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 Neqotiation, Ddmission Control, Pricing, SLA Management, Metering…	OpenNebula, Enomaly, Eucalyptus , OpenQRM, …
虛擬化 Virtualization VM, VM management and Deployment	Xen, KVM, VirtualBox, QEMU, OpenVZ, …
硬體設施 Hardware Infrastructure: Computer, Storage, Network	

Hadoop

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

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

Sector / Sphere

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

Sector-Sphere

National Center for Data Mining
University of Illinois at Chicago
UIC

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

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。

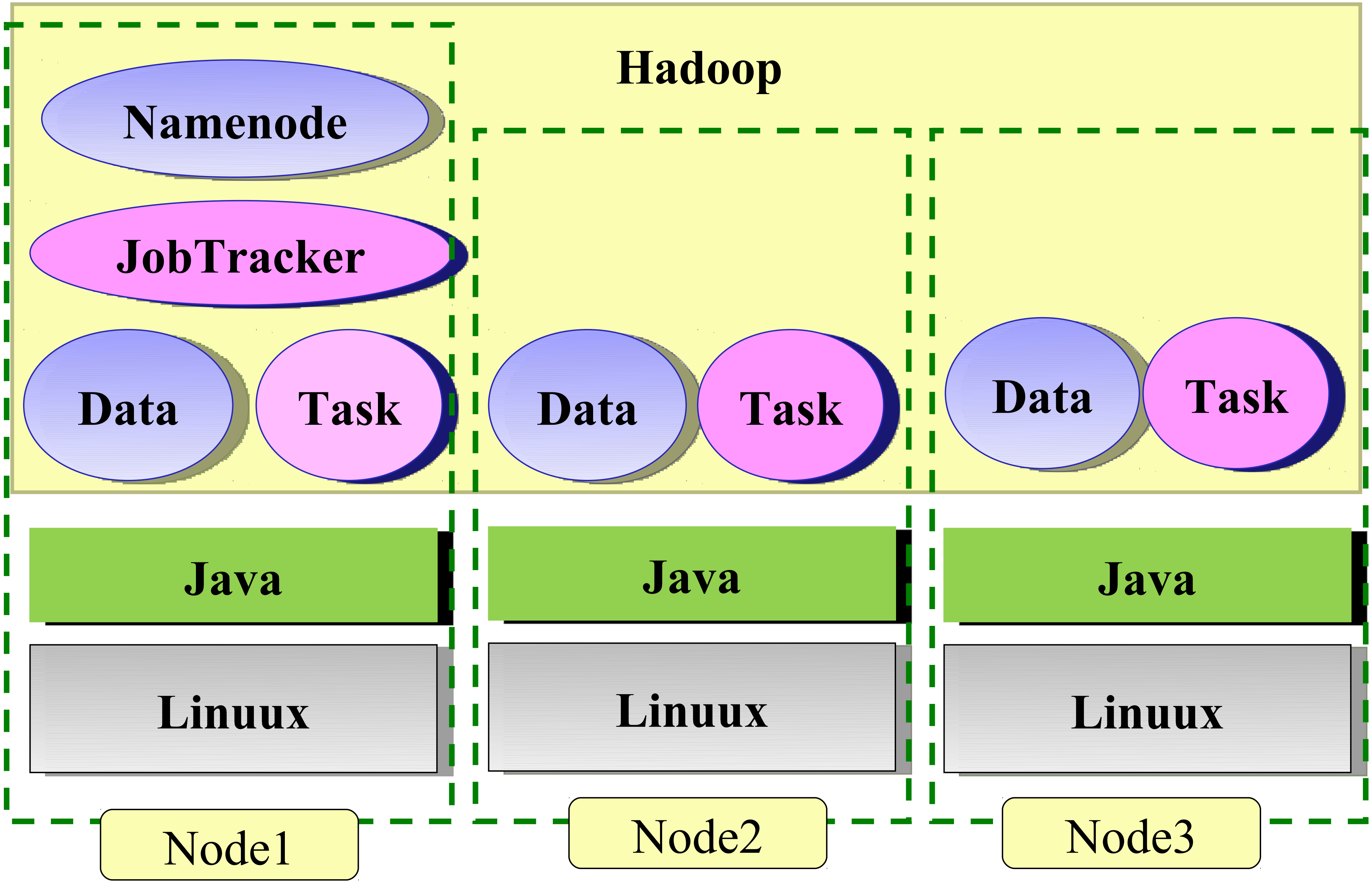
亦可以想像成一個處理海量資料的生產線，只須學會定義 *Map* 跟 *Reduce* 工作站該做哪些事情。

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

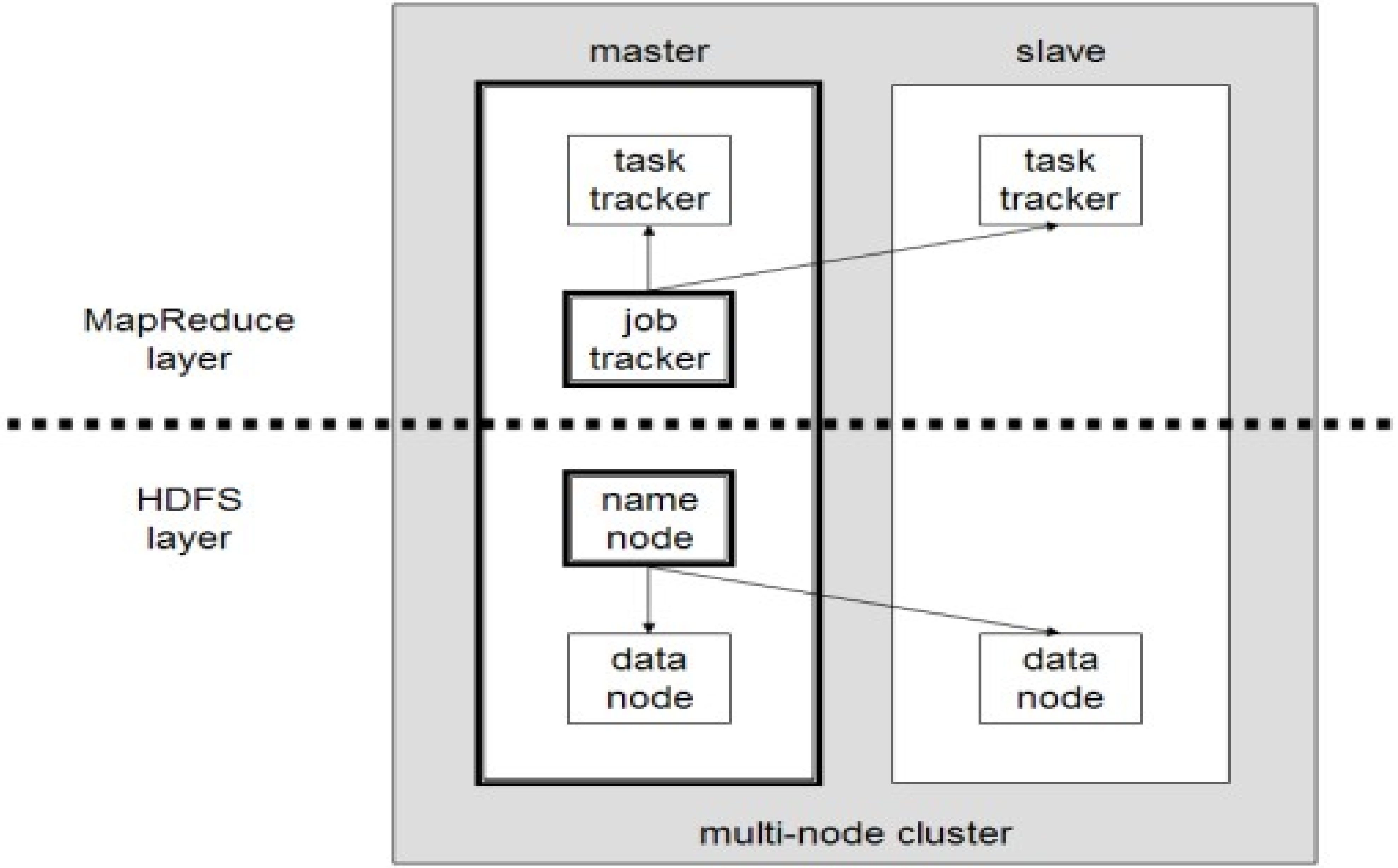
Scheduler
程序排程

File System
檔案系統

Distributed Operating System of Hadoop
Hadoop 建構成一個分散式作業系統



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



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

名稱節點 NameNode

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

資料節點 DataNode

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

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

JobTracker

- Master Node

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

•只能有一個

TaskTracker

- Worker Nodes

•運作 Map/Reduce 的工作
•管理儲存、回覆運算結果

• 可多個

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.

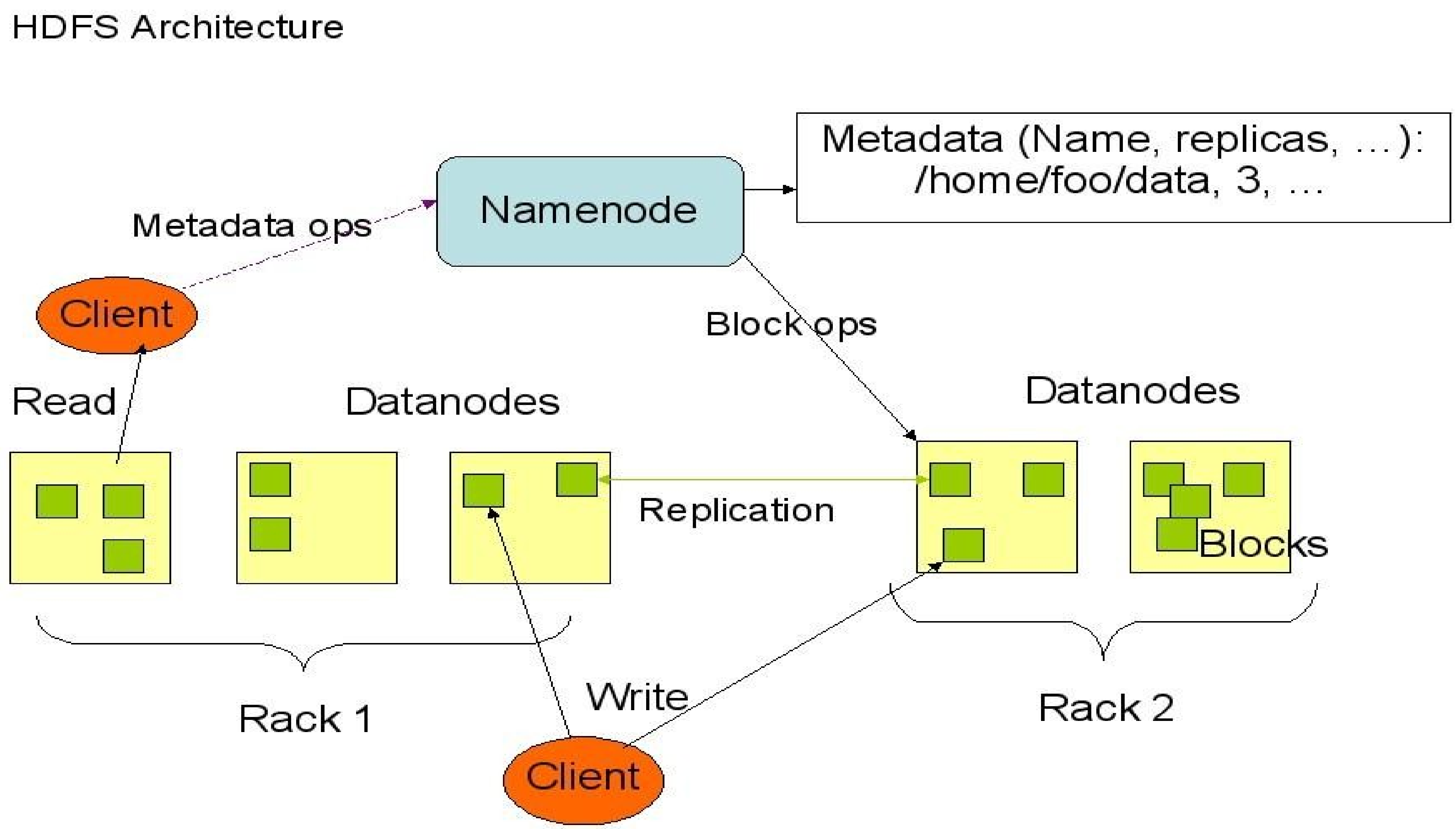
Features of HDFS …
HDFS 的特色是 …

- 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 …

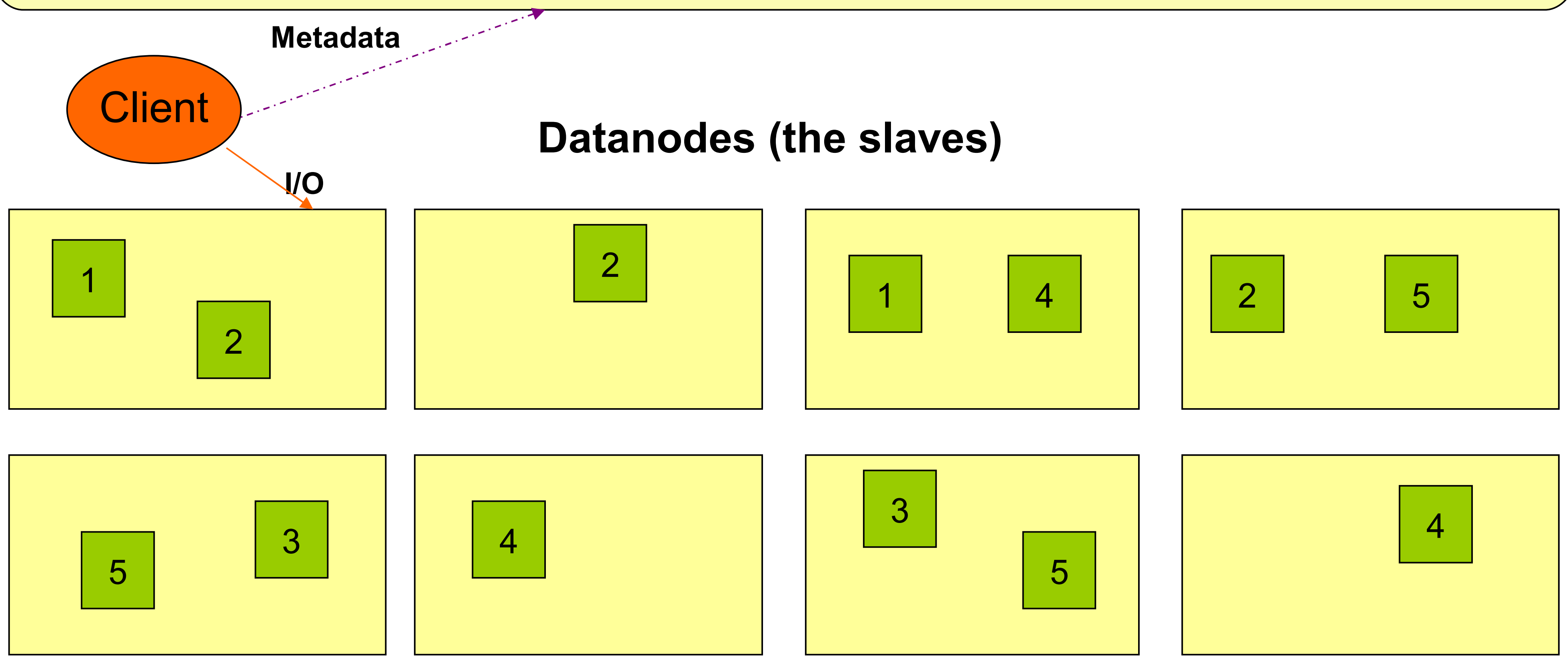


How does HDFS work …
HDFS 如何運作 …

Namenode (the master)

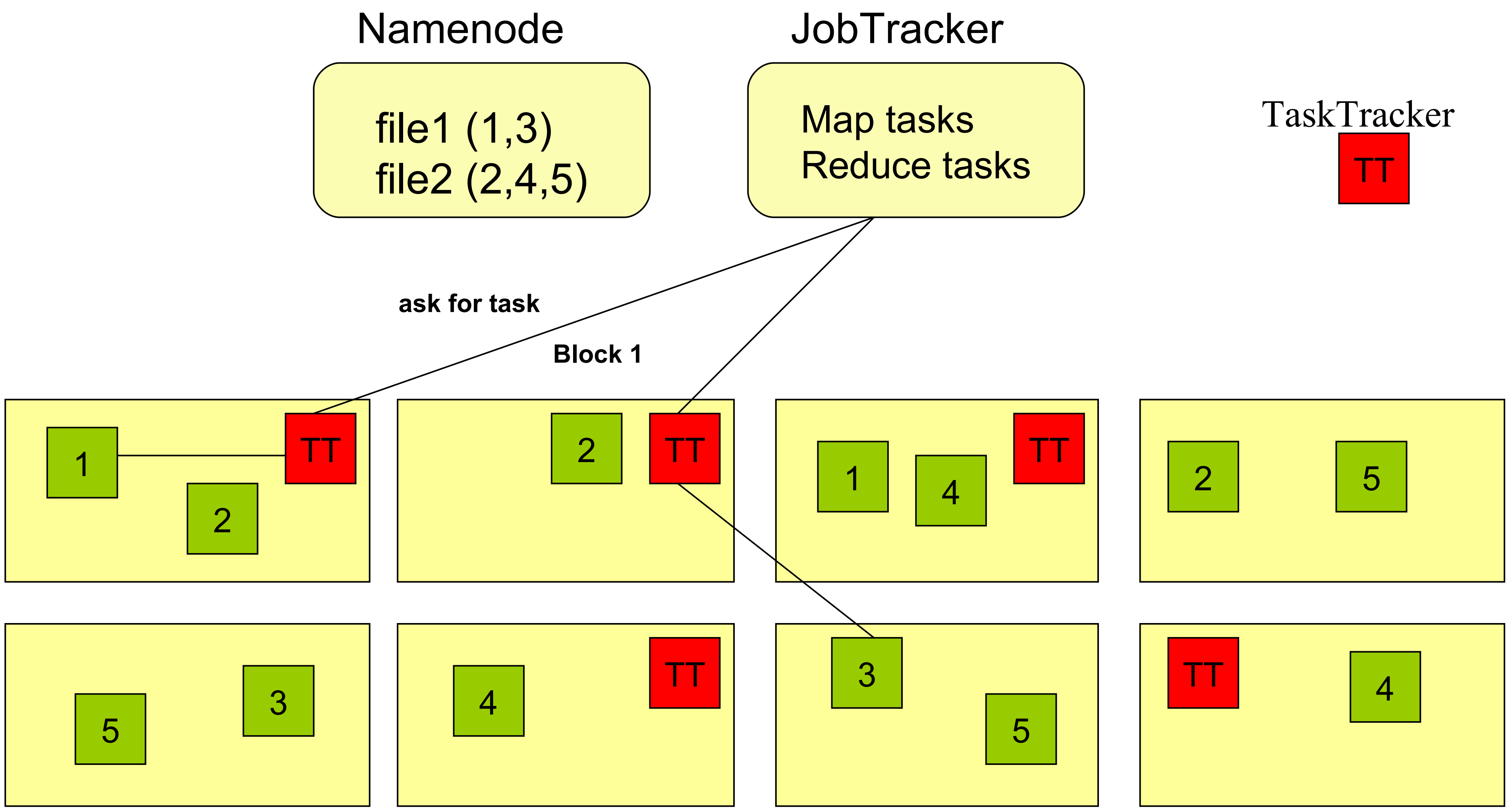
Path and Filename – Replication , blocks

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



About Data locality …
HDFS 如何達成在地運算 …

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



About Fault Tolerance …
HDFS 如何達成容錯機制 …

資料崩毀
Data Corrupt

網路或資料
節點失效
Network Fault
DataNode Fault

名稱節點錯誤
NameNode Fault

- 資料完整性 Data integrity
  - checked with CRC32
  - 用副本取代出錯資料
  - Replcae corrupt block with replication one
- Heartbeat
  - Datanode send heartbeat to Namenode
- Metadata
  - FSImage 、 Editlog 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]]
```

Questions?

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

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

Powered by DRBL

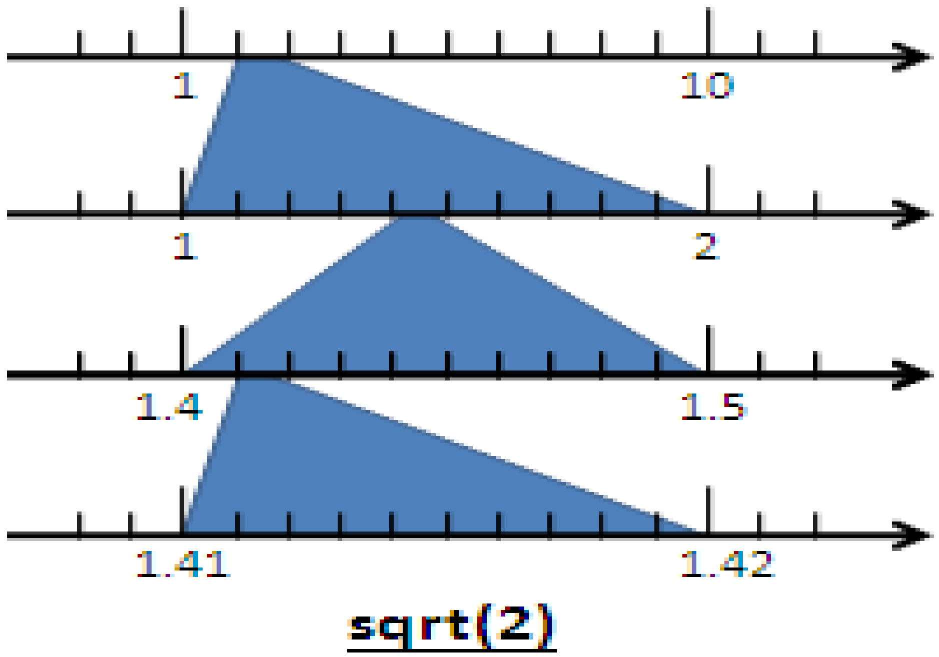
MapReduce 簡介

Introduction to MapReduce

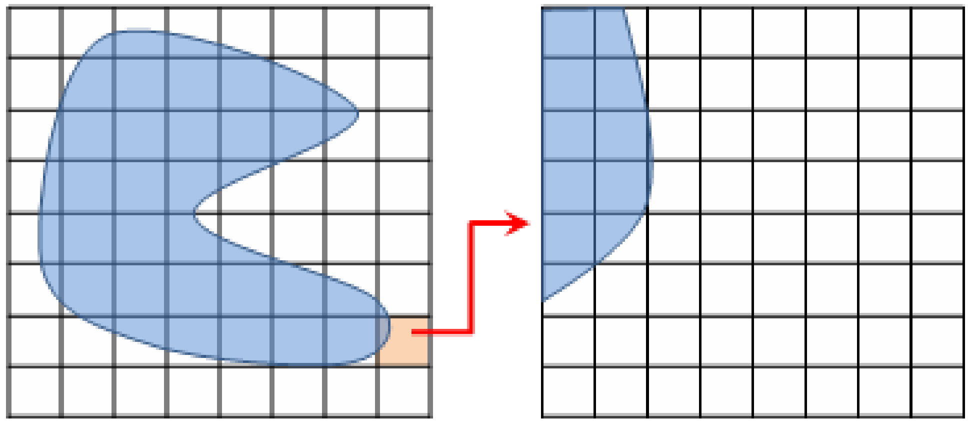
Jazz Wang
Yao-Tsung Wang
jazz@nchc.org.tw

Divide and Conquer Algorithms
分而治之演算法

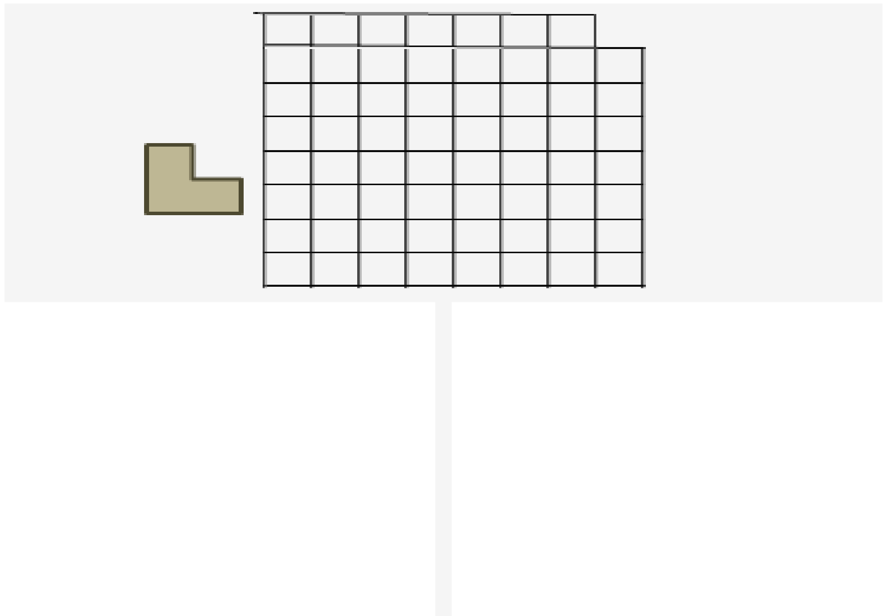
Example 1:



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？

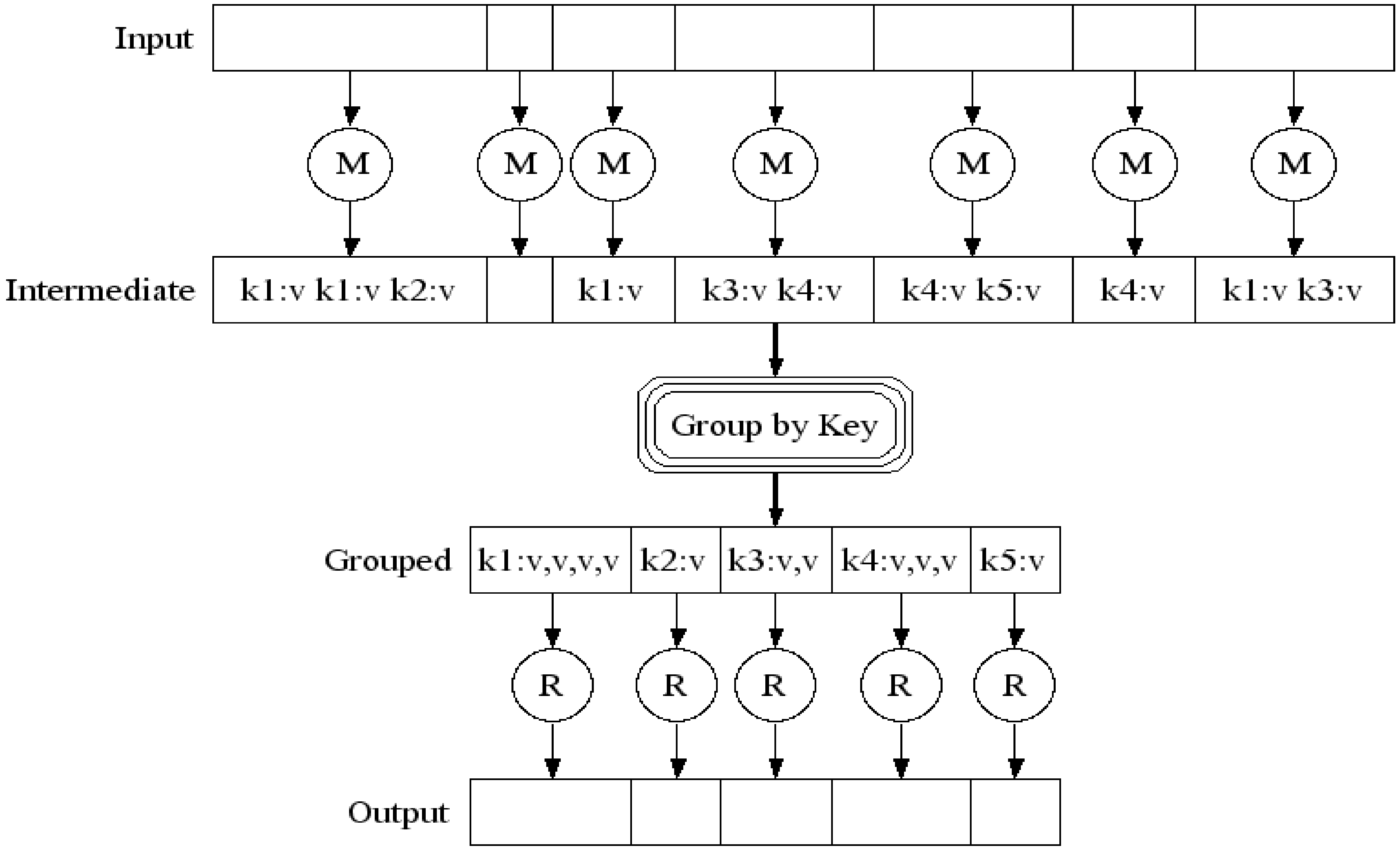
Ex： (1,1,1,1,1) or (1,2,1,1)

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 → N
    - Ex. [1,2,3,4] – (*2) -> [2,4,6,8]
  - Reduce(...): N → 1
    - [1,2,3,4] - (sum) -> 10
- Logical view of MapReduce
    - Map(k1,v1) -> list(k2,v2)
    - Reduce(k2, list (v2)) -> list(v3)

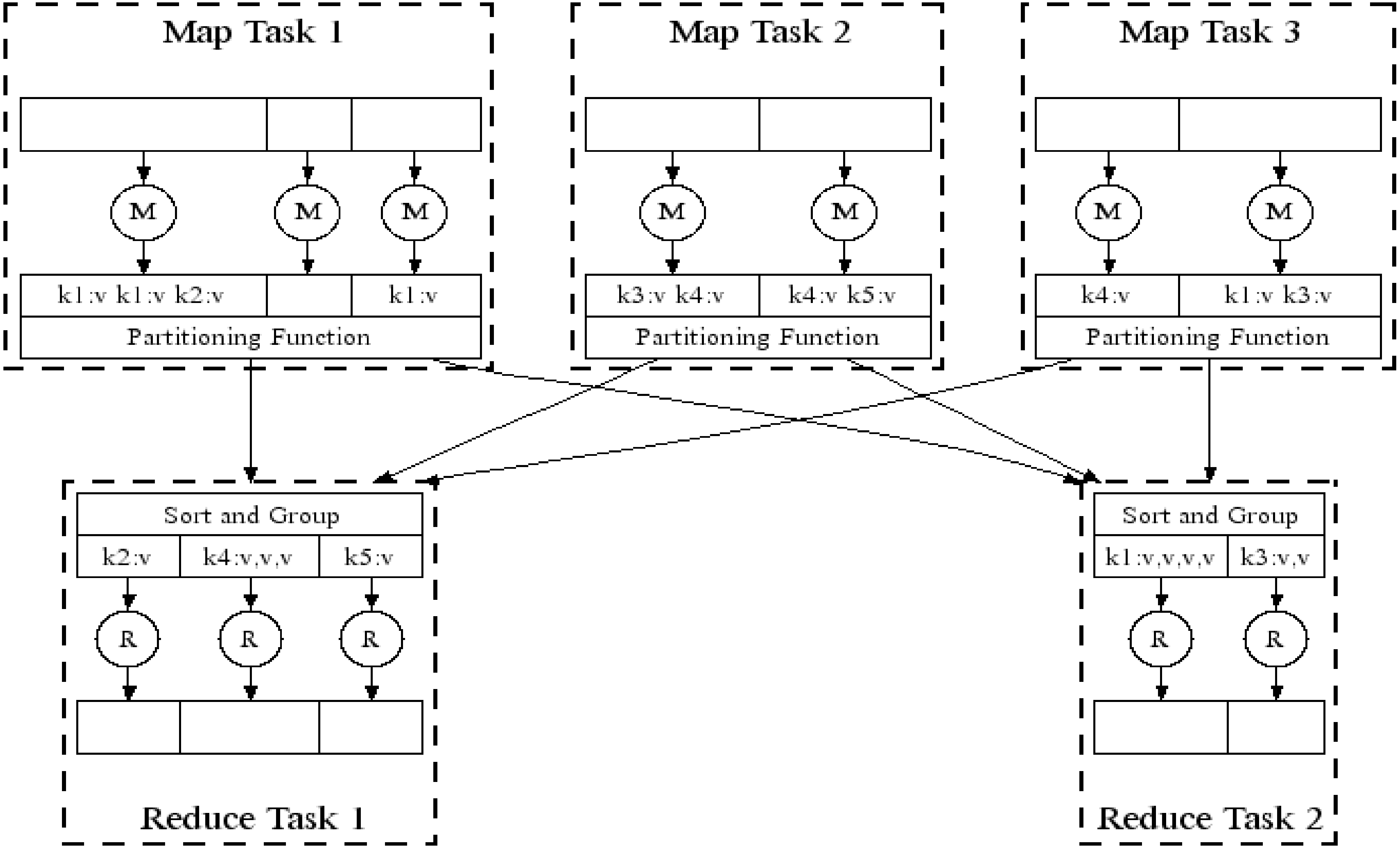
Source: http://en.wikipedia.org/wiki/MapReduce

Google's MapReduce Diagram
Google 的 MapReduce 圖解



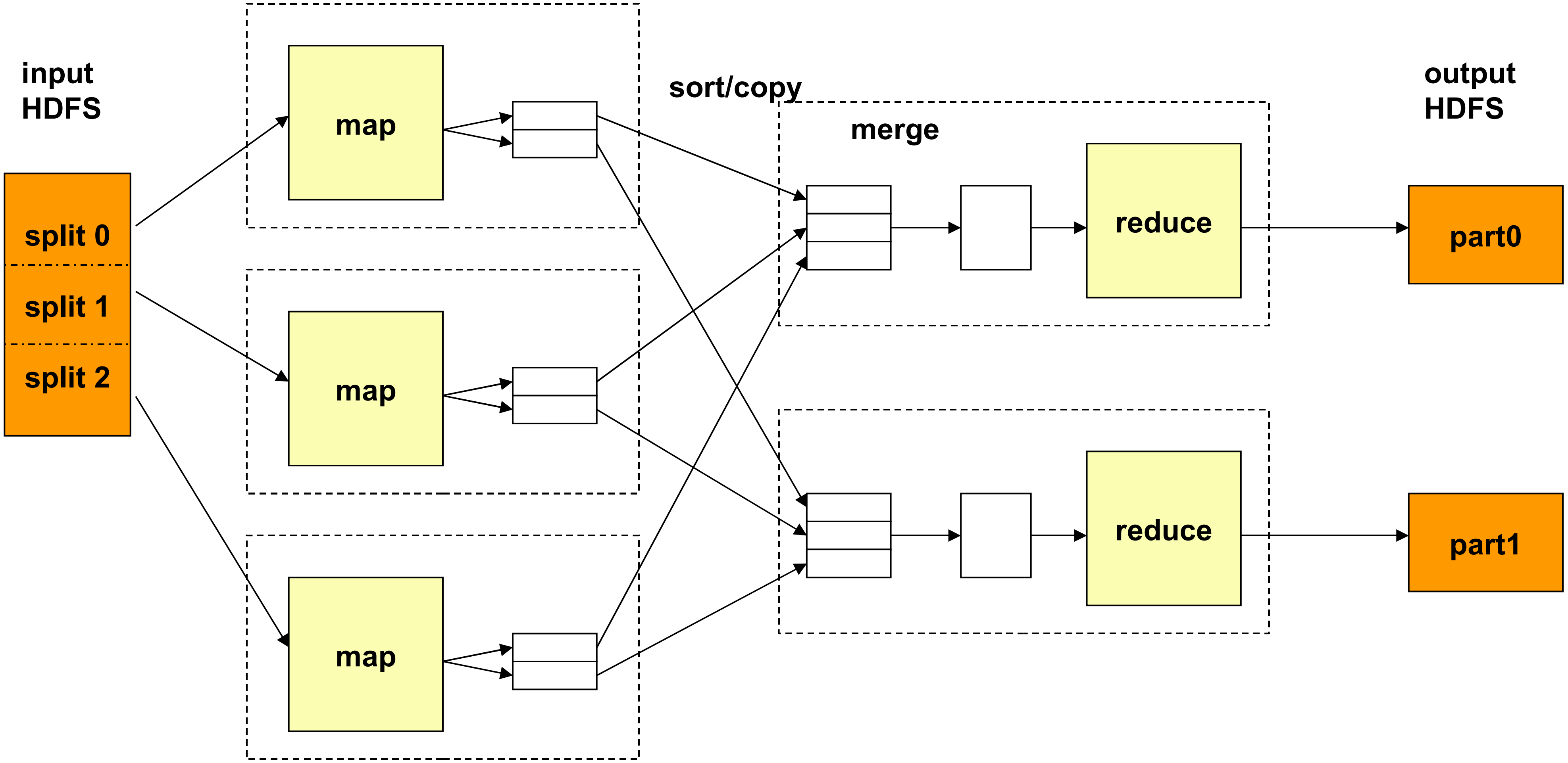
Google's MapReduce in Parallel

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

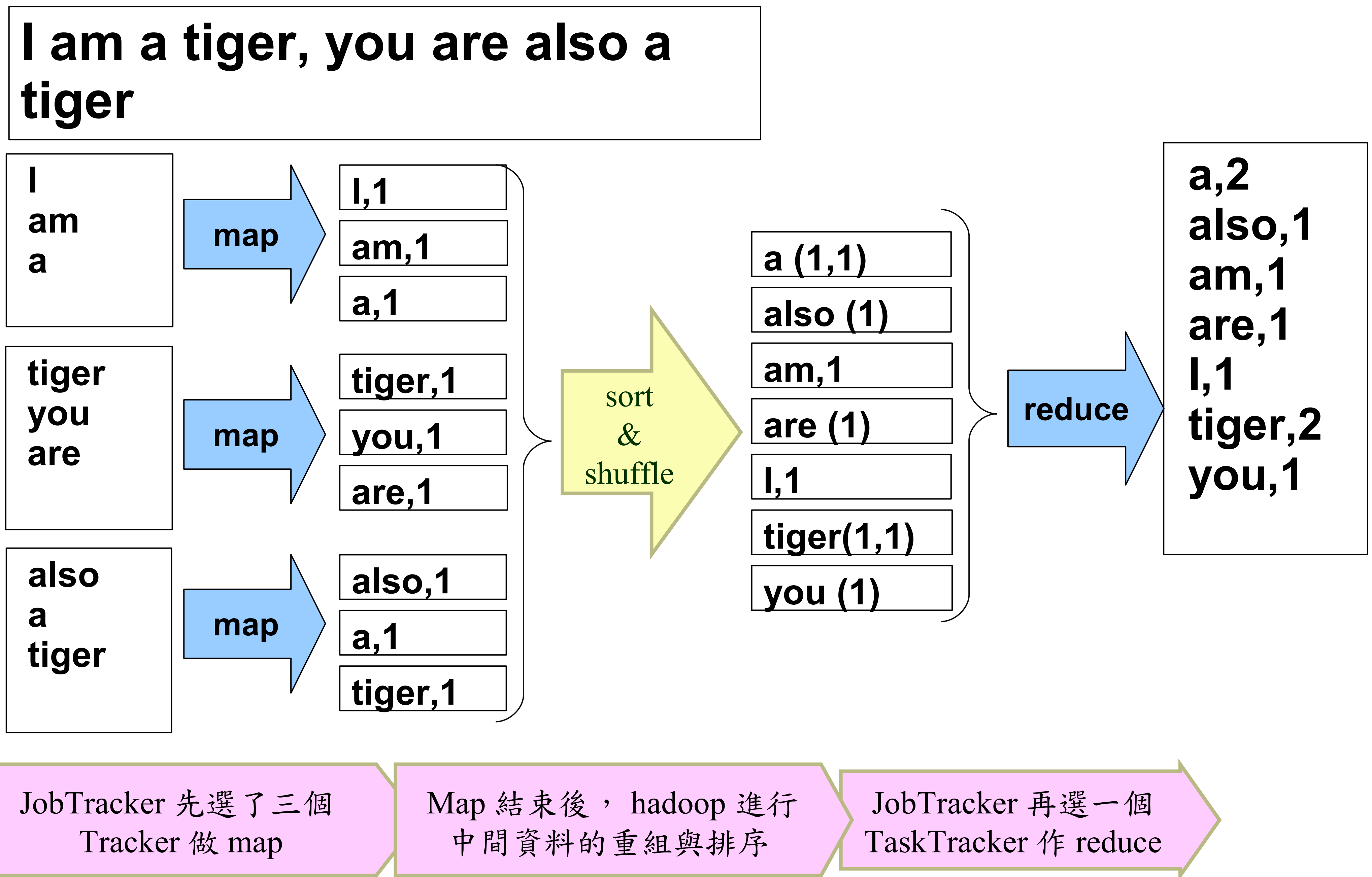
JobTracker 選數個 TaskTracker 來作 Map 運算，產生些中間檔案

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)

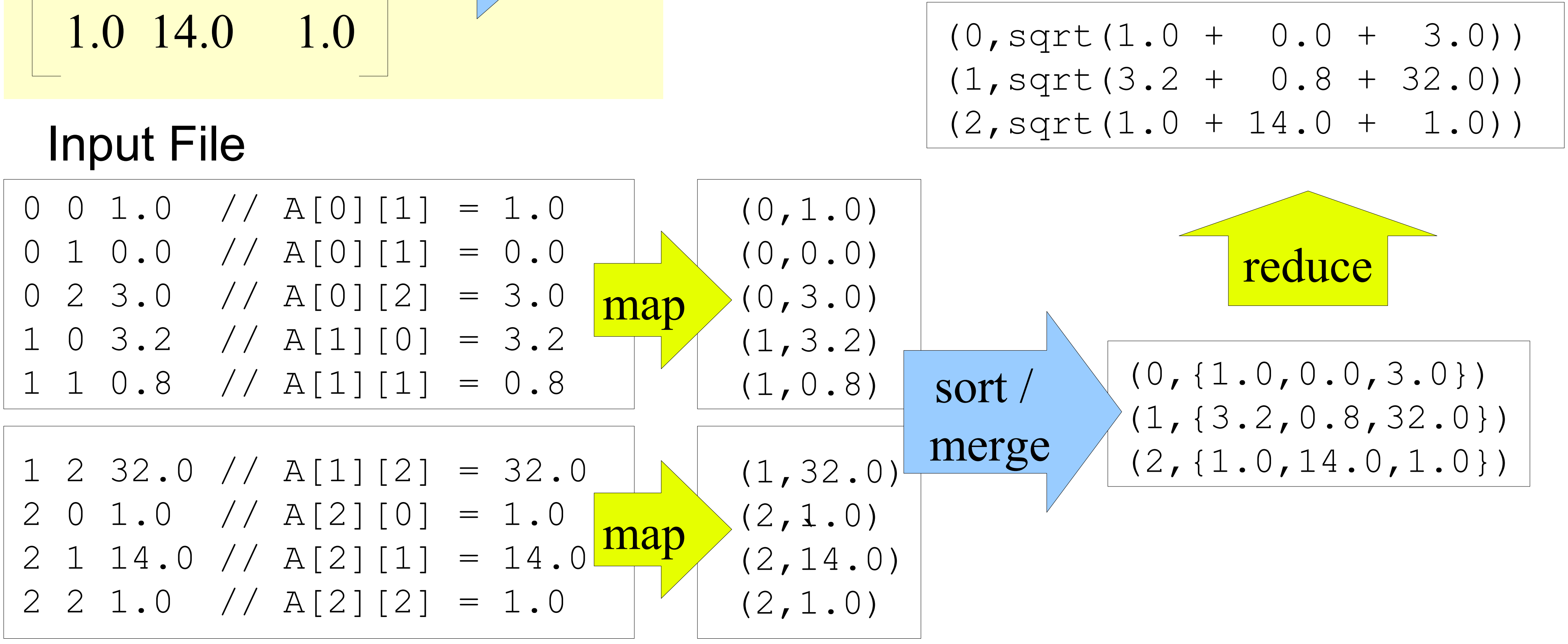


MapReduce by Example (2)

MapReduce 運作實例 (2)

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

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



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
```

```
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
```

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

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

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

```
(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

MapReduce 程式設計入門

MapReduce Programing 101

Jazz Wang

Yao-Tsung Wang

jazz@nchc.org.tw

Program Prototype (v 0.20)

Map 區

Reduce 區

設定區

```
Class MR{
      static public Class Mapper …{
      }
       static public Class Reducer …{
      }
      main(){
          Configuration conf = new Configuration();
          Job job = new Job(conf, “job name");
          job.setJarByClass(thisMainClass.class);
          job.setMapperClass(Mapper.class);
          job.setReduceClass(Reducer.class);
          FileInputFormat.addInputPaths(job, new Path(args[0]));
          FileOutputFormat.setOutputPath(job, new Path(args[1]));
            Other configure 其他的設定參數程式碼
          job.waitForCompletion(true);
}}
```

Mapper 程式碼

Reducer 程式碼

Program Prototype (v 0.18)

Map 區

Reduce 區

設定區

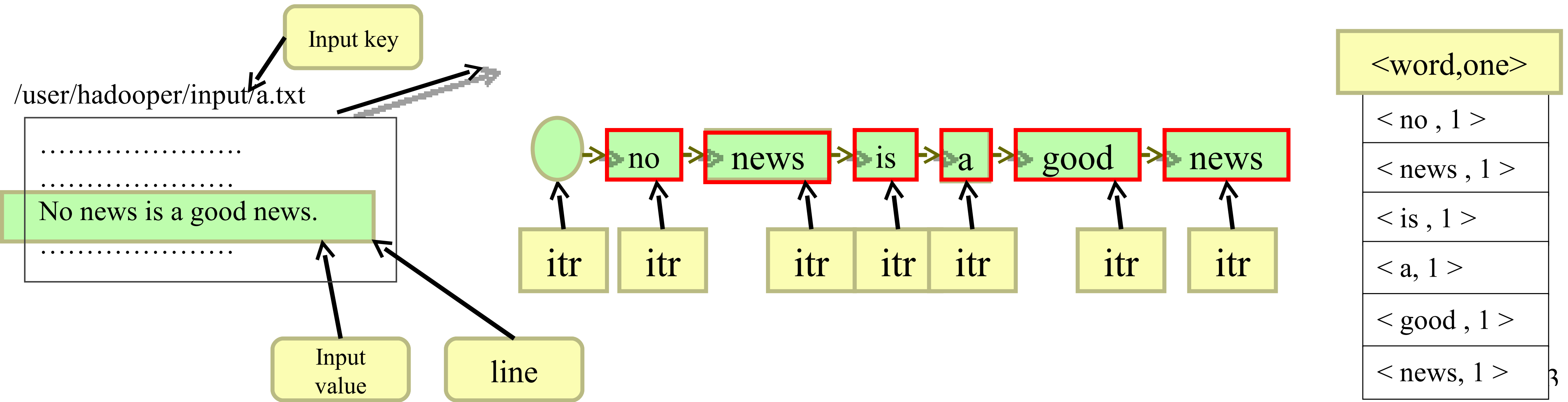
```
Class MR{
    static public Class Mapper …{
    }
    static public Class Reducer …{
    }
    main(){
        JobConf conf = new JobConf( MR.class );
        conf.setMapperClass(Mapper.class);
        conf.setReduceClass(Reducer.class);
        FileInputFormat.setInputPaths(conf, new Path(args[0]));
        FileOutputFormat.setOutputPath(conf, new Path(args[1]));
        Other configure 其他的設定參數程式碼
        JobClient.runJob(conf);
}}
```

Map 程式碼

Reduce 程式碼

Word Count - mapper

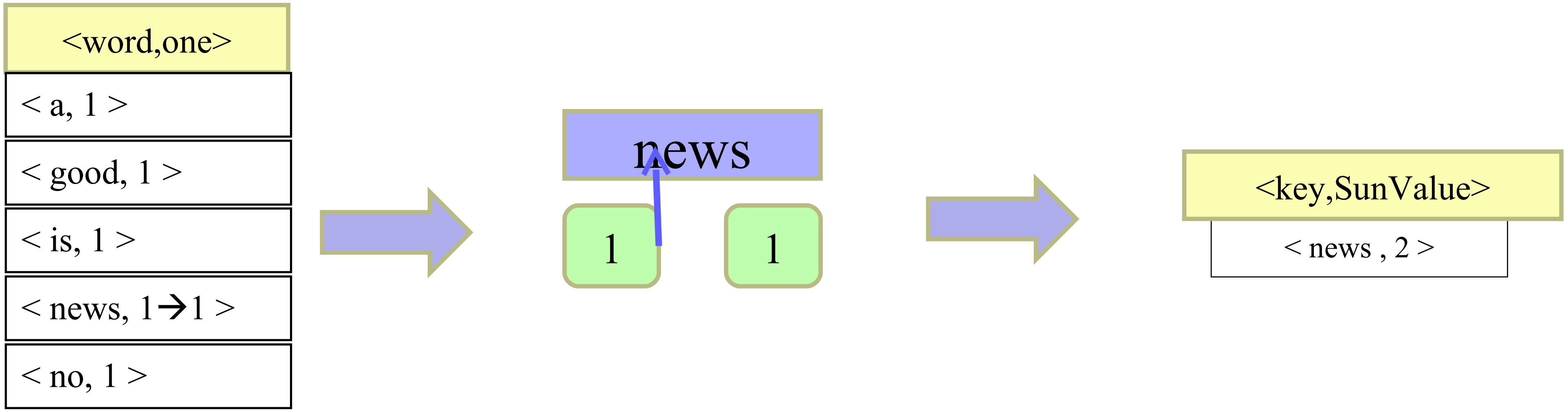
```
class MyMapper extends Mapper<LongWritable, Text, Text, IntWritable> {
        private final static IntWritable one = new IntWritable(1);
        private Text word = new Text();
        public void map( LongWritable key, Text value, Context context)
        throws IOException , InterruptedException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



Word Count - reducer

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

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



Word Count – main program

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

Questions?

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

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

Powered by DRBL

Hadoop 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^{TM}, 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^{TM} based).

Hadoop Pipes (C++, Python)

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

Hadoop Streaming

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

Running Hadoop Streaming

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

… More …

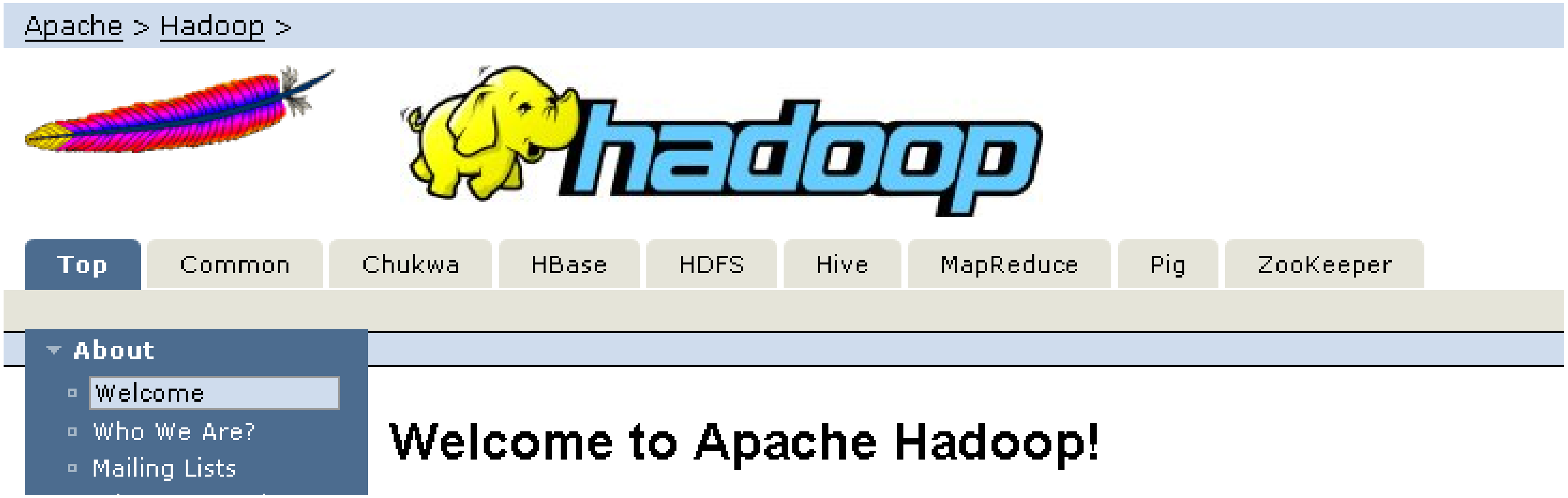
Hadoop Streaming with shell commands (1)

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

Hadoop Streaming with shell commands (2)

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

There are serveral Hadoop subprojects



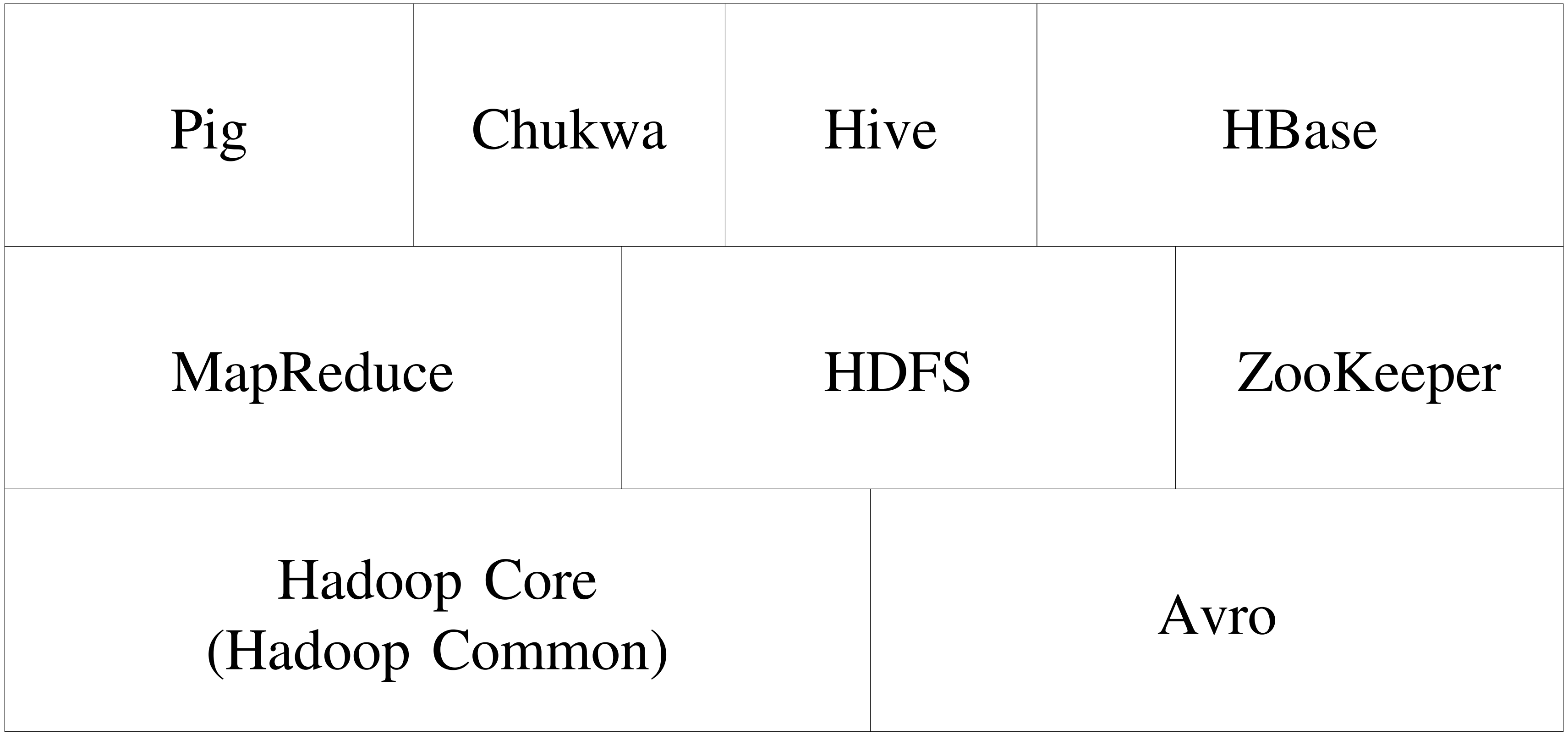
Welcome to Apache Hadoop!

- 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



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

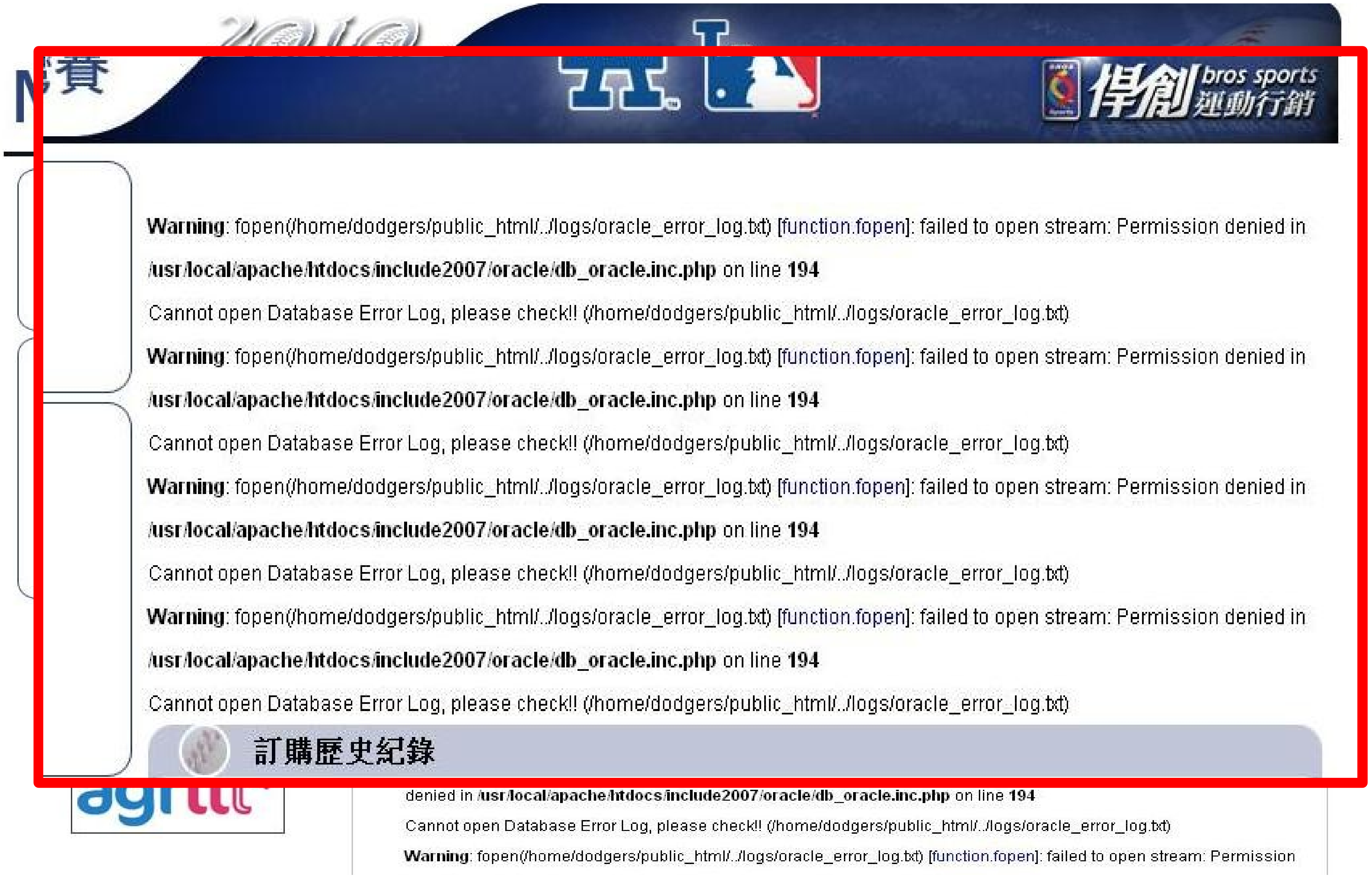
- http://mahout.apache.org/
- Mahout is a scalable machine learning libraries.
- implemented on top of Apache Hadoop using the map/reduce paradigm.
- Mahout currently has
  - Collaborative Filtering
  - User and Item based recommenders
  - K-Means, Fuzzy K-Means clustering
  - Mean Shift clustering
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

HBase 雲端資料庫

Introduction to HBase

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

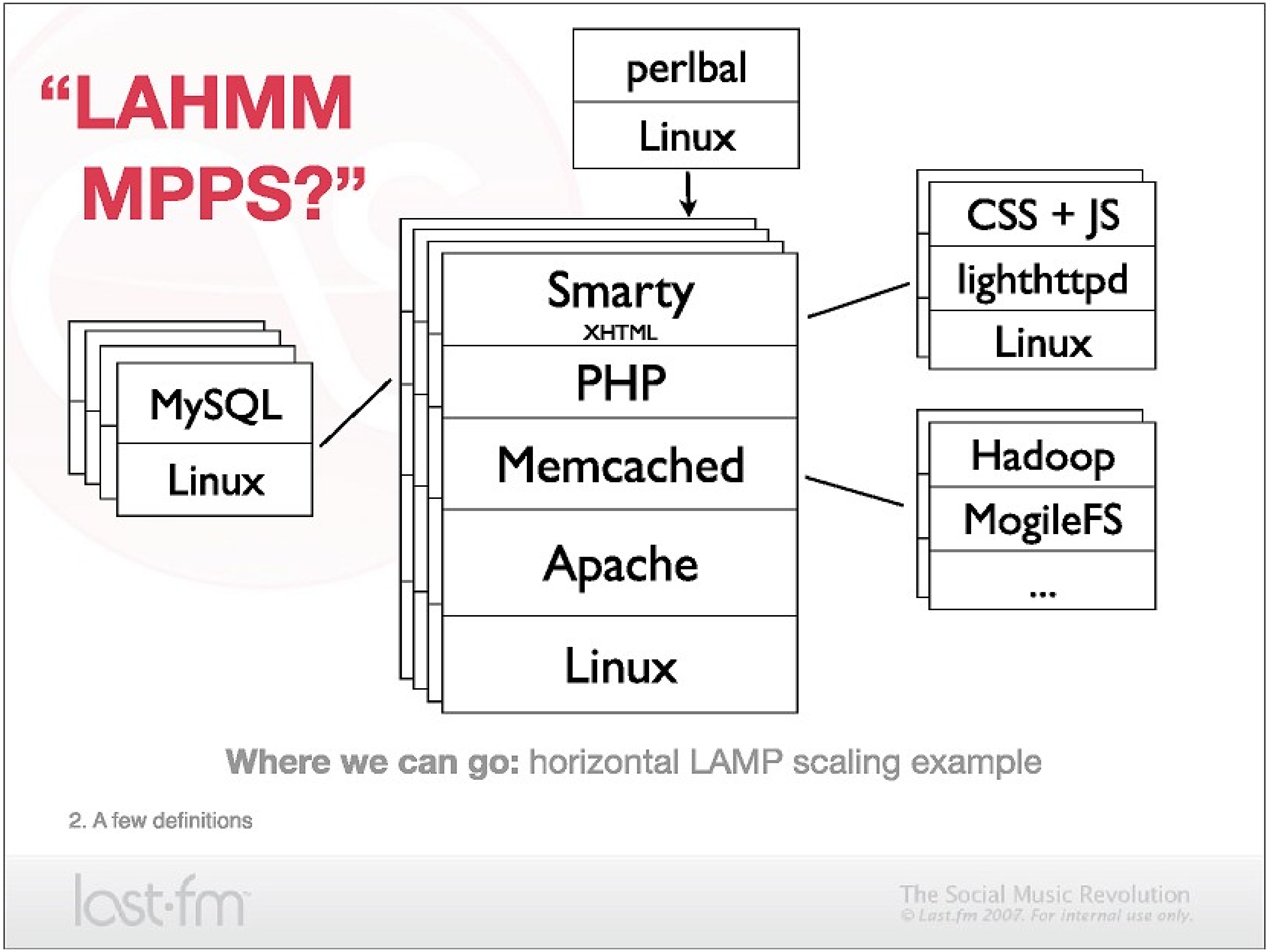
It's all about SCALE!!



```
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)
```

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

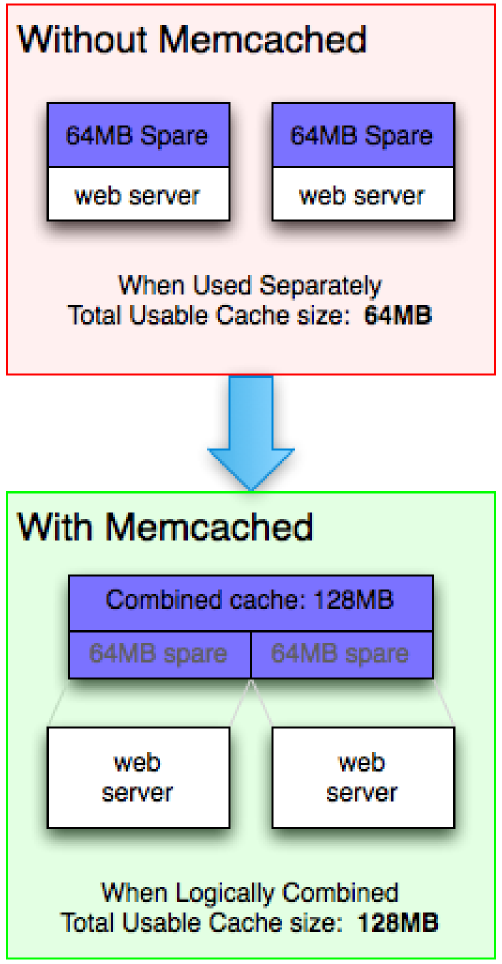
How to scale up web service in the past ?



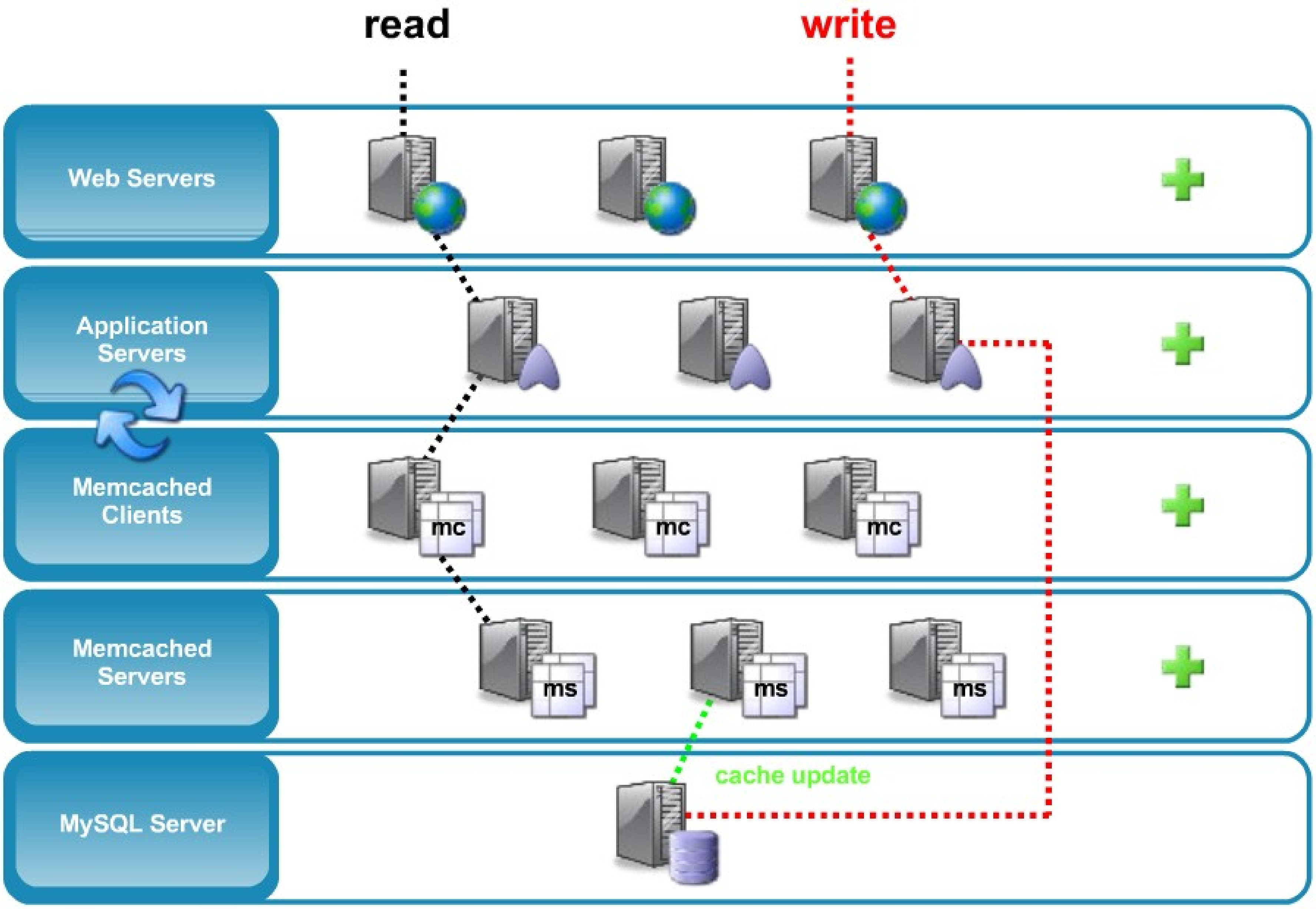
Source: http://www.slideshare.net/mallipeddi/scalable-lamp-development-for-growing-web-apps

Tools used by large scale websites

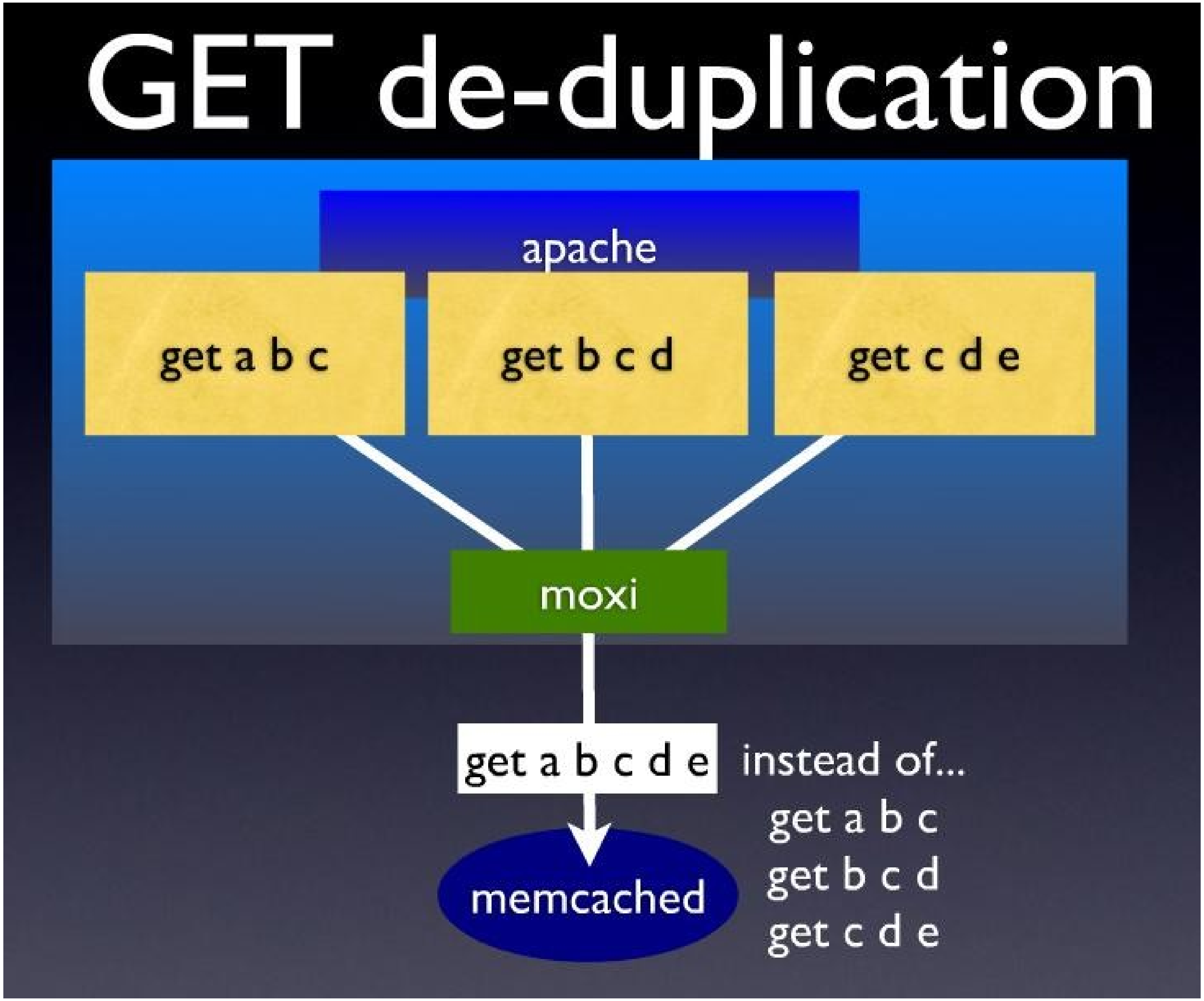
- Perlbal - http://www.danga.com/perlbal/
  - 多個網頁伺服器的負載平衡
  - Load balancer
- MogileFS - http://www.danga.com/mogilefs/
  - 分散式檔案系統
  - Distributed File System fo small files
  - 有公司認爲 MogileFS 比起 Hadoop 適合拿來處理小檔案
- memcached - http://memcached.org/
  - 共享記憶體 ??
  - Share Memory
  - 把資料庫或經常讀取的部分，用記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



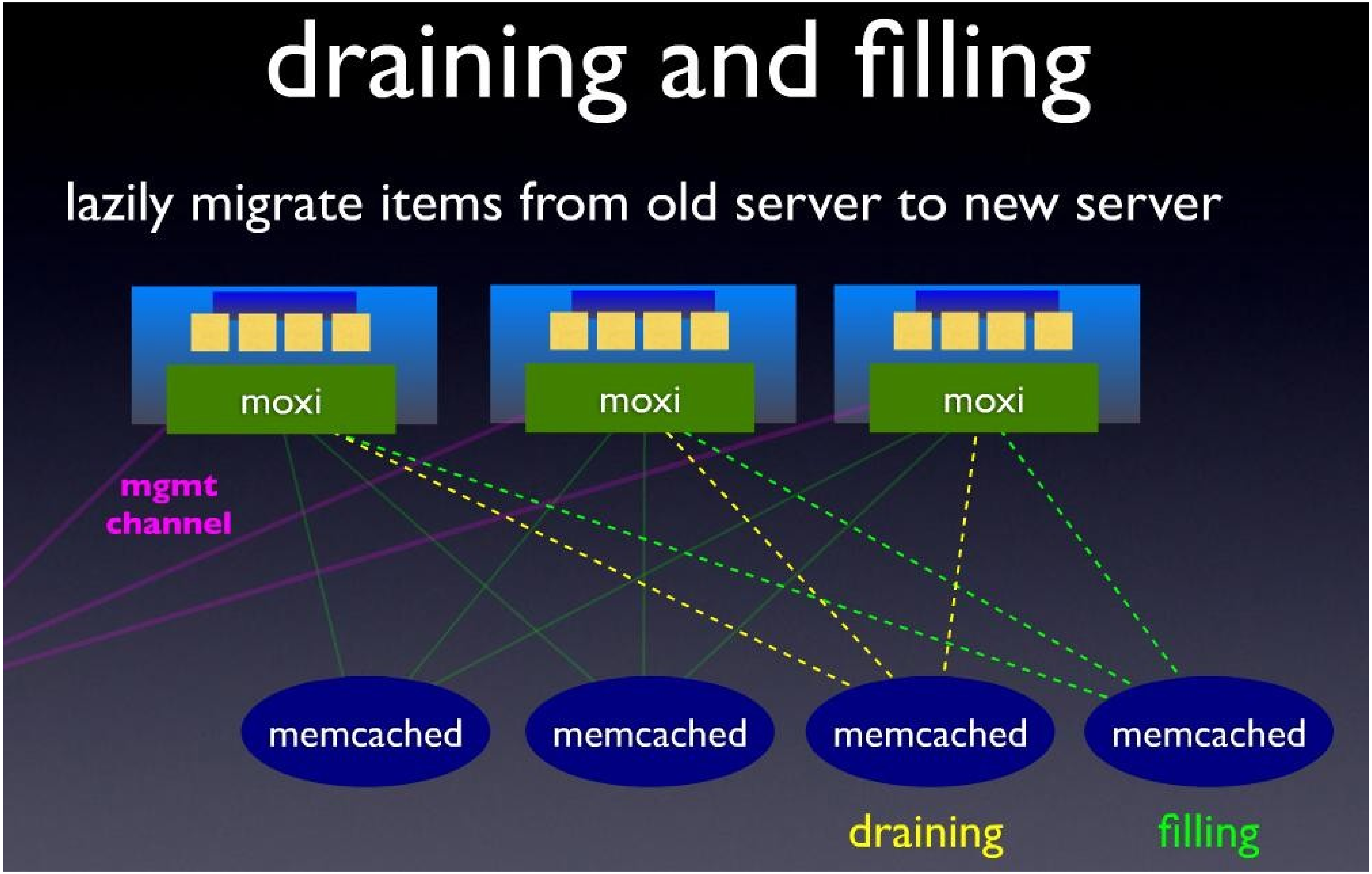
Memcached & MySQL



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



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



draining and filling

lazily migrate items from old server to new server

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

HBase is ..

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

Benefits

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

Who use HBase

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

Backdrop

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

HBase Is Not …

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

HBase Is Not ...(2)

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

Why Bigtable?

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

Why Bigtable? (2)

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

Why HBase ?

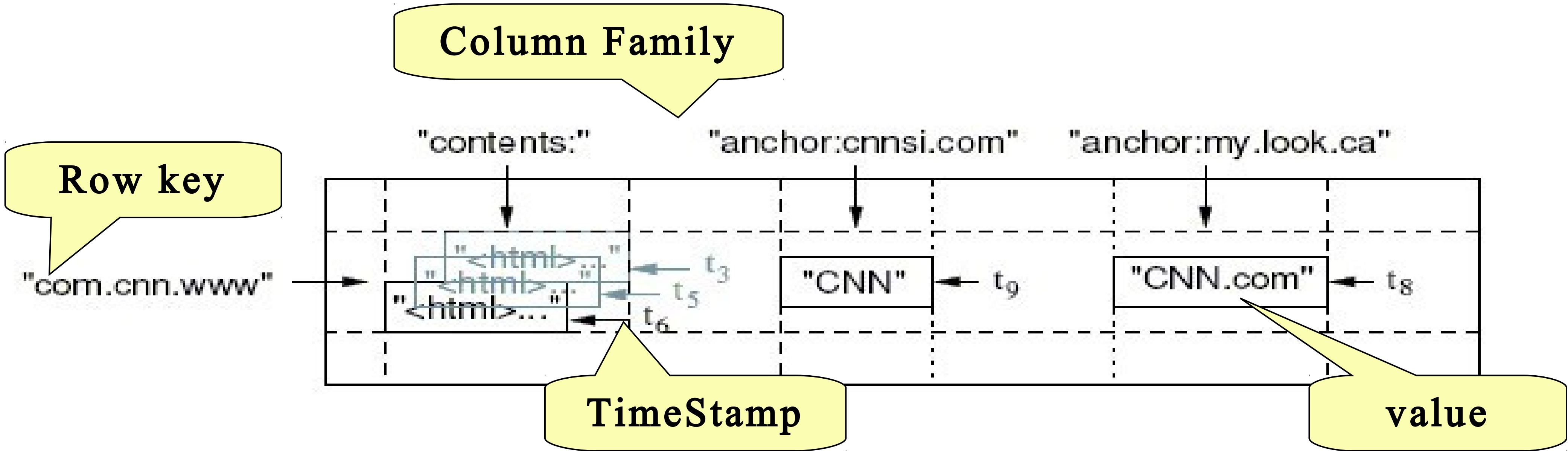
- HBase is a Bigtable clone.
- It is open source
- It has a good community and promise for the future
- It is developed on top of and has good integration for the Hadoop platform, if you are using Hadoop already.
- It has a Cascading connector.

HBase benefits than RDBMS

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

Data Model

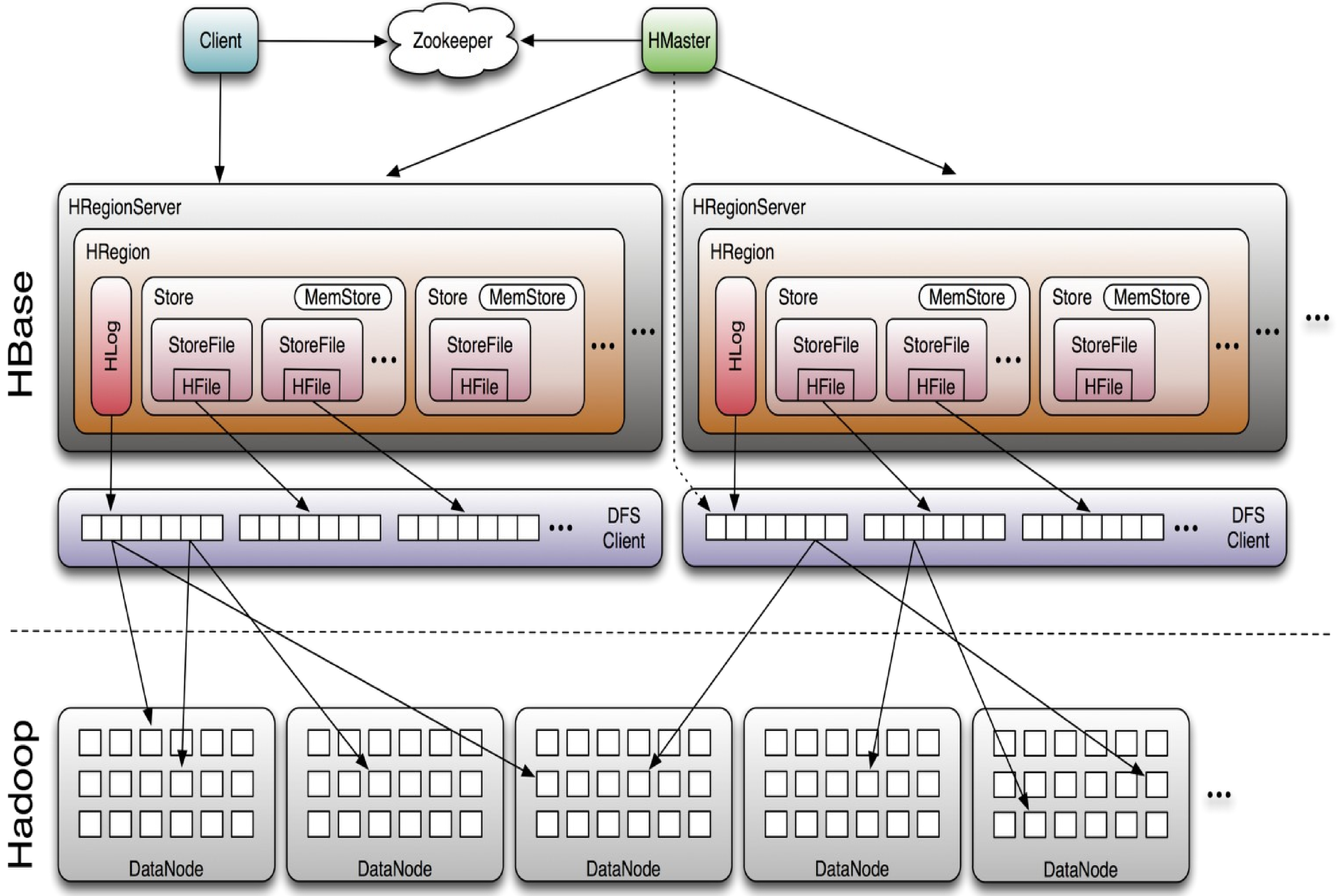
- Tables are sorted by Row
- Table schema only define it's *column families* .
  - Each family consists of any number of columns
  - Each column consists of any number of versions
  - Columns only exist when inserted, NULLs are free.
  - Columns within a family are sorted and stored together
- Everything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



Members

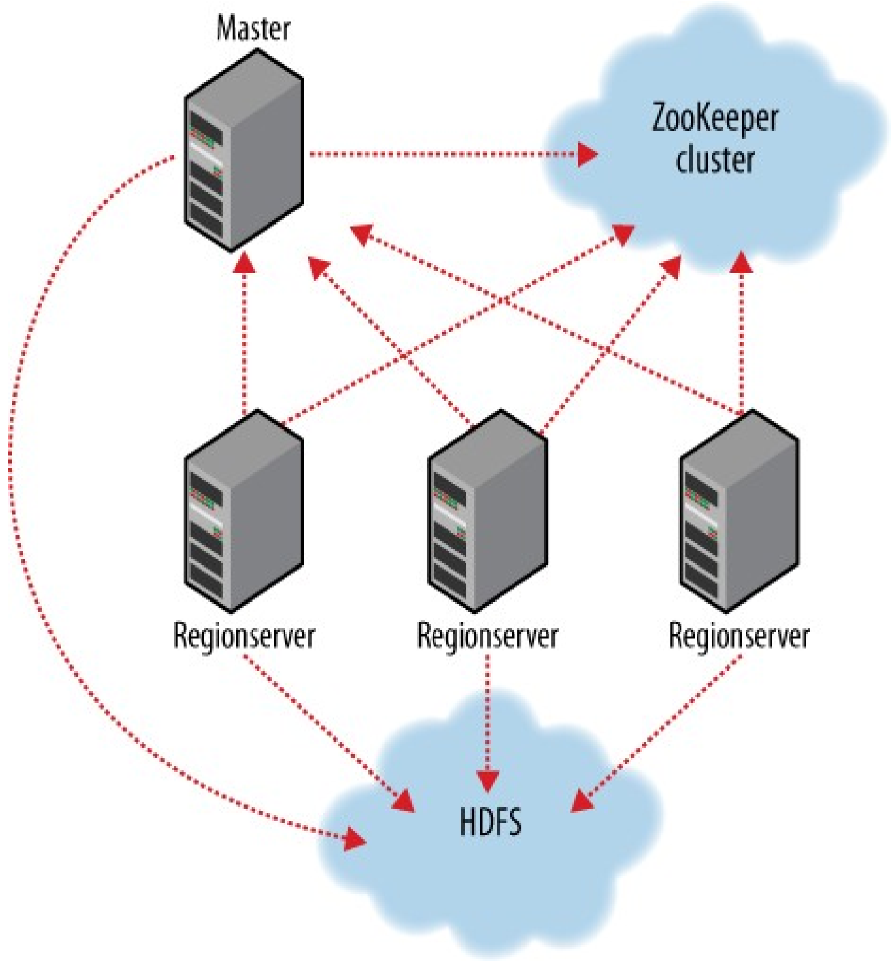
- *Master*
  - Responsible for monitoring region servers
  - Load balancing for regions
  - Redirect client to correct region servers
  - The current SPOF
- *regionserver* slaves
  - Serving requests(Write/Read/Scan) of Client
  - Send HeartBeat to Master
  - Throughput and Region numbers are scalable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

Architecture

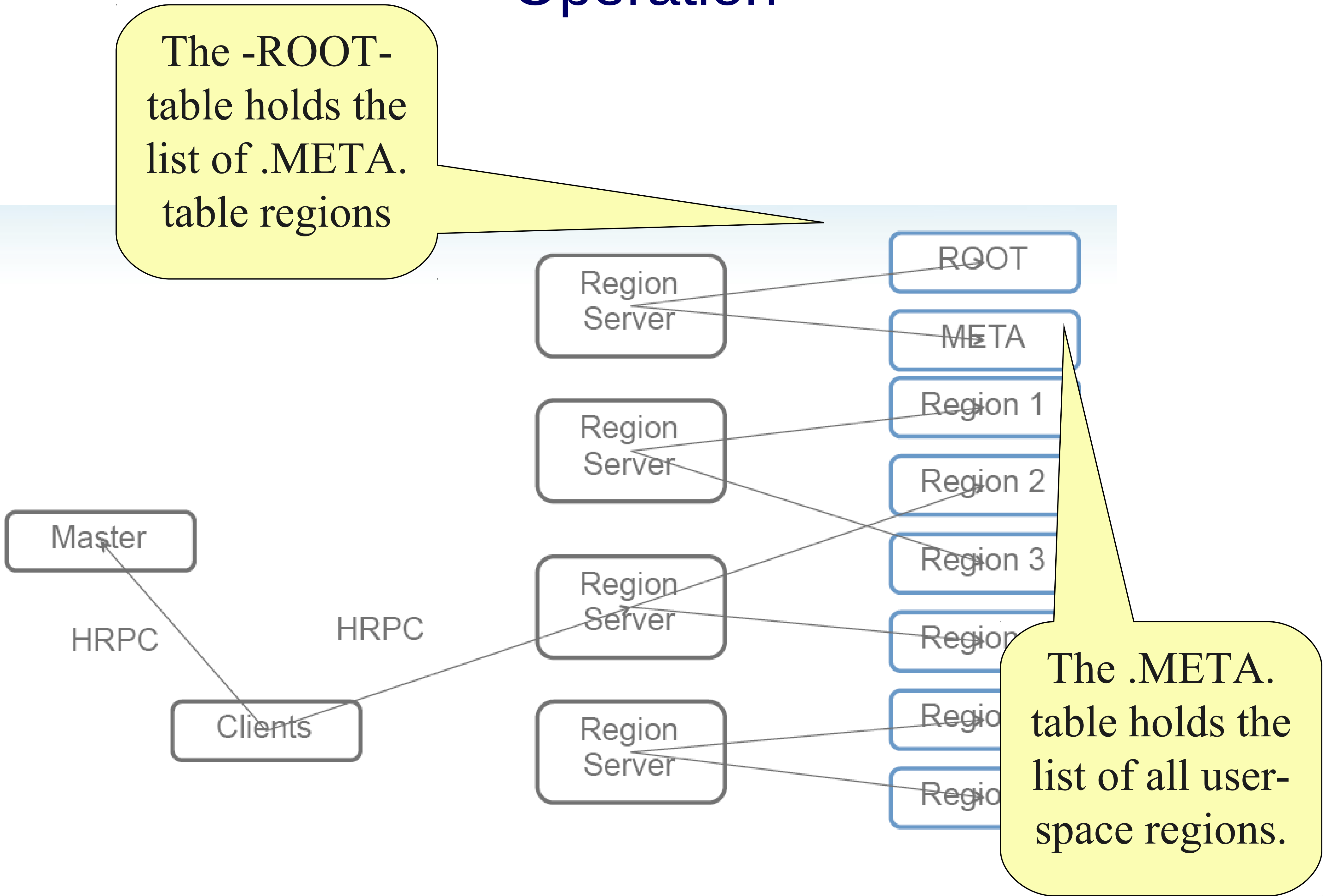


ZooKeeper

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



Operation



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

Questions?

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

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

Powered by DRBL