

# Products

There is a lot of software out there ready to run with  
mod\_perl and/or help you with your programming project.

Last modified Mon Jun 29 03:41:16 2009 GMT

- 1. Apache/Perl Modules

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.

- 2. Application Servers and Toolkits based on mod\_perl

There are several application servers and toolkits available designed to run under mod\_perl, which might provide you with inline-Perl coding, or MVC (Model/View/Controller) frameworks.

- 3. mod\_perl Related Software

A short list of some software, open source and commercial, based on or compatible with mod\_perl, which might be interesting for anyone using mod\_perl

# 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](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](mailto: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

## **2 Application Servers and Toolkits based on mod\_perl**

## 2.1 Description

There are several application servers and toolkits available designed to run under `mod_perl`, which might provide you with inline-Perl coding, or MVC (Model/View/Controller) frameworks.

You probably want to look at [Choosing a Templating system](#) for more information and in-depth analysis of some of the ones listed here.

The following frameworks are sorted alphabetically.

## 2.2 Apache::ASP

`Apache::ASP` (<http://www.apache-asp.org/>) provides an Active Server Pages port to the Apache Web Server with Perl scripting only, and enables developing of dynamic web applications with session management and embedded perl code. There are also many powerful extensions, including XML taglibs, XSLT rendering, and new events not originally part of the ASP API

## 2.3 Apache::PageKit

`Apache::PageKit` (<http://pagekit.org/>) is a web application framework that uses `HTML::Template` and XML to separate the Model, View, Content and Controller. Provides elegant solutions to many difficult web programming problems, including session management, language localization, authentication, form validation, and co-branding.

## 2.4 AxKit

`AxKit` (<http://axkit.org/>) is an XML Application Server for Apache. It provides on-the-fly conversion from XML to any format, such as HTML, WAP or text using either W3C standard techniques, or flexible custom code. `AxKit` also uses a built-in Perl interpreter to provide some amazingly powerful techniques for XML transformation.

## 2.5 The bivio OLTP Platform (bOP)

The bivio OLTP Platform (<http://www.bivio.biz/hm/why-bOP>) (bOP) is a declarative, multi-layered Perl application development framework. Adapters have been written for Apache/`mod_perl`, `sendmail`, Oracle, and PostgreSQL. The Model layer generates SQL for CRUD, lists, and HTTP forms. Views are content-independent hierarchies of widgets. Tasks control security, execution order, and state transitions. Facades map qualified names to URLs, strings, fonts, colors, icons, files, and views. bOP provides a full-suite of services including dynamic tracing, configuration, logging, class loading, type hierarchy, unit testing, and acceptance testing.

## 2.6 Catalyst

Catalyst is an elegant web application framework, extremely flexible yet extremely simple. It's similar to Ruby on Rails, Spring (Java), and Maypole, upon which it was originally based. Its most important design philosophy is to provide easy access to all the tools you need to develop web applications, with few restrictions on how you need to use these tools. However, this does mean that it is always possible to do things in a different way. Other web frameworks are initially simpler to use, but achieve this by locking the programmer into a single set of tools. Catalyst's emphasis on flexibility means that you have to think more to use it. We view this as a feature. For example, this leads to Catalyst being more suited to system integration tasks than other web frameworks.

Intro document <http://search.cpan.org/~jrockway/Catalyst-Manual-5.700501/lib/Catalyst/Manual/Intro.pod>

Main site page <http://catalyst.perl.org/wiki>

## 2.7 CGI::Application

`CGI::Application` is intended to make it easier to create sophisticated, reusable web-based applications. This module implements a methodology which, if followed, will make your web software easier to design, easier to document, easier to write, and easier to evolve.

`CGI::Application` builds on standard, non-proprietary technologies and techniques, such as the Common Gateway Interface and Lincoln D. Stein's excellent `CGI.pm` module. `CGI::Application` judiciously avoids employing technologies and techniques which would bind a developer to any one set of tools, operating system or web server.

There's a good article on perl.com about this framework <http://www.perl.com/pub/a/2001/06/05/cgi.html>.

## 2.8 DataBreeze Web Database Platform

DataBreeze is written entirely in Perl and utilizes `mod_perl` within Apache to speed performance. Feedback from the Perl/`mod_perl`/Apache community is important. You may download the software for evaluation at <http://www.databreeze.com> or simply logon and try DataBreeze directly online.

DataBreeze is very different from typical development tools because it utilizes a pre-built application framework that requires no programming. DataBreeze provides a template environment for building database applications that list, search, explore and edit relational data. The system includes integrated menus, forms, tree-views, security, session/state management and more, all in a Web Browser. DataBreeze can also be extended through the use of Event programs using Perl.

This is not free software but nothing is hidden, the complete Perl source code is included so that you may explore how it works.

## 2.9 Embperl

`Embperl` (<http://perl.apache.org/embperl/>) is a system for building dynamic websites with Perl. It gives you the power to embed Perl code in your HTML documents and the ability to build your Web site out of small reusable objects in an object-oriented style. You can also take advantage of all the usual Perl modules, (including `DBI` for database access) use their functionality and easily include their output in your web pages. `Embperl` has several features which are especially useful for creating HTML, including dynamic tables, form field processing, URL escaping/unescaping, session handling, and more.

## 2.10 Gantry

`Gantry` (<http://www.usegantry.org>) is a web application framework for Apache/`mod_perl`, CGI and Fast-CGI. `Gantry`'s object oriented design allows for a pragmatic, modular approach to URL dispatching. `Gantry` supports MVC and initiates rapid development, offering an organized coding scheme for web applications.

It can take advantage of its sister project `Bigtop`, which allows you to describe the data of your application and have it generate or regenerate your database schema and module objects without destroying the portions you have customized.

`Gantry` also has a robust configuration abstraction layer `Gantry::Conf` that allows you to change how your application retrieves its configuration data without requiring any code changes.

## 2.11 Interchange

`Interchange` (<http://www.icdevgroup.org/>) is a flexible, high performance application server that handles state management, authentication, session maintenance, click trails, filtering, URL encodings, security policy. It's made up of the following components: database abstraction layer, generic templating system, transaction routing rules, customer information object, universal localization scheme, security blackout definition, profiles, filters, search language, and session management.

Some of `Interchange`'s many modules are transaction management, pricing, personalization, payment processing, reporting, customer service, and search.

## 2.12 Jifty

`Jifty` is a way to build web applications. Homepage <http://jifty.org/view/HomePage>

## 2.13 Mason

`Mason` (<http://www.masonhq.com/>) is a powerful Perl-based web site development and delivery engine. With `Mason` you can embed Perl code in your HTML and construct pages from shared, reusable components. `Mason` solves the common problems of site development: caching, debugging, templating, simulating browser conditions, maintaining development and production sites, and more

## 2.14 Maypole

Maypole is a Perl framework for MVC-oriented web applications, similar to Jakarta's Struts. Maypole is designed to minimize coding requirements for creating simple web interfaces to databases, while remaining flexible enough to support enterprise web applications.

Homepage <http://maypole.perl.org/>

## 2.15 OpenInteract

OpenInteract (<http://www.openinteract.org/>) is a web application environment written in perl and geared to run on the Apache web server using the mod\_perl plugin module. The environment is built to be not only friendly to people editing and changing a website's content, but also for the developers who can write code (or complex templates) and create entire applications.

## 2.16 OpenFrame

OpenFrame (<http://openframe.fotango.com/>) is an open source application framework for distributed media applications. What all this buzzword-compliant mumbo-jumbo really means is that with OpenFrame you can write a single application for multiple downstream clients (ie, Web, WAP, iDTV, Email) and not have to worry about much except the presentation.

## 2.17 PLP

PLP (<http://plp.juerd.nl/>) is yet another Perl embedder, primarily for HTML documents. Unlike with other Perl embedders, there is no need to learn a meta-syntax or object model: one can just use the normal Perl constructs. PLP runs under mod\_perl for speeds comparable to those of PHP, but can also be run as a CGI script.

## 2.18 The Template Toolkit

The Template Toolkit (<http://template-toolkit.org/>) is a fast, powerful and easily extensible template processing system written in Perl. It is ideally suited (but not limited) to the creation of static and dynamic web content. The `Apache::Template` module provides an `Apache/mod_perl` interface to the Template Toolkit, providing a quick and simple way to integrate its processing power into your web site.

## 2.19 Uttu

Uttu (<http://uttu.tamu.edu/>) is a web-application driver that provides support for writing application frameworks. This support includes configuration, database connectivity, caching, uri-to-filename translation, and application installation. Uttu makes working with multiple virtual hosts and `HTML::Mason`, Template Toolkit, or `AKIT` almost trivial.

## **3 mod\_perl Related Software**

## 4 mod\_perl Related Software

---

### 4.1 Applications running under mod\_perl

These application give you an idea of the range of products available for use with mod\_perl, which are granted extra speed and/or functionality by taking advantage of the Apache API.

#### 4.1.0.1 Content Management Systems

- Krang is an Open Source web-publisher / content-management system designed for large-scale magazine-style websites. It is a 100% Perl application using Apache/mod\_perl and MySQL, as well as numerous CPAN modules.  
Krang provides a powerful and easy to use story and media editing environment for magazine editors, as well as a complete template development environment for web designers. On the back-end, Perl programmers can customize Krang to control the data entered in the story editor and add code to drive the templates to build output. Krang can be enhanced with add-ons containing new skins and other new features. Krang easily handles large data sets and can manage multiple websites in a single installation.
- Bricolage is a full-featured, open-source, enterprise-class content management system. It offers a browser-based interface for ease-of use, a full-fledged templating system with complete programming language support for flexibility, and many other features. It operates in an Apache/mod\_perl environment, and uses the PostgreSQL RDBMS for its repository.
- Slashcode is the codebase behind the highly popular Slashdot site. It provides content management and discussion possibilities for its users, and can support a high load of users, with self-moderation because of the high number of comments.
- Everything is an information management system, using a combination MySQL and Perl to create a flexible system of entering, linking, and retrieving information. Its codebase is for example used to run Perl Monks, a Perl community site.
- Scoop is a "collaborative media application". It falls somewhere between a content management system, a web bulletin board system, and a weblog. Scoop is designed to enable your website to become a community. It empowers your visitors to be the producers of the site, contributing news and discussion, and making sure that the signal remains high. Was written originally for Kuro5hin.org, which is still the biggest Scoop site.
- The WYPUG website is powered by a CMS called Shapeshifter that is written entirely in Perl and designed to run under mod\_perl (in Registry mode). It's a template-based, database-driven CMS with some unique features. It's not finished yet, hence the lack of documentation or an advocacy website - but it's not far away. The system will be a commercial product once finished, but it'll be released under an Open Source license. Inquiries should go to Digital Word.
- Movable Type is a web-based personal publishing system. It is designed to ease maintenance of regularly-updated news or journal sites, like weblogs. The Movable Type system runs on your own webserver, giving you complete control over your site.
- Metadot provides secure, web-based collaboration and content management software that allows non-technical users to build and manage intranets, extranets, and project spaces.
- WebGUI (pronounced web-goey) is one of the most popular open source content management

systems (CMS) available today. It is a platform built to allow average business users to build and maintain complex web sites. It is modular, pluggable, and platform independent. It was designed to allow the people who create the content, to manage it online, rather than content management taking up the time of the busy IT Staff.

- Callisto CMS is an XML/XSL Web-based Content Management System built on AxKit. It supports WYSIWYG XML content editing, transactional site deployment to multiple target web servers, multiple users, virtual hosting support and other features.
- EplSite WorkFlow System and Portal Web based on EmbPerl and MYSQL database.
- Mason-CM is a web content management system for arbitrary file based web productions, based on Mason. It's optimized to manage multiple sites of SMB's, together with multiple graphic designer companies or professional content contributors. Mason-CM has all the basic needs of web content management as versioning, rollback, staging and productive area, trashcan, locking and search. Files, ASCII and binaries, can be viewed or downloaded. It supports remote filesystems and more.
- Spine is a web based Content Management System, implemented in mod\_perl and released under the GPL (GNU Public License). It requires Apache version 1 or 2, a mod\_perl installation to match and a database (PostgreSQL or MySQL). It features mixed static/dynamic content, separated template and content administration, granular privileges, user-friendly URLs, file manager, plugins, ...

#### 4.1.0.2 Forum Software

See above too, many of the CMSes above feature very good discussion capabilities.

- mwForum is an open-source, Web-based discussion forum system. mwForum is based on Perl CGI scripts, uses a MySQL database and is compatible with mod\_perl for optimal performance. mwForum's design goals are comfortable operation and administration, stability, security and speed, a professional and consistent look, and lean and clean code to make customization easy.
- WebBoard is the number-one selling message board software in the marketplace with over one million sites using it as their primary means for communication. The software has long served as a highly reputable online education tool that can run both as a stand-alone product and as an integrated element within other learning management systems. Many installations of WebBoard are also used for customer service and corporate intranet communication.

#### 4.1.0.3 Webmail servers

- acmemail is a multiuser IMAP/POP3 to Web gateway (or webmail program). It reads mail from a mail server and displays it using HTML on a web server. It is written in Perl, has full support for MIME (and mod\_perl) and is quite pretty.
- @Mail is a fully featured Web Mail server that allows you to set up Web Mail at your domain or server existing POP3 accounts. Including full source code, under a commercial license.

@Mail Includes auto-account creation via Sendmail/Exim/Qmail , scheduler, SQL mailbox support, spell check, group address-book, user preferences, multiple accounts, migration scripts from other mail-servers, attractive DHTML interface, Anti-Virus integration, User Payment service and in-depth Web-based administration of accounts.

- WING is an Open Source Apache/mod\_perl based system which allows users to access email held on an IMAP server via any web browser.

#### 4.1.0.4 Calendars

- Chronos is a Web agenda/calendar for Intranets (but it can be used from anywhere). It can send reminders by email. It can schedule multi-user events. It is fast and light on resources. It is not proprietary, ugly nor buggy.
- CyberCalendar is a calendar of events program for the web and PDAs written entirely in Perl.
- WebCalendar is a powerful group calendaring and scheduling system. It is a web based application that makes it easy to publish and manage calendar events on the internet. WebCalendar includes an impressive set of business class features. Its easy to use interface, rich functionality, and sophisticated scheduling system makes it one of the best calendar applications on the internet.

#### 4.1.0.5 Other

Of course, not every application running under mod\_perl is a Content Management System or a Forum!

- Loggerithim is monitoring package for Linux and Solaris (with early support for OS X and Win32). It allows you to collect metrics from your hosts and analyze them with graphs and whatnot. You can also set thresholds for proactively spotting problems. Modules can be written to supplement the stock data collection.
- DnsZone is a fully functional application for Apache/mod\_perl. It implements a web based interface for managing zones with a nameserver running BIND.
- DAD is a comprehensive website ad management system. It is a mod\_perl application with advanced targeting capabilities, a comprehensive administrative interface, and automatic client reporting. Because it uses mod\_perl, DAD can take advantage of Apache API elements like subrequests and notes, and it can rely on persistent database connections through DBI. These features allow DAD to efficiently implement advanced targeting rules that would be impossible under CGI.
- Ringlink is a Perl program that provides the tools you need to run one or more rings of web sites, i.e. systems of links between web sites of a similar topic, content, or interest.
- Request Tracker (RT) is an industrial-grade ticketing system. It lets a group of people intelligently and efficiently manage requests submitted by a community of users. RT is used by systems administrators, customer support staffs, NOCs, developers and even marketing departments at over a thousand sites around the world. It can run under mod\_perl for performance benefits, and is the software behind [rt.cpan.org](http://rt.cpan.org).
- OTRS is an open source Ticket Request System and email management system with many features to manage customer telephone calls and email. The system is built to allow your support, sales, pre-sales, billing, internal IT, helpdesk, etc. department to react quickly to inbound inquiries. It is useful for people who receive many emails and want to answer them with a team of agents. It has been tested on Linux, Solaris, AIX, FreeBSD, OpenBSD, and Mac OS 10.x.
- Namp! is Not Another Music Player. In combination with the Apache web server and mod\_perl, Namp! lets you create a virtual jukebox of your MP3, WAV and Ogg Vorbis music files, browse them over the web, create playlists, and stream them.
- Mod\_Survey is a web questionnaire system implemented as a mod\_perl module for Apache. It is used as a content-handler for ".survey" files, or in other words for files containing surveys described in an

XML-based tag notation defined in the Survey v3.0.0 DTD. The system targets an audience consisting of people who need powerful questionnaire and data handling features.

The Mason project also has a list of applications running under mod\_perl and HTML : : Mason.

---

## 4.2 Toolkits for use with mod\_perl

See the Apache/Perl module list and Application Servers and Toolkits based on mod\_perl for a list of modules and toolkits running allowing you to tackle web development more easily under mod\_perl.

---

## 4.3 Compatible products

### 4.3.1 *Freeware:*

- Ralf S. Engelschall's mod\_ssl
- Ben Laurie's Apache-SSL

... and many more. Most of the modules for Apache should be able to be used more or less integrated with mod\_perl.

### 4.3.2 *Commercial:*

- RedHat's Stronghold, which embeds a number of Apache modules, including SSL support and Matt Sergeant's AxKit, which is an XML Application Server running under mod\_perl.
- 

## 4.4 Products allowing fast web programming and integration with web servers

### 4.4.1 *Freeware, source code distribution:*

#### 4.4.1.1 Perl

Other software solutions exist to run Perl faster than with CGI for web application development. You may want to try these if mod\_perl isn't what you need.

- SpeedyCGI is a way to run CGI perl scripts persistently, which usually makes them run much more quickly.
- FastCGI is a language independent, scalable, open extension to CGI that provides high performance without the limitations of server specific APIs.
- PPerl turns ordinary perl scripts into long running daemons, making subsequent executions extremely fast. It forks several processes for each script, allowing many processes to call the script at once.

- Ben Sugars' nsapi\_perl (Perl plugin for Netscape - not actively developed anymore)

#### 4.4.1.2 Other Languages

Many other languages feature Apache modules to facilitate web development, many being inspired by mod\_perl.

- The Tcl-Apache integration project, providing similar functionality to mod\_perl but with the Tcl programming language.
- Mod\_python is an Apache module that embeds the Python interpreter within the server.
- mod\_ruby embeds the Ruby interpreter into the Apache web server, allowing Ruby CGI scripts to be executed natively.
- PHP is a widely-used general-purpose scripting language that is especially suited for Web development and can be embedded into HTML.
- Tomcat is the servlet container that is used in the official Reference Implementation for the Java Servlet and JavaServer Pages technologies.

#### ***4.4.2 Commercial, binary distribution:***

- ActiveState's PerlEx (Perl plugins for O'Reilly WebSite, Netscape and IIS)
- 

If you know of other mod\_perl related software products, please send a description to the docs-dev mailing list.



# Table of Contents:

<b>Products</b>	1
<b>Apache/Perl Modules</b>	3
1 Apache/Perl Modules	3
1.1 Description	4
1.2 The Module List	4
1.3 Where Are The Modules Kept?	4
1.4 Playing Your Part	4
1.4.1 Module Naming Conventions	5
1.5 Porting CPAN modules to mod_perl 2.0 Status	6
1.5.1 Ported 3rd-party Modules	6
1.6 Porting in Process	7
1.7 mod_perl 2.0 Specific Modules	7
1.8 Maintainers	7
1.9 Authors	7
<b>Application Servers and Toolkits based on mod_perl</b>	9
2 Application Servers and Toolkits based on mod_perl	9
2.1 Description	10
2.2 Apache::ASP	10
2.3 Apache::PageKit	10
2.4 AxKit	10
2.5 The bivio OLTP Platform (bOP)	10
2.6 Catalyst	11
2.7 CGI::Application	11
2.8 DataBreeze Web Database Platform	11
2.9 Embperl	12
2.10 Gantry	12
2.11 Interchange	12
2.12 Jifty	12
2.13 Mason	12
2.14 Maypole	13
2.15 OpenInteract	13
2.16 OpenFrame	13
2.17 PLP	13
2.18 The Template Toolkit	13
2.19 Uttu	13
<b>mod_perl Related Software</b>	14
3 mod_perl Related Software	14
4 mod_perl Related Software	15
4.1 Applications running under mod_perl	15
4.1.0.1 Content Management Systems	15
4.1.0.2 Forum Software	16
4.1.0.3 Webmail servers	16
4.1.0.4 Calendars	17
4.1.0.5 Other	17

Table of Contents:

4.2	Toolkits for use with mod_perl . . . . .	18
4.3	Compatible products . . . . .	18
4.3.1	Freeware: . . . . .	18
4.3.2	Commercial: . . . . .	18
4.4	Products allowing fast web programming and integration with web servers . . . . .	18
4.4.1	Freeware, source code distribution: . . . . .	18
4.4.1.1	Perl . . . . .	18
4.4.1.2	Other Languages . . . . .	19
4.4.2	Commercial, binary distribution: . . . . .	19