

# SMARTS code, version 2.9.5

## For Mac OSX

### Quick Start Guide



Christian A. Gueymard  
Solar Consulting Services, USA  
December 2005

#### **Installation**

The distribution package is compressed. Decompression (e.g., by double-clicking) will create the SMARTS\_295\_Mac folder.

#### **Important**

Read the **License Agreement** in file 'AAgreement\_LICENSE.txt' and, *if and only if you agree with its terms*, proceed further as described below.

#### **Folder content**

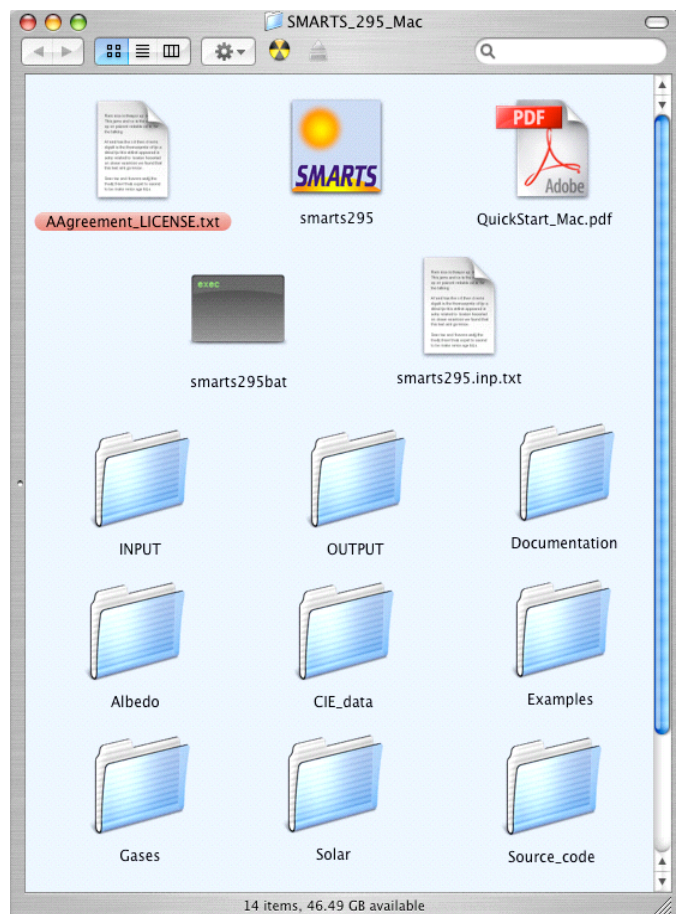
A typical screenshot of the SMARTS\_295\_Mac folder is shown here. The exact appearance depends on your own settings in the Finder Preferences and View options.

The clickable Standard script is the 'smarts295' file with the SMARTS icon.

The Command Line executable is the 'smarts295bat' file with a dark gray icon.

The 'smarts295.inp.txt' file is a sample input file that you can use to familiarize yourself with the two execution modes described below.

The User's Manual can be found in the Documentation folder.



## Normal execution

To run in Standard mode, double-click on ‘smarts295’ (whose icon is based on the SMARTS logo, and whose name may alternatively be listed as ‘smarts295.command’ if the Finder Preferences are set to “Show all file extensions”). This will launch the ‘smarts295\_cmd’ executable in the `Source_code` folder, and will open a single-use Terminal window. A welcome message appears, as follows.

```
*****
Welcome to SMARTS, version 2.9.5!
*****
SMARTS_295> Use standard mode with default input file?
[If YES (or Y), execution will start immediately
using the default input file smarts295.inp.txt]
(Y/N) ==>
```

If you type “y” or “yes” and hit Enter, execution begins immediately. A final message similar to this one is displayed when execution ends:

```
Total CPU time:  0.122 sec
logout
[Process completed]
```

The Terminal window’s name becomes Completed Command, at which point it can be closed.

Execution of the sample input file must have created two separate output files: ‘smarts295.out.txt’ and ‘smarts295.ext.txt’, both located at the root of the `SMARTS_295_Mac` folder.

If you rather type “n” or “no” and hit Enter, the following message appears:

```
$$$ SMARTS_295> What is the path to the input file?
* Type only "." if in the same folder
* Do NOT type the last "/" of the chain
* 2000 characters max. ==>
```

Type the path pointing to your input file or type “.” if the file resides in the current folder. For instance, type ‘INPUT’ if it is located in the `INPUT` folder of `SMARTS_295_Mac`. Hit Enter and the program will ask for the proper filename:

```
$$$ SMARTS_295> Generic name for all input/output files
* without any extension
* 100 characters max.)? ==>
```

For instance, type “Test\_for\_July” if the input file is ‘Test\_for\_July.inp.txt’. Hitting Enter results in:

```
$$$ SMARTS_295> You chose the following filenames:
Input:  INPUT/Test_for_July.inp.txt
Output: INPUT/Test_for_July.out.txt
Spreadsheet-ready: INPUT/Test_for_July.ext.txt
Smoothed results: INPUT/Test_for_July.scn.txt
$$$ SMARTS_295> Is this OK? (Y/N) ==>
```

Type “y” if it is OK, or “n” if not, and hit Enter. If you choose “yes”, execution starts normally and finishes identically as before. If you choose “no”, the previous message reappears and the cycle continues until you reply “yes”. The output files will be located in the same folder as the input file, e.g., in the `INPUT` folder for this example. (For convenience, a window size of at least 120x36 is recommended; this size can be saved for future use in the Terminal Preferences.)

To use any sample input file contained in one of the example subfolders within the **Examples** folder, move it to the root of the **SMARTS\_295\_Mac** folder, thus replacing any pre-existing file of the same name. However, before undertaking the execution of any new run using an input file with the same name, it is imperative to either rename the existing output files, or to move them to the **OUTPUT** folder or any other folder. *Failing to do so would prevent proper execution.*

Power users can change this feature by altering the appropriate **OPEN** statements in the Fortran code, so that at anytime they are overwritten. Recompile the program and run again. The source code resides in the **Source\_Code** folder and can be compiled with any Fortran77 compiler. An open-source (free) Fortran compiler (g77) is available (<http://www.gnu.org/software/fortran/fortran.html>). It may be installed in various ways, for instance through the combined use of Fink (<http://fink.sourceforge.net/>) and Xcode (<http://developer.apple.com/tools/xcode/index.html>), which can be both obtained freely. Once g77 is installed, the Terminal command

```
g77 -o smarts295 -O3 -fdollar-ok -w smarts295.f
```

should complete without any errors and create an optimized executable called ‘smarts295’. (Note that there is no space before “-ok” and the “-O3” optimization option uses a capitalized O.) Finally, move or copy this new executable to the **SMARTS\_295\_Mac** folder.

### **Command-line (batch) execution**

In batch mode the program runs without user interaction. This means that once the command “./smarts295bat” is issued, the program will read the default input file (‘smarts295.inp.txt’), calculate the solar spectrum, and pass the results to the default output files (e.g. ‘smarts295.out.txt’, ‘smarts295.ext.txt’, and ‘smarts295.scn.txt’). The file ‘smarts295bat’ has been produced by activating the command “batch=.TRUE.” (line 188 of the code) and recompiling the code as described above. This executable must reside at the root of the **SMARTS\_295\_Mac** folder.

To run the Command Line executable from the Terminal, start by moving to the directory where you have extracted the package if you are not there yet:

```
cd SMARTS_295_Mac
```

and hit Enter. Then type

```
./smarts295bat
```

and hit Enter. This first message appears:

```
*****  
Welcome to SMARTS, version 2.9.5!  
*****
```

and, when the program stops, a final message similar to this one appears:

```
Total calculation time: 0.124 sec  
[Solar:/SCIENCE/SMARTS_295_Mac] chris%
```

It is clear that the actual execution time will vary according to your computational power. Execution of the sample input file must have created two output files: ‘smarts295.out.txt’ and ‘smarts295.ext.txt’ at the root of the **SMARTS\_295\_Mac** folder. To avoid problems, it is recommended to move the output files to the **OUTPUT** folder (or any other folder) before the next run.

## Scripting capabilities

Unix offers a series of shell commands that can enhance the batch capabilities of the SMARTS code. A sample shell file, called ‘smarts295script.csh’, is provided in the **Source\_code** folder. It makes use of the ‘smarts295bat’ executable compiled for pure batch mode (see previous section). The content of this file is listed below. A user familiar with shell scripting can take further advantage of the batch mode and create even more complicated scripts.

```
#!/bin/csh
# Use "source smarts295script.csh" to execute this file
# or
# "chmod u+x smarts295script.csh"
# and then
# "./smarts295script.csh"
#
foreach i (01 02 03 04 june july 20050914)
cp -f smarts_${i}.inp.txt smarts295.inp.txt
./smarts295bat
mv smarts295.out.txt smarts_${i}.out.txt
mv smarts295.scn.txt smarts_${i}.scn.txt
mv smarts295.ext.txt smarts_${i}.ext.txt
end
```

This example script assumes that seven input files are present in the root folder, with a variety of naming conventions (for demonstration purposes): ‘smarts\_01.inp.txt’, ‘smarts\_02.inp.txt’, ‘smarts\_03.inp.txt’, ‘smarts\_04.inp.txt’, ‘smarts\_june.inp.txt’, ‘smarts\_july.inp.txt’, and ‘smarts\_20050914.inp.txt’.

When the script is activated (after it is moved to the root of the main folder, and the instructions from the comments in the first few lines of the script are followed), the content of each of these files is copied to the default input file with the help of the “cp” command (copy). Then the ‘smarts295cli’ executable reads this input file, make the appropriate calculations and pass the output to the default files. In order not to overwrite the default output files, they are renamed with the help of the “mv” command (move). Once the “end” statement is reached, the “foreach” command will move to the next argument. If all arguments are used, the user will be returned to the standard prompt.

It is clear that the user can change the arguments of the “foreach” command to suit his/her needs. All text files and the executable are assumed to reside in the root directory of the code.

## What’s next?

At this point, you are certainly eager to customize the default input file to obtain the results you really want. Use an appropriate text editor, such as TextEdit, TextWrangler or BBEdit, to modify the input file (‘smarts295.inp.txt’) according to your needs, based on the detailed explanations provided in Section 6 of the User’s Manual. Save it as TEXT (Unix format), with the same filename, or any other name of your choice, in the format ‘any\_file\_name.inp.txt’ (e.g., ‘Test\_for\_July.inp.txt’). Note that the default input filename is ‘smarts295.inp.txt’, and remember that filenames are case sensitive if the program is invoked with a command line from a Unix shell (e.g., with Terminal).

## System and Hardware requirements

System: Mac OSX, version 10.2 or later; preferably 10.4 or later.

Processor: PowerPC G4 or G5 (not tested on G3 nor on Intel-based machines).



© Solar Consulting Services, 2005