

Resound API

Resound is designed to be extended by modules you write to meet your own needs. The idea here is: why write a sound editor to do some thing you need to do, when you can just write the part you need and plug it into a full-featured editor to begin with?

Resound's API consists of four files: *Module.h*, the object that your Resound Module must descend from, *ModuleProtocol.h*, which describes how you communicate with Resound, *ModuleSound.h*, a category of additional useful sound methods, and *ModuleMenuNode.h*, a description of the "Menu Nodes" you need to create to inform Resound about how to build your Modules menu when it loads your module. Each of these four files comes with an accompanying documentation file.

How Do I Write a Module?

Most of the description of the Resound API is found in Resound's on-line help. You should check there for tips on building modules, caveats, and more complete descriptions of each of the pieces above. Start with the help option *The Resound API*.

Can't think of anything to make a module for? Check out *Module Ideas*. I'd love to hear of anything you're working on.

Examples

The API comes with a "Walk Through" tutorial for building *Zero.rmod*, a very simple module which zeroes out sound selections. Try this tutorial before tackling a real project.

Resound also comes with the source code to one fairly large module, *Math.rmod*, which shows quite a lot more than just the simple Zero example, including loading nib files and multiple menu options. You're welcome to use any of the code in *Math.rmod* or the example tutorial in your own modules.

A Late Note on Symbols

Modules can't link against libraries. Only the application itself. That means that if you use a function or method whose symbol isn't in Resound's symbol table, your module cannot load.

Resound has been linked against several major libraries, and includes all of their symbols using the **-all_load** option. This adds about 150K to the application, but it's the only way I can give you at least a fighting chance of being able to use common functions in your libraries. Nonetheless, it's possible you may try to use a symbol that Resound doesn't have internally. If you need me to link against some odd library not included in Resound, send me mail.

If you have any questions, feel free to contact me at **seanl@cs.umd.edu**

Sean Luke
U Maryland at College Park