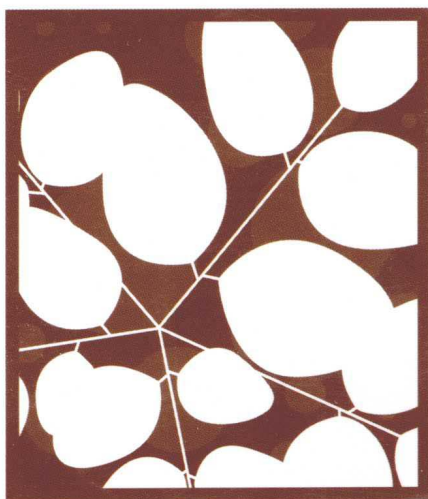


Broadview®
www.broadview.com.cn

诗意的边缘

PHP 顶级框架 Zend Framework 开发实战

欧雪冰 著



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

诗意的边缘

PHP顶级框架 Zend Framework 开发实战

欧雪冰 著

电子工业出版社

Publishing House of Electronics Industry

北京•BEIJING

内 容 简 介

本书通过一个完整 Web 站点的开发实例,全面讲解企业级框架 Zend Framework 从开发环境设置到框架安装、配置,再到网站开发、部署的全过程。全书紧紧围绕着一个包括 CMS、社区和博客的 Web 应用实例,循序渐进,带领读者突破重重阻碍,步入 Web 开发的前沿技术领域。本书可帮助那些希望借助框架进行开发的读者顺利地熟悉 Zend Framework 的基本结构、规范和开发流程,轻松掌握常用的 Zend Framework 组件,敏捷、稳健地开发 Web 2.0 应用程序。

本书适合那些具备 PHP 开发基础,进而希望学习使用框架来提升开发能力的读者。也适合那些掌握了 PHP 面向过程的开发方式,正在转向 PHP 面向对象编程的学习者,通过対 Zend Framework 框架的学习,可以更加迅速、规范地掌握 MVC 架构和面向对象的思想 and 语法。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有,侵权必究。

图书在版编目(CIP)数据

诗意的边缘: PHP 顶级框架 Zend Framework 开发实战 / 欧雪冰著. —北京: 电子工业出版社, 2012.6
ISBN 978-7-121-17127-7

I. ①诗… II. ①欧… III. ①PHP 语言—程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字(2012)第 101809 号

策划编辑: 孙学瑛

责任编辑: 董 英

印 刷: 北京东光印刷厂

装 订: 三河市皇庄路通装订厂

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编: 100036

开 本: 787×980 1/16 印张: 21.75 字数: 557 千字

印 次: 2012 年 6 月第 1 次印刷

印 数: 4000 定价: 59.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888。

质量投诉请发邮件至 zlts@phei.com.cn, 盗版侵权举报请发邮件到 dbqq@phei.com.cn。

服务热线:(010) 88258888。

这本书是写给哪些人看的

本书适合有一定 PHP 基础和开发网站经验的读者学习，以进一步提高网络应用程序的开发效率。Zend Framework 是 PHP 官方推出的重量级框架，拥有强大的功能和稳健的性能，因为完全采用面向对象思想和模式构建，所以相对于仅熟悉面向过程开发的 PHP 程序员来说显得比较难。本书通过一个完整的网络应用实例，带领读者轻松愉快地掌握该框架。

这不是一本关于诗歌的书，虽然名字叫《诗意的边缘》，但网络程序开发的诗意依托于程序代码，如果说本书除了讲解技术之外，还有一点“文学性”的话，那么这种“文学性”离开程序开发的过程，显得累赘、造作而枯燥，并无可读性。

这是一本 Zend Framework 的入门级实例教程，但不是 PHP 的，如果你没有学过 PHP，不曾了解过面向对象编程，这本书对你来说可能会有点难——Zend Framework 入门时的学习曲线陡峭是出了名的。但是，也有一种可能：本书也可能适合一些开发经验不多，但却有着使用各类开源程序来搭建网站的丰富经验的使用者，如果他们想进入开发领域，那么读这本书可能是一条捷径，他们可以绕过 PHP 面向过程的编码方式，直接从面向对象入手——我这么说可能会引起争议，因为在很多过来人看来，学好面向过程的原生代码是基本功课，这之后才能开始步入面向对象和框架的领域——我觉得凡事没有绝对，PHP 也在进化之中，我编写这本书的主要目的，就是希望一些刚开始接触网站开发的读者能够以面向对象的框架为起点，比较顺利地步入网站开发的行程。如果你是这样的一个新手，本书的过程讲解细致周全，学习起来不会太吃力，然而可能会有一部分内容对你来说显得比较难，需要你以很大的耐心去泡 Google、百度，做一些补习功课。

如果你胸怀大志，但既没有学过任何计算机语言，也没有接触过做网站这件事，那么这本书对你来说就很困难了，建议你还是从一些更为基础的书籍着手学习，如可以先学习 HTML，并尝试多了解一些 PHP 的开源程序和 PHP 的基础知识，对网络编程具备一定的了解之后，再来学习这本书。

框架的诱惑

尽管 PHP 社区对是否使用框架依然存在争论，但毋庸置疑的是，框架是一种诱惑，无论

你目前是否还在抵抗它。如果你还在抵抗那么可敬的 PHPer，请继续你快乐潇洒的纯 PHP 代码之路，如果你已决定投诚，那么勇敢的 PHPer，准备好迎接新的挑战吧！

在一些测试数据中，运用框架比运用原生 PHP 代码要慢很多，但是这些测试结果并不是实际生产环境数据的真实反映。事实上 echo "hello world" 的简单测试不具有实际意义，因为在实际使用中，即使是使用原生 PHP，开发者也免不了会创建一些自己的类库，并在运行时加载，再加上代码缓存、页面缓存等技术的综合运用，使得一些成熟框架在经过充分的性能优化后，与 PHP 原生代码相比，性能下降得并不像通常想象得那么大。

框架的产生就是原生代码演进到一定程度后的一个结果，可以说开发框架的诞生和发展是行业发展、技术水平提高的必然，使用开发框架将大大提高项目开发效率和促进行业内技术经验共享，从而更好地提升整个行业的开发效率和知识积累。因此选择开发框架可以说是大势所趋。

使用框架，可以使编程看起来不那么像体力劳动，你可以有更多的时间坐下来思考点什么。但这是指使用框架，不是指学习——学习还是一种繁重的体力劳动——直到哪一天你宣称你学会了，你才可以使用它。

那么，在众多的 PHP 框架之中，我们为什么选择 Zend Framework 呢？下面的分析或许会对你有所帮助。

Zend Framework 作为 PHP 的官方框架，拥有非常庞大的用户基础和世界顶级的开发人员支持，发展非常稳定，前景很好，但这并不表示它是唯一的、最好的选择。根据开发的需求来决定是否使用它吧！如果只是一个很小的应用并且以后不会有太多的扩展和维护，使用 PHP 原生代码就很好，使用框架反而降低效率。除了 Zend Framework，PHP 还有许多优秀框架可选，如 CodeIgniter、CakePHP、Yii、Symfony、ThinkPHP、QeePHP/FleaPHP 等。其中 ThinkPHP、QeePHP/FleaPHP 是国产的优秀框架，笔者使用的第一个框架就是 ThinkPHP，印象很不错。在这里也推荐一下 Yii 框架，它是一个身在美国的华人主导开发的，开发效率和运行效率都非常高，有很多令人耳目一新的功能。另外，开发网络程序除了代码级别的框架外，还有一些应用级别的框架可以选用，如 Drupal，它已是一个完善的 CMS，其核心就包含有论坛、博客等功能，你可以通过在此基础上写模块来扩展功能，可以省很多力气。但相比开发级的 PHP 框架而言，Drupal 在灵活度和程序的运行效率上要差一个量级（Drupal 7 版本变得很慢），而且对于 PHP 开发人员来说，学习起来也并不容易，甚至比以上所提到的某些开发框架还要难一些。

Zend Framework 的发展非常快，但国内 PHP 社区对它的看法似乎还停留在几年前。当听

说 Windows 8 中的应用程序将主要采用 HTML 5 和 JavaScript 语言来开发的时候,我意识到在未来,网络编程会以一种更加不可预料的发展趋势发展,编程语言需要飞速进化,以适应这个时代的需要。所以不管怎么说,更新知识、学习新技术都是必要的,对很多程序员来说,学习新技术本身就是一个不可抗拒的诱惑。

由于本书是讲 Zend Framework 框架的,所以下面主要介绍一下该框架的优点和缺点。

Zend Framework 的优点

1. 在所有框架里, Zend Framework 显得很强大,给人的印象总的来说就是功能强大、稳健、速度中等。要获得理想的执行效率,需要有好的设计,在开发过程中注意调优,对效率不高的部分,要想办法进行优化。
2. 代码严谨规范,是纯正的 OOP 框架,对 PHP 的侵入性较其他框架低,因为规范,所以适合多人开发,结合培训成本、协作成本、开发和维护成本等方面综合考虑,在大、中型应用中还是比较理想的开发框架,对于小型应用来说,门槛略高,效能问题突出,显得有点得不偿失。
3. 至于 Zend Framework 效率问题,使用当前最新版本在实际开发和运营中体会到的问题并不严重,实际运行效果要远好于网上传言。
4. 与其他框架一样, Zend Framework 使得应用程序的维护变得更容易。
5. 学好 Zend Framework,再去学其他框架会很容易,一般几天就可以上手。

Zend Framework 的缺点

没有任何事物是完美的, Zend Framework 的缺点也很明显。

1. 与某些轻量级的框架相比, Zend Framework 显得很臃肿,当然也有点慢(在最新的版本中已大为改观)。Zend Framework 肥大臃肿的印象,很大程度上来自于它的下载压缩包的大小,而实际上这个压缩包里有许多不属于框架本身的示例、文档和扩展,你可以选择精简版的压缩包进行下载。另外框架中许多不用的组件是可以删除的,由于 Zend Framework 设计的松耦合性,除少数的一些核心组件外,删除某些组件对框架的使用没有影响。
2. 学习曲线陡峭,入门比较难。对大多数人来说,这确实可以算是一个很大的缺点,这

也是影响它在国内普及的主要因素之一。Zend Framework 的文档大而全，但却并不友好，显得太理论化了，难以理解，与实际应用之间的差距较大，而相关的优秀中文教程书籍稀缺，这无疑使学习 Zend Framework 变得更加困难。不过这也是本书出现的原因，希望读者朋友们看过本书之后，对这种状况的看法能有一些改观。

最后我想说的是，如果一个框架已经完美了，也就没有对它怀以梦想的乐趣了。相信随着语言的进化，以及硬件条件的变革（如云服务的使用），在不远的将来，Zend Framework 引起纠结的这些问题会逐渐淡化，被新的问题所取代。

保持简单

简洁是智慧的灵魂，冗长是肤浅的藻饰。——莎士比亚

很多人喜欢把事情搞复杂，而我喜欢 KISS (Keep It Simple, Stupid!) 原则：“保持简单而且傻瓜化”，本书的写作也以此作为指导思想。我将简单作为设计的首要目标，让初学者能够轻松入门。我提醒自己时刻记得自己编写这本书的初衷：自己在学习 Zend Framework 的过程中，看到很多把开发网络程序这么一件有趣的事情搞得无比复杂、枯燥的教程，有时真的像天书一样让人不明所以，经过反复琢磨弄明白之后，才发现那些难以理解的东西原来都应该是很容易理解的，于是我渐渐地想把这些东西理清楚，分享给为此深感痛苦的初学者们，以免他们像我一样走许多弯路。很多人不肯用心去学习和理解知识，而是人云亦云地用大堆的术语把自己和别人的脑袋搞乱，使人彻底丧失学习的信心和兴趣，这种不好的习惯在教育领域比比皆是，不知毁了多少人的求知欲，误了多少人的前程！

例如，在本书实例网站的开发中，在许多地方使用了静态处理，只要能静态写的就不用动态，能够直接用简单办法实现的，绝不为了所谓的优雅而绕很大的弯子。在网站的架构中，注重全局的概念完整性，即功能和复杂度应成比例，在此基础上安排结构和划分粒度。嗯，“粒度”，这是一个术语，好像有点复杂，不用管它了……

格式约定

本书采用了一些格式约定来帮助你的阅读。

1. 文字框：文字框分为命令框、代码框、说明框、警示等。

2. 粗体字代码：粗体字代码表示新添加的代码，以此与本已存在的代码区别出来，方便你识别新添加代码所处的位置。

代码提供

本书中所有的示例代码均可以在博文视点的网站下载，网址为 www.broadview.com.cn。对付费买本书的读者朋友们，我还将提供以下力所能及的服务。

1. Bug 的回馈和修正。
2. 对典型疑难问题的解答。

为了提供以上服务，作者在 CSDN 开设了博客：<http://blog.csdn.net/unityoxb>，关于本书的资讯均在博客中予以公布，一些置顶信息分别用于提供本书的代码下载、勘误和问题答复等，欢迎访问。

学习建议

以下是作者自己的一些学习心得，写在这里以供借鉴。

专心专注

学编程有点像修炼，心静神宁可以让学习更有效率，浮躁要不得。

不要贪多嚼不烂，一定要精于某一方面。

安排好时间

Zend Framework 内容庞杂，学好绝非一朝一夕之功，要有序有节奏，一天一天坚持下去才能有好的收获。

干这一行，熬夜似乎是免不了的，但即便非得熬，也要熬得科学，保证必要的睡眠，调节好生物钟，让作息规律一些。

善待自己，友爱他人

PHP 的学习过程是一件非常耗神的事，学进去之后，常常会废寝忘食，所以要注意劳逸结合，不要因在计算机前沉迷而忽略了与家人说说话或出去游玩的重要性。

每天在学习开始前，给自己倒一杯热茶或咖啡，并且记着喝它。

有时翻翻官方手册

你可以不时地去官网看看有什么变化，如有一个思路对头的文档作者重写了手册，或者哪位翻译大家把手册译得适合阅读了。官方手册试图告诉我们 Zend Framework 能做一切，但却没展示给我们如何去做，尤其是如其所宣称的那样“简单地”去实现某个实用的功能，但即便如此，官方手册到目前为止毕竟还是唯一“权威”的技术文档，我们还是得花点时间去啃一啃。

If you tell me, I'll soon forget. If you show me, I'll remember forever. —Chinese Proverb

这是从一本讲 Dojo 的书的开头摘来的，落款说是中国谚语——耳听为虚，眼见为实？我更喜欢直译：如果你告诉我，我很快就忘了，如果你展示给我，我会永远记得。

开发工具

本书中的实例在 LAMP 平台上进行开发，测试浏览器采用 Firefox 5.0+，并没有在更多的浏览器上进行测试，根据经验，本书中的代码在 IE 7+、Chrome 等浏览器下运行应该没有严重的问题，但在 IE 6 下，可能会有一些不正常的显示，主要是由 CSS 兼容性引起的，由于这不是本书要涉及的内容，所以没有对其进行专门的修正，因此建议大家在学习中，最好避免使用 IE 6，以免不必要的问题干扰学习。

第 1 部分 开发前的准备工作

第 1 章 创建开发环境	2
1.1 配置开发环境	3
1.1.1 Windows 系统下 XAMPP 集成软件包的下载和安装	4
1.1.2 Linux 系统下 Apache、MySQL、PHP 的安装	9
1.1.3 Zend Server CE 开发环境	11
1.1.4 phpMyAdmin 工具	11
1.2 模仿更真实的服务器环境	12
1.2.1 劫持一个域名	13
1.2.2 启用 Rewrite 模块	14
1.2.3 创建虚拟主机	14
1.3 开发工具	16
1.3.1 Zend Studio	16
1.3.2 vim 编辑器	18
1.4 本章小结	20
第 2 章 安装并初步了解 Zend Framework 框架	21
2.1 Zend Framework 的下载和安装	22
2.2 创建 Zend Framework 项目	23
2.2.1 使用 ZF tool	23
2.2.2 使用 Zend Studio	30
2.3 了解 Zend Framework 的 MVC 结构	33
2.3.1 Controller 控制器	37
2.3.2 Model 模型	38
2.3.3 View 视图	38

2.3.4 路由和分发规则	39
2.4 本章小结	41
第 3 章 layout 模板布局	42
3.1 准备整站 CSS 文件	43
3.2 理解 layout	43
3.3 开启 layout 布局模板	44
3.4 规划 layout	45
3.4.1 应用不同的 layout	47
3.4.2 避免过度设计	47
3.5 在 Bootstrap 引导文件中添加资源	47
3.6 让布局更完整和符合标准	48
3.7 本章小结	49

第 2 部分 开发网站的前台

第 4 章 基本页面的实现	52
4.1 准备好数据库	53
4.1.1 数据库的设计	54
4.1.2 创建数据库	55
4.1.3 添加示例数据	56
4.1.4 在 application.ini 文件中配置数据库信息	56
4.2 显示一篇新闻文章	60
4.2.1 创建文章模型和方法	61
4.2.2 创建新闻文章控制器	64
4.2.3 创建新闻频道首页视图	65
4.2.4 改进模型，让它适应更复杂的查询条件	66
4.2.5 改进 NewsController 控制器	67
4.3 显示一个新闻文章列表	68
4.3.1 扩充 Page 模型，添加 getPages 方法	69
4.3.2 扩充 News 控制器	70

4.3.3	使用 <code>partialLoop</code> 助手输出循环	71
4.4	创建文章详细信息页面	72
4.4.1	创建 <code>PageController</code> 控制器并添加 <code>detail</code> 方法	73
4.4.2	创建 <code>detail</code> 页面视图	74
4.4.3	在 <code>detail</code> 页面添加其他文章的链接	74
4.5	本章小结	76
第 5 章	对页面进行分类和导航	78
5.1	页面组织和分类的数据库设计	79
5.1.1	设计分类数据库	79
5.1.2	创建分类数据库并添加示例数据	80
5.2	创建 <code>Category</code> 模型	81
5.3	创建 <code>CategoryController</code> 分类控制器	82
5.4	输出分类到视图	83
5.5	创建新频道	84
5.5.1	准备一些数据	85
5.5.2	创建 <code>About</code> 控制器	85
5.5.3	创建 <code>About</code> 视图	85
5.6	创建导航菜单和面包屑路径	87
5.6.1	了解 <code>Zend_Navigation</code> 组件	87
5.6.2	编写 XML 文件	88
5.6.3	在 <code>bootstrap.ini</code> 引导文件中进行配置	89
5.6.4	在 <code>layout</code> 布局模板和视图中调用	90
5.6.5	结合 <code>JavaScript</code> 代码完成导航栏制作	90
5.7	本章小结	92
第 6 章	用户注册、登录和注销	93
6.1	准备用户数据表	94
6.2	实现用户注册功能	95
6.2.1	创建注册表单	95
6.2.2	处理表单	101

6.2.3	增加用户页面	105
6.2.4	用 E-mail 通知用户注册成功	107
6.2.5	避免同名用户	108
6.3	实现用户登录和注销	110
6.3.1	创建登录表单	111
6.3.2	使用 Zend_Auth 组件实现登录	112
6.3.3	在 layout 中构建用户面板	115
6.4	注销账户	118
6.5	账户管理	119
6.5.1	修改账户页面	119
6.5.2	更新用户信息	120
6.5.3	修改密码	126
6.5.4	忘记密码的处理方法	127
6.6	本章小结	131
第 7 章	博客	132
7.1	准备数据库	133
7.1.1	添加博客分类数据	133
7.1.2	创建 Tag 标签相关数据表	133
7.2	发表博客	134
7.2.1	创建博客表单	134
7.2.2	为表单扩展 Page 模型	140
7.2.3	实现博客 Tag 功能	143
7.2.4	博客表单的处理	145
7.3	博客的显示	147
7.3.1	博客首页	147
7.3.2	使用 Zend_Paginator 进行分页显示	148
7.3.3	博客内容页面	154
7.4	博客的更新与删除	157
7.5	为博客添加评论功能	162
7.5.1	创建数据库	162

7.5.2	评论表单	162
7.5.3	在控制器中增加验证码	164
7.5.4	评论表单的处理	166
7.5.5	显示评论	170
7.6	本章小结	171
第8章	社区	172
8.1	数据库设计和准备工作	173
8.2	更换布局	176
8.2.1	创建新的论坛布局模板	176
8.2.2	控制器中使用新布局文件	177
8.3	社区的模型	177
8.3.1	ForumCategories 模型	178
8.3.2	Forum 模型	178
8.3.3	ForumTopic 模型	179
8.3.4	ForumPost 模型	182
8.4	社区首页	185
8.4.1	首页控制器	186
8.4.2	视图	187
8.5	发表主题	188
8.5.1	表单	188
8.5.2	控制器	189
8.5.3	视图	190
8.6	为表单添加所见即所得编辑器	191
8.6.1	安装和配置	192
8.6.2	在视图中部署	204
8.6.3	用装饰器修正错误	205
8.7	为编辑器添加图片上传功能	206
8.8	板块页面	208
8.8.1	控制器	208
8.8.2	视图	209

8.9 主题页面	212
8.9.1 控制器	212
8.9.2 视图	213
8.10 回复主题	217
8.10.1 控制器	217
8.10.2 视图	218
8.11 帖子管理	218
8.11.1 修改模型	218
8.11.2 主题的更新和删除	220
8.11.3 回复的更新和删除	221
8.12 论坛、博客和用户页面的整合	223
8.12.1 User 控制器	223
8.12.2 用户 account 视图	224
8.13 本章小结	226

第 3 部分 开发网站后台和其他特殊功能

第 9 章 框架结构和后台管理模块的创建	228
9.1 什么是模块	229
9.2 创建 admin 模块	230
9.3 模块 Bootstrap 引导文件	231
9.4 为 admin 模块规划模板	232
9.4.1 模板文件	232
9.4.2 给 admin 应用新模板	233
9.5 启用 Dojo 框架	236
9.5.1 部署 Dojo	238
9.5.2 启用 Zend_Dojo 组件	239
9.6 本章小结	242
第 10 章 实现后台管理功能	243
10.1 管理分类	244

10.1.1	分类管理的准备工作	245
10.1.2	后台分类首页	246
10.1.3	创建分类	251
10.1.4	更新分类	256
10.1.5	删除分类	257
10.2	管理用户	258
10.2.1	后台用户首页	258
10.2.2	添加用户	260
10.2.3	更新用户	268
10.2.4	删除用户	270
10.2.5	用户信息页面	271
10.3	文章和页面的管理	274
10.3.1	文章列表	275
10.3.2	添加文章	276
10.3.3	编辑文章	281
10.3.4	删除文章	282
10.3.5	构建条件过滤表单	283
10.4	管理论坛	288
10.4.1	后台论坛模型	288
10.4.2	论坛分区	292
10.4.3	移动论坛分区位置	295
10.4.4	论坛板块	298
10.5	用自定义视图助手实现 RSS 调用	299
10.6	本章小结	303
第 11 章	用户认证、授权和管理	304
11.1	认识 Zend_Acl 组件	305
11.2	理解基本概念	306
11.2.1	开发 Acl 插件	309
11.2.2	在配置文件中开启插件	313
11.3	本章小结	313

第 12 章 网站的优化和部署	314
12.1 性能测试	314
12.2 使用 Zend_Db_Profiler 了解数据库查询情况	320
12.3 使用 Zend_Cache 缓存数据	322
12.3.1 Zend_Cache 缓存原理	323
12.3.2 配置文件	323
12.4 服务器空间的选择	324
12.5 Zend Framework 在服务器上的安装	325
12.6 重新修改配置文件	326
12.7 本章小结	326
附录 A Zend Framework 的组件索引	328