

Larry 设计 Perl 5 的一个主要目标就是鼓励核心发行版之外开发与革新。Perl 4 有几个旁支，因为，举例来说，没什么简单的方法能让它连接到关系数据库。Larry 希望人们能够创建并维护他们自己对语言的扩展而无须使 Perl 分裂为千百个不兼容的混杂语言。

你可以为扩展而引入某种技术上的机制，但是你也需站在社区角度考虑此事。那些不在人群中共享的扩展和增强使得每个人必须自己动手对其进行编译、测试、调试和维护。如果没人能够找到或者找到了也不能修正甚至根本无权使用，那么共享扩展和库同样也是毫无意义的。

幸运的是，我们有 Perl 社区的存在。它强壮且健康。它欢迎任何级别的参与者——并不只有那些产出并共享代码的人。考虑利用无数其他 Perl 程序员的知识 and 经验，并同样地分享 你自己的能力。

社区网站

Perl 的主页位于 <http://www.perl.org/>。这个站点提供了文档、源代码、教程、邮件列表以及若干重要的社区项目。如果你是 Perl 新手，那么友善的 Perl 新手邮件列表就是你提问并得到准确解答的地方。参见 <http://beginners.perl.org/>。

<http://dev.perl.org/> 是一个值得一提的子域名，Perl 5、Perl 6 主站为 <http://www.perl6.org/>，甚至是 Perl 1 核心开发的中心站点。

Perl.com 每月发布若干关于 Perl 编程的文章和教程。其存档可追溯到20世纪。参见 <http://www.perl.com/>。

CPAN (*cpan*) 的中心位置位于 <http://www.cpan.org/>，虽然有经验的用户通常在 <http://search.cpan.org/> 上花去更多时间。这个承载着可重用、免费 Perl 代码的 软件发布中心枢纽是 Perl 社区的基本组成部分。

PerlMonks，位于 <http://perlmonks.org/>，此处可敬的社区站点专门用于 Perl 编程问答及其他讨论。它于2009年十二月过完十周岁生日，使其成为编程语言专用 Web 社区中持续时间最长者之一。

若干社区网站提供新闻和评论。最古老的是 <http://use.perl.org/>，它提供自主期刊已有许多年头。不少知名 Perl 黑客曾在那里执笔。<http://blogs.perl.org/> 是一个用于撰写 Perl 主题博客的新兴社区站点。

其他一些站点则集中展示了 Perl 黑客们对 Perl 的沉思，包括 <http://perlsphere.net/>、<http://planet.perl.org/> 以及 <http://ironman.enlightenedperl.org/>。后者是来自 Enlightened Perl Organization (<http://enlightenedperl.org/>) 倡议的一部分，以期增加 Perl 在线出版物的数量和质量。

Perl Buzz (<http://perlbuzz.com/>) 定期收集并重新发布一些最有趣、有用的 Perl 新闻。

开发网站

Best Practical Solutions (<http://bestpractical.com/>) 为 CPAN 作者以及 Perl 5、Perl 6 的开发维护着一份他们产品——流行的需求跟踪系统——RT的安装。每一个 CPAN 模块的发行版有着 属于自己的 RT 队列，search.cpan.org 的模块页上有对应的链接，指向 <http://rt.cpan.org/>。Perl 5 和 Perl 6 各自有一个位于 <http://rt.perl.org/> 的 RT 队列。

Perl 5 Porters (也称 p5p) 邮件列表是 Perl 5 开发讨论的中心点。参见 <http://lists.cpan.org/showlist.cgi?name=perl5-porters>。

Perl 基金会 (<http://www.perlfoundation.org/>) 为所有有关 Perl 5 的内容提供了一个 Wiki 参见 <http://www.perlfoundation.org/perl5>。

许多 Perl 黑客使用 Github (<http://github.com/>) 来存放他们的项目 包括本书的书稿, 位于 http://github.com/chromatic/modern_perl_book/。一个特别值得注意的 Github 页便是 Gitpan, 它存有记录每一个 CPAN 模块历史的 Git 代码库。参见 <http://github.com/gitpan/>。

活动

物理世界同样有着众多活动。Perl 社区每年举办不少会议、workshop 和研讨会。特别地, 社区运营着 YAPC —— Yet Another Perl Conference (“又一次 Perl 大会”) —— 是一场成功的、地方性、低成本且在多个大洲举办的会议模型。参见 <http://yapc.org/>。

Perl 基金会 Wiki 于 http://www.perlfoundation.org/perl5/index.cgi?perl_events 列举了其它社区活动。

同样还有本地 Perl Monger 小组 (Perl 小贩组?!) 不时地碰头举办技术讲座和社会交流。参见 <http://www.pm.org/>。

IRC

当 Perl 贩子们不出现在本地会议或大会上或 workshop 中, 他们中的许多人会互相协助并在 IRC —— 一种英特网早期发展起来、基于文本的群组聊天系统 —— 上聊天。不少最为出名且有用的 Perl 项目拥有自己的 IRC 频道, 例如 #moose 和 #catalyst。

Perl 社区的主 IRC 服务器是 <irc://irc.perl.org/>。其他值得一提的频道包括 #perl-help —— 为一般性 Perl 编程求助而设, 以及 #perl-qa —— 为测试和质量问题专用。注意 #perl 不是一般性的求助频道, 相反, 它是一个为讨论其参与者想讨论的任何事情而设的通用聊天室。 而且它对提出编程基础问题的人通常不十分友好。

CPAN

Perl 5 是一门实用的语言。它帮助你完成工作。恒久务实的 Perl 社区对语言进行了扩展, 并将成果供给全世界使用。如果你有问题需要解决, 别人很可能早就为此编写了代码并把它传到了 CPAN 上。

一个现代化的编程语言和它的库之间的界限是模糊的。是什么成就了一门语言? 区区语法吗? 还是核心库? 抑或其外部库的可用性, 和因在自己的项目中使用它们而带来的方便?

不管对于其他语言你如何回答上述问题, 现代化 Perl 编程大量用到 CPAN (<http://www.cpan.org/>)。CPAN, 或 Comprehensive Perl Archive Network, 是一项为重新分发可重用 Perl 代码而设的上传、镜像系统。它是全世界库代码量最大的存档, 如果不是, 加上“之一”。

CPAN 镜像模块的发行版，意图成为可重用 Perl 代码的总集。单一发行版可能包含一个或多个模块，或是自包含的 Perl 库代码。每一个发行版生存于 CPAN 上各自的名称空间并含有自身的元信息。你可以对每一个发行版进行构建、安装、测试和升级。发行版也可能互相依赖。为此，从 CPAN 客户端安装这些发行版会比手动安装要容易得多。

CPAN 每月增加数以百计的注册贡献者，和诸多发行版内数以千计的被索引的模块。这些数字还没有将更新算在内。2010 年八月末，search.cpan.org 报告了 8396 个上传者，85146 个模块，以及 20824 个发行版。

CPAN 自身仅是一个镜像服务。作者们上传包含模块的发行版，CPAN 将它们送至镜像站点，用户和 CPAN 客户端可以从那里下载、配置、构建、测试和安装发行版。CPAN 的成功应归功于其简洁，还有数以千计的志愿贡献者基于 CPAN 创造更大的价值。尤其，社区标准经过革新，能够识别一组织精良的 CPAN 发行版的属性和特点。这些标准包括：

- 某安装能与 CPAN 自动安装工具协作的标准；
- 描述每个发行版所含内容及其依赖的元信息标准；
- 描述发行版功能和使用条款的文档及许可标准。

额外的 CPAN 服务提供了综合自动测试和报告每个 CPAN 发行版是否遵循打包和分发要求以及在各式平台、不同版本的 Perl 下是否工作正确。每个 CPAN 发行版在 <http://rt.cpan.org/> 上有各自的单据队列，供缺陷报告以及和其他作者合作之用。发行版的历史遗留版本、评分、文档注解和其他有用的信息同样可以从 CPAN 获取。所有这些都可以通过 <http://search.cpan.org/> 得到。

摩登 Perl 安装版包含两个连接、搜索、下载、构建、测试和安装 CPAN 发行版的客户端，CPAN.pm 以及 CPANPLUS。他们行为相仿，使用何者只是喜好的区别。本书推荐使用 CPAN.pm 仅由于其普遍性。

如果你使用 CPAN.pm 的最近版本（截止写作时，1.9402 是最新的稳定版），CPAN.pm 的配置无须你做过多决定。对于任一完整的 Perl 安装来说，你可以以如下方式启动此客户端：

```
$ B<cpan>
```

安装一个发行版：

```
$ B<cpan Modern::Perl>
```

由 Eric Wilhelm 编写的优秀的 CPAN.pm 配置教程（<http://learnperl.scratchcomputing.com/tutorials/configuration/>）值得反复阅读，特别是问题排除那一小节。

即便 CPAN 客户端是 Perl 5 发行版的核心模块，你还是需要安装例如 make 之类的标准开发工具还可能外加一个 C 编译器来安装任何你想安装的发行版。Windows 用户请参考 Strawberry Perl (<http://strawberryperl.com/>) 和 Strawberry Perl Professional。Mac OS X 用户需要安装“开发者工具”。Unix 和类 Unix 用户请咨询你本地的系统管理员。

作为对你设置 CPAN 客户端和相应环境来构建和安装发行版的回报，你得到了可以完成任何任务的代码库的访问权——从数据库访问到能处理现存几乎所有网络设备的协议分析工具再到声音和图像处理库以及你系统上共享库的包装器。

没有了 CPAN 的摩登 Perl 仅是另一种平凡的语言。有了 CPAN 的摩登 Perl 则是惊人的。

认真的 Perl 开发人员通常会管理他们自己的 Perl 库路径甚至是完整的安装版。若干项目有助达成这些目的。

App::cpanminus 是一个新兴的 CPAN 客户端，以速度、简洁和零配置为目标。安装它简单到：

```
$ curl -LO http://xrl.us/cpanm
$ chmod +x cpanm
```

App::perlbrew 是一个可以安装管理多版本 Perl 的系统，允许你在版本和配置之间切换。安装它简单到：

```
$ curl -LO http://xrl.us/perlbrew
$ chmod +x perlbrew
$ ./perlbrew install
$ perldoc App::perlbrew
```

The local::lib CPAN 发行版让你可以在自己的用户目录下安装和管理各类 CPAN 模块，而不必将它们一股脑儿装进系统。这是既维护 CPAN 发行版又不影响其他用户的有效方法。只是安装过程较前两个复杂。参见 <http://search.cpan.org/perldoc?local::lib> 以获得更多细节。

这三个发行版项目都倾向于假设它们会用于类 Unix 环境（比如某个 GNU/Linux 版本甚至是 Mac OS X）之下。Windows 用户可以参考 Padre 一体化下载（<http://padre.perlide.org/download.html>）。