
序

這本書的目的是協助你撰寫出更好的 Perl 程式：事實上，就是你可能寫出的最佳 Perl 程式。本書收集了 256 則指導方針，涵蓋撰碼技藝的各個面向，包括配置、名稱選擇、資料和控制結構的選擇、程式重組、介面設計和實作、模組化、物件導向、錯誤處理、測試、以及除錯。這些指導方針的發展和調整，歷經 22 年程式設計生涯的淬鍊，產生能夠共同協調運作、清晰、強健、有效率、可維護、以及簡明的程式。

提醒你，這可不容易達成。簡明是清晰的天敵，而效率是減損可維護性的因子。此外，為了讓程式強健所做的防護，也會侵蝕清晰、效率、簡明、以及維護性。有時，你就是無法「面面俱到」。

本書並非試圖提供一組放諸四海皆準的唯一最佳實務。量測程式品質有很多方式，而評斷程式也有多種面向，因為不同程式設計師會有不同的評估觀點。每位程式設計師和每個程式設計團隊，對何謂程式最重要且最不可或缺的屬性，都有自己的看法。

相反的，本書提供的是一組最佳實務：這組最佳實務前後連貫、用途廣泛、目標平衡、而且是根據程式撰寫的真實經驗，而非某人認為程式應該如何打造的象牙塔理論。最重要的是，這組實務可實際運作，而且世界各地已經有眾多開發人員在使用。如同 Perl 本身，這些指導方針是為了協助你把工作做好，而不是變成開發之路的絆腳石。

如果你是有經驗的開發人員，幾乎可以肯定你不會喜歡後頭所有的建言。你會發現其中有些原則並不自然，或者違反直覺；而其他人可能會覺得很死板，不像 Perl。也許這些原則看起來就像是以不必要的不同方式撰寫軟體，有別於你長久以來根深柢固、覺得很舒服的撰碼習慣。

當你閱讀本書建言時，試著將這些感覺放到一旁。當你要做任何程式改良時，就回頭看一看這本書：分析一下新實務所提出的論據；問自己是否掉進那些新實務試著避開的陷阱；考慮一下書中所建議的撰碼技巧是否可能值得一試。

想一想這些議題（對你當前撰寫程式的方式保持意識清醒），會有很大的益處，即使最後你根本不採用任何一道建言。

本書內容

第 1 章《最佳實務》說明為何值得花時間重新評估你當前的撰碼實務。文中討論撰碼風格的演化，提出三項寬廣準則，說明現有或提議中的撰碼實務應該加以重新評估。本章也說明為何良好的撰碼習慣是很要緊的，以及提出如何培養良好撰碼習慣的建言。

第 2 章《程式配置》要解決諸多爭議性十足的程式配置議題。本章建議如何設立區塊分界字元，區分內建函式以及次常式和變數之關鍵字，將運算子、終結符號、分界字元、以及其他標號擺在何處，如何以一致的空白使用增加可讀性，程式列和區塊縮排的最佳寬度，如何設立值清單，從何處把冗長運算式打斷。最後的結論提出方便的工具，讓多數配置任務自動化。

第 3 章《命名規則》介紹一系列指導方針，協助你替變數、次常式、以及名稱空間選擇更具有描述性的名稱。此外，也示範說明一致性命名手段的各種元素如何協同運作，以改良程式的整體維護性（使其更具可讀性，並減少解謎用的註解）。

第 4 章《值和運算式》提供一組簡單規則，當你建立字元字串、數字、以及清單時，可協助你避開常見的陷阱。主題包括如何避免意外的變數安插（以及非安插）、對非列印字元所採取的可靠且可讀的手法、定義常數、避免未修飾字（bareword）、並且馴服即席文件（heredoc）、逗號、長數字和清單。

第 5 章《變數》探索使用變數的強健手法。文中說明套件變數或標點變數的內在缺點，在可行之處，提出較安全的替代選項，在無替代選項時，建議較安全的實務作法。本章的後半部會介紹幾種有效率、維護性高的技巧，以處理陣列和雜湊表中的資料（使用「容器切片」機制）。

第 6 章《控制結構》檢視 Perl 豐富的控制結構，鼓勵你使用那些易維護、不易出錯、或較有效率的控制結構。本章提出一組簡單的指導方針，以決定對特定任務而言，`for`、`while`、或 `map` 之中哪一個才是最恰當的選擇。我們也會討論有效使用反覆器變數，包括必須同時以鍵和值繞行雜湊表項目的情況。

第 7 章《說明文件》建議一系列技巧，讓你在替程式撰寫說明文件時比較不無聊，因此，你就更可能願意寫。文中提倡一種範本式的手法以撰寫使用者和技術說明文件，討論何時、何處、以及如何撰寫有用以及準確的註解。

第 8 章《內建函式》討論使用 Perl 最著名之內建函式的較佳方式，包括 `sort`、`reverse`、`scalar`、`eval`、`unpack`、`split`、`substr`、`values`、`select`、`sleep`、`map`、以及 `grep`。文中也總結出其他許多有用的「非內建」的內建函式（來自標準 Perl 發行套件和 CPAN 上的兩個模組）

第 9 章《次常式》說明一些有效率以及具有維護性的方式，以撰寫 Perl 次常式，包括位置、命名、以及可有可無引數的使用，引數驗證及預設值，安全的呼叫及傳回約定，各種情境中可預測的傳回值，以及為什麼應該避免次常式原型及隱性傳回值。

第 10 章《I/O》說明如何以可靠方式開啟和關閉檔案、何時使用以列為基準的資料輸入、如何正確偵測互動式應用程式、提示的重要性、以及如何長時間非互動式任務中提供最佳回饋給使用者。

第 11 章《參考位址》提出建言，解開 Perl 眾多提領（解參照）語法的神秘之處，討論為何符號參考位址製造的問題比能解決的問題更多，然後建議你避免使用循環式參考位址鏈，以免造成記憶體漏洞。

第 12 章《正規表示式》介紹使用正規表示式的指導方針。文中建議使用延伸格式，針對 Perl 令人困惑的「單列」和「多列」比對模式，提出一種簡單但罕見的解決辦法，提醒你比對空白時的危險性，說明如何避免使用易出錯的數值變數，介紹強健手法，以建立複雜但仍具維護性的正規表示式，提出一些讓緩慢比對最佳化的指示，最後以說明不要使用正規表示式的時機作為結論。

第 13 章《錯誤處理》提倡以例外事件為基礎的一致性手法處理錯誤，然後說明為什麼例外事件會比特殊傳回值或旗標更好的原因。文中也建議你使用例外事件物件，探索其宣告、建立、拋出、捕捉、以及處理的細節。

第 14 章《命令列處理》說明命令列介面的設計和實作（針對個別程式和應用程式集組）。文中會建議幾個模組，使你的命令列介面更為一致、可預測，同時也能大量減少實作這些介面所需付出的精力。

第 15 章《物件》和第 16 章《類別階層》提供強健而有效率的手法，建立 Perl 之物件和類別階層。此手法可提供完全的封裝物件，而無須受到效能懲罰，支援單一繼承和多重繼承，而不會碰到常見的問題，諸如屬性衝撞、不完全的初始設定、部分解構、或者不正確的方法自動載入。第 16 章也會介紹一個新模組，讓這些強健有效率的類別可以用半自動的方式建立。

第 17 章《模組》談的是非物件導向模組，探索建立模組、設計其介面、宣告以及檢查其版本編號、以及將現有程式重構成模組的最佳方式。本章也會討論 Perl 標準程式庫和 CPAN 上眾多免費可用之現有模組。

第 18 章《測試和除錯》鼓勵你使用測試，提倡以核心 `Test::` 模組進行測試驅動的設計和開發。此外，也提出一些技巧，說明有效之除錯技術，包括說明各種模組，以及可以讓除錯簡單一點的免費工具。

第 19 章《其他主題》提供各種主題的指導方針，諸如版本控制、和其他語言寫成的程式互動、處理組態檔、文字格式化、繫結變數、對程式進行效能評量和側寫、快取機制技術、以及有關重構的一般性建言。

附錄 A《Perl 必要之最佳實務》以三份清單，總結出本書中最重要的 30 則指導方針。附錄 B《Perl 最佳實務》列出全部 256 則指導方針，而且有各章次和位置的交叉參考資訊。附錄 C《編輯器組態》提供 *Vim*、*Vile*、*Emacs*、*TextWrangler*、以及 *BBEdit* 文字編輯器的有用組態選項。附錄 D《推薦的模組和工具》列出本書建議的各種模組及交叉參考資訊，而且對當中最有用的次常式提出簡要的說明。附錄 E 列出簡短的參考文獻。

本書印刷體裁

下列是本書所用的印刷體裁：

斜體字

表示強調、新術語、URL、電子郵件、檔名、路徑、以及 Unix 工具。

定寬字

表示命令、變數、屬性、函式、類別、名稱空間、方法、模組、值、檔案內容、命令的輸出、以及破壞所建議之實務的程式範例。

定寬粗體字

表示應該由使用者逐字輸入的命令和其他文字，以及示範說明所建議之實務的程式範例。

定寬斜體字

表示文字應由使用者提供之值替換，以及程式範例中的註解。

程式範例

這本書是協助你把工作做好。一般而言，你可以在你的程式和說明文件中使用本書的程式碼。除非你要重製重要的程式碼，否則無需取得我們的許可。例如，使用本書程式碼片段寫程式，並不需要取得我們的許可。但是，把 O'Reilly 書籍的程式範例燒成光碟片販賣或散佈，就需要取得授權。引用本書的文句和範例程式碼來回答問題，不需要取得許可。把本書大量的程式範例整合到你產品的說明文件中，則需要取得授權。

雖然不是必要，但註明來源，我們會很感謝。註明來源通常包括書名、作者、出版商、以及 ISBN。例如，「Perl Best Practices, by Damian Conway. Copyright 2005 O'Reilly Media, Inc., ISBN 碼」。

如果你覺得你對書中程式範例的使用情況有別於上述情況，不用客氣，盡量和我們連絡：permissions@oreilly.com。

誌謝

每本書都是一大群人的成果。寫這本書時，我受益於 Perl 社群中最天才人士的協助、建議、以及支援。我想表達我最深沉的感謝：

感謝我的第一位編輯 Linda Mui，她似乎有無限的耐心，堅持不懈的信念，給我寬廣的自由，終於找到我的書的主題。

感謝我的第二位編輯 Nat Torkington，他重新燃起我的寫作熱忱，十年來的夥伴關係，不斷的鼓勵和支持我。

感謝我的第三組編輯 Allison Randal 和 Tatiana Apandi，他們相當優雅、沉著、有同理心、能給好建議、相當有效率、而且提供了非常務實的協助。

感謝 Sean M. Burke，他給予我細心的技術分析、無價的文法建言、醒目的問題、以及相當傑出的補充說明（我真希望能用到）。

感謝 Nancy Kotary，她以超強的文字編輯功力改善我的寫作能力。

感謝 Larry Wall，五年來給予我的友誼和智慧，而且教會我一件事：謙遜。

感謝 Abigail，他提出一種撰碼 Perl 的好手法，而且對許多主題都提出無價之建言，特別是正規表示式和 Inside-Out 物件。

感謝 Nate Bailey，他過去幾年來毫不懈怠地支援我，協助我鍛字鍊句，一開始就寫出有意義的句子，此外，也謝謝他熟練的反社會二手知識。

感謝 Hildo Biersma，他對大型團隊環境的開發實境有專業的觀點，而且給了相當多優良建議，在單單一本書中實在無法消耗。

感謝 Chris Devers，對本書措辭做了較佳的統一，當一個段落就夠時，讓我節省額外一章的量。

感謝 Richard Dice，引領令人信服的經濟性論據，讓審查流程一直在軌道上，以及為我找到這個完美的類似生意運算元。

感謝 Stephen Edmonds，他找出三個非常微妙的撰碼錯誤，才得以從印刷品中排除惱人的結果。

感謝 Paul Fenwick，他從教師的觀點看待這些想法的表達，而且當我破壞自己的指導方針時，他就不給糖吃。

感謝 Garrett Goebel，他對細節特別注意，而且有超強能力可以從那些特定的建言推論出絕佳的通用建言。

感謝 Mick Goulish，他給我熱情的鼓勵，在我壓力大時，給我多角度的建言，還有在我正好需要的時候，講一些我聽過最好笑的笑話。

感謝 Uri Guttman，他能看出其他人看不見的東西。

感謝 Trey Harris，他是專家級的程式設計師，激發我找出更棒的方式，而非僅僅是簡單方式。

感謝 Brand Hilton，他找出打字錯誤的能力超強，還有發現程式範例臭蟲的詭異能力，以及他對「**unless**」英勇的捍衛。

感謝 Steven Lembark，他給了許多傑出的建言，協助我把精神集中在真實世界的開發人員的需求上。

感謝 Andy Lester，他對最佳撰碼實務的理解和熱情讓我產生不少靈感。

感謝 Greg London，他看得見真正的程式設計師需要什麼，當我的幽默感行不通時，他也會老實告訴我。

感謝 Tim Maher，多年來多虧他的友誼和支持，還有偶爾不同意見的珍貴聲音。

感謝 Bill Odom，他分享許多他的智慧和經驗，仁慈地讓我借用他的許多絕佳想法。

感謝 Jacinta Richardson，她給了我許多文法和句法上的卓越建言，還有她對隨便寫作絕不妥協的立場。

感謝 Bill Ricker，他對一些指導方針、模組、以及版本給了寶貴的說明，還有他能從大處著眼看出不利於公司組織的建言，此外，還有他對計算理論、實務、以及歷史的卓越知識。

感謝 Randal Schwartz，他擠出時間對這本書提出回饋；他手上同時還有兩本書在修改。

感謝 Peter Scott，他分享了他那無與倫比的經驗、知識、以及智慧；在悲傷的時候，他和 Grace 也給予我仁慈和支持。

感謝 Mike Stok，他對 Perl 程式設計有著獨到的觀點，從中流瀉出許多有價值的建言。

感謝 Tony Stubblebine，有人要評語的話，他會毫不留情痛批，尤其是他對說明文件的穩健建言。

感謝 Andrew Sundstrom，他以獨到的方式把戰士、哲學家、以及詩人揉合在他生活的各個層面，包括程式設計在內。

感謝 Bennett Todd，他的鼓勵和建言我仰賴了五年，也熬出了兩本書。

感謝 Merijn Broeren、Dave Cross、以及 Tom Christiansen，他們給予本書慷慨的協助。

感謝 Dave Rolsky、Brian Ingerson、以及 Eric J. Roode，我對他們已經很棒的模組提出不客氣的建言，他們也以寬厚的胸襟接納。

感謝我的 Perl 同事：Joe Hourcle（他對 BBEdition 和 TextWrangler 提出了詳盡而即時的協助），John McNamara、Jim Mahoney、Scott Lanning、以及 Michael Joyce（Emacs 的建言都歸功於他們），以及其他新手、和尚、修士、住持、以及聖人，協助我把編輯器組態附錄做出來：*artist*、*barrachois*、*Fletch*、*InfiniteLoop*、*jplindstrom*、*leriksen*、*Limbic~Region*、*runrig*、*Smylers*、以及 *steff*。

感謝我的父母 Sandra 和 Jim Conway，以及我的岳父和岳母 Fred 和 Corallie Stock，他們一直都相信我。

最後，要感謝我最愛的 Linda，她是我所有靈感的泉源，以及我所做一切的理由。