

自動化測試軟體：Rational Robot

陳思豪 曹世強 林盈達

國立交通大學資訊科學系

新竹市大學路1001 號

TEL：(03) 5712121 EXT. 56667

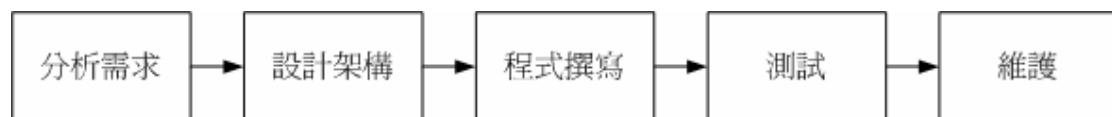
E-MAIL：{cschen, weafon, ydlin}@cis.nctu.edu.tw

摘要

軟體工程的生命週期簡單可分為分析需求，設計架構，程式撰寫，測試，維護五個階段。每個階段都有其對應的軟體開發輔助工具。在測試階段中，為了大量節省人力成本，我們需要能夠做到自動化測試的工具，自動化測試需要軟體能夠記錄測試過程的各種動作，再重複的播放並記下結果。Rational 是一家專門為軟體工程各階段製作開發輔助工具的公司（已被 IBM 併購），其中一項名為 Robot 的產品就是為了自動化測試而設計，其提供多樣的功能如錄製與播放、用 **關鍵字** 來檢查系統資訊，提供的測試模式有 GUI 程式的錄製與播放測試、client/server 架構的程式作多 session 的測試、支援不同 IDE 上的測試等。

1. 簡介

常見的軟體工程流程大約如圖 1 所示，首先必須分析需求，確定開發環境與所使用的工具，擬定整個系統架構，並切割成小區塊，明確定義各區塊的輸出入，然後開始分工撰寫程式。撰寫完成後交付測試，而後成為產品出售。產品賣出後，需要針對客戶的新需求或是使用後發現的問題進行維護。如果能夠在測試階段就



圖一 軟體工程流程

而針對以上的各個階段的需求，在市面上已經有許多輔助軟體開發的工具出現，這些工具恰巧對應於圖一的五個步驟，而可分為底下五種類型：

1. 專案管理工具：提供專案需求訂定，監視專案發展

2. 視覺化模組發展工具: 將許多功能模組化成一個個物件, 開發者用圖表的方式去表現各個物件間的關係, 定下關係之後才撰寫內部的程式。
3. 多人同步版本控制系統: 多人同時修改單一程式時, 會產生多版本控制與衝突的問題, 因此需要一個控制系統來維護。
4. 功能與效能測試系統: 前者須重複地測試來能檢驗出系統穩不穩定, 程式是否有錯。而後者則需要模擬多人同時使用系統, 以找出系統瓶頸, 負載上限等。
5. 除錯器與執行期分析工具: 除錯器可以將程式資訊一五一十的報告出來或是設定中斷點除錯, 而執行期分析工具, 則可以分析出程式執行時的可能瓶頸處, 或那邊可能作了多餘的動作而浪費系統效能。

接下來我們將焦點放在第四點所談到的測試系統上。很明顯的, 可以看出其工作大部分是繁雜而重複性的。因此一個高度自動化的測試軟體是非常必要的。他被期待能很便利的將測試方法, 過程, 及結果記錄下來, 然後自動重複執行這些工作, 如此才能節省大量人力, 避免人為疏失。另外, 這類型的軟體通常還有一個用途, 當我們要確定某個資料來源 (例如某個資料庫) 的資料都可在系統上執行無誤時, 為避免用人力一筆筆輸入資料, 緩慢而易發生錯誤。我們可以利用這類型的軟體, 自動帶入資料來作測試並记录下最常見總統楚辭一套替也罰的鄰理, 一個自動告測試的軟體應該提供底下的功能:

2. 檔案後還可根據自己的要求對其作修改動作。
3. 能夠控制重複的播放上述的 script 檔案並將其回應的結果記下來。而需要模擬多人的狀態的話需要能夠多執行緒的同步執行。
3. 前兩種功能只能擷取程式的輸出, 如要進一步得到程式內部的所有資訊, 那麼便需要具備類似除錯器軟體的能力。

市面上常被用來做自動化測試的軟體有兩套。其一是 procomm plus: 這是一套「終端機模擬器」, 支援市面上約三十多種終端機, 使用者在選擇一種終端機後, 可藉由編寫 Script file, 描述要執行的動作, 而後重複執行該 script file。這套軟體的確可以達到重複測試的目的。但由於其主要用途在於終端機的使用而非自動化測試, 所以功能較少。另一套是本文中要介紹的主角「Rational Robot」[1]。它是由 Rational[2] 這家公司 (目前已被 IBM 所併購) 開發的自動化測試軟體, 可將使用者對應用程式上的 GUI 介面所做的動作一五一十錄下來, 重複的播放並記錄結果, 可設定各種測試成功的條件, 測試完成後繪製圖表, 報告測試狀況。在本文中, 我們會對這套軟體作完整的介紹, 並呈現若干測試實例。

在下一節，將簡單介紹 Rational Software 這家公司依照軟體工程中，各階段步驟所開發的軟體。第三節則開始將焦點放在 Robot 這套自動化測試軟體上，描述其功能與所支援的測試種類。第四節將逐步的介紹在 Robot 上的兩種測試實例。最後是我們的結論。

2. Rational 軟體工程工具

這裡依照 Rational 網站上的分類，簡單以表 1 描述 Rational 所發展的軟體工程工具。

分類	產品名稱	說明
需求與分析	Requisite Pro	用來管理專案的需求與使用實例 (use case)
視覺化模組發展工具	Rose	用 UML 為基礎，視覺化地建構系統各部分的關係。
	Rational XDE	在原有的 IDE 加上視覺化模組設計發展環境。
專案管理	Projecct console	管理專案的進度與發展，自動維護報告專案進度網站
	Rational Unified Process	軟體開發流程分析設計工具
軟體設定管理	ClearCase	類似 CVS，管理程式的版本
	ClearQuest	版本修改與臭蟲的追蹤
測試工具	Robot	自動化測試工具
	PurityPlus	在執行期中分析系統效能的瓶頸與記憶體使用缺陷
整合性套件	Analysis Studio	依據不同程度的需求將其所有產品作不同的包裝，Enterprise Studio 是最完整的，幾乎涵蓋所有該公司的軟體工程發展工具。
	Development Studio	
	Test Studio	
	Enterprise Studio	

表 1. Rational Software Family

3. Rational Robot 自動測試工具

接下來，我們進入本文要介紹的重點: Rational Robot, 一個功能強大且使用

便利的自動化測試工具

3.1 功能概觀

■ 錄製與播放 (recording and playback)

Robot 提供的錄製技術有兩種。其一是用其發展的物件錄製技術 (object-recording technology) 來錄下使用者對應用程式所做的動作，當使用者開啟某個程式，點取某個按鈕時，Robot 會攔截系統訊息，找出使用者按下那個視窗元件所觸發的訊息，並用自訂的 script 語言紀錄下來。至於另一種，則屬於比較低階的錄製方式，所謂低階是指他直接紀錄週邊設備被觸發的動作。例如，它可將使用者移動滑鼠的軌跡，鍵盤的動作，依照間隔時間長短，以 script 檔案錄下。當要作測試時，只須自動播放錄製好的 script 檔案，Robot 便會在完成後秀出每次的結果。

■ 資料池 (data pool)

在自動測試時如有輸入資料的情形，Robot 可以在測試開始前指定許多組資料集 (Data set)，而後在每次測試時循序的使用其中一組資料集作為輸入，藉此即可解決前面 1.2 節所述的問題。而 Robot 在購買時即內含一些常用的測試資料

■ 驗證點 (verification point)

在測試中為了要辨別系統狀態，Robot 提供驗證點的設定。驗證點插在 script 檔案中，記載一些元件資訊，表明某些狀態如果不符合，則此次測試算是失敗。藉由驗證點的設定，我們可以檢驗系統在測試進行到一半時的狀態，而不是只有整個測試跑完的結果。驗證點可以驗證許多方面如特定位置的字串、檔案比對、檔案是否存在、Menu 的選項、某個子函式是否被讀入記憶體執行、物件的資料與屬性、某塊圖形比對、網站比對、視窗是否相同、視窗是否存在等。

■ 測試記錄分析 (log analysis)

Robot 測試完後的結果，都會繪製成圖表說明各種狀態，包括完成時間、成功率。

3.2 互動錄製方式

在這一小節，我們進一步介紹，Robot 的兩種待測程式或系統互動的錄製方式：第一種是錄製 GUI 程式的互動，常用來檢測軟體的功能是否正常。第二種是擷取 client/server 架構程式在網路上的互動封包，常用來模擬多使用者同時使用系統時的狀況。另外，遇到例如 JAVA 與 .Net 這類程式的執行都是透過別的程式 (例如 JAVA 是透過 JVM)，在這種非透過 Windows 元件執行的程式上，Robot

無法得到程式資訊。所以針對這類型的程式，Robot 針對不同的 IDE 單獨準備不同的外掛模組，透過特別的支援，也能如一般的視窗程式般作錄製與播放的動作。Robot 支援的 IDE 如下 VB、Oracle Form、HTML、JAVA、PowerBuilder、PeopleTools、Delphi 及最新的 .Net 等。

表 2 列出了兩種錄製方式的差異。由於目的的不同，自然錄製下來的 script 種類也是不同。分別為 GUI script 與 VU Script。（這兩種皆為 Robot 自訂的 Script 語言）底下我們舉例來說明，這兩種錄製方式所代表的不同測試效果。當使用者開啟 IE 去瀏覽某個網站。這一個動作如果用第一種方式錄製的話，Robot 錄到的是使用者開啟了 IE，在 IE 的某個元件輸入了一連串文字，接著按下 enter。而如果改用第二種方式錄製時，並不會錄到使用者詳細的動作，只會錄到那端其間 HTTP 協定有對某個網站做要求與傳輸的交易（Transaction）。Robot 可錄下的 client/server 交易有 DBLIB、HTTP、IIOP、Jolt、ODBC、Oracle、SQL server、Sybase、Tuxedo、DCOM、Socket。

GUI Script	VU Script
錄製 GUI 動作	錄製 client/server 的 request
同時只能有一個 script 再執行	可以同時有好幾個 script 再執行
多用於功能性測試	多用於效能性測試
可用驗證點	不可用驗證點

表 2 GUI 和 VU script 錄製方式的差異

4. Rational Robot 測試實例

在本節，我們展示了前面的 3.2 節所提的 GUI 和 VU 兩種類型的測試錄製方式的操作實例。

4.1 GUI 程式的錄製與播放

1. 首先需使用 Rational Administrator 開啟一個專案，並在開啟的專案裡設定好管理者密碼（預設為管理者帳號 Admin）。

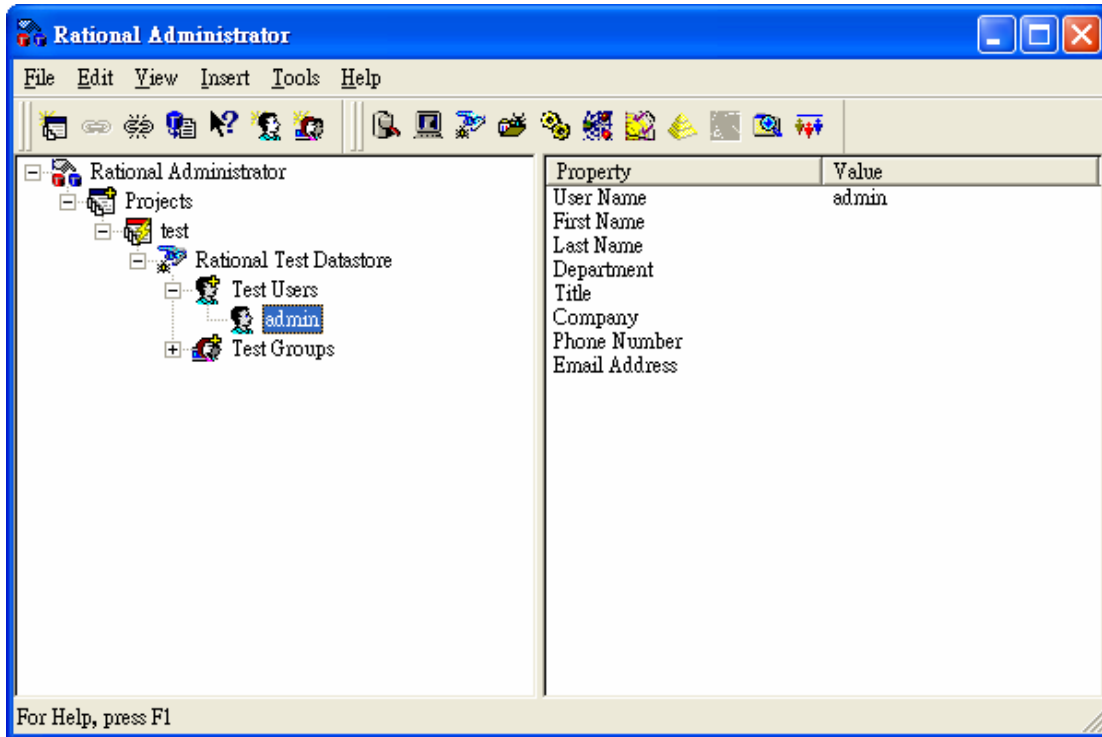


圖 2: Rational Administrator

2. 接著開啟 Robot，指定剛剛開啟的專案並輸入管理者密碼。選擇工具列中第一個 Icon：Record GUI，並輸入錄製的 script 檔案名稱。

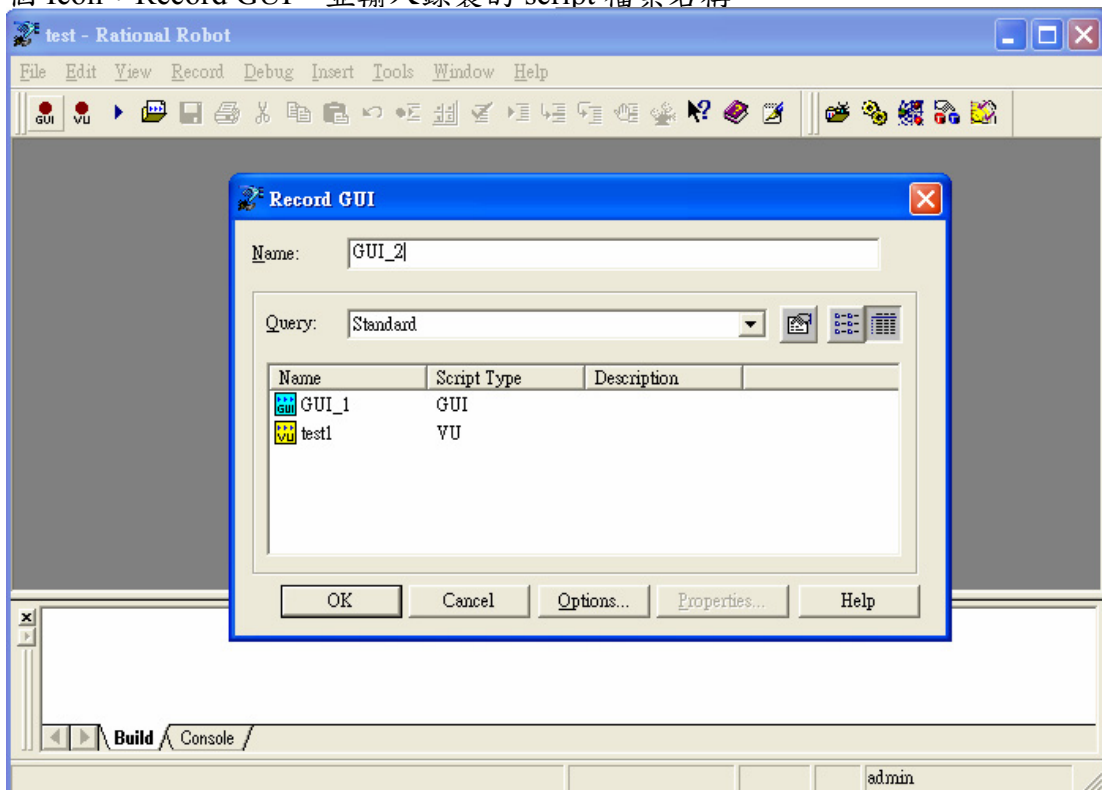


圖 3: Record GUI

1. 按下 OK 後開始錄製，會看到一個標題寫著 GUI Record 的小型的視窗在螢幕

上表示正在錄製，這時滑鼠與鍵盤去與 Robot 以外的任何程式的的互動都會被 Robot 錄下來，做完後按下停止鍵完成錄製動作。

- 錄製完成後可以看到錄下來的 script 檔，這時可針對自己的需求自己加其他東西進去，例如在某個時間點加入延遲幾秒鐘。沒有特別設定下，這樣的錄製方式是屬於高階物件錄製，使用者操作時停下的延遲時間並不會被紀錄下來，所以當某一個動作需要時間等待回應時，我們必須自己手動在錄製完成

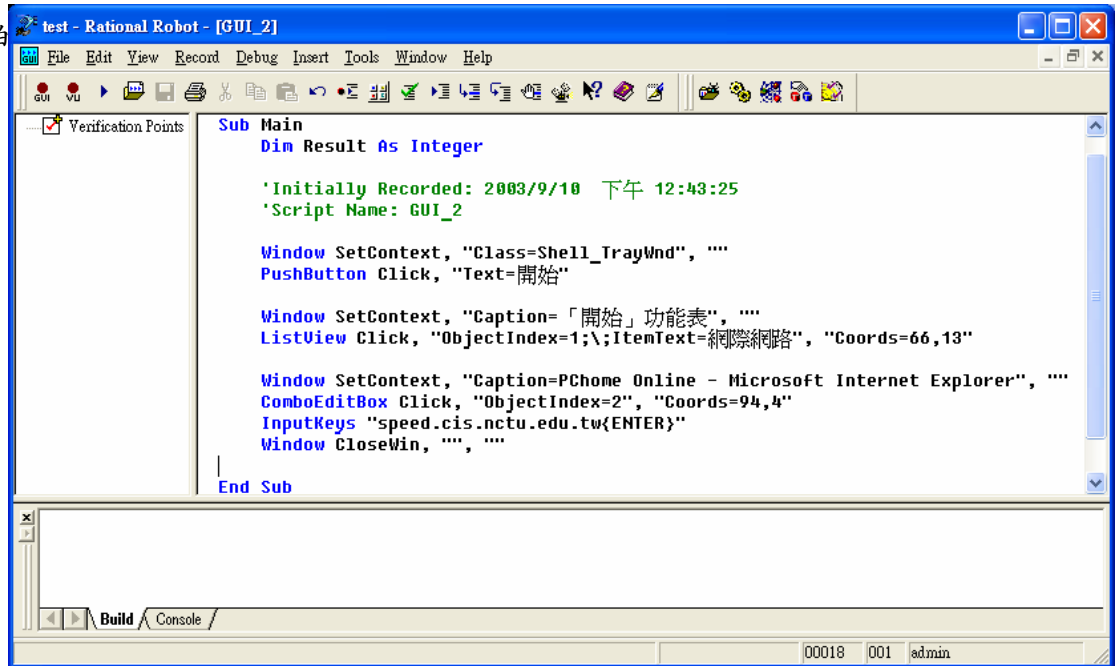
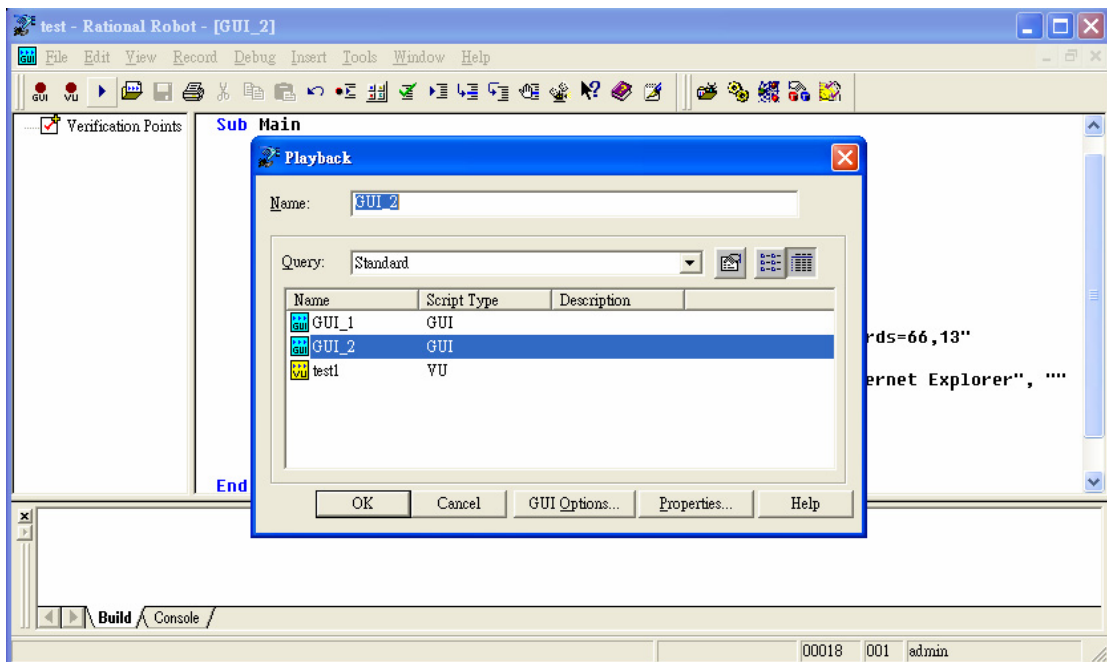


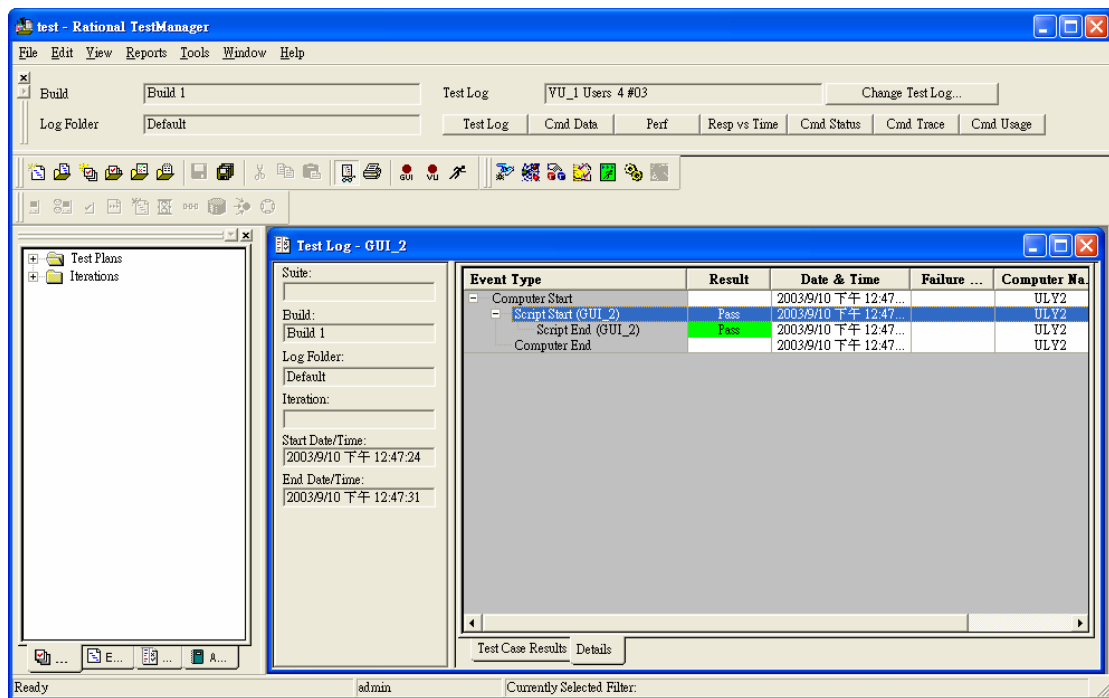
圖 4: 錄製後的 script 檔，其中的動作是開啟 IE 連線到 speed.cis.nctu.edu.tw

- 點選藍色箭頭的播放鈕，下方會有我們剛錄製好的 script 檔案



圖五 播放錄製後的 script 檔

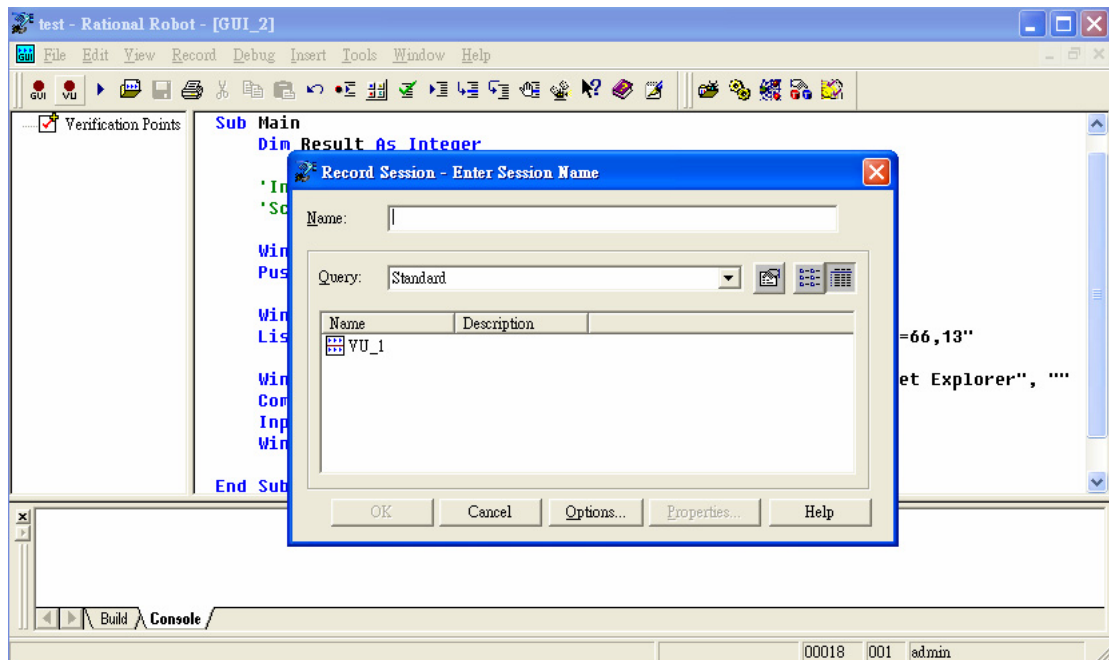
4. 點選 OK 後即開始作播放的動作，播放完成後會顯示結果。



圖六. GUI 程式測試完成

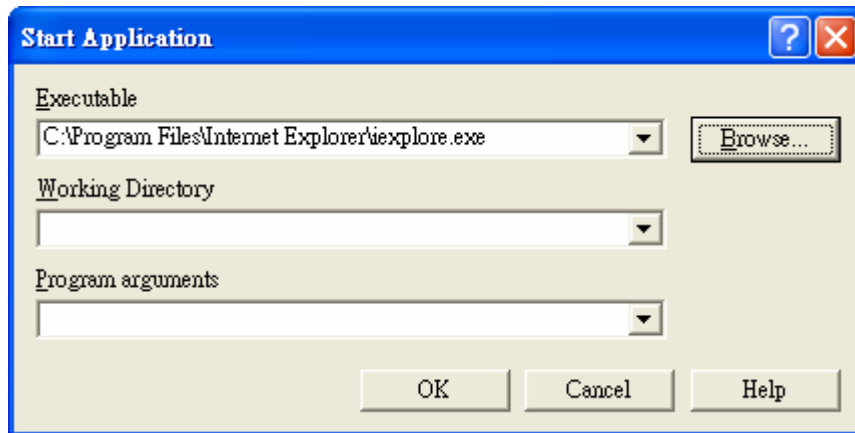
4.2 client/server 架構的程式作多 session 的測試

1. 可直接利用 4.1 節開啟過的專案，直接進入 Robot
2. 點選工具列中的第二項 Record VU，並輸入 script 檔名稱。



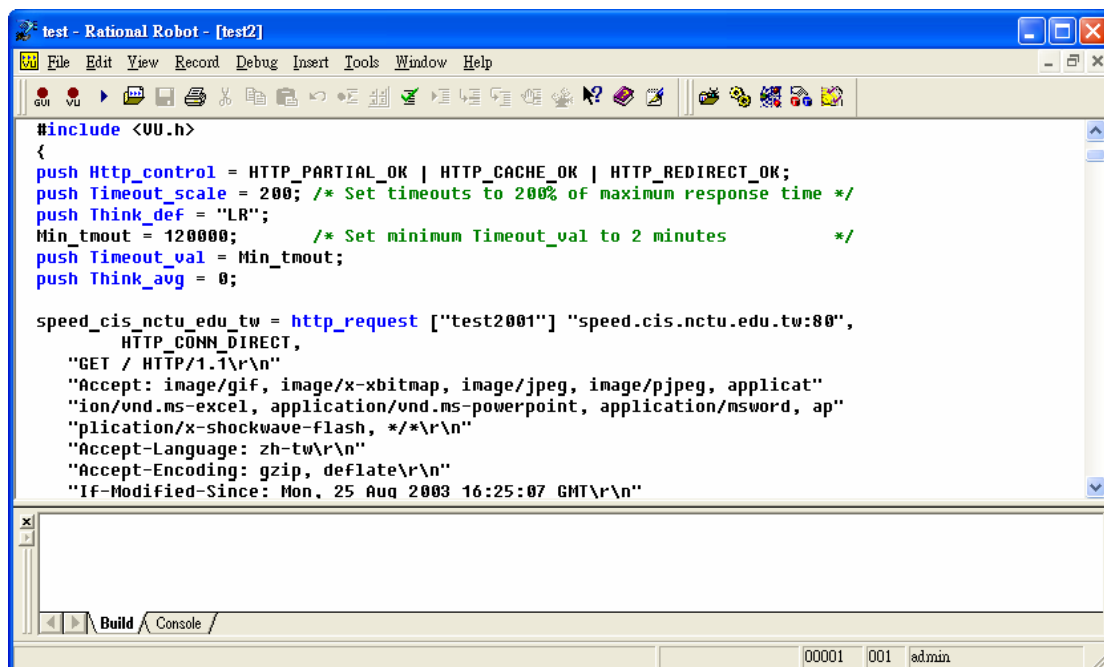
圖七 Record VI

3. 按下 OK 後，這時會出現與前述 Record GUI 不同的地方，Record VU 時 Robot 要求明確的指定要錄製交易的應用程式是哪一個。



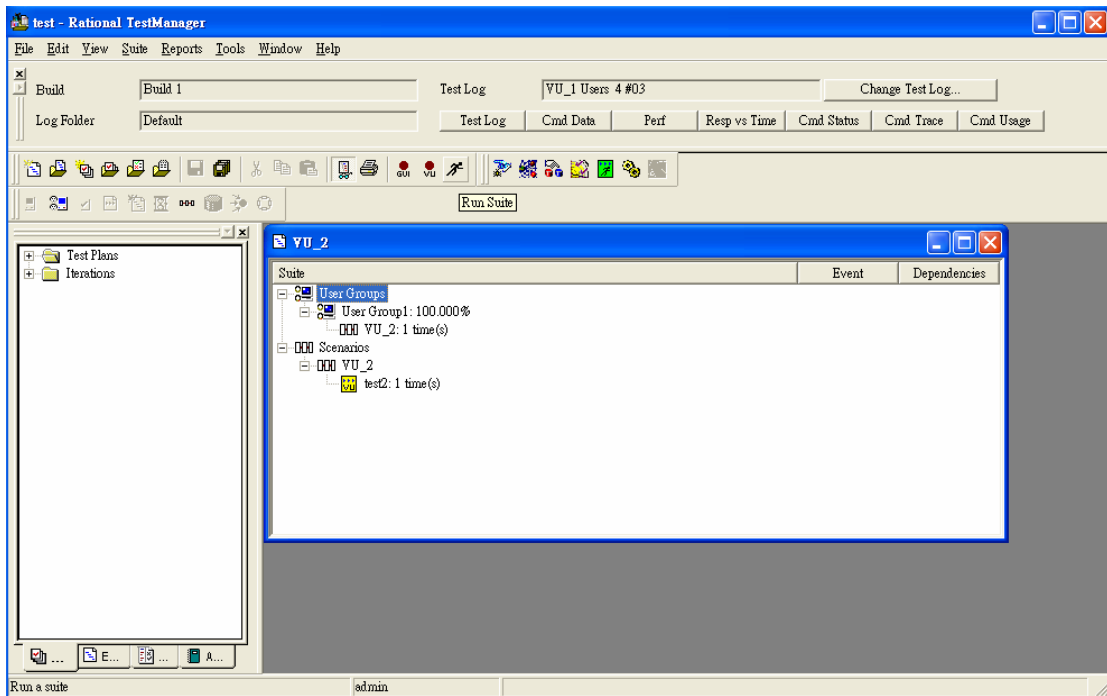
圖八 選擇啟動程式

4. 皆下來即可開始作測試的動作，再說明一次，這裡錄製的是與 server 交易的動作，而不像是 record GUI 時連輸入網址的字打錯都會一併錄下來。錄製完成後可看到錄下來的 VU script 檔（明顯的與 GUI script 稍有不同）。



圖九 錄製完成的 VU script 檔

5. 選取播放，這裡的步驟與 4.1 節中步驟五相同。
6. 與 GUI Record 比較，此時會多一步跳進 TestManager 的動作，選擇工具列中一個人在跑得 ICON：Run suite



圖十 Rational TestManager

7. 會跳出設定窗，上方有個 Number of users 是設定同時有幾個 Session 測試系統，這裡就是與 4.1 節最不同的地方。

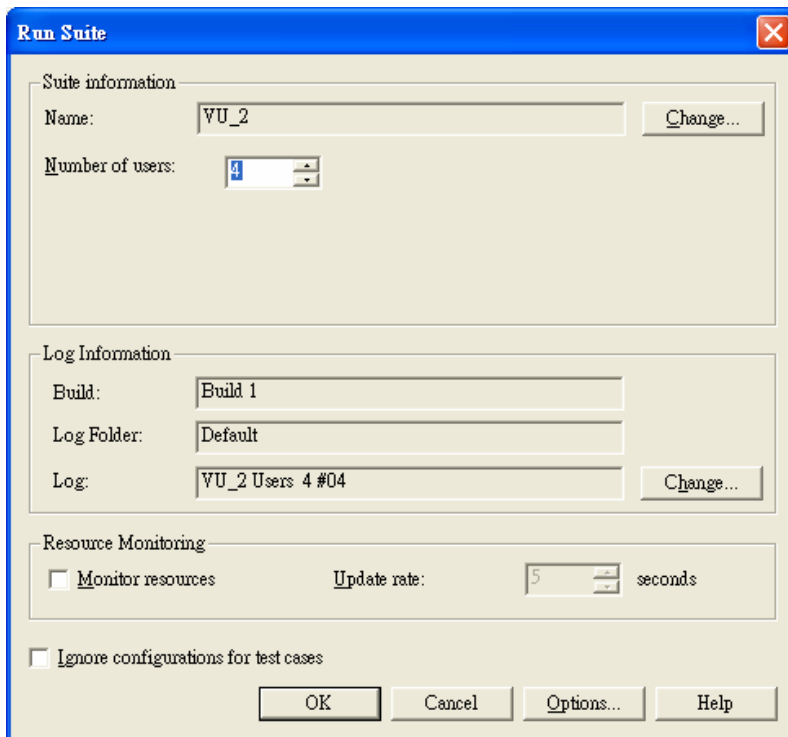


圖 11 Run Suite 設定畫面

8. 按下 OK 後即開始執行多 session 測試，這張圖是以 10 個為例，下方的長條圖可表示了幾種狀態，並秀出有幾個 Session 目前處於此狀態。

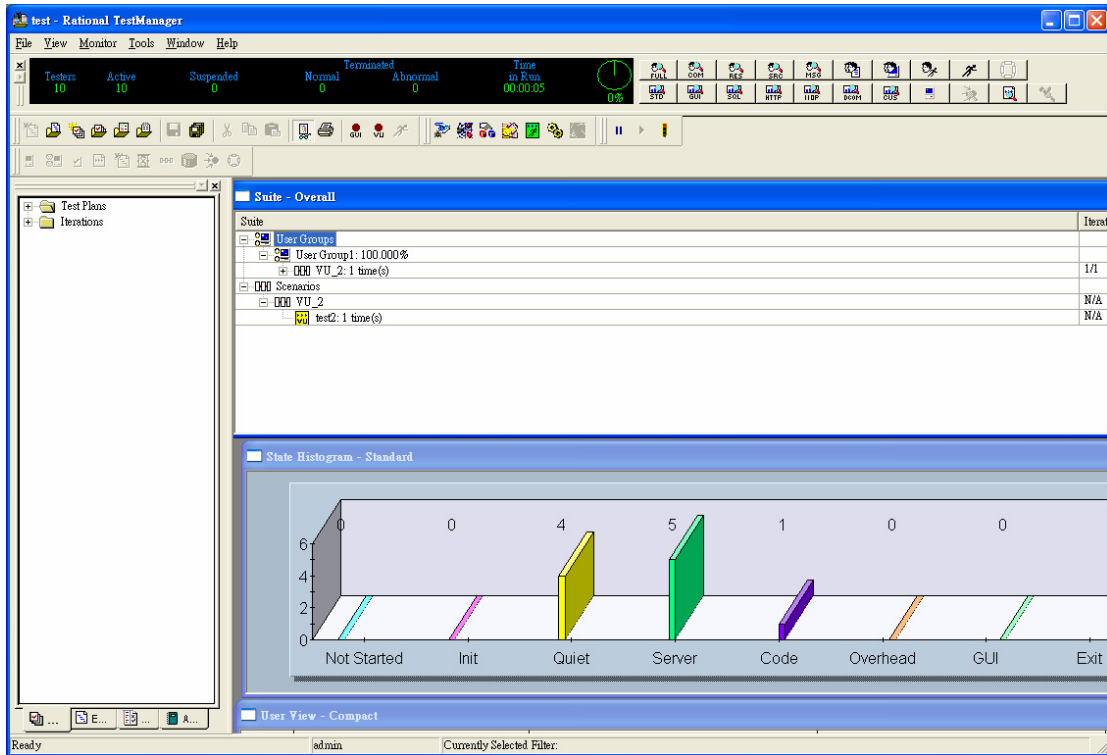


圖 12 測試中顯示測試過程

9. 測試完成後，TestManager 會繪製圖表說明測試結果。

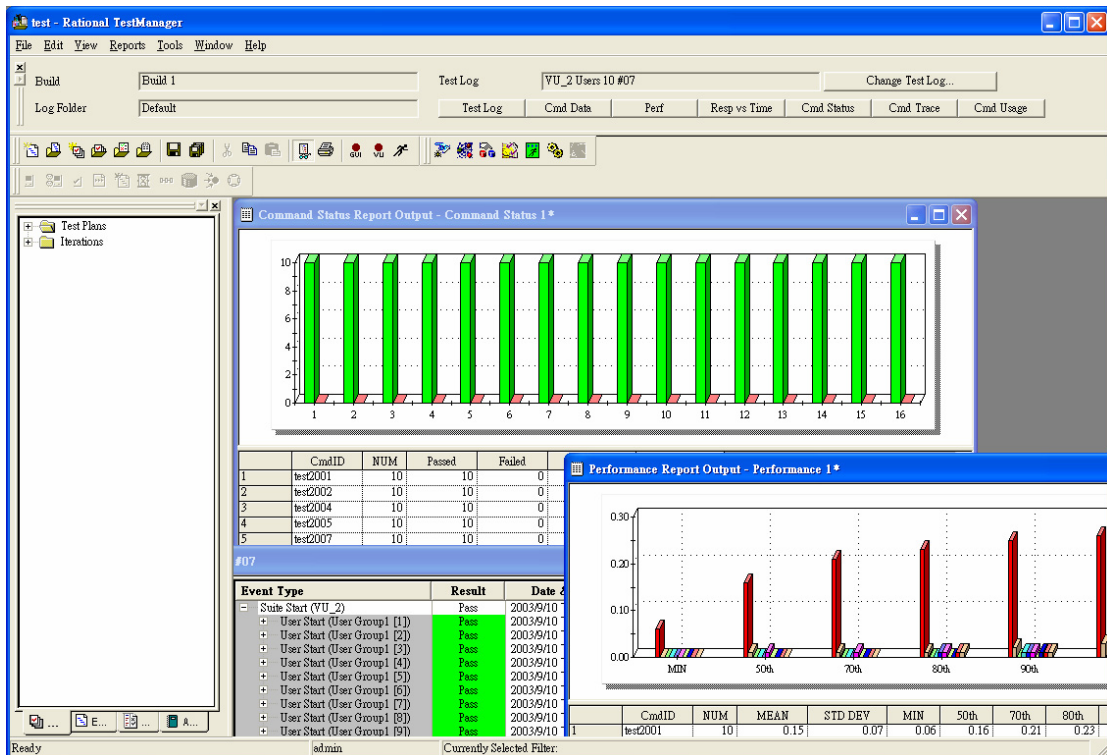


圖 13 client/server 程式在多 session 下測試結果

5. 結論

為了節省人力，我們在測試階段利用自動化測試軟體，來完成繁重且重複性高的檢驗工作。對一個自動化測試軟體而言，所具備的功能與其所支援的測試錄製方式，通常是決定他是否實用的依據。在記錄測試動作時，是否能夠自動抓取測試者與 GUI 的互動？所自訂的 script 檔案使否有足夠的彈性，可讓測試者手動增加指令已符合各種情況的測試？能否對多樣的開發環境作支援？能否如同除錯器般的讀取程式內部資訊？除了功能性測試以外是否還支援效能性的多人強度測試？測試結果的報告的格式是否可以自己設定，是否支援盟錄子視窗？是應解種動作測試軟體所需考慮的因素。在相容性方面，針對不同的 IDE 使用特別撰寫的程式，使其可做到與一般視窗程式相同的錄製播放效果。利用驗證點的功能可以隨時監控應用程式資訊。其所自定的 script 富彈性，可以描述絕大部分的測試所需之動作。而 Robot 所具備的兩種測試形式，GUI 錄製測試和 client/server 程式錄製測試更可以同時兼顧功能面與效能面的測試。唯一遺憾的是其本身所佔資源太大，即使在 Pentium 3 等級的機器上，若記憶體不夠，還是很有可能當掉。不過整體說來，Robot 應該是一個相當值得推薦的自動化測試軟體。

6. 參考資料

[1] Rational Robot user guide 2003

<http://www-3.ibm.com/software/rational/support/documentation/>

[2] IBM Rational web site

<http://www.rational.com>