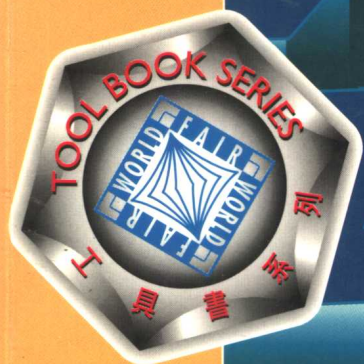


電腦名詞大全



AN INFORMAL DICTIONARY OF COMPUTER TERMS



Peachpit Press

ROBIN WILLIAMS
with **STEVE CUMMINGS**

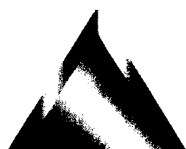


**JARGON, AN INFORMAL DICTIONARY
OF COMPUTER TERMS**

電腦名詞大全

Robin Williams, Steve Cummings ■

緯輝電子出版公司
WORLD FAIR PUBLISHING
1997



Jargon, an informal dictionary of computer terms

CHINESE EDITION

© 1997 by World Fair Publishing.

Authorized translation of the English edition

© 1993 by Robin Williams and Steve Cummings. This translation is published and sold by permission of Robin Williams and Steve Cummings, the owner of all rights to publish and sell the same, through Leed and Wood Co. Ltd.

Original title Jargon, an informal dictionary of computer terms

Copyright© 1993 by Robin Williams and Steve Cummings.

First publishing in the United States by Peachpit Press; Berkeley, CA 94710, U.S.A.

All rights reserved including the right of reproduction in whole or in part in any form.

《電腦名詞大全》是《工具書》系列之一。

1997 年由緯輝電子出版公司翻譯出版。

原英文版 Jargon, an informal dictionary of computer terms 由 Peachpit Press 於 1993 年出版。
版權所有。

電腦名詞大全

作 者：Robin Williams, Steve Cummings

翻 譯：周克希 薛密 于玉 李偉德 周克言
諸葛勤 李旭輝 朱埔達 黃小峰 張芄

審 校：周克希

出 版：緯輝電子出版公司 World Fair Publishing
香港新界大埔汀麗路36號中華商務印刷大廈10字樓
電話：2666 4808

發 行：萬昌科技有限公司 Techpro Technology Ltd.
香港銅鑼灣告士打道 311 號皇室堡 1101 室
電話：2576 1328

製 作：緯輝電子出版公司 World Fair Publishing

印 刷：美迪製作公司 Media Productions
香港新界大埔汀麗路36號中華商務印刷大廈10字樓

1997 年 1 月 第 1 版 第 1 次印刷

© 1997 緯輝電子出版公司

ISBN 962-7966-37-1

作者前言

寫這樣一本書吃力得很。我不是專業人員，許多專業性很強的概念，真是費了我不少精力。在我寫這本書（一邊寫一邊叫苦不迭）的時候，好些人（尤其是男性的電腦行家）都勸我不必寫這種沒人會看的勞什子。但是我認為，電腦正在成為我們每個人生活的組成部分，有關電腦的術語幾乎隨處可見——報章雜誌，目錄手冊，推銷廣告，到處都會碰到這些術語。我看過一些讀者寫給雜誌的信，他們抱怨雜誌的文章用了那麼些縮略詞，叫人看了不知所云。可是你知道嗎，即使告訴了你這個縮略詞代表哪些詞，你照樣可能莫名其妙——就算你知道 SCSI 表示 Small Computer Systems Interface（小型電腦系統介面），那又怎麼樣呢？

我之所以想寫這麼一本書，正是因為手邊的電腦詞典（我一共有 12 本電腦詞典，還不算兩本儲存在磁碟裏的）以及那麼些詞彙手冊（我收集了近 20 本）實在令我失望得很。釋義都那麼簡短，彷彿作者都假定許多東西我是早就該懂的，而且誰也不肯告訴我，這些概念跟日常生活有些什麼關係，或者我為什麼應當了解這些概念。正是讀了詞條 SIMM 的釋義（這個詞條偏偏不按字母順序排在“SIMM”的位置，而是排在“Single In-line Memory Module”的位置，倒像我事先就該明白 SIMM 是什麼詞的縮寫似的），我才真正意識到我有責任自己動手來寫一本詞典，好讓像我和我母親這樣的普通人都能看得懂。那條 SIMM 的釋義是這樣的：

一種記憶體模塊，裏面裝有在電腦中添加 256K 或 1M 隨機存取記憶體所需的晶片。
SIMM 通常插入母板或邏輯電路板。

這一釋義假定我不僅知道什麼是記憶體，什麼是記憶體模塊，什麼是晶片，而且明白 256K 是什麼意思，1M 是什麼意思，隨機存取是什麼意思，母板和邏輯電路板又是什麼意思。可要是這些東西我都懂了的話，我恐怕也就不會來查 SIMM 這個詞條了。（再說我也不明白，電腦詞典幹嗎非得把開頭一句寫成不完整的句式。）

哦，我真不該這麼盛氣凌人、盡發牢騷——確實也有幾本詞典相當出色，我曾經從中獲益匪淺，而且倘若你看了這本詞典的釋義後還想再多了解些情況的話，我很願意推薦你去參閱這幾本詞典。不過，我寫這本書的宗旨跟大多數詞典都有所不同。這本書並不是一本包羅萬象的參考書——其宗旨是為初學的普通用戶提供必要的訊息，幫助你完成日常的工作。而且（我希望能）讓你明白你究竟為什麼要那樣操作。

Steve Cummings（我跟他從沒見過面！）撰寫的許多詞條以及有關 PC 機的所有釋義，確實為本書增色不少。跟他合作是很愉快的事，我們彼此改動刪節對方的文字，完全可以做到不存芥蒂。我很願意能有機會跟他再度合作。

全書體例

這本詞典收進了各類電腦的有關術語。不過，如果你完完全全是個 PC 機用戶，你很可能不會在意 Mac 機的內容，反過來，Mac 機用戶（我也在其中）又未必會對 PC 機的內容感興趣。因此，本書釋義內容的編排充分考慮到了這一點，使你無須多花冤枉工夫。

一段文字前面如果沒有圖標，表示其中內容適用於所有種類的電腦。



在這個圖標後面的段落，其內容專指 Mac 機而言。



在這個圖標後面的段落，其內容專指 PC 機而言。

本書中，“PC 機”一詞泛指自最初的 IBM PC 機及以後、能與之使用相同軟體的電腦。從 1981 年最初的 IBM PC 機推出以來，成百家公司已在 IBM PC 機的基礎上推出許許多多電腦型號。這些電腦無論正式型號叫什麼名稱，都統稱為 PC 機。IBM 公司本身在 1987 年已不再在型號名稱中使用 PC 的字樣，但我們仍把 IBM 公司出品的這些新型號稱為 PC 機，因為它們運行的是 PC 軟體。

同樣，“DOS”一詞指用於 PC 機的任何牌號、任何版本的 DOS。DOS，即 Disk Operating System（磁帶操作系統）是至今為止用於 PC 機的最流行的基本控制軟體（亦即操作系統）。你用的可能是 MS-DOS——最常見的牌號——或 PC-DOS、IBM DOS 或 DR DOS，但說到底，它們的工作方式是大致相同的。這些 DOS 版本之間即使有某些重要的差異（比如說 MS-DOS 2.11 與 MS-DOS 6 之間）——基本命令卻始終是一樣的。

不管怎麼說，希望本書能對你有用。請把你喜歡或不喜歡本書的訊息反饋給我，以便再版時可以修訂得更完善些。

HB 01/01

你知道嗎？……

為什麼螢幕上的標尺不以英吋為單位？參見 resolution。

為什麼你知道硬式磁碟片並未佔滿，卻仍要把“記憶體取出”的訊息暫時寄存一下？參見 RAM。

怎樣自己動手安裝插卡？參見 add-in board。

你的 PC 機上要不要裝 Windows？參見 Windows。

什麼時候紅色和綠色混和會生成黃色？參見 additive color。

為什麼螢幕上有些字形看上去很平滑，有些則並非如此？參見 font。

為什麼記憶體控制面板中沒有應用虛擬記憶體的選項？參見 32-bit clean。你在別人的文檔中見到的那些異型字符，在哪兒能找到？參見 Key Caps (Mac) 或 Character Map (PC)。

為什麼有些字形以城市命名，有些卻不然？參見 fonts (city-named)。

每英吋行數與每英吋點數有什麼區別？參見 lines per inch。

為什麼小寫字母稱為“下盤”字母？參見 lowercase。

怎樣在聯機網路上表示你在咯

咯發笑？參見 checkbox。

我的哥哥長得什麼模樣？參見 propeller head。

你的軟體的真正所有權屬於誰？（不是你！）參見 license agreement。

為什麼在文檔名中不能用冒號？參見 illegal character。

Mac 機機箱上的那個小塑料片是什麼東西？參見 programmer's switch。

一台三色（紅，綠，黃）顯示器怎樣顯示黑色或白色？參見 RGB。

怎樣從螢幕上獲取圖片？參見 screen capture。

用印表機列印的照片為什麼都是由點組成的？參見 halftone。

數據機盒上閃爍的小燈意味著什麼？參見 modem。

發生系統性故障時，怎樣盡可能地恢復？參見 Ctrl-Alt-Del (PC) 或 force quit (Mac)。

電腦怎樣才能具有 PMS？參見 Pantone Matching System。

何時一組悅耳的和弦會表示

出了嚴重故障？參見 Chimes of Doom。

Mac 機開機後，怎樣放映影片？參見 Startup Movie。

為什麼用了 ATM 以後，螢幕上（甚至非 PostScript 印表機上）的字型會變成那麼平滑？參見 fonts (ATM)。

哪些人住在硅谷？參見 Silicon Forest。

在 Mac 機，怎樣使用內置螢幕保護程式？參見 screen saver。

選購 VGA 顯示器時要注意些什麼事情？參見 VGA。

你是否在用 System 6 或 System 7？參見 System 7。

如果電腦運行有些異常，該怎麼辦？參見 virus。

怎樣用鍵盤選取和激活按鈕？參見 Tab Key。

在哪兒能找到手控撥動式開關？參見第 595 頁。

用哪種軟體可以很方便地做成雙欄文字？參見 table。

有哪些事是真正很要緊的？參見 Three Rules of Life。

這麼些縮略詞是從哪兒來

的？參見 TLA。

頁邊 (margin) 和縮格 (indent) 有什麼區別？參見 indent。

看到“悲傷 Mac”圖標怎麼辦？參見 Sad Mac。

為什麼不能把兩台電腦連在同一個外部硬式磁碟片上？參見 SCSI address。

你是否犯了個過失，用了那個讓人一眼就能看出你是非專業水平的符號？參見 smart quotes。

坐在電腦跟前為什麼不能摩挲氣球？參見 antistatic device。

怎樣為 Mac 機文檔製作文具簿？參見 template。

怎樣防止應用程式因記憶體不夠（其實有時還有許多記憶體空著）而造成系統性故障？參見 application heap。

在聯網通訊中，對方無法看見你的表情，那麼怎樣才能交流某些微妙的情緒呢？參見 bawdy language。

怎樣鍵入 pinata 一詞中 n 上方的那一彎？參見 tilde。

為什麼把開機稱為“啟

動”？參見 boot。

怎樣將文檔以某種格式傳送給對方，既能保證對方的程式能打開該文檔，又能保存粗體、斜體及字形選擇等格式化特徵？參見 Rich Text Format。

怎樣在 Windows 中用快捷鍵來選取和激活按鈕。參見 buttons。

怎樣將文檔中的數字送入電腦，或者從電腦送入文檔？參見 Calculator。

怎樣用 Mac 機來談情說愛？參見 Startup Sounds。

大寫鎖定 (Caps Lock) 與打字機上的換檔鎖定 (Shift Lock) 有什麼區別？參見 Caps Lock。

怎樣預防腕部肌腱勞損？參見 carpal tunnel Syndrome。

什麼東西會發生“哞夫！”的叫聲？參見 Dogcow。

僅有一份原始文檔時，為什麼不應對其進行壓縮？參見 compress。

那位墨菲的俏皮理論是否真的算得上生活法則？參見

Murphy's Law。

從哪兒可以看出是誰在主宰一切？參見 motherboard。

有什麼簡單的辦法可以消除 WDEF 病毒？參見 WDEF。

有什麼辦法可以完全無償地略微提高一些電腦運行速度？參見 rebuild the desktop。

本書是怎樣排版印製的？參見 desktop publishing。

有些 ROM 是攪雜的，有些則是淨化的，其原因何在？參見 32-bit addressing。

鐳射印表機是怎樣進行工作的？參見 laser printer。

有什麼辦法可以馬上看出你的 Macintosh 字形是不是下裝的？參見 downloadable font。

跟顯示器的正面、側面或背面離得太近會有什麼問題？參見 ELF。

時至深夜，這個國家的電腦網上在搞些什麼名堂？參見 electronic sex。

目錄

作者前言	ii
全書體例	iii
你知道嗎？	iv
詞條：符號與數字	1
詞條：A 至 Z	13



正襟危坐開始工作之前，Robin 總要先到盥洗室裏去調整一下個性。

SYMBOLS AND NUMBERS 符號和數字

符號

下列符號原則上都按字母順序列作本書詞條。

ALT	Alt（交替）鍵
&	和記號
⌘, ⌘	Apple（蘋果）鍵
*	星號（asterisk；亦作 star 或 splet）
\	反斜槓；參見 directory
•	著重號

SYMBOLS

^	插入記號
^^^	咯咯地笑；參見 bawdy language
:	冒號
⌘	Command (命令) 鍵
⌃	Control (控制) 鍵
•	點 (文檔名中作 dot, 參見 extension; 版本中作 point, 參見 version)
esc	Escape (返回) 鍵
F1, F2, etc.	F 鍵, 功能鍵
⌵	I-beam 形光標
⌥ ⌘	Option (選項) 鍵
	Power On (電源接通) 鍵
#	磅記號, 亦可作數字記號或空格符號
?	問號光標; 參見 online help
⇧	Shift (換檔) 鍵
↵	軟回車
*	星號; 參見 asterisk
.*	星點星
⇥	Tab (製表符) 鍵
	快樂 Mac
	悲傷 Mac
	? (啟動時); 參見 Sad Mac
	X (閃爍)

020 , 030 , 040

它們分別是 68020 , 68030 , 68040 型 Macintosh 晶片的俗稱。參見本段稍後的有關釋義。

1-bit, 2-bit, 8-bit, 16-bit, 20-bit, 24-bit, 32-bit 1 位、2 位、8 位、16 位、20 位、24 位、32 位

“位元”一詞視上下文而含義有所不同。

它可以指電腦中央處理機 (CPU) 的“寄存器”所能儲存訊息的多少，這是衡量電腦速度和性能的一個指標；參見 8-bit computer 。

它可以指處理器用以發送訊息的匯流排 (bus) 的容量大小；參見 8-bit computer 和 8-bit slot 。

它可以指一台電腦可以編址（使用）的記憶體 (memory) 的多少；參見 24-bit 和 32-bit addressing 。

它也可以指顯示器上像素的深度，圖像的色彩和解析度就是由此生成的；參見下面有關灰度和色彩的釋義。

位元 (bit) 是很小的電子信號單位。不論上下文如何，位元數愈大，則意味著功能愈強、速度愈快或色彩愈豐富。

1-bit, 2-bit, 8-bit, 16-bit, 24-bit, 32-bit color 1 位元、2 位元、8 位元、16 位元、24 位元、32 位元色

說到灰度或色彩時，位數表示能夠傳送到熒幕上的像素 (pixels, 亦即點) 中去的訊息位的多少。訊息位愈多，色彩或灰度愈豐富。而且位數愈多，所佔用的記憶體空間也愈多，這就是說，熒幕上的色彩數目並不是由你所用的彩色顯示器（某些 PC 機顯示器除外）的類型，而是由有效視頻記憶體的數量所決定的。在一台彩色顯示器上，位深度 (bit depth) 可取 1 至 32 之間的任一數目，但 8 位元和 24 位元是目前最常見的標準。

Monochrome 單色

一個圖像是 1 位元的，意即只有一位元訊息傳遞到熒幕上的像素中去。該訊息可以開啟（白）或關閉（黑）該像素。所有的 1 位圖像都是黑白圖像。在一台單色顯示器上，像素只能處理這一位數據，因此你看見的只有黑白兩色。（有時候，所謂“黑色”實際上是綠色或黃褐色的。）

Grayscale 灰度

在一台灰度顯示器上，每個像素可以接受 1 到 8 位元數據不等，從而可以顯示 1 到 256 種灰色層次。

如果每個像素有 2 位數據，就有四種可能的開關情形：開／開，關／關，開／關和關／開。每種情形所顯示的灰色層次（其中包括黑色和白色）都是不同的。

如果每個像素有 4 位數據，你就有 16 (2⁴) 種不同的灰色層次。

如果每個像素有 8 位數據，就有 256 種 (2⁸) 可能的組合。這是灰度顯示器所能顯示的灰色層次的上限，其實我們的眼睛也無法辨別更多的層次了。

24-bit, 32-bit color 24 位，32 位色

在一台彩色顯示器上，每個像素都有三位一體的三個點——一個紅點，一個綠點，一個藍點。每個點可以處理多達 8 位元的數據，這樣每個像素總共就可以處理 24 位元數據。三種色點中每種都有 256 種可能的色彩，因而 24 位元彩色就能在顯示器上顯示一千六百萬 (256 的三次方) 種色彩。其中許多色彩差異極其細微，就連眼光最銳利的行家也無法區分。可以這麼說：一千六百萬種色彩已經是綽綽有餘了。（在具有三色點的情形下，怎樣得到黑色和白色？所有的點都開啟時，該像素是白色；所有的點都關閉時，像素是黑色。）

現在，你常常會聽到 32 位元色的說法，其實這只是一種說法罷了。外加的那 8 位元並不另外提供色彩，而只是提供屏蔽和通道的功能而已（參見 channels）。

4 DOS 4 DOS 軟體

4 DOS 是一種軟體產品（自然是用於 DOS 系統的），它能使 DOS 本身變得更易使用並功能更強。舉例來說，有了 4 DOS，你用一條命令就可以複製一張列表上的所有文檔；沒有 4 DOS 的話，你就得分別對每個文檔鍵入複製命令。這個軟體價格不貴，而且是共享軟體 (shareware)，因此你可以先試後買。如果你經常要用標準 DOS 命令，4 DOS 確是值得用的小程式。

從技術角度說，4 DOS 取代的是與 DOS 配套供應的標準“命令處理程式”COMMAND.COM（參見 COMMAND.COM）。

8-bit, 16-bit, 32-bit computer 8 位、16 位、32 位電腦

32 位元（或 8 位、16 位）電腦指的是中央處理機 (CPU)，亦即運行電腦的晶片的能力。可以同時處理 8 位元 (bit) 訊息的 CPU（電腦的最低限度配置）稱為 8 位元電腦。如果 CPU 能同時處理 16 位元訊息，顧名思義，它就稱為 16 位元電腦，32 位元電腦情形也相仿。

但你恐怕未必知道，即使 CPU 本身能同時處理 16 或 32 位元數據，也並不意味著這台電腦能夠處理這樣的訊息。數據經 CPU 處理後，還得通過匯流排 (bus) 送往電腦的其他部件。好多帶有 16 位元處理器 (CPU) 的電腦的外部數據匯流排僅有 8 位寬度，因此實際上不論 CPU 是否 16 位元——反正只有 8 位元數據能被同時使用。（不錯，你可以把匯流排想像成一個帶輪子的小車子，它裝載著訊息駛往各處。）

廣告上號稱 32 位元機的電腦也往往如此；其中有些電腦上外部數據匯流排只有 16 位元寬。

但如今，絕大多數電腦都已裝備了與處理器匹配的數據匯流排。

8-bit slot, 16-bit slot 8 位元槽，16 位元槽



PC 機上有幾種不同的擴充槽 (expansion slot)（電腦內部供加入插卡以擴充性能的接插器）。當初的 IBM PC 機中的槽口一次僅能在母板 (motherboard) 與槽內插卡之間傳送 8 位元 (bit) 訊息；這些就是 8 位元槽。最先具有 16 位元槽的 PC 機是 PC/AT 機，這一標準已為今天的大多數 PC 機所沿用（參見 ISA 和 bus）。16 位元槽實際上在同一行上有兩個分隔的槽口；朝向電腦後背的槽口完全就是個 8 位元槽。換句話說，你可以把一塊 8 位元插卡插在一個 16 位元槽裏（反之不然）。EISA 槽都是 32 位元槽，它們不僅能接受 32 位卡，也能接受標準的 8 位元和 16 位元卡。此外尚有 32 位元 Micro Channel（微通道）槽，它只接受 Micro Channel 卡。亦可參見 Full-length slot。

24-bit addressing, 32-bit addressing 32-bit clean, clean ROMS, dirty ROMS 24 位元尋址，32 位元尋址，32 位元淨化，淨化 ROM，攪雜 ROM

請記住，你的硬式磁碟片 (hard disk) 就是你儲存所有文檔的地方。當你打開一個文檔工作時，電腦就會把該文檔的一個副本放入記憶體 (memory)，這就像從文檔櫃 (硬式磁碟片) 中取出一個文檔放在辦公桌 (相當於電腦的記憶體) 上。

可以把電腦記憶體想像成數以百萬計的許許多多小房間，每個房間都有個地址 (address)。電腦放在每個小房間裏的訊息都是以位元組 (byte) 亦即微小的電子脈衝的形式貯存的。(實際上，每個脈衝僅是一位訊息；一排八位元才是一個位元組。)

24 位地址就是由 24 個數字排在一起組成的地址。這些數字非 1 即 0，分別代表電子脈衝接通或斷開。每個脈衝就是一位元 (bit)。24 位元地址看上去有如這樣：100100111010110001010001。這一長串數字指定了儲存訊息的記憶體晶片中的一個小房間。

你不妨想想看，假如你家的地址是由 24 個數字組成的，那麼即使這些數字只能取 1 或 0，也會有一千六百萬 (2^{24}) 種可能的有效地址。如果電腦的 CPU (中央處理器，即運行電腦的晶片) 採用 24 位元地址，那麼電腦就有可能向一千六百萬個不同的記憶體位置 (亦即地址) 發送或從那兒接收數據。電腦能尋址的記憶體愈多，能運行的項目就愈大，而且能同時運作的項目也愈多。



一千六百萬位元組 (因為每個記憶體位置存放一個位元組) 就是 16 兆位元組，因而你只要安裝適當的記憶體晶片，就可以對 16 兆記憶體進行尋址 (使用) 了。

如果你的電腦的處理器 (晶片) 是用於 IBM PC/AT 和兼容機的 80286，那你就可以對 16 兆記憶體進行尋址。386 和 486 處理器則可進行 32 位元尋址 (用 Windows 3.0 時可尋址記憶體僅限於 16 兆，而 3.1 則可使用全部記憶體)。



但在 Mac 機上，卻有些使人掃興的情況。早期的 Mac 機採用 68000 處理器 (Mac 512, Plus, SE 以及最初的 classic 型號均如此)，只有 4 兆記憶體可用作 RAM (隨機存取記憶體) ——直接與我們有關的記憶體區。其餘 12 兆則編址用於系統其他部件，如母板、顯示器、ROM 等等。從 Mac II 型和 68020 處理器開始，Mac 機僅將 8 兆記憶體用於其他部件，從而電腦可以將

8 兆記憶體用作 RAM。採用 24 位元地址時，你最多只能使用 8MB（兆）記憶體，即便增加記憶體晶片也仍是這樣。（但可借助於 Connectix 公司的 Maxima 和 Optima 等軟體產品擴展記憶體。參見下文。）

32-bit addressing 32 位尋址

一台電腦採用 32 位元地址，就意味著地址中含有 32 個數字。每個數字都可以取 1 或零，於是總共有 40 億 (2^{32}) 個不同的有效地址。4 億位元組（每個記憶體位置存放一個位元組）就是 4096 兆位元組 (megabyte)，亦即 4 千兆位元組 (gigabyte)。因此，採用 32 位地址時，從理論上說可以有 4 千兆位元組的記憶體地址。

不過，目前在硬體製造工藝上還存在某些局限性，因而你並不能當真儲存並使用（尋址）所有這 4 千兆位元組。配有 80386 或 80486 微處理器的 PC 機應該可以對 4 千兆位元組的記憶體進行尋址，但是能擁有這麼多記憶體晶片的 PC 機至今尚未造出。一台標準化 Mac 機（Mac II 系列）“僅”能尋址 128 兆位元組，而有些 Quadras 電腦則能尋址 256 兆位元組。

32-bit clean, clean ROMS, dirty ROMS 32 位元淨化，淨化 ROM，攪雜 ROM

最初的 Mac 機就具有 32 位元尋址能力。但由於其他軟體和硬體的限制，它僅能用到其中的 24 位元（原因在於 24 位元地址匯流排——它只能尋找 1 千 6 百萬，而不是 40 億的地址）。對若干年前生產的 Mac 機而言，這一局限性似乎並不顯得怎麼重要。於是，軟體工程師們就把另外 8 位元（24 位元以外的）在操作系統 (operating system)、ROM 乃至在某些應用程式和擴展程式（初始化模塊）中自行派用場了。正如 Sam Hunting 所說的，這就好比有人用郵政編碼的附加的四個數字去派表示 I♥U！（我愛你！）之類的用場。

後來，Apple 公司開始生產（從 Mac II 起）採用 32 位元匯流排的 Mac 機。這一來挺不錯——硬體（匯流排）就此可以使用全部 32 位元地址了，可是 Mac II、IIx、IIcx 及 SE/30 中的 ROM（軟體，實際上是指 firmware 即固件）仍在把“外加的”8 位元另作它用。當時這個問題還不要緊，因為操作系統畢竟還無須用到全部 32 位元（好比郵局還不需要附加的編碼）。

然後，Apple 公司又有了兩項重大舉措：推出採用 32 位元尋址模式的 System 7；IIcx 之後生產的所有 Mac 機（諸如 ci，fx，si，LC，Classic II，Quadras，PowerBooks，Centris 等等）都配備了新的“淨化”ROM（之所以稱“淨化”，是因為它們不再用另外那 8 位元地址去幹別的事情了）。倘若你有一台這樣的新型電腦，你大概會注意到 Memory Control Panel（記憶

體控制面板) 上有個打開 32 位元尋址功能的選項。這些 Mac 機由於具有淨化 ROM，就能處理 32 位元地址，也就是說可以使用更多的記憶體。(如果你在具有淨化 ROM 的 Mac 機上安裝 8 兆以上的記憶體，必須打開 32 位元尋址選項，否則電腦無法使用超過 8 兆部分的記憶體。)

但由於 Mac II、IIx、IIcx 和 SE/30 都有“攙雜”ROM，你在這些電腦上就無法啟動 32 位元尋址功能——在記憶體控制面板上甚至干脆就沒有這一選項。攙雜 ROM 傳送的訊息，由於這後八位中的附加訊息的緣故，可能會尋錯地址。這一來電腦就會亂了套。這情形好比郵局要用到附加的四位編碼了，結果卻被 I♥U！之類的訊息弄得無所適從。

如果你在具有淨化 ROM 的 Mac 機上採用 32 位元尋址模式，那麼在使用應用程式，尤其是擴展程式 (extension，在 System 6 中稱為 INIT，即初始化模塊) 時仍可能會出岔子 (程式設計人員在擴展程式中利用了那 8 位元附加地址)。這後 8 位元地址中包含的訊息可能引起系統的混亂，甚至會造成系統性故障。程式如果並不使用這 8 位元附加地址，就稱為“32 位元淨化”的。(如果你需要用前面所說的程式，可通過記憶體控制面板關閉 32 位元尋址功能。)

因此，32 位元淨化 ROM 就是這樣一種只讀記憶體晶片 (即裝在電腦裏面無法變動的晶片)，當初軟體工程師在設計時沒有拿其中最高的八位元地址搞過鬼。

32 位元淨化應用程式 (或擴展程式，亦即 INIT) 則是這樣一種應用程式，當初程式設計人員沒有用最後那 8 位元 (即通常作為附加地址的那 8 位元) 地址來編制過稀奇古怪的訊息程式。

要在 Mac II、IIx、IIcx 或 SE/30 上使用 32 位元地址，可用 MODE 32。這是 Connectix 公司免費發放的一種控制面板設備 (cdev)。在有軟體可供下裝的佈告牌 (bulletin board) 和線上服務 (online service) 上都可得到 MODE 32，或者你也可以與當地的用戶協會 (user group) 聯繫或打電話 800.9505880 給 Connectix 公司。該公司還有其他極棒的軟體和名叫《Mac 機記憶體指南》的極好的小冊子 (甚至可免費奉送) 專門介紹 Mac 機記憶體 (其特點，如何選擇等等)。

24-bit, 32-bit memory 24 位元、32 位元記憶體

24 位元或 32 位元記憶體，跟 24 位元或 32 位元尋址實際上是一回事，因為尋址就是在記憶體中尋定地址的意思。

24-mode, 32-bit mode 24 位元模式，32 位元模式

24 位元模式指電腦採用 24 位元地址，32 位元模式的電腦則採用 32 位元地址。

32.bis , 42.bis

參見 V 部的 V.32 和 V.42 詞條。這些術語都指遠程通訊 (telecommunication) 標準。

32-bit QuickDraw 32 位元 QuickDraw 程式



32 位元 QuickDraw 亦稱彩色 QuickDraw (color QuickDraw)，這一程式能賦予 Mac 機極佳的色彩性能（從而使顯示器也顯得具有更高的分辨率）。最初的 QuickDraw 版本僅能與 8 種顏色（而不是 8 位色）配合工作，而現在的 Color QuickDraw 版本已能與一千六百七十萬種顏色配合工作。Color QuickDraw 是新型 Mac 機 ROM（只讀記憶體）中內置的例行程式。它也可以作為一種系統擴展程式 (extension)。

286、386、486 晶片



這幾個數字 286、386、486，分別是 80286、80386、80486 處理器 (processor，即運行電腦的晶片) 的簡稱。其定義詳見本節下文。

300，1200，2400，9600 bps 或波特 (baud)

這幾個數字都既可指電腦序列埠 (serial port)（或印表機及其他電腦外圍設備的序列埠）的速度，也可指數據機 (modem) 的數據傳送速率，但許多數據機的速率都以 bps (bits per second，即每秒數據位數) 為準，波特 (baud) 的準確度相對來說較低。有關波特的定義詳見 baud。

680x0 晶片



680x0（讀若“sixty-eight oh x oh”）系指下面列出的整個 Motorola 微處理器系列。680x0 不論寫起來還是說起來，都要比“68000，68020，68030 或 68040”省事得多。

68000、68020、68030、68040 晶片



這幾個數字代表 Motorola 公司為 Mac 機設計的一系列晶片 (chip)。每台 Mac 機都採用其中一種晶片來運行電腦。由於這種晶片是運行電腦的，所以它們稱為中央處理機或 CPU，有時也稱為微處理器 (microprocessor) 或簡稱處理器 (processor)。一般來說，數目愈大，電腦運行速度愈快，性能也愈好（不過