快速佈署叢集式搜尋引擎- Crawlzilla

楊順發　郭文傑　陳威宇
國家高速網路與計算中心
{shunfa, rock, waue}@nchc.org.tw

摘要

Nutch 是目前最知名也是最好的 OpenSource 搜尋引擎專案之一，適用於搜尋企業內部網站及個人使用，但由於繁瑣的設定過程，許多使用者進行安裝及系統設定時，往往會遇到與多阻礙，尤其是叢集設定部份，往往會因為細部設定不正確，導致系統無法正常使用，進而無法體驗 Nutch 強大的搜尋功能。有鑑於此，本論文將整合 Nutch 搜尋引擎及其相關套件，開發一個完整且容易上手的安裝工具 - Crawlzilla，讓使用者可輕易安裝使用、快速佈署屬於自己或企業內部的搜尋引擎。除了單機版本的 Nutch 安裝外，本篇論文也提供叢集安裝功能，使用者可以視自己的硬體資源，選擇是否建立效率更佳的叢集式搜尋引擎環境。除此之外，Crawlzilla 也提供人性化的操作介面，讓使用者透過此一操作介面管理系統上單機/叢集中的系統狀況，在前端作業中，Crawlzilla 亦將所有功能整合於前端網頁中，提供更直觀的操作介面供使用者打造屬於自己或企業內部的搜尋引擎。

關鍵字：搜尋引擎、雲端運算、開放原始碼

Abstract

Nutch is the most well-know and one of the best search engine project for crawling internal enterprise or personal web sites, but many users encounter difficulties to setup and use due to the complicated operation process. This paper will integrate Nutch search engine and related useful packages to develop a complete and convenient installation software – Crawlzilla. This tool can assist users to quickly deploy their own private search engine within the intranet. Crawlzilla also supplies cluster installation feature to build cluster-type search engine environment depend on hardware resources. In addition, Crawlzilla provides a friendly user interface, allowing users to operate and manage system through this interface. Most of this system's operation functions will be integrated in the web page to help users creating their own or internal search engines in a more intuitive interface.

Keywords: Search Engine, Cloud Computing、Open Source

1. 前言

網際網路的發達，使得網頁資訊以非常快的速度增加，如何幫助使用者在短時間內搜尋到所需要的資訊，是搜尋引擎的重要功能之一。由於現有的搜尋引擎工具，如著名的 Google、Yahoo…等，都僅針對公開網頁提供搜尋服務，而較高機密性質的網站（如：企業內部網站）並不適用此類型的搜尋引擎建立索引並使用相關的搜尋服務。基於這些因素，許多高機密性質的網站必須建立屬於自己的搜尋引擎環境，雖然利用現有開放原始碼軟體 -Nutch[3]，建立搜尋引擎環境，可以節省花費成本，但繁瑣的設定過程，使用者不但在安裝前需要花不少時間研究如何安裝及使用，且在安裝過程中往往因為設定不正確，而產生許多狀況，使得 Nutch 搜尋引擎無法正常啟動使用，讓許多使用者放棄使用 Nutch。因此，本篇論文基於開放原始碼軟體-Nutch，實做一套快速佈署叢集式搜尋引擎環境，簡化 Nutch 的設定步驟、開發友善的管理介面、提供使用者更直觀的系統安裝及操作環境、並整合 Nutch 操作功能於前端網頁中，讓使用者僅需透過網頁介面即可操作使用。此外，於後端的系統中，我們也針對 Nutch 專案預設系統做了些許的更動，提供多重網頁爬取、多重網頁索引…等使用上更俱彈性的功能。

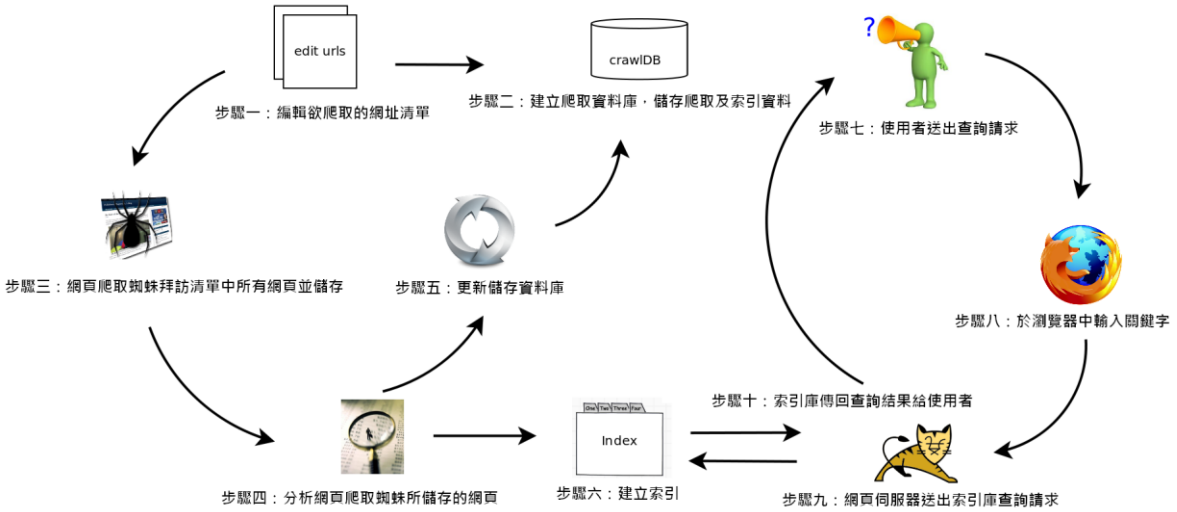
本篇的論文架構如下：第二章將依序介紹搜尋引擎的原理及 Hadoop 與 Nutch 之間的運作關係；第三章為 Crawlzilla 的系統設計及架構；第四章為系統實作部份，我們將於此章節介紹實作方法及系統展示；第五章為未來本專案將持續新增的功能及結論。

2. 動機及相關研究

本章節，我們將簡介搜尋引擎的運行原理，進而將焦點轉至搜尋引擎後端處理的 Hadoop[4]及 Nutch 的工作上，並詳細介紹兩個專案間的運作關係。

2.1 Nutch

Nutch 是一個架構於 Hadoop 之上，基於 Lucene[5]函式庫所開發的開放源碼軟體，隸屬於 Apache 的專案，由於採 OpenSource 授權方式，因



圖(一) Nutch 運作流程及其各系統元件

此程式碼完全公開，對於進階的使用者而言，可更彈性的修改程式碼，以符合進階使用者的需求，設計屬於自己的 Data Mining 工具，Nutch 的目標是讓每位使用者都能最小的成本運作屬於自己的網頁搜尋引擎，並提供高質量的搜尋結果，主要的功能可分為兩個部份，其一為 Crawl，目的是將設定好爬取的網站進行網頁爬取且建立網頁索引資料庫；另一個功能為 Search，透過網頁介面供使用者進行關鍵字查詢，也支援搜尋許多的文件檔案格式及不同的伺服器種類。

搜尋引擎的運作原理是利用自動搜尋機器人（或稱網路蜘蛛）的程式連結各個網頁的超連結，將搜尋到的連結與關鍵字整理成一份索引庫，當使用者輸入關鍵字送出查詢時，搜尋引擎會依據搜尋機器人所建立的索引資料庫中進行搜尋，並將搜尋結果送回網頁端。因此，在使用者下關鍵字前，搜尋引擎必須先建立好搜尋的索引資料庫，當使用者送出搜尋時，搜尋引擎便能很快速的在索引資料庫搜尋到結果並建立連結以網頁的形式呈現給使用者。我們以 Nutch 的運作流程為例，如圖（一）所示，首先，必須先建立欲爬取的網址清單（Step1），在搜尋機器人開始透過清單爬取網址前，會先建立一個索引資料庫，提供儲存爬取過後的結果（Step2,3），接著系統會將爬取過的網頁進行文字分析，且將分析過得結果送至一開始所建立的 CrawlDB 中，並且建立索引函式庫，重複這些爬取分析的步驟，直到預先設定的爬取深度（Step4~6），目前為止的系統步驟（Step1~6），都必須在使用者下查詢指令前完成，否則使用者查詢到的結果都將為空值，在 Step7~10 中，當使用者透過瀏覽器查詢關鍵字指令時，網頁伺服器（Tomcat）將會把客戶端的請求轉化為 Lucene 查詢並送至 Index 函式庫中，且將查詢的結果送至瀏覽器供使用者取得資訊，此一為 Nutch 整個運作的流程，也涵蓋了大部分搜尋引擎的運作模式。

在系統安裝的部份，Nutch 的安裝流程非常繁瑣，不但安裝前使用者必須先鑽研安裝設定及執行方法，接著須下載、安裝、設定及後續的網頁爬取…等步驟，即使 Nutch 是一套功能非常完整的搜尋引擎，但若加上設定及操作過程中的細節錯誤，尤其是當使用者欲建立叢集式搜尋引擎環境時更容易造成設定錯誤，這些因素往往會導致系統無法正常啟動使用，因而造成許多使用者使用上的困擾，不但花費許多時間且系統還是無法正常運作，因此，本論文的目的為簡化這些繁瑣的安裝及設定流程，不但可以透過 Crawlzilla 安裝單機或建立叢集式的 Nutch 搜尋引擎環境，也開發了友善的操作介面提供使用者管理單機或叢集環境及執行 Nutch 網頁爬取…等功能，讓使用者可以更輕鬆的上手 Nutch！

2.2 Hadoop

Hadoop 為 Apache 的專案之一，原主要為 Nutch 的一部分，後來獨立為一個專案，用來處理與儲存大量資料的雲端運算平台，由 Dong Cutting 所開發，以 Java 程式語言所開發，主要參考並實作 Google MapReduce[1]及 GFS[2] （Google File System）的雲端運算工具，使用的檔案系統為 HDFS，並具有分散式運算及運算節點備援等能力，在此一章節中，我們僅介紹 Hadoop 的核心運算及儲存架構。

Hadoop 實作 Google 所提出的 MapReduce 方法，做為解決問題的主要函式，基本解決問題的策略為 Divide and Conquer，主要為所有需要解決的問題拆解成兩個作業階段，Map 及 Reduce，所有待解決的問題，將於 Map 階段分析出程式所設定的結果，透過 Reduce 階段將這些結果進行合併，最後彙集成

最後完整的結果，並保存於 Hadoop 的檔案系統 HDFS 中，預設的檔案備份數量為三份，使用 HDFS 檔案系統，除了可以減少檔案毀損的風險外，優點是可以與 MapReduce 搭配，分散且平行的處理這些大量工作，可大幅的減少系統執行時間。Nutch 透過 Hadoop 中的 MapReduce 進行網頁爬取，預設僅有一個運算節點進行此項工作，此篇論文我們考慮了網頁爬取的深度越深，爬取的網站越多，消耗的運算資源也越多，因此加入叢集運算，透過增加硬體資源的方式提升運算效率。

3. Crawlzilla

於本章節中，我們將介紹 Crawlzilla 專案及其設計理念與架構，所有有關提及的功能也將於下個章節中進行系統展示。

3.1 Crawlzilla 簡介

Crawlzilla 專案已於 2009 年推出第一版，目的為幫助以單一機器、debian 系列為主的 linux 平台使用者快速佈署 Nutch 以及相關套件；並提供一個以 shell script 為主的 cdialog 介面，方便使用者爬取目標網頁以及設定爬取的深度。而 2010 年版本又加入了更多的新功能，如支援全系列的 Linux；加入叢集架構安裝與管理模式，動態的加入或移除運算和儲存節點，以增加運算速度；原本單調的使用者介面也改以用網頁來設計，這樣可以延續遠端控制 Crawlzilla 平台，並且提供更方便、簡潔、完整的操作步驟。

在客製化的搜尋引擎中，Crawlzilla 主要為使用者提供了友善的安裝介面與管理環境，使用者僅需要設定一組密碼及網路位址，即可全自動架設好一單機/叢集式的客製化搜尋引擎環境，透過系統管理介面及網頁操作環境，使用者可以更直觀的管理及執行網頁爬取建立索引的任務，此一版本亦提供可多工的網頁爬取功能，一次同時爬取並建立多個索引資料庫，而在後端的索引資料庫中，我們亦設計了多重索引的功能，使用者可以視自己的使用需求，在同一時間分別索引一個以上的索引資料庫，使用者可視自己的使用狀況，更彈性的佈署屬於自己的搜尋引擎。

3.2 系統架構及設計

Crawlzilla 主要分為三個部份，第一部份為系統安裝，目的是將所有的安裝程序，都整合於這個作業階段，協助使用者安裝 Nutch，再來透過 Crawlzilla 的管理介面管理系統狀態，如管理主機或叢集上的 Hadoop 程序及查看其系統狀態、管理 Tomcat 網頁伺服器以及作相關的系統設定…等，最後為安裝完後的管頁管理介面，使用者可以透過此一網頁管理介面，設定欲爬取的網站，讓系統執行，此一網頁也提供資料庫瀏覽及 Hadoop 系統狀態資訊提供使用者瞭解目前的系統資訊。

在軟體開發的過程中，為了使搜尋引擎運作的更有效率，因此加入了開發叢集式搜尋引擎環境，主要的好處除了可以更有效率的爬取網站上的資料，更可於執行多工爬取的任務時，大幅減少叢集內閒置的運算節點，提高網頁爬取建立索引資料庫的效率，以減少系統執行時間。

3.2.1 系統安裝

系統安裝的部份，採用以文字對話方式作為使用者介面協助安裝，好處是可以減少被安裝端主機必須安裝圖形介面函式庫的問題，並且還能讓使用者透過 ssh 遠端連線方式來操控。安裝的過程中，會自動匯入使用者的語系以提供更親善的使用環境，而使用者也僅需在五個步驟內即可完成安裝流程。此外，為了讓安裝後的系統可以順利操作，在安裝的過程中，系統會為使用者建立一帳號 -nutchuser，作為啟動 Nutch 相關軟體套件（如：Hadoop、Tomcat）的身份，所有叢集中的電腦也會使用相同的帳號密碼做為叢集運算節點彼此間溝通的工具，系統安裝的設計原則為簡化所有步驟，使用者僅需輸入一組密碼及選擇一組網路設備即可完成安裝，以符合 easy 安裝的設計理念。叢集安裝部份，目前系統設計為在 Master 安裝完後產生相關的安裝檔案，使用者僅需複製系統中的安裝檔至新的運算節點中進行安裝即可動態的加入新的運算節點至叢集中。

3.2.2 Crawlzilla 系統管理

因 Nutch 安裝後，所有的管理均須透過終端機輸入指令才可執行，即便使用者已順利完成安裝，若因使用者對終端機的操作不熟悉，仍無法順利執行系統進行搜尋，有鑑於此，本篇論文開發一套系統管理介面，目前提供的功能如下：

- *Cluster 狀態檢查：提供使用者瀏覽目前系統狀態，方便瀏覽 Hadoop 中的 Datanode 是否已被啟動。*
- *Cluster 狀態設定：提供使用者啟動/停止叢集中的 Datanode 及 Tasktracker。*
- *Server 狀態設定：提供使用者啟動/停止單機或叢集上 Master 節點的狀態。*
- *Tomcat 狀態設定：提供使用者啟動/停止伺服器及更改 port 號碼。*
- *Crawlzilla 亦提供英文版本，以便國外使用者進行系統操作。*

由於叢集設定需透過 root 權限修改相關的主機 host 名稱，因此 Crawlzilla 在安裝的過程中，必

須更動*/etc/hosts* 檔案設定叢機主機，以便叢集運算節點間的溝通，在系統移除時，Crawlzilla 也會將 */etc/hosts* 檔案還原至安裝前的狀態，不會造成系統移除後，系統仍保有一些無關的設定檔。

3.2.3 Crawlzilla 網頁管理系統

此一小節，我們僅描述網頁管理系統中最具特色的重點功能，網頁管理中基本的會員登入...等基本元素將不在此進行贅述；在網頁管理系統中，主要提供的功能項目如下：

- *Crawl：此一項目提供了建立索引資料庫最主要的網頁爬取功能，我們簡化了大部份的指令，使用者僅需要自行設定索引資料庫名稱、欲抓取的網址清單及爬取的網頁深度即可提交網頁爬取任務，此一功能亦支援多重網頁爬取，可同時提交多個工作項目，提高運算節點的使用率。*
- *資料庫管理：此一項目提供使用者瀏覽已建立的索引資料庫資訊，如起始 URL、本機索引路徑、文件檔數量及資料庫更新日期...等資訊，使用者可於進行瀏覽後刪除無效或資訊太過老舊的索引資料庫。*
- *系統狀態：此一項目提供使用者瀏覽系統狀態，如任務執行狀況、叢集數量...等。*

除上述功能外，為了使搜尋引擎更彈性及發揮單一主機/叢集的最大效益，我們也於 Nutch 的系統中做了部份修改，利用現有已建立的索引資料庫進行多重索引功能，於網頁管理的右側，我們建立了現有索引資料庫的搜尋引擎連結，使用者可視自己的使用需求，在同一個系統環境(單機/叢集)中，個別建立不同的搜尋引擎。

4. 系統實作與展示

本章節將介紹 Crawlzilla 專案的實作與系統展示；系統安裝及系統管理以 Shell Script 及 Dialog 作為主要的開發工具；在網頁管理介面中，由於 Nutch 專案中的網頁搜尋介面採 JSP 與 Tomcat 做搭配，為系統主要的執行環境，為考量系統管理及維護上的一致性，專案亦採用 JSP 搭配 Tomcat 作為開發及執行網頁的主要工具，在接下來的章節中，將逐一展示系統的操作及執行過程。

4.1 實驗環境

本實驗於虛擬機器 VirtualBox 下安裝測試，機器設定名稱分別為 VBox1 及 VBox2，作業系統均為 Ubuntu 10.04LTS，Sun Java 版本為 1.6.0.20，其他需安裝的套件分別為 OpenSSH、OpenSSH-Server、Dialog…等，因考慮作業系統版本之套件安裝方式不一，故套件檢查部份，僅能檢測出系統缺少的套件，而無法自行為使用者安裝；本次系統安裝分別採單機模式及叢集模式安裝建立 Crawlzilla 於 VBox1，並動態加入 VBox2 建立叢集式 Crawlzilla 環境，詳細的安裝測試過程，將描述下一小節。

4.2 Crawlzilla System Demo

本章節以 Crawlzilla 的三大元件進行系統使用步驟說明及展示，依序為(1)系統安裝、(2)系統後端環境管理、(3)索引資料庫的建立及搜尋引擎展示。

圖(二) 成功安裝後的 Crawlzilla 網頁系統首頁，使用預設密碼登入後，即可開始進行網頁爬取，搜尋...等系統操作

4.2.1 系統單機及叢集安裝

此一小節，我們將展示 Crawlzilla 安裝流程，首先，我們先於虛擬機器名稱為 VBox1 使用 Crawlzilla
安裝單機版 Crawlzilla，隨後並加入 VBox2 將其安裝成為叢集式環境，安裝步驟簡述如下：

- *Step1：系統將會依序檢查執行環境是否已安裝所需套件。*
- *Step2：系統會建立 nutchuser 帳號，作為執行 Crawlzilla 的身份，使用者須在此一步驟鍵入密碼。*
- *Step3：等待系統安裝設定完成，在網頁瀏覽器中鍵入預設網址 http://localhost:8080 或 http://your.system.ip.address:8080 即可瀏覽 Crawlzilla 安裝後的畫面，如圖(二)所示。*

單機安裝流程與上述步驟 Step1~Step3 相同，當系統出現圖(二)之畫面，即為安裝完成；若須建立叢集環境，除上述步驟外，還須執行下列安裝步驟，以下以 VBox2 執行 Client 安裝為例：

- *Step4：將 Crawlzilla 主機上的安裝檔案複製至 Client 中。*
- *Step5：進入 Client 資料夾後，以 root 身份執行 Client Install 即可開始安裝。*
- *Step6：至 Master 主機上執行系統管理介面，以加入新的運算節點至叢集中。*

其中，Step4 及 Step5，在 Crawlzilla 主機上安裝完成後，終端機會列印出相關的提示字元，僅須於 Client 端輸入這些提示字元即可將 Crawlzilla 主機上的系統安裝檔複製至 Client 主機中，並以 root 身份執行安裝即可。

系統安裝完裝完成後，於瀏覽器中輸入 *http://your.system.ip.address:8080* 即可開啟網頁管理頁面執行任務並瀏覽相關資訊頁面，詳細的網頁操作方式將於 4.2.3 章節做進一步的描述。

4.2.2 Crawlzilla 管理介面

Crawlzilla 管理介面，主要目的為讓使用者可輕鬆管理單機/叢集系統，執行畫面如圖(三)所示，所有的功能已於章節 3.2.2 中描述，透過此一介面，也可直觀的進行系統管理，因此，此一小節將不予進行贅述。

圖(三) Crawlzilla 系統管理介面，提供使用者最直觀的系統管理方式

4.2.3 Crawlzilla 網頁管理介面

此一章節將描述如何於網頁管理介面中進行建立索引資料庫，使用者僅需設定資料庫名稱、爬取的網頁清單及爬取深度即可送出 Job，由 Hadoop 執行網頁分析及爬取的任務，由於執行時間需視硬體資源及爬取網址深度…等其他因素而定，因此任務送出後至執行完成時間無法準確的預估，以國網中心中文網站為例，爬取的目標網址為 http://www.nchc.org.tw/tw/ ，深度兩層，執行時間約為 12 分鐘，執行完成後，於資料庫管理網頁可預覽資料庫目前資訊及相關的統計資料，如圖(四)所示，此一頁面除了顯示索引資料庫資訊外，也可直接點選進行刪除，在資料庫名稱的欄位上，也已建立專屬的搜尋引擎連結，搜尋引擎將針對此一資料庫進行搜尋服務，以此索引資料庫為例，於搜尋引擎輸入關鍵字-"nchc"，執行結果如圖(五)所示，使用者可依自己的需求，分別建立多重索引資料庫，更彈性的打造屬於自己的搜尋引擎。

圖(四) 索引資料庫預覽頁面介面參考圖，提供詳細的索引資料庫概況，若資料已過時，可直接於此一操作頁面進行刪除

圖(五) 以國網中心中文網站為例，爬取深度兩層，搜尋關鍵字 nchc 之執行結果

5. 結論與未來展望

對於企業高機密資訊的內部網站而言，考量資訊安全因素，無法使用類似 Google 開放式搜尋引擎平台為內部網頁建立專屬索引搜尋資料庫，因此必須透過其他方式為自己的企業內部建立專屬的搜尋引擎，本篇論文透過 Linux 下的開發工具及 JSP 網頁程式語言，設計一套簡化 Nutch 安裝過程的軟體-Crawlzilla，除了簡化原有系統的繁瑣安裝步驟，也提供相關的系統管理工具及網頁管理介面，目的是提供友善的操作介面，讓使用者可以順利安裝執行。除此之外，使用者也可視硬體資源數量，動態

的將硬體資源加入叢集環境中做為新的運算節點，以提升運算效率，在第四章節中，我們一系列展示了 Crawlzilla 系統安裝、管理及網頁介面的操作方式，提供了一系列較易上手，較直觀使用方法的套件工具。Crawlzilla 目前也持續更新於 Google Code Project Hosting 平台上- Crawlzilla [6]專案中，在未來，除了完整的提供英文版本供國外使用者使用外，此一專案將針對中文分詞功能進行研究，提供中文分詞索引功能，讓中文使用者可以更完整的於客製化搜尋引擎中使用中文搜尋。

參考文獻

[1] J. Dean and S. Ghemawat, MapReduce: Simplified Data Processing on Large Clusters, In Proceedings of the 6th Conference on Symposium on Opearting Systems Design & Implementation - Volume 6, San Francisco, CA, December 06 - 08, 2004.

[2] S. Ghemawat, H. Gobioff and S. T. Leung, The Google File System, 19th ACM Symposium on Operating Systems Principles, Lake George, NY, October, 2003.

[3] The Apache Software Foundation, Nutch, available at: http://nutch.apache.org/ , accessed 5 June 2010.

[4] The Apache Software Foundation, Hadoop, available at: http://hadoop.apache.org/ , accessed 5 June 2010.

[5] The Apache Software Foundation, Lucene, available at: http://lucene.apache.org/ , accessed 5 June 2010.

[6] Crawlzilla @ Google Code Project Hosting, available at: http://code.google.com/p/crawlzilla/, accessed 15 Sep 2010.