Version 2
August 2007

Beginning with TortoiseCVS

A guide to begin using the TortoiseCVS client for source code management on the CropForge collaborative software development site

IRRI

This work was funded by the Generation Challenge Programme
http://www.generationcp.org

CONTENTS

1. Introduction

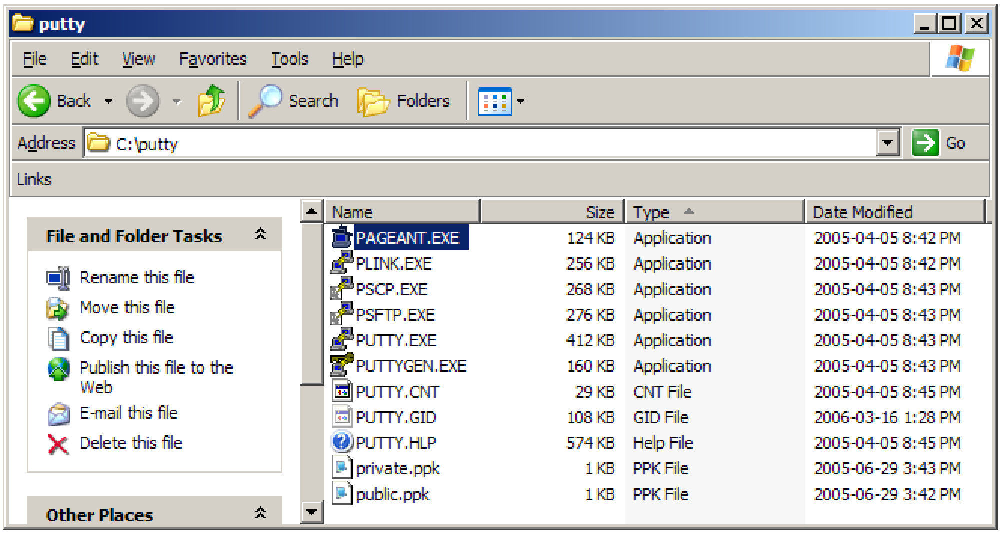
The Concurrent Versioning System (CVS) enables developers to work concurrently by using the **copy-modify-merge** model. Developers download a working copy of the source code from a CVS server, make changes, and subsequently upload those changes into the CVS repository. The source code is usually in a directory tree that contains the files that make up the software project.

This manual describes the steps needed to create an initial local copy of the source code from a CVS repository located on the CropForge server. The client software used is TortoiseCVS.

It is assumed that the TortoiseCVS software has already been installed in the client machine, and that the user has an account on CropForge. Also, the user must be a developer in the project from which the source code will be downloaded. Furthermore, it is assumed that the authentication software PuTTY has been installed, that a pair of public and private keys have been generated, and that the public key has been uploaded into the user's CropForge account. Refer to CodeOnCropForge.pdf for details on how to do these steps.

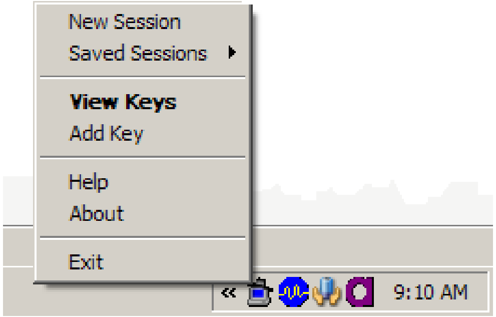
2. Loading the private authentication key

It is assumed that the PuTTY software is being used for authentication with the CropForge server. The default PuTTY installation should have resulted in a directory **c:\putty**. The public and private keys in the example below were saved under the names **private.ppk** and **public.ppk**.



Name	Size	Type	Date Modified
PAGEANT.EXE	124 KB	Application	2005-04-05 8:42 PM
PLINK.EXE	256 KB	Application	2005-04-05 8:42 PM
PSCP.EXE	268 KB	Application	2005-04-05 8:43 PM
PSFTP.EXE	276 KB	Application	2005-04-05 8:43 PM
PUTTY.EXE	412 KB	Application	2005-04-05 8:43 PM
PUTTYGEN.EXE	160 KB	Application	2005-04-05 8:43 PM
PUTTY.CNT	29 KB	CNT File	2005-04-05 8:45 PM
PUTTY.GID	108 KB	GID File	2006-03-16 1:28 PM
PUTTY.HLP	574 KB	Help File	2005-04-05 8:45 PM
private.ppk	1 KB	PPK File	2005-06-29 3:43 PM
public.ppk	1 KB	PPK File	2005-06-29 3:42 PM

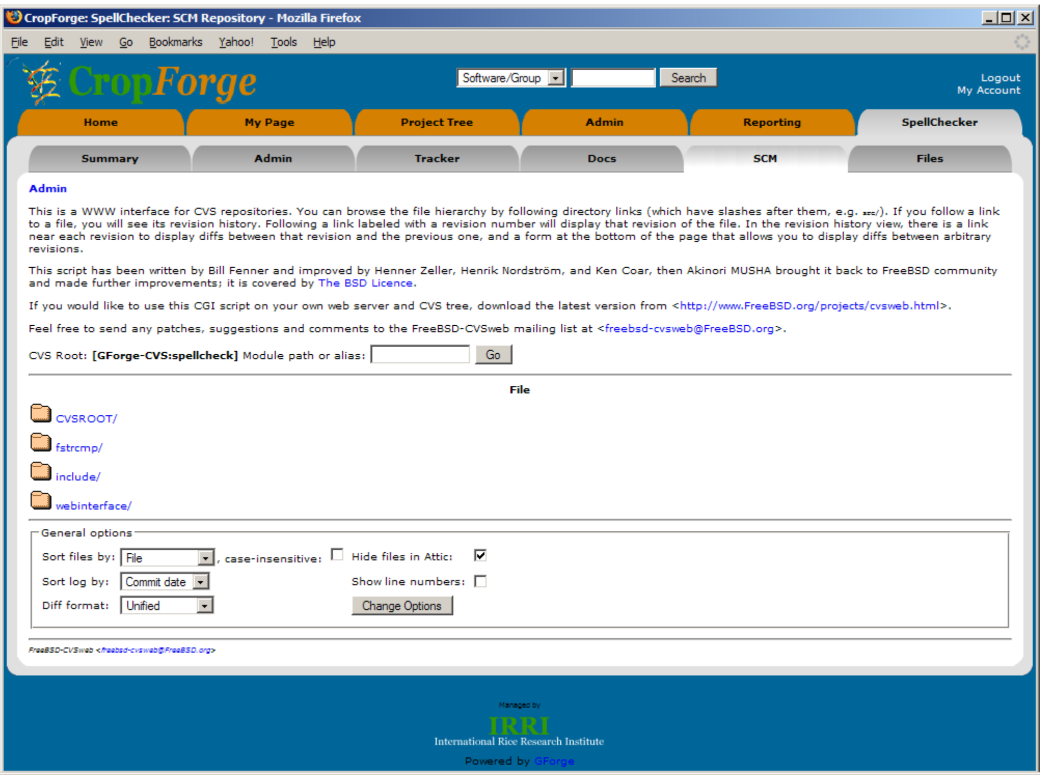
After starting the PAGENT.EXE program (PuTTY authentication agent), a corresponding icon appears in the taskbar. Right-click on the icon, select the option **Add Key**, and select the file containing your private key. In this example, the private key is stored in the file **c:\putty\private.ppk**.



It is assumed that your public key has already been uploaded to the CropForge server. Whenever authentication between you and the CropForge server is needed, it will now be done automatically by the PAGENT software using your locally loaded private key and your public key available on the CropForge server. Again, refer to CodeOnCropForge.pdf for details on how to do this.

3. Initial downloading of a module

The project used in this example is called **spellcheck**, and it consists of three modules: **fstrcmp**, **include**, and **webinterface**. The screen shot below shows the source code repository of this project on CropForge.



Admin

This is a WWW interface for CVS repositories. You can browse the file hierarchy by following directory links (which have slashes after them, e.g. src/). If you follow a link to a file, you will see its revision history. Following a link labeled with a revision number will display that revision of the file. In the revision history view, there is a link near each revision to display diffs between that revision and the previous one, and a form at the bottom of the page that allows you to display diffs between arbitrary revisions.

This script has been written by Bill Fenner and improved by Henner Zeller, Henrik Nordström, and Ken Coar, then Akinori MUSHA brought it back to FreeBSD community and made further improvements; it is covered by The BSD Licence.

If you would like to use this CGI script on your own web server and CVS tree, download the latest version from <http://www.FreeBSD.org/projects/cvsweb.html>.

Feel free to send any patches, suggestions and comments to the FreeBSD-CVSweb mailing list at <freebsd-cvsweb@FreeBSD.org>.

CVS Root: **[GForge-CVS:spellcheck]** Module path or alias: Go

File

CVSROOT/
fstrcmp/
include/
webinterface/

General options

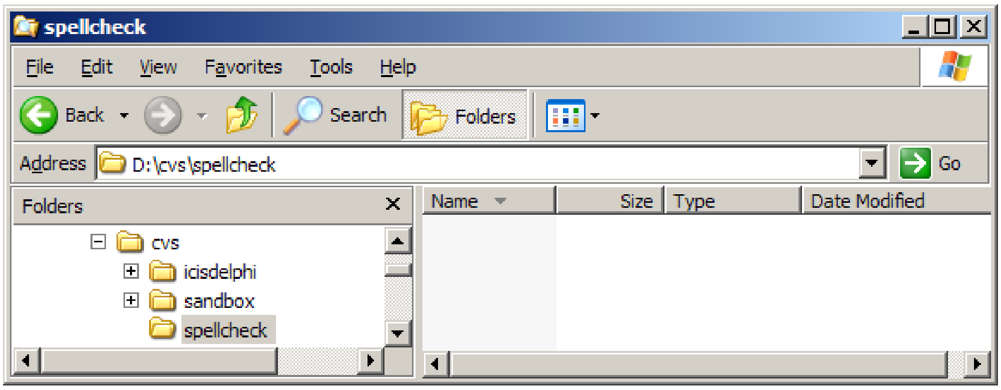
Sort files by: File, case-insensitive: □ Hide files in Attic: ☑
Sort log by: Commit date Show line numbers: □
Diff format: Unified Change Options

FreeBSD-CVSweb <freebsd-cvsweb@FreeBSD.org>

In the following steps, a separate local copy of the source code is created for each module, reflecting the structure of the project's CVS source code repository.

3.1. Creating a local project folder

The first step is to create a project folder on your local hard disk for maintaining the source code under CVS. In this example, there is already a directory **\cvs** that contains the code of two projects, **icisdelphi** and **sandbox**. To the **\cvs** directory, a new subdirectory is added with the same name as the CropForge project to be downloaded. It is recommended to give the local directory the same name as the project to be downloaded from the CropForge server.



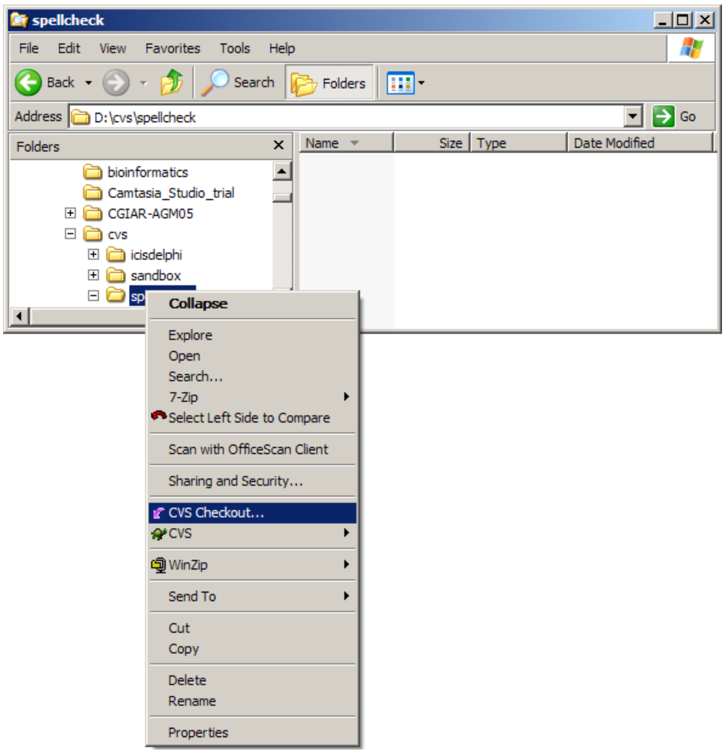
Address: D:\cvs\spellcheck

Note: Only the project directory needs to be created. The module directories will be created as part of the initial source code download (checkout) process.

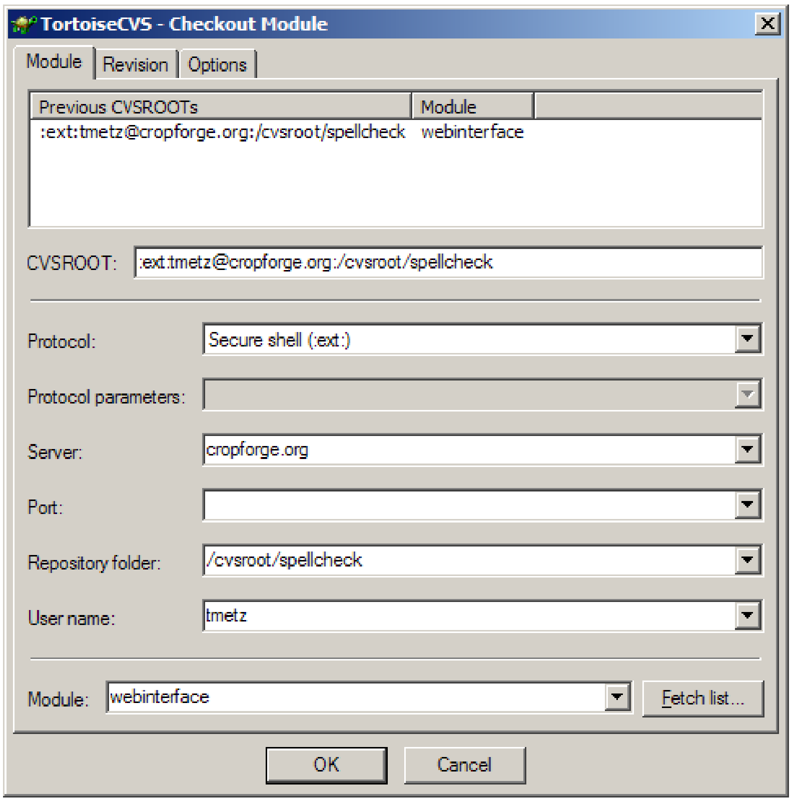
3.2. Checking out the first module

The initial downloading of a source code directory is also called **checking out**. To initiate the checkout of a module, right-click on your local project directory (**spellcheck** in this example).

The Explorer menu should show two additional options, **CVS Checkout…** and **CVS**. These additional options become available after the installation of the TortoiseCVS software. Select the **CVS Checkout…** option.



The form below should appear.



Previous CVSROOTs	Module
:ext:tmetz@cropforge.org:/cvsroot/spellcheck	webinterface

CVSROOT: :ext:tmetz@cropforge.org:/cvsroot/spellcheck

Protocol: Secure shell (:ext:)

Protocol parameters:

Server: cropforge.org

Port:

Repository folder: /cvsroot/spellcheck

User name: tmetz

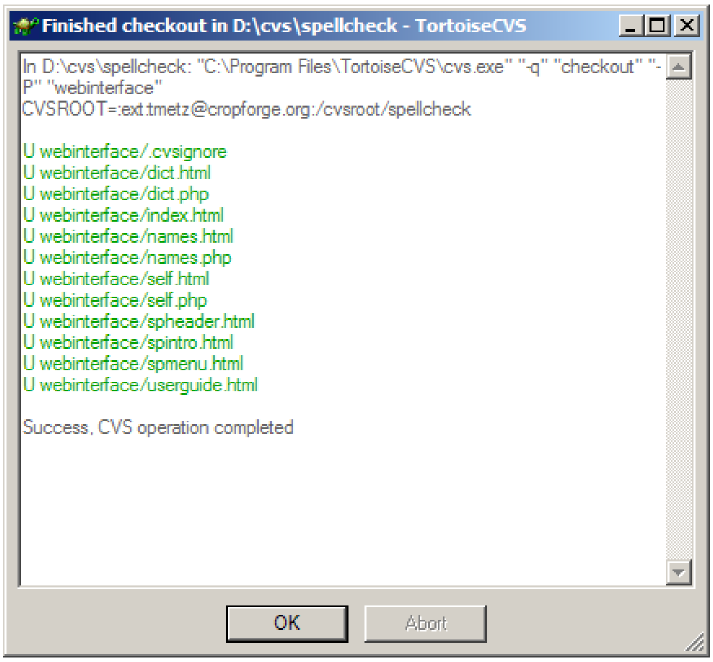
Module: webinterface

This form needs to be filled out to provide TortoiseCVS with the necessary information to contact the CVS repository on the CropForge server, and to request a copy of a directory tree. It is suggested to complete the form using the following steps:

- Select the protocol: **ext**
- Provide the server name: **cropforge.org**
- Provide the repository (project) folder: **/cvsroot/**spellcheck
- Provide your CropForge user name: tmetz
- Provide the module name: webinterface

Note: Supply the appropriate project name (this example: spellcheck) and module name (this example: webinterface), as shown in the CropForge web interface. You must also supply your own user name (this example: tmetz). User-specific information is indicated in a blue underlined font.

After clicking the **OK** button, the following window should appear and show the dialog between the TortoiseCVS client and CropForge CVS server.

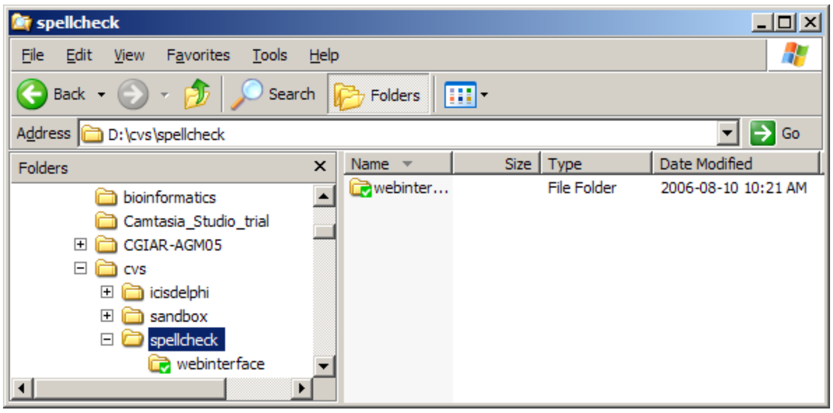


```
In D:\cvs\spellcheck: "C:\Program Files\TortoiseCVS\cvs.exe" "-q" "checkout" "-
P" "webinterface"
CVSROOT=:ext:tmetz@cropforge.org:/cvsroot/spellcheck

U webinterface/.cvsignore
U webinterface/dict.html
U webinterface/dict.php
U webinterface/index.html
U webinterface/names.html
U webinterface/names.php
U webinterface/self.html
U webinterface/self.php
U webinterface/spheader.html
U webinterface/spintro.html
U webinterface/spmenu.html
U webinterface/userguide.html

Success, CVS operation completed
```

The dialog shows the CVS command sent to the server and the resulting actions. The lines in a green font indicate that a number of files have been downloaded to your local project and module directory.

3.3. Inspecting your local module copy

A module directory (webinterface) should have been added to your project directory (spellcheck). TortoiseCVS uses special icons for folders and files to indicate the local status of these folders and files in relation to the CVS repository on CropForge.

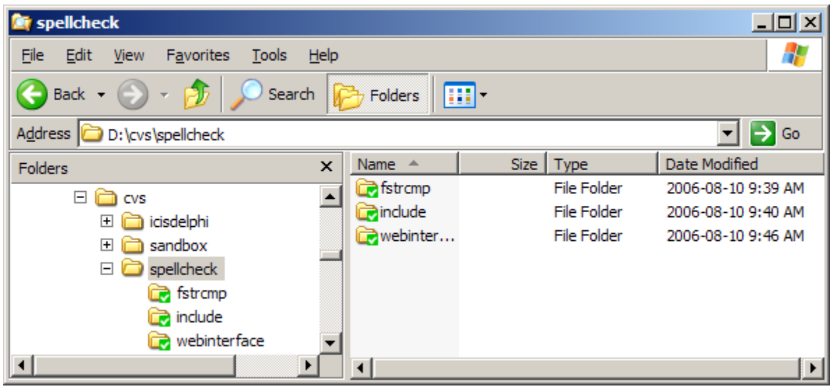


Address D:\cvs\spellcheck

Name	Size	Type	Date Modified
webinter...		File Folder	2006-08-10 10:21 AM

3.4. Checking out the remaining modules

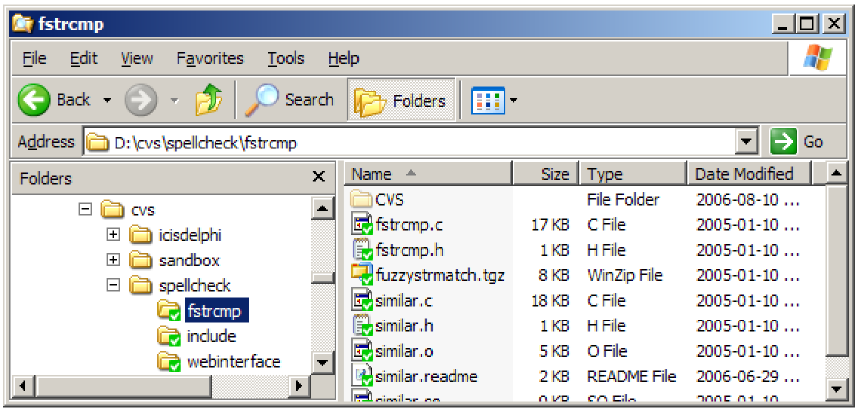
After repeating the checkout process for the remaining modules, the local project directory should now have all the module directories that are available in the CVS project repository.



Address D:\cvs\spellcheck

Name	Size	Type	Date Modified
fstrcmp		File Folder	2006-08-10 9:39 AM
include		File Folder	2006-08-10 9:40 AM
webinter…		File Folder	2006-08-10 9:46 AM

Displaying a module directory content shows that the special icons are also used for files. Furthermore, a special directory CVS has been created. This special directory contains all the necessary status and communication information that is needed by the TortoiseCVS client to work with the CVS server. The information in this special directory is automatically maintained, and should not be edited by the user.

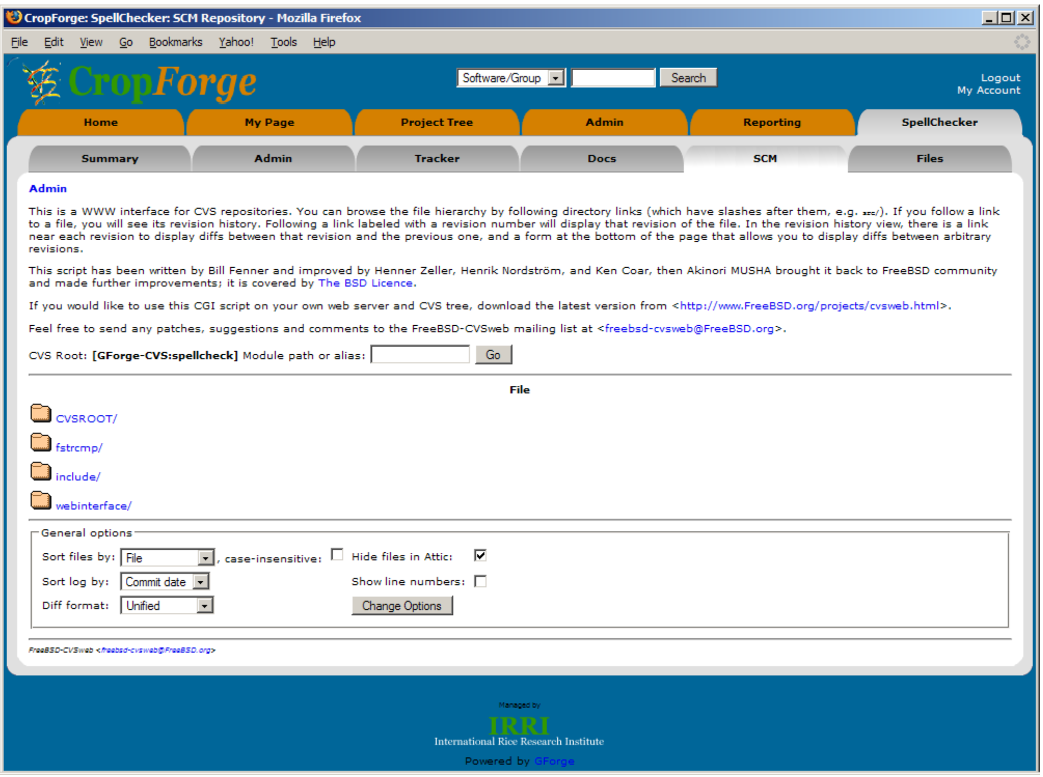


Address D:\cvs\spellcheck\fstrcmp

Name	Size	Type	Date Modified
CVS		File Folder	2006-08-10 …
fstrcmp.c	17 KB	C File	2005-01-10 …
fstrcmp.h	1 KB	H File	2005-01-10 …
fuzzystrmatch.tgz	8 KB	WinZip File	2005-01-10 …
similar.c	18 KB	C File	2005-01-10 …
similar.h	1 KB	H File	2005-01-10 …
similar.o	5 KB	O File	2005-01-10 …
similar.readme	2 KB	README File	2006-06-29 …

4. Initial uploading of a module

The project used in this example is called **spellcheck**, and it already consists of three modules: **fstrcmp**, **include**, and **webinterface**. The screen shot below shows the source code repository of this project on CropForge.



Admin

This is a WWW interface for CVS repositories. You can browse the file hierarchy by following directory links (which have slashes after them, e.g. src/). If you follow a link to a file, you will see its revision history. Following a link labeled with a revision number will display that revision of the file. In the revision history view, there is a link near each revision to display diffs between that revision and the previous one, and a form at the bottom of the page that allows you to display diffs between arbitrary revisions.

This script has been written by Bill Fenner and improved by Henner Zeller, Henrik Nordström, and Ken Coar, then Akinori MUSHA brought it back to FreeBSD community and made further improvements; it is covered by The BSD Licence.

If you would like to use this CGI script on your own web server and CVS tree, download the latest version from <http://www.FreeBSD.org/projects/cvsweb.html>.

Feel free to send any patches, suggestions and comments to the FreeBSD-CVSweb mailing list at <freebsd-cvsweb@FreeBSD.org>.

CVS Root: **[GForge-CVS:spellcheck]** Module path or alias: Go

File

- CVSROOT/
- fstrcmp/
- include/
- webinterface/

General options

Sort files by: File, case-insensitive: □ Hide files in Attic: ☑

Sort log by: Commit date Show line numbers: □

Diff format: Unified

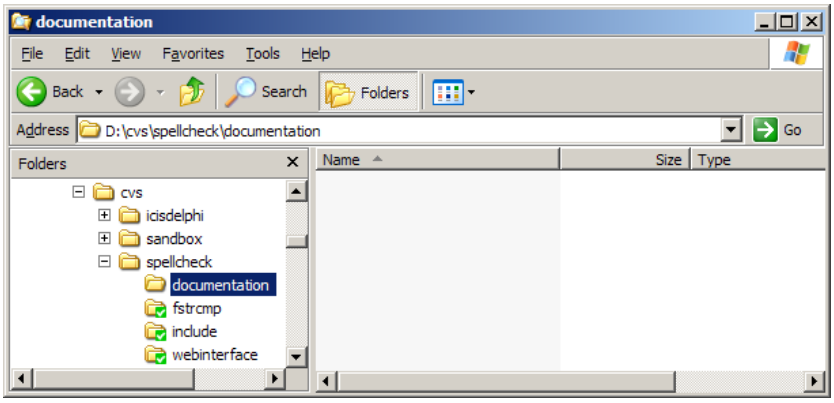
Change Options

FreeBSD-CVSweb <freebsd-cvsweb@FreeBSD.org>

We want to create an additional module in this project, and upload the content of the module to the project's CropForge repository. Note that the steps described below work in exactly the same way when adding the first module to a project's source code repository.

4.1. Creating the local module directory

The first step is to create a project folder on your local hard disk for maintaining the source code under CVS. In this example, there is already a directory **\cvs** that contains the code of two projects, **icisdelphi** and **sandbox**. To the **\cvs\spellcheck** directory, a new subdirectory is added (**documentation**) that will hold the content of the new module. The naming of the directories is up to the user, but it is recommended to use names consisting of single words in lowercase without punctuation.

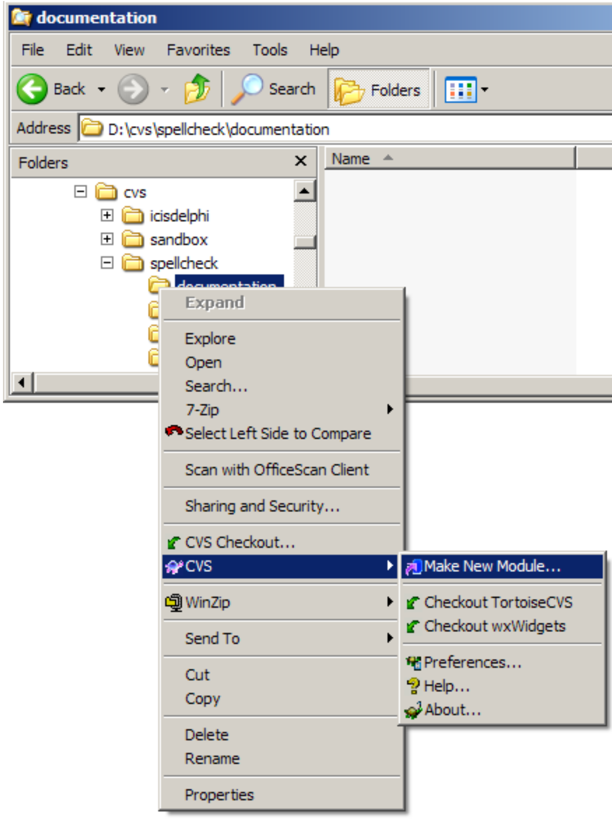


Address: D:\cvs\spellcheck\documentation

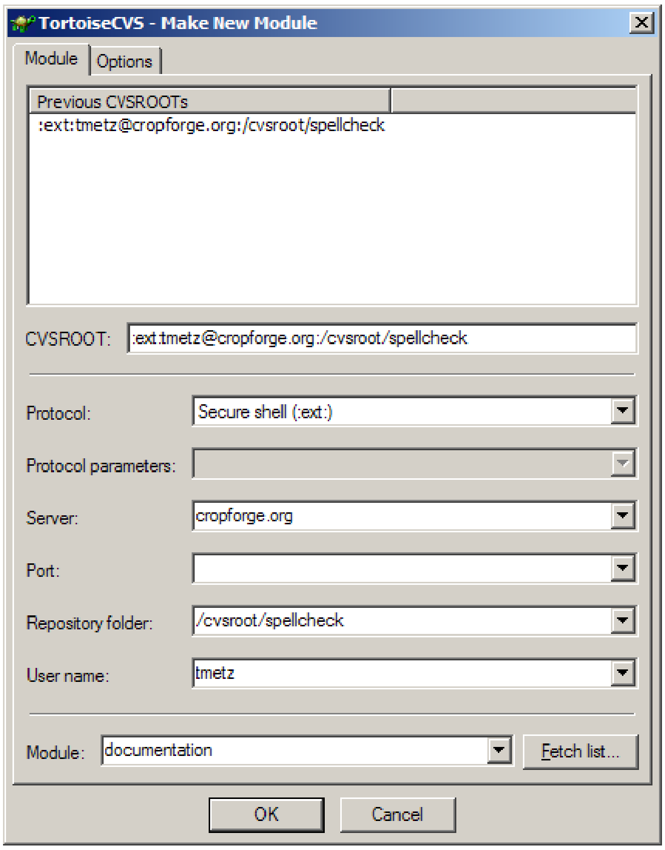
In the Explorer window, the new folder appears initially as an ordinary folder, with the default folder icon.

4.2. Creating a new module in the CropForge CVS repository

To create a corresponding repository folder in the project's CVS repository on CropForge, right-click on the directory name (**documentation**) and select the **CVS** option followed by the **Make New Module…** suboption.



The following form should appear.



Previous CVSROOTs
:ext:tmetz@cropforge.org:/cvsroot/spellcheck

CVSROOT: :ext:tmetz@cropforge.org:/cvsroot/spellcheck

Protocol: Secure shell (:ext:)

Protocol parameters:

Server: cropforge.org

Port:

Repository folder: /cvsroot/spellcheck

User name: tmetz

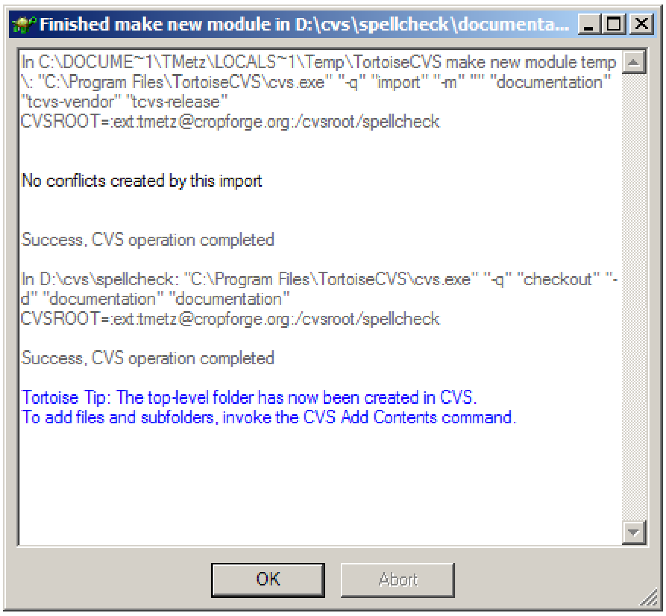
Module: documentation

This form needs to be filled out to provide TortoiseCVS with the necessary information to contact the CVS repository on the CropForge server, and to request the creation of a new module. It is suggested to complete the form using the following steps:

- Select the protocol: **ext**
- Provide the server name: **cropforge.org**
- Provide the repository (project) folder: **/cvsroot/**spellcheck
- Provide your CropForge user name: tmetz
- Provide the module name: documentation

Note: Supply the appropriate project name (this example: spellcheck) and module name (this example: documentation), as shown in the CropForge web interface. You must also supply your own user name (this example: tmetz). User-specific information is indicated in a blue underlined font.

After clicking the **OK** button, the following window should appear and show the dialog between the TortoiseCVS client and the CropForge CVS server.



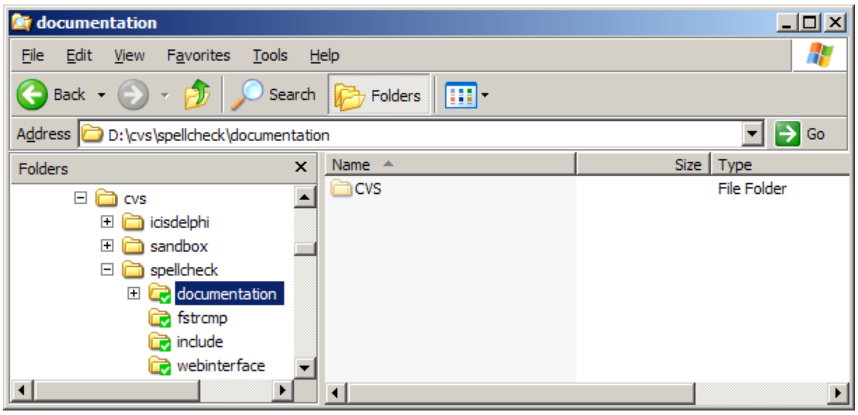
Finished make new module in D:\cvs\spellcheck\documenta...

```
In C:\DOCUME~1\TMetz\LOCALS~1\Temp\TortoiseCVS make new module temp
\: "C:\Program Files\TortoiseCVS\cvs.exe" "-q" "import" "-m" "" "documentation"
"tcvs-vendor" "tcvs-release"
CVSROOT=:ext:tmetz@cropforge.org:/cvsroot/spellcheck


No conflicts created by this import


Success, CVS operation completed

In D:\cvs\spellcheck: "C:\Program Files\TortoiseCVS\cvs.exe" "-q" "checkout" "-
d" "documentation" "documentation"
CVSROOT=:ext:tmetz@cropforge.org:/cvsroot/spellcheck

Success, CVS operation completed

Tortoise Tip: The top-level folder has now been created in CVS.
To add files and subfolders, invoke the CVS Add Contents command.
```

The dialog shows the CVS command sent to the server and the resulting actions. The lines in a blue font list the other steps that are necessary to add files and subfolders to the CVS repository.

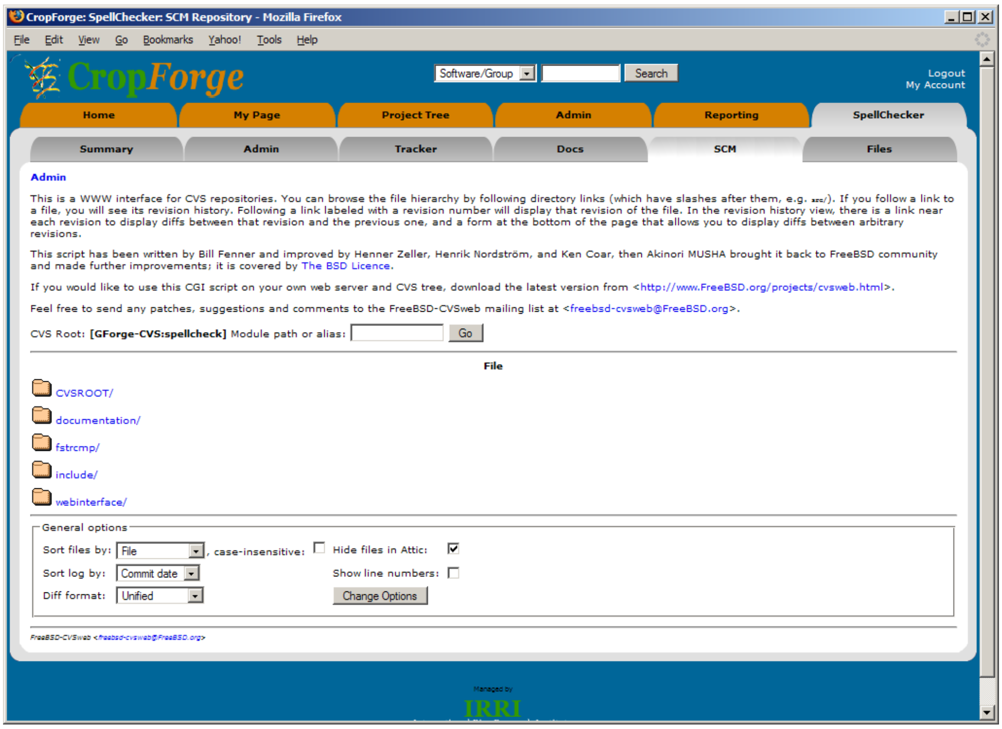
The Explorer window below shows that the local directory, representing a new module (documentation), is now under CVS control. Note the change in the folder icon from a normal folder icon to one with a green check/tick.

Furthermore, a special hidden directory named **CVS** has been created. This special directory contains all the necessary status and communication information that is needed by the TortoiseCVS client to work with the CVS server. The information in this special hidden directory is automatically maintained, and should not be edited or deleted by the user.



Name	Size	Type
CVS		File Folder

When we view the CVS repository of the **spellcheck** project on the CropForge web interface, we see that a corresponding new module subdirectory has been created.



Admin

This is a WWW interface for CVS repositories. You can browse the file hierarchy by following directory links (which have slashes after them, e.g. src/). If you follow a link to a file, you will see its revision history. Following a link labeled with a revision number will display that revision of the file. In the revision history view, there is a link near each revision to display diffs between that revision and the previous one, and a form at the bottom of the page that allows you to display diffs between arbitrary revisions.

This script has been written by Bill Fenner and improved by Henner Zeller, Henrik Nordström, and Ken Coar, then Akinori MUSHA brought it back to FreeBSD community and made further improvements; it is covered by The BSD Licence.

If you would like to use this CGI script on your own web server and CVS tree, download the latest version from <http://www.FreeBSD.org/projects/cvsweb.html>.

Feel free to send any patches, suggestions and comments to the FreeBSD-CVSweb mailing list at <freebsd-cvsweb@FreeBSD.org>.

CVS Root: **[GForge-CVS:spellcheck]** Module path or alias: Go

File

CVSROOT/
documentation/
fstrcmp/
include/
webinterface/

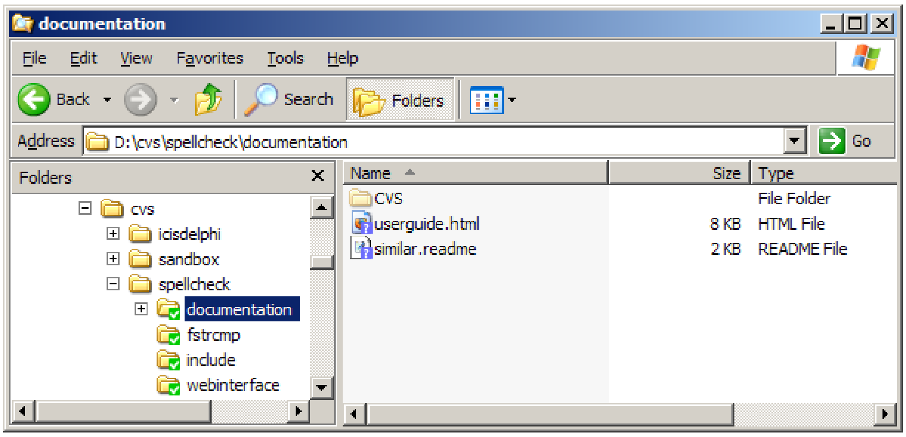
General options

Sort files by: File, case-insensitive: □ Hide files in Attic: ☑
Sort log by: Commit date Show line numbers: □
Diff format: Unified Change Options

FreeBSD-CVSweb <freebsd-cvsweb@FreeBSD.org>

4.3. Copying content into the new module directory

The next step is to copy files into the local module directory (**documentation**). In the current example, we use two text files. Note that the file icons have a blue question mark, indicating that the CVS status of these files is unknown.

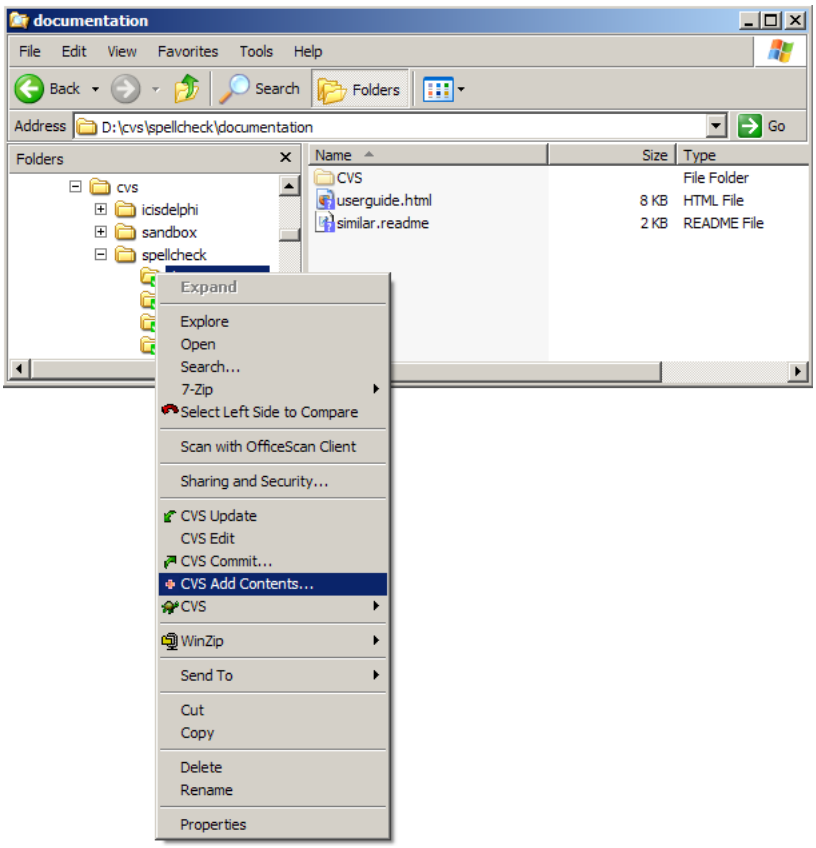


Address D:\cvs\spellcheck\documentation

Name	Size	Type
CVS		File Folder
userguide.html	8 KB	HTML File
similar.readme	2 KB	README File

4.4. Adding content to the local CVS copy

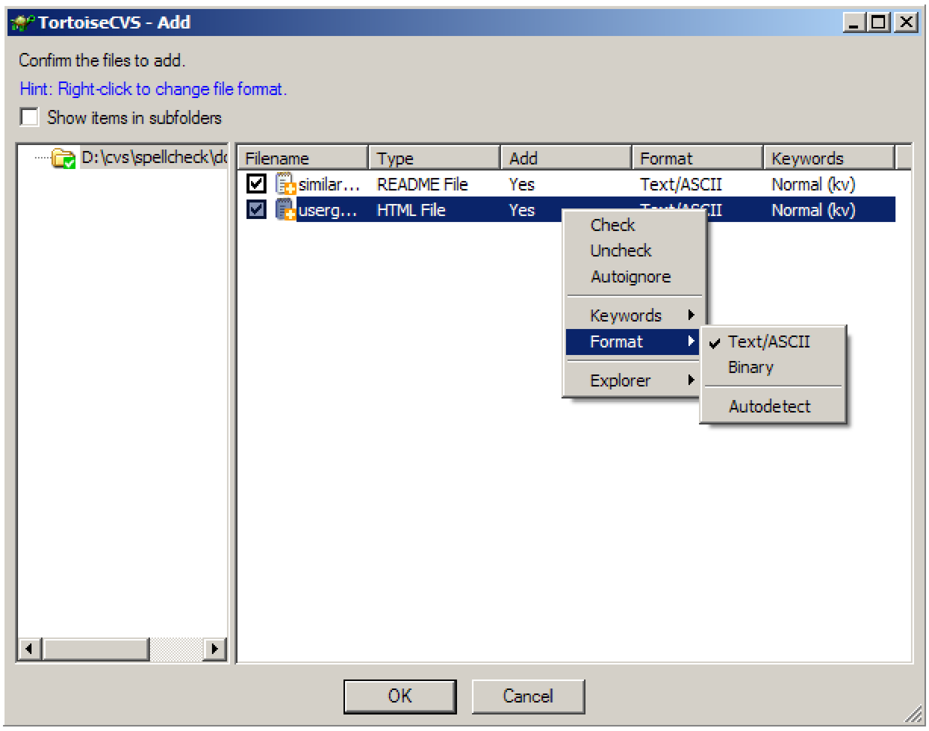
The next step is to add the content of the new module directory (**documentation**) to the local CVS source code copy. Right-click on the directory name and select the option **CVS Add Contents…**



Address D:\cvs\spellcheck\documentation

Name	Size	Type
CVS		File Folder
userguide.html	8 KB	HTML File
similar.readme	2 KB	README File

The following Add window shows which files will be added to the CVS repository. It is important to add only the files that need to be under source code control. They also need to have the correct file format (text/binary) set in order for the CVS function to work correctly.



Confirm the files to add.

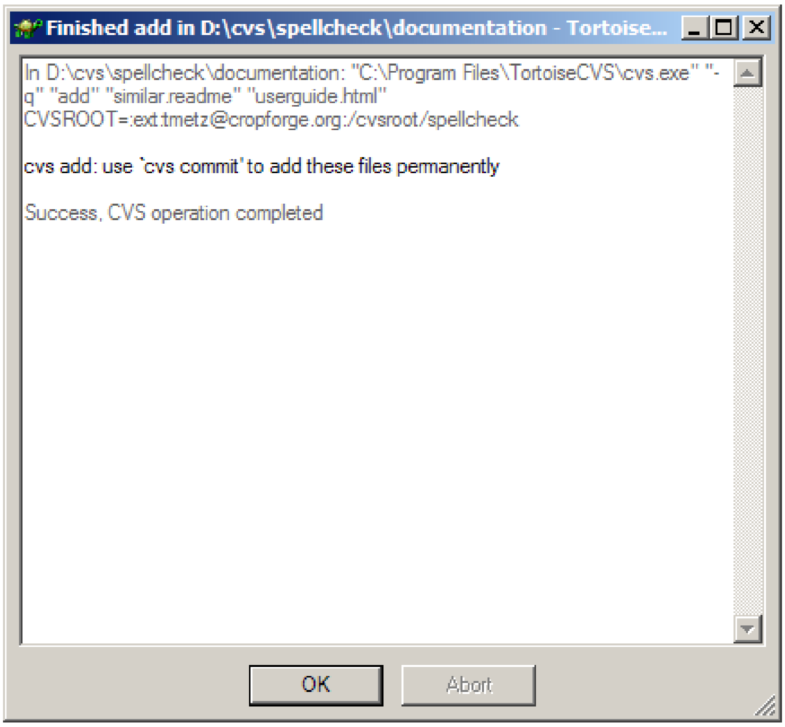
Hint: Right-click to change file format.

□ Show items in subfolders

D:\cvs\spellcheck\dc

	Filename	Type	Add	Format	Keywords
☑	similar...	README File	Yes	Text/ASCII	Normal (kv)
☑	userg...	HTML File	Yes	Text/ASCII	Normal (kv)

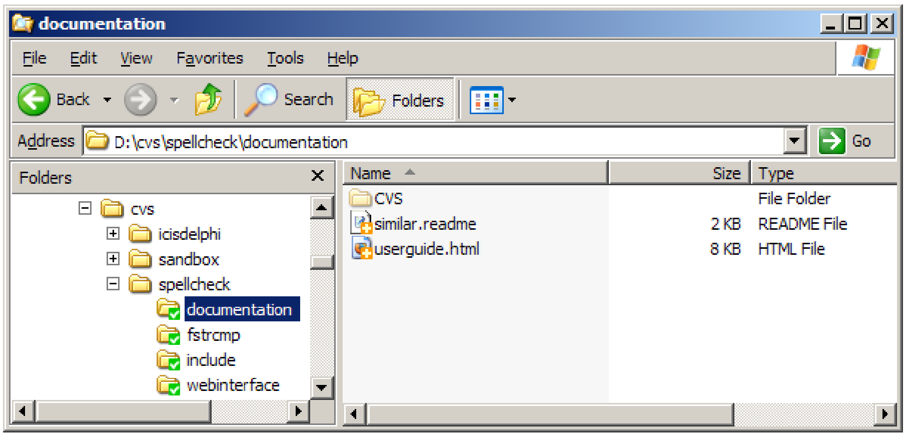
After clicking the **OK** button, the following window should appear and show the dialog between the TortoiseCVS client and the CropForge CVS server.



Finished add in D:\cvs\spellcheck\documentation - Tortoise...

```
In D:\cvs\spellcheck\documentation: "C:\Program Files\TortoiseCVS\cvs.exe" "-
q" "add" "similar.readme" "userguide.html"
CVSROOT=:ext:tmetz@cropforge.org:/cvsroot/spellcheck

cvs add: use `cvs commit' to add these files permanently

Success, CVS operation completed
```

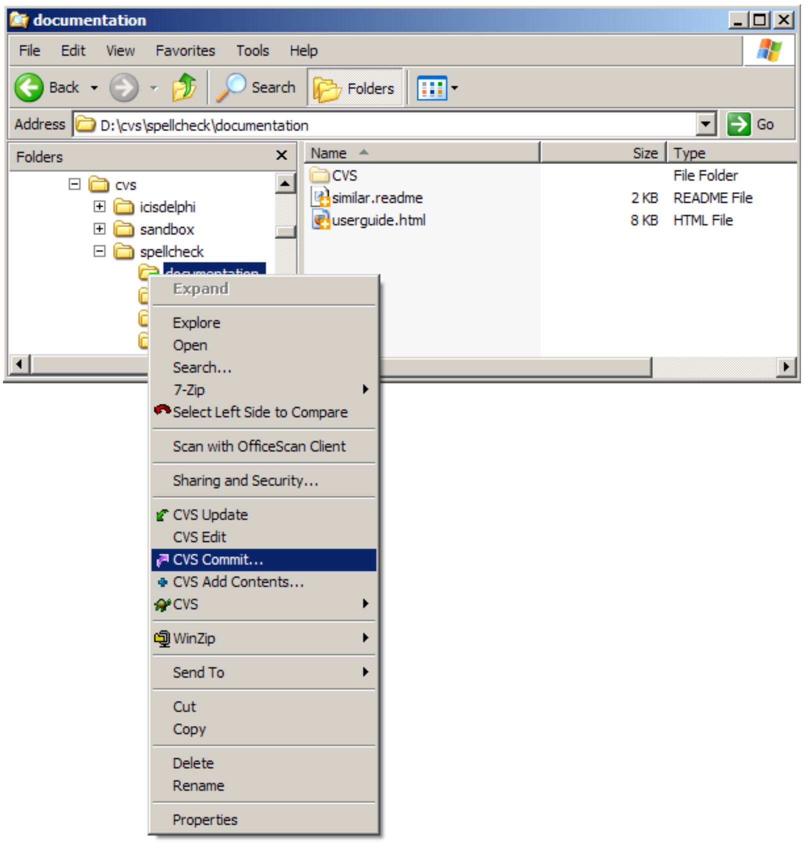
The Explorer window below shows that the files have now been added to the local CVS directory. Note the change in the file icons from a blue question mark to an orange plus.



Name	Size	Type
CVS		File Folder
similar.readme	2 KB	README File
userguide.html	8 KB	HTML File

4.5. Committing content to the CropForge CVS repository

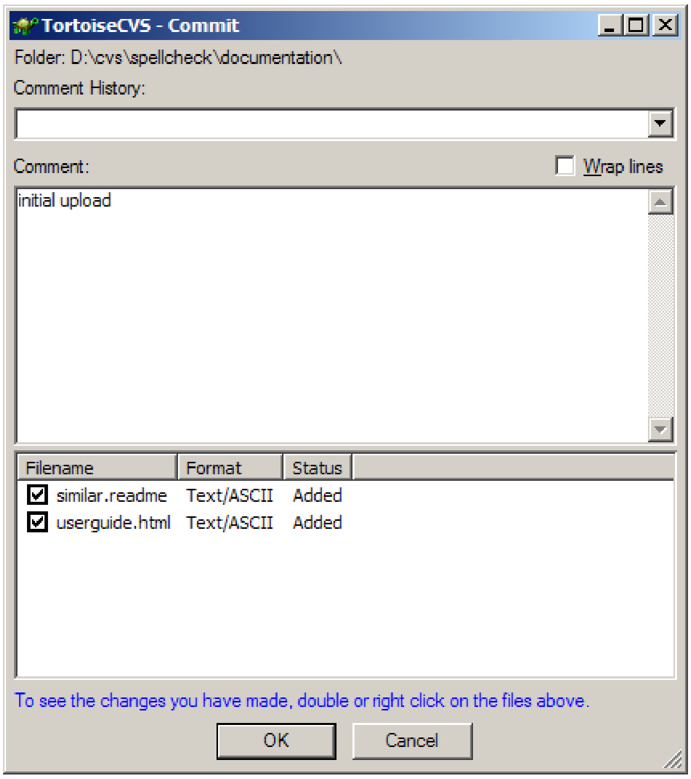
The last step is to add (commit) the content of the new module directory (**documentation**) to the CVS source code repository on CropForge. Right-click on the directory name and select the option **CVS Commit…**



Address D:\cvs\spellcheck\documentation

Name	Size	Type
CVS		File Folder
similar.readme	2 KB	README File
userguide.html	8 KB	HTML File

The following window allows the user to check which files will be added to the CVS repository on the CropForge server. A comment can be added to the commit process, in this example indicating an initial upload.



Folder: D:\cvs\spellcheck\documentation\

Comment History:

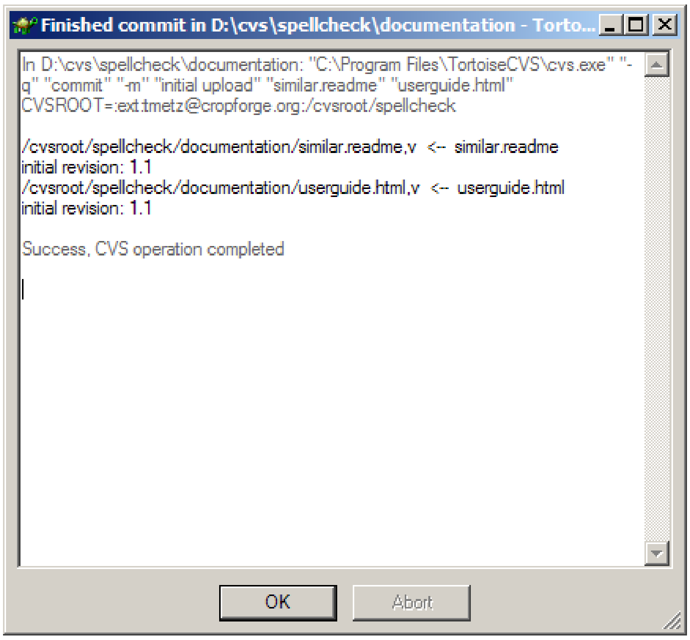
Comment: □ Wrap lines

initial upload

Filename	Format	Status
☑ similar.readme	Text/ASCII	Added
☑ userguide.html	Text/ASCII	Added

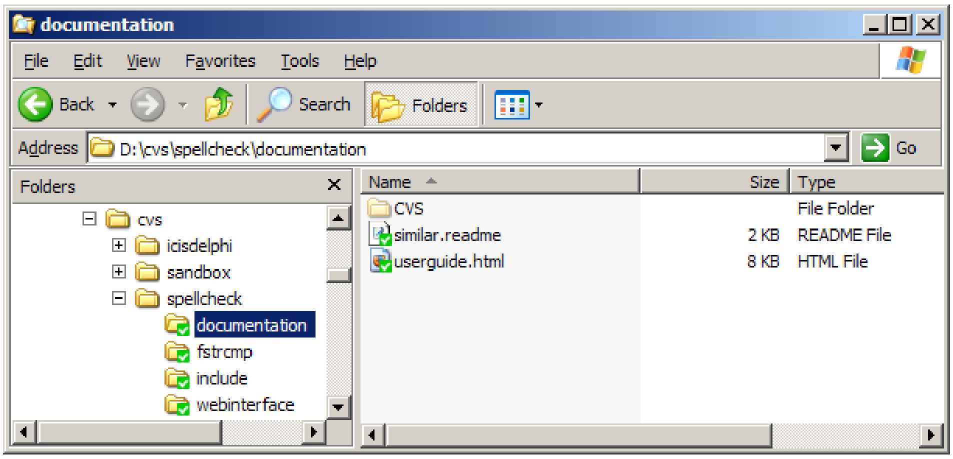
To see the changes you have made, double or right click on the files above.

After clicking the **OK** button, the following window should appear and show the dialog between the TortoiseCVS client and the CropForge CVS server.


Finished commit in D:\cvs\spellcheck\documentation - Torto...
OK
Abort

```
In D:\cvs\spellcheck\documentation: "C:\Program Files\TortoiseCVS\cvs.exe" "-
q" "commit" "-m" "initial upload" "similar.readme" "userguide.html"
CVSROOT=:ext:tmetz@cropforge.org:/cvsroot/spellcheck

/cvsroot/spellcheck/documentation/similar.readme,v  <--  similar.readme
initial revision: 1.1
/cvsroot/spellcheck/documentation/userguide.html,v  <--  userguide.html
initial revision: 1.1

Success, CVS operation completed
```

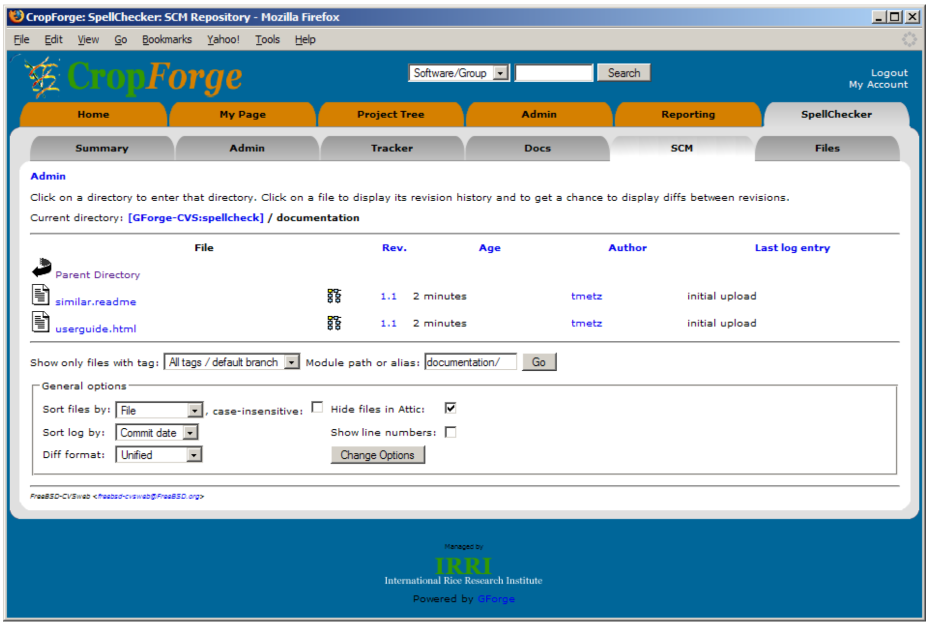
The Explorer window below shows that the files have now been added to the CVS directory on the CropForge server. Note the change in the file icons from an orange plus to a green check/tick.



Name	Size	Type
CVS		File Folder
similar.readme	2 KB	README File
userguide.html	8 KB	HTML File

4.6. Inspecting you new module copy on CropForge

You can now inspect the content of the new module in the CVS repository on the CropForge server. There are columns for file name, revision number, age (time since upload), author of the upload, as well as the comment attached to the upload process.



Click on a directory to enter that directory. Click on a file to display its revision history and to get a chance to display diffs between revisions.

Current directory: [GForge-CVS:spellcheck] / documentation

File	Rev.	Age	Author	Last log entry
Parent Directory				
similar.readme	1.1	2 minutes	tmetz	initial upload
userguide.html	1.1	2 minutes	tmetz	initial upload

Show only files with tag: All tags / default branch Module path or alias: documentation/ Go

General options

Sort files by: File , case-insensitive: □ Hide files in Attic: ☑

Sort log by: Commit date Show line numbers: □

Diff format: Unified Change Options

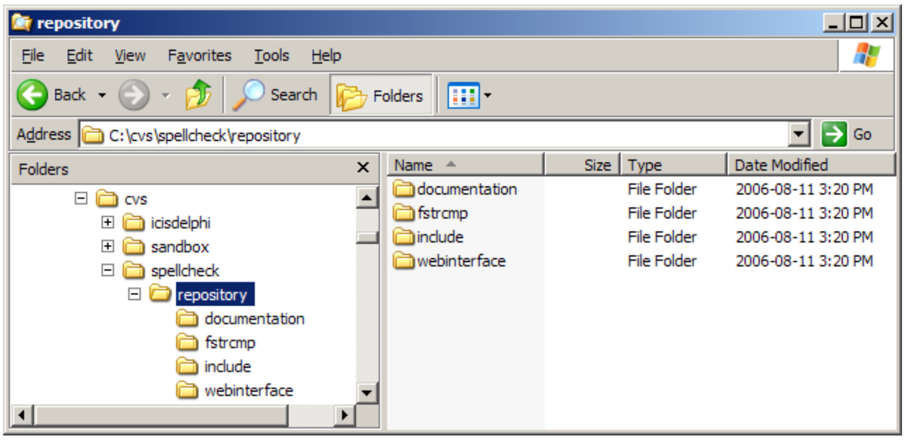
FreeBSD-CVSweb <freebsd-cvsweb@FreeBSD.org>

4.7. Alternative module structure for a project

Instead of splitting a project into separate modules that need to be handled independently by the CVS client and the CVS server, users may want to have only a single module for an entire project. In this case, an additional directory level is needed.

The Explorer window below shows such a structure. In this case, there would be only single module (repository) in the project **spellcheck**, which could be created, and all the directories and files within it could be added and committed, each in a single step.

For a project with components that are not tightly connected (e.g., a back-end and a user front-end), coded in different programming languages, or developed by different developers, a structure with separate modules may be preferable to a structure with a single module, as shown below.



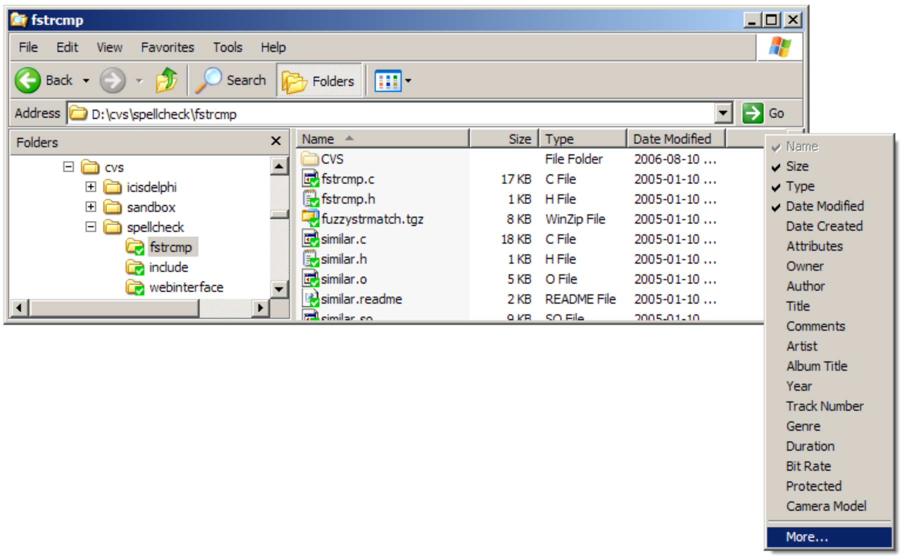
Address C:\cvs\spellcheck\repository

```
cvs
  icisdelphi
  sandbox
  spellcheck
    repository
      documentation
      fstrcmp
      include
      webinterface
```

Name	Size	Type	Date Modified
documentation		File Folder	2006-08-11 3:20 PM
fstrcmp		File Folder	2006-08-11 3:20 PM
include		File Folder	2006-08-11 3:20 PM
webinterface		File Folder	2006-08-11 3:20 PM

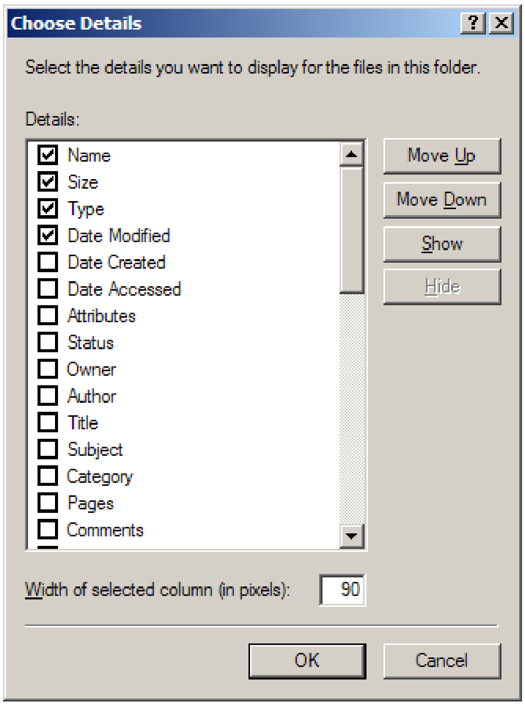
5. Displaying CVS file attributes in the Explorer

Windows Explorer can be configured to display the CVS file attributes, in addition to the normal attributes shown. In a module directory that is under CVS control, right-click on the file attributes title bar. A list of currently activated and additionally available attributes is shown.



Name	Size	Type	Date Modified
CVS		File Folder	2006-08-10 ...
fstrcmp.c	17 KB	C File	2005-01-10 ...
fstrcmp.h	1 KB	H File	2005-01-10 ...
fuzzystrmatch.tgz	8 KB	WinZip File	2005-01-10 ...
similar.c	18 KB	C File	2005-01-10 ...
similar.h	1 KB	H File	2005-01-10 ...
similar.o	5 KB	O File	2005-01-10 ...
similar.readme	2 KB	README File	2005-01-10 ...
similar.so	9 KB	SO File	2005-01-10

The **More…** option leads to the dialog where file attributes can be selected and ordered. Select the four CVS-specific file attributes in addition to the currently selected file attributes. Click the **OK** button.

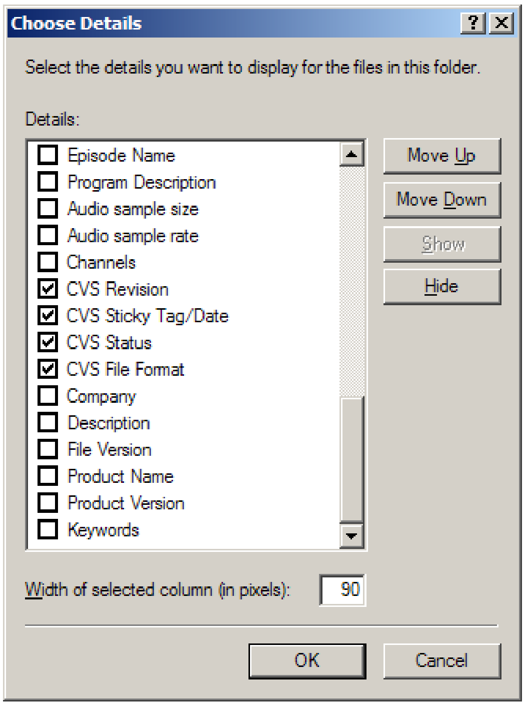


Select the details you want to display for the files in this folder.

Details:

☑ Name
☑ Size
☑ Type
☑ Date Modified
☐ Date Created
☐ Date Accessed
☐ Attributes
☐ Status
☐ Owner
☐ Author
☐ Title
☐ Subject
☐ Category
☐ Pages
☐ Comments

Width of selected column (in pixels): 90



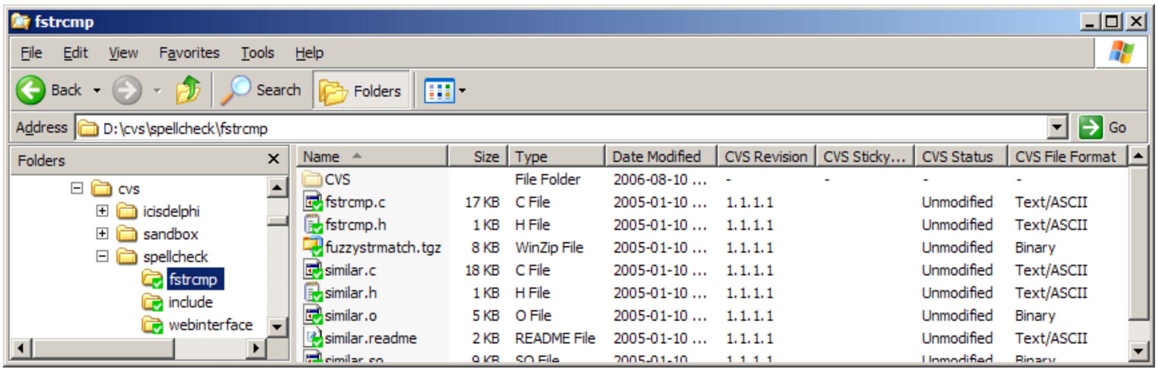
Select the details you want to display for the files in this folder.

Details:

☐ Episode Name
☐ Program Description
☐ Audio sample size
☐ Audio sample rate
☐ Channels
☑ CVS Revision
☑ CVS Sticky Tag/Date
☑ CVS Status
☑ CVS File Format
☐ Company
☐ Description
☐ File Version
☐ Product Name
☐ Product Version
☐ Keywords

Width of selected column (in pixels): 90

Windows Explorer should now show the CVS file attributes in addition to the default file attributes. These additional CVS file attributes are very helpful in maintaining your local source code version. The above procedure needs to be done for every module subdirectory.



Name	Size	Type	Date Modified	CVS Revision	CVS Sticky…	CVS Status	CVS File Format
CVS		File Folder	2006-08-10 …	-	-	-	-
fstrcmp.c	17 KB	C File	2005-01-10 …	1.1.1.1		Unmodified	Text/ASCII
fstrcmp.h	1 KB	H File	2005-01-10 …	1.1.1.1		Unmodified	Text/ASCII
fuzzystrmatch.tgz	8 KB	WinZip File	2005-01-10 …	1.1.1.1		Unmodified	Binary
similar.c	18 KB	C File	2005-01-10 …	1.1.1.1		Unmodified	Text/ASCII
similar.h	1 KB	H File	2005-01-10 …	1.1.1.1		Unmodified	Text/ASCII
similar.o	5 KB	O File	2005-01-10 …	1.1.1.1		Unmodified	Binary
similar.readme	2 KB	README File	2005-01-10 …	1.1.1.1		Unmodified	Text/ASCII
similar.so	9 KB	SO File	2005-01-10	1.1.1.1		Unmodified	Binary

6. Resources

- CropForge server: http://cropforge.org
- TortoiseCVS software: http://www.tortoisecvs.org
- TortoiseCVS help: Start -> Programs -> TortoiseCVS -> Help
- PuTTY software: http://www.chiark.greenend.org.uk/~sgtatham/putty/
- CVS book: http://cvsbook.red-bean.com
- CVS manual: http://ximbiot.com/cvs/manual/