

- Blog
- Paste
- Ubuntu
- Wiki
- Linux
- Forum

搜索

進入

搜索

- 頁面
- 討論
- 編輯
- 歷史
- 简体
- 繁體

- 導航
 - 首頁
 - 社群入口
 - 現時事件
 - 最近更改
 - 隨機頁面
 - 幫助
- 工具箱
 - 鏈入頁面
 - 鏈出更改
 - 所有特殊頁面
- 個人工具
 - 登入

跟我一起寫**Makefile**:後序

出自**Ubuntu**中文

*導

* |概述 + **MakeFile**介紹 + 書寫規則 + 書寫命令 + 使用變量 + 使用條件判

*航斷 |

* |使用函數 + **make**運行 + 隱含規則 + 使用**make**更新函數庫文件 + 後序 |

目錄

■ 1 後序

後序

終於到寫結束語的時候了，以上基本上就是GNU make的Makefile的所有細節了。其它的產商的make基本上也就是這樣的，無論什麼樣的make，都是以文件的依賴性為基礎的，其基本是都是遵循一個標準的。這篇文檔中80%的技術細節都適用於任何的make，我猜測"函數"那一章的內容可能不是其它make所支持的，而隱含規則方面，我想不同的make會有不同的實現，我沒有精力來查看GNU的make和VC的nmake、BCB的 make，或是別的UNIX下的make有些什麼

樣的差別，一是時間精力不夠，二是因為我基本上都是在Unix下使用make，以前在SCO Unix和IBM的AIX，現在在Linux、Solaris、HP-UX、AIX和Alpha下使用，Linux和Solaris下更多一點。不過，我可以肯定的是，在Unix下的make，無論是哪種平台，幾乎都使用了Richard Stallman開發的make和cc/gcc的編譯器，而且，基本上都是GNU的make（公司里所有的UNIX機器上都被裝上了GNU的東西，所以，使用GNU的程序也就多了一些）。GNU的東西還是很不錯的，特別是使用得深了以後，越來越覺得GNU的軟件的強大，也越來越覺得GNU的在操作系統中（主要是Unix，甚至Windows）"殺傷力"。

對於上述所有的make的細節，我們不但可以利用make這個工具來編譯我們的程序，還可以利用make來完成其它的工作，因為規則中的命令可以是任何Shell之下的命令，所以，在Unix下，你不一定只是使用程序語言的編譯器，你還可以在Makefile中書寫其它的命令，如：tar、awk、mail、sed、cvs、compress、ls、rm、yacc、rpm、ftp……等等，等等，來完成諸如"程序打包"、"程序備份"、"製作程序安裝包"、"提交代碼"、"使用程序模板"、"合併文件"等等五花八門的功能，文件操作，文件管理，編程開發設計，或是其它一些異想天開的東西。比如，以前在書寫銀行交易程序時，由於銀行的交易程序基本一樣，就見到有人書寫了一些交易的通用程序模板，在該模板中把一些網絡通訊、數據庫操作的、業務操作共性的東西寫在一個文件中，在這些文件中用些諸如"@@N、###N"奇怪字串標註一些位置，然後書寫交易時，只需按照一種特定的規則書寫特定的處理，最後在make時，使用awk和sed，把模板中的"@@N、###N"等字串替代成特定的程序，形成C文件，然後再編譯。這個動作很像數據庫的"擴展C"語言（即在C語言中用"EXEC SQL"的樣子執行SQL語句，在用cc/gcc編譯之前，需要使用"擴展C"的翻譯程序，如cpre，把其翻譯成標準C）。如果你在使用make時有一些更為絕妙的方法，請記得告訴我啊。

回頭看看整篇文檔，不覺記起幾年前剛剛開始在Unix下做開發的時候，有人問我會不會寫Makefile時，我兩眼發直，根本不知道在說什麼。一開始看到別人在vi中寫完程序后輸入"!make"時，還以為是vi的功能，後來才知道有一個Makefile在作怪，於是上網查啊查，那時又不願意看英文，發現就根本沒有中文的文檔介紹Makefile，只得看別人寫的Makefile，自己瞎碰瞎搞才積累了一點知識，但在很多地方完全是知其然不知所以然。後來開始從事UNIX下產品軟件的開發，看到一個400人年，近200萬行代碼的大工程，發現要編譯這樣一個龐然大物，如果沒有Makefile，那會是多麼恐怖的一樣事啊。於是橫下心來，狠命地讀了一堆英文文檔，才覺得對其掌握了。但發現目前網上對Makefile介紹的文章還是少得那麼的可憐，所以想寫這樣一篇文章，共享給大家，希望能對各位有所幫助。

現在我終於寫完了，看了看文件的創建時間，這篇技術文檔也寫了兩個多月了。發現，自己知道是一回事，要寫下來，跟別人講述又是另外一回事，而且，現在越來越沒有時間專研技術細節，所以在寫作時，發現在闡述一些細節問題時很難做到嚴謹和精練，而且對先講什麼后講什麼不是很清楚，所以，還是參考了一些國外站點上的資料和題綱，以及一些技術書籍的語言風格，才得以完成。整篇文檔的提綱是基於GNU的Makefile技術手冊的提綱來書寫的，並結合了自己的工作經驗，以及自己的學習歷程。因為從來沒有寫過這麼長，這麼細的文檔，所以一定會有很多地方存在表達問題，語言歧義或是錯誤。因些，我迫切地得等待各位給我指證和建議，以及任何的反饋。

最後，還是利用這個後序，介紹一下自己。我目前從事於所有Unix平台下的軟件研發，主要是做分佈式計算/網格計算方面的系統產品軟件，並且我對於下一代的計算機革命——網格計算非常地感興趣，對於分佈式計算、P2P、Web Service、J2EE技術方向也很感興趣，同時，對於項目實施、團隊管理、項目管理也小有心得，希望同樣和我戰鬥在“技術和管理並重”的陣線上的年輕一代，能夠和我多多地交流。我的MSN是：haoel@hotmail.com（常用），QQ是：753640（不常用）。（注：請勿給我MSN的郵箱發信，由於hotmail的垃圾郵件導致我拒收這個郵箱的所有來信）

我歡迎任何形式的交流，無論是討論技術還是管理，或是其它海闊天空的東西。除了政治和娛樂新聞我不關心，其它只要積極向上的東西我都歡迎！

最最後，我還想介紹一下make程序的設計開發者。

首當其衝的是： Richard Stallman

開源軟件的領袖和先驅，從來沒有領過一天工資，從來沒有使用過Windows操作系統。對於他的事迹和他的軟件以及他的思想，我無需說過多的話，相信大家對這個人並不比我陌生，這是他的主頁：<http://www.stallman.org/>。這裏只貼上一張他的近照：



第二位是：Roland McGrath

個人主頁是：<http://www.frob.com/~roland/>，下面是他的一些事迹：

- 1) 合作編寫了並維護GNU make。
- 2) 和Thomas Bushnell一同編寫了GNU Hurd。
- 3) 編寫並維護着GNU C library。

4) 合作編寫並維護着部分的GNU Emacs。

在此，向這兩位開源項目的鬥士致以最真切的敬意。

(全文完)

取自 "<http://wiki.ubuntu.org.cn/index.php?title=%E8%B7%9F%E6%88%91%E4%B8%80%E8%B5%B7%E5%86%99Makefile:%E5%90%8E%E5%BA%8F&variant=zh-hant>"

本頁面已經被瀏覽2,067次。

- 此頁由Dbzhang800於2008年4月14日 (星期一) 22:06的最後更改。
 - 關於Ubuntu中文
 - 免責聲明