Hbase介紹

資料模型(Data Model) 與系統架構 (Architecture)

趨勢科技研發實驗室

Public 2009/5/13

課程大綱

- HBase 簡介
- HBase 資料模型
- HBase 系統架構
- 參考資料

Introduction to HBase

- 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 以 Hadoop 分散式檔案系統 (HDFS) 為基礎， 提供類 Bigtable 功能
- Hbase同時提供Hadoop MapReduce程式設計。

HBase 發展史

- 2006 年 11 月
  - Google 發佈 BigTable 相關論文
- 2007 年 2 月
  - 建立初始的 HBase 原型，用來做為 Hadoop 的構成要素
- 2007 年 10 月
  - 第一個可使用的 HBase
- 2008 年 1 月
  - Hadoop 成為 Apache 頂層專案，而 HBase 則成為子專案
- 2008 年10 月
  - HBase 0.18 版與 0.19 版發佈

HBase並不是 …

- 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(Aatomicity)與交易 (transaction)功能.
  - 只有一種資料型態 (bytes)

為什麼使用Bigtable?

- 關聯式資料庫(Relational Database) 適用做資料異動 C.R.U.D (create, retrieval, update, delete) 等操作，主要因為這動作在記憶體中進行。對於超大量的資料分析，資料分散在多個節點的情況下，關聯式資料庫系統就不適用了。現有可於大量資料的分析的聯式資料庫軟體價格昂貴，且仍只適用在特定用途。

- 這裏所謂的大量資料分析是指
  - Big queries – 整個資料表的存取
  - Big databases - 100 Terabytes以上的資料

- Bigtable的可以配合MapReduce框架，進行複雜的分析與查詢。

為什麼使用HBase?

- HBase實作Bigtable的概念。
- 它是開放原始碼
- 架構在Hadoop分散式儲存系統(HDFS)上。
- HBase是Apache的專案之一，未來的支援及維護上較有保障。

邏輯資料模型 (Logical 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

Row Key	Time Stamp	Column (Family) "content:"	Column (Family) "anchor:"	
com.cnn.www	t9	"<html>…"	"anchor:cnnsi.com"	"CNN"
	t8		"anchor:cnnsi.com" "anchor:my.lock.ca"	"CNN" "MyLook"
	t6	"<html>…"		

實體資料模型 (Physical Data Model)

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

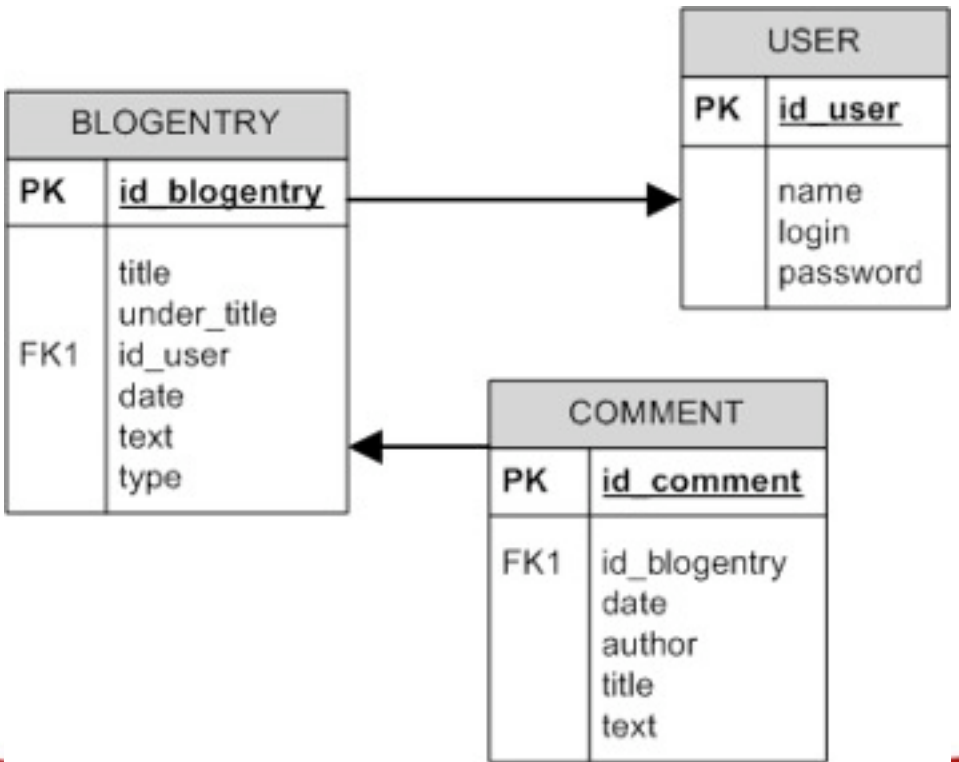
Row Key	Time Stamp	Column (Family) "content:"
com.cnn.www	t9	"<html>…"
	t6	"<html>…"

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

實際個案討論 – 部落格

- 邏輯資料模型
  - 一篇 Blog entry 由 title, date, author, type, text 欄位所組成。
  - 一位User由 username, password等欄位所組成。
  - 每一篇的 Blog entry可有許多Comments。每一則comment由 title, author, 與 text 組成。

- ERD



部落格 – HBase Table Schema

Table	Row Key	Family	Attributs
blogtable	TTYYYYMMDDHHmmss	info:	Always contains the column keys author,title,under_title. Should be IN-MEMORY and have a 1 version
		text:	No column key. 3 versions
		comment_title:	Column keys are written like YYYMMDDHHmmss. Should be IN-MEMORY and have a 1 version
		comment_author:	Same keys. 1 version
		comment_text:	Same keys. 1 version
usertable	login_name	info:	Always contains the column keys password and name. 1 version

- Row key
  - type (以2個字元的縮寫代表)與 timestamp組合而成。
  - 因此 rows 會先後依 type 及 timestamp 排序好。方便用 scan () 來存取 Table的資料。
- BLOGENTRY 與 COMMENT的”一對多”關係由comment_title, comment_author, comment_text 等column families 內的動態數量的column來表示。 每個Column的名稱是由每則 comment的 timestamp來表示，因此每個column family的 column 會依時間自動排序好

HBase 架構

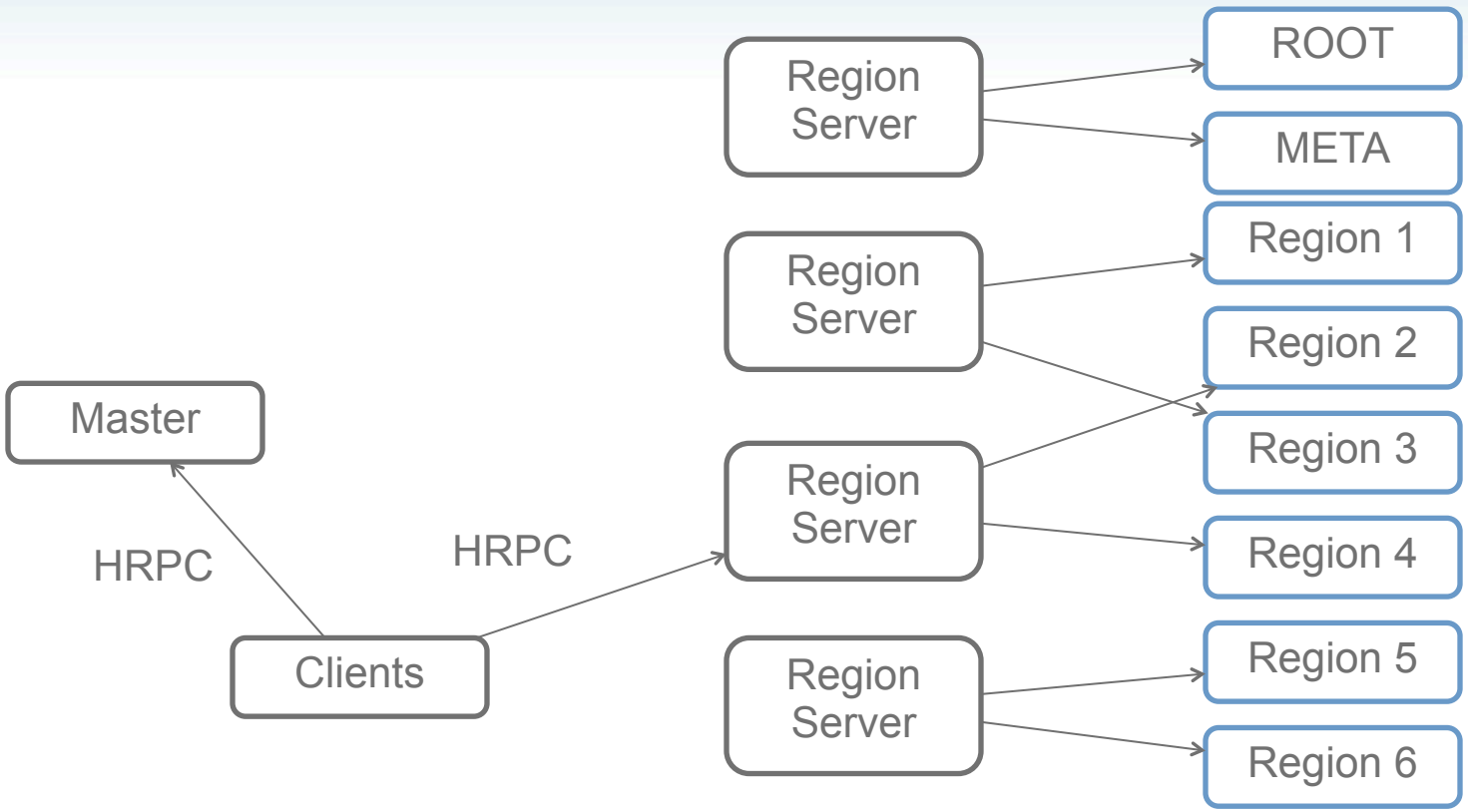
Region	Row Keys	Column Family “Content:”
Region 1	00000	…
	00001	…
	…	…
	09999	…
Region 2	10000	…
	…	…
	…	…
	29999	…

- 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。

HBase 架構

- Region Servers
  - 負責處理使用者的request (write/read/scan)
  - 定時送 heartbeat 給 master
  - 增加 region servers將可增加整體的throughput
- HBase Master
  - 負責管理region servers
  - 適度分配regions給region servers
  - 負責處理使用者的查詢，並提供資料所在的region server資訊。
  - 目前master為single point of failure

HBase 架構



HBase的Client Interface

- Java client
  - *get(byte [] row, byte [] column, long timestamp, int versions);*
- Non-Java clients
  - Thrift server
    - Sample ruby, c++, & java (via thrift) clients
  - REST server
- TableInput/OutputFormat for MapReduce
- HBase Shell
  - *./bin/hbase shell YOUR_SCRIPT*

參考資料

- Google BigTable 論文
  - http://labs.google.com/papers/bigtable.html
- HBase 官方網站
  - http://hadoop.apache.org/hbase/
- HBase 官方 wiki
  - http://wiki.apache.org/hadoop/Hbase