Hadoop 進階課程

HBase 資料庫應用

< V 0.20 >

王耀聰 陳威宇

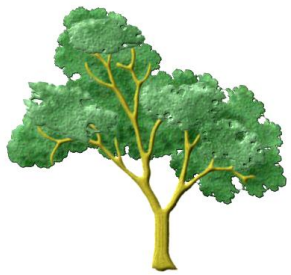
Jazz@nchc.org.tw

waue@nchc.org.tw

財團法人國家實驗研究院

國家高速網路與計算中心

NATIONAL CENTER FOR HIGH-PERFORMANCE COMPUTING



一、導論

原本我們使用關聯式資料庫好好的，為何又要有新資料庫的儲存架構型態，是有其必要？或新技術可完全取而代之？還是只是一個等待泡沫化新技術的濫觴？

海量資料緒論

- Hadoop 能運算海量資料，然後呢？
  - 其實 Hadoop運算出來的結果也不少
- 如何處理 Hadoop 運算出來的資料？
  - 再用Hadoop運算一次??
- 海量資料也需要整理
  - 排序
  - 搜尋
  - 選擇

RDBMS / 資料庫

- **Relational DataBase Management System =** 關聯式資料庫管理系統
  - **Oracle、IBM DB2、SQL Server、MySQL…**

資料庫管理系統（DBMS）（檢視 · 討論 · 編輯 · 歷史）

概念

資料庫 · 資料庫模型 · 資料庫儲存結構 · 關聯 (資料庫) · 關聯模型 · 分布式資料庫 · ACID · Null值
關聯式資料庫 · 關聯代數 · 關聯演算 · 元組關聯演算 · 域關聯演算 · 資料庫正規化 · 參照完整性 · 關聯式資料庫管理系統
主鍵 · 外來鍵 · 代理鍵 · 超鍵 · 候選鍵

資料庫元件

觸發器 · 檢視 · 資料庫表 · 指標 (資料庫) · 事務日誌 · 資料庫事務 · 資料庫索引
儲存程式 · 資料庫分割

SQL

分類： 資料查詢語言DQL · 資料定義語言DDL · 資料操縱語言DML · 資料控制語言DCL
指令：SELECT · INSERT · UPDATE · MERGE · DELETE · JOIN · UNION · CREATE · DROP · Begin work · COMMIT · ROLLBACK · TRUNCATE · ALTER
安全： SQL資料隱碼攻擊 · 參數化查詢

資料庫管理系統的實施

實施型式

關聯式資料庫 · 檔案型資料庫 · Deductive · 維度化資料庫 · 階層式 · 物件資料庫 · 物件關聯式資料庫 · Temporal · XML資料庫

資料庫產品

物件型 (對比) · 關聯型 (對比)

資料庫成分

查詢語言 · 查詢最佳化器 · 查詢計畫 · 嵌入式SQL · ODBC · JDBC · OLE DB

RDBMS 碰上大資料

- RDBMS 的好處
  - 提供了很多而且很豐富的操作方式
  - SQL語法普遍被使用
- 但當資料量愈來愈大時，會遇到單台機器的”囧”境
  - 網路頻寬有限
  - 空間有限
- 走向多台機器架構

跨足多台機器的 RDBMS

- 讀取的 query 比寫入的 query 多
  - **Replication**
- slave 過多時，造成每台記憶體內重複 cache 相同元素
  - **Memcached**
- 寫入的 query 超過單台可以負荷的量時，replication 技術則導致每台Slave 一起掛
  - **Sharding**
  - 依照 id，把資料拆散到各台 （如 Flickr）

多台機器的 RDBMS的缺點

- 需要 application server 或是 library配合，否則第三方程式找不到資料放在哪個node
- 無法隨意使用 JOIN 及 transaction，即使可以硬要使用效能也很差
- 設計 schema 時必須注意，當一個 cluster 愈來愈大時要 rebalance

是否非RDBMS不可？

- Web 2.0 網站很多時候
  - 不需要transaction
  - 減少JOIN 次數
  - 多次 SELECT 拉資料
- 一開始寫在一台DB主機的SQL程式無法再套用於後來多台SQL主機的架構上
  - 程式有可能全部重寫

將RDBMS簡化吧

- RDBMS -> key-value DataBase
  - 簡化掉不需要的功能，到只剩下key-value的架構
    - GET(key)
    - SET(key, value)
    - DELETE(key)
- 類似 Excel

Distributed key-value System

- key-value DataBase -> Distributed key-value DataBase
- 加強 key-value 架構的 scalability，使得增加機器就可以增加容量與頻寬
- 適合管理大量分散於不同主機的資料
- 通稱為 NoSQL DataBase

常見的 NoSQL

OpenSource

- HBase (Yahoo!)
- Cassandra (Facebook)
- MongoDB
- CouchDB (IBM)
- SimpleDB (Amazon)

Commercial

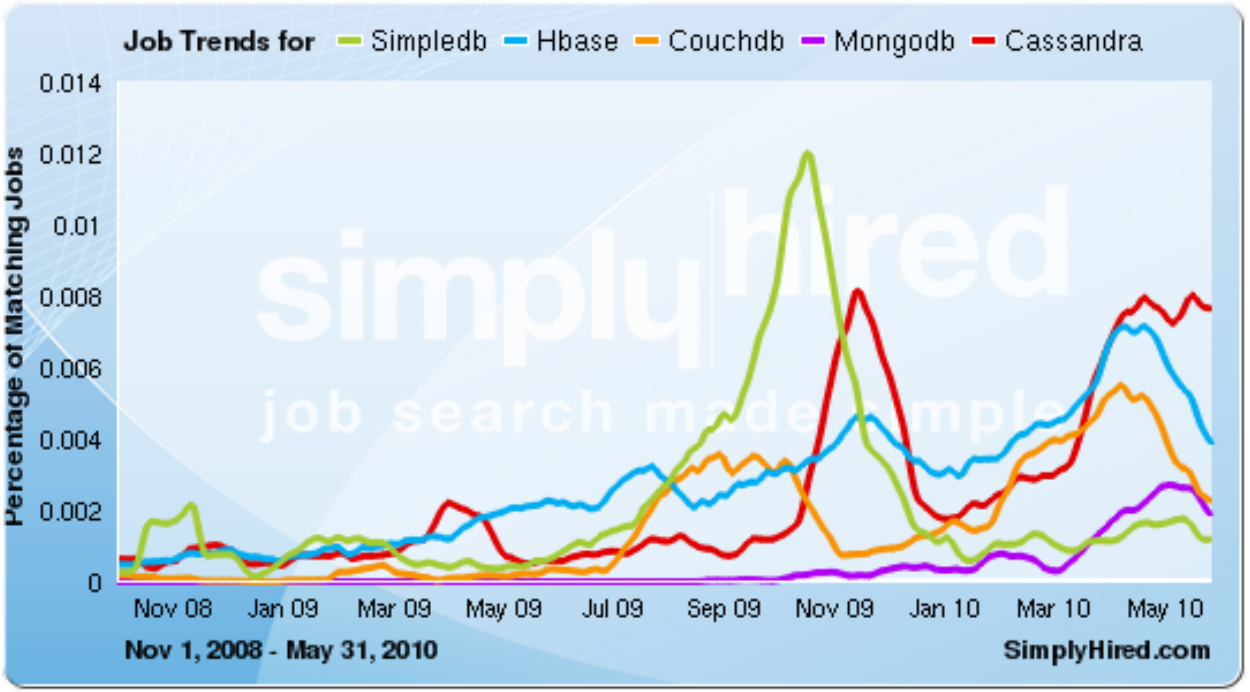
- BigTable (Google)

2010 年 NoSQL 職缺排行榜

1. Cassandra
2. HBase
3. CouchDB
4. MongoDB
5. SimpleDB

(2010-07-25)

Simpledb, Hbase, Couchdb, Mongodb, Cassandra Trends



Simpledb, Hbase, Couchdb, Mongodb, Cassandra 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:

- Simpledb jobs increased 357%
- Hbase jobs increased 745%
- Couchdb jobs did not change or there is no data available
- Mongodb jobs increased 18,480%
- Cassandra jobs did not change or there is no data available

一、HBase 介紹

介紹HBase如何而來，它的 Why, What, How ….，以及它的架構

HBase, *Hadoop database*, is an open-source, distributed, versioned, column-oriented store modeled after Google' Bigtable. Use it when you need random, realtime read/write access to your Big Data.

BigTable ?

- Bigtable: 一個結構化數據的分佈式存儲系統
- Google Style的數據庫，使用結構化的文件來存儲數據
- 不支持關聯或是類似於SQL的高級查詢。
- 大規模處理、高容錯性
- PB級的存儲能力
- 每秒數百萬的讀寫操作

HBase

- 設計概念與結構類似Bigtable
- HBase 以 Hadoop 分散式檔案系統 (HDFS) 為基礎，提供類Bigtable 功能
- HBase 是具有以下特點的儲存系統：
  - 類似表格的資料結構 (Multi-Dimensional Map)
  - 分散式
  - 高可用性、高效能
  - 很容易擴充容量及效能
- HBase 適用於利用數以千計的一般伺服器上，來儲存Petabytes級的資料。
- HBase同時提供Hadoop MapReduce程式設計。

開發歷程

- 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
- 2010.3
  - HBase graduates from Hadoop sub-project to Apache Top Level Project
- 2010.7
  - HBase 0.20.6 released

誰使用 HBase

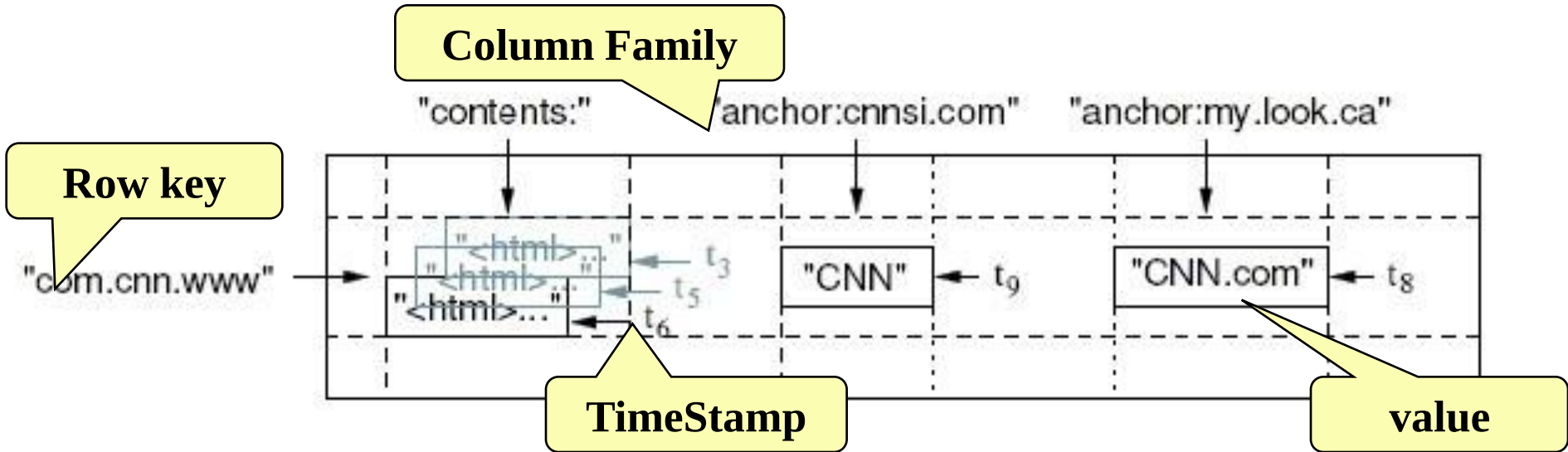
- Adobe
  - 內部使用 (Structure data)
- Kalooga
  - 圖片搜尋引擎 http://www.kalooga.com/
- Meetup
  - 社群聚會網站 http://www.meetup.com/
- Streamy
  - 成功從 MySQL 移轉到 Hbase http://www.streamy.com/
- Trend Micro
  - 雲端掃毒架構 http://trendmicro.com/
- Yahoo!
  - 儲存文件 fingerprint 避免重複 http://www.yahoo.com/
- More
  - http://wiki.apache.org/hadoop/Hbase/PoweredBy

為什麼使用HBase?

- 不是關聯式(Relational)資料庫系統
  - 表格(Table)只有一個主要索引 (primary index) 即 row key.
  - 不提供 join
  - 不提供 SQL語法。
- 提供Java函式庫, 與 REST與Thrift等介面。
- 提供 getRow(), Scan() 存取資料。
  - getRow()可以取得一筆row range的資料，同時也可以指定版本 (timestamp)。
- Scan()可以取得整個表格的資料或是一組row range (設定start key, end key)
- 有限的單元性(Atomicity)與交易 (transaction)功能.
- 只有一種資料型態 (bytes)
- 可以配合MapReduce框架，進行複雜的分析與查詢

Data Model

- Table依 *row key* 來自動排序
- Table schema 只要定義 *column families* .
  - 每個column family 可有無限數量的 columns
  - 每個column的值可有無限數量的時間版本(timestamp)
  - Column可以動態新增，每個row可有不同數量的columns。
  - 同一個column family的columns會群聚在一個實體儲存單元上，且依 column 的名稱排序。
  - byte[] 是唯一的資料型態(Row, Family: Column, Timestamp) Value



Data Model

- HBase實際上儲存Table時，是以column family為單位來存放

Row Key	Time Stamp	Column (Family) "content:"
com.cnn.www	t9	"<html>…"
	t6	"<html>…"

Row Key	Time Stamp	Column (Family) "anchor:"	
com.cnn.www	t9	"anchor:cnnsi.com"	"CNN"
	t8	"anchor:cnnsi.com"	"CNN"
		"anchor:my.loc	"MyLook"

HTable 成員

Table, Family, Column, Qualifier , Row, TimeStamp

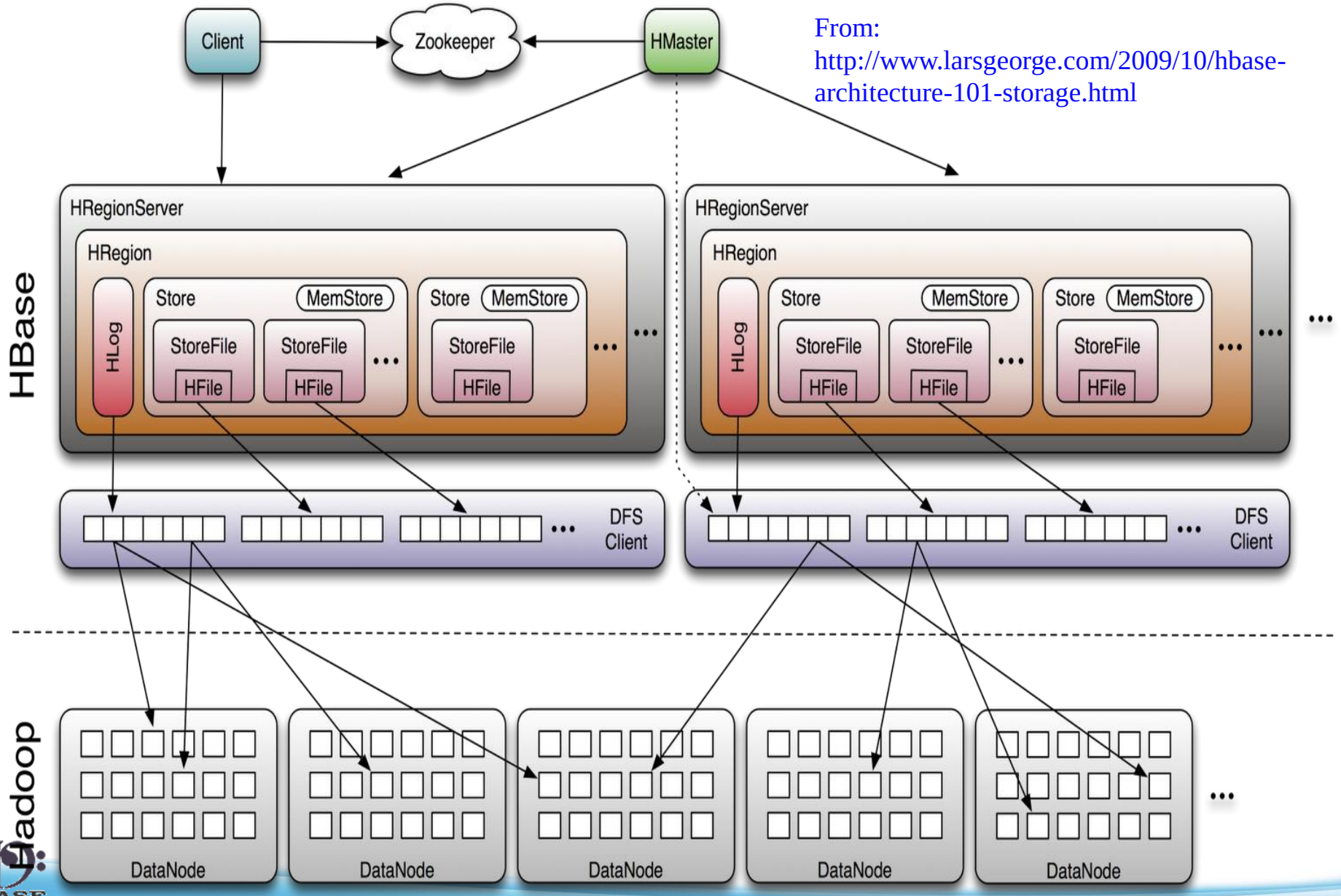
		Contents	Department		
			news	bid	sport
t1	com.yahoo.news.tw	“撿到學雜費，硬要分三成”	“tech”		
t2		“科研論文評比 5校進500大”	“tech”		
t3		“罰蹲立300下！班長「住院」 師懊悔”	“tech”		
t1	com.yahoo.bid.tw	“… iphone 4G 9/17 日上市”		“ 3C ”	
t1	com.yahoo.sport.tw	“Nadal 大滿貫”			“MBA”

Regions

- 表格是由一或多個 region 所構成
  - Region 是由其 startKey 與 endKey 所指定
- 每個 region 可能會存在於多個不同節點上，而且是由數個HDFS 檔案與區塊所構成，這類 region 是由 Hadoop 負責複製

Region	Row Keys	Column Family "Content"
Region 1	00000	…
	00001	…
	…	…
	09999	…
Region 2	10000	…
	…	…
	…	…
	29999	…

HBase 與 Hadoop 搭配的架構

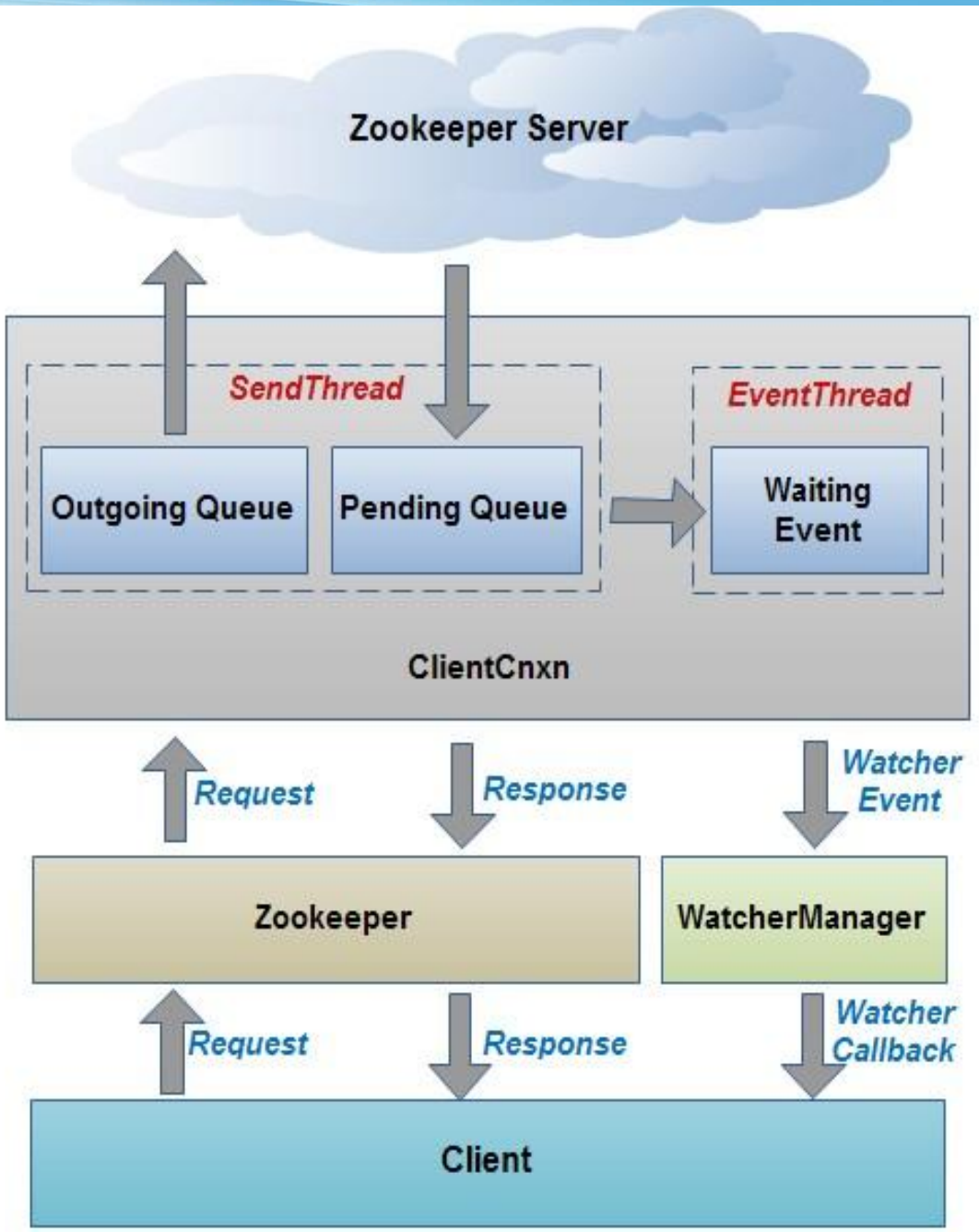


HBase 0.20 特色

- 解決單點失效問題（single point of failure）
  - Ex: Hadoop NameNode failure
- 設定檔改變或小版本更新會重新啟動
- 隨機讀寫（Random access）效能如同MySQL

Zookeeper ?

- Hadoop的正式子項目
- 針對大型分散式系統的可靠協調系統
- Google的Chubby
- 存儲一些配置信息,確保文件寫入的一致性等
- Master / Client 架構，Master 可由選舉而得



From: http://www.spnguru.com/?p=211