How to change code...

1 Introduction

This document describes the necessary steps to change code from the model MOHID. The source code of MOHID is stored in a data base located in the Server *Einstein*. If you are about to change code from MOHID always:

- keep in mind that you are not the only one who works with MOHID
- don't keep a file checked-out for a longer time
- keep in mind that the model should be easily usable

2 Changing Code

To change on or more source files from the MOHID model you should follow the steps described below.

2.1 Check-out source code

To check-out source code you should:

- Get the latest version of all the source files
- Check-out the files which you are going to change
- Proceed with the changes of the source code

2.2 Testing

2.2.1 Functional Verification

"Functional verification is the core of software testing. It is validating if the basic functionality of the software is correct. This is testing whether the software does what it is supposed to do. This type of testing can usually be done in a single environment."
This means to check the basics of the new features implemented in MOHID's source code.

2.2.2 Black box testing

Black box testing is testing that is focused on what the software is supposed to do. Thus, in black box testing we are checking to see if the functionality is correct, without concerning ourselves with how the software achieves this.

2.2.3 White box testing

"White (or rather clear) box testing is when the tester uses his knowledge of how the software module or component is implemented. Contrast this with black box testing".
In this type of test is important that all new/changed code lines are passed at least one time.

2.2.4 Regression testing

It is an unfortunate fact of life in software development that when you change one part of the code, ***other parts, for one reason or another, stop working properly***. In regression testing we are simply redoing **old tests** to ensure that nothing that was working has now for some reason **stopped working.**

2.2.5 More information

More information about software testing can be get from:
http://www.aptest.com/resources.html
http://www.priortesting.com/testing_glossary.html

2.3 Check-In code

To check-in the changed code you should follow the following steps.

- Get the latest version of all files which you haven't checked-out
- Select all files from the project, like shown in Figure 2-1.
- Set the compiling diagnostic like shown in Figure 2-1

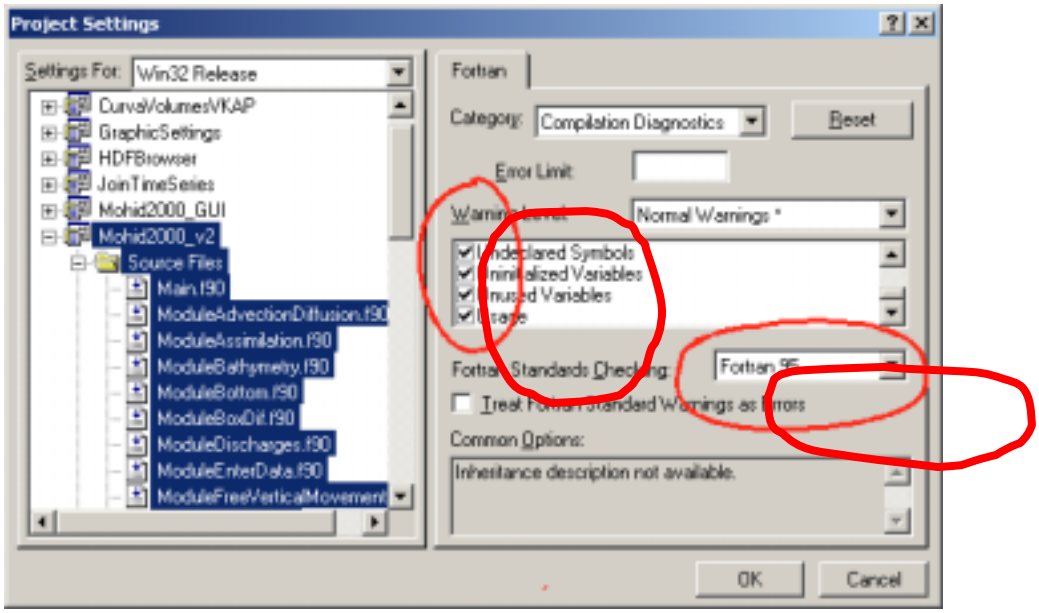

Figure 2-1: Selecting all source files

Rebuild the whole project (Clean, Rebuild All) and remove all warnings in the source files which you are going to check-in.

3 Other considerations

3.1 Subroutine GetData

The subroutine GetData has several optional arguments, among them the so called *ClientModule* and the *Default*.

The optional argument should always be filled with the calling module. This step is fundamental for the future of the graphical user interface.

Example:

```
call GetData(NewProperty%ID%Description,                    &
             AppWaterProperties%AppEnterData1, iflag,       &
             SearchType   = FromBlock,                      &
             keyword      = 'DESCRIPTION',                  &
             ClientModule = 'ModuleWaterProperties',        &
             Default      = 'No description',               &
             STAT         = STAT_CALL)
```

The argument *Default*, case it exists, should always be used. There are many partes in the source code where the default value is filled, using the argument flag.

Example (OK)

```
call GetData(NewProperty%ID%Description,                    &
        AppWaterProperties%AppEnterData1, iflag,            &
        SearchType   = FromBlock,                           &
        keyword      = 'DESCRIPTION',                       &
        ClientModule = 'ModuleWaterProperties',             &
        Default      = 'No description',                    &
        STAT         = STAT_CALL)
```

Example (WRONG)

```
call GetData(NewProperty%ID%Description,                    &
        AppWaterProperties%AppEnterData1, iflag,            &
        SearchType   = FromBlock,                           &
        keyword      = 'DESCRIPTION',                       &
        ClientModule = 'ModuleWaterProperties',             &
        STAT         = STAT_CALL)
…
    if (iflag == 0) NewProperty%ID%Description = "No Description given"
```

3.2 Real numbers

If you use real numbers without decimal fraction, always add a point (.) at the end of the number.

Example (OK)

```
Mean = (a+b) / 2.
```

Example (WRONG)

```
Mean = (a+b) / 2
```