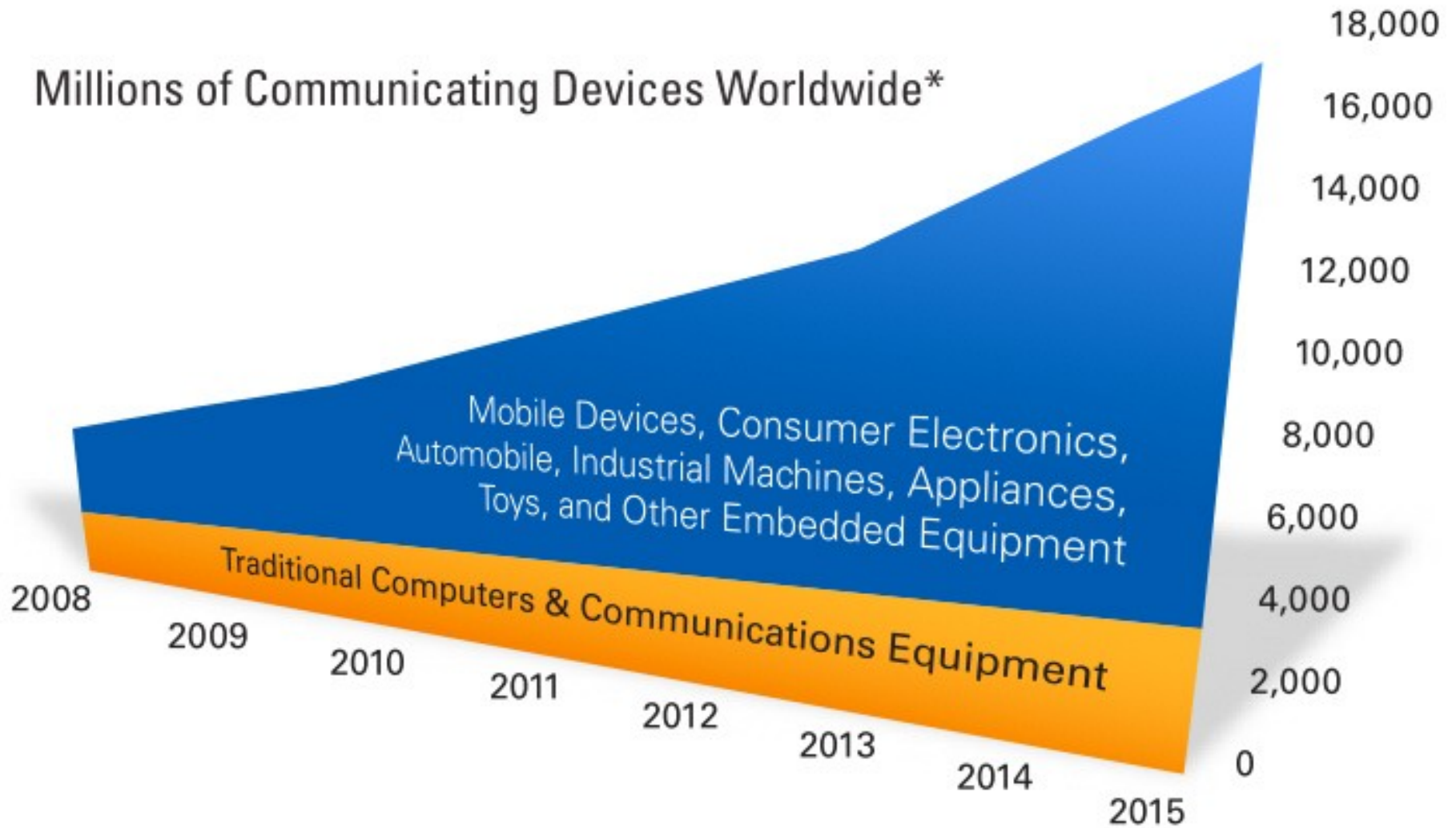
端

symbian  
OS



多元，中小廠  
Diversify，  
SMB

# 全球連網裝置急速成長中



Source: IDC Device Base Model, 2009

\*Excludes voice- and SMS-only phones

# 物聯網的時代來臨

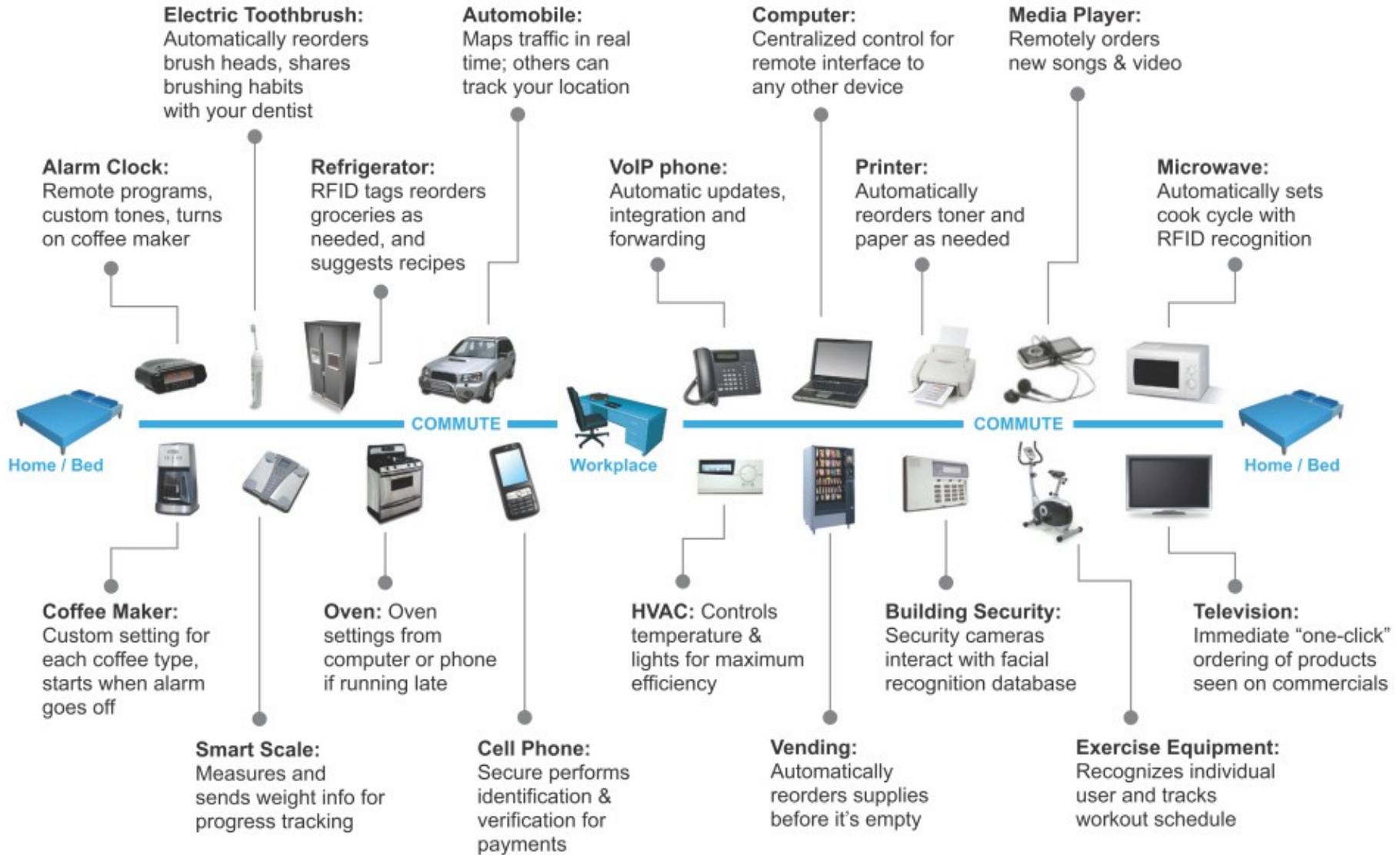
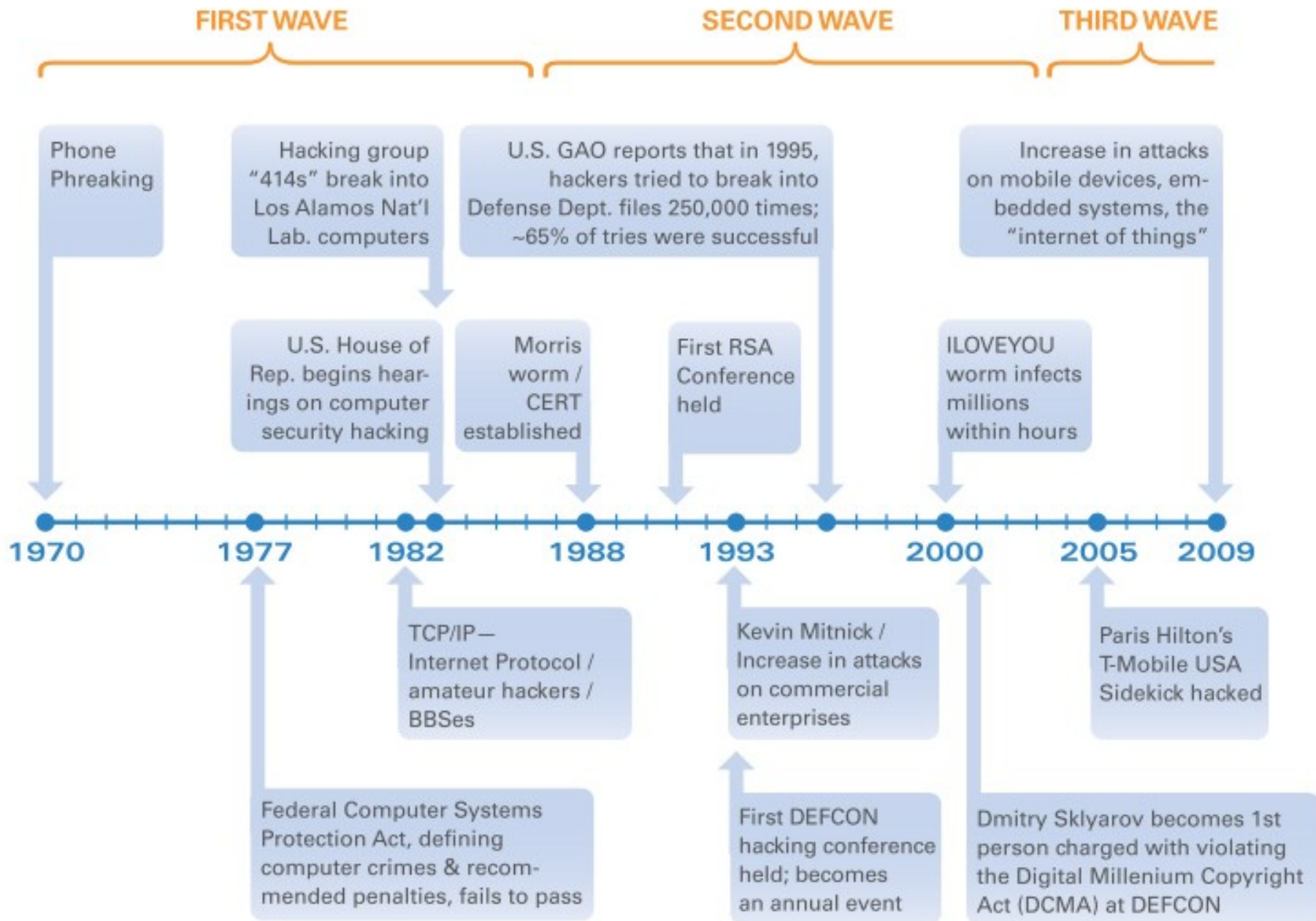


Figure 3. The Internet of Things

# 第三波網路入侵對象將鎖定在『物聯網』



# 針對行動裝置的各種資安問題與經驗

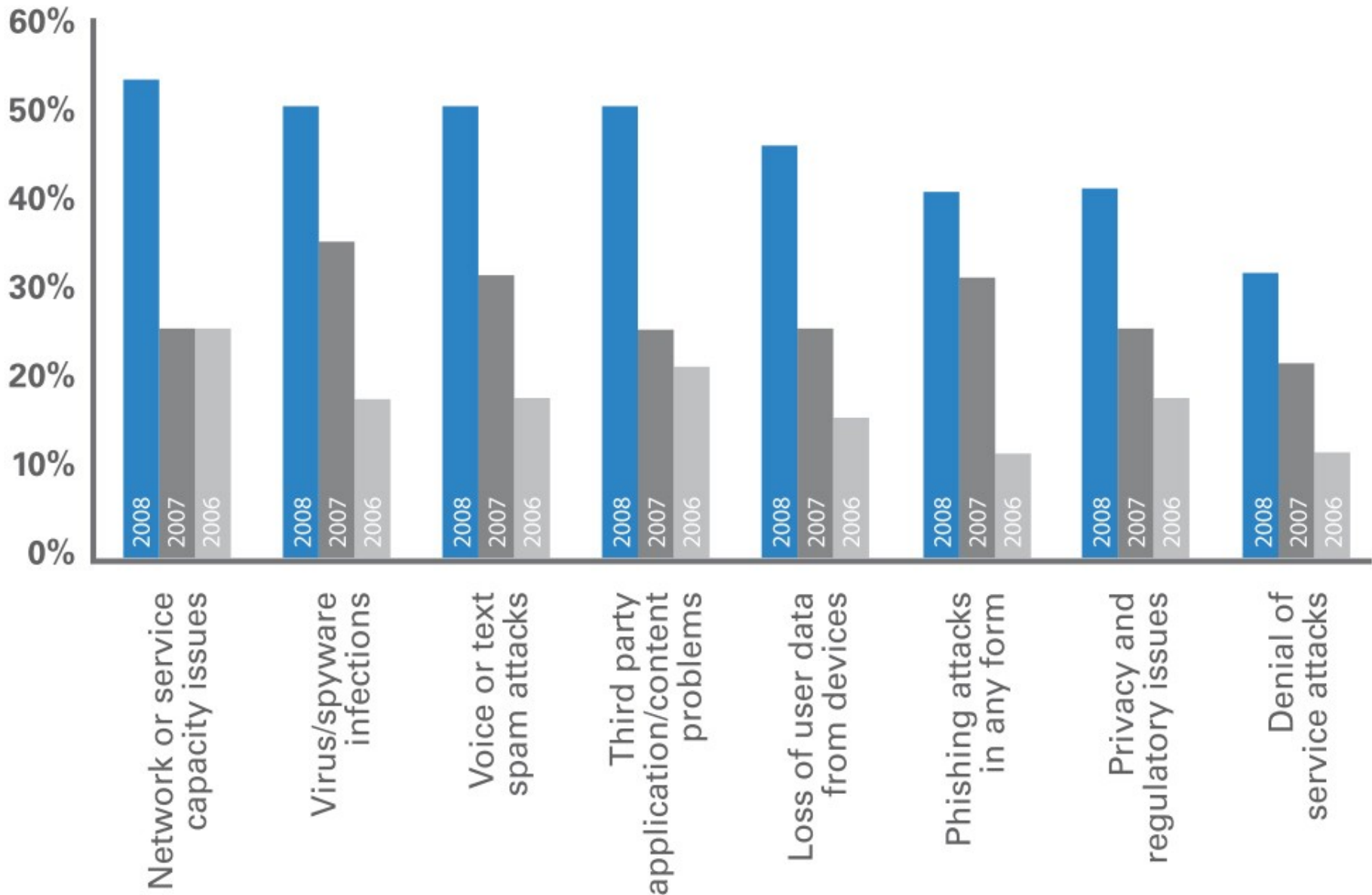
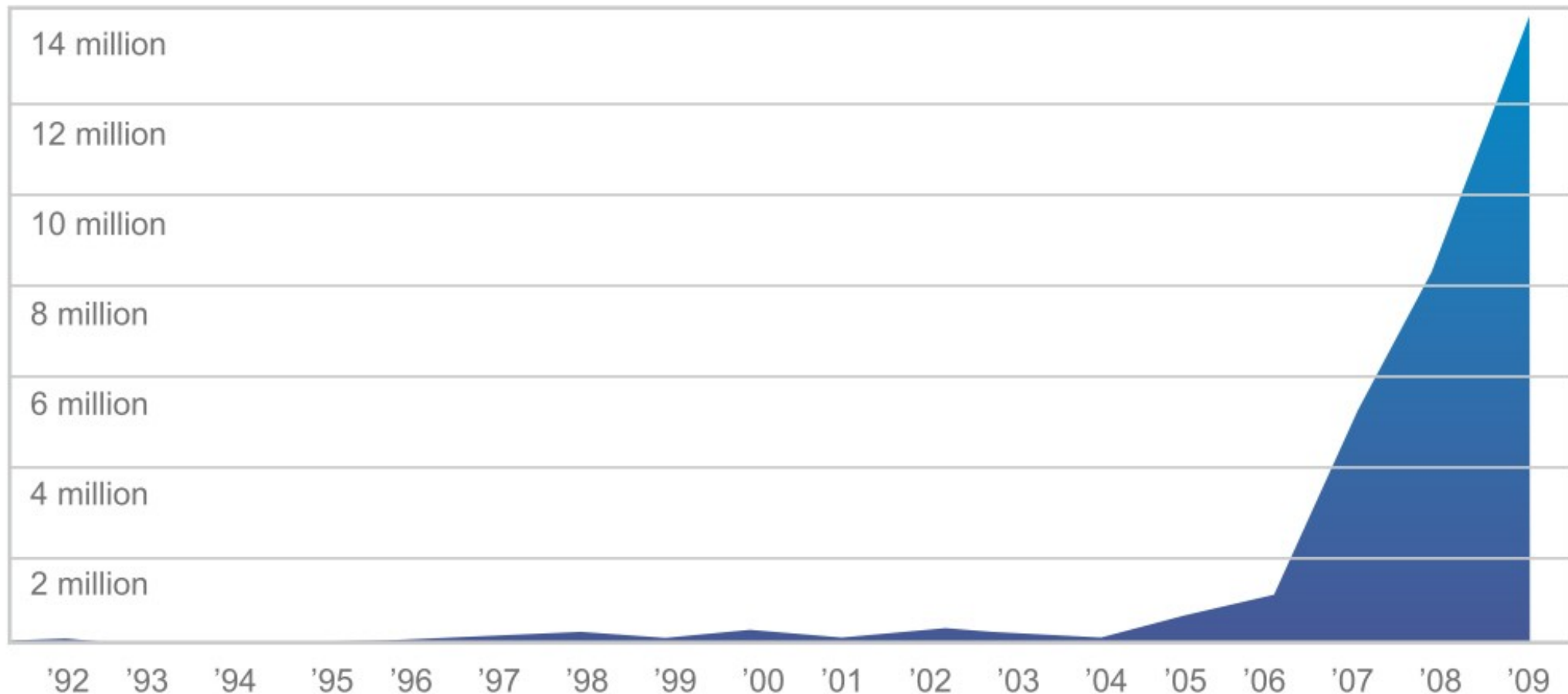


Figure 6. The increase in security issues experienced by mobile device users from 2006 to 2008; % of respondents. McAfee *Mobile Security Report 2009*

圖片來源：[Attacks on Mobile and Embedded Systems: Current Trends](#) by Mocana

# 網路惡意程式 (Malware) 逐年激增

Malware detected by year



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

Source: AV LABS

圖片來源：

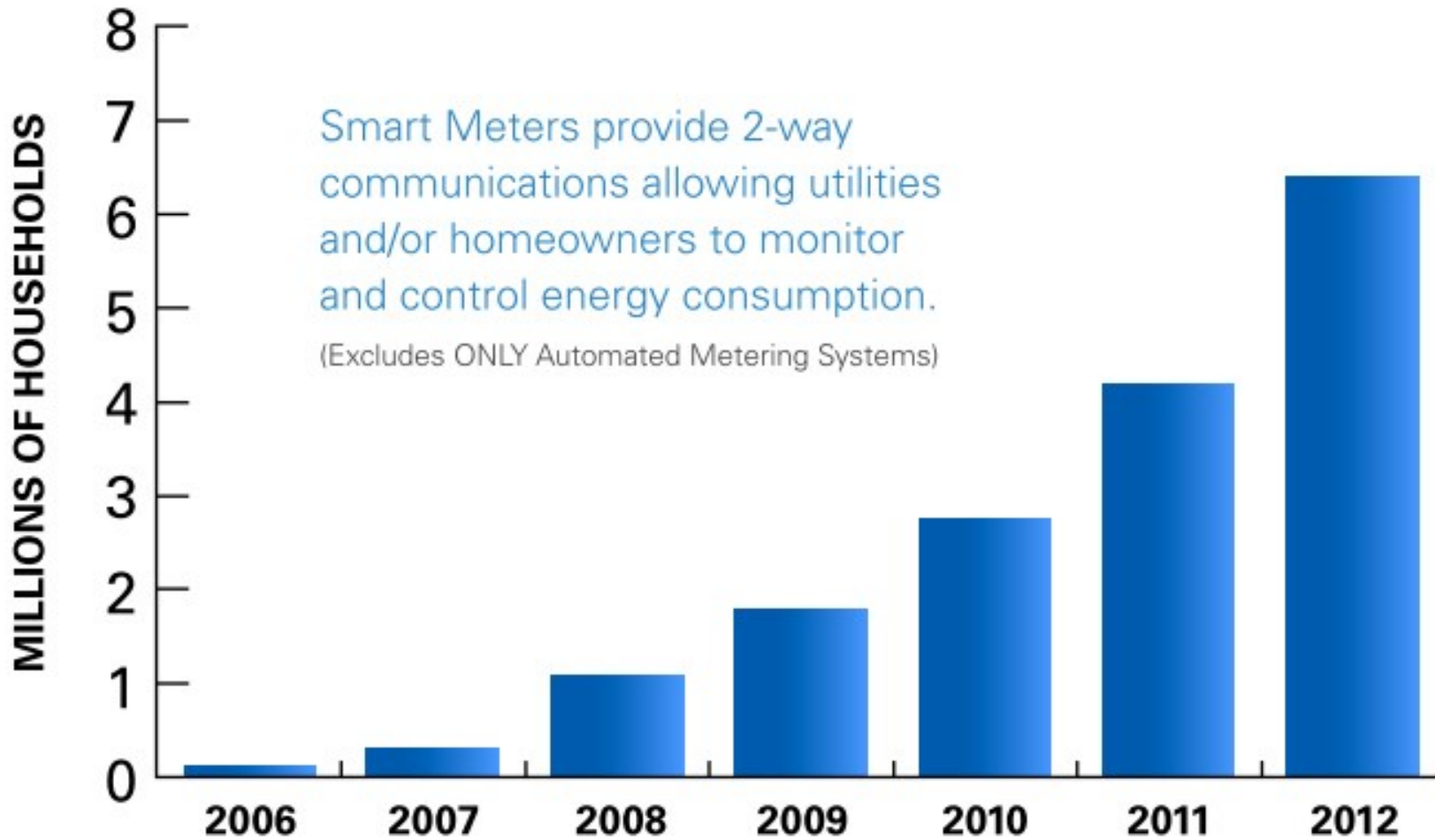
[U.S. Unprepared for Internet Device Flood: Unaddressed Security Problems & Talent Drought Threaten Long-Term Commercial, Government Interests](#) 11

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



# 如果你家的智慧電錶被入侵會怎樣？

## U.S. Households with Smart Meters



© Copyright 2009 - Parks Associates

圖片來源：

U.S. Unprepared for Internet Device Flood: Unaddressed Security Problems & Talent Drought Threaten Long-Term Commercial, Government Interests 12

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

# 再來談談「雲的安全」

用雲端  
處理資安

**Dealing Security  
issues using Cloud**

**Data Security  
In the Cloud**

雲內部  
的資安管制

**Security Issues  
Inside the Cloud**

雲端資料  
安全性

端本身  
的資安威脅

**Security Threats  
to Internet of Things**





VM 2

A yellow rounded rectangle containing a blue circular icon of a computer monitor and a server tower, representing a virtual machine.

VM 1

A yellow rounded rectangle containing a blue circular icon of a computer monitor and a server tower, representing a virtual machine.

VMM

An orange rounded rectangle labeled 'VMM' above a blue server rack icon, representing a Virtual Machine Monitor.

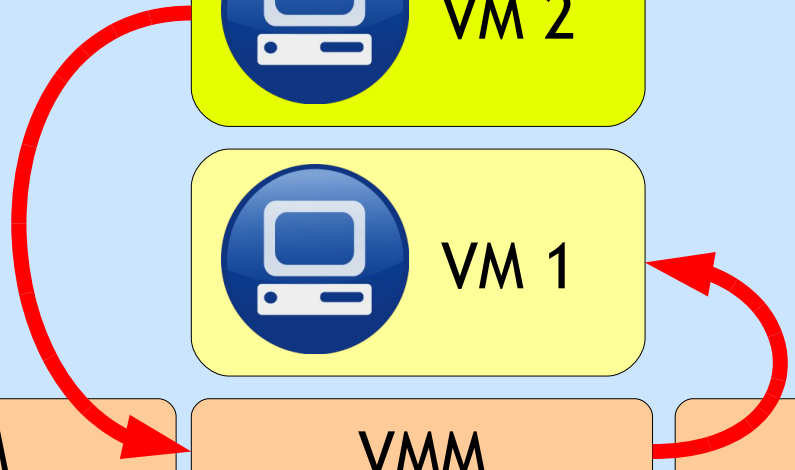
VMM

An orange rounded rectangle labeled 'VMM' above a blue server rack icon, representing a Virtual Machine Monitor.

VMM

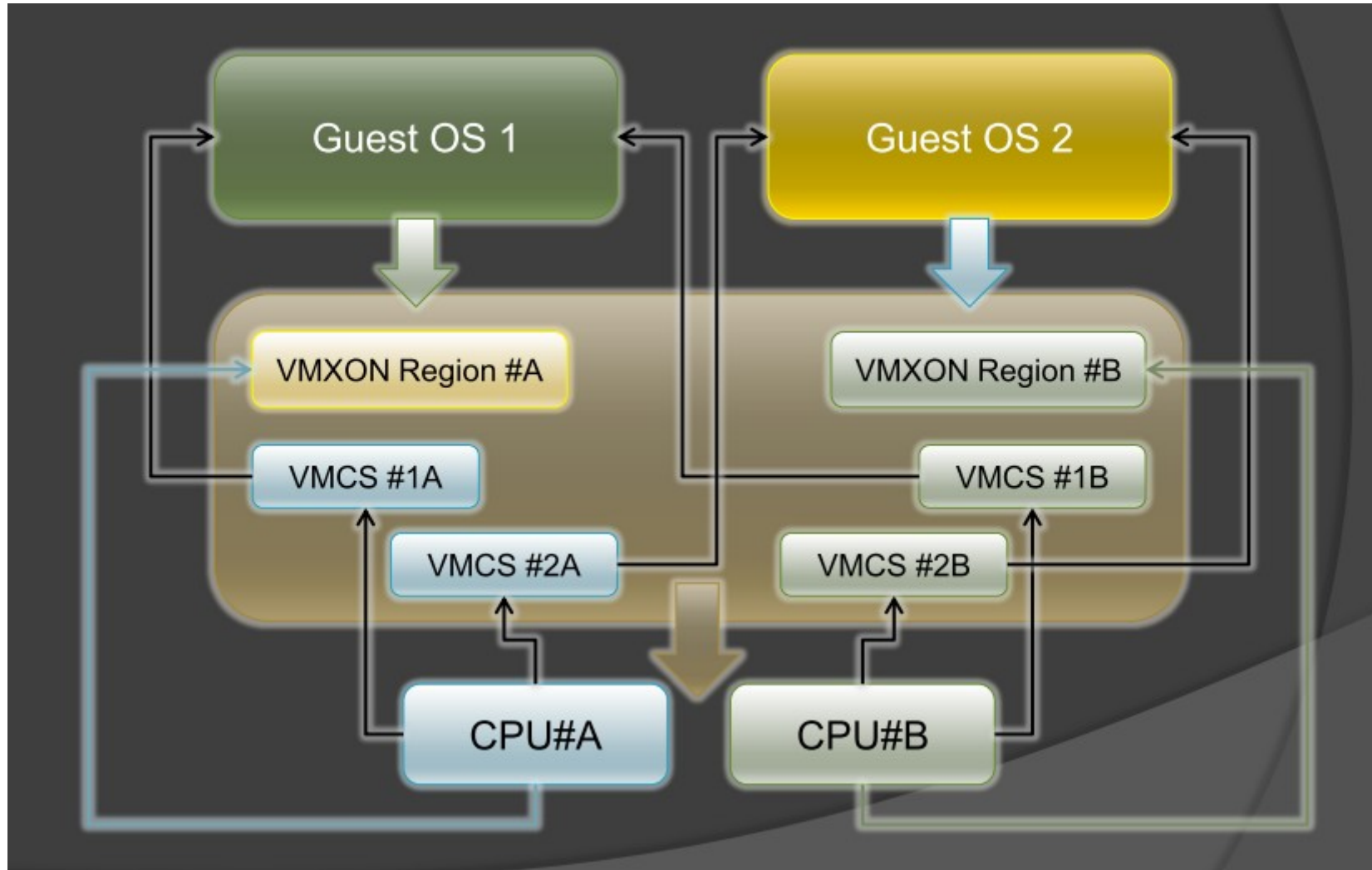
An orange rounded rectangle labeled 'VMM' above a blue server rack icon, representing a Virtual Machine Monitor.

VMM

An orange rounded rectangle labeled 'VMM' above a blue server rack icon, representing a Virtual Machine Monitor.

# 虛擬化衍生的新興資安問題

透過虛擬機器，竊取鍵盤輸入、植入後門.....



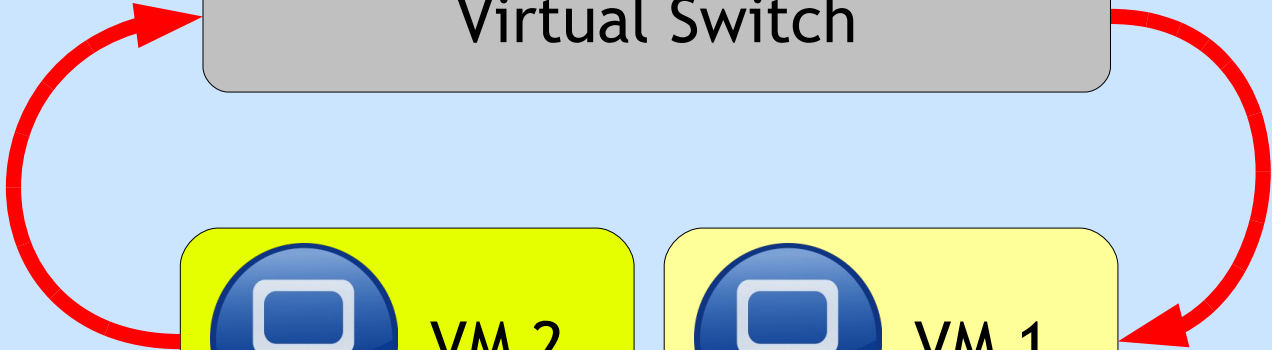
圖片來源： Hacks in Taiwan Conference 2010

[http://www.hitcon.org/hit2010/download/6\\_New%20Battlefield%20For%20Malware%20Game.pdf](http://www.hitcon.org/hit2010/download/6_New%20Battlefield%20For%20Malware%20Game.pdf)

王大寶 & PK / Hypervisor - New Battlefield For Malware Game 虛擬機 - 惡意程式攻防的新戰場 16



Virtual Switch



 VM 2

 VM 1

VMM



VMM



VMM



VMM



# 三談「資料安全」

用雲端  
處理資安

**Dealing Security  
issues using Cloud**

**Data Security  
In the Cloud**

雲內部  
的資安管制

**Security Issues  
Inside the Cloud**

雲端資料  
安全性

端本身  
的資安威脅

**Security Threats  
to Internet of Things**

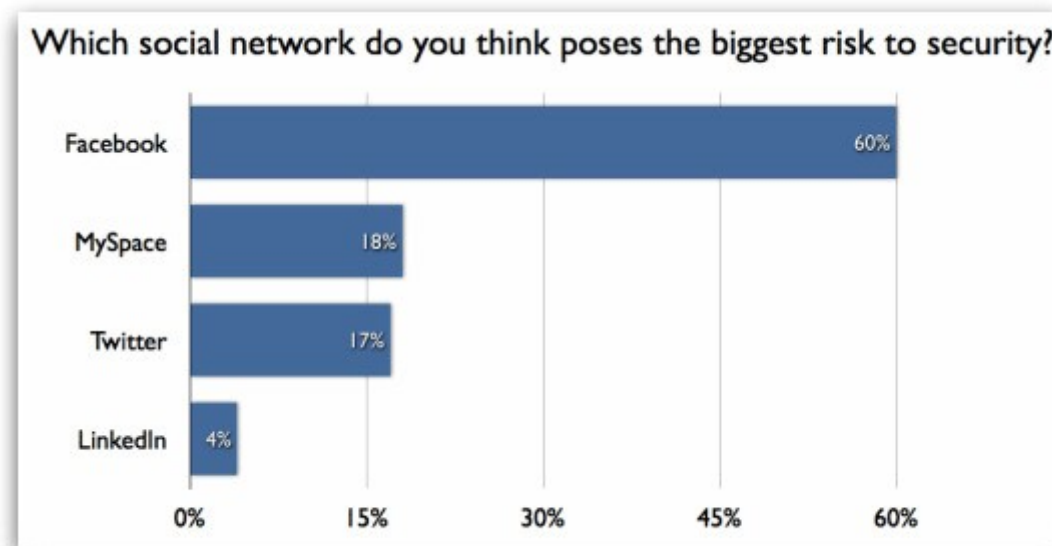
# Ex. 無名照片外流、臉書個資外洩

轟動一時黑澀會妹妹容瑄親密自拍照片外流



圖片來源：

[Wikileaks and Facebook Privacy / Security: Do we care?](#)



圖片來源：

Report Ranks Facebook As Greatest Corporate Security Risk

<http://www.allfacebook.com/facebook-corporate-risk-2010-02>



# 進入今天的主題：用雲端處理傳統資安問題

今天的重點

用雲端  
處理資安

**Dealing Security  
issues using Cloud**

**Data Security  
In the Cloud**

雲內部  
的資安管制

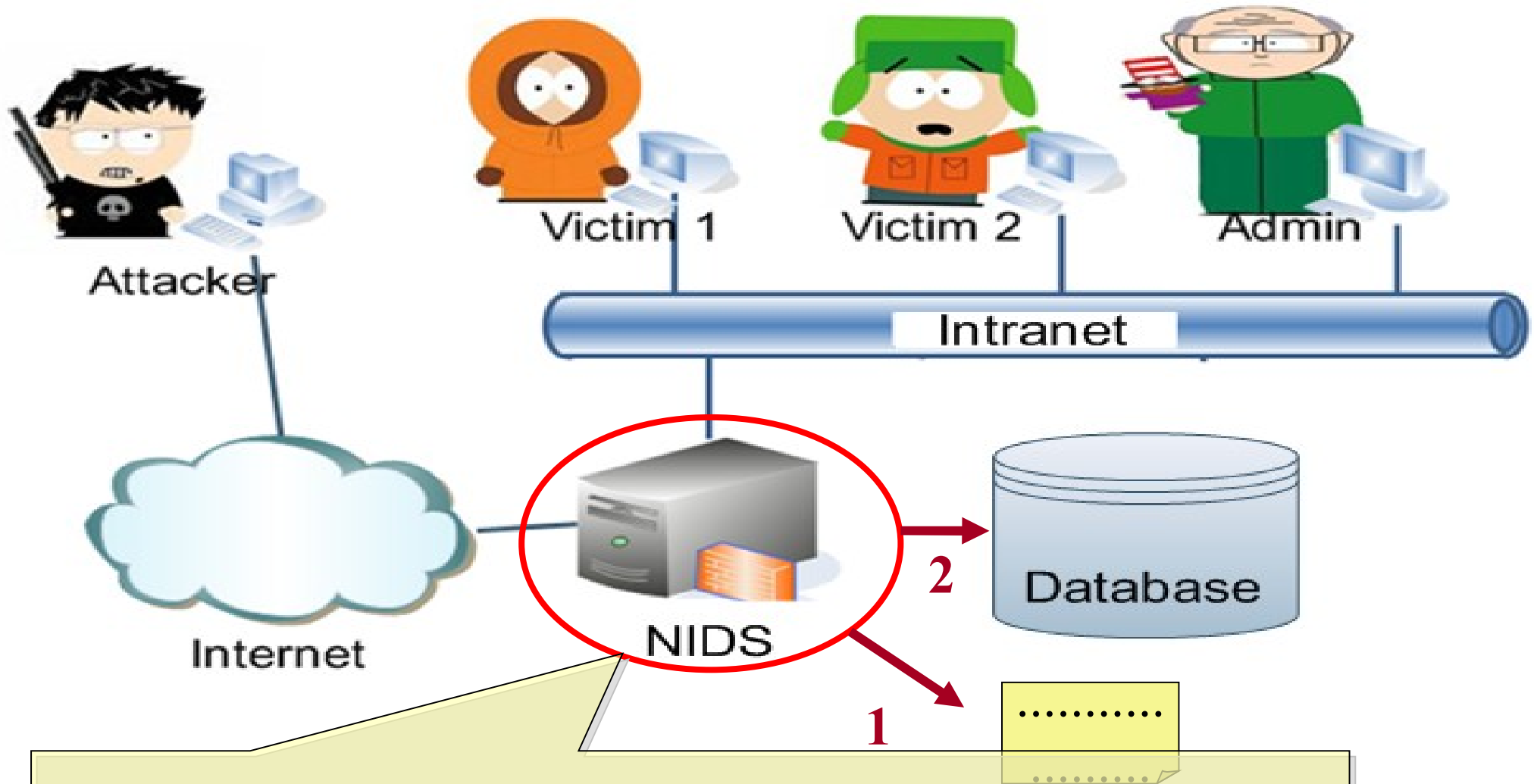
**Security Issues  
Inside the Cloud**

雲端資料  
安全性

端本身  
的資安威脅

**Security Threats  
to Internet of Things**

# 使用入侵偵測系統 (NIDS) 來找出入侵訊息



當入侵偵測系統偵測到網路上有異常封包時，就會產生警訊以告知有攻擊發生。警訊通常有兩種形式：  
1. 紀錄成 log 檔 2. 紀錄到資料庫

# 傳統 NIDS 的警訊型態 (1) 紀錄在日誌檔內

## 入侵偵測系統所產生警訊日誌檔內一小段內容

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

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

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

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

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

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

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

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

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

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

# 傳統 NIDS 的警訊型態 (2) 紀錄在資料庫內

以下為利用瀏覽器透過網頁方式呈現警訊資料庫的內容

The screenshot shows a Mozilla browser window displaying the 'Basic Analysis and Security Engine (BASE) Query Results' page. The page title is 'Basic Analysis and Security Engine (BASE)'. The navigation menu includes 'Home', 'Search', and 'AG Maintenance'. A status message indicates 'Added 0 alert(s) to the Alert cache' and the query was performed on 'Thu October 14, 2004 22:04:44'. A table lists search criteria: Meta Criteria, IP Criteria, TCP Criteria, and Payload Criteria, all set to 'any'. A 'Summary Statistics' box lists: Sensors, Unique Alerts (classifications), Unique addresses (source | destination), Unique IP links, Source Port (TCP | UDP), Destination Port (TCP | UDP), and Time profile of alerts. Below this, it states 'Displaying alerts 1-50 of 81 total'. The main content is a table of alerts with columns for ID, Signature, Timestamp, Source Address, Dest. Address, and Layer 4 Proto.

<input type="checkbox"/>	ID	< Signature >	< Timestamp >	< Source Address >	< Dest. Address >	< Layer 4 Proto >
<input type="checkbox"/>	#0-(1-84)	[snort] NETBIOS SMB IPC\$ share unicode access	2004-10-08 11:25:41	192.168.1.100:1613	192.168.1.4:139	TCP
<input type="checkbox"/>	#1-(1-83)	[snort] NETBIOS SMB IPC\$ share unicode access	2004-10-08 11:25:31	192.168.1.100:1608	192.168.1.4:139	TCP
<input type="checkbox"/>	#2-(1-82)	[snort] NETBIOS SMB IPC\$ share unicode access	2004-10-08 11:25:05	192.168.1.100:1601	192.168.1.4:139	TCP
<input type="checkbox"/>	#3-(1-80)	[snort] (http_inspect) OVERSIZE CHUNK ENCODING	2004-10-04 22:25:41	192.168.1.4:42164	67.19.245.228:80	TCP
<input type="checkbox"/>	#4-(1-81)	[snort] (http_inspect) OVERSIZE CHUNK ENCODING	2004-10-04 22:25:41	192.168.1.4:42163	67.19.245.228:80	TCP

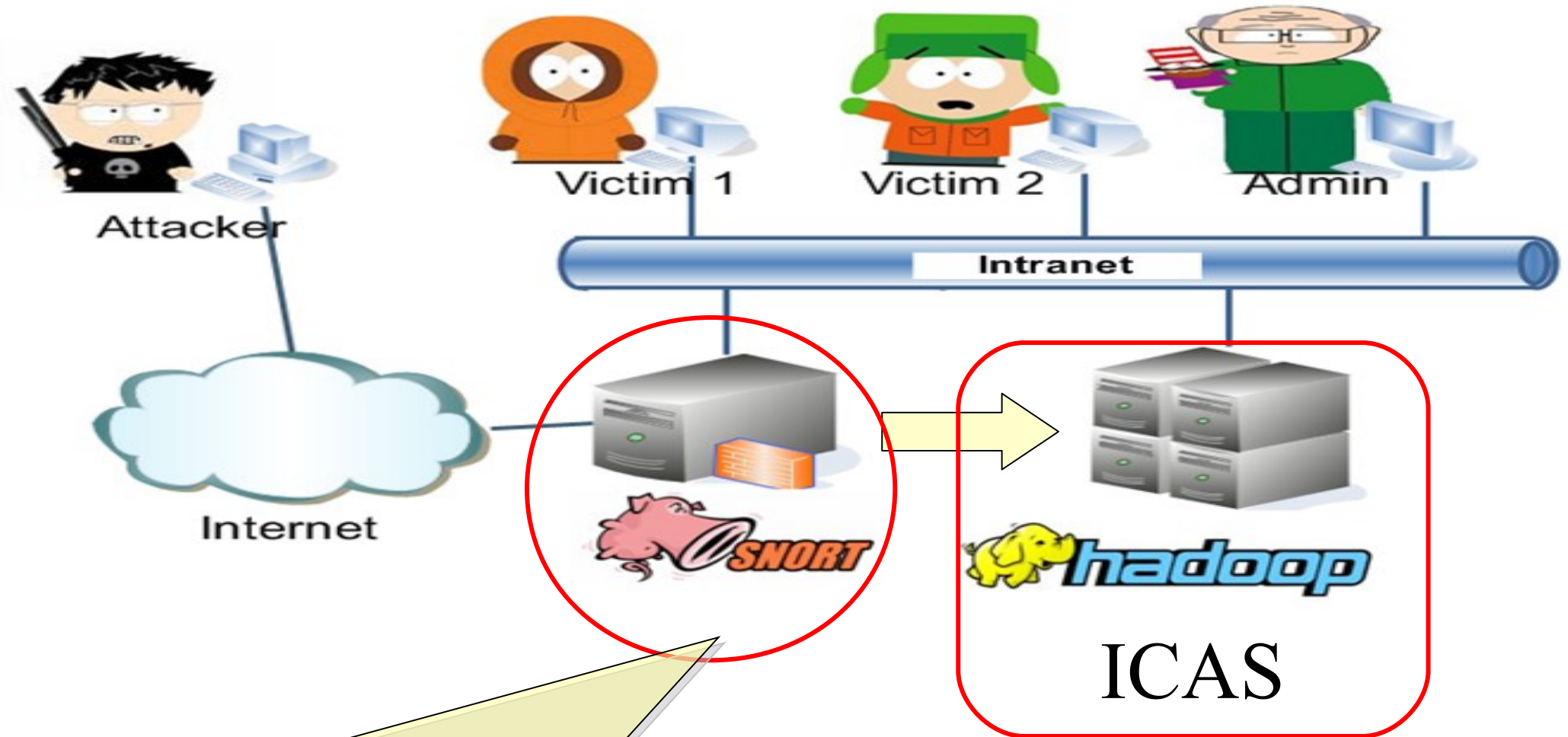
## 以上作法的缺點

- 警訊僅被『忠實』地被記錄下來，無法顯示彼此間的關聯性，因此系統管理者難以瞭解全部攻擊情形
- 過多的警訊，使得容易忽略重要內容
- 完全依賴單一資料庫，當資料量一大，該台主機的讀寫效率將成為瓶頸

# 使用雲端運算的解決方案：ICAS

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

# 透過 ICAS 協助分析 IDS 的警訊



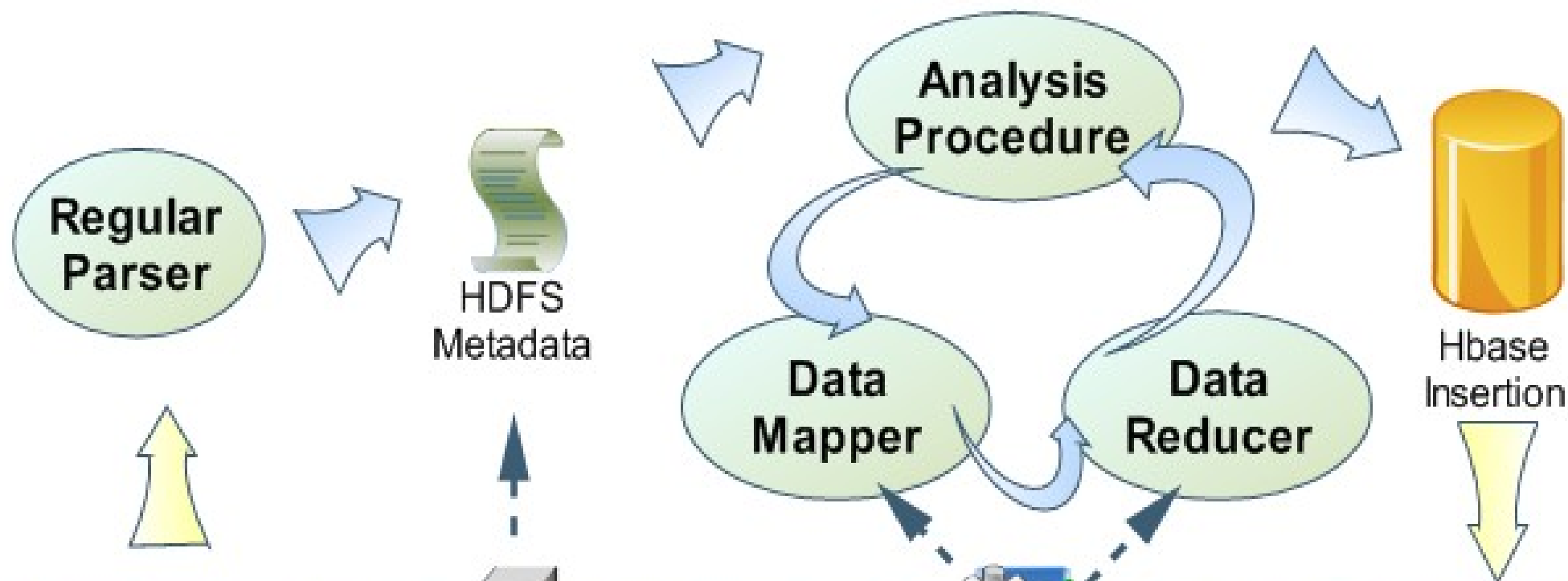
可多個 NIDS 共同產生警訊後，傳送至 ICAS，分析演算法目前有 ICAS-I 及 ICAS-II

# ICAS-I

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



# ICAS-I 流程圖



**Intrusion  
Detectoin  
System**



**HDFS**



**JobTracker**



**hadoop**

**Cloud Platform**



**HBASE**

**Database**

# ICAS-I 整合後的警訊結果

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



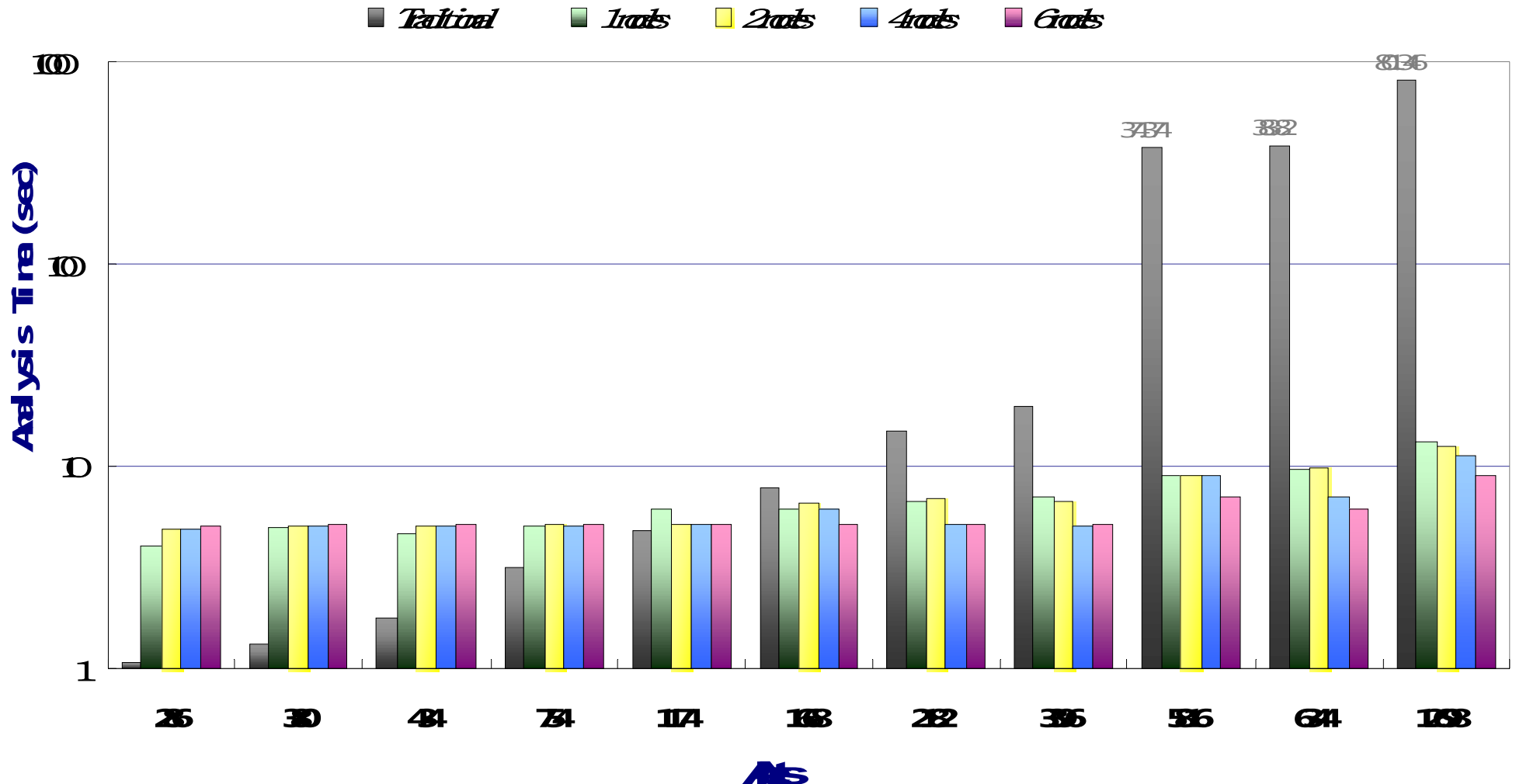
Key		Values				
Host_1	Trojan	Sip1,Sip2	80,443	4077,5002	tcp	T1,T2,T3
Host_2	Trojan	Sip1	443	5002	tcp	T4
Key		Values				

# ICAS-I 效能數據的環境

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

# ICAS-I 效能分析時間圖

## The Consuming Time of Each Number of Data Sets



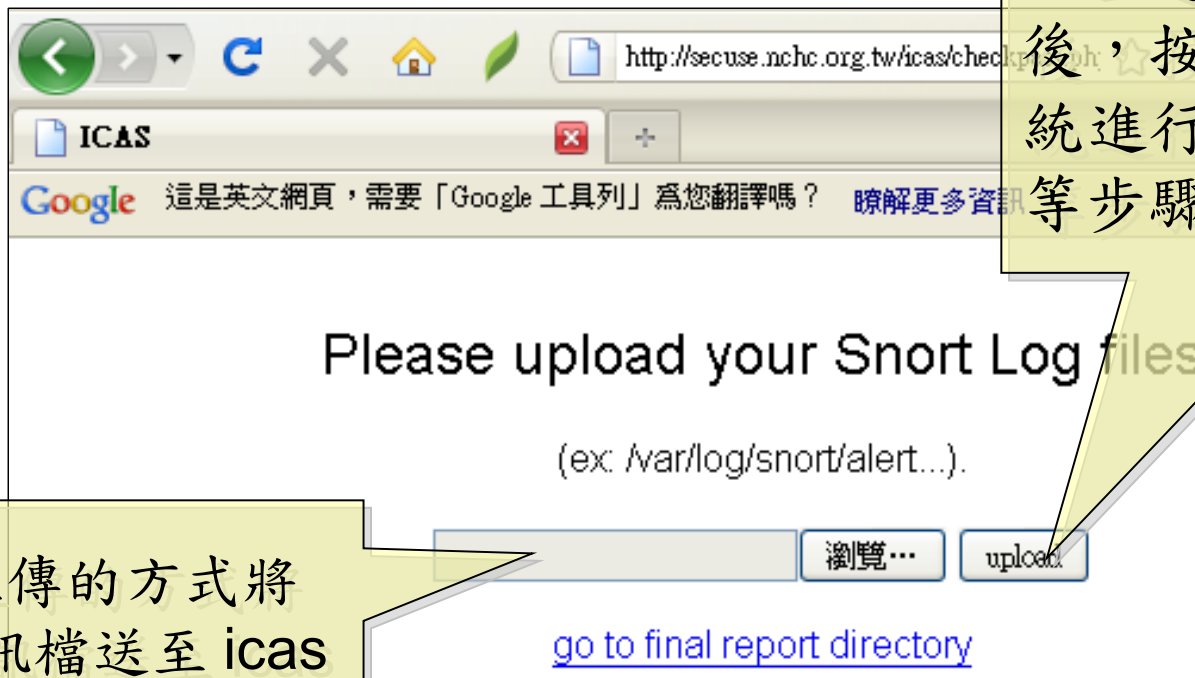
# ICAS-I 效能數據表

## Throughput Data Overall

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

# ICAS-II

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

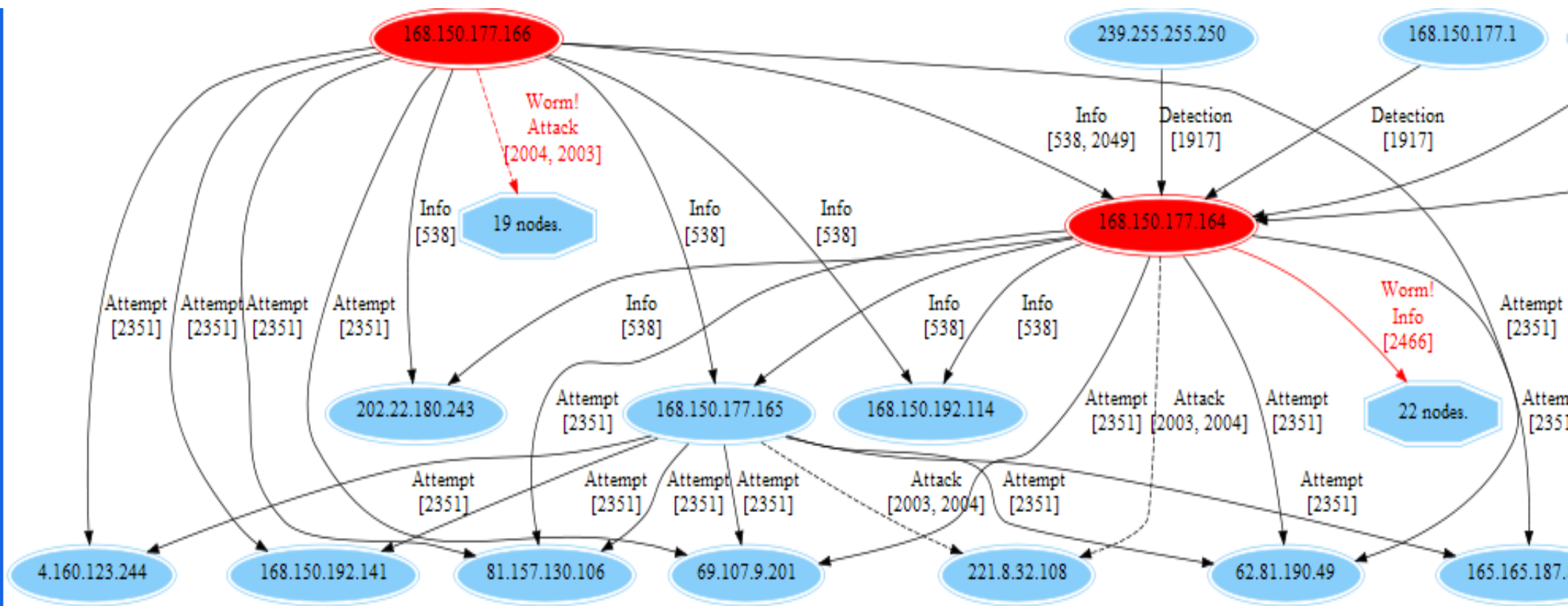


一旦選定需分析的日誌檔後，按下 upload 鈕，系統進行上傳→分析→繪圖等步驟

透過網頁上傳的方式將 snort 的警訊檔送至 icas 分析

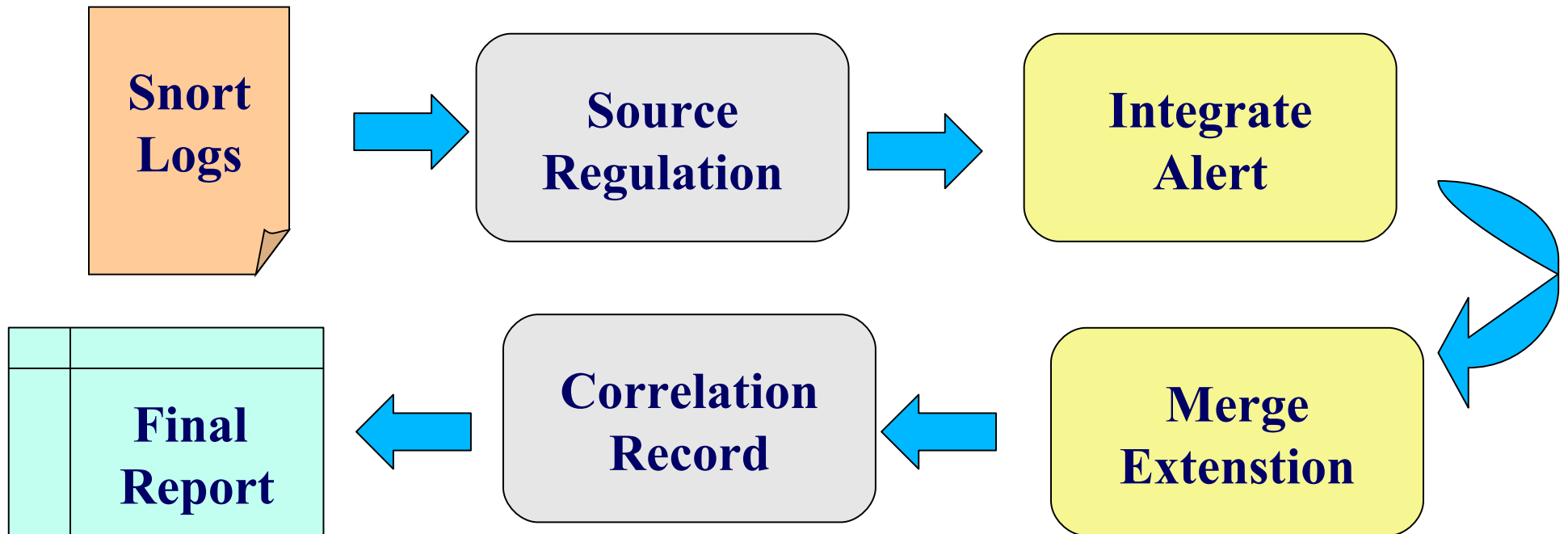
# ICAS-II 所產生的報表：警訊關聯圖

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



# ICAS-II 的分析流程

- Hadoop v 0.20





# ICAS-II 結論

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

# ICAS 總結

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



## Questions?

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

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





# CRAWLZILLA

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

Shun-Fa Yang 、 Wei-Yu Chen 、 Wen-Chieh Kuo  
Free Software Lab. @ NCHC

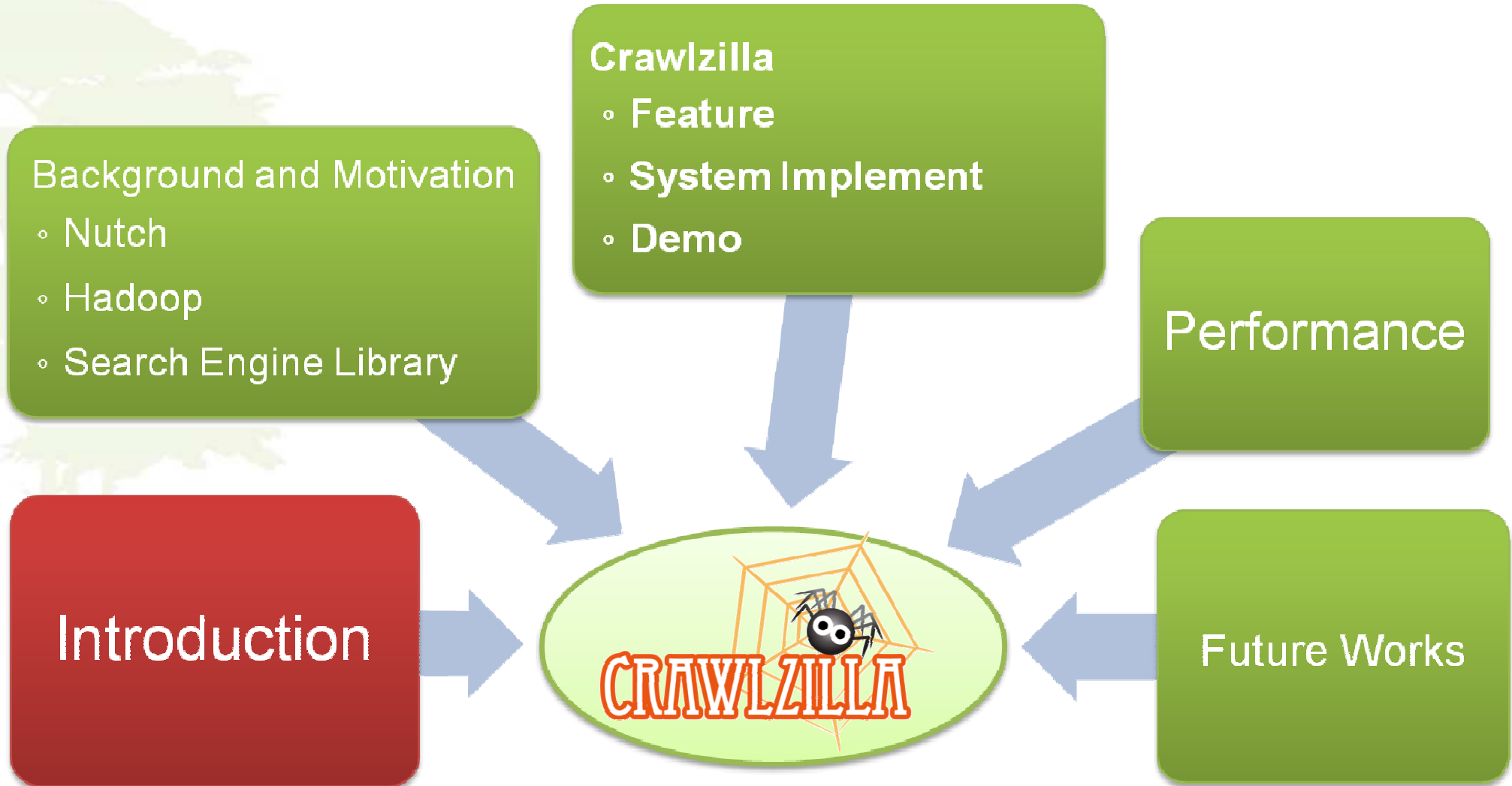
INVENSIVE 2011 May 23, 2011

TAIWAN

[www.nchc.org.tw](http://www.nchc.org.tw)  
National Applied  
Research Laboratories



# Outline

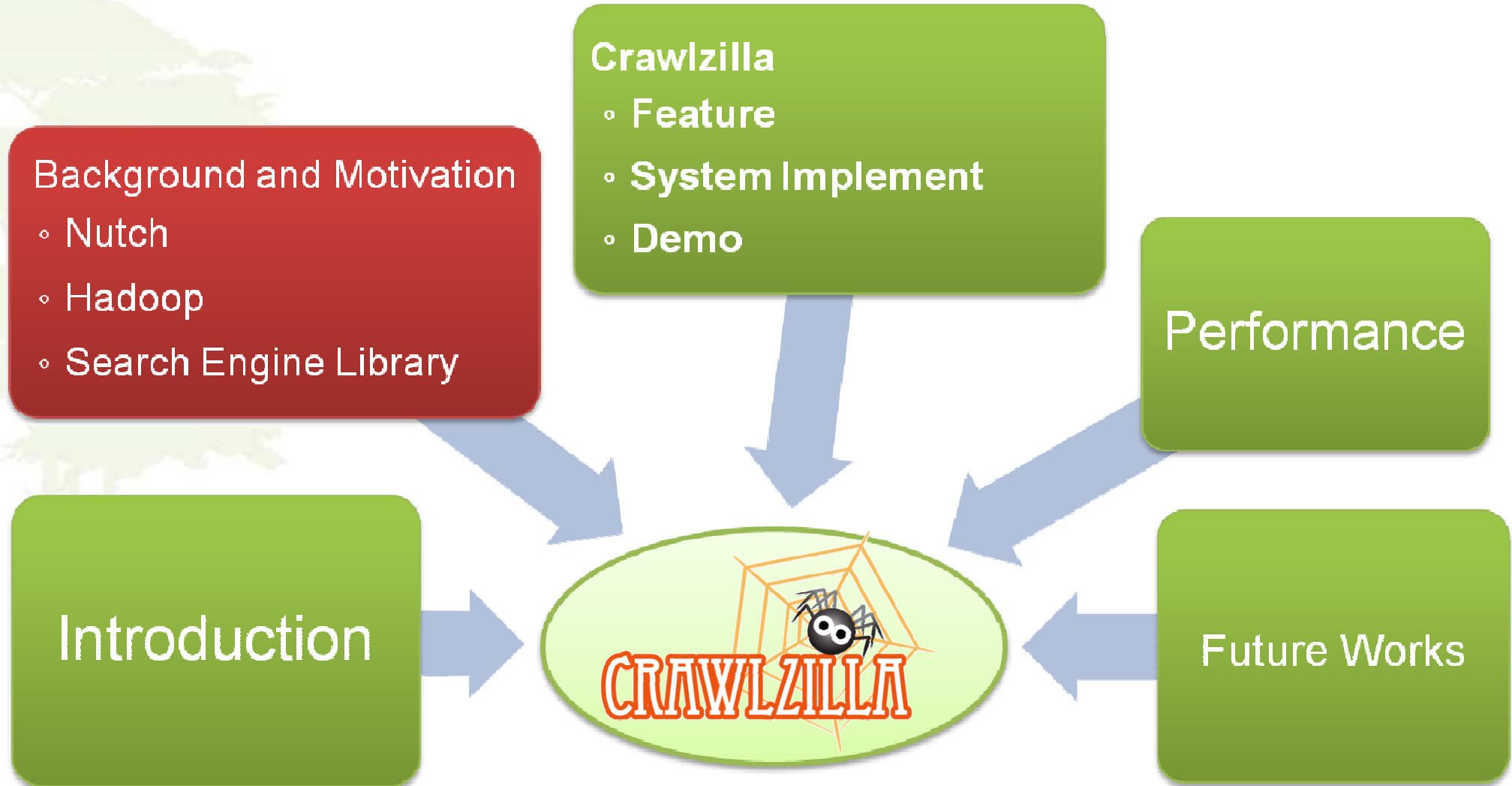


# Introduction

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



# Outline



# Background and Motivation

Search Engine workflow

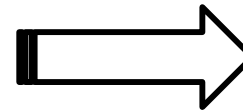
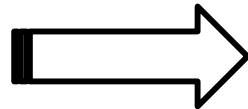
Related Open Source Projects

Compare with Other Projects



# Search Engine workflow – Phase 1

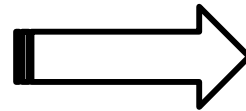
- **Crawling the Web**



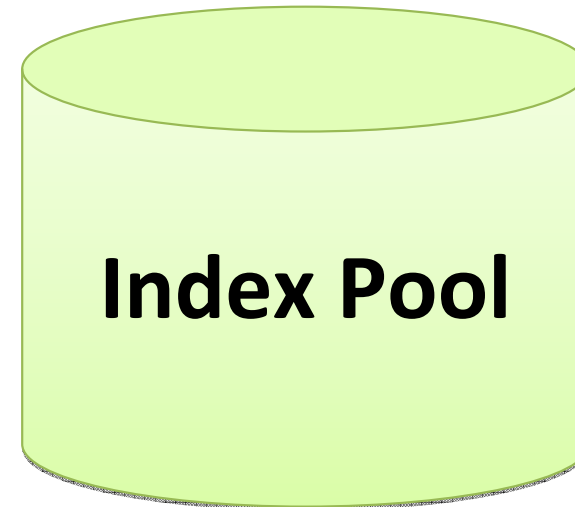
**Crawler visits the web pages of the links**

# Search Engine workflow – Phase 2

- **Building the Index Pool**

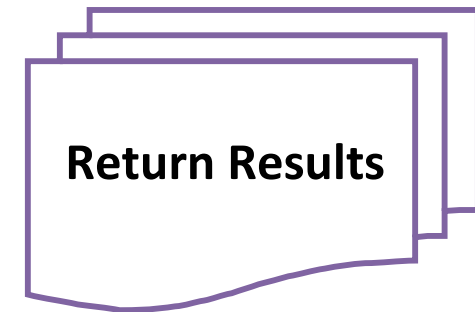
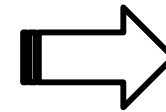
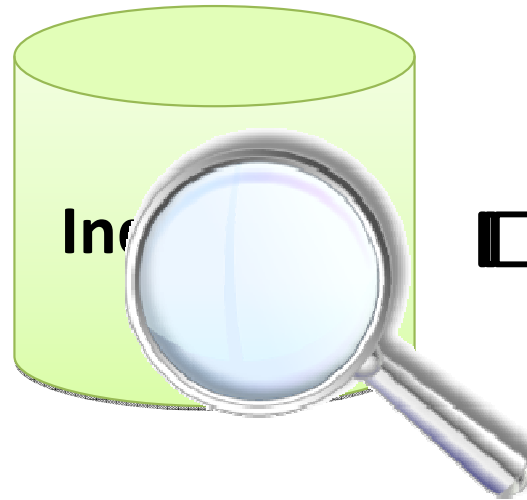
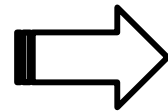


Parse Contents



# Search Engine workflow – Phase 3

- **Serving Queries**



User Sent a Query

Search from Index Pool

# Background and Motivation

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

# Background and Motivation

- **If Build Search by Yourself ...**

- Setup Hadoop
- Deploy System Configure Files
- Debug Errors...
- ...
- ...
- ...

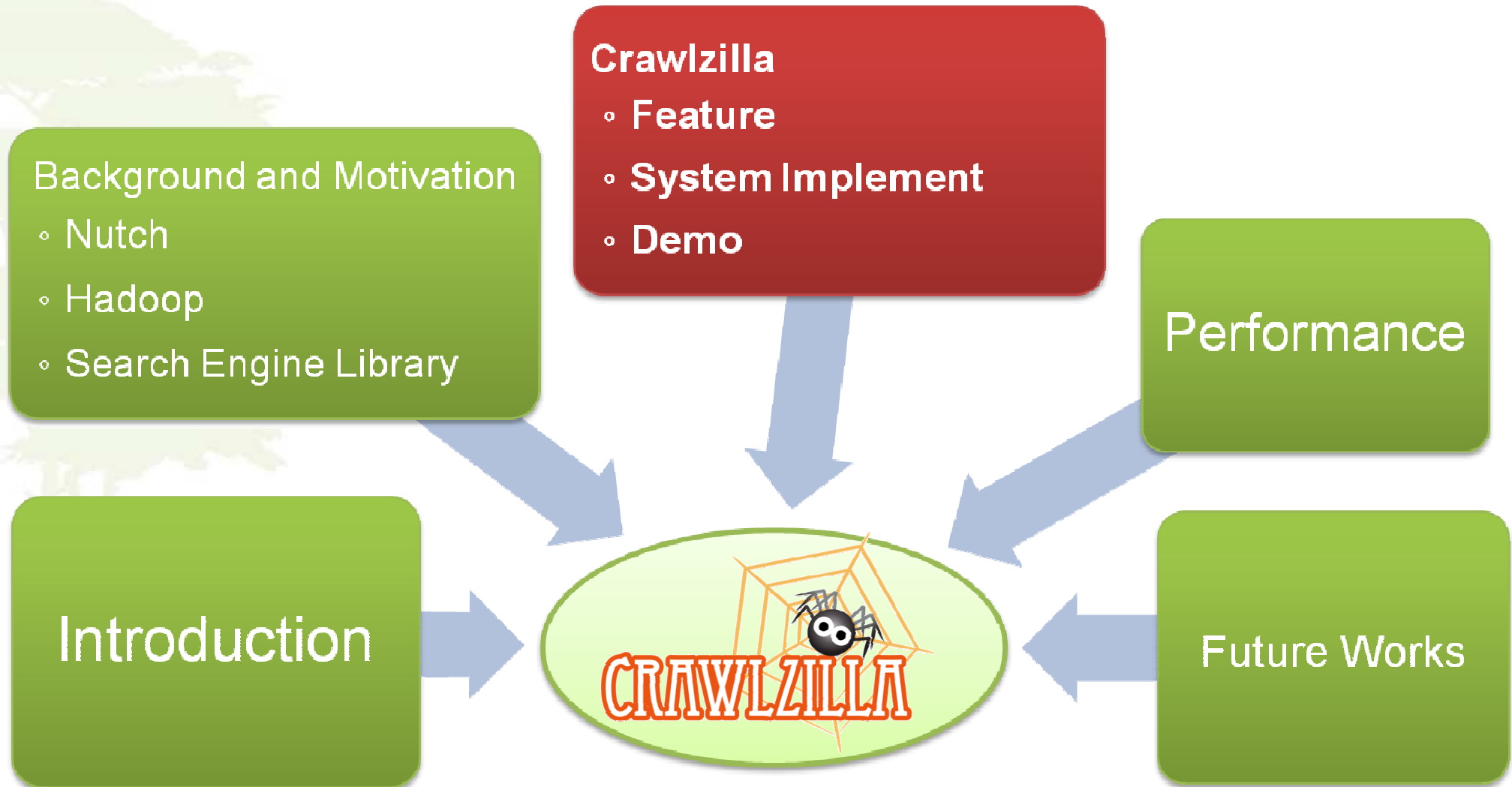
# Compare with Other Projects

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

# Goal

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

# Outline



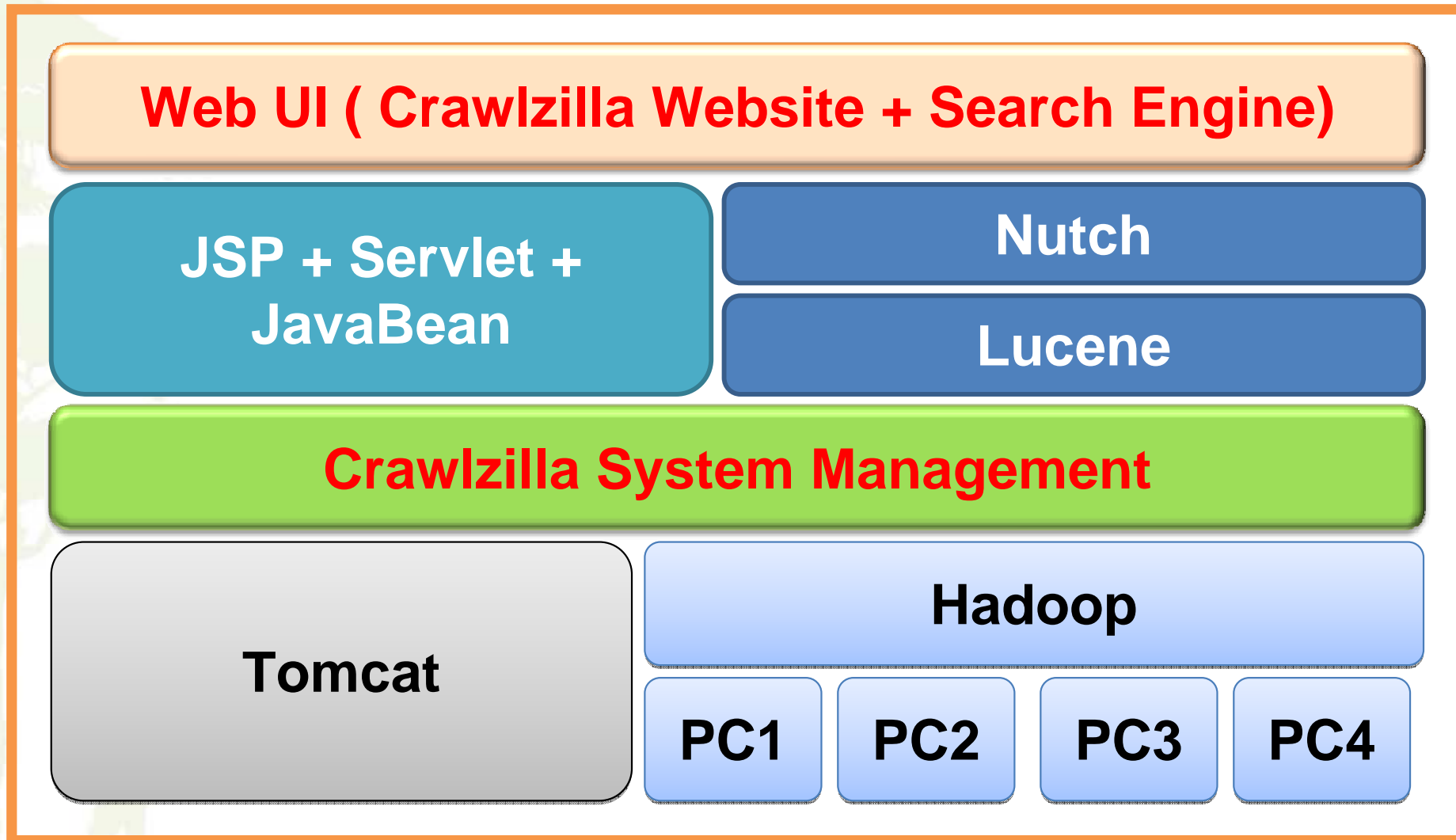


# Crawlzilla Feature

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



# Crawlzilla Architecture



# System Implement

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

# Web Management

(Model 2)



## Setup PW

這是你第一次登入安全考量, 預設的密碼不該被使用

原密碼為	●●●●●●
新設定的密碼	<input type="password"/>
確認新設定的密碼	<input type="password"/>

送出 重設

## Capture Lucene index pool

索引庫管理

索引庫名稱	建立時間	刪除索引庫	預覽統計資料	嵌入搜尋引擎到網頁的語法
nchc-en_3	2010-08-24 16:16:14	Delete	Preview	embed code
nchc-tw_3	2010-08-24 15:22:48	Delete	Preview	embed code

資料總覽

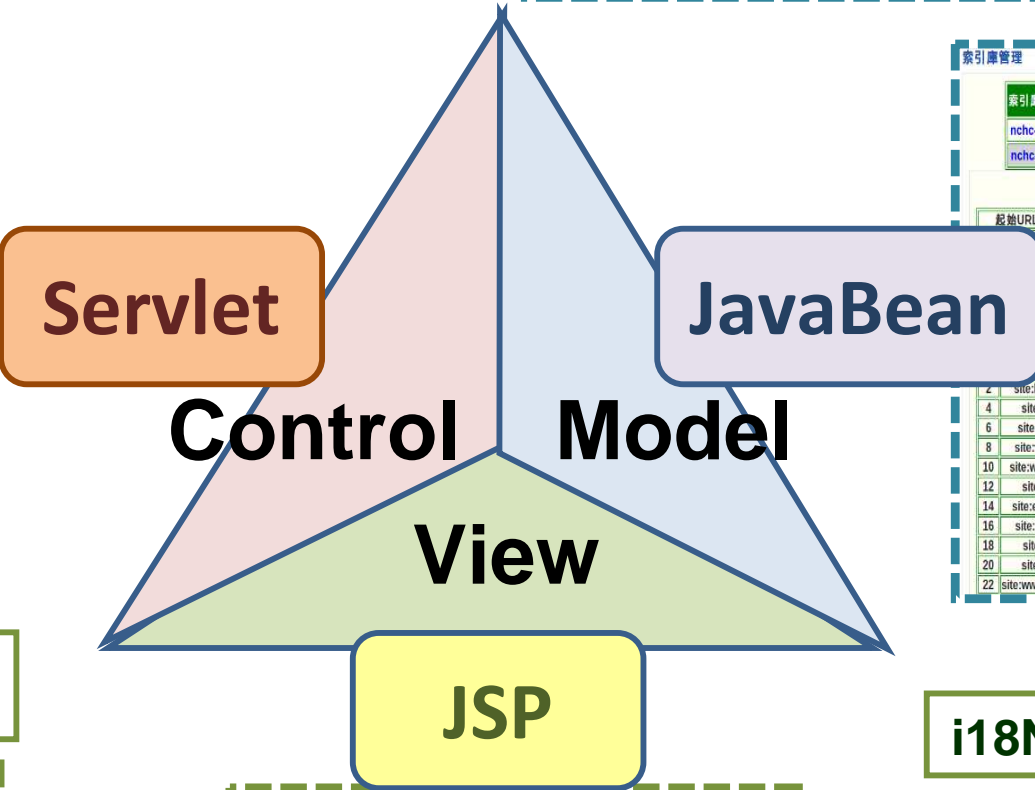
起始URL: http://www.nchc.org.tw/tw/

文件檔數量: 37095, 1036

使用者名稱: crawler

被搜尋分析到的網址:

內容	引用次數	排序	內容	引用次數
www.nchc.org.tw	336	1	site:pccluster.nchc.org.tw	87
site:bioinfo.nchc.org.tw	66	3	site:www.nar.org.tw	57
site:edu.nchc.org.tw	53	5	site:service.nchc.org.tw	35
site:acta.nchc.org.tw	28	7	site:colife.nchc.org.tw	14
site:wlanrc.nchc.org.tw	13	9	site:elib.nchc.org.tw	13
site:www.medicalgrid.org	13	11	site:volunteer.nchc.org.tw	9
site:www.stpi.org.tw	7	13	site:noc.twaren.net	7
site:ecogrid.nchc.org.tw	6	15	site:www.sipa.gov.tw	3
site:asp.104ehr.com.tw	3	17	site:viml.nchc.org.tw	3
site:www.ym.edu.tw	2	19	site:www.tnu.edu.tw	2
site:www.usc.edu.tw	2	21	site:www.ssvs.tp.edu.tw	2
site:www.smelearning.org.tw	2	23	site:ecocam.nchc.org.tw	2



## Setup and Drive Crawl Procedure

CrawlZilla Management Page

How To Use

1. Input Index Pool Name
2. Input URLs (see below example)
  - http://www.nchc.org.tw
  - http://www.google.com
  - http://code.google.com/j/harvest/
3. Choose Depth, then Submit!

Index Pool Name:

Input Crawl URLs:

Crawl Depth Setup: Choose Crawl Depth:

## System Status

系統狀態

Jobid	Priority	User	Name	Map % Complete	Map Total	Maps Completed	Reduce % Complete	Reduce Total	Reduce Completed
job_201009011521_0222	NORMAL	crawler	nchc NCHC-2_insegments/20100901134618	50.00%	2	1	0.00%	1	0

Completed Jobs

Jobid	Priority	User	Name	Map % Complete	Map Total	Maps Completed	Reduce % Complete	Reduce Total	Reduce Completed
job_201009011521_0222	NORMAL	crawler	nchc NCHC-2_insegments/20100901134618	100%	2	2	100%	1	1

## CrawlZilla 網頁管理介面

登入管理系統

請輸入管理者密碼

送出 重設

## i18N language setup

Home Crawl Manage System

Setup

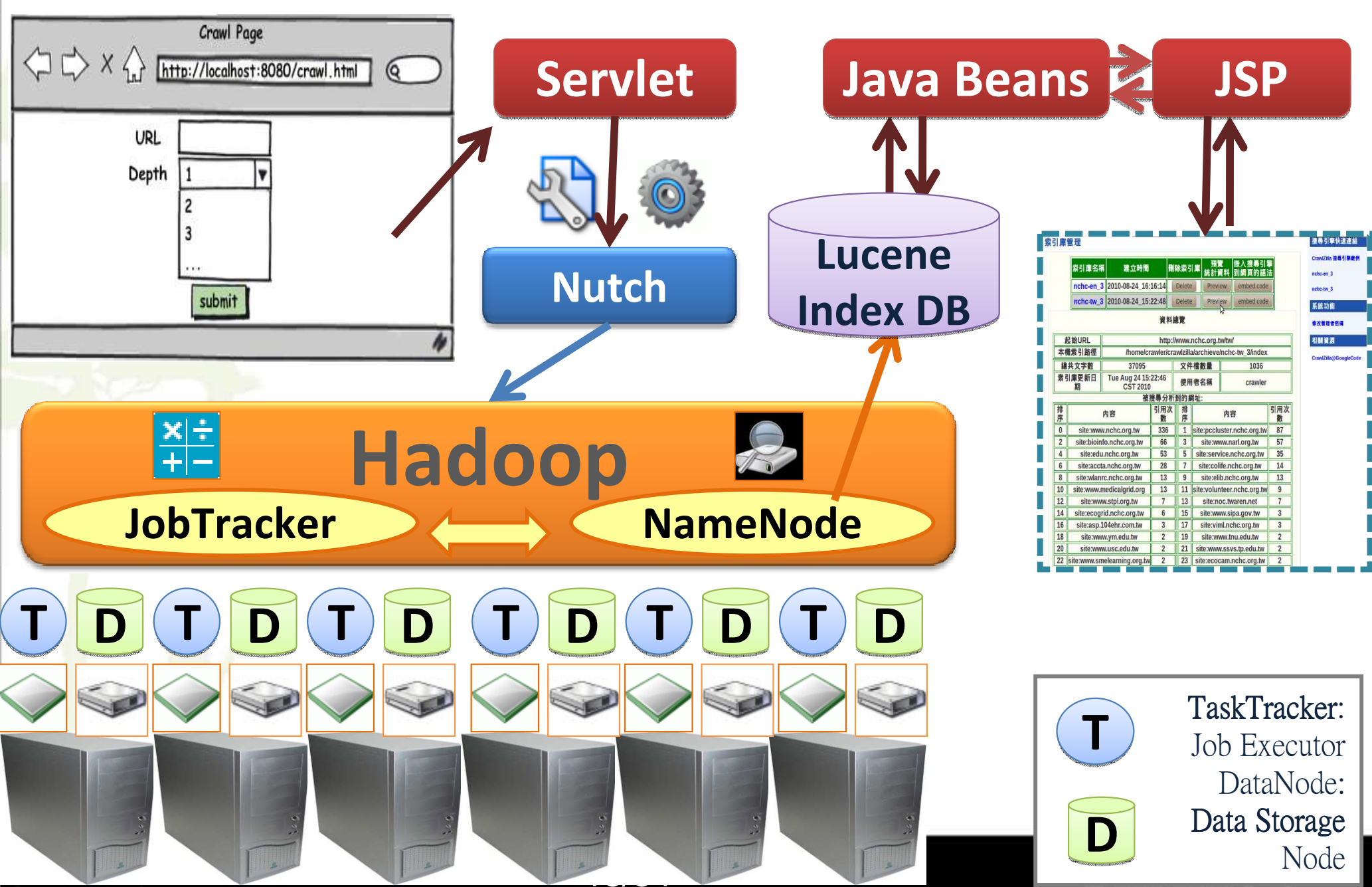
Engine Name:

Admin Email:

Choose Language:

submit

# System Implement – Web Parser



索引庫管理

索引庫名稱	建立時間	刪除索引庫	預覽	加入搜尋引擎統計資料	加入搜尋引擎引網頁的語法
nchc-en_3	2010-08-24 16:16:14	Delete	Preview	embed code	
nchc-tw_3	2010-08-24 15:22:48	Delete	Preview	embed code	

資料總覽

起始URL	http://www.nchc.org.tw/tw/
本機索引路徑	/home/crawler/crawlzilla/archieve/nchc-tw_3/index
總共文字數	37095
文件檔數量	1036
索引庫更新日期	Tue Aug 24 15:22:46 CST 2010
使用者名稱	crawler

被搜尋分析到的網址:

排序	內容	引用次數	排序	內容	引用次數
0	site:www.nchc.org.tw	336	1	site:pccluster.nchc.org.tw	87
2	site:bioinfo.nchc.org.tw	66	3	site:www.narl.org.tw	57
4	site:edu.nchc.org.tw	53	5	site:service.nchc.org.tw	35
6	site:accta.nchc.org.tw	28	7	site:colife.nchc.org.tw	14
8	site:wlanrc.nchc.org.tw	13	9	site:colife.nchc.org.tw	13
10	site:www.medicalgrid.org	13	11	site:volunteer.nchc.org.tw	9
12	site:www.stpl.org.tw	7	13	site:ncc.waren.net	7
14	site:ecogrid.nchc.org.tw	6	15	site:www.sipa.gov.tw	3
16	site:asp.104hr.com.tw	3	17	site:viml.nchc.org.tw	3
18	site:www.ym.edu.tw	2	19	site:www.tnu.edu.tw	2
20	site:www.usc.edu.tw	2	21	site:www.svsr.tp.edu.tw	2
22	site:www.smelearning.org.tw	2	23	site:ecocam.nchc.org.tw	2

TaskTracker:  
 Job Executor  
 DataNode:  
 Data Storage Node



# Live Demo I

## Crawlzilla Install

- (1) **Master** Install
- (2) Cluster **Slave** Install

# Live Video Demo

- Master Install ([Demo Video also @ YouTube](#))





# Live Video Demo

- Slave Install ([Demo Video also @ YouTube](#))



A faint, light green silhouette of a tree and a person with arms raised, located on the left side of the slide.

# Live Demo II

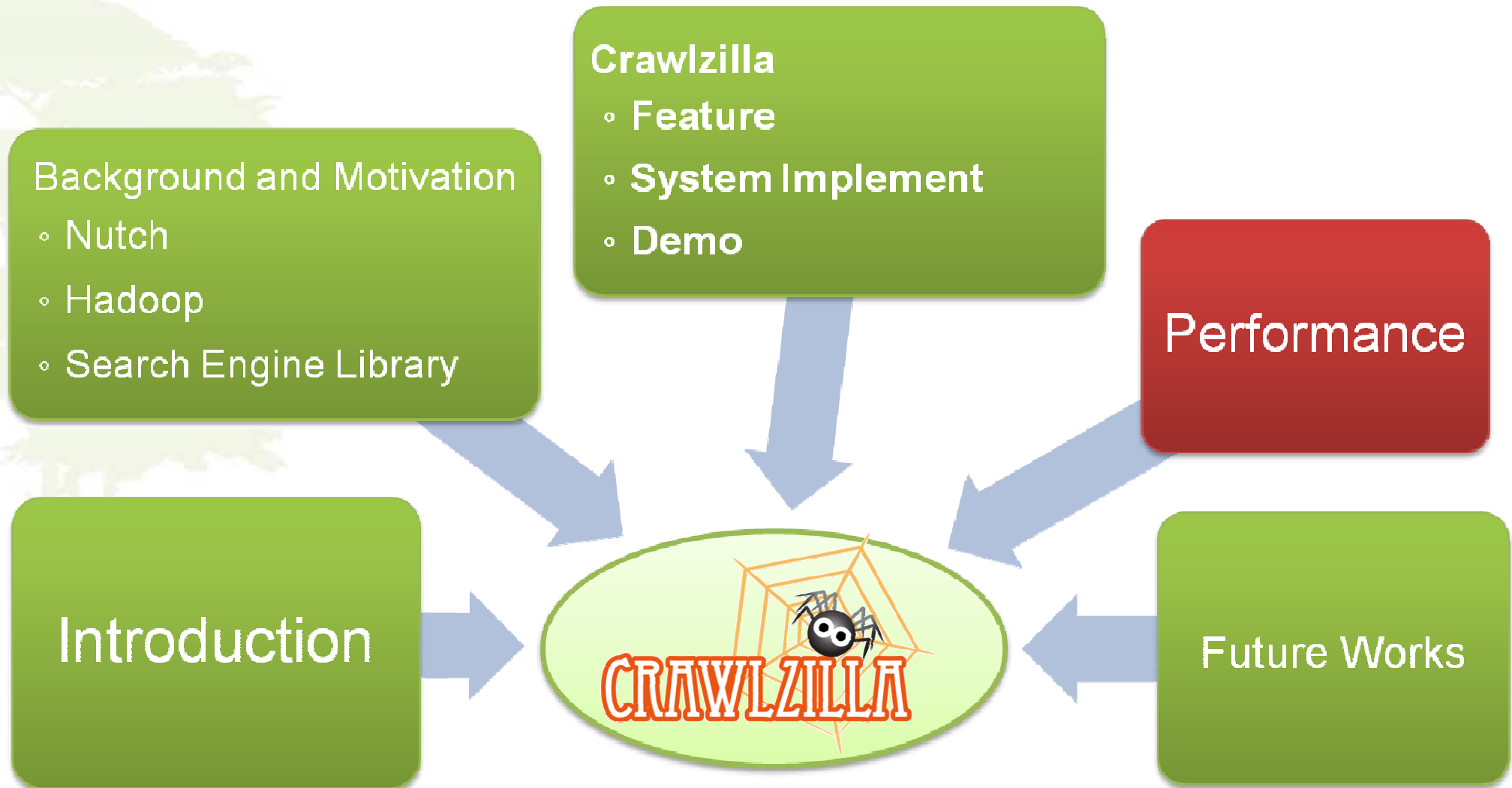
## Dialog Management

# Live Demo III

## Web Management

- (1) Crawl Setup
- (2) Search Engine **Index Pool**
- (3) **Search it!**

# Outline



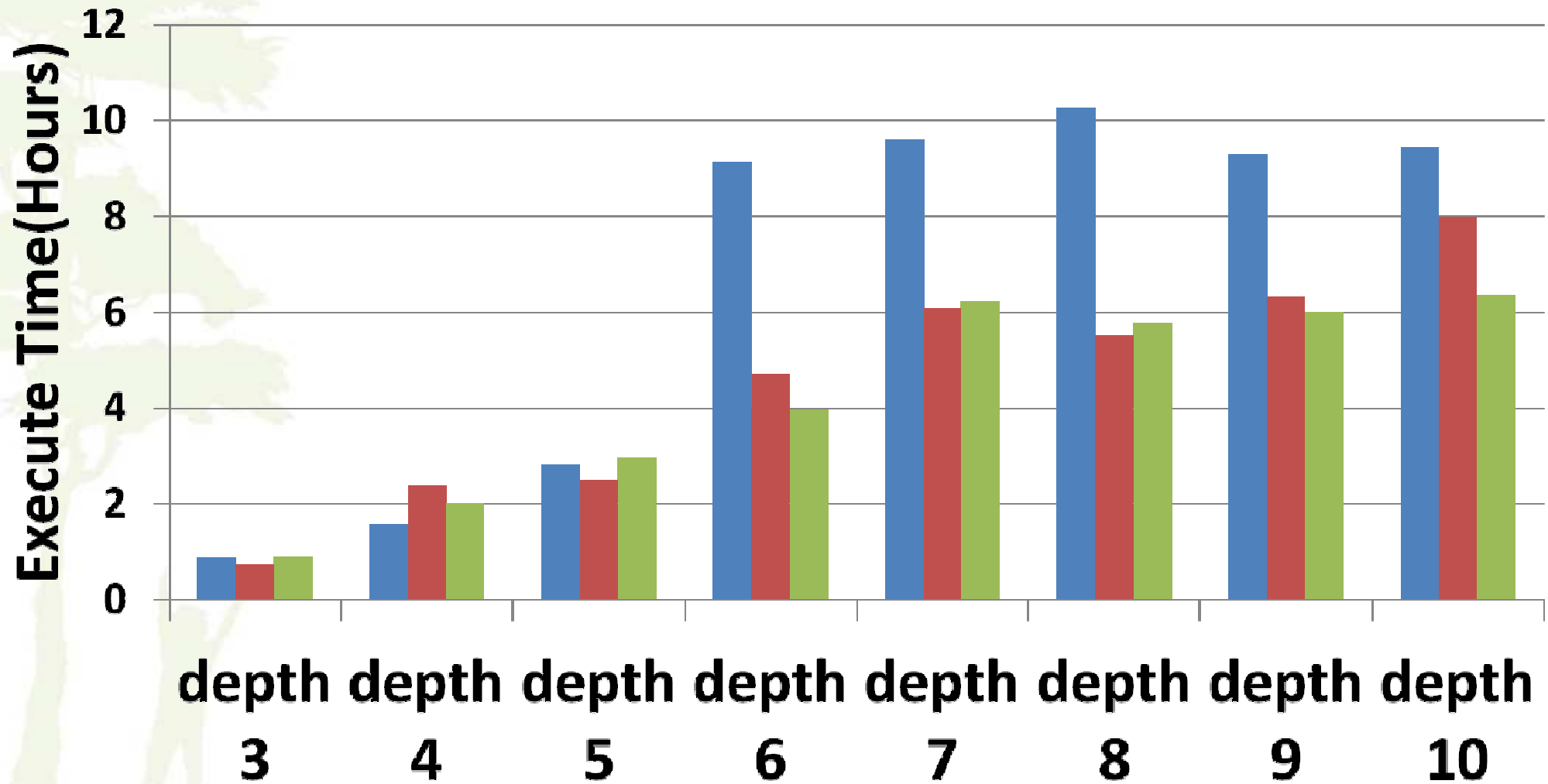
# Performance

## Experiment Environment

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

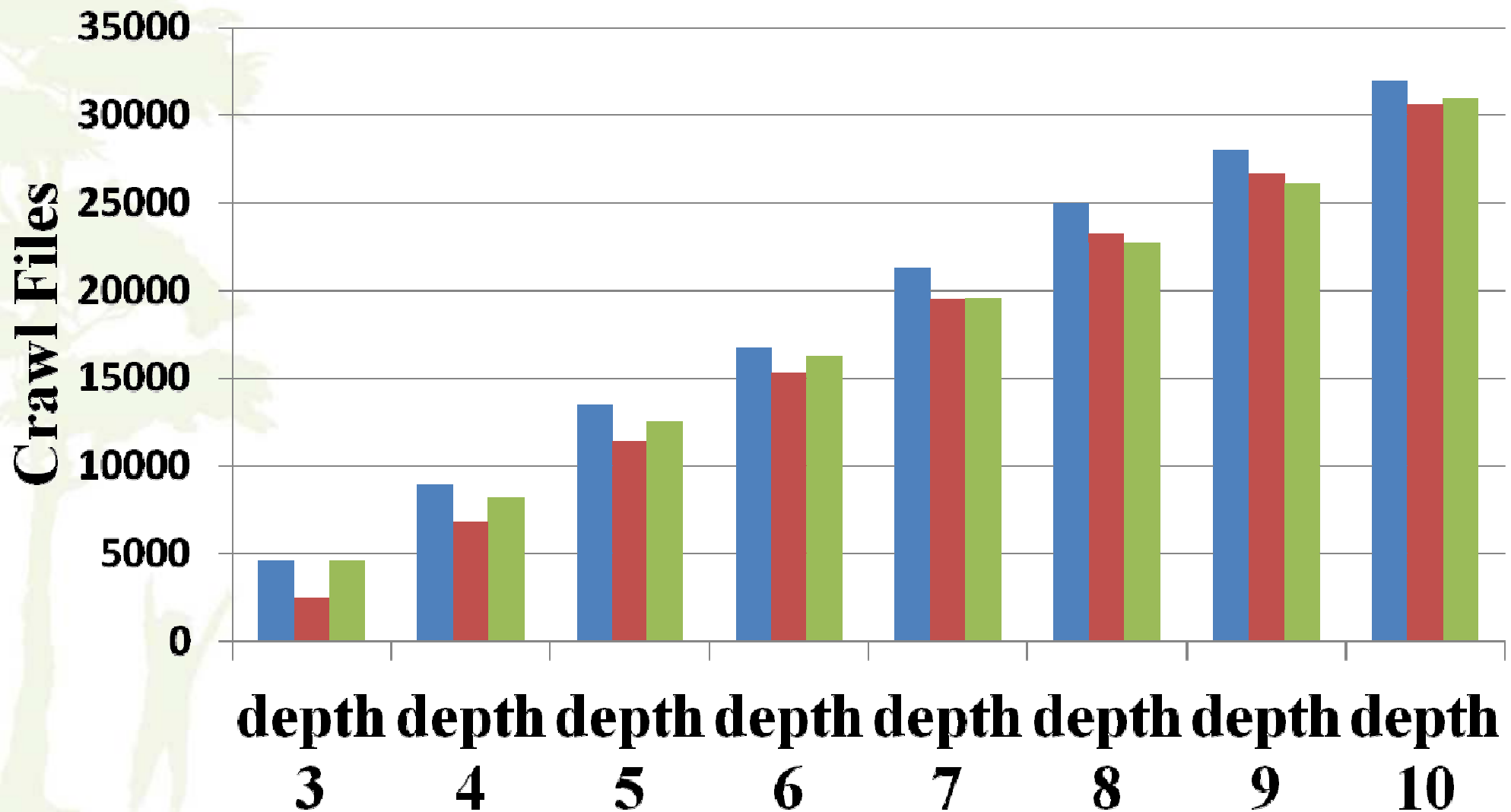
# Execute Time

■ node number 1 ■ node number 3 ■ node number 6



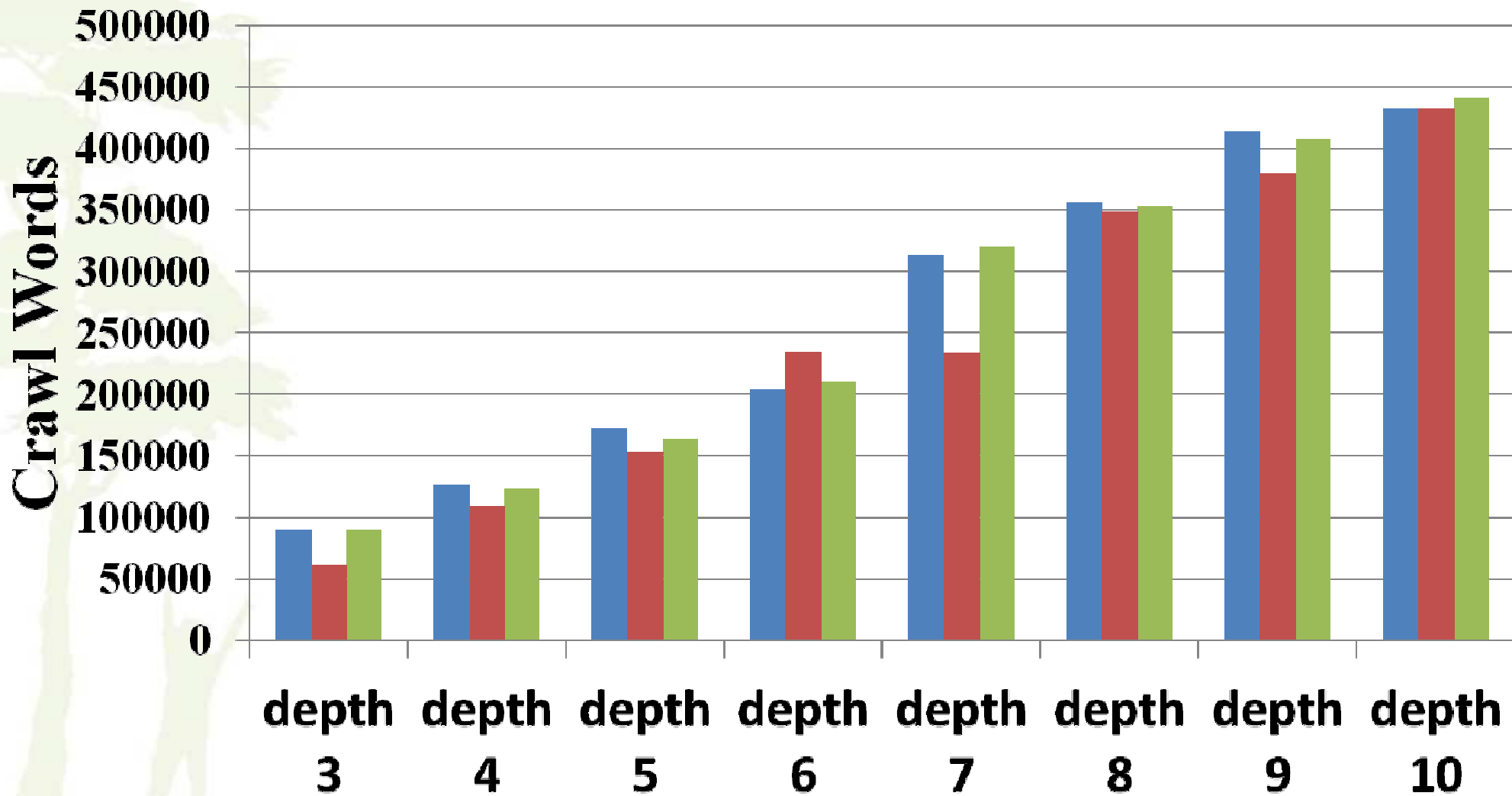
# Crawl Files

■ node number 1 ■ node number 3 ■ node number 6



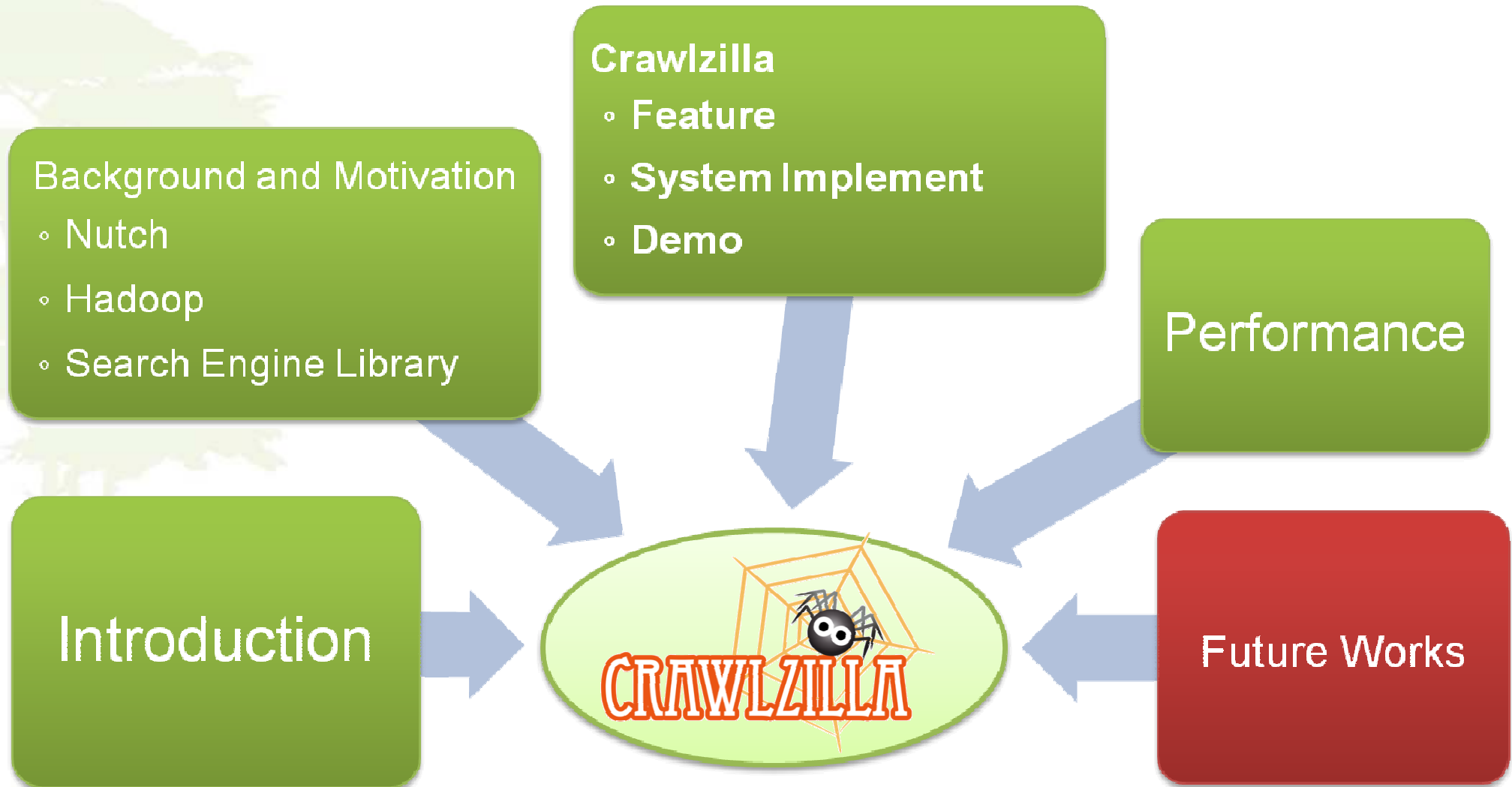
# Crawl Words

■ node number 1 ■ node number 3 ■ node number 6





# Outline



# Future Works

- **New Version**

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

# Reference

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

# Enjoy your search engines!!! Start from Here!



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

# Thank You!

## Q & A

