

國立交通大學

科技法律研究所

碩士論文

開放原始碼模式下的法律糾紛與風險

-- 從 SCO v. IBM 案出發

Legal Issues in Open Source Model

-- *starting from SCO v. IBM*

研究 生：李界昇

指 導 教 授：王明禮 博士

中華民國九十三年七月

開放原始碼模式下的法律糾紛與風險

-- 從 **SCO v. IBM** 案出發

Legal Issues in Open Source Model

-- *starting from SCO v. IBM*

研究 生：李界昇

Student: Jieh-Sheng Lee

指 導 教 授：王明禮

Advisor: Dr. Ming-Li Wang



A Thesis
Submitted to Institute of Technology Law

College of Management

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

in

Technology Law

July 2004

Hsinchu, Taiwan, Republic of China

中 華 民 國 九 十 三 年 七 月

開放原始碼模式下的法律糾紛與風險

研究生：李界昇

指導教授：王明禮

國立交通大學 科技法律研究所 碩士班

摘要

開放原始碼運動為軟體產業開創新的創作模式，在開放原始碼模式下的軟體持續增加，累積的成果有目共睹。但開放原始碼日益普及的背後並非沒有任何隱憂，尤其智慧財產權方面的法律議題，更關係著開放原始碼模式未來的發展。2003年SCO控告IBM侵權案，夾著可觀的商業利益及複雜的訴訟關係，並直接挑戰開放原始碼模式的法律基礎，正突顯了相關議題的潛在破壞力，也因而受到高度重視。另外，SCO後續衍生出幾個相關案件，這些案件加上前者便成了本文探討開放模式下法律糾紛的切入點。

從訴訟中各方紛陳的意見及盤根錯節的事實，本文進一步探討開放原始碼模式下其他潛在的風險。其中可以發現，幾個重要的考量多半與開放原始碼獨特的公眾授權模式有關，並且隨著開放原始碼的普及，瞭解公眾授權模式也變得比以前更加重要。針對已知的風險類型以及近年來智慧財產權制度的演變，有識之士指出軟體專利可能成為開放原始碼模式未來最大的挑戰，而值得研究。綜合各方對專利制度的批評與建議，加上開放原始碼模式的啟發，本文嘗試提出創設「公益專利」的想法。藉由運作機制的勾勒及可能利弊的初步分析，希望不但能為開放原始碼軟體營造一個更健全的制度環境，也能為專利制度的改革提供一個新的思考面向。

Creative Commons的成立與茁壯，將開放模式的典範延伸至電腦軟體以外的領域，給關心公眾創作領域者新的期待。受其啟發，本文初步提出「公眾智慧財產權」的概念，以同時涵蓋開放原始碼、公益專利及Creative Commons三者，希望藉此拋磚引玉，將來有機會發展出一以貫之的理論架構。最後，就產業面，則希望台灣廠商能善用各種開放原始碼軟體，強化既有以硬體為主的市場競爭力，並留意開放原始碼模式下特有的相關法律議題。

Legal Issues in Open Source Model

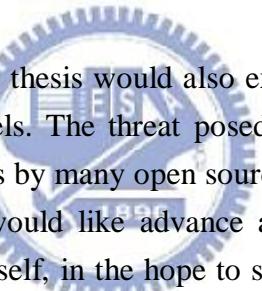
Student: Jieh-Sheng Lee

Advisor: Dr. Ming-Li Wang

Institute of Technology Law
National Chiao Tung University

ABSTRACT

The open-source movement presents the world a new software development model. Its continuing success can be attested by the exponential growth of software projects under its wings. Pitfalls and challenges loom ahead, however. Intellectual property-related legal issues, in particular, deserve our special attention for their disrupting potential, of which the recent *SCO v. IBM* case and its surrounding legal entanglements serve as a timely reminder. Hence an in-depth analysis of important issues raised by the cases should be a good starting point for this thesis.



Going beyond the SCO cases, this thesis would also explore other risks associated with the various open-source licencing models. The threat posed by software patents stands out during the analysis, echoing the concerns by many open source pundits. After reviewing other proposed solutions so far, the author would like to advance a new one—the “public patent” mechanism inspired by the movement itself, in the hope to strengthen the long-term viability of the open source movement, and add more fuel to the big debate of patent system reform.

The thesis would also touch on the Creative Commons and its likes that seek to extend applications of the open-source model to intellectual innovations beyond computer software. A primitive idea called “public intellectual property” is suggested to potentially better rationalize the assortment of “open” innovations—open-source software, public patents and the Creative Commons—in a unified theory. Sketchy as it is, the idea is thrown out as a stepping stone of further brainstorming.

In closing, the thesis would outline a few suggestions catered to Taiwanese IT industries, which might benefit from incorporating open source software into their development strategies.

誌 謝

有幸在工作數年後暫時停下腳步，回到學校專心完成一篇碩士論文，要感謝周遭許多人的照顧與支持。首先，要感謝指導老師王明禮教授的提攜與栽培，從論文研究的方向、重要的參考資料到許多細節的釐清，王明禮老師都能給予熱心的指導，並提供寶貴而專業的正反面意見。王明禮老師所不知道的是，其實在論文指導以外，學生從老師的課堂上也受益良多，單從老師對許多議題的反覆辯證中，便能在法學思考上收到醍醐灌頂之效。如果論文寫作過程中所遇到的瓶頸能有些突破的話，首先應歸功於老師身教的激勵與思維的啟發。

其次，要感謝口試委員劉靜怡教授及王敏銓教授，兩位老師在口試中提出的問題與建議，讓學生有機會把論文的嚴謹度再往上提升，並強化部分論述讓重點更加突顯。另外，學生有機會帶著理工的背景跨入科技法律領域進修，並從所上良好的學習環境、紮實的課程設計及優秀的老師們身上受益匪淺，更需要感謝所長劉尚志教授高瞻遠矚設立科技法律研究所，讓莘莘學子們有機會朝跨領域發展。

最後，感謝妻子在過去三年來的體諒與鼓勵，讓我得以在工作、課業與家庭生活三者間保持平衡而無後顧之憂，彼此在互相扶持及關懷的歲月裡，讓我感受到生命中幸福的一面，願意面對各種挑戰一同開創美好的未來。至於為什麼希望跨領域繼續進修，我想激勵自己不斷努力的是父母勤勉刻苦的身影，上一代在困頓的環境下為子女所犧牲奉獻的，大概是自己一輩子也報答不完的。

目 錄

中文摘要	i
英文摘要	ii
誌謝	iii
目錄	iv
圖表目錄	viii
第一章 緒論	1
第一節 研究背景與動機	1
第二節 研究方法與步驟	2
第三節 研究範圍與限制	3
第四節 研究章節與架構	3
第五節 開放原始碼簡介	4
第一項 何謂原始碼	4
第二項 如何開放	5
第三項 開放原始碼的重要性	9
第二章 SCO v. IBM 及相關案例	11
第一節 案情簡介	11
第二節 深入分析	24
第一項 SCO 本身涉嫌違反 GPL 授權條款的規定	24
第二項 SCO 未能釐清原始碼的爭議範圍	27
第三項 原始碼相同未必等於著作權之侵害	29
第四項 SCO 質疑開放原始碼社群控管原始碼的嚴謹度	32
第三節 開放原始碼社群的立場與回應	33

第一項 自由軟體基金會	33
第二項 開放原始碼協會	34
第三項 USENIX 協會	38
第四項 群策群力、抽絲剝繭	39
第五項 撤銷 SCO 的散布權	39
第六項 補償措施	40
第四節 其他國家之案例	41
第一項 德國	41
第二項 澳洲	42
第五節 小結	43
第三章 開放原始碼模式與法律風險	44
第一節 侵害著作權	44
第二節 授權方式的複雜度	49
第三節 授權合約的效力	58
第四節 擔保條款	62
第五節 他人違反公眾授權	66
第六節 商標	69
第七節 軟體專利	72
第八節 其他觀點	74
第一項 反方陣營的觀點	74
第二項 開放原始碼運動的障礙	80
第三項 意識型態	81
第四項 資訊科技的改變	82
第九節 小結	83
第四章 軟體專利制度的改革	84

第一節 軟體專利的演進.....	84
第二節 軟體專利引發的爭議或批評	87
第一項 軟體專利存在的必要性	88
第二項 對習知技藝人士的誤解	90
第三項 專利制度是否優於著作權制度	92
第四項 專利制度因時制宜的必要性.....	93
第五項 專利文件對程式設計師的助益.....	93
第三節 開放原始碼模式與軟體專利	94
第一項 軟體專利的威脅是否存在	94
第二項 軟體專利是否涵蓋原始碼的使用	96
第三項 專利制度下的差別待遇	99
第四節 制度改革的建議.....	100
第一項 暫停或取消軟體專利	100
第二項 提升軟體專利的品質	101
第三項 縮短專利保護期限	102
第四項 專利的合理使用原則	103
第五項 降低訴訟的誘因	105
第五節 建議作法	106
第一項 理念	106
第二項 開放原始碼模式的啟發	108
第三項 公益專利庫的可能性	112
第四項 公益專利庫如何運作	114
第五項 公益專利庫的利弊	116
第五章 展望未來	118
第一節 開放原始碼模式的前景	118

第一項 軟體應用面的擴大.....	118
第二項 Creative Commons	119
第二節 公眾智慧財產權.....	122
第三節 台灣產業的機會.....	125
第六章 結論	127
參考文獻	129



圖 表 目 錄

圖一 原始碼、目的碼及函式庫的關係	5
圖二 GPL 與 LGPL 的比較	8
圖三 SCO 相關案件之關係	18
圖四 雙軌授權模式	50
圖五 著作權保護與促進公益、私利的關係	109
圖六 軟體缺乏著作權保護的缺點	110
圖七 公眾授權模式與促進公益、私利的關係	110
圖八 軟體專利的潛在威脅	111
圖九 目前專利制度與促進公益或私利的關係	112
圖十 軟體放棄專利權保護的缺點	113
圖十一 公益專利模式與現行專利制度的關係	114
圖十二 Creative Commons 與促進公益、私利的關係	122
表一 SCO 相關案件中訴訟爭點及背景事實之整理	24
表二 常見公眾授權間之相容關係	55

第一章 緒論

第一節 研究背景與動機

開放原始碼運動(Open Source Movement)近年來如火如荼般展開，影響範圍日益擴大，不僅為軟體產業開創新的創作模式，創作成果也備受肯定，得以與傳統專屬軟體一較高下。近年來網際網路蓬勃發展，開放原始碼軟體功不可沒，網路上許多重要的功能與架構，都可以看到開放原始碼軟體的身影，對一般消費者，開放原始碼模式下產生的 Linux 系統，也逐漸成為另外一種選擇。開放原始碼運動在落實無償分享的同時，在商業應用方面也發展出各種模式，讓不同的公司有可能以適合自己的方式獲利。許多國家或機構，則著手將開放原始碼列為公共政策的考量重點之一，以解決資訊化引發的許多問題。

不過，在開放原始碼運動蓬勃發展的背後，並非沒有任何挑戰或隱憂。相對於目前常見的專屬軟體，開放原始碼仍有許多技術上或商業上的障礙需要克服，這些障礙透過資訊技術的進步或市場機制的自由競爭，應該可以逐步跨越，讓開放原始碼運動持續下去。但在技術與商業之外，法律議題卻不容忽視，尤其是智慧財產權方面的議題，可能成為開放原始碼模式能否持續發展的重要關鍵。例如，開放原始碼的「公眾授權」(public licensing) 模式與傳統授權契約的觀點不同，授權模式是否具有效力？違反授權條款時該如何處理？不同的授權間彼此的法律關係為何？這些都是亟待釐清的問題。

2003 年發生的 SCO v. IBM¹一案，正好凸顯了法律議題對開放原始碼所產生的衝擊。過去在開放原始碼運動中並非沒有出現任何法律糾紛，但就發生的次數及規模來說，應可視為零星的個案而未受到廣泛的注意。唯獨 SCO 控告 IBM 一案，因夾雜著可觀的商業利益且訴訟關係複雜，原告並採取擴大爭議的策略將開放原始碼模式下的主流系統 Linux 牽涉其中，因此受到資訊界的高度重視。簡略地說，在該案中 SCO 指控 IBM

¹ See Caldera Systems, Inc. v. International Business Machines Corp., No. 2:03cv0294 (D. Utah complaint filed March 6, 2003), available at <http://www.sco.com/ibmlawsuit> (visited Jul. 23, 2004).

違反合約，故意將 SCO 所屬的程式碼流入開放原始碼社群，SCO 並針對開放原始碼特有的公眾授權模式加以挑戰，認為公眾授權缺乏強制力，再加上其他的諸多指控，SCO 案已明顯對開放原始碼運動構成過去所沒有的威脅，而為開放原始碼社群所無法忽略。

開放原始碼運動長期發展下來，已達到一定的成績，預期未來將有更多的廠商投入其中，以智慧財產權常常成為商業競爭策略的一環，將來引發的爭議可能有增無減。例如，開放原始碼模式如果對專屬軟體模式的廠商威脅過大，專屬軟體廠商是否可能透過現行法制中對開放模式比較不利的部分，壓制開放原始碼運動的進展，以維持專屬模式下的既得利益？還有，近幾年軟體專利的快速演變，引發許多爭議，對開放原始碼運動是否可能造成不利的影響？對於關心開放原始碼運動的人來說，這些都是值得以法律觀點加以研究的課題。

第二節 研究方法與步驟

本文在研究方法上以文獻整理、實務分析與抽象思考為主，在研究步驟上則從實際案例切入，找出相關文獻加以研讀，並思考將實務運作中的上位概念抽離出來適用到其他法律客體的可能性。在實務方面，本文透過 SCO 引發的相關案例，從爭點中分析開放原始碼模式所遇到的問題，並嘗試整理開放原始碼社群涉及的其他法律糾紛，探討開放模式潛在的風險。開放原始碼模式下的風險逐項加以釐清後，針對未來潛在的最大問題則另以專章討論，以進一步研究法律制度面調整或改革的可能性。在抽象思考方面，本文將「開放模式」視為開放原始碼模式的上位概念，發現跨出資訊領域以外「開放模式」已逐漸應用到其他創作領域。至於蒐集相關文獻或資料，除了使用國內外常見的期刊、論文、判例等資料庫外，訂閱相關的電子報持續觀察開放原始碼社群的動態並追蹤訴訟案的進行，也有助於更新案情並彙整網路上的各種意見。在資料的整理上，本文採取「爭點探討」(argument outline)的方式分析問題，輔以「心智圖法」(mind mapping)²的

² 心智圖法於 1970 年代初期由英國學者 Tony Buzan 所創，是一種將放射性思考具體化的方法，相關細節可參見 <http://www.mindmapping.com.tw>.

建構方式逐步構思、調整論文綱要，並透過 FreeMind³軟體工具加以落實。

第三節 研究範圍與限制

本文研究範圍以美國案例與文獻為主，原因有二，一是資訊產業在美國蓬勃發展，相關法律及判例較多，就智慧財產權領域，美國在國際上亦扮演主導角色，二是科技與法律之間的互動與演變，美國學術界提出數量可觀的研究成果，得以成為進一步探討的資料來源。本文希望在既有的文獻中，整理相關論點並針對部分課題提出一些建議。考量科技與法律關連的範圍廣泛，為避免失之龐雜，除了 SCO 案之外，本文重心將以著作權、專利、公眾授權及其他智慧財產權領域為主，對部分實務案例的整理，則希望有助於產業界瞭解開放原始碼模式，並在考量是否採用開放原始碼軟體時，多一點法律方面的觀點。

本文介紹的相關案例，由於部分的訴訟仍在進行，因此研究上難免受到一些限制，分析的結果未來也可能因為加入新的變數，而有修正的可能。尤其 SCO 一案特別複雜，案情持續變化，高度不確定性增加研究的難度及對議題論斷過早的風險，雖然如此，本文仍試圖在已知的法律事實與不同觀點中，找出對開放原始碼模式重要的議題與建議，並在逐步分析的過程中，形成自己的立場與見解。

第四節 研究章節與架構

本文論述的主軸從 SCO 案切入，探討該案與開放原始碼模式相關之處後，進一步擴大分析與開放原始碼模式相關的其他糾紛與風險。針對未來軟體專利可能成為主要的潛在風險，則研究其近年來演變與法制面改革的可能。最後，將開放模式擴大到原始碼以外，對未來智慧財產權領域的可能轉變提出一點觀察。

在章節的安排上，第一章簡要介紹開放原始碼運動，解釋相關專有名詞，指出議題

³ FreeMind 為一開放原始碼軟體，可以從 <http://sourceforge.net/projects/freemind> 取得。

的重要性，為進入後續章節做準備。第二章進入複雜的 SCO v. IBM 案，分析雙方爭點，列出重要事件，並追溯開放原始碼模式的演變與細節，做為法律上的事實基礎。在各方互相質疑、批評、聲援或回應的過程中，許多議題將一一浮現。第三章從開放原始碼模式的角度，整理該模式下相關的法律風險，以及相關的其他法律糾紛。規模較小或時間較早的部分案例，如有助於評估開放原始碼模式的風險，則一併加以探討。第四章專門討論軟體專利，軟體專利制度如何演變？為何爭議不斷？軟體專利對開放原始碼模式產生何種影響？負面影響是否需要制度面的改革？智慧財產權制度如何在公益與私利間取得平衡？第五章展望未來，以不同的面向觀察開放模式的延伸，以及實務上正在發生的趨勢，並對台灣產業可能的機會提出一點建議。第六章總結論文要點。

第五節 開放原始碼簡介

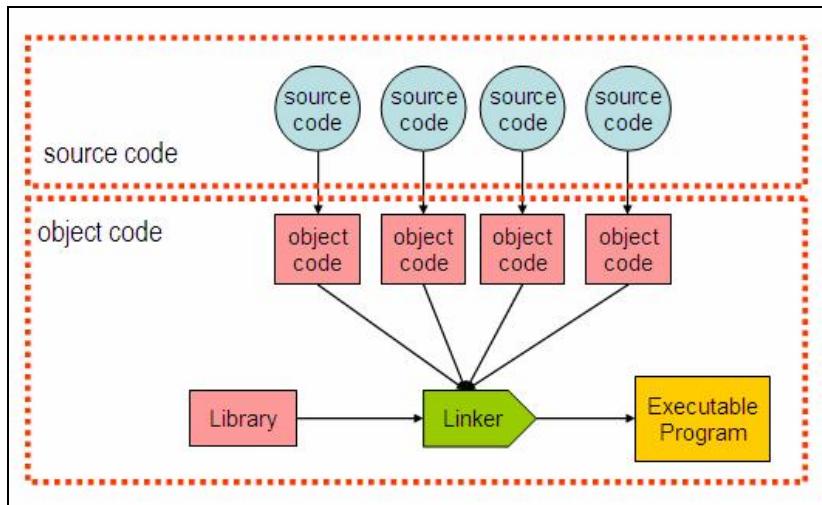
開放原始碼中可探討的議題不少，為避免失之龐雜，在此擇要從底下幾個角度切入，做為簡介。



第一項 何謂原始碼

原始碼(source code)是程式設計師創作的成果，形式上一般以人工可閱讀的高階電腦語言所組成，所謂「寫程式」，即是設計師以特定語言描述電腦執行功能的創作過程。相對於原始碼的是「目的碼」(object code)，目的碼是電腦硬體據以執行的指令，形式上以晶片可讀取的二元碼所組成，非人工所能閱讀，目的碼透過「編譯」(compile)的步驟產生，不同的電腦語言透過不同的編譯器(compiler)將原始碼編譯轉換成目的碼。軟體的改良原則上必須透過原始碼的修改才能達成，因此，是否取得原始碼便成為能否修改軟體的關鍵。

根據複雜度的不同，一個軟體的原始碼可能由單一個檔案到上百個檔案所組成，甚至更多，軟體中的細部功能往往也不是全部自己開發，而是藉助外部的函式庫(library)，函式庫通常同樣以目的碼形式存在，由其他人或底層的作業系統所提供之。原始碼、目的碼及函式庫彼此之間的關係可由下圖表示：



圖一 原始碼、目的碼及函式庫的關係⁴

透過圖示，一則瞭解資訊技術的細節，一則有助於後續探討開放原始碼的法律議題。例如，可執行的軟體通常由數個原始碼構成，個別的原始碼可能採取開放模式或專屬模式，或者具備不同的授權條款，各個原始碼的所有人可能並不相同，而且原始碼編譯成目的碼之後，可能採取不同的授權方式供他人使用，加上自己的軟體如何與他人提供的函式庫連結，必須受到函式庫本身授權條款的限制等。以這些問題為例，如果對應到圖示本身，基本上可以視為圖中的每個物件依據所有權人及授權條款的不同，可能與其他物件形成不同的法律關係。將不同關係或狀態的組合套到圖上，開放原始碼下法律議題的複雜度將跟著增加，除此之外，實務上也會衍生出一些需要探討的問題。例如，原始碼如果公開，應該透過何種機制確保達到公開的目的？公開後是否允許他人修改、散布衍生的原始碼？允許他人修改、散布時，可否要求他人繼續公開衍生的創作？如果原始碼並非同一人所有，可否合併？各個原始碼的授權條款相異時，該如何處裡？使用他人開放的目的碼，是否自己的原始碼必須跟著公開？類似這些問題，如果事先能對原始碼與目的碼之間的關係具備一定的瞭解，將有助於解決技術實務上真正碰到的法律問題。

第二項 如何開放

⁴ 關係圖參考 <http://www.webopedia.com/TERM/L/link.html> 而來。

就字義解釋，「開放」表示揭露原始碼之意。一般傳統專屬軟體模式，原始碼保持在廠商手中，以著作權、營業秘密的方式保護，消費者拿到的是可執行的目的碼。所謂購買軟體，消費者取得的是使用軟體的授權，授權標的多半不包括取得原始碼在內。取得原始碼等於取得軟體廠商大部分的智慧財產，對軟體廠商有害而無利，一般無法閱讀高階電腦語言的使用者，多了原始碼也沒有太大用處。因此，專屬軟體模式多半等同於「封閉」原始碼模式。

對於封閉原始碼模式，部分程式設計師有不同的想法，認為封閉的模式應該打破，程式設計師如能自由閱讀、修改、散布自己或他人的原始碼，將可促進資訊科技的進步。開放原始碼運動在這樣的精神下，出現許多程式設計師及團體參與其中，與特別重要的首推「自由軟體基金會」(Free Software Foundation, FSF)⁵與「開放原始碼協會」(Open Source Initiative, OSI)⁶。本文所謂「開放原始碼軟體」，簡言之，可以視為符合自由軟體基金會定義的「自由軟體」(Free Software)或符合開放原始碼協會標準的軟體，兩個單位在理念上雖有不同，但重疊相容之處較多，無礙於本文研究，故等同看待⁷。

對自由軟體基金會來說，自由軟體具備幾項特徵，其中重要值得提出者如：軟體散布時必須揭露原始碼、不得對原始碼本身收費等。為確保自由軟體的延續，自由軟體基金會採取特殊的 GPL (General Public License)公眾授權⁸，其中規定對自由軟體，使用者可自行複製、修改、散布其原始碼，但必須附上並遵守 GPL 授權條款的規定。散布衍生著作時，必須採用相同的 GPL 條款再公開原始碼，除非衍生著作只供自己使用，則無需公開原始碼。如果違反 GPL 規定，將自動喪失 GPL 條款賦予的權利。因為使用者不需要簽名，使用者可以選擇接受或不接受 GPL 條款及 GPL 軟體，如果使用者選擇修改或散布了自由軟體，則表示同意接受 GPL 條款的規範。另外，由於程式設計師無償

⁵ See Free Software Foundation, at <http://www.fsf.org> (visited May. 3, 2004).

⁶ See Open Source Initiative, at <http://www.opensource.org> (visited May 3, 2004).

⁷ 理念上的不同仍造成負面影響，有礙於開放原始碼運動的推展。See generally Teresa Hill, *Fragmenting the Copyleft Movement: The Public Will Not Prevail*, 1999 UTAH L. REV. 797 (1999).

⁸ See GNU, *General Public License*, at <http://www.gnu.org/licenses/licenses.html#GPL> (visited May 7, 2004).

提供原始碼，在 GPL 授權下不提供任何擔保，所有風險由使用者自負。

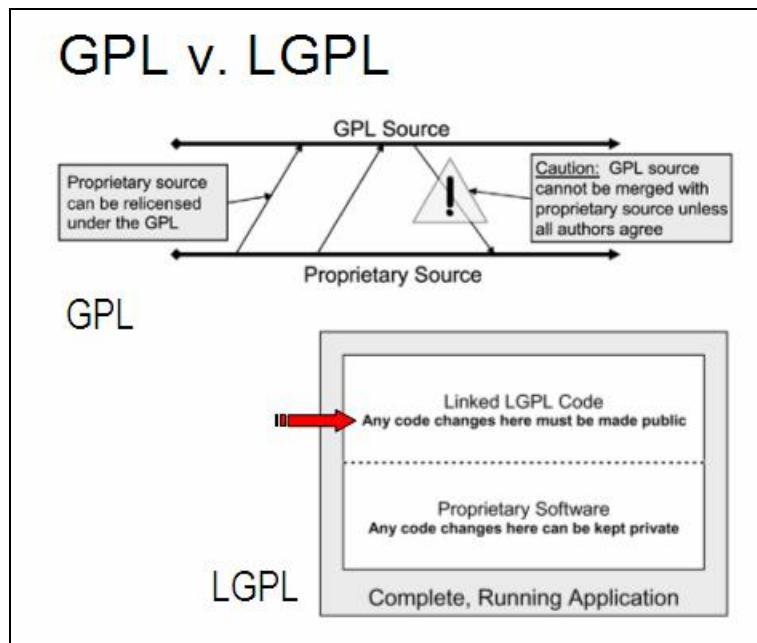
圖 1-1 提到函式庫的使用，在 GPL 的規定下，如果函式庫以 GPL 條款授權，其他專屬軟體連結到函式庫時，也構成軟體的使用，必須跟著開放其原始碼。這樣的規定對許多軟體來說，可能過於嚴苛，因此，自由軟體基金會提出強制性較弱的 LGPL (Lesser GPL)⁹。以軟體的目的，大部分函式庫努力的目標是增加使用的人數，以擴大其影響力，影響力越大，越可能成為業界的典範或標準，通常，如果函式庫位於系統架構較為基礎的層次，受到其他軟體使用的可能性也相對增加。在這些特性下，如果開放原始碼的函式庫要求相關者一律開放，反而可能劃地自限，降低專屬軟體採用的意願，因此，LGPL 的提出，剛好解決這個問題。

根據跨平台軟體最大集散地 freshmeat 網站的統計¹⁰，其所公布的各類軟體中，GPL 佔有率約為百分之七十，LGPL 排名第二，佔有率略超過百分之五，其餘則由數十種不同的授權模式瓜分，由此可以看出 GPL 與 LGPL 在開放原始碼議題中的代表性。GPL 與 LGPL 的差異，另外可參考下圖。



⁹ See GNU, *Lesser General Public License*, at <http://www.gnu.org/licenses/licenses.html#LGPL> (visited May 7, 2004).

¹⁰ See Freshmeat, *License breakdown*, Nov. 2003, at <http://www.dwheeler.com/frozen/freshmeat-stats-20031110.html> (visited May 3, 2004).



圖二 GPL 與 LGPL 的比較¹¹

所謂開放原始碼，另外可以從「開放原始碼定義」(Open Source Definition)¹²加以瞭解。開放原始碼協會創立於自由軟體基金會之後，之所以另闢蹊徑，主要用意在於透過重新定義，吸引並廣納包含商業團體在內的各方人士，一同在傳統專屬軟體模式外開創新的可能。該協會採用開放原始碼運動做為整個產業模式轉變的代名詞，並提出「開放原始碼定義」，定義中包含十項標準，凡公眾授權模式符合全部規定者，即可稱為開放原始碼授權模式，目前符合者有五十二種授權¹³，在這些授權條款下公開的軟體則稱為開放原始碼軟體。其十項定義分別為：(1)免費散布軟體，(2)要求散布原始碼，(3)衍生著作採原著作的相同授權，(4)維持原作者的人格權與著作完整性，(5)不得歧視任何團體或個人，(6)不得歧視任何領域，(7)授權本身足以賦予所述之權利，(8)授權不得針對特定產品，(9)授權不得限制其他軟體，(10)授權必須技術中立。其中第一、二、三項與本文研究相關，與自由軟體的規範也類似，事實上，GPL 公眾授權符合上述開放原始碼

¹¹ MARTIN FINK, THE BUSINESS AND ECONOMICS OF LINUX AND OPEN SOURCE 46-47 (2002).

¹² See OSI, *The Open Source Definition Version 1.9*, at <http://www.opensource.org/docs/definition.php> (visited May 3, 2004).

¹³ See OSI, *The Approved Licenses*, at <http://www.opensource.org/licenses> (visited May 3, 2004).

定義下的規定，因此，GPL 授權的自由軟體亦可視為開放原始碼軟體。

第三項 開放原始碼的重要性

從統計數字中¹⁴，可以觀察到開放原始碼的佔有率不斷提升，這一點除了反應開放原始碼影響的層面擴大外，在開放原始碼模式的重要性中有兩點值得提出。首先是軟體的安全性與穩定性；開放社群認為：「對原始碼注視的目光如果夠多，臭蟲將無所遁形。」

¹⁵臭蟲是軟體的瑕疵，通常是程式設計師的無心之過，或是軟體複雜度升高後不可避免的副產品。雖是無心之過，後果可能十分嚴重，例如，1996 年六月，歐洲太空總署發射無人巨型火箭 Ariane 5，升空後不到四十秒爆炸。該火箭造價七十億美金，歷時十年，爆炸原因是火箭內部軟體誤將一個 64 位元數值塞到 16 位元資料，結果釀成大禍¹⁶。

網路普及之後，電腦病毒與木馬程式逐漸取代臭蟲，成為資訊安全的主要威脅。例如，2004 年五月，網路病毒 Sasser 蔓延並波及國內郵政系統，全國三分之一郵局的電腦受害，儲匯繳款作業全面停擺，據估計，該病毒在全球造成百萬電腦中毒¹⁷。無獨有偶，約於同一時間，刑事警察局偵九隊經過相當期間的監控後公開表示，國內資訊安全已出現大規模的「木馬屠城記」，並呼籲企業與民眾自行檢查電腦是否遭到駭客入侵。據估計，總共約有上百家企業受害，其中包括四家以上網路銀行遭盜領，金額達二、三千萬，不少高科技業者則有機密資料外流之虞¹⁸。不論軟體的瑕疵是有心還是無意，開

¹⁴ See David A. Wheeler, *Why Open Source Software/Free Software (OSS/FS)?*, at http://www.dwheeler.com/oss_fs_why.html (visited May 4, 2004).

¹⁵ 由開放原始碼協會(Open Source Initiative)創始人 Eric Raymond 所提出之概念：「Given enough eyeballs, all bugs are shallow.」。See Eric S. Raymond, *The Cathedral and the Bazaar*, at http://www.firstmonday.dk/issues/issue3_3/raymond (visited June 20, 2004).

¹⁶ See James Gleick, *A bug and a crash*, Dec. 1, 1996, available at <http://www.around.com/ariane.html> (visited May 4, 2004).

¹⁷ 該病毒針對微軟公布的 MS04-011 漏洞攻擊，是有史以來漏洞公布後最快出現的病毒，未更新系統者幾乎難以倖免。有人認為，微軟定期公布系統弱點，如同燈塔般向駭客指出自己要害。參見 93 年 5 月 4 日中國時報、聯合報。

¹⁸ 參見 93 年 5 月 4 日工商時報。

放模式如同陽光法案，讓原始碼公諸於世，將有助於提高軟體的穩定性與安全性。原始碼透明度提高，蓄意遭植入不當功能的機會將下降，即使發生，相關社群也能發揮把關的效果¹⁹。相對之下，傳統專屬軟體則一切仰賴原廠的品管與服務。

其次，開放原始碼有助於數位鴻溝的跨越。除了取得原始碼沒有成本外，開發中國家投入開放原始碼運動，將有機會站在巨人的肩膀上，提升自己的資訊技術。每年購買昂貴的專屬軟體²⁰或支付廠商強迫升級的費用，則可投入研發或基礎建設，以便讓技術生根。目前約有 25 個國家提出 70 多個與開放原始碼軟體有關的法案、政策或聲明，歐盟在建立電子化政府及資訊交換平台上，也特別提出詳盡的「遷徙指南」²¹，說明相關單位應如何以開放原始碼的軟體取代現有的專屬軟體。2003 年六月，德國慕尼黑市政府決議將 Windows 系統逐步轉換成 Linux 系統，涵蓋範圍約一萬四千台電腦，由於數量可觀，華盛頓郵報以「慕尼黑革命」稱之²²，可算是歐盟中比較顯著的例子。在亞洲，日本、韓國與中國大陸則合作密切，2004 年四月，三國資訊通訊相關部會首長，於北京達成協議，共同以開發 Linux 系統為標準化藍本，以便打破目前 Microsoft 獨霸市場的局面²³。在國內，產官學界發起「阿里山計畫」，希望以五年時間推動 Linux 系統，經濟部則設立「自由軟體入口網站」²⁴，作為推廣開放原始碼的平台。從上述諸多事件可以看出，開放原始碼已形成一股潮流，在資訊領域佔有相當的重要性。

¹⁹ 曾有人試圖在 Linux 新版的系統核心中植入安全漏洞，但隨即被其他社群成員發現而移除。參見 Kevin Poulsen, *Linux kernel backdoor blocked*, Nov. 7, 2003, at http://www.theregister.co.uk/2003/11/07/linux_kernel_backdoor_blocked (visited May 4, 2004).

²⁰ Rishab Aiyer Ghosh 在 Open Source: A case for Open Source: A case for developing countries 一文中指出，以 Microsoft Office 在美國售價\$800 為例，如採用平均國民所得作基準，換算成開發中國家如越南，則售價相當於\$68,587。

²¹ 報告原文共 148 頁，涵蓋硬體、伺服器、網路及常用軟體等遷移及規劃建議，可供政府單位參考外，對產業界亦有價值，文件網址為 <http://europa.eu.int/ISPO/ida/export/files/en/1618.pdf> (visited May 24, 2004).

²² 吳端輝，Linux 的慕尼黑革命，參見

<http://taiwan.cnet.com/enterprise/column/0,2000062893,20087554,00.htm> (visited May 24, 2004).

²³ 參見 93 年四月資策會科技法律要聞「日韓大陸達成協議 合作開發 Linux 軟體」，網址為 <http://stlc.iii.org.tw/tlnews/02-93-04.htm> (visited May 24, 2004).

²⁴ <http://www.oss.org.tw>

第二章 SCO v. IBM 及相關案例

第一節 案情簡介

該案爭議的焦點主要圍繞在 Unix 程式碼的所有權上。Unix 是一套 AT&T 於 1960 年代末期到 1970 年代初期開發出來的作業系統²⁵，由於功能強大，逐漸成為學術單位及政府機關採用，並擴大為商業上使用。在 AT&T 廣泛授權的策略下，目前 Sun Microsystems、Hewlett-Packard Co. 和 IBM 手頭上都擁有當年來自於 AT&T 專屬的程式碼。針對 AT&T 專屬的程式碼，許多程式設計師與研究人員為了享有更大的自由度，在 Unix 一路發展的過程中開啟不同的計畫，其中最重要的是 1970 年代加州大學推動的改良版本 BSD²⁶，除了強化許多功能外，目標在於移除所有 AT&T 專屬的程式碼，藉此避免 AT&T 主張其著作權。到了 1980 年代，MIT 研究員 Richard Stallman 開啟另外一個 GNU 計畫，希望創造出與 Unix 類似而不含 AT&T 程式碼的新系統，而 1990 年代，則加入了當時為芬蘭赫爾辛基大學學生的 Linus Torvalds，Torvalds 希望在自己的電腦而非大型主機上執行類似 Unix 的程式，於是開發出一套作業系統的基礎核心，這部分的創作後來與 GNU 計畫形成意外而完美的互補²⁷，並透過開放原始碼的模式，快速發展成非商業主流以外絕佳的替代系統²⁸。

Stallman 除了技術上的貢獻外²⁹，並成立自由軟體基金會提倡軟體應該採取自由流通、複製、執行、修改的理念，為了確保該理念推行無礙，並保護該理念下貢獻的原始碼有效地維持在公眾可得的範圍內，避免出現搭便車的情況，Stallman 在 1989 年提出了

²⁵ Unix 並非完全由 AT&T 獨立開發完成，參見本章第二節第二項「SCO 未能釐清原始碼的爭議範圍」。

²⁶ See Berkeley Software Distribution, at <http://www.bsd.org> (visited Apr. 2, 2004).

²⁷ Richard Stallman 指出 GNU/Linux 是比較正確的稱呼，因為原本 Linux 所指的是作業系統的核心，搭配 GNU 的工具之後成為完整的作業系統。但一般為求簡便，大多直接以 Linux 稱之。

²⁸ See Evan P. Schultz, *Is the Future Free?*, Oct. 13, 2003, available at <http://www.law.com/jsp/article.jsp?id=1066080435935> (visited Apr. 1, 2004).

²⁹ 曾隱居一年半載，以超過十萬行程式寫出廣為流通的編譯器軟體 GCC，為其後自由軟體的推展打下基礎。

GNU General Public License (GPL)公眾授權。GPL的特點在於，軟體必須附上原始碼供他人檢視或修改，他人改良後如加以散布，也必須再以GPL做為授權條款公佈改良的原始碼。透過這種方式，程式設計師可以確保自由軟體的理念得以延續，如薪火相傳般不斷增加自由軟體的數量。

到了1993年，AT&T對Unix市場已失去興趣，於是將Unix系統實驗室賣給Novell公司，Novell發現Unix無利可圖，於1995年將Unix再轉賣給SCO當年的前身公司。當年AT&T擁有Unix時，曾授權給IBM開發自己專屬的Unix系統，稱為AIX，根據SCO的說法，SCO前身公司當年取得Unix的所有權，曾有一段時間與IBM密切合作，將Unix移植到一般個人電腦常用的Intel處理器之上。然而，過去五年來IBM的競爭策略轉向，開始大力推動開放原始碼的主流系統Linux，Linux可在一般個人電腦上執行，與SCO的產品直接競爭，這一點成為日後爭訟的導火線。SCO認為IBM為了朝服務導向的獲利模式轉型，刻意採取不當方式降低了Unix系統的市場價值，甚至，後來直接將AIX的程式碼以GPL的方式對外公開。基於此，SCO於2003年三月對IBM提起訴訟，指控侵害其營業秘密、著作權、違反授權合約及構成不公平競爭，要求數十億美元的損害賠償，同時，SCO認為GPL公眾授權有違憲之虞。

IBM擁有充足的法律資源，Linux為其競爭策略中不可或缺的要件之一，加上本身擁有充分的智慧財產權，例如，專利數量長年高居榜首，因此，IBM除了駁斥SCO所有的指控外，並反控SCO侵害其四項專利。IBM表示與Novell或SCO前身公司所簽訂的合約，已對Unix的使用取得永久而且無可撤回的完整權利。更進一步，IBM指出SCO的前身公司Caldera當年經營Linux市場時，也透過GPL的規範運作，亦即，SCO自己也經由GPL釋出一些強化Linux核心功能的程式碼，如今，反過來質疑其他善用GPL的廠商，其立場十分薄弱。

在片面宣布IBM無權繼續在AIX系統使用Unix程式碼後，SCO接著對上千個Linux商業用戶寄出警告函，表示Linux實質上是缺乏授權的Unix衍生系統，為避免侵權，建議用戶向SCO付費以取得合法授權。在控告IBM之前，SCO曾對外宣布SCOsouce授權方

案³⁰，SCO表示，客戶不斷提出在Linux上使用Unix共用函式庫的需求，加上本身意識到智慧財產權的價值以及對股東的責任，為了強化公司的無形資產，SCO有必要採取合理的授權，透過授權，一方面增加Linux上可以執行的Unix軟體，一方面增加Linux用戶³¹。部分用戶收到警告函之後，同意支付授權費用，例如，一家網路主機代管公司為確保客戶不受訴訟的干擾，向SCO取得授權。對該業者來說，此舉固然讓部分客戶放心，不過，也意外引起部分客戶的批評，批評者認為付出授權費用等於不明就理向SCO低頭³²。

唯恐Linux的形象受到貶損，2003年八月，Linux市場上最大的經銷商Red Hat公司，另外對SCO提起訴訟，請求法院以簡易判決方式，認定Linux並未侵害SCO的著作權或營業秘密，SCO則提起反訴。法官認為，系爭核心問題仍為Unix所有權，歸屬的認定與SCO v. IBM案相同，為節省法院資源，無需另起爭端，因此，法官裁決該案應該暫緩等到前案明朗之後再做定奪³³。

在煙硝四起的爭論中，一直視Linux為主要競爭者的Microsoft，同意向SCO簽署正式合約取得Unix授權。Microsoft表示，此舉有助於維持Windows系統與Unix系統的相容性介面，同時能夠合法使用SCO的智慧財產權，Microsoft強調支付的授權費用，與SCO指控Linux系統侵權無關，類似的智慧財產權授權合約十分常見，不值得外界大驚小怪。過去Microsoft雖曾向AT&T取得Unix授權，但受到產品線擱置的影響，授權也早已失去實質效力，因此，有再向SCO取得授權的必要。SCO提供授權以來，前兩個合約金額超過一千萬美元，Microsoft簽約是其中之一，但Microsoft拒絕透露支付的額度³⁴。2003年

³⁰ See SCO, *SCO Intellectual Property License for Linux*, Jan. 22, 2003, at <http://www.thescogroup.com/scosource/linuxlicense.html> (visited May 14, 2004).

³¹ See SCO, *SCOsouce Q&A*, at http://www.thescogroup.com/scosource/Final_SCOsouce_QandA.html (visited May 14, 2004).

³² See Robert McMillan, *SCO Linux licensee has second thoughts on deal*, Mar. 25, 2004, at http://www.infoworld.com/article/04/03/25/HNscolicensee%20_1.html (visited May 16, 2004).

³³ See Steven J. Vaughan-Nichols, *SCO and Red Hat Case Goes Into Hibernation*, Apr. 15, 2004, at <http://www.eweek.com/article2/0,1759,1568127,00.asp> (visited May 16, 2004).

³⁴ See Juan Carlos Perez, *Microsoft licenses Unix technology from SCO*, May 19, 2003, at <http://www.nwfusion.com/news/2003/0519mssco.html> (visited May 17, 2004).

十月，創投公司BayStar證實募集五千萬美元投資SCO進行訴訟，與Microsoft高階主管居中牽線有關。在公開之前，外界對Microsoft扮演的角色並不知情，一直到開放原始碼社群公布一封SCO委外策略公司(S2 Strategic Consulting)的內部信件，信中指出Microsoft準備將八千六百萬美元以不同管道挹注SCO，整個事件才揭露出來。隨後SCO證實信件的真實性，但表示該外部顧問對事情的真相有所誤解³⁵。

2004年一月，SCO案加入新的變數。當年從AT&T手中買下Unix系統的Novell公司，經過多年，Novell迫於市場遭Microsoft蠶食鯨吞，營運策略上出現重大轉變，Novell於2003年突然買下Linux第二大經銷商SuSE，一舉成為Linux市場上的重要廠商。針對SCO宣稱擁有Unix所有權，Novell公開表示，當初與AT&T簽訂的合約中，已排除控制衍生著作的權利，同樣的條款適用於SCO前身公司向Novell所購得的Unix系統，因此，如今SCO宣稱IBM非法散布其Unix相關的衍生著作，是無法有效成立的說法。Novell表示SCO並未取得Unix系統的著作權後，SCO決定反擊，另外提起訴訟控告Novell涉嫌誹謗，在起訴狀中表示Novell對Unix所有權的不實陳述已影響SCO的營運³⁶。隨後，Novell也向法院提出聲請，表示SCO在無法證明Novell的說法不成立、且無法證明擁有Unix的著作權之前，法院應撤銷該訴訟³⁷。本案與IBM案密切相關，如果Novell的主張成立，則SCO根據擁有Unix著作權為前提的諸多指控可能頓失立場。

智慧財產權訴訟常常與商業競爭策略有關，SCO相關案件也不例外。Novell主張擁有Unix以保護Linux，對於訴訟對手SCO的出現，其實與過去自己的策略有關。SCO當年從Novell手中取得Unix時，為Canopy Group旗下的子公司Caldera，2004年五月，美國猶他州上訴法院支持地方法院的見解，在Novell控告Canopy違反合約一案中，維

³⁵ See Robert McMillan & Joris Evers, *Microsoft behind \$50 million SCO investment*, Mar. 11, 2004, at <http://www.nwfusion.com/news/2004/0311mssco.html> (visited May 17, 2004).

³⁶ See David Becker, *SCO sues Novell over copyright claims*, Jan. 20, 2004, at <http://news.com.com/2100-7344-5143977.html?tag=nl> (visited Apr. 2, 2004).

³⁷ See David Becker, *Novell raps SCO claims*, Feb. 11, 2004, at <http://news.com.com/2100-7344-5157484.html> (visited Apr. 2, 2004).

持 Novell 勝訴的判決。受該案的影響，Novell 當年透過 Canopy 擔任白手套控告 Microsoft 的策略，也跟著公布開來。1996 年 Canopy 及子公司 Caldera 控告 Microsoft 違反反托拉斯法的規定，在 1990 年代初期以不正當競爭方式，迫使另外一套相容作業系統 DR DOS 退出市場。爭訟四年後，Microsoft 以兩億五千萬美元達成和解。DR DOS 原為 Novell 產品，Novell 唯恐直接控告 Microsoft 將引發對方報復，於是將系統以一億美元賣給 Canopy，並於合約中要求 Canopy 必須控告 Microsoft，如果勝訴，所得部分金額另以權利金名義歸 Novell 所有，Canopy 同時不得揭露 Novell 的計畫。惟 Microsoft 敗訴之後，Novell 與 Canopy 就權利金的計算發生爭執，Canopy 堅持必須扣除律師費用及其他開銷後，才加以計算，Novell 無法同意，雙方於是走上法庭。地方法院根據相關文件，認為 Canopy 扣除部分費用為無理由，Canopy 不服提出上訴，Canopy 在上訴法院敗訴後，是否再上訴最高法院尚未可知。但有人認為，當年 Caldera 從 Microsoft 和解案所取得的資金，正資助著現在 SCO 對 IBM、Novell 及其他廠商興訟，Novell 當年的商業策略可能陰錯陽差地養虎為患，並回過頭來與自己為敵³⁸。

回到 SCO 案的主軸，基於訴訟策略的考量，SCO 將打擊面延續到商業用戶端。2004 年三月，SCO 分別對使用 Linux 及 Unix 的用戶提出訴訟。SCO 控告汽車零件經銷商 AutoZone 使用 Linux 系統侵害 SCO 原始碼的著作權，另外控告汽車製造商 DaimlerChrysler 違反使用 Unix 的軟體合約³⁹。處於消費者立場，一般對這類訴訟多傾向於息事寧人，SCO 可能抓住這種心理上的弱點，希望藉此增加授權的營收，或者提高消費者對 IBM 要求賠償的壓力。SCO 表示，RIAA⁴⁰針對網路上非法流通音樂，一口氣控告許多消費者，發揮了伸張智慧財產權的正面效果，值得 SCO 借鏡。DaimlerChrysler 公司被告之後，要求法院駁回 SCO 的聲請，認為相關事實及理由並不充分，對於系爭

³⁸ See Bob Mims, *Novell wins breach-of-contract dispute with Canopy Group*, May 15, 2004, at <http://www.sltrib.com/2004/May/05152004/business/business.asp> (visited May 16, 2004).

³⁹ 關於 SCO 與 DaimlerChrysler 之間訴狀及答辯的對照，請參見 Pamela Jones, *SCO v. DaimlerChrysler: Complaint and Answer Chart*, Apr. 30, 2004, at <http://www.groklaw.net/article.php?story=20040430012823325> (visited May 16, 2004).

⁴⁰ Recording Industry Association of America.

軟體，DaimlerChrysler 公司已經超過七年未曾使用，與 SCO 之間未曾簽訂任何合約，也無義務提出 SCO 口中符合約定的證明⁴¹。AutoZone 所屬的全國零售聯盟，以一千四百萬家機構及全美五分之一工作人口的代表性，站在支持 AutoZone 的立場，則發表正式聲明，表示 SCO 與 Novell 爭訟定案之前，沒有理由控告 Linux 或 Unix 用戶⁴²。

2004 年三月，審理 SCO v. IBM 案的法官裁決表示，SCO 提出的證據尚不足以支持 IBM 將 Unix 專屬原始碼流入 Linux 系統的指控，雙方應提出更多的證據，其中包括：SCO 必須明確指出流入 Linux 系統的是哪些程式碼、System V 版本中哪些 Unix 原始碼與 IBM 所屬系統的侵權有關、SCO 自己提供散布的原始碼有哪些、IBM 則需提供自己 Unix 系統中上百項軟體的相關資訊，以利 SCO 進一步指出與本案有關的項目⁴³。

主導投資 SCO 訴訟的創投公司 BayStar，於 2004 年四月公開表示，可能將投資的兩千萬美元撤離。BayStar 公司認為，SCO 的管理階層表現不佳，花太多時間跟 Linux 陣營的運動人士爭辯，而未專注於訴訟策略與資源的運用上，真正重要的是法院對智慧財產權歸屬的認定，而非法庭外的意氣之爭。受此事件影響，SCO 股價在短時間內大幅下滑⁴⁴。開放原始碼社群普遍認為，BayStar 公司自 1998 年成立以來，從未對自己投資的公司公開發表負面聲明，所挹注的資金也直接影響訴訟案背後的資源，由 BayStar 的聲明，可以臆測 SCO 在訴訟中可能已居下風，連投資者都考慮全身而退。針對此點，有人則提出不同意見，根據 BayStar 的聲明，BayStar 希望 SCO 專注在智慧財產權的訴訟案上，SCO

⁴¹ See Stephen Shankland, *SCO suits target two big Linux users*, Mar. 4, 2004, at <http://asia.cnet.com/newstech/industry/0,39001143,39170786,00.htm> (visited May 16, 2004); Robert McMillan, *DaimlerChrysler responds to SCO lawsuit*, at <http://www.computerworld.com/softwaretopics/os/linux/story/0,10801,92742p2,00.html> (visited May 16, 2004).

⁴² See National Retail Federation, *SCO Group Lawsuit Appears Unfounded*, available at <http://www.nrf.com/content/default.asp?folder=press/release2004&file=scosuit0504.htm&bhcp=1&bhqs=1> (visited May 16, 2004).

⁴³ See Chris Preimesberger, *SCO Group, IBM exchange new evidence during 'discovery'*, Apr. 19, 2004, at <http://www.newsforge.com/article.pl?sid=04/04/19/058215> (visited May 16, 2004).

⁴⁴ See Steve Lohr, *Investor's Pullout Stirs Doubts About SCO Group*, Apr. 22, 2004, at <http://www.nytimes.com/2004/04/22/technology/22sco.html?ex=1084680000&en=99b12a3b14bc67a6&ei=5070> (visited May 14, 2004).

隨後低調裁員，兩件事加起來，很可能是SCO準備轉變為訴訟型公司的前兆，純粹以訴訟為業，不需要太多人手，重要的是手上擁有的智慧財產權，以及如何從其他來源取得或代理他人的智慧財產權，然後發起訴訟。開放原始碼社群最不需要的，便是一家到處藉侵害智慧財產權名義找人麻煩的公司⁴⁵。在資金撤出部分，BayStar發表聲明後不久，當時一起投資SCO的加拿大皇家銀行，也準備撤出三千萬美元全部資金，將股份轉換或轉賣給BayStar，據估計，加拿大皇家銀行七個月前的投資約損失一半。BayStar如取得股份，對SCO的影響力將跟著提高⁴⁶。

SCO針對先前主張GPL違憲，因提不出強而有力的理由，於是在2004年五月改變說法。SCO寄發國會的公開信中表示，著作權法制建構在美國憲法的基礎上，國會與最高法院認為追求利潤是科學進步的動力，因此，宣稱軟體應該免費，實與上述精神有違。在這種情況下，SCO認為GPL削弱了智慧財產權制度的效力，違反美國憲法的精神。SCO進一步表示，自訴訟以來受到各方惡意的攻擊，如網站遭到灌爆而停擺，SCO仍願意堅持下去，為的就是希望GPL引發的開放原始碼軟體，不會毫無限制地威脅美國企業所追求的資本主義⁴⁷。

本案事實紛陳、關係錯綜複雜，對資訊界來說，Unix系統發源至今衍生的派別不計其數，加上後起之秀Linux系統的重要性與日俱增，可謂許多系統都成了SCO案潛在的影響範圍，因此，不難理解何以SCO相關訴訟受到各方的矚目，至於對開放原始碼運動的影響，如果採正面的觀點，可能也是開放原始碼陣營向大眾澄清、消除外界疑慮的良機。

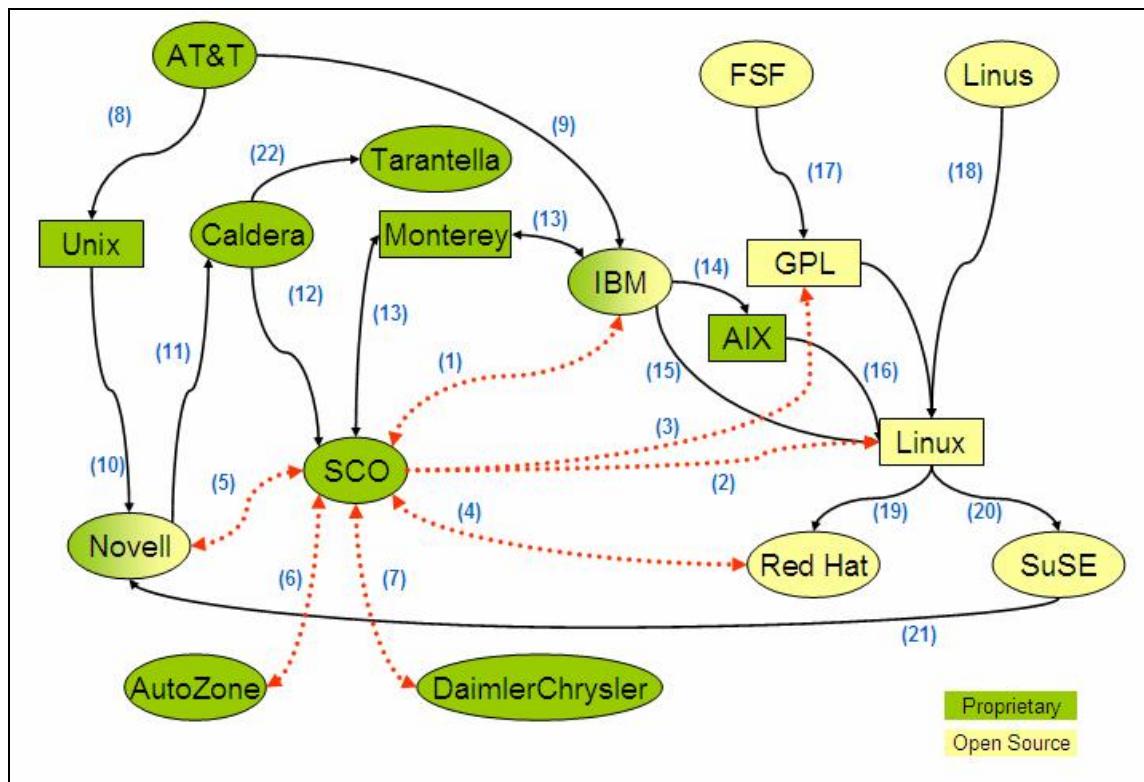
底下以圖示列出SCO相關案件中的關係，並以表格列出圖中重要法律關係或事實關

⁴⁵ See Brian Proffitt, *Is BayStar SCO's Bully or Savior?*, May 7, 2004, at http://linuxtoday.com/news_story.php3?ltsn=2004-05-07-022-26-OP-BZ-CD (visited May 14, 2004).

⁴⁶ See Stephen Shankland, *Canadian bank backs away from SCO*, May 7, 2004, at http://zdnet.com.com/2100-1104_2-5208396.html (visited May 14, 2004); Chris Preimesberger, *Major SCO Group investors reposition stock holdings*, May 7, 2004, at <http://business.newsforge.com/business/04/05/07/1942220.shtml?tid=20&tid=85> (visited May 14, 2004).

⁴⁷ See Gene J. Koprowski, *SCO Changes Legal Tactics in Federal Court*, May 3, 2004, at <http://www.linuxinsider.com/perl/story/33614.html> (visited May 14, 2004).

係的說明，做為後續分析或觀察訴訟案的重點。



圖三 SCO 相關案件之關係



上圖 關係	訴訟爭點及背景事實之整理
(1)	<p>I SCO於控告IBM的訴狀中主張：</p> <ul style="list-style-type: none"> ■ IBM違反當初與AT&T簽署的軟體合約，合約中規定IBM取得的軟體應限於內部使用，權利不得移轉，且衍生著作視為原軟體的一部份。IBM推動Linux系統時將Unix技術進行外部商業用途，違反合約的規定。 ■ IBM違反政府對出口管制的規定，且違反當初AT&T合約中未經同意不得將軟體予以出口的規定。 ■ IBM違約對SCO的市場造成重大影響，故賠償金額不得少於十億美元。 ■ IBM與AT&T的合約終止後，繼續重製並散布Unix軟體、相關工

具及文件，已構成著作權的侵害。SCO擁有Unix原始碼、目的碼、相關文件及衍生著作的著作權，並向著作權局登記後受到保護。

- SCO及Unix系統先前的擁有者在Unix上面投入可觀的資源，投資期間超過二十年，金額超過十億美金，受到IBM的影響，Unix系統的價值可能大幅減損，相關技術並落入Linux領域構成不公平競爭。其他不公平競爭的行為還包括：侵害營業秘密、侵害原始碼、違反合約、誘導他人違反保密條款、以虛偽不實的方式與SCO合作開發軟體等。
- 受到IBM不公平競爭的影響，SCO在Intel平台上的Unix營收大幅減少，並失去成為該平台領導廠商的機會，未來更可能失去所有Unix的市場。除了請求損害賠償金額不得少於十億美元外，訴訟中可預見因IBM不公平競爭所導致的損失並應於未來一併計算，法院並應對IBM課以損害性賠償。

| IBM於反訴中主張：

- SCO違反前身公司當初與Novell、IBM簽署的合約，意圖終止IBM對Unix取得永久且不可撤回的授權，且未能對指控部分提出適當的通知。
- SCO對IBM自己的AIX系統及Linux相關系統的不實陳述，已影響客戶採購上的決策，特別是公開否定上述系統的合法性，表示IBM無權製造、銷售、授權他人使用上述系統，對IBM造成實質的傷害。
- SCO惡意對IBM及開放原始碼社群所屬智慧財產權的不實指控，已構成不公平競爭。
- SCO蓄意擴大對Unix權利範圍的主張，已對IBM與客戶的商業往來構成妨礙，有關IBM將SCO的智慧財產大規模複製到Linux系

	<p>統的指控，也損及IBM在Linux社群及相關開放原始碼社群的長期合作。</p> <ul style="list-style-type: none"> ■ 受到SCO不公平及詐欺行為的影響，IBM要求在SCO為惡意且知情的情況下，法院應判定賠償金額加重為三倍。 ■ SCO違反GPL授權條款。SCO將IBM以GPL條款釋出的原始碼併入自己的產品並加以散布，應受GPL授權的規範，IBM並對SCO違反GPL條款所構成的不利益要求損害賠償。 ■ SCO複製、修改、散布GPL條款下的軟體，不得增加GPL原本所沒有的限制，SCO既已同意採用GPL條款，應受禁反言的約束。 ■ IBM擁有本身釋出的原始碼著作權，並完成多項著作權登記，原始碼本身亦標有著作權聲明，IBM採用GPL條款對外授權，SCO如果違反該授權條款，即構成著作權的侵害。 ■ IBM認為本身使用、重製、改良Linux系統的行為，並未侵害、誘使或輔助侵害任何SCO專屬而有效的著作權。 ■ SCO侵害IBM所屬編號4,814,746、4,953,209及5,805,785三項美國專利，SCO知情之後繼續製造、使用、銷售、提供銷售侵權之產品，已構成故意侵害之情形，IBM對此請求三倍金額之懲罰性損害賠償。
(2)	<ul style="list-style-type: none"> SCO認為如果不是IBM將Unix的功能納入Linux系統，Linux在2.4.x版以後的系統核心將不足以執行高階的企業運算功能，如系統的延展性、效能的提升、數個系統形成叢集等功能。
(3)	<ul style="list-style-type: none"> SCO質疑GPL的效力。 SCO認為Linux開放原始碼的特性破壞了其專屬軟體的保密性。
(4)	<ul style="list-style-type: none"> Red Hat於控告SCO的訴狀中主張： ■ Red Hat使用、銷售、散布或客戶使用的任何Linux系統，皆未侵

害SCO所屬的著作權，Red Hat希望法院以簡易判決方式，確定Red Hat未侵害SCO著作權以消除疑慮。

- Linux系統的原始碼以公開方式揭露，無法構成營業秘密的侵害，有關Linux系統的使用或銷售，並未侵害SCO的營業秘密。Red Hat希望法院以簡易判決確認Red Hat未侵害SCO營業秘密以消除疑慮。
- SCO透過報章媒體等公開方式散布不實、誤導的陳述，對Red Hat客戶、潛在的使用者及其他Linux軟體的使用者已構成詐欺行為，並損及Red Hat的商業利益，對此Red Hat要求損害賠償。
- SCO惡意的行為對Red Hat已造成不可回復的傷害，如果其行為不受禁制令所禁止，Red Hat所受的損害將繼續擴大。
- SCO的行為已構成不公平競爭。
- SCO的侵權行為對Red Hat與客戶的關係造成損害，並損及潛在的商業利益及商譽。

I SCO在答辯中主張：

- Red Hat對於SCO將控告其侵害著作權一事，所陳述的事實並未達到構成合理顧慮(reasonable apprehension)的程度，法院應駁回其訴。SCO對外曾明白表示如果需要控告Red Hat，必定在IBM一案定讞之後，SCO也未曾接觸Red Hat指控其產品侵害SCO的著作權或營業秘密。
- 即使法院決定受理，也應進一步考量與IBM案的重疊程度，事實上關於著作權侵害及其他爭議，大多已為IBM案所涵蓋。
- Red Hat訴狀中所引述的SCO公開聲明，應為SCO在憲法第一修正案下受言論自由保護的部分，SCO對進行中案件的陳述也受到普通法下訴訟豁免原則doctrine of litigation immunity的保護，因

	此，法院應駁回其訴。
(5)	<p> SCO控告Novell時主張：</p> <ul style="list-style-type: none"> 根據當初的採購合約，SCO獨自擁有任何與Unix及UnixWare原始碼及文件有關的著作權，Novell公開表示其為上述著作權之所有人，已構成所有權的毀謗(slander of title)，並損害SCO的商譽及與客戶的關係。 Novell對SCO造成商業上不利益為惡意之行為，法院應課以懲罰性損害賠償。 法院應對Novell施以禁制令，要求Novell將Unix及UnixWare中註冊的著作權讓渡給SCO，並收回之前關於Unix所有權的不實陳述。 <p> Novell答辯中主張：</p> <ul style="list-style-type: none"> 當初採購合約中已將Unix的著作權排除在合約範圍之外，SCO未能依著作權法規定提出所有權移轉的有效證明，則無法證明其為所有權人，自然也無法證明Novell的主張為不實之陳述。 如果要構成所有權毀謗，原告必須提出明確的損害，SCO於訴狀中只提出推測性的一般損害，因此法院應駁回其訴。 Novell表示當初與AT&T的合約中，已排除控制衍生著作的權利，同樣的條款應適用於Caldera向Novell所購得的Unix系統。
(6)	<p> SCO控告AutoZone時主張：</p> <ul style="list-style-type: none"> Linux系統中含有SCO專屬的原始碼，AutoZone使用Linux系統已構成對著作物的使用、複製、修改或散布，在未取得SCO適當授權之前，形同對SCO智慧財產權的侵害。 法院應禁止AutoZone繼續使用侵權的軟體，SCO並要求損害賠償。

(7)	<ul style="list-style-type: none"> SCO控告DaimlerChrysler時主張： <ul style="list-style-type: none"> Unix軟體授權書規範Unix系統的使用，其中允許授權人向被授權人要求提供符合授權規範的證明，DaimlerChrysler拒絕提供SCO為確保被授權人遵守合約的要求，已違反原本授權的規定。 在軟體合約中，被授權人只能將Unix系統供內部使用，軟體涵蓋的範圍包括衍生著作在內，DaimlerChrysler將Unix系統轉換成Linux系統並不在原本的授權範圍之內。
(8)	<ul style="list-style-type: none"> AT&T於1970年代初期開發出Unix系統，其中包含來自外界貢獻的原始碼，AT&T並採取廣泛授權的策略。
(9)	<ul style="list-style-type: none"> IBM曾取得AT&T的正式授權。 SCO認為在授權關係中，被授權人應維持授權軟體的保密性，不得對被授權單位員工以外的第三者揭露原始碼。
(10)	<ul style="list-style-type: none"> AT&T於1993年將Unix系統實驗室賣給Novell公司。
(11)	<ul style="list-style-type: none"> Novell於1995年將Unix再轉賣給Caldera。
(12)	<ul style="list-style-type: none"> Caldera原為Linux廠商，併購當年Santa Cruz Operation公司伺服器部門之後，改名為Caldera Systems Inc，之後再改名為現在的SCO Group Inc，即SCO v. IBM一案中之原告。 Caldera當初經營Linux市場時，曾透過GPL釋出強化Linux核心功能的程式碼。
(13)	<ul style="list-style-type: none"> SCO與IBM曾就如何將Unix系統移植到Intel平台上進行Monterey合作計畫。
(14)	<ul style="list-style-type: none"> IBM曾於AT&T擁有Unix時取得授權，開發出自己專屬系統稱之為AIX。
(15)	<ul style="list-style-type: none"> IBM商業策略轉型後，對Linux及開放原始碼運動投入可觀資源。
(16)	<ul style="list-style-type: none"> IBM將AIX部分技術貢獻至Linux系統。

	<ul style="list-style-type: none"> 但SCO主張IBM違約後失去繼續使用Unix軟體及修改衍生著作的權利，IBM應歸還相關軟體或予以銷毀，如AIX系統。IBM當初與AT&T之間關於再授權合約中的權利，應一併予以終止，不得再對外散布Unix系統的目的碼。
(17)	<ul style="list-style-type: none"> 自由軟體基金會(FSF)成立於1984年，創設GPL公眾授權之重要規定包括：軟體散布時必須揭露原始碼、不得對原始碼本身收費、使用者可自行複製、修改、散布其原始碼，但必須附上並遵守GPL授權條款的規定、散布衍生著作時必須採用GPL條款再公開原始碼、不提供任何擔保，所有風險由使用者自負等。
(18)	<ul style="list-style-type: none"> Linus Torvalds於1991年開發出作業系統的核心，與自由軟體基金會的GNU計畫日後共同構成目前習稱的Linux系統。
(19)	<ul style="list-style-type: none"> Red Hat為Linux市場上最大之經銷商。
(20)	<ul style="list-style-type: none"> SuSE原為Linux市場上第二大之經銷商。
(21)	<ul style="list-style-type: none"> Novell於2003年買下Linux第二大經銷商SuSE，加入開放原始碼陣營。
(22)	<ul style="list-style-type: none"> Santa Cruz Operation公司伺服器部門為Caldera併購後，放棄Unix市場改名為Tarantella Inc。

表一 SCO 相關案件中訴訟爭點及背景事實之整理

第二節 深入分析

參考 Groklaw 網路社群⁴⁸的分析，SCO 案涉及的爭議至少包括下列幾項。

第一項 SCO 本身涉嫌違反 GPL 授權條款的規定

SCO 本身違反 GPL 公眾授權的規定。自由軟體基金會表示，SCO 散布的軟體中，並非只有核心部份的 Linux 系統，而是整體的 GNU/Linux 系統。大部分在 GNU 名下元

⁴⁸ See Pamela Jones, *Digging for Truth, at*

<http://www.groklaw.net/staticpages/index.php?page=20030929022014462> (visited Apr. 2, 2004).

件其著作權都屬於自由軟體基金會，受到 GPL 條款的保護，Linux 本身也採用 GPL 條款。GPL 最重要的精神在於原始碼的揭露，賦予使用者自行修改的權利，並要求再散布時必須揭露自己改良的部份。如今，SCO 宣稱其營業秘密落入 GNU/Linux 系統中，而本身卻繼續採用 GPL 條款散布開放原始碼程式，明顯違反 GPL 中第七款的規定⁴⁹。在過去二十年，自由軟體基金會已盡適當的注意義務，在處理著作權的移轉時，要求程式設計師確認沒有侵害他人的營業秘密、專利或著作權，並謹慎檢查 GPL 條款下的程式碼，才將原始碼併入整個系統，因此，SCO 對 GNU/Linux 侵害他人智慧財產權的指控可謂空穴來風⁵⁰。

在 Progress Software v. MySQL AB⁵¹一案中，自由軟體基金會法務長 Moglen⁵²於預審時指出：GPL 是十分單純的著作權授權條款，但不像一般契約上要求對價的義務。關於授權，目前軟體界一般常見的做法是，創作者在著作權法下保留後續創作的權利，如修改軟體，而將特地情況下的權利賦予取得授權的使用者，如使用軟體的權利。所謂購買軟體，嚴格來說是取得對方授予的使用權，而非軟體本身的所有權。GPL 的概念與其相異，公眾授權中所規範的並非保留創作者的權利，而是放棄著作權法下部分專屬的權利，讓一般大眾能夠使用、複製及修改軟體，如果取得 GPL 軟體純粹供私人使用，則自行複製及修改完全不需負擔其他義務。但如果涉及軟體的再散布，無論是否修改軟體，都必須遵守 GPL 條款，繼續釋出創作者的部份權利。簡言之，使用者如果改良受 GPL 條款保護的軟體，再對外散布時必須遵守三項規定⁵³：

⁴⁹ 參見 GPL §7，原文為：If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all.

⁵⁰ See f2 network, *SCO using scare tactics to get licensing fees*, May 22., 2003, at <http://www.smh.com.au/articles/2003/05/22/1053196678272.html> (visited Apr. 2, 2004).

⁵¹ 195 F.Supp.2d 328 (2002).

⁵² 哥倫比亞大學法學院教授，自願無償為自由軟體基金會擔任法務工作，確保 GPL 的完整與落實。從十三歲到擔任法官助理前，採兼差或全職方式擔任程式設計師，並有六年時間任職於 IBM 從事研發工作，故其資訊方面的專業能力無庸置疑。

⁵³ See GNU Project, *Affidavit of Eben Moglen on Progress Software vs. MySQL AB Preliminary Injunction Hearing*, at <http://www.gnu.org/press/mysql-affidavit.html> (visited Apr. 2, 2004).

- (i) 散布行為必須而且只能在 GPL 條款下進行，並且不得附加額外的授權條件⁵⁴。
- (ii) 散布的內容必須包含原始碼，使他人得以閱讀、修改程式，不得只提供目的碼供機器執行⁵⁵。
- (iii) 散布時需提供一份 GPL 條款，讓使用者知道有關使用、複製、修改及散布的權利義務，特別是散布時必需遵守的規定⁵⁶。

根據 GPL 第五款，由於使用者不需要簽名，使用者可以不接受整份授權的規範，但不接受的話，則無法取得修改及散布的權利，反之，任何修改及散布的動作將視為接受授權條款的規範。SCO 散布 GPL 授權下的軟體，等於接受了 GPL 的規範，而有遵守相關規定的義務。

GPL 的效力可以從 *Progress Software v. MySQL AB* 一案中得到參考。*MySQL AB* 是一家瑞典的軟體公司，採用開放原始碼模式開發資料庫軟體，在開放原始碼社群取得極高的佔有率，2001 年美國 *Progress Software* 公司涉嫌違反 GPL 條款的規定，修改 *MySQL AB* 公司的軟體，加強部分功能後擴大應用，但散布時並未提供原始碼。同時，該公司註冊 mysql.org 網址，也有誤導他人混淆軟體所有權之嫌，而侵害 *MySQL AB* 的商標權。*MySQL AB* 於是在 2001 年七月提起訴訟，關於商標權部分，則與 GPL 無關，地方法院法官在預審時以初步禁制令(preliminary injunction)，裁決 *Progress Software* 在判決前不得繼續以 MySQL 商標行銷產品。Moglen 於預審程序中指出，根據取得不同版本的軟體

⁵⁴ 參見 GPL §2 (b)，原文為：You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

⁵⁵ 參見 GPL §3 (a)，原文為：Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

⁵⁶ 參見 GPL §1，原文為：You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

加以測試，的確發現 Progress Software 在自己發行的軟體中未提供應有的原始碼，以致於無法重新編譯出據以執行的同一份程式，違反了 GPL 第三款的規定，而違反授權條款的規定，根據 GPL 第四款，則喪失了複製、修改及繼續散布軟體的權利。針對違反 GPL 散布軟體的規定，法官並未准予禁制令。預審時法官表示，對她而言，GPL 視為可執行(enforceable)且具有拘束力(binding)的授權許可，應無疑問，但被告同意遵守 GPL 條款後，對 MySQL AB 公司而言似乎未造成不可恢復的損害⁵⁷。不過，Progress Software 隨後同意支付適當的權利金取得 MySQL AB 的授權，兩家公司於 2002 年 11 月達成和解，於是原本可能是第一次正式在法庭上檢驗 GPL 效力的機會，遂移轉到後來的 SCO v. IBM 一案上。

第二項 SCO 未能釐清原始碼的爭議範圍

SCO 案中爭的原始碼範圍未定，引發不少爭議。SCO 指控 IBM 將其擁有的原始碼流入 Linux 社群後，一直不願公布原始碼受侵害的細節，面對各方的質疑，SCO 以契約上保密條款為由，不肯對外揭露遭到剽竊的原始碼，對於這部份，雙方似乎只能進入採證階段才可能取得更多細節。對開放原始碼社群而言，擁有的程式碼本在光天化日之下，隨時可受公評，面對 SCO 指控後拒絕提出證據，並在權利歸屬尚未確認前向大眾宣稱 Linux 侵害了 Unix 的智慧財產權，使用者應考慮支付權利金給 SCO，開放原始碼社群認為，SCO 的警告函明顯影響了 Linux 系統的推廣，在網路上，並引發一連串的反彈聲浪。Torvalds 認為，該訴訟爭議的焦點應該是 SCO 與 IBM 當年的契約問題，而非 Linux 系統中的智慧財產權，對他而言，SCO 的諸多指控並沒有事實根據。開放原始碼運動人士 Bruce Perens 認為，SCO 的意圖似乎是希望在 Linux 周圍營造一種懼怕、不確定與疑慮(FUD)⁵⁸的氣氛，如果 Linux 真的與 SCO 的智慧財產權有關，只要 SCO 一公布，

⁵⁷ See Free Software Foundation, *Judge Saris defers GNU GPL Questions for Trial in MySQL vs. Progress Software*, Mar. 1, 2002, at <http://www.gnu.org/press/2002-03-01-pi-MySQL.html> (visited Apr. 2, 2004).

⁵⁸ FUD 為 fear、uncertainty、doubt 的縮寫，為開放原始碼社群常見用詞之一，用來指稱對方針對特定議題或對象加以抹黑。

開放原始碼陣營不僅可以辨其真偽，必要時，社群隨時可以重寫有爭議的程式碼，像 SCO 這種在疑雲重重中不斷放槍的行徑，並沒有辦法解決問題⁵⁹。SCO 面對各方的質疑，提出的主要說法是，如果太早公佈受到剽竊的程式碼，開放原始碼社群可能暗中進行部分修改，進而影響未來法院對證據的認定，對 SCO 產生不利的結果⁶⁰。這樣的說法放在開放原始碼的機制下可能難以成立，原始碼一旦在網路上公布，覆水難收，世界各地的伺服器不久便能儲存著複製版本，新舊版本的累積在分散式架構下各自管理，如果有人企圖在網際網路上湮滅證據的話，以開放原始碼的環境而言，應是難上加難。

回顧資訊界的發展，Unix 相關系統在 Windows 系統崛起之前或之後，一直屬於主流作業系統之一，研究 Unix 歷史的學者 Warren Toomey 表示，SCO 宣稱擁有 Unix System V 原始碼的所有權，即使屬實，SCO 自己可能也弄不清楚在 System V 之前錯綜複雜的程式碼來源。在 SCO 案出現後，部分 1970 年代加入 Unix 創作的資深設計師們，懷疑當年的心血可能被列為 SCO 手中打擊 Linux 社群的工具。SCO 在 2003 年八月的年度論壇中，對部份簽署保密協定的與會人士揭露一部份疑遭剽竊的原始碼，試圖平息質疑的聲浪並強化自己的指控，但隨後德國某網站公佈其中兩張投影片，技術專家指出，投影片上的原始碼不但在 Lion's Commentary on Unix 一書和 1984 年的網路佈告欄上可以找到，更重要的是，SCO 的前身 Caldera 公司，曾經採用較 GPL 設限更少的 BSD 公眾授權，流通含有相同程式碼的 V7 版 Unix⁶¹。1970 年代當時著作權觀念尚未普及，一些從學術研究中產生的原始碼併入 AT&T 及其後 Unix 系統時，很可能都沒有加註適當的權利歸屬，尤其此次爭訟標的 System V 更為龐雜，完全不屬於 AT&T 或 SCO 等商業團體的程式碼可能不在少數，而且，裡面也含有不少 BSD 授權下的創作，讓所有權的歸屬與侵害的認定顯得更加複雜⁶²。

⁵⁹ See Stephen Shankland, *SCO: Unix code copied into Linux*, Mar. 1, 2003, at http://news.com.com/2100-1016_3-999371.html (visited Apr. 2, 2004).

⁶⁰ *Id.*

⁶¹ See f2 network, *SCO shows 'copied' code at forum, denials follow*, Aug. 20, 2003, at <http://www.smh.com.au/articles/2003/08/20/1061261197329.html> (visited Apr. 2, 2004).

⁶² See Sam Varghese, *SCO may not know origin of code, says Australian UNIX historian*, Sep. 9, 2003, at

隨著訴訟的推移，SCO 於 2004 年二月修訂對 IBM 第一回合及第二回合質詢的補充答辯⁶³，首次明確列出受到侵害的原始碼，並舉出 Linux 涉嫌涵蓋 IBM 所屬 AIX 或 Dynix/ptx 系統下的數項功能⁶⁴，由於列出的數量不在少數，因此，有待專家進一步分析才能確認 SCO 的指控是否屬實。如果 IBM 將系爭原始碼透過 GPL 方式納入 Linux，則爭議的焦點將回到當初雙方合約的細節，以及 SCO 是否取得 Unix 相關系統的所有權。

第三項 原始碼相同未必等於著作權之侵害

Unix 在 1990 年代曾發生侵害著作權訴訟 Unix System Laboratories, Inc. v. Berkeley Software Design, Inc.⁶⁵，值得此次 SCO 案參考。在 Unix 發展史上，由於創始機構 AT&T 逐步提高授權費用，部分廠商為了移植 Unix 系統上的許多功能以擴大應用範圍，紛紛寄望 BSD 支派的系統能夠加速脫離 AT&T 的範圍，走出原始碼權利金的陰影。在 BSD 的授權條款下，衍生著作可以用目的碼或原始碼的形式流通，也沒有支付權利金的要求，唯一必須遵守的是保留原本的著作權聲明，在後續改良系統中明白指出加州大學及相關程式設計師的貢獻，不得侵害他人的著作人格權⁶⁶。1983 年 BSD 系統在支援 Internet 的發展上相當成功，因而受到許多公司採用，並衍生出後來許多系統。到了加州大學柏克萊分校推出 4.4Lite 版，一般認為 BSD 已正式脫離 AT&T 的影響，整套原始碼不再受到任何限制，於是部分人士籌組 BSDI 公司，正式對外發行整套系統。不幸的是，BSDI 直接以 Unix 名義行銷，引起 AT&T 法務人員的關切，而且，發行的軟體中含有一小部份屬於 AT&T 的原始碼。這件事讓自由軟體的命運因而改觀，並間接促成日後 Linux 系

<http://www.smh.com.au/articles/2003/09/09/1062902037394.html> (visited Apr. 2, 2004).

⁶³ See Pamela Jones, *Exhibit I to IBM's Report on SCO's Compliance*, Feb. 17, 2004, at <http://www.groklaw.net/article.php?story=20040215015800694> (visited Apr. 2, 2004).

⁶⁴ 例如 read copy update、journaling file system、enterprise volume management system、AIO (Asynchronous I/O) 和 "scatter gather" I/O 程式碼。

⁶⁵ 832 F.Supp. 790 (1993).

⁶⁶ Marshall Kirk McKusick, *Twenty Years of Berkeley Unix: From AT&T-Owned to Freely Redistributable*, in OPEN SOURCES: VOICES FROM THE OPEN SOURCE REVOLUTION 40-41 (Chris Dibona, Mark Stone & Sam Ockman eds., 1999).

統的崛起。

面對對方侵權的事實，AT&T 旗下 USL (Unix System Laboratory)不僅控告 BSDI，也將加州大學柏克萊分校列為被告，纏訟期間 BSD 支派無法推出新系統，眼見訴訟曠日廢時，校方決定完全退出 BSD 業務。1993 年三月，紐澤西州地方法官聽審 AT&T 提出的禁制令申請後，以長達四十頁的意見書駁回大部分的指控，將爭點限縮到侵害著作權及營業秘密兩點。令人意外的是，1994 年一月雙方達成和解，AT&T 不再試圖壓制「免費 Unix」的出現，甚至允許 BSD 系統可以使用殘存在內的 AT&T 原始碼⁶⁷。和解之後，BSD 系統從 18,000 個檔案中移除其中三個，對部份檔案進行細微修改，並把 USL 的著作權聲明加到七十個檔案中，而得以繼續散布整套原始碼⁶⁸。

法官在判決書⁶⁹的結論中指出，AT&T 對於版本代號 32V 的 Unix 系統，並未擁有著作權。法官認為：根據「出版說」(Publication Doctrine)，著作所有人在出版著作時如未加註適當的著作權聲明，將失去著作權的保護，該說目前只適用於 1978 年一月一日前出版的作品，系爭 32V 系統發表於 1978 年，因此適用 17 U.S.C. §405(a)⁷⁰及普通法中「有限出版說」(limited publication rule)的規定，在 1989 年三月一日之後出版的著作，遵照伯恩公約的規定，而不再適用舊的出版說。原告在 1992 年登記 32V 系統的著作權，推定其著作權有效，因此，被告如有相反意見，需負擔舉證責任，證明原告廣泛公佈 32V 系統的原始碼而未標示適當的著作權聲明。被告指出，原告在缺乏適當聲明的情況下，將 32V 系統的原始碼散布給上千個取得授權的單位或個人，因此，如果法院認為著作權的有效性成立，則原告需符合§405(a)除外條款的規定，而原告對缺乏著作權聲明的原始

⁶⁷ See Ian F. Darwin, *Why Caldera Released Unix: A Brief History*, Mar. 1, 2002, at <http://www.linuxdevcenter.com/pub/a/linux/2002/02/28/caldera.html> (visited Apr. 2, 2004).

⁶⁸ McKusick, *supra* note 66, at 45.

⁶⁹ Unix System Laboratories, Inc. v. Berkeley Software Design, Inc., Civ. No. 92-1667 (D. N. J., 1993), available at <http://cm.bell-labs.com/cm/cs/who/dmr/bsdi/930303.ruling.txt> (visited May 24, 2004).

⁷⁰ 著作物缺乏著作權聲明並公開散布時，如符合下列情況，並不當然失去其著作權：(1)缺乏著作權聲明的數量佔所有公開散布數量的一小部分。(2)公開散布之前或之後五年內，正式登記其著作權，並在發現缺乏著作權聲明之後，已盡合理的努力加以補充。

碼加以散布，且散布的數量不在少數，加上著作權登記日期已超過五年期限，後續未盡合理的努力在原始碼上補充著作權聲明，因此，依據 17 U.S.C. §405(a)，原告並未取得有效的著作權。接著，法院進一步探討普通法中的「有限出版說」。

在「有限出版說」下，創作人公佈作品，如果是「在特定的目的下針對選定的族群進行交流，而不涉及散布、重製、銷售等權利」，則未喪失著作權的保護⁷¹，例如，在同儕間流通手稿方便溝通、將施工草圖發給外包商競標參考用等。原告辯稱，32V 系統散布的對象只有教育單位、政府單位及仔細篩選過的企業用戶。前兩者可視為特定族群，但關於企業用戶，雖然採證時原告列出其評選步驟與條件，但法院並不認為，在有限出版說的意義下，這種方式符合原本「選定」的規定。如果對方同意妥善保管取得的智慧財產且具備支付費用的能力，即可成為選定族群中的一員，則實務上租用也可算是「選定」的範圍。1989 年之前的著作權法對缺乏聲明的出版品，在保護上並未賦予如此廣泛的認定，17 U.S.C. §101 明白表示租賃是發行的一種形式，依原告對有限出版說的說法，租賃則不算是公開發行，法院認為，這應該不是當初國會制定著作權法時的原意。因此，原告如果無法進一步證明其擁有 32V 系統的著作權，則侵害著作權的指控將無法成立。

BSD 案中值得思考之處包括：(1)即使原始碼相同，也未必直接等同於著作權的侵害，以 Unix 體系盤根錯節的關係，如果要確認所有權人，可能需要追蹤歷史軌跡以檢驗不同原始碼版本的權利歸屬，對 SCO 案而言，關鍵之一在於當初雙方簽署的契約內容，是否移轉原始碼的所有權？(2)訴訟對延緩科技進展的影響。一般認為，如果不是因為 BSD 系統訴訟纏身，當年 Torvalds 很可能不需要另外開發一套免費的系統核心，BSD 案對資訊界的影響，就某種角度，延緩了資訊科技的普及與進展，迫使有識之士另起爐灶而已。Linux 如今反受訴訟牽連，預計影響層面可能隨訴訟的持續而擴大開來，如此發展，是否可能促使開放原始碼社群重新以 BSD 系統為主力，目前雖無明顯跡象，但值得觀察。

⁷¹ Rexnord, Inc. v. Modern Handling Systems, Inc., 379 F. Supp. 1190, 1197 (D. Del. 1974).

第四項 SCO 質疑開放原始碼社群控管原始碼的嚴謹度

SCO 曾公開對外表示，開放原始碼社群的運作模式有瑕疵，品質上無法確保程式碼的來源是否乾淨。針對這點，Torvalds 表示，當初接受 IBM 貢獻的程式碼時即非常小心，SCO 提到 IBM 將 ”Read Copy Update” 程式流入 Linux 社群，其實該原始碼在 IBM 沒有同時授權使用相關專利之前，有很長一段時間無法併入 Linux 系統。從拒絕 IBM⁷²、IBM 回信澄清經併購後取得專利權⁷³、到 Linus 本人收到確認後才准予放行，可以看出社群管控外來原始碼的嚴謹度，SCO 在別人已申請專利的功能中宣稱含有其營業秘密，更是突顯其指控的薄弱。

Moglen 代表自由軟體基金會指出，參與 GNU 計劃的程式設計師，必須遵守基金會的規定，不得在正職簽署的保密協定中涉及 GNU 計劃下所開發的程式，也不得帶入其他不屬於自由軟體類型的原始碼，尤其跟 Unix 相關的部份，應特別注意⁷⁴。因此，如果 SCO 指控的是 GNU 下的軟體侵害了著作權或營業秘密，以基金會過去運作的經驗來看，成立的可能性不高。不過，由於參與開放原碼運動同時從事全職工作的比例不在少數，將來是否可能出現雇主以涉及職務上創作為由對員工及社群主張權利⁷⁵，則猶待觀察。Moglen 表示，基金會成立以來，在納入他人貢獻的原始碼之前，如貢獻者涉及職務上的創作，即要求其必須與雇主確認放棄日後爭訟的可能，Moglen 並提議隨著這幾年軟體信託管理機制的出現，或許將來自由軟體可以納入保險等配套措施，藉以提高外界對自由軟體與開放模式的信任⁷⁶。

⁷² See Andrea Arcangeli, *Re: 2.4.10pre7aa1*, Sep. 11, 2001, at <http://www.cs.helsinki.fi/linux/linux-kernel/2001-36/0393.html> (visited Apr. 2, 2004).

⁷³ See Dipankar Sarma, *Re: 2.4.10pre7aa1*, Sep. 11, 2001, at <http://www.cs.helsinki.fi/linux/linux-kernel/2001-36/0394.html> (visited Apr. 2, 2004).

⁷⁴ See Eben Moglen, *SCO v. IBM*, Jun. 12, 2003, at http://www.linuxuser.co.uk/articles/lud_special-free_software_matters.html (visited Apr. 2, 2004).

⁷⁵ Eben Moglen 於 2004 年 2 月 23 日應邀至哈佛商學院演講，在場學生提出此問題。演講題目為：SCO and After SCO: The Legal Future of Free Software，錄影資料參見 http://media.law.harvard.edu:8888/ramgen/encoder/2004-02-23_ae_0630-0830.rm (visited Apr. 2, 2004).

⁷⁶ See MadMax, *Eben Moglen's Harvard Speech - The Transcript*, Feb. 26, 2004, at <http://www.groklaw.net/article.php?story=20040226003735733> (visited Apr. 2, 2004).

第三節 開放原始碼社群的立場與回應

第一項 自由軟體基金會

2004 年一月，自由軟體基金會招開記者會，公開駁斥 SCO 對 GPL 及基金會的不實陳述。自由軟體基金會認為 SCO 的陳述不僅毫無根據，其本身繼續透過 GPL 散布 Linux 系統而獲利，如果真有專屬的營業秘密或智慧財產權，SCO 何以繼續以公眾可複製、修改、再散布的方式公開自己的原始碼⁷⁷？2003 年八月 SCO 遭於輿論壓力首次提出疑遭侵害的原始碼細節，不到幾個小時，開放原始碼社群即辨別 SCO 指控是否可能成立，而分析結果顯示 SCO 的蒐證能力可能頗有問題⁷⁸。SCO 比對出 Linux 中雷同的原始碼，其實是同一位設計師貢獻到不同社群的結果，其早期作品經由 BSD 系統進入 System V 系統，SCO 宣稱擁有 System V 而找到相同的原始碼，卻忽略了早期原始碼非其所有。其他的例子則顯示出 SCO 比對原始碼的技巧薄弱，或者誤判，例如，許多程式碼在缺乏著作權聲明的情況下成為公共財，早已不成立侵權問題⁷⁹。

如果自由軟體基金會創始的 GPL 條款無效，則 SCO 必須說明自己在何種授權情況下散布他人的軟體。根據 GPL 第六款「不得附加任何條款散布原 GPL 保護的創作」，SCO 另外要求繳交權利金的授權聲明，明顯違反 GPL 第六款的規定，因此，根據 GPL 第四款「違反 GPL 的規定自動喪失繼續散布的權利」，SCO 既已缺乏散布他人原始碼的權利卻繼續從事散布行為，則明顯侵害了其他人的著作權。

在許多自由軟體的協助下，SCO 才能將許多程式移植到後來立基的 System V 系統，基金會對此樂見其成，但對 SCO 過河拆橋反過來攻擊自由軟體則深表遺憾。至於，SCO 認為 GPL 違反美國憲法，基金會則無法理解允許他人複製、修改、散布著作將違反憲

⁷⁷ See Free Software Foundation, *Setting the Record Straight: The Free Software Foundation, the General Public License and SCO versus IBM*, at <http://www.fsf.org/press/2004-01-14-record-straight.html> (updated Feb. 11, 2004).

⁷⁸ See Bruce Perens, *Analysis of SCO's Las Vegas Slide Show*, at <http://www.perens.com/SCO/SCOSlideShow.html> (visited Apr. 2, 2004).

⁷⁹ 參見第三項。

法上的什麼權利⁸⁰。如果 SCO 的說法有效，則其他類似 GPL 的授權許可，如 BSD、Apache、MIT/X11 甚至微軟的「分享原始碼方案」(Shared Source)，都可能成為無效的授權模式⁸¹。

自由軟體基金會當年成立 GNU 計劃(GNU is Not Unix)，抱持的便是與 Unix 相異而另闢蹊徑的立場，因此特別提醒程式設計師不要接觸 Unix 陣營的原始碼。著作權保護的是表達而非概念。在不涉及表達層面的原始碼下，GNU 的程式將不會侵害他人的著作權。SCO 直接將 GNU/Linux 與 Unix 類比，顯然忽略了 GNU 計劃所預設的原則。目前 GNU 旗下各種軟體已經普及到許多系統，可與 Linux 以外的其他系統核心結合，即使 SCO 案影響了 Linux 的進展，SCO 也無法阻止數百萬程式設計師在其他平台上繼續創作自由軟體⁸²。

第二項 開放原始碼協會

開放原始碼協會認為⁸³，SCO 市場失利後的掙扎，必須先放在軟體界質變的角度觀察。自 1970 年代中期以來，半導體製造技術遵循莫爾定律，每十八個月，晶片的運算能力便加倍成長，但軟體技術的進展卻始終無法跟上。到了 1990 年代，硬體與軟體的落差加大，隨著軟體應用面複雜度的增加，有識之士終於體認到傳統開發模式的限制，傳統以集中方式開發軟體，一直無法兼顧低出錯率及高複雜度的同步要求，軟體界因此浮現所謂的「軟體危機」。因應軟體危機，部分人士認為，軟體的生產過程需要更加嚴謹，在明確的規則下仔細監控每個環節以提高品質，像汽車的生產線看齊，才能解除軟體危機。不幸的是，這樣的做法並未奏效，出人意料之外的是，解決軟體危機的答案來

⁸⁰ See Eben Moglen, *SCO: Without Fear and Without Research*, Nov. 24, 2003, at <http://www.gnu.org/philosophy/sco/sco-without-fear.html> (visited Apr. 2, 2004).

⁸¹ See Eben Moglen, *SCO Scuttles Sense, Claiming GPL Invalidity*, Aug. 18, 2003, at <http://www.gnu.org/philosophy/sco/sco-preemption.html> (visited Apr. 2, 2004).

⁸² See Richard Stallman, *SCO, GNU and Linux*, at <http://www.gnu.org/philosophy/sco/sco-gnu-linux.html> (updated Sep. 4, 2003).

⁸³ See Eric Raymond, *OSI Position Paper on the SCO-vs.-IBM Complaint*, at <http://www.opensource.org/sco-vs-ibm.html> (updated Aug. 22, 2003).

自於網路串聯的 Unix 社群。自 Unix 發展以來，許多頂尖的程式設計師們逐漸形成一種全然不同於商業界的做法，這些程式設計師以分散式小組的方式，取代大型集中的開發團隊，透過同儕檢閱與開放標準的遵行，達到比強化階層管理與程序控制更好的效果，更重要的是，網路上推崇公開、分享的精神而不藏私，造就了後來開放原始碼運動如滾雪球般的進展。1977 年柏克萊版的 Unix、1983 年 GNU 計劃和 X 視窗團隊，都是 Unix 界廣為人知的範例，等到 1991 年 Torvalds 開發新的系統核心，開放原始碼的傳統已經融入網路文化之中。

開放原始碼發展數年後，在許多方面明顯優於過去封閉式的開發模式，不僅軟體的出錯率降低，成本也跟著下滑，到了 1997 年，許多人恍然大悟，原來這是解決軟體危機的良方。不少優秀的程式設計師投身開放原始碼陣營，隨後 Linux 的崛起、IBM 大力採用、到後來 SCO 前身公司 Caldera 自己也投入其中，無非是大勢所趨，以經濟學家熊彼得的名言「創造性破壞」稱之，亦不為過。

開放原始碼模式與許多產業大者恆大、以經濟規模取勝的經驗剛好相反，一般人在初步理解上難免遲疑或質疑，誤以為軟體業的產值將如 SCO 所公開警告的，將因原始碼的開放而隨風而去。誠然，對於仍身處封閉原始碼陣營的廠商，如 SCO、Microsoft 等堅持保護獲利關鍵的原始碼而反對公開，其立場可視為營運策略下的自然反應，但用另外一種角度，開放原始碼模式不過是市場本身自我修正的一種演化過程，消費者沒有理由不能多一種選擇，讓市場決定最後的贏家。

SCO 對 IBM 興訟之後，開放原始碼協會公開發表聲明，針對 SCO 指控⁸⁴中虛偽不實之處一一澄清，藉此匡正大眾視聽。開放原始碼協會指出，原告 SCO 全名為 SCO Group Inc.，前身是一家 Linux 廠商，稱為 Caldera，併購當年 Santa Cruz Operation 公司伺服器部門之後改名為 Caldera Systems Inc. (10/11/2000)，Santa Cruz Operation 公司放棄 Unix 市場之後改名為 Tarantella Inc. (02/12/2001)，而 Caldera 則於 05/15/2003 再改名為現在的 SCO Group Inc.。為方便討論，必要時以 SCO/Caldera 和 SCO/Tarantella 做為區別，單指

⁸⁴ 訴狀內容請參見 <http://www.sco.com/ibmlawsuit/amendedcomplaintjune16.html> (visited Apr. 2, 2004)。

SCO 時表示此次訴訟原告 SCO Group Inc.。

開放原始碼協會分析 SCO 的起訴書後表示⁸⁵：

- (1) SCO 認為 Unix 是 AT&T 貝爾實驗室所開發的作業系統與相關軟體，廣泛用於各種企業的運算環境。事實上，開放原始碼協會認為對 Unix 採取上述定義有誤導之虞，因為並沒有一套可以單純稱之為 Unix 的系統，而與其他非 Unix 系統明顯區隔開來。技術上來說，Unix 是 Open Group 的註冊商標⁸⁶，用來涵蓋符合該組織技術規格定義下的各種作業系統，由於後續衍生關係複雜，如果不以嚴格符合規格為限，則廣義上資訊界有超過上百個系統可以泛稱為 Unix 系統。SCO 自己有兩套產品，稱之為 UnixWare 與 OpenServer。
- (2) SCO 認為經過一連串的併購，SCO 已取得關於 Unix 的所有權利，任何 Unix 廠商使用或散布 Unix 系統，都需要取得 SCO 的授權。開放原始碼協會認為，事實上 SCO 可能取得權利的來源為 AT&T、Novell 和 SCO/Tarantella，這些公司曾對手上逐漸過時 Unix 版本(System V)放棄部分權利，因此，SCO 爭執的智慧財產權標的，權利已有所減損。例如，從 1993 年 BSD 案和解中可以看出，AT&T 原始碼的衍生著作並沒有太大的侵權問題，而且，當年許多原始碼釋出時採取永久授權，也強化許多廠商使用 Unix 相關系統的合法性。IBM 從 AT&T 手中取得永久授權，後續不含任何權利金，衍生著作的所有權歸 IBM 所有並得對外授權，便是其中的一個例子。另外，成千上萬份 Unix 原始碼流通在外超過三十年，根據這些原始碼主張侵害營業秘密，令人難以理解。
- (3) SCO 認為 Linux 系統根據 Unix 原始碼衍生而來，免費供他人使用，缺乏所有權。事實上，開放原始碼協會認為 Linux 並非 Unix 的複製品，而是網路上自動自發的合作成果。Torvalds 於 1999 年曾表示，Linux 與 Unix 類似，但並非 Unix 的

⁸⁵ See Rob Landley & Eric Raymond, *Halloween IX: It Ain't Necessarily SCO*, at <http://www.opensource.org/halloween/halloween9.php> (updated Aug. 28, 2003).

⁸⁶ See The Open Group, *Who Owns UNIX?*, at <http://www.opengroup.org/comm/press/who-owns-unix.htm> (updated Jen 17, 2003).

衍生版本，不像 BSD 系統的原始碼可追溯出 AT&T 的淵源。為了相容的緣故，Linux 在系統功能介面參考 Unix，但系統核心則從頭寫起，並未觸及 Unix 的原始碼。

(4) SCO 認為包括 IBM 在內的 Unix 經銷商，受契約的限制，不得對外散布或揭露 Unix 專屬的原始碼，例如，將原始碼流入 Linux 社群，同時，由 System V 版本衍生的軟體受到同樣的限制，因此，Linux 是未經授權的 Unix 衍生系統。開放原始碼協會認為，事實上 SCO/Caldera 自己銷售 Linux 系統長達八年，同時加入一些 Linux 相關的開發計畫，經過這麼長的時間，如果以善意違反 GPL 條款抗辯，恐難成立。同時，SCO 一直無法具體指出外流的原始碼為何，也值得非議。

(5) SCO 認為 Unix 系統廣泛用於前一千大企業，故權利受到侵害的損失相當可觀。開放原始碼協會認為，事實上 SCO 自己產品的佔有率並不高，1995 年到 1998 年間投入大量資源開發 UnixWare 系統，市場上反應並不好，SCO 主要的營收來源是低階應用的 OpenServer 產品線。SCO 口中的 Unix 市場以 UnixWare 為主，公司全部產品的整體營收還不到重量級廠商如 Sun、IBM、HP 的十分之一，而其中主要的貢獻也不是來自 Unix。

(6) SCO 認為 Unix 系統完全由 AT&T 貝爾實驗室開發，內部使用成效顯著後，再對外廣泛授權。開放原始碼協會認為，事實上來自 AT&T 之外的貢獻相當可觀，其中某些成員目前也參與 Linux 的開發。當年 AT&T 未販賣 Unix 系統，主要是 1956 年受到反托拉斯法的限制，因此，1973 年首次對外發表 Unix 時，發表的場合是學術研討會。一直到 1984 年分家後，AT&T 才有機會將 Unix 當成商品看待，在這之前，AT&T 缺乏投入 Unix 研發的商業誘因，但在圍繞著 Unix 大量的原始碼，已聚集可觀的人氣，早期開放原始碼模式逐漸浮現，許多人並在加州大學柏克萊分校的協調下貢獻新的原始碼。

(7) SCO 認為 SCO 擁有 System V 版本的 Unix 及衍生的任何著作權。開放原始碼協會認為，除了 BSD 案對主張可能侵害 AT&T 原始碼持保留意見外，AT&T 本身也在 System V 版本中納入一些 BSD、SunOS、Xenix 或其他來源的原始碼，當

初 AT&T 預設衍生著作將歸其所有，因此，部分原始碼並未徵得創作者的同意，逕行刪除他人的著作權後，便直接納入 System V 系統。另外，Novell 也表示當初與 SCO/Tarantella 交易時，並未移轉 System V 系統的著作權，著作權仍歸 Novell 所有。

由上可知，開放原始碼協會認為 SCO 的指控漏洞百出，且多與事實不符，諸多疑點有待法院中逐一澄清。

第三項 USENIX 協會

2004 年二月，SCO 執行長應邀至哈佛法學院演講及發函致國會後，USENIX 協會隨後也公開向國會駁斥 SCO 的說辭。USENIX 在公開信⁸⁷中表示自 1975 年起，該協會致力於促進工程師、科學家、系統管理者等各種專業人員，齊力為電腦運算世界推動技術上的進步。在 SCO 及 Linux 發跡之前，協會就已見證了分享創新成果能為資訊科技所帶來的好處，而非如 SCO 所述，強調保護自己的機密或收取高額的權利金才是進步的動力。SCO 認為 GPL 規定不收權利金，將威脅到美國資訊科技產業，而 SCO 本身卻繼續使用各種開放原始碼軟體，表裡不一十分明顯。SCO 表示，開放原始碼將侵蝕智慧財產權制度的地基，其說法也缺乏根據。在任何智慧財產權相關的法律中，並沒有規定創作者必須向他人收費才能授權，因此，程式設計師應有自行裁量的權利。SCO 另外辯稱開放原始碼可能在法律禁止的情況下，將資訊技術流入敵人手中，其實是自己混淆了出口管制措施與智慧財產權法兩個不同的領域。SCO 唯一可能說對的是，市場上每多一份開放原始碼軟體，潛在就壓縮了一份專屬封閉軟體的生存空間。如果開放原始碼證明是較好的模式，的確，將產生這樣的效果，而長久以來，訴諸市場力量一直是最好的管制，高價的產品刺激低價或優質新產品的出現，形成資本市場運作的基調，SCO 希望緩和開放原始碼帶來的競爭壓力，可以理解，但不是智慧財產權法的目的。

⁸⁷ See USENIX, *An Open Letter from the USENIX Association Rebutting SCO's Position on Open Source Software*, available at <http://www.usenix.org/about/sco.html> (updated Mar. 10, 2004).

第四項 群策群力、抽絲剝繭

Linux 是許多程式設計師以熱情與專業、累積多年建立的成果，在網路上受到廣泛的支持與尊敬，SCO 公開對 Linux 加以指控，不管最後真相如何，就注定引發網路社群的強烈反彈。許多網站上可以看出，討論發言中的情緒性反應相當明顯，不過，在批評的衝動之外，社群也展現了螞蟻雄兵的力量，不僅相關討論熱烈，諸多疑點及專業術語也常常能深入辯證而得以澄清。例如，案發不久，即有法務助理 Pamela Jones 義務成立專屬網站⁸⁸，一邊蒐羅相關法律文件，一邊引發各方討論、追蹤最新案情演變，不少議題在短時間內常常累積上百封回應，充分顯示網路社群關切的程度，至於其他報導資訊動態的新聞網站⁸⁹，累積的討論也相當可觀。在關注案情之外，許多人也付諸行動，例如，SCO 一公布疑遭剽竊的原始碼，資深設計師立刻巨細靡遺地加以分析⁹⁰，SCO 執行長與自由軟體基金會法務長分別應邀至哈佛法學院演講，熱心人士隨即熬夜將演講內容逐字謄出加以公佈，至於整理 Unix 歷史資料⁹¹，呼籲協尋相關證人，則是另外的例子。

第五項 撤銷 SCO 的散布權

對於 SCO 明知 GPL 條款，一方面繼續散布受 GPL 保護的各種軟體，一方面又宣稱 GPL 無效甚至違憲，開放原始碼部分人士以行動抵制。例如，社群成員之一 Insecure.org 於 2004 年二月推出新版軟體時，特別對外點名，SCO 的行為已違反 GPL 第四款的規定，自動喪失繼續散布 GPL 軟體的權利，為了具體回應，Insecure.org 決定撤銷 SCO 繼續散布其軟體的權利，並溯及過去舊版軟體及 SCO 旗下所有產品線，同時，停止對 SCO 系統平台的各種支援⁹²。杯葛 SCO 的舉動是否發揮示範作用，鼓動社群內其他人士的跟



⁸⁸ See Groklaw, at <http://www.groklaw.net> (visited Apr. 2, 2004).

⁸⁹ See, e.g., Slashdot, at <http://www.slashdot.org> (visited Apr. 2, 2004).

⁹⁰ See, e.g., Groklaw, *Signal.h -- Part 2 of Warren Toomey's look at the ABI Files*, Mar. 1, 2004, at <http://www.groklaw.net/article.php?story=2004022923000172> (visited Apr. 2, 2004).

⁹¹ See The Unix Heritage Society, *Unix Timeline*, at <http://www.tuhs.org/Images/unixtimeline.gif> (visited Apr. 2, 2004).

⁹² See Network Mapper, *Free availability of the Nmap Security Scanner version 3.50*, Feb. 20, 2004, at <http://www.insecure.org/stf/Nmap-3.50-Release.html> (visited Apr. 2, 2004).

進，因不同軟體採用授權條款不同，故有待觀察，不過此舉已引起廣泛注意。如 SCO 置 Insecure 的聲明於不顧，繼續散布該軟體，則可能演變成另外一起訴訟，讓 SCO 相關訴訟的爭議更加複雜⁹³。

第六項 補償措施

SCO 曾公開對 GPL 免除擔保加以攻擊，表示使用開放原始碼軟體，用戶自己將負擔額外的風險。SCO 認為自己的智慧財產權不當流入 Linux 系統，有權利也有必要對 Linux 用戶發動訴訟，向 RIAA 看齊。為了降低客戶對 SCO 案的疑慮，Linux 相關廠商開始提供補償方案，為客戶築起一道「訴訟上的防火牆」，以利 Linux 持續推廣。例如，併購 SuSE 之後的 Novell 公司及 Linux 銷售商 HP 與 Sun，各自提出補償條款，佔有率最高的 Red Hat 公司則提出「開放原始碼保證方案」，願意替換掉任何能確定侵害他人著作權的程式碼，讓客戶可以安心在訴訟糾紛外不受波及。此外，Red Hat 發起基金，投入百萬美元，做為 SCO 案中捍衛程式設計師及學術單位的法律費用，Torvalds 本身投效的開放原始碼發展實驗室(OSDL)，也設置獨立基金加入捍衛開放原始碼的行列。SCO 控告的主要對象 IBM，除了資助 OSDL 及 Novell 併購 SuSE 之外，倒沒有特別為客戶提供類似的補償方案⁹⁴。

這些補償方案是出於保留 SCO 勝訴的可能、或者是其商業模式下所必須搭配的策略，可能各有不同考量。例如，HP 提供的方案中要求客戶支付額外費用，保障以 SCO 案影響為限，對 HP 來說，補償方案是客戶需求心理存在的情況下，潛在的另一種收入來源。相反地，Novell 不對客戶另外收費，但表示萬一敗訴，賠償金額將限定在原本產品價格的 125%，總額最高也不得超過一百五十萬美元。Novell 所盤算的是，當初將 Unix 賣給 SCO 時自身仍保有一定的智慧財產權，因此對客戶提供擔保的風險不高。Sun 擔保

⁹³ See Slashdot, *USENIX Responds to SCO; Fyodor Pulls NMap*, Feb. 27, 2004, at <http://slashdot.org/article.pl?sid=04/02/27/1317219&mode=thread> (visited Apr. 2, 2004).

⁹⁴ See David Berlind, *The SCO legal train: Know your options*, Feb. 18, 2004, at http://techupdate.zdnet.com/techupdate/stories/main/SCO_legal_train.html (visited Apr. 1, 2004).

的賠償金額沒有上限，但補償範圍以間接與自己的「Java 桌面系統」有關為限，因此，對客戶的保護程度有限。除了這些抬面上的大廠外，市場上出現一家新的「開放原始碼風險管理」(OSRM)公司⁹⁵，計劃以保險方式提供擔保，來降低採用開放原始碼的風險。OSRM 異於上述補償方案之處，在於涵蓋 SCO 案以外的任何開放原始碼軟體，至於一旦涉訟時能請求的費用，則視事先支付的保險金額而定。初步計算後發現，以 Novell 最高補償一百五十萬美元為例，OSRM 的保險費約四萬五千元，即可取得類似的保障，SCO 目前希望對 Linux 客戶索取每台伺服器每年 149 塊美元，如果有三百台機器，則付給 SCO 的費用跟付給 OSRM 便差不多，但 OSRM 涵蓋的範圍更多、期間更長⁹⁶。

第四節 其他國家之案例

第一項 德國

2004 年四月，德國慕尼黑地方法院針對一個違反 GPL 授權的訴訟，准許原告所聲請的假處分，被告必須停止銷售產品或揭露相對的原始碼。被告產品為無線網路路由器，內部含有原告以 GPL 條款釋出的軟體，被告銷售產品時，並未遵守 GPL 的規定提供原始碼，也沒有附上 GPL 授權條款，經原告提出警告後，被告仍拒絕遵守授權條款，原告於是提起訴訟，要求被告履行 GPL 條款中的所有義務。就目前所知，這個裁定可能是法院首次認定 GPL 有效並強制執行的案例⁹⁷。原告對外表示，在本身的開放原始碼軟體之外，就其所知，類似以硬體含 GPL 軟體但違反授權條款的產品，可能高達 170 種，因此，此次判決有重要的指標意義⁹⁸。

⁹⁵ See Open Source Risk Management, at <http://www.osriskmanagement.com> (visited Apr. 1, 2004); Steven J. Vaughan-Nichols, *Open-Source Insurance Is Arriving*, Mar. 18, 2004, at <http://www.eweek.com/article2/0,1759,1550967,00.asp> (visited Apr. 1, 2004).

⁹⁶ See *supra* note 94.

⁹⁷ See Harald Welte, *netfilter project was granted a preliminary injunction against Sitecom GmbH*, Apr. 14, 2004, at <http://www.netfilter.org/news/2004-04-15-sitecom-gpl.html> (visit May 16, 2004).

⁹⁸ See Stephen Shankland, *GPL gains clout in German legal case*, Apr. 22, 2004, at <http://news.com.com/2100-7344-5198117.html?tag=nefd.hed> (visited May 16, 2004).

第二項 澳洲

就目前所知，SCO 並未在澳洲控告任何公司，但澳洲「競爭及消費者委員會」⁹⁹已收到兩份針對 SCO 而來的訴狀。2003 年七月，澳洲以推動開放原始碼及自由軟體為宗旨的業界團體 OSV (Open Source Victoria)，注意到 SCO 以 Unix System V 原始碼登記註冊著作權，並對 Linux 系統提出諸多不利指控。OSV 認為指控缺乏事實根據，除了呼籲業界收到 SCO 警告函時應審慎評估是否支付權利金，OSV 同時向競爭及消費者委員會遞出訴狀¹⁰⁰，要求委員會正式調查 SCO 一連串的商業行為，是否對澳洲數十萬的 Linux 用戶，已形同不正當的威脅或勒索。同時，OSV 希望 Linux 用戶如果收到警告函，可考慮加入 OSV 的行列，另外向競爭及消費者委員會提出相同的聲請¹⁰¹。

2004 年三月，澳洲西部一家小型系統廠商 CyberKnights，向競爭及消費者委員會控告¹⁰²SCO 在澳洲及紐西蘭的事業單位，表示 SCO 對 Linux 的公開聲明，已導致其實質上的商業損失，且 CyberKnights 寄出三封正式信函試圖溝通，也未能得到妥善的回應。在訴狀中，CyberKnights 請求委員會禁止 SCO 對外宣稱擁有或得以控制 Linux 系統、SCO 必須公開承認採用 GPL 的合法性、SCO 必須賠償 CyberKnights 的損失，及 SCO 如果拒絕遵守委員會的規範，則不得在澳洲進行商業活動或廣告。CyberKnights 並以美國少數已向 SCO 支付授權費用的廠商為例，佐證 SCO 對其他公司的威脅確實存在¹⁰³。起訴律師對外表示，SCO 應受自己過去的行為所拘束，是十分合理的要求。根據 SCO 過去的歷史，曾取得 Unix 授權者應無需另外向 SCO 支付費用，新的 SCOSource 授權方案或許

⁹⁹ ACCC, Australian Competition and Consumer Commission.

¹⁰⁰ See OSV, *OSV Official Complaint About SCO to the ACCC*, available at <http://www.osv.org.au/index.cgi?tid=120> (visited May 17, 2004).

¹⁰¹ See OSV, *OSV Files Complaint against The SCO Group with ACCC*, Jul. 23, 2004, at <http://www.osv.org.au/index.cgi?tid=74> (visited May 17, 2004).

¹⁰² See Complaint from CyberKnights Pty Ltd to Mr Graeme Samuel, Chairman, Australian Competition and Consumer Commission (Mar. 3, 2004), available at <http://www.groklaw.net/pdf/ACCC-complaint-mar2004-noaddr-US.pdf>

¹⁰³ See Pamela Jones, *CyberKnights Runs With the Ball - 2nd Complaint Filed With Australia's ACCC*, Mar 3, 2004, at <http://www.groklaw.net/article.php?story=20040303041041270> (visited May 17, 2004).

還可有機會適用於新的使用者，但應無法向過去取得授權者主張。如果前述論點成立，SCO 受過去行為所拘束，新的使用者其實可以從已有授權的經銷商再取得系統，而無需向 SCO 支付費用。而且，如果 GPL 公眾授權無效，SCO 應該提供效力相同的其他授權，而非出爾反爾，對曾經採用 GPL 發行的軟體加以翻案，如果使用者從 SCO 取得同等效力的授權，則無需支付額外的授權費用。簡言之，不論新舊，使用者都沒有必要向 SCO 支付授權費用。上述主張的基礎與禁反言的法律原則類似，與 SCO 是否為 Unix 原始碼的所有權人，並沒有直接關係，就某個角度來說，SCO 如為所有權人可能更有利於上述主張，因為 SCO 過去的授權將具有法律基礎，將不利其事後改口否定曾賦予他人的權利¹⁰⁴。

第五節 小結

開放原始碼運動的商業影響力日增，2004 年三月舉辦第一次「開放原始碼商業會議」，會中便把 SCO 案視為未來整個運動可能訴訟纏身的開端，具有一定的指標意義。與會律師表示，SCO 案的爭議焦點在於契約及著作權等，但可預見未來，可能出現以專利為主的侵權糾紛¹⁰⁵。在深入探討軟體專利與開放原始碼模式的關係之前，下一章將藉由過去或其他進行中的法律糾紛，討論開放原始碼模式下的其他潛在風險，以便對相關議題能有更廣泛的瞭解。

¹⁰⁴ See Brendan Scott, Esq., *About "The Second Australian Complaint"*, Feb 25, 2004, at <http://www.groklaw.net/article.php?story=20040225094810723> (visited May 17, 2004).

¹⁰⁵ See Matt Hicks, *SCO Just Start of Open-Source Lawsuit Wave*, at <http://www.eweek.com/article2/0,1759,1550914,00.asp> (visited Apr. 1, 2004).

第三章 開放原始碼模式與法律風險

SCO 案所涉及的，主要以 IBM 自己的 AIX 系統及 Linux 系統為主，如果訴訟結果對被告不利，對這兩個系統的用戶衝擊最大，但潛在地，SCO 案可能影響開放原始碼運動本身及成千上萬的軟體開發計劃。以此觀之，受影響的使用者範圍也可能擴大許多，例如，開放原始碼中著名的 Sendmail 軟體負責了網路上約八成的電子郵件、Apache 軟體負責超過一半的 Web 網站等。開放原始碼社群累積的資源有如寶山，受 SCO 案及相關訴訟的影響，使用者目前是否只能暫時觀望？在單純使用之外，如果積極加入開放原始碼社群，是否應考量其他風險？所需要留意的法律議題又有哪些？一般使用者該如何看待？本章希望藉由整理相關案例，評估潛在的風險來源及可能的因應對策，做為實務上是否及如何採行開放原始碼的參考。

第一節 侵害著作權



原始碼受著作權保護，使用他人的原始碼或將自己的原始碼公開，首要的問題在於是否侵害他人的著作權或自己的著作權受到侵害，開放原始碼對一般專屬軟體公司及開放原始碼社群在侵害著作權的議題上，是否各有不同的考量？或者對兩者而言存在相同的疑慮？對於著作權侵害的問題，有哪些實務案例可供參考？實務上又該如何降低使用開放原始碼軟體或採用開放原始碼模式的風險？這些問題是本章一開始希望探討的問題。

首先，軟體開發過程中必須避免侵害他人原始碼的著作權，這一點並非開放原始碼社群特有的問題。原始碼的貢獻來自四面八方時，似乎較難確保原始碼並未侵害他人著作權。專屬軟體陣營即常以這一點理由質疑開放原始碼模式的可行性。不過事實上，開放原始碼的效益普及開來，傳統封閉式軟體同樣面臨原始碼來源的控管問題。例如，2003 年 IBM 併購加拿大一家軟體公司時，發現接收的專屬原始碼中含有數十個開放原始碼

軟體，併購金額隨即下降兩千萬美元¹⁰⁶。對資訊業而言，傳統專屬軟體可能含有開放原始碼的程式，而硬體可以燒入軟體執行特定功能，因此，開放原始碼授權的糾紛除了可能出現在專屬軟體上，也可能出現在硬體中，或者，出現在不同的開放原始碼授權之間。SCO 案把這個問題聚焦在鎂光燈下，讓原始碼的控管逐漸成為實務上不可忽視的議題，在資訊相關產業內，不管是否屬於開放原始碼社群，可能都無法置身事外。

軟體公司在尚未瞭解開放原始碼模式之前，往往對開放原始碼產生許多誤解。例如：一般專屬軟體公司主要顧慮的是自己開發的程式如果沾上開放原始碼，是否必須被迫跟著開放？有些反對開放原始碼的廠商抓著這種心裡將開放原始碼模式貼上「具病毒感染力」的標籤，希望不知者因而躊躇不前。事實上，開放原始碼提供的一種選擇，如果併入外部的原始碼供自己內部使用，則沒有必須跟著開放的問題，但如果涉及再散布，一般公眾授權為了防範搭便車的行為，則要求必須對外公開自己創作改良的部份。至於細節的部份，應視不同的授權條款而定，例如，LGPL 允許其他程式以連結方式加以散布、利用其原始碼，而無須公開自己的原始碼。

原始碼來源的問題並非只是存在於開放陣營與專屬軟體公司之間，即使理念接近的開放原始碼社群中，也可能發生類似的著作權糾紛。實務上曾出現兩個知名的開放原始碼團隊就原始碼疑遭剽竊發生爭執，可資參考。2003 年八月，網站伺服器軟體佔有率最高的 Apache 軟體基金會成立新的開發計劃 Geronimo，吸引不少程式設計師加入，同年十一月，Apache 軟體基金會收到另一家開放原始碼軟體公司 JBoss 的律師函¹⁰⁷，信中表示，JBoss 注意到彼此的原始碼有多處雷同，而且之前曾經反映但未得到 Apache 的具體回應。在資訊界，新一代的系統架構有兩大主流，一是微軟主導獨占的.NET 技術，一是 Sun 創立且受大部分廠商支持的 J2EE 技術，JBoss 公司早期即投入 J2EE 陣營，並以

¹⁰⁶ See O'Reilly Developer Weblogs, *Who are the copyright infringers in open source?*, at <http://www.oreillynet.com/pub/wlg/4291#infringers> (visited Apr. 1, 2004).

¹⁰⁷ Letter from TESTA, HURWITZ & THIBEAULT, LLP, to Mr. Greg Stein, Chairman, The Apache Software Foundation (Oct. 31, 2003), available at <http://nagoya.apache.org/eyebrowse/GetAttachment?listName=geronimo-dev@incubator.apache.org&msgId=1119494&attachId=1> (visited May 6, 2004).

開放原始碼模式成功佔有一席之地，Apache 基金會在社群內享有盛名，新成立的 Geronimo 即針對 J2EE 技術而來，希望延續旗下資源再下一城，因而，無可避免地立刻對包括 JBoss 在內的其他廠商形成競爭壓力。

開放原始碼軟體因將原始碼完全公開，不同軟體開發團隊彼此互相學習、模仿的情形就屢見不鮮，而在互相學習中求進步本也就是開放原始碼運動的主要目的之一。但彼此採用的公眾授權條款如果不同，相對地在授權的範圍或效力上也容易出現爭執。Apache 和 JBoss 之間的爭議即肇因於此。JBoss 在律師函中列出程式碼實質相同或類似之處，表示從某些程式碼片段的註解相同可推測 Geronimo 的原始碼應從 JBoss 程式衍生而來，此外，有些原始碼的變數名稱也與 JBoss 相同，而且相關的檔案又在核心範圍內，因此 JBoss 認為最可能的原因便是 Geronimo 直接抄襲 JBoss 原始碼而來。JBoss 的軟體採用限制較少的 LGPL 授權方式，但衍生著作仍須遵守規定繼續使用 LGPL 授權，因此對 JBoss 而言，Apache 基金會採用自己的 ASL (Apache Software License)¹⁰⁸ 散布 Geronimo 而含有其原始碼，已違反 LGPL 條款的規定。LGPL 不允許原始碼以封閉形式改良散布，Apache 的授權雖然仍為開放模式，不抵觸 LGPL 的精神，但 Apache 的 ASL 公眾授權類似 BSD，其限制不多，JBoss 的程式碼如果併入 Geronimo，後續可能採取不同的授權條款釋出，而違反當初 JBoss 採用 LGPL 的用意。JBoss 雖然保留 Apache 善意侵害的可能，但要求 Apache 必須拿出對策妥善處置。Apache 基金會經過密切討論後認為¹⁰⁹，JBoss 指控的一部分原始碼反而來自於 Apache 基金會先前的軟體，因此，應該注意的是 JBoss 而不是 Apache，JBoss 該檢查其是否違反 ASL 公眾授權的規定。至於部份雷同的程式碼則出於同一人之手，程式設計師以不同的授權將同一份作品貢獻到不同的開發計劃並未違法，因為程式設計師未轉讓其所有權。另外，Apache 基金會認為還有一部分的程式碼則毫無關連。

¹⁰⁸ See Apache Software Foundation, *Apache License, Version 2.0*, at <http://www.apache.org/licenses/> (visited Apr. 1, 2004)

¹⁰⁹ See Apache Software Foundation, *Board of Directors Meeting Minutes*, Jan. 21, 2004, at http://www.apache.org/foundation/records/minutes/2004/board_minutes_2004_01_21.txt (visited Apr. 1, 2004).

從以上案情可以看出，同樣在開放理念下，不同團體仍可能發生侵害著作權或違反授權條款的情況，因此，採取開放原始碼模式時，除了留意著作權可能受專屬軟體公司侵害外，對開放原始碼社群的成員也應注意。專屬軟體因原始碼封閉的關係，惡意侵害著作權的比例可能較高，開放原始碼社群因原始碼開放的關係，惡意侵害的比例可能較低，但即使是善意侵害，侵害著作權的結果仍為原始碼所有權人所不欲見。事實上，當專屬原始碼軟體將開放原碼據為私有時，還比較容易進行價值對立的二元判斷，開放原始碼社群往往可以口徑一致對外，但當社群內出現類似 Apache 與 JBoss 的爭議時，往往夾雜著理念細節上的歧異，而較難判定內鬨中的是非，在這種情況下，可能必須求諸於授權條款的細節，並回歸著作權法的規定，以保障原始碼所有人的權益。

法律風險的控管除了提高注意義務外，軟體問題的解答可能回到軟體技術本身¹¹⁰。面對偵測原始碼來源及處理不同公眾授權的難處，2003 年一家新的軟體公司(Black Duck)開始研究如何有效管理內部專屬原始碼與外來開放原始碼的合併或隔離，在這之前，開放原始碼協會創始人 Eric Raymond 曾開發比對工具，協助程式設計師從大海撈針中找出可疑的程式碼，Black Duck 公司所做的，則進一步納入公眾授權的偵測。尤其對許多公司，法務單位對於資訊人員是否採用新的開放原始碼軟體，可能所知不多，如果疏於智慧財產權的風險管理，等到事後出了問題，以軟體工程「越晚發現、越難解決」的原則，再處理可能已緩不濟急。另外，在全球人力成本結構的轉變下，將軟體開發工作外包已成趨勢，如何確保回收的程式沒有侵害他人智慧財產權，一樣是將來無可迴避的問題¹¹¹。據估計，在社群最大集散地 SourceForge.net 上，2004 年初時約有七萬四千個開發專案，註冊人數超過七百七十萬人，每週並以約 2~3% 的速度成長，一週的軟體下載量約三到四百萬次，這些統計尚未包括其他獨當一面的重要網站¹¹²。在公眾授權部分，

¹¹⁰ 學者 Paul Goldstein 在探討著作權演進的書中 (Copyright's Highway)，第六章標題為 The Answer to the Machine Is in the Machine，討論技術問題回到技術本身解決的可能性，在此採取類似觀點。

¹¹¹ See Steven J. Vaughan-Nichols, *Black Duck Hunts Open Source's Legal Pitfalls*, Jan. 16, 2004, at <http://www.eweek.com/article2/0,4149,1442452,00.asp> (visited Apr. 1, 2004).

¹¹² See, e.g., The Linux Kernel Archives, at <http://www.kernel.org> (visited Apr. 1, 2004); Apache Software Foundation , at <http://www.apache.org> (visited Apr. 1, 2004); MySQL, at <http://www.mysql.org> (visited Apr. 1,

約三分之二的軟體專案採用 GPL，其他三分之一可能採用開放原始碼協會認證通過的三十多種授權之一，或者採用其他自創或散見的二十幾種授權¹¹³。如果忽略原始碼的控管，開發軟體時所承擔的風險可能隨著開放原始碼數量的增加而提高。

原始碼的開放，固然增加了抄襲的誘因及引起新的著作權爭議，不過，整體而言，開放模式——從網路到程式碼到其他著作——也會降低剽竊他人智慧財產的難度，因為在一個開放環境裡這種行為比較容易遭人發現。例如，2003 年中一家出版社發現 SCO 未經授權，將其網路上開放但未允許散布的電子書，整章納入 SCO 自己的線上文件庫，SCO 於 2004 年與出版社達成和解¹¹⁴。部分人士對於 SCO 一邊指控他人剽竊其原始碼，一邊剽竊他人的著作，提出不少批評，並認為 SCO 在法院裡的可信度將因此受到影響。同樣的，開放原始碼的軟體一旦發生抄襲或被抄襲的問題，因其原始碼是公開的，是非的斷定也相對較為容易。

最後，以開放原始碼模式開發軟體時除了上述幾項考量外，現有著作權法對軟體的保護範圍也值得注意。軟體的著作權侵害除了針對原始碼，程式設計上的其他要素也可能成為侵害的對象。軟體中可受著作權保護的內容，受不同判例的影響，有寬、窄兩種不同範圍的認定標準。範圍寬鬆者以 *Whelan Associates, Inc. v. Jaslow Dental Laboratory, Inc.*¹¹⁵ 一案為代表，第三巡迴法院認為軟體在原始碼的字義侵害之外，著作權法保護的範圍可擴及其他非字義的部分，例如程式的結構、順序或組織等¹¹⁶。範圍狹窄者以 *Computer Associates International v. Altai, Inc.*¹¹⁷ 一案為代表，第二巡迴法院在該案中採取不同的見解，法院認為進行軟體的著作權侵害判斷時，應該先刪去不受著作權保護的

2004).

¹¹³ See Black Duck Software, *Whitepaper: Solutions for Software IP Risk Management*, at <http://www.blackducksoftware.com/wprequest.php> (visited Mar. 25, 2004).

¹¹⁴ See Don Marti, *SCO Copies from Book, Settles with Publishing Company*, May 11, 2004, at <http://www.linuxjournal.com/article.php?sid=7578&mode=thread&order=0&thold=0> (visited May 14, 2004).

¹¹⁵ 797 F.2d 1222 (3d Cir. 1986).

¹¹⁶ See Patrick K. Bobko, *Open-Source Software and the Demise of Copyright*, 27 RUTGERS COMPUTER & TECH. L.J. 51, 67 (2001).

¹¹⁷ 982 F.2d 693 (1992).

部分，再依整體決定保護的範圍。法院提出將軟體先抽象化、過濾不受保護的部分、再加以比對的三步測試法，以此限縮著作權法保護軟體或需要比對的範圍，避免著作權法過度保護產生類似專利保護的強度，或是造成他人為迴避侵害而不得不重新製作類似的輪子¹¹⁸。對開放原始碼模式，著作權保護範圍採狹義角度限縮，可降低侵權的風險，目前大部分法院傾向採用第二巡迴法院的見解，等於對原始碼以外的侵害從嚴認定¹¹⁹，市場上開放原始碼軟體為了與既有的專屬軟體競爭，在操作介面、資料格式或系統功能上通常以相容為原則，在這種情況下，目前實務上的見解可能並未對開放原始碼社群構成太大的限制。

第二節 授權方式的複雜度

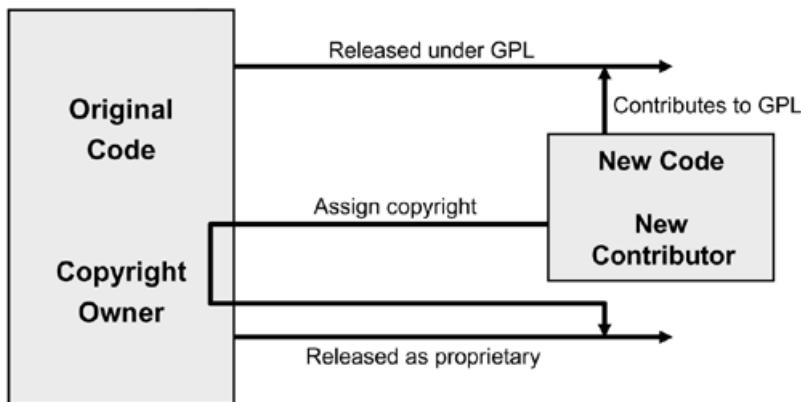
考量是否採用開放原始碼模式時，瞭解公眾授權是不可或缺的一環。如果對公眾授權的規範不夠瞭解，實務上可能存在風險而不自知。公眾授權的運作如何與商業模式搭配讓開放原始碼軟體得以獲利，也日益受到重視。另外，實務上因授權細節的差異引發糾紛的相關案例，在本節中則值得一併瞭解。

首先，雙軌授權模式讓採用開放原始碼模式者能夠獲利，對於開放原始碼軟體能否在商業環境下持續發展特別重要，有必要先提出說明。一般認為，開放原始碼模式在直覺上與既有的獲利模式相違背，因此如何從開放模式中獲利，一直受到懷疑。授權方式對開放模式的重要性除了規範當事人間相關權利義務外，部分原因在於適當的授權策略可能是目前開放原始碼模式最有效的獲利方法。例如，相對於知名度較高的 Linux 廠商如 Red Hat 及 Cygnus 等成功範例證明了透過服務獲利的可行性，有人認為，真正值得注意的反而是像 MySQL 及 Gluecode 之類的公司，在缺乏當年 IPO 水漲船高的協助下，這些公司以雙軌授權方式穩定獲利，才算是通過市場的考驗。雙軌模式對開放社群與商業用途採取差別待遇，讓商業客戶成為主要的收入來源，以開放社群做為擴大市佔率的

¹¹⁸ *Id.*, at 70-71, 90.

¹¹⁹ *Id.*, at 72.

基地，因此，建立了一種既開放又能獲利的商業模式¹²⁰。雙軌授權間的關係可藉由下圖表示：



圖四 雙軌授權模式¹²¹

其次，採用開放原始碼模式者必須注意到授權許可的複雜度可能導因於授權種類的繁多¹²²及條款的演進，實務上發生自由軟體基金會與 Apache 基金會就授權相容性的爭議，可供參考。例如，GPL、LGPL、BSD 及開放原始碼定義下的數十種授權許可，分別可符合各種不同的需求，程式設計師必要時可自創新的公眾授權許可，同一份授權許可隨著時代的演進，也可能出現不同的版本，這些因素提高了選擇授權方式與深入細節上的難度。在實務上，由於 SCO 案提高了開放原始碼社群對智慧財產權議題的警覺度，Apache 軟體基金會於是在 2004 年二月，將 2000 年公布的 1.1 版授權許可更新為 2.0 版，推出新版的目的在於：允許其他軟體專案可以重複使用該授權許可、授權許可得以指到同一份資料而不需要包含在每個檔案中、澄清外來原始碼貢獻時的授權條件、原始碼涉及專利時需取得授權、及標示 Apache 著作權權利歸屬時，可以在授權許可之外以其他檔案表示。整個改版的目的除了簡化部分實務操作面的不便，藉此，Apache 基金會也希望與其他開放原始碼的授權模式相容，並保有 Apache 原本同時幫助非營利組織及商

¹²⁰ See Joe Barr, *Matt Asay introduces Open Source Business Conference*, Mar. 15, 2004, at <http://www.newsforge.com/article.pl?sid=04/03/12/1955258> (visited Mar. 17, 2004).

¹²¹ FINK, *supra* note 11, at 41.

¹²² 關於常見幾種授權的比較，如 GPL、BSD、MPL、NPL、SCSL 等，可參見 Natasha T. Horne, *Open Source Software Licensing: Using Copyright Law to Encourage Free Use*, 17 GA. ST. U. L. REV. 863 (2001).

業團體的理念¹²³。對於此次改版，自由軟體基金會則表示新版的公眾授權與 GPL 並不相容¹²⁴，「相容」的意義是該公眾授權下的軟體得以和 GPL 下的軟體合併構成更大的軟體，而持續符合 GPL 條款的規範¹²⁵。

自由軟體基金會主張 Apache 基金會的授權與自己的授權方式不相容，主要爭點在於 Apache 基金會希望預防他人故意在開放原始碼中埋入專利，但預防的條款對 GPL 而言形同額外的限制，而不能符合相容性的規定。自由軟體基金會認為，ASL 2.0 視為一種自由軟體的授權方式，原則上並沒有問題，但因額外附加 GPL 中所沒有的限制，造成與 GPL 不相容，雖然 ASL 中規定專利權無適用餘地的幾種情況，與自由軟體的精神並無抵觸，但在定義上仍構成相容性的問題。根據 GPL 第二款 b 項，軟體必須以整體方式透過相同的 GPL 條款授權給第三人，GPL 第六款規定，原始碼再散布時，取得原始碼的第三人即取得 GPL 下相同的授權，散布時不得對該第三人加上 GPL 條款中原本所無的限制¹²⁶。在著作權的範圍內，Apache 基金會表示可以理解上述條款的用意，目的在於確保 GPL 下的創作始終維持在 GPL 領域內，但兩個基金會追求的目標不完全相同，Apache 基金會希望專注於社群內協力開發上的支援，對於外來的授權許可，Apache 基金會需要的是確保社群內的程式設計師免於被追訴的風險，同時，Apache 基金會希望盡量擴大原始碼的應用，讓非開放原始碼陣營也有權利善用 Apache 下的原始碼。對於開放原始碼社群，Apache 同樣希望 GPL 規範下的軟體可以自由使用 Apache 授權下的原始碼，這是為什麼 ASL 第四款中規定，使用者可以在自行修改的部分增加著作權聲明，也可以針對修改的部分或對衍生著作的整體，提供額外或不同的授權條款，以供使用、複製或散布。為了確保使用者及貢獻者免於專利訴訟，ASL 第三款中要求明確的專

¹²³ See Apache Software Foundation, *Licenses*, at <http://www.apache.org/licenses/> (visited Apr. 1, 2004).

¹²⁴ See Free Software Foundation, *GPL-Incompatible, Free Software Licenses*, at <http://www.fsf.org/licenses/license-list.html#GPLIncompatibleLicenses> (visited Apr. 1, 2004).

¹²⁵ See Free Software Foundation, *Various Licenses and Comments about Them*, at <http://www.fsf.org/licenses/license-list.html#TOCIntroduction> (visited Apr. 1, 2004).

¹²⁶ See Apache Software Foundation, *Apache License v2.0 and GPL Compatibility*, at <http://www.apache.org/licenses/GPL-compatibility.html> (visited Apr. 1, 2004).

利授權，並附加互惠基礎下的終止條款。在第三款中，程式設計師貢獻原始碼如涉及專利權，則需賦予使用者永久、全球、非獨佔、無償、無授權金、且不可撤回的授權許可，如果使用者對含有該原始碼的軟體發動專利訴訟，宣稱該軟體整體或其中一部份構成直接或輔助性的專利侵權，訴訟發動當日，使用者在 ASL 下被授予的任何權利將立刻終止。

相對地，GPL 第七款規定，如果使用者受限於其他因素無法符合 GPL 的規定，例如，受到法院判決或其他合約的影響，在無法完全遵守 GPL 的情況下，使用者只有停止散布整個軟體一途。以專利授權而言，如果沒有免除權利金的收取，則無法在 GPL 的規範下散布整個軟體。自由軟體基金會認為單以侵害專利權的聲明，尚不構成對再散布的限制，在正式判決或禁制令之前，如果有人宣稱其專利權受到侵害，專利權人仍可散布 GPL 下疑似侵權的軟體。ASL 在專利權人發動訴訟時即終止其權利，故在時間點上比 GPL 嚴格，對 GPL 而言等於增加了額外的限制¹²⁷。亦即，Apache 基金會希望預防有人故意在開放原始碼中使用自己的專利，等到事隔多年之後再對他人追索權利金¹²⁸。

上述 ASL 與 GPL 之間的相容性是一個例子，用來說明不同授權許可之間的交互影響以及可能造成的認知差異。對程式設計師而言，加入開發專案時，需要瞭解所屬授權條款的保障與限制，對使用者而言，不同公眾授權如何降低法律糾紛的風險，則可能影響採用上的考量，對封閉原始碼陣營，主要顧慮則為能否合法納入他人的原始碼，並符合其相關公眾授權的規定。

由於商業力量的加入，開放原始碼運動的法律議題比以前複雜，類似 ASL 與時俱進的情況可能增加，就提升公眾授權的細緻度而言，授權版本的更新有其正面意義，但對一般使用者及程式設計師而言，則可能附帶提高了理解上的難度。Moglen 表示，目前 GPL 是自由軟體基金會 Stallman 於 1991 年時創設，沿用至今，不僅軟體界本身經歷快速的變化，網路的生態也截然不同，例如，最顯著的變化是建立在開放原始碼之上已

¹²⁷ *Id.*

¹²⁸ See David Rubenstein, *Apache Rewrites License*, Feb. 15, 2004, at <http://www.sdtimes.com/news/096/story4.htm> (visited Apr. 1, 2004).

形成可觀的產業，因此，GPL 勢必也需要跟著調整。事實上，Moglen 與 Stallman 正著手研議 GPL 第三版的內容¹²⁹，但改版的複雜度是一大挑戰。GPL 誕生之時，只要說服個體戶和研究人員，讓他們相信公眾授權所帶來的好處，便能推行無礙，如今現實環境大不相同，不僅從獨立的程式設計師到大型公司如 IBM、HP，都可能有不同的意見，開放原始碼運動擴及全球後，對不同國家的法律也需要納入考量。受 SCO 案的影響，大家對於如何在自由軟體下保障應有的權利、原始碼的來源是否有瑕疵等問題，顯得比以往更為關切。過去，一般多半偏重軟體推出後的授權問題，而忽略預防勝於治療的重要性，對於這一點，有人已開始呼籲開放原始碼社群應該多花時間留意軟體推出之前的細節，以正本清源。例如，開放原始碼社群應如何在各方原始碼「組裝」的過程中，透過公眾授權的規範建立防禦性機制，便值得加以探討。如果從市場機制出發，或許透過第三人擔保的模式來降低風險也是一種可能¹³⁰。

在開放原始碼授權的選擇上如有不慎，後續可能引發合作伙伴的離去或遭遇授權轉換的困難。例如，開放原始碼的資料庫廠商 MySQL 在 2004 年三月修改授權條款，希望彌補與另一重要軟體 PHP 之間的裂痕，並平息在授權模式上的爭議。在網際網路上，以 PHP 開發網頁程式搭配 MySQL 是十分常見的組合¹³¹，MySQL 之所以修正授權條款，目的在允許 PHP 恢復以往搭贈 MySQL 元件的作法。PHP 本身同樣屬於開放原始碼軟體，因此此一事件顯示出即使在開放陣營內部，考慮採用何種公眾授權時也不可輕率決定。MySQL 營運策略的特殊之處在於提供兩種授權模式，針對其他開放原始碼軟體，MySQL 採用一般公眾授權模式，針對傳統專屬軟體，則進行商業授權收取權利金。這種雙重授權的作法只有在本身擁有全部原始碼的著作權時才適用，原本 MySQL 以 GPL 涵蓋資料庫軟體，而採限制較少的 LGPL 涵蓋供他人呼叫用的函式庫，後來決定改以

¹²⁹ 就資訊科技的角度，新版授權條款將涵蓋三個領域：(1) Web Service 架構下的線上授權，(2)專利帶來的負面影響，(3) Trusted Computing 概念下以硬體規範軟體執行的潛在影響。

¹³⁰ See David Halperin, *The Future of the GPL*, Mar. 15, 2004,
<http://www.linuxinsider.com/perl/story/33121.html> (visited Apr. 1, 2004).

¹³¹ Linux、Apache、MySQL 加上 PHP 簡稱為 LAMP，是開放原始碼軟體的黃金組合之一。

GPL 規範函式庫的部分時，等於相對增加了新的限制，因而對 PHP 在搭配、散布軟體上造成影響。LGPL 允許其他開放或專屬軟體透過連結的方式使用其軟體，但 GPL 則不允許，如果專屬軟體希望連結 GPL 的程式，本身也必須接受 GPL 的規範。因為 MySQL 將授權許可全部改為 GPL，PHP 於是決定停止搭配 MySQL 元件，以免後續開發的程式必須受到 GPL 的規範而強制開放。因此，PHP 如果繼續搭配 MySQL 使用，將失去過去 LGPL 賦予的部分彈性，而 MySQL 雖然性能優異，在開放原始碼下仍有許多資料庫可供選擇，PHP 團隊於是準備在推出新版時搭配另外一個受限較少的 SQLite 資料庫。

MySQL 之所以需要變更授權方式，當初並非針對 PHP 而來，主要是因為有些專屬軟體公司辯稱，透過 LGPL 授權的函式庫存取 MySQL 資料庫並未違反授權規定。MySQL 公司發現，部分專屬軟體公司以此將自己的產品與 MySQL 資料庫緊密結合，而不願在搭售 MySQL 系統時支付商業授權下應該支付的權利金，因此，MySQL 公司只好將函式庫的部分也改為 GPL，以杜絕搭便車的行為。這樣的改變雖然確保了權利金的收入，但也波及 PHP 及其他開放原始碼社群的重要軟體，例如，Red Hat 決定不再為客戶更新 MySQL 資料庫的版本，以免增加授權搭配上的額外限制。唯恐失去市場，MySQL 只好再次調整授權方式，在本身擁有全部原始碼的基礎下，MySQL 公司增加了新的「自由軟體與開放原始碼軟體的例外條款」，允許相同理念的軟體能像以往自由搭配 MySQL 的產品，但繼續禁止商業公司迴避權利金的支付¹³²。

從以上事件可以看出授權模式的影響重大，至於採用何種授權，實務上可參考開放原始碼社群人士整理的條款重點並考量自己的需求，再加以決定。如果以 GPL 觀點，常見公眾授權間的相容關係可以由下表簡要看出¹³³。對函式庫(library)的使用不論技術上是靜態連結(static link)或使用共用函式庫(shared library)，就法律觀點都視為「結合」，而屬於原函式庫的衍生作品。在下表中，「專屬軟體的連結」指封閉原始碼應用程式或

¹³² See Stephen Shankland, *MySQL addresses open-source license problem*, Mar. 12, 2004, at <http://news.com.com/2100-7344-5173014.html> (visited Apr. 1, 2004).

¹³³ See Zack Rusin, *Open Source Licenses*, at http://developer.kde.org/documentation/licensing/licenses_summary.html (visited Apr. 15, 2004).

函式庫連結到開放原始碼授權下的應用程式或函式庫。「散布作品」中的「作品」指的是結合後的軟體含有開放原始碼授權的應用程式或函式庫，「散布更動後的程式碼」指根據公眾授權修改的應用程式或函式庫再加以散布，最後一欄則是是否相容的問題。

授權類型	是否允許專屬軟體的連結？	是否允許散布作品？	是否允許散布更動後的程式碼？	與 GPL 是否相容？
GPL	不允許，連結後的軟體視為整體	如果採用與 GPL 相容的授權，則允許	只有採用 GPL 時允許	是
LGPL	允許，連結的軟體未視為衍生作品的一部份	允許，但需連同 LGPL 函式庫的原始碼一起散布	只有採用 GPL 或 LGPL 時允許	是
BSD	允許	允許	允許	是
Apache	允許	允許	允許，但不得將自己的衍生作品標為 Apache 原創	否
Common Public License	授權條款未定	授權條款未定	只有採用 CPL 或相容授權時允許	否
MIT (X11)	允許	允許	允許	是
Sun Public License	允許	允許	同樣採用 SPL 時允許	是
W3C	允許	允許	允許	是

表二 常見公眾授權間之相容關係

有人認為對於授權的相容性問題，實際運作經驗的結論是最好一開始便採取與 GPL 相容的授權，因為如果與 GPL 不相容，很可能失去社群成員的支持而無法成長，或到

一定規模後各自分道揚鑣。根據 freshmeat 於 2003 年十一月的統計，網站上公開下載的 32,592 個軟體中約百分之七十為 GPL，排名第二為 LGPL 佔 5.29%，第三為 BSD 佔 4.82%，明顯看出 GPL 相容為開放原始碼實務的主流。如果統計另外一個開放原始碼重要集散地 SourceForge.net，或估算 Linux 經銷商 Red Hat 出貨量所含 GPL 原始碼的比例，也可以得到 GPL 影響力最大的結論。因此，一般建議採用開放原始碼模式應盡量走 GPL 路線，如果需要自訂授權模式，也應與 GPL 相容。「相容」在規範上的意義是，不同授權下的原始碼能否直接合併成為一個軟體，讓原始碼更容易流通、整合。

採用 GPL 除了跟隨主流外，另有重要的實益。GPL 代表開放原始碼運動下的核心理念，受到廣泛的支持，如果授權模式不與 GPL 相容，理念上的差異容易增加程式設計師的疑慮，甚至引發團隊的分裂。例如，Unix 領域知名的 XFree86 視窗系統，過去一直在 MIT/X 公眾授權下廣為流傳，與 GPL 相容，其後團隊領導者為了幫助開發人員累積名聲，決定增加額外條款，而導致與 GPL 不相容。核心成員對修改授權一事意見紛歧，部分人士即使未必完全擁護 GPL 的理念，也強烈質疑引發相容性問題是否值得。Stallman 及其他團體或公司如 Red Hat、Debian、Mandrake 和 OpenBSD 等，皆公開表示關切或反對，其中有人認為新授權讓原始碼較無法自由流通，有人則認為 GPL 相容性應該是首要考量，即使出現這些不同的聲音，XFree86 領導者仍維持修改授權條款的決定。決定之後不久，幾家具有代表性的公司及團體決定另起爐灶，成立 X.Org 基金會，正式從 XFree86 分家，並吸引支持 GPL 相容性的程式設計師加入，數月後，X.Org 推出成立後第一版的「X 視窗」系統，並得到諸多經銷大廠的背書與採用，原本的 XFree86 系統則逐漸失去昔日的舞台¹³⁴。

上面例子反映出將 GPL 相容轉換成不相容所面對的風險與挑戰，如果反方向從原本不相容轉換成相容，另外也會遭遇不同的困難。例如，Linux 雜誌上多次獲獎的文字編輯器 vim，經過相關人士長期討論後，才達成朝 GPL 相容前進的共識；瀏覽器 Mozilla

¹³⁴ See David A. Wheeler, *Make Your Open Source Software GPL-Compatible. Or Else.*, Apr. 7, 2004, at <http://www.dwheeler.com/essays/gpl-compatible.html> (visited Apr. 17, 2004).

團隊成員經過複雜的重新授權後，才讓手頭上所有的原始碼及授權條款與 GPL 相容；開放原始碼社群票選代表社群參加電腦展的 Plone 系統，也花了許多時間脫離第一版不相容的授權；加州大學柏克萊分校原本的 BSD 授權，由於附加特殊的廣告條款而與 GPL 不相容，也在 1999 年略加調整後與 GPL 相容。有了這些經驗，專家建議不如一開始採用 GPL 相容授權，如果之後才轉換成相容模式並非不可能，但重新要求原作者一一授權，往往曠日廢時，尤其軟體專案規模越大，執行上越困難，找不到原作者或原作者不同意時，另行改寫也成了不小的負擔。凡此種種，都是一開始採取相容策略的理由。歸納而言，發起開放原始碼專案時有三種較佳的選擇，第一是直接採用 GPL 條款，第二是採取與 GPL 相容的授權模式，第三則採取雙軌授權，相同的原始碼一邊採用 GPL 相容授權，一邊採用特殊需求下的授權（如商業考量），供使用者自行選擇授權模式¹³⁵。

原始碼所有人授權後如果反悔，由於本身為著作所有權人，自得終止 GPL 下的對外授權。他人根據 GPL 授權所創作的衍生著作，以美國著作權法§103(b)為例，衍生著作部分的所有權不受影響，但來源著作既失去授權依據，他人自然不得再散布原始著作的部分，衍生程式如果需要使用該部分的功能，則有重寫該部分的必要¹³⁶。不過，如果對授權方式事後改變心意，實務上似乎只有改採雙軌授權的例子，而沒有完全從開放原始碼模式退出造成他人衍生程式出現重大缺口的情況，這一點除了創作者當初認同公眾授權的理念，而較少改變外，創作者通常可考慮從下一版本才開始採用其他的授權模式。社群中重要的原始碼如果出現缺口，以網路上豐沛的人力，填補亦非難事，原創作者將作品留在開放原始碼社群中還有掛名的價值，因此，完全撤銷授權似乎未見實益，這是或許實務上較少發生的原因。

有鑑於公眾授權在開放原始碼運動中肩負防弊興利的關鍵角色，學者分析後發現¹³⁷，現行的公眾授權以 GPL 及 BSD 為例，仍存在許多改進空間，並建議以制訂產業標

¹³⁵ *Id.*

¹³⁶ See David McGowan, *Legal Implications of Open-Source Software*, 2001 U. ILL. L. REV. 241, 299-300 (2001).

¹³⁷ See generally Robert W. Gomulkiewicz, *De-Bugging Open Source Software Licensing*, 64 U. PITTS. L. REV. 75

準的方式成立專責機構主導公眾授權的增修。開放原始碼社群中善於除錯(debug)的程式設計師比比皆是，雖然軟體臭蟲在所難免，但常有人加以修正，相對地，公眾授權也存在一些漏洞，需要加以「除錯」，而這部分卻一直受到忽略。例如，部分 GPL 及 BSD 條款定義不明或容有爭議，前者長期未加修訂，後者調整的步調緩慢，在資訊科技快速變化及開放原始碼應用擴大下，授權條款實有增修的必要。以 GPL 首款為例，條款中規定「本授權不涵蓋複製、散布及修改以外的行為」，對於軟體必須「執行」才發揮作用的特性，卻漏未規定，雖然從其他條款的文義解釋中，可以推知授權範圍應涵蓋「執行」在內，但以重要性而言，應該明確羅列為佳。另外，由於 GPL 不提供任何擔保，如果條款未明確列出「執行」一項，以美國為例，軟體執行錯誤的責任負擔可能須遵守統一商法典(Uniform Commercial Code)第二篇關於默示擔保及損害賠償的預設規定，而失去 GPL 當初免除程式設計師瑕疵擔保責任的效果。BSD 授權的情況類似，就契約的嚴謹度亦有值得推敲之處，例如，條款中賦予他人「使用」及「再散布」的權利，但未明白表示賦予「修改」原始碼的權利，他人如果無法修改原始碼，授權範圍未免大打折扣。

雖然開放原始碼協會訂出「開放原始碼定義」的十項標準，自由軟體基金會也致力於 GPL 的推廣與落實，但為使開放原始碼運動在法律議題上長治久安，設立專責機構制訂產業標準來改善公眾授權，仍有其必要性。藉由專責機構，一來可以聚集法律與資訊兩方的專業人士，二來可以定期進行授權條款的增修，符合產業的變化與需求¹³⁸。藉由專責機構的設立，訂定相關運作程序，除了有助於公眾授權實體內容的進步，在追蹤違反公眾授權的侵權糾紛及損害賠償的執行上，也能發揮主動出擊的作用，以建立開放原始碼運動下的秩序。

第三節 授權合約的效力

(2002).

¹³⁸ *Id.*, at 100-103.

GPL 以片面方式宣告的公眾授權是否構成有效的契約？過去認為並不特別重要，因為即使公眾授權無效，原始碼基本上仍受著作權法的保護。GPL 賦予下游使用者有權使用軟體，如果使用者挑戰 GPL 的效力，自己很可能失去權源，反而落入侵害他人著作權的困境，因此，使用者應該希望看到 GPL 有效而得到使用授權¹³⁹，這可能也是長久以來 GPL 的效力未受質疑的原因。此次 SCO 案夾雜其他複雜的因素，打破了過去的推論。

就契約的形式而言，GPL 片面宣告的方式類似於一般購買軟體時常見的拆封合約。拆封合約的有效性建立在買賣雙方事先知情的基礎上，雙方在合理的情況下知道付款後才取得契約的內容，因此能形成有效的契約。至於其中的條款是否顯失公平則是另外一回事。就契約的形成而言，彼此知道拆封條款的存在，即構成一種合意的基礎，因此，拆封合約的重點在於標示是否充分？對方是否知情？如果買方在締結買賣契約時已經知道必須接受其後的條款，取得事後條款時又未加反對，則事後的條款將視為買賣契約的一部份。一般如果必要，買方可以要求事先檢閱條款的內容以求確保自己的權益¹⁴⁰。

不過，在 Step-Saver Data Systems, Inc. v. Wyse Technology¹⁴¹一案中，拆封條款並未免除瑕疵擔保的責任。原告是被告的加值型經銷商，因被告提供的軟體含有瑕疵而要求賠償，被告當初在拆封授權書中免除任何擔保責任，因此，訴訟爭點為該棄權條款是否為雙方合約中的一部份而有效。就事實部分，原告以電話向被告訂購軟體時，曾於電話上表示隨後將寄出採購訂單，在訂單中記載相關數量、出貨及付款條件，被告則同意寄出產品並附上發票，發票上載明與訂單相同的資料與條款，被告從未在電話上或發票中表示瑕疵擔保責任的免除，因此，原告認為雙方在電話上已締約完成。被告後來出貨時，軟體外盒上的拆封條款實質改變了當初契約的內容，原告認為不能視為契約的一部份。法院同意契約成立於拆封條款之前，並認定原告並未明白表示接受拆封條款，因此，拆封條款不能隨後附加而修改原本的合約內容。關於被告表示，鑑於軟體在性質上難以避

¹³⁹ See McGowan, *supra* note 136, at 289.

¹⁴⁰ *Id.*, at 291.

¹⁴¹ 939 F.2d 91 (3d Cir. 1991).

免毫無臭蟲的情況下，法院如果排除拆封條款的適用，將增加業者的負擔而危及軟體業的生存，法院則表示不同意，法院認為，重點在於當事人是否在契約成立之前知道棄權條款的存在，還是該條款於契約成立後才提出¹⁴²。對於 GPL 授權，目前尚未出現有人主張事前不知道 GPL 中棄權條款的規定，之後卻因軟體瑕疵而求償的案例。但將 GPL 視為另外一種形式的拆封條款，實務上應注意使用者是否知道公眾授權的存在，以現行使用者自行下載軟體的作法，在程序上似乎有進一步加強的地方。

至於拆封條款相關的其他案例，則可能支持 GPL 具強制力(enforceable)的觀點。在 ProCD v. Zeidenberg¹⁴³一案中，第七巡迴法院認為，軟體外包裝的合約如果明白表示盒內含有額外的條款且為買方所知，且外包裝合約在當初締約範圍之內，則外包裝上的聲明亦為合約的一部份，盒內額外的條款應具有強制力¹⁴⁴。之後，在 Hill v. Gateway 2000¹⁴⁵一案中繼續延伸相同的觀點，並擴大加以認定。在該案中，原告透過電話訂購一套電腦，被告取得產品規格及付款資料後，同意接受訂單，但雙方均未提及交貨時箱內含有額外的條款，外箱上也沒有如 ProCD 案般加以標示，但法院仍認為箱內的條款有效，原因在於，被告由公開廣告中表示對商品提供有限擔保及終身支援，法院認為這一點足夠支持消費者應該知道實際買賣中將含有額外條款的推論，原告事先未加詢問，貨物運抵後才檢查，並持有該商品超過拆封合約上約定的三十天期限，因此，雙方的買賣契約成立。在 M.A. Mortenson Co. v. Timberline Software Corp.¹⁴⁶一案中，賣方的棄權條款成為爭議的焦點，該案與 Step-Saver 案類似，原告向被告訂購軟體，軟體包裝上明白標示授權條款，軟體首次執行時電腦畫面上再顯示一次，不幸軟體含有瑕疵，造成原告執行後在應用上有所損失，因此原告提起訴訟，被告則辯稱已標示棄權條款為雙方合約的一部份，被告應得以免責。法院的主要考量的是，被告在交易過程中，是否得知或可得而知所購

¹⁴² See McGowan, *supra* note 136, at 292.

¹⁴³ 86 F.3d 1447 (7th Cir. 1996).

¹⁴⁴ See McGowan, *supra* note 136, at 293.

¹⁴⁵ 105 F.3d 1147 (7th Cir. 1997).

¹⁴⁶ 998 P.2d 305 (Wash. 2000).

買的軟體將受額外條款的約束，根據相關證據及資料，法院認為原告知情，因此，被告在合約中的棄權條款有效，對方不得主張瑕疵擔保責任，原告因而敗訴。回到 GPL 是否具強制力的問題。參考上述案例，重點之一在於著作所有權人及他人輾轉再授權出去，是否對原始碼標示適當的聲明，讓一般使用者取得原始碼時對 GPL 條款可得而知？關於這一點，GPL 第一款規定「...必須明顯而適當地在每一份原始碼上標示著作權聲明及棄權條款...」，如果 GPL 條款隨同原始碼散布，或者透過連結將條款細節標示在網站上，使用者在可以得知公眾授權條款的情況下，GPL 應具有強制力¹⁴⁷。

如果比照拆封條款辦理仍有疑義，另外可以考慮採用一般專屬軟體常用的「點選式授權合約」(click-wrap licensing agreement)。一般軟體安裝時，多半要求使用者在畫面上閱讀授權協議書，點選按鈕表示同意後，才允許繼續安裝。確認使用者得知授權條款的細節，除非合約內容顯失公平，否則使用者違反授權規定時其抗辯將難以成立。只是，許多開放原始碼下載後即可執行，無需安裝，上述建議未必完全試用，因此，有人建議應該從網站上下載時，即增加「點選式授權合約」的機制，讓 GPL 公眾條款的強制力得以落實¹⁴⁸。不過，網路下載的形式不只一種，有些可能無法像網頁般增加點選的機制，例如 FTP 檔案傳輸，或是下載軟體中包含許多個別的開放原始碼軟體，要求使用者逐一點選「同意」按鈕，可能造成實務上的困擾。對於這些問題，或許需要從技術上努力，以各種檢便可行的方式取得使用者的合意，以確保公眾授權條款事後的強制力¹⁴⁹。

授權時，以限制部分權利做為對方接受合約所承擔的義務，而不要求支付授權費用，最早出現於 1985 年的 GNU Emacs 授權合約中。採取這種作法在當時相當獨特，但受到資訊社群的認同，經建議後，將該授權方式擴大到原本的 Emacs 軟體以外，促成 1989 年第一版 GPL 公眾授權的誕生¹⁵⁰。就契約的實質而言，一般授權可能要求每一份

¹⁴⁷ See McGowan, *supra* note 136, at 294-97.

¹⁴⁸ See Lawrence Rosen, *Rough draft of Open Source click-wrap notice*, Aug. 6, 2002, at <http://newsvac.newsforge.com/newsvac/02/08/06/2219227.shtml?tid=11> (visited May 14, 2004).

¹⁴⁹ See Lawrence Rosen, *Manifestation of Assent*, Dec. 6, 2002, at <http://www.linuxjournal.com/article.php?sid=6409> (visited May 14, 2004).

¹⁵⁰ Sam Williams, *FREE AS IN FREEDOM: RICHARD STALLMAN'S CRUSADE FOR FREE SOFTWARE* 124-125 (2002).

著作支付一定比例的費用，以取得授權的合法性。GPL 公眾授權模式只不過交換的對價與一般契約有別，因此，在契約的效力方面應該不受影響，亦即，GPL 授權條款應屬有效，違反 GPL 的規定即失去授權的基礎，是否侵害原始碼則回歸著作權法的規定¹⁵¹。

在契約的效力上，附帶值得注意的是時效問題。以 GPL 為例，授權條款中並未明訂授權為期多久，其代表永久授權，或是經過一段合理時間後授權將失去效力，則有待將來實務上加以處理¹⁵²。

第四節 擔保條款

GPL 條款中規定授權人得自行提供擔保或補償，但一般程式設計師在無償貢獻原始碼的情況下，多半採取 GPL 預設的免除擔保條款，因此，使用者面對開放原始碼軟體，必須在免費取得與自行負擔風險間衡量。開放原始碼社群宣稱開放原始碼軟體的品質優良，卻又不願意提供擔保，看似矛盾，實則不然，程式設計師追求品質與願意負擔訴訟風險，必須當成兩件不同的事情看待¹⁵³。另外，相較於專屬軟體的授權合約，開放原始碼未提供擔保似乎是一項弱點，不過如果深究，可以發現一般專屬軟體的合約對消費者的保障，實際上可能沒有消費者想像的多，例如，軟體造成損害，其賠償額度以當初購買成本為限，超出金額則不在賠償範圍之內。因此，評估時如考量開放原始碼可以免費取得，實際上缺乏擔保似乎並不影響一般人採用開放原始碼的意願，開放原始碼軟體出現瑕疵，使用者除了回報問題並寄望下一版改良外，很少聽到因瑕疵造成損害而引發糾紛。

¹⁵¹ See Ben Kremer, *Open source software: What is it and how does it work?*, Feb. 17, 2004, at <http://www.freehills.com/CA256AD900137BAA/page/Listing-ait-Open+source+software:+What+is+it+and+how+does+it+work%3F00201A46?opendocument&l=50-Publications~&2=~&3=~&REFUNID=~undefined> (visited May 16, 2004).

¹⁵² See Dennis M. Kennedy, *A Primer on Open Source Licensing Legal Issues: Copyright, Copyleft and Copyfuture*, 20 ST. LOUIS U. PUB. L. REV. 345, 373 (2001).

¹⁵³ See Robert W. Gomulkiewicz, *How Copyleft Uses License Rights to Succeed in the Open Source Software*, 36 Hous. L. Rev. 179, 192 (1999).

在 SCO 案中，缺乏擔保條款成了 SCO 質疑開放原始碼模式可靠度的一項論點。在 SCO 案之前，是否採用開放原始碼軟體，一般考量的是軟體的成熟度、是否符合需求、後續維護的成本等，較少顧慮原始碼是否侵害他人智慧財產權或埋入不良的功能，而可能引發後續的糾紛，畢竟原始碼置於光天化日之下，如果蓄意侵害他人權利，事跡敗露可能只是遲早而已。但受到 SCO 案的影響，開放原始碼軟體的權利瑕疵擔保，逐漸受到重視而成為是否採用的參考。針對這個問題，如前所述，部分採用開放原始碼的商業團體開始提供不同程度的擔保，以消除客戶的疑慮。

要減少外界對開放原始碼缺乏擔保的疑慮，或許在商業策略之外的另外一種可能是，在公眾授權契約中要求程式設計師擔保其創作未侵害他人著作權。程式設計師如果本身為程式的創作者，在創作主義下即擁有著作權，擔保未侵害他人著作權應沒有太大負擔，因為創作前並沒有類似專利檢索的查證義務。不過，考量無償貢獻原始碼卻需要負擔擔保責任，仍可能降低採取開放原始碼模式的意願，因此，已知的公眾授權並未如上所述，要求創作者負擔瑕疵擔保責任，甚且，將軟體專利納入考慮，要求創作者擔保未侵害他人專利，實務上則可能更是難以實行。考量這些因素，可預期開放原始碼社群對軟體的擔保或補償，未來可能仍以商業團體為主，而較難要求大部分獨立無償的程式設計師負擔，商業團體保留部分獲利做為補償損害或定紛止爭的資源，似乎是比較可行的作法。使用者面對缺乏商業團體擔保的軟體，事實上也應該不用特別顧慮，對重量級的開放原始碼軟體來說，創作者為提高市場佔有率或自己的知名度，自然會負擔應有的注意義務，審慎遵守智慧財產權法的相關規定。例如，SCO 案發生後，Torvalds 思考如何讓 Linux 系統防範於未然、避免受到下一個 SCO 攻擊時，曾建議開發團隊及相關人士採取簽名背書的作法，在經手的原始碼上寫下自己的大名，為原始碼的流通與版本的更迭留下軌跡，如此一來，不僅多年後如果涉訟其舉證相對容易，串連每個人都有一些信得過的朋友的信任關係，也將築起一道阻擋不實指控的強韌防線，如果落實這種作法，將來一般使用者拿到原始碼軟體，不僅信任程度可能提高，對開放原始碼模式的運

作機制也將更加瞭解¹⁵⁴。巧合的是，歷史似乎總是重複出現，當年 BSD 開發團隊與 AT&T 之間爭訟的原因之一，即在於原始碼的權利歸屬不清，軟體開發程序缺乏適當的紀錄，以致於雙方最後必須以訴訟方式逐一檢驗權利的歸屬，相對地，Linux 系統過去雖然並非沒有相關記錄，但仍有許多有待補強的地方¹⁵⁵。

再從法制面分析，即使目前缺乏擔保條款在實務上，似乎受到程式設計師與使用者兩邊所接受，而維持公眾授權的現況也沒有太大問題。不過隨著資訊相關法制的變革，可能逐漸浮現不同的衝突或考量。以美國「統一電腦資訊交易法」(UCITA, Uniform Computer Information Transactions Act)為例，如果法案由統一法的層次逐漸落實到各州州法，以其「電腦資訊」定義的範圍廣義，開放原始碼的公眾授權應涵蓋在內，則開放原始碼的實際運作將受到 UCITA 的規範。UCITA 基本精神之一在於彌補一般軟體授權的不足，在規範有所缺漏的情況下納入契約中應有的預設條款，因此，如果將 UCITA 適用到開放原始碼的公眾授權，預設條款可能有違開放社群的本意，例如，UCITA 可能對授權期間、擔保範圍、默示擔保或其他限制有所規定，讓目前創作者在缺乏擔保條款下仍負擔部分責任。以 BSD 授權為例，授權條款規定包括直接損害在內的任何侵害，都不在擔保的範圍之內。由於直接損害對契約的成立有根本上的重要性，如果預先免除，可能會讓法院對公眾授權的效力產生不同的認定¹⁵⁶。不過，UCITA 並非完全對開放原始碼模式不利，其中仍可能產生正面的效果，例如，贊成使用者未簽名的拆封合約有效，將有助於支持公眾授權契約的強制力¹⁵⁷。

契約中提供擔保條款，目的之一在於降低交易風險，並促成交易，軟體授權異於其他一般有體財產的交易之處，在於交易客體本質上難以確保毫無瑕疵。軟體需要以人工

¹⁵⁴ See Linus Torvalds, *RFD: Explicitly documenting patch submission*, May 24, 2004, at <http://www.newsforge.com/article.pl?sid=04/05/24/1214237> (visited May 26, 2004); Jay Lyman, *Linux kernel developers to log their trust*, May 24, 2004, at <http://www.newsforge.com/article.pl?sid=04/05/24/1555237> (visited May 26, 2004).

¹⁵⁵ See Jan Stafford, *Usenix president: Linux needs better paper trail*, May 24, 2004, at http://searchenterpriselinux.techtarget.com/qna/0,289202,sid39_gci966588,00.html (visited May 26, 2004).

¹⁵⁶ See Kennedy, *supra* note 152, at 374.

¹⁵⁷ *Id.*, at 369.

撰寫，細微的邏輯錯誤可能當掉整個程式，以及軟體功能不斷增加所帶來的複雜度，都是無法保證軟體百分之百沒有臭蟲的原因，即使大型軟體公司具備各種資源及理由將瑕疵降至最低，也盡量只提供最低限度的擔保，便可看出臭蟲瑕疵像是軟體本質上揮之不去的不定時炸彈，誰也難以保證到底。如前所述，使用者必須在購買專屬軟體取得少量的擔保及缺乏擔保但免費的開放原始碼軟體之間選擇，自行評估承受多少風險以及付出多少對價來降低風險，相對地，程式設計師則必須評估需要取得多少對價以提供風險的擔保，在雙方評估之間是風險的配置問題。開放原始碼社群既然缺乏對價填補承擔風險所需要的付出，採取拒絕擔保將風險完全轉嫁到使用者身上，便不難理解，加上使用者自己有選擇接受與否的自由，不提供擔保的正當性又提高一些。軟體瑕疵的風險在雙方你情我願的情況下，其分配或許可以自然形成一種的平衡，例如，維繫企業每天正常運作者採專屬軟體，非關鍵性應用則採開放原始碼軟體等。因此，類似 UCITA 法律介入的風險即在於破壞上述的平衡，雖然立意良好，但可能在降低交易風險的同時，卻無助於交易的促成。



本文認為，促進市場交易是目的，降低交易風險只是一種可能的手段，過去在買賣雙向對價關係中所形成的擔保概念，不必然可以直接適用於開放原始碼模式的單向授權關係，法律應回到最初規範的目的，思考過去的手段是否在新的情境下依舊有益於目的的達成，才能類推適用。在防弊的思維下，把風險擺在使用者這一端，可能有違消費者保護的精神，把風險擺在程式設計師那一端，則可能壓制了創作後分享的意願，看起來思考上沒有出路。實則，在防弊之外或許可以朝興利的另外一個方向思考，例如，開放原始碼模式的特性之一為榮譽制度，許多程式設計師貢獻心力為的是建立個人的名聲、擴大影響力，並得到同儕的肯定會或追隨，在這種情況下，後續是否寄望經濟上的利益在所不問，其個人名聲的累積已成為追求的目標，內在已具有足夠的動機加以經營、維護。能夠反映個人名聲的因素包括社群參與的程度、原始碼創作的品質、數量、重要性等，如果制度運作上能凸顯出這些貢獻度，讓個人的名聲與開放原始碼軟體的優劣產生連結，則軟體瑕疵可能自然地在另外一種誘因下降低。在榮譽制度的強化下，使用者依舊承擔使用軟體的風險，創作者也沒有提供額外的擔保，但因為找出適當的誘因提升軟

體的品質，開放原始碼模式下的交易風險將有機會降低，同時有益於交易的促成。或許可以說，程式設計師採取姜太公釣魚的模式，在使用者願者上鉤的交易中並非一無所獲，其取得的對價其實是名聲的累積而非經濟上的收入，因此，制度的設計如果刺激正確的誘因，例如，建立百大優質開放原始碼軟體的榮譽制度或獎金制度，程式設計師可能自然願意不斷提升魚餌的品質。

第五節 他人違反公眾授權

理論上，開放後的原始碼受到公眾授權條款的保護，但因為後續原始碼由誰下載？下載後是否加以修改、散布？他人是否遵守授權條款的規定？就實務上，並沒有任何機制可以隨時偵測或追蹤違反授權的發生，因此，許多程式設計師對開放模式的運作抱有疑慮，部分企業用戶則擔心來源軟體隱含權利瑕疵而小心迴避。相較之下，專屬軟體只散布執行用的「目的碼」，且多半在使用授權書中強調原始碼為營業秘密，不得對目的碼進行任何還原工程分析原始碼，因此，專屬軟體在規範所有權人的授權範圍方面，似乎比較容易落實。開放原始碼社群對於他人侵權行為是否缺乏著力點，而明顯劣於封閉原始碼模式？在形成結論之前，除了前面提過 MySQL 事件¹⁵⁸外，還有幾個案例值得參考。

2001 年九月，自由軟體基金會發表聲明指出，FSMLabs 公司試圖在 GPL 軟體上附加專利限制，明顯違反 GPL 條款的規定。FSMLabs 產品(RTLinux)為一種即時作業系統，其中含有已審核通過的專利技術，FSMLabs 雖採用不同的授權條款，但仍使用 GPL 原始碼，因此引起自由軟體基金會的注意。雙方就爭議事項協調後，FSMLabs 同意他人如果維持在 GPL 條款規範下，可採用另外一種授權方式，無償使用 RTLinux 的專利技術。但 FSMLabs 做為開放原始碼陣營的一員，取得軟體專利的動作仍引起部分人士的批評，部分人士認為軟體專利不應該存在，遑論開放原始碼成員涉足其中，就這一點，FSMLabs 公司有自己的看法。FSMLabs 表示，該公司對 Linux 系統核心有所貢獻，營運狀況良好，

¹⁵⁸ 參見第二章第二節第一項。

是開放原始碼模式下少數獲利的公司之一，公司發展的策略在於同時開發專屬及開放的軟體，專利的申請是公司重視智慧財產權的證明之一，社群人士應該平心以待¹⁵⁹。

2001 年十月，FSMLabs 與自由軟體基金會達成協議，提出新版的「RTLinux 專利開放授權」¹⁶⁰將符合 GPL 的規定，公司開放原始碼產品 RTLinuxFree 也將持續採用 GPL 授權，以利開放原始碼社群的成長。在「專利開放授權」中，FSMLabs 將美國專利編號 5,995,745 的使用權無償賦予 GPL 軟體使用，自由軟體基金會也希望藉此加速 Linux 在嵌入式系統與即時作業系統的影響力。在這些領域中，專屬軟體廠商派別林立，藉由營業秘密與特殊規格鎖住客戶十分常見，如果自由軟體在其他領域的成功可以帶入這些封閉的領域，無疑是一件好事。經過澄清，自由軟體基金會也發現先前聲明含有誤解的成分，正式加以撤回，並感謝 FSMLabs 多年來對自由軟體的支持¹⁶¹。

2003 年二月，ARM Linux 開發人員發現，專屬系統 RISC OS 5 中含有 GPL 軟體，英國 Castle Technology 公司涉嫌將 Linux 核心程式併入自己的封閉系統，燒到機器的唯讀記憶體中加以販售，但未公布原始碼，違反了 GPL 條款的規定¹⁶²。2003 年七月，業餘玩家測試 Linksys 公司的 WRT54G 無線網路基地台時發現，韌體中的執行碼含有 GPL 軟體，雖然該設備核心原本即以 Linux 為基礎，出現 GPL 軟體本不足為奇，但部分程式明顯遭到修改，修改後的原始碼卻不知公布何處。經詢問後，原廠指出公布在公司的網站，經比對後發現，廠商公布的只是修改前的版本¹⁶³。由於 Linksys 及母公司 Cisco 知名度不低，在網路上特別引人注意，批評聲浪並隨之而來。事後發現，剽竊原始碼的行

¹⁵⁹ See Grant Gross, *RTLinux leader: GPL violation may be a mix-up*, Sep. 17, 2001, at <http://www.newsforge.com/article.pl?sid=01/09/17/1726229> (visited Apr. 17, 2004).

¹⁶⁰ See FSMLabs, *The RTLinux Open Patent License, version 2.0*, Oct. 5, 2001, at http://www.fsmlabs.com/products/rtlinuxpro/rtlinux_patent.html (visited Apr. 17, 2004).

¹⁶¹ See Free Software Foundation, *FSF and FSMLabs come to agreement on a GPL-compliant version of RTLinux Open Patent License*, Oct. 12, 2001, at <http://www.gnu.org/press/2001-09-18-RTLinux.html> (visited Apr. 18, 2004).

¹⁶² See Chris Williams, *RISC OS 5 GPL breach accusations*, Feb. 7, 2003, at <http://www.drobe.co.uk/riscos/artifact556.html> (visited Apr. 18, 2004).

¹⁶³ See michael, *Linksys and the GPL, Again*, Jul. 31, 2003, at <http://developers.slashdot.org/article.pl?sid=03/07/31/1350217&tid=106> (visited Apr. 18, 2004).

為來自委外的程式設計師，Cisco 與 Linksys 雖沒有可直接歸責之處，但相關人士與原廠聯繫時，原廠似乎對 GPL 興趣缺缺。專家指出，在嵌入式系統領域比較容易對 GPL 產生誤解，尤其小公司或委外人員，在缺乏法務資源下，技術人員對公眾授權的細節可能一知半解，以為原始碼修改後燒到硬體中，他人無法修改原始碼，便不受 GPL 的約束。

¹⁶⁴ 經過一段時間，Linksys 公司終於將原始碼公布在網路上¹⁶⁵。

2004 年一月，MPlayer 公司對外聲明，表示 KISS Technologies 公司涉嫌在產品中單以執行碼散布 MPlayer 的開放原始碼，在 KISS Technologies 的產品中，執行碼燒錄在唯讀記憶體成為韌體的一部份¹⁶⁶。對此，KISS Technologies 公司除駁斥指控外，並懷疑程式碼如果相同，也可能是從公司外流出去，輾轉而為 MPlayer 軟體所用，加上 GPL 的效力未曾正式在法院中檢驗，KISS Technologies 公司對 GPL 的效力也持保留態度¹⁶⁷。KISS Technologies 公司表示沒有意願採用 GPL 授權，但有興趣者可以自行附上回郵信封索取原始碼。

從上面幾個案例，加上網路上還可找到其他爭議，顯見原始碼開放後的確較難避免他人違反授權規定。不過，改良後的原始碼容易藏私，是否意謂著提高他人剽竊的誘因？可能未必全然如此，原因在於 GPL 原本就允許他人使用原始碼，有問題的部分是對外散布的改良版本，如果改良版本達到相當的知名度，將吸引足夠的目光，剽竊遭到揭發應是指日可待。因此，如果他人故意違反 GPL 的規定，到頭來面對指控與輿論的壓力，往往必須改寫有所爭議的部分，或者全盤加入 GPL 陣營，不管採取哪一種方式，都不如一開始審慎決定來得妥當。

至於是否違反 GPL 條款，自由軟體基金會建議檢查的項目包括：(1) 對方散布的軟

¹⁶⁴ See LinuxDevice.com, *Linksys GPL issues raise embedded concerns*, Oct. 14, 2003, at <http://www.linuxdevices.com/news/NS7719522768.html> (visited Apr. 18, 2004).

¹⁶⁵ See Linksys, *GPL Code Center*, at <http://www.linksys.com/support/gpl.asp> (visited May 14, 2004).

¹⁶⁶ See CowboyNeal, *MPlayer Alleges KISS Technology Violating GPL*, Jan. 3, 2004, at <http://slashdot.org/articles/04/01/03/146209.shtml?tid=117> (visited Apr. 18, 2004).

¹⁶⁷ See simoniker, *Kiss Technology Counters MPlayer GPL Arguments*, Jan. 12, 2004, at <http://slashdot.org/articles/04/01/12/228247.shtml?tid=185&tid=188&tid=97> (visited Apr. 18, 2004).

體是否包含授權書？(2)是否清楚列出授權書涵蓋的軟體範圍？是否故意製造一種程式受 GPL 保護但實際並未涵蓋在內的錯誤印象？(3)散布的軟體是否含有原始碼？(4)如果只包含執行用的二元碼，是否清楚標示可取得原始碼的來源？(5)原始碼是否完整無缺？是否故意連結到非屬自由軟體的模組才能執行？自由軟體基金會表示，如果侵害屬實，當事人可以記下軟體名稱、負責人、發行單位、聯絡資訊、及上述檢查結果，通知自由軟體基金會處理，並知會原始碼所有權人，於涉訟時由其主張權利¹⁶⁸。

自由軟體基金會之所能夠代為主張權利，配套作法是要求創作者將著作權轉讓給基金會，以取得事後要求侵權者遵守 GPL 的權利基礎，不用每次一一尋找原作者。其他的開放原始碼授權方式通常沒有採取這種作法，因此，在權利侵害的救濟上可能較不明確，也缺乏類似自由軟體基金會的機構代為主張¹⁶⁹。

第六節 商標

嚴格說來，商標並未因開放模式而有所差別，但與開放原始碼社群有關的兩個案例，仍值得一提。首先是 Lindows 與 Microsoft 兩家公司關於「Windows」商標的訴訟。Lindows 公司成立於 2001 年，以 Linux 為基礎，致力於開發更加親和好用的系統，並以低價方式在市場上直接與 Windows 競爭，推出的產品稱為 LindowsOS。2004 年四月美國西雅圖地方法院裁定，駁回 Lindows 要求 Microsoft 在歐洲訴訟必須等到美國訴訟結束後才能繼續進行的聲請。Microsoft 認為，Lindows 名稱明顯淡化 Windows 商標，致公眾於混淆，於是在歐洲七個國家同時提起訴訟，並先後於芬蘭、瑞典及荷蘭取得勝訴¹⁷⁰。商標訴訟延伸到歐洲之前，事實上，Microsoft 於 2001 年十二月，在美國西雅圖地方法院對 Lindows 公司提起訴訟，雙方就商標是否受到淡化及原商標的有效性進行攻防。

¹⁶⁸ See Free Software Foundation, *Violations of the GPL, LGPL, and GFDL*, at <http://www.gnu.org/licenses/gpl-violation.html> (visited Apr. 18, 2004).

¹⁶⁹ See Kennedy, *supra* note 152, at 371.

¹⁷⁰ See Pham-Duy Nguyen, *Judge rules Microsoft can sue Lindows.com*, Apr. 3, 2004, at <http://www.oaklandtribune.com/Stories/0,1413,82~10834~2060749,00.html> (visited Apr. 12, 2004).

Microsoft 向法院要求 Lindows 必須先停止繼續使用該名稱，遭法院駁回，因此，Lindows 得以在美國繼續使用其商標。法院認為，如果一個字被認定為普通名詞(generic)，則恆為普通名詞，不受後續市場行銷所左右，亦與投資金額無關。關於 Windows 在 1985 年使用時是否為普通名詞，則有待陪審團認定¹⁷¹。由於 Windows 系統是 Microsoft 的主要命脈，商標權如果無效，顯然非同小可，因此，Microsoft 不惜動用各種資源，在世界各地對 Lindows 舉訟。Lindows 預期後續訴訟將進入第九巡迴上訴法院，如果判定 Windows 為普通名詞，Microsoft 可能再上訴到最高法院¹⁷²。

法院在駁回 Lindows 聲請簡易判決時表示，就事實部分，Microsoft 在 1980 年代初期進行「圖形使用介面」的開發，其中特色之一為「視窗」介面的使用，但毫無疑問地，1985 年推出 Windows 1.0 版之前，已有幾家公司發展出畫面視窗可以重疊的操作方式。根據 1993 年二月美國專利商標局的回覆意見，Windows 一詞廣泛用於公眾及相關產業，用來指涉特定類別的物品或軟體，邏輯上看來，用「視窗」(window)和「視窗化」(windowing)一詞表示圖形作業環境的一種，似乎言之成理。如果一個詞彙為普通名詞，即使廠商賦予第二種意義，也無法將普通名詞轉換為商標。在本案中，Lindows 公司提出許多資料，試圖證明 1980 年代初期，一般電腦使用者把「視窗」特性當成圖形作業環境的代名詞，而 Windows 系統只是其中一種，Microsoft 對此則提出相反的證據反駁。整個案子的重心落在事實審，陪審團需決定「Windows」一詞是否為通稱。

在歐洲的訴訟中，法國法院駁回 Microsoft 的禁制令申請，對 Lindows 有利，但 Lindows 公司及經銷商在荷蘭則面臨法院裁定每日十萬歐元罰款的壓力，而且涉及多國訴訟對小公司造成龐大負擔。研議之後，Lindows 公司決定維持在美國的公司名稱，但將國際行銷的產品名稱先改為 Linspire。值得注意的是，Lindows 公司原本販賣的

¹⁷¹ See Lindows, *Michael's Minutes*, Apr. 6, 2004, at http://www.lindows.com/lindows_michaelsminutes.php (visited Apr. 12, 2004); Lindows, *MS Law Suit*, at <http://info.lindows.com/lindows> (visited Apr. 12, 2004).

¹⁷² See CIO Today, *Lindows: Microsoft's Latest Demand Impossible*, Mar. 15, 2004, at http://cio-today.newsfactor.com/story.xhtml?story_title=Lindows_Microsoft_s_Latest_Demand_Impossible&story_id=23359 (visited Apr. 12, 2004).

LindowsOS 系統，經過此次事件後，開放一段時間提供網路上輸入折價代碼免費下載，代碼為 lindows，試圖延續舊商標知名度的作法是否引發 Microsoft 的抗議，有待觀察。可以確定的是，Lindows 在美國就「windows」一詞的商標訴訟仍持續進行¹⁷³。

第二個案子為關於 Open Group 機構與 Apple 公司對 Unix 商標權的糾紛。Apple 新一代作業系統係根據 BSD 版本的 Unix 而來，2003 年六月，Open Group 機構控告 Apple 公司行銷新系統 Mac OS X 時，未得到授權卻使用 Unix 商標，明顯侵害該機構的商標權，要求 Apple 應仿照其他公司，繳交每年十萬美元的授權費。Apple 公司隨後提出反訴，認為 Unix 一詞應該看成普通名詞，不得做為商標使用，對一般大眾而言，知道 Unix 商標屬於 Open Group 者微乎其微。此案因雙方都需要較長的前置時間準備、協商，法院同意延後進入訴訟程序的日期，最後的結果尚未可知。但分析指出，此案對 Open Group 的重要性大於 Apple 公司，Apple 如果敗訴，主要是繳交年費或將 Unix 一詞從行銷資料中排除，但判決如果相反，則 Open Group 將失去重要的獲利來源與經營基礎，該機構從 1994 年起即取得 Unix 商標開始對外授權¹⁷⁴。Open Group 除了提供商標授權外，同時負責授權前的認證測試，確保受測系統符合 Unix 規格，對於 Apple 一案，Open Group 表示，對方公開宣稱 Mac OS X 系統以 Unix 為基礎(*Unix-based*)，卻未通過必要的驗證，因此，更凸顯 Open Group 提起訴訟的必要。對於這一點，Apple 則反駁，強調 Unix 有許多衍生派別，其採用的 FreeBSD 雖是 Unix 的一種，但網路上廣為流通，應該沒有送到 Open Group 測試的必要，而且，Apple 並沒有直接將 Unix 當成自家產品的註冊商標，只是用這個業界通用的名詞描述 Mac OS X 系統的一項特徵，與業界常見作法並無相異之處¹⁷⁵。雖然本案雙方當事人與 SCO 案無關，但 Unix 一詞如果判定為普通名詞而無商

¹⁷³ See Steven J. Vaughan-Nichols, *Good-Bye Lindows, Hello Linspire*, Apr. 14, 2004, at <http://www.eweek.com/article2/0,1759,1567656,00.asp> (visited. Apr. 17, 2004).

¹⁷⁴ See Bryan Chaffin, *More On Apple vs. The Open Group In Use Of "Unix"*, Apr. 8, 2004, at <http://www.macobserver.com/article/2004/04/08.8.shtml> (visited Apr. 13, 2004); Gillian Law, *Court grants delay in Apple/Open Group Unix lawsuit*, Jun. 13, 2003, at <http://www.computerweekly.com/Article122619.htm> (visited Apr. 13, 2004).

¹⁷⁵ See Ian Fried, *Apple in court dispute over Unix*, Jun. 11, 2003, at http://news.com.com/2100-1016_3-1015814.html?tag=fd_top (visited Apr. 14, 2004).

標權，SCO 將必須更詳盡指出自己擁有的版本為何，而難以在訴狀中用 Unix 一詞概括表述權利受到侵害的範圍。

第七節 軟體專利

Microsoft 與美國司法部就反托拉斯案達成和解時，承諾公布更多技術文件，讓其他廠商得以在較高的透明度中競爭，並與 Windows 平台互通相容。為此，Microsoft 於 2002 年對外提供部分技術授權方案，但特別將 GPL 排除在外。例如，Samba 為開放原始碼中著名軟體，提供與 Windows 系統共享檔案和印表機的功能，並採行 GPL 授權模式順利運作。Samba 開發團隊過去在 Microsoft 沒有提供技術文件的情況下，成功達到高度的相容性，讓 Linux 與 Windows 之間得以交換資料。但 Microsoft 表示上述技術含有軟體專利，Microsoft 就本身的技術對外授權，允許其他同業免費使用其專利，證明 Microsoft 在反托拉斯案和解的誠意，但 Microsoft 特別表明 GPL 與 LGPL 條款下的軟體，將不在授權的範圍之列。這一點呼應了 Microsoft 將開放原始碼運動視為大敵的立場，也引起開放原始碼社群對軟體專利的關切，因為 GPL 和 LGPL 代表成千上萬的開放原始碼軟體，此次 Samba 軟體受到直接影響，下次是那個軟體則猶未可知。

Samba 團隊分析後指出¹⁷⁶，Microsoft 專利技術並非該系統功能的唯一實作方法，至少，Samba 採取不同的技巧已達到相容的效果，而沒有侵權的疑慮。事實上，該專利保護的技術已經在 Microsoft 後來新版的技術協定中廢棄，因此影響有限。至於 GPL，過去 Samba 靠著 GPL 而成功，未來仍將繼續採用，不受 Microsoft 的影響。不過，有人指出即使如此，技術的主導權仍不屬於開放原始碼社群，雖然 Samba 提供的效能意外超過 Microsoft 自己的產品¹⁷⁷，但 Microsoft 可以搭配類似的授權合約，不斷更新技術並施以專利保護，繼續將 GPL 排除在授權範圍之外，以達到阻礙競爭的目的。另外一個類

¹⁷⁶ See Samba, *CIFS, Microsoft and the Samba Team*, at http://de.samba.org/samba/ms_license.html (visited Apr. 12, 2004).

¹⁷⁷ See Roger Howorth, *Samba beats Windows*, Oct. 13, 2003, at <http://www.vnunet.com/News/1144289> (visited Apr. 12, 2004).

似的例子是 OpenOffice，OpenOffice 以開放原始碼模式直接與 Microsoft Office 競爭。Microsoft 於 2004 年取得歐洲專利局編號 EP1376387 的專利¹⁷⁸，內容為關於以 XML 格式儲存文件的技術，專利通過後引發多方注意，主要原因在於 XML 已是電子資料交換時廣為接受的標準格式，部分國家並考慮規定官方資料必須遵守 XML，確保資料交換的流通性，不受特定廠商以專屬格式箝制。Microsoft 取得 387'專利，顯然讓類似的規定可能流於形式，而無法達到原本開放標準的目的。取得專利後，Microsoft 對於專利技術另外公開授權，授權書中表示，對於遵守 Microsoft Office 規格的存取功能，在清楚標示授權來源後，他人得以無償使用其專利，但如果被授權人對 Microsoft 發動訴訟，Microsoft 有權終止其授權。Microsoft 既申請專利，又願意無償授權，重點也在於排除 GPL 的適用，Microsoft 授權書中明訂 GPL 軟體不在授權之列。競爭對手 OpenOffice 採用雙軌授權，其中公眾授權的部分便是 GPL，而 OpenOffice 一開始採用 XML 做為文件的內部格式，很可能落入'387 專利的範圍，Microsoft 未來是否就此發動專利訴訟，仍有待觀察¹⁷⁹。

在個別案例之外，軟體專利似乎有獨大的可能。2004 年四月，Microsoft 宣布以 4.4 億美元與 InterTrust Technologies 公司和解，取得對方關於多媒體反盜版技術的所有專利，其中多項專利將有助於 Microsoft 加速「數位版權管理」(DRM)系統的研發與應用。InterTrust 於 2001 年提起訴訟，指控 Microsoft 侵害其十一項專利，產品線涵蓋作業系統的各個版本，甚至包括 Xbox 遊戲機在內。Microsoft 和解後，不僅本身取得合法授權，Microsoft 的客戶同樣無需另外取得 InterTrust 的授權¹⁸⁰。差不多同一時間，Microsoft 與長期競爭對手 Sun 公司達成和解，雙方簽署長達十年的協議，Microsoft 支付美金十六億中，九億用來平息過去所有的專利糾紛，並約定不得互相控告，以達到取得交互授權的效果，剩下七億則解決 Sun 公司在美國及歐洲反托拉斯案中的敵對立場。分析家指出，

¹⁷⁸ See EPO, *Word-processing document stored in a single XML file*, at <http://v3.espacenet.com/textdoc?DB=EPODOC&IDX=EP1376387&QPN=EP1376387> (visited Apr. 14, 2004).

¹⁷⁹ See peterh, *Microsoft Justifies XML Patent*, Jan. 25, 2004, at <http://www.nzoss.org.nz/portal/modules.php?name=News&file=article&sid=290> (visited Apr. 14, 2004).

¹⁸⁰ See Alexander Wolfe, *Microsoft Assembles Hefty Patent Arsenal*, Apr. 12, 2004, at <http://www.eweek.com/article2/0,1759,1566393,00.asp> (visited Apr. 14, 2004).

Sun 公司在 Unix 市場逐漸落後，主因之一在於 Linux 系統的崛起，Microsoft 雖然支付可觀的金額，但同時累積專利實力、削弱反托拉斯制裁的正當性、並與昔日對手共同面對 Linux 的挑戰，可謂一石三鳥之計。雖然和解金額不低，但分析家認為，Microsoft 過去半年逐步強化智慧財產權的保護與授權，纏訟在身的二十多件專利糾紛如果逐一清除障礙，將有利於後續以軟體專利大幅對外擴張版圖¹⁸¹。據估計，Microsoft 擁有四千個以上的專利技術，另有五千四百個在申請程序中¹⁸²。

以上初步可以看出軟體專利對開放原始碼社群的潛在影響，Microsoft 只是傳統專屬軟體模式下的主要代表，不表示軟體專利的爭議只侷限於 Microsoft 一家。軟體專利的排他性與開放社群自由分享的機制之間，存在著許多的問題，這部分留待後面以專章討論。

第八節 其他觀點

在前述各項法律風險之外，對於評估是否採用開放原始碼，還可以從下面幾個角度觀察。

第一項 反方陣營的觀點

反對開放原始碼最力的以 Microsoft 為首，這一點不難理解。2003 年 Microsoft 向美國證管會送交季報時，正式將開放原始碼運動列為公司營運模式的一大威脅，並對運動人士主張公家單位應積極使用開放原始碼軟體表示強烈反對¹⁸³。因此，透過 Microsoft 的觀點，可以採取另外一種角度思考開放原始碼模式的利弊，以便在正反兩方意見中加以評斷。

¹⁸¹ See Stephen Shankland, *Sun settles with Microsoft, announces layoffs*, Apr. 2, 2004, at http://news.com.com/2100-1014_3-5183848.html?tag=st_lh; Matt Hicks, *Sun, Microsoft Promise End to Patent Feud*, Apr. 2, 2004, at <http://www.eweek.com/article2/0,4149,1561360,00.asp?kc=EWRSS03119TX1K0000594> (visited Apr. 14, 2004).

¹⁸² See Mary Jo Foley, Microsoft Watch and Darryl K. Taft, *Microsoft Plays Intellectual Property Licensing Catch-Up*, Dec. 3, 2003, at <http://www.eweek.com/article2/0,1759,1403030,00.asp> (visited Apr. 14, 2004).

¹⁸³ See David Legard, *Microsoft: Open source threatens our business model*, Feb. 5, 2003, at <http://www.computerworld.com/softwaretopics/software/story/0,10801,78203,00.html> (visited Apr. 1, 2004)

Microsoft 在季報中表示，傳統商用軟體公司對消費者更有利，消費者可以信賴一家公司所提供的專業技術，該公司推出的平台也可吸引其他開發人員的加入，共同為消費者創造各種增值軟體，並保有軟硬體的相容性。另外，Microsoft 面對開放原始碼帶來的許多特點，經過消化吸收之後，推出折衷的「分享原始碼方案」¹⁸⁴，在對外公佈的提議中羅列開放原始碼運動的要點以及 Microsoft 本身的立場¹⁸⁵，其中的重點包括：(1)肯認開放原始碼運動由來已久，其獨特的開發模式與授權機制形成軟體界的一個重要領域。(2)開放原始碼運動的參與者有商業及非商業的兩種動機，就商業目的而言，主要在提升相關硬體、服務或專屬軟體的獲利。(3)消費者採用開放原始碼或專屬軟體，考量的因素從成本、價值、透明度、安全性、穩定性到相容性，不一而足。(4)開放原始碼社群中的理念不盡相同，最具理想色彩的可以自由軟體當代表。(5)商用軟體公司的經營理念與自由軟體社群的理念有本質上的差異，對公眾採購、研發政策及智慧財產權法制自然有不同的看法。(6)商用軟體公司支持中立的政府採購法，政府應根據競爭產品的優劣進行評比，而非理念。(7)Microsoft 相信政府資助的基礎研究應將成果廣為分享，開放原始碼在這部分有其助益。(8)商用軟體公司需要廣泛的智慧財產權保護，並反對削弱商標、營業秘密、著作權及專利的效力。

在 Microsoft 眼中，開放原始碼運動最重要的兩個特色是「集體創作」與「公眾授權」，前者異於獨家掌控的集中開發模式，後者允許原始碼的自由複製、修改及流通，與專屬軟體單獨將執行碼授權出去不同。Microsoft 提議「分享原始碼」方案時特別強調，使用者評估軟體的因素很多，原始碼沒有成本看似誘人，但考量長期則整體成本未必划算。Microsoft 指出，Red Hat 的研究報告顯示採用 Linux 比其他專屬的 Unix 划算，但 IDC 機構的分析結果則顯示 Linux 的整體成本反而比 Windows 來得高。因此，單看低成本的

¹⁸⁴ 微軟為避免失去政府單位的市場，針對特定對象有限度地開放其程式碼，以供檢閱。有關「分享原始碼」模式與「開放原始碼」模式的比較，請參見 Craig Mundie & Michael Tiemann, *Shared Source vs. Open Source*, Aug. 9, 2001, at http://www.linuxdevcenter.com/pub/a/linux/2001/08/09/oscon_debate.html?page=1 (visited May 4, 2004).

¹⁸⁵ See Microsoft, *Shared Source Initiative: Open Source Software*, at <http://www.microsoft.com/resources/sharedsource/Government/opensource.mspx> (last updated Feb. 2004).

原始碼將失之偏頗，至於系統整合的成本、使用上的難易¹⁸⁶與熟悉度，則是另外一回事。

再從軟體的透明度來看，Microsoft 認為開放原始碼固然把一切攤在陽光下無所遁形，但其所提出的「分享原始碼方案」則有異曲同工之效，雖然只有特定機構在一定條件下才能取得 Microsoft 的原始碼，但對於確保程式內沒有隱藏爭議性的功能或檢查其安全性，仍達到相同的效果。況且，開放原始碼對大部分使用者來說，如果不具備閱讀原始碼的能力，其實開放的作用不大，一般使用者對軟體的需求，還是基本的功能和完善的售後支援。對於原始碼在眾目睽睽下比較安全的說法，Microsoft 則舉專門追蹤安全漏洞的 CERT 機構為例，2002 年統計資料顯示 Windows 有五項安全漏洞，Linux 及 Solaris 則有十二項，以此證明藉由自願方式來修補軟體漏洞的效果較低。而且，Microsoft 認為開放原始碼發佈的修補程式是否經過嚴格的測試，也是另外一個問題，如果品質不良，挖東牆補西牆，可能製造新的漏洞或引發更多相容性的問題。另外，政府單位逐漸依賴國際性的安全認證，如 Common Criteria 程序，開放原始碼在這方面也往往付之闕如。針對各式各樣新舊不一的軟硬體，Microsoft 投入上千萬美元確保自己的產品能夠執行無礙，開放原始碼陣營則缺乏足夠的資源，而使用者一旦自行修改原始碼，更可能提高維持品質或相容上的難度¹⁸⁷。

在政策方面，不同的理念導致不同的看法。自由軟體陣營認為程式接近言論，在言論自由的概念下應保持自由狀態，商用軟體公司則認為軟體是一種智慧財產，應受到相關法律的保護。對公共研發政策，有人認為軟體研發的成果應回饋給社會大眾，並採用類似 GPL 的公眾授權模式，有人則認為，公眾授權如果限制了商用軟體利用研發成果，則加諸技術移轉的限制未免太大，而與政策上帶動私人企業創新與擴大應用的立意相

¹⁸⁶ 即使是 Eric S. Raymond 投入開放原始碼運動多年，仍不免呼籲程式設計師們注意平民百姓的需求，2004 年三月其在網路上公布使用某開放原始碼軟體的坎坷經驗，認為難用的軟體將讓 Microsoft 日益茁壯，引起社群的注意。細節請參見 Eric S. Raymond, *The Luxury of Ignorance: An Open-Source Horror Story*, at <http://www.catb.org/~esr/writings/cups-horror.html> (visited Apr. 2, 2004)。不同意見請參見 John Gruber, *Ronco Spray-On Usability*, at http://daringfireball.net/2004/04/spray_on_usability (visited Apr. 2, 2004)。

¹⁸⁷ See Microsoft, *supra* note 185.

左。如果採用公眾授權，應以 BSD 這類允許開放原始碼社群及商業應用並行的模式為佳。因理念的不同，雙方對智慧財產權的看法也不同。如果程式碼是一種言論的形式，而非財產，則自由軟體社群所訴求的將弱化某些智慧財產權法的保護，例如，反對軟體專利而影響軟體界的進步。商用軟體公司依賴廣泛的智慧財產權法，其不僅確保投資得到必要的保護，也增加軟體界繼續創新的誘因¹⁸⁸。

Microsoft 於 2001 年提出「分享原始碼方案」時指出¹⁸⁹，該方案的精神在於融合傳統的「商用軟體發展模式」與「開放原始碼模式」的優點，成為符合本身追求商業利益的一種授權方式，同時，具備實務上不同情況所需要的彈性。在授權方案下，Microsoft 的原始碼可以交到客戶、商業夥伴、獨立開發人員、研究人員、學生或其他人士手中，但智慧財產權仍屬於 Microsoft 所有，因為智慧財產權對軟體界的創新十分重要。在方案中，Microsoft 因應不同的需求，可提供各種授權專案，並決定取得原始碼的範圍。Microsoft 進一步說明，「商用軟體發展模式」是傳統的方式，藉由研發的集中與投資，將想法轉換成實際可行的產品，同時自行負擔成敗的風險，產品如果順利問世，則由銷售狀況及附帶支援的服務中獲得利潤。在這種模式下，原始碼等於整個研發的結晶，讓他人取得原始碼十分罕見。過去，這種模式成功造就了蓬勃發展的軟體產業，也符合一般人對原始碼並不關心的狀況。「商用軟體發展模式」的主要優點是，使用者可以依靠單一公司的專業，而利之所趨，商業公司將盡力讓軟體穩定、合法、好用，並與其他軟硬體相容。使用者同時可以依賴其專業人員提供的服務與支援，並讓軟體得到必要的更新。上千個軟體公司透過這個模式營運，如果有什麼缺點，主要是外部的程式設計師無法參與開發過程，減少了創新的來源，同時，產品一旦失敗，外人也難以接手另求發展或改良。當然，商用軟體必須付費取得。開放原始碼模式在 Microsoft 看來，推動的主力是同儕間的肯定或自我的成就感，而非利潤導向，至於市場上的經銷商，則由相關週邊的軟硬體或支援服務來取得利潤。過去這幾年，開放原始碼造就了許多優異的軟體，

¹⁸⁸ *Id.*

¹⁸⁹ See Microsoft, *Basic Principles of Software Source Code Licensing*, at <http://www.microsoft.com/resources/sharedsource/Articles/LicensingBasics.mspx> (last updated Mar. 2004).

也吸引不少程式設計師及公司的投入，但對一般人來說仍認識不多。開放原始碼的優點是透過社群運作激發創新與進步，並將軟體成本壓到最低，但在程式碼之外，教育訓練、服務、支援等費用都可能超過軟體本身的價格。另外一個主要的缺點則是，並沒有一個主要單位負責所有的事情，不同版本的軟體可能造成衝突，開放原始碼模式的長期獲利也有待市場檢驗。在該模式下，特別是自由軟體陣營，以 GPL 條款歧視商用軟體公司，主張程式碼中的智慧財產權不應為任何人所有，並限制未來軟體在授權上的可能¹⁹⁰。

因此，Microsoft 提出「分享原始碼方案」，類似「開放原始碼模式」，在一定條件下提供內部的原始碼，讓他人能夠據以推動創新與改良，例如，將原始碼授權給大學機構，進行新一代軟體的研發與教學，將有助於軟體界長遠的進步。同時，該方案保留了「商用軟體發展模式」的優點，藉由保護研發人員的智慧財產權，商業利益將持續成為創新的誘因。綜合兩者，「分享原始碼方案」能夠在自由市場的運作下，一邊激發創新，一邊開放原始碼，可謂魚與熊掌兼得¹⁹¹。

「分享原始碼方案」讓原始碼交到他人手上，意謂著風險的增加。2004 年初，網路上首次公開流傳 Windows 系統原始碼，為數相當可觀，引起廣泛注意。Microsoft 證實原始碼外流後表示，內部的安全控管沒有漏洞，可能的問題出在「分享原始碼方案」下的廠商，疑似管理不當以致原始碼遭到竊取¹⁹²。對於此次事件，Microsoft 雖然公開呼籲下載未經授權的原始碼是違法的行為，但實際上效果有限。電腦安全專家憂心指出，過去原始碼在封閉的情況下，Windows 系統的安全漏洞便層出不窮，如今，駭客們取得相當比例的原始碼，將有機會深入系統的核心，發掘更多的弱點，做為下一波攻擊的基礎。網路上許多有心協助的安全專家，在無法檢視原始碼的情況下，只好愛莫能助，原始碼開放與封閉差別也可由此看出。

2004 年四月，出乎意料之外，Microsoft 以開放原始碼模式釋出一項工具軟體，並

¹⁹⁰ *Id.*

¹⁹¹ *Id.*

¹⁹² See Paul Roberts, Joris Evers, *Microsoft: Shared Source not to blame for leak*, Feb. 13, 2004, at http://www.infoworld.com/article/04/02/13/HNmssourceleak_1.html (visited Apr. 12, 2004).

透過最大的開放原始碼網站 SourceForge 公布，同時採用符合「開放原始碼定義」的「通用公眾授權」(Common Public License)。CPL 是不折不扣的開放原始碼授權，而非先前「分享原始碼」的變形，這意謂著 Microsoft 正式跨過自己長期固守的界線，開始參與開放原始碼社群。是否 Microsoft 將全盤轉向開放原始碼模式，將所有的原始碼陸續公布出來？業界人士分析後發現，答案應該是否定的。Microsoft 公布的 WiX 技術，雖非耳熟能詳的重量級軟體，但亦非無足輕重，該軟體的主要功能在於協助軟體安裝，在未來 Windows 平台上扮演要角，透過該工具程式，開發人員可以簡化軟體安裝設定的機制。因此，Microsoft 釋出 WiX 原始碼而非其他軟體，實有策略上為新版 Windows 系統鋪路的用意，希望未來更多的開放原始碼軟體能夠順利安裝到 Windows 平台上。另外，可能藏有玄機之處在於，不確定 Microsoft 是否同時將相關技術及資料格式申請專利？Microsoft 宣稱釋出原始碼，是基於對開放模式成功經驗的觀察，但以 Office 採用特定 XML 資料格式並取得專利的情況下，不免引發猜忌，難以確定 Microsoft 真正的目的為何¹⁹³。對 SCO 案來說，Microsoft 採用開放原始碼公眾授權，或許 SCO 要主張 GPL 無效將更為不易。



開放原始碼人士並指出，Microsoft 的目的是希望程式設計師由 GNU/Linux 系統往 Windows 系統轉向。Microsoft 採用 CPL 而非 GPL 授權方式，背後有其原因。首先，CPL 與 GPL 有相容性問題，CPL 授權下的原始碼可否直接納入 GPL、BSD 或其他開放原始碼公眾授權軟體中，答案是否定的，相對地，GPL 原始碼同樣不能與 CPL 原始碼合併。除非自己是原始碼的所有人，或得到所有權人的授權，才能將相同的原始碼放在不同的授權下，而避開相容性的問題。如果程式設計師開發新的模組，增強 CPL 授權軟體的功能，而該模組並非原程式的衍生著作，則不需強制公開該模組的原始碼，這或許說明了為什麼 Microsoft 決定使用 CPL 而非 GPL。基於這些分析，部分人士建議不要加入 WiX 開發計畫，但不同意見認為，Microsoft 如今採用開放原始碼模式，不論企圖為何，

¹⁹³ See Steven J. Vaughan-Nichols, *MS Open-Source Move Is Straight from Playbook*, Apr. 6, 2004, at <http://www.eweek.com/article2/0,1759,1563644,00.asp> (visited Apr. 12, 2004).

過去多年抹黑開放原始碼運動的許多說辭，都將不攻而破，就這一點不妨站在支持立場¹⁹⁴。

第二項 開放原始碼運動的障礙

根據財星五百大資訊長的訪問資料，2004年「開放原始碼商業會議」列出開放原始碼運動遇到的六項障礙¹⁹⁵，除了前述SCO引發的相關問題及授權的複雜度外¹⁹⁶，其他值得瞭解的障礙為：(1)缺乏正式支援。企業考量整體成本時，免除授權金雖然節省了部分的費用，但並非表示成本一定最低。隨著經銷商的增加並穩定獲利，對Linux的支援可望日益提升，硬體廠商搭配Linux的出貨量增加後，也足以提供更多的支援，前者如Red Hat、Novell，後者如IBM、HP。至於每個團隊的支援程度，仍需視個案而定，商用軟體公司的支援不必然優於開放原始碼團隊，例如，Apache在網站伺服器上的支援便超越Microsoft同類產品許多。而新的開發計畫如果忽略必要的支援，也不足以形成一定的規模，因此，用戶支援應逐漸成為各開發團隊重要的考量，而有機會拉近與商用軟體公司的距離。(2)軟體變動的幅度較大。不論是開放或封閉，軟體一旦發行，後續更新及修補的動作似乎成了一種常態，開放原始碼因為集體創作及回饋的特性，變動的次數較一般專屬軟體來得頻繁，因而可能增加後續維護的成本。但反面而言，由於軟體瑕疵無法避免，修補次數的增加也可能代表安全及效率的提升。對於不願意隨時更新版本的企業用戶，部分廠商提供定期更新的方案，試圖從技術上處理軟體生命週期管理上的難題。另外，開放原始碼中不少性質相同的軟體，隨著競爭關係的消長，也增加選用上

¹⁹⁴ See Pamela Jones, *Microsoft Wishes to Tempt Developers With Its Code*, Apr. 8, 2004, at <http://www.groklaw.net/article.php?story=20040408113135111> (visited Apr. 13, 2004).

¹⁹⁵ See Dan Farber, *Six barriers to open source adoption*, Mar. 20, 2004, at http://techupdate.zdnet.com/techupdate/stories/main/Six_barriers_to_open_source_adoption.html (visited Apr. 1, 2004)

¹⁹⁶ 授權模式的複雜度可以再舉一例，例如，Gluecode以開放原始碼為基礎，提供「企業原始碼授權」(ESL)供客戶延伸功能，但不允許客戶將原始碼再散布或授權給第三者，對於一般企業用戶而言，這樣的限制不大，對於其他軟體廠商，Gluecode則提供另外一種收費的「OEM原始碼授權」，並增加支援的幅度。細節可參考http://www.gluecode.com/website/html/prod_licensing.htm。

的難度。(3)缺乏發展藍圖。對企業用戶，系統中長期的規劃有其必要，開放原始碼團隊在這部分提供的能見度相對較低，因而提高選用上的疑慮。但相反地，開放原始碼運作的機制比較民主，公開辯證的機會也多，社群在過程中往往追求長期利益而非短程效果，也沒有遭到單一廠商鎖住系統平台的問題。(4)功能上的不足。目前 Linux 所肩負的是一般初階伺服器的應用，對於高階的資料中心或企業資源規劃系統，仍未取得可觀的佔有率，在一般用戶端方面，佔有率預計到 2007 年也只有百分之五，推究起來，功能有待加強應是主因之一。針對不足之處，廠商持續投入資源開發新的功能，以期進入高階或終端用戶的市場¹⁹⁷。(5)獨立軟體商的加入。專屬軟體常常將自己定位為開發平台，吸引更多的獨立軟體商投入，以擴大整體的佔有率與影響力，雖然原始碼的開放有利於獨立軟體商的採用，但形成開發平台的局面較不明顯，因此，影響範圍有仍擴大的空間。

第三項 意識型態

意識型態雖然不算法律問題，但對於是否採用開放原始碼來說，仍為不可忽視的風險因素之一。開放原始碼運動形成國際性規模，針對特定議題，已開發國家與開發中、未開發國家可能出現不同的立場，即便在資訊強國如美國，內部也存在不同的意識型態。史丹佛法學院教授 Lawrence Lessig 公開表示，華盛頓方面部分極端人士，將開放原始碼視為保護智慧財產權的嚴重威脅，已經到了「麥卡錫主義」的程度，在他們眼中，支持公眾領域(public domain)與開放原始碼基本上就是支持共產主義。Lessig 認為，智慧財產權制度應該回歸當初立法的目的，真正重要的是在公益與私利之間是否取得平衡而促進創新與進步¹⁹⁸。以著作權為例，當初國會制訂著作權法為的是落實憲法精神，以保障創作者在一段時間內對作品享有排他權利做為誘因，藉以促進科學及實用技藝的進步，最後達到增進公眾福祉的目標。因此，其中重要的是公眾的福祉，而非創作者個別

¹⁹⁷ 例如，MySQL 正開發叢集 (cluster) 版本，並支援 stored procedure，Novell 計畫讓 Linux 支援新一代 64 位元的電腦等。

¹⁹⁸ See John Pallatto, *Lessig: Open-Source Industry Must Lobby for Political Backing*, Mar. 17, 2004, at <http://www.eweek.com/article2/0,1759,1550036,00.asp> (visited Apr. 1, 2004)

的權利，最高法院曾強調相同的觀點，指出重要的是讓創作者的心力最後裨益於社會大眾。由此可以看出著作權法背後是「社會導向」的思維，而非以「作者導向」出發¹⁹⁹。賦予智慧財產權保障個人創作不受侵害，讓私利得以成為經濟上有效的誘因，為的是促進公益的累積，前者是手段，後者是目的，過去手段與目的之間似乎只是單行道，認為創作者關心的只是手段部分，落實了智慧財產權制度之後，公益的目的自然水到渠成，對此，開放原始碼運動則注入反向的思考，將目的的達成擺在前面，再回頭找出調整手段的方式。關於公益、私利與智慧財產權的關係，容後面章節詳述²⁰⁰。

第四項 資訊科技的改變

資訊科技的進步及系統架構的演變，可能影響公眾授權的適用範圍。例如，網際網路興起之後，主機代管業者出現，電腦運算及儲存空間如水電瓦斯一般，可依照需求自行根據用量或時間租用。在這種環境下，對許多公司來說，資訊委外的可行性提高，「應用服務供應商」(ASP, Application Service Provider)成為一種可行的商業模式。採取這種模式，供應商集中管理各種應用軟體及網路頻寬，供不同的客戶使用時，與過去不同，軟體的執行不再需要將目的碼散布出去，在這種情況下，既然不涉及散布問題，開放原始碼軟體的衍生功能，依照授權條款則沒有強制公開的問題，因此，ASP 業者可以一邊採用開放原始碼軟體，一邊改良功能對外供客戶使用，但不需要將衍生的原始碼公布出來。這種作法相當程度上等於合法規避了 GPL 公眾授權的規定，以當初 GPL 未曾預料到的技術手段，脫離 GPL 的規範。

除了上述網路應用模式的轉變，讓開放原始碼授權可能出現漏洞外，另外一種稱為 Web Service 的資訊標準，可能在技術層面達到同樣的結果。一般瀏覽網頁，使用者透過瀏覽器取得遠端的資料，顯示在螢幕上，Web Service 的概念類似，同樣由使用者端向遠端伺服器發出需求，不同的是，要求伺服器回應的並非只是資料，而是執行特定功

¹⁹⁹ See Teresa Hill, *Fragmenting the Copyleft Movement: The Public Will Not Prevail*, 1999 UTAH L. REV. 797, 800 (1999).

²⁰⁰ 參見第四章第五節第二項「開放原始碼模式的啟發」。

能，亦即，在取得資料之外同時具備執行伺服器上軟體元件的能力，這種新的業界標準對於跨網路的應用整合相當有用，等於讓不同電腦之間透過網路能夠以標準的語言互相溝通。採取這種模式，背後代表著各種不同軟體元件的整合，與前述模式類似，使用者執行軟體時，核心功能可能都在遠端執行，而不是在自己的電腦上，因此，執行的目的碼不需要加以散布，如採用開放原始碼軟體並予以改良，在這種情況下，拒絕揭露改良部分的原始碼可能也沒有違反 GPL 的規定。

上述兩點說明資訊科技的變化，可能讓公眾授權浮現力有未逮之處，如果需要防堵，實務上可以考慮將新的使用樣態列入規範，應不至於對整份公眾授權產生重大影響。只是授權條款修正後，如何兼顧過去已釋出的軟體，未來能否涵蓋新的技術變化，則有待將來探討。

第九節 小結

開放原始碼社群日益成長，集散規模最大的 SourceForge 網站統計，到 2004 年三月為止，登記在案的開發計畫約七萬八千個，網站註冊的人數約八十一萬，單日網頁瀏覽次數約九百萬次，單日下載檔案約八十四萬次，單日寄送郵件一百四十萬封。Amazon 所屬 Alexa 網站統計全球資料流量顯示²⁰¹，SourceForge 在前十萬個網站中排名第兩百五十八名，過去十二個月，該網站已超越 IBM、Sun 及 Cnet.com 等知名網站，顯見開放原始碼運動持續成長的趨勢。然而，在擴大影響範圍的背後並非沒有隱憂，除了顯而易見的 SCO 案之外，前面大略提到，未來軟體專利可能成為開放原始碼模式下首要的風險來源。Torvalds 於 2004 年三月出席公開研討會時，再次呼籲社群重視軟體專利帶來的威脅。事實上，他認為 Linux 在 2.6 版之後，開發重心從系統核心轉往與使用者息息相關的桌面環境，將繼續提高市場佔有率，但未來所面對的挑戰，主要不是技術問題，在所有非技術性問題中，軟體專利將成為最大的威脅²⁰²。

²⁰¹ See Alexa, *Daily Traffic Rank Trend*, at <http://sourceforge.net/traffic.php> (visited Apr. 12, 2004)

²⁰² See Andrew Donoghue, *Patents 'biggest threat' to Linux - Torvalds*, Mar. 23, 2004, at

第四章 軟體專利制度的改革

開放原始碼模式未來最主要的威脅在於軟體專利，底下分別探討軟體專利的演進、軟體專利引發的各種爭議以及各方的改革建議，受到開放原始碼模式的啟發，本文則提出另外一種觀點下可能的作法。

第一節 軟體專利的演進

軟體是否為可以申請專利？過去三十年間爭議不斷，從最初將軟體排除在適格標的之外，逐步接受可以透過硬體方式實施而准予專利，到純粹軟體可以申請專利，近來，並擴大到所實施的商業方法也能申請專利，可以看出實務上逐漸透過判例賦予軟體適格地位的演變²⁰³。美國專利法上涵蓋的適格標的包含程序、機械、製造等²⁰⁴，其中與軟體較接近的是同樣不具實體特性的程序部分，早期認為單純描述自然法則或實施心智步驟的程序無法取得專利，數學演算法在 *Gottschalk v. Benson*²⁰⁵ 一案中經最高法院判定不在專利適格標的範圍內，其後一般認定軟體等同於演算法的集合而同樣無法取得專利的保護。*Benson* 案經過六年，法院在 *Parker v. Flook*²⁰⁶ 一案中，針對化學轉換程序中監測臨界值加以警示的發明，否定該方法經電腦實施的可專利性，法院所持的理由是，該發明唯一的新穎性在於軟體，而軟體並非適格標的，因此，整個發明無法取得專利保護。然而，三年後法院改變了見解，在 *Diamond v. Diehr*²⁰⁷ 一案中，發明人將習知的方程式透過電腦軟體應用在塑膠模具的製造上，而獲得專利保護，法院認為，發明人並非限制他

<http://www.groklaw.net/staticpages/index.php?page=20030929022014462> (visited Apr. 12, 2004)

²⁰³ See Julie E. Cohen & Mark A. Lemley, *Patent Scope and Innovation In The Software Industry*, 89 CALIF. L. REV. 1, 7-14 (2001).

²⁰⁴ 35 U.S.C.A. §101 (“Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.”).

²⁰⁵ 409 U.S. 63 (1972).

²⁰⁶ 437 U.S. 584 (1978).

²⁰⁷ 450 U.S. 175 (1981).

人使用已習知的方程式，而是針對該方程式在特定應用領域上加以保護，在軟體本身之外的應用讓整個發明具備了可專利性。隨著電腦應用範圍的擴大，到 1990 年代初期為止，實務上多根據 Diehr 案採取變通方法，以硬體裝置包含軟體功能的方式，讓許多以軟體為主的發明得以落入適格標的範圍，而在整體發明上取得專利的保護。1994 年聯邦法院對 *In re Alappat*²⁰⁸ 一案中的判決，為軟體專利開啟新的一頁，法院在判決中表示，一般電腦執行特定軟體後，可視為特定功能的電腦，而得以申請專利保護。在 *Alappat* 案後，只要將機器中實施的軟體本身透過專利申請範圍具體闡釋，不再需要迂迴地以保護整體發明的說法來包裝純粹保護軟體的意圖。隔年，關於「在機器中實施」的部分，專利局於 *In re Beauregard*²⁰⁹ 一案中同意 IBM 的申請方式，凡「儲存在媒體上」的電腦程式即可成為審查基準所考慮的發明。到了 1998 年，軟體以更純粹的方式成為適格標的，無需依附在儲存媒體上，在 *State Street Bank & Trust v. Signature Financial Group*²¹⁰ 一案中，針對純以軟體計算來管理財務資訊的一項發明是否為專利適格標的，聯邦法院推翻地方法院根據過去 *Freeman-Walter-Abele* 測試法的見解，認為即便缺乏實體裝置，只要能產生有用、明確而具體的結果，軟體本身即可申請專利。隨後聯邦法院於 *AT&T v. Excel Communications*²¹¹ 一案中再度肯認新的見解，軟體可專利性的問題於是塵埃落定²¹²。

相對於美國，歐洲採取不同的政策。歐洲議會(Parliament)於 2003 年九月投票表決通過，維持將軟體及商業方法排除在專利適格標的之外的規定，但關於軟體的可專利性，正反雙方仍持續爭論不休。例如，歐盟部長理事會(Council of the Ministers)於 2004 年初起草新的提案，希望推翻議會的決定，讓透過電腦實施的演算法和商業方法都能受到專利保護。另外，即使在法無明文許可的情況下，反對軟體專利的團體 FFII²¹³認為，

²⁰⁸ 33 F.3d 1526 (Fed. Cir. 1994).

²⁰⁹ *In re Beauregard*, 53 F.3d 1583 (C.A.Fed. 1995).

²¹⁰ 149 F.3d 1368 (Fed. Cir. 1998), cert. Denied, 525 U.S. 1093 (1999).

²¹¹ 172 F.3d 1352 (Fed. Cir. 1999).

²¹² See Cohen & Lemley, *supra* note 203.

²¹³ Foundation for a Free Information Infrastructure 為位於慕尼黑之一非營利組織，致力於支持建立在著作

歐洲專利局其實已核發三萬個以上的軟體相關專利²¹⁴，對軟體的創新造成負面影響，其中以美國和日本所申請的專利佔大宗。到了 2004 年五月，出乎開放原始碼社群意料之外，歐盟部長理事會對先前歐洲議會的決議通過修正案，以兩票些微差距，逆轉了軟體不得申請專利的立場。在修正案中，前一年增加的許多限制不復存在，歐盟可能朝比美國標準更加寬鬆的方向改變，整個結果類似於執委會(European Commission)最早建議的版本。歐盟部長理事修正的指令傳回歐洲議會，將於 2004 年九月會議中再次進行討論及表決，但受限於多數決的門檻，一般預料翻案的難度較高。如果得以翻案成功，程序上修正案將再度回到部長理事會手中，部長理事會如果堅持自己的版本，雙方則進入為期六週的協商期間。反對新版本者指出，部長理事會將改版的細節交由獨立專家群制訂，其中參與者多半來自於各國專利局或受大企業所左右，專利數量的增加對這些人來說等於影響力的提高，因此，即使反對人士以美國專利糾紛攀升的後果做為警告，仍無法動搖決策的結果²¹⁵。附帶一提，歐盟部長理事會此次主席為愛爾蘭籍，愛爾蘭過去為西歐最貧窮、工業化程度最低及人口最少的國家，在歐盟財政補貼的協助下，不過十年，這個人口不到 400 萬的國家，在 1998 年取代美國成為全球最大的軟體出口國，並成為該國第一大產業²¹⁶，做為軟體輸出大國的地位或許可以說明愛爾蘭籍主席在主導翻案背後的利益考量。

美國與歐洲部分國家對軟體的可專利性採取不同看法，雖然在 TRIPs (Trade-Related Aspects of Intellectual Property Rights)的規範下軟體為專利適格標的，但未來國際上是否會在 TRIPs 的調和下對軟體專利採取一致的看法，以及歐盟最後拍板定案的結果為何，

權、自由競爭及開放標準之上關於公共資訊財的發展，其代表超過三百位會員、七百家公司及五萬位支持者，網址為 www.ffii.org。

²¹⁴ See European Software Patent Statistics, at <http://swpat.ffii.org/patents/stats/index.en.html#appland> (visited Apr. 1, 2004); Current Situation in Europe, at <http://swpat.ffii.org/news/03/situ0923/index.en.html> (visited Apr. 1, 2004).

²¹⁵ See Munir Kotadia, *EU approves software patent changes*, May 18, 2004, at <http://news.zdnet.co.uk/business/legal/0,39020651,39155028,00.htm> (visited May 26, 2004);

²¹⁶ 許昭瑾，眼光向外才有未來：台灣不應閉門造車，電子時報，93 年 5 月 26 日。

目前尚未可知²¹⁷。以現況而言，不管是在美國或是歐洲，就開放原始碼社群既有的數量及規模而言，軟體相關專利都將造成實質上的影響。以演進的趨勢觀察，美國近幾年的演變包括：專利法涵蓋標的範圍的擴大、專利核發的數量增加、1982 年起負責專利訴訟的聯邦法院比過去法院更傾向於承認專利的有效性²¹⁸。專利的影響力增加，可能相對壓縮了公眾創作的自由度，相對地，開放原始碼運動則持續擴大公眾可用的智慧財，未來公眾所擁有的創作空間或資源，受到兩者交錯影響可能更為明顯。

在我國，智慧財產局曾於民國八十七年十月修訂專利審查基準，以第八章第二節²¹⁹將電腦軟體相關發明納入特定技術領域，正式將軟體專利列入審核基準。根據章節的前言，我國電腦軟體相關發明的專利審查基準，係依據現行專利法及專利法施行細則，參酌專利審查基準總則篇後，以美國相關專利審查基準為主，以日本相關專利審查基準為輔，加以訂定。由於日本受美國影響，故實際上我國等於採取美國的立場，而非歐盟部分國家的意見。在理由方面，智慧財產局表示軟體工業蓬勃發展，各國均認有必要保障軟體中的實質技術功能，以促進產業創新，因此，透過專利法保護軟體發明是各先進國家普遍的做法，我國自然有跟進的必要。在實務操作面，可以從審查基準中發現電腦軟體不需要像過去依附在硬體上，才能結合成為專利的適格標的，審查基準中並表示，有關以電腦可讀取的記錄媒體為形式的發明，以往不屬於可准予發明專利的範疇，但考量美、日等國已列為可准予專利標的，為順應世界潮流，智慧財產局也將記錄媒體形式之發明納入審查範疇。由上述幾點可知，電腦軟體在我國已納入專利保護範圍，美國實務判例與產業運用軟體專利的態度，對國內未來的發展應有相當程度的參考價值。

第二節 軟體專利引發的爭議或批評

²¹⁷ See Eric Allen Engle, *When is Fair Use Fair?: A Comparison of E.U. and U.S. Intellectual Property Law*, 15 TRANSNAT'L LAW. 187, 221 (2002).

²¹⁸ See Maureen A. O'Rourke, *Toward a Doctrine of Fair Use in Patent Law*, 100 COLUM. L. REV. 1177, 1178 (2000).

²¹⁹ 參見 http://www.tipo.gov.tw/patent/patent_law/explain/patent_law_3_1_8.asp#b (visited May 22, 2004).

軟體可專利性的問題在歷史演進中一直成為關注的焦點，大家將心力放在可否准予專利的爭辯上，以致於忽略了通過軟體專利後應有的配套措施。例如，在美國專利局錄用足夠而且合適的審查員，並在分類系統中能夠妥善處理軟體專利之前，成千上萬的相關專利已經通過審核，為未來埋下了爭議的因子。而軟體的創新往往藏於原始碼中，不如其他學科廣見於出版刊物，更增加了發明人及專利審查員在搜尋前案上的難度，因此，比起其他領域，後來許多軟體專利在新穎性和進步性方面受到較高程度的質疑，並不令人意外²²⁰。開放原始碼運動持續擴大加上網路普及後，各種軟體技術有機會公開流傳，雖有助於先前技藝的搜尋或提高未來專利的品質，但換個角度看，開放原始碼也暴露在專利侵權的風險中。由於開放原始碼運動日益擴大，本節試圖從不同角度分析常見的爭議焦點，希望透過這些分析釐清軟體專利對開放原始碼可能的影響。

第一項 軟體專利存在的必要性

專利制度是推動進步的一種手段，在探討手段與目的之間的關連性時，可以加上軟體產業的本質，從分析中指出軟體專利的爭議。一般而言，產業投入研發的費用越高，透過專利制度保障投資的需求越強。軟體公司比起半導體產業所費不貲的廠房設備，或製藥業長年實驗的費用支出，相對來說進入門檻低了許多，因此，軟體產業訴諸於專利制度提供壟斷、保護投資的必要性較低²²¹。其次，除了專利之外，產業的創新動力是否受其他誘因的影響？例如，市場的競爭、同儕間肯定、為造福他人或純粹基於興趣、學習新知等。如果這些誘因的效果越好，透過制度性壟斷的必要性就越低，專利制度如果運用不當，反而可能壓制了其他有效的誘因，破壞產業既有的創新機制²²²。開放原始碼社群認為過去沒有軟體專利，軟體產業一樣推陳出新，顯見程式設計師受到許多誘因的激勵，繼續讓專利以外的誘因促進產業的進步，並沒有什麼問題。更何況，對開放模式

²²⁰ See Julie E. Cohen & Mark A. Lemley, *Patent Scope And Innovation In The Software Industry*, 89 CALIF. L. REV. 1, 12-13 (2001).

²²¹ See Dan L. Burk & Mark A. Lemley, *Policy Levers in Patent Law*, 89 VA. L. REV. 1575, 1582-83 (2003).

²²² *Id.*, at 1587.

下大部分無償貢獻的程式設計師，專利的花費與繁複的程序構成相當高的障礙，令人望之卻步。2003年十月，美國聯邦貿易委員會發表一份超過三百頁的研究報告²²³，探討如何在市場競爭、專利法制與公眾政策三者間取得平衡。報告中指出，針對軟體界，競爭的強化比專利制度更能促進產業的創新，其中，開放原始碼模式列為兩個有效的替代方案之一²²⁴。

現行軟體專利制度存在過度壟斷的風險。觀察聯邦法院對軟體專利的態度，可以發現認定符合進步性要件的難度很高，不過一旦通過，軟體專利往往涵蓋範圍廣泛。聯邦法院強調，軟體專利的進步性，只有以前從未出現任何軟體執行該項功能時，才得以成立。大部分的專利顯然無法通過這種嚴格的標準，但通過後，則幾乎不受前案技術的限制。在這種情況下，專利往往不只保護了該項功能的實施方式，而是該項功能的整個概念成了專利本身保護的標的。尤其，軟體專利不需要在專利說明書中揭露原始碼，習知技藝者是否能夠從高階描述及其他資料中順利實施專利技術，其實值得懷疑。結果是，第一個以軟體落實創新想法的人，壟斷了同一類型的軟體功能，後人再也沒有機會採用不同技巧實施相同概念²²⁵。軟體技術進步快速，廣泛壟斷的特性加上保護期間甚長，軟體專利可能帶來反效果，在目的的達成上違背專利法設立的初衷。法院對進步性設立高標準，除了讓軟體專利涵蓋廣泛而出現全盤通吃的效用，對後續發明人，軟體漸進式創新的成果也容易遭到否定，而通不過進步性的門檻。專利制度的出發點在於促進產業創新，手段的實施應有助於目的的達成，賦予專利權人獨佔的權利後，如果無益於產業創新，或結合了其他誘因後整體反而產生負效果，則軟體專利制度可能有修正的必要。

就開放模式的效益，另外可以從全球資訊網(WWW, World Wide Web)快速成長得到佐證。網際網路上第一台網站架設以來，網頁的數量一直呈指數上揚，之所以如此，很

²²³ See Federal Trade Commission, *To Promote Innovation: The Proper Balance of Competition and Patent Law and Policy*, Oct. 2003, available at <http://www.ftc.gov/os/2003/10/innovationrpt.pdf> (visited May 6, 2004).

²²⁴ 另外一項建議則認為著作權法即能有效促進軟體的創新，不需要專利法的介入。著作權法保護軟體的表達方式(expression)非而背後的概念(idea)，其保護程度已足以讓軟體成為獲利可觀的產業。參見註223，第155頁。

²²⁵ See Burk & Lemley, *supra* note 221, at 1594-99.

重要的一個原因即在於網頁的公布，其實也可視為採取了開放原始碼模式。製作網頁採取不同的語言(HTML)，網頁中如果沒有額外加入特定高階語言以增加功能，通常以標準的 HTML 語言描述網頁的內容，即可符合大部分網站的需求。使用者瀏覽網頁，實際上是將遠端伺服器的資料下載到個人電腦，再透過個人電腦的瀏覽器根據指定的格式顯示出來，在這種方式下，使用者如果需要，隨時可以檢視全部下載的原始資料。由於原始資料中含有網頁的內容與對應的格式，從中便可以學習他人製作網頁的技巧與 HTML 的語法，在剪貼之間增進自己製作網頁的能力。WWW 採取免費的標準語言，系統設計上讓網頁內容開放，是得以快速發展的主要因素²²⁶。如同開放原始碼遵守免費的公開標準，讓原始碼開放流通，是開放原始碼運動持續擴大的原因。開放模式在此所展現的效益，有助於與專利賦予壟斷保護的方式相比較。

專利制度從權利界定與保護的精神出發，與開放模式的理念相左，其封閉性甚至可能壓縮開放空間，而實務上專利涵蓋的範圍也有增無減。在 State Street 一案中，聯邦法院將專利的保護擴大涵蓋商業方法，商業方法引發的爭議不下於軟體專利。在該案中，負責的聯邦巡迴上訴法院法官之一 Giles S. Rich，當年曾於國會主導專利法的修正，提出「太陽底下由人製成的任何事物」都可能申請專利的說法，1952 年專利法修正後，專利法涵蓋範圍於是逐漸跨出科學與工業領域²²⁷。由於商業方法專利多半與軟體有關，例如，透過軟體將某種演算法實施在特定的應用領域中，以產生實際有用的結果，因此，探討軟體專利問題，間接有助於未來商業方法專利的支持或修正，軟體專利存在與否，也牽動商業方法是否應該存在的問題。

第二項 對習知技藝人士的誤解

就「習知技藝人士」概念，學者 Dan L. Burk 與 Mark A. Lemley 認為存在謬誤之處，並指出誤用後對軟體專利的潛在影響。專利法雖未以產業細分適用條款，但學者研究後

²²⁶ See Julia Alpert Gladstone, *Why Patenting Information Technology and Business Methods is not Sound*, 25 HAMLINE L. REV. 217, 232 (2002).

²²⁷ *Id.*, at 222.

發現，法院其實在實務上對不同產業有不同的作法。例如，法院對軟體專利與生物科技專利在見解上剛好相反，對於軟體專利，「得以具體實施」及「最佳實施例」的標準不高，但進步性的認定從嚴。「得以具體實施」及「最佳實施例」受「習知技藝人士」的概念所影響，「習知技藝人士」的技術水準越高，揭露資訊較少也能符合「得以具體實施」的要求，但相對地進步性越不易達到。「習知技藝人士」影響專利的核駁，同時也影響專利通過後涵蓋的範圍，例如，均等論同樣需要在「習知技藝人士」的概念下推敲不同的發明是否實質相同。美國專利法§112 要求專利權人揭露充足的資訊，目的之一在允許競爭對手進行迴避設計，促成技術的進步，同時在專利過期之後，該技術得以落入公眾領域成為免費的公共財。軟體專利不要求揭示原始碼、流程圖或細部設計，主因在於法院認為從描述高階功能到實際完成軟體，高階功能比較重要，專利說明書只要交代清楚，剩下寫程式不過是「單純像打字員」按部就班便可完成的工作。法院認為，軟體專利的重點在於元件或步驟的組合，而非程式的細節，程式透過結構化的電腦語言撰寫，從想法到可執行的軟體，對習知技藝人士，落實上並非難事。另外，專利法§112 最佳實施例的規定，因軟體的特性也不適用，法院在 *Fonar Corp. v. General Electric Co.*²²⁸ 一案中表示，習知技藝人士代表具備寫程式的能力，軟體開發不需要進行實驗測量各種數據，而習知技藝在該領域中既已成為一般技能，便沒有在最佳實施例中揭露的必要。上述幾個觀點共同指出，法院對習知技藝人士定位為能力較高（由認定寫程式的技術層次較低而來），相對地，專利說明書揭露的要求較低²²⁹。

法院認為寫程式按部就班便可完成，與實務不符。事實上，以高階功能描述新的想法，對大部分軟體專案來說進度可能不到一半，在充足的資源下，軟體可能經過長期的開發，最後仍不見得順利完成。在開發過程中，程式設計技巧也扮演關鍵角色，因此，習知技藝人士的概念如果建立在程式設計像生產線製造般簡單，不僅誤解軟體產業的特

²²⁸ 107 F.3d 1543 (Fed. Cir. 1997).

²²⁹ See Dan L. Burk & Mark A. Lemley, *Is Patent Law Technology-Specific?*, 17 BERKELEY TECH. L.J. 1155, 1156-63 (2002).

性，程式開發階段的許多創新技巧也完全受到忽略²³⁰。開放原始碼模式下直接以原始碼競爭，軟體的優劣決定市場的佔有率，如果專利制度促進軟體產業競爭的只是創意，不問是否真正落實，忽略了軟體工程的重要性大於創意本身，便賦予壟斷形式，則軟體專利可能在保護面不完全妥當的情況下，產生不良的結果。

第三項 專利制度是否優於著作權制度

回到軟體應該如何保護以促進創新的問題，有人認為著作權法才是軟體創新的黑洞，應該採用專利制度取代著作權制度。論者認為，軟體界常看到程式設計師「重新發明輪子」，原因在於著作權法基本上禁止軟體還原工程，雖然在 *Atari Games Corp., v. Nintendo of America*²³¹ 與 *Sega v. Accolade*²³² 案中，為了學習對方未受保護的想法及程序，可以藉著作權法合理使用來阻卻違法，但成立與否仍須受到其他條件的限制，如為了相容性目的、無其他可行方式等。無法藉由還原工程學習他人的技術，程式設計師只好重新發類似的輪子。相反地，專利說明書可以有效揭露技術的特徵，讓他人能夠在迴避或改良中發明出更好的輪子，忽略他人專利不能免責的特性及交互授權的價值，則有助於強化參與專利制度的必要性，讓軟體技術的揭露與累積朝形成正方向循環²³³。可惜的是，論者未對軟體專利引發的爭議加以深究，其支持軟體專利的理由主要在於硬體產業與軟體產業的比較，以 Intel 晶片設計的進步速度遠超過 Microsoft 軟體進步的速度，背後的原因是硬體界著重專利，而軟體界以著作權為主，因此，專利制度應優於著作權制度。這樣的推論可能有速斷之嫌，事實上，論者提出開放原始碼運動是對著作權法形成技術黑洞的反撲，觀點可能比較可採，至於認為開放原始碼具「病毒感染力」且「反智慧財產權」，可能在誤判之餘，也忽略了探討對於促進產業創新方面，開放原始碼模式優於專利制度的可能性。

²³⁰ *Id.*, at 1192.

²³¹ 975 F.2d 832.

²³² 977 F.2d 1510.

²³³ See Mark A. Haynes, *Commentary: Black Holes of Innovation in the Software Arts*, 14 BERKELEY TECH. L.J. 567, 567-575 (1999).

第四項 專利制度因時制宜的必要性

專利制度的基本架構百年來沒有變化，而科技日新月異，過去二十年新興領域如資訊、通訊、半導體、生物科技等，已讓當初採機械觀的專利制度捉襟見肘。從質疑專利制度的本質出發，Burk 與 Lemley 指出，現有專利制度的規範對各種產業一體適用，忽略了不同產業本質上的差異，並不妥當。例如，生物科技與資訊科技的產業性質截然不同，用同一套制度推動其進步與創新，很可能陷入顧此失彼的困境。事實上，學者分析實際判例後發現，專利制度雖然形式上條文只有一套，法院其實對不同產業已採取著不同的侵害判斷標準，判斷歧異之處包括進步性、能否據以實施、書面揭示程度等²³⁴。如果實務上對不同產業實質上採取不同的判斷標準，是否應考慮在立法政策上調整專利法，根據產業特性量身設計不同的條款²³⁵？

同樣的問題放在軟體產業，專利制度是否需要因為開放原始碼模式異於傳統的專屬模式，而有差別待遇的必要？本文初步認為，不管原始碼是否開放，軟體專利在侵害鑑定的判斷上並沒有不同的必要，但如果站在開放原始碼社群的立場，制度中相關程序是否有修正的必要，例如專利相關費用的減免等，則有討論的餘地，這部分容後面建議方案中再述。



第五項 專利文件對程式設計師的助益

受到專利法中「禁止銷售原則」(on-sale bar)的影響，最高法院認為尚未開發程式碼之前，只要能描繪出大致的功能或進行商業交易上的邀約，該軟體發明便處於得申請專利保護的狀態。如果發明人等到程式寫完，才提出專利申請書，則可能已經違反禁止銷售原則下的時效規定²³⁶。在這種情況下，專利文件可能只含有高階的功能描述，缺乏實

²³⁴ See Burk & Lemley, *supra* note 221, at 1576-77.

²³⁵ Burk 與 Lemley 認為，如果為不同產業量身訂做專利法，可能遭遇更多建構法律體系上的困難，而最後實務上也不見得達到所要的結果，因此，建議形式上維持現狀，將裁量空間留給法院針對個案評量，可能比較妥當。

²³⁶ See Burk & Lemley, *supra* note 229, at 1170-71.

際的實施細節，程式設計師從閱讀專利說明書中，能夠學習多少新技術，其實值得懷疑。在軟體開發專案的生命週期中，讓前段偏向系統或架構上的規劃得以受到專利保護，忽略後段實作上的挑戰與技術，是另外一個問題。

自軟體專利成立以來，鮮少聽到程式設計師從專利文件中收到醍醐灌頂之效，資訊技術的精進主要仍在於解決各種實務上的問題，或者從一般媒體吸收或網路上交流。網路上常聽到的，反而是軟體專利如當頭棒喝一般，簡單得令人難以置信，程式設計師在訝異於他人取得顯而易見的專利後，往往從批評該專利到反對整個專利制度本身，分別提出不同程度的質疑。如前所述，軟體專利揭露程度較低，可能造成專利制度的傾斜，無法有效在促進公益與保障私利之間平衡，這樣的情況並受到「手段功能用語」(means-plus-function)²³⁷的影響而加深。以手段及功能描述專利技術，根據判例，保護的範圍將受到說明書中實際揭露結構的限制，因此，專利代理人在通過審核的前提下，往往刻意籠統敘述手段及功能涵蓋的標的，以期擴大專利保護範圍，結果造成程式設計師閱讀專利說明書時，反而不知所云，無法從中學習，相反地，透過還原工程可能還比較容易瞭解技術的內涵²³⁸。專利說明書對資訊科技有多少幫助，是程式設計師對目前專利制度的另外一點質疑。



第三節 開放原始碼模式與軟體專利

軟體專利引發的各種議放在開放原始碼模式下，有幾點值得分析。

第一項 軟體專利的威脅是否存在

雖然軟體專利引發爭議，但是否必然直接對開放原始碼模式造成威脅？值得進一步

²³⁷ 國內智慧財產局於2004年四月公布新修正的專利法施行細則，正式採納美國的作法，增加「手段功能用語（means plus-function language）做為界定專利申請範圍的方式之一。其特色是不須詳細敘述元件的結構或材料，而僅須以實現某一特定功能的手段或步驟的方式來表現，可省略相關複雜說明，大幅簡化申請專利範圍的撰寫。參見 93/04/09 經濟日報 11 版。

²³⁸ See Burk & Lemley, *supra* note 229, at 1164-66.

討論。如前所述，因前案資料不完備而通過的許多軟體專利，固然造成專利品質的下降，但有學者認為，這些品管不良的專利如果進入法院，在法院對進步性採取高標的現況下，專利存活的機率應該不高，因此，無需高估垃圾專利的負面影響。即使前案搜尋不易，涉訟時雙方資源遠超過專利審查人員核駁一個案子所花的時間，如果前案存在，也應該找得出來。加上法院看重軟體專利對高階功能的描述，而非程式技巧的細節，許多根本解決相同問題的軟體，彼此之間的類推適用可能顯而易見，法院最後判定專利無效的機率很高²³⁹。不過，對開放原始碼社群，軟體專利無效的可能性高，並不等同於威脅因此降低，理由有二：(1)專利訴訟常常成為商業競爭的策略手段，即使明知勝訴機會不大，興訟打擊對方的情況也屢見不鮮。專屬軟體公司如果受到開放原始碼模式的威脅，是否可能採取訴訟手段打壓，並非毫無可能。(2)開放社群的程式設計師多半未從原始碼中獲得實質報償，如果涉訟，也將缺乏足夠的資源據理力爭。

在前案搜尋的部分，開放原始碼社群倒是可能發揮螞蟻雄兵的力量，有助於讓不當的專利中箭落馬。例如，公眾專利基金會於 2003 年底成立，為一非營利法律服務組織，致力於保護公眾免於專利系統的侵害，其中，尤以不當核發的專利與專利政策的缺失為首。2004 年四月，公眾專利基金會針對 Microsoft 取得 FAT 檔案系統的專利，向專利局要求重審，理由是根據找出的前案，該專利無法通過進步性的門檻，因此建議專利局應該撤銷專利。除此之外，公眾專利基金會認為哥倫比亞大學對同一件發明取得數個重複的專利，容有爭議空間，其他還有幾個專利也需要評估是否要求重審。藉由徵詢網路上的意見，公眾專利基金會希望能逐漸清除不當的專利²⁴⁰。美國專利局接受公眾專利基金會提出的理由，決定重審其所提的專利爭議²⁴¹，可以證明公眾專利基金會的運作模式開始發揮一定的效用。

²³⁹ See Burk & Lemley, *supra* note 229, at 1168-69.

²⁴⁰ See PUBPAT, *PUBPAT challenges Microsoft patent to protect competition in software market*, Apr. 15, 2004, at <http://www.pubpat.org/Protecting.htm> (visited Apr. 21, 2004).

²⁴¹ See PUBPAT, *Patent Office Grants PUBPAT Request to Reexamine Cotransformation Patent*, May 10, 2004, at http://www.pubpat.org/Axel_Reexam_Granted.htm (visited May 11, 2004).

第二項 軟體專利是否涵蓋原始碼的使用

有人認為，「原始碼」可能非軟體專利中所稱的「軟體」，因此軟體專利並未直接影響開放原始碼社群。關於原始碼的複製、修改、散布是否可能侵害他人的專利？論者認為，軟體專利在專利局眼中，「軟體」的定義其實並不清楚，所指的是電腦執行時使用的「目的碼」？還是包含編譯器處理之前的「原始碼」在內？認定不同，侵害軟體專利的樣態也可能不同。例如，程式設計師甲開發軟體，善意侵害乙的軟體專利，甲之後加入開放原始碼專案，貢獻自己的原始碼，所公布的原始碼為善意第三人丙下載，丙研究原始碼後發現不符合需求，予以廢棄，乙是否得主張丙下載原始碼的行為構成專利的侵害²⁴²？如果甲公布「目的碼」為丙下載，是否構成侵害軟體專利？

對於上述問題，可以從探討原始碼與軟體的關係著手。如果原始碼儲存在硬碟上，等同於「電腦讀取媒體上的軟體」，則單純下載行為即構成專利的侵害，因為下載行為等於製造一份受專利保護的標的。就這個角度，開放原始碼社群日常的運作等於暴露在侵權的風險下，程式設計師即使不把原始碼拿來編譯、執行，都可能使用、製造了受專利保護的軟體。一般社群人士對軟體專利往往偏重於質疑其進步性，似乎忽略了此處所述「原始碼如果等同軟體」所隱含的風險。在美國專利法上，專利標的可分為：程序(process)、機器(machine)、製造品(manufacture)、和物之組合(composition of matter)，最高法院對專利適格標的採廣義見解，讓「太陽底下由人製成的任何事物」都能納入某一種類別，而受到專利保護。論者認為，軟體專利放在各個類別下分析，可以得到只有「目的碼」符合專利侵害構成要件的結論，單純原始碼或許仍負擔輔助侵害或間接侵害的風險，但並沒有直接侵害的可能²⁴³。分析軟體專利侵害的樣態，如採上述分類進行，可簡述如下：(1)軟體與機器：在 State Street 案中，聯邦法院提出新的「實際效用說」，軟體如果產生「實用、具體、有形」的結果，包含該軟體的機器便可成為適格標的，受專利

²⁴² See Daniel Lin, Matthew Sag & Ronald S. Laurie, *Source Code Versus Object Code: Patent Implications for the Open Source Community*, 18 SANTA CLARA COMPUTER & HIGH TECH. L.J. 235, 236 (2002).

²⁴³ *Id.*, at 237-40.

保護。電腦「產生」實際效用，必須執行「目的碼」，原始碼在編譯之前，並無法使電腦執行特定功能，因此，在此說下「軟體」的概念只限於「目的碼」而非「原始碼」。

(2)軟體與方法：State Street 案發生後一年，聯邦法院在 AT&T Corp. v. Excel Communications, Inc.²⁴⁴案中，把相同的概念類推適用到方法專利。法院認為方法專利不要求在實體上轉換標的物的性質，重要的是方法中所用的數學演算法，是否產生實際有用的效果。採取方法專利的觀點，專利侵害只有在實施該方法時才成立，亦即，必須執行該方法專利保護的軟體，而軟體執行所涉及的是「目的碼」而非「原始碼」。(3)軟體與製造品：將軟體存在磁片或其他媒體申請專利，過去曾視為印刷物而不符合適格標的，但聯邦法院對 In re Beauregard²⁴⁵一案的判決，讓專利局改變審查基準，將電腦可讀取媒體上的軟體，一同視為製造品而成為專利適格標的。法院的見解為，磁片上的程式雖然像靜態的印刷物，但載入電腦後，具有動態執行改變電腦記憶體資料的能力，因此，與印刷物的特性有別。根據此項見解，載入電腦得以執行的是「目的碼」而非原始碼，因此，軟體專利的「軟體」不包含原始碼在內²⁴⁶。

簡言之，不管採取哪一種法定適格標的類型，軟體專利的「軟體」指的是據以執行的「目的碼」，不包括原始碼在內。在這種情況下，論者認為，開放原始碼社群受到軟體專利的威脅，可能沒有想像中嚴重，因為原始碼的使用、複製、散布等，除非涉及輔助或誘導的行為，否則並沒有侵害他人專利的顧慮。相對地，目的碼的使用、製造、販賣等行為，則帶有侵害他人專利的風險。不過，散布或販賣原始碼時，如果原始碼視為專利標的部分原件，或者該行為視為主動誘導他人侵害專利權，而當事人對侵害軟體專利知情，則可能無法免責²⁴⁷。

軟體專利是否如上所述，對開放原始碼的運作影響不大？值得進一步探討之處包括：(1)原始碼最終的目的是產生可執行的軟體，即使善意侵害他人軟體專利可以免責，

²⁴⁴ 172 F.3d 1352, 1358 (Fed. Cir. 1999).

²⁴⁵ 53 F.3d at 1584.

²⁴⁶ See Lin, Sag & Laurie, *supra* note 242, at 240-46.

²⁴⁷ *Id.*, at 253-56.

知情之後，軟體開發仍面臨迴避設計的問題，如果無法迴避，投資的心力可能全盤皆墨，與其事後面臨困境，不如在原始碼階段即出現侵權爭議，檢驗專利的效力。原始碼與目的碼只有「編譯」一線之隔，以此區分是否為軟體專利保護標的，從技術觀點看，意義可能不大，如果以此支持侵權的抗辯，開放原始碼社群似乎只取得鴟鳥式的短期效果，專利侵權如果成立，目的碼無法使用、製造、散布，唇寒齒亡，原始碼也將失去存在的價值。因此，對開放原始碼社群，同意軟體專利只包含目的碼而未涵蓋原始碼，可能弊多於利。(2)區分原始碼與目的碼的二分法，實務上並非完全成立。「編譯」之所以需要，主要在於執行效率的提升，軟體以目的碼形式流通，為的是讓電腦直接執行，不需要每次從原始碼編譯成目的碼再執行，許多常見的電腦語言採取這種方式運作，但並非沒有其他的可能性。事實上，許多電腦語言下的軟體直接以原始碼流通、執行，在這種情況下，執行時仍需要將原始碼轉換成目的碼，只不過每次執行時才「編譯」，目的碼形式隱藏在系統裡，使用者無所察覺。這種每次「即時編譯」的特性稱為「直譯」，看似沒有效率，但透過許多軟體技術仍可讓整體效能維持在相當不錯的程度，雖然無法與「編譯」相比，但直譯式語言以原始碼直接做為流通與執行的形式，在程式的可讀性上較佳，完全將人類無法閱讀的目的碼排除在外。簡單地說，採用直譯式語言，以原始碼的觀點，沒有所謂封閉式的軟體存在。電腦上只要裝有該語言的「直譯器」，取得原始碼即可加以執行，在這種情況下，以能否「執行」來區分軟體專利下「軟體」的定義，在目的碼之外是否應該包含原始碼，可能更加缺乏實益。如果堅持以「可直接執行」的特性描述軟體的形式，可能得到不同電腦因是否安裝直譯器，而對軟體有不同定義的結果，亦即，軟體專利的侵害認定必須將不同的電腦設定考量在內。

根據以上分析，軟體區分成原始碼與目的碼，對解說資訊技術時有其必要，但對於探討軟體專利引發的法律爭議，其必要性則有待商榷。本文認為，軟體專利應該涵蓋目的碼與原始碼兩種形式下的軟體，如果涉及專利侵害，開放原始碼社群並不能置身事外，亦即，原始碼的使用、製造、散布等行為，仍然具有著侵害他人專利的風險。因此，軟體專利如果對專屬軟體公司帶來負面的效果，對開放原始碼模式可能放大其效果，因為原始碼公布出來，專利權人隨時可以搜索相關軟體，進行專利侵害鑑定。相對地，專

屬軟體其原始碼封閉，他人要進行鑑定的難度較高。

第三項 專利制度下的差別待遇

除了侵害鑑定的門檻不同，軟體專利對專屬軟體模式與開放原始碼模式，還有其他不可避免的差別待遇。差別待遇來自於模式上的不同，雖非制度設計上的公平性問題，但考量開放模式的價值，仍有討論的必要，並可成為修正專利制度的參考。軟體專利特別不利於開放模式有下列幾點：(1)新穎性。專利技術申請保護之前，如果公諸於世，則失去新穎性，無法取得專利。開放原始碼模式為了分享與合作，原始碼一開始即公開，後續改良如有技術特徵，也直接反映在原始碼上，就此，在申請專利上顯然無法通過新穎性的要求。(2)進步性。開放原始碼在網路效應下，軟體常常出現逐步但快速的進化，新的想法與改良來自於四面八方，聚沙成塔，實際上更符合軟體工程上理想的開發模式。但以此方式，每次進步的幅度可能並不明顯，相對專屬軟體多半累積足夠的技術特徵後才推出新版，兩相比較，開放原始碼軟體如果真的申請專利，符合進步性要件的可能性也較低。(3)原始碼的揭露。原始碼雖然不是促進軟體進步的唯一媒介，但對大部分程式設計師，原始碼顯然具備正面的臨摹價值，對軟體產業的進步有益無害。目前專利說明書並未要求揭露原始碼，對專屬軟體模式，未構成任何負擔，相對地，開放原始碼軟體如果取得專利，或採開放模式前取得專利，公布原始碼的動作將超出專利審核要件的要求。這一點無礙於專利的審核，但可看出專利制度在促進技術的進步上，原始碼的封閉模式與開放模式有不同的貢獻度，而貢獻較多者並未得到任何審核上的優惠。專利制度是否值得據此採取差別待遇，仍有待研究。值得思考的是，回溯到專利法的精神，在於促進技術的進步與長期的公益，如果某種手段能夠更加落實專利法目的，專利審核上採取不同的標準是否合理？是否有助於目的的達成？(4)成本負擔：專利技術從紀錄研發過程、撰寫專利說明書、聘請專利代理人、提出申請到取得專利之後的維護，皆需要額外的人力與財力，如果涉及侵權糾紛，後續聘請專利律師及相關訴訟費用更加可觀，這些成本對開放原始碼社群來說，負擔過於沈重。加上採用開放原始碼，本身並非以財務收入為考量，專利授權的誘因無法成立，這或許也說明了開放原始碼運動長期發展下

來，為何在專利的申請及保護上付之闕如。據估計，一個專利的相關費用約二萬五千美元，以微處理器為例，如果本身缺乏專利，進入市場的門檻約上億美元，軟體界雖然沒有類似的數據可供參考，但可發現為了在專利制度下達到防禦效果所投入的資源，也逐漸超出一般小公司所能負荷的程度，更遑論一般個人²⁴⁸。

第四節 制度改革的建議

到目前為止，已有不少學者呼籲專利制度應該加以改革，並提出不同的看法，歸納來說，至少包含下列幾項：

第一項 暫停或取消軟體專利

美國是第一個將可專利標的涵蓋到軟體和商業方法的國家，由於爭議不斷及效果未明，因此，學者認為國會應要求專利局定期檢討專利制度的效益與衝擊，特別是針對近幾年不斷擴張的部分，應該採取經濟的角度加以分析，專利局如果無法對擴張的必要性提出最低限度的證明，國會應考慮加以取消，以避免出現不當的壟斷形式²⁴⁹。如果採取這種措施，對開放原始碼社群的持續發展可能有利，可以暫時免除軟體專利的威脅，同時，隨著開放原始碼模式下軟體不斷地增加，前案技術持續累積，將來再開放軟體專利的話，可能影響程度也會相對縮小。

雖然資訊界有人認為，美國軟體專利已經到了每寫一千行程式就可能侵害他人專利的程度，百分之九十的技術屬於顯而易見而不應賦予專利保護，整個制度應該反省檢討等，但同時也有人指出，不少新的軟體的確帶有高度技術性，需要受到制度上的保障與獎勵。例如，MP3 音樂格式在誕生之前，發明人的確投入相當可觀的研發與時間，軟體專利的出現確實建立相當程度的誘因²⁵⁰。因此，暫停或取消軟體專利的爭議性及難度可

²⁴⁸ See Gladstone, *supra* note 226, at 232-233.

²⁴⁹ LAWRENCE LESSIG, THE FUTURE OF IDEAS: THE FATE OF THE COMMONS IN A CONNECTED WORLD 259 (2001).

²⁵⁰ See Eugenia Loli-Queru, *Not Everyone Understands the Patent Situation*, May 15, 2004, at

能最高，原因除了缺乏充足的實證資料估算制度引發的利弊外，直接對專利制度中既得利益者構成挑戰，也讓此一建議相對於其他方案更不容易達成。

第二項 提升軟體專利的品質

在軟體技術早在軟體專利之前出現的情況下，專利申請時如何有效地找出前案做為核駁的基準，便成了影響軟體專利品質的一項關鍵²⁵¹。前案搜尋的部分除了是專利審查員的責任範圍，有學者認為也應該賦予申請人部分責任，在缺乏完整的前案資料庫之前，一同對軟體專利的品質分擔把關的責任。除了前案之外，進步性的要求也應該提高。例如，跟生物科技產業相比，軟體的創新比較模糊，在核准專利保護之前應該格外審慎。進步性的認定係根據習知技藝人士的判斷而來，因此，何謂習知技藝人士應有更明確的參考標準。為避免專利涵蓋範圍失之過廣，專利說明書揭露的程度及均等論的適用限度，都有列入修正的必要，以便將軟體專利限縮在合理的保護範圍²⁵²。

品質不良的另外一個因素在於人力不足，針對這一點，2004年三月美國眾議院於以懸殊的票數通過新的「專利與商標費用現代化法案」²⁵³，根據法案，專利相關收入將盡量留在專利局本身，作為提升人力、擴大編制的基礎。面對科技的日新月異，專利審查的複雜度高於以往，審查期間冗長而且不確定性高，一直為人所詬病²⁵⁴。以半導體為例，平均三到四年的審查期明顯落後市場變化的步調，透過該法案，增加人力之外將有機會強化前案資料庫的建置。不過，科技的快速進步是否持續拉大專利品質的落差，仍有待

http://www.osnews.com/story.php?news_id=7055 (visited May 26, 2004).

²⁵¹ 例如，網路主機代管為客戶提供子網域的功能，在業界沿用數年後仍出現專利侵權訴訟，業者認為，專利局當初之所以核准，主因在於許多習知技術並未以專利的形式儲存在前案的資料庫中。相關報導可參見 Dennis McCafferty, *Subdomain Patent Sparks Concerns*, Mar. 26, 2004, at <http://thewhir.com/features/subdomain.cfm> (visited Mar. 27, 2004).

²⁵² See Burk & Lemley, *supra* note 221, at 1688-89.

²⁵³ Patent and Trademark Fee Modernization Act of 2004, H.R. 1561, 108th Cong. (2004)

²⁵⁴ 關於美國專利局內部作業、審查員的養成及專利審查的不確定性，請參見 Scott Wolinsky, *An Inside Look At The Patent Examination Process*, Oct. 2003, at <http://www.volpe-koenig.com/showarticle.asp?Show=12> (visited May 14, 2004).

觀察。據估計，藉由保留盈餘轉化成內部資源，美國專利局在五年內可以多通過十四萬個專利，以這個數量，應可抒解目前積案的困境²⁵⁵。專利核准數量的提高，代表專利制度效益的放大，應准予專利而未准予專利的案件，能夠得到較佳的時效，但對尚有爭議的專利類型，如軟體專利、商業方法專利，反面來看也可能代表負面效益的擴大。

第三項 縮短專利保護期限

網際網路的普及強化了軟體擴散的速度，軟體如果開發成功，成為市場主流通常不過三到五年的光景，如果需要加以保護的話，類似的年限應該較為恰當。專利保護年限如果符合軟體界的生態，應有助於達到以壟斷刺激創新但又不至於過當壓制了後續創新的可能²⁵⁶。如果採用這種做法，對開放原始碼社群來說，釋出的創作自由度雖然無法跟廢除軟體專利相比，但受到軟體專利的鉗制應能有效減緩，至少，基礎性專利所構成的阻礙可以在較為合理的期限內消失。

過去著作權法擴大範圍涵蓋電腦軟體後，在軟體專利定案前曾出現類似調整保護期限的建議。學者從避免市場失靈的角度出發，建議把著作權法放在一旁，根據電腦軟體的特性量身訂做一套新的保護措施，讓制度與實務之間更加契合。例如，電腦軟體具有可執行的特性，異於一般靜態的著作，軟體中使用者操作的人機介面可能需要大量的研究，才能找出最佳的設計，但新的操作介面問世之後，模仿跟進的障礙並不高，因此，如果缺乏適當的法律保護，廠商投入的研發資源可能難以回收，願意再投資的意願也將跟著下滑，長期下來，將不利於創新的出現。對於類似問題，論者認為針對值得保護的客體賦予適當的排他期間，將能夠預防市場的失靈，同時，透過對軟體以個別處理的方式，也不需要冒著不斷擴張著作權法引發其他副作用的風險²⁵⁷。惟實務界後來並沒有如

²⁵⁵ See David Lammers, *Legislation advances to speed patents*, Mar. 15, 2004, at <http://www.eetimes.com/sys/news/OEG20040315S0013> (visited Mar. 18, 2004).

²⁵⁶ See Margaret Kane, *Bezos calls for patent reform*, Mar. 10, 2000, at <http://news.zdnet.co.uk/software/0,39020381,2077612,00.htm> (visited Apr. 1, 2004).

²⁵⁷ See Pamela Samuelson, Randall Davis, Mitchell D. Kapor & J.H. Reichman, *A Manifesto Concerning the Legal Protection of Computer Programs*, 94 COLUM. L. REV. 2308, 2310-16 (1994).

上述建議，為電腦軟體發展出量身訂做的保護機制，而是讓著作權法下力有未逮之處，逐漸進入專利的保護範圍，上述以「適當排他期間」增加保護的概念，則遺漏在軟體專利應有的配套考量之外，讓專利期間維持對所有產業一體適用，而對軟體產業產生負面影響。

第四項 專利的合理使用原則

合理使用原則是著作權法底下的概念，需要在專利法討論，原因之一在於專利說明書不需要揭露原始碼，而可能阻礙資訊技術的進步。過去在 *Northern Telecom, Inc. v. Datapoint Corp.*²⁵⁸ 一案，聯邦法院駁回地方法院的見解，認為專利權人無需揭露原始碼，也可以符合「據以實施」與「最佳實施例」兩個審核要件，至於所謂習知技藝人士的水平，應該從個案判斷。法院見解在 1990 年形成判例，之後法院便未就是否應揭露原始碼一事再做考慮²⁵⁹。實務上，法官通常沒有技術背景，法官可能每隔幾年才遇到一個專利訴訟，專家證人雖然提供意見，但法官本身仍需要在相當時間內掌握專利法與該技術領域的重點，在這種情況下，其判斷的正確性便有商榷的餘地²⁶⁰，或者，至少應有在不同意見中重新檢討的必要。目前軟體同時可受營業秘密法、著作權法及專利法的保護，原始碼也不需要揭露，是否可能保護的密度過高？

為了避免過度保護，壓制後續他人創作或使用上應有的空間，合理使用原則應納入考慮，以減緩智慧財產權制度的保護強度。例如，在市場競爭上，還原工程讓競爭對手得以一窺內部以便進行迴避設計，對促進競爭有正面的作用。在著作權法領域，法院接受將電腦軟體的還原工程視為合理使用的樣態之一，但合理使用的概念在專利法領域並不存在，實務上也未發展出類似的原則。由於專利說明書不需要揭露原始碼，軟體在外觀上不具實體，也不像一般機械裝置得以拆開來一探究竟，因此，透過專利來瞭解軟體技術便受到相當的限制，專利概念下以揭露技術內涵換取獨佔權利的機制，在軟體專利

²⁵⁸ 908 F.2d at 943 (Fed. Cir. 1990).

²⁵⁹ See Burk & Lemley, *supra* note 229, at 1199-1200.

²⁶⁰ *Id.*, at 1197.

中似乎有所失衡，在無法取得原始碼的情況下，還原工程可能是唯一徹底瞭解專利技術的手段。不過，採用還原工程有侵害專利之嫌，專利法條文規定的侵害類型廣泛，使用、製造、販賣、進口等都包括在內，還原工程中採用反組譯(decompile)方式時，很可能即構成「使用」樣態，或者在記憶體中構成「重製」，如果不是預先取得授權，則可能已侵害他人專利。由此可以看出，一方面還原工程對軟體界有存在的必要，得以探求軟體專利的技術特徵，一方面還原工程則受到專利侵害的限制，而難以安全實施²⁶¹。在兩難的情況下，制度上應該如何採用著作權法下合理使用的原則，以避免專利制度的過度保護？

事實上，還原工程及揭露原始碼只是針對軟體而來的問題，在軟體領域之外，專利法中是否應該增加合理使用原則並一體適用，則是另外一個更根本的問題。學者認為，過去假設專利權人將有效運用授權發揮專利技術的價值，近幾年受到可專利範圍不斷擴張的影響，市場可能已出現失靈的現象，讓專利制度無法達到原本立法的目的。因此，從法律經濟分析的角度，為了解決市場失靈的問題，有必要考慮移植著作權法下合理使用的概念，在專利法領域定義不同的判斷準則²⁶²，讓未來的發明人不會在過度保護專利權人的情況下，失去創新的空間。目前專利局、國會及法院等並非對專利制度可能有礙科學進步視而不見，但思考上多半侷限於制度內操作細節的調整，而未能跳出框框重新審視受保護及不受保護之間的界線為何。專利數量大量成長，後續創新如果可能出現，面對諸多相關專利的授權成本過高，也可能變得難以實施，整個專利制度反而形成對大眾福祉不利(anticommons)的結果。為了避免情況持續惡化，應有必要在專利制度中增加得以侵權的合理使用原則²⁶³。

雖然著作權制度與專利制度有所不同，例如，著作權一開始取得保護的門檻很低，但後續不少原則或學說可以將權利限縮在不至於對公眾有害的範圍，相反地，取得專利保護的門檻較高，需要經過專利局審查其是否符合相關要件，權利的範圍也在事前審查

²⁶¹ See Burk & Lemley, *supra* note 221, at 1589-92.

²⁶² See O'Rourke, *supra* note 218, at 1177.

²⁶³ *Id.*, at 1178-80.

中決定，因此，事後限縮的概念較不發達，其他如權利保護期間的長短、獨立創作是否可能侵權等，也有不同的差異²⁶⁴。不過，著作權法及專利法在立法的目的上是相同的，最後皆以增進公眾福祉為依歸，也沒有理由兩者在誘因設計上必須採取不同的作法²⁶⁵。

著作權法出現合理使用原則，從經濟分析的觀點，所考量的是市場失靈時可能過度保護了著作權人而一般大眾無法有效運用其著作²⁶⁶，目前專利制度中雖然存在一些限縮專利權以避免市場失靈的作法，如要求重新審查專利的效力、主張反均等論(reverse doctrine of equivalents)、主張侵害對象為阻擋性專利(blocking patent)、主張實驗用途而免責、主張授權人為專利濫用(patent misuse)等，但深入分析後可以發現，這些原則並無法等同於著作權法中的合理原則，也無法解決所有市場失靈的問題或各有不同的侷限。例如，重新審查專利只能得到有效或無效的結果，但當事人需要的可能是不受追訴地使用部分技術的空間，主張實驗用途則無法讓還原工程的使用完全免責等²⁶⁷。綜合上述考量，專利制度應考慮合理使用原則的採用。至於採用哪些原則，學者提出五個判斷因素：(1)專利侵害促進技術進步的程度，(2)專利侵害的目的，(3)市場失靈對能否取得授權的影響力，(4)專利侵害對創作誘因降低的程度以及對社會整體利益提高的程度，(5)專利技術的本質²⁶⁸。

上述原則是否為實務界所接受，以及如何在個案中加以落實，皆有待考驗。從本文的觀點，提出合理使用原則是跳出框框的第一步，藉由釐清專利保護與公眾福祉之間的界線，在壁壘分明下賦予合法的漏洞或增加判斷的彈性，應有助於緩和專利密度過高的副作用。稍後的建議作法或許可視為再往前一步，將考量的重點跨過界線本身，以公眾福祉為出發點，思考開放模式應用在專利領域的可能性。

第五項 降低訴訟的誘因

²⁶⁴ *Id.*, at 1184-87.

²⁶⁵ *Id.*, at 1196.

²⁶⁶ *Id.*, at 1188.

²⁶⁷ *Id.*, at 1192-95.

²⁶⁸ *Id.*, at 1205.

專利訴訟的成本與潛在收益十分龐大，動輒上億的賠償金額時有所聞，考量軟體開發的成本較低，有學者認為可以改用收取合理權利金及強制授權的方式解決專利侵害糾紛，以消弭投機暴發心態下的濫訴之風²⁶⁹。不過，這種方式對開放原始碼社群來說可能助益不大，在自發無償貢獻的程式設計師眼中，涉及任何訴訟都將是無妄之災。或許有幫助的是，在降低訴訟的誘因之後，如果傳統專屬軟體受到的專利糾紛因而降低，開放原始碼社群所需顧慮的軟體專利也間接地跟著減少²⁷⁰。只是如何有效降低訴訟的誘因，實務上判斷或操作的細節為何，則可能有待從經濟分析的角度進一步研究。

第五節 建議作法

開放原始碼運動到目前為止，有效累積了各方的原始碼，下一階段是否可能出現類似的機制，在著作權外累積專利權，是本節思考的重點。

第一項 理念

智慧財產權保護範圍的不斷擴張，可視為人類智慧資產的另一種「圈地運動」(enclosure movement)，在財產權的保護下，各種智慧資產有效地開發、運用，固然有正面的經濟效益，但不斷擴大保護範圍，智慧資產落入私人手中，如果造成長期壟斷，是否應該思考「國家公園」存在的可能，以保障日益縮減的公眾領域²⁷¹？過去，發源於十五世紀的圈地運動透過財產權的賦予，讓無主荒廢的土地在私人手中，大幅提升其生產力，以擺脫「公共財的悲劇」(tragedy of commons)。在以前缺乏財產權的情況下，土地不是過度開發就是任其荒廢，造成資源的浪費，賦予財產權保護的機制後，雖然不免出現大地主以強欺弱的現象，但農作及畜牧產量的增加，減少了飢餓的人口，對整體社會



²⁶⁹ See LESSIG, *supra* note 249, at 259~261.

²⁷⁰ See Sean Gallagher, *Patents of Mass Destruction*, Sep. 9, 2003, at <http://www.eweek.com/article2/0,1759,1516486,00.asp> (visited Apr. 1, 2004).

²⁷¹ See generally James Boyle, *The Second Enclosure Movement and the Construction of the Public Domain*, 66 LAW & CONTEMP. PROB. 33 (2003).

福祉顯然利多餘弊²⁷²。

財產權制度經過歷史證明有效，因此，主張強化智慧財產權者，認為智慧財產權的保護應該延續同樣的思維。不過，這樣的推論有值得討論的空間。首先，與實體財產權不同的是，智慧財產權不具實體，沒有與他人競爭使用的問題，使用一份智慧財產並不影響他人仍可使用相同的智慧財產，因此，「公共財的悲劇」中相同資源過度使用的顧慮並不成立，如果是「反公共財的悲劇」(tragedy of anticommons²⁷³)中，因大家都有排他權以致於無人得以使用共有的資源，這種情況也不會出現。但相對地，智慧創作因不具實體，同樣無法排除他人的使用，以著作為例，數位化之後盜版比以前容易，創作心血如果無法保護，創作者一開始可能便裹足不前，社會整體的創作量可能因而下降，因此法律有必要介入，讓智慧財產權的設立與保護應運而生。另外，盜版的成本越低，保護的強度相對也可能需要提高²⁷⁴。

相對於私人財產，公眾領域的資源可供大眾使用，而不需要事先取得私人的許可，例如，實體世界中的道路、公園等。在數位世界中，網際網路、開放原始碼可視為公眾領域的資源²⁷⁵，這些資源提供了大眾創作的養分與空間，如果網際網路不是對外開放、遵守公開標準，很可能無法達到今日普及的程度，進一步成為各種加值運用的平台與管道。對於架構在網際網路上的各種應用，網際網路如同一種「數位環境」，有其存在及維護的必要，開放原始碼形成的公眾資源，可視為軟體技術蓬勃發展的肥沃土壤，也值得提升層次將整體視為另外一種「環境」。對數位資源採取環境觀點，為的是思考「環境保護」的必要，以及私人財產權不斷擴張後的界限？跨過了界限，公眾領域的環境權是否便受到侵害，而不利於整體社會長期發展的福祉？在環境保護的觀點下，學者認為公眾領域應該出現類似綠色和平組織、環境防禦基金、自然保育等組織²⁷⁶，以便在第二

²⁷² *Id.*, at 35.

²⁷³ See generally Michael A. Heller, *The Tragedy of the Anticommons: Property in the Transition from Marx to Markets*, 111 HARV. L. REV. 621 (1998).

²⁷⁴ *Id.*, at 41-43.

²⁷⁵ *Id.*, at 62-63.

²⁷⁶ *Id.*, at 73.

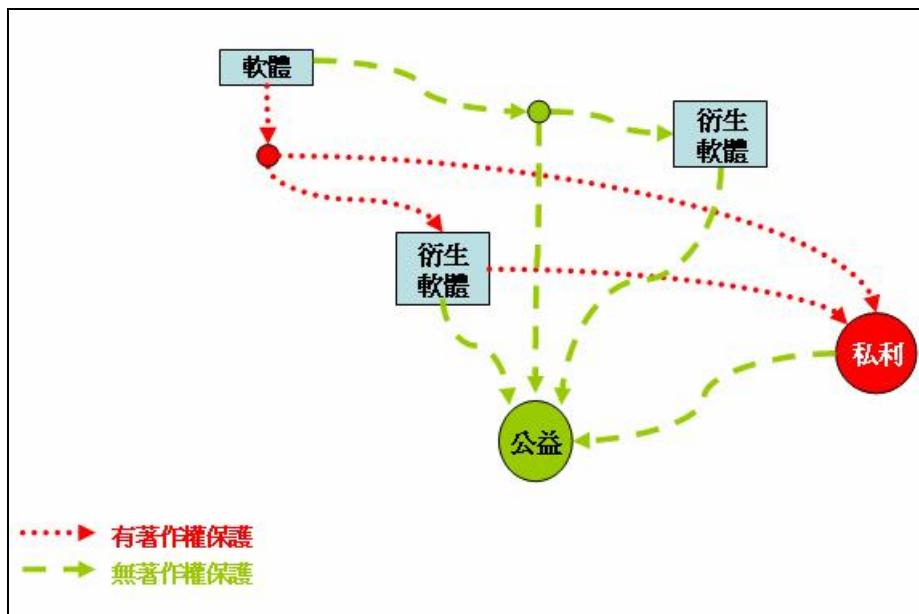
次圈地運動中，為公眾創作保留必要的生存空間。

另外在理念上，本文認為「開放」並非對抗「專屬」或「封閉」，彼此並非水火不容，開放所代表的是一種額外的可能性。開放原始碼提供開放的可能，並非強迫採取開放模式，這一點時而受到誤解，例如，曾有學者以「法律上的開放」與「事實上的開放」區分，分析開放原始碼社群形成事實上的開放後，無法推論出法律上開放的必然，認為開放原始碼社群要求所有軟體必須揭露原始碼，有其值得批評之處²⁷⁷。以此論述，可能有所誤解，開放原始碼的理念並非用來拆除現行的法律制度，而是同樣在著作權法下提出另外一種可能，不強迫理念不同者必須放棄其他任何保護原始碼的權利，如據為營業秘密。對於軟體專利，本文與開放模式的理念相同，建議的作法並非用來推翻既有的制度，而是在既有的制度下探討另外一種可能。

第二項 開放原始碼模式的啟發

智慧財產權制度的目的在於取得公益與私利之間的平衡。如前所述，在財產權制度下，透過私人財產的保障，能促進資源的有效運用，間接達到提升整體社會福祉的效果。智慧財產權延續傳統財產權觀點，希望同樣採取間接方式，保障個人智慧創作的成果，以短期私利的追求促進長期公益的累積。從公益與私利的觀點，開放原始碼模式可以視為在既有制度中開創出「第三種」可能，希望保護公眾的智慧財產權，以避免私人智慧財產權受到過度保護。關於軟體創作與著作權保護，第一種可能方式是「無著作權保護」，創作成果歸為公共財，這種模式直接對公益有利，但對創作者個人的獲利相當有限，也可能落入他人搭便車的困境。第二種可能方式為「有著作權保護」，這是目前運作的制度，公益的達成間接來自於私利的誘因。進一步論述之前，現行制度可以採下圖表示：

²⁷⁷ See generally Mathias Strassery, *The Case Against Open Sources*, 2001 STAN. TECH. L. REV. 4 (2001).

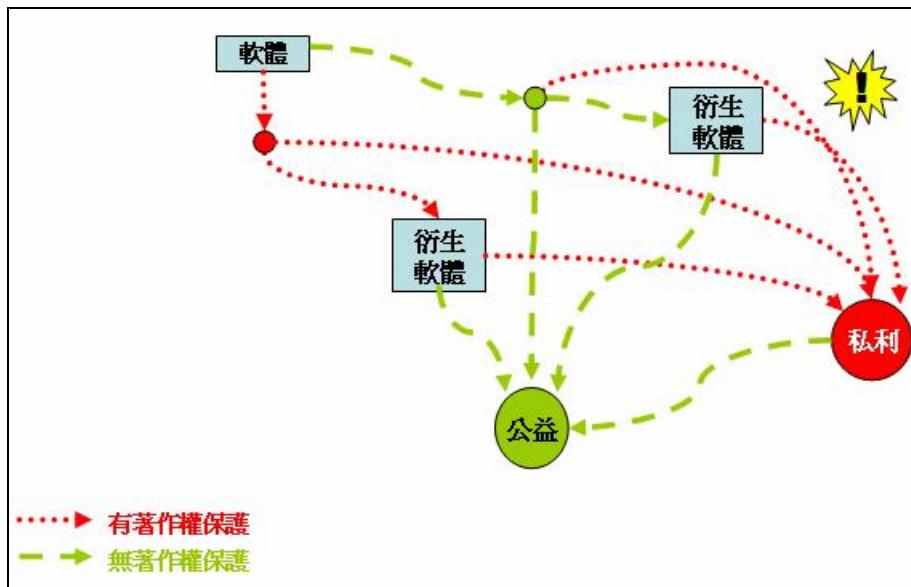


圖五 著作權保護與促進公益、私利的關係

上圖為傳統上軟體受著作權保護的狀況，原軟體及衍生軟體視為著作及衍生著作，可受著作權法的保護。軟體廠商透過授權方式，向使用者提供可執行的目的碼，交換使用者支付的價金，經濟上的誘因及市場上贏者全拿的特性成為驅動軟體產業成長的主要因。

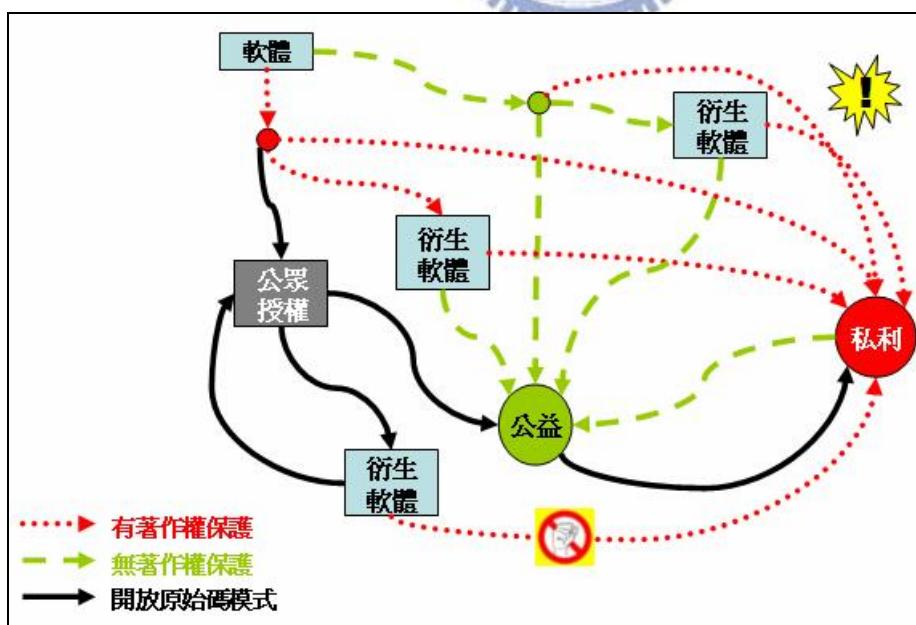
著作權法採創作主義，程式設計師寫出程式碼即享有著作權的保護，不過，所有權人可以放棄相關權利，讓自己的創作實質上等同於公共財，供他人自由使用。自動放棄著作權在其他創作類型似乎比較少見，但在軟體界，並非難以理解。與其他文學或藝術創作相比，軟體的進化受益於軟體元件的「重複使用」與「逐步改良」，前者節省重新發明輪子的資源，後者代表衍生創作的必要，在這兩種情況下，權利的放棄似乎有助於軟體的進步。Unix 領域在早期階段，許多程式設計師重視原始碼的分享與流通勝於權利的伸張，反應出軟體界運作的本質，同時，對許多程式設計師而言，智性上的挑戰與同儕間的肯定，比財務上的回饋來得重要。因此，部分開放原始碼人士對智慧財產權制度抱持敵視的態度，多少可以理解。如果所有的程式設計師認同原始碼一律成為公共財的理念，放棄著作權保護，讓所有軟體及衍生軟體歸入公眾領域，或許對公益的提升最為直接，但失去私利誘因的利弊難以評估，實際執行上也會出現漏洞。例如，在著作權存在的情況下，放棄權利者，將無法避免他人「搭便車」的行為，衍生軟體可能落入他人

著作權的保護範圍，而與原作者增進公益的初衷相違，甚至，改良後回過頭來與原軟體競爭，讓原創作的效用往私利的範圍集中，如下圖驚嘆號處所示：



圖六 軟體缺乏著作權保護的缺點

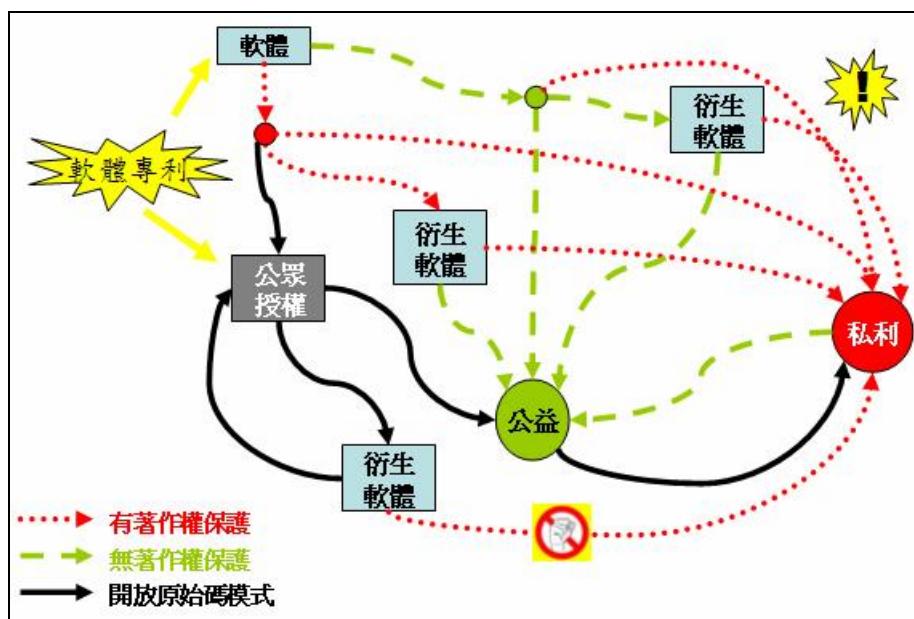
開放原始碼模式採取類似 GPL 的公眾授權，目的在於解決上面的問題，既不挑戰著作權法存在的必要性，也不要求犧牲小我貢獻於公眾領域的高度理想色彩，而是在現有的法律體系中，開創第三種可能，如下圖：



圖七 公眾授權模式與促進公益、私利的關係

在開放模式下，原始碼的公開與無償取得，直接有助於公益的累積，間接對創作者帶來私利，如名聲的累積或雙軌授權的收入。另外，公眾授權的約束則防止了搭便車的行為，衍生軟體無法直接落入私人手中再對外獲利。開放原始碼運動到目前為止，除了SCO案夾雜著商務契約上的糾紛外，就開放模式本身而言，可以視為獲得相當程度的成功，這一點從參與人數的增加、開發專案的持續進行與專屬軟體公司無法忽視開放模式的壓力等，可以得到佐證。目前雖無統計上的數字，但從網路少數的個案報導來看，應可推論開放原始碼模式下出現糾紛的比例不高，因此，一般認為開放原始碼運作的狀況大致良好。

然而，開放原始碼模式在著作權下運作無礙，並不代表毫無危機，在非技術障礙中，一般認為，未來最大的風險來源在於軟體專利。軟體專利如果核發過於浮濫，將對開放原始碼軟體構成威脅，圖示如下：



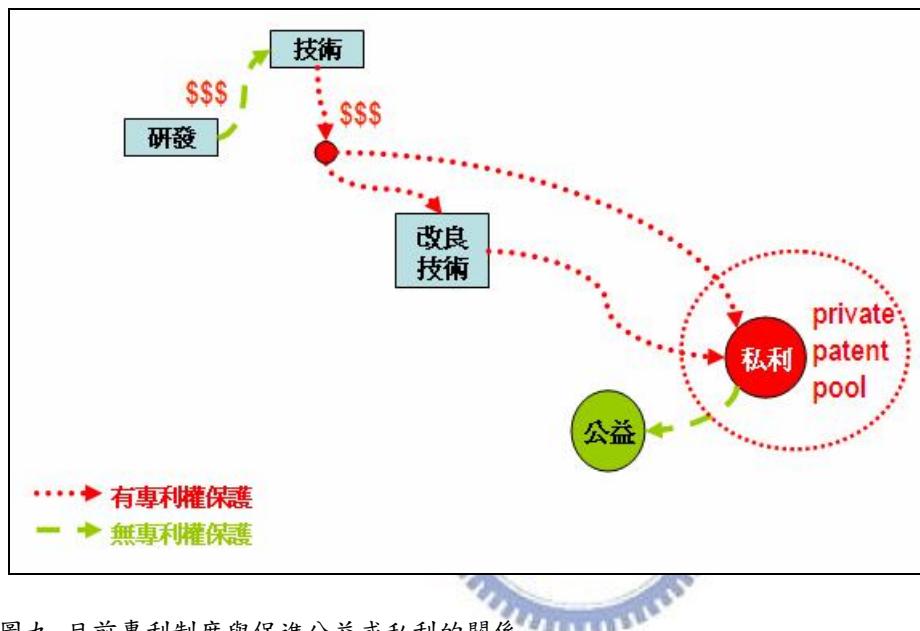
圖八 軟體專利的潛在威脅

開放原始碼社群對軟體專利的威脅並非視若無睹，但除了前述建議的各種對策外，似乎仍以推翻或阻止軟體的可專利性為主流意見，例如，歐盟反軟體專利人士的立場即在於避免走上美國的路線。儘管如此，在國際智慧財產權制度調和的趨勢及利益團體的運作下，或許歐盟接受美國軟體及商業方法的專利制度，只是遲早的事。在歐盟抗爭及美國部分人士呼籲專利制度改革時，往往討論的是專利制度的有無，以及其優劣比較，

這樣的思考模式似乎與早期討論軟體是否需要著作權保護類似，在「有」或「無」兩者之間爭辯外，本文認為或許存在著類似開放原始碼模式的第三種可能，亦即，承認軟體的可專利性，但採取一種可直接增進公益、間接增進私利的路徑。

第三項 公益專利庫的可能性

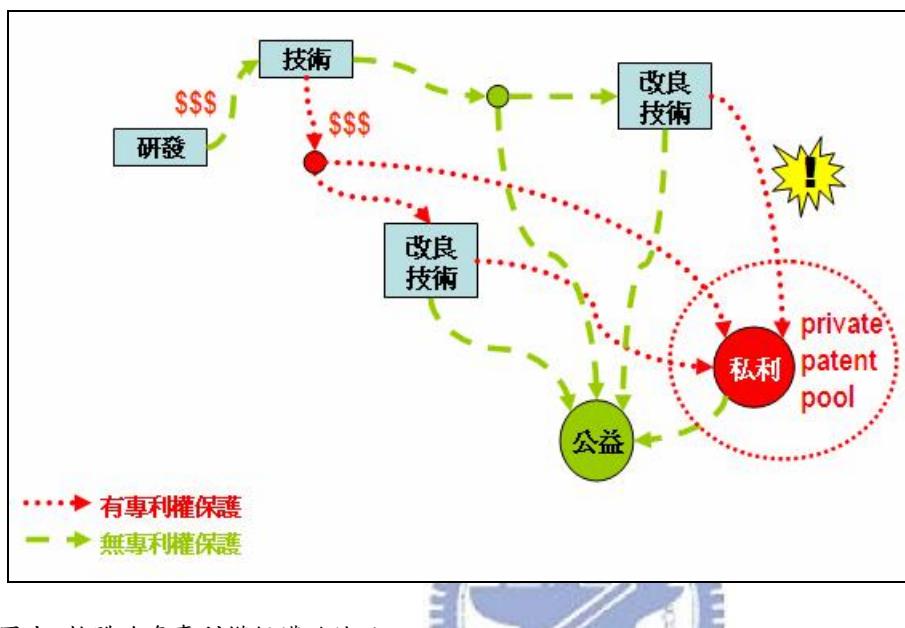
前項以圖示討論軟體與著作權保護、開放原始碼模式的關係，在此，將同樣的思考模式放到專利領域，目前專利運作機制可以用下圖表示：



在上圖中，與著作權不同者，首先是專利相關費用的出現，從研發、申請到維護，可能所費不貲，這一點對無償的程式設計師，明顯構成進入障礙。第二點不同是，多個專利可以集合成專利庫，形成授權談判的籌碼或達到防禦的效果，在不涉及專利權濫用及反托拉斯法的情況下，專利的累積有大者恆大的趨向，這一點對獨立創作的個人或小公司，相對也較為不利。為了與後續建議的專利類型區隔，在此傳統的專利稱為私人專利(private patent)，聚集之後稱為私人專利庫(private patent pool)。

與著作權情況相同的是，如果放棄智慧財產權的保護，雖然可能直接讓創作成果進入公眾領域，但創作人本身則受益不大。放棄專利權保護技術所面對的風險是，對方可能研發出相同的技術，並取得專利保護，而反過來索取權利金。在自己獨立開發該技術的情況下，雖有先使用的抗辯空間，但舉證責任在己且權利受限，仍構成一定的困擾。

因此，在專利實務上，通常發明人考慮將可申請專利但放棄保護的技術公布，讓該技術喪失新穎性，防止他人申請專利形成防禦效果。採取這種作法，固然有節省專利費用及避免專利糾紛的好處，但他人仍可能改良技術後取得專利，對原技術的創作者構成威脅。納入這項因素，申請專利的必要性反過來又提高一些。軟體技術與專利保護的關係，可以採下圖表示：



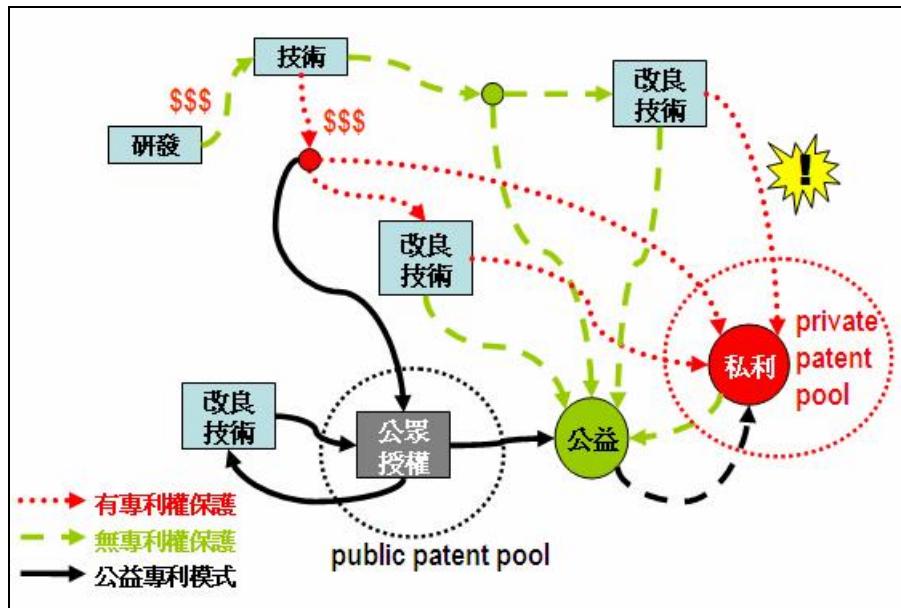
圖十 軟體放棄專利權保護的缺點

針對開放原始碼模式，專利制度上值得考慮的是增加第三種可能，將專利區分為「公益專利」²⁷⁸與「私人專利」。就專利的要件審查上，因為技術領域相同，無差別待遇的必要，但就相關程序，如考量兩者的目的不同，則存在相對優惠的可能，這部分稍後討論。

公益專利以直接增進公益為目的，如同原始碼直接開放供大眾無償使用，由於專利核准後即公開，故公益專利的重點並非公開，而在於無償供大眾使用，無償使用衍生搭便車問題，所以類似 GPL 的授權模式應運而生，在此以「公益專利授權」(PPL, Public Patent License)稱之。公益授權的精神是，受公益授權保護的專利技術，必須無償供他人使用，

²⁷⁸ 不論是專利核准或者早期公開，專利說明書必須公諸於眾，因此，專利本身即具有「開放」的特性，故「開放專利」雖然在字義上較容易與「開放原始碼」產生連結，但實質上並未突顯出本文建議的精神所在，故改以「公益專利」稱之，直接把以增進公益的目的性彰顯出來。

他人如果加以改良，再取得專利時必須採取相同的公益授權，以確保公益專利不會直接遁入私人專利領域，彼此關係如下圖所示：



圖十一 公益專利模式與現行專利制度的關係

第四項 公益專利庫如何運作

在一個運作良好的制度中，手段的執行應有助於目的的達成。過去專利制度面對新興技術領域，逐步調整其規範的手段及認定的標準，如軟體的可專利性及適格要件，引發諸多爭議，針對爭議所提出的修正方案，多半圍繞著「手段」的細節或範圍加以討論，例如，專利期限是否縮短、專利說明書揭露幅度增加等，似乎較少從「目的」的達成直接思考改革的可能。公益專利的提出，可視為從這個觀點切入，在屬性上公益專利直接與專利法的目的相契合，達成目的之後，才回過頭來探究執行過程應採取哪些手段加以配合。

公益專利如果成為一種法定的專利類型，與現行專利制度在程序上有何異同？亦即，在本文建議的改革方案中，公益專利與私人專利有何異同？關於這個問題，可以放在專利從有到無的不同階段下分析，並藉此說明建議方案在實務運作上的考量。首先，技術在研發過程中應盡的保密義務，並未因公益專利或私人專利而有不同，專利的核駁仍需要考量技術的新穎性，不論為公益或私利，都沒有必要為缺乏新穎性的技術賦予專

利保護。開放原始碼模式從提出想法到完成軟體，所有開發過程多半直接公布在網路上，這一點將無法符合新穎性的要求，開放原始碼模式在這一點可能必須有所調整，例如，在未提出專利申請之前，採行適當的配套保密措施。

提出申請時，在申請費用上本文建議採取差別待遇，公益專利既然以增進公益為目的，在費用的減免上應有考量的空間，甚至全部減免，因為，任何費用都可能構成無償程式設計師考量是否申請公益專利的門檻。甚至，如果為了促進公眾領域技術的增加，也可考慮納入獎勵措施，例如，取得專利後發給專利獎金等，如此一來，則可能填補發明人的在專利局規費以外的支出，如聘請專利代理人等。對專利局來說，申請費用如果減免，後續專利的維護費用應可比照辦理。發明人取得專利權，並非代表費用支出的結束，專利期間可能涉及侵害糾紛。在專利訴訟中，如果侵害他人專利，而本身缺乏任何專利做為交互授權的基礎，則損害賠償可能在所難免，如果該軟體本身具備其他公益專利，則公益專利庫將做為交互授權的基礎。如果在專利訴訟中，發生的是公益專利受他人侵害，則權利的主張將成為發明人的負擔，與 GPL 侵權由自由軟體基金會協助類似，在此可考慮由其他專責機構負責，例如，成立公益專利基金會等。簡言之，專利相關費用的負擔是目前開放原始碼模式融入專利制度的一大障礙，藉由公益專利的創設及排除其他障礙等配套措施，將有機會讓開放原始碼模式強化專利制度的價值，專利制度也不至於抑制開放原始碼模式的發展，兩者得以相容並存，如同 GPL 與著作權互相彰顯對方存在的必要。

概念上，為了不同的目的，可以考慮採取不同的手段。原始碼的開放並不必然表示單純採取公眾授權，而無法與其他方式併存，前面提到的雙軌授權模式，便是一個例子。透過雙軌授權，原始碼在公眾授權之外，可以平行對其他商業應用採取專屬授權，並由專屬授權中獲利，這是開放模式下直接增進公益、間接促進私利的一種樣態。專利制度涉及各種費用，申請人能否獲利支付費用，將與制度設計能否運作有關，因此，著作權雙軌授權的獲利模式可以納入考量，以探討調整專利制度的其他可能。例如，在雙軌授權下，專利技術基本上可視為屬於私人專利，而非上述公益專利，但透過公眾契約的方式，賦予公益專利的效果，如同開放原始碼基本上仍受私人著作權保護一般，但同時以

GPL 公眾授權賦予一般大眾使用權利。透過公眾契約釋出私人專利，用意在於擴大公眾專利庫的範圍，讓開放原始碼社群得以無償使用這些專利技術，或衍生出其他的公眾專利。至於是否直接採用公益專利，將專利的使用、製造等權利賦予大眾，或是採用私人專利，再平行以公眾授權及專屬授權兩種途徑釋出技術，則由發明人自行考量。關於實務操作的細節，有待將來進一步研究，目前建議創設新的專利型態，影響層面已經不小，實行難度亦高，程序性的細節或許在正當性與必要性確定後，再探討也不遲。

第五項 公益專利庫的利弊

創設公益專利或者讓私人專利透過公眾授權釋出，主要的目的在於形成公益專利庫，讓開放原始碼社群得以與私人專利庫在平等的制度下競爭，防止軟體技術不斷落入私人領域，避免開放原始碼在專利制度不利下只能藉由著作權加以保護。公益專利的上位概念來自開放原始碼模式，直接增進公益、間接促進私利，但公益專利的目的在相當程度上，則是保護開放原始碼模式的運作，避免專利制度失衡壓縮開放原始碼運動的生存空間。

如果軟體與專利本質上不應該合在一起，公益專利有可能成為專利制度的一種退出機制。目前軟體專利在成為一種法定的保護型態後，基本難以再將軟體排除在專利適格標的之外，在商業競爭下為求武器平衡，軟體專利的數量似乎只有如走向不歸路般地持續成長。專利制度涵蓋技術領域一直擴大而無法縮減，即使立法介入調整，如何調和既有專利權人與未來發明人之間的利益，也是一大難題。在這種情況下，公益專利型態的出現，或許可以成為有意退出專利武器競賽者的一種選擇，既保有自己專利技術的使用，又未放棄與競爭對手交互授權的制衡效果。如果重量級廠商對公益專利庫挹注大量專利，在整體規模上超出私人專利庫甚多，在實力懸殊的情況下，也許引發私人專利的全面棄守，讓實質上只剩下公益專利，到了這個地步，則軟體專利可視為名存實亡。武器競賽到了所有人繳械、互不攻擊的情況，則武器亦不復存在。當然，採取這種思考方式，前提是如部分人士呼籲，軟體專利只是加速既得利益的壟斷，而有害於資訊技術的進步，對此，公益專利庫或許可能成為軟體專利制度不斷擴張的一種降溫機制或退路。

創設公益專利的缺點是，對現行制度衝擊不小，即便配套措施設計得當，推行上也可能困難重重，或者，在政府介入解決現有問題後引發其他的問題。之所以需要對制度造成衝擊，主要原因在於專利的強制性及費用負擔。GPL 公眾授權透過契約的方式，不需要修正著作權法即可形成開放原始碼運動，主因之一在於原始碼本身對原作者即具備實用價值，通常解決了自己遇到的問題，順手將原始碼開放出來的成本很低，而潛在可能獲得他人的建議或者加入開發的行列，因此，基本上對許多程式設計師而言利多於弊。相對於專利，如果希望單以契約方式鼓吹開放原始碼社群達到類似的效果，等於要求無償發明人負擔額外的費用，可能變得陳義過高，在實務難以推行。因此，有必要從制度上思考其他的可能性。智慧財產權制度在跨國合作上難以自成一格，以 TRIPs 調和國際間智慧財產權法制的角度，新的公益專利所面臨的問題將更為複雜。另外，公益專利庫如果形成，與公平交易法的規範之間，仍有諸多課題值得討論，例如，公益專利是否過度限制市場競爭，或者反而解決軟體專利過度限制市場競爭的問題？反托拉斯法下分析授權契約的合理原則，在公益專利類型下將如何適用或修正？公益專利是否可能過度壓制了目前以私利為誘因的機制？這些問題因涉及另外一個複雜的領域，有待將來深入探討。如果需要從量化的方式分析利弊，未來或許也可考慮採取經濟分析的角度切入相關的問題。

最後，公益專利的概念與既有的諸多建議並不衝突，在推動上可以兼容並存，例如，如果縮短軟體專利的保護期限，公益專利與私人專利一視同仁跟著縮短期限，如果前案資料庫更加完備提升了審查品質，公益專利與私人專利的品質也將同時受益。因此，公益專利提出與各方建議平行的一種新的可能性，在制度上增加保護的類型，但沿襲目前專利制度的許多實務操作，並同樣寄望實務操作面的其他改善能讓整體專利制度更趨於完善。如果上述各方的建議未能落實，在專利制度整體品質日趨下滑的情況下，或許更彰顯出思考公益專利的必要性與迫切性。

第五章 展望未來

第一節 開放原始碼模式的前景

開放原始碼模式未來的發展，可以從底下幾個面向觀察。

第一項 軟體應用面的擴大

開放原始碼擴大應用範圍，可以從縱向深度及橫向廣度說明，在縱向上，許多開放原始碼軟體持續強化功能，並逐漸跟上硬體升級的腳步。例如，資料庫系統從處理單一資料庫進化到處理數個資料庫形成的叢集(cluster)，對大型資料中心來說，資料庫叢集的平行處理及容錯能力，是能否省下專屬軟體昂貴的授權費用、轉向開放原始碼的重要考量。又如，在硬體支援上，許多廠商只考慮 Windows 及 Intel 平台，Centrino 筆記型電腦推出時，只支援 Windows 系統，而無法讓 Linux 系統執行，便是一個例子。隨著 Linux 佔有率的提升，硬體廠商開始提供不同的驅動程式，付之闕如時，廠商除了收到開放原始碼社群的意見外，社群中也可能自行開發以支援新硬體。因此，可以預期開放原始碼軟體與專屬軟體所處的外在條件，彼此的差距正在逐漸縮小。當年 Stallman 拿不到印表機的驅動程式原始碼而提倡自由軟體，就這一點，其願望可算已經實現。

在橫向上，開放原始碼軟體的種類及使用人口，都有增加的趨勢。開放原始碼軟體過去通過許多伺服器應用的考驗，市場影響力不斷提升，可謂證明了開放原始碼模式的效益，未來備受注目的，則是一般用戶端的接受度。例如，對 Microsoft 而言，Linux 在作業系統方面挑戰 Windows，OpenOffice 在應用軟體方面挑戰 Microsoft Office，兩者可能直接對 Microsoft 的主要收入來源造成影響，加上 Windows 下一代系統 Longhorn 延後推出，讓開放原始碼社群更有機會迎頭趕上，剩下的問題是一般使用者是否願意轉移或嘗試 Linux 系統，以及 Linux 系統在桌面環境下是否具體反映使用者的需求等。就使用人口的普及，開放原始碼除了讓經濟上的弱勢族群降低取得資訊工具的成本外，未來可能進一步擴大使用的族群。例如，隨著人口老化，可考慮在 Linux 上建立一套老年人專用的電腦系統，透過網路，老年人之間可形成互助社群，簡單的遊戲軟體可以延緩腦力

退化的速度等。至於系統開發的人力來源，等這一代的程式設計師再過二十年，自己即將成為使用者時，可能就不是太大的問題。類似的小眾應用唯有在開放原始碼存在時才比較可能出現，專屬軟體公司無可厚非必須追求最大的獲利，無法顧及所有族群，對比之下，開放原始碼則追求不同的目的，同時附帶為其他人鋪陳了許多開放的可能性。

另外，開放原始碼的應用逐漸跨出軟體的範疇，以兩種方式進入硬體領域。第一種是將軟體燒入硬體中，執行特定的功能。最明顯的例子是嵌入式 Linux 系統(embedded Linux)，可結合各種不同的硬體裝置擴大應用層面，例如，手錶、PDA、手機、機器人、數位家電等，都可以看到 Linux 的存在。Pingtel 公司決定將網路電話相關軟體以開放原始碼模式釋出，方便硬體廠商加以結合、節省成本，則是另外一個例子²⁷⁹。至於軟體燒入硬體可能成為一種侵權的樣態，則如前面 GPL 軟體燒入網路路由器裝置時所述²⁸⁰。開放原始碼對硬體的第二種影響，則是可能改變硬體開發的模式，例如，IBM 計畫將微處理器的設計對外開放，藉由開放原始碼模式的優點，提高市場佔有率或加快開發速度。微處理器是硬體系統的核心，IBM 公布晶片的細節資訊及配套的設計工具，有助於系統廠商及軟體人員開發新的應用，或提早符合客製化的不同需求。雖然 IBM 對於部分設計核心仍保有控制權（如微處理器的指令集），但基本上開放之後，客戶可以擺脫以往專屬架構的限制，彼此之間也能在許多方面合作無間。IBM 強調合作、開放的模式，能否在市場佔有率上挑戰其他不斷追求執行速度的專屬晶片，如 Intel，則有待觀察²⁸¹。

第二項 Creative Commons

如果將「開放模式」視為開放原始碼模式的上位概念，跨出資訊界軟硬體以外，可以發現開放模式逐漸落實到更為廣泛的創作領域中。例如，成立於 2001 年的 Creative

²⁷⁹ See CXOtoday, *VoIP Goes Open Source: Talk To Become Cheap*, Mar. 26, 2004, at http://www.cxotoday.com/cxo/jsp/index.jsp?section=News&subsection=Software&subsection_code=4&file=template1.jsp&storyid=773 (visited May 21, 2004).

²⁸⁰ 參見第二章第四節第一項。

²⁸¹ See John G. Spooner, *IBM opens up its Power line*, Mar. 31, 2004, at http://news.com.com/2100-1006_3-5182742.html (visited May 21, 2004).

Commons 組織，參考 GPL 的精神，希望透過開放而具有彈性的授權方式，在日益強調個人著作權的趨勢中，致力保留屬於公眾自由創作的空間，以便在私人與公眾之間取得平衡。根據 Creative Commons 的公眾授權，在預設的情況下，大眾有機會自由展示、複製、傳播、修改他人的著作，而無需另外徵求創作者的同意，採用公眾授權時，創作者可以根據自己的意願，事先將部分權利賦予公眾，不需要像過去等待他人徵詢時，才賦予著作權法專屬的各種權利。不論事先賦予權利或事後徵詢授權，雖然創作者與使用者最後都可能達成相同的合意，但對著作使用的效率與衍生著作的累積卻有不同的效果。另外，著作權法為了因應數位化之後複製容易、保護困難的處境，日益朝「原則禁止、例外許可」的方向位移，並輔以立法方式禁止迴避科技保護措施²⁸²，在這種可能從 Copyrights 變成 Copywrongs²⁸³的情況下，Creative Commons 等於反向提供另外一種「原則許可、例外禁止」的可能性。

Creative Commons 公眾授權涵蓋的標的包括文字、影像、音樂等各種承載著作權的媒介，相對創作者可能是作家、導演、演奏家等。授權條款中定義著作權人及被授權人的權利與義務，並尊重創作者選擇不同授權條款的需求，例如，著作權人是否允許他人修改著作？是否允許著作做為商業用途？是否要求他人使用時必須標註原創作者？是否要求他人的衍生著作遵守上述相同的規定？透過不同的符號，創作者可以簡便地在自己的著作上標註授權的範圍或條件，以交叉組合方式設定自己對公眾授權的需求²⁸⁴。2003 年，Creative Commons 模式朝國際化跨出第一步，提出 iCommons 計畫(international Commons)，目前已有十多個國家加入，計畫的目的在於將公眾授權條款轉換成符合當地語文及法律架構的版本，做為 Creative Commons 在各國落地生根的基礎。各地累積的資源透過國際交流，可進一步形成全球性的公眾資源庫。在台灣，中央研究院資訊所經

²⁸² 深入分析請參閱馮震宇教授「數位內容之保護與科技保護措施」一文(月旦法學，93 年二月)。

²⁸³ See generally SIVA VAIDHYANATHAN, COPYRIGHTS AND COPYWRONGS: THE RISE OF INTELLECTUAL PROPERTY AND HOW IT THREATENS CREATIVITY (2003).

²⁸⁴ 細節請參見 <http://creativecommons.org/learn>。

過籌備階段，在 iCommons Taiwan 計畫²⁸⁵下，2004 年五月正式在網路上公布第一版的 Creative Commons 公眾授權條款，希望在規定一個月內廣納不同的意見後，讓最終的版本更趨完備，並做為台灣未來累積公共資源的法律基礎。同時，iCommons Taiwan 計畫提供許多參考資料，方便後續推廣時由各方採用，國內目前已出現研究成果以 Creative Commons 授權方式釋出²⁸⁶，期待後續如風行草偃般推廣開來。美國最高法院網站上提供多媒體資料²⁸⁷，如大法官審理案件與正反雙方辯證的全程錄音，便透過 Creative Commons 方式對外授權，發揮推廣法學教育的正面示範作用。Lessig 於 2004 年出版 Free Culture 一書，在傳統出版通路之外以 Creative Commons 免費釋出電子檔²⁸⁸，是另外一個例子。2004 年五月英國 BBC 表示準備將旗下的許多錄音及錄影資料，在參考 Creative Commons 後訂出自己類似的授權供大眾使用，以便 BBC 在公眾文化創作領域取得先行者優勢，則是另外一個受到注目的例子²⁸⁹。

除了 Creative Commons，將開放模式應用在原始碼之外還有其他例子。例如，自由軟體基金會的文件原本連同原始碼以 GPL 方式釋出，但顧及文件與程式的性質有異，基金會於是另外制訂「自由文件授權書」(FDL, Free Document License)，做為公眾授權的基礎。一般開發軟體的過程中，為了軟體工程上的要求，或是輔助一般使用者在操作上所需，相關文件的製作十分常見，透過自由文件的授權方式，技術文件可以與 GPL 軟體並行，同時讓 copyleft 的精神一以貫之。在自由文件的授權下，使用者可自行複製、散布、修改文件，並允許進行商業或非商業上使用。在 Creative Commons 順利推廣之後，FDL 是否仍有必要存在，值得觀察。另外，1998 年 OpenContent 組織曾推動類似的理念並獲得初步成果，在 Creative Commons 出現後，由於其授權彈性及操作細節更為周

²⁸⁵ <http://www.openfoundry.org/icommons/>

²⁸⁶ 參見 tw-iCommons 網站日誌，消息發佈之網址為：

<http://www.openfoundry.org/icommons/archives/000030.html> (visited May 21, 2004).

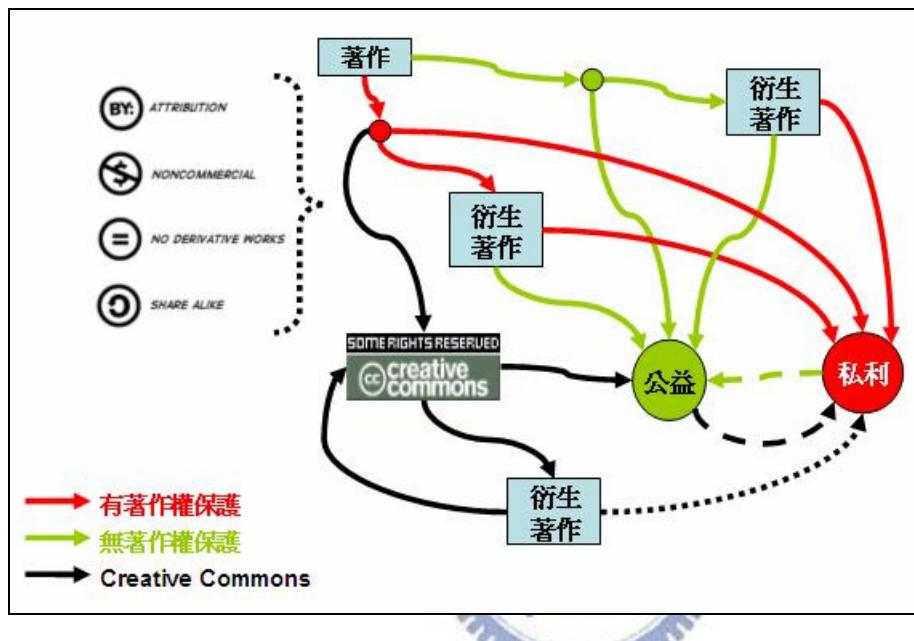
²⁸⁷ 參見 <http://www.oyez.org/oyez/frontpage> (visited May 22, 2004).

²⁸⁸ 參見 <http://www.free-culture.cc> (visited May 22, 2004).

²⁸⁹ See Simon Perry, *BBC Creative Archive licensing to be based on Creative Commons*, May 26, 2004, at http://digital-lifestyles.info/display_page.asp?section=distribution&id=1254 (visited May 27, 2004).

延，OpenContent 創始人決定終止計畫，加入 Creative Commons 組織²⁹⁰。以此演變，未來電腦軟體以外的著作，或許可統一由 Creative Commons 涵蓋，以累積成豐富多樣的公眾資源。

在上一章，本文曾以圖示說明著作權與公眾授權模式之間的關係，以及公眾授權模式為何、如何類推適用到軟體專利，同樣的思維可以沿用到 Creative Commons 領域，採下圖表示：



圖十二 Creative Commons 與促進公益、私利的關係

與開放原始碼的 GPL 相比，Creative Commons 除了涵蓋標的不同外，可以看出授權的樣態更為細緻，尊重著作權的立場從傳統預設「保留所有權利」翻轉成「保留部分權利」，未保留的部分等於公眾已取得授權。在這種情況下，使用者可以降低侵害他人智慧財產權的顧慮，並擺脫著作權「合理使用」的不確定性，有助於提高後續創作的自由度，同時，在 Creative Commons 下也有機會建立累積公眾資源的回饋機制。

第二節 公眾智慧財產權

²⁹⁰ See OpenContent, *OpenContent is officially closed*, Jun. 30, 2003, at <http://www.opencontent.org/index.shtml> (visited May 22, 2004).

在開放原始碼模式下，從某個角度，可以視為「所有權」與「使用效益」同時併存，但主體不同，開放原始碼軟體歸程式設計師所有，但透過公眾授權供大眾使用，雖然程式設計師自己通常也使用該軟體，但就實際效用來說，如果軟體成功，大眾使用累積的效益可能遠大於程式設計師個人的使用²⁹¹。因此，如果從「法益」的觀點，程式設計師對開放原始碼軟體的所有權及大眾的使用效益，可能是兩個值得保護的客體。之所以造成這種現象，主因在於智慧財產數位化的結果，數位化帶來無耗損的複製，網路的普及讓流通的成本趨近於零，兩者共同為大眾使用的效益產生加乘效果，例如 Linux 系統應用廣泛便是一個例子。另外，智慧財產不具實體的特性，讓「分享」的行為成為利人但不損己的舉動，程式設計師為個人目的開發的軟體，並不會因為分享出去而受損，相反地，可能收到善意的批評及建議，如果不以財務回饋為創作目的，分享出去可能有利而無害。

說明程式設計師「所有權」與大眾「使用效益」同時併存的原因及背景後，可以將開放原始碼的公眾授權視為聯繫兩者的橋樑，讓原屬於個人的使用效益以單行道方式複製到類似公眾領域之中，讓一般大眾皆能善加使用，但使用時不得違反公眾授權條款反向進行搭便車的行為。所謂「類似」公眾領域表示並不完全等於公眾領域，在公眾領域中的資源，並沒有人主張其所有權，一般人可以任意使用公眾領域的資源，也沒有搭便車的問題，以不盡貼切的方式比喻：因為車子已過了智慧財產權法律保護的期限，車主等於不存在了，而開放原始碼軟體等於尚有車主的車子，只不過車門上貼了 GPL 授權條款，他人可以免費使用但使用時必須遵守相關規定。這些有主的愛心車集合起來，並沒有適當的法律名詞代稱，概念上或許比較接近 Creative Commons。

缺乏法律名詞隱含的可能是法律關係的模糊或者法律客體的缺漏，這一點值得再思考一步。前述所有權與使用效益的併存，或許可以視為兩種法益的分離，分離並非毫無關係，而是彼此都可能成為法律保護的客體。所有權的保護沒有問題，目前法律制度即保障私人的所有權，但保護大眾使用的效益是否可以提升為保護大眾的使用權？本文持

²⁹¹ 事實上，創作者本身也可視為大眾的一員。

贊成意見外，認為如果涵蓋前述公益專利的創設，或許可以稱之為「公眾智慧財產權」(public intellectual property right)，目前的智慧財產權則稱為「私人智慧財產權」，彰顯權利主體的不同。在這種概念下，開放原始碼軟體及 Creative Commons 下的創作可以視為「公眾智慧財產權」中受著作權保護的財產，公益專利庫則等於「公眾智慧財產權」領域中屬於專利的部分，營業秘密因要求保密的特性與「公眾智慧財產權」的公開性質相斥，故無適用餘地。

學者對圈地運動省思後提出國家公園的概念，固然有助於指出「保存」公眾領域的必要性，並佐以環境權強調背後的理念，但實體世界的資源只會減少、智慧創作成果卻能不斷增加，兩者本質上的差異似乎隱含著修正觀點的可能，本文認為以「公眾智慧財產權」領域取代國家公園的概念，或許是一種可行的想法。在「公眾智慧財產權」下，使用權歸大眾所有，所有權歸創作者所有，一般大眾使用智慧財產時則負擔遵守公眾授權的義務。「公眾智慧財產權」等於介於目前私人財產權與公眾領域的第三種可能，在模式上與前述開放原始碼模式、公益專利庫、Creative Commons 的圖示相吻合，在傳統財產權概念與公眾領域彼此相對的二分法中，或許非黑即白之間可以存在灰色的地帶。有鑑於目前智慧財產權制度，舉著避免「公共財災難」(tragedy of commons)的大旗不斷擴張而逐漸變質，導致創作上避免侵害他人智慧財產權的注意義務提高不少，增加了自由創作的限制或負擔，對此，或許賦予「公眾智慧財產權」的法律地位可以為「公共財的繁榮」(prosperity of commons)打下一個基礎。

過去著作權法為了因應市場失靈產生合理使用原則，專利密度不斷提高後，則有學者提出相同的建議，希望避免市場失靈妨礙大眾福祉的增進而偏離專利法當初的目的。在目前智慧財產權的概念下，行為如涉及權利的侵害，可以視為原則禁止、合理使用為例外，在「公眾智慧財產權」的概念下，合理使用則轉變為原則，創作者未賦予的部分才是例外。如果類似的概念得以推動，未來，一般大眾可能不需要在「私人智慧財產權」的叢林中努力尋找「合理使用」(fair use)的空隙，在「公眾智慧財產權」領域預設賦予「公眾使用」(public use)的權利下，唯一需要注意的只是使用上附帶的規範。

第三節 台灣產業的機會

在全球資訊產業的版圖中，台灣在硬體方面表現突出，以兩兆雙星中的半導體為例，從上游 IC 設計、晶圓製造、封裝測試、主機板到各種消費性電子產品，處處可以看到高度的市場佔有率。隨著國際化佈局，硬體產業逐漸將勞力密集的製造段轉移到成本更低的區域，朝設計代工或建立自有品牌兩種方向升級。反觀軟體產業，一則受到內需市場太小的限制，一則受到印度及中國大陸軟體代工的威脅，所考量的並非升級的問題，而是生存的問題。在先天不良後天失調的情況下，開放原始碼能帶來什麼樣的契機？

首先，必須先瞭解的是，開放原始碼模式對一般使用者有利，但可能對傳統專屬軟體公司不利。開放原始碼陣營與一般軟體公司競爭的並非個別的產品，競爭的其實是開放模式與專屬模式，雖然各自可能在不同的應用領域中互有優勢，或者相容，但開放模式瓜分專屬模式的市場，已是不爭的事實。國內大部分軟體公司採用傳統專屬模式，因此，開放原始碼的崛起對許多公司來說，可能是個壞消息。不過，正因為國內軟體產業的研發較弱，特別凸顯出善用開放原始碼軟體的必要，部分軟體的應用層面過去可能因此處於低度開發的狀態而具備成長的潛力。如果忽視開放原始碼，一味埋首閉門造車、重新發明輪子，則趕上先進資訊化國家可能更是遙遙無期。

從這個角度切入，善用國內硬體產業的優勢，或許可以考慮透過開放原始碼軟體降低成本，進一步強化競爭力或擴大市場。例如，開放原始碼軟體常見的問題之一在於硬體的相容性較低，或者缺乏支援，即使最新版本的 Linux 系統，也可能無法支援各種新型的電腦週邊裝置，受此影響，使用者往往降低採用的意願。台灣做為硬體設計製造的主要基地，如能主動增加對 Linux 等開放原始碼系統的支援，提供最新的驅動程式，除有助於開放原始碼的推展外，間接也能吸引不同的消費族群，增加市場佔有率。目前國內已有廠商嘗試朝這方面努力，例如，推出新款筆記型電腦搭配 Linux 系統供消費者選擇²⁹²。

²⁹² See David Tzeng & Steve Shen, *Taiwan's makers test markets with Linux-based notebooks*, Mar. 22, 2004, at <http://www.digitimes.com/NewsShow/Article.asp?datePublish=2004/03/22&pages=A1&seq=1> (visited May 22,

其次，在硬體支援軟體之外，開放原始碼軟體也可成為硬體的核心，以增加硬體應用的範圍與彈性，這一點在嵌入式系統方面最為明顯。台灣在消費性電子產品表現不錯，隨著無線網路及家電應用的整合，客廳已成為相關廠商的下一波戰場，如果台灣能保持製造上的品質與速度，善用低成本的開放原始碼軟體強化系統功能，假以時日建立自有品牌，或許將來可以捨棄全球平均工時最長的紀錄，在產業升級後一樣繁榮富足，並於全球經濟版圖中佔有一席之地。

至於如何在國內推動開放原始碼模式，除了經濟部工業局推動「自由軟體鑄造場計畫」²⁹³，有助於吸引人才投入開放原始碼軟體的開發外，在公共政策方面，應可考慮是否特別提倡開放原始碼軟體，以政府單位發揮帶頭作用。例如，政府單位採購資訊系統時，是否應優先考慮使用開放原始碼軟體？或者在部分情況下強制規定？如果強制規定可能阻礙自由市場的競爭機制，至少在政策上，應考慮堅持採用各種業界的開放標準，讓格式透明、資料得以流通、系統能夠互相整合，以避免獨家壟斷、強迫升級的情況出現。至於開放原始碼軟體的中文化，或是其他與中文相關的資訊技術，則是另外一塊國外缺乏誘因、國內有待努力的領域。開放原始碼帶來許多新的機會，國內投入相關領域者逐漸增加，惟大部分從技術及商業觀點出發，對於相關法律議題除了著墨較少外，往往也忽略許多潛在的風險來源，或對各種糾紛樣態產生誤解而裹足不前，本文希望在論文研究之餘，對這一點落差能夠略盡棉薄之力，增加可供產業界參考的素材。

2004).

²⁹³ 參見官方網站 <http://www.openfoundry.org> (visited May 22, 2004).

第六章 結論

關於開放原始碼模式中的法律議題，本文研究時正巧遇到 SCO v. IBM 一案如火如荼地進行，SCO 後續並衍生出幾個關係複雜的案件，這些實務案例成為探討開放模式下法律糾紛的切入點。在各方紛陳的意見及盤根錯節的事實中，進一步引發個人瞭解開放原始碼模式下其他潛在風險的興趣。針對已知的風險類型以及近年來智慧財產權制度的演變，有識之士指出軟體專利可能成為開放原始碼模式下一階段最大的挑戰，而值得本文繼續一探究竟，在探究的過程中，本文則試著提出一些不同的觀點，最後，本文則從開放模式逐漸擴散的影響力中，觀察未來可能的演變。

對於 SCO 相關案件，由於複雜度太高，以論文有限的篇幅，在研究上力有未逮之處恐怕不在少數，本文僅能就重要的爭點及關係，發揮蒐集整理並試圖釐清的作用，而難以在此預測最後的勝負。不過，不論 SCO 案最後的結果為何，可以確定的是，訴訟案對開放原始碼社群帶來許多衝擊，衝擊之後引發諸多人士的回應與省思，在回應過程中真相往往越辯越明，透過這些省思帶來強化開放原始碼模式的契機，而有助於防範下一次法律糾紛於未然。未來 SCO 如果敗訴，其他人下次挑戰開放原始碼模式的難度可能提高，SCO 如果勝訴，則開放原始碼社群將有去蕪存菁的必要。因此，長期來看，訴訟案的出現未嘗不是一件好事，至於其他與開放原始碼軟體相關的糾紛，也具有同樣正面的意義。本文對 SCO 相關案件，初步進行了蒐集資料、釐清爭點及分析交錯關係的工作，希望對有興趣瞭解或追蹤後續發展的人有所幫助。

另外，為了預防糾紛的產生，有必要在糾紛出現之前瞭解可能的爭議來源，對背後的風險因素加以探討，其中，可以發現幾個重要的考量多半與開放原始碼獨特的公眾授權模式有關，而且隨著開放原始碼的普及，瞭解公眾授權模式也變得比以前重要。不論是一般使用者或是程式開發人員，在公眾授權條款下各有不同的權利與義務，如果沒有對開放模式加以瞭解，貿然採用事後可能發現意料之外的限制。對開放原始碼模式增加認知的程度，也有助於判斷諸多爭議下的是非對錯，例如，就已知的事實，SCO 的主張是否為有理由等。本文以一整章篇幅探討開放原始碼的糾紛，目的在於將使用開放原始

碼軟體或採用開放原始碼模式的風險降低，希望有利於實務上對開放原始碼運動的推動及未來處裡上的參考，如果將來出現新的糾紛類型或風險因素，有興趣者也能夠繼續補充或延伸對這方面的瞭解。

在公眾授權條款之外，侵害軟體專利可能成為開放原始碼軟體未來無法避免的糾紛類型，事實上，軟體專利引發的爭議並非針對開放原始碼軟體而來，其他一般專屬軟體也在涵蓋範圍之內，只是開放模式在軟體專利的負面影響中，可能更加無法招架，加上缺乏累積專利的適當機制，使得原始碼雖然得以不斷累積，卻未能累積軟體專利的數量。本文從各方對專利制度的批評與建議，以及受開放原始碼模式的啟發，嘗試提出創設公眾專利的想法，並簡要設計運作的機制、分析可能的利弊，希望藉此為開放原始碼軟體形成保護網，或者讓專利制度的改革多一種思考面向。任何涉及制度面改革的建議必定遭遇落實上的困難，但正因為挑戰的難度而凸顯出論文研究的必要，希望公眾專利的概念在此發揮拋磚引玉的效果，未來有興趣者可針對許多初步的想法再深入探究，從經濟分析的角度或其他觀點，提出更為完備的理論基礎或制度設計。

至於開放模式的未來，跳出軟體領域，除了看到 Creative Commons 逐漸形成一股新的潮流，令關心公眾創作領域者有所期待之外，本文也意識到前述公眾專利可能是「公眾智慧財產權」概念下的一種樣態，在公眾智慧財產權概念下的著作權領域，本質上則與 Creative Commons 互相契合。Creative Commons 參考開放原始碼模式而來，開放原始碼軟體為了在軟體專利制度下得以平等競爭，激發出公益專利的想法，公益專利朝上位概念發展，則引發思考公眾智慧財產權的可能性，並回到 Creative Commons 本身，有機會形成一以貫之的理論架構。惟顧及論文篇幅及避免脫離開放原始碼的主軸太遠，這部分只有留待將來研究。本文最後對台灣產業提出的一點建議，除了寄望產業善用國際上各種開放原始碼軟體，並強化既有以硬體為主的市場競爭力外，同時希望本文對相關法律議題的整理，能夠成為認識開放原始碼模式的一點參考。

參考文獻

一、中文著作

1. 楊崇森，專利法理論與應用，三民書局，2003 年 7 月。
2. 曾陳明汝，兩岸暨歐美專利法，翰蘆圖書出版，2004 年 2 月。
3. 王盈勛，微軟生存之戰：軟體巨人如何因應開放原始碼，商周出版，2003 年 9 月。
4. 馮震宇，高科技產業之法律策略與規劃，元照出版，2003 年 4 月。
5. 馮震宇，智慧財產權發展趨勢與重要問題研究，元照出版，2003 年 10 月。
6. 劉尚志、陳佳麟合著，電子商務與電腦軟體之專利保護－發展、分析、創新與策略，翰蘆圖書出版，二版，2001 年 9 月。
7. 陳起行，美國軟體專利論，1966-1991，學林文化出版，2002 年 2 月。
8. 陳家駿、羅怡德合著，公平交易法與智慧財產權－以專利追索為中心，五南圖書出版，1999 年 11 月。
9. Lawrence Lessig 著，劉靜怡譯，網路自由與法律，商周出版，2002 年 7 月。
10. Frederick P. Brooks, Jr.著，錢一一譯，人月神話：軟體專案管理之道，2004 年 4 月。
11. Kevin G. Rivette & David Kline 合著，林柳君譯，閣樓上的林布蘭，經典傳訊出版，2000 年 6 月。
12. Peter Wayner 著，蔡憶懷譯，開放原始碼：Linux 與自由軟體運動對抗軟體巨人的故事，商周出版，2000 年 8 月。



二、中文期刊論文

1. 朱俊銘，電腦軟體相關智慧財產權法制之探討－從開放原始碼運動出發，國立臺灣大學法律學研究所碩士論文，2002 年 6 月。
2. 劉靜怡，電子商務與智慧財產權：軟體原始程式碼開放運動與連網時代的共同創新未來，月旦法學雜誌，第 80 期，2001 年 11 月。

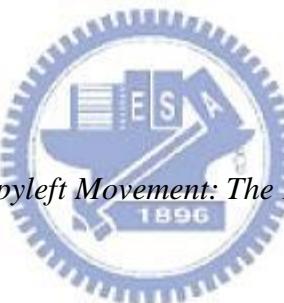
3. 吉玉成，商業方法軟體專利之標的適格性研究，科技法學評論，第一卷第一期，2004年4月。
4. 陳人傑，開放原始碼授權條款之法律分析－以通用公共授權為中心(上)，科技法律透析，2002年6月。
5. 陳人傑，開放原始碼授權條款之法律分析－以通用公共授權為中心(下)，科技法律透析，2002年7月。

三、英文著作

1. JOSHUA GAY, FREE SOFTWARE, FREE SOCIETY: SELECTED ESSAYS OF RICHARD M. STALLMAN (2002).
2. SAM WILLIAMS, FREE AS IN FREEDOM: RICHARD STALLMAN'S CRUSADE FOR FREE SOFTWARE (2002).
3. ERIC S. RAYMOND, THE CATHEDRAL & THE BAZZAR (2001).
4. CHRIS DiBONA, SAM OCKMAN & MARK STONE, OPEN SOURCES: VOICES FROM THE OPEN SOURCE REVOLUTION (1999).
5. DAVID R. KOEPSELL, THE ONTOLOGY OF CYBERSPACE: PHILOSOPHY, LAW AND THE FUTURE OF INTELLECTUAL PROPERTY (2003).
6. GREGORY A. STOBBS, SOFTWARE PATENTS (2000).
7. MARTIN FINK, THE BUSINESS AND ECONOMICS OF LINUX AND OPEN SOURCE (2003).
8. ROD DIXON, OPEN SOURCE SOFTWARE LAW (2004).
9. PAUL GOLDSTEIN, COPYRIGHT'S HIGHWAY: FROM GUTENBERG TO THE CELESTIAL JUKEBOX (2003).
10. JOSEPH FELLER & BRIAN FITZGERALD, UNDERSTANDING OPEN SOURCE SOFTWARE DEVELOPMENT (2002).
11. LAWRENCE LESSIG, THE FUTURE OF IDEAS: THE FATE OF THE COMMONS IN A CONNECTED

WORLD (2002).

12. SIVA VAIDHYANATHAN, **COPYRIGHTS AND COPYWRONGS: THE RISE OF INTELLECTUAL PROPERTY AND HOW IT THREATENS CREATIVITY** (2003).
13. STEVEN W. LUNDBERG & STEPHEN C. DURANT, **ELECTRONIC AND SOFTWARE PATENTS: LAW AND PRACTICE** (2000).
14. RUSSELL C. PAVLICEK, **EMBRACING INSANITY: OPEN SOURCE SOFTWARE DEVELOPMENT** (2000).
15. GLYN MOODY, **REBEL CODE: INSIDE LINUX AND THE OPEN SOURCE REVOLUTION** (2001).
16. LINUS TORVALDS & DAVID DIAMOND, **JUST FOR FUN: THE STORY OF AN ACCIDENTAL REVOLUTIONARY** (2001).
17. ROBERT W. HAHN, **GOVERNMENT POLICY TOWARD OPEN SOURCE SOFTWARE** (2002).



四、英文期刊論文

1. Teresa Hill, *Fragmenting the Copyleft Movement: The Public will not Prevail*, 1999 UTAH L. REV. 797 (1999).
2. Patrick K. Bobko, *Open-Source Software and the Demise of Copyright*, 27 RUTGERS COMPUTER & TECH. L.J. 51 (2001).
3. Natasha T. Horne, *Open Source Software Licensing: Using Copyright Law to Encourage Free Use*, 17 GA. ST. U. L. REV. 863 (2001).
4. David McGowan, *Legal Implications of Open-Source Software*, 2001 U. ILL. L. REV. 241 (2001).
5. Robert W. Gomulkiewicz, *De-Bugging Open Source Software Licensing*, 64 U. PITTS. L. REV. 75 (2002).
6. Dennis M. Kennedy, *A Primer on Open Source Licensing Legal Issues: Copyright, Copyleft and Copyfuture*, 20 ST. LOUIS U. PUB. L. REV. 345 (2001).

7. Robert W. Gomulkiewicz, *How Copyleft Uses License Rights to Succeed in the Open Source Software*, 36 Hous. L. Rev. 179 (1999).
8. Julie E. Cohen & Mark A. Lemley, *Patent Scope and Innovation in the Software Industry*, 89 CALIF. L. REV. 1 (2001).
9. Eric Allen Engle, *When is Fair Use Fair?: A Comparison of E.U. and U.S. Intellectual Property Law*, 15 TRANSNAT'L LAW. 187 (2002).
10. Maureen A. O'Rourke, *Toward a Doctrine of Fair Use in Patent Law*, 100 COLUM. L. REV. 1177 (2000).
11. Dan L. Burk and Mark A. Lemley, *Policy Levers in Patent Law*, 89 VA. L. REV. 1575 (2003).
12. Julia Alpert Gladstone, *Why Patenting Information Technology and Business Methods is not Sound*, 25 HAMLINE L. REV. 217 (2002).
13. Dan L. Burk & Mark A. Lemley, *Is Patent Law Technology-Specific?*, 17 BERKELEY TECH. L.J. 1155 (2002).
14. Mark A. Haynes, *Commentary: Black Holes of Innovation in the Software Arts*, 14 BERKELEY TECH. L.J. 567 (1999).
15. Daniel Lin, Matthew Sag & Ronald S. Laurie, *Source Code Versus Object Code: Patent Implications for the Open Source Community*, 18 SANTA CLARA COMPUTER & HIGH TECH. L.J. 235 (2002).
16. Pamela Samuelson, Randall Davis, Mitchell D. Kapor and J.H. Reichman, *A Manifesto Concerning the Legal Protection of Computer Programs*, 94 COLUM. L. REV. 2308 (1994).
17. James Boyle, *The Second Enclosure Movement and the Construction of the Public Domain*, 66 LAW & CONTEMP. PROB. 33 (2003).
18. Mathias Strassery, *The Case Against Open Sources*, 2001 STAN. TECH. L. REV. 4 (2001).
19. Michael J. Madison, *Reconstructing the Software License*, 35 LOY. U. CHI. L.J. 275

(2003).

20. Bruce Abramson, *Promoting Innovation in the Software Industry: A First Principles Approach to Intellectual Property Reform*, 8 B.U. J. SCI. & TECH. L. 75 (2002).
21. Patrick K. Bobko, *Linux and General Public Licenses: Can Copyright Keep "Open Source" Software Free?*, 28 AIPLA Q.J. 81 (2000).

五、網站資源

1. Free Software Foundation, at <http://www.fsf.org>
2. Open Source Initiative, at <http://www.opensource.org>
3. David A. Wheeler, at <http://www.dwheeler.com>
4. Berkeley Software Distribution, at <http://www.bsd.org>
5. The SCO Group, at <http://www.thescogroup.com>
6. InfoWorld, at <http://www.infoworld.com>
7. eWeek, at <http://www.eweek.com>
8. Net World Fusion, at <http://www.nwfusion.com>
9. CNET, at <http://news.com.com>
10. Groklaw, at <http://www.groklaw.net>
11. ComputerWorld, at <http://www.computerworld.com>
12. NewsForge, at <http://www.newsforge.com>
13. The New York Times, at <http://www.nytimes.com>
14. Linux Today, at <http://linutxtoday.com>
15. ZDNet, at <http://zdnet.com.com>
16. Linux Insider, at <http://www.linuxinsider.com>
17. The Sydney Morning Herald, at <http://www.smh.com.au>
18. Linux Dev Center, at <http://www.linuxdevcenter.com>

19. Bruce Perens, at <http://www.perens.com>
20. The Open Group, at <http://www.opengroup.org>
21. USENIX, at <http://www.usenix.org>
22. Slashdot, at <http://www.slashdot.org>
23. The Unix Heritage Society, at <http://www.tuhs.org>
24. Open Source Risk Management, at <http://www.osriskmanagement.com>
25. Open Source Victoria, at <http://www.osv.org.au>
26. O'Reilly, at <http://www.oreillynet.com>
27. Apache Software Foundation, at <http://www.apache.org>
28. Linux Journal, at <http://www.linuxjournal.com>
29. KDE Developer's Corner, at <http://developer.kde.org>
30. Lindows, at <http://www.lindows.com>
31. Microsoft, at <http://www.microsoft.com>
32. SourceForge, at <http://sourceforge.net>
33. Foundation for a Free Information Infrastructure, at <http://swpat.ffi.org>
34. PUBPAT, at <http://www.pubpat.org>
35. Federal Trade Commission, at <http://www.ftc.gov>
36. Creative Commons, at <http://creativecommons.org>
37. Free Culture, at <http://www.free-culture.cc>
38. 自由軟體入口網站：<http://www.oss.org.tw>
39. 自由軟體鑄造場計畫網站：<http://www.openfoundry.org>