Linux PCMCIA HOWTO 中文版

作者: David Hinds, dhinds@hyper.stanford.edu
譯者: 謝崑中 HSIEH Kun-Chung, 台灣 中華民國 Taiwan, Republic of China (NOT People's Republic of China) (*linuxer.bbs@cis.nctu.edu.tw*)
1998年2月19日 v2.5版

本文件主要描述如何在 Linux 下安裝及使用 PCMCIA 卡服務程式並回答大家常問到的問題。這份文件最新的版本會放在 <ftp://hyper.stanford.edu/pub/pcmcia/doc>。另外 HTML 版本會被放在 <http://hyper.stanford.edu/HyperNews/get/pcmcia/home.html>。

目錄

1 一般資訊及硬體要求

1.1 簡介

給 Linux 使用的卡片服務程式是個完整支援 PCMCIA 的套件。 它包含了組成 PCMCIA 卡服務程式應用程式介面的一套可載入模組、一套給 特別卡片使用的客户驅動程式式、一個可反應出卡片插入、抽出並依需要來 載入或卸下驅動程式的卡片管理精靈。 它支援“熱機交換”PCMCIA 卡 片，因此呢，卡片就可以在任何時間插入或抽出了。

本軟體仍在發展中，所以可能在程式裏有小虫，請您小心使用。我將盡力把 我所知道的問題修正，可是如果您不告訴我您在使用上所發生的程式問題， 也許我就無法知道問題所在。因此，如果您使用這個程式，請來信告訴我您 使用經驗，不論是好是壞。

如果您有任何怎樣改進這份文件的寶貴意見，也請讓我知道 (`dhinds@hyper.stanford.edu`)。

1.2 著作權宣告與解除修款

1.3 最新版本是哪個？我可從哪兒拿到？

目前最新的卡片服務程式爲 3.0 版， 小幅修改或除虫過的版本爲 3.0.1、 3.0.2 等等如此類推。

新的原始程式碼可從 hyper.stanford.edu 內的 /pub/pcmcia 目錄找到， 叫做 pcmcia-cs-3.0.?.tar.gz 。 在那兒通常會有數個版本在。通常我只會把最新的幾個修改版放在此處。新 版本通常都包含了未測式完成的程式碼，因此我也會把上一個穩定的修正版 放在那兒以備不時之需。現在的穩定版爲 2.9.12 版，要使用哪個版本較好 呢？隨您的意思啦。在 CHANGES 檔案內會做簡單的版本差異比較。

hyper.stanford.edu 會被對映到 sunsite.unc.edu (以及所有 的 sunsite 對映網站上) 的 /pub/Linux/kernel/pcmcia 內。

假使您不想自己編譯 PCMCIA 程式的話， 市面上主要的 Linux 供應商如 Slackware、 Red Hat、Caldera 、Yggdrasil 等等都有已編譯好的程式可使 用。

1.4 哪些系統被支援呢？

這些程式應該可以在 Linux 相容的筆記型電腦上正常工作。常見的 PCMCIA 控制器都有支援， 這些包括以下廠商： Intel、 Cirrus、 Vadem、VLSI、 Ricoh 及 Databook chips。 亦支援 IBM 及 Toshiba 的筆記型電腦。只要 是直接插上 ISA 介面槽 (SCSI-to-PCMCIA 或 IDE-to-PCMCIA) 不管是的 桌上型電腦之 PCMCIA 插槽裝置亦受支援。

CardBus 橋接控制器遵循著 “Yenta” 暫存器規格 (包括 TI、Cirrus、SMC、O2Micro、Omega Micro 和 Ricoh 的晶片) 都被支援，但 對於 32 位元的 CardBus 卡片仍只屬於實驗性質。驅動程式版本比 3.0 版還早的在 CardBus 插槽中也只能支援到 16 位元的卡。 由於筆記型電腦硬 體科技的快速進步，新型控制器時常出現，因此，新的產品出現在市場上與 對映的驅動程式支援的上市可能會有些時間上的出入 。

在 Hyundai 筆記型電腦上使用的 Motorola 6AHC05GA 控制器並不被支援。 在 HP Omnibook 600 上的 PCMCIA 控制器亦不被支援。

1.5 哪些PCMCIA卡被支援呢？

目前的版本有網路卡、 數據卡、串列埠卡、SCSI 介面卡、ATA/IDE 硬碟卡 以及支援 SRAM 和快閃記憶體卡的記憶卡驅動程式，SUPPORTED.CARDS 檔案內包含了各版本中有支援的 PC 卡名稱（至少在一個系統上可使用皆列 入）。

但不在列表內的 PC 卡也有可能可以使用。基本上，所有的數據卡都可以使 用。 有些未列入的網路卡若他們是被支援的卡之 OEM 版本亦可使用。其他 IO 卡（如緩衝記憶體卡、音效卡等等）大部份目前都不支援， 除非有人爲 它們寫驅動程式。

1.6 什麼時候我的新卡片才會被支援呢？

遺撼的是，這些廠商並沒有付給我任何酬勞來寫設備驅動程式。因此，如果 您想讓您的卡被支援，您至少要做點事。 理想上，我較喜歡像做 Linux 核 心那樣的模式亦即我主要負責 PCMCIA 的核心程式碼，其他的朋友則貢獻在 特殊的 PC 卡的驅動程式上。在 SUPPORTED.CARDS 檔案中提到有些卡 所用驅動程式正在進步中。 在我能力所及的地方我會儘量幫忙， 不過透過 Email 來做除錯核心設備驅動程式並不是很有效率的。

製造商若有興趣讓自己公司的產品被 Linux 所支援可與我連絡做進一步的 洽詢與安排。

1.7 郵寄名冊及其他資訊來源

我曾經維護一個 Linux PCMCIA 的使用者郵寄名冊。而現在我已將我的網頁 調整成″超媒體新聞″網站，裡頭有一系列有關 Linux PCMCIA 的資訊項目： 有針對不同型式的卡片之安裝與設定、PCMCIA 卡之相關程式設計與除錯。這個 Linux PCMCIA 資訊網頁在 http://hyper.stanford.edu/HyperNews/get/pcmcia/home.html 上。使用者可以申請經由電子郵件的方式來通知您較感興趣的新訊息。我期 望這樣子可以將這些超過 HOWTO (″怎麼做″) 範圍的問題討論變爲一個 實用的資訊寶庫。

有個針對筆記型電腦使用 Linux 的郵寄名冊叫做 “linux-laptop” 郵寄 名冊。 若要更進一步的資訊， 請寄一封含有 “help” 一字的電子郵件到 majordomo@vger.rutgers.edu。 若您要訂閱這份資訊，請寄一封含有 “subscribe linux-laptop” 的電子郵件到上面相同的 Email 地址。這份 郵寄名冊是一份有關於 Linux PCMCIA 的不錯論壇哦。

Linux Laptop 網頁在 http://www.cs.utexas.edu/users/kharker/linux-laptop 上， 裏面有連結至特定的筆記型電腦在安裝 Linux （及 PCMCIA ）時的資 訊。還有一份可尋找系統建構資訊的資料庫。

1.8 怎麼不提供 PCMCIA 的二進位碼呢？

對我而言，提供二進位碼會困擾我。這有點複雜，因爲有些功能只能在編譯 時被選擇，而且 PCMCIA 模組又那麼依賴合適的核心建構。所以，我只能提 供伴隨著合適的核心之未編譯的模組。況且，當要安裝到一個乾淨的系統上 時是最需要未編譯的模組的。 一般來說，每個 Linux 供應商會需要在安裝過程中設定 PCMCIA 來使用它的功能。每個供應商都有自己的程序，所以由 我來供應開機及根磁碟給他們好像有點不太適合。

PCMCIA 現在已是許多主要 Linux 供應商的一部份了， 這包含 Red Hat、 Caldera、 Slackware、Yggdrasil、Craftworks 及 Nascent Technology。

1.9 爲什麼 PCMCIA 套件這麼大呢？

噢! 首先我要說的是，它眞的不大。所有的驅動程式模組加起來只需約 200 K 的磁碟空間而已。 公用程式部份加起來約 70K，放在 /etc/pcmcia 裏的 資料約 30K。當執行時，PCMCIA 的核心模組佔用約 48K 的系統記憶體。通 常 cardmgr 精靈除了在卡片被插入或移出時外都會被移出記憶體的。 所有 套件的大小並不會跟 DOS 的卡片服務程式執行時有多大的差別。

這與 DOS 的單點啓用器相比似乎還比它們的好。 特別是有人並不會使用到 PCMCIA 的所有功能：比如說能源管理或熱機交換功能。 因爲單點啓用器大 部份只需支援一種或少許的卡片以及支援少許的 PCMCIA 控制器，所以可以 做得小一些。如果有人要寫個天才型的數據機開啓器，到後頭可能會把許多 卡片服務的功能都加進去才能應付不同的卡片廠商以及控制器了。

2 編譯與安裝

2.1 準備工作與核心設定

在開始前， 您應該想想是否需要自己來編譯 PCMCIA 套件。 因爲大部份的 Linux 供應商版本已含有已編譯過的 PCMCIA 驅動程式套件。一般上來說， 如果您需要最新的驅動程式或您更動的 Linux 供應

商版本內的核心和您原 本的 PCMCIA 驅動程式無法相容時才會需要自己動手做。 雖然編譯 PCMCIA 套件在技術上並不難，但還是需要您對 Linux 已有一定的熟悉的。

下列的東西必須在您開始安裝PCMCIA程式之前先行安裝妥當：

- 一個版本 2.0.* 或 2.1.* 系列的核心原始程式樹。
- 一套合適的模組公用程式。
- (選擇性) “XForms” X11 使用者界面工具組

目前的驅動程式套件事實上可在核心版本 1.2.8 開始的核心版本上正常工 作。而用舊一點的核心版本就背道而馳了些，最好不要這樣。

您還需要完整的 Linux 原始程式樹， 不是只有最新版的核心映像檔就可用 來編譯 PCMCIA 套件的。PCMCIA 模組含有一些對核心原始程式檔案的參考 指令， 也許您會想做個不含那些不必要的驅動程式的核心， 但是安裝 PCMCIA 並不需要您這麼做。

目前穩定的的核心程式的原始檔及修補檔可以在 <ftp://sunsite.unc.edu/pub/Linux/kernel/v2.0> 或 <ftp://tsx-11.mit.edu/pub/linux/sources/system/v2.0> 內找到。 發展中的核心程式可在 v2.1 目錄內找到，目前的模組公用 程式也可在同樣的地方找到。

在核心 2.0 及 2.1 版所用的 Linux 原始程式樹內的 Documentation/Changes 檔案中描述了核心程式所需要的各種類別的系統元件。您可檢視它並核對您 的系統是否爲最新版本，尤其當您已更新了核心程式。 如果您使用 2.1 版 核心，請確認您用了正確的程式庫和模組工具的組合。

當您建構核心時，如果您計畫要用 PCMCIA 網路卡，您應該打開網路支援並 關上正常的 Linux 網路卡驅動程式， 這包括口袋型可攜式型介面卡。 PCMCIA 網路卡驅動程式都設計成可載入式模組， 所以將驅動程式編譯到核 心程式內只會浪費空間而已。

如果您要使用 SLIP、PPP 或 PLIP，可在建立核心時將這些功能打開，或者 使用這些功能的可載入模組版本。 但是在核心 1.2.X 版本下建構作業時有 小缺撼，就是不能爲載入式模組設定建構選項 (例如 SLIP 壓縮)，所以如 果您需要這個功能，只好將 SLIP 連結到核心程心裏了。

爲了要使用 PCMCIA 拓樸鏈網路卡， 您的核心程式應該將 “Token Ring driver support” (CONFIG_TR) 這項目開啓，並將 CONFIG_IBMTR 這項關閉。

如果您要使用 PCMCIA IDE 卡，針對核心 1.3.72 至 2.1.7 版， 核心建構 時應該要將 CONFIG_BLK_DEV_IDE_PCMCIA 功能打開，舊版的核心並不 支援可移式 IDE 設備，而較新的核心就不需這麼特別設定了。

如果您要使用 PCMCIA SCSI 卡， 在建構核心時應將 CONFIG_SCSI 功 能打開。 並且，將您要使用的 SCSI 設備項目之驅動程式打開 (SCSI 磁碟 機、磁帶機、唯讀光碟機等等)。所有給特定介面卡的低層次驅動程式應該 要關掉，因爲它們只會佔用空間而已。

如果您想把給 PCMCIA 設備使用的驅動程式模組化，請修改 /etc/pcmcia/config 來指定哪些模組在哪種卡片使用下需被載入。例如，如果串列埠驅動程式被 模組化了，那麼串列埠設備的定義應該如下：

```
device "serial_cs"
  class "serial" module "misc/serial", "serial_cs"
```

在套件裏還有個 X 環境使用的卡片狀態公用程式 cardinfo。 此程式 是奠定在一個免費供應的使用者介面工具套件 XForms 程式庫。通常被附在 大部份的 Linux 供應商的產品內。 如果您想建造 cardinfo ，您得在 建構 PCMCIA 套件前安裝好 XForms 和所有的一般 X 程式頭檔及程式庫。

2.2 安裝

以下是安裝過程的簡要步驟：

- 解開 pcmcia-cs-3.0.?.tar.gz 到 /usr/src目錄內
- 在新的 pcmcia-cs-3.0.? 目錄內執行 “make config”
- 執行 “make all”，再執行 “make install”。
- 爲您的系統在 /etc/pcmcia 內自訂 PCMCIA 啓動手稿和選擇檔案。

如果您想安裝其他非主要 PCMCIA 分享套件程式所包含的客户端驅動程式， 將它們解開到 PCMCIA 原始程式樹的頂層，再依正常的建造指令，即可將這 增加的驅動程式編譯及安裝起來。

執行 “make config” 會列出一些建構選項，請檢查您的系統並比對 這些選項是否符合安裝 PCMCIA 支援的條件。請小心檢查此命令的輸出訊息，尤其哪邊有問題時。

如果您想編譯個 PCMCIA 套件給其他機器使用，在建構手稿執行到出現指定 目標目錄時請指定個替代的目錄，它應該是個絕對路徑，所有的 PCMCIA 工 具將被安裝在這目錄的相對位置裏。然後，您就可以 tar 這個目錄起 來，再把它拷貝到其他機器上，解壓縮到相對目錄上後再安裝它們到適當的地方。

如果您想編譯個 PCMCIA 套件給其他機器使用，在建構手稿執行到要求您指 定目標目錄時請指定個替代的目錄， 它應該是個絕對路徑，所有的 PCMCIA 工具將被安裝在這目錄的相對位置裏。然後，您就可以 tar 這個目錄 起來，再把它拷貝到其他機器上，解壓縮到相對目錄上後再安裝它們到適當的地方。

如果您在其他電腦上做交叉編譯，您可以指定替代檔名給編譯器及連結器。 這對混合著 a.out 及 ELF 系統時很有助益。此手稿會要求輸入做爲除錯時 使用的編譯旗標。

某些支援公用程式 (cardctl 和 cardinfo) 可以在安全或可信 的形式下編譯。在安全形式下可以防止非 root 的使用者更改卡片的建構。 可信的形式下允許一般使用者執行暫停、繼續、重置卡片的命令及改變目前 的建構。建構手稿會問你要將公用程式編譯成安全或可信的形式，內定爲安 全形式。

有些核心程式建構選項會影響到 PCMCIA 的工具。建構手稿（一般狀況下） 會在執行核心程式時推斷出這些影響。另外，如果您是編譯來安裝到其他機 器上的，它能從核心原始程式樹裏讀到這些建構，或是互動地設定每個選項。

執行 “make all” 後，接著 “make install” 將建立並安裝 核心程式模組及公用程式。 核心模組會被安裝在 /lib/modules/< 版本 >/pcmcia 內。 cardmgr 和 cardctl 兩程式被安 裝在 /sbin 目錄內。 如果 cardinfo 有被建立的話，它會 被安裝在 /usr/bin/X11 裏。

建構檔會被安裝在 /etc/pcmcia 目錄內。如果您安裝在一個舊版 本之上，您的舊版建構手稿會被備份起來不會被蓋掉，被保存下來的檔案名 稱會被加個 *.~1~、*.~2~ 之類的檔名。

如果您不知道您的 PCMCIA 控制晶片是什麼廠牌的， 您可以使用 cardmgr/ 目錄下的 probe 公用程式來探測它。目前最主要 的晶片有：Databook TCIC-2 晶片及 Intel i82365SL 相容晶片。

在少數情況下，probe 命令將無法自動地判斷您的控制器之類別。 如 果您有個 Halikan NBD 486 系統， 它有個放在不同地方的 TCIC-2 控制器： 此時您就需要修改 rc.pcmcia 來載入 tcic 模組， 並且將 PCIC_OPTS 參數設定成 “tcic_base=0x02c0”。

有些使用 Cirrus 控制器的系統， 包括 NEC Versa M 系統的 BIOS 會在系 統啓動時將控制器置於一個特別的懸置狀態下。在這類系統上，probe 命令並無法找到任何已知的 PCMCIA 控制器。 如果發生這種事， 請修改 rc.pcmcia 將 PCIC 設爲 i82365， PCIC_OPTS 爲 “wakeup=1”。

2.3 安裝完後（使用BSD 啓動手稿的系統）

有些 Linux 供應商，包括 Slackware，使用 BSD 的安排方式做爲系統啓動 手稿的方排。 如果有 /etc/rc.d/rc.M 這個檔的系統就屬這類型 的。安裝在 /etc/rc.d 內的 rc.pcmcia 手稿會控制啓動和 關閉 PCMCIA 系統。 在 “make install” 時會用 probe 命令 來決定您的控制器種類並適當修改 rc.pcmcia 的內容。您可在系統啓 動檔 /etc/rc.d/rc.M 內加上一行如下面的指令來執行 PCMCIA 啓動手稿。

```
/etc/rc.d/rc.pcmcia start
```

將這行插排在哪行前面都可以，只要 PCMCIA 驅動程式在 syslogd 後 啓動即可。

2.4 安裝完後（使用System V 啓動手稿的系統）

2.5 安裝完後（使用 System V 起始手稿的系統）

Red Hat、 Caldera 及 Debian 等供應商的 Linux 使用 System V 型的安排系統啓動 檔案。 如果有個 /etc/init.d 或 /etc/rc.d/init.d 目錄的系統就是這類型。 rc.pcmcia 手稿會被安裝到 /etc/rc.d/init.d/pcmcia 或是 /etc/init.d/pcmcia 下。它會自動啓動 PCMCIA 功能，不必去修改任何的啓動手稿了。

如果 /etc/sysconfig 目錄在系統內， 那麼有另一個建構檔 /etc/sysconfig/pcmcia 會被建立起來做爲啓動選項。 如果您需 要更改任何的模組選項 (如 PCIC= 或 PCIC_OPTS= 設定)， 只 需編輯這個建構檔而不必修改眞正的 PCMCIA 啓動手稿。此檔案也不會被往 後的安裝而覆蓋掉。

有些以前的版本使用 /etc/sysconfig/pcmcia-scripts 目錄而不 是現在的 /etc/pcmcia 目錄。現在所有的系統的版本都已或將改 用 /etc/pcmcia 目錄了，。

2.6 插槽驅動程式選項

有些 PCMCIA 控制器擁有其他的功能可能會也可能不會被特別的系統所採用。而插槽驅動程式也沒有辦法偵測到是否這些功能被採用與否。請檢查您的 插槽驅動程式的 man 說明頁看看您的控制器有哪些其他的功能。

低階的插槽驅動程式 tcic 及 i82365 有許多匯流排分時參數特 別是在採用高速處理器的系統上時需做些修改。分時發生問題的症狀像卡片 辨認問題、在重負載下的閉鎖、錯誤比率高，或是造成設備性能變差。詳情 請參閱相關的 man 頁會有更多的細節，在此簡述如下：

- Cirrus 控制器有許多可建構的分時參數。 最重要的應該是 cmd_time 旗標， 它用來決定 PCMCIA 匯流排週期的長度。 速度快的 486 系統（如 DX4-100 ）下將它從 6 (內定值) 加大成爲 12 或 16 時可看到它效果。
- Cirrus PD6729 PCI 控制器有 fast_pci 旗標，如果 PCI 匯流排的速 度比 25 MHz 快時應該設定它。
- Vadem VG-468 控制器及 Databook TCIC-2 控制器內的 async_clock 旗標必須更改爲與 PCMCIA 匯流排和主匯流排相對的週期速度。更改這個旗 標會使某些操作上增加等待的時間。然而，我好像還沒聽說有哪個筆記型電 腦需要這麼做的。

- pcmcia_core 模組有個 cis_speed 參數， 它是用來更改取存卡 片的卡片資訊結構 Card Information Structure (CIS) 之記憶體速度。有 些速度快的匯流排時序的系統上，增加這個參數（例如，減慢卡片的存取） 可以解決卡片辨識的問題。
- 這裏我們並非討論分時的議題，但如果您的系統上有一個以上的 ISA-to-PCMCIA 控制器或在擴充塢上有其他的插槽時，i82365 模組應 該和參數設爲 1 的 extra_sockets 一起被載入， 但對於 PCI-to-PCMCIA 或 PCI-to-CardBus 橋接器者就不必這麼做。

所有的這些選項應修改 /etc/rc.d/rc.pcmcia來建構，例如：

```
# Should be either i82365 or tcic
PCIC=i82365
# Put socket driver timing parameters here
PCIC_OPTS="cmd_time=12"
# Put pcmcia_core options here
CORE_OPTS="cis_speed=500"
```

這裏有一些特殊系統的分時設定：

- 在 ARM Pentium-90 或 Midwest Micro Soundbook Plus 上， 使用 “freq_bypass=1 cmd_time=8”。
- 在 Midwest Micro Soundbook Elite，使用 “cmd_time=12”。
- 在 Gateway Liberty 上，試試 “cmd_time=16”。

2.7 系統資源設定

卡片服務系統應該自動地去避免分配到已被其他標準設備使用了的 IO 埠和插斷。它也會試著偵測本身與不知名設備間的衝突，只不過還不是很完 全可靠。 如果遇到這種情形， 您需要將 /etc/pcmcia/config.opts 內的資源給排除掉。

以下是某些特定的筆記型電腦之資源設定：

- 在 AMS SoundPro, 排除 irq 10.
- 在有些 AMS TravelPro 5300 models, use memory 0xc8000-0xcffff.
- 在 BMX 486DX2-66, 排除 irq 5, irq 9.
- 在 Chicony NB5, use memory 0xda000-0xdffff.
- 在 Compaq Presario 1020, 排除 port 0x2f8-0x2ff, irq 3, irq 5.
- 在 HP Omnibook 4000C, 排除 port 0x300-0x30f.
- 在 Micron Millenia Transport, 排除 irq 5, irq 9.
- 在 NEC Versa M, 排除 irq 9, port 0x2e0-2ff.
- 在 NEC Versa P/75, 排除 irq 5, irq 9.
- 在 NEC Versa S, 排除 irq 9, irq 12.

- 在 NEC Versa 6000 系列, 排除 port 0x300-0x33f, irq 9, irq 10.
- 在 ProStar 9200, Altima Virage, 及 Acquiline Hurricane DX4-100, 排 除 irq 5, port 0x330-0x35f. 可以使用 memory 0xd8000-0xdffff.
- 在 Siemens Nixdorf SIMATIC PG 720C, 使用memory 0xc0000-0xcffff, port 0x300-0x3bf.
- 在 TI TravelMate 5000,使用 memory 0xd4000-0xdffff.
- 在 Toshiba T4900 CT, 排除 irq 5, port 0x2e0-0x2e8, port 0x330-0x338.
- 在 Twinhead 5100, HP 4000, Sharp PC-8700 和 PC-8900, 排除 irq 9 (音效), irq 12.
- 在 MPC 800 系列，排除 irq 5, port 0x300-0x30f 是給 CD-ROM 用的。

3 解決安裝與建構的問題

在本章節裏會指出一些常見的 PCMCIA 子系統的失敗模式。請您試著在 這些例子中找出您所遇到的問題之症狀。本章只描述”一般的錯誤”問題， 因此並不針對特定的卡片或驅動程式。

想要除錯在我們試著經由 PCMCIA 裝置來安裝 Linux 時遇到的 PCMCIA 驅 動程式問題幾近乎不可能。甚至您能從症狀中知道是哪方面的問題，想要修 改安裝磁片又很難，尤其是無法在 Linux 系統下存取時。 要自訂安裝磁片 完全要仰賴 Linux 供應者的的選擇了，這也不在本文件的範圍內。 但一般 來說， 最佳的步驟是先使用其他的方法來安裝好 Linux， 然後拿到最新的 PCMCIA 驅動程式後，再來除錯那些仍存在的問題。

3.1 基本 PCMCIA 核心模組並沒載入

症狀:

- ”核心版本不符合”之錯誤訊息在 PCMCIA 啓動手稿執行時出現。
- 在啓動後， `lsmod` 並沒秀出任何的 PCMCIA 模組。
- `cardmgr` 執行報告 “no pcmcia driver in /proc/devices” 在系統日誌中。

核心模組中包括它的版本資訊會在模組被載入時與現在的核心相核對。檢查 的方式視 `CONFIG_MODVERSIONS` 這項核心選項來看。 如果這項目是否 定的， 核心版本號碼就會被編譯到每一個模組內，而 `insmod` 會檢查 這項是否與執行中的核心是相符合的。 如果 `CONFIG_MODVERSIONS` 是 yes，核心所提報的每個符號會被做成一份檢查總覽 (Checksum)。這些程式 碼都會被與相對應的程式碼相比對後編譯成模組。這麼做旨在讓模組們減少 版本依賴度， 因爲檢查總覽只會在核心介面更動時才會跟著變動， 且對於小小的核心更新升級幾乎維持與原來相同。在實務上，檢查總覽已變成更加 的嚴格，因爲有許多的核心介面都依賴是在編譯時期時核心選項的設定。而 且，檢查總覽已變成一個判斷相容度的極端悲觀的工具了。

有些 PCMCIA 模組需要核心服務程式，但這些服務程式可能存在或不存在， 這完全要看核心的建構。 例如，SCSI 控制卡驅動程式就需要核心已被建構 支援了 SCSI 了。網路驅動程式就需要支援網路的核心。如果核心缺少了一 需要的功能，`insmod` 可能會報告出有未定義的符號而不去載入該模組。

這樣繼續的結果是，核心模組緊密地與核心版本以及許多的核心建構選項的 設定相結合。一般來說，結核心 2.0.31 版的一組被編譯好的模組並無法被 其他的核心 2.0.31 版本上使用。除非有特別地注意到將兩個建構成相同的 設定。這個問題，就讓那些供應已編譯好的核心模組的工作變得有點奇怪了。

您有幾種選項：

- 如果您擁有的是 Linux 供應版內之未經編譯的驅動程式， 請檢查您所使用 的核心是和該供應版一起的未經編譯的核心。如果您想使用未經編譯的模組，一般來說你得使用與它想伴的核心。
- 如果你重新建構或升級你的核心了，你可能需要編譯和安裝新的 PCMCIA 套 件。 如果你已經有安裝了核心原始樹的話，做這件事就得容易了。 請參考 PCMCIA-HOWTO 有更詳細的指示。
- 在某些情形下，與其他系統元件的不相容可能會導致無法正確載入核心模組 元件。 如果您自己升級核心， 請注意詳列模組原始檔案樹內之 `Documentation/Changes` 檔案內針對模組公用程式及二進位公具 程式中列明的最小需求 (“minimal requirements”)。

3.2 插斷掃描失敗

症狀:

- 當 PCMCIA 驅動程式被載入時系統卻動也不動，就算並沒有卡片插著時也一 樣。
- 系統日誌在系統當機鎖死前顯示成功地偵測到 PCMCIA 控制器，但還沒顯示 插斷偵測的結果時。

在辨視 PCMCIA 控制器之後，插槽驅動程式會偵測空著的插斷號碼。這個動 作會爲每個顯然是空著的插斷做程式化, 然後產生一個“ 軟的 ”插斷， 來看看是否這個插斷可以被正確地被偵測到。有些時候，偵測到一些特殊的 插斷時會影響到其他的系統設備。

這麼偵測的理由是，我們要辨視出眞正空著可用的插斷。 (例如，那些不是 被任何其他 Linux 設備驅動程式所預留著的， 也並非實體上已連接著 PCMCIA 控制器的，或是已連接著其他的設備但並沒有驅動程式的。)

有二種繼續的方法:

- 插斷探測工作可以使用插槽驅動程式內的 `irq_list` 參數設定來限制 只對某些插槽實施而已。例如“`irq_list=5, 9, 10`” 會限制只對這 三個插斷做掃描探測而已。所有的 PCMCIA 設備會被限制只能使用這幾個插 斷而已 (假如它們略過了偵測動作)。你可能需要嚐試幾次失敗並再接再厲地才能找到哪些插斷可以被安全地偵測使用的。
- 插斷探測工作可以被完全地關閉掉，在載入插槽驅動程式時使用了“do_scan=0” 選項。這麼做，會讓原定的插斷清單被使用著，它們已經避免使用那些已經 被其他設備所佔用了的插斷。

另一個方法，我們可以使用在 PCMCIA 啓動手稿中指定 `PCIC_OPTS` 的 設定，例如:

```
PCIC_OPTS="irq_list=5,9,10"
```

3.3 記憶體偵測失敗

症狀:

- 主驅動程式在卡片並不存著時被正確地載入，而且在系統日誌內也沒有任何 錯誤。
- 系統當機動不了和或任何卡片插入但在任何嗶聲響起前就重新開機。

或是：

- 任何卡片插入時會產生一個高音的嗶聲，接著低沈的嗶聲。
- 任何卡片都被誤認 “anonymous memory cards”。
- 系統日誌報告說有很多的記憶體範圍已被排除在外了。

主模組程式在第一次插入卡片使做一定記憶體掃描。這個動作有潛在可能地 干涉到其他記憶體映射的設備。另外，pre-3.0.0 版本前的驅動程式套件還 會做比現今的驅動程式版本更進一步的掃描。記憶體窗是被定義在 `/etc/pcmcia/config.opts` 內。 預設的窗口很大，所以它可能會 幫助來限制掃描到較窄的範圍。比較合理的範圍可試看看包含進以下的位址：0xd0000-0xdffff, 0xc0000-0xcffff, 0xc8000-0xcffff, 或 0xd8000-0xdffff。

如果你有 DOS 或 Windows 版的 PCMCIA 驅動程式， 你就可以 you may be able to deduce what memory region those drivers use. 請記得 DOS 的 記憶體位址通常都使用“段”位址形式，也就是它會將尾巴的十六位元 數字省略掉（所以 0xd0000 的絕對位址就是 0xd000 ）。 記得在改 `/etc/pcmcia/config.opts` 時要確認這項。

3.4 錯誤地偵測卡片的插入與抽出

症狀：

- 在開機使卡片有插著並被偵測到且正確地被建構了。
- 驅動程式不會反應出卡版被插入或移出，或是記錄在系統日誌、或時嗶聲響。

一般來說，卡槽驅動程式 (`i82365` 或 `tcic`) 會自動地偵測並選 擇一個適合的插斷來傳送卡片狀態的更動。 某些 Intel 相容控制器的自動 插斷偵測不能工作。 包含 Cirrus 晶片和裝在 IBM ThinkPads 上的晶片。 如果在偵測時設備無法起動，它的插斷也會是閒置的。這種狀態下，卡槽驅 動程式也許會挑到一個已被其他裝置使用中的插斷來使用。

在 `i82365` 和 `tcic` 的驅動程式裏的 `irq_list` 選項可以 用來限制哪些插斷可以被測試的。這個插斷列表可被限制成只被 PCMCIA 卡 所使用或用來監控卡片狀態的改變。 另外 `cs_irq` 選項可明白地設定哪個插斷要被用來監控卡片狀態的改變的。

如果您無法找到可正常工作的插斷號碼，還有一個票選狀態模式可用：不論 是 `i82365` 或 `tcic` 都接受 `poll_interval=100` 這選項， 用來票選卡片的每秒的改變狀態。如果您的系統已短缺可被 PCMCIA 卡使用 的插斷時這個選項也可以被使用。特別是在系統內有一種以上的 PCMCIA 控 制器時就必須注意這點了。

所有的這些選項必須在 `PCIC_OPTS=` 這行來設定， 看您的系統是設在 `/etc/rc.d/rc.pcmcia` 裏或是 `/etc/sysconfig/pcmcia`。

3.5 兩張卡之間的資源相衝突

Symptoms:

- 兩張卡片在各自獨自使用時可以工作，
- 但當兩張卡一起被插著時，卻只有一個可以正常工作。

通常這就表示已經和某個 Linux 不知道的系統設備相衝突了。PCMCIA 設備 是被動態建構的，所以，例如，插斷是在被需要時被分配的，而不是特別被 指定到特別的卡片或是插槽的。現在有一個可用資源的清單，卡片會在他們 被建構時依序地被指派給資源的。在這種狀況下，最後被建構的卡片會被指派到一個並非是空閒著的資源上了。

您可檢查系統日誌有哪些資源被非正在工作的卡片所佔用著。在 /etc/pcmcia/config.opts 裏把這些排除在外， 再重新啓動 cardmgr 精靈來再載入資源資料庫。

3.6 設備建構並沒有完成

症狀：

- 當一個卡片被插入時，確實可聽到一個高音的嗶聲響。
- 接下來的卡片不管是插入或移出都不被理睬。

這表示卡片已被成功地辨視了。但是 cardmgr 因某些原因已無法完成 建構程序。最有可能的原因是在卡片設定手稿的某一步驟被困住了。當一個 網路卡被插入時並沒有接上一個正活動中的網路上時，網路手稿被困住了， 這就是最好的例子。

要找出問題出在哪裏，你可以手動執行一個設定手稿來看看它是被困在哪兒 的。這個手稿就放在 /etc/pcmcia 目錄內。他們會使用二個參數：設備名稱及動件。 cardmgr 精會把記錄建構的命令記錄在系統日誌 內。 例如， 在系統日誌中顯示出 ‘./network 命令開始了 eth0” 是被 cardmgr 最後一個執行的命令，以下的命令會追蹤這個手稿：

```
cd /etc/pcmcia
sh -x ./network start eth0
```

4 使用方法以及特色

4.1 用來建構以及監控 PCMCIA 設備的工具

如果所有的模組都被正確地載入了，在沒有卡片被插入的狀態下， lsmod 命令的輸出會看起來像以列的樣子。

```
Module                  Size  Used by
ds                      5640   2
i82365                 15452   2
pcmcia_core            30012   3  [ds i82365]
```

系統日誌也應該包含插槽驅動程式對於哪些主控制器被發現了、有幾個插槽 被偵測到了的輸出。

4.1.1 cardmgr建構精靈

cardmgr 精靈負責監控 PCMCIA 插槽，有需要時載入 其他的驅動程式，並且執行使用者階層的手稿來反應卡片的插入與移出。它 會把它的動作記錄在系統日誌內並用嗶聲來回應卡片狀態的改變。嗶聲的音 調表示建構步驟是成功還是失敗。兩個高音聲響表示步片已被成功地辨視與 建構了。一高音一低音聲響表示卡片被辨視，但因某些原因無法被建構。一 低音嗶聲表示卡片無法被辨視。

Cardmgr 會把每一個插槽的設備資訊記錄在 /var/run/stab 裏。以下是 /var/run/stab 的列表例子：

```
Socket 0: Adaptec APA-1460 SlimSCSI
0       scsi    aha152x_cs      0       sda     8       0
0       scsi    aha152x_cs      1       scd0    11      0
Socket 1: Serial or Modem Card
1       serial  serial_cs       0       ttyS1   5       65
```

在這幾行所描述的設備中，第一個欄位表示插槽、第二爲設備類別、第三爲 驅動程式名稱、第四被用來指定在同一驅動程式下使用的每個設備取號碼、 第五個是設備名稱、最後兩個欄位是這個設備的主要及次要號碼。（如果可 找到的話）

cardmgr 精靈依據被存在 /etc/pcmcia/config 內的已知之 卡片資料庫來建構這些卡片。這個檔案描述了許多個別的驅動程式，以及如 何辨視許多的卡片，還有哪個驅動程式屬於哪個卡片的。這個檔案的格式在 pcmcia(5) 的 man 頁裏有詳細的資料。

4.1.2 cardctl 及 cardinfo 公用程式

cardctl 命令可以用來檢查插槽的狀態， 或看它們是如何被建構的。 它也可以被用來警示一個卡片的建構狀態。 這兒有個 “cardctl config” 命令的輸出例子：

```
Socket 0:
Socket 1:
  Vcc = 5.0, Vpp1 = 0.0, Vpp2 = 0.0
  Card type is memory and I/O
  IRQ 3 is dynamic shared, level mode, enabled
  Speaker output is enabled
  Function 0:
    Config register base = 0x0800
      Option = 0x63, status = 0x08
    I/O window 1: 0x0280 to 0x02bf, auto sized
    I/O window 2: 0x02f8 to 0x02ff, 8 bit
```

“cardctl suspend” 和 “cardctl resume” 兩個命令可以用 來在不需卸下它的相關驅動程式的狀況下關閉卡片。“cardctl reset” 命令會重新設定及建構卡片。 “cardctl insert” 和 “cardctl eject” 摸擬當一個實體的卡片插入或抽出時所做的動作，這包含載入或卸下驅動程式以及建構或關閉設備。

如果您正執行 X， cardinfo 公用程式會產生一個圖 示來顯示所有的 PCMCIA 插槽的現況， 內容和執行 “cardctl config” 類似。它還提供其他 cardctl 功能的圖形化功能。

4.1.3 插入與抽出卡片

在理論上，你可以在任何時間插入或移出 PCMCIA 卡片。然而，把正在被其 他程式使用中的卡片移出來並不是個好主意。在核心 1.1.77 版以前的系統 常會在串列數據機卡片被移出時造成當機，但現在已有修正方法了。

4.1.4 卡片服務程式與進階能源管理

卡片服務程式可以編譯成支援 APM (進階能源管理)， 如果你已安裝了這個套件到你的系統內的話。 APM 在核心 1.3.46 （含）以後就被整合了。 現在這套件被 Rick Faith (faith@cs.unc.edu) 所維護，

APM 工具可以由 <ftp://ftp.cs.unc.edu/pub/users/faith/linux> 拿到。如果你的系 統被偵測出有個相容的版本在的話， PCMCIA 模組會被自動地建構成支援 APMM 功能。

如果沒有藉著 APM， 你可以在懸置你的筆記電腦前執行 “cardctl suspend”，以及在回覆後執行 “cardctl resume”，才能適當地關 上或重啓你的 PCMCIA 卡。 這對正在使用中的數據機並不適用，田因序 列驅動程式並無法儲存與回應數據機的運作參數。

APM 在有些系統上似乎有點不太穩定。 如果你遇到 APM 和 PCMCIA 共用時 有任何問題，請您在提出程式小虫報告前先試著把問題簡化。

有些驅動程式， 比如說 PCMCIA SCSI 驅動程式，無法從懸置重新動作週 期下回覆過來。 在使用 PCMCIA SCSI 卡時， 在懸置系統前先執行 “cardctl eject”。

4.1.5 關閉 PCMCIA 系統

要卸下整個的 PCMCIA 套件， invoke rc.pcmcia with:

```
/etc/rc.d/rc.pcmcia stop
```

這個手稿會花幾秒鐘來執行，給全部的客户自訂驅動程式來完美地關閉。如果有個 PCMCIA 設備正被使用中，這樣關可能不會很完整地關閉，有些 核心模組可能不會被卸下。 要防止發生這種狀況， 請在執行 rc.pcmcia 之前使用 “cardctl eject” 來關閉所有的插槽。 cardctl 命令的離開狀態會顯示如果有任何的插槽不能被關閉的。

4.2 PCMCIA建構手稿的概要

每一個 PCMCIA 設備都會有個相對應的 “ 類別 ” 來描述它應該如何被建 構與管理的。這些類別與設備驅動程式都被組織在 /etc/pcmcia/config 內。目前共有五種輸出輸入設備的類別 (網 路、 SCSI、cdrom、硬碟以及串列) 以及二種記憶體設備類別 (記憶體與 FTL)。在每一個類別中，都有二個手稿放在 /etc/pcmcia/config 內：一個主要建構手稿 (例如， /etc/pcmcia/scsi 是 SCSI 設 備的)，以及一個選項手稿 (例如， /etc/pcmcia/scsi.opts)。 主手稿在卡片被插入時會被啓用來建構該設備，當卡片被抽出時關掉該設備。對於支援多種設備的卡片，針對各別不同的設備的手稿都會被啓用。

這 config 手稿會從 /var/run/stab 裏解出一些有關於設備上的 資料開始。 每一個手稿會構築一個 “ 設備位址 ”，它是獨一無二的位址 用來描述該設備在 ADDRESS shell 變數內被要求被建構的方式。它會被傳遞到 *.opts 手稿上， 該手稿再回應有關於在這位址上的設備應 該要怎樣地建構才可以的資訊。有些設備， 它的設備位址就是插槽號碼。 還有其他的設備，它會包含另外的有用資訊可以用來決定如何來建構該設備。例如，網路設備會傳遞他們的硬碟乙太網路位址做爲設備位址的一部份， 來讓 network.opts 手稿能夠使用這項資訊來從許多不同的建構中做 正確的選擇。

所有的設備位址的首要部份是目前的 PCMCIA “scheme”。這個參數是被用 在只有一個外部使用者指定變數時用來對設備建構之多種設定的。對於它的 應用， 例如我們可以有個 “home” scheme 以及一個 “work” scheme， 它們包括了不同的網路建構參數的組合。 目前的 scheme 可以使用 “cardctl scheme” 命令來做選擇。如果沒有 scheme 被指定的話就 以內定值來設定。

一般原則上，當要建構 Linux 在筆記型電腦上時，PCMCIA 的設備只能被經 由 PCMCIA 設備手稿來建構。 請不以試著以建構固定設備的方式來建構 PCMCIA 設備。然而，有些 Linux 供應版商提供已接在他們自已的設備建構 工具上的 PCMCIA 套件。在這種情況下，以下的幾個主題可能無法使用，而 供應版商應該會有他們自己的這類文件。

4.3 PCMCIA 網路卡

Linux 的乙太型網路卡介面通常被稱做 eth0、eth1 等等。環形 卡亦同樣地被看待。 然而它們被稱做 tr0、 tr1 等等。 ifconfig 命令是被用來觀看及修改網路介面狀態的。Linux 的另一特 點是網路介面並不會像其他設備一樣地被看成是一個在 /dev 裏 頭的檔案。所以如果您在 /dev 內找不到它們時請不要覺得驚訝。

當一個 PCMCIA 乙太網路卡被偵測到時，它會被指定爲第一個可用的介面卡 名字， 通常爲 eth0。Cardmgr 命令會執行 /etc/pcmcia/network 手稿來建構它的介面。

請勿將您的 PCMCIA 乙太網路卡建構在 /etc/rc.d/rc.inet1 內，因爲當這個手稿被執行時也許卡片並不在插槽裏。 在 rc.inet1 內 除了回歸的項目之外， 其它的命令列請將他們註解掉。 而另外修改 /etc/pcmcia/network.opts 來配合您的本地網路設定。 network 和 network.opts 手稿只有在您的乙太網路卡存在時才 會被執行。 如果您的系統有自動網路建構程式， 它也許是也許不是 PCMCIA-aware 。 請從您的 Linux 供應版本的文件中查看看是否 PCMCIA 網 路設備是否需要使用自動化工具來建構或只需編輯 network.opts 即 可。

傳到 network.opts 的設備位址中包含了四個以逗號來分開的欄位： scheme、插槽號碼、設備提議及卡片硬體的乙太網路位址。設備提議是用來 給有許多網路介面的卡片取號碼之用，通常爲 0。如果您有許多不同使用目 地的卡片時，另一按照插槽位置而建構的選項應該被設定，如同在：

```
case "$ADDRESS" in
*,0,*,*)
    # definitions for network card in socket 0
    ;;
*,1,*,*)
    # definitions for network card in socket 1
    ;;
esac
```

或是，它們可能被以自己的硬體位址被建構，如下：

```
case "$ADDRESS" in
*,*,*,00:80:C8:76:00:B1)
    # definitions for a D-Link card
    ;;
*,*,*,08:00:5A:44:80:01)
    # definitions for an IBM card
esac
```

4.3.1 網路設備參數

以下的參數可在 network.opts內被使用：

IF_PORT

指定乙太網路收發器類型， 這是針對那些無法自動偵測的卡使用。 請看 “man ifport” 內有收發器名稱。

BOOTP

布林 (y/n) 設定： 表示是否主機的 IP 位址和分路資訊是否可以經由 BOOTP 協定來獲取。

IPADDR

定IP位址。

NETMASK, BROADCAST, NETWORK

基本網路參數： 請參閱 networking HOWTO

GATEWAY

給主機的子網路使用的通訊閘 IP 位址。只要是在這個子網路以外的封包都 會被轉到這個閘道。

DOMAIN

給主機的網路 domain name，被用來建立 /etc/resolv.conf。

DNS_1, DNS_2, DNS_3

主機名稱或是 IP 位址給這個介面的 nameservers， 被加到 /etc/resolv.conf內。

MOUNTS

A 給這個介面使用來做 NFS 掛入點的列表。

IPX_FRAME, IPX_NETNUM

對 IPX 網路: frame type 及網路號碼將被傳輸到 ipx_interface 命令上。

例如：

```
case "$ADDRESS" in
*,*,*,*)
    IF_PORT="10base2"
    BOOTP="n"
    IPADDR="10.0.0.1"
    NETMASK="255.255.255.0"
    NETWORK="10.0.0.0"
    BROADCAST="10.0.0.255"
    GATEWAY="10.0.0.1"
    DOMAIN="domain.org"
    DNS_1="dns1.domain.org"
    ;;
esac
```

若想自動掛上或卸下網路檔案系統，首先將檔案系統加到 /etc/fstab 內，在掛上選項上加上 noauto。在 network.opts 內 使用 MOUNTS 變數列明檔案系統的掛入點。 這特別是在使用 cardctl 或 cardinfo 命令來關閉當網路檔案系統是用這個方法 建構而掛上的網路卡時。如果在沒預警下網路卡被移出時，是無法完全地卸 下網路檔案系統的。

除了一般的網路建構參數之外，network.opts 手稿可用來在介面已經 被建構起後或被關閉後做另外的動作時使用。 如果 network.opts 定 義了一個叫 start_fn 的命令層功能呼叫， 當介面被建構後它會被網 路手稿所啓用，而介面名稱將會被傳送到這個功能函數做爲第一個參數（也 只有這個）。相同地，如果 stop_fn 被定義了，那麼它會在關閉一個 介面前被啓用。

收發器類型可以用 IF_PORT 來設定。它可以是 PCMCIA 之前版本的數 值表示或是識別收發器的關鍵字。所有的網路驅動程式儘可能將它設定爲自 動偵測（如果可能的話）介面或是 10baseT 的。 ifport 命令可用來 檢查或設定目前的收發器類型，例如：

```
# ifport eth0 10base2
#
# ifport eth0
eth0    2 (10base2)
```

目前的 3c589 驅動程式版本會自動偵測網路的連結， 但還無法完全發揮它 的功能。若想以自動偵測來工作，在網路卡被建構前網路線應該是已連接著 網路卡上的。要不然，等網路接好後，您可以下列命令迫使驅動程式檢查連 結：

```
ifconfig eth0 down up
```

4.3.2 特定卡片的評語

- 使用 IBM CCAE 和 Socket EA 卡在網路設備被建構時， 您需挑個接收器類 型 (10base2、10baseT 、AUI)，並確認在您的系統日誌上所記錄的接收器與 您所連結的相同。
- 給 SMC、Megahertz、 Ositech 及 3Com 卡片的驅動程式應該會自動偵測連 接著的網路類型 (10base2 或 10baseT)。 當驅動程式被載入時會設 定收發器類型來訂定義卡片的"第一次猜想"。
- Farallon EtherWave 事實上是和 3Com 3c589 相同的， 但有個特殊的收發 器。 雖然 EtherWave 使用 10baseT 類的連接方式， 但它的收發器需要把 3c589 建構成 10base2 模式。
- 如果您使用 IBM CCAE、NE4100、 Thomas Conrad 或 Kingston 介面卡時遇 到問題， 請試著在 pcnet_cs 模組內的 mem_speed=# 選項 增加記憶體存取時間。在標準的 config.opts 檔案內有個教你如何做 的範例。請試看看速度在 1000 ns(奈秒)以上。
- New Media 乙太網路卡在接到某些系統上時，可能需要在 pcmcia_core 模組被載入時， 以 io_speed=# 選項來增加 IO 埠的存取時間。 在啓動手稿的 CORE_OPTS 選項可以設定它。
- New Media 乙太網路卡驅動程式的多工傳播支援並不完整。最新的驅動程式 版本可與多工傳播核心一起使用，但會略過多工傳播封包。不規則模式應該 可以工作正常了。
- 給 IBM 及 3Com token ring 卡的驅動程式在卡片尚未接上一個 ring 上時 就做初始化是會有點問題。所以請一定記得在開機前一定要把這些卡片和 網路連接在一起。 這個驅動程式也需要空置的 IO 埠在 0xa20-0xa27 範圍 內。在有些系統上，自動 IO 埠衝突檢查器會錯誤地判定說這個埠範圍並不 是空置的。 遇到這種情況， 我們可以把埠檢查器給關閉掉。 只要在 載入 pcmcia_core 模組時設定 probe_io=0即可。
- Newer Linksys 及 D-Link 的卡片有個特殊的選擇收發器方式，目前還不被 Linux 的驅動程式所能控制。 只有一個變通的方法是以 DOS 開機並使用廠 商所付的公用程式來選擇收發器後，再暖開機進到 Linux 裏。 我現在也正 在尋找首次的測試者使用 Linux 的公用程式來做到這些功能。
- 對 WaveLAN 無線網路卡來說，Jean Tourrilhes (jt@hplb.hpl.hp.com) 寫了一篇無線的 (設備) wireless HOWTO 放在 <http://www-uk.hpl.hp.com/people/jt/Linux/Wavelan.html> 內。

4.3.3 診斷網路卡的問題

- 您的卡片有被辨視成乙太網路卡嗎? 請查看系統日誌並確認 cardmgr 已正確地辨視了這張卡，而且已啓動了一種網路驅動程式了。如果沒有，您 的卡片也可能還是可以使用，只要它與任一種被支援的卡片相容的話。只要 是這卡片是與 NE2000 相容的卡片都可以做到。

- 卡片有妥當地被建構了嗎? 如果你使用的是被支援的卡片， 而且它已被 cardmgr所辨視了，但它還是無法工作，這可能是因爲它與其他的設備 發生插斷或埠的衝突了。請查明這張卡片使用了嗎些資源 (從系統日誌裏找)， 然後試著在 /etc/pcmcia/config.opts 內把這些給排除掉來 迫使卡片使用其他的資源。
- 如果你的卡片似乎已妥當地建構了，但有時候還是會鎖死不能動，尤其是在 高負載時，你可能需要試著改變你的插槽驅動程式的分時參數。詳細資料請 看 2.6 (2.3) 這章。
- 當你在存取網路使遇到像 “network unreachable” 這類的訊息， 可能是 你在 /etc/pcmcia/network.opts 有設定錯誤的地方。 另外，錯 誤地建構的卡片通常會安靜地發生問題。
- 要診查在 /etc/pcmcia/network.opts 內的錯誤， 開始請先試著 以它們的 IP 位址來 ping 同一次網路的其他系統。然後試著 ping 你的閘 道，然後其他次網路內機器。在試完這些較簡單的測試後，再以機器的名字 來 Ping。
- 請要確認你的問題眞的個 PCMCIA 的問題。 你可以在 DOS 下以廠商供應 的驅動程式來驅動看看是否可以工作，這樣將有助於判斷。並再次複檢你對 /etc/pcmcia/network.opts 手稿內的任何修改。並確認你的纜線、“T” 接頭、終端電阻器等等的元件都可正常工作。

4.4 PCMCIA 串列與數據機設備

Linux 的串列設備都是被經由 /dev/cua* 和 /dev/ttyS* 特殊設備檔案來做取用。 ttyS* 的設備被使用在進來的連接，例如直 接地連接終端機。cua* 的設備被使用在往外的連接，比如說數據機。 而每一個實體串列埠都各有 ttyS 和 cua 兩個設備檔： 要使用 哪個適當的設備到你的應用上都由您來決定。

對於串列設備的建構，您可使用 setserial 命令來檢查與修改。

當一個 PCMCIA 串列卡或數據卡被偵測到時，它會被指定成爲第一個可用的 串列設備。通常是 /dev/ttyS1 (cua1) 或 /dev/ttyS2 (cua2)，這完成看原已內建的串列埠數目。 ttyS* 設備會被 報告在 /var/run/stab 內。 原內定的串列設備選項手稿 /etc/pcmcia/serial.opts 會將之連結到相對應的 cua* 設 備檔案成爲 /dev/modem。這樣就很便利了。

請勿使用 /etc/rc.d/rc.serial 來建構 PCMCIA 數據機。這個手 稿應該只能被用來建構那些非可移動的設備。如果你想要設定你的數據機來 做任何特別的動作時，請修改 /etc/pcmcia/serial.opts 這個手 稿。 另外， 也請勿使用 setserial 來改變 PCMCIA 串列設備的設備 IO 埠和插斷設定。 這樣做只會告訴串列驅動程式到不對的地方尋找串列設 備，但不會改變卡片硬體方面眞正被建構的事實。串列建構手稿允許您指定 其他的 setserial 選項， 以及是否您該在 /etc/inittab 內爲這個埠加一行設定。

快傳送到 serial.opts 的設備位址共有三個由逗點所分開的欄位：第 一是 scheme、 第二是插槽號碼以及第是 device instance。 The device instance 會爲支援多串列埠的卡選用多個數據，但對於單一埠卡來說， 它 永遠都是零。 如果你常使用不只一個的 PCMCIAl 數據機，你可能需要針對 各別插槽內的數據機做不同的設定。如下：

```
case "$ADDRESS" in
*,0,*)
    # Options for modem in socket 0
    LINK=/dev/modem0
    ;;
*,1,*)
```

```
    # Options for modem in socket 1
    LINK=/dev/modem1
    ;;
esac
```

如果 PCMCIA 數據卡在 Linux 啓動時即被建構好了， 那它可能已被錯誤地 被辨識成爲一般的內建串列埠了。這雖然無害處，但是，當 PCMCIA 驅動程 式接手來控制數據機時， 它會被指派成不一樣的設備插孔。 最好是 parse /var/run/stab 或使用 /dev/modem， 而不是期望 PCMCIA 數據機總是被指派成相同的設備。

如果你建構你的核心來載入基本的 Linux 串列埠驅動程式成爲一個模組， 你必須編修 /etc/pcmcia/config 來指出該模組必須被載入。 編 修串列設備那列如下：

```
device "serial_cs"
  class "serial" module "misc/serial", "serial_cs"
```

4.4.1 串列裝置參數

以下的參數可以在 serial.opts內被定義：

LINK

爲符號連結 (symbolic link)指定個要被建立的路徑 “dialout” 或是 /dev/cua* 設備。

SERIAL_OPTS

指定有哪些選項將被傳送到 setserial 命令。

INITTAB

如果有指定的話，它將被用來在 inittab 中爲該設備建立一個條目。

例如：

```
case "$ADDRESS" in
*,*,*,*)
    LINK="/dev/modem"
    SERIAL_OPTS=""
    INITTAB="/sbin/getty"
```

4.4.2 診斷串列設備的問題

- 您的卡被認爲是個數據機嗎? 請檢查系統日誌並確認 cardmgr 有正確 地辨識了這張卡，並且啓動了 serial_cs 驅動程式。如果沒有，你可 能需要在你的 /etc/pcmcia/config 檔案中加入一條應該被適當 地辨識的條目。細節請看章節 6.1 (3.6)。
- 數據卡成功地被 serial_cs 所建構了嗎? 再來一次， 請檢查系統日誌，再 看看從 serial_cs 驅動程式所出來的訊息。 如果你看到 “register_serial() failed”， 你可能遇到了 I/O 埠與其他設備相衝突 了。 另外一個情報是， 如果這個設備被報告說是個 8250; 大部份現在的 PCMCIA 數據卡應該是被辨視成 16550A UART 類型的。如果您覺得遇上了一 個埠衝突了，請編修 /etc/pcmcia/config.opts 把被配置給數據 卡的埠範圍給排除。

- 遇上了插斷衝突嗎? 如果系統日誌看似很好，但數據卡就是好像不能工作， 請試試用 setserial 來把 irq 改成 0， 再看看數據機是否可以工作 了。這樣會讓串列驅動程式使用一個慢一些的票選模式而不使用插斷來工作。如果這樣可以修正這個問題，那就是在你的系統中有某個其他的設備使用 了被 serial_cs 所選用的插斷在工作著。你應該在 /etc/pcmcia/config.opts 中加一行命令將這個插斷給排除掉。
- 如果數據機好像只能慢慢地工作，這就很明顯地是個插斷衝突的問題了。
- 請確認你所遇到的問題眞的是個 PCMCIA 的問題。使用原廠商的驅動程式在 DOS 下看看是否可以正常工作即可知道。另外，在你已確定你可以做簡單的 連接前，請不要使用更複雜的方式來測試，比如説 SLIP 或 PPP。如果簡單 的連接正常但使用 SLIP 時卻不是，你的問題應該就是出在 SLIP 上，而不 是 PCMCIA。
- 如果你得到發自核心的訊息指出 serial_cs 模組無法被載入。 那就表示你 的核心並不支援串列設備。如果你已將串列驅動程式編譯成一個模組了，你 必須修改 /etc/pcmcia/config 來讓 serial 模組應該 在載入 serial_cs 前就被載入了。

4.5 PCMCIA SCSI卡

目前所有被支援的 PCMCIA SCSI 卡都工作的像下列的 ISA 匯流排卡的樣子： Qlogic、Adaptec AHA-152X 或是 Future Domain TMC-16x0。PCMCIA 驅 動程式被建立成與 PCMCIA 特別程式碼相連結 (如 qlogic_cs.c、 toaster_cs.c 或 fdomain_cs.c) 以及正常的 Linux SCSI 驅動 程式。

當一個新的 SCSI 主卡被偵測到時， SCSI 驅動程式會尋找連接著的設備。 請檢查系統日誌你的設備被正確地偵測到了。新的 SCSI 設備會被指定爲第 一個可用的 SCSI 設備檔。第一個 SCSI 硬碟是 /dev/sda，第一 個 SCSI 磁帶機是 /dev/st0， 還有， 第一個 CD-ROM 會是 /dev/scd0。

在 1.3.X (含) 以後的核心裏，PCMCIA 核心驅動程式可以從系統核心中找 出有哪些 SCSI 設備有連接在介面卡上。它們會被列表在 /var/run/stab， 而 SCSI 構建手稿 /etc/pcmcia/scsi 也會被每一個連接著的 設備所呼叫來建構或關閉該設備。內定的手稿並不會做任何建構 SCSI 設備 的動作，但它會妥當地在介面卡被移走時卸下在 SCSI 設備上檔案系統。

在 1.2.X 核心裏，PCMCIA 驅動程式無法自動地推測哪個設備被安裝在哪個 特定的 SCSI 介面卡上。然而，如果你有一個正常的 SCSI 設備之建構，你 可以將這些設備列明在 /etc/pcmcia/scsi.opts 中。例如，如果 你正常都有個 SCSI 磁碟以及一個 CD-ROM，你可以使用以下：

```
# For 1.2 kernels: list of attached devices
SCSI_DEVICES="sda scd0"
```

被傳到 <scsi.opts 的設備位址有點複雜， 那是因爲有許多種類的東 西可以被連接到 SCSI 介面卡上。這些位址包含了不是 6 個就是 7 個以逗 號來分開的欄位資料： 現在的架構、設備種類、插槽號碼、SCSI 頻道、識 別號、邏輯單位號碼，以及選擇性的分割號碼。設備種備爲 “sd” 指磁碟、“st” 指磁帶、“sr” 指 CD-ROM 設備， 還有 “sg” 指一般的 SCSI 設備。 大部份的設定中，SCSI 頻道還有邏輯單位號碼都設爲 0。對於有許 多磁碟分割區的磁碟設備，scsi.opts 會先被以 5 個欄位的位址所呼 叫。這手稿應該要設一個 PARTS 變數來列明這些分割。 然後， scsi.opts 會被每一個磁區以較長的 7 欄位位址所呼叫使用。

如果你的核心沒有針對特定的 SCSI 設備給上層驅動程式 (磁碟、磁帶等)，那麼這些設備就不會被 PCMCIA 驅動程式所建構。但有個邊際效應是， 該設備的名字在 /var/run/stab 內會變爲像 “sd#nnnn” 的名字。這 “nnnn” 是個 4 個數字的十六進位號碼。只要 cardmgr 無法轉譯一個 SCSI 設備的識別號碼到相對應的 Linux 設備名字時都會發 生這種事。

要將上層的 SCSI 驅動程式模組化成只有當 PCMCIA SCSI 卡被偵測到時它 們才會被載入是有可能的。要這麼做，你需要編修 /etc/pcmcia/config 來告訴 cardmgr 有哪個另外的模組在介面卡被建構時也要將它載入。 例如：

```
device "aha152x_cs"
  class "scsi" module "scsi/scsi_mod", "scsi/sd_mod", "aha152x_cs"
```

會在載入一般的 PCMCIA 驅動程式模組前將主要的 SCSI 模組和上層的磁碟 驅動程式模組給載入。PCMCIA 建構手稿不會自動地偵測被模組化了的 SCSI 模組，所以你必需手動建構選項來將 SCSI 支援的功能打開。

每次在開機您的筆記型電腦前先將 SCSI 設備開機，或是在插入控制卡前， 這樣子 SCSI 匯流排會在控制卡被建構後正確地被終止。另外還要注意抽出 SCSI 控制卡。 記得要確認在抽出控制卡前被連接著的 SCSI 設備都被正確 地卸下與關閉了。 最好的確認方法是在實體抽出卡片前使用 cardctl 或 cardinfo 來要求卡片的移出動作。 就這樣，所有的 SCSI 設備必 須在插上 SCSI 控制卡前已開啓了電源，而在你抽出控制卡和或關機筆記 電腦前都須要保持著它們是連接著的。

使用一般的 ISA 匯流排介面卡時可能會有潛在的複雜性， 尤其是無法啓動。 SCSI 匯流排搭載一個爲了適當地使用被動式 SCSI 終端器所必須的“終端電源”信號。PCMCIA SCSI 卡並不供應終端電源，所以如果需要該電 源的話，就必須由外接的設備來供應它。有些外接的 SCSI 設備可以被建構 成供應這個終端電源。其他的，如 Zip 磁碟以及 Syquest EZ-Drive，它們 使用主動式終端電阻就不需要如此了。有時候，可能需要使用一個特別的終 端電阻器如 APS SCSI Sentry 2，它有個外接的電源供應。當您建構 SCSI 設備串接時，請注意是否其中有任何的設備需要或可以提供終端電阻電源的。

Adaptec APA-460 SlimSCSI 卡並不被支援。這張卡原本是以 Trantor 的名 稱被販賣的， 當 Adaptec 併購了 Trantor 後，他們仍繼續以 Adaptec 的 商標來賣 Trantor 的卡，反正，APA-460 不被現有的任何一 Linux 驅動程 式所相容。

我不確定要寫個驅動程式會有多難， 我不認爲有任何人可以從 Adaptec 那 兒拿到技術資訊。

(未被支援的) Trantor SlimSCSI 可用以下方法被識別：

```
Trantor / Adaptec APA-460 SlimSCSI
FCC ID: IE8T460
Shipped with SCSIworks! driver software
```

(未被支援的) Adaptec SlimSCSI 可用以下方法被識別：

```
Adaptec APA-1460 SlimSCSI
FCC ID: FGT1460
P/N: 900100
Shipped with EZ-SCSI driver software
```

4.5.1 SCSI 設備的參數

以下的參數可以被定義在 scsi.opts內：

DO_FSTAB

布林 (y/n) 設定：指定是否這個設備應該在 /etc/fstab 內加個 條目。

DO_FSCK

布林 (y/n) 設定：指定是否檔案系統應該在被掛上前以 “fsck -Ta” 來檢查一下。

DO_MOUNT

林 (y/n) 設定： 指定是否這個設備應該在卡片被插入的同時被自動地被 掛上。

FSTYPE, OPTS, MOUNTPT

檔案系統型態、 掛上選項、以及被用來做爲 fstab 的項目及或掛上該設 備的掛上點。

例如，以下是一個建構一個 SCSI ID 爲 3、有 2 個分割區的磁碟設備還有 一個 SCSI ID 爲 6 的 CD-ROM 之手稿：

```
case "$ADDRESS" in
*,sd,*,0,3,0)
    # This device has two partitions...
    PARTS="1 2"
    ;;
*,sd,*,0,3,0,1)
    # Options for partition 1:
    #  update /etc/fstab, and mount an ext2 fs on /usr1
    DO_FSTAB="y" ; DO_FSCK="y" ; DO_MOUNT="y"
    FSTYPE="ext2"
    OPTS=""
    MOUNTPT="/usr1"
    ;;
*,sd,*,0,3,0,2)
    # Options for partition 2:
    #  update /etc/fstab, and mount an MS-DOS fs on /usr2
    DO_FSTAB="y" ; DO_FSCK="y" ; DO_MOUNT="y"
    FSTYPE="msdos"
    OPTS=""
    MOUNTPT="/usr2"
    ;;
*,sr,*,0,6,0)
    # Options for CD-ROM at SCSI ID 6
    PARTS=""
    DO_FSTAB="y" ; DO_FSCK="n" ; DO_MOUNT="y"
    FSTYPE="iso9660"
    OPTS="ro"
    MOUNTPT="/cdrom"
    ;;
esac
```

4.5.2 診斷SCSI卡的問題

- 使用 aha152x_cs 驅動程式 (被 Adaptec、 New Media 以及有些牌子 所使用)，似乎在 SCSI 脫離連接再連接支援時使用磁帶機時常會有問題。要將這個問題解除，請在 /etc/pcmcia/config.opts 中加入下 列這行：

```
module "aha152x_cs" opts "reconnect=0"
```

- 如果你將 SCSI 支援編譯成模組了 (CONFIG_SCSI is “m”)，在 建構 PCMCIA 時， 你必須指定要把建立 SCSI 驅動程式。 你也要修改 /etc/pcmcia/config 在適合的 *_cs 驅動程式被載入 前載入 SCSI 模組。
- 如果在 SCSI 匯流排被探測到時看到 “aborting command due to timeout” 訊息時，大部份是你有個插斷衝突了。

4.6 PCMCIA 記憶卡

1tt/memory_cs/ 驅動程式負責所有型態的設憶卡， 並且還提供對於那些有 其他功能的卡的 PCMCIA 記憶體位址空間直接存取功能。當被載入時，它建 立了一個字元與區塊設備的組合體。 請看這模組的 man 頁中有個完整的設 備取名規則的描述。區塊設備被用來做磁碟類型的存取 (建立和掛上檔案系統等等)。而字元設備被用來做 " 未加工 " 的未被緩衝之隨意位置上讀與 寫。

被傳送到 memory.opts 的設備位址包含了 2 個欄位： 架構和插槽號 碼。這個選項被放在相對應的記憶卡內的第一個共同記憶分割區中。

有些舊型的記憶卡， 還有大部份的簡單靜態 RAM 卡，都缺少 “ 卡片資訊 架構 ” (CIS)，e PCMCIA 卡用來辨視它們自已的一種 scheme。 一般來說， cardmgr 會將任何缺少 CIS 的卡片視爲一個簡單記憶卡， 並載入 memory_cs 驅動程式。 因此，有個邊際效應是，一般的識別問題是其 他型式的卡片可能會錯誤地被偵測爲記憶卡了。

memory_cs 驅動程式使用了 heuristic 來猜測這些卡片的容量。 heuristic 在寫入保護的卡片上並無法工作， 有時還有可能會出錯誤。 如果卡片被誤判了，當你在使用像 dd 或 mkfs 這樣的命令時， 它的容量應要明確地被指定。

4.6.1 記憶設備參數

下列參數可以放在 memory.opts內：

DO_FSTAB

  布林 (y/n) 設定：指定是否這個設備應該在 /etc/fstab 內加個 條目。

DO_FSCK

  布林 (y/n) 設定： 指定是否檔案系統應該在被掛上前以 “fsck -Ta” 來檢查一下。

DO_MOUNT

  布林 (y/n) 設定： 指定是否這個設備應該在卡片被插入的同時被自動地被 掛上。

FSTYPE, OPTS, MOUNTPT

  檔案系統型態、 掛上選項、以及被用來做爲 fstab 的項目及或掛上該設 備的掛上點。

以下範例會針對插上任一插槽的記憶卡做自動掛上動作的手稿：

```
case "$ADDRESS" in
*,0,0)
    # Mount filesystem, but don't update /etc/fstab
    DO_FSTAB="n" ; DO_FSCK="y" ; DO_MOUNT="y"
    FSTYPE="ext2" ; OPTS=""
```

```
        MOUNTPT="/mem0"
        ;;
    *,1,0)
        # Mount filesystem, but don't update /etc/fstab
        DO_FSTAB="n" ; DO_FSCK="y" ; DO_MOUNT="y"
        FSTYPE="ext2" ; OPTS=""
        MOUNTPT="/mem1"
        ;;
    esac
```

4.6.2 使用快閃記憶卡

傳到 ftl.opts 內的設備位址有 3 或 4 個欄位： the scheme、插槽 號碼、區號碼以及另外的選項，分割號碼。大部份的快閃卡只有一個快閃記 憶體區，所以大部份的區號碼都是 0 的。

要將快閃記憶卡當成一般的磁碟來使用， 首先要建立 FTL，或“ 快閃轉譯層級 (flash translation layer)”，它是一個使 用 ftl_format 命令在設備上做成的一個分割。 這層資料隱藏了快閃 記憶體程式的設備指定細節，並讓卡片看起來就像是個簡單的區塊設備。例 如：

```
ftl_format -i /dev/mem0c0c
```

記得這個命令是經由“ 未加工的 ” 記憶體卡片介面來存取卡片的。一經 格式化後， 這個卡就可以經由 ftl_cs 驅動程式將它當成一般的區塊 設備那樣地做存取了。例如：

```
mke2fs /dev/ftl0c0
mount -t ext2 /dev/ftl0c0 /mnt
```

被命名成 FTL 類的設備都有點怪怪的。Minor device numbers 有三個部份：卡片號碼、卡片內的區域號碼以及選項－在區域號碼內的分割。這兒的區 域可以被看成爲沒有分割表格的單一區塊設備 (就像軟體片一樣) 或者它 也可以被分割，就像是個硬碟設備一樣。“ftl0c0” 設備是卡片 0，共同 記憶體區域爲 0， 整個區域。 如果這個區域已被分割了， 那麼從 “ftl0c0p1” 到 “ftl0c0p4” 的設備是主分額區 1 到 4 了。

快閃記憶體卡片有兩種主要的格式: FTL 型式， 以及 Microsoft 的快閃檔案系統 Flash File System。FTL 格式是最有彈性的，因爲它允許一般的任何高階檔案系統 (ext2、ms-dos 等等) 都可視爲一 般磁碟設備般地來使用。FFS 則完全是個不同的檔案系統類型了。Linux 目 前無法處理用 FFS 格式化的卡片。

4.7 PCMCIA ATA/IDE 卡式硬碟機

對 ATA/IDE 磁碟機的支援需要 1.3.72 或更新的核心版本。 驅動程式內 指定 PCMCIA 的部份是 fixed_cs。 請記得在抽出 ATA/IDE 卡前請使 用 cardctl 或 cardinfo 程式把它先關閉， 因爲這個驅動程式 並沒有被設計成可“防止熱置換的”。

傳送到 fixed.opts 內的設備位置包含三或四個欄位: 目前的 scheme、插槽號碼、磁碟的序號以及選項的分割號碼。ide_info 可以 被用來獲得 IDE 磁碟的序號。 使用 SCSI 設備時，fixed.opts 會在 第一次被呼叫給整個設備做設定。 如果 fixed.opts 回應了一個在 PARTS 變數內的一串分割資訊， 命令手稿接著就會爲每一個分割做設 定。

4.7.1 ATA/IDE 固定磁碟設備參數

以下的參數可以被定義在 fixed.opts內：

DO_FSTAB

布林 (y/n) 設定：指定是否這個設備應該在 /etc/fstab 內加個 條目。

DO_FSCK

布林 (y/n) 設定：指定是否檔案系統應該在被掛上前以 “fsck -Ta” 來檢查一下。

DO_MOUNT

布林 (y/n) 設定： 指定是否這個設備應該在卡片被插入的同時被自動地被 掛上。

FSTYPE, OPTS, MOUNTPT

檔案系統型態、 掛上選項、以及被用來做爲 fstab 的項目及或掛上該設 備的掛上點。

以下是個 fixed.opts 檔案的例子用來掛上任一個 ATA/IDE 卡的第一 個分割到 /mnt 目錄上。

```
case "$ADDRESS" in
*,*,*)
    PARTS="1"
    ;;
*,*,*,1)
    DO_FSTAB="y" ; DO_FSCK="y" ; DO_MOUNT="y"
    FSTYPE="msdos"
    OPTS=""
    MOUNTPT="/mnt"
    ;;
esac
```

4.7.2 診斷 ATA/IDE 卡的問題

- 有些 IDE 磁碟允許 PCMCIA 的規格使用比最大可允許卡片設定起來的時間 更長的時間來起動。要使用這些卡片，請載入 pcmcia_core 模組配合 下面的設定:

  ```
  CORE_OPTS="unreset_delay=400"
  ```

- 要使月 ATA/IDE CD-ROM 設備，你的核心必須將 CONFIG_BLK_DEV_IDECD 功能打開且編譯進去。通常標準的核心都已支援了，然而如果你想編譯個自 定的核心請記得要謹愼些。

4.8 多功能卡

從 1.3.73 Linux 核心開始，幾個驅動程式可以共用單一個插斷，像串列埠 驅動程式與一個乙太網路驅動程式。在新的核心下使用多功能卡片時，所有 的卡片可以不用在載入或御下驅動程式的狀況下使用卡片的所有功能。

很多的硬體廠商自已做了不太能互相相容的 (有些可說是獨立的) 插斷共 享方法來模擬使用二個卡的功能。有些卡片 (Diamonds 的 Ositech Jack、、3Com 的 3c562、Linksys) 的驅動程式可以正確地支援這樣的模擬存取， 但是有些卡 (特別是 Megahertz) 則不能。

早期的核心並不支援插斷分享給不同的設備驅動程式。所要 PCMCIA 驅動程 式不可能建構此卡片讓它可模擬乙太網路卡與數據卡的存取。乙太網路卡與 串列埠驅動程式會被自動地載入。然而，內定上是乙太網路驅動程式 “ 擁 有 ” 這張卡片的插斷。要使用數據機，你可以卸下乙太網路驅動程式，然後使用以列方法重新建構串列埠:

```
ifconfig eth0 down
rmmod 3c589_cs
setserial /dev/modem autoconfig auto_irq
setserial /dev/modem
```

第二個 `setserial` 會查証出該埠在先前是由乙太網路驅動程式所使用 的插斷。

5 進階的主題

5.1 PCMCIA 設備的資源分配

理論上， 哪一個插斷被使用到哪個設備上都沒關係。 只要兩個 (或以上) 設備不被建構成使用相同的插斷即可。在 `/etc/pcmcia/config.opts` 裏你可以發現有個地方用來排除那些被非 PCMCIA 設備所使用著的插斷。

同樣地，也沒有辦法可以直接指定 I/O 位址給一個 PCMCIA 卡來使用。 `/etc/pcmcia/config.opts` 檔案允許你指定可被所有的 PCMCIA 設備所使用的埠範圍， 或是排除那些會和其他設備相衝突的埠範圍。

在修改了 `/etc/pcmcia/config.opts`後，你可以使用 “`kill -HUP`” 來重新啓動 `cardmgr`。

被用來監視卡片狀態的插斷是在 `cardmgr` 處理 `/etc/pcmcia/config` 檔案之前由低階的插槽驅動程式模組 (`i82365` 或 `tcic`) 所選擇 的。因此改變這個檔案並不會有什麼影響。要設定這個插斷，在插槽驅動程 式被載入時，設定在 `/etc/rc.d/rc.pcmcia` 內的 `PCIC_OPTS` 變數使用 `cs_irq=` 選項。

所有的客户端卡片驅動程式都有個叫做 `irq_list` 的參數。 被用來指 定哪些插斷可用試著用來配置給他們的。這些驅動程式選項應該被設定在你 的 `/etc/pcmcia/config` 檔案裏。例如:

```
device "serial_cs"
  module "serial_cs" opts "irq_list=8,12"
  ...
```

會指定串列埠驅動程式只使用 irq 8 或 irq 12。 不管 `irq_list` 的設定如何， 卡片服務程式 Card Services 絕不會分配已被其他設備所使 用了的插斷，或是在建構檔裏被排除的插斷。

5.2 我怎樣才能爲家庭及工作的使用做各別的設備設定呢?

我們可以很簡單地使用 PCMCIA “scheme” 支援來達成。 使用兩種不同的 建構 schemes，分別叫 “home” 及 “work”。 以下是個有 scheme 指定 的設定的 `network.opts` 手稿範例:

```
case "$ADDRESS" in
work,*,*,*)
    # definitions for network card in work scheme
```

```
        ...
        ;;
    home,*,*,*|default,*,*,*)
        # definitions for network card in home scheme
        ...
        ;;
    esac
```

PCMCIA 設備位址的首要部份都是建構 scheme。 在這個例子中， 第二個 “case” 句子會從 “homee” 和 “default” schemes 中挑選一個。 因 此，不管任何理由 scheme 並沒被設定了，內定值是 “home” 設定。

現在，要在二個設定集裏頭挑選一個，可執行:

```
cardctl scheme home
```

或者是

```
cardctl scheme work
```

`cardctl` 命令相當於關掉所有的卡片後再重啓動它們。 不管 PCMCIA 系統是否被載入，這個命令都可安全地被執行。但是如果你在當時正在使用 其他的 PCMCIA 設備時這個命令可能會失敗。 (雖然他們的建構並不完全依 靠 scheme 的設定。)。

要知道目前 PCMCIA scheme 的設定，執行:

```
cardctl scheme
```

5.3 從 PCMCIA 設備開機

要在一個 PCMCIA 設備上做個檔案系統需要耍個小把戲可行， 因爲 Linux 的 PCMCIA 系統並沒有設計成連接到核心程式上。它的主要元件，可載入式 核心模組以及使用者模式的 cardmgr 精靈是仰賴在一個已經在執行的系統 上。核心的 “initrd” 以允許 Linux 使用一個有最小化的根 映像檔、載入驅動程式的暫時性的動態記憶體磁碟來開機，然後再重新掛上 一個不同的根檔案系統。這個暫時的根系統可以建構 PCMCIA 設備，然後再 把它重新掛成一個根系統。

initrd 映像檔絕對必須存放在一個可開機的設備上: 一般來說這就無法放 到 PCMCIA 設備上了。這是因爲 BIOS 的限制，可不是核心的限制。在這裏， 能夠分辨出 “ 可開機的 (boot-able)” 設備 (例如，任何可以做開機 使用的設備) 與 “ 可做根系統的 (root-able)” 設備 (例如，可以被掛 入成爲根系統的設備) 的不同處也是很重要的。 “ 可開機的 (Boot-able) ” 設備是由 BIOS 所決定的， 通常是內部軟碟機和硬式磁碟 機。 “ 可做根系統的 (root-able)” 設備是任何可以在被載入即即被核心 所支援的區塊設備。 initrd 的功能可使更多的設備變成 “ 可做根系統的 (root-able)”，而非 “可開機的 (boot-able)”。

有些 Linux 供應版本允許安裝到 PCMCIA SCSI 卡接著的設備上。使得它們 無意地多出了可支援從 PCMCIA SCSI CD-ROM 設備來安裝的邊際效應。 然而， 到目前爲止， 並沒有任何 Linux 安裝工具支援建構一個可以從 PCMCIA 根檔案系統上開機到 Linux 的合適 “initrd”。 設立一個以 PCMCIA 根檔案的 Linux 系統需要您使用其他的 Linux 系統來產生一個 “initrd” 映像檔案。如果沒有可用的其他 Linux 系統，另一個替代方法 是做最小安裝到非 PCMCIA 磁碟上，建立一個 initrd 映像檔，然後再重新 安裝到 PCMCIA 目地設備上。

在 Linux Bootdisk-HOWTO 裏有一些關於設定開機磁碟片的一般資訊，但卻 沒有任何有關於 initrd 的資訊。 initrd 的主要文件被放在最新的核心程 式碼供應版本裏頭， 在 `linux/Documentation/initrd.txt`

。在開始前，你應該要先讀一下這份文件。 對於 lilo 的熟悉也是有助 益的。 要使用 initrd 也需要你已有個將 CONFIG_BLK_DEV_RAM 和 CONFIG_BLK_DEV_INITRD 功能打開後所編譯出來的核心。

是個高等建構技術， 需要您對 Linux 以及 PCMCIA 系統有一定程度的熟 悉度。請確認在您閱讀完所有相關的文件後再開始。下面的“食譜”應 該是可行的， 但是一旦不依照範例來做的話， 就會很快地將你推向未知及“不被支援的”地方，那你就得自己來的。

這方法需要你一定要使用 PCMCIA 驅動程式 2.9.5 或最新的版本才可以。 舊的 PCMCIA 套件或是個別的元件並不能在 initrd 裏使用。請不要將不同 的版本的元件混合了。

5.3.1 pcinitrd 輔助員手稿

pcinitrd 手稿用來建立一個使用 PCMCIA 根磁區分割來啓動 Linux 的基本 initrd 映像檔。 該映像檔裏包含一個最小的目錄層級結構、 以及有用的設備檔案、一些二進位檔案、共用程式庫檔以及一套 PCMCIA 驅動程式模組。 當你執行 pcinitrd 時，要指定要要包含進去映像檔 裏面的驅動程式模組名稱。 而主要 PCMCIA 核心元件 pcmcia_core 及 ds會被自動地包含進去。

例如，假設你的筆記電腦使用一個 i82365 同容的 PCMCIA 主控制器，而你 想從一個接著 Adaptec SlimSCSI 卡而裝有根檔案系統的硬碟上啓動 Linux。你可以建立一個合適的 initrd 映像檔，方法如下:

```
pcinitrd -v initrd pcmcia/i82365.o pcmcia/aha152x_cs.o
```

來自訂 initrd 啓使排程，你可以用下面的命令來使用 “loopback” 設備 來掛上該映像檔案:

```
mount -o loop -t ext2 initrd /mnt
```

然後編修 linuxrc 手稿。 PCMCIA 建構的檔案會被安裝到映像檔案裏 頭的 /etc 內， 當然你也可以自已設定。 細節請看 pcinitrd 的 man 頁。

5.3.2 建立一個 initrd 開機軟碟片

在建立了 pcinitrd 映像檔後， 你可以利用拷貝核心的方式來做一片 可開機的磁碟片， 並把被壓縮的 initrd 映像檔、支援給 lilo 使用 的檔案到一張空白磁碟片上。 下面的列子， 我們假設我們需要的 PCMCIA 根設備是 /dev/sda1:

```
mke2fs /dev/fd0
mount /dev/fd0 /mnt
mkdir /mnt/etc /mnt/boot /mnt/dev
cp -a /dev/fd0 /dev/sda1 /mnt/dev
cp [kernel-image] /mnt/vmlinuz
gzip < [initrd-image] > /mnt/initrd
```

建立一個包含下面內容的 /mnt/etc/lilo.conf 檔案:

```
boot=/dev/fd0
compact
image=/vmlinuz
    label=linux
    initrd=/initrd
    read-only
    root=/dev/sda1
```

最後，執行 lilo 如下:

```
lilo -r /mnt
```

當 `lilo` 被執行時配合使用 `-r` 選項，它會做所有與被指定的根 目錄的相關動作。 在 `/mnt/dev` 裏建立設備檔案的理由是， 當它是在替代的根目錄模式下 `lilo` 並無法使用在 `/dev` 內 的檔案。

5.3.3 安裝 **initrd** 映象檔到非**Linux** 磁碟內

initrd 功能一般都使用在當內部的硬碟機被分派給其他的作業系統使用時。 Linux 核心以及 initrd 映像檔可以被放在非 Linux 的磁區分割內， 而且 `lilo` 或是 `LOADLIN` 可以被設定由這些映像檔來開啓 Linux 系統。

假設你有個已被建構好合適的根設備，且 initrd 映像檔被建立在另外的系 統上了，啓動 Linux 系統的最容易方法是使用 `LOADLIN`，如下:

```
LOADLIN <kernel> initrd=<initrd-image>
```

一旦你可以在你的目標機器上啓動 Linux， 你就可以安裝 `lilo` 來允 許直接地啓動 Linux 了。 例如，假設 `/dev/hda1` 是個非 Linux 的目標磁區分割，且 `/mnt` 可以被用做爲掛入點。 首先，在目標 磁區分割上建立一個放 Linux 檔案的子目錄:

```
mount /dev/hda1 /mnt
mkdir /mnt/linux
cp [kernel-image] /mnt/linux/vmlinuz
cp [initrd-image] /mnt/linux/initrd
```

在這個例子中， 假設 `/dev/sda1` 是我們要做爲 Linux 根分割的 磁碟， 它是個經由 PCMCIA SCSI 卡所掛上的 SCSI 硬碟機。 要安裝 `lilo`，請建立一個有以下內容的 `lilo.conf` 檔:

```
boot=/dev/hda
map=/mnt/linux/map
compact
image=/mnt/linux/vmlinuz
        label=linux
        root=/dev/sda1
        initrd=/mnt/linux/initrd
        read-only
other=/dev/hda1
        table=/dev/hda
        label=windows
```

在 `boot=` 這行裏說要在指定的設備之主開機記錄 MBR(master boot record) 裏安裝開機載入程式。 `root=` 這行指出在載入 initrd 映像 檔後需要使用的根檔案系統，如果核心映像檔裏已這麼建構了那就不需這麼 做了。 在 `other=` 那行是用來描述說有其他的作業系統被安裝在 `/dev/hda1` 裏。

要安裝這個例子裏的 `lilo`，使用:

```
lilo -C lilo.conf
```

記得在這個案子裏，`lilo.conf` 檔案使用了包括 `/mnt` 在內 的絕對位址。我用這個例子的原因是目標檔案系統並不一定支援使用 `boot=` 和 `root=` 選項來建立 Linux 設備檔案的建立。

6 使用未被支援的卡片

6.1 建構無法被辨視的卡片

現在我們假設你的卡片已被現有的驅動程式所支援，我們所需要做的是在 /etc/pcmcia/config 中加入一個條目來告訴 cardmgr 如何 來辨視這張卡，哪些驅動程式需要連結到這張卡片上。建構檔案的格式細節 請查看 pcmcia 的 man 頁。 如果你插入一張不被認得的卡片， cardmgr 正常來說會記錄一些辨視資訊到系統日誌上， 我們可以利用 這些資訊來建造它的建構條目。

以下是個 cardmgr 如何將未支援卡的報告到 /usr/adm/messages 內的例子。

```
cardmgr[460]: unsupported card in socket 1
cardmgr[460]: version info: "MEGAHERTZ", "XJ2288", "V.34 PCMCIA MODEM"
```

相對應到/etc/pcmcia/config 的條目可以是:

```
card "Megahertz XJ2288 V.34 Fax Modem"
  version "MEGAHERTZ", "XJ2288", "V.34 PCMCIA MODEM"
  bind "serial_cs"
```

你可以使用 "*" 來代替不曉得的字串，像版本號碼啦。 當你寫新的建構 條目時，請小心地把完整的字串拷貝過去，保持原有的大小寫體以及空白字 元。還有，請確認建構條目與被報告到日誌檔案裏的資料有同樣數目的字串。

請牢記您幾乎可以指定任何的驅動程式給一張卡使用，但如果你只是胡亂地 亂弄，並不是個很有生產效率的作法。你也可能很幸運地發現現有的驅動程 式已支援了你的卡片。然而，大部份的結果是那些驅動程式並無法工作，也 有可能會產生邊際效應像鎖死你的系統了。不像一般的設備驅動程式只探測適當的卡片，探測 PCMCIA 設備的工作是由 cardmgr 所執行，驅動程 式本身在試著與設備溝通前並不會做啥麼確認動作的。

在編修 /etc/pcmcia/config 之後， 你可以暗示 cardmgr 重新載入該檔案，使用:

```
kill -HUP `cat /var/run/cardmgr.pid`
```

如果你設定好了一張新的卡片，請寄一份設定給我，這樣子我就可以把它包 含進標準建構檔案裏。

6.2 增列對 NE2000相容乙太網路卡的支援

首先， 先查看看這張卡是否已被 cardmgr 所辨識。 有些沒有被列在 SUPPORTED.CARDS 內的卡片事實上是有些被支援的卡的 OEM 版本。如 果你發現了像這樣的卡片，請讓我知道，這樣才能把它加到列表內。

如果你的卡片沒被辨別到， 請依照 6.1 (3.6) 內 的步驟爲你的卡片建立一個建構條目，並把這張卡片連繫上 pcnet_cs 驅動程式。 重新執行 cardmgr 即可使用被更新的建構檔案了。

如果 pcnet_cs 驅動程式說它無法決定你的卡片之硬體乙太網路位址，那麼請編修你的新建構條目將這張卡片連繫上記憶體卡片驅動程式 memory_cs 上。 重新執行 cardmgr 即可使用被更新的建構檔案了。你需要知道你的卡片的硬體乙太網路位址的值。這個位址是個以二個十 六位元字元號碼所排成的六個數字，通常它會被印在卡片上。如果它沒有印 在卡片上，你可以使用 DOS 的驅動程式來顯示這個位址。 只要你一知道這 個值，請執行:

```
dd if=/dev/mem0a count=20 | od -Ax -t x1
```

然後尋找你的位址之輸出。只有偶數位元組被定義了，因此可以忽略傾印資 料內的奇數位元組。記下位址的第一個位元組的十六位元位移。現在，編修 modules/pcnet_cs.c 並尋找 hw_info 構造。你需要爲你的 卡片建立一個新的條目。第一個欄位就是記憶體位移。接著的三個欄位就第 硬體位址的前三個位元組。最後一欄位含有卡片的特別功能的旗標; 開始時 請試著將它設爲 0。

編修完 pcnet_cs.c 後， 編譯與安裝新的模組。 再次修改 /etc/pcmcia/config， 然後改變卡片的連繫從 memory_cs 換成 pcnet_cs。順著指示重新載入建構檔案後就完成了。請您寄給我 一份你的新 hw_info 檔案以及建構的條目內容。

如果你沒有在十六進位傾印內找到你的卡片的硬體位址。最後的一個方法， 當 pcnet_cs 模組初始化時想要 “hard-wire(硬體接合)” 是有可 能的。請編修 /etc/pcmcia/config.opts 並加入 hw_addr= 選項，像這樣:

```
module "pcnet_cs" opts "hw_addr=0x00,0x80,0xc8,0x01,0x02,0x03"
```

在適宜的地方代換掉你自己的卡片之硬體位址。當然，請小心如果你已經做 了那麼多了，那你的卡片就不太像是個 NE2000 相容卡了。事實上，我並不 確定是否有 任何 卡片在經過了前面兩種方法之一後還不能用的了。

6.3 PCMCIA 軟碟機介面卡

在 Compaq Aero 及少數一些筆記型電腦所使用的 PCMCIA 軟碟機介面卡目 前尚未被支援。 主要障礙在於 Aero 似乎使用自訂的 PCMCIA 控制器來使 DMA 支援軟碟機。在不曉得這是如何辦到的情況下，我們就無法找出解決的 方法。

如果在 Aero 被開啓時軟碟機介面卡是插著的， Aero 的 BIOS 就會建構該 卡，此時 Linux 會把它視爲一般的軟碟機。當 Linux 的 PCMCIA 驅動程式 被載入時， 它們會查察到這卡已被建構並已附著在 Linux 的驅動程式上， 而這插槽也會被忽略不管了。所以，如果在開機時卡片即插著了那就可以使 用，但它不具備熱機交換的功能。

6.4 支援 Xircom 卡到底怎麼了？

在目前的 PCMCIA 套件中有含了一個古援 Xircom 乙太網路及乙太網路 / 數據機的驅動程式。這要感謝 Werner Koch 先生。我有設立一個專門討論 Xircom 驅動程式論譠的超媒體新聞，在 <http://hyper.stanford.edu/hypernews/get/pcmcia/xircom.html>。

以前 Xircom 卡並沒被支援是因爲 Xircom 公司有不能洩露他們的卡片之技 術資訊。後來，他們放鬆這些規定，現在主動提供驅動程式的資訊。

7 除錯小技巧及程式設計資訊

7.1 提出有用的小虫報告

提出小虫報告的最好的方法是使用在 Linux PCMCIA 資訊站的超媒體新聞訊 息列表。這樣可讓其他人知道最新的問題有哪些 (並修改或改變方法)。這 兒是小虫報告內應讓有的資料：

- 您的系統類型，以及 probe 命令的輸出訊息。
- 您使用哪些 PCMCIA 卡。
- 您的 Linux 核心程式版本及 PCMCIA 驅動程式版本。
- 您對 /etc/pcmcia 檔或 PCMCIA 啓動手稿所做的改變。
- 在系統日誌檔內所有與 PCMCIA 有關的訊息。

所有的 PCMCIA 程式模組和 cardmgr 精靈所傳到系統日誌檔的訊息。 通常爲 /var/log/messages 或 /usr/adm/messages。 當追蹤一個問題時這應該時第一個要察看的地方。當您提出小虫報告時請連 同包括這個檔案。 如果您在找系統訊息有任何問題時， 請檢查 /etc/syslogd.conf 來看有哪些不同的訊息類別被處理了。

在提出小虫報告前，請您檢查一下確認您使用的是最新版的驅動程式套件。 如果能先看已被我改正除錯後的報告的話會讓人稍稍高興一下，不然就有點 沒建設性地辜負我的心血了。

如果你的問題是核心的部份，從錯誤的地方之錯誤傾印只有你能夠追縱錯誤 位址— EIP 才有用。 如果錯誤是在主要核心內，看看在 System.map 內的位址，找出錯誤的功能函數。如果出錯的地方是在可載入式模組內，那 麼就很難追縱了。 使用目前的模組工具 “ksyms -m” 會提出一份每 一個可載入模組的基位址。選取包含了 EIP 位址的模組，然後把 EIP 減掉 基位址即可獲得模組內的位移。 然後，執行 gdb 在該模組上，使用 list 命令找到位移。 這項功能只有在你有使用 -g 選項在編譯 該模組時加入了除錯資訊的功能。

如果你沒有使用網路，小虫報告也可以寄到 dhinds@hyper.stanford.edu 來給我，我較希望你能把小虫報告貼到我的網站上，這樣子其他人也都可以 看到。

7.2 低階 PCMCIA 除錯輔助

PCMCIA 模組含有許多條件編譯的除錯碼。 大部份的這些碼都在前置處理器 PCMCIA_DEBUG 定義的控制下。如果它沒被定義，除錯碼就不會被編譯。如果設定位 0，控制碼會被編譯進入但不會被啓用。愈大的數字指定會變 得更冗長了。 以 PCMCIA_DEBUG 定義來建立的模組都會有個整數參數 pc_debug，它會控制它的輸出之多寡。 這可以在模組被載入時加以調 整，在不需重新編譯下使得輸出可以被控制成以每個模組爲單位了。

在 PCMCIA 供應版內的 debug_tools/ 子目錄內有一些除錯的工 具。 dump_tcic 和 dump_i365 兩個公用程式會產生 PCMCIA 控 制器的完整暫存器的傾印，並且將許多暫存器資訊的解碼。如果你有對相關 的控制晶片做存取的話， 這些資訊最最有用的了。 dump_tuples 公用程式列出了卡片的 CIS (卡片資訊結構)，並將一些較重要的資料解碼 出來。 dump_cisreg 公用程式顯示卡片的本地端建構暫存器 (local configuration registers)資料。

有時候 memory_cs 記憶體卡驅動程式用來除錯很好用。它可以與任何 的 PCMCIA 卡相連接，而且不會干擾到其他的驅動程式。它可以被用來對任 何卡片的屬性記憶體或通用記憶體的直接存取。

7.3 爲新卡片寫卡片服務驅動程式

Linux PCMCIA 程式設計師指引是 Linux PCMCIA 介面的最好文件。 最新的 版本你都可以從 hyper.stanford.edu 的 /pub/pcmcia/doc 目錄內或是在網站 <http://hyper.stanford.edu/HyperNews/get/pcmcia/home.html> 內找到。

對於那些接近於一般的 ISA 介面設備來說， 你也許可以使用已存在的 Linux 驅動程式來驅動。有時候，最大的障礙是修改一個已存在的驅動程式 使它可以在開機後處理加入或移出設備。在現行的驅動程式中，只有記憶體 卡是唯一"自我包含的"驅動程式－並不依賴 Linux 的核心的其他部 份來做苦工。

在很多例子中，要支援一張新卡的最大障礙在於從它的製造版那兒得到技術 資訊。 要知道問誰才對或是解釋哪些資訊是必需的也很難。 然而， 只有少數例子外，在沒有從製造廠得到技術資訊的情況下要寫個該卡的驅動 程式幾乎很難。

我寫了一個含了備註來解說許多有關一個驅動程式如何與 Card 服務程式相 溝通的架構驅動程式。 你可以在 PCMCIA 原始檔案的 `modules/skeleton.c` 內找到。

7.4 給 PCMCIA 客户自定驅動程式的作者的指引

我決定若要供應所有的 PCMCIA 客户端驅動程式來成爲 PCMCIA 套件的一部 份的話，這樣並不適合我。每一個新的驅動程式都會讓主要套件漸漸地難以 來維護。而且包含的這些驅動程式也會不請自來地將維護的工作從作者那兒 轉移到我的身上。因此，我會基於使用者的需求以及可維護性來以個案的方 式決定是否要包含哪些供應的驅動程式。對於那些不被包含在核心套裝的驅 動程式，我建議這些驅動程式的作者可以使用下面的方案來打包您的驅動程 式作爲供應用。

驅動程式的檔案應該被安排放在與 PCMCIA 來源供應版商的相同目錄結構下，如此，驅動程式就可被解開到完整的 PCMCIA 原始程式樹的上面了。一個 驅動程式應該包含原始程式檔案 (在 `./modules/`)， man 頁 (在 `./man/`)，建構檔案 (在 `./etc/`)。 在最上層的目錄內 也應該有個 README 讀我檔案。

最上層目錄也應該包含一個 makefile，它是一個組合用來執行 "`make -f ...` all" 以及 "`make -f... install`" 編譯驅動程式並安裝 適當的檔案。如果這個 makefile 有個 .mk 附加檔名，那麼它會自動 地被上層的 `Makefile` 命令加上 `all` 以及 `install` 目標 地時來執行。

以下是一個 makefile 如何被建立的例子：

```
# Sample Makefile for contributed client driver
FILES = sample_cs.mk README.sample_cs \
        modules/sample_cs.c modules/sample_cs.h \
        etc/sample etc/sample.opts man/sample_cs.4
all:
        $(MAKE) -C modules MODULES=sample_cs.o
install:
        $(MAKE) -C modules install-modules MODULES=sample_cs.o
        $(MAKE) -C etc install-clients CLIENTS=sample
        $(MAKE) -C man install-man4 MAN4=sample_cs.4
dist:
        tar czvf sample_cs.tar.gz $(FILES)
```

這個 makefile 使用 2.9.10 版本（含）以後的 PCMCIA 套裝程式所定義的 安裝目標地。它還包含了一個 "dist" 目標地來給驅動程式的作者方便性。 你也許想要加上版本編號到最後的套裝檔名上。 (例如， `sample_cs-1.5.tar.gz`)。一個完整供應版可以如下：

```
sample_cs.mk
README.sample_cs
modules/sample_cs.c
modules/sample_cs.h
etc/sample
```

```
etc/sample.opts
man/sample_cs.4
```

以這樣的安排，當供應版本驅動程式被解開時，它會變爲 PCMCIA 原始程式 樹的必要成員。這樣它就可以使用 PCMCIA 檔頭檔案以及檢查使用者系統建 構的機能、自動相關性檢查，就像是個“一般的”客户端驅動程式一樣。

我接受那些依照這個規格所準備的客户端驅動程式將它們放在我的 FTP 檔 案傳輸站 hyper.stanford.edu 的 /pub/pcmcia/contrib 目錄內。在這個目錄內的 README 檔案會述明如何解開供應的驅動程式。

PCMCIA 客户端驅動程式介面一直以來都沒有變動很多， 並且還都有保留向 後相容的功能。一客户端驅動程式並不需在主要的 PCMCIA 套件小部份的改 版時就得升級一次。我也會試著通知那些供應驅動程式的作者對於他們的驅 動程式需要更動的地方。

7.5 給 Linux 供應版本維護人員的導引

如果您使用的供應商版本提供系統建構工具程式使您須注意 PCMCIA 部份， 請使用在 /etc/pcmcia 內的 *.opts 檔案來”掛上” 那些功能。如果使用者編譯及安裝新版的 PCMCIA 套件時它們將不會被更動。如果您修改了主建構手稿後再安裝個新的 PCMCIA 套件時，這將會悄悄地 把您已自訂的手稿給覆蓋而中斷您之前與建構工具間的連接。如果您不曉得 怎麼來寫個合適的選項手稿，您可以與我連絡。

如果您能將您使用的供應商版本中有關 PCMCIA 套件的使用與本文件之不同 的地方寫成文件將對其他的使用者以及我本人有助益的。特別是，請在文件 上付上啓動手稿及建構手構的不同處。

如果您想做 Linux 供應版的 PCMCIA，最好也把非 PCMCIA 主要程式的其他 分享驅動程式一起包括進去。爲了方便維護，我會盡力地將核心套件的大小 限制在一定範圍內，除非有我覺得會被大家感興趣的部份才會再加進去。如 前面所說，其他的驅動程式會被分開地供應。對於被整合和分開於核心部份 的驅動程式之界定是隨意且有些是有其歷史性的，因此我們不能以爲它們在 品質上有任何的不同。

後記:

譯者按: 在翻譯本篇文章的過程中，共遇到二次翻譯到一半而原作者修正文 件及重新編排的狀況。因此，本譯文可能有翻譯不周延或錯字之處，煩請發 現錯誤地方的朋友來信到 *linuxer.bbs@cis.nctu.edu.tw* 給我，以便修正，謝謝您!