

序

當本書第一版的作者 — Karim Yaghmour — 於 1997 年任職一家硬體製造商的時候，曾率先提議在嵌入式系統中使用 Linux，他的建議在當時不僅令人感到驚訝，也受到某些質疑。今日，大多數嵌入式系統不是已經在使用 Linux，就是正在考慮使用 Linux。的確，有不少大公司及政府機關在其嵌入式軟體的需求上逐漸仰賴 Linux。

本書第一版頗受歡迎，但自本書第一版發行以來，Linux 核心與相應的工具取得了不少的進展，這使得 Linux 更具吸引力。其中最主要的是，出現了若干的即時擴充與配套環境，這會在本書第二版最後三章加以探討。

此外，由於本書第一版、熱心的開放源碼與自由軟體程式員已經簡化了 GNU/Linux 組件（此處之所以提到 GNU 是因為，要建立有效的 Linux 系統，需要用到來自 GNU 這個自由軟體計畫的工具）的建構與安裝程序，因此本書第二版會介紹許多令人感到驚奇的高階工具，包括 Eclipse 以及用於替嵌入式 Linux 系統「打造建構工具」(build the build tools) 的各種工具。不過，我們仍會提供替大量的低階資訊給那些有需要的人，以協助您了解它們在背後用到了哪什麼輔助工具。

要跟上 Linux 各部分與相關工具的爆炸性進展，最好的辦法就是由嵌入式與即時 Linux 相關議題的各種專家來告訴你。因此，本書第二版的作者加入了幾位 GNU/Linux 社群中關鍵的參與者，包括了那些進行核心開發或建立相關專案的人士。

本書的特點

Linux 在一些嵌入式應用的成功，引發廣泛的興趣和熱烈的迴響，這導致了「嵌入式 Linux」相關文章、網站、公司及文件的氾濫。除了一閃即逝的公告，雜誌裡的文章，數以百計的專案和產品，都在宣告了將 Linux 使用於嵌入式系統的輕而易舉，然而正在探索有用指引的專業開發者們，卻仍在為用於建立嵌入式 Linux 系統的基本方法和技術尋覓解答。

目前大部分的文件都只著重在，如何使用各類套裝、即用的跨平台開發工具，以及目標二元碼（target binary）【譯註】。有的文件則只侷限在如何在嵌入式目標上執行 Linux。

本書與現有的文件不同，除了你想要使用的 Linux 之外，它並不會假定你手邊有哪些工具，或是限定專案的範圍。本書只會要你透過 Internet 連線下載必要的套件、瀏覽特定的線上文件，以及透過專案的郵遞論壇（mailing list）與其他開發者一起分享經驗。你還需要一部開發主機，以及目標硬體的說明文件，但我不會要求你向任何廠商購買任何產品或服務。

除了提供最大的自由度，以及讓你掌控自己的設計之外，這麼做最接近帶頭在嵌入式系統中使用 Linux 的先驅者們所採行的方式。這些先驅者對 Linux 所做的事，本質上包括將 Linux 拆開以符合應用的需要，以及根據他們目的來定義 Linux。因此，Linux 突破 embedded 世界的方式不同於許多軟體廠商將其產品擴展至新應用領域的做法。身為嵌入式系統開發者的你可能會發現，相較於廠商所販賣的產品，Linux 比較容易應用到你的設計中。

本書的做法就是提供所有的細節，以及討論在嵌入式系統中使用 Linux 時將會遇到的各種困境，讓你得以把 Linux 應用在設計中。雖然本書的內容不可能涵蓋所有的嵌入式設計，但是書中所提供的資源卻可讓你輕易獲得必要的資訊，使你得以在嵌入式系統中定義以及使用 Linux。

撰寫本書的目的在於拉近「於設計中使用開放原碼和自由軟體之嵌入式系統開發者」跟「建立和維護這些開放原碼和自由軟體套件之開發者」的距離。儘管有許多主流嵌入式系統開發者（其中不乏高水準的程式設計者）會仰賴第三方（third-party）提供其所需要的嵌入式 Linux，無疑地他們也有機會對其所仰賴的開放原碼（open source）和自由軟體（free software）專案做出貢獻。最後，這種動力，將可確保 Linux 持續成為嵌入式系統首選的作業系統。

本書的讀者

本書鎖定的第一種讀者，是那些想要在未來或目前的專案中使用 Linux 之有經驗的嵌入式系統設計者。本書會假定這一類讀者已經熟悉開發嵌入式系統所使用的技術和術語，像是交叉編譯（cross-compiling）、進行 BDM 或 JTAG 除錯，以及如何處理不完善或

●.....
譯註 目標（target）係嵌入式系統的代名詞，通常用於軟體開發期間，用來區分嵌入式系統與開發主機。目標二元碼（target binary）係指開發板上以二元（0 與 1）形式存在記憶體中供 CPU 執行的機器碼（machine code）或目的碼（object code）。

不完全的硬體。如果你是這一類的讀者，你可能會想要跳過前面一些章節中，與嵌入式系統開發背景知識有關的部分。然而，你可能需要閱讀一些前面的章節（尤其是第 2 章），因此這些章節包含了在嵌入式系統中使用 Linux 核心的特別提示。

本書鎖定的第二種讀者，是那些想要熟悉嵌入式 Linux 系統開發工具和技術之無經驗的嵌入式系統開發者。然而，本書並非嵌入式系統的入門書，不過若你想探究本書所討論到的一些課題，你可以參考一些入門性質的教科書。

如果你是個對 Linux 已經很熟悉的玩家（power user）或系統管理者，本書應該可以協助你自行定義 Linux 的安裝。例如，若你發現到發行套件（distribution）安裝了過多的套件（package），而你想要從零開始建立自己的發行套件（custom distribution），本書有許多部分應該可以派上用場，特別是第 6 章的內容。

最後，本書應該可以協助那些想要瞭解如何建立與操作 Linux 系統的程序設計者或 Linux 的熱衷者。儘管本書並未說明如何建立一般用途發行套件（general-purpose distribution），不過本書所提到的許多技術，在一定程度上，可以媲美用於定義嵌入式 Linux 安裝的一般用途發行套件。

本書的範疇及背景知識

要在嵌入式系統中展現 Linux 的最佳能力，你需要具備以下背景知識（這些知識在許多書籍中都交代得很清楚）：

嵌入式系統

一般而言，你必須熟悉嵌入式系統的開發、程式設計和除錯，這包括軟體和硬體兩方面。

Unix 系統管理

你必須能夠勝任各種的系統管理工作，像是硬體的組態、系統的設定、維護，以及利用 shell 命令稿讓管理工作自動化。

Linux 裝置驅動程式

你必須知道如何對各種 Linux 裝置驅動程式進行開發及除錯。

Linux 核心的運作原理

你必須盡可能瞭解核心的運作原理。

GNU 軟體開發工具

你必須具備有效利用 GNU 工具的能力。這包括瞭解許多常被認為是晦澀難懂的選項和公用程式。

本書會假定你至少熟悉基本概念中所論述的每個項目。另一方面，閱讀本書，你無須知道，例如，Linux 裝置驅動程式是如何建立的，或與嵌入式系統開發有關的每一件事。翻閱本書的時候，若看到符合你的嵌入式系統中的 Linux 用法，你可能會覺得需要取得與這個 Linux 用法的進一步資訊。

儘管本書只討論如何在嵌入式系統中使用 Linux，但是對想要在嵌入式系統中使用 BSD 的開發者來說，多少會有一些幫助。不過本書所作的許多說明，都必須依據 BSD 與 Linux 間的差異，重新詮釋過。

本書的架構

本書由四大部分構成。第一部分由第 1 到 3 章組成。這三章的內容涵蓋建構任何嵌入式 Linux 系統必備的基本知識。儘管這三章並未提到任何實作程序，不過它們都是建構嵌入式 Linux 系統時不可或缺的知識。

第二部分由第 4 到 9 章組成。這幾章的內容都非常重要，其中安排了用來建構任何嵌入式 Linux 系統的基本步驟。不管你的系統的用途或功能是什麼，都必須閱讀這幾章的內容。

第三部分（也就是本書第一版的最後一個部分）由第 10 和 11 章組成。雖然這兩章所包含的內容都很重要，不過並非建構嵌入式 Linux 系統的要素。

本書最後一個部分由第 12 到 14 章組成。這三章對即時（real-time）這個議題做了深入的探討，其中包括即時的的不同應用，以及何時應該考慮各種的實作品以及可用的變體。我們很幸運和榮幸能夠邀請到「Xenomai」以及「Linux 核心之 RT patch」的實作者來撰寫這幾章的內容。

第 1 章，入門指引，將深入嵌入式 Linux 系統的世界。本章首先會介紹基本定義，接著會從實務的觀點來探討嵌入式 Linux 系統的議題，包括從嵌入式系統的觀點來探討開放原碼（open source）和自由軟體（free software）的版權問題。然後會介紹本書其他部分將會用到的範例系統，以及本書將會用到實作方法。

第 2 章，基本觀念，用於描述建構任何嵌入式 Linux 系統時所必須具備的基本觀念。

第 3 章，所支援的硬體，完整地介紹了 Linux 所支援的嵌入式硬體，以及告訴讀者何處可以找到實作這些硬體的驅動程式（driver）和子系統（subsystem）。本章還會探討處理器的架構、匯流排和介面、I/O、儲存裝置、一般用途的網路功能、工業等級的網路功能，以及系統監控。

第 4 章，開發工具，將說明如何安裝與使用各種用於建構嵌入式 Linux 系統的開發工具。這包括了探討如何將 Eclipse 用於嵌入式 Linux 開發，以及如何從零開始建構與安裝 GNU 工具鏈元件。此外還會分節討論 Java、Perl、Python 和其他語言，而且會一節專門探討可用來與嵌入式目標互動的各種終端機模擬程式。

第 5 章，核心方面的考量，將探討在嵌入式系統中 Linux 核心的選擇、組態、交叉編譯、安裝以及使用。

第 6 章，根檔案系統的組成（這一章的內容由 Michael Opdenacker 負責更新），將說明如何使用本書稍早所提到的元件來建構一個根檔案系統，這包括 C 程式庫的安裝，以及建立適當的 */dev* 項目。比較值得注意的是，本章的說明涵蓋了 BusyBox、*embutils* 以及 System V *init* 的安裝和使用。

第 7 章，儲存裝置的操作（這一章的內容由核心開發者 David Woodhouse 負責更新），將解說如何操作以及為嵌入式 Linux 系統設置儲存裝置。本章的重點在於固態儲存裝置，例如原生的 flash 和 DiskOnChip 裝置，以及 MTD 子系統。

第 8 章，根檔案系統的規劃，將說明如何為嵌入式系統的儲存裝置設定第 6 章所建立的根檔案系統。這包括了建立（基於 JFFS2、CRAMFS 或其他特定檔案系統的）檔案系統映像，以及經由 NFTL 使用磁碟形式的檔案系統。

第 9 章，設定 Bootloader，將針對每個嵌入式 Linux 架構探討各種可用的 bootloader（開機載入器）。並且會進一步探討如何在 DiskOnChip 裝置上使用 GRUB 以及 U-Boot。此外還包含以 BOOTP/DHCP、TFTP 和 NFS 進行網路開機。

第 10 章，設定網路服務，會把重點擺在提供各種網路服務（像是 SNMP、SSH 和 HTTP）之軟體套件的組態、安裝和使用。

第 11 章，除錯工具（這一章的內容由 Michael Boerner 負責更新），將探討為嵌入式 Linux 系統開發軟體時可能遭遇到的主要除錯議題。這包含了在跨平台開發環境中使用 *gdb*、使用 Eclipse、進行追蹤、效能分析以及記憶除錯。

第 12 章，即時 Linux 入門，將說明即時功能的價值，而且會持平地討論，你何時需要各種的即時功能，同時會介紹你可以使用 Linux 達到即時行為的各種方法。本章的作者為 Xenomai Real-Time System 的創立者與維護者 Philippe Gerum。

第 13 章，Xenomai 即時系統（本章的作者也是 Philippe Gerum）為如何以 Xenomai 達到即時目標以及如何與嵌入式 Linux 有效地結合提供了一個高階觀點。

第 14 章，RT patch，Linux 核心的 RT 修補類似於 Xenomai，本章將會說明如何啟用它的功能。本章的作者是此修補的關鍵開發者 Steven Rostedt。

儘管第 7 到 9 章看起來像是各自獨立，不過它們的內容卻有高度的關連性。舉例來說，要瞭解第 7 章所探討之目標儲存裝置的設定，得先瞭解第 8 章所探討之目標檔案系統的結構，反之亦然。此外，儲存裝置的設定還必須先瞭解第 9 章所探討之 boot loader 的規劃與操作，反之亦然。因此，建議各位先一口氣讀完第 7 到 9 章，再執行這幾章所提到的任何指令。這樣當你設置目標系統的時候，你才有辦法按照這幾章所描述的程序進行操作。

本書所用到的硬體

正如我們將在第 3 章所見，Linux 支援了各式各樣的硬體。由於篇幅有限，本書只能選用幾種嵌入式系統作為各種程序的測試平台。這些系統中有些是主流市場上的商品，例如 OpenMoko-based NEO 1973。我刻意將它們列入，主要是讓有興趣的讀者可以輕易找到用來學習建立嵌入式 Linux 系統的材料。當然，你也可以使用老舊的 x86 PC 來作實驗，不過這樣比較沒意思就是了。

為了解說可以使用 Linux 的各種目標架構，我在各章所舉的範例，會刻意使用不同的目標硬體。儘管各章的說明基於不同的架構，不過各章所提到的命令卻可同時應用到其他架構上。例如，若某章所舉的例子中需要用到 *arm-linux-gcc* 命令（用於 ARM 的 *gcc* 編譯器），則只要將它代換成 *powerpc-linux-gcc* 命令，就可以將這個例子應用到 PPC 目標架構上。若某章所列出的架構不止一個，則其所探討的主要架構會列在第一個。例如，第 5 章所舉的命令範例，會將重點擺在 PowerPC 上，而與 ARM 有關的命令只會提到幾個。

除非特別聲明，否則主機的架構總是不同於目標的架構。舉例來說，雖然第 4 章所提到的是，以 PPC 主機為 x86 目標建構工具。不過只要稍微或不須修改，相同指令就可以在 SPARC 或 S/390 上面執行。請注意，本書前幾章大部分的內容與架構無關，所以無須提供任何架構專屬的命令。

軟體的版本

當然，嵌入式 Linux 系統所依靠的中心軟體就是 Linux 核心。本書將焦點擺在 2.6 版的 Linux 核心上，尤其是 2.6.22。以 2.6 版為依據所做的變更，大概只會影響到本書前面幾章所提供的一些資訊。也就是說，新版本所支援的硬體大概會比第 3 章所列的還多。不過就 2.6 版來說，本書所提到的基本程序不太可能需要變更。

此外，本書探討了 40 種以上開放原碼及自由軟體套件的組態、安裝和使用。每種套件的維護各自獨立，而且開發步調也各不相同。因為這些套件會與時俱進（change over time），有可能本書所提到的套件的版本，在讀者閱讀的當時已經過時了。為了降低軟

體更新對內文所造成的影響，本書會盡可能讓內文與版本無關。由於與軟體的變動無關，所以全書的結構和各章的結構不太可能需要更動。再者，本書所提到的許多套件已經問世一段時間，所以它們不太可能會有什麼重大的變動。例如，本書所使用的 GNU development toolchain（開發工具鏈），其中各元件的安裝、設定和使用程序，相對來說，已經好多年沒什麼變動了，所以未來也不太可能有任何重大的變動。這同樣適用於本書所提到的大多數軟體套件。

本書的慣例

定寬字

用來表示程式碼的內容或命令的輸出，以及程式碼中的關鍵字。

定寬粗體字

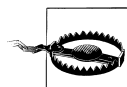
用來表示使用者所鍵入的文字。

斜體字

用來表示目錄的名稱、程式和命令的名稱、命令列選項和網址。



用來表示秘訣、建議或通用註解。



用來表示警告或注意事項。

使用範例程式

本書的宗旨是協助你搞定工作。一般而言，你可以將本書的程式碼用在你的程式裡，或是在文件裡提及，而無須要求我們的同意，除非你想大幅度引用。舉例來說，使用本書範例裡的幾個程式片段，無須經過我們的同意；但如果打算將 O'Reilly 書籍裡的範例程式燒錄成光碟來銷售或散佈，則需要授權。引用本書內文或程式片段來回答問題，不需要授權；但如果大量引用本書範例到你的產品說明書裡，則需要知會我們。

如果你引用本書（內文或範例程式），我們會感謝你註明出處，但沒要求你必須得這麼做。如果你願意，請註明書名、作者、出版公司以及 ISBN。例如：「Building Embedded Linux Systems, by Karim Yaghmour, Jon Masters, Gilad Ben-Yossef, and Philippe Gerum. Copyright 2008 Karim Yaghmour and Jon Masters, 978-0-596-52968-0」。

建議和問題

歐萊禮公司是世界性的電腦資訊出版公司。我們永遠樂意聽到讀者對出版品的意見，包括如何讓本書可以更好的建議、指正本書的錯誤、或是讀者建議本書往後改版時，應該再加進來的其他主題。以下是本公司的聯絡資料：

美商歐萊禮股份有限公司台灣分公司

電話：(02) 2709-9669 傳真：(02) 2703-8802

網頁：<http://www.oreilly.com.tw>

電子郵件：mail@oreilly.com.tw

與本書有關的線上資訊（可能包括勘誤、範例程式、相關連結）：

原文書

<http://www.oreilly.com/catalog/9780596529680>

中文書

http://www.oreilly.com.tw/product2_linux.php?id=a242

作者將會替本書架設一個網站：

<http://www.embeddedlinuxbook.org/>

第一版致謝詞

「E quindi uscimmo a riveder le stelle.」^註，是義大利詩人但丁在《神曲》(Divine Comedy) 第一部《地獄篇》(Inferno) 的結語。這句話可能會讓讀者誤以為，我撰寫本書時沒有一點喜悅之情。但丁的這句話明確地表達了我完成你手上這本書當時的感覺。特別是我必須承認，在嵌入式系統中使用 Linux 的資訊多如牛毛，要加以彙整談何容易，更遑論將它們一股腦兒全塞進一本書，好告訴讀者建立嵌入式 Linux 系統的實際方法。還好，我背後有許多有能力的人願意協助我。

●.....

註 「離開此地，只見閃耀燦爛的滿天繁星。」

首先，要感謝我的編輯 Andy Oram。就像協助但丁（Dante）遊歷靈界的維吉爾（Virgil），Andy 帶領著我渡過撰寫本書的每個階段。除了別的協助之外，他還耐心地糾正我非成語的成語，以確保我的文章不會辭不達意，並且鐵面無私地指出哪些章節的內容不夠深入。你現在之所以會讀到比較好的文章，這都得益於 Andy 的指教。同樣地，我要感謝 Ellen Siever，本書一開始是我跟她一起合作的。但是我們的合作關係卻在本書完成之前提早結束了，有許多點子最後之所以能夠成功放入本書，都要歸功於她建設性的回饋。

我真的非常幸運，本書能夠遇到這麼一個傑出的審閱者團隊，我非常感謝他們付出大量的時間和精神閱讀、改正以及指出本書的種種問題。他們是 Erik Andersen、Wolfgang Denk、Bill Gatliff、Russell King、Paul Kinzelman、Alessandro Rubini、David Schleef 和 David Woodhouse。其中，我特別要感謝 Alessandro 追求盡善盡美的執著。本書接下來的內容，如果還能找到任何錯誤，毫無疑問地這都是我的問題。

撰寫關於如何在嵌入式系統中使用 Linux 的書，必須用到許多不同的硬體。這些嵌入式硬體的價錢通常很貴，我要感謝那些提供設備給我使用的公司和個人。我特別要感謝 Kontron 公司的 Stephane Martin，提供了一張 Teknor VIPer 806 實驗版給我；DENX Software Engineering 公司的 Wolfgang Denk，提供了一張 TQ components TQM860L PPC 實驗版給我；以及 Zee2 公司的 Steve Papacharalambous 和 Stuart Hughes，提供了一個 uCdimmm 系統給我。

身為開放原碼和自由軟體社群的使用者與貢獻者，我屢次受益於社群中其他成員所分享出來的知識與成果，這鼓舞並促成了我撰寫本書的意願。基於此，我要感謝許多人。首先，我要感謝 Michel Dagenais 教授的信任與指導，讓我有機會毫無拘束地去探索這個未經探勘的領域（uncharted terrain）。Linux Trace Toolkit 的開發（這是我碩士論文的一部分），讓我與開放原碼和自由軟體社群的關係越來越密切。身為其中的一份子，我遇到了許多洞察力卓越的人，以及受到了許多協助，我非常感激他們。他們是 Jacques Gelinas、Richard Stallman、Jim Norton、Steve Papacharalambous、Stuart Hughes、Paolo Mantegazza、Pierre Cloutier、David Schleef、Wolfgang Denk、Philippe Gerum、Loic Dachary、Daniel Phillips 和 Alessandro Rubini。

最後，我一定要說，我欠 Sonia 一份情，是她異於常人的耐心，讓我得以無後顧之憂地將全部的時間投入本書的測試與撰寫。由於她的支持與關心，讓本書得以順利完成。「La main invisible qui a écrit les espaces entre les lignes est la sienne et je lui en suis profondément reconnaissant.」【註】

●.....

註 「這本書的字裡行間裡充滿了你的倩影，感謝妳為我所做的一切。」

第二版致謝詞

在 Karim 第一次提到要更新《建構嵌入式 Linux 系統》這本書的時候，我無法想像這將是多麼有趣和瘋狂的一件事。當時正值我從英國移民美國的最後階段，忙碌的生活讓我沒有喘息的機會。一路上，在一些了不起的朋友以及本書其他作者的協助之下，把我的想法轉變成你現在所閱讀的這本書。我們的共同願望是，為嵌入式 Linux 增加一本可供參考的文獻。

首先，我想要感謝我的朋友 Karim Yaghmour 讓我修改他的原稿，Andy Oram 耐心的指導和編輯絕技，以及 Isabel Kunkle 協助 Andy 把一群繁忙的作者結合在一起。我還要感謝 Marlowe Shaeffer 以及歐萊禮工作團隊對細節堅定不移的關注，尤其是在出書計劃接近尾聲的時候。

我還想要感謝本書的其他作者願意接受挑戰，並幫忙把這項出書計劃進行下去，他們是 Michael Boerner、Michael Opdenacker、Steven Rostedt、Gilad BenYossef (Codefidence 公司的首席技術長)、Phillipe Gerum 和 David Woodhouse。對於多年來我所認識的朋友們，即使我們每年只在 Linux Symposium 研討會上見一次面，但是我很感謝你們的幫助，使得本書的整體品質得以提升。同樣的，我要感謝 Tim Rikers、Vince Skahan、Mark VandenBrink 等人所提供的審查意見，以及偶爾會聽我談及本書的許多其他人。本書的內容若還能找到任何錯誤和遺漏之處，完全是我的責任。

如果沒有全世界成千上萬人的努力，嵌入式 Linux 系統將變得毫無意義。其中有些人參與了本書第一或第二版的寫作，而且在許多人的協助之下，使得 Linux 成為嵌入式開發者最有價值和可行的選項。要列出這些人的名字——感謝似乎不太可能，所以我要對這些人表達我們最真誠的謝意——我也要鼓勵你們去感謝那些對你們的開發專案提供上游 Linux 核心的人。也請你們透過自己覺得最適當的方式來鼓勵你們的員工和客戶去做相同的事。

此外，我想要感謝我的朋友和家人們，對我的許多追求和隨意的瘋狂舉動，永無止境地支持。我的媽媽和爸爸這些日子很少看到我（我住在 3,000 英里以外的另一個國家，這是一個難以處理的時間延遲），但正如你們所期望的，你們一直都是最好的父母，儘管你們的兒子入了美國籍成為了「叛國者的狗」（感謝，老爸，這就是你古怪的歷史洞察力！）如同我的最小妹妹 Holly，我的姊姊 Hannah 和姊夫 Joe Wrigley（另一個紅帽製造商！）總是叫人驚奇。我的祖母常寫信告訴我家中所發生的事，我總是期待遠離電腦閱讀她的來信。

這個出書計劃之所以能夠成功還要感謝許多朋友的貢獻，即使他們對此事並不瞭解。這些人包括 Deepak Saxena、Hussein Jodiyawalla、Bill Weinberg、Alison Cornish、Grace Mackell、Andrew Schliep、Ginger Diercks、Kristin Mattera 和 James Saunders、Karen Hopkins、Andrew Hutton，以及 Emilie Moreau（還包括 Denali 和 Nihao）、Madeleine 和 Chris Ball、Tim Burke、Lon Hohberger、Chris Lumens、Jon Crowe、Rachel Cox、Catherine Nolan、Toby Jaffey（以及 Sara 和 Milly）、David Brailsford、Jeff 和 Nicole Stern、Catherine Davis、Mary-Kay 和 Luke Jensen、Philippe De Swert、Matt Domsch、（Secret Lab 的）Grant Likely、Hetal Patel、Mark Lord、Chris Saul、Dan Scrase 以及 David Zeuthen。還要特別感謝 Sven-Thorsten Dietrich 和 Aaron Nielson 在正確的時刻志同道合的瘋狂行徑。

最後，感謝我的好朋友諾丁漢大學的 David Brailsford，以及 Oxford Instruments 的 Malcolm Buckingham 和 Jamie McKendry 相信我，讓我實驗 Linux 和超導磁體，以及 MontaVista 英國公司的 Ian Graham 讓我有機會參與大型的專案。我還欠 Steamballon 的 Andrew Hutton 和 Craig Ross（以及 Linux Symposium 的主辦者）一份人情，感謝他們這些年來支持我在嵌入式方面的努力。我還要特別感謝 Gary Lamb（Global Engineering Services—我們的嵌入式團隊）、Clark Williams 和 Red Hat 公司的 Tim Burke 持續地支持我，以及我在 Red Hat 公司和其他大型 Linux 公司的所有朋友。

—Jon Masters 于 Cambridge, Massachusetts

