行政院國家科學委員會專題研究計畫　期中進度報告

一個高效能生物網格環境與其入口網站之研製(2/3)

計畫類別：個別型計畫
計畫編號：NSC94-2213-E-029-002-
執行期間：94 年 08 月 01 日至 95 年 07 月 31 日
執行單位：東海大學資訊工程與科學系

計畫主持人：楊朝棟

計畫參與人員：陳松毅、韓祖棻、傅俊賓、陳俊仁

報告類型：精簡報告

處理方式：本計畫可公開查詢

中　華　民　國 95 年 6 月 1 日

行政院國家科學委員會補助專題研究計畫 □成果報告 ■期中進度報告

一個高效能生物網格環境與其入口網站之研製(2/3)

計畫類別：■個別型計畫　　□整合型計畫
計畫編號：NSC 94－2213－E－009－002－
執行期間：　94 年　8 月　1 日至　95 年　7 月 31 日

計畫主持人：楊朝棟
共同主持人：
計畫參與人員：陳松毅、韓祖棻、傅俊賓、陳俊仁

成果報告類型(依經費核定清單規定繳交)：■精簡報告　□完整報告

本成果報告包括以下應繳交之附件：
□赴國外出差或研習心得報告一份
□赴大陸地區出差或研習心得報告一份
□出席國際學術會議心得報告及發表之論文各一份
□國際合作研究計畫國外研究報告書一份

處理方式：除產學合作研究計畫、提升產業技術及人才培育研究計畫、列管計畫及下列情形者外，得立即公開查詢
□涉及專利或其他智慧財產權，■一年□二年後可公開查詢

執行單位：東海大學資訊工程與科學系

中　華　民　國　　95 年　　5 月　31 日

一個高效能生物網格環境與其入口網站之研製(2/3)

High Performance Computing Lab

Department of Computer Science and Information Engineering

Thughai University

目　錄

前　言

歷史背景：

計算機科學(Computer Science)的發展，為人類在科學領域的探索上，提供了許多十分強韌的研究方法與實驗工具。然而，隨著科學家們所探討的問題日趨於複雜，所需考慮的各種內外情況更加之稠密，無形之中便增加了龐大的計算量。因此若要提升分析問題之效率，如何更進一步地提高工具的計算效能，將是未來科學發展所必然要面對的重要課題。

早期科學家對於高度計算能力的需求，大多仰賴計算能力強大的超級電腦(Super Computer)。然而由於超級電腦造價昂貴，往往大為降低許多規模較小之學術研究單位的購買意願。然而藉由與其他擁有超級電腦的單位合作研究，所必須付出之費用與心力亦不低，常常成為令人十分頭痛的問題。

然而自從 PC 叢集(PC - Cluster)的問世，讓一些面臨經費短缺的單位看到了希望。PC 設備的成本低廉，加上具有結合多台機器計算資源的優異特性，藉此產生強大的計算效能，試問，這麼實惠的計算工具，誰不躍躍欲試？

近幾年來，探討計算效能的領域更興起了一股「網格熱」。由於網格計算相似於 PC 叢集計算，尤其「可結合性質相異之機器」這項不同於 PC 叢集計算的特性，無形中提升了網格計算的研究價值。

何謂網格計算：

網格計算(Grid Computing)目的是用來整合大型網路環境下個各種資源(如處理器、記憶體、磁碟空間、其他可用資源等)，進而利用這些整合起來的資源作大量資料的分析或是高效能的計算應用，像是資料採礦或者是基因的演算等。

以目前的企業來說，其企業內部的電子計算機相關資源通常沒有做到完善的整合，一個公司內可能有便宜的伺服器，快速的區域網路，網際網路，以及分佈於各地的各種資源等等。而集合分散的運算資源之外，Grid Computing 能夠經由網路管理組織內任何一個可使用的運算資源，進而降低伺服器的閒置時間，提升資源的使用效率。

過去網格運算多用於學術研究中，目前業界除昇陽、IBM 力推網格運算技術應用於企業市場之外，Microsoft、企業軟體供應商甲骨文亦踏入網格計算的應用

領域。對企業而言，只需運用現有科技成本的一小部份，即可有效運用且聯結網路現有設備的資訊資源，進而獲得更為強大的運算能力，當然是企業所必須致力追求的目標。在此架構之下，企業可直接取得電腦、資料、軟體，或者是儲存設備。此外，對於誰可以分享、可分享哪些資源、什麼條件可以允許分享等，網格計算同樣提供了嚴格的控管機制與清楚的角色界定。

另外一方面，叢集計算(Cluster Computing)產生的主因是因為高速乙太網路的普及和個人電腦價格的滑落。藉由利用高速乙太網路建置相互連結的架構，讓價格低廉的個人電腦或是小型電腦做相互的聯結，整合成一組虛擬的"計算平台"，並在上面進行高效能計算或資料分析等問題，這就是我們所知的叢集系統(Cluster system)。叢集計算的環境在建置上傾向集中式管理，將所有的計算資源作整合，再藉由高速網路的便利而讓需要高效能或是需即時計算的問題，能夠以最快的時間計算出結果或是提供高速且巨量的資料分享。

縱使目前叢集計算的環境可以獲得很好的計算效能，但叢集計算平台需要集中式的管理和建置，所使用之機器的性質亦須相近(例如:整套叢集系統其處理器必須全是 Pentium III 架構，而不容許有一顆 Pentium II 的存在)，因此無法將這樣的計算平台和資源供予網路上其他的資源分享。

事實上，網格計算的概念是建構在叢集系統之上的，其不但具有叢集計算的優點，更具有整合「異質」機器的特色，這使得我們能夠以更簡單的方法與更低廉的研究成本，簡易地得到經過整合而產生強大計算能量的網格環境，進而替我們解決更多複雜的科學問題。早在 10 多年前，網格計算已在高速運算領域多獲驗證。過去研究單位執行工程運算時，多採用矩陣、向量計算方式；乃至後期大型主機內部 CPU 排列方式，皆採棋盤與網格方式處理。

詳細來說，網格計算其實可以視為幾個主要概念的構成：

資源分享

網格第一個概念是「分享資源」。當你進入網格去使用遠端資源，這些遠端資源將協助你去完成一部電腦無法完成的任務。比如說一項很複雜的數值模擬，不只簡單地檔案交換，還包括遠端存取軟體、電腦及資料。甚至存取遠端感測器、望遠鏡及所有權不屬於你的其他設備。 網格的難題：並非無條件獲取資源或給予資源。實際上，應該建立一個資源的使用機制，資源提供者應該決定何種使用者可以被信賴，同時也應該建立使用者的使用權限。

安全存取

存取政策 (Access policy)：資源提供者及使用者，必須清楚定義何種資源可被分享？ 誰可以分享？及分享的條件？
認證 (authentication)：需要建立一套機制，確定使用者及資源的身份。
授權 (authorization)：需要一套機制，決定此操作是符合共同定義的分享關係。

資源使用

網格的第三個概念是有效利用資源。因為無論你擁有多少電腦資源，都還是會花費許多時間在排隊等待使用資源。但如果有一個機制，可以有效率地自動分配工作到不同的計算資源，便可以減少排隊等待的時間。在網格中，可以經由適當的資源分配，安排使用者執行工作。其中中介軟體便是擔任分配資源的角色，所以開發此軟體也是目前世界上網格計畫主要的發展方向。

零距離

高速網路的連接使全球網格變為可能。十年前，將大量資料傳到其他速度更快的電腦去處理，是不明智的行為，因為緩慢的傳輸速度，會讓快速的處理效能變得毫無意義。

由於「距離」問題永遠存在，這表示「速度」也會是個永久性的問題。有些科學家為分析龐大資料，需要每秒數十億 (gigabits) 的傳輸速度來配合。但另外有些科學家，因為要執行某些需要「處理器間一致通訊」的複雜計算，且為了即時確定處理器間的通訊內容，反而需要極慢的傳輸速率來配合。

為確認網格資料的即時傳輸，需要處理器間一致的通信。而為了避免通訊瓶頸，網格發展者必須掌握錯誤發生時的補救方法。比如說當計算錯誤、傳輸錯誤或是電腦當機時，所應有的補救方法。想要達到上述需求，必須解決許多高效能網路問題。這包括了傳輸通訊協定的適當性，及發展高效能乙太網路交換的技術。

開放標準

開放標準，可以協助將各種網格應用程式，相容於其他的網格區域。這或許過於理想化，畢竟各軟體公司為了自身利益，不願意讓他人分享標準。然而，網格的本質就是共享，每個人應在不違反自身利益的原則下，設立共同開放的標準。

PART I Biozilla

前言

本部份分為三個主要的章節：RRS Service-日誌網路服務、XML Parser of Bio Software 與 Biozillia Family。

我們在 Web Service 方面開發了「RSS Service-日誌網路服務」，可以給程式或是使用者輕鬆建立屬於自己的 RSS。

生物軟體處理方面，則開發了負責處理各輸入輸出檔的 XML Parser 供作資料轉換再利用。

Biozilla Family 則針對使用生物資訊軟體方面，開發了一系列輕鬆、容易上手的使用介面與工具。

1. RSS Service-日誌網路服務

1.1 關於 RSS Service

RSS Service 是一個 Web Service。可以應用在 Web Application、個人用戶、SOP 上訂做自己的 RSS Feed。

1.2 什麼是 RSS

RSS 的英文全稱是 Really Simple Syndication，是一種透過 XML (eXtensible Markup Language)特性所制定的格式，讓網站的管理者可以把網頁內容傳給訂閱戶。這是個有點像電子報和新聞群組(Newsgroup)的東西，但是賦予讀者更大的自訂能力和更豐富的資料。

將 RSS 技術應用在獲取來源端的即時訊息是一個很重要的演進，對於使用者也可以很清楚的知道這些新聞是從何而來，以及這些資訊是否對自己是有用的。以下是 RSS 發展的優點：

一	即時性(Timely)： 對於 RSS 的訂閱者而言，可以最快的得到最新訊息以及頭條新聞。而不用被動式的去每個網站上去搜索。
二	具有成本效益(Cost-effective)： 在傳輸和發送的成本減少是很巨大的。如對於新聞郵件的發送提供者不需要花費太多的金費，對每個訂閱者來寄信散撥訊息。
三	統一的標準： RSS 有其一定的標準定義的<Tag>，有提供 RSS 的網站都依循此標準，不但可以方便解讀以及管理。
四	RSS 可整合電子郵件： 在透過 RSS 等軟體可以將拿到的 RSS 訊息完美的轉換成你的電子郵件的格式。這也意味著訂閱者會依照自己的偏好來訂閱，並且也可避免電子郵件的垃圾信和病毒。
五	隱私性和安全性： 對於訂閱者而言，並不需要提供自己的電子信箱；而發行者並不能利用電子郵件重複不斷的寄廣告信或是垃圾

	信件。RSS 代表著不能不正當地使用網路來作為廣播媒體傳送相同的訊息給大量未要求傳送訊息的使用者的人，對於訂閱者而言是另外一種的安全以及隱私。

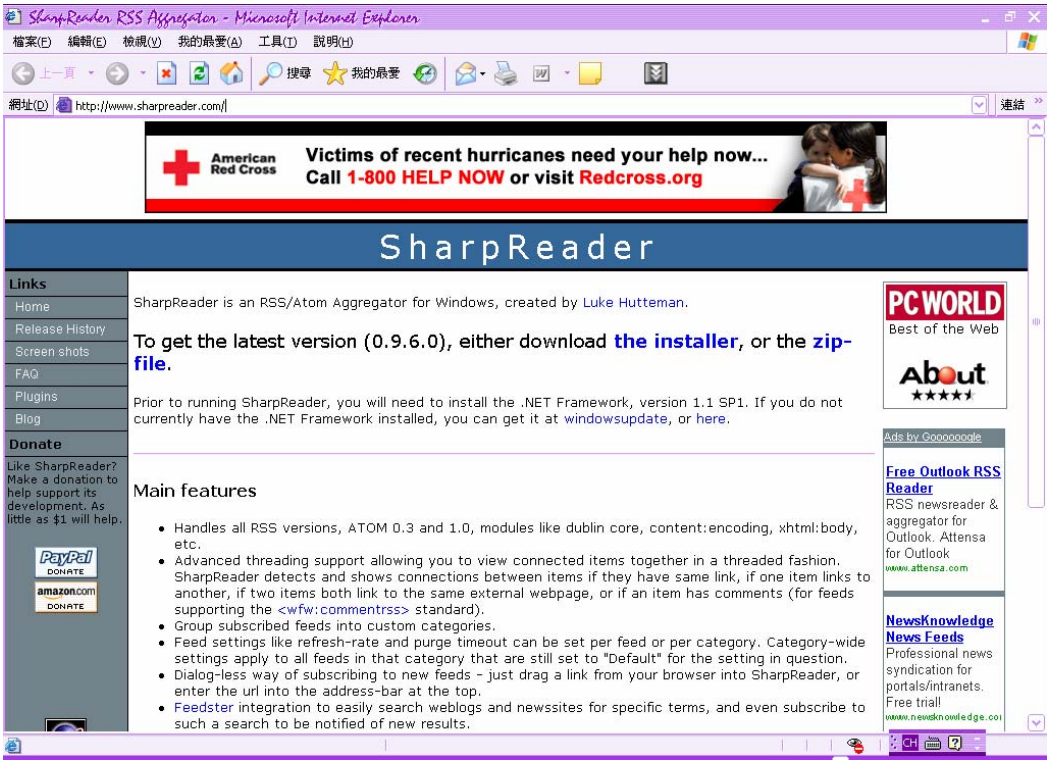
1.3 如何使用 RSS

閱讀 RSS 必須使用閱讀器(Reader)。閱讀器是可以讀取 RSS 規範的語法並將其轉換成文章列表可供閱讀。因為 RSS 是開放的格式，所以閱讀器的實作多如過江之鯽。以下將以 SharpReader 作為使用 RSS 的範例。

- 安裝 SharpReader

Step 1.安裝.Net Framework
http://www.microsoft.com/taiwan/netframework/downloads/howtoget.htm

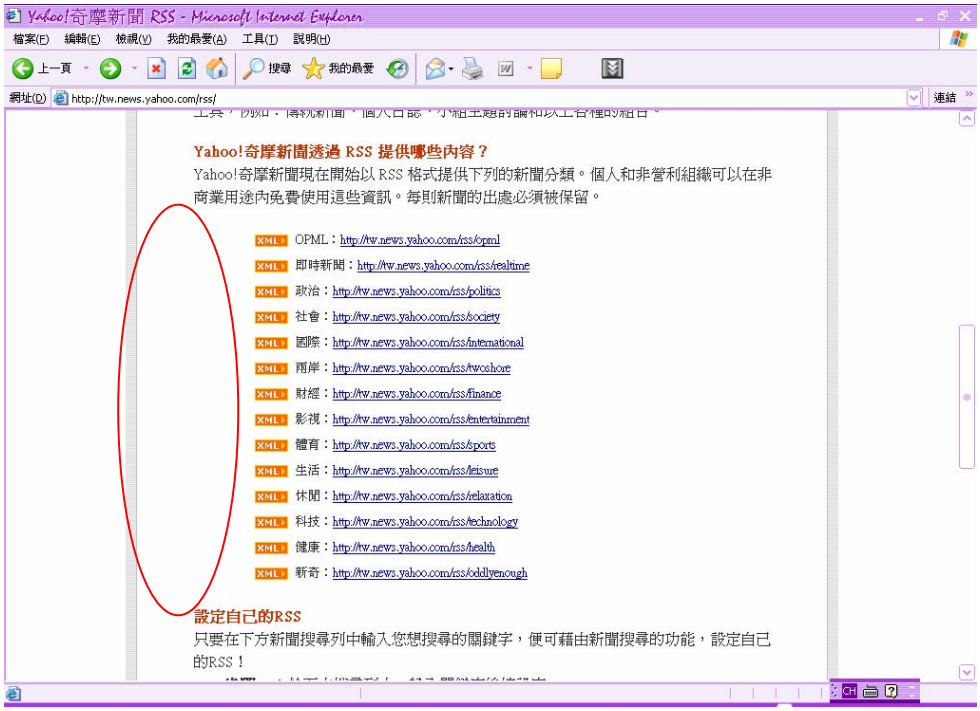
Step 2.至 SharpReader 的網站 http://www.sharpreader.com/ 下載閱讀器並安裝。



- 使用 SharpReader 訂閱相關網站的 RSS

Step 1.先找到該網站可訂閱 RSS 的連結。通常 RSS 的連結都以特定圖示表示。如

下圖 Yahoo 新聞訂閱 RSS http://tw.news.yahoo.com/rss/。



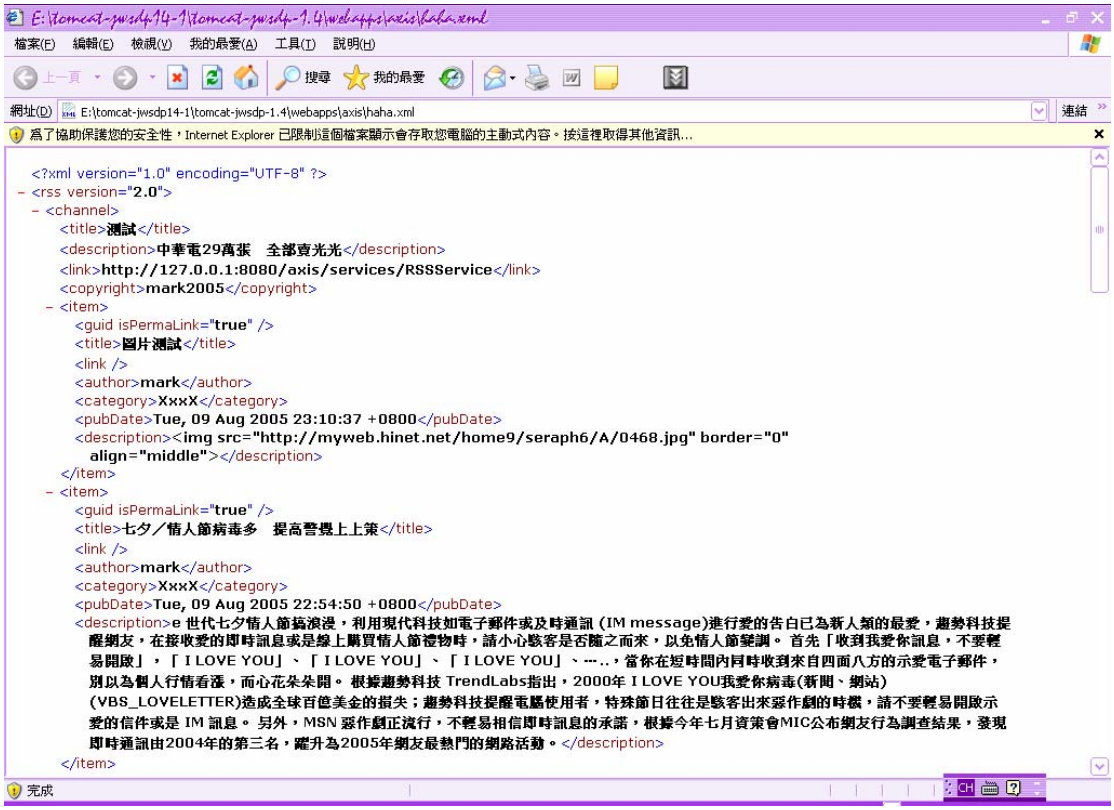
Step 3.RSS 網址的內容是 XML 檔，是給閱讀器閱讀的。將該 RSS 網址複製下來，貼上 SharpReader 的 Address 欄位內，按下 Enter 開始讀取該 Feed。結果如下圖。

Step 4.如果要訂閱該 Feed，按下 Subscribe 按鍵即可在左邊樹狀結構上自動新增一個新節點。SharpReader 會在固定時間內去讀取網路上該 RSS XML 檢查是否更新。

1.4 RSS 的規範

RSS 是 XML 的一種應用，基本上 RSS 必須符合 W3C 的 XML 規範。目前關於 RSS Tag 的規範從 0.91、0.92 一直到現在的 2.0 版本。鑒於本 RSS Service 是由 2.0 規範寫成的，在此將以 RSS 2.0 做為介紹。

下圖是一個 RSS 2.0 XML 範例。



首先 Root 元素由<rss>元素組成，內含 Attribute 為 version="2.0"表示該 RSS XML 是由哪版的規範組成的。接下來必須要有一個<channel>元素包含於<rss>內。<channel>元素可以說是正式描述該 RSS 資訊內容的起點。

在<channel>內，有三個元素是必要的。

<title>	該 RSS 的標題。
<link>	與該 RSS 相關的 Web 首頁或是網頁資訊。
<description>	該 RSS 的描述。

以下是<channel>內可選擇元素；

<language>	RSS 使用的語言種類，例如 en-us，簡體中文是 zh-cn。它方便 Reader 組織同一語言的站點。可以使 W3C 預定義的值。
<copyright>	channel 內容的版權聲明。
<managingEditor>	對於該 channel 內容負責的個人的 Email 位址。
<webMaster>	對該 channel 的技術支援負責的個人的 Email 位址。

<pubDate>	該 channel 內容的公佈日期。例如，一個根據紐約時間按日更新的 channel 每 24 小時公佈日期就滾動一次。即該 channel 的更改的時間。所有 RSS 中使用的日期時間遵守 RFC 822 規範，年份可以是兩位或者四位（首選四位）。
<lastBuildDate>	上次 channel 內容更改的時間。
<category>	說明 channel 屬於哪一個或多個分類，其規則和<item>級別的 category 元素一樣 generator 說明用於生成該頻道的程式。
<docs>	RSS 檔所使用格式的說明文檔所在的 URL。它可能指向本文檔。它有助於讓人理解該 RSS 檔。
<cloud>	允許進程註冊為“cloud”， channel 更新時通知它，為 RSS 提供實現了一種輕量級的發佈-訂閱協定。
<image>	指定一個能在 channel 中顯示的 GIF、JPEG 或 PNG 圖像。
<skipDays>	告訴 Reader 那一天的更新可以忽略。
<rating>	關於該 channel 的 PICS 評價。
<textInput>	定義可與 channel 一起顯示的輸入框。
<skipHours>	告訴 Reader 哪些小時的更新可以忽略。

任意 channel 內可以有數個<Item>。<item>可以說是一個故事，一篇主題。更簡單的說，<item>好像是每封郵件，裡面包含的內容就是<item>內所描述的。利用 Reader 看到的資訊內容都是由每個<item>元素所組成的。

<item>內可描述的元素；

<title>	item 的標題。
<link>	item 的 URL。
<description>	item 的大綱。
<author>	item 作者的 Email 地址。
<category>	item 的一個或多個分類。
<comments>	item 的注釋頁的 URL。
<enclosure>	item 有關的媒體物件。
<guid>	item 聯繫在一起的永久性鏈結。

<pubDate>	item 是什麼時候發佈的。
<source>	item 來自哪個 RSS 頻道，當把 item 聚合在一起時非常有用。

在 RSS Service 中並不是把所有 RSS 規範中的元素加入到 Service 中，而是挑選出常用的 Tag 作為 RSS 產生依據。這樣一來 XML 也不會因為無謂的 Tag 而導致容量過大。

以下為 RSS Service 所支援的 Tag：

<rss> <channel> <title> <description> <link> <copyright> <item> <guid> <author> <category> <pubDate>

1.5 Web Service 簡介

Web 服務(Web Services)代表著可以從 Web 上存取的一個單位的商業、應用、或者是系統的功能。 Web 服務的英文原名是 Web Services，翻譯成網路服務會和一般作業系統的 Network Services 混淆，所以將其稱之「Web 服務」。

Web 服務可用無線的手機、桌上電腦、或是從一個應用程式，甚至從另一個 Web 服務上面，透過網路去呼叫調用的功能，以提供某種服務，諸如是 B2C、B2B 或是 P2P 的網上服務，而這網路可以是 Internet、Intranet 或是 Extranet。

Web 服務是在網路上可被其它程式用標準網路協定呼叫的軟體元件，它用 XML 來做程式間溝通的媒介。

Web 服務是讓那些在環境之下提供服務的軟體模組元件整合在一起，協同工作的一組開放性技術及標準。它用的標準及網路協定有:

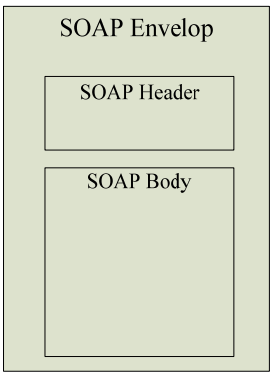
XML	形容結構化資料
SOAP	簡單物件存取協定
WSDL	描述服務細節
UDDI	服務搜尋
Web 服務之安全技術及標準	WS-Addressing、WS-Security、WS-notification…

1.5.1 SOAP

稱為簡單物件存取協定 (SOAP，Simple Object Access Protocol) 也是以 XML 來規範的協定。SOAP 是以訊息傳遞機制來達成傳統的程式驅動。由於 SOAP 沒有自己的底層通訊協定，故需使用其他的協定(如 HTTP、SMTP、FTP 等)，因此 SOAP 的彈性極佳。

SOAP 訊息是以一個稱為 Envelope 來表示，並內含 SOAP Header 及 SOAP Body 。

SOAP Header 是協助寄件者能依相關資訊送交到最終的收件者，SOAP Body 是指訊息實際的內容。SOAP 架構圖如下所示。



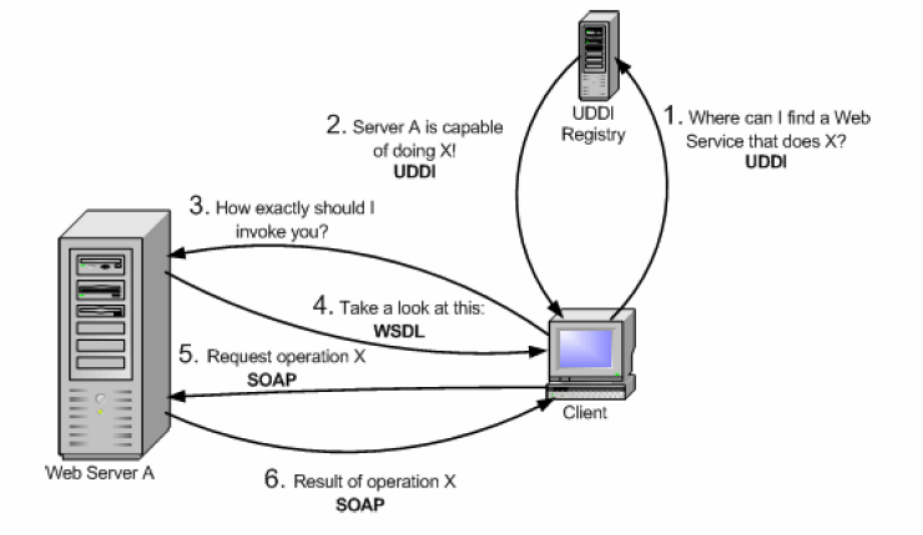
1.5.2 WSDL

WSDL(Web Services Description Language)主要是描述 Web Services 的細節，也是使用 XML 格式之語言。Web Service 描述自己有哪些方法可以呼叫，呼叫參數型態，回傳型態可以說都是透過 WSDL 描述而成的。如果了解 WSDL，便了解該 Service 所傳達運作的細節。

1.5.3 UDDI

UDDI 為 Web Service 的註冊服務。傳送的內容以 XML 為主。Web Service 可以向 UDDI 註冊相關資訊。其他使用者或是 Web Service 可以透過 UDDI 查詢有哪些可用的服務。UDDI 可以說是在 Web Service 裡面的電話簿一樣。

1.5.4 典型 Web Service 運作過程



1.對於一個 client 他要如何得知哪裡有提供怎樣的 Web Service 可以供他使用?首先他必須連到 UDDI(Universal Description Discovery and Integration)的主機，當一個服務的提供者(Service Provider)開發某個 Web Service 之後，可以向 UDDI Server 註冊，在上面描述其 service 的一些相關資迅，之後 client 端向 UDDI Server 查詢其所需的 Web Service。

2.之後 UDDI Server 會回覆，告知那個 server 有提供這樣的 Service。

3.之後取得提供服務 server 的位置之後，client 端就可以與之取得連線，但 client 端必須知道要如何去啟動這個服務，所以要知道服務的詳細描述，例如他有提供哪些 method 可以使用，接受以及回傳怎樣的訊息。

4.提供服務的 server 同時也會提供服務的 WSDL(Web Services Description Langugaes)，client 端可以透過 wsdl 來了解到，這個服務本身的細節，如何去啟動、那些 method 可以使用以及接受那些參數的傳入回傳怎樣的訊息，有了這些資訊 Client 端就可以依照其格式來啟動服務。

5.之後服務的啟動就是透過 SOAP(Simple Object Access Protocol)來完成，SOAP 會定義了 client 端的 request 以及 server 端的 reponse 格式。

6.最後 server 會回傳 SOAP message，來告知執行的結果，不管是正確或者是錯誤的結果，都會回傳回來。

1.6 開發 RSS Service 動機

1 **讓軟體自己能寫 Blog**

由於 RSS Service 是透過 Web Service 技術所組成。Web Service 技術具有了跨平台性，因此無論是什麼語言所寫成的軟體皆可以呼叫該 Service 進行 RSS 的產生。適合的軟體有 Web Application，如 jsp、php、asp 亦或是 J2EE 等網路伺服器，還包括其他長時間在網路上運作的程式，如網管軟體。透過 RSS Service 可以產生日誌，特定事件的通知，讓軟體也能寫自己的 Blog，且不用針對 RSS 核心重新撰寫。

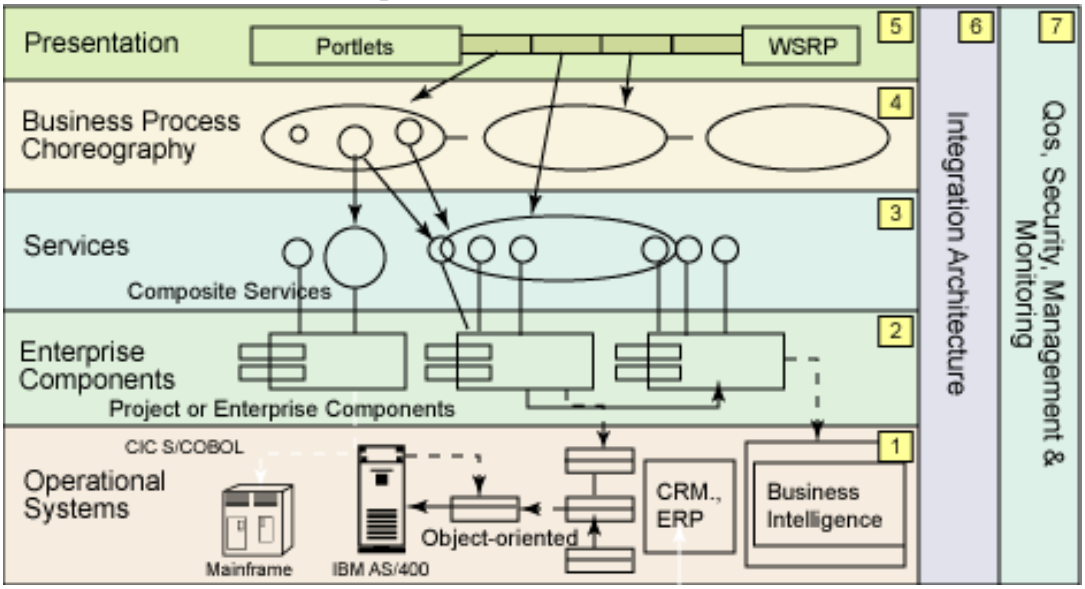
2 **使用者可以訂製自己的 RSS**

使用者可以透過 SOAP 溝通工具產生自己的 RSS。相關應用可以是作為特定資訊發布。尤其是在不支援互動語言網路空間的使用者可以訂做自己的網站更新資訊，而不用因為沒有支援互動語言或是不會撰寫程式而傷腦筋。

3 **作為 SOA 的一環**

由於 Web Service 的特性，RSS Service 可以成為 SOA(Service Oriented Architecture)架構中軟體的一環。不但提高了軟體的可重用性，同時也能在服務導向中隨意與其他 Web Service 重新整合成一個新的應用程式。

圖表 1 IBM develop works tw-以服務為基礎的塑膜與架構



在上圖 SOA 架構中，RSS Service 所處的就是在第三層的 Services 層，可供為連結其他 Service 成為一個 Business Process Choreography 使用。

1.7 RSS Service 軟體特性

- ✓ 支援 RSS 文件標準。
- ✓ 自訂 RSS 名稱。
- ✓ 自訂 XML 位址。
- ✓ 自訂 Item 最大數量。
- ✓ 支援同時多個 RSS 在 RSS Service 上運作。
- ✓ 自動讀取寫入設定檔，使 RSS Service 支援 Stateful 狀態。
- ✓ 協力工具協助設定 RSS Service。
- ✓ 提供跨平台呼叫需求。

1.8 部署 RSS Service 至伺服器

建議必要環境	測試環境
可支援 Tomcat 執行之軟硬體環境即可	AMD 1800+ 512MBRam Windows XP
Apache Tomcat	Apache Tomcat 4.1 –jwsdp 1.5
Axis 1.2	Axis 1.2

註：在這裡是以 Axis 作為 Container 部署。若使用的 Container 不同，部署方式會有所不同，請參閱該 Container 部署文件。

Step 1.複製所需檔案
將包裝好 RSS Service 的 jar 封裝檔複製進 axis 所屬的/WEB-INF/lib 中。

Step 2.部署服務
此步驟主要為將 wsdd 描述檔部署 axis 所屬的 servlet 中。Wsdd 檔內容如下。

```
<?xml version="1.0" encoding="utf-8" ?>
<deployment xmlns="http://xml.apache.org/axis/wsdd/"
xmlns:java="http://xml.apache.org/axis/wsdd/providers/java">
<service name="RSSService" provider="java:RPC">
   <parameter name="className"
     value="org.twbbs.hellokitty.service.RSSService.RSSService" />
   <parameter name="allowedMethods" value="*" />
```

```
    <parameter name="scope" value="application" />
</service>
</deployment>
```

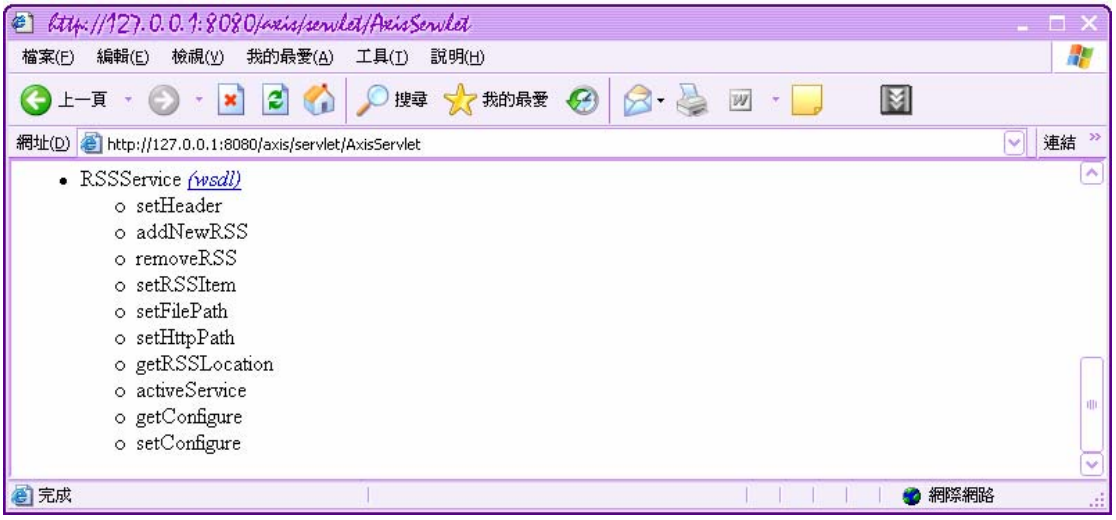
首先確認 Tomcat 已經運行。在命令提示元下執行以下指令

```
java -Djava.ext.dirs=%AXISHOME%\lib
org.apache.axis.client.AdminClient -saxisdemo\services\AdminService
–p8080 deploy.wsdd
```

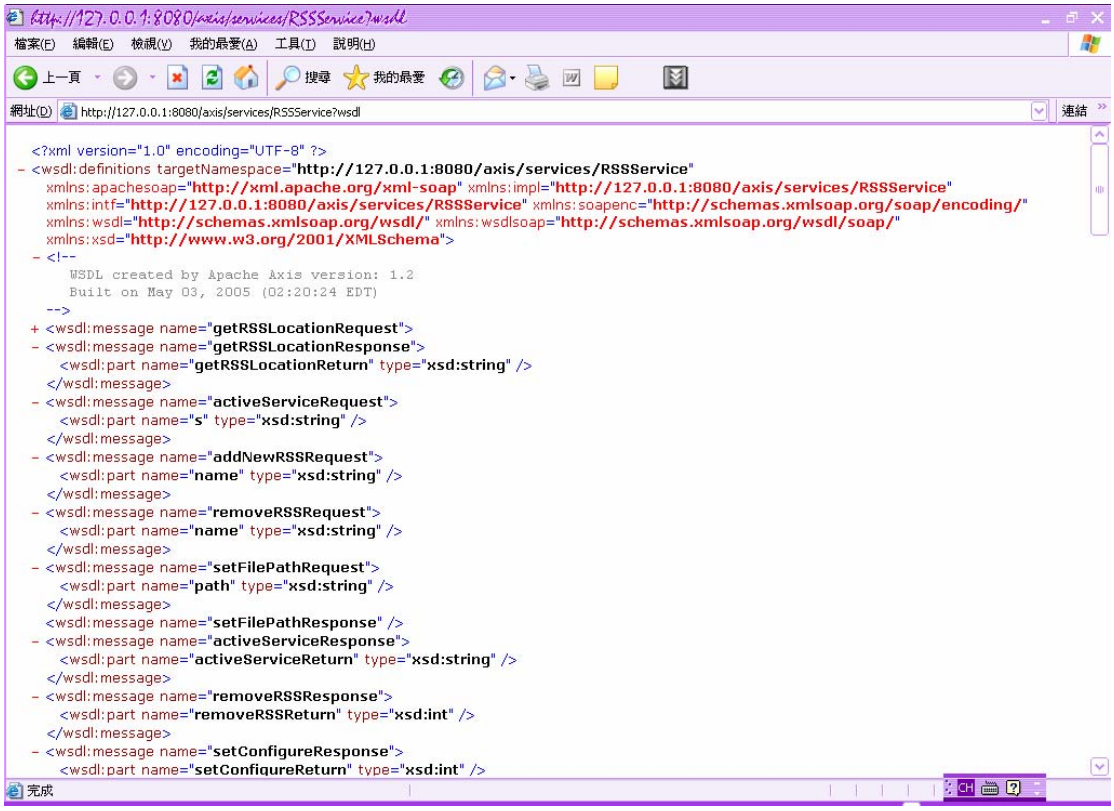
deploy.wsdd 是 wsdd 的檔名。

Step 3.確認部署狀況

觀看部署狀況 http://127.0.0.1:8080/axis/servlet/AxisServlet。若 wsdd 部署成功應會出現 RSS Service 所屬名稱與對應方法。如下圖所示。



按下名稱旁的 wsdl 連結，若 jar 所屬位置正確，便會即時產生對應的 wsdl 文件。

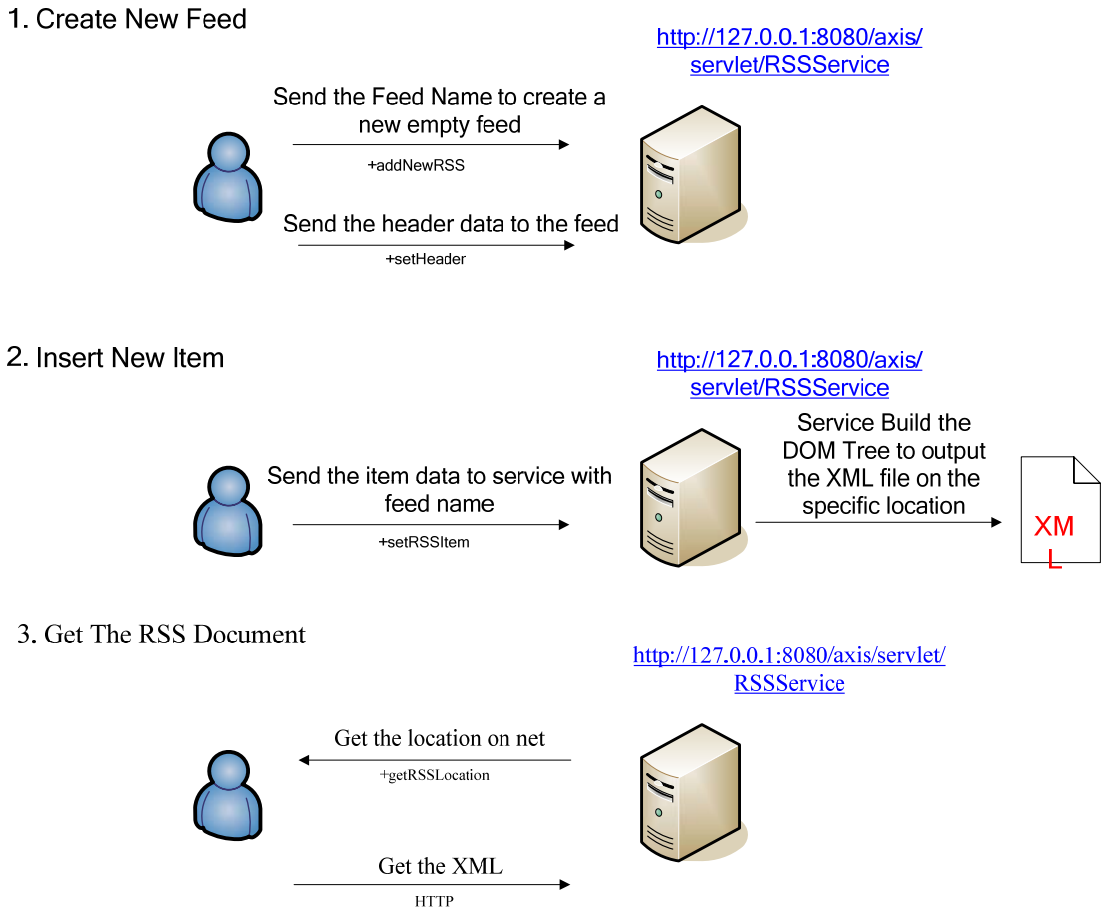


1.9 RSS Service 呼叫方法介紹

setHeader (String rssname,String title,String description,String link,String copyright)	設定該 RSS 文件的名稱，標題，描述，連結以及版權。在第一次產生 RSS 文件時候需要對 Header 做設定。
addNewRSS (String name)	新增一個 RSS 文件，name 為該 RSS 文件名稱。
removeRSS (String name)	移除該 RSS 文件，name 為該 RSS 文件名稱。
setRSSItem (String rssname,String guid,String title,String link,String author,String category,String	新增一個 item 進入 RSS 文件中，相當於新增一條新資訊。參數為想要新增的 RSS 名稱、guid、item 標題、連結、作者、類別、發布日期、描述內容。若不想對某些

pubDate,String description)	參數做設定，送出空字串即可。發布日期輸入”null”字串，Service 會自動產生符合標準的日期字串。
setFilePath (String path)	設定 RSS 文件位址所在檔案路徑。
setHttpPath (String path)	設定 RSS 文件在 Http 協定上所屬檔案路徑。
getRSSLocation (String rssname)	取得特定 RSS 文件在 Http 協定上絕對路徑。
setMaxItem(String i)	設定單一文件中最大 item 數量。
getConfigure ()	取得設定內容，為一字串。
setConfigure (String setting)	設定 RSS Service。此方法為 setFilePath，setHttpPath，setMaxItem 的綜合方法。

RSS Service 基本運作如下：



1.10 RSS Service Tool Box 介紹

在此 RSS Service 中提供了方便的工具來對 RSS Service 做設定以及調測。而這些動作也都是透過 Web Service 溝通而完成的。有了這些工具，在 Service 維護以及運作上會有相當的助益。

1.10.1 RSS Service Configure ToolBox

此工具是設定 RSS file 在系統中的位置以及在網路上的位置，也能設定 item 的數量。藉由軟體寫好的分析字串，可以將資訊分別顯示在 GUI 上並做修改設定。

1.10.2 RSS Service Test Feed ToolBox

這是可以編輯送出 header 及 item 的工具。在選定 Service 的 Address 之後，可以手動建立新的 feed，並根據 feed 名稱對 header 及 item 做設定。之前所談及的一般使用者可以利用此工具更新他們自己的 RSS。

1.11 小結

RSS Service 在軟體及個人上，提供了一個資訊發布較佳的解決方案。使用者只要藉由通用的 RSS Reader 便可以讀取資訊，滿足了簡單性及方便性的需求。在程式開發上可藉由組合方式幫助縮短開發時程。

2. XML Parser of bio software

2.1 簡介

XML(eXtensible Markup Language)具有平台可交換性，在不同平台上可以輕易操作使用 XML。在可閱讀性上也非常高，定義良好的標籤可以讓使用者輕易閱讀

出此 XML 所包含的訊息。對於程式設計師而言，XML 簡化了資料處理的流程步驟，透過一些標準定義的 XML 語法，如 XPath、XQury，可以直接定位出所要的資訊。因此以下為將雜亂無章，可利用性低的輸入輸出生物軟體格式轉換成適當的 XML。

2.2 ClustalW

2.2.1 ALN 格式

a.未轉換前範例格式

```
CLUSTAL W (1.82) multiple sequence alignment


gi|17865154|gb|AAL47160.1|AF45  ------------------------------------------------
gi|1709489|sp|P54790|ORC3_YEAS  ------------------------------------------------
gi|18599218|ref|XP_002914.4|   ------------------------------------------------
gi|6981604|ref|NP_037172.1|    ------------------------------------------------
gi|6320339|ref|NP_010419.1|    ------------------------------------------------
gi|461721|sp|Q00553|CFTR_MACMU  ------------------------------------------------
gi|116140|sp|P26361|CFTR_MOUSE  ------------------------------------------------
gi|14141185|ref|NP_066388.1|   MQKSPLEKASFISKLFFSWTTPILRKGYRHHLELSDIYQAPSADSADHLS
gi|461723|sp|P34158|CFTR_RAT    ------------------------------------------------
gi|461720|sp|Q00552|CFTR_CAVPO  ------------------------------------------------
gi|116142|sp|P26363|CFTR_XENLA  ------------------------------------------------
gi|9966877|ref|NP_065132.1|     ------------------------------------------------
gi|13124088|sp|Q9R0A1|CLC2_MOU  ------------------------------------------------
gi|1706485|sp|P54861|DNM1_YEAS  ------------------------------------------------
gi|6753432|ref|NP_034030.1|    ------------------------------------------------
gi|17224460|gb|AAL36985.1|AF28  ------------------------------------------------
gi|1705763|sp|Q00554|CFTR_RABI  ------------------------------------------------
gi|461719|sp|P35071|CFTR_BOVIN  ------------------------------------------------
gi|14753227|ref|XP_004980.4|   ------------------------------------------------
gi|2506121|sp|Q00555|CFTR_SHEE  ------------------------------------------------
gi|116141|sp|P26362|CFTR_SQUAC  ------------------------------------------------
gi|18574439|ref|XP_083829.1|   ------------------------------------------------
gi|2506123|sp|P14772|BPT1_YEAS  ------------------------------------------------
gi|6978669|ref|NP_036965.1|    ------------------------------------------------
gi|17224462|gb|AAL36986.1|AF28  ------------------------------------------------
gi|18266164|gb|AAL67487.1|AF45  ------------------------------------------------
gi|6678848|ref|NP_032602.1|    ------------------------------------------------
gi|1706521|sp|P32892|DRS1_YEAS  ------------------------------------------------
gi|12860348|dbj|BAB31925.1|    ------------------------------------------------
gi|18034783|ref|NP_542148.1|   ------------------------------------------------
gi|6981606|ref|NP_037171.1|    ------------------------------------------------
gi|2133118|pir||S64758          ------------------------------------------------
gi|17488612|gb|AAL40378.1|AC08  ------------------------------------------------
gi|1705762|sp|P13569|CFTR_HUMA  ------------------------------------------------
     Name Block                          Hit Block

gi|17865154|gb|AAL47160.1|AF45  ------------------------------------------------
gi|1709489|sp|P54790|ORC3_YEAS  ------------------------------------------------
gi|18599218|ref|XP_002914.4|   ------------------------------------------------
gi|6981604|ref|NP_037172.1|    ------------------------------------------------
gi|6320339|ref|NP_010419.1|    ------------------------------------------------
…
```

b.轉換程式

Class 路徑

org.twbbs.hellokitty.clustalw.XMLParser.aln.XMLParser.XMLParser

使用說明

java org.twbbs.hellokitty.clustalw.XMLParser.aln.XMLParser.XMLParser [input file name] + [Output file name]

使用範例

java org.twbbs.hellokitty.clustalw.XMLParser.aln.XMLParser.XMLParser CFTR_ALN.out aln.xml

API

XMLParser public XMLParser() Default Constructor
XMLParser public XMLParser(java.lang.String filename, java.lang.String outname) Process Constructor **Parameters**: filename - String outname - String

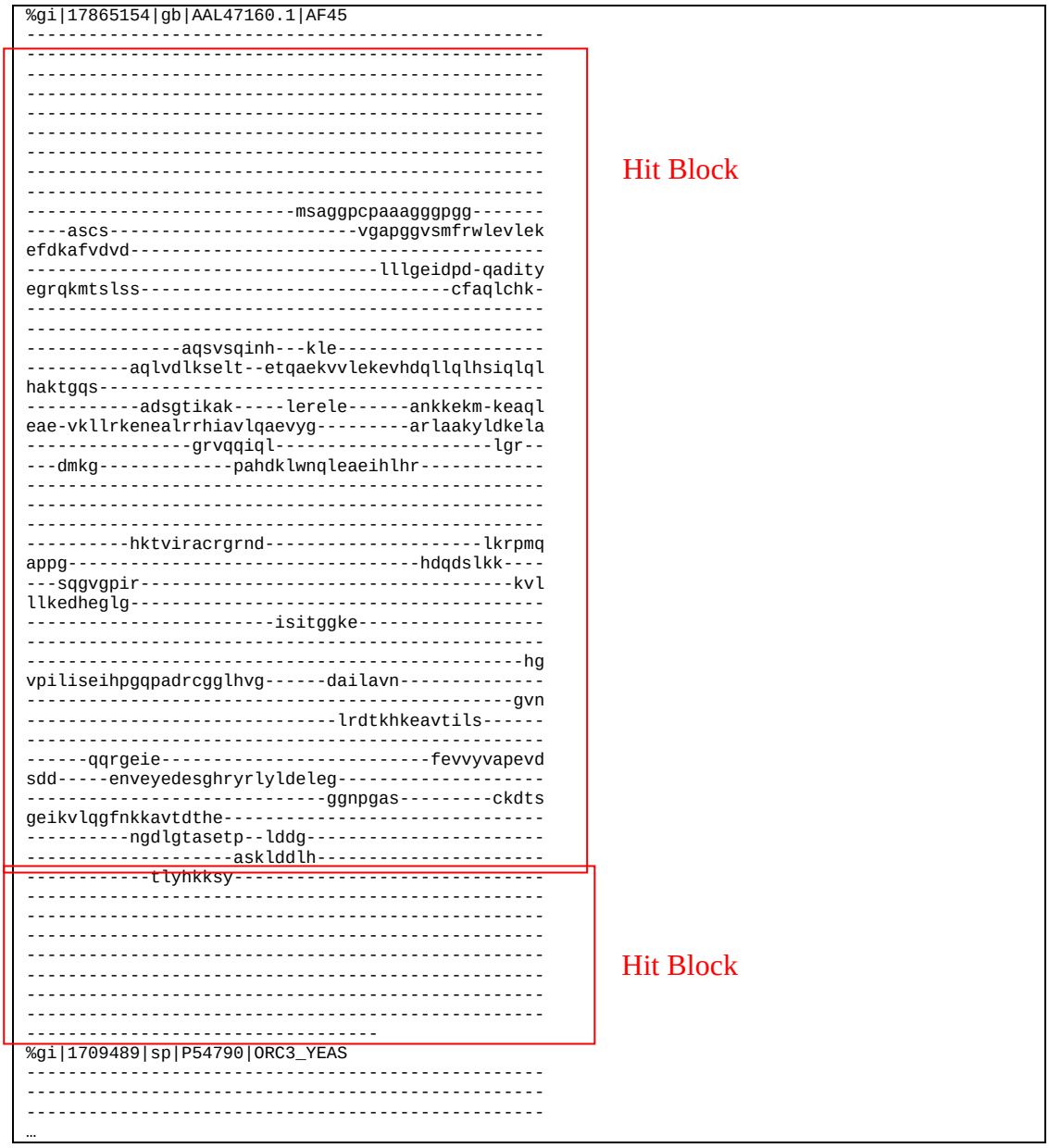
c.轉換結果部份範例

```
<?xml version="1.0" encoding="UTF-8" ?>
- <CLUSTALW type="aln" version="(1.82)">
  - <hits>
    - <block>
      - <hit>
          <parameter>gi</parameter>
          <parameter>17865154</parameter>
          <parameter>gb</parameter>
          <parameter>AAL47160.1</parameter>
          <parameter>AF45</parameter>
          <marks>----------
            ----------
            ----------
            ----------
            ----------
            EKLEREWDREQASKKNPQLIHALRRCFFWRFLFYGILLYLGEVTKAVQPV----------
            ----------
            ----------
            ----------
            ----------
            ----------
            ----------
            ----------
            ----------
            ----------
            ----------
            ----------
            ----------
            ----------
            ----------
            ----------
            ----------
            ----------
            LLGRIIASYDPENKVERSIAIYLGIGLCLLFIVRTLLLHPAIFGLHRIGM----------
            ----------
            ----------
            ----------
            ----------
```

2.2.2 GDE 格式

a. 未轉換前範例格式

Name Block



b.轉換程式

Class 路徑

org.twbbs.hellokitty.clustalw.XMLParser.gde.XMLParser.XMLParser

使用說明

java org.twbbs.hellokitty.clustalw.XMLParser.gde.XMLParser.XMLParser [input file name] + [Output file name] ([mode]) mode: 0-Seperate marks 1-Same marks (Default)

使用範例

```
java org.twbbs.hellokitty.clustalw.XMLParser.gde.XMLParser.XMLParser
CFTR_ALN.out gde.xml
```

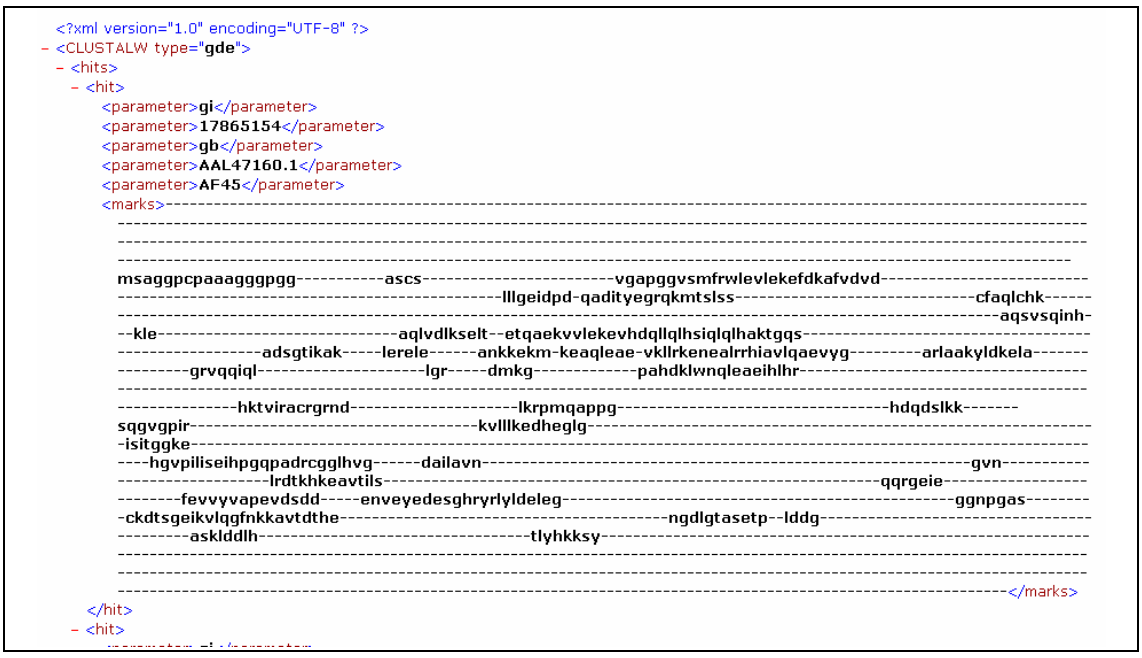
API

XMLParser
public XMLParser()
Default Constructor

XMLParser
public XMLParser(java.lang.String filename,
java.lang.String outname)
Process Constructor Mode 1
Parameters:
filename - String
outname - String

XMLParser
public XMLParser(java.lang.String filename,
java.lang.String outname,
int mode)
Process Constructor
Parameters:
filename - String
outname - String
mode - int

c.轉換結果部份範例



2.2.3 GCG 格式

a. 未轉換前範例格式

```
PileUp


  MSF: 2584  Type: P    Check:  6854   ..

Name: gi|17865154|gb|AAL47160.1|AF45 oo  Len: 2584  Check:  3975  Weight:  3.8
Name: gi|1709489|sp|P54790|ORC3_YEAS oo  Len: 2584  Check:  8702  Weight:  4.6
Name: gi|18599218|ref|XP_002914.4| oo  Len: 2584  Check:  9253  Weight:  1.5
Name: gi|6981604|ref|NP_037172.1| oo  Len: 2584  Check:  8565  Weight:  1.5
Name: gi|6320339|ref|NP_010419.1| oo  Len: 2584  Check:  7592  Weight:  1.9
Name: gi|461721|sp|Q00553|CFTR_MACMU oo  Len: 2584  Check:  7017  Weight:  0.1
Name: gi|116140|sp|P26361|CFTR_MOUSE oo  Len: 2584  Check:  4863  Weight:  0.1
Name: gi|14141185|ref|NP_066388.1| oo  Len: 2584  Check:  1276  Weight:  1.5
Name: gi|461723|sp|P34158|CFTR_RAT oo  Len: 2584  Check:  4197  Weight:  3.4
Name: gi|461720|sp|Q00552|CFTR_CAVPO oo  Len: 2584  Check:  4898  Weight:  5.4
Name: gi|116142|sp|P26363|CFTR_XENLA oo  Len: 2584  Check:  4649  Weight:  0.1
Name: gi|9966877|ref|NP_065132.1| oo  Len: 2584  Check:  1272  Weight:  5.0
Name: gi|13124088|sp|Q9R0A1|CLC2_MOU oo  Len: 2584  Check:   401  Weight:  5.0
Name: gi|1706485|sp|P54861|DNM1_YEAS oo  Len: 2584  Check:  8151  Weight:  4.2
Name: gi|6753432|ref|NP_034030.1| oo  Len: 2584  Check:   401  Weight:  3.4
…
//
```

Information Block

Name Block

```
gi|17865154|gb|AAL47160.1|AF45
gi|1709489|sp|P54790|ORC3_YEAS
gi|18599218|ref|XP_002914.4|
gi|6981604|ref|NP_037172.1|
gi|6320339|ref|NP_010419.1|
gi|461721|sp|Q00553|CFTR_MACMU
gi|116140|sp|P26361|CFTR_MOUSE
gi|14141185|ref|NP_066388.1|
gi|461723|sp|P34158|CFTR_RAT
gi|461720|sp|Q00552|CFTR_CAVPO
gi|116142|sp|P26363|CFTR_XENLA
gi|9966877|ref|NP_065132.1|
gi|13124088|sp|Q9R0A1|CLC2_MOU
gi|1706485|sp|P54861|DNM1_YEAS
gi|6753432|ref|NP_034030.1|
gi|17224460|gb|AAL36985.1|AF28
gi|1705763|sp|Q00554|CFTR_RABI
gi|461719|sp|P35071|CFTR_BOVIN
```

Name Block

Hit Block

```
..... ..... ..... ..... .....
..... ..... ..... ..... .....
..... ..... ..... ..... .....
..... ..... ..... ..... .....
..... ..... ..... ..... .....
..... ..... ..... ..... .....
..... ..... ..... ..... .....
MQKSPLEKAS FISKLFFSWT TPILRKGYRH HLELSDIYQA PSADSADHLS
..... ..... ..... ..... .....
..... ..... ..... ..... .....
..... ..... ..... ..... .....
..... ..... ..... ..... .....
..... ..... ..... ..... .....
..... ..... ..... ..... .....
..... ..... ..... ..... .....
..... ..... ..... ..... .....
..... ..... ..... ..... .....
```

Hit Block

b.轉換程式

Class 路徑

org.twbbs.hellokitty.clustalw.XMLParser.gcg.XMLParser.XMLParser

使用說明

java org.twbbs.hellokitty.clustalw.XMLParser.gcg.XMLParser.XMLParser [input file name] + [Output file name] ([mode])

使用範例

java org.twbbs.hellokitty.clustalw.XMLParser.gcg.XMLParser.XMLParser

CFTR_ALN.out gcg.xml

API

XMLParser
public XMLParser()
Default Constructor

XMLParser
public XMLParser(java.lang.String filename,
java.lang.String outname)
Process Constructor
Parameters:
filename - String
outname - String

c.轉換結果部份範例

```
<?xml version="1.0" encoding="UTF-8" ?>
- <CLUSTALW type="gcg">
  - <header>
      <MSF>2584</MSF>
      <type>P</type>
      <check>6854</check>
    </header>
  - <hits>
    - <hit>
        <parameter>gi</parameter>
        <parameter>17865154</parameter>
        <parameter>gb</parameter>
        <parameter>AAL47160.1</parameter>
        <parameter>AF45</parameter>
        <type>oo</type>
        <length>2584</length>
        <check>3975</check>
        <weight>3.8</weight>

          <marks>.....
      </hit>
    - <hit>
        <parameter>gi</parameter>
        <parameter>1709489</parameter>
        <parameter>sp</parameter>
        <parameter>P54790</parameter>
        <parameter>ORC3_YEAS</parameter>
        <type>oo</type>
        <length>2584</length>
        <check>8702</check>
        <weight>4.6</weight>
```

2.2.4 phylip 格式

a. 未轉換前範例格式

```
   34   2584
gi|1786515 ---------- ---------- ---------- ---------- ----------
gi|1709489 ---------- ---------- ---------- ---------- ----------
gi|1859921 ---------- ---------- ---------- ---------- ----------
gi|6981604 ---------- ---------- ---------- ---------- ----------
gi|6320339 ---------- ---------- ---------- ---------- ----------
gi|461721| ---------- ---------- ---------- ---------- ----------
gi|116140| ---------- ---------- ---------- ---------- ----------
gi|1414118 MQKSPLEKAS FISKLFFSWT TPILRKGYRH HLELSDIYQA PSADSADHLS
gi|461723| ---------- ---------- ---------- ---------- ----------
gi|461720| ---------- ---------- ---------- ---------- ----------
gi|116142| ---------- ---------- ---------- ---------- ----------
gi|9966877 ---------- ---------- ---------- ---------- ----------
gi|1312408 ---------- ---------- ---------- ---------- ----------
gi|1706485 ---------- ---------- ---------- ---------- ----------
gi|6753432 ---------- ---------- ---------- ---------- ----------
gi|1722446 ---------- ---------- ---------- ---------- ----------
gi|1705763 ---------- ---------- ---------- ---------- ----------
```

Hit Block

Name Block

b.轉換程式

Class 路徑

org.twbbs.hellokitty.clustalw.XMLParser.phylip.XMLParser.XMLParser

使用說明

java
org.twbbs.hellokitty.clustalw.XMLParser.phylip.XMLParser.XMLParser
[input file name] + [Output file name] ([mode])
mode: 0-Block marks
1-Line marks
2-Same marks
3-Same marks with trim
4-Line marks with trim

使用範例

java
org.twbbs.hellokitty.clustalw.XMLParser.phylip.XMLParser.XMLParser
CFTR_phylip.out phylip.xml

API

XMLParser
public XMLParser()
Default Constructor

XMLParser
public XMLParser(java.lang.String filename,
java.lang.String outname)
Process Constructor Mode 1
Parameters:

filename - String
outname - String

XMLParser

public XMLParser(java.lang.String filename,
java.lang.String outname,
int mode)

Process Constructor

Parameters:

filename - String
outname - String
mode - int

c.轉換結果部份範例

```
<?xml version="1.0" encoding="UTF-8" ?>
- <CLUSTALW type="phylip">
  - <hits>
    - <hit>
        <parameter>gi</parameter>
        <parameter>1786515</parameter>
        <marks>-----------------------------------------------</marks>
        <marks>-----------------------------------------------</marks>
        <marks>-----------------------------------------------</marks>
        <marks>-----------------------------------------------</marks>
        <marks>-----------------------------------------------</marks>
        <marks>-----------------------------------------------</marks>
        <marks>-----------------------------------------------</marks>
        <marks>-----------------------------------------------</marks>
        <marks>-----------------------------------------------</marks>
        <marks>-------------------------MSAGGPCPAAAGGGPGG-------</marks>
        <marks>----ASCS-----------------------VGAPGGVSMFRWLEVLEK</marks>
        <marks>FDKAFVDVD-----------------------------------------</marks>
        <marks>-----------------------------------LLLGEIDPD-QADITY</marks>
        <marks>GRQKMTSLSS-----------------------------CFAQLCHK-</marks>
        <marks>-----------------------------------------------</marks>
        <marks>-----------------------------------------------</marks>
        <marks>---------------AQSVSQINH---KLE--------------------</marks>
        <marks>---------AQLVDLKSELT--ETQAEKVVLEKEVHDQLLQLHSIQLQL</marks>
        <marks>AKTGQS-----------------------------------------------</marks>
        <marks>----------ADSGTIKAK-----LERELE------ANKKEKM-KEAQL</marks>
        <marks>AE-VKLLRKENEALRRHIAVLQAEVYG---------ARLAAKYLDKELA</marks>
        <marks>---------------GRVQQIQL--------------------LGR--</marks>
        <marks>--DMKG------------PAHDKLWNQLEAEIHLHR------------</marks>
        <marks>-----------------------------------------------</marks>
        <marks>-----------------------------------------------</marks>
        <marks>-----------------------------------------------</marks>
        <marks>---------HKTVIRACRGRND--------------------LKRPMQ</marks>
```

2.2.5 PIR 格式

a. 未轉換前範例格式

```
>P1;gi|17865154|gb|AAL47160.1|AF45
-----------------------------------------------
-----------------------------------------------
-----------------------------------------------
-----------------------------------------------
-----------------------------------------------
-----------------------------------------------
-----------------------------------------------
-----------------------------------------------
-------------------------MSAGGPCPAAAGGGPGG-------
----ASCS-----------------------VGAPGGVSMFRWLEVLEK
EFDKAFVDVD-----------------------------------------
-----------------------------------LLLGEIDPD-QADITY
EGRQKMTSLSS-----------------------------CFAQLCHK-
-----------------------------------------------
-----------------------------------------------
---------------AQSVSQINH---KLE--------------------
```

Name Block

Hit Block

```
----------AQLVDLKSELT--ETQAEKVVLEKEVHDQLLQLHSIQLQL
HAKTGQS-----------------------------------------
-----------ADSGTIKAK-----LERELE------ANKKEKM-KEAQL
EAE-VKLLRKENEALRRHIAVLQAEVYG---------ARLAAKYLDKELA
---------------GRVQQIQL--------------------LGR--
---DMKG------------PAHDKLWNQLEAEIHLHR------------
-------------------------------------------------
-------------------------------------------------
-------------------------------------------------
----------HKTVIRACRGRND--------------------LKRPMQ

>P1;gi|1709489|sp|P54790|ORC3_YEAS

-------------------------------------------------
-------------------------------------------------
-------------------------------------------------
-------------------------------------------------
…
```

b.轉換程式

Class 路徑

org.twbbs.hellokitty.clustalw.XMLParser.pir.XMLParser.XMLParser

使用說明

java org.twbbs.hellokitty.clustalw.XMLParser.pir.XMLParser.XMLParser
[input file name] + [Output file name] ([mode])
mode: 0-Seperate marks
1-Same marks (Default)

使用範例

java org.twbbs.hellokitty.clustalw.XMLParser.pir.XMLParser.XMLParser
CFTR_pir.out pir.xml

API

XMLParser
public XMLParser()
Default Constructor

XMLParser
public XMLParser(java.lang.String filename,
java.lang.String outname)
Process Constructor Mode 1
Parameters:
filename - String
outname - String

XMLParser
public XMLParser(java.lang.String filename,
java.lang.String outname,
int mode)
Process Constructor

Parameters:
filename - String
outname - String
mode - int

c.轉換結果部份範例

```
<?xml version="1.0" encoding="UTF-8" ?>
- <CLUSTALW type="pir">
  - <hits>
    - <hit>
        <parameter>gi</parameter>
        <parameter>17865154</parameter>
        <parameter>gb</parameter>
        <parameter>AAL47160.1</parameter>
        <parameter>AF45</parameter>
        <marks>----------------------------------------------------------------------------------------------------
          ----------------------------------------------------------------------------------------------------
          ----------------------------------------------------------------------------------------------------
          ----------------------------------------------------------------------------------------------------
          MSAGGPCPAAAGGGPGG----------ASCS----------VGAPGGVSMFRWLEVLEKEFDKAFVDVD----------
          ----------LLLGEIDPD-QADITYEGRQKMTSLSS----------
          CFAQLCHK----------
          ------AQSVSQINH---KLE----------AQLVDLKSELT--ETQAEKVVLEKEVHDQLLQLHSIQLQLHAKTGQS----
          ----------ADSGTIKAK-----LERELE------ANKKEKM-KEAQLEAE-
          VKLLRKENEALRRHIAVLQAEVYG---------ARLAAKYLDKELA----------GRVQQIQL----------LGR-----DMKG-
          ----------PAHDKLWNQLEAEIHLHR----------
          ----------HKTVIRACRGRND----------
          ----LKRPMQAPPG----------HDQDSLKK-------SQGVGPIR----------
          KVLLLKEDHEGLG----------ISITGGKE----------
          ----------HGVPILISEIHPGQPADRCGGLHVG---
          ---DAILAVN----------GVN----------
          LRDTKHKEAVTILS----------QQRGEIE----------
          FEVVYVAPEVDSDD-----ENVEYEDESGHRYRLYLDELEG----------GGNPGAS---------
          CKDTSGEIKVLQGFNKKAVTDTHE----------NGDLGTASETP--LDDG----------
          ----------ASKLDDLH----------TLYHKKSY----------
          ----------
          ----------
          ----------
          -</marks>
```

2.2.6 NEXUS 格式

a. 未轉換前範例格式

```
#NEXUS
BEGIN DATA;
dimensions ntax=2 nchar=110;
format missing=?
symbols="ABCDEFGHIKLMNPQRSTUVWXYZ"
interleave datatype=PROTEIN gap= -;

matrix
gi|345664   GANPTHLADFQVTVSWDSGGEDGGLQGPATLLATVDELSHLQSEEPGAPH
gi|345661   GANPTHLADFQVTVSWDSGGEDGGLQGPATLLATVDELSHLQSEEPGAPH

gi|345664   LGSGANPTHLA-----------------------------------
gi|345661   LGSGANPTHLAAQLSTILEKPPRPGAGSIEREDVFHYFENLIGVITLPMP

gi|345664   ----------
gi|345661   PRQVTVQVTV
;
end;
```

Information Block

Hit Block

Name Block

b.轉換程式

Class 路徑

org.twbbs.hellokitty.clustalw.XMLParser.nexus.XMLParser.XMLParser

使用說明

java org.twbbs.hellokitty.clustalw.XMLParser.nexus.XMLParser.XMLParser [input file name] + [Output file name]

使用範例

java org.twbbs.hellokitty.clustalw.XMLParser.nexus.XMLParser.XMLParser CFTR_nexus.out nexus.xml

API

XMLParser public XMLParser() Default Constructor --- **XMLParser** public XMLParser(java.lang.String filename, java.lang.String outname) Process Constructor **Parameters**: filename - String outname - String

c.轉換結果部份範例

```
<?xml version="1.0" encoding="UTF-8" ?>
- <CLUSTALW type="nexus">
  - <header>
    - <dimensions>
        <ntax>2</ntax>
        <nchar>110</nchar>
      </dimensions>
      <format_missing>missing</format_missing>
      <symbols>ABCDEFGHIKLMNPQRSTUVWXYZ</symbols>
    - <interleave>
        <datatype>PROTEIN</datatype>
        <gap>-</gap>
      </interleave>
    </header>
  - <hits>
    - <hit>
        <parameter>gi</parameter>
        <parameter>345664</parameter>
        <marks>GANPTHLADFQVTVSWDSGGEDGGLQGPATLLATVDELSHLQSEEPGAPHLGSGANPTHLA----------
          ----------</marks>
      </hit>
    - <hit>
        <parameter>gi</parameter>
        <parameter>345661</parameter>

        <marks>GANPTHLADFQVTVSWDSGGEDGGLQGPATLLATVDELSHLQSEEPGAPHLGSGANPTHLAAQLSTILEKPPRPGAGSIEREDVFHYFENLIG'
      </hit>
    </hits>
  </CLUSTALW>
```

2.2.7 DND 格式

a. 未轉換前範例格式

```
(
(
(
(
(
(
(
(
gi|18599218|ref|XP_002914.4|:-0.04593,
(
gi|17865154|gb|AAL47160.1|AF45:-0.01106,
gi|1709489|sp|P54790|ORC3_YEAS:0.01106)
:0.04593)
:0.02878,
gi|6981604|ref|NP_037172.1|:-0.02878)
:0.02051,
gi|6320339|ref|NP_010419.1|:-0.02051)
:0.10035,
gi|461721|sp|Q00553|CFTR_MACMU:-0.10035)
:0.07261,
gi|116140|sp|P26361|CFTR_MOUSE:-0.07261)
:0.22213,
(
gi|14141185|ref|NP_066388.1|:0.02551,
gi|461723|sp|P34158|CFTR_RAT:0.06917)
:0.02671)
:0.05177,
(
(
(
(
(
…
```

( )所包含的代表 Tree 之子點

b.轉換程式

Class 路徑

org.twbbs.hellokitty.clustalw.XMLParser.dnd.XMLParser.XMLParser

使用說明

java org.twbbs.hellokitty.clustalw.XMLParser.dnd.XMLParser.XMLParser [input file name] + [Output file name]

使用範例

java org.twbbs.hellokitty.clustalw.XMLParser.dnd.XMLParser.XMLParser CFTR_dnd.out dnd.xml

API

XMLParser public XMLParser() Default Constructor **XMLParser** public XMLParser(java.lang.String filename, java.lang.String outname) Process Constructor **Parameters**: filename - String outname - String

c.轉換結果部份範例

```
<?xml version="1.0" encoding="UTF-8" ?>
- <CLUSTALW type="dnd">
  - <level>
    - <level>
      - <level>
        - <level>
          - <level>
            - <level>
              - <level>
                - <level>
                  - <level>
                    gi|18599218|ref|XP_002914.4|:-0.04593,
                    <level>gi|17865154|gb|AAL47160.1|AF45:-0.01106,gi|17865154|gb|AAL47160.1|AF45:-
                      0.01106,gi|1709489|sp|P54790|ORC3_YEAS:0.0110</level>
                    gi|18599218|ref|XP_002914.4|:-0.04593,:0.0459
                  </level>
                  :0.02878,:0.02878,gi|6981604|ref|NP_037172.1|:-0.0287
                </level>
                :0.02051,:0.02051,gi|6320339|ref|NP_010419.1|:-0.0205
              </level>
              :0.10035,:0.10035,gi|461721|sp|Q00553|CFTR_MACMU:-0.1003
            </level>
            :0.07261,:0.07261,gi|116140|sp|P26361|CFTR_MOUSE:-0.0726
          </level>
          :0.22213,

            <level>gi|14141185|ref|NP_066388.1|:0.02551,gi|14141185|ref|NP_066388.1|:0.02551,gi|461723|sp|P34158|CFTR_RAT
          :0.22213,:0.0267
        </level>
        :0.05177,
      - <level>
        - <level>
```

2.2.8 Input 格式

a. 未轉換前範例格式

```
>gi|18599218|ref|XP_002914.4| ATP-binding cassette, sub-family C (CFTR/MRP), member 5 [Homo sapiens]
MKDIDIGKEYIIPSPGYRSVRERTSTSGTHRDREDSKFRRTRPLECQDALETAARAEGLSLDASMHSQLR
ILDEEHPKGKYHHGLSALKPIRTTSKHQHPVDNAGLFSCMTFSWLSSLARVAHKKGELSMEDVWSLSKHE
SSDVNCRRLERLWQEELNEVGPDAASLRRVVWIFCRTRLILSIVCLMITQLAGFSGPAFMVKHLLEYTQA
TESNLQYSLLLVLGLLLTEIVRSWSLALTWALNYRTGVRLRGAILTMAFKKILKLKNIKEKSLGELINIC
SNDGQRMFEAAAVGSLLAGGPVVAILGMIYNVIILGPTGFLGSAVFILFYPAMMFASRLTAYFRRKCVAA
TDERVQKMNEVLTYIKFIKMYAWVKAFSQSVQKIREEERRILEKAGYFQSITVGVAPIVVVIASVVTFSV
HMTLGFDLTAAQAFTVVTVFNSMTFALKVTPFSVKSLSEASVAVDRFKSLFLMEEVHMIKNKPASPHIKI
EMKNATLAWDSSHSSIQNSPKLTPKMKKDKRASRGKKEKVRQLQRTEHQAVLAEQKGHLLLDSDERPSPE
EEEGKHIHLGHLRLQRTLHSIDLEIQEGKLVGICGSVGSGKTSLISAILGQMTLLEGSIAISGTFAYVAQ
QAWILNATLRDNILFGKEYDEERYNSVLNSCCLRPDLAILPSSDLTEIGERGANLSGGQRQRISLARALY
SDRSIYILDDPLSALDAHVGNHIFNSAIRKHLKSKTVLFVTHQLQYLVDCDEVIFMKEGCITERGTHEEL
MNLNGDYATIFNNLLLGETPPVEINSKKETSGSQKKSQDKGPKTGSVKKEKAVKPEEGQLVQLEEKGQGS
VPWSVYGVYIQAAGGPLAFLVIMALFMLNVGSTAFSTWWLSYWIKQGSGNTTVTRGNETSVSDSMKDNPH
MQYYASIYALSMAVMLILKAIRGVVFVKGTLRASSRLHDELFRRILRSPMKFFDTTPTGRILNRFSKDMD
EVDVRLPFQAEMFIQNVILVFFCVGMIAGVFPWFLVAVGPLVILFSVLHIVSRVLIRELKRLDNITQSPF
LSHITSSIQGLATIHAYNKGQEFLHRYQELLDDNQAPFFLFTCAMRWLAVRLDLISIALITTTGLMIVLM
HGQIPPAYAGLAISYAVQLTGLFQFTVRLASETEARFTSVERINHYIKTLSLEAPARIKNKAPSPDWPQE
GEVTFENAEMRYRENLPLVLKKVSFTIKPKEKIGIVGRTGSGKSSLGMALFRLVELSGGCIKIDGVRISD
IGLADLRSKLSIIPQEPVLFSGTVRSNLDPFNQYTEDQIWDALERTHMKECIAQLPLKLESEVMENGDNF
SVGERQLLCIARALLRHCKILILDEATAAMDTETDLLIQETIREAFADCTMLTIAHRLHTVLGSDRIMVL
AQGQVVEFDTPSVLLSNDSSRFYAMFAAAENKVAVKG
>gi|18574439|ref|XP_083829.1| ATP-binding cassette, sub-family C (CFTR/MRP), member 2 [Homo sapiens]
MLEKFCNSTFWNSSFLDSPEADLPLCFEQTVLVWIPLGYLWLLAPWQLLHVYKSRTKRSSTTKLYLAKQV
…
```

Name Block

Content Block

b.轉換程式

Class 路徑

```
org.twbbs.hellokitty.clustalw.XMLParser.input.XMLParser.XMLParser
```

使用說明

```
java
org.twbbs.hellokitty.clustalw.XMLParser.input.XMLParser.XMLParser
[input file name] + [Output file name]
```

使用範例

```
java
org.twbbs.hellokitty.clustalw.XMLParser.input.XMLParser.XMLParser
CFTR_input.out input.xml
```

API

XMLParser
public XMLParser()
Default Constructor

XMLParser
public XMLParser(java.lang.String filename,
java.lang.String outname)
Process Constructor
Parameters:
filename - String
outname - String

c.轉換結果部份範例

```
<?xml version="1.0" encoding="UTF-8" ?>
- <CLUSTALW type="input">
  - <source>
      <parameter>gi</parameter>
      <parameter>18599218</parameter>
      <parameter>ref</parameter>
      <parameter>XP_002914.4</parameter>
      <description>ATP-binding cassette, sub-family C (CFTR/MRP), member 5 [Homo sapiens]</description>

        <content>MKDIDIGKEYIIPSPGYRSVRERTSTSGTHRDREDSKFRRTRPLECQDALETAARAEGLSLDASMHSQLRILDEEHPKGKYHHGLSALKPIRTT
    </source>
  - <source>
      <parameter>gi</parameter>
      <parameter>18574439</parameter>
      <parameter>ref</parameter>
      <parameter>XP_083829.1</parameter>
      <description>ATP-binding cassette, sub-family C (CFTR/MRP), member 2 [Homo sapiens]</description>

        <content>MLEKFCNSTFWNSSFLDSPEADLPLCFEQTVLVWIPLGYLWLLAPWQLLHVYKSRTKRSSTTKLYLAKQVFVGFLLILAAIELALVLTEDSGQ
    </source>
  - <source>
      <parameter>gi</parameter>
      <parameter>14753227</parameter>
      <parameter>ref</parameter>
```

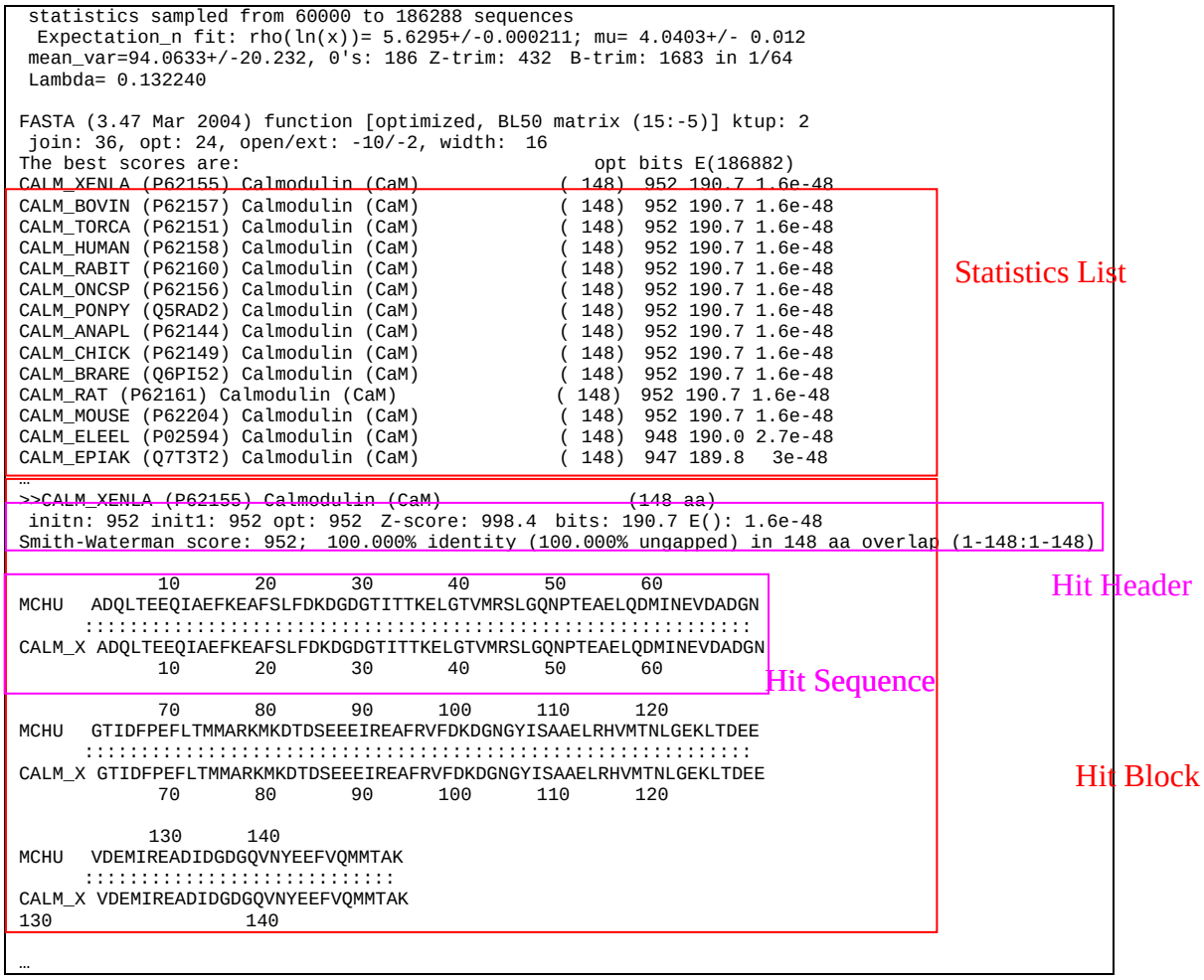
2.3 FASTA 格式轉換

FastA 格式比較複雜，但是包含許多資訊。來源檔中可以分成許多部分來看。以下是來源檔的分析。

```
  1>>>MCHU - Calmodulin - Human, rabbit, bovine, rat, and chicken 148 aa

67622695 residues in 186882 sequences
```

Header Information



b.轉換程式

Class 路徑

org.twbbs.hellokitty.fasta.XMLParser.XMLParser

使用說明

java org.twbbs.hellokitty.fasta.XMLParser.XMLParser [input file name] + [Output file name]

使用範例

java org.twbbs.hellokitty.fasta.XMLParser.XMLParser fasta.out fasta.xml

API

XMLParser

public XMLParser()

Default Constructor

XMLParser

public XMLParser(java.lang.String filename,
java.lang.String outname)

Process Constructor

Parameters:

filename - String

outname - String

c.轉換結果部份範例

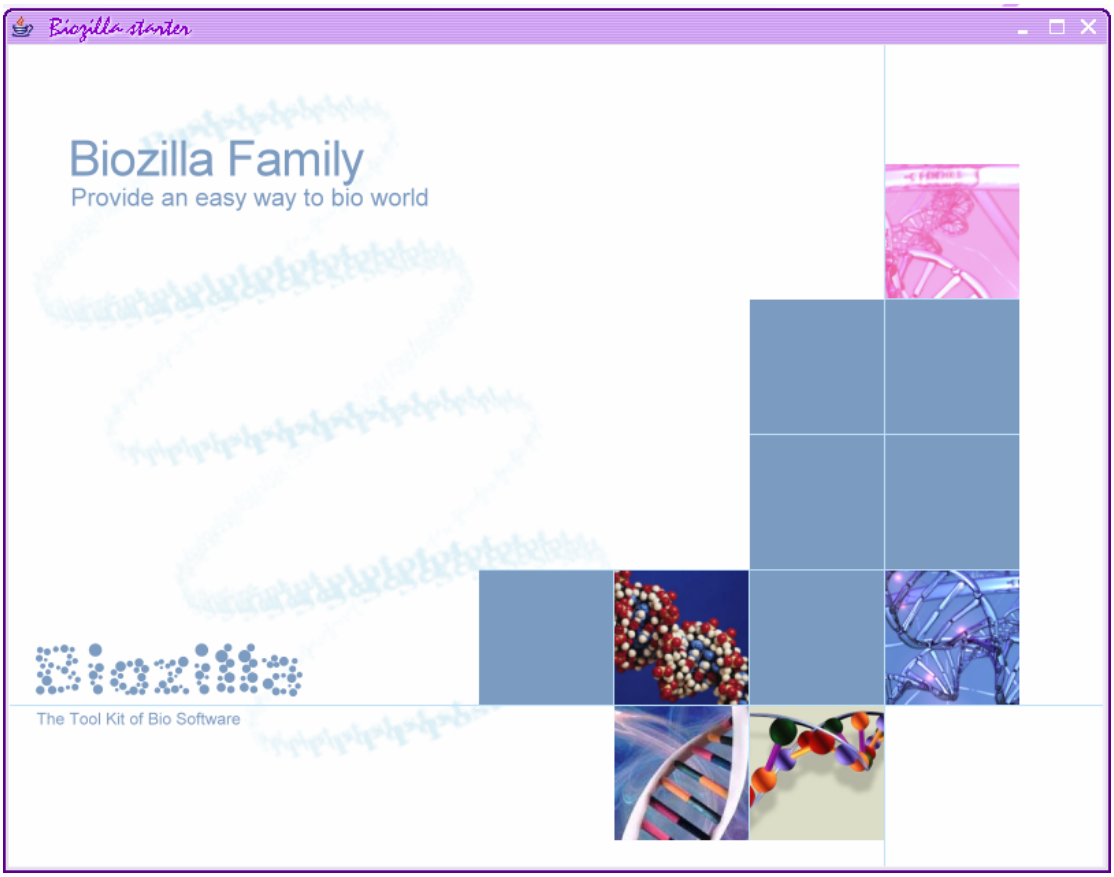

```xml
<?xml version="1.0" encoding="UTF-8" ?>
- <fast_report>
  - <statistics>
      <comment>statistics sampled from 60000 to 186288 sequences</comment>
      <rho>5.6295+/-0.000211;</rho>
      <mu>4.0403+/-+/-0.012</mu>
      <mean_var>r=94.0633+/-20.232,</mean_var>
      <zeros>186</zeros>
      <Ztrim>432</Ztrim>
      <Btrim>1683</Btrim>
      <in>1/64</in>
      <Lambda>0.132240</Lambda>
    </statistics>
  - <parameters>
      <matrix>BL50</matrix>
      <ktup>2</ktup>
      <join>36,</join>
      <opt>24,</opt>
      <gap_pen>-10/-2,</gap_pen>
      <width>16</width>
    </parameters>
    <best />
  - <hits>
    - <hit header="CALM_XENLA (P62155) Calmodulin (CaM) (148 aa)" initn="952" init1="952" opt="952" zscore="998.4" bits="190.7"
        E="1.6e-48" Smith_Waterman="952" identity="100.000%" indentitiy_ungapped="100.000%">
      - <overlap>
          <length>148</length>
          <from>1-148</from>
          <to>1-148</to>
        </overlap>
      - <alignment>
          <alignment_query>ADQLTEEQIAEFKEAFSLFDKDGDGTITTKELGTVMRSLGQNPTEAELQDMINEVDADGNGTIDFPEFLTMMARKMKDTDSEEE
          <alignment_subject>ADQLTEEQIAEFKEAFSLFDKDGDGTITTKELGTVMRSLGQNPTEAELQDMINEVDADGNGTIDFPEFLTMMARKMKDTDSEE
        </alignment>
      </hit>
    - <hit header="CALM_BOVIN (P62157) Calmodulin (CaM) (148 aa)" initn="952" init1="952" opt="952" zscore="998.4" bits="190.7"
        E="1.6e-48" Smith_Waterman="952" identity="100.000%" indentitiy_ungapped="100.000%">
      - <overlap>
          <length>148</length>
          <from>1-148</from>
          <to>1-148</to>
        </overlap>
      - <alignment>
          <marks>::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::</marks>
          <alignment_query>ADQLTEEQIAEFKEAFSLFDKDGDGTITTKELGTVMRSLGQNPTEAELQDMINEVDADGNGTIDFPEFLTMMARKMKDTDSEEE
          <alignment_subject>ADQLTEEQIAEFKEAFSLFDKDGDGTITTKELGTVMRSLGQNPTEAELQDMINEVDADGNGTIDFPEFLTMMARKMKDTDSEE
        </alignment>
      </hit>
    - <hit header="CALM_TORCA (P62151) Calmodulin (CaM) (148 aa)" initn="952" init1="952" opt="952" zscore="998.4" bits="190.7"
        E="1.6e-48" Smith_Waterman="952" identity="100.000%" indentitiy_ungapped="100.000%">
      - <overlap>
          <length>148</length>
          <from>1-148</from>
          <to>1-148</to>
        </overlap>
      - <alignment>
          <marks>::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::</marks>
          <alignment_query>ADQLTEEQIAEFKEAFSLFDKDGDGTITTKELGTVMRSLGQNPTEAELQDMINEVDADGNGTIDFPEFLTMMARKMKDTDSEEE
```

3. Biozilla Family

3.1 Biozilla

Biozilla 是一套針對生物軟體及網格所開發出來的協力工具集合。裡面包含了 Sequence Generator、2XML Parser、Input Sequence Spilter、Input Sequence Linker、BioCosmos。軟體中主要需求為簡單好用且功能完整，並搭配上親和近人的軟體介面。對於從事生物基因研究人員可以說是不可或缺的工具。

Biozilla 中藉由啟動 starter，可選擇所需要的相關工具。如下圖。



以下將針對各工具做詳盡介紹。

3.2 Biozilla Family-Sequence Generator

3.2.1 軟體介紹

Sequence Generator 是用來自訂隨機產生的序列，並且產生對相容於 Input Sequence Linker 所需要的 XML 格式。當想要對生物軟體比對的序列作測試時，便可以使用此序列產生器。藉由預先描述好的 XML 設定檔或是手動產生出規定的大小及 XML 檔，便可以結合 Input Sequence Linker 做序列合併成新的 Input 檔作為生物軟體的輸入使用。

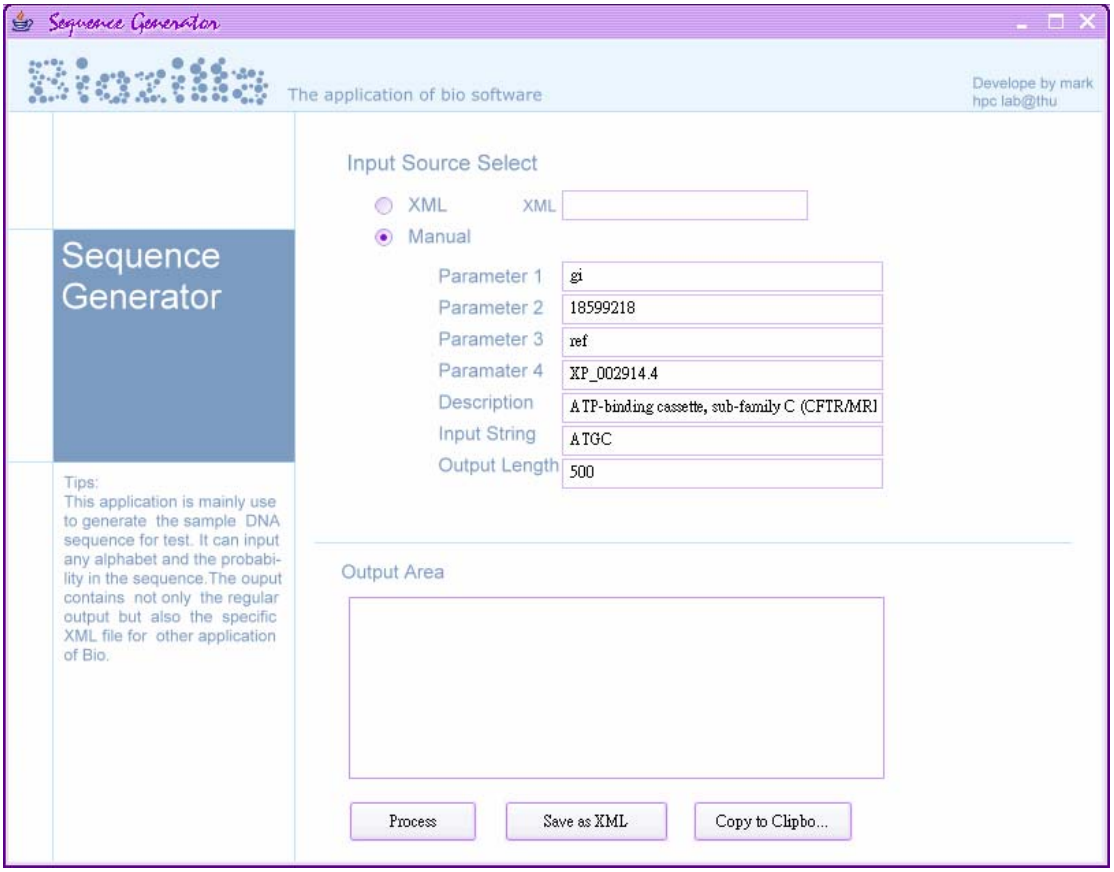
3.2.2 軟體特性

- ✓ 自訂標頭說明。
- ✓ 自訂產生字元。
- ✓ 自訂產生長度。
- ✓ 自訂各字串產生機率。
- ✓ 自訂 XML 描述檔。
- ✓ 產生 XML 序列檔可供再利用。

3.2.3 軟體使用

使用手動輸入

Step 1.軟體畫面如下。選擇 Manual，表示由手動輸入。

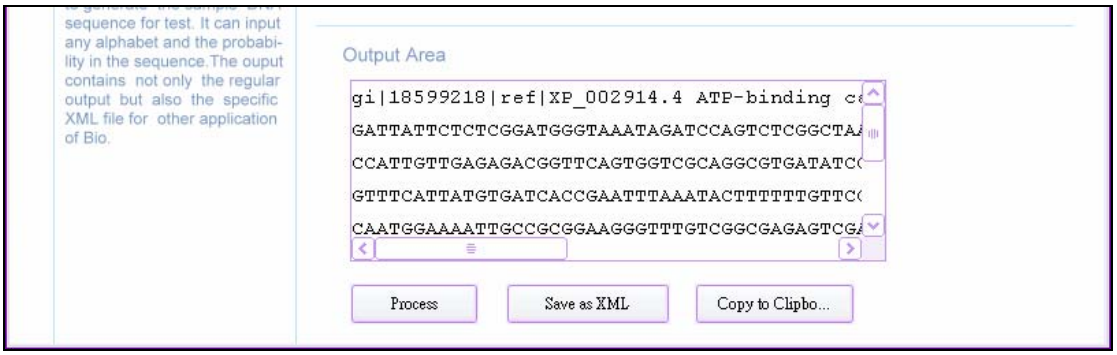


Step 2.各輸入參數說明如下。

Parameter 1	序列之識別參數，通常為 gi。
Parameter 2	序列之識別參數，通常為一串數字。
Parameter 3	序列之識別參數。
Parameter 4	序列之識別參數。
Description	對於該序列的描述。
Input String	想要參與隨機產生的字元，輸入為字串程式會自動分析。
Output Length	輸出的字串長度。

在此手動模式下，所有字元產生機率均相等。若想要針對每個字元作機率設定，應採用 XML 描述檔方式輸入。

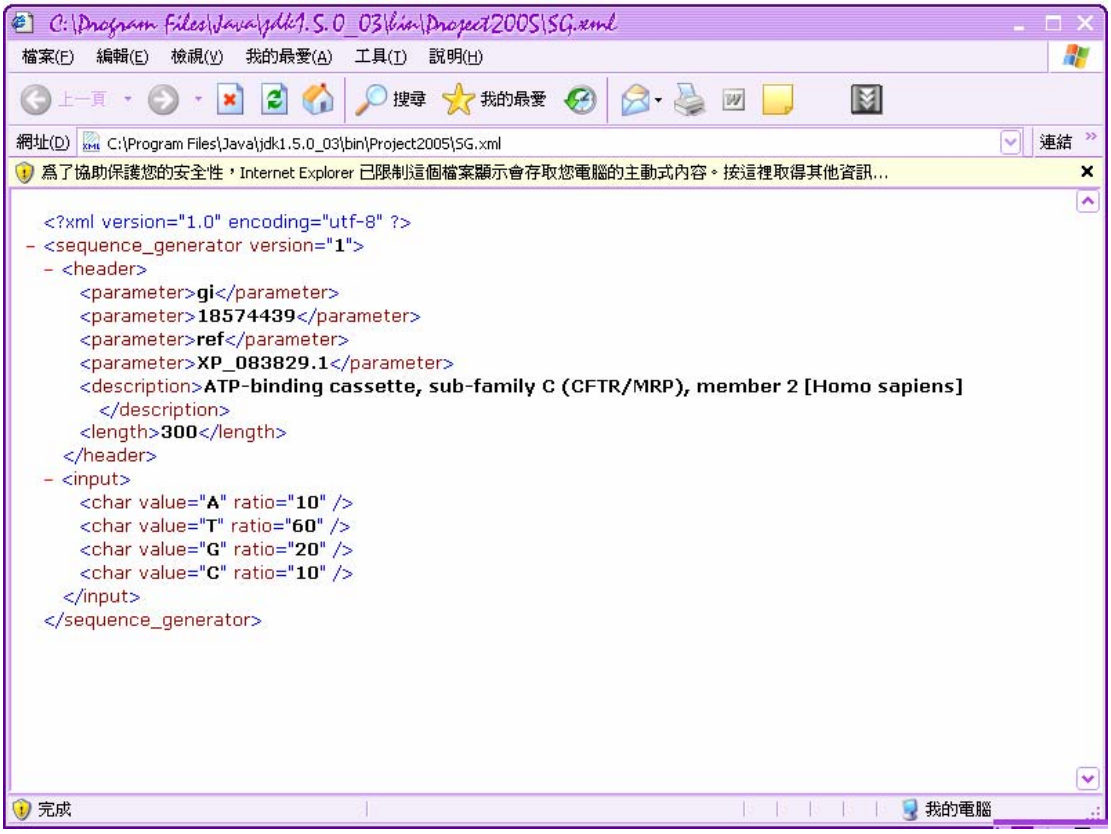
Step 3.按下 Process 按鈕，產生之結果如下圖。



Step 4.按下 Save as XML 可以儲存為 Input Sequence Linker 所用的 XML。Copy to Clipboard 為複製到剪貼簿供使用者再利用。

使用 XML 描述檔作部署

Step 1.XML 範例描述檔如下。



各元素簡介如下。

<sequence_generator>	Root 元素，為最頂端元素。version=”1”表示此為版本 1 之文件。

<header>	標頭元素。裡面包含的是各個參數以及描述，亦包含產生字串的長度。
<parameter>	序列參數。
<description>	序列描述。
<length>	序列長度。
<input>	字元輸入元素。包含所有字元描述。
<char>	想要顯示的字元。value=”A”表示欲產生字元為 A，ratio=”10”表示所佔比率為 10。

Step 2.在 Input Source Select 選擇 XML，並填入該 XML 描述檔名稱。

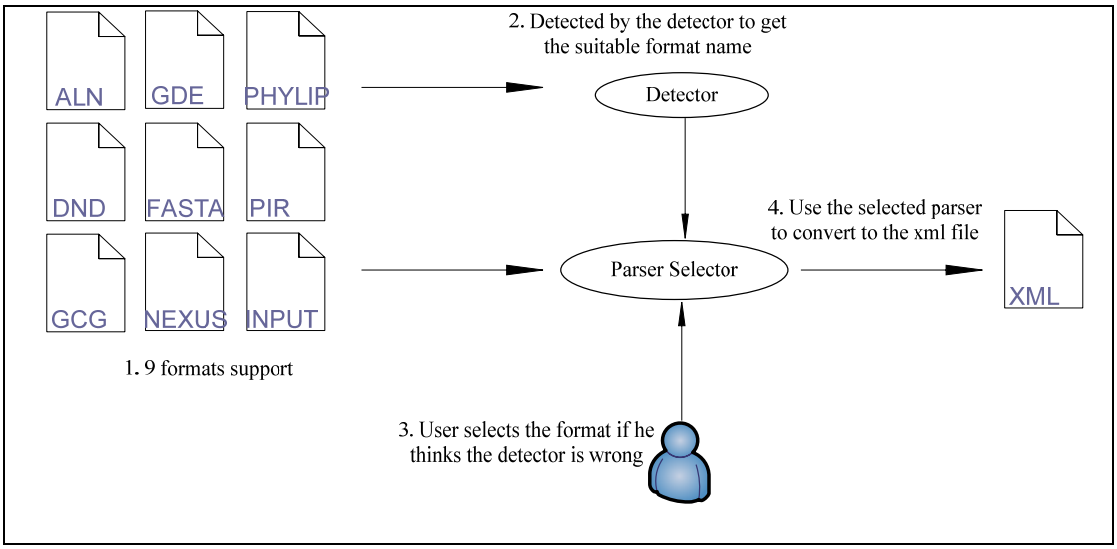
Step 3.按下 Process 按鈕。所產生之操作與 Manual 相同。

3.3 Biozilla Family-2XML Parser Selector

3.3.1 軟體介紹

2XML Parser Selector 為將生物軟體的輸入輸出檔轉為自訂規格的 XML 檔，並賦予自動判斷來源檔為何種格式。由於 XML 具有跨平台以及高度可利用性，將文字檔案轉為 XML 可方便作為之後使用。

軟體執行流程如下：



3.3.2 軟體特性

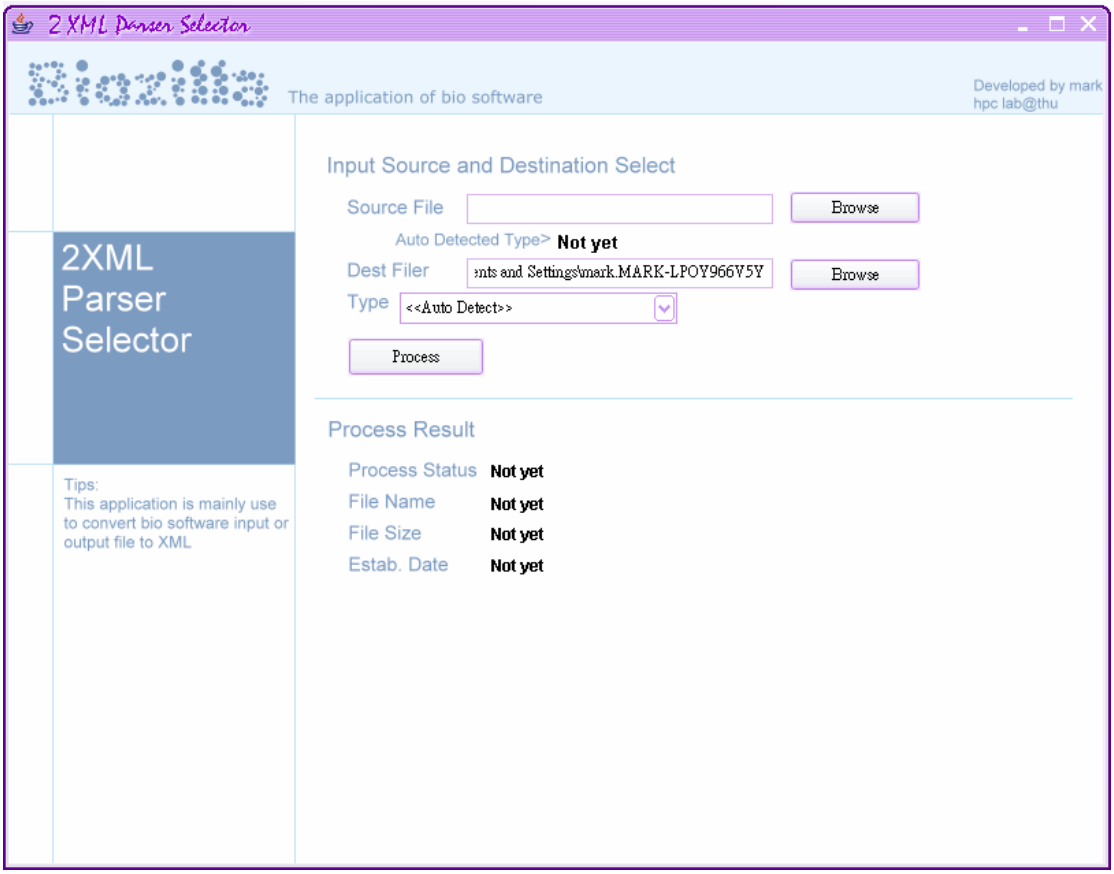
✓ 整合 ALN、INPUT、DND、FASTA、GCG、GDE、NEXUS、 PHYLIP

、PIR 格式轉換。

✓ 自動判別格式功能。

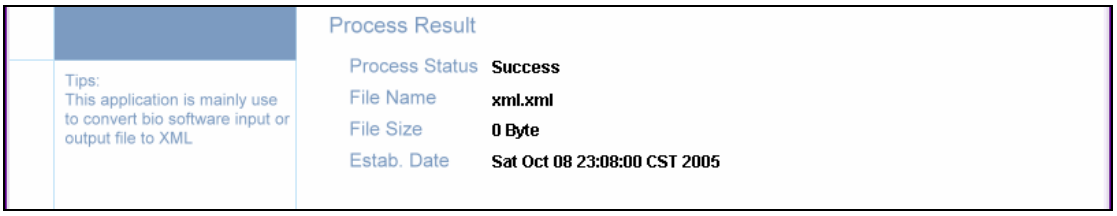
3.3.3 軟體使用

Step 1. 軟體執行畫面如下。Source File 內使用 Browse 選擇檔案。也可以使用拖曳方式將檔案拉入視窗內，程式自動會加入該檔案。



Step 2.選定檔案後，程式會自動判斷該格式，並且顯示出來。圖例中使用檔案為 GDE 格式，程式自動判斷為 GDE。若判斷為不正確，可自行選定適當的格式做轉換。

Step 3.設定好存檔名稱後，按下 Process 按鈕即會開始轉換。轉換完成後會顯示該轉換狀態、檔案名稱、大小、以及建立日期。

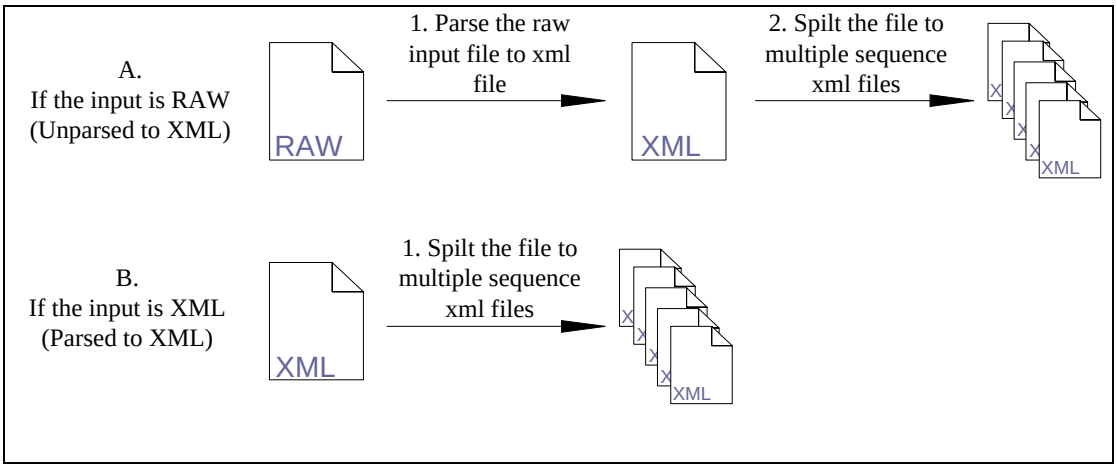


3.4 Biozilla Family-.Input Sequence Spilter

3.4.1 軟體介紹

.Input Sequence Spilter 為將標準的 ClustalW input 檔打散成許多 XML 序列檔。由於 input 檔是由許多基因序列組成的，藉由打散成許多單一的檔案可以將這些序列檔經由 Input Sequence Linker 再利用組合成自訂的 Input 檔。這樣一來不僅提高了序列的可利用度，在生物軟體應用上也有相當大的助益。

圖表 2 程式流程圖



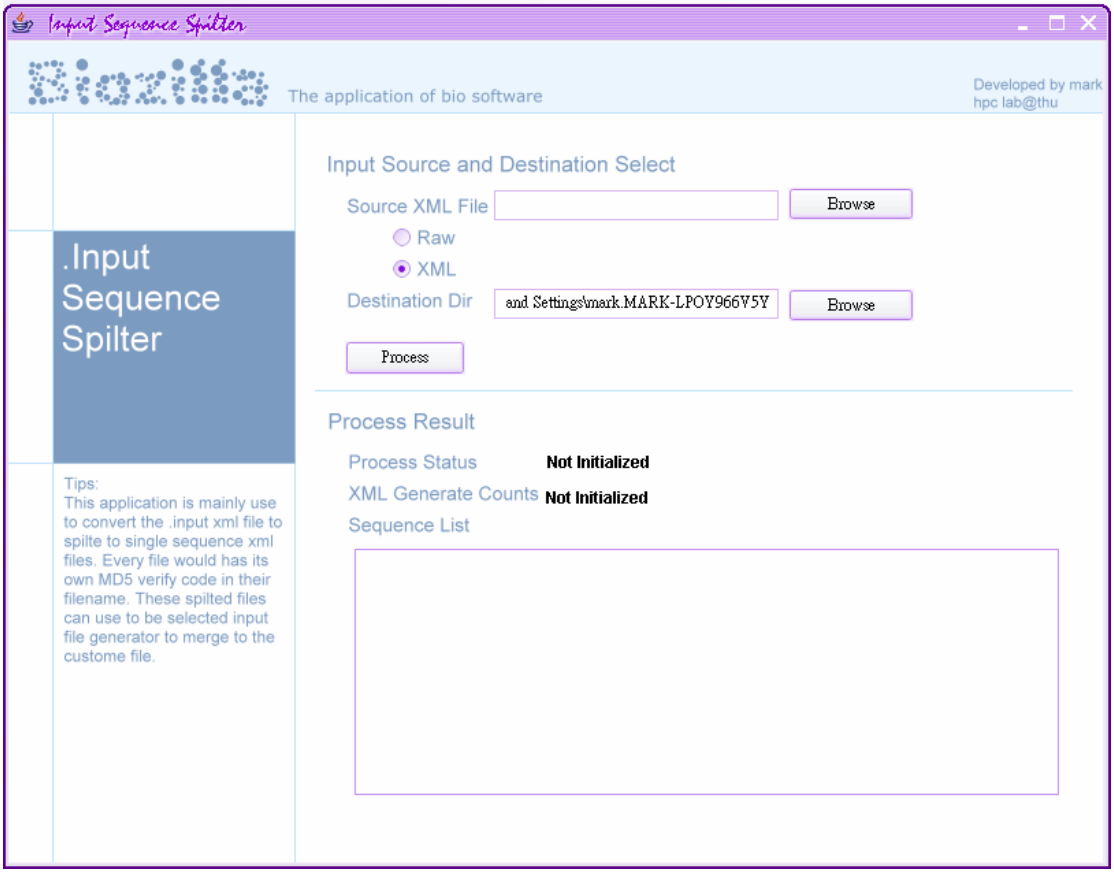
3.4.2 軟體特性

- ✓ 支援原始 input 格式以及經過 XMLParser 過後的來源檔。
- ✓ 自動分析來源檔並輸出多個可再利用 XML 檔案。
- ✓ 輸出檔可供 Input Sequence Linker 再利用。
- ✓ 輸出檔名附加了相關描述，使辨識檔案簡單明瞭。
- ✓ 輸出檔名附加 MD5 編碼，便於檢測序列是否有人為更改過。

3.4.3 軟體使用

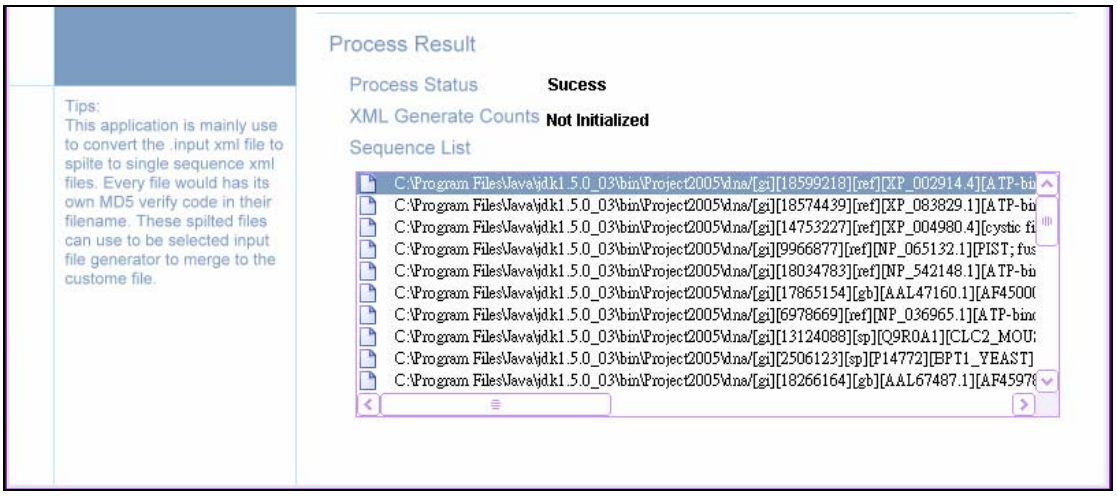
Step 1.程式執行畫面如下。來源檔可以選擇是 Raw(原始檔)或是已經處理過的

XML 檔。Raw 僅是多進行一道 XML 轉換的工作而已。在此來源檔皆可以使用檔案拖曳方式輸入。

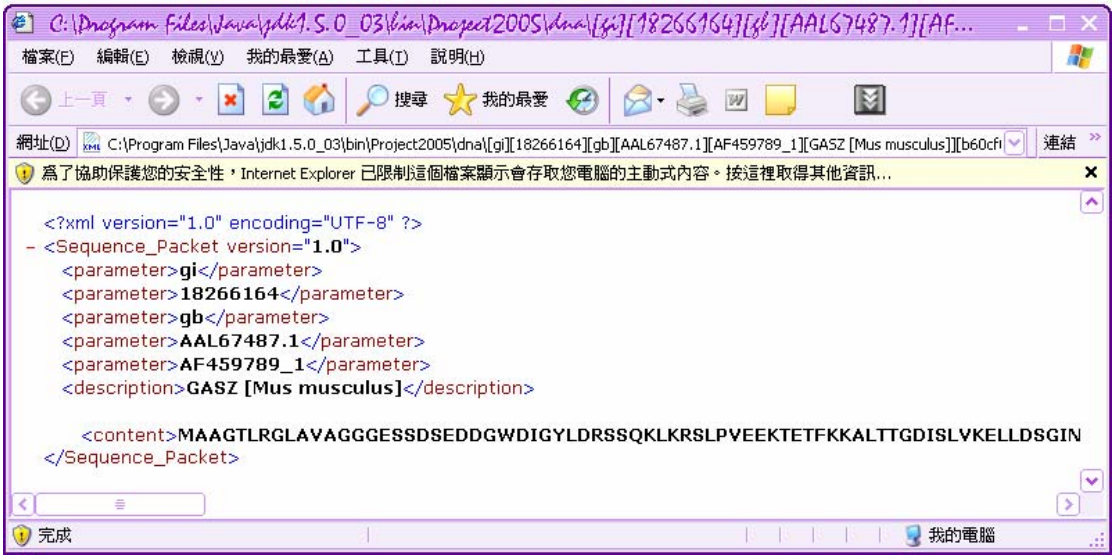


Step 2.選定輸出目錄。由於產生的量是根據來源檔之內容，因此必須使用目錄選擇。

Step 3.按下 Process 後，在 Process Result 區域會顯示相關訊息，包括產生的檔案列表。



輸出 XML 格式簡介，範例如下圖



XML 元素說明。

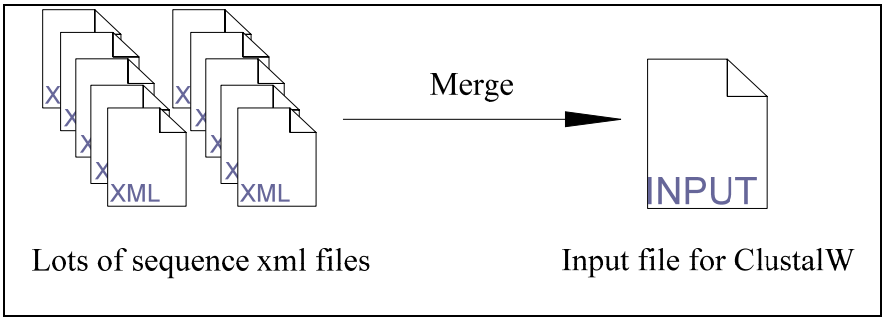
<Sequence_Packet>	Root 元素。Version=”1.0”為版本編號。
<parameter>	序列參數。
<description>	序列描述。
<content>	序列內容。

3.5 Biozilla Family-Input Sequence Linker

3.5.1 軟體簡介

Input Sequence Linker 主要功能為產生自訂的 input 檔案供生物軟體使用。輸入檔案為 Input Sequence Spilter 以及 Sequence Generator 所產生的 XML 檔。

圖表 3 軟體基本運作流程。



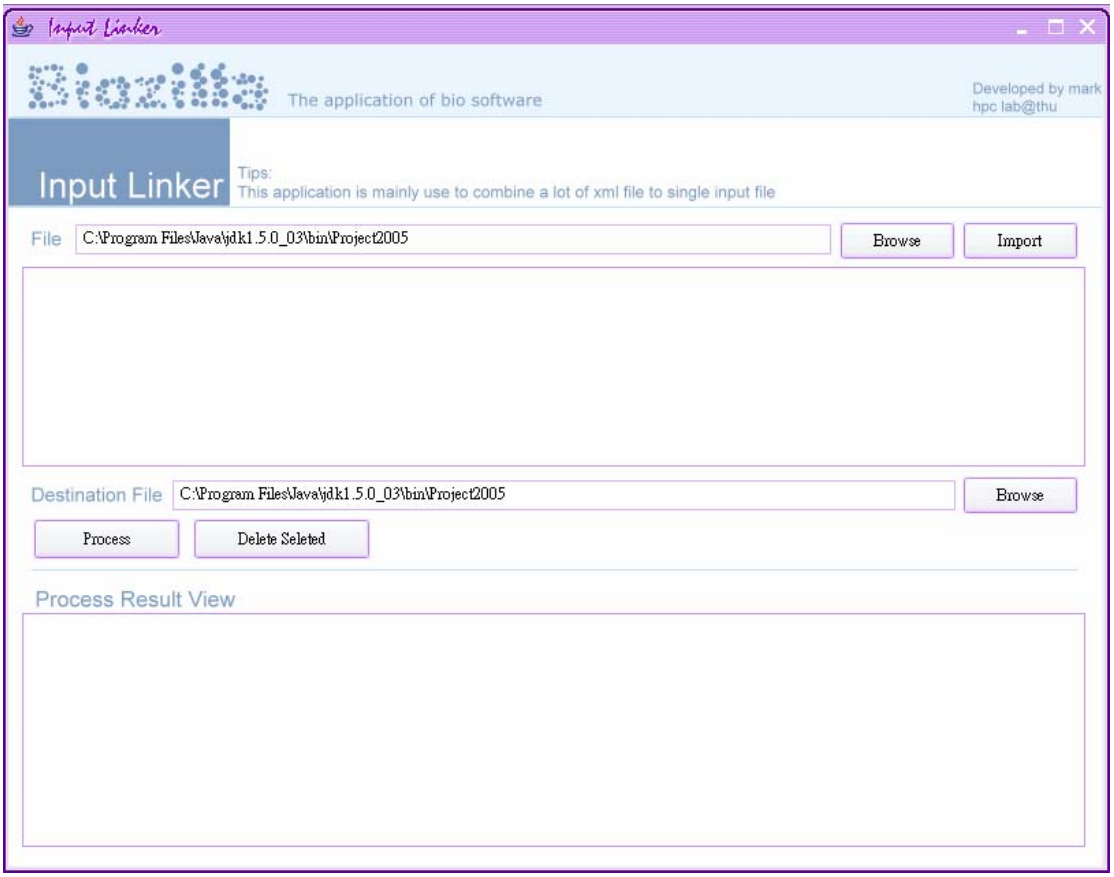
3.5.2 軟體特性

✓ 產生符合生物軟體輸入格式的 input 檔。

✓ 介面操作容易，可輕易增加許多檔案供合併。

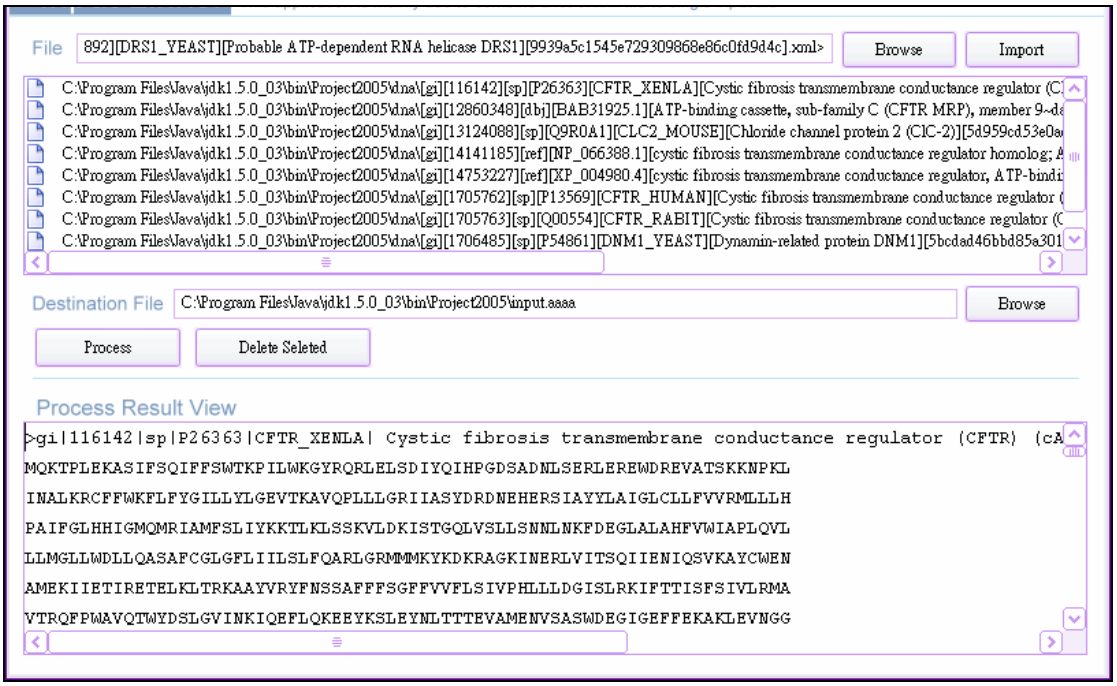
3.5.3 軟體使用

Step 1.軟體畫面如下



Step 2. 選擇合適的輸入檔案。可以使用瀏覽方式加入多個檔案，使用方式是先按下 Browse 選擇許多檔案，接下來利用 import 按鈕加入清單中。或是使用拖曳方式將多個檔案一次拖入視窗中。

Step 3. 設定好輸出檔案位置後，按下 Process 按鈕即可以開始進行 input 檔生成動作。程式中設定序列長度為 70 字元時自動換行。檔案產生完畢時會在結果視窗顯示轉換後檔案的內容。

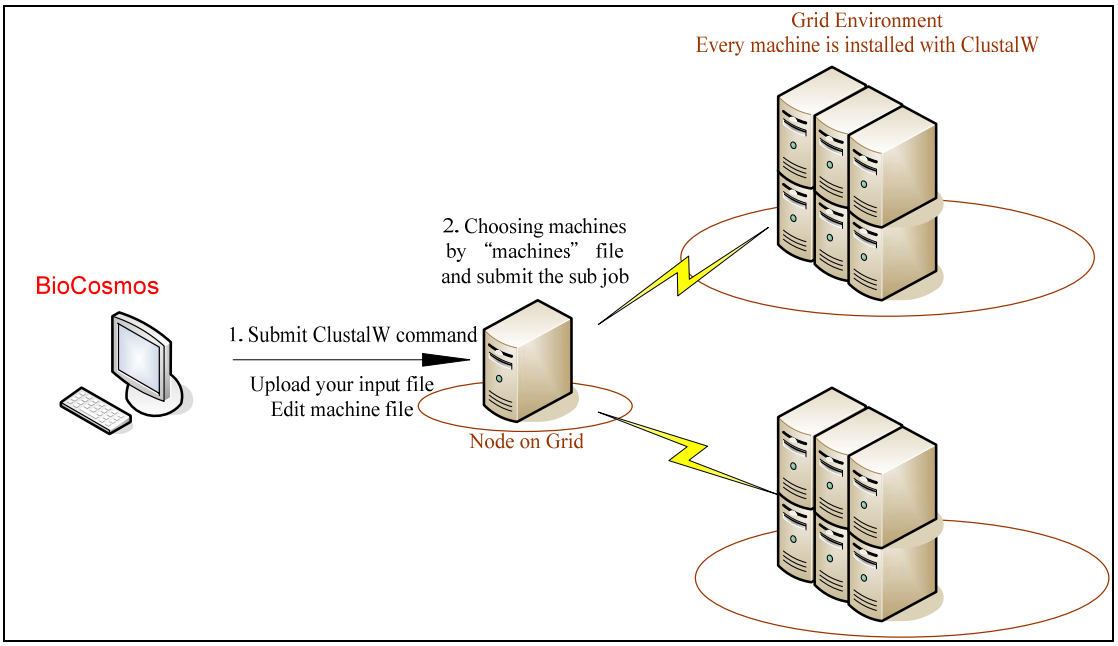


3.6 Biozilla-BioCosmos

3.6.1 軟體簡介

BioCosmos 是可以利用 Grid 執行生物軟體的簡單強大工具。透過洗鍊的 GUI 讓使用者能非常容易上手，不會因為需要執行許多系統方面的指令而感到懼怕。另外也針對執行結果做了分析。對於專心從事生物領域研究的人，利用 Grid 加速基因比對，可以說相當有助益。

圖表 4 軟體執行流程圖

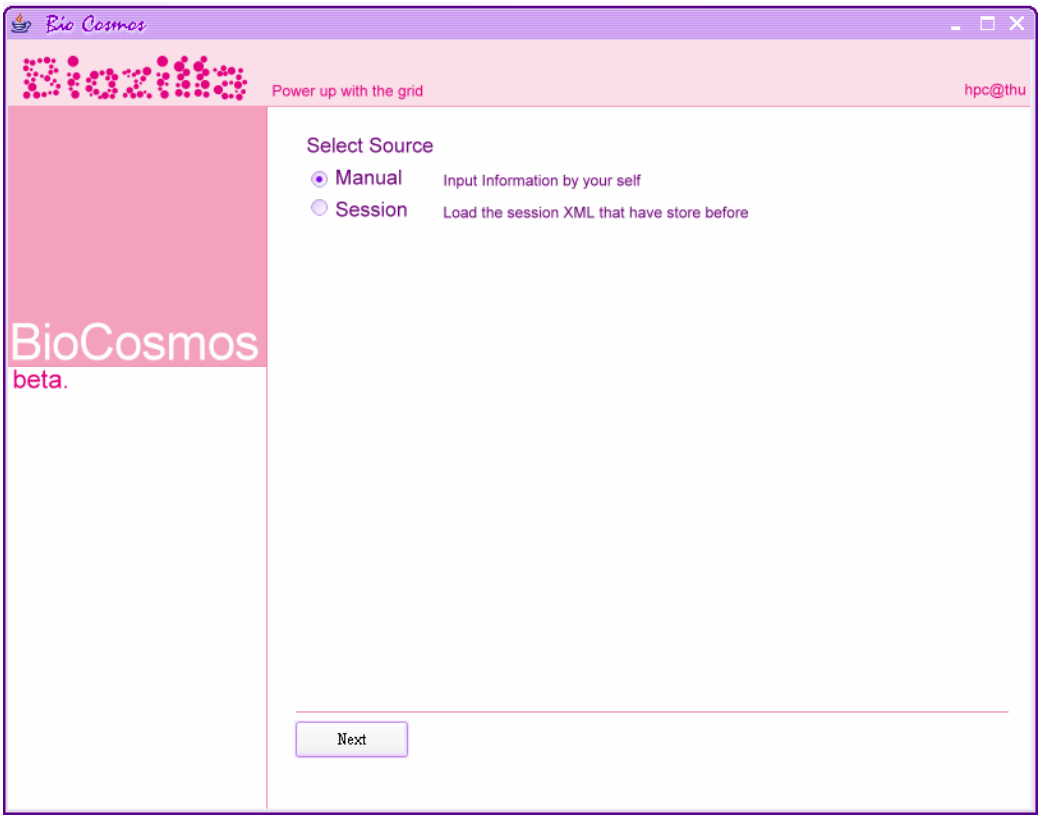


3.6.2 軟體特性

- ✓ 結合 Grid 技術與 ClustalW，基因比對在彈指之間。
- ✓ 使用介面親和力強。
- ✓ 可儲存或讀取每次執行的 Session 狀態。
- ✓ 可動態設定 ClustalW 執行目錄。
- ✓ 可動態針對該機器修改 ClustalW 的 machines 設定。
- ✓ 可上傳序列至 ClustalW 供軟體執行。
- ✓ 可分析執行結果。

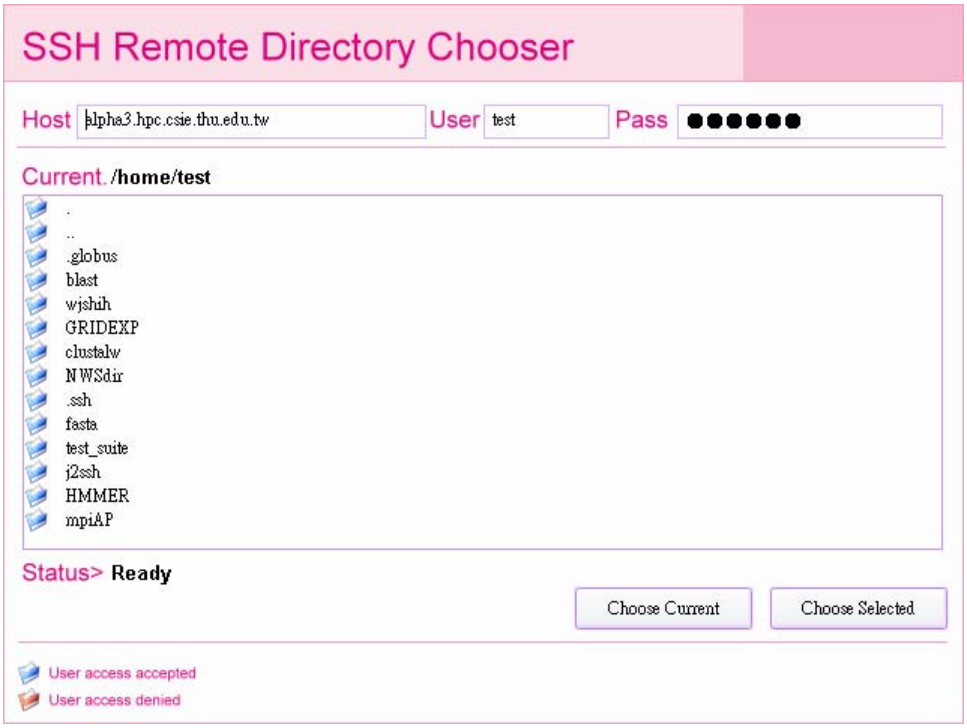
3.6.3 軟體使用

Step 1.軟體啟動執行如下。Manual 是指手動輸入所有訊息。Session 是指說可以讀取之前儲存的 Session 狀態。在這裡使用 Manual 對執行進行設定。按下 Next 進入下一個畫面。

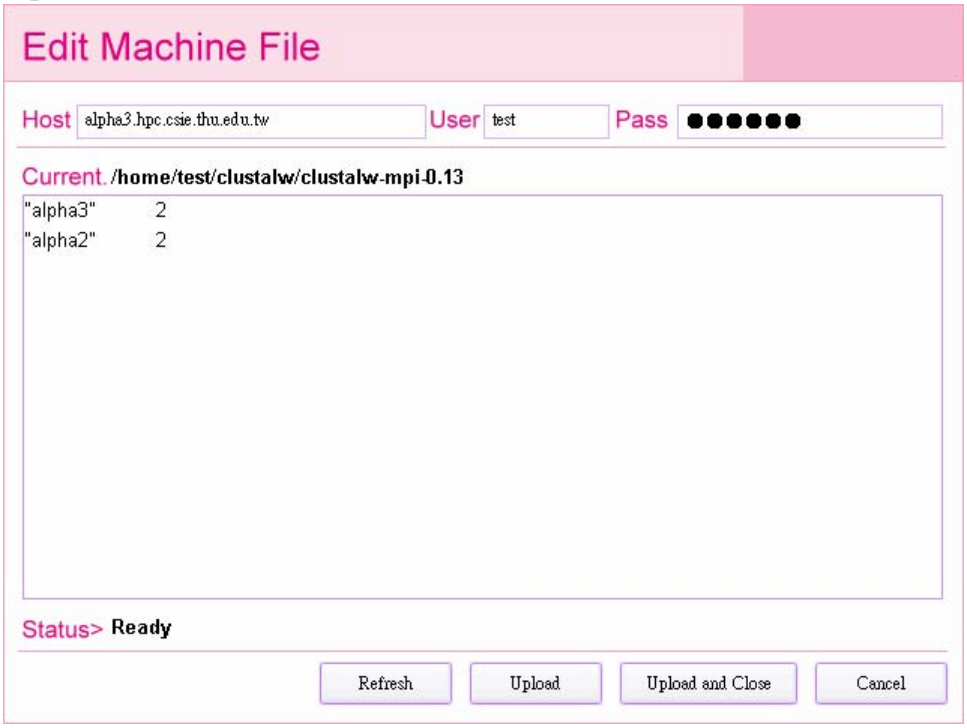


Step 2.在這裡中可以針對機器、Grid 環境、ClustalW 參數做設定，分別填入相關資訊。

若執行的工作目錄不可確定，可以使用 Work dir.旁的 Browse 按鈕開啟對話框選擇遠端執行目錄。如下圖。

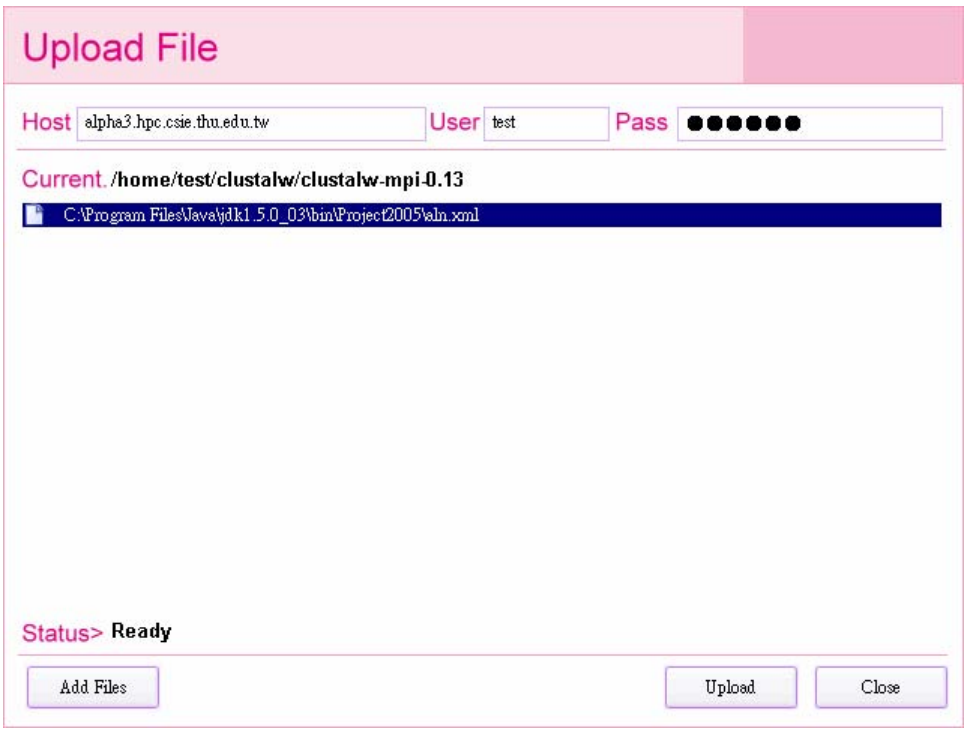


另外，提供了 Machines 編輯工具可同步針對該機器編輯 Grid 中協同工作的節點。在設定好 Host、User、Pass、Work dir.之後，按下左下的 Edit Machine File，便產生即時編輯視窗。如下圖。程式會自動讀取遠端 machines 內容。編輯完後可按 Upload 更新遠端 machines 內容。



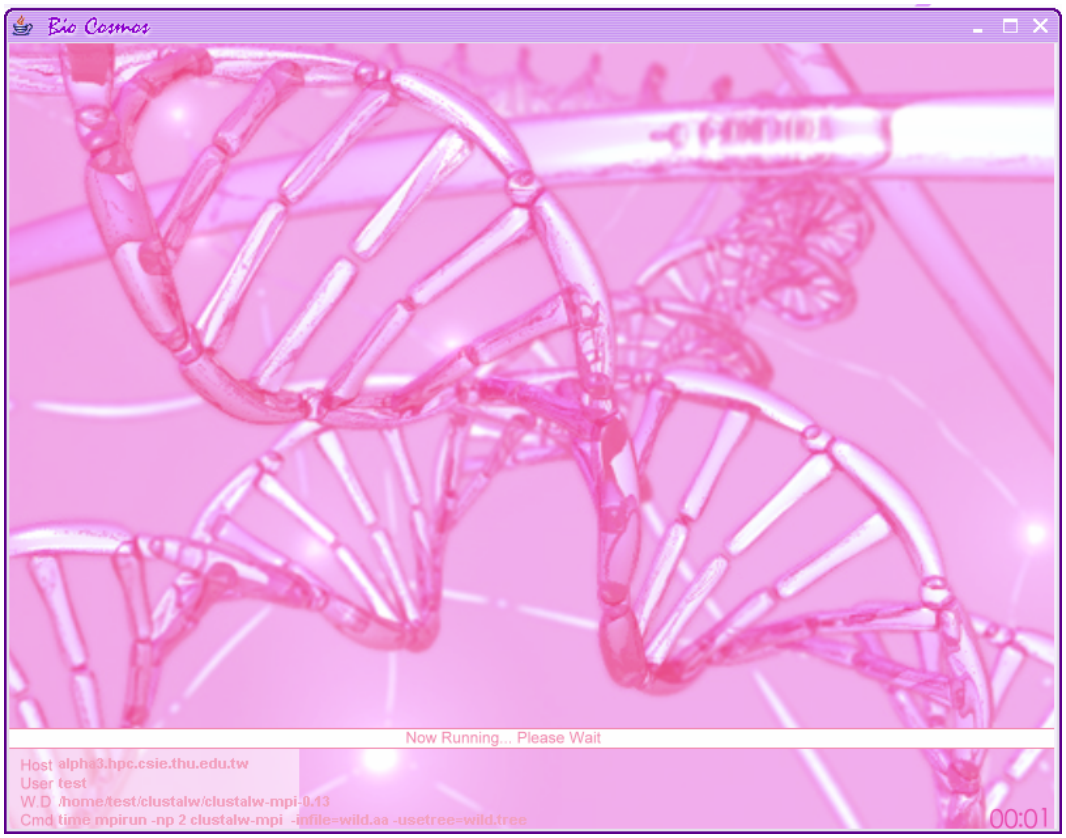
若有基因序列檔或是 Tree 檔案想要上傳，可利用左下 Upload File 按鈕開啟上傳

視窗。如下圖。可選擇 Add Files 一次選多個檔案上傳。按下 Upload 即可上傳檔案。



將一切資訊填入完畢之後。主畫面中 Session 按鈕可儲存目前所填入的資訊。按下後會彈出下圖對話視窗。將想要儲存的名稱填入即可。

Step 3.按下 Next 即便進入了執行視窗。執行中會顯示機器、使用者、目錄及指令資訊。並有經過時間計時器的提示。如下圖。



Step 4. 執行完畢時，會出現執行結果。如下圖。按下 Next 即可針對結果進行進一步分析。

結論

在將來除了將 Resource Broker 移植成 Web Service，在 Biozilla 方面也將結合 Resource Broker 的 Web Service 提供一體化的執行作業，並針對執行結果作更加詳盡的數字及圖表分析。在建置環境上也將提供好用的通用工具提供多台環境的佈署。

PART II MPI 生物資訊軟體

1.生物資訊學概要

1.1 序列比對和資料庫搜索

比較是科學研究中最常見的方法，通過將研究物件相互比較來尋找物件可能具備的特性。在生物資訊學研究中，比對是最常用和最經典的研究手段。

最常見的比對是蛋白質序列之間或核酸序列之間的兩兩比對，通過比較兩個序列之間的相似區域和保守性位元點，尋找二者可能的分子進化關係。進一步的比對是將多個蛋白質或核酸同時進行比較，尋找這些有進化關係的序列之間共同的保守區域、位元點和 profile，從而探索導致它們產生共同功能的序列模式。此外，還可以把蛋白質序列與核酸序列相比來探索核酸序列可能的表達框架；把蛋白質序列與具有三維結構資訊的蛋白質相比，從而獲得蛋白質折疊類型的資訊。

比對還是資料庫搜索演算法的基礎，將查詢序列與整個資料庫的所有序列進行比對，從資料庫中獲得與其最相似序列的已有的資料，能最快速的獲得有關查詢序列的大量有價值的參考資訊，對於進一步分析其結構和功能都會有很大的幫助。近年來隨著生物資訊學資料大量積累和生物學知識的整理，通過比對方法可以有效地分析和預測一些新發現基因的功能。

1.2 序列兩兩比對

序列比對的理論基礎是進化學說，如果兩個序列之間具有足夠的相似性，就推測二者可能有共同的進化祖先，經過序列內殘基的替換、殘基或序列片段的缺失、以及序列重組等遺傳變異過程分別演化而來。序列相似和序列同源是不同的概念，序列之間的相似程度是可以量化的參數，而序列是否同源需要有進化事實的驗證。通過大量實驗和序列比對的分析，一般認為蛋白質的結構和功能比序列具有更大的保守性，因此粗略的說，如果序列之間的相似性超過 30%，它們就很可能是同源的。

FASTA 是第一個被廣泛應用的序列比對和搜索工具包，包含若干個獨立的程式。FASTA 為了提供序列搜索的速度，會先建立序列片段的“字典”，查詢序列先會在字典裡搜索可能的匹配序列，字典中的序列長度由 ktup 參數控制，缺省的 ktup=2。FASTA 的結果報告中會給出每個搜索到的序列與查詢序列的最佳比對結果，以及這個比對的統計學顯著性評估 E 值。BLAST 是現在應用最廣泛的序列相似性搜索工具，相比 FASTA 有更多改進，速度更快，並建立在嚴格的統計學

基礎之上。

1.3 多序列比對

顧名思義，多序列比對就是把兩條以上可能有系統進化關係的序列進行比對的方法。目前對多序列比對的研究還在不斷前進中，現有的大多數演算法都基於漸進的比對的思想，在序列兩兩比對的基礎上逐步優化多序列比對的結果。進行多序列比對後可以對比對結果進行進一步處理，例如構建序列模式的 profile，將序列聚類構建分子進化樹等等。

目前使用最廣泛的多序列比對程式是 CLUSTALW(它的 PC 版本是 CLUSTALX)。CLUSTALW 是一種漸進的比對方法，先將多個序列兩兩比對構建距離矩陣，反應序列之間兩兩關係；然後根據距離矩陣計算產生系統進化指導樹，對關係密切的序列進行加權；然後從最緊密的兩條序列開始，逐步引入臨近的序列並不斷重新構建比對，直到所有序列都被加入為止。

CLUSTALW 對輸入序列的格式比較靈活，可以是前面介紹過的 FASTA 格式，還可以是 PIR、SWISS-PROT、GDE、Clustal、GCG/MSF、RSF 等格式。輸出格式也可以選擇，有 ALN、GCG、PHYLIP 和 GDE 等，用戶可以根據自己的需要選擇合適的輸出格式。

用 CLUSTALW 得到的多序列比對結果中，所有序列排列在一起，並以特定的符號代表各個位元點上殘基的保守性，“*”號表示保守性極高的殘基位點；“.”號代表保守性略低的殘基位點。

◆ *已安裝且正常運作之 MPI 生物資訊軟體*

(1) **mpiBLAST :**

是一套在蛋白質資料庫或 DNA 資料庫中進行相似性比較的分析工具，能迅速與公開資料庫進行相似性序列比較。所產生出的結果中的得分是對一種對相似性的統計說明。採用一種局部的演算法獲得兩個序列中具有相似性的序列。BLAST 是現在應用最廣泛的序列相似性搜索工具，相比 FASTA 有更多改進，速度更快，並建立在嚴格的統計學基礎之上。

(2) **FASTA :**

是一個蛋白質或 DNA 的快速且具高靈敏度的比對程式，速度與靈敏度可藉由設定 ktup 參數由使用者自行調控。其功能類似 BLAST，與 BLAST 的差異在於 FASTA 的搜尋速度較慢，但靈敏度較 BLAST 高。

(3) **ClustalW :**

CLUSTALW 是一個 DNA 與蛋白質多條序列比對程式, 它可從不同序列的比對中找出具有生物意義的序列。CLUSTALW 可從多條序列中計算出最適當的比對方式，並且排列出來，使使用者異於觀察序列的一致性，相似度與不同處。序列間的演化關係可經由設定 Output 格式獲得演化關係圖。

CLUSTALW 是一種漸進的比對方法，先將多個序列兩兩比對構建距離矩陣，反應序列之間兩兩關係；然後根據距離矩陣計算產生系統進化指導樹，對關係密切的序列進行加權；然後從最緊密的兩條序列開始，逐步引入臨近的序列並不斷重新構建比對，直到所有序列都被加入為止。

◆ *各套軟體所跑的程式*

(1) **mpiBLAST :**

Program	Query sequence type	Database sequence type	Alignment sequence type
blastn	nucleotide	nucleotide	nucleotide
blastp	protein	protein	protein
blastx	nucleotide	protein	protein
tblastn	protein	nucleotide	protein
tblastx	nucleotide	nucleotide	protein

比對類型	資料庫	查詢序列	簡述
blastp	蛋白質	蛋白質	可能找到具有遠源進化關係的匹配序列。
blastn	核苷酸	核苷酸	適合尋找分值較高的匹配，不適合遠源關係。
blastx	蛋白質	核苷酸	適合新 DNA 序列和 EST 序列的分析。
tblastn	核苷酸	蛋白質	適合尋找資料庫中尚未標注的編碼區。
tblastx	核苷酸	核苷酸	適合分析 EST 序列。

(2) **FASTA :**

fasta, mpfasta	mp34compfa	Run the fasta program. It compares nucleotide sequences against nucleotide databases or protein sequences against protein databases.
fastx, mpfastx	mp34compfx	Run the fastx program. It compares a DNA sequence to a protein database in three forward or reverse frames. Uses a simpler algorithm than fasty; runs faster than fasty.
fasty,	mp34compfy	Run the fasty program. It compares a DNA sequence

mpfasty		to a protein database in three forward or reverse frames. Uses a more complex algorithm than fastx and runs more slowly.
tfastx, mptfastx	mp34comptfx	Compares a protein sequence to a nucleotide database. Simpler and faster algorithm than tfasty.
tfasty, mptfasty	mp34comptfy	Compares a protein sequence to a nucleotide database. More complex and slower algorithm than tfastx.
fasts, mpfasts	mp34compfs	Run the fasts program. It compares a short peptide sequence to a protein database.
tfasts, mptfasts	mp34comptfs	Compare a short protein sequence to a nucleotide database.
ssearch, mpcompsw	mp34compsw	Use the Smith-Waterman algorithm to query nucleotides sequences against a nucleotide data bases or to query protein sequences against protein databases.
Programs that "summarize the effectiveness of a search" and "require super-family-labeled databases".		
mscompfa	ms34compfa	Use for comparisons made using fasta.
mscompfx	ms34compfx	Use for comparisons made using fastx.
mscompfy	ms34compfy	Use for comparisons using fasty.
mscomptfx	ms34comptfx	Use for comparisons made using tfastx.
mscomptfy	ms34comptfy	Use for comparisons using tfasty.
mscompsw	ms34compsw	Use for comparisons using ssearch.
mscompss	ms34compss	Use for comparisons using ssearch. Allows for higher gap penalties than mscompsw allows.
"Programs to report the scores and alignments of the highest scoring unrelated sequence (require super-family-labeled databases). These programs are used to evaluate super-family labelling."		
mucompfa	mu34compfa	Use for comparisons made using fasta.
mucompfx	mu34compfx	Use for comparisons made using fastx.
mucompfy	mu34compfy	Use for comparisons using fasty.
mucomptfx	mu34comptfx	Use for comparisons made using tfastx.
mucomptfy	mu34comptfy	Use for comparisons using tfasty.
mucompsw	mu34compsw	Use for comparisons using ssearch.

(3) **ClustalW :**

只有單一指令 clustalw-mpi，其餘皆由指定參數完成複雜工作

DATA (sequences)

-INFILE=file.ext:	input sequences.
-PROFILE1=file.ext and -PROFILE2=file.ext :	profiles (old alignment).

VERBS (do things)	
-OPTIONS :	list the command line parameters
-HELP or -CHECK :	outline the command line params.
-ALIGN :	do full multiple alignment.
-TREE :	calculate NJ tree.
-BOOTSTRAP(=n) :	bootstrap a NJ tree (n= number of bootstraps; def. = 1000).
-CONVERT :	output the input sequences in a different file format.

PARAMETERS (set things)	
*****General settings:******	
-INTERACTIVE :	read command line, then enter normal interactive menus
-QUICKTREE :	use FAST algorithm for the alignment guide tree
-TYPE= :	PROTEIN or DNA sequences
-NEGATIVE :	protein alignment with negative values in matrix
-OUTFILE= :	sequence alignment file name
-OUTPUT= :	GCG, GDE, PHYLIP, PIR or NEXUS
-OUTORDER= :	INPUT or ALIGNED
-CASE :	LOWER or UPPER (for GDE output only)
-SEQNOS= :	OFF or ON (for Clustal output only)
-SEQNO_RANGE=:	OFF or ON (NEW: for all output formats)
-RANGE=m,n :	sequence range to write starting m to m+n.
*****Fast Pairwise Alignments:*****	
-KTUPLE=n :	word size
-TOPDIAGS=n :	number of best diags.
-WINDOW=n :	window around best diags.
-PAIRGAP=n :	gap penalty
-SCORE :	PERCENT or ABSOLUTE
*****Slow Pairwise Alignments:*****	
-PWMATRIX= :	Protein weight matrix=BLOSUM, PAM, GONNET, ID or filename
-PWDNAMATRIX= :	DNA weight matrix=IUB, CLUSTALW or filename

-PWGAPOPEN=f :	gap opening penalty
-PWGAPEXT=f :	gap opening penalty
*****Multiple Alignments:*****	
-NEWTREE= :	file for new guide tree
-USETREE= :	file for old guide tree
-MATRIX= :	Protein weight matrix=BLOSUM, PAM, GONNET, ID or filename
-DNAMATRIX= :	DNA weight matrix=IUB, CLUSTALW or filename
-GAPOPEN=f :	gap opening penalty
-GAPEXT=f :	gap extension penalty
-ENDGAPS :	no end gap separation pen.
-GAPDIST=n :	gap separation pen. range
-NOPGAP :	residue-specific gaps off
-NOHGAP :	hydrophilic gaps off
-HGAPRESIDUES= :	list hydrophilic res.
-MAXDIV=n :	% ident. for delay
-TYPE= :	PROTEIN or DNA
-TRANSWEIGHT=f :	transitions weighting
*****Profile Alignments:*****	
-PROFILE :	Merge two alignments by profile alignment
-NEWTREE1= :	file for new guide tree for profile1
-NEWTREE2= :	file for new guide tree for profile2
-USETREE1= :	file for old guide tree for profile1
-USETREE2= :	file for old guide tree for profile2
*****Sequence to Profile Alignments:*****	
-SEQUENCES :	Sequentially add profile2 sequences to profile1 alignment
-NEWTREE= :	file for new guide tree
-USETREE= :	file for old guide tree
*****Structure Alignments:*****	
-NOSECSTR1 :	do not use secondary structure-gap penalty mask for profile 1
-NOSECSTR2 :	do not use secondary structure-gap penalty mask for profile 2
-SECSTROUT=STRUCTURE or MASK or BOTH or NONE :	output in alignment file

-HELIXGAP=n :	gap penalty for helix core residues
-STRANDGAP=n :	gap penalty for strand core residues
-LOOPGAP=n :	gap penalty for loop regions
-TERMINALGAP=n :	gap penalty for structure termini
-HELIXENDIN=n :	number of residues inside helix to be treated as terminal
-HELIXENDOUT=n :	number of residues outside helix to be treated as terminal
-STRANDENDIN=n :	number of residues inside strand to be treated as terminal
-STRANDENDOUT=n:	number of residues outside strand to be treated as terminal
*****Trees:*****	
-OUTPUTTREE=nj OR phylip OR dist OR nexus	
-SEED=n :	seed number for bootstraps.
-KIMURA :	use Kimura's correction.
-TOSSGAPS :	ignore positions with gaps.
-BOOTLABELS=node OR branch :	position of bootstrap values in tree display

2 各套軟體的輸入格式(INPUT FORMAT)

2.1 mpiBLAST

尚不確定，但大多數為 FASTA 格式的檔案。

2.2 FASTA

The fasta3 programs know about three kinds of sequence files(four under VMS): (1) plain sequence files - files that contain nothing but sequence residues - can only be used as query sequences. (2) FASTA format files. These are the same as plain sequence files, each sequence is preceded by a comment line with a '>' in the first column. (3)

distributed sequence libraries (this is a broad class that includes the NBRF/PIR VMS and blocked ascii formats, Genbank flat-file format, EMBL flat-file format, and Intelligenetics format. All of t he files that you create should be of type (1) or (2). FASTA format files (ones with a '>' and comment before the sequence) are preferred, because they can be used as query or library sequence files by all of the programs. I have included several sample test files, *.aa and *.seq as well as two small sequence libraries, prot_test.lib and gst.nlib. The first line may begin with a '>' by a comment. Spaces and tabs (and anything else that is not an amino-acid code) are ignored.

2.3 ClustalW

All sequences must be in 1 file, one after another. 7 formats are automatically recognised: NBRF-PIR, EMBL-SWISSPROT, Pearson (Fasta), Clustal (*.aln), GCG-MSF (Pileup), GCG9-RSF and GDE flat file. All non-alphabetic characters (spaces, digits, punctuation marks) are ignored except "-" which is used to indicate a GAP ("." in MSF-RSF).

3. 各套軟體的輸出格式(OUTPUT FORMAT)

3.1 mpiBLAST :

產生出的檔案其檔名及格式自訂。

3.2 FASTA :

輸出的檔案同 mpiblast 一樣可自訂其格式和檔名，且其產生出的結果分成三個部份:

(I) A histogram
(II) A list of related sequences
(III) A list of sequence alignments

3.3 ClustalW :

6 different alignment formats (CLUSTAL, GCG, NBRF-PIR, PHYLIP, GDE, NEXUS, and FASTA).

4. MPI 生物資訊軟體安裝說明

4.1 mpiBLAST 安裝

There are five steps to installing mpiBLAST from source
(1) Install MPI (if not already installed)
(2) Download mpiBLAST and the matching NCBI Toolbox release
(3) Patch the NCBI Toolbox with the mpiBLAST patch and compile it
(4) Compile and install mpiBLAST
(5) Configure mpiBLAST by editing the ~/.ncbirc file

1. **Installation of NCBI toolbox**
   1.1 In order to compile mpiBLAST, we need some libraries of NCBI toolbox

   1.2 Getting the source tarball from NCBI
   ftp://ftp.ncbi.nih.gov/toolbox/ncbi_tools/old/20040204/ncbi.tar.gz

   1.3 Extracting the source tarball
   $ tar -xzvf ncbi.tar.gz

   1.4 Installing NCBI toolbox
   $ ./ncbi/make/makedis.csh

2. **Installation of MPICH-g2**
   2.1 In order to compile mpiBLAST, we need mpich libraries for C and FRTRAN compiler on GlobusToolkit

   2.2 Getting the mpich-1.2.6 source tarball and extracting it
   http://www.hpc.csie.thu.edu.tw/doc/GT3.0.2%20Installation/mpich1.2.6.tar.gz

   2.3 Installing MPICH-g2
   $ ./configure --with-arch=LINUX --with-device=globus2:--flavor=gcc32dbg \ --prefix=<INSTALLATION_PREFIX>
   $ make
   $ make install

3. **Installation of mpiBLAST package**

3.1 Getting the source tarball and extracting
http://mpiblast.lanl.gov/releases/mpiBLAST-1.2.1.tar.gz

3.2 Configuring mpiBLAST
$./configure --prefix=/path/to/mpiBLAST-1.2.1 --with-ncbi=/path/to/ncbi

--prefix=/path/to/install/directory —
Specifies the location where mpiBLAST will be installed

--with-ncbi=/path/to/ncbi_toolbox/ —
Specifies the path of NCBI Toolbox installation
[N.B.] If the *--prefix* configuration option is not specified, mpiBLAST will be placed in */usr/local*

3.3 Building mpiBLAST
$ make

3.4 Installing mpiBLAST
$ make install

3.5 Edit the mpiblast.conf in the /mpiBLAST/etc
/blast/db/share/
/blast/db/yeast_db/

3.6 Edit the ~/.ncbirc configuration file in the user's home directory

[NCBI]
Data=/path/to/shared/storage/data

3.7 Getting the database you want
ftp://ftp.ncbi.nih.gov/blast/db/

3.8 Formatting a database Before processing blast queries the sequence database must be formatted with mpiformatdb

mpiformatdb -N 25 -i nt

3.9 ex. yeast.nt

$ mpiformatdb -f /usr/local/mpiBLAST-1.2.1/etc/mpiblast.conf -N 12 -i yeast.nt -o T -p F

-N The number of fragment you want to divide
-i Database name
-o Parse options
T – True: Parse SeqId and create indexes
F – False: Do not parse SeqId. Do not create indexes

-p Type of file
T – protein
F – nucleotide

3.10 Querying the database

$ mpirun -np nodes mpiblast --config -file=/path/to/mpiblast.conf -p blastn -d nt – i blast_query.fas -o blast_results.txt

-p Program Name [String]
-d Database [String]
default = nr
-i Query File [File In]
default = stdin
-e Expectation value (E) [Real]
default = 10.0
-m alignment view options:
7 = XML Blast output,
default = 0
-o BLAST report Output File [File Out] Optional
default = stdout

4.2FASTA 安裝

1. Getting the source tarball and extracting it
ftp://ftp.virginia.edu/pub/fasta/fasta34t23b8a.shar.Z
$ zcat fasta34t23b8a.shar.Z | sh

2. Edit Makefile.mpi4

   --- for lam/mpi---
   MPI_ROOT = /etc/lam
   PLIB = -L${MPI_ROOT}/lib – llam

   ---for mpich---
   MPI_ROOT = /usr/local/mpich
   PLIB = -L${MPI_ROOT}/lib – lmpichg2

3. **$** make -f Makefile.mpi4

4. make all

5. Getting the FASTA database
   ftp://ftp.uniprot.org/pub/databases/uniprot/knowledgebase/uniprot_sprot.fasta.gz

6. **$** gunzip uniprot_sprot.fasta.gz

7. Query sequence
   $ time mpirun -np 2 mp34compfa mwkw.aa uniprot_sprot.fasta

4.3 ClustalW 安裝

1. Make sure you have MPICH or LAM installed on your system

2. Getting the clustalw-mpi-0.13 source tarball and extracting it
   http://web.bii.a-star.edu.sg/~kuobin/clustalw-mpi/clustalw-mpi-0.13.tar.gz
   $ tar -zxvf clustalw-mpi-0.13.tar.gz

3. Edit Makefile

   -------Makefile-------
   CC = mpicc
   CFLAGS = -c –g
   or
   CFLAGS = -c -O3

4. make

5. $mpirun –np n clustalw-mpi –infile=dele.input

5. 附錄

5.1 FASTA 所接參數列表

-a (fasta3, ssearch3 only) show both sequences in their entirety.

-A force Smith -Waterman alignments for fasta3 DNA sequences. By default, only fasta3 protein sequence comparisons useSmith -Waterman alignments.

-B Show normalized score as a z-score, rather than a bit-score in the list of best scores.

-b # Number of sequence scores to be shown on output. In the absence of this option, fasta (and tfasta and ssearch) display all library sequences obtaining similarity scores with expectations less than 10.0 if optimized score are used, or 2.0 if they are not. The -b option can limit the display further, but it will not cause additional sequences to be displayed.

-c # Threshold score for optimization (OPTCUT). Set "-c 1" to optimize every sequence in a database.

-E # Limit the number of scores and alignments shown based on the expected number of scores. Used to override the expectation v alue of 10.0 used by default. When used with -Q, -E 2.0 will show all library sequences with scores with an expectation value <= 2.0.

-d # Maximum number of alignments to be displayed. Ignored if "-Q" is not used.

-F # Limit the number of scores and alignments shown based on the expected number of scores. "-E #" sets the highest E()-value shown; " -F #" sets the lowest E()-value. Thus, "-F 0.0001" will not show any matches or alignments with E() < 0.0001. T his allows one to skip over close relationships in searches f or more distant relationships.

-f Penalty for the first residue in a gap (-12 by default for proteins, -16 for DNA, -15 for FAST[XY]/TFAST[XY]).

-g Penalty for additional residues in a gap (-2 by default for proteins, -4 for DNA, -3 for FAST[XY]/TFAST[XY]).

-h Penalty for frameshift (fastx3/y3, tfastx3/y3 only).

-H Omit histogram.

-i Invert (reverse complement) the query sequence if it is DNA. F or tfasta3/x3/y3, search the reverse complement of the library sequence only.

-j # Penalty for frameshift within a codon (fasty3/tfasty3 only).

-l file
L ocation of library menu file (FASTLIBS).

-L Display more information about the library sequence in the alignment.

-M low-high
R ange of amino acid sequence lengths to be included in the s earch.

-m # Specify alignment type: 0, 1, 2, 3, 4, 5, 6, 9, 10
-m 0 -m 1 -m 2 -m 3 -m 4
MWRTCGPPYT MWRTCGPPYT MWRTCGPPYT
MWRTCGPPYT
::..:: ::: xx X ..KS..Y... MWKSCGYPYT ----------
MWKSCGYPYT MWKSCGYPYT
In addition -m 10 is a new, parseable format for use with other programs. See the file"readme.v20u4" for a more complete description. -m 5 provides a combination of -m 4 and -m 0. -m 6 provides -m 5 plus HTML formatting. -m 9 provides percent identify and coordinates with the initial list of high scores, as well as conventional -m 0 alignments.

-M low-high
I nclude library sequences (proteins only) with lengths b etween low and high.

-n Force the query sequence to be treated as a DNA sequence. T his is particularly useful for query sequences that contain a large number of ambiguous residues, e.g. transcription factor binding sites.

-O Send copy of results to "filename." Helpful for environments without STDOUT (mostly for the Macintosh). -o Turn off default optimization of all scores greater than OPTCUT. Sort results by "initn" scores (reduces the accuracy of statistical estimates).

-p Force query to be treated as protein sequence.

-Q,-q
Q uiet - does not prompt for any input. Writes scores andalignments to the terminal or standard output file. -r Specify match/mismatch scores for DNA comparisons. The d efault is "+5/-4". "+3/-2" can perform better in some cases.

-R file
S ave a results summary line for every sequence in the sequence library.The summary line includes the sequence identifier, superfamily number (if available) position in the library, and the similarity scores calculated. This option can be used to evaluate the sensitivity and s electivity of different search strategies (Pearson, 1995, Pearson, 1998).

-s file
S pecify the scoring matrix file. fasta3 uses the same s coring matrices as Blast1.4/2.0. Several scoring matrixfiles are included in the standard distribution. For protein sequences: codaa.mat - based on minimum mutation matrix; idnaa.mat - identity matrix; pam250.mat - the PAM250 matrix developed by Dayhoff et al. (Dayhoff et al., 1978); p am120.mat - a PAM120 m atrix. The default scoring matrix is BLOSUM50 ("-s BL50"). Other matrices available from within the program are: PAM250/"-s P250", PAM120/"-s P120", PAM40/"-s P40", PAM20/"-s P20", MDM10 - MDM40/"-s M10 - M40" (MDM are modern PAM matrices from Jones et al. (Jones et al., 1992),), BLOSUM50, 62, and 80/"-s BL50", "-s BL62", "-s BL80".

-S Treat lower -case characters in the query or library s equences as"low-complexity" ("seg"-ed) residues. Traditionally, the "seg" program (Wootton and F ederhen, 1993) is used to remove low complexity regions in DNA sequences by replacing the residues with an "X". When t h e "-S" option is used, the FASTA33 programs provide a potentially more informative approach. With "-S", lower case characters in the query or database sequences are treated as "X"'s during the initial scan, but are treated as normal residues during the final alignment display. Since statistical significance is calculated from th e similarity s core calculated during the library search, when the lower case residues are "X"'s, low complexity regions will not produce statistically significant matches. However, if a significant alignment contains low complexity regions, their alignmen is shown. With "-S", lower case characters may be included in the alignment to indicate low complexity regions, and the final alignment score may be higher than the score obtained during the search. T he pseg program can be used to produce databases (or query sequences) with lower case residues indicating low complexity regions using the command: pseg database.fasta -z 1 -q > database.lc_seg (seg can also be used with some post processing, see readme.v33tx.)

-w # Line length (width) = number (<200)

-x # Specify the penalty for a match to an 'X', independently of the PAM matrix. Particularly useful for fastx3/fasty3, where termination codons are encoded as 'X'.

-X Specifies offsets for the beginning of the query and library sequence. For example, if you are comparing upstream regions for two genes, and the first sequence contains 500nt of upstream sequence while the second contains 300 nt of u p s tream sequence, you might try:

fasta -X "-500 -300" seq1.nt seq2.nt

If the -X option is not used, FASTA assumes numbering starts with 1. (You should double check to be certain the negative numberingw orks properly.)

-y Set the width of the band used for calculating "optimized" scores. For proteins and ktup=2, the width is 16. For proteins with ktup=1, the width is 32 by default. For DNA the width is 16.

-z -1,0,1,2,3,4,5

-z -1 turns off statistical calculations. z 0 estimates the significance of the match from the mean and standard deviation of the library scores, without correcting for library sequence length. -z 1 (the default) uses a weighted regression of average score vs library sequence length; -z 2 uses maximum likelihood estimates of Lambda and K; -z 3 uses Altschul-Gish parameters (Altschul and Gish, 1996); -z 4 – 5 uses two variations on the -z 1 strategy. -z 1 and -z 2 are the best methods, in general.

-z 11,12,14,15

e stimate the statistical parameters from shuffled copies of each library sequence. This doubles the time required for a search, but allows accurate statistics to be estimated for libraries comprised of a single protein family.

-Z db_size

set the apparent size of the database to be used when calculating expectation E() values. If you searched a database with 1,000 sequences, but would like to have the E()-values calculated in the context of a 100,000 sequence database, use '-Z 100000'.

-1 sort output by init1 score (for compatibility with FASTP - do not use).

-3 translate only three forward frames