論搜尋引擎以程式在網路上自動抓取資料時可能面臨之法律問題及其解決之道

廖先志

國立交通大學資訊管理研究所博士班研究生 / 台灣桃園地方法院檢察署檢察官

Hsien-Jyh Liao

Email : hjliao1@mail.moj.gov.tw

陳鍾誠

國立金門技術學院資訊管理系助理教授

Assistant Professor, Information Management Department, Nation Kinmen Institute of Technology

Email : ccc@kmit.edu.tw

【摘要】

搜尋引擎必須以 crawler 程式（又稱 spider 程式）來自動抓取網頁並建立索引，如果 crawler 程式僅僅循著網頁所提供的超連結來搜尋網頁並抓取內容，稱爲一般性的 crawler 程式；如果不論網頁是否提供超連結，crawler 程式會自行計算並找到網頁的所有內容並加以抓取，此種 crawler 則稱爲深度 crawler。而 crawler 抓取網頁內容的步驟可以細分爲「取得資料」及「儲存並建立索引」二大步驟。在「取得資料」階段中，深度 crawler 雖然是自行透過演算法來取得網頁的所有內容，但本文認爲仍不至於構成非法存取（unauthorized access）。此外，不論是一般的 crawler 或是深度 crawler，如果取得網頁內容時會耗費網站資源而干擾網站的正常運作，就可能構成如美國 eBay 案中討論的財產侵害（trespass to chattel）。在「儲存並建立索引」階段中，原則上應該不會侵害網頁擁有者之重製權，然而、有些搜尋引擎 (例如 Google) 將其取得的內容以「庫存頁面」（cache）的方式允許使用者存取，此時即有爭議發生，但本文以爲，由於搜尋引擎的主要目的是在使網路使用者更容易接觸網頁，所以此種「重製」與「散布」行爲原則上應有著作權法「合理使用」原則的適用，故不會構成侵害著作權，但仍應考慮搜尋引擎與原網站之間是否處於競爭關係，以及所抓取之資料量佔原網站之比例等因素綜合判斷。要解決搜尋引擎與網站間可能發生的法律爭議，除可以強化現行的 robot exclusion 協定外，網站也可以考慮增強自動過濾 crawler 的功能，以杜絕爭議。

關鍵字：搜尋引擎、crawler、侵權行爲、著作權法

壹、引論

自 1991 年全球資訊網出現之後，網路上的資料量出現爆炸性的成長，甚至產生了資訊過多的問題， Yahoo 與 AltaVista 等入口網站於 1994 年開始出現，這類網站有系統的蒐集網頁並加以分類、過濾，以供使用者查詢，基於這樣的功能，這類網站被統稱爲「搜尋引擎」(search engine)。時至今日，搜尋引擎蒐集的資料越來越多，在網路世界中的影響也越來越大，以 Google 爲例，至 2004 年底爲止，它的資料庫中就有 80 億 5 千 8 百萬個網頁，11 億 8 千 7 百萬張圖片，10 億個新聞組訊息，6 千 6 百個列印目錄，4 千 5 百個新聞訊息[1]。同時， Google 和 Yahoo 甚至雙雙入選爲 15 個影響人類的網站[2]。

搜尋引擎的一個非常重要的組成部分，就是先行利用 crawler 在網路上取得網頁內容[3]，而取得對象是希望將所有網路上的資料都包含進來，所以我們甚至可以說只要曾經在網路上撰寫過的資料，幾乎都有可能被搜尋引擎抓下，搜尋引擎在取得內容後，會對這些資料建立索引，以方便查詢。而在這樣的「取得資料—儲存—建立索引」的過程中，搜尋引擎必須大量的向網站抓取資料，取用網頁的內容後儲存下來，這些動作引發了許多的法律爭議。本文的目標在釐清這些法律爭議，這不但對於搜尋引擎的經營者非常重要，同時對網站的管理者，甚至是曾經在網路上發表資料的人來說，都必須瞭解這個問題，如此才能在個人資料被搜尋引擎取用後，主張並維護自己的權利。

在這個法律議題上，目前的文獻大都是針對發生後的個案[4]，或是針對單一法律問題來進行討論[5]，而缺乏以搜尋引擎爲出發點的完整分析。因此，本文將先介紹搜尋引擎的架構與基本功能，區分出組成單元與步驟，然後分析各個步驟中可能產生的法律問題及避免方式。接著，再列舉網站所可能採取的幾種防止 crawler 之方式，並分析其法律意涵。最後，我們提出結論與未來的研究方向。

貳、搜尋引擎與 crawler 簡介

2.1 背景

搜尋引擎是由一組具有網頁蒐集、儲存、索引、查詢等功能的程式所集合而成的 [6]，其中網頁蒐集的程式稱爲 crawler 或 spider，crawler 程式的設計原理是利用追蹤網頁上的超連結以不斷尋找新網頁，這些被蒐集到的網頁經組織後會有系統的存在硬碟中，接著一個稱爲 Indexer 的程式會對這些檔案中出現的每個字詞都建立索引，以便查詢。如此當使用者在搜尋引擎的首頁中輸入查詢字詞時，搜尋引擎就會根據索引，找出所有曾經出現過該字詞的檔案，並傳回查詢結果給使用者，下圖顯示了搜尋引擎的基本架構。

---

[1] 見 http://zh.wikipedia.org/wiki/Google（last visited Sep. 30 2006）

[2] 見英國觀察家報，http://observer.guardian.co.uk/review/story/0,,1843263,00.html（last visited Sep. 30 2006）

[3] 這些內容包含討論區、BBS、論文、個人的網路日記(又稱部落格)等資料。另外，此處的內容，暫時先限定爲文字內容，也不包含以縮圖方式呈現的「圖形搜尋」，因爲以縮圖建立索引時，所應考慮的技術問題雖然與處理文字內容的技術類似，但提供使用者搜尋結果的表示方式則大不相同，而美國法院也在 Kelly v. Arriba Soft Corporation (280 F.3d 934 (CA9 2002)及 2006 年的 Perfect 10 v. Google Inc. et al. (CASE NO. CV 04-9484 AHM (SHx))二個案件中表示不同的結論，故此部分將留待將來專文討論。

[4] 林發立，" Internet 的優勢與問題--從 TicketmasterCorp. v. Tickets.com Inc.「深入連結」一案談起"， 萬國法律，第 112 期，頁 49-52，2000 年。Pamela Samuelson, "Unsolicited Communications as Trespass?", Comm. ACM, Vol. 46 No. 10, P15-20, Oct. 2003。Kevin Emerson Collins, "Cybertrespass and Trepass to Documents", Clev. St. L. Rev., Vol. 54, P41-66, 2006。

[5] 郭寶明，"网站搜索引擎提供者著作权侵权风险的法律分析"， www.law-lib.com/lw/lw_view.asp?no= 1397（last visited Sep. 30 2006）

[6] 參李曉明、閻宏飛、王繼民, "搜尋引擎 – 原理、技術與系統"，科學出版社, 2004 (簡體) 及 Brin S, Page L. "The anatomy of large-scale hypertextual Web search engine", In Proceedings of the 7th International World Wide Web conference/ComputerNetworks, Amsterdam, 1998.

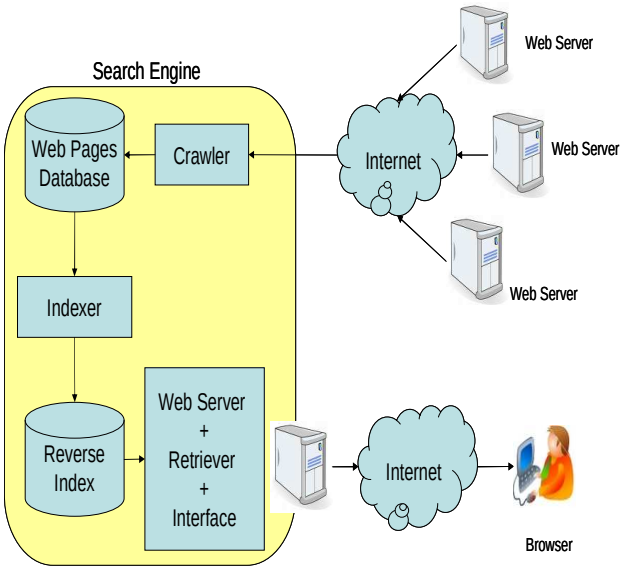

圖 1　搜尋引擎的基本架構圖

上述架構乃是一個簡化的結果，某些搜尋引擎的設計還加上了其他的元件以增進搜尋效能，例如：排序 [7]、壓縮、分散資料等等，然而就本文的法律分析而言，上述的架構已經足夠，以下將依照上述的這個架構簡介搜尋引擎的各個組成部分。

2.2 網頁抓取程式

圖 1 中的網頁抓取程式 crawler 是一個追蹤網址以取得網頁的程式，詳細的做法大致上可分爲下列幾個步驟：

1. 事先在網址資料庫中放入一些網址，稱爲起始網址。(此部分通常爲手動而非自動)
2. Crawler 從網址資料庫中取得一個待存取的網址 (URL)[8]。
3. Crawler 根據該網址取得網頁內容 (HTML 格式的文件)。
4. Crawler 剖析網頁的內容後，取得其中的所有超連結 (Hyperlink)，並將這些超連結所連接的新網址加入網址資料庫中。
5. Crawler 將所取得的網頁存入磁碟或資料庫中，以供搜尋引擎建立索引之用。
6. 回到步驟 2 ，以取得並處理下一個網址。

如此只要一開始時網址資料庫中有少數的網址，crawler 就能根據這些網址來取得網頁，然後找到更多的網址，進而取得更多的網頁，如此反覆循環，不斷的擴大網址資料庫，經過這樣的程序，只要曾經被連結過的網頁大部分都會被囊括進資料庫中，下圖顯示了 crawler 的運作過程。

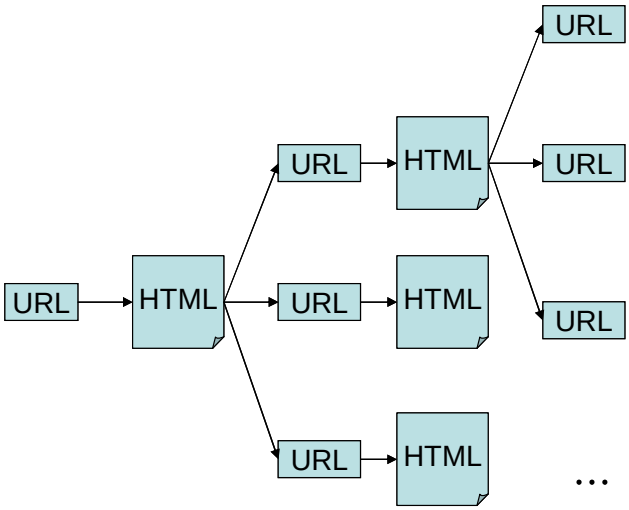

圖 2　Crawler 抓取網頁、取出網址的過程

然而由於 crawler 的運作過程中，並沒有包含徵詢網站維護者或網頁製作者同意的機制，爲了避免發生這樣的爭議，於是 發展出了 robot exclusion standard (RES) [9] 機制，RES 可以用來向搜尋引擎表達禁止

[7] Google 的排序方法請參考 Page L, et al. “The PageRank Citation Ranking : Brining Orderto the Web”, Stanford Digital Library Technologies Project, 1998. (http://citeseer.ist.psu.edu/page98pagerank.html last visited Sep. 30 2006)

[8] 所謂的 URL，是 Universal Resource Locator 的縮寫，直譯成中文可以叫「全球資源定址器」，URL 實際上就是一個位址，這個位址可以引導瀏覽器或 crawler 存取到 URL 所指向的內容，這個內容可能是一個檔案或者是從資料庫中查詢所組合出來的一些紀錄。參見 David Fox,Tory Downing 著，深入 HTML3 WEB 設計，松格資訊有限公司，1995，頁 14。

[9] 參考 Martijn Koster , “A Standard forRobot Exclusion”, 網址: http://www.robotstxt.org/wc/norobots.html（last visited Sep. 30 2006）及 Wong C. “Web Client Programming with Perl” , O’Reilly, 1997, Appendix C. 網址: http://www.oreilly.com/openbook/webclient/appc.html. （last visited Sep. 30 2006）

crawler 搜尋的意思。RES 的運作方式很簡單，只要網站維護者在根目錄底下放置一個 robot.txt 的檔案，指定哪些資料夾或檔案不希望被搜尋引擎搜尋，那遵守 RES 協定的搜尋引擎就不會這些資料夾中的網頁，以避免侵犯著作權 [10]。以下是一個 Robot.txt 的範例：

```
User-agent: webcrawler
Disallow:

User-agent: lycra
Disallow: /
User-agent: *
Disallow: /tmp
Disallow: /logs
```

圖 3　Robot.txt 的一個範例

上述範例中，第一個段落表示 webcrawler 這個程式可以抓取所有的資料夾，第二個段落表示 lycra 這個程式不能抓取根目錄以下的所有資料，第三個段落表示所有其他程式都不能抓取 /tmp 與 /logs 這兩個資料夾。

雖然 RES 可以告知 Crawler 不要蒐錄該網站的內容，然而 Crawler 是否遵循 RES 則完全根據 Crawler 的意願，並無強制性，一個不遵守 RES 的 Crawler 仍然能夠抓到這些網頁，因爲 RES 並無防護能力。另外還有一個更嚴重問題，就是網頁內容的著作者通常並非網站維護者，也不見得知道 RES 協定的運作方式，因此 RES 無法完全解決搜尋引擎的法律爭議。

1995 年以前，全球資訊網所採用的是靜態呈現技術，也就是所有網頁內容都是固定的，網站維護者必須將網頁一頁一頁傳送到網站上。隨著網路技術的發展，出現了所謂的動態網頁呈現技術，這種技術乃是在網站伺服器上加裝一個程式，該程式會根據使用者在網頁查詢表單中所填入的值，動態的從資料庫中取出資料，組合成網頁後傳回給瀏覽器，圖 4 顯示了 eBay 首頁中的表單範例。

圖 4 動態網頁中的表單欄位–以 ebay 爲範例

由於動態網頁背後資料庫所包含的資料量通常相當龐大，因此、有人稱這些不被超連結所連結到的所有資料組成的 Web 爲 Deep Web[11] (又稱 Invisible Web 或 Hidden Web)，而可由超連結所連結到的部分則稱爲 Surface Web。傳統 Crawler 追蹤 URL 的方式並無法抓取到 Deep Web 中所包含的資料，因而有一種稱爲 Deep Crawler[12] 的程式被發展出來，試圖抓取 Deep Web 中的資料。要存取這些資料必須傳送一組表單的參數給網站，使用者可以很容易的閱讀表單後填入適當的值於欄位中，因爲這類的表單通常是爲人類而設計的，但程式卻通常無法理

[10] Google 與 Yahoo 等較知名的網站大多有遵循 Robot Exclusion Standard. 請參閱 Google 網頁 http://www.google.com/support/webmasters/bin/answer.py?answer=35302 與 Yahoo 網頁 http://help.yahoo.com/help/us/ysearch/slurp/slurp-02.html

[11] 參考 Wikipedia 維基百科, “Deep Web”, 網址：http://en.wikipedia.org/wiki/Deep_web

[12] 參考 Michael K. Bergman, "The Deep Web: Surfacing Hidden Value". The Journal of Electronic Publishing 7 (1), 2001.及 Sriram Raghavan and HectorGarcia-Molina "Crawling the Hidden Web". In Proceedings of the 27th International Conference on Very Large Data Bases (VLDB), 129-138, 2001.

解這些表單，因而，因此要設計一個全自動的 Deep Crawler 以抓取 Deep Web 的資料相當不容易，目前的技術通常使用的是半自動的方法，也就是由人類操作者配合程式一同決定參數，然後再由 Deep Crawler 進行後續的抓取工作的方法。

2.3 網頁資料庫

Crawler 取得資料並儲存於網頁資料庫 (Web Pages Database) 中，這些資料分爲兩類：第一類是大量的新網址，第二類是每個網址所對應到的網頁。這些網址和網頁都會被儲存起來，儲存的方式也有兩種：一種是儲存在檔案系統中，另一種是儲存在資料庫中，就法律的角度而言，這兩種儲存方式純粹是技術上的不同，在此不進行詳細描述。

2.4 索引建立程式

一旦這些網頁被下載儲存在資料庫之後，索引程式 (Indexer) 將會被啓動，這個索引程式會對網頁中的每一個字詞都進行索引，也就是讀取每一個網頁後，將其中每一個字詞所出現的位置進行紀錄，紀錄的方式通常有兩種，一種稱之爲指紋檔 (Signature File)，一種稱爲反轉索引檔 (Reverse Index)，由於反轉索引檔的查詢速度較快，因此目前的搜尋引擎幾乎都是使用反轉索引檔技術。

2.5 反轉索引檔

所謂的反轉索引檔 (Reverse Index) [13] 是一個以字詞爲核心的索引檔案，其結構如下：

$Word_1$: $d_{11}d_{12}$ …. D_{1k}
…
$Word_n$: $d_{n1}d_{n2}$ …. d_{nr}

圖 5　索引檔案結構

上圖中的 $Word_1$, $Word_2$, …, $Word_n$ 都是字詞，而其後的 $d_{11}d_{12}$ …. d_{nr} 是文件索引，代表的是曾經出現該字詞的檔案文件。在電腦上，爲了節省儲存空間，反轉索引檔中的 $d_{11}d_{12}$ …. d_{nr} 通常以一個整數代表，該整數稱爲文件代號，搜尋系統根據該代號、配合一個 (代號、文件) 的對映表格，就可以取得文件的 URL 與網頁內容。

建立反轉索引檔的目的，其實是爲了進行快速查詢之用，由於網路上的文件數量龐大，如果在使用者輸入查詢詞彙後，再一個一個檔案進行比對，那對數億的網頁而言，可能會花上數十天的時間才能查到資料，但在建立反轉索引檔之後，只要不到一秒鐘，就可以查出所有包含該查詢詞彙的網頁了。

2.6 網站伺服器

在建立完索引檔後，搜尋引擎還需要提供一個網路介面，此時搜尋引擎必需提供一個網站伺服器(Web Server)，讓使用者可以查詢到這些網頁資料，因此搜尋引擎的提供者必須建立一個網站，然後設計一個查詢介面，在使用者輸入查詢詞彙後，呼叫查詢程式去取得曾經出現該詞彙的網頁，然後將這些網址與網頁顯示在銀幕上，讓使用者瀏覽點選，這個過程中牽涉到的三個元件是「網站伺服器」(Web Server)、「查詢程式」(Retriever)與「查詢介面」(Query Interface)。

所謂的Web Server是一個電腦加上一組程式，該程式會等待客戶端(Web Client)的連線，而這個 Web Client 程式，就是我們所常使用的瀏覽器，目前常用的瀏覽器有微軟的 Internet Explorer，開放原始碼的 Mozilla 與 Firefox 等等。以 Google 爲例，當使用者在瀏覽器的網址列上打上 Google 的網址

[13] Baeaz-Yates R, Riberiro-Neto B, “Modern Information Retrieval”, Addison Wesley, Longman, 1999.

http://www.google.com 時，瀏覽器會傳送一個要求訊息給 Google 這一台伺服器，當 Google 收到該訊息時，就將其首頁 (包含了查詢畫面) 的 HTML[14] 檔案傳回給瀏覽器，當瀏覽器收到這個檔案後，會將檔案以網頁的方式呈現在瀏覽器的銀幕上，下圖顯示了這個連線與回傳的過程。

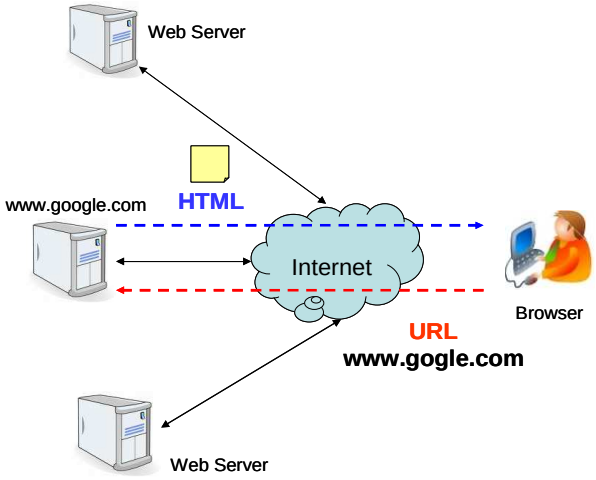

圖 6　使用者透過瀏覽器連線到網站的過程

2.7 查詢介面

上述過程中，www.google.com 所傳回的 HTML 文件，實際上是一個查詢介面(Query Interface)，其畫面如下：

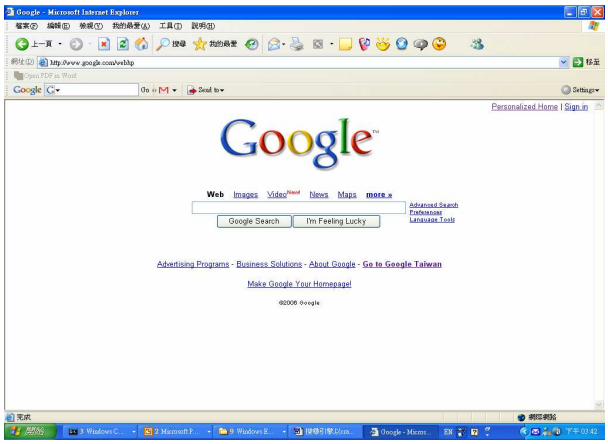

圖 7 瀏覽器根據 HTML 網頁繪製的畫面

雖然上述的畫面看來不像一個文件，反而像是一個圖片，但實際上，是瀏覽器根據 HTML 文件指示所畫出來的一個畫面，其 HTML 文件的部分內容顯示如下：

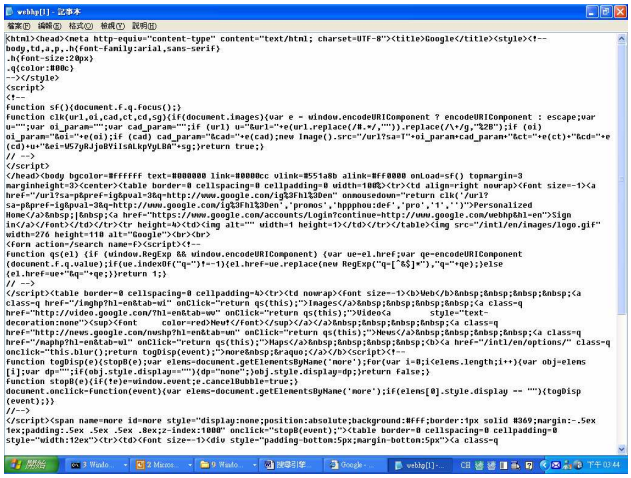

圖 8　Google 所實際傳回的網頁內容

上述 HTML 文件雖然複雜，但實際上就是指示瀏覽器畫出 Google 首頁的命令文件，其中、最重要的內容是一段稱爲 FORM 的標記，這個標記包含了畫面中的輸入欄位，爲了讓讀者容易理解，我們將 Google 的 FORM 內容簡化如下：

```
<form action=/search name=f>
    <input maxlength=2048 size=55
name=q value="" title="Google Search"><br>
    <input type=submit
value="Google Search" name=btnG>
</form>
```

圖 9　簡化後的 Google 的 FORM 內容

其中 <input maxlength=2048 size=55 name=q value="" title="Google Search"> 所代表的是一個輸入欄位，而 <input type=submit value="Google Search" name=btnG> 所代表的是畫面中的 Google Search 按鈕，而最外層的 form 是表單的標記，其中的 action 欄位所指定的 action=/search 代表的是「當按鈕被按下時將欄位訊息傳送給 /search 這個程式」，這個程式就是搜尋引擎中的查詢程式。

14 W3C. “HTML 4.01specification”, WWW Consortium (W3C) http://www.w3.org/TR/1999/REC-html401-19991224/

2.8 查詢程式

當查詢程式 (Retriever) 收到網路上所傳來的查詢詞彙後，會根據這個詞彙到 Reverse Index 中取出對應的文件代號串列，然後再根據這些文件代號，取出對應的網址與網頁，最後將這些資料加以封裝後，轉換成 HTML 格式的網頁文件傳回給使用者的瀏覽器，當瀏覽器接收到這個 HTML 文件時，會再度將其轉換成圖形畫面顯示在銀幕上。

然而網路上的網頁數量實在太多，往往符合查詢條件者會有上千萬個，此時就須仰賴排序的功能，傳統的排序方法通常是以語意相關度爲主的方法，然而在網路資料量太大的狀況下，語意符合的文件非常的多，使用語意相關度也就不具有太大的意義，於是 Google 採用了利用網頁被連結次數作爲網頁重要性的判斷方法 [15]，這也是 Google 受歡迎的原因之一。

下圖是 Google 搜尋 book 這個詞彙時，所顯示的畫面，其後的圖是這個畫面的 HTML 文件內容。

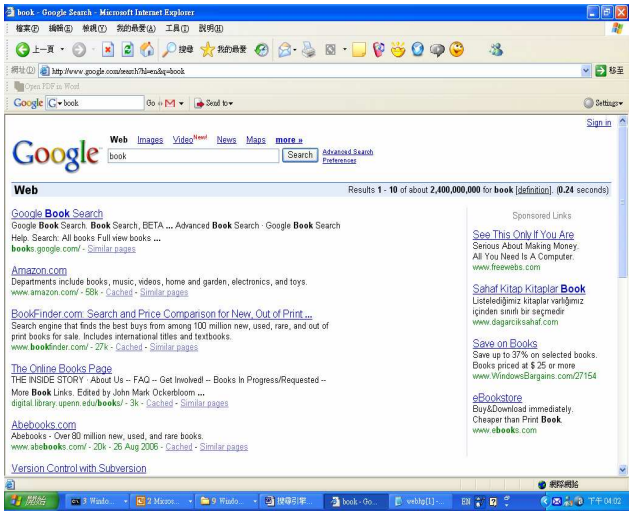

圖 10　Google 搜尋結果的畫面

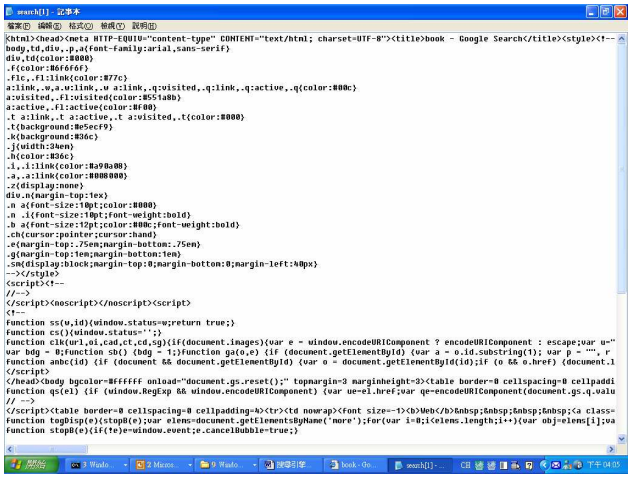

圖 11　Google 搜尋結果的網頁 HTML 文件

由以上的分析，我們可以將 crawler 抓取資料的過程區分爲「取得資料階段」與「儲存並處理階段」，其中取得資料階段是指 crawler 找出 URL，再依照 URL 去抓取網頁的資料；而儲存階段則是 crawler 將資料抓回後，先儲存起來，以備將來的運用。因此，以下本文將依照上面的描述的二個階段，以法律的角度來分析其所可能遭遇到的法律問題。

參、Crawler 之法律分析

3.1 取得資料階段

Crawler 的第一個重要步驟就是從資料庫中取出一個 URL， Crawler 依照這個 URL 發出需求（request），要求取得這個 URL 所指的資料，然後等待伺服器回傳（reply）資料，不論是一般的 crawler 或是深度 crawler 其取得資料的過程大致都是如此，所不同處在於一般性 crawler 是從資料庫中取出 URL；但深度 crawler 使用的 URL 不是網頁所直接提供的，而是 crawler 以演算法計算或人工輔助猜測出可能的 URL 組合，然後嘗試以這些計算出來的 URL 來取得資料。在這個階段中，本文以爲，可能碰到的法律問題有二個，以下分別討論之：

[15] 同註 7。

3.1.1 無權限存取[16]（unauthorized access）

就一個一般的 crawler 來說，上述的「Request--reply」過程與一般使用者瀏覽網頁時，取得網頁資料的過程大致上並無不同，只是 crawler 是以程式自動爲之而已，因此此時發生法律爭議的可能性不大。但是深度 crawler 則有所不同，其最主要的不同處就是在於深度 crawler 的 URL 並不是網頁所直接提供，而是由 crawler 自己計算出來，換句話說這些隱藏（hidden）的 URL 是 crawler 自己「發現」的，也就是說，這些 URL 並非是網頁自願性的提供出來，這就牽涉到一個問題：此時，crawler 是否算是在「侵入」網站？crawler 是否是在進行非法存取？

美國最早有關「侵入網站」的法律是 1984 年立法的「防止電腦詐欺與濫用法」（The ComputerFraud and Abuse Statute，CFAA），即美國聯邦法規 18USC、第 47 章、第 1030 條（18 U.S.C1030），其主要目的在保護電腦與網路的正確性、整體性和安全性。該法先後在 1986 年、1994 年修定，最後一次則是在 1996 年的資訊基礎建設保護法（The National Infrastructure Protection Act of 1996）中修定，該次修正重點主要在於爲了使適用範圍能夠涵蓋到所有在州際商業或通訊中的電腦，乃將其所保護的客體範圍由原先的「聯邦利益之電腦」（federal interest computers） 擴大到及於「受保護之電腦」（protected computers）的概念，使之不再僅限於對政府或金融機構的電腦系統，只要無權限或超越權限而存取與網際網路係處於連線狀態之電腦，就有可能構成本法規定之犯罪[17]。

基於美國上述的法律及相關案例，本文以爲，雖然深度 crawler 取得的URL並非是網頁主動在網頁中展示，而是以自行計算的方式取得，但是若先不考慮 crawler 這個因素，以一般使用者上網瀏覽網頁的狀況來看，網站仍然可能因爲使用者輸入不同的查詢條件而讓使用者可以存取到隱藏的URL所指的特定網頁資料，只是深度 crawler 是採取自動的作法，而且 crawler 會以演算法盡量算出所有的URL[18]，換言之，深度 crawler在這個地方與人工瀏覽最大不同的只是在「自動」與「全面」二個地方而已，因此，從這個角度來看，本文以爲，不論是深度或一般 crawler，與一般使用者在「是否有權限取得」這一點上沒有任何不一樣的地方，所以並不會構成無權限存取。

但是，也就是因爲 crawler 是「自動」，而且深度 crawler 還具有「全面」的特性，實際上 crawler 在與網頁伺服器之間的關係還是與使用者以人工方式瀏覽有相當的不同，其中最大的不同之處就在於 crawler 所發生的需求（request）數目相當龐大，這使得網頁伺服器必須投入相當比例的資源來處理，此時，就可能導致下面我們要探討的另外一個法律問題：是否會有侵害財產權（trespass to chattel）的問題？

3.1.2 侵害財產權（trespass to chattel）

這方面第一個指標性判決是發生在美國的 eBay, Inc. v. Bidder's Edge, Inc[19]。該案事實大致如下：原告 eBay 網路拍賣網站是全球最大的拍賣網站，主要業務是提供其使用者相互間買賣物品的平台，在案發時，eBay 大約有七百萬個註冊客戶，每天有將近 40 萬個新增的拍賣物品。被告 Bidder’s Edge 公司（以下稱 BE 公司）則是一個集成網站（aggregation

[16] 此處本文以無權限存取取代國內學界常見的「非法入侵」。因爲「非法入侵」這個名詞，或許可以比較具象化的表現出電腦系統遭到攻擊，其資料就如同銀行金庫中的錢一樣，被入侵者任意取走，但是實際上，如果考量到電腦系統的現實狀況，我們就會發現，實際上大多數被入侵的系統裡面並沒有進駐一個「入侵者」，所謂的入侵，常常只是沒有權限的使用者可以任意從外部存取系統內的資料而已，而美國法中也是用「unauthorized access」或「exceed authority access」來稱呼這樣的狀況，因此，本文以下都將使用無權限存取，以避免誤會。

[17] Michael Hatcher, Jay McDonnell & Stacy Ostfeld, ComputerCrimes, 36 Am. Crim. L. Rev.,at 399-402 (summer, 1999).

[18] Alexandros Ntoulas, Petros Zerfos, Junghoo Cho, "Downloading Textual Hidden Web Content by Keyword Queries" ", Proceedings of the Joint Conference on Digital Libraries (JCDL), p100-109, June 2005。

[19] eBay, Inc. v. Bidder’s Edge, Inc., 100 F. Supp. 2d 1058 (N.D. Cal. 2000)

site），該網站提供的一項功能讓線上買家可以在該網站中找到數個網站所提供商品並，而不必分別到不同拍賣網站去投標，其所涵蓋之拍賣網站包括 eBay。爲了達到上面所說的功能，BE 每天存取 eBay 達 10 萬次以複製 eBay 網站內的拍賣物品，大概複製 eBay 網站資料庫約 1.10%。

審理該案的法官最後認爲被告 BE 公司的行爲，已經構成了侵害財產權（trespass to chattel），他認爲 BE 的行爲主要造成了兩點侵害[20]，第一：BE 所發出需求封包數量已經達到 eBay 每日處理總量的 1%，而如果不遏止這樣行爲的話，可能會有其他的網站也從事類似的行爲，這將導致 eBay 網站的崩潰。第二：被告 BE 公司的行爲確實干擾（interference）了 eBay 公司對該公司網站的排他性控制[21]。

如果依照上述的判決所持的理由，一旦 crawler 發生大量的 request 並存取網頁內容，就有可能因爲干擾網站的正常運作而構成侵權行爲。而這種情形不論是否是一般的 crawler 或是深度 crawler 其實都有可能發生，只是深度 crawler 通常會產生大量不論有效或無效的 URL 來存取網站，所以比較有可能會造成侵權的狀況，像是上面介紹的 eBay 案中，造成侵權的正是可以存取特定格式資料的深度 crawler。或許也因爲這個原因，如 Google 的 GoogleBot，Yahoo 的 Yahoo Slurp 等目前較知名搜尋引擎的 crawler 都會遵循 Robot Exclusion Standard，所以這些 crawler 會在如下列的請求封包表頭的 User-Agent 欄位告知對方自己的名稱，以方便網站判斷是否要接受 crawler 的請求，並且以定期的方式傳送連線請求[22]，以避免目標伺服器負擔過重，干擾到對方的正常運作，如下圖所示：

```
GET /index.htm HTTP/1.0
Host: ccc.kmit.edu.tw
Accept: */*
User-Agent: Mozilla/5.0 (compatible; Yahoo! Slurp China; http://misc.yahoo.com.cn/help.html)
If-Modified-Since: Wed, 06 Sep 2006 11:51:02 GMT
Accept-Encoding: gzip, x-gzip
```

圖 12　Yahoo Slurp 的表頭

但是也有一些 crawler 不遵循定期傳送，盡可能不干擾對方的原則，例如： Teleport 這套市面上販售的 crawler 中，就允許使用者設定以很快的速度傳送請求，這可能會造成網站的過於忙碌，另外 Teleport 也可以僞裝成 Internet Explorer 等瀏覽器，以避免被目標伺服器拒絕連線[23]，其設定如下圖：

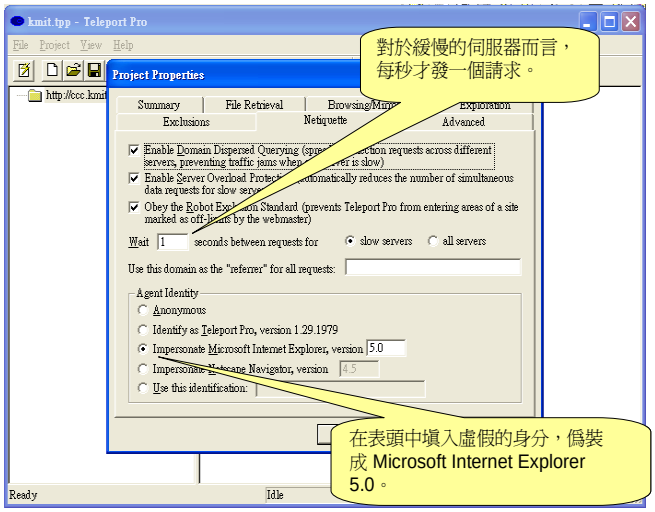

圖 13　Teleport 的設定畫面

[20] Mark A. Lemley etc., “Software and Internet Law”, P948, 2003。

[21] 在本案之後，有許多判決是依照 eBay 相同的理由認爲類似的行爲會構成侵權。例如 Register.com, Inc. v. Verio, Inc., 126 F. Supp.2d 238(S.D.N.Y. 2000)、OysterSoftware, Inc. v. Forms Processing, 2001WL 1736382(N.D. Cal. Dec6, 2001)，但是，在 TicketmasterCorp. v. Ticket.com, Inc, 2000 WL 525390, 2000 U.S. Dist LEXIS 12987, Copy. L. Rep(C.D. Cal., August 10 2000)中，該法官則質疑 eBay 案法官將 trespass to chattel 適用在網站伺服器上的決定，認爲存取對公眾公開的網站，並不對網站造成實質上的侵害，並因而判決原告 Ticketmaster 公司敗訴。

[22] 爲避免造成對方的負擔，間隔時間通常不宜定的太短。

[23] 例如經過本文作者自己的測試，Google 本身就透過辨認身分的方式來拒絕 crawler 透過它進行查詢。

但即使 Crawler 爲了避免干擾到對方的正常運作，每隔一段時間才發送請求，但是這個功能又指出了另外一個問題：多久的間隔，多少的數量會被認爲是侵權？在 eBay 案中，BE 公司每日會發生 8 萬到 10 萬個需求封包，約佔當時 eBay 每日處理量的百分之一左右，這個比例乍看之下很小，但是如果考慮到這是單一使用者所耗費的資源，其實比例應該算是相當高，但是確實很難就比例多少算是侵權規定統一的標準，在不同的案件中，法官可能有不同的考量，例如在 Intel Corp. v. Hamidi[24]中，身爲 Intel 前員工的 Hamidi，在 21 個月的期間內，對 Intel 的 3 萬 5 千名員工散發了 6 封容量相當巨大的郵件，加州最高法院最後以 4 比 3 的比數，認爲此舉並不構成侵害財產權。

雖然很難建立一個特定的標準，但是考量上述的幾個案例，本文以爲，或許可以從考量以下幾個因素來綜合評估是否構成侵權：

1. 網站伺服器的處理能力及平均每日流量：以上面的 eBay 案爲例，eBay 有數百萬的客戶，而 BE 這個單一客戶卻每天都要佔去百分之一的流量，比例上似乎是超過了預期，應該會排擠到其他客戶。
2. Crawler 是否是平均地對每一個網站都發生相同的需求封包，還是僅針對某一個或幾個特定的網站：就算是 crawler 所發出的需求確實會干擾到網站的正常運作，有可能造成這種結果的原因是因爲網站伺服器的使用量過小，此時，就不應該僅歸責於 crawler，而應該考量 crawler 是否是對其他網站也一視同仁的發出這麼多數量的需求封包，如果是一視同仁地發出需求，而其他網站並沒有因而被干擾，此時就有可能認爲 crawler 並沒有特別的侵權故意。
3. Crawler 之目的：如果 crawler 是基於商業目的蒐集網頁，甚至蒐集後的網頁內容可能被提供給原網頁的競爭對手，此時就應該採取比較嚴格的標準。而 Hamid 案的法官也在判決理由中也附帶提到，如果該案傳送的是大量的商業性電子郵件，就有可能會構成侵權。

[24] Intel Corp. v. Hamidi, 30 Cal.4th 1342, 71P.3d 296, 1Cal.Rptr.3d 32 (Cal. 2003)

3.2 儲存

搜尋引擎通常都會將搜尋回來的資料[25]儲存一份以便進行進一步的處理。而這裡的儲存，已經超過著作權法中「暫時性重製」的範圍[26]，因此本文以爲這種行爲已經屬於「重製」。而這些被儲存的資料，有時候會被使用者直接存取，這個功能在 Google 這叫做「庫存頁面」，或是「CACHE」，此時，搜尋引擎還可能會構成著作權法上的「散布」[27]。而依照美國著作權法第 106 條，重製與散布都是著作權人的專屬權利，因此，此處搜尋引擎的重製和散布，是有可能構成侵害著作權的行爲，而庫存頁面的散布並不屬於 crawler 的工作範圍，因此，以下本文將只討論「重製」的部分。

本文以爲，對這個問題，我們要先注意到某些搜尋引擎會將其存取到的網頁內容製作索引後即刪除，雖然這種重製並不完全符合著作權法規定的「暫時性重製」，但是如果考慮到搜尋引擎僅是爲建立索引而必須將一部份的網頁內容存在永久性的儲存媒介中，但實際上如果網頁內容量夠小的時候，根本可以僅儲存在記憶體內，以「暫時性重製」的方式就足夠處理，因此，本文以爲這種因「建立索引需要」而儲存的行爲，如果在建立完索引就立刻刪除時，不論是從擴大「暫時性重製」或是「合理使用」的觀點來看，應該可以認爲並未侵害「重製權」。

但對某些會永久性的儲存或甚至會再以「庫存頁面」將網頁內容散布出去的搜尋引擎而言，

[25] 這裡先將網頁的內容假定爲具有著作權，當然有部分網站的內容並不具有創作高度，因此不具備著作權法保護的要件，例如前述的 Ticketmaster 案中原告的網頁內容即是如此，但搜尋引擎碰到的網頁，應該相當的比例都是具有創作高度而受到著作權法保護。

[26] 羅明通，著作權法 I，頁 414，2002 年。

[27] 美國將網路上之接觸也認爲屬於著作權法上之「散布」，但是在我國則屬於著作權法中之「公開傳輸權」。參見羅明通，著作權法 II，頁 483，2002 年。

則應另外加以考量：首先，應考慮著作權人是否已經明示或默示同意？第二，搜尋引擎建立庫存頁面的目的爲何？是否會影響甚至取代原網頁的地位？亦即即使是侵害到重製權，但是否會有「合理使用」的適用而排除侵權的可能，也有進一步考慮的必要。

就著作權人是否已經同意或是授權他人使用方面，而這裡所謂的同意或授權，當然不單純限制於明示的授權，默示授權也包含在內[28]。網頁連上網際網路，原則上當然應該認爲已經允許其他使用者瀏覽，而依照現階段的瀏覽器技術，都是先將內容暫存在記憶體內而構成了暫時性的「重製」，因此，我們最少可以說，暫時性的重製是網站擁有者，也就是著作權人所允許的[29]，但是這樣的同意是否可以直接推論到著作權人也已經允許永久性的重製，仍然值得懷疑。另一方面，雖然如果網站擁有者不願意搜尋引擎進行重製，則其在現行的”robot exclusion standard”，可以設定爲”nonarchive”，表示 crawler 這個網頁是不願意被重製並製作成庫存頁面的，也就是說，著作權人是有機會來表達不願意被重製的意願，但如果著作人捨此不爲，本文以爲，仍然不應該認爲著作權人已經默示的同意願意搜尋引擎製作成庫存頁面，因爲目前並沒有任何實際數據可以證明設定 “robot exclusion standard”已經形成網際網路界中大家都接受的共通習慣，所以著作權人並無任何義務或習慣來作此表示，當然就不可以將此處的「沈默」認爲是「默示同意」[30]。

另外，依照美國著作權法的規定，即使是未經著作權人的同意也不當然會構成侵害著作權，在美國著作權法中仍有許多除外的規定，而其中與此最有關聯的就是「合理使用」（fair use），關於合理使用的判斷，依照美國著作權法第 107 條的規定，是否合理使用，應該要判斷以下 4 個因素：1.使用的目的 2.著作物的性質 3.使用的比例 4.是否影響原著作的潛在市場或經濟價值。在這 4 個因素中，基於搜尋引擎在網際網路的重要性以及特殊性，本文以爲最重要的判斷因素是第 1 個和第 4 個，也就是「使用目的」以及「是否影響原著作之經濟價值」，分別討論如下：在現在的網路世界中，搜尋引擎的主要目的是在使網路使用者可以更容易的找到每個人需要的資訊，因此，其初始用途並非基於商業的目的[31]，而儲存的目的，也主是在建立索引，因此在使用目的上，基本上應該認爲符合合理使用。另外，在經濟價值的減損上，一般認爲搜尋引擎是可以提高網站的經濟價值的，這可以從優先搜尋需要支付費用這一點明顯的看出。因此，綜合來說，搜尋引擎的儲存，本文以爲，可以認爲是合理使用美國法院在 2006 年初的 Blake A. Field v. Google Inc.[32] 案中，也認定 Google 的 CACHE 屬於「合理使用(fair use)」，不構成著作權侵害。

肆、防止 crawler 方式之分析

4.1 現有之防止 crawler 方式

在現有的技術中，本文以爲，專門用來防止 crawler 的技術大致有以下三個：robot exclusion，crawler 偵測、Captcha，以下分別介紹[33]：

4.1.1 Robot Exclusion[34]

Robot Exclusion Standard (RES) 是目前最普遍的一套標準，其目的在提供網站管理者及網頁

[28] De Forest Radio Tel. & Tel. Co. v. United States, 273 U.S. 236, 241(1927)

[29] 爲了討論簡化起見，這裡我們先將網頁擁有者與著作權人視爲同一人，以下都用著作權人來代表。

[30] 但美國法院在 2006 年的 Blake A. Field v. Google Inc 中，認爲不設定 nonarchive 就已經屬於默示授權，本文對此表示仍然有疑義。見 Blake A. Field v. Google Inc.,(NO.CV-S-04-0413-RCJ-LRL)。

[31] 雖然，搜尋引擎在今天已經逐漸成爲另外一種商業媒體，但是，就搜尋功能的本身而言，本文以爲仍然並不能說這是商業用途，而且也同樣有許多非商業用途的搜尋引擎存在，如專門提供檢索資訊科學論文的 Citeseer(“http://citeseer.ist.psu.edu”)

[32] 同註 29。

[33] 另外，還有以帳號管制也可以防止 crawler，但是本文認爲帳戶的目的不是針對 crawler，而且其主要的功能是在管制使用者權限，所以在此不列入討論。

[34] 參考 Martijn Koster , “A Standard forRobot Exclusion”, 網址: http://www.robotstxt.org/

所有人一套可以限制 crawler 的機制。其又可分爲提供網站管理者使用的 robot exclusion protocol（以下簡稱 REP）與網頁所有人使用的 robot exclusion META tag。前者之指令包含「是否准許 crawler 抓取資料」（Allow，Disallow）；而後者的指令則主要爲「是否准許搜尋引擎之 crawler 將該網頁抓取並建立索引」（Noindex）及「是否准許 crawler 繼續追蹤該網頁上之鏈結以連到其他網頁」(NoFollow)。如下表:

表 1 : Robot Exclusion Standard 的指令與功能

Robot Exclusion Protocol	
放置處	網站根目錄
指令	Allow/Disallow 功能 ： 是否准許 crawler 抓取資料(預設爲允許 Crawler 存取)
指令	??? 功能 ： crawler 抓取資料的週期爲多久是適當的
Robot Exclusion META tag	
放置處	網頁開頭
指令	Noindex 功能 :是否准許搜尋引擎之 crawler 將該網頁抓取並建立索引(預設爲可索引)
指令	Nofollow 功能 ： 是否准許 crawler 繼續追蹤該網頁上之鏈結連到其他網頁(預設爲可追蹤)

4.1.2 自動偵測

Crawler 既然是一種自動程式，其運行就有一定的軌跡可循，因此資訊技術的角度來看，還是可以利用某些技術，例如分析抓取資料的 log 來找出哪些抓取資料的動作是由 crawler 發出的，如果發現是 crawler 的話，就可以考慮採取某些動作來防止[35]。本文前面提到部分 crawler 會在 head 中顯示『agent』一詞，網站管理者或網頁所有人可以利用這個 head 來判斷抓取資料的是否是 crawler，這是一種簡單的自動偵測技術。

4.1.3 Captcha

Captcha 是 "**C**ompletely **A**utomated **P**ublic **T**uring test to tell **C**omputers and **H**umans **A**part" 的縮寫[36]，最初是由卡內基美隆大學所發展出來的，如果從字面上直譯，應該要翻譯爲「用以分辨人類或電腦的自動杜林測試」，看起似乎不容易理解，但是對網際網路有多一點接觸的使用者應該都有碰過，像是申請 MSN messager 時，我們都會碰到系統要求我們輸入一串歪歪斜斜的英文數字或字母，其目的就是確定帳號是由人工親自申請，而非已有人以程式自動申請。這串字母的前提假設是這串字母只有人才能看的懂，自動程式則無法看懂。

4.2 現有防止方式之分析

如果從「是否是網站主動阻絕 crawler」或是必須「網站僅能被動等待 crawler 配合」這一點來看，我們可以將以上的三種方式區分如下表

表 2: 以主動與否區分防止方式

	防止技術
主動	自動偵測、Captcha
被動	Robot exclusion

此時不免讓我們產生一個疑問，既然 Robot exclusion 必須要 crawler 遵守，網站管理者無法主動透過這套機制阻止 crawler 抓取資料，那 Robot exclusion 有什麼價值？爲何還會有這麼網站使用它？本文以爲，Robot exclusion 還是有其功能的，從法律的角度來看，Robot exclusion 最少具備一個非常重要的功能：表達網站管理者與網頁所有人對 crawler 抓取資料的意願。Robot exclusion standard （以下簡稱 RES）的「Allow/Disallow」表明了「是否准許某一個 crawler 來抓取資料」。另外，Robot exclusion

[35] P.-N. Tan and V. Kumar, "Discovery of web robot sessions based on their navigational patterns", Data Mining and Knowledge Discovery, Vol.6(1), P9-35, 2002.

[36] 參考 Wikipedia : http://en.wikipedia.org/wiki/Captcha

META tag 的「Noindex」則表明該網頁是否准許搜尋引擎之 crawler 抓取該網頁並建立索引，而「Nofollow」則是表明網頁所有人是否准許 crawler 繼續循該網頁上之鏈結連到其他網頁，我們可將之整理如下表。

如果更進一步以法律的角度來看表 1 中每一個指令的意思，如果 crawler 違反網站管理者以「Allow/Disallow」明白表示的意思而仍然繼續抓取資料，就可能會構成上面 3.1.1 節所討論的「非法存取」。另外，如果 crawler 違反 Noindex 而仍然抓取資料並製作索引，則可能會侵害到網頁所有人的重製權，如果將之顯示，則侵害散布權，但此時仍然有可能以「合理使用」排除侵權，要視具體個案是否符合「合理使用」的相關規定。但是就「Nofollow」來說，本文以爲，不遵守並不一定會有法律責任，這要區分情形來看，如果所連到的對象是屬於該網頁所有人所有，例如子網頁，則可能就如同違反 Allow/Disallow 一樣構成非法存取，但是如果該鏈結對象根本與該網頁所有人無關，則不會構成任何法律問題。

伍、小結及未來研究方向

搜尋引擎在現在的網路中日益重要，因爲唯有透過搜尋引擎，才能夠使得網路使用者在眾多的網站中找到個人所需要的資料。而提供這種服務的搜尋引擎，其最主要的部分之一就是要透過 crawler 程式來定期取得各網站的網頁內容，才能夠快速地搜尋。

本文將 crawler 在網路上取得網頁內容的過程，依照其功能及先後關係，分爲「取得資料」與「儲存並建立索引」二大基本步驟。就「取得資料」而言，深度 crawler 因爲存取時會自行計算出所有可能的 URL，而這些 URL 是網頁並未直接提供出來的，但是，本文以爲，URL 仍然是一般網頁瀏覽者透過正常方式而可能取得的，crawler 只是以自動的方式產生並存取，所以我們不能說深度 crawler 超過存取權限，因此，並不會發生無權限存取的爭議。但是，正因爲深度 crawler 是透過自動產生的，而這些封包大量的傳輸到網頁伺服器後，可能會影響到伺服器的正常運作，這也就是 eBay 案的產生背景，在該案中，法官肯認深度 crawler 所發生出的大量封包確實可能會干擾到網站的正常運作，因此，在目前的網路運作實務上，不論是一般的 crawler 或是深度 crawler，都會自我節制所傳出封包的數量以及發出的時間，以盡量避免干擾到網站伺服器的運作。

另外，在「儲存」的過程，如 Google 類提供的庫存頁面（CACHE），實際上已經涉及著作權法上的重製，但是，基於考量 Google 類的搜尋引擎功能的本質，本文以爲，這裡的重製行爲應該仍有「合理使用」的適用，故不構成侵害著作權。

其實，搜尋引擎是網路世界的一項非常重要的工具，但是，也由於這項工具的日益重要，在它上面也產生了許多商機，進而衍生出了許多網站與搜尋引擎間的利益衝突，網站方面有 robot exclusion，crawler 偵測、captcha 等三種防止 crawler 的方式，其效果各有不同，已如前述，但目前實務上最常見的 robot exclusion，要發揮真正的防止效果，本文以爲，還是必須依靠法律的強制力。

而法律雖然是調和這些衝突的一個重要方式，但是這也只是一個選項而已，在技術面上，這方面能夠著墨的地方還有很多，例如雖然上述 Blake A. Field v. Google 案中法官以合理使用的觀點認爲 Google 的 CACHE 並未侵害著作權，但是，Google 如果能夠在使用者要求閱覽 CACHE 內容時，可以多加一些判斷的條件，例如先確認資料來源網站是否能傳回該原始資料，若可以、則將使用者導到該網站上，若來源網站暫時無法傳回資料，才啓動 CACHE 的功能，傳回 CACHE 的資料，如此、則更可以避免這方面的爭議。另一方面，對於許多被搜尋的網站來說，目前除了採取 Robot Exclusion Standard 外，幾乎都沒有針對 Crawler 作任何主動的防護，但法律應該是網路世界中的最後一道防線，如果網站經營者能夠在網站伺服器或防火牆的設置上多加一些防護

的功能[37]，或許可以將著作權侵害與財產權侵害的問題減少，也更可以避免隨之而來的法律爭議與訴訟。另外，現行的 robot exclusion 過於簡略，就法律的觀點來看，還有不少必須要補充的地方，一個更完整的 robot exclusion，也是值得努力的方向。

參考文獻

1. 李曉明、閻宏飛、王繼民, “搜尋引擎 – 原理、技術與系統”, 科學出版社, 2004 (簡體)
2. 羅明通，著作權法論 I、II，台北，台英國際商務法律事務所，2002 年第 4 版
3. David Fox,Tory Downing 著，江永祥、廖先志譯，深入 HTML3 WEB 設計，松格資訊有限公司，1995，頁 14。
4. 林發立，” Internet 的優勢與問題--從 TicketmasterCorp. v. Tickets.com Inc.「深入連結」一案談起”， 萬國法律，第 112 期，頁 49-52，2000 年。
5. Baeaz-Yates R, Riberiro-Neto B. “Modern Information Retrieval”, Addison Wesley, Longman, 1999.
6. Mark A. Lemley etc., “Software and Internet Law”, Aspen Publishers, 2003。
7. Kevin Emerson Collins, ”Cybertrespass and Trepass to Documents”, Clev. St. L. Rev., Vol. 54, P41-66, 2006。
8. Michael Hatcher, Jay McDonnell & Stacy Ostfeld, ComputerCrimes, 36 Am. Crim. L. Rev.,at 399-402 (summer, 1999).
9. Michael K. Bergman, "The Deep Web: Surfacing Hidden Value". The Journal of ElectronicPublishing 7 (1), 2001.
10. Pamela Samuelson, “Unsolicited Communications as Trespass?”, Comm. ACM, Vol. 46 No. 10, P15-20, Oct. 2003。
11. P.-N. Tan and V. Kumar, “Discovery of web robot sessions based on their navigational patterns”, Data Mining and Knowledge Discovery, Vol.6(1), P9-35, 2002.
12. Anália G. Lourenço, Orlando O. Belo, “Catching web crawlers in the act**”,** Proceedings of the 6th international conference on Web engineering (ICWE), P265-272, 2006。
13. Alexandros Ntoulas, Petros Zerfos, Junghoo Cho, "Downloading Textual Hidden Web Content by Keyword Queries"", Proceedings of the Joint Conference on Digital Libraries (JCDL), p100-109, June 2005。
14. Brin S, Page L. “The anatomy of large-scale hypertextual Web search engine. In Proceedings of the 7th International World Wide Web conference/ComputerNetworks, Amsterdam, 1998.
15. Sriram Raghavan and HectorGarcia-Molina "Crawling the Hidden Web". In Proceedings of the 27th International Conference on Very Large Data Bases (VLDB), 129-138, 2001.
16. Page L, et al. “The PageRank Citation Ranking : Brining Orderto the Web”, Stanford Digital Library Technologies Project, 1998.

[37] Anália G. Lourenço, Orlando O. Belo, **“**Catching web crawlers in the act**”,** Proceedings of the 6th international conference on Web engineering (ICWE), P265-272, 2006。P.-N. Tan and V. Kumar, “Discovery of web robot sessions based on their navigational patterns”, Data Mining and Knowledge Discovery, Vol.6(1), P9-35, 2002.