

1 Apache/Perl Modules

1.1 Description

On CPAN you will be able to find a number of Perl modules created to run under `mod_perl`. We will give a presentation of them here and ways to get at these modules.

Apache/Perl modules are somewhat different from the many others found on CPAN (<http://www.cpan.org/>). Most Perl modules are "building blocks"; developers use them to build applications or even more building blocks. It just so happens that Apache/Perl modules are encapsulated inside a namespace and `.pm` file. However, this does not make them all building blocks, many are just like Apache modules written in C: you install them, edit the server configuration files and you're done.

Before you start a new module, consider this: Apache/Perl modules are useless outside of `mod_perl` and Apache. Do your best to design such that your `Apache::*` module will make good use of the building blocks found on CPAN, putting together new building blocks where appropriate and simply glue them together with the Apache API, keeping the `Apache::*` modules very small.

These modules will also serve as good examples, showing you where modules can step in during a stage of a request with `Perl*Handler` callbacks. You'll also see how and when to use the Apache API methods.

1.2 The Module List

Like other CPAN modules, the Apache/Perl modules are listed on the Perl module list, under the *World Wide Web* category: http://www.cpan.org/modules/00modlist.long.html#ID15_WorldWideW . Browse down the page to find the `Apache::*` modules, organized by their role in the request cycle.

1.3 Where Are The Modules Kept?

Modules listed with the **APML** as is contact are part of the `mod_perl` distribution package. Other modules can be found on CPAN in the <http://www.cpan.org/modules/by-module/Apache/> directory. If they have not made it to CPAN yet, they might be found in the `mod_perl contrib` directory at <http://perl.apache.org/dist/contrib/>. Otherwise, feel free to ask the contact author or the list how you can go about getting your hands on a copy. The modules listed with **MPB** as the author were present in the *Writing Apache Modules with Perl and C* book by Lincoln Stein and Doug MacEachern. If you can't find these modules on CPAN, look for them at <http://modperl.com:9000/book/source/>.

1.4 Playing Your Part

Apache and Perl are world-wide collaborative efforts; naturally, the Apache/Perl integration project is the same. You've started on the right foot simply by reading this document, thanks! Please make sure you've read the main Perl module list at <http://www.cpan.org/modules/00modlist.long.html>, all that is said there is relevant here!

If you wish to share your module(s) in the Perl way, by putting them on CPAN, send a request to modules@perl.org to get set up with a PAUSE user id and directory. See <http://www.cpan.org/modules/04pause.html> for more details.

HINT: For a nice set of template files try this:

```
% h2xs -AX -n Apache::YourPackageName
```

1.4.1 Module Naming Conventions

Like for normal Perl modules on CPAN, the `Apache::` namespace conventionally used for `mod_perl` modules needs to keep a certain organization. This is why you should choose a multi-level name for your module, categorizing it into a namespace that you feel fits your module. There should be a very good reason for a module to not have a nested name (ie. living directly under `Apache::`), that should only be the case for the core API modules and older modules which may keep their names.

Here are some examples of namespaces that you may want to use:

```
Apache::App      Applications: any module serving content with a
                  point-and-click interface *

Apache::Auth     Modules related to general user authentication/authorization

Apache::Authen  Modules solely related to authentication

Apache::Authz   Solely related to authorization

Apache::Access  Only access control

Apache::Config  Configuration-related things

Apache::Devel   Development-related modules (debug/devel help)

Apache::Filter  Filtering modules
  ::Output     Output filters (1.0 style Filter/OutputChain or 2.0 style)
  ::Input      Input filters

Apache::Framework Modules providing application frameworks (templating
                and/or publication frameworks) *

Apache::Log     Logging modules

Apache::Method  Method handlers (PATCH/PUT and the like)

Apache::MIME    MIME-type modules

Apache::Monitor Monitoring modules

Apache::Persistent Modules providing for persistency in connections

Apache::Proxy   Proxies

Apache::RPC     RPC/SOAP modules
```

```

Apache::Trans    URI Translation handlers

Apache::Util     General-utility modules, usually not handlers but
                  providing an API to use in your code

```

*: If you are writing a small application or templating module, contained in one module, you can use these namespaces. However, if you are planning a substantial framework with many inter-related modules, you should probably go with a top-level namespace outside of `Apache:::`.

For an illustration of how this would look for the older modules, you can look at this example categorization.

Note: *Of course* none of these categories are cast in stone: you should feel free to choose the module name that suits your module best instead of trying to follow these general indications. If in doubt, just ask around at the `mod_perl` list; you are sure to get some assistance in naming your module! Furthermore, you should look at the way it's done on CPAN, and apply it under the `Apache:::` namespace.

1.5 Porting CPAN modules to mod_perl 2.0 Status

If you want to port your or someone else's module to `mod_perl 2.0`, please refer to the these documents: porting Perl modules to `mod_perl 2.0` and `mod_perl 1.0` compatibility reference.

1.5.1 Ported 3rd-party Modules

The following modules have already been ported to `mod_perl 2.0` and are available from CPAN. If you know of other modules that have been ported but not listed here, please contact the `modperl` list, so we can update this information:

Module Name	Required Dist Package
-----	-----
Apache::ASP	Apache-ASP-2.55
Apache::DBILogin	Apache-DBILogin-2.03
Apache::AuthCookie	Apache-AuthCookie-3.05
Apache::AuthExpire	Apache-AuthExpire-0.38
Apache::AuthNetLDAP	Apache-AuthNetLDAP-0.25
Apache::AuthPerLDAP	Apache-AuthPerLDAP-2.01
Apache::AuthenNTLM	Apache-AuthenNTLM-2.04
Apache::AuthenPasswd	Apache-AuthenPasswd-0.12
Apache::AuthenSmb	Apache-AuthenSmb-0.70
Apache::AuthenURL	Apache-AuthenURL-2.02
Apache::AuthzNetLDAP	Apache-AuthzNetLDAP-0.07
Apache::AuthzPasswd	Apache-AuthzPasswd-0.11
Apache::Clean	Apache-Clean-2.00_4
Apache::DB	Apache-DB-0.10
Apache::DProf	Apache-DB-0.10
Apache::GeoIP	Apache-GeoIP-1.215
Apache::PAR	Apache-PAR-0.30
Apache::Peek	Apache-Peek-1.03
Apache::Scoreboard	Apache-Scoreboard-2.01
Apache::SessionManager	Apache-SessionManager 1.00
Apache::SmallProf	Apache-DB-0.10

Apache::Syntax::Highlight::Perl	Apache-Syntax-Highlight-Perl-1.00
Apache::VMonitor	Apache-VMonitor-2.0
CGI	CGI.pm-3.10
CGI::Cookie	CGI.pm-3.10 (comes in the CGI dist)
Apache::Request	Apache2-Request (comes in libapreq2 dist)

1.6 Porting in Process

The following modules are in the process of being ported to mod_perl 2.0. Please contact the porter if you wish to help the porting.

If you have started the porting, in order to avoid duplicated efforts please contact the modperl list, and let us know what module you are porting. This information should also help others to help you to do the porting and testing.

Module	Porters
-----	-----
Apache::Language	Philippe M. Chiasson <gozer AT cpan.org>
Apache::AutoIndex	Philippe M. Chiasson <gozer AT cpan.org>
Apache::Icon	Philippe M. Chiasson <gozer AT cpan.org>
Apache::Template	Geoffrey Young <geoff AT modperlcookbook.org>
Apache::AuthzNIS	

1.7 mod_perl 2.0 Specific Modules

The following modules have gone beyond the porting stage and are specifically designed for mod_perl 2.0 and the new features and functionality it brings. Each is available from CPAN. If you know of other modules that have been written for the 2.0 API but not listed here, please contact the modperl list, so we can update this information:

Module	Maintainers
-----	-----
Apache::AuthenHook	Geoffrey Young <geoff AT modperlcookbook.org>
Apache::IncludeHook	Geoffrey Young <geoff AT modperlcookbook.org>
Apache2::DebugFilter	Philip M. Gollucci <pgollucci AT p6m7g8.com>
Apache2::Filter::HTTPHeadersFixup	Philip M. Gollucci <pgollucci AT p6m7g8.com>

1.8 Maintainers

The maintainer is the person you should contact with updates, corrections and patches.

- Per Einar Ellefsen <pereinar (at) oslo.online.no>

1.9 Authors

- Per Einar Ellefsen <pereinar (at) oslo.online.no>

1.9 Authors

- Doug MacEachern

Table of Contents:

1	Apache/Perl Modules	1
1.1	Description	2
1.2	The Module List	2
1.3	Where Are The Modules Kept?	2
1.4	Playing Your Part	2
1.4.1	Module Naming Conventions	3
1.5	Porting CPAN modules to mod_perl 2.0 Status	4
1.5.1	Ported 3rd-party Modules	4
1.6	Porting in Process	5
1.7	mod_perl 2.0 Specific Modules	5
1.8	Maintainers	5
1.9	Authors	5