

# Scilab Contest Instructions

## ***Submission of the toolbox***

With each toolbox you have to give:

- One page in English with the name of the toolbox, the authors and their affiliation, and a brief description of the toolbox.
- A paper in English which describes the requirements for using the toolbox such as the operating system (Windows, Linux...), the compilers (C, C++...), the libraries (JAVA...), Scilab version and other relevant information if needed, together with a small user's guide which explains how to install the toolbox with Scilab.

## ***Technical points***

The purpose of these technical points is that your toolbox can be used by other people using Scilab. After the requirements for using the toolbox are fulfilled, the user must be able to use the toolbox with his standard Scilab version. This is an important point because then your toolbox can be put on Scilab Web site and so be easily available for all Scilab community.

- The Scilab distribution that must be used for the contest is the official stable version 4.12 or higher. It can be downloaded from Scilab Web site **[www.scilab.org](http://www.scilab.org)** or **[www.scilab.org.cn](http://www.scilab.org.cn)**.
- The toolbox will be given as a compressed file using standard file compressor such as zip or gzip. The compressed file must only include the toolbox and possibly all what is necessary to use it (such as additional libraries for instance): the toolbox must work with standard Scilab versions already installed.
- The toolbox must include on-line help files describing its purpose and the on-line help of each user callable Scilab function if any. The help files must be written using the standard XML language using Scilab DTD.
- A demonstration of the toolbox, using for instance Scilab “demo” files, will be appreciated.
- We recall that for building and loading toolboxes into Scilab, standard “builder.sce” and “loader.sce” files can be defined using the “get\_absolute\_file\_path”, “genlib”, “ilib\_build”, etc... Scilab functions. These functions allow, in most cases, to create system independent “loader.sce” and “builder.sce” files.