

### 59.7.2 Trying to do too much

Your module will be part of a developer's toolkit. It will not, in itself, form the **entire** toolkit. It's tempting to add extra features until your code is a monolithic system rather than a set of modular building blocks.

### 59.7.3 Inappropriate documentation

Don't fall into the trap of writing for the wrong audience. Your primary audience is a reasonably experienced developer with at least a moderate understanding of your module's application domain, who's just downloaded your module and wants to start using it as quickly as possible.

Tutorials, end-user documentation, research papers, FAQs etc are not appropriate in a module's main documentation. If you really want to write these, include them as sub-documents such as `My::Module::Tutorial` or `My::Module::FAQ` and provide a link in the SEE ALSO section of the main documentation.

## 59.8 SEE ALSO

### *perlstyle*

General Perl style guide

### *perlnewmod*

How to create a new module

### *perlpod*

POD documentation

### *podchecker*

Verifies your POD's correctness

### Packaging Tools

*ExtUtils::MakeMaker*, *Module::Build*

### Testing tools

*Test::Simple*, *Test::Inline*, *Carp::Assert*, *Test::More*, *Test::MockObject*

### <http://pause.perl.org/>

Perl Authors Upload Server. Contains links to information for module authors.

### Any good book on software engineering

## 59.9 AUTHOR

Kirrily "Skud" Robert <skud@cpan.org>