



SHIFTS Downloading and Installation Instructions

Trevor Fulton <trevor.fulton@blueskyinc.ca>

SPIRE-BSS-DOC-XXXXXX

07 April 2008

SHIFTS Downloading and Installation Instructions

Trevor Fulton

Published Version 2.0, 01 April 2008

Table of Contents

1. Introduction	1
1.1. Acronyms	1
1.2. Scope of Document	1
1.3. Documents	1
1.3.1. Applicable Documents	1
1.3.2. Reference Documents	1
1.4. Document History	2
2. CVS Instructions	3
3. SHIFTS Installation	8
4. Updating the SHIFTS Software Package	12
A. Appendix	14
A.1. SHIFTS user group	14
A.2. IDL Header Example	14
A.3. Modifying Windows© Environment Variables	15

Chapter 1. Introduction

1.1. Acronyms

Table 1.1. Acronyms

Short Form	Full Name
AIG	Astronomical Instrumentation Group at the University of Lethbridge
CVS	Concurrent Versions System
GUI	Graphical User Interface
HSCDT	Herschel Science Center Development Team
HSO	Herschel Space Observatory
IDL	Interactive Data Language™
SHIFTS	Simulator for the Herschel Imaging Fourier Transform Spectrometer
SPIRE	Spectral and Photometric Imaging REceiver
TBD	To Be Determined
TBW	To Be Written

1.2. Scope of Document

This purpose of this document is to present downloading and installation instructions for the SHIFTS software package. In addition, instructions on how to contribute to the SHIFTS software package will be presented.

1.3. Documents

1.3.1. Applicable Documents

- AD01 J. V. Lindner, SHIFTS: Simulator for the Herschel Imaging Fourier Transform Spectrometer, Master of Science Thesis, University of Lethbridge, 02 February 2006.
- AD02 J. V. Lindner, SPIRE Architectural Design Document for the Simulator for the Herschel Imaging Fourier Transform Spectrometer, SPIRE-UOL-DOC-xxxxxx, Version 0.1, 23 May 2006

1.3.2. Reference Documents

- RD01 A Developer's Manual -- Herschel Data Processing, HERSCHEL-HSC-DOC-0625, Version 3.1.1278, 29 May 2007
- RD02 SHIFTS Test Module, A. Katz, 11 June 2007.
- RD02 <http://research.uleth.ca/spire>
NEED DOC NAME AND NUMBER

1.4. Document History

Table 1.2. Version and Date

Issue	Date	Changes
Version 1.0	17 July 2007	First Issue
Version 2.0	16 April 2008	<ul style="list-style-type: none">• Fixed some minor typos.• Modified the updating instructions in Chapter 4.• Added a verification step to the installation process in Chapter 3.• Changed Rob Zondag <rzondag@rssd.esa.int> to Daniel Galan <dgalan@sciops.esa.int> as Herschel CVS administrator in Chapter 2.

Chapter 2. CVS Instructions

This section describes where to find the SHIFTS software package and includes instructions on how to retrieve the package.

The source code, calibration information, and documentation relating to the SHIFTS simulator can be found on the HSCDT CVS repository.

Users wishing to access and/or contribute to the SHIFTS software package will need to set up an account on the Herschel CVS server. Please contact the Herschel CVS custodian Daniel Galan <dgalan@sciops.esa.int> (and please cc: Steve Guest <s.guest@rl.ac.uk>) for further details.

Once a CVS account is established, the next step is to connect to the HSCDT CVS repository. The following set of instructions apply to Windows© users; Linux users may consult Section 2.3 of RD01.

1. **Download and install a CVS client program.** Visit http://ximbiot.com/cvs/wiki/index.php?title=CVS_Clients#Windows for a list Windows©-compatible CVS GUI clients. This remainder of this section will include examples from the WinCvs (<http://www.wincvs.org/>) client.
2. **Set the CVSROOT environment variable**¹. For example, the user "tfulton", would set the CVSROOT environment variable to the the following:

```
:pserver:tfulton@cvs.rssd.esa.int:/local/repositories/HERSCHEL_CVS
```

3. **Define a local root folder for the SHIFTS package.** This is the folder into which the SHIFTS software package will be written. Please note that this folder must be accessible by both the CVS and IDL programs. For the purposes of this document, the folder C:\SPIRE\CVS\ will be used as an example root folder.
4. **Login to the CVS repository.** In WinCvs, one can login to a repository by selecting "Admin→Login..." as shown in Figure 2.1.

¹ See Section A.3 for instructions on how to modify/create Windows© environment variables.

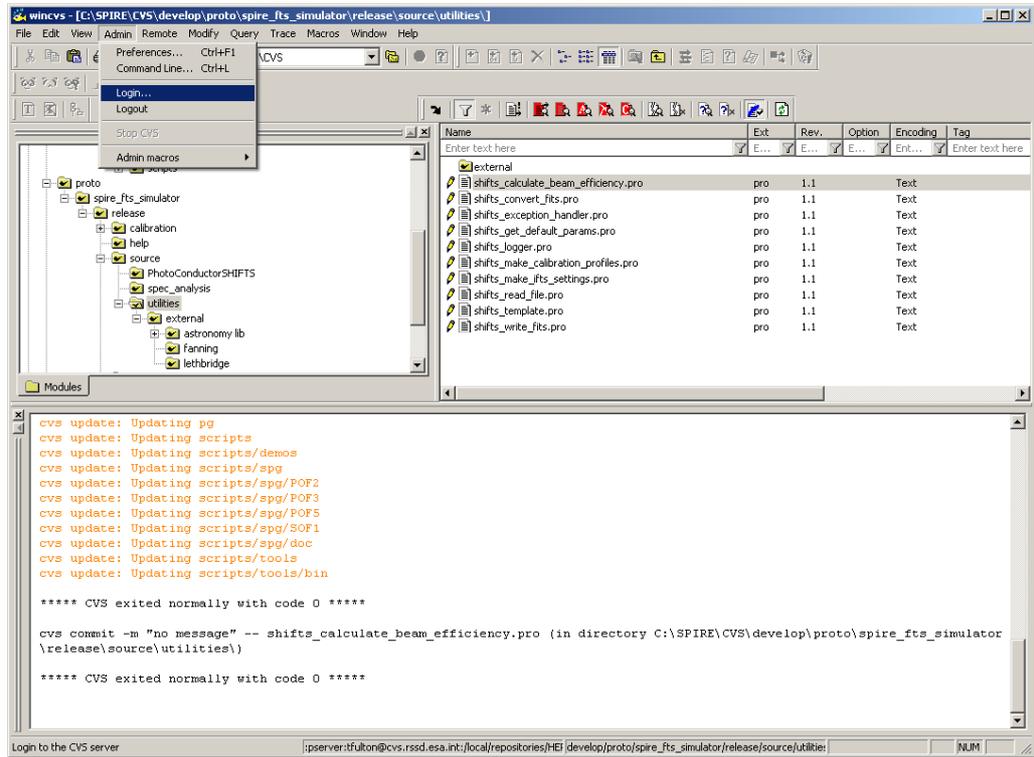


Figure 2.1. Using WinCvs to login to a repository

This will cause the "Login settings" window to appear (Figure 2.1).

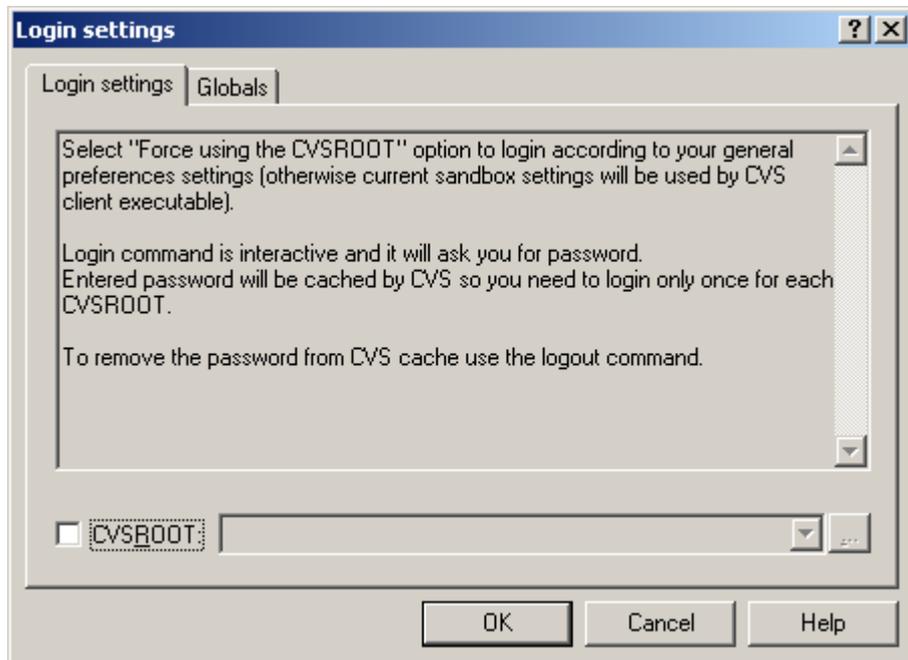


Figure 2.2. Example of WinCvs Login settings

Deselect the CVSROOT checkbox (so as to use the value written in the CVSROOT environment variable) and select "OK". Then type in the appropriate password at the prompt in order to complete the login process.

5. **Checkout the SHIFTS software package.** This is the step where the SHIFTS simulator software is actually downloaded from the server to the host. The SHIFTS software package may be downloaded in two different ways, depending on whether the user is intending to simply use the simulator or whether the user intends to contribute to the SHIFTS package.
 - a. **Developers.** For SHIFTS developers, it may be best to checkout the latest version of each SHIFTS software module. This may be accomplished with the command

```
cvsc checkout -P -- develop/proto/spire_fts_simulator/release/
```

In WinCvs, one first needs to select "Remote→Checkout module..." (Figure 2.3)

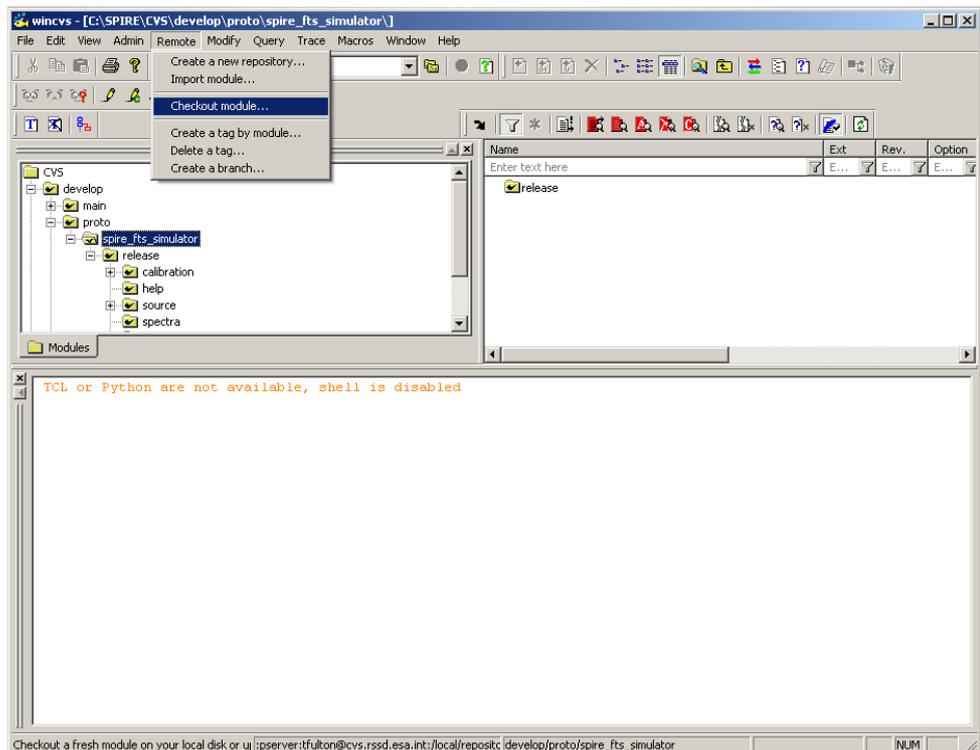
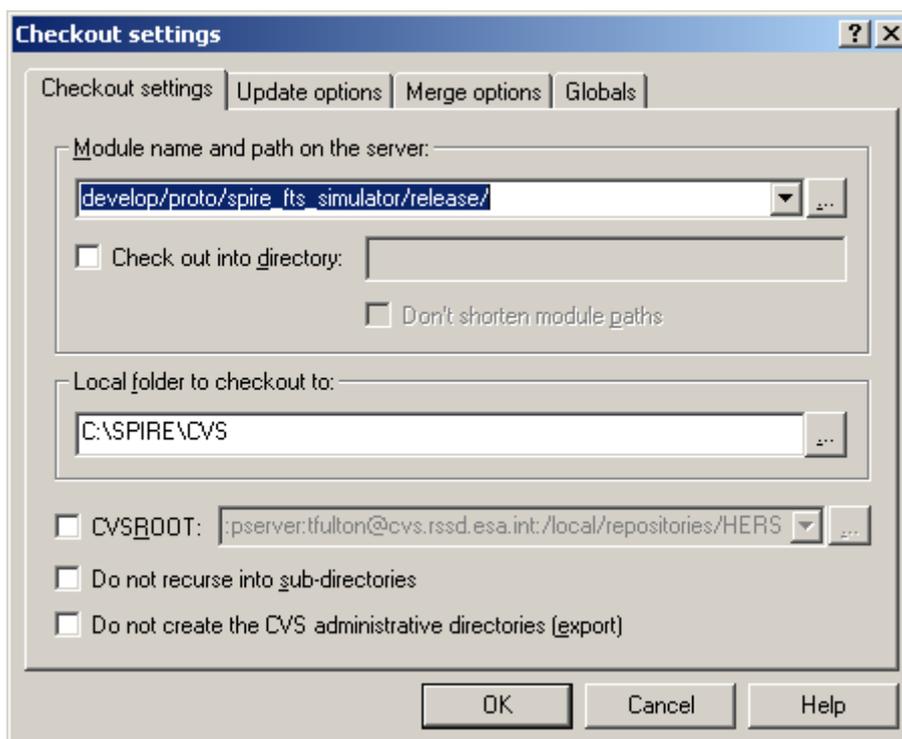


Figure 2.3. Using WinCvs to checkout the SHIFTS software package.

This will bring up the "Checkout settings" window as shown in Figure 2.4.



Please note the item in the Local folder to checkout to: field. This folder will be the root folder on the local system for the SHIFTS software package.

Figure 2.4. CVS checkout settings.

Developers should set the fields of this window to the following values:

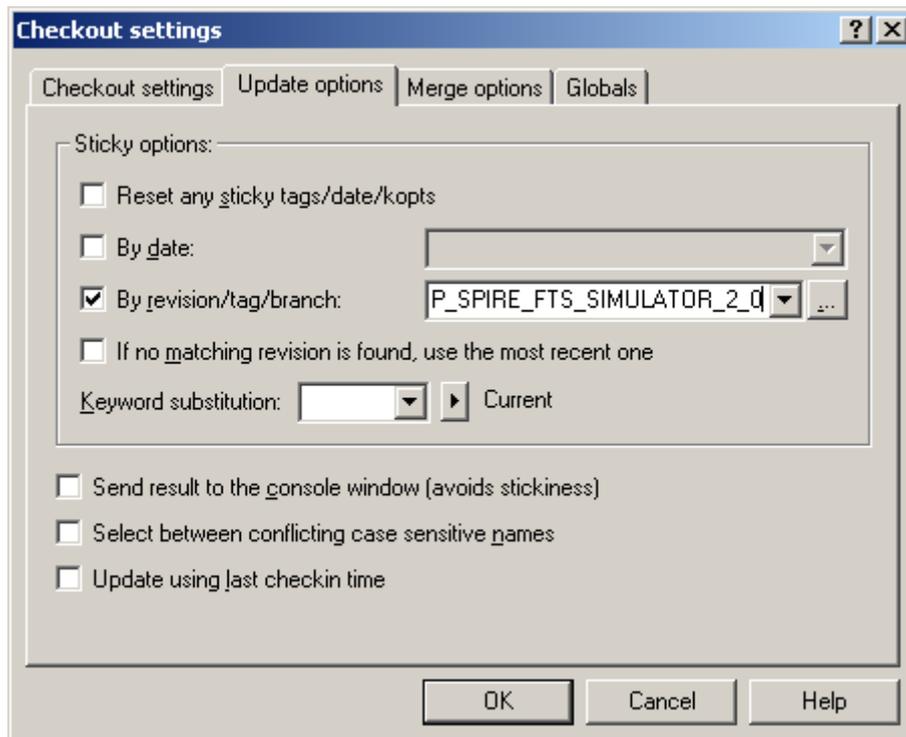
- Set "Module name and path on the server:" to `develop/proto/spire_fts_simulator/release/`
- Set "Local folder to checkout to:" to the root folder chosen above.
- Ensure that the CVSROOT checkbox is not selected (CVSROOT will then be determined from the system environment variable).

- b. **General Users.** For general SHIFTS users, it may be best to download the latest release version of the SHIFTS software package. This may be accomplished by checking out the latest release tag version of SHIFTS. The command to accomplish this is:

```
cvs checkout -P -r P_SPIRE_FTS_SIMULATOR_2_0 --
develop/proto/spire_fts_simulator/release/
```

where the argument "P_SPIRE_FTS_SIMULATOR_2_0" may be replaced with the whichever release version tag is desired.

In WinCvs, one should follow the same steps as listed above for developers but should also modify the "Checkout settings" tab as shown in (Figure 2.5).



When using the checkout by tag option, the Checkout settings should be the same as above.

Figure 2.5. CVS checkout by tag name.

Chapter 3. SHIFTS Installation

The following steps are required to install the SHIFTS software package:

1. **Retrieve the SHIFTS software from the CVS repository.** See Chapter 2 for instructions.
2. **Add the SHIFTS source directory to the IDL path.** An entry is added to the IDL search path by selecting File→Preferences which brings up a window similar to that shown in Figure 3.1

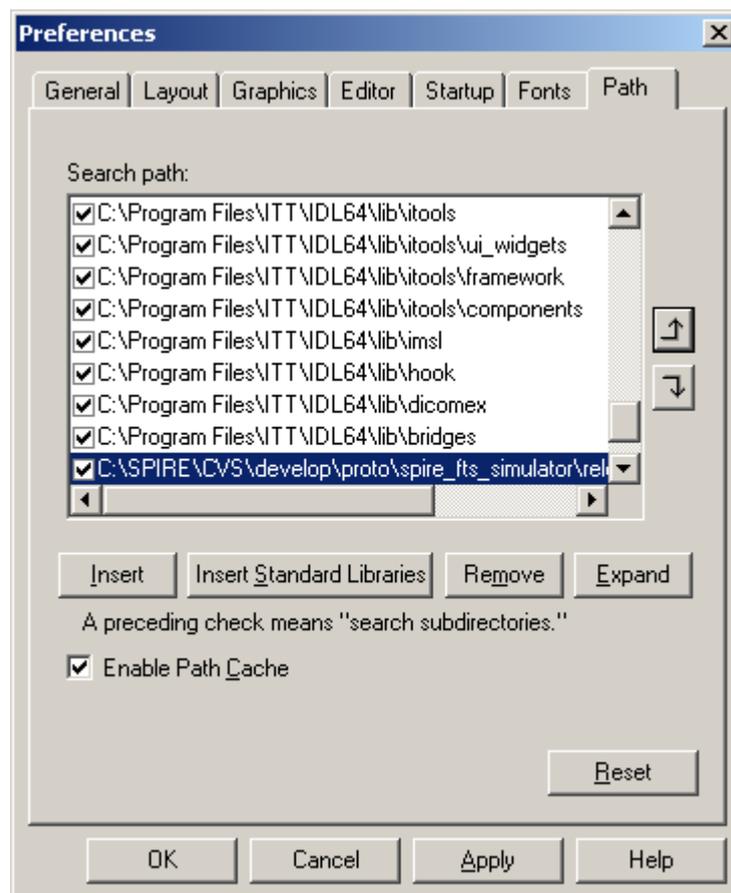


Figure 3.1. Modifying the IDL search path.

A path may be added to the list by selecting "Insert" and then navigating to the desired folder. Finally, check the box to the left of the newly added path to ensure that subfolders of that folder will also be searched.

It should be noted that the SHIFTS package on the CVS repository contains some IDL source code that comes from external libraries. These are procedures and functions that are widely available and that SHIFTS requires to execute properly.

The layout of the software within the SHIFTS package is as shown in Figure 3.2.

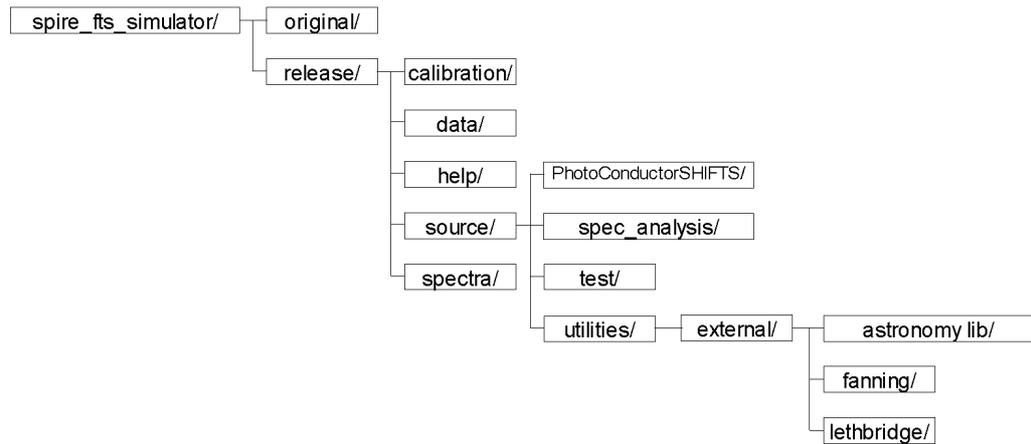


Figure 3.2. Layout of the SHIFTS software package

The external IDL packages are located within the SHIFTS source tree in the `utilities/external/` folder. Below this folder are three subfolders:

- `astronomy lib` The IDL Astronomy User's Library (<http://idlastro.gsfc.nasa.gov/>).
- `fanning` Selected IDL programs from David Fanning (<http://www.dfanning.com/>).
- `lethbridge` Selected IDL programs developed at Blue Sky Spectroscopy (<http://www.blueskyinc.ca/>) and in the AIG (<http://www.uleth.ca/phy/naylor/>).

If the user's local system already contains these packages then they may be deleted from the local copy of SHIFTS.

3. **Verify the installation of SHIFTS** The master test control program, `shifts_master_test_control.pro`, may be used to verify that the installation of SHIFTS was a success (see Figure 3.3). This program will test most SHIFTS modules for complete execution and physically accurate results. An overview of the specific testing process is presented in [RD02].

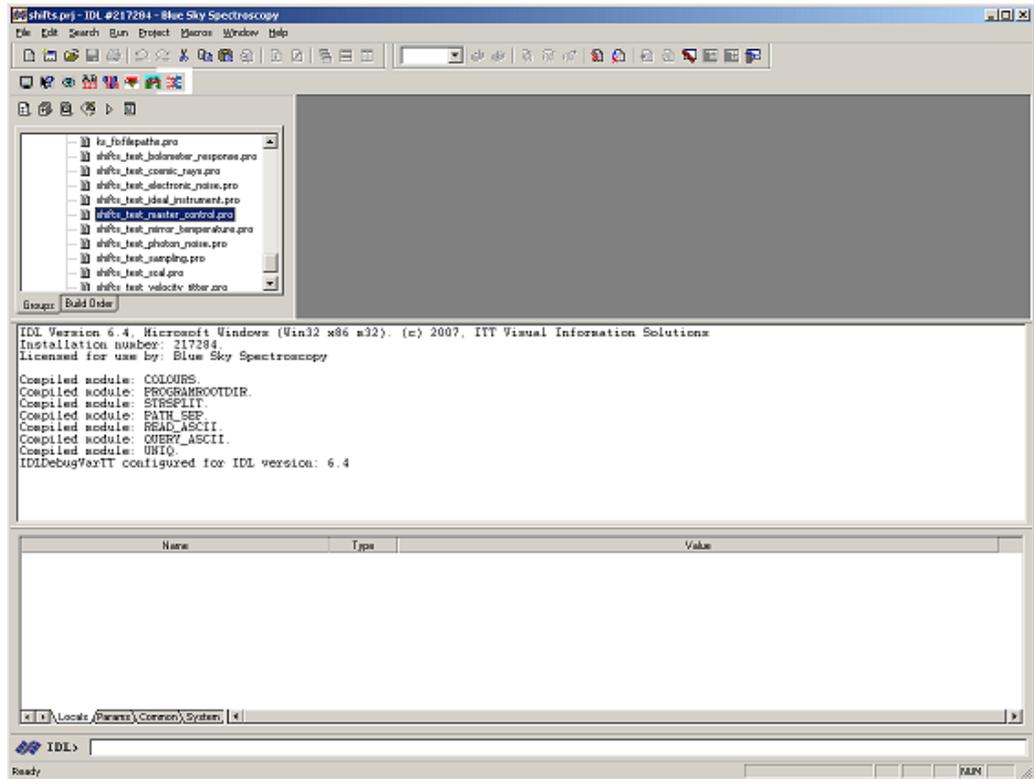


Figure 3.3. The SHIFTS master test module

If all tests pass successfully, then the message ALL TESTS HAVE PASSED SUCCESSFULLY. will appear in the IDL console, compare Figure 3.4.

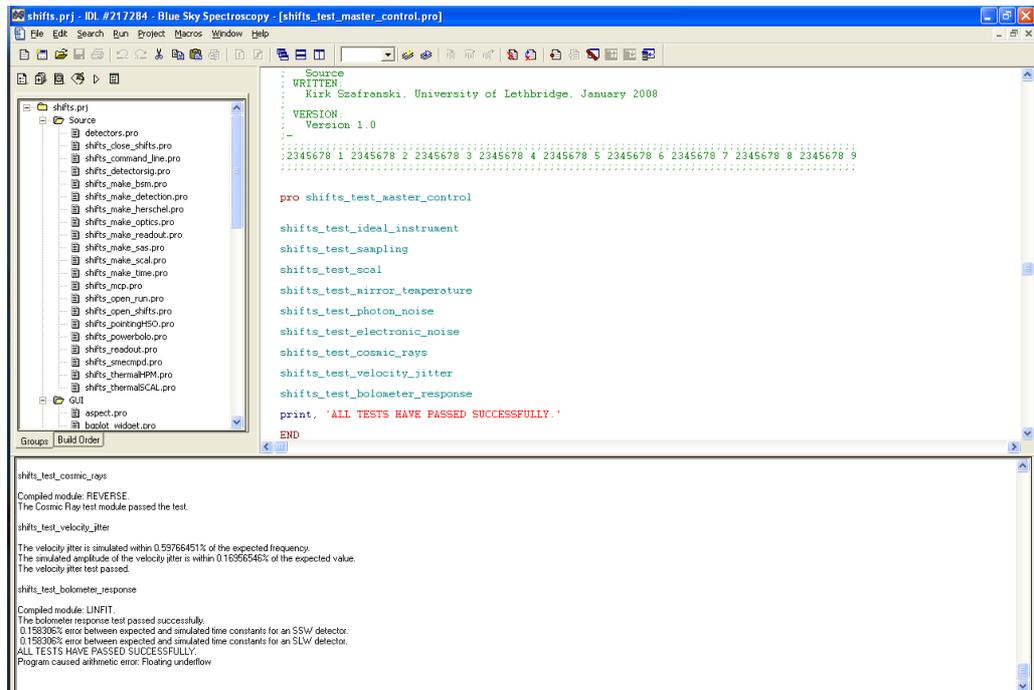


Figure 3.4. A successful pass of the SHIFTS test module

If the test fails, SHIFTS will stop execution, display an error message, and take the user to the point in the module that had a problem. If there is an error with SHIFTS itself then the `shifts_exception_handler` will display an internalized error code as is typical for any SHIFTS internal error [REF??].

Please report the specific behavior and error message when contacting support on SHIFTS.



Note

The test modules will create data products in the data folder of SHIFTS.

Chapter 4. Updating the SHIFTS Software Package

This section describes the procedures that should be followed when making modifications or additions to the SHIFTS software package.

1. **Add comments to IDL code.** Each new IDL source file added to the SHIFTS package should contain the standard SHIFTS header, an example of which is shown in Figure A.1.

When modifying existing source files within the SHIFTS package, please include a message related to the modification in the "MODIFICATION HISTORY" section of the standard header.

2. **Resolve CVS conflicts.** A conflict may arise using CVS when two or more developers are modifying the same file at the same time. In such a situation, the conflicts must be resolved before the modifications can be committed to the CVS repository.
3. **Test the modified software package.** The full suite of test harnesses should be executed (see Chapter 3) to ensure that any modifications do not cause errors in SHIFTS as a whole.



Important

It is advised that any modules that are to be added to SHIFTS should contain automated unit-level test harnesses. Please refer to RD02 for a guide on the design of a SHIFTS unit-level test harness and how to add new test harnesses to the testing suite.

4. **Commit the modifications to the CVS repository.** This is the step where the changes made by the developer actually get written to the CVS repository.



Note

In the case of file modification, the older version of the modified file still exists on the repository and may be retrieved if desired.

5. **Tag the package.** The tags for the SHIFTS software package fall into two categories:

Release Tags These are tags that should be used to denote a release version of the SHIFTS simulator. The format of these tags should be `P_SPIRE_FTS_SIMULATOR_AA_BB` where AA denotes the major version number and BB denotes the minor version number.

Contributors that use the version tag should send an notification to SHIFTS user group (Section A.1).

Development Tags These are tags that contributors may use in order to mark a stage of their own development of the SHIFTS software. These tags should be of the form `UPDATE_SPIRE_FTS_SIMULATOR_AA_BB` where AA and BB are defined in the same manner as above.

For example, if the current release tag is `P_SPIRE_FTS_SIMULATOR_2_1` then the next development tag should be `UPDATE_SPIRE_FTS_SIMULATOR_2_2`. When the development code is ready for release, the next release tag would be `P_SPIRE_FTS_SIMULATOR_2_3`.

Contributors that use the version tag should send an notification to SHIFTS user group (Section A.1).

Warning



Contributors should not use tags that begin with D_. Tags of this nature are reserved in the HSCDT CVS system.

6. **Modify the CHANGELOG file.** Modifications to the CHANGELOG file should be related with the tag action. The CHANGELOG file should not be modified unless a new tag is added. Moreover, the CHANGELOG file should be tagged along with the rest of the package.

The CHANGELOG file should contain comments relating to the additions and/or modifications to the package. The following is a generic example of a CHANGELOG entry:

```
TAG_ASSOCIATED_TO_THIS_CHANGES (yyyy-mm-dd) [user_initials]
- Added/Modified ... [comments]
- Added/Modified ... [comments]
- Files affected:
  * Added:
    file1
    file2
    ...
  * Modified:
    file1
    file2
    ...
```



Note

CVS has a feature that may be use to get a list of the files that have been modified or added.

```
cvs -n -q update -I ! -I CVS
```

('n' do nothing, '-q' for quiet, 'update' query update, '-I' for ignore)

The newest entries should always be added to the top of the CHANGELOG file.

If issues arise that are related to updating the SHIFTS software that are not covered in this section, please contact Trevor Fulton <trevor.fulton@blueskyinc.ca>.

Appendix A. Appendix

A.1. SHIFTS user group

The entries in the table below are the members of the SHIFTS user group. Please notify all of the members of this group whenever any modifications are made to the SHIFTS software package. In addition, please address any problem reports or change requests relating to SHIFTS to the members of this list.

Table A.1. SHIFTS User Group.

Name	Email Address
Trevor Fulton	trevor.fulton@blueskyinc.ca
Peter Davis	peter.davis@blueskyinc.ca
Edward Polehampton	E.T.Polehampton@rl.ac.uk
Bruce Sibthorpe	sib@roe.ac.uk
Giorgio Savini	giorgio.savini@astro.cf.ac.uk

A.2. IDL Header Example

Figure A.1 shows an example of the standard header that should be located at the top of each IDL source file that is part of the SHIFTS package.

```
;/+
;/
;/ $Id: shifts_close_shifts.pro,v 1.1 2007/07/13 20:44:29 tfulton Exp $
;/ $Date: 2007/07/13 20:44:29 $
;/ $Author: tfulton $
;/ $Revision: 1.1 $
;/
;/ NAME:
;/ shifts_close_shifts
;/ PURPOSE:
;/ Closes the execution of SHIFTS
;/ EXPLANATION:
;/ If the user sets the logging output to print to file, the log file
;/     is closed in the run directory. If the logging level is debugFine
;/     or greater, the output of the IDL profiler is written to a file.
;/ USE:
;/ shifts_close_shifts
;/ INPUTS:
;/
;/ OPTIONAL INPUTS:
;/
;/ OUTPUTS:
;/
;/ OPTIONAL OUTPUTS:
;/
;/ KEYWORDS:
;/
;/ CALLS:
;/ shifts_logger
;/ shifts_exception_handler
;/ ERROR HANDLING:
;/ -IDL catch
;/ COMMON:
;/ common1: hk
;/ RESTRICTIONS:
;/
;/ KNOWN SIDE EFFECTS:
;/
;/ CATEGORY
;/ Source
;/ WRITTEN:
;/     John Lindner, University of Lethbridge (May 2005)
```


Figure A.2. Generic Windows© System Properties Window.

2. Select Advanced→Environment Variables. This will bring up a window similar to that shown in Figure A.3:

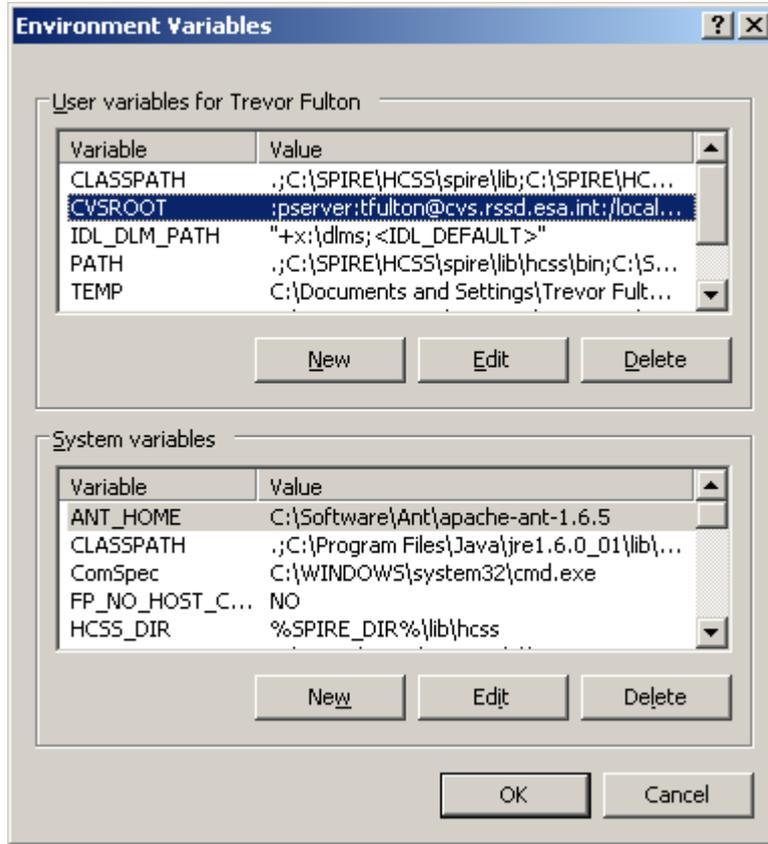


Figure A.3. Generic Windows© Environment Variables Window.

3. To add a new user or system environment variable, select one of the "New" buttons. This will bring up a window similar to that shown in Figure A.4:



Figure A.4. Example of how to add a new system variable.

The fields of this window may be modified as desired.

4. To modify an existing user or system environment variable, select one of the "Edit" buttons. This will bring up a window similar to that shown in Figure A.5

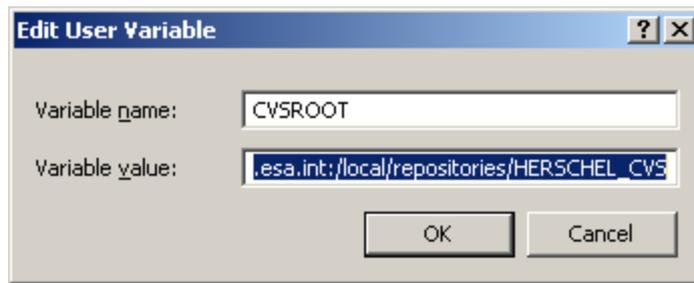


Figure A.5. Example of how to modify an existing system variable.

The fields of this window may be modified as desired.