

initpackage: initialize a package of local models

The new **initpackage** command initializes a package of local models from their source code and from a model component description file in “model.dat” format which defines the component’s name, type, function call, and its parameter names and initial settings. Further details of the file format, function and parameter specifications are given in Appendix C, [Adding Local Models To XSPEC](#). **initpackage** is NOT CURRENTLY SUPPORTED on **Cygwin**. See the **static_initpackage** subsection below for the alternative local model build method for **Cygwin** users.

Syntax: **initpackage** <name> <description file> [<directory>] [-udmget]

The <name> argument names the package. For internal reasons package names must be lowercase: the initpackage command will force lower case and warn the user if the argument contains uppercase letters. Also there should be no numerals in the package name.

The <description file> argument specifies the model component description file. The third argument <directory> is optional and specifies the location of the source code. If it is not given, the value of the setting LOCAL_MODEL_DIRECTORY given in the user’s Xspec.init file will be used. Finally, the <description file>, if not specified as an absolute pathname, will be read from the same directory as the source code.

Another optional argument is “-udmget”, for local model libraries containing Fortran code which makes use of XSPEC’s now-obsolete udmget function for dynamic memory allocation. None of the functions in XSPEC’s built-in models library use udmget anymore, and the necessary xsudmget.cxx file no longer resides there. If a user still requires this code for their own local models, they should add “-udmget” (without the quotes) at the end of the command line. **initpackage** will then copy the files xsudmget.cxx and xspec.h into the user’s local model directory.

initpackage performs the following tasks:

- reads the model description file

- writes code that will load the new component calculation functions

- writes a makefile that will drive the compilation and installation of the new code

- invokes the compiler and builds the library.

A separate command, **lmod**, actually loads the library. This two step process makes it easier to determine where the user is during the process if compilation failures arise. Further, if the model is complete and working correctly, only the **lmod** command need be invoked.

Initpackage can also be run as a stand-alone program outside of XSPEC. When used like this however, after initpackage has finished the user must manually run hmake to build their library. XSPEC performs this part automatically using a script file.

CYGWIN Users: static_initpackage

XSPEC dynamic libraries are not currently supported on Cygwin. Therefore local model libraries must be statically built outside of the XSPEC shell, and the xspec.exe executable must be re-linked to pick up the new library. Unlike on other platforms, local model libraries cannot be loaded at runtime with the **lmod** command. We have provided a static variation of the initpackage command for Cygwin users:

Syntax: **static_initpackage** <description file> <directory>
 static_initpackage none

Unlike initpackage, this command can only be run as a stand-alone application outside of XSPEC. The <description file> argument is the same as for initpackage. The <directory> is the full path to your local model directory and is a required argument (also unlike initpackage). There is no <name> parameter since all static local model libraries will be assigned the name “static_local_mods.” This command will perform similar tasks to initpackage, but for a static library build, and then will automatically re-link the XSPEC executable to pick up the new library.

The “static_initpackage none” option will restore the default (empty) static model library to the XSPEC executable. It will not remove or alter anything in the user’s local model directory.