**ASUS - 12月工作筆記**

內容

[置頂其他注意事項 1](#_Toc440471707)

[其他搜尋的資料 1](#_Toc440471708)

[前端技術待學習技術思考？ 2](#_Toc440471709)

[week1 - 1/4(一)~1/10(日) 2](#_Toc440471710)

[IIS log - 資料檔案相關 (2015資料重新統整) 2](#_Toc440471711)

[IIS log - 相關原理筆記 (2015資料重新統整) 3](#_Toc440471712)

[IIS log time\_taken趨勢觀察 (2015資料重新統整) 3](#_Toc440471713)

[IIS log與ASUS log分析統整 - 2016/1/6 3](#_Toc440471714)

[資訊統整 - 2016/1/8 4](#_Toc440471715)

[week2 - 1/11(一)~1/17(日) 4](#_Toc440471716)

[上週五課會John交代事項： 4](#_Toc440471717)

[本週工作項目 5](#_Toc440471718)

[IIS log與Speedera log分析統整 5](#_Toc440471719)

[GitHub學習 5](#_Toc440471720)

※※※※※※※※※※※※※※※※※※※※ 分隔線 ※※※※※※※※※※※※※※※※※※※※

# 置頂其他注意事項

最多僅用三層目錄

目錄命名第一階層：為 "周次 - 日期"

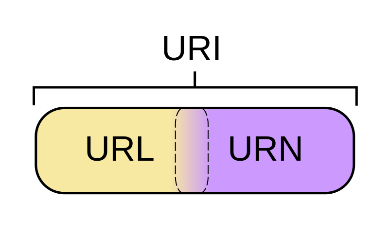
目錄命名第二階層：為 "專案名稱 - 標題內容"

* 一些專案的欄位定義及統計使用excel來記錄，並放置於專案的資料夾。
* 把所有專案流程文件打在一個word檔會比分散在各project資料夾好，因為未來要找資料會比較容易整合，且可用日期及內容來設立目錄，方便查詢及整合。

# 其他搜尋的資料

括弧(Bracket)的種類 - <https://en.wikipedia.org/wiki/Bracket>

統一資源標誌符 URI與URL的區別！- [統一資源識別元（Uniform Resource Identifier，或URI)](https://zh.wikipedia.org/wiki/%E7%BB%9F%E4%B8%80%E8%B5%84%E6%BA%90%E6%A0%87%E5%BF%97%E7%AC%A6)



* 用於標識唯一書目的ISBN系統是一個典型的URN使用範例。
* 在類Unix作業系統中，一個典型的URL位址可能是一個檔案目錄。
* URL是一種URI，它標識一個網際網路資源，並指定對其進行操作或取得該資源的方法。可能通過對主要存取手段的描述，也可能通過網路「位置」進行標識。

# 前端技術待學習技術思考？

1. HTML5 + CSS3
2. JavaScritp + Jquery
3. JSP + SQL

# week1 - 1/4(一)~1/10(日)

## IIS log - 資料檔案相關 (2015資料重新統整)

2015/12/24晚上全部下載回來解壓縮，解一個晚上。

在使用R將檔案轉換為RDS的過程中發現，每個檔案欄位數量不同，並不能直接放在一起變成一個Impala Table，而且每個壓縮檔裡面的log檔檔案名稱都一樣，放同一個HDFS資料夾不知道會不會被覆蓋掉！

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 數 | 資料夾名稱 | 欄 | RDS轉檔 | 轉入impala |
| 1 | TP-COAP-10-2015-11.zip | 18 | 已轉為RDS (欄位未命名) | tmp\_iis\_log\_co2 |
| 2 | TP-COAP-11-2015-11.zip | 18 | 已轉為RDS (欄位未命名) | tmp\_iis\_log\_co2 |
| 3 | TP-COAP-V01-201511.zip | 14 | 已轉為RDS | tmp\_iis\_log\_co |
| 4 | TP-COAP-V02-2015-11.zip | 14 | 已轉為RDS | tmp\_iis\_log\_co |
| 5 | TP-COAP-V03-2015-11.zip | 14 | 已轉為RDS | tmp\_iis\_log\_co |
| 6 | TP-COAP-V04-2015-11.zip | 14 | 已轉為RDS | tmp\_iis\_log\_co |
| 7 | TP-COAP-V05-2015-11.zip | 14 | 已轉為RDS | tmp\_iis\_log\_co |
| 8 | TP-COAP-V06-2015-11.zip | 14 | 已轉為RDS | tmp\_iis\_log\_co |
| 9 | TP-COAP-V07-2015-11.zip | 14 | 已轉為RDS | tmp\_iis\_log\_co |
| 10 | TP-COAP-V08-2015-11.zip | 14 | 已轉為RDS | tmp\_iis\_log\_co |
| 11 | TP-MKTSPAP-V01-201511-2.zip | 15 | 放到145去轉檔 | tmp\_iis\_log\_co3 |
| 12 | TP-MKTSPAP-V01-2015-11.zip | 15 | 11個檔案分成4-7來做RDS | tmp\_iis\_log\_co3 |
| 13 | TP-MKTSPAP-V02-2015-11.zip | 15 | 30個檔案分成10-10-10來做RDS | tmp\_iis\_log\_co3 |
| 14 | TP-SPWACAP-01-2015-11.zip | 14 | 已轉為RDS | tmp\_iis\_log\_co |
| 15 | TP-SPWACAP-02-2015-11.zip | 14 | 已轉為RDS | tmp\_iis\_log\_co |
| 16 | TP-SPWACAP-03-2015-11.zip | 14 | 已轉為RDS | tmp\_iis\_log\_co |
| 17 | TP-COMEDIA-V01-2015-11.zip | 14 | 已轉為RDS | tmp\_iis\_log\_co |
| 18 | TP-COMEDIA-V02-2015-11.zip | 14 | 已轉為RDS | tmp\_iis\_log\_co |

目前僅拿出藍色部份八個SERVER來進行分析

## IIS log - 相關原理筆記 (2015資料重新統整)

網際網路資訊服務（英语：Internet Information Services，簡稱IIS），在中國大陸被稱為互聯網信息服務，是由微軟公司提供的基於運行Microsoft Windows的網際網路基本服務。



12月日誌裡面有紀錄現在存放的檔案路徑（在http://192.168.88.116/）

目前在釐清官網關係圖，其存放位置於20151225 - IIS\_log分析資料夾：<官網架構圖.png>

↑John 2015/12/30寄給我

## IIS log time\_taken趨勢觀察 (2015資料重新統整)

可以與之前的ASUS Website log所繪製出的time\_taken趨勢圖一同進行觀察

目前將時間鎖定在11/26至11/30，圖形存放於：

1. [..\20151224 - Asus\_Website\_Log 分析\wedbsite\_log\_statstic.xlsx](../20151224%20-%20Asus_Website_Log%20分析/wedbsite_log_statstic.xlsx)
2. <IIS_log_statstic.xlsx>

## IIS log與ASUS log分析統整 - 2016/1/6

用Impala + R 去算ASUS Website log各國家不同流量的情況

經IIS\_log找出的一些異常花費時間的uri，可以看出幾個重點國家可以抓出來看看流量趨勢：

如us、sg(Singapore)、fr(France)

所給的資料目前可以統計繪圖，不同國家、不同時段的流量趨勢。

限制

1. 一些資料以及趨勢要先計算出來才有辦法找出有沒有什麼因果關係，並依此為根據來繼續探索下去，當不知道正確的異常點為何的時候，只能不斷的猜測與估計，累積經驗。
2. IIS\_log目前沒有sc-bytes的欄位，因此無法計算檔案下載了多少，花了多少錢。
3. IIS\_log目前沒有x-custom的欄位，因此無法區分國別。
4. ASUS\_Website有sc-bytes的欄位，但無法用來預估cache。
5. IP難以追查，且有很多問題導致無法用以作為用戶的判別。

思考

1. ASUS website log可拿來做區分的方式：日期、時段、國別

觀察ASUS Website log的流量趨勢可發現1400-1600流量最高、下載花費時間1600-2000最慢！

目前觀察IIS log也可以發現與ASUS Website log的花費時間有同樣的趨勢，其動作花費時間在1600-2000為高峰！（這部分可以套用上次的圖形以及這次的圖形來做說明）

1. 繪製兩個log的流量趨勢圖形（單日）- 已完成11/26 - 11/30的log

[..\2015年工作\20151225 - IIS\_log分析\ASUS\_abnormal\_url\_1130\_1to5.xlsx](../2015年工作/20151225%20-%20IIS_log分析/ASUS_abnormal_url_1130_1to5.xlsx) ↓第四個表格

經由 IIS log 設定管制界線找出，loading最費時的幾個uri後，拿來搜尋ASUS Website log 的資料來觀察所消耗的流量！ 其實排序找最大的幾個來看即可！例如搜尋出每日前50個最花流量的url。

R檔在116 → RImpala - IIS\_log\_analyse\_order\_time.R

此檔案根據IIS log 搜出cs\_uri\_stem：/websites/global/products/n1YzcJITwZrK4LaT/sse/sse2.swf

R檔在116 → RImpala - IIS\_log\_and\_ASUS\_Website\_log.R

然後依據上述uri 查詢ASUS Website log 的cs\_url，並可計算這條網址所占用的流量！

（一天約6GB，一個月約180GB）1GB = 0.58758 US$ = 19.15511 NT$ (一個月約3420元)

這網址主要是一個影片撥放！觀看cs\_referer欄位可以發現是由主機板的網頁而來：https://www.asus.com/de/Motherboards/MAXIMUS\_VII\_RANGER/

發現：ASUS Website log的sc\_bytes欄位當中，值最大的通常為sc\_status為200的，304通常sc\_bytes欄位值很小。

觀看IP欄位可以發現有些是重複IP，很可能是經常有重複LOADING

另外藉由這樣的查詢，可以看出哪些國家的Client對Server之間的傳輸time\_taken會很長。

因此可以開始去看IIS log於1400-2000這段時間的資料，看是那些uri造成網路流量速度降低！

搜尋每日超出正常範圍的資料，主要是觀察會不會有好幾天都出現同樣的URI，以此判斷是否為異常。

RImpala - IIS\_log\_and\_ASUS\_Website\_log\_Status.R

各種不同流量的佔比統計！

## 資訊統整 - 2016/1/8

目前有可能接增加的任務

1. java轉檔建立功能的程式
2. c# - Tim2他們的一些工作流程，如AMAX
3. 下周三要開始教R語言  
   目前教的方法就是直接在R上面打指令來執行！

# week2 - 1/11(一)~1/17(日)

## 上週五課會John交代事項：

1. IIS Log中找出的異常請提出讓網管與Marketing確認
2. Analytics工作者請將R script上傳git，並寫上註記
3. R課程於2016一月每星期四晚上6:30~8:30 (Howard/Teddy)
4. Big Data資料轉換作業交接: [FTP2IMPALA](ftp://FTP2IMPALA) (Tim2/Howard)

## 本週工作項目

1. R - Shiny 以及 RStudio產出HTML研究 < 已記錄至技術R筆記 >
2. R教學文件的書寫 < ok >
3. 學習如何使用GitHub
4. 請paul幫忙開Git權限

## IIS log與Speedera log分析統整

查找IP位址：<http://www.whatip.com/ip/66.249.65.120>

使用tmp\_akamai\_co\_speedera\_original\_log查詢後若要對照至IIS log則需要將tmp\_akamai\_co\_speedera\_log的f\_pos欄位的 "/www.asus.com/" 這一段刪除，只留後面的去搜尋cs\_uri\_stem的欄位

對比IIS\_log以及 Speedera log的以下邏輯資料時間及狀態是否對得起來

===============================================================

WHERE log\_date = '2015-11-30'

AND CAST(time AS TIMESTAMP) > '20:30:00'

AND CAST(time AS TIMESTAMP) < '20:40:00'

AND cs\_uri\_stem like '%ROG-Republic-Of-Gamers%'

AND cs\_uri\_stem like '%.png'

================================================================

# GitHub學習

Gerrit ← 這也是一個版本控管工具

Github ← <https://github.com/> 基本上這東西看看官方文件就可以懂了！不太需要買書。

以下為待研究：

* 程式專案的分支（Branch）
* GitHub讓程式專案飛上雲端
* Fork讓Git檔案庫分家後再合併
* Bitbucket比GitHub更好用
* GitLab完全免費再加送整個系統
* 利用Windows分享資料夾或是Git Daemon當成Git Server
* 利用HTTP/HTTPS存取Git Server
* 使用SSH認證和加密的Git Server