

CVS基本使用

CVS介紹與使用觀念

CVS (Concurrent Versions System) 是一個版本控制系統。CVS 常見於程式開發或者文件撰寫時，所使用的一種軟體版本控制的工具。CVS 可透過網路運作，對於多人協同開發程式、多人文件撰寫有極為強大的效果。

CVS的使用可以讓開發人員可以隨時隨地的針對同一檔案進行編修，而不用擔心自己修改好的部分會不會被其他開發者所覆蓋或者移除，因為CVS版本控制系統會幫您妥善處理好檔案的合併與復原動作。CVS可以紀錄下某人於某時所新增加的程式或文件內容。並且可以很輕易的比對出兩個版本的差異，以及該內容由誰所新增，借此可強化版本的控管與程式註解的能力。例如：目前當紅的WIKI (多人網路筆記) 有些也採用CVS作為後端文件版本控制。可以多人同時進行一份檔案的編修，多人開發的網站或網頁，也可以透過 CVS 來管理，方便又有效率。

CVS伺服器端(Server)可以在所有的UNIX和Linux作業系統上執行。CVS有多種用戶端(Client)工具可供使用，CVS用戶端程式則可以在跨平台環境中使用。以下將說明如何在Linux平台安裝CVS，以及介紹CVS的基本使用方法與Windows用戶端的使用方法。

CVS安裝 (Linux平台)

CVS是一種主從式軟體，可以在Unix和Linux平台執行。只要在Unix/Linux伺服器上安裝CVS時，就等於自動安裝了CVS的伺服器與用戶端程式，任何人只要登入到該主機就可以使用command line的方式來執行CVS用戶端程式。以下將簡述在Linux平台上安裝CVS。

安裝步驟：

0.首先以root帳號登入，以下的安裝步驟均以root來執行。

1.至 <http://www.cvshome.org> 取得最新版本的CVS原始碼包裝。或者直接取得 cvs-1.11.18.tar.gz(<https://ccvs.cvshome.org/files/documents/19/537/cvs-1.11.18.tar.gz>)的檔案進行編譯安裝。

執行以下指令

```
# wget https://ccvs.cvshome.org/files/documents/19/537/cvs-1.11.18.tar.gz
```

```
[root@lx4 root]# whoami
root
[root@lx4 root]# wget https://ccvs.cvshome.org/files/documents/19/537/cvs-1.11.18.tar.gz
```

2.下載完成cvs-1.11.18.tar.gz檔案時，請將壓縮檔解開並進行編譯安裝。

執行以下指令

```
# tar xzf cvs-1.11.18.tar.gz
```

```
# cd cvs-1.11.18
```

```
# ./configure ; make ; make install
```

```
[root@lx4 root]# ls -l
總計 2904
-rw-r--r--  1 root    root      2969415 11月 12 06:02 cvs-1.11.18.tar.gz
[root@lx4 root]# tar xzf cvs-1.11.18.tar.gz
[root@lx4 root]# ls -l
總計 2908
drwxrwxrwx  14 hyh      500          4096 11月 12 02:29 cvs-1.11.18
-rw-r--r--  1 root    root      2969415 11月 12 06:02 cvs-1.11.18.tar.gz
[root@lx4 root]# cd cvs-1.11.18
[root@lx4 cvs-1.11.18]# ./configure ; make ; make install
```

3.必須建立一個CVS用戶群組，之後只要將主機上的使用者加入至該CVS用戶群組，才可來使用CVS的儲存庫。

執行以下指令

```
# useradd cvs
```

```
# vi /etc/group
```

```
[root@lx4 cvs-1.11.18]# useradd cvs
[root@lx4 cvs-1.11.18]# vi /etc/group

cvs:x:503:cyj,cyj2
cyj:x:504:
cyj2:x:505:
:x
```

(這裡示範將cyj、cyj2這兩位使用者加入到CVS用戶群組中)

4.CVS需要一個資料庫，稱之為CVS儲存庫(repository)，這裡面將存放往後匯入於CVS中的所有專案的資料。依照FHS(<http://www.pathname.com/fhs>)的規定，儲存庫較佳的存放位置應該設置在 /var/lib/cvsroot 。請依照下列指令執行，建立第一個CVS儲存庫。

執行以下指令

```
# mkdir /var/lib/cvsroot
```

```
# chown cvs.cvs /var/lib/cvsroot
```

```
# chmod 2775 /var/lib/cvsroot
```

```
# cvs -d /var/lib/cvsroot/ init
```

```
# chown -R cvs.cvs /var/lib/cvsroot
```

```
[root@lx4 cvs-1.11.18]# mkdir /var/lib/cvsroot
[root@lx4 cvs-1.11.18]# ls -ld /var/lib/cvsroot/
drwxr-xr-x  2 root  root    4096 12月 30 00:34 /var/lib/cvsroot/
[root@lx4 cvs-1.11.18]# chown cvs.cvs /var/lib/cvsroot/
[root@lx4 cvs-1.11.18]# chmod 2775 /var/lib/cvsroot/
[root@lx4 cvs-1.11.18]# ls -ld /var/lib/cvsroot/
drwxrwsr-x  2 cvs   cvs    4096 12月 30 00:34 /var/lib/cvsroot/
[root@lx4 cvs-1.11.18]# cvs -d /var/lib/cvsroot/ init
[root@lx4 cvs-1.11.18]# ls -la /var/lib/cvsroot/
總計 12
drwxrwsr-x  3 cvs   cvs    4096 12月 30 00:36 .
drwxr-xr-x 20 root  root    4096 12月 30 00:34 ..
drwxrwsr-x  3 root  cvs    4096 12月 30 00:36 CVSROOT
[root@lx4 cvs-1.11.18]# chown -R cvs.cvs /var/lib/cvsroot/
[root@lx4 cvs-1.11.18]#
```

5.到此為止，CVS儲存庫已經建立完畢，未來所有匯入的專案就會存放在儲存庫目錄的子目錄裡，而在此示範的儲存庫目錄就是 /var/lib/cvsroot。

CVS快速入門與使用方法

由於CVS的指令使用實為繁雜且難以理解，若欲瞭解更深入的CVS使用方法，可參考CVS速查表(<http://linux.tnc.edu.tw/techdoc/CVSdoc/cvs-quick-ref.html>)，或是完整的CVS技術手冊(<http://www.oreilly.com.tw/chinese/network/cvs.html>)。

在這個部分裡，僅會說明CVS最常見的基本使用方法，並且以Unix/Linux命令列上所執行的CVS用戶端程式的使用作為示範。以下將簡單敘述如何把專案匯入CVS儲存庫，並且進行多人協同開發。

1. 把專案匯入儲存庫
2. 從儲存庫取出專案副本
3. 將專案副本變動後的資料交付至儲存庫
4. 更新本地端的專案副本
5. 檔案發生衝突該如何處理
6. 替儲存庫中的專案新增檔案
7. 替儲存庫中的專案移除檔案

1. 把專案匯入儲存庫

在建立完成儲存庫之後，就可以將自己的專案匯入CVS儲存庫中，未來此專案的所有變動皆可以透過CVS指令來達成控制與管理的功用。

現在有一個專案名稱叫做websir，他們的專案組長為cyj，以下示範cyj組長如何將他們的websir專案匯入到CVS儲存庫。

```
$ ls -l
```

```
$ cd websir (進入專案的資料夾中)
```

```
[cyj@lx4 cyj]$ ls -l
總計 4
drwxrwxr-x  2 cyj  cyj  4096 12月 30 01:49 websir
[cyj@lx4 cyj]$ cd websir/
[cyj@lx4 websir]$
```

執行cvs指令

```
$ cvs -d /var/lib/cvsroot import websir websir_project ver_0-1 (將專案匯入儲存庫)
```

```
[cyj@lx4 websir]$ pwd
/home/cyj/websir
[cyj@lx4 websir]$ cvs -d /var/lib/cvsroot import websir websir_project ver_0-1
```

之後會開啟一個編輯視窗，依照個人需求與否，可在此寫入一些專案簡介等等。

```
CVS: -----
CVS: Enter Log. Lines beginning with `CVS:' are removed automatically
CVS: -----
CVS:
: x
```

儲存離開後將會看到下列訊息，表示專案websir已經匯入成功了！

```
"/tmp/cvs8dUYBT" 53L, 278C 已寫入
N websir/html1
No conflicts created by this import
[cyj@lx4 websir]$
```

專案的匯入到此已完成。其主要所用的CVS命令簡單解釋如下：

```
$ cvs -d <儲存庫路徑> import <專案> <專案的名稱> <專案的版本>
```

```
$ cvs -d /var/lib/cvsroot import websir websir_project ver_0-1
```

如此即可將專案匯入至CVS儲存庫中。

2. 從儲存庫取出專案副本

專案匯入之後，CVS把專案跟檔案集中存放在儲存庫裡，而之後我們所編修的個人副本則存放在自己的本地端目錄下。用來建立個人副本的CVS命令如下：

```
$ cvs -d <儲存庫路徑> checkout <專案的名稱>
```

以下示範cyj2使用者要將其websir專案副本取出到自己的本地端目錄中。

```
$ mkdir ~/cvs
```

```
$ cd ~/cvs
```

執行cvs指令

```
$ cvs -d /var/lib/cvsroot/ checkout websir (從儲存庫中取出專案副本)
```

```
[cyj2@lx4 cyj2]$ mkdir ~/cvs
[cyj2@lx4 cyj2]$ cd ~/cvs
[cyj2@lx4 cvs]$ cvs -d /var/lib/cvsroot/ checkout websir
cvs checkout: Updating websir
U websir/html1
[cyj2@lx4 cvs]$ ls -l
總計 4
drwxrwxr-x    3 cyj2      cyj2          4096 12月 30 02:18 websir
[cyj2@lx4 cvs]$
```

如此即可從從儲存庫中取出websir專案副本。所有的專案組員都必須在自己的家目錄中執行上述的CVS指令來取出專案副本。未來此專案所有的修改動作均在自己家目錄的專案副本中執行，如此可達成多人協同開發的作用。

3. 將專案副本變動後的資料交付至儲存庫

從儲存庫調出專案副本到自己的家目錄後，就可以對該專案的檔案進行編輯修改。檔案的變動之處僅會在自己的家目錄中，不會跟儲存庫的檔案同步化。因此必須執行CVS命令：**cvs commit**，把家目錄的專案變動之處交付到儲存庫中。這裡必須要注意的重點是，執行cvs commit命令時，建議必須在本地端(家目錄)專案資料夾的根目錄中執行才行(沙箱的根目錄)。

以下示範cyj2使用者將其websir專案的變動交付至儲存庫。

\$ cd ~/cvs/websir (進入到本地端websir專案的根目錄中)

執行cvs指令

\$ cvs commit (將專案副本變動後的資料交付至儲存庫)

```
[cyj2@1x4 cvs]$ cd ~
[cyj2@1x4 cyj2]$ ls -l
總計 4
drwxrwxr-x    3 cyj2    cyj2          4096 12月 30 02:18 cvs
[cyj2@1x4 cyj2]$ cd ~/cvs/websir/
[cyj2@1x4 websir]$ cvs commit
cvs commit: Examining .
[cyj2@1x4 websir]$
```

如下圖，將資料交付至儲存庫時，可以在這裡寫入一些交付的訊息或註解資料。

```
CVS: -----
CVS: Enter Log.  Lines beginning with `CVS:' are removed automatically
CVS:
CVS: Committing in .
CVS:
CVS: Modified Files:
CVS:    html1
CVS: -----
:x
```

編輯畫面儲存之後離開，即完成交付的動作。

```
"/tmp/cvstIwIm2" 43L, 325C 已寫入
Checking in html1;
/var/lib/cvsroot/websir/html1,v <-- html1
new revision: 1.2; previous revision: 1.1
done
[cyj2@1x4 websir]$
```

4. 更新本地端的專案副本

專案若是由多人協同開發時，一個相同的檔案隨時都會有許多人正在撰寫與編修，因此專案的組員至少每天在開始工作之前，或者是工作至一段時間後，必須執行CVS命令：**cvs update**，以確保個人專案副本(沙箱)能保持最新的狀態。這裡必須要注意的重點是，執行cvs update命令時，建議必須在本地端(家目錄)專案資料夾的根目錄中執行才行(沙箱的根目錄)。

以下示範cyj2使用者將其websir專案的個人副本做更新的動作。

\$ cd ~/cvs/websir (進入到家目錄中的websir專案根目錄中)

執行cvs指令

\$ cvs update -d (本地端專案副本更新至最新的狀態)

```
[cyj2@lx4 cyj2]$ cd ~/cvs/websir/  
[cyj2@lx4 websir]$ cvs update -d  
cvs update: Updating .  
U html1  
[cyj2@lx4 websir]$
```

當update命令執行時，會產生已修改之檔案的清單。在每個檔名的左邊會有一個大寫的字母，這些字母是用來回報清單中每個檔案的狀態，分別的意義如下說明。

U：更新成功，較新的專案版本已經取代個人副本中舊的版本。

A：代表新增，但尚未加入至儲存庫，需要執行cvs commit。

R：代表移除，但尚未從儲存庫中移除，需要執行 cvs commit。

M：檔案變更之處已安全的合併，但尚未更新至儲存庫，必須要執行cvs commit。

C：儲存庫的副本跟自己個人專案副本的檔案有衝突，必須人工協調處理。

?：此檔案位於個人副本工作目錄中，但儲存庫找不到，CVS不知如何處理。

5. 檔案發生衝突該如何處理

專案若是由多人協同開發時，一個相同的檔案隨時都會有許多人正在撰寫與編修。通常如果一個檔案的變更之處在不同列上時，則CVS會自動合併檔案，反之，如果檔案變更之處在同一列時，則CVS會回報衝突，並建立一個檔案，內容含有該列的不同之處，並以特殊符號加以標示。在多人協同開發時，常常會發生檔案衝突的狀況。以下舉例示範：

cyj2使用者在html1這個檔案的第一行填寫20個a。

```
aaaaaaaaaaaaaaaaaaaaaa
~
:x
```

寫完之後cyj2使用者執行了cvs commit的動作將檔案變更交付至儲存庫。

```
[cyj2@lx4 websir]$ cvs commit
cvs commit: Examining .
[cyj2@lx4 websir]$
```

剛好在這個同時，cyj使用者也在html1這個檔案的第一行填寫了20個b。

```
bbbbbbbbbbbbbbbbbbbbbb
~
:x
```

寫完之後cyj使用者也執行cvs commit命令將檔案變更交付，這時候由於兩位使用者分別在同一的檔案的同一行寫了不一樣的資料，CVS無法進行合併，因此就會發生如下的衝突事件。

```
[cyj@lx4 websir]$ cvs commit
cvs commit: Examining .
cvs commit: Up-to-date check failed for `html1'
cvs [commit aborted]: correct above errors first!
[cyj@lx4 websir]$ cvs update
cvs update: Updating .
RCS file: /var/lib/cvsroot/websir/html1,v
retrieving revision 1.9
retrieving revision 1.10
Merging differences between 1.9 and 1.10 into html1
rcsmerge: warning: conflicts during merge
cvs update: conflicts found in html1
C html1
[cyj@lx4 websir]$
```

cyj使用者可以察看html1檔案，CVS會自動把衝突的地方以特殊符號加以標示。

```
[cyj@lx4 websir]$ cat html1
<<<<<<< html1
bbbbbbbbbbbbbbbbbbbbbb
=====
aaaaaaaaaaaaaaaaaaaaa
>>>>>>> 1.10
[cyj@lx4 websir]$
```

此時CVS無法解決衝突事件，必須由cyj使用者與cyj2使用者互相協調，協調究竟是要填寫a還是b。協調成功之後，cyj使用者必須重新編輯html1檔案，並且找到衝突標示符號的區域，手動編修檔案，然後再把最後修改完成的檔案交付給儲存庫，如此可解決檔案衝突的問題。

```
[cyj@lx4 websir]$ vi html1
[cyj@lx4 websir]$ cat html1
bbbbbbbbbbbbbbbbbbbbbb
[cyj@lx4 websir]$ cvs commit
cvs commit: Examining .
[cyj@lx4 websir]$
```

6. 替儲存庫中的專案新增檔案

要為儲存庫裡的專案新增檔案，首先必須在本地端個人專案副本中建立該檔案。然後在本地端個人專案副本的根目錄中執行以下CVS命令：

```
$ cvs add <檔案名或目錄名>
```

cvs add執行之後，會將相對應的檔案標示為新增，在cvs commit執行過後，才會把檔案實際的加入至儲存庫中。

以下示範cyj使用者將其websir專案新增一個檔案html2，並且交付至儲存庫中。

```
$ cd ~/cvs/websir (進入到家目錄中的websir專案根目錄中)
```

```
$ touch html2
```

執行cvs指令

```
$ cvs add html2 (將websir專案新增一個檔案html2)
```

```
$ cvs commit (將新增檔案結果交付至儲存庫中)
```

```
[cyj@lx4 cyj]$ cd ~/cvs/websir/
[cyj@lx4 websir]$ ls -l
總計 8
drwxrwxr-x   2 cyj      cyj      4096 12月 30 04:22 CVS
-rw-rw-r--   1 cyj      cyj      21 12月 30 04:02 html1
[cyj@lx4 websir]$ touch html2
[cyj@lx4 websir]$ ls -l
總計 8
drwxrwxr-x   2 cyj      cyj      4096 12月 30 04:22 CVS
-rw-rw-r--   1 cyj      cyj      21 12月 30 04:02 html1
-rw-rw-r--   1 cyj      cyj       0 12月 30 04:23 html2
[cyj@lx4 websir]$ cvs add html2
cvs add: Re-adding file `html2' (in place of dead revision 1.2).
cvs add: use 'cvs commit' to add this file permanently
[cyj@lx4 websir]$ cvs commit
[cyj@lx4 websir]$
```

7. 替儲存庫中的專案移除檔案

要從儲存庫中移除檔案，首先必須在本地端個人專案目錄中移除該檔案。然後在本地端個人專案副本的根目錄中執行以下CVS命令：

`$ cvs remove <檔案名或目錄名>`

同樣的在cvs remove 檔案之後，必須執行cvs commit，如此刪除的動作才會生效，儲存庫中相對應的檔案將會被移除。

以下示範cyj使用者將其websir專案的檔案html2移除，並且交付至儲存庫中。

`$ cd ~/cvs/websir` （進入到家目錄中的websir專案根目錄中）

`$ rm html2` （移除html2檔案）

執行cvs指令

`$ cvs remove html2` （將websir專案檔案html2移除）

`$ cvs commit` （將移除檔案結果交付至儲存庫中）

```
[cyj@lx4 cyj]$ cd ~/cvs/websir/
[cyj@lx4 websir]$ ls -l
總計 8
drwxrwxr-x   2 cyj      cyj          4096 12月 30 04:34 CVS
-rw-rw-r--   1 cyj      cyj           21 12月 30 04:02 html1
-rw-rw-r--   1 cyj      cyj           0 12月 30 04:23 html2
[cyj@lx4 websir]$ rm html2
[cyj@lx4 websir]$ cvs remove html2
cvs remove: scheduling `html2' for removal
cvs remove: use 'cvs commit' to remove this file permanently
[cyj@lx4 websir]$ cvs commit
[cyj@lx4 websir]$
```

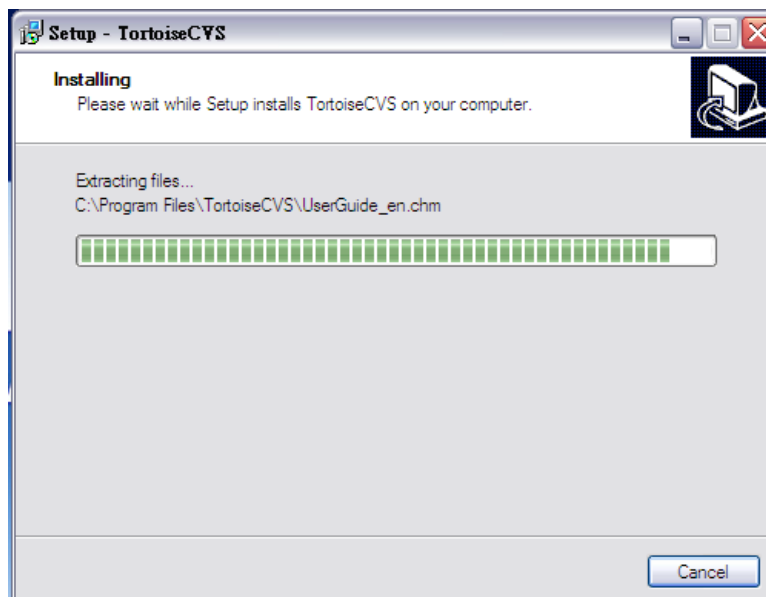
Windows用戶端使用方法

除了前面所示範的command line使用方式以外，我們也同樣可以在Windows上安裝CVS用戶端軟體來使用CVS。目前比較多人使用的CVS Client Tool有WinCVS以及TortoiseCVS。在這個部分裡，僅會說明CVS最常見的基本使用方法，並且以For Windows TortoiseCVS用戶端程式的使用作為示範。

首先必須下載TortoiseCVS軟體

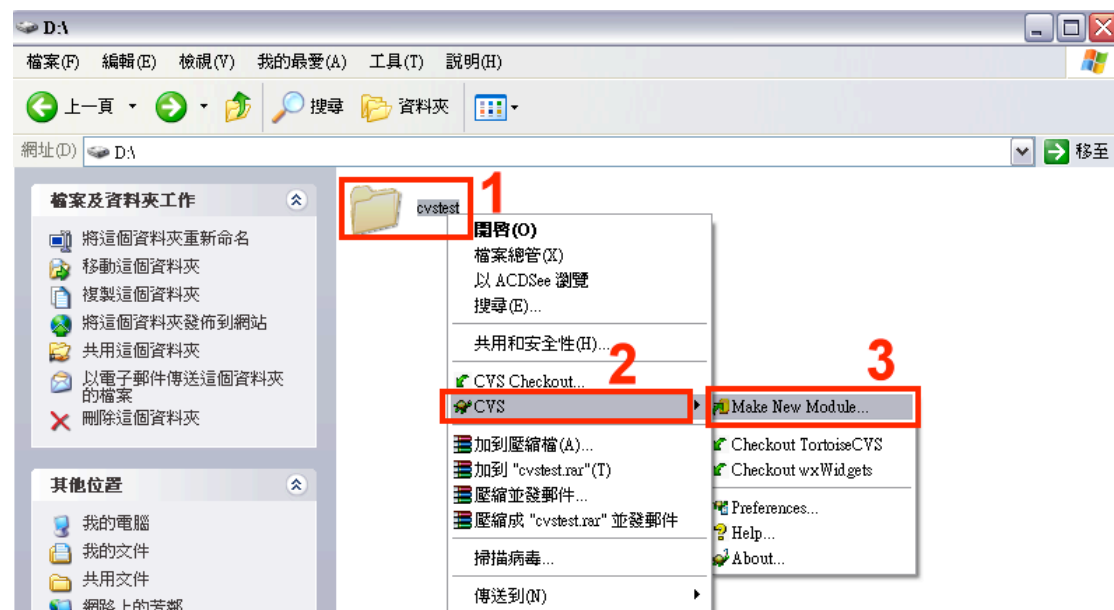
<http://www.tortoisecvs.org/download.shtml>

下載完畢後進行安裝

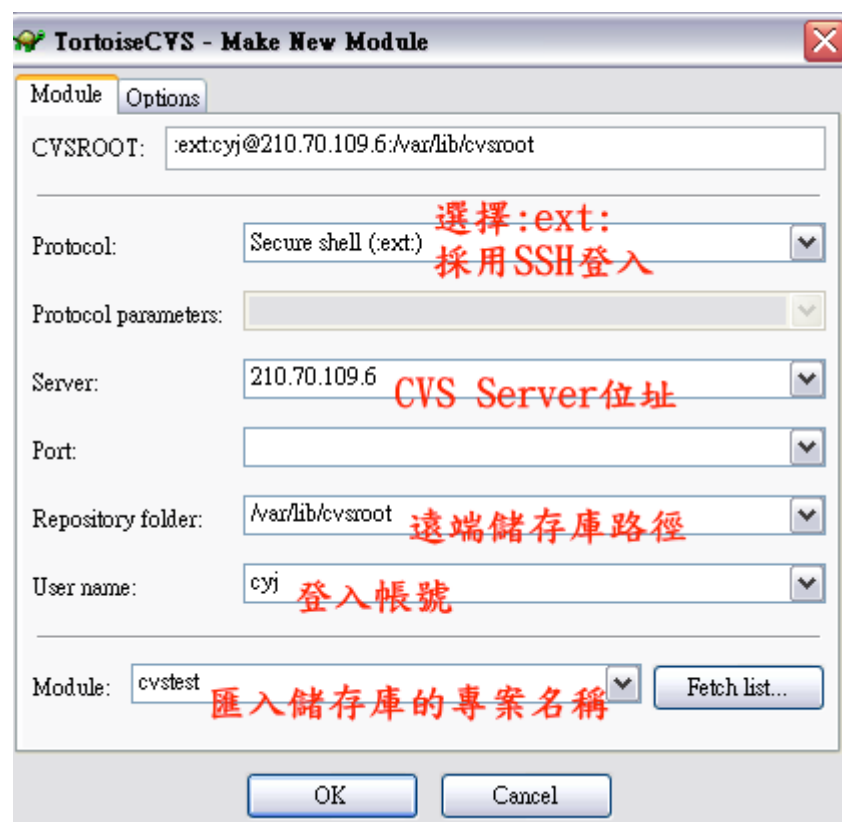


1. 把專案匯入儲存庫

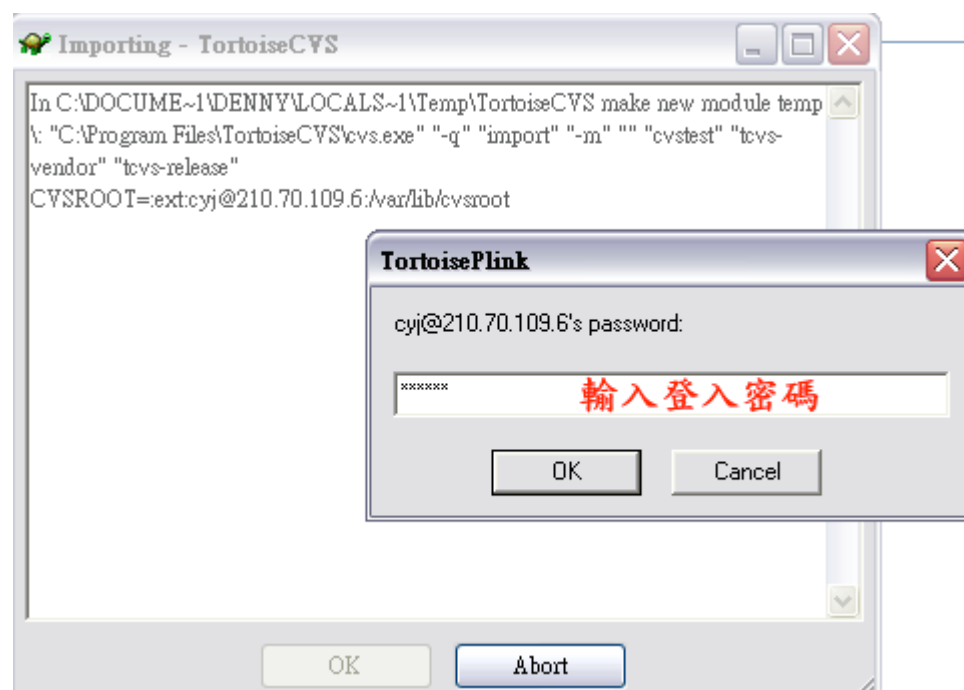
以下示範將D:\cvstest 專案匯入至儲存庫。



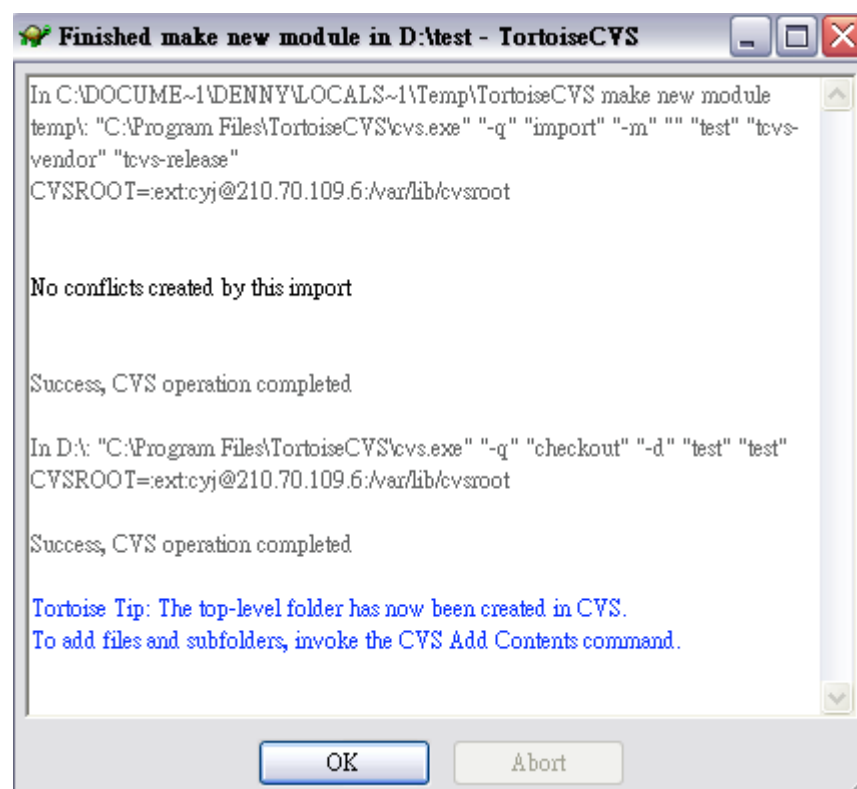
按照圖示1至3的步驟執行



依照圖示紅字部分依序輸入遠端CVS Server的資訊。



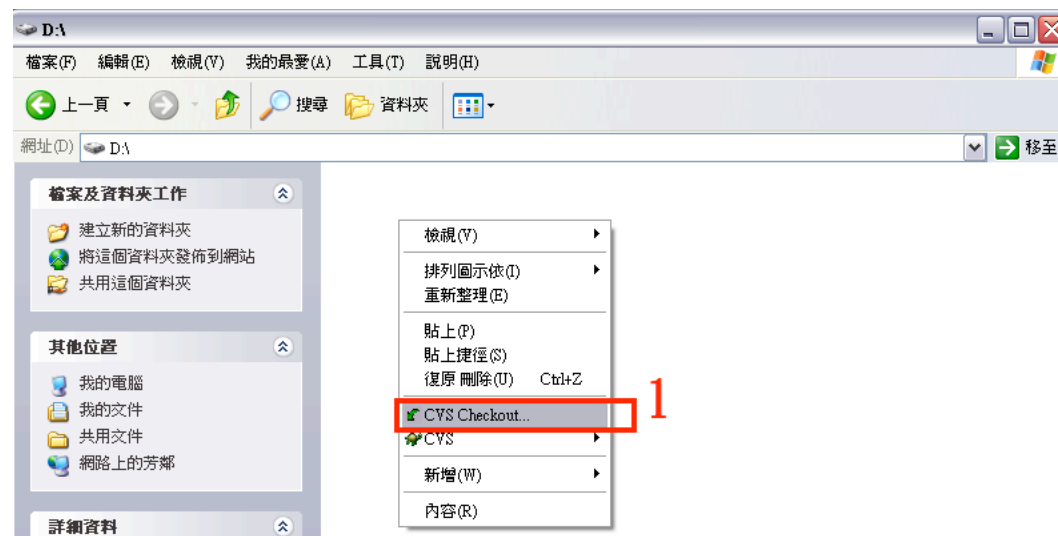
輸入登入密碼。



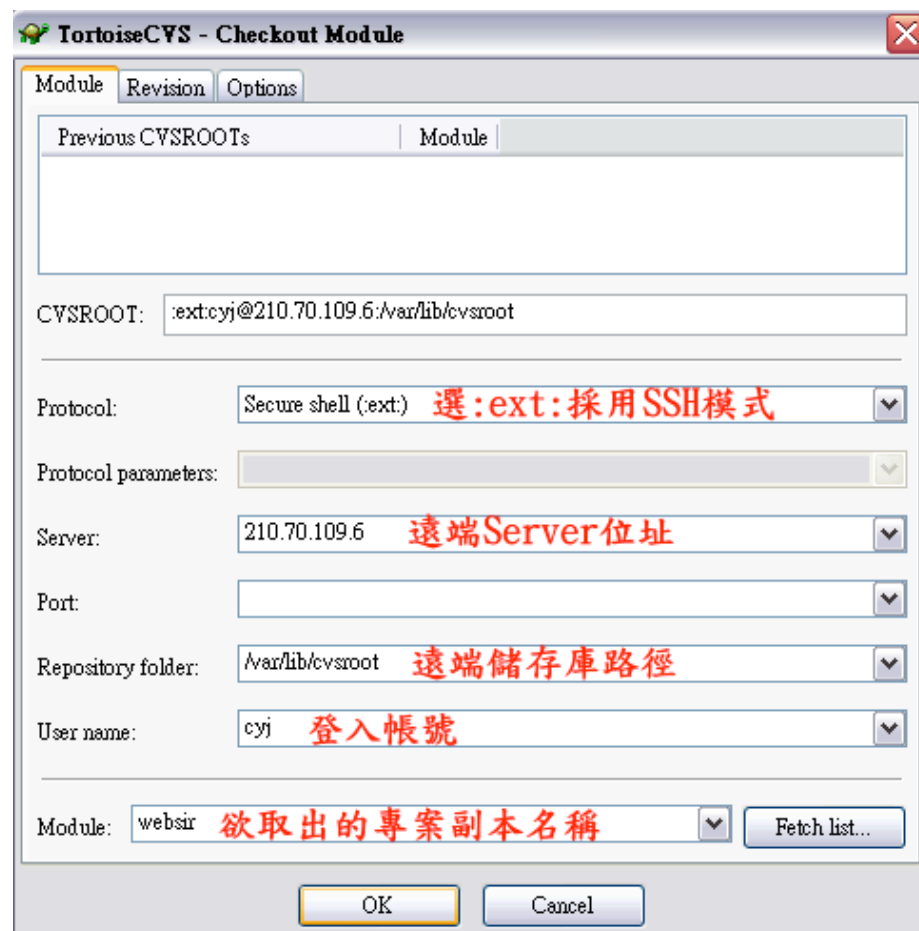
專案匯入成功！

2. 從儲存庫取出專案副本

以下示範將websir專案從遠端儲存庫取出到D:\websir上。



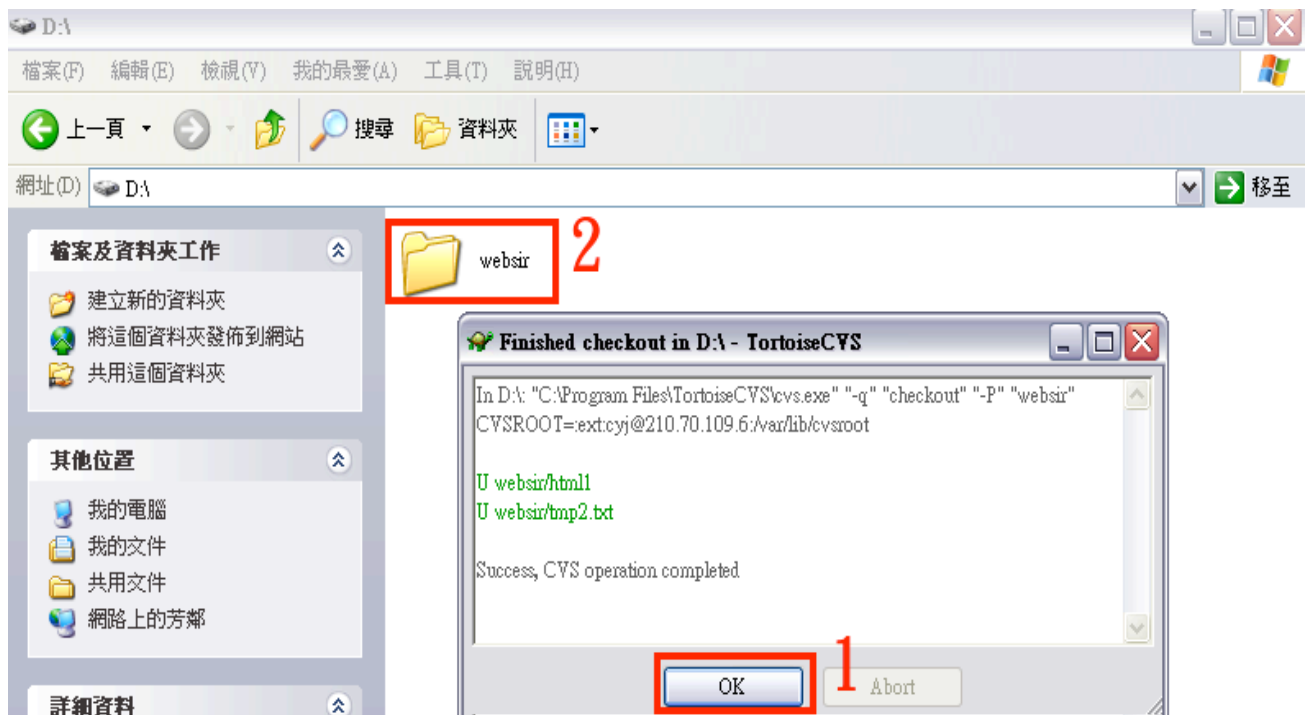
點選CVS Checkout。



依照圖示紅字部分依序輸入遠端CVS Server的資訊。



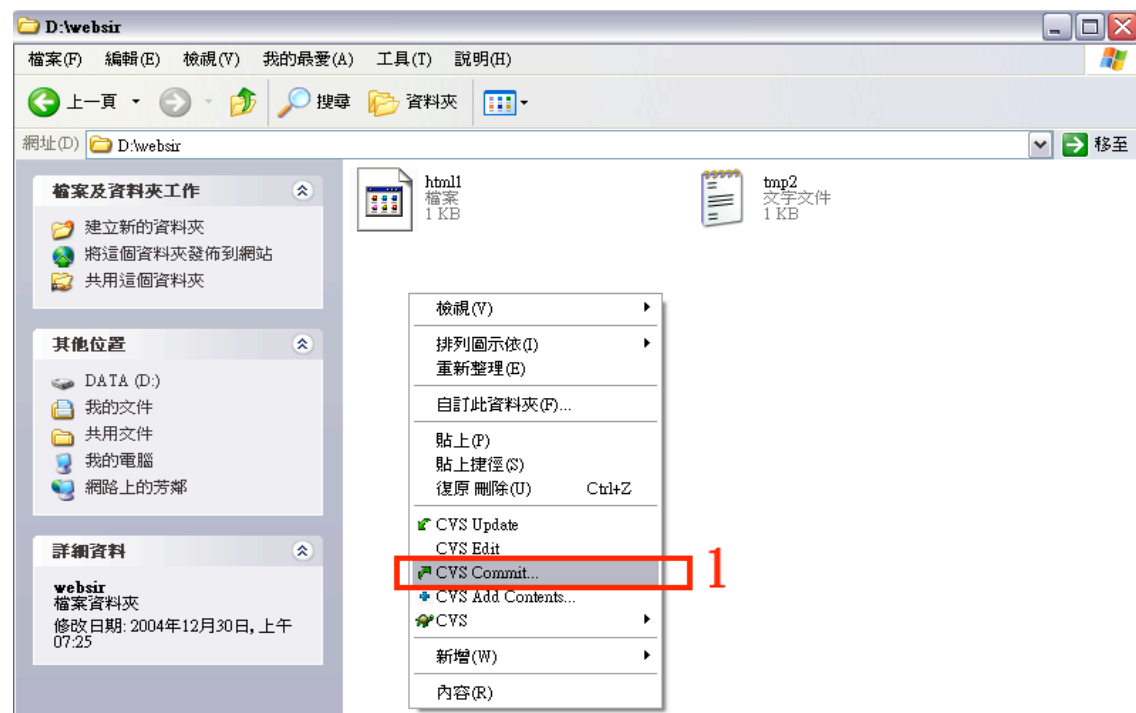
輸入登入密碼



成功的將websir專案從遠端儲存庫取出至D:\websir !

3. 將專案副本變動後的資料交付至儲存庫

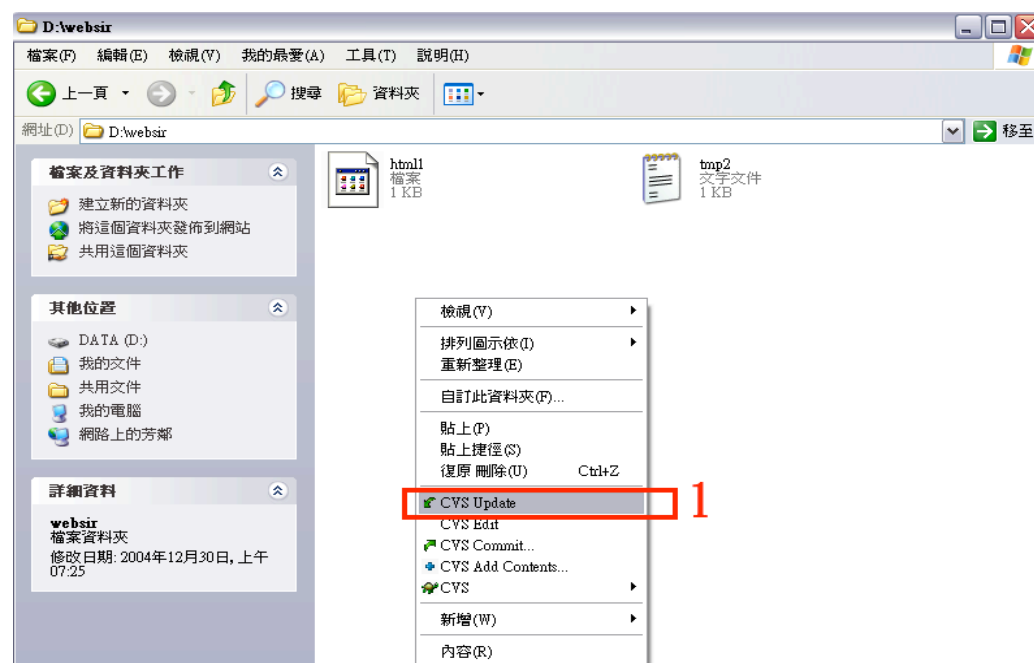
以下示範將D:\websir專案的檔案commit至遠端儲存庫。



點選CVS Commit即可。

4. 更新本地端的專案副本

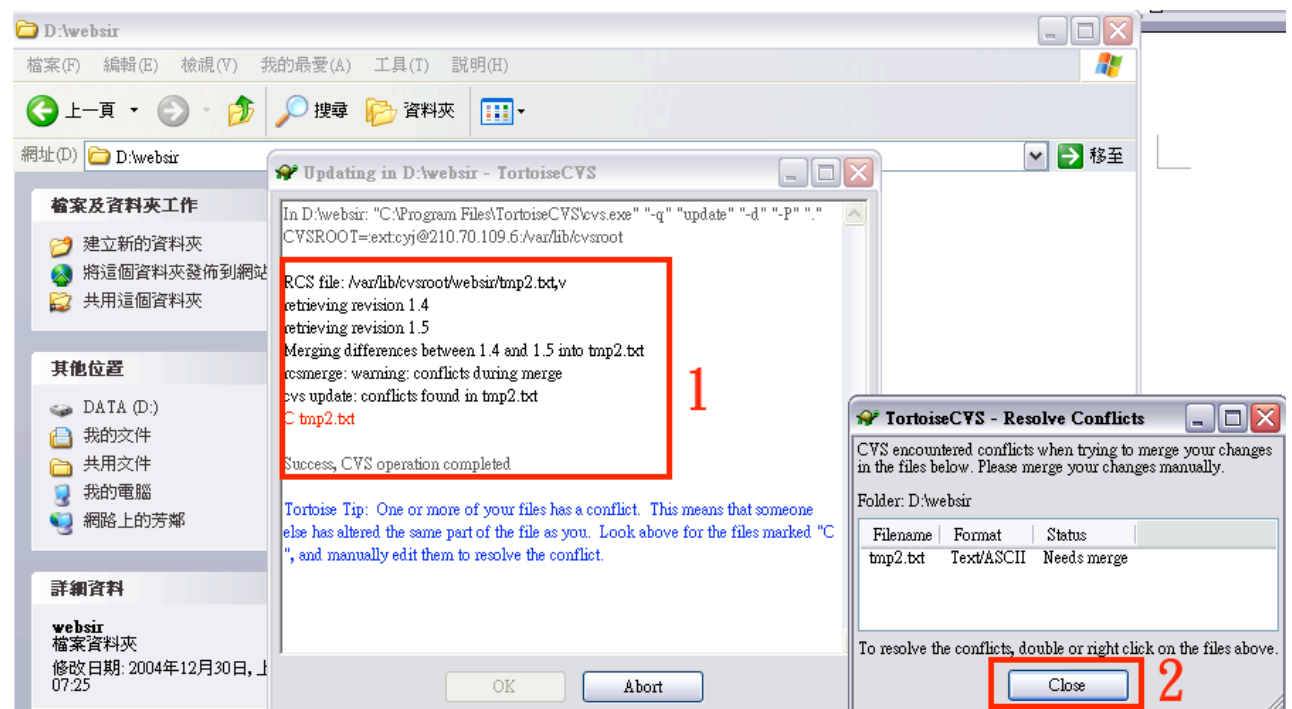
以下示範將遠端儲存庫websir專案update至D:\websir。



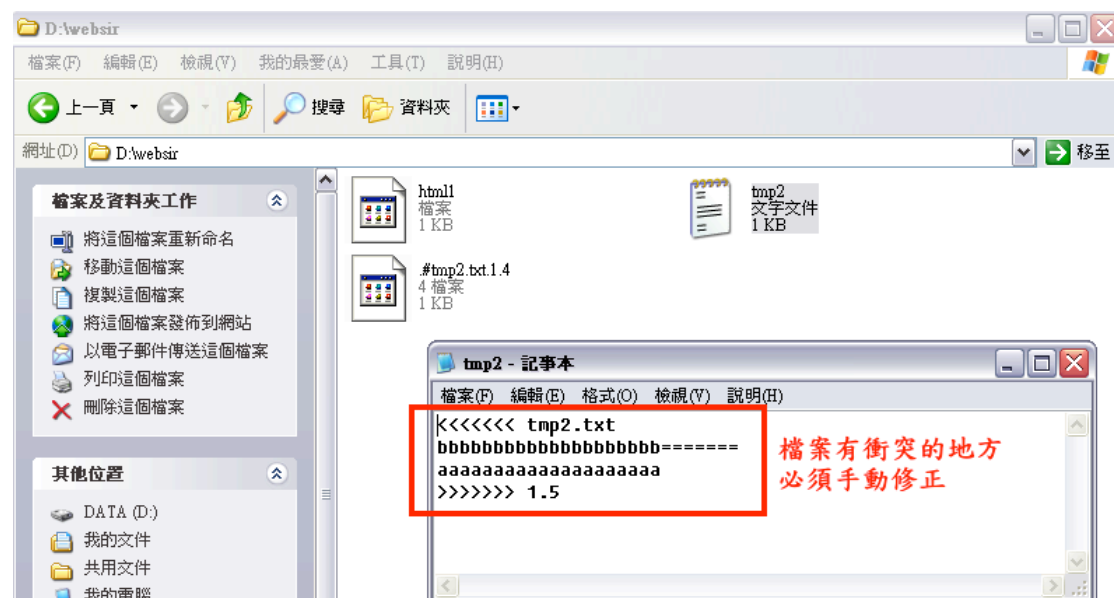
點選CVS Update即可。

5. 檔案發生衝突該如何處理

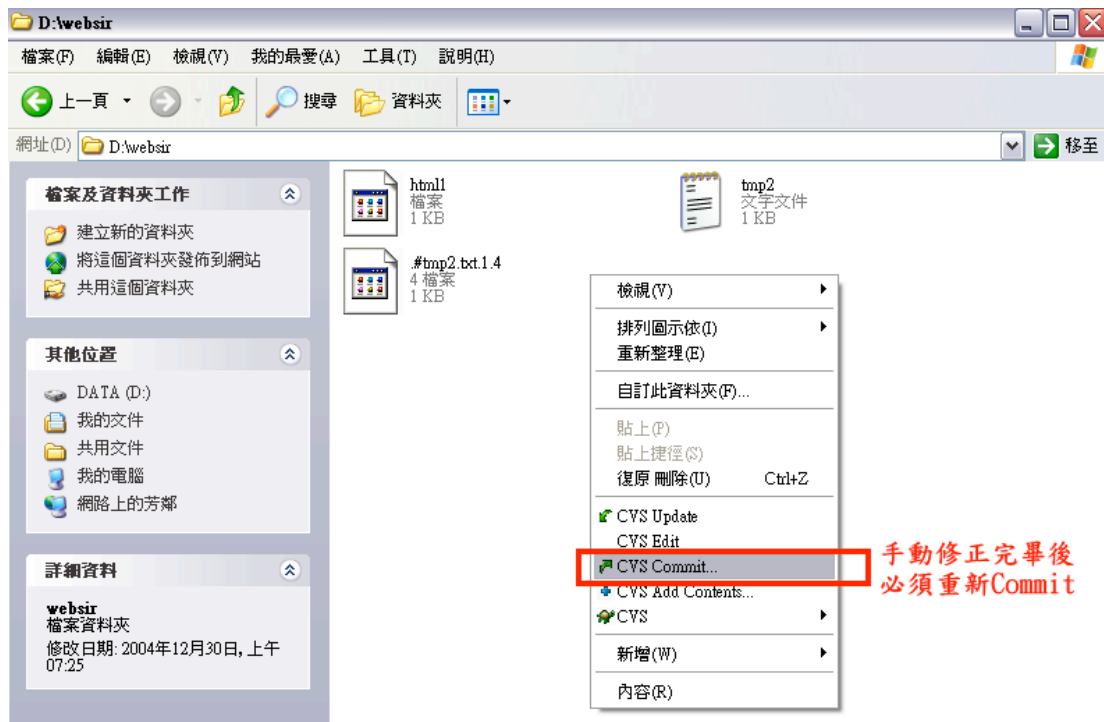
以下示範若update/commit時發生檔案衝突該如何處理。



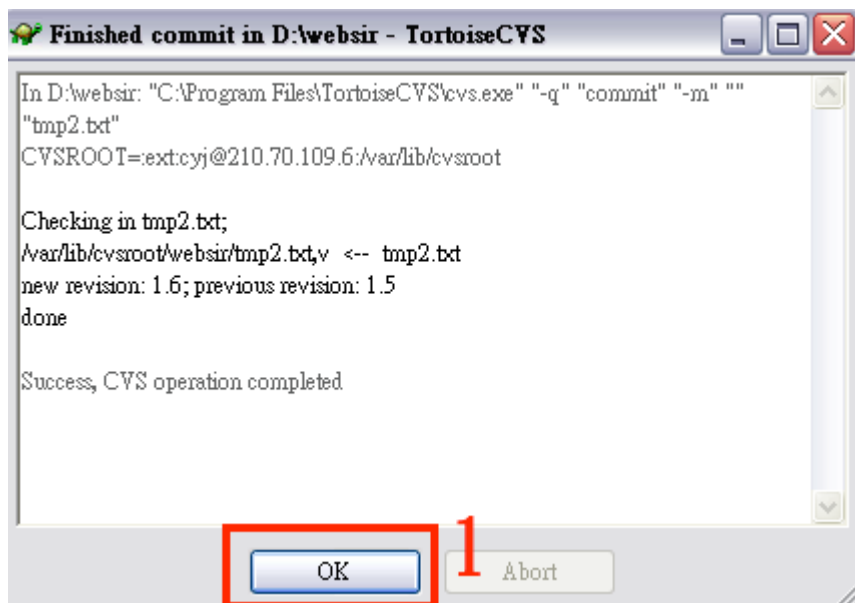
畫面顯示檔案發生衝突，必須手動修正。



檔案衝突的地方必須手動修正。



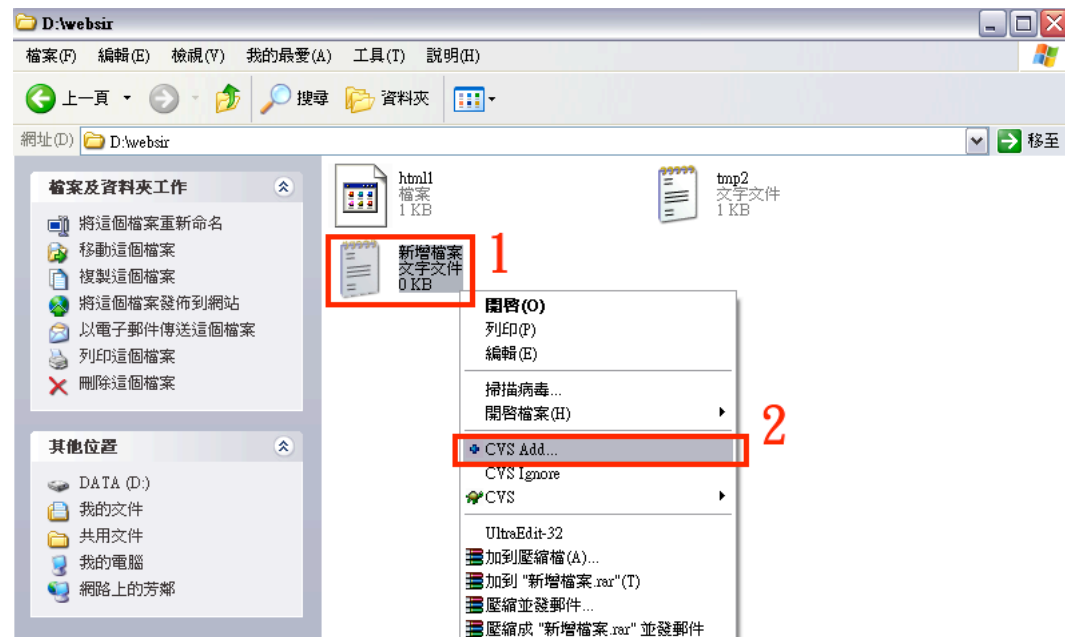
手動修正完畢必須重新Commit。



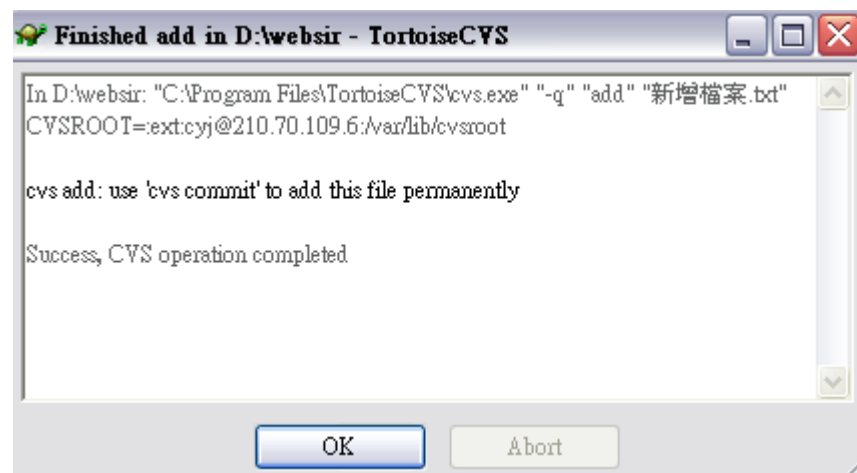
衝突修正完成，檔案交付成功。

6. 替儲存庫中的專案新增檔案

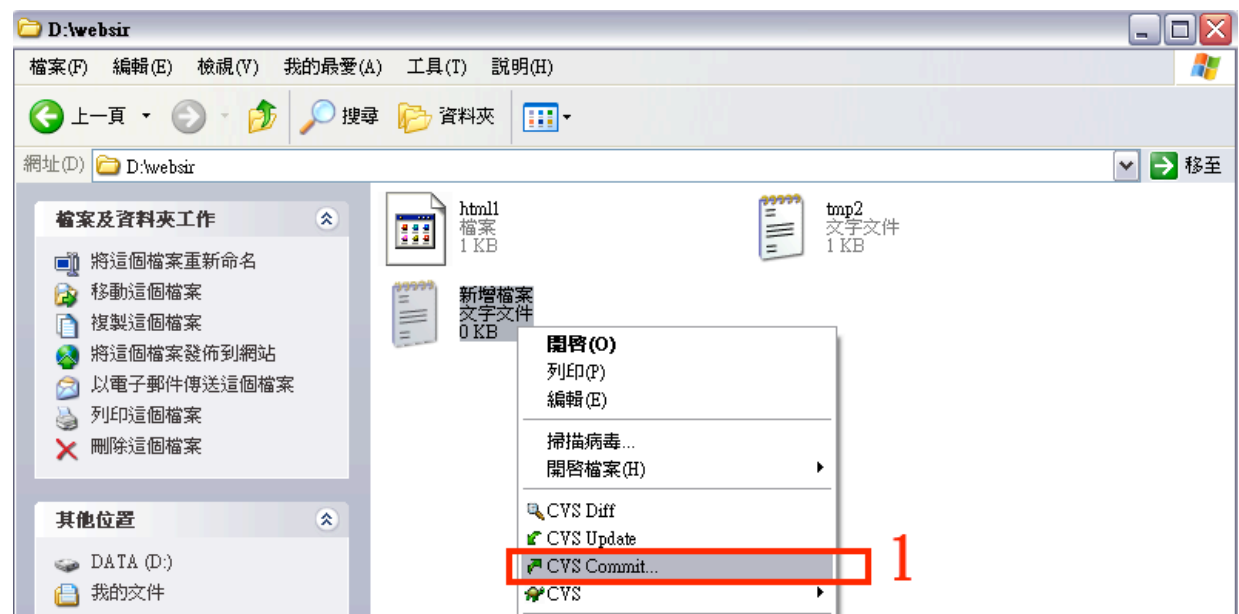
以下示範在D:\websir專案中新增檔案至遠端儲存庫。



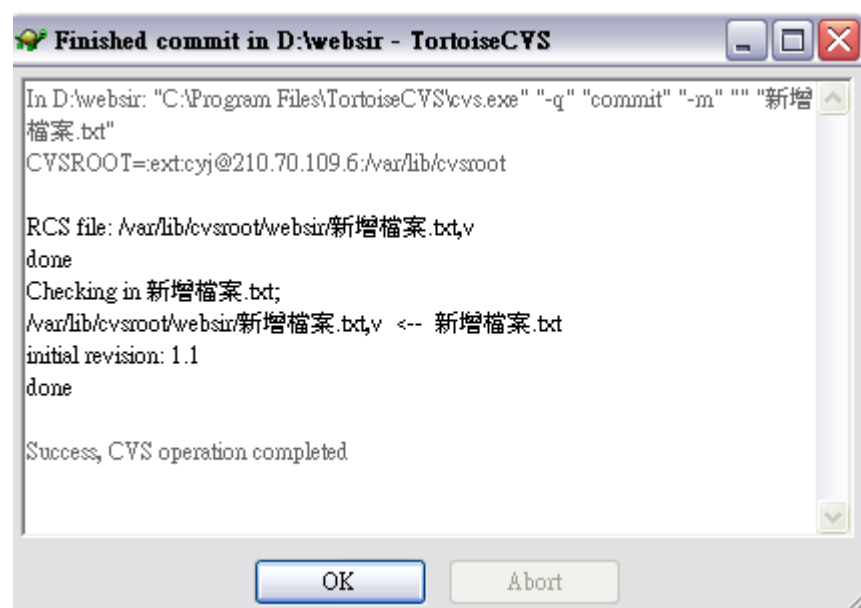
首先建立新檔，接著點選新檔按滑鼠右鍵，選擇CVS Add。



檔案新增成功，需要在執行一次CVS Commit才行。



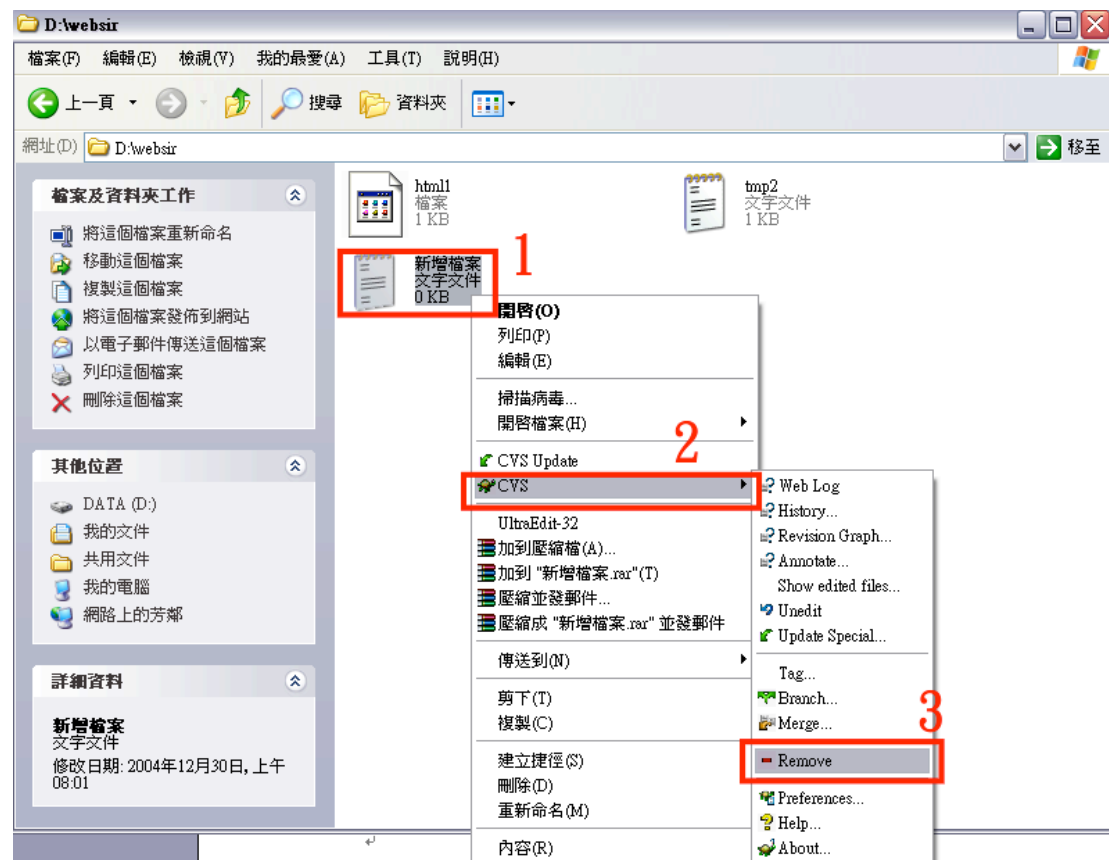
執行CVS Commit。



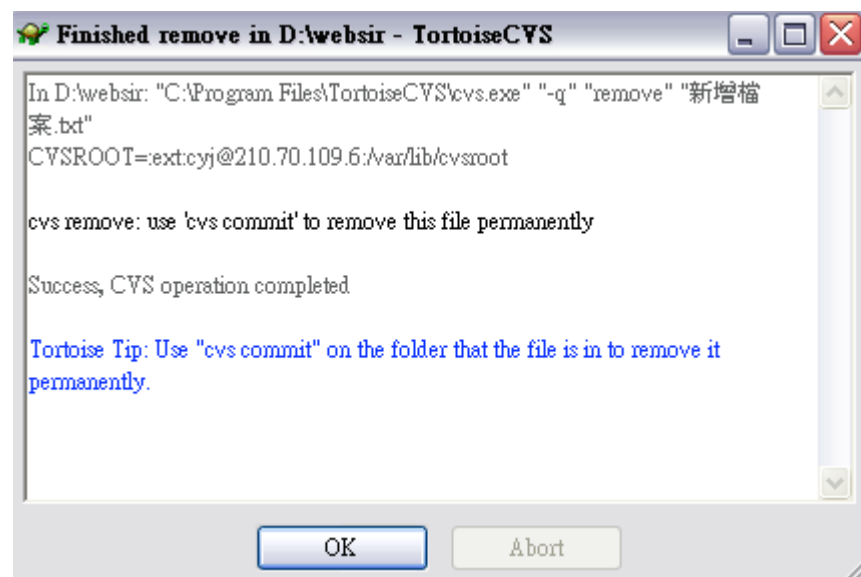
檔案新增成功。

7. 替儲存庫中的專案移除檔案

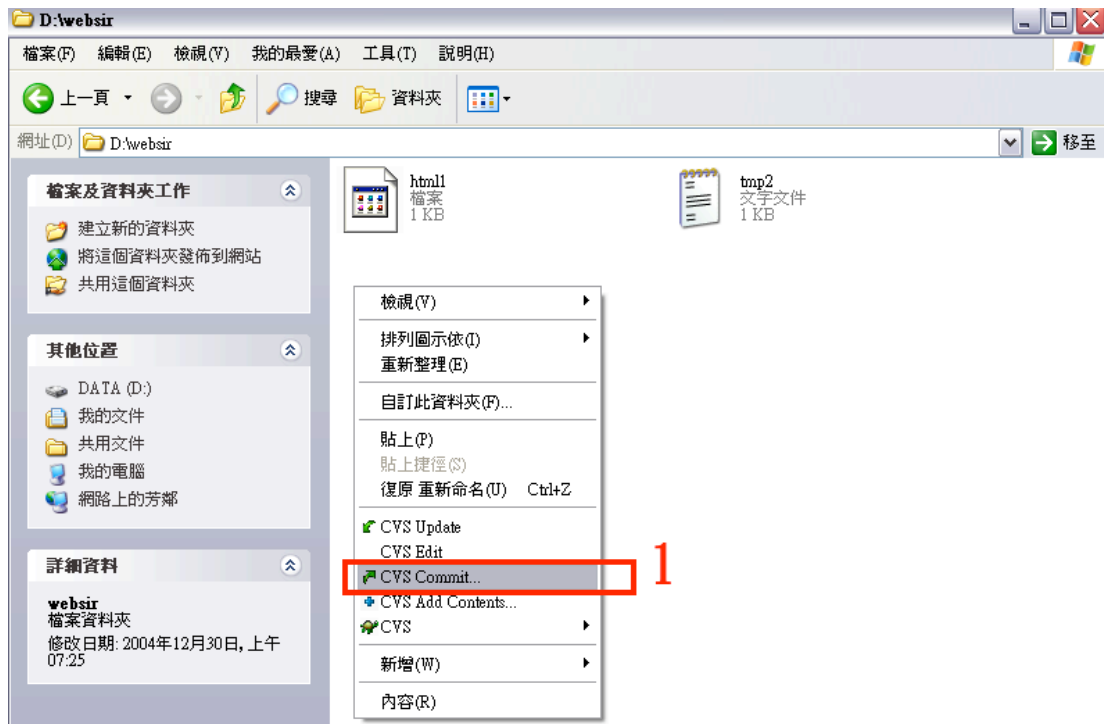
以下示範將遠端儲存庫中的檔案移除。



首先選擇欲刪除的檔案名稱，接著按滑鼠右鍵，選擇CVS、Remove。



檔案Remove成功，需在執行一次CVS Commit才行。



執行CVS Commit。



已成功將遠端儲存庫中的檔案移除。

更多參考資源

CVS 入門 (<http://linux.tnc.edu.tw/techdoc/cvs/book1.html>)

精通CVS (<http://www.oreilly.com.tw/chinese/network/cvs.html>)