「Full-text index、Graph index、Time series

index非結構化之索引技術」

**Graph索引引擎安裝及操作說明書**

**財團法人資訊工業策進會**

2017/12/14

**目錄**

[一、 布署流程與環境需求 1](#_Toc501026655)

[布署流程 1](#_Toc501026656)

[所需檔案 1](#_Toc501026657)

[環境需求 1](#_Toc501026658)

[二、 前置作業 1](#_Toc501026659)

[三、 布署elasticsearch 2](#_Toc501026660)

[四、 配置tomcat及相關war檔 4](#_Toc501026661)

[五、 連線設定檔 4](#_Toc501026662)

[六、 延伸設定 5](#_Toc501026663)

[七、 API Document 6](#_Toc501026664)

圖目錄

[圖 1、布署開始執行畫面 2](#_Toc501026665)

[圖 2、public key執行畫面 3](#_Toc501026666)

[圖 3、elasticsearch設定執行畫面 3](#_Toc501026667)

[圖 4、elasticsearch安裝完成畫面 3](#_Toc501026668)

[圖 5、tomcat安裝與啟動畫面 4](#_Toc501026669)

[圖 6、Open Index起始畫面 4](#_Toc501026670)

[圖 7、Open index連線設定畫面 4](#_Toc501026671)

# 布署流程與環境需求

## 布署流程

1. 布署elasticsearch
2. 配置tomcat及相關war檔

## 所需檔案

* OPINX.tar.gz

## 環境需求

1. Ubuntu 14.04
2. Oracle-Java 8
3. Elasticsearch –2.3.4
   * **elasticsearch-analysis-ik-1.9.4**
   * **elasticsearch-analysis-mmseg-1.9.4**
4. apache-tomcat-7.0.72
5. 4 core CPU/2G ram \*3或以上配備

# 前置作業

　 解壓縮OPINX檔案至(/home/localadmin)

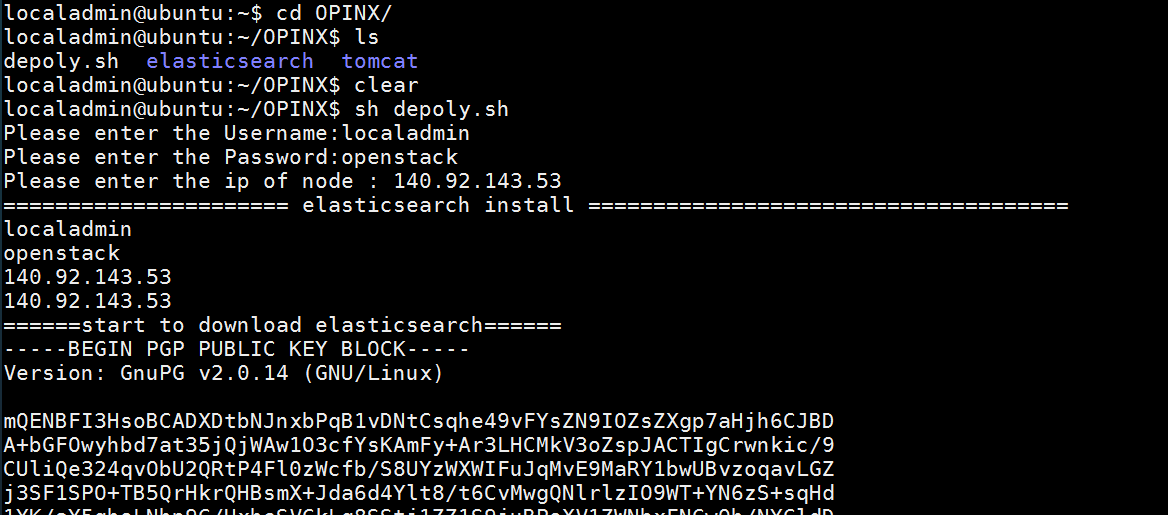
* tar –zxvf OPINX.tar.gz

# 布署elasticsearch

　　執行depoly.sh(使用root權限執行)

* cd OPINX/
* sudo sh depoly.sh

　　布署執行畫面如下：



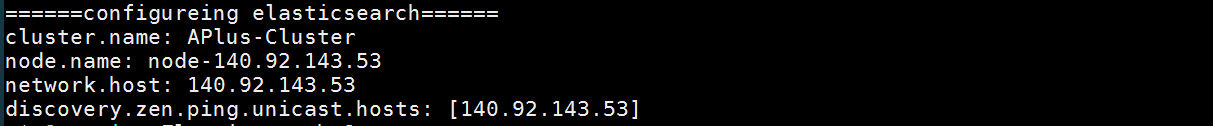
資料來源：本計畫

圖 1、布署開始執行畫面

　　布署過程中會檢查所需要的open-jdk-8，需要手動輸入accept通過認證條款，下圖為搬移所需要的elasticsearch plugins、elasticsearch 下載所需的public key：

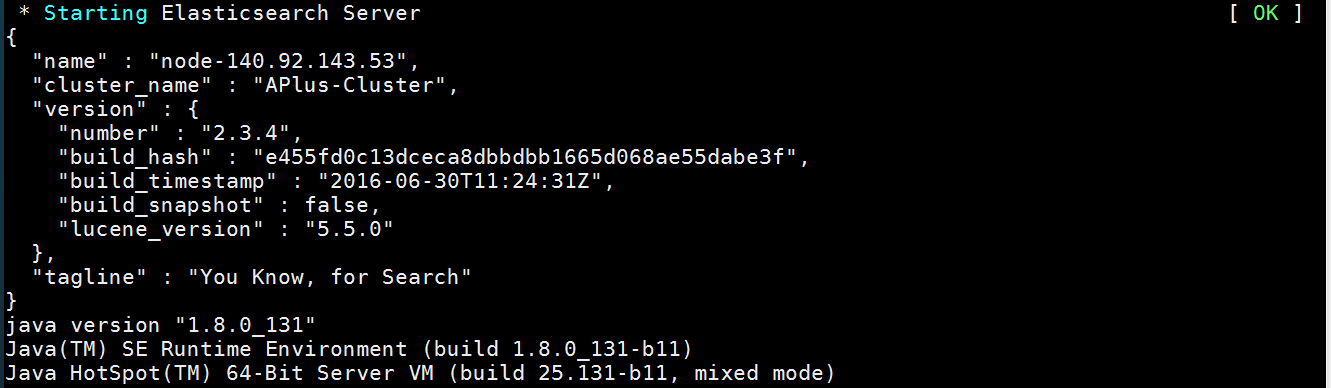
資料來源：本計畫

圖 2、public key執行畫面



資料來源：本計畫

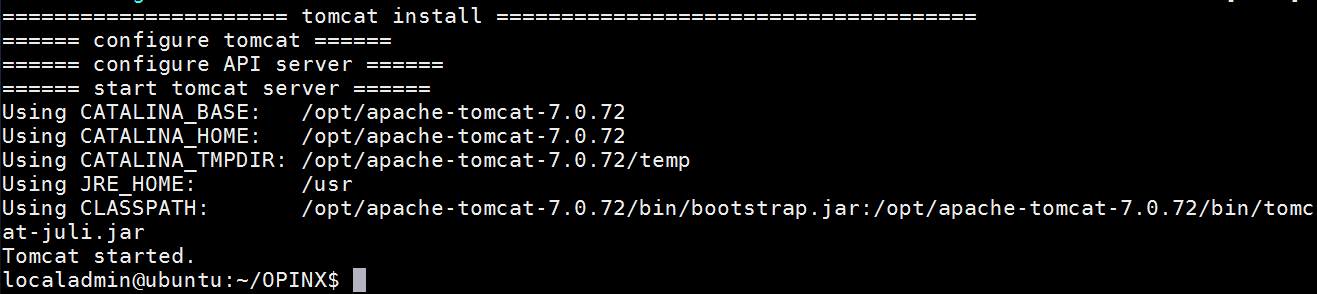
圖 3、elasticsearch設定執行畫面



資料來源：本計畫

圖 4、elasticsearch安裝完成畫面

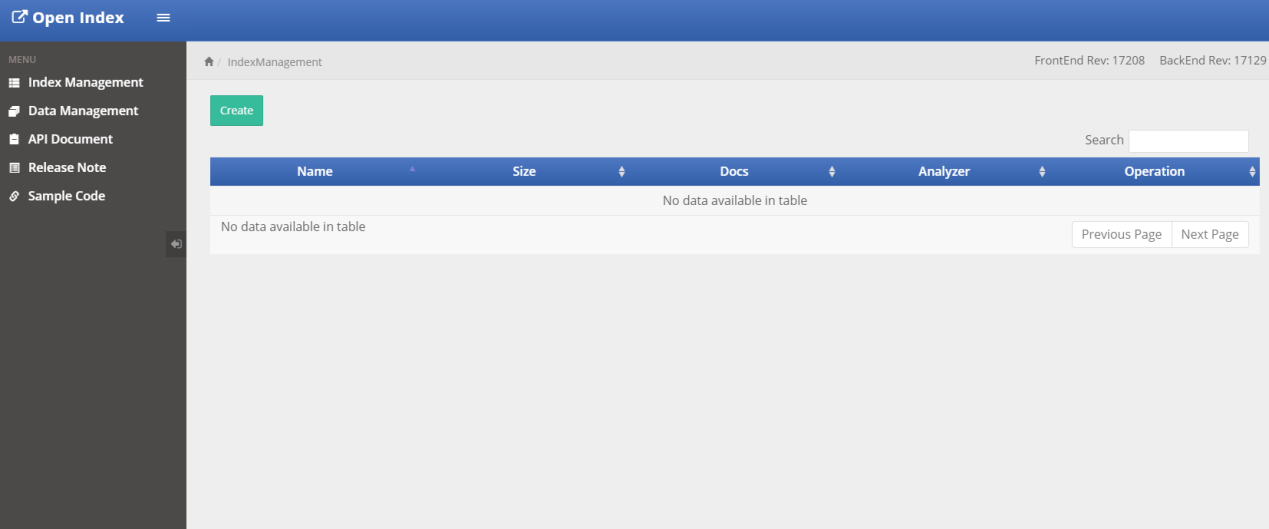
# 配置tomcat及相關war檔



資料來源：本計畫

圖 5、tomcat安裝與啟動畫面

　　開啟瀏覽器輸入<http://ip:8080/iServDB>即可看到Open Index的操作頁面：



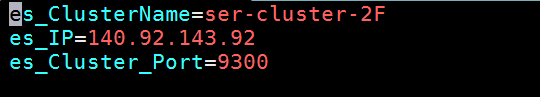
資料來源：本計畫

圖 6、Open Index起始畫面

# 連線設定檔

　　Open index API：

* + /opt/apache-tomcat-7.0.72/webapps/OpenIndex\_BackEnd/WEB-INF/classes/OPINX\_ini.properties



資料來源：本計畫

圖 7、Open index連線設定畫面

# 延伸設定

* 新增Elasticsearch node
  + wget -qO - https://packages.elastic.co/GPG-KEY-elasticsearch| apt-key add –
  + echo "deb https://packages.elastic.co/elasticsearch/2.x/debian stable main" | tee –a /etc/apt/sources.list.d/elasticsearch-2.x.list
  + add-apt-repository ppa:openjdk-r/ppa –y
  + apt-get update
  + apt-get install openjdk-8-jdk --yes --force-yes
  + apt-get install elasticsearch=2.3.4 --yes --force-yes
* 修改連線設定檔
  + vim /etc/elasticsearch/elasticsearch.yml
  + cluster.name: APlus-Cluster
  + node.name: $hostname
  + network.host: 0.0.0.0
  + discovery.zen.ping.unicast.hosts: ["192.168.111.63","192.168.111.64","192.168.111.65"]

service elasticsearch restart

* 驗證安裝成功
  1. curl <http://localhost:9200>

# API Document

* 新增索引資料庫：Create Index

***Request***

***URL***: /OpenIndex\_BackEnd/restAPI/create?

***Method***: GET

***URL*** ***Params***:

index\_name: 索引資料庫名稱 (只允許輸入小寫英文字母及數字)

index\_type: 索引資料表名稱 (與index\_name同)

analyzer:

ik\_max\_word ,

ik\_smart ,

mmseg\_simple ,

mmseg\_complex ,

mmseg\_maxword

***Reponse***

{

"status": true/false(成功與否),

"err\_msg": 錯誤訊息,

"index\_List": 列出所有index清單

}

* 刪除索引資料庫：Drop Index

***Request***

***URL***: /OpenIndex\_BackEnd/restAPI/drop?

***Method***: GET

***URL*** ***Params***:

index\_name: 索引資料庫名稱

***Reponse***

{

"status": true/false(成功與否),

"err\_msg": 錯誤訊息

}

* 查詢索引資料庫映射：Get Mappings

***Request***

***URL***: /OpenIndex\_BackEnd/restAPI/get?

***Method***: GET

***URL*** ***Params***:

index\_name: 索引資料庫名稱

***Reponse***

{

"status": true/false(成功與否),

"err\_msg": 錯誤訊息,

"index\_Mappings": index的mapping資訊

}

* 新增索引資料：Insert Data

***Request***

***URL***: /OpenIndex\_BackEnd/restAPI/insert

***Method***: POST

***Data*** ***Params***:

{

"index\_Name": 索引資料庫名稱,

"id\_Num": 插入資料之編號,

"content\_text": 資料內容,

"timeStamp":插入資料之時間(format: yyyy-MM-dd)

}

***Eample***:

{

"index\_Name": "test",

"id\_Num": "1",

"content\_text": "just test",

"timeStamp":"2016-01-01"

}

***Reponse***

{

"status": true/false(成功與否),

"err\_msg": 錯誤訊息,

"inserted\_Data\_info": 插入資料的詳細資訊

}

* 修改索引資料：Update Data

***Request***

***URL***: /OpenIndex\_BackEnd/restAPI/update

***Method***: POST

***Data*** ***Params***:

{

"index\_Name": 索引資料庫名稱,

"id\_Num": 插入資料之編號,

"content\_text": 資料內容,

"timeStamp":插入資料之時間(format: yyyy-MM-dd)

}

***Eample***:

{

"index\_Name": "test",

"id\_Num": "1",

"content\_text": "just test",

"timeStamp":"2016-01-01"

}

***Reponse***

{

"status": true/false(成功與否),

"err\_msg": 錯誤訊息,

"updated\_Data\_info": 插入資料的詳細資訊

}

* 刪除索引資料：Delete Data

***Request***

***URL***: /OpenIndex\_BackEnd/restAPI/delete?

***Method***: GET

***URL*** ***Params***:

index\_name: 索引資料庫名稱

id\_Num: 欲刪除索引資料之編號

***Reponse***

{

"status": true/false(成功與否),

"err\_msg": 錯誤訊息,

"deleted\_Data\_info": 刪除資料的詳細資訊

}

* 查詢索引資料：Search Data

***Request***

***URL***:/OpenIndex\_BackEnd/restAPI/select?

***Method***: GET

***URL* *Params***:

index\_name: 索引資料庫名稱

filter: 查詢索引關鍵字，運算子有AND(&&),OR(||),NOT(!),+,-，ex:A AND B表示取A跟B都符合的資料(交集)，A OR B表示取A或者B符合的資料，+與-則放在查詢的詞前面，表示一定要有或者一定沒有

＊優先順序由左至右，NOT>AND>OR，+,-只影響右邊的詞且可以使用"ABC"定義為一個詞

limit\_start: 查詢起始資料編號(至少為0)

limit\_num: 查詢資料結果筆數(至少為1)

timestamp\_start: 時間查詢條件(起始時間點)，格式為yyyy-MM-dd

timestamp\_stop: 時間查詢條件(終止時間點)，不輸入則為"now"，格式為yyyy-MM-dd

＊ timestamp\_start與timestamp\_stop皆不輸入，則為查詢全部資料

***Reponse***

{

"status": true/false(成功與否),

"err\_msg": 錯誤訊息,

"deleted\_Data\_info": 刪除資料的詳細資訊

}