推薦序

很多生命科學領域的研究者和學生，當初選擇生命科學，就是因為不想碰數學。因此他們對使用電腦從事數據分析，往往有著莫名的恐懼。這種現象在二十一世紀的今天，演變成生物學研究的一個阻力。因為現在的科技，可以在極短時間之內產生非常大量的基因定序或是基因表現數據。沒有生物資訊分析，這些實驗數據大概只能靜靜地躺在硬碟的一角，沒有重見天日的機會。這本書可以當做生命科學學生進入生物資訊領域，使用生物資訊方法的第一本書。作者從什麼是電腦，如何採購一台適合的電腦開始，逐步帶領對資訊科學完全陌生的讀者，進入生物資訊的世界。

生物資訊是一個很廣泛的領域，一本入門小書當然沒有辦法介紹所有主題，所以作者李鎮宇決定把重點放在一個近年來非常熱門的領域，那就是新世代的高通量基因定序。和市面上的教科書不同，這本書採取的策略是，「如何第一次分析高通量基因定序就上手」。讀者不必擔心會被太多理論所淹沒，反而會驚訝於原來在自己電腦上安裝分析基因定序的軟體是多麼的簡單。他們會發現，以後再也不必低聲下氣的去懇求生物資訊學者，來分析自己產生的大量實驗數據。

對於生物資訊領域的學生，你也不要小看這本書。作者告訴你最新的電腦技術，就是利用繪圖卡來輔助生物資訊研究。對於繪圖卡相關的生物資訊應用，我在國內還沒有看過中文的介紹。本書作者先從概念上簡介什麼是繪圖卡輔助計算，再帶領讀者建立一個自己的程式開發環境。對有志於程式開發的生物資訊學生是一大福音，相信可有效的推廣將繪圖卡應用在高效能計算的觀念。最後本書還會教你怎樣建立生物資訊分析流程，怎樣把好幾個單獨的軟體串接成一個完整的，不需要人工干預的流程。生物學者會發現，他們再也不用三更半夜跑到實驗室做實驗了。只

碁峰 www.gotop.com.tw

指令：mv

作用：移動檔案

Move 即移動、搬移，指令是 mv。應該不難想像他的用途就是把檔案搬遷到其他資料夾。不意外，指令是：

```
mv <檔案名稱> <資料夾名稱>
```

例如把 seq1.fastq 這個檔案搬到 original 這個資料夾內，就是：

```
mv seq1.fastq original/
```

有沒有發現一件事，不管是檔案或是資料夾的操作，如何都沒有「更改名稱」。難道沒有 Rename 的相關指令嘛？答案一樣是這個 mv，改檔名的用法變成：

```
mv <原始檔案名稱> <新的檔案名稱>
```

資料夾也是這樣的模式。例如把 original 的檔案夾改叫 raw，就是「`mv original/raw/`」，即使裡面有檔案也沒關係。

下面將示範如何把改名後的 raw 資料夾內，裡面的 seq1.fastq 改名為 seq1.fastq.bak。指令為：

```
mv seq1.fastq seq1.fastq.bak
```

這一連串的操作，看圖最清楚了。

碁峰 www.gotop.com.tw

```
eric — eric@cctv: ~/bookExample/rawData/raw — ssh eric@cctv.eric.lv — 80×24
eric@cctv:~/bookExample/rawData$ ls
original  seq1.fastq
eric@cctv:~/bookExample/rawData$ mv seq1.fastq original/
eric@cctv:~/bookExample/rawData$ ls
original
eric@cctv:~/bookExample/rawData$ ls original/
seq1.fastq
eric@cctv:~/bookExample/rawData$ mv original/ raw/
eric@cctv:~/bookExample/rawData$ ls
raw
eric@cctv:~/bookExample/rawData$ cd raw/
eric@cctv:~/bookExample/rawData/raw$ ls
seq1.fastq
eric@cctv:~/bookExample/rawData/raw$ mv seq1.fastq seq1.fastq.bak
eric@cctv:~/bookExample/rawData/raw$ ls
seq1.fastq.bak
eric@cctv:~/bookExample/rawData/raw$
```

(1) 首先用 `ls` 指令確認目前操作中的資料夾內所有的檔案還有資料夾，之後把 seq1.fastq 移到 original 資料夾裡面。

(2) 馬上再來一次 `ls`，看看是否移動成功。

(3) 列出 original 資料夾的內容（`ls` 還可以這樣用，直接針對特定的子資料夾顯示所有內容）。

(4) 把 original 資料夾改成叫 raw，再列出目前的資料夾內狀態（到目前為止都還在 bookExample/rawData 這個資料夾裡面）。

(5) 用 `cd` 指令切換到 raw 的資料夾內，然後用 `ls` 列出所有項目。

(6) 最後把唯一的檔案 seq1.fastq 用 `mv` 指令改名為 seq1.fastq.bak。

檔案的操作基本上就只有這樣，你應該又困惑了：如何沒有複製貼上的功能？這不是我不教你，是 Unix 架構的作業系統本身的特性就是這樣。完全是安全考量跟實際需求，尤其是配合權限的設計之後。檔案並不能讓你隨意剪貼，需要時請用搬移的功能。因為亂剪貼，你可能不知道貼去哪了。不信你看 Mac OS 作業系統的滑鼠右鍵功能表也沒有剪下的功能對吧？

碁峰 www.gotop.com.tw

3.3 關於品質分數的標準

早在 1992 年的時候，就已經有定序品質的概念。到了 1995 年，Bonfield 還有 Staden 發表了一個以對數概念為基礎的錯誤率評估標準，至今已是各個定序平台間的互通的準確度比較參考，也就是 Phred Score。如果你得到了一組 FASTQ 格式（FASTQ 格式也已經是標準）的序列檔案，裡面除了序列之外也包含了品質分數。一個基本的 FASTQ 檔案，用文字編輯器或者 cat 指令印出來，內容長這樣：

```
@SEQ_ID
GATTTGGGGTTCAAAGCAGTATCGATCAAATAGTAAATCCATTTGTTCAACTCACAGTTT
+
!''*((((***+))%%%++)(%%%%).1***-+*''))**55CCF>>>>>>CCCCCCC65
```

看起來好像沒有分數啊！（還有另一種沒有帶分數的序列檔案叫 FASTA）事實上在加號後面那一行，雖然沒有直接用數字呈現分數，但是每個字元其實就是第二行各鹼基對照下來的品質分數，理論上有多少個鹼基下面就要有同樣數量的對應字元。應該也有人看出來了（尤其是資訊背景的你），這些字元代表 ASCII 的編碼裡面轉換的十進位數字。

所謂的 ASCII 碼，是指美國信息標準交換代碼（American Standard Code for Information Interchange）。是一套在六零年代末期就已經發展出來的電腦編碼，主要就是把我們常用的英文字母、標點符號還有特殊字元如@、＃等，賦予一個二進位、十進位、十六進位的數字。你現在用的鍵盤，上面的每一個字元都有一個編碼。ASCII 編碼表，每一本計算機概論課本都有，或者可以翻本書最後的附錄。如果你真的好奇的查起來上面範例裡面出現的字元，! 就是 33、* 為 42、＋是 43，這樣應該可以感受想到用 ASCII 編碼定義品質分數的人實在很聰明。用一個字母取代，至少可以省下一半的資料傳輸量（品質分數都用十進位十位數顯示的話，就要用兩個數字字元）。

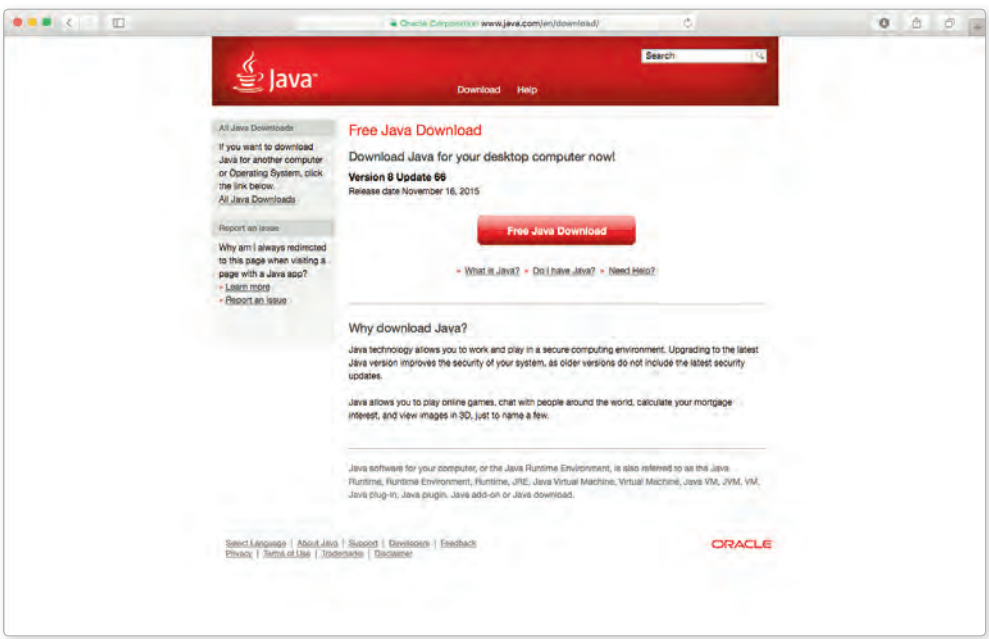
碁峰 www.gotop.com.tw

3.4 實戰品質校驗

通常檢體（Sample）送生技公司或者自己用實驗室的高通量定序平台定序後，會得到 FASTQ 檔案，我們可以把這個檔案送給軟體分析，然後產出關於這個序列的概要報告並畫出統計圖，幫助判斷定序的資料品質好不好。針對兩種狀況，介紹兩套用來進行品質分析的軟體。

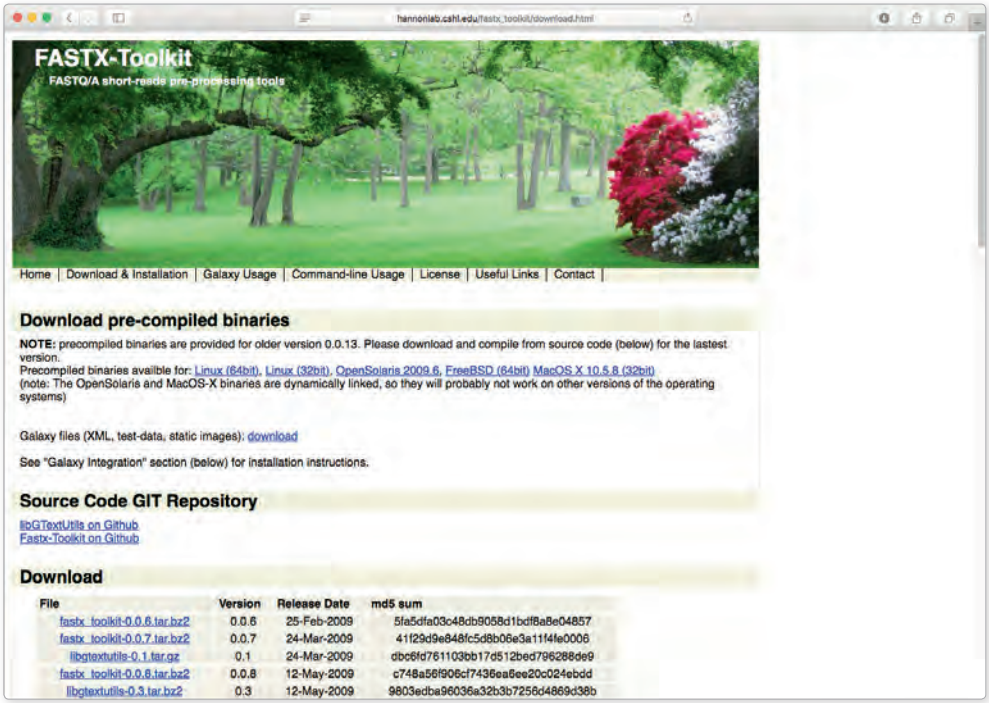
第一套是 FastQC，這是一個用 Java 開發的軟體，可以接受 FASTQ、BAM 還有 SAM 檔案（後兩者會在序列比對的章節介紹），接著產生完整的報告，不管要數字或是統計圖都有。

要在你的 PC 或是 Mac 桌面環境使用 FastQC，請先確認你的電腦有安裝 Java 虛擬機器。OS X 近年來基於安全的問題，已經不隨作業系統預載 Java。所以不論 Windows 或 Mac 的使用者請上 Oracle 的網站下載（http://java.com/download）。



Windows 或 OS X 的使用者如果沒有安裝 Java 虛擬機器，可以到官方網站（http://java.com/download）免費下載並安裝。

碁峰 www.gotop.com.tw



FASTX-Toolkit 的下載頁面：http://hannonlab.cshl.edu/fastx_toolkit/download.html

在 FASTX-Toolkit 的下載頁面可以看到一開始是 Download pre-compiled binaries，並且還有幾個版本的作業系統。光是 Linux，就有分成 64 位元還有 32 位元的版本。

這是如何一回事？對電腦系統跟軟體架構不熟的人而言，的確是需要先教育。64 位元架構的處理器如 Core i3、Core i5、Core i7、Xeon 還有 AMD 的 Athlon，是目前電腦系統的主流，本身也向下相容 32 位元。但是 32 位元沒有辦法執行 64 位元的應用程式。至於電腦安裝的作業系統，本身也有 64 或是 32 位元的區別。一般而言，現在都會直接安裝 64 位元的版本。

然後在設計軟體的時候，程式設計師辛苦完成了一個軟體專案，必須要告訴電腦系統如何用正確且有效率的方法執行。軟體專案通常是原始碼（Source code），打開來就可以看到程式是如何寫的。然後每一種電腦系統上，都會有一個編譯器（Compiler），負責把這些原始碼轉換成機器能夠執行的結果，這個動作叫編譯（Compile）。大家都知道電腦的世界只有 0 與 1，也就是二進位的架構，轉出來的

碁峰 www.gotop.com.tw

總而言之，序列比對的工具很難有絕對的好壞。讀者就按自己的訴求還有該工具設計的導向決定是否使用吧！例如有些序列比對工具的對象是較小的基因體，例如酵母菌，如果你的研究是人類基因體可能就不太適合。

到了 2015 年，Rahman 等人發表的論文 benchNGS: An approach to benchmark short reads alignment tools 中，選擇被評比的工具與 2011 年論文的項目幾乎無異，筆者認為這些比對工具的趨勢已經進入「大者衡大」的時代，只有老牌的工具普遍被研究人員接受。新的工具幾乎沒有生存空間。所幸該論文評比後的結果是各家工具的表現都不錯，所以現在使用序列必對工具倒也不用太計較，選自己上手的來用就好。

表 4-1：現有高通量定序序列比對工具

名稱	主要演算法	網址	備註
Bowtie	BWT	http://bowtie-bio.sourceforge.net/index.shtml	
Bowtie2	BWT	http://bowtie-bio.sourceforge.net/bowtie2/index.shtml	改用 FM index 為索引格式
BWA	BWT	http://bio-bwa.sourceforge.net/	
mr(s)FAST	Seed-and-extend	http://mrfast.sourceforge.net/	
Novoalign	Hash	http://www.novocraft.com/	付費商業軟體
SHRiMP	Smith-Waterman	http://compbio.cs.toronto.edu/shrimp/	主要比對 colorspace 序列
SOAP2	BWT	http://soap.genomics.org.cn/soapaligner.html	未開放原始碼

BWT 演算法編解碼示意

目前有許多序列比對工具的核心都使用 BWT 演算法壓縮資料，然後再加上一些小地方的改變，實作高速的序列比對。這個小節將說明 BWT 主要的工作原理。以使用 BWT 演算法編碼 ^BANANA@ 這個字為例（要考慮頭尾，^ 是頭 @ 是尾），

碁峰 www.gotop.com.tw

要存檔成 hg19，指令是 `./bowtie-build ../UCSChg19/hg19.fa hg19`，執行之後就開始建立索引。會在目前的 bowtie 根目錄下建立一連串 hg19 開頭的索引檔案，因為我沒有另外指定路徑。

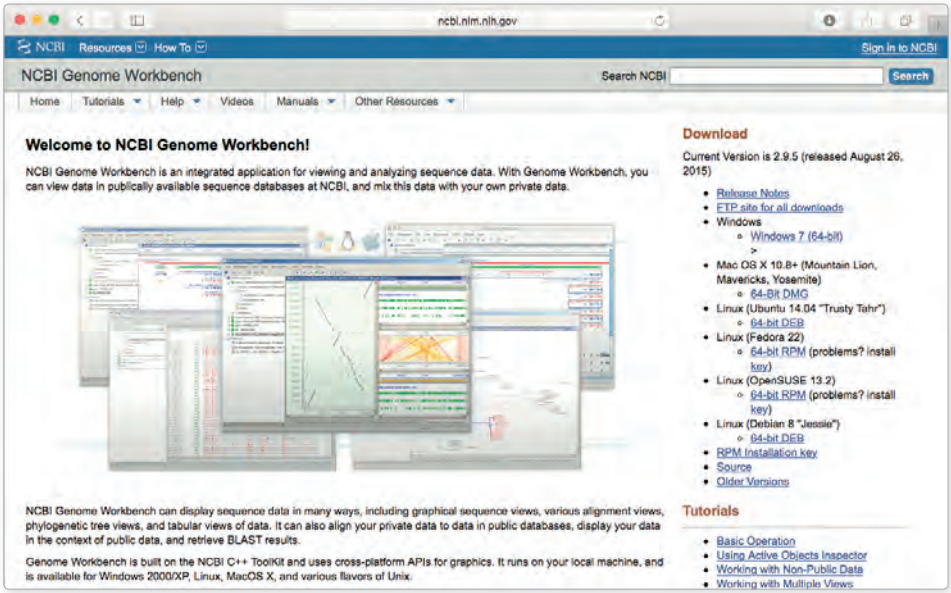
```
bowtie-1.1.2 — -bash — 80×24
    offMask: 0xffffffe0
    isaRate: -1
    isaMask: 0xffffffff
    ftabChars: 10
    eftabLen: 20
    eftabSz: 80
    ftabLen: 1048577
    ftabSz: 4194308
    offsLen: 90540952
    offsSz: 362163808
    isaLen: 0
    isaSz: 0
    lineSz: 64
    sideSz: 64
    sideBwtSz: 56
    sideBwtLen: 224
    numSidePairs: 6467211
    numSides: 12934422
    numLines: 12934422
    ebwtTotLen: 827803008
    ebwtTotSz: 827803008
    reverse: 0
Total time for backward call to driver() for mirror index: 01:23:23
Erics-Mac-mini:bowtie-1.1.2 eric$
```

建立索引的時間跟原始基因體參考序列的大小還有電腦性能有關，我大概花了快一小時半才建立完成。記得要看到 Total time for backward call to driver() for mirror index 的字樣出現，並出現新的命令輸入提示字元才算完成。

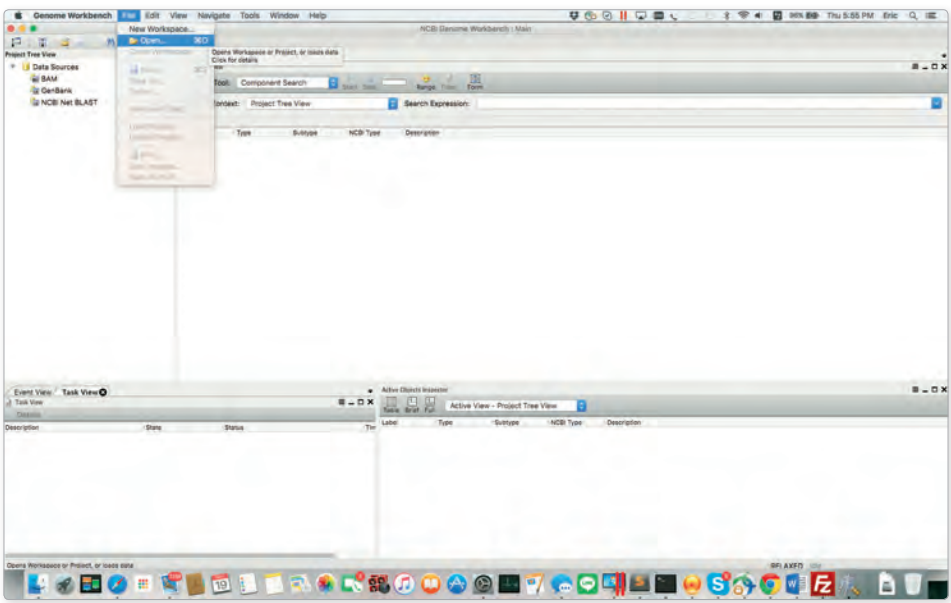
表 4-2：Bowtie 已經預先建立好的參考序列索引

參考序列名稱	物種	參考序列名稱	物種
H. sapiens, UCSC hg18	人類	B. taurus, UMD v3.0	黃牛
H. sapiens, UCSC hg19	人類	C. familiaris, UCSC canFam2	家犬
H. sapiens, NCBI v36	人類	G. gallus, UCSC, galGal3	家雞
H. sapiens, NCBI v37	人類	D. melanogaster, Flybase, r5.22	黑腹果蠅
H. sapiens, NCBI GRCh38	人類	A. thaliana, TAIR, TAIR9	阿拉伯芥
M. musculus, UCSC mm8	家鼠	C. elegans, Wormbase, WS200	線蟲
M. musculus, UCSC mm9	家鼠	S. cerevisiae, CYGD	釀酒酵母
M. musculus, NCBI v37	家鼠	E. coli, NCBI, st.536	大腸桿菌
R. norvegicus, UCSC rn4	褐鼠		

碁峰 www.gotop.com.tw

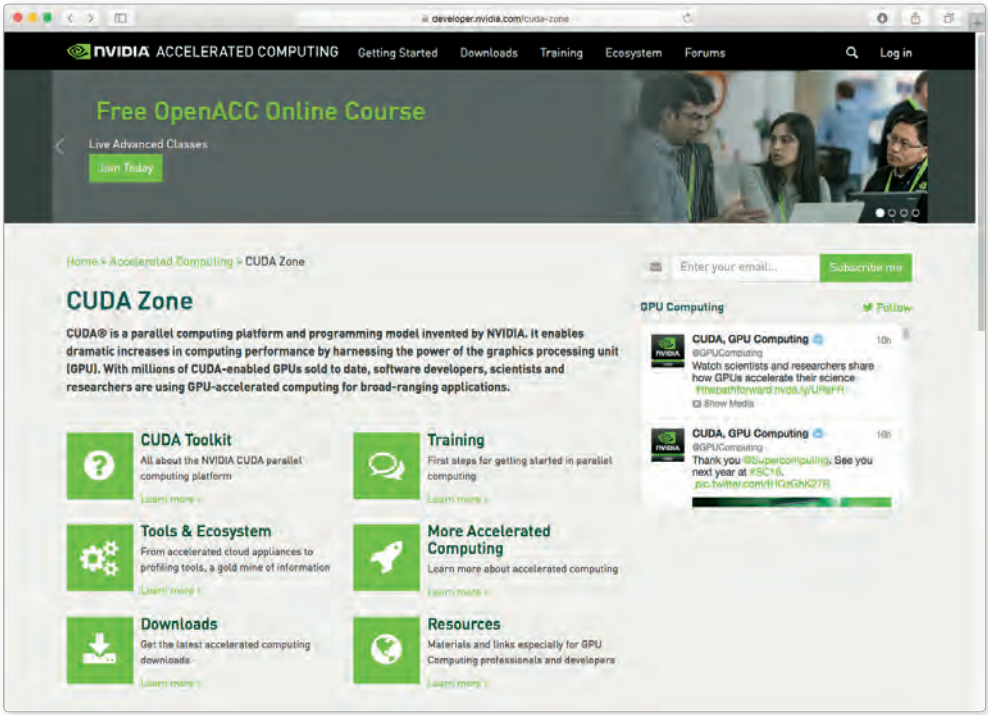


NCBI 的 Genome Workbench 網頁（www.ncbi.nlm.nih.gov/tools/gbench），右邊有各個作業系統適用的 Genome Workbench，請下載自己需要的版本。我後面使用 Mac 平台的版本示範，但是其他平台的 Genome Workbench 用起來都一樣。



檢查網路連線後，主程式開啟。我們要開啟之前轉好的 BAM 檔，請從檔案（File）的功能表選擇開啟檔案（Open）。

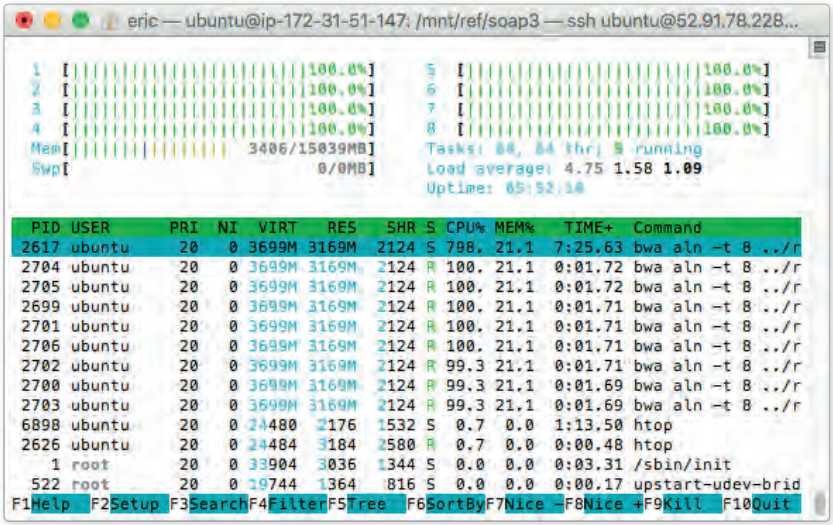
碁峰 www.gotop.com.tw



NVIDIA 網站的 CUDA 開發者專區，提供了完整的 CUDA 驅動程式與開發套件、程式範例。加上實用的教學課程，讓有需要的人可以駕馭 GPU 帶來的平行運算。

以 Ubuntu 14.04 LTS 平台的 CUDA 7.5 環境建置為例，一開始先檢查繪圖卡的硬體是否有正確被偵測到。輸入指令 `lspci | grep -I nvidia` 就能看到是否有 NVIDIA 的繪圖卡正確安裝。

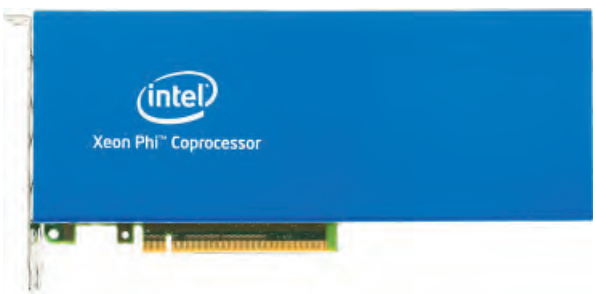
碁峰 www.gotop.com.tw



使用一般的多核心處理器，8 個 Xeon 核心一次全部使用，傳統的序列比對工具不見得會輸 GPU。

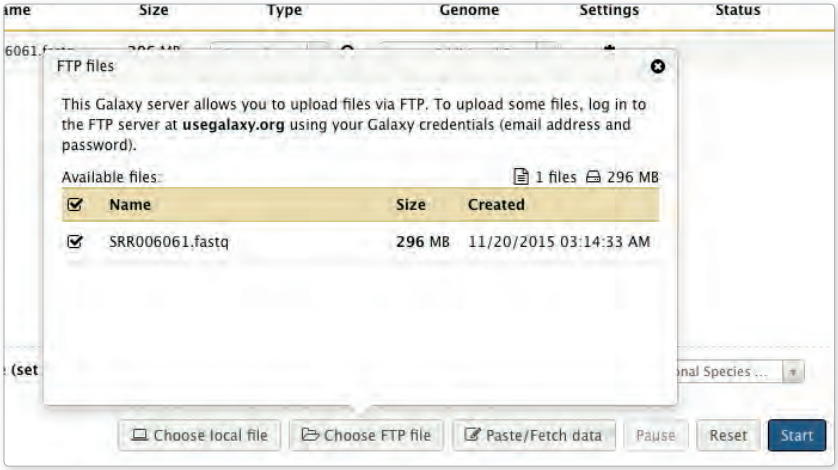
截至 2015 年底，已經有不少雲端基礎設施服務平台（Infrastructure as a service, IaaS）提供按時計價的 CUDA 硬體運算服務。如果不敢貿然投資 CUDA 硬體，花點錢用外面的公有雲測試或暫時少量計價使用也不失為解決之道。

本章最後也跟讀者介紹 Intel Xeon Phi 這個主打超級電腦的產品。原理類似 NVIDIA 的 CUDA，只是把改成用多核心處理器實作。完稿前最新一代的產品預計 2015 年末推出的，高達 72 核心，內建 16GB 的記憶體。從上面筆者實測 CPU 對 GPU 的結果，就能推測這東西威力驚人，肯定超越 GPU。不過價格當然不便宜，目前上市的舊版一片官方報價超過十三萬台幣，新版估計相去不遠。

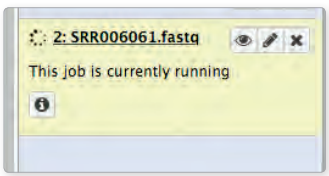


產品定位是「協同處理器」的 Intel Xeon Phi，主要用途是在商業與科學的計算。現在「大數據」的應用，估計 Xeon Phi 也能出不少力。

碁峰 www.gotop.com.tw

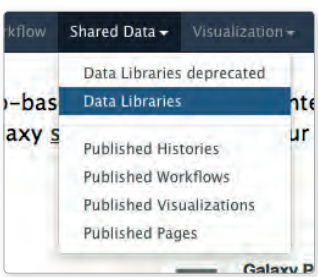


待 FTP 上傳完成之後，在原本 Get Data 的頁面（記得要先登入），就可以看到用 FTP 上載的檔案。勾選之後按下 Execute 的按鈕，就會匯入到工作區內。



開始匯入 FTP 上傳的檔案，Galaxy 主畫面最右邊會用黃色顯示正在作業的進度。

關於資料的取得，最後再介紹一種方式，適用於手上沒有序列資料，又想要實際操作 Galaxy 平台的人。方法是使用 Galaxy 平台上被分享出來的檔案（通常是由官方放上去的），匯入到自己的工作區中。這個方式速度最快，完全不需要上傳。讀者實際操作時可能會跟我的範例不太一樣，因為不見得平台的管理者都會放上同樣的檔案。即使是 Galaxy 官方的主站，哪一天資料會被下架也說不定。



從 Galaxy 主畫面最上面的 Shared Data 裡面，找到 Data Libraries 的項目，用滑鼠點選進入。

www.gotop.com.tw