**MallBuilder技术文档**

[1.简介说明 2](#_Toc356577113)

[2.开发环境 2](#_Toc356577114)

[3.目录结构 2](#_Toc356577115)

[Api/结口目录 2](#_Toc356577116)

[4.系统架构 5](#_Toc356577117)

[**（1）**操作系统 5](#_Toc356577118)

[LAMP网站架构图 6](#_Toc356577119)

[Web服务器、缓存和PHP加速 6](#_Toc356577120)

[1、squid + Apache + PHP + eAccelerator 6](#_Toc356577121)

[2、nginx/Apache + PHP（fastcgi） + eAccelerator 7](#_Toc356577122)

[3、nginx + Apache + PHP + eAccelerator 7](#_Toc356577123)

[5.模板结构 7](#_Toc356577124)

[6.会员商铺模板制作 8](#_Toc356577125)

[7.新的模块制作 8](#_Toc356577126)

[8.二次开发规则 9](#_Toc356577127)

[《1》系统核心数据处理文件均放在includes文件夹中， 9](#_Toc356577128)

[《3》第三方插件lib 文件夹 9](#_Toc356577129)

[9.商城模块 10](#_Toc356577130)

[10.标签调用 11](#_Toc356577131)

# 1.简介说明

MallBuilder非免费软件，是商用软件，此文档为〔**机密文档**〕，在没有经过我们书面授权的前提下禁止一切传播复制活动，软件当中的任何代码不经我们书面许可不允许用到第三方软件当中，此文档的拷贝仅为合作商业用户二次开发参考之用。使用范围受主合同约束。未获商业授权之前，不得将本软件用于商业用途，不得对本软件或与之关联的商业授权进行出租、出售、抵押或发放子许可证，无论如何，即无论用途如何、是否经过修改或美化、修改程度如何，只要使用 MallBuilder 的整体或任何部分，未经书面许可，页面页脚处的 MallBuilder名称和网站（http://www.mall-builder.com） 的链接都必须保留，而不能清除或修改，使用者必须遵循GPL国际通用协议。

# 2.开发环境

开发语言为php,数据库为mysql,　Web服务器为apache或者iis,建议使用apache。

要求支持Rewrite,以获得更好的搜索引擎优化效果。

文当编码统一为UTF-8编码。Mysql字符集统一为UTF-8。

开发工具任选，editplus,ZEND,dremwaver以及您喜欢的任意一款开发工作，只是注意一下文档编码设为UTF8。

为了开发方便推荐一款集成安装包appserv，该款安装包安装非常简单，集成了apache+mysql+php以及开发所需的所有功能。

# 3.目录结构

Admin/为网站管理员目录，管理员的所有操作程序都写在此目录下面。

Api/结口目录

　　－ad.php 前台广告调用程序

　　－external\_call\_js.php　外部ＪＳ调用标签程序

　　－mail\_box.php　商铺中漂浮留言本程序。

　　－mail\_box\_js.php　商铺中漂浮留言本调用代码。

－uc.php　uc结口代码，ＤＺ的程序规定必须放此目录。

Cache 缓存目录，所有页面的代码缓存文件都分门别类的放在此处，里面的子目录不再作详细介绍。

Config 网站所有配置文件都放在此处，包括从后台生成的配置文件，某些文件从后台生成的目的是为了程序不去直接读取数据库，减少程序对库的读取次数，从而提高网站的负载能力。

Index\_cat 是类别字母索引文件。

Shop\_config 是会员自定义商铺格式的缓存文件。

　 Config.inc.php　是数据库的配置文件，其中包括数据库用户名和密码。

Cron\_config.php 是系统计划任务的配置文件，由管理员后台提交任务的时候自动生成。

Nave\_menu.php 是网站的主菜单配置文件，管理员提交的时候会自动更新。

Point\_config.php　是积分配置缓存文件，管理员在后台更改积分机制的时候会自动更新。

Stop\_ip.php 阻止某个ＩＰ地址访问时的缓存配置文件。

Table\_config.php 是所有网站数据表的配置文件。

Uc\_config.php　是进行ＵＣ整合时产生的配置文件。

Usergroup.php 是前台会员组相关的缓存配置文件，后台更新的时候此文件会得到更新。

Watermark\_config.php 是水印相关的配置文件，由管理员后台生成。

Web\_con\_type.php是独立页的配置文件，由管理员后台生成。

Web\_config.php 是网站总体参数的配置文件。

Databackup　是管理员在后台进行数据库备份时生成的备份文件存放目录。

Image 网站图片目录，包括管理员前后台等所有图片。每个文件目录名和模板的目录名相符。

Includes 网站的核心处理文件存放目录。

Crons 系统计划任务脚本存放目录。

Label 调用标签的处理层，负责从数据库调用出来，然后把取出来的结果送给模板，模板接收到数据后进行输出，关于模板如何接受如何处理将会在下面进行详细讲述。

Admin\_class.php 前台会员中心操作的基础类。

Arrcache\_class.php 是缓存类，此缓存类并非smarty当中的缓存类。是我们自己开发的。

bbs\_passport\_fun.php　ＤＺ６．０的整合结口函数。

Chinese\_class.php　是简繁转换的功能类库。

Cron\_inc.php 系统计划任务脚本的处理类。

Db\_class.php 数据库类。

filter\_class.php 非法词屏蔽

Function.php　公用函数库。

Global.php 系统总的集成文件，每一个页面都需包括此文件。

insert\_function.php 第一代标签的调用库，已不推荐使用，但还是可以用。方便。

lang\_class.php 多语言调用

Page\_calass.php 翻页类用于开启重写模式下的调用。

Page\_utf\_class.php 翻页类，用于非重写模式下的调用。

Plugin\_ 开头的所有文件为网站的主模板处理函数。

plugin\_blog\_class.php 会员分享函数

Plugin\_fav\_class.php 会员收藏夹的类

Plugin\_sms\_class.php　会员短消息类

Plugin\_temp\_class.php　商铺模板设置类

Plugin\_tradealter\_class.php　会员订阅类，提醒等。

Rand\_func.php 验证码处理函数。

Shop\_page\_class.php 用于商铺的翻页函数类，因为商铺的重写规则不一样。

Smarty\_config.php 是smarty类初始化配置的函数。

Static\_inc.php 是统计功能类。

Tag\_inc.php 标签的处理类。

User\_shop\_class.php 会员商铺类。

Install 是自动安装程序向导程序目录，一般就是第一次安装的时候使用，安装以后会自动生成一个isinstall.lock的文件，防止二次恶意安装，为了安全，之后建议将install目录删除。

Lang 是语言包目录，里面分有cn中文，en,英文，当然您也可以再增加其它目录。

Lib 是库文件，第三方公共用库。

　　Csv 是产品批量上传时候的cxcel模板。

Fckeditor 是国外一个著名的免费编辑器，由于功能强大，方便使用，多语言以及兼容性，很多公司都在使用。

Phpmailder　国外的一个邮件发送类，功能很强大，他的诱惑力对于做技术来说是不可抵挡的。

Smarty 现在ＰＨＰ的ＭＶＣ开发模式，几呼无一例外都在使用它，对互联网来说功不可莫。

Upload 一个很牛的人开发的一个flash 大文件上传组件主要是用作视频上传。

Big5-gb.table 繁简对照库二进制形式的，原因是更小。

Tinyipdata.dat 是一个ＩＰ地址库，很全，很小，读起来很快，二进制的。

Script javascript集中营，网站上所有用到的ＪＳ都存放在此处。

Templates 网站所有模板存放位置。

Templates\_c　模板编译目录，在开发过程中不用理会，可以在管理员后台－》工具－>清空、缓存中对此编译文件进行清空。

Tools 网站数据信息清理

Uc\_client ucenter的整合文件目录，他们要求这样子放的。只能放在跟目录，目录名必须一致。

Uploadfile 是网站上所有上传的附件存放地。

.htaccess 是针对apache服务器的地址重写文件。

404.php 是自定义的４０４页面。

Aboutus.php 是网站底部相关连接的调用文件。

Ads\_services.php 是**广告服务说明页**

Ajax\_back\_end.php是ＡＪＡＸ的处理后台

Change\_link.php 友情连接交换页。

City.php 是ajax后台，包括无刷新切换城市等。

Company\_list.php 是公司库列表页。

Company\_category.php 公司频道页。

Footer.php　页尾的页面，所有页面都包括。

Home 个人主页调用与处理

Index.php　索引主页，做分发用。

Login.php 登录页面。

Login\_statu .php登录类别

Lostpass.php 找回密码页。

Main.php 前台用户后台的调用与处理。

Member\_services.php 不同的用户组所拥有的权限列表页。

register.php 个人注册

rewiew\_detail.php 企业注册

Robots.txt　搜索引擎权限控制文档。

shop.php 会员商铺

sql.txt 数据字段升级

Search.php 高级搜索页。

Search\_result.php 高级搜索结果页。

Shop.php 会员商铺集中调用分发的页面。

Sub\_sity.php 是二级域名分站切换页。

Upload.php　上传。

Module 是模块目录。

Product 商城模块

　　　　　Album相册模块。

Announcement公告模块。

Brand 品牌模块。

Company 企业模块

Logistics 物流模块

Message 邮件模块

Others 收藏分享

Payment 支付系统的相关配置文件。

Promotion 促销模块

Report 举报功能模块

Special 专题模块。

Tg 团购模块

News 新闻模块。

Vote 投票模块。

# 4.系统架构

MallBuilder采用了国际最为流行的LAMP（Linux-Apache-MySQL-PHP）网站架构，该框架包括：Linux操作系统，Apache网络服务器，MySQL数据库，Perl、PHP或者Python编程语言，所有组成产品均是开源软件，是国际上成熟的架构框架，很多流行的商业应用都是采取这个架构（比如百度http://www.baidu.com/index.php,淘宝网 http://www.taobao.com/index.php等等），和Java/J2EE架构相比，LAMP具有Web资源丰富、轻量、快速开发等特点，微软的.NET架构相比，LAMP具有通用、跨平台、高性能、低价格的优势，因此LAMP无论是性能、质量还是价格都是企业搭建网站的首选平台。

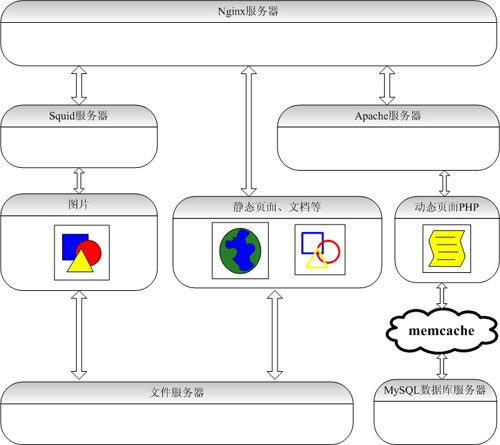
对于大流量、大并发量的网站系统架构来说，除了硬件上使用高性能的服务器、负载均衡、CDN等之外，在软件架构上需要重点关注下面几个环节：使用高性能的操作系统（OS）、高性能的网页服务器（Web Server）、高性能的数据库（Databse）、高效率的编程语言等。

**（1）**操作系统

　　Linux操作系统有很多个不同的发行版，如Red Hat Enterprise Linux、SUSE Linux Enterprice、Debian、Ubuntu、CentOS等，每一个发行版都有自己的特色，比如RHEL的稳定，Ubuntu的易用，基于稳定性和性能的考虑，操作系统选择CentOS（Community ENTerprise Operating System）是一个理想的方案。

　　CentOS（Community ENTerprise Operating System）是Linux发行版之一，是RHEL/Red Hat Enterprise Linux的精简免费版，和RHEL为同样的源代码，不过，RHEL和SUSE LE等企业版，提供的升级服务均是收费升级，无法免费在线升级，因此要求免费的高度稳定性的服务器可以用CentOS替代Red Hat Enterprise Linux使用。

LAMP网站架构图



Web服务器、缓存和PHP加速

　　Apache是LAMP架构最核心的Web Server，开源、稳定、模块丰富是Apache的优势。但Apache的缺点是有些臃肿，内存和CPU开销大，性能上有损耗，不如一些轻量级的Web服务器（例如nginx）高效，轻量级的Web服务器对于静态文件的响应能力来说远高于Apache服务器。

　　Apache做为Web Server是负载PHP的最佳选择，如果流量很大的话，可以采用nginx来负载非PHP的Web请求。nginx是一个高性能的HTTP和反向代理服务器，Nginx以它的稳定性、丰富的功能集、示例配置文件和低系统资源的消耗而闻名。Nginx不支持PHP和CGI等动态语言，但支持负载均衡和容错，可和Apache配合使用，是轻量级的HTTP服务器的首选。

　　Web服务器的缓存也有多种方案，Apache提供了自己的缓存模块，也可以使用外加的Squid模块进行缓存，这两种方式均可以有效的提高Apache的访问响应能力。Squid Cache是一个Web缓存服务器，支持高效的缓存，可以作为网页服务器的前置cache服务器缓存相关请求来提高Web服务器的速度，把Squid放在Apache的前端来缓存Web服务器生成的动态内容，而Web应用程序只需要适当地设置页面实效时间即可。如访问量巨大则可考虑使用memcache作为分布式缓存。

　　PHP的加速使用eAccelerator加速器，eAccelerator是一个自由开放源码PHP加速器，优化和动态内容缓存，提高了性能PHP脚本的缓存性能，使得PHP脚本在编译的状态下，对服务器的开销几乎完全消除。它还有对脚本起优化作用，以加快其执行效率。使PHP程序代码执效率能提高1-10倍。

　　具体的解决方案有以下几种：

1、squid + Apache + PHP + eAccelerator

　　使用Apache负载PHP，使用squid进行缓存，html或图片的请求可以直接由squid返回给用户。很多大型网站都采用这种架构。

2、nginx/Apache + PHP（fastcgi） + eAccelerator

　　使用nginx或Apache负载PHP，PHP使用fastcgi方式运行，效率较高。

3、nginx + Apache + PHP + eAccelerator

此方案综合了nginx和Apache的优点，使用Apache负载PHP，nginx负责解析其他Web请求，使用nginx的rewrite模块，Apache端口不对外开放。

**（2）**数据库

　　开源的数据库中，MySQL在性能、稳定性和功能上是首选，可以达到百万级别的数据存储，网站初期可以将MySQL和Web服务器放在一起，但是当访问量达到一定规模后，应该将MySQL数据库从Web Server上独立出来，在单独的服务器上运行，同时保持Web Server和MySQL服务器的稳定连接。

　　当数据库访问量达到更大的级别，可以考虑使用MySQL Cluster等数据库集群或者库表散列等解决方案。

　　总的来说，LAMP架构的网站性能会远远优于Windows IIS + ASP + Access（例如月光博客）这样的网站，可以负载的访问量也非常大，国内的大量个人网站如果想要支撑大访问量，采用LAMP架构是一个不错的方案。

综上所述，基于LAMP架构设计具有成本低廉、部署灵活、快速开发、安全稳定等特点，是Web网络应用和环境的优秀组合。

# 5.模板结构

系统所有模板均存放与templates目录，该目录下面又存放不同风格的模板，在templates/default文件夹中以admin\_开头的为会员办公室模板，以user\_开头的为会员商铺模板，以及系统前台的所有模板，单独风格的模板，除一些公用的模板外，存放与以风格名称命名的文件中，比如templates/blue简称蓝色模板，文件夹中就存放了一些模板和样式表page.css，需要用到的模板如果在blue文件中没有，会自动调用templates/default中的模板.以user\_开头的文件夹为会员商铺模板，在templates\_c编译目录中存在一个与风格模板文件夹同名的文件夹，用于存放改风格的模板编译文件。

Example:**如何制作MallBuilder模板**

第一，在templates/下面新建一个文件夹，用于存放模板及ＣＳＳ。  
第二，在templats\_c新建同名文件夹，此目录为模板缓存目录。  
第三，在image/下面新建同名文件夹用于存放图片。  
第四，进管理员后台－》系统设置－》选择你新制作的模板名即可前台预览使用。  
  
default模板为系统默认模板目录，此目录不可以删除，程序会自动搜索你自己创建的模板目录里面相对应的文件，如果找不到会自动从default里调用，在制作模板的时候可以参考我们其中一个模板案例如blue模板，里面只有少数几个文件，在大多数情况下您只需将header.htm,footer.htm文件重新定义制作就行，另外我们对模板结构中的ＣＳＳ名称进行了标准化的命名，您不需要更改html档，只需写ＣＳＳ就行了，如果您对某个页面有特殊需求，可以将此模板复制进您所制作的模板目录中，进行编辑修改。  
模板语言为smarty，可以参看smarty手册  
模板标签为　<{ 开头， }>结尾  
１.循环：<{foreach item=list from=$\*\*\*\*}> \*\*\*\*\*\*\*\* <{/foreach}>  
2. 判断如<{if \*\*\*\*}> \*\*\*\*\*\*\*\*\*\*\*\*<{/if}>  
3.变量显示为<{$\*\*\*\*}>

# 6.会员商铺模板制作

在templates文件夹中，以user\_开头，如user\_templates就是会员商铺模板，在商铺模板文件中，默认的只存放一个改模板的框架文件space\_temp\_inc.htm和一个存放该商铺模板的图片的文件夹img，其他使用的模板如果在该文件夹中不存在，系统会自动使用templates/default中以space\_ 开头的模板文件。

Example:

制作会员商铺模板时可以复制现在已有的会员商铺模板，重命名为user\_templates4，user\_templates5.......6,7,8等  
放入/templates/下面，同时在/templates\_c/下建立同名空文件夹，作为模板编译后存放的目录，如果是linux用户，确保该目录属性为777可写权限  
会员模板中space\_temp\_inc.htm这个文件最为重要，是商铺的框架，  
调整时注意标签变量不要删除搞错了，变量以<{开头，}>结尾。具体可以参看smarty手册.

# 7.新的模块制作

admin/管理后台。  
includes/类的存放目录。  
lang/语言包，里面可以分为多种，如cn,en,jp等等。  
templates/模板目录，里面有default,或者其它子目录，如果管理员后台定义的模板目录里面找不到模板文件，会取default里面相关的文件。  
table\_config.php每一个模块里面用到的数据表，要在里面配置一下。  
config.php 菜单配置项，如管理员后台菜单，和用户中心菜单。  
admin\_开头的PHP文件是用户中心发布和管理信息的文件

// 以下是另一个技术做的教程===================================

1.在网站的module文件夹下新建一文件夹,取名你的模块名（下面已 wuliu 为例）；  
2.在module文件夹下的找到video文件夹，把里面的文件复制一份到wuliu下；  
3.在wuliu文件夹下找到config.php文件.把里面的video都改成wuliu,中文字改成你想要的；  
4.在wuliu文件夹下找到table\_config.php文件中，user\_video 改成你数据库中表的名字（不加前缀），VIDEO随便改（区分大小写的），这是在你文件中用的表名；  
5.在wuliu文件夹下找到index.php文件,把include里面的module/video改成module/wuliu；  
6.在wuliu文件夹下找到admin\_video.php,把include里面的module/video改成module/wuliu；  
7.在wuliu文件夹下找到includes文件夹下的plugin\_video\_class.php文件，把sql语句中的VIDEO表名改为你在  
table\_config.php文件中改的名字；  
8.其余的就是根据你数据库中的字段来改plugin\_video\_class.php文件中的sql语句了。  
  
其余admin文件夹存放的是你的后台文件。  
image文件夹是存放你的图片的。  
lang是语言包。  
templates文件夹是存放你的前台展示的html页面。  
config.php文件中分两部分1.用户中心是个人用户后台；2.管理员后台。  
table\_config.php文件是文件中对数据库中表名的重命名。

# 8.二次开发规则

（一）数据存储层：

系统数据表统一定义名称在config/table\_config.php中，对照数据表的说明，可

以在数据表中添加和修改字段的属性

（二）数据处理层：

《1》系统核心数据处理文件均放在includes文件夹中，

global.php 为全局文件

db\_class.php封装了系统所需要的数据库处理类

founction.php 包含了系统常用的自定义函数

以\_class结尾的文件为栏目数据处理文件

crons文件夹为计划任务脚本存放文件

label文件夹为标签数据处理文件，负责从数据库读取数据

insert\_function.php为以前的系统标签处理文件

user\_shop\_class.php为会员商铺处理类

《2》语言包文件lang

系统语言包存放与lang文件夹中，目前里面有en,cn两个文件，在en和cn文件

中，存放了admin，front，payment，user\_admin，user\_space文件夹分别存放系统

系统后台管理，前台模板，支付结构，会员办公室和会员商铺的语言包，每个语

言表中的文件均与对应的文件名称相同，语言包文件中为数组的形式，如增加语言，

只需添加一个文件夹把以上文件夹放进去即可

《3》第三方插件lib 文件夹

该文件夹中存放了fck编辑器，phpmailer邮件类，smarty模板类，upload上传类，

本文件夹无需修改

《4》接口api文件夹

ad.php 为广告调用标签

external\_call\_js.php为站外调用标签

mail\_box.php，mail\_box\_js.php为会员商铺留言板文件

Uc.php为dz和ucenter的整合接口文件

《5》config文件夹

该文件夹用于存放系统的配置文件

Config.inc.php为数据库链接账号配置文件，可手动修改

cron\_config.php为计划任务配置文件，系统生成

point\_config.php为系统积分配置文件，系统生成

stop\_ip.php为ip阻止配置文件，系统生成

uc\_config.php为uc配置文件，系统生成

watermark\_config.php为水印配置文件，系统生成

web\_con\_type.php为站点独立页面菜单，系统生成

shop\_config文件夹，为会员的商铺配置文件，系统生成

Usergroup.php为会员组权限配置文件，系统生成

nav\_menu.php为站点导航菜单，系统生成

web\_config.php为站点全局配置，系统生成

table\_config.php为数据库表名称配置文件，可以手动修改

开发规则：

系统常用的数据处理类均存放于includes文件中，按照功能命名为

plugin\_功能模块\_class.php,具体操作实例主要分布在main.php，用于会员办

办公室数据处理，shop.php用于会员商铺数据处理。City.php为ajax查询，

Footer.php用于载入前台页面的语言包，载入独立页面菜单、版权和替换URL

Rewrite 路径。

在前台需要的地方修改/添加处理函数

在includes中修改/添加数据库处理类

修改数据表，添加需要的字段

在模板中修改/添加语言包中对应的变量

如添加新的模板或者说明文字，在语言包lang/cn和en中添加新的语言包文件

# 9.商城模块

|  |  |
| --- | --- |
| add\_address.htm | 添加收货地址 |
| admin\_buyorder.htm | 买家订单管理 |
| admin\_deliver.htm | 订单管理 |
| admin\_edit\_price.htm | 修改价格 |
| admin\_invoice.htm | 发票信息 |
| admin\_orderadder.htm | 收货地址编辑 |
| admin\_orderdetail.htm | 订单详情 |
| admin\_product.htm | 发布商品 |
| admin\_product\_batch.htm | 淘宝导入 |
| admin\_product\_cat.htm | 商品分类 |
| admin\_product\_list.htm | 商品列表 |
| admin\_product\_storage.htm | 仓库商品 |
| admin\_return.htm | 退换货 |
| admin\_sellorder.htm | 卖家订单管理 |
| cart.htm | 购物车 |
| comment.htm | 评论 |
| confirm\_order.htm | 结算 |
| product\_detail.htm | 商品详情 |
| product\_index.htm | 网站首页 |
| product\_list.htm | 商品列表 |
| shop\_cart.htm | 我的购物车 |
| shop\_footer.htm | 店铺底部 |
| shop\_header.htm | 店铺头部 |
| shop\_order.htm | 支付 |
| space\_product\_detail.htm | 举报中心 |
| space\_product\_list.htm | 店铺列表 |
| ucomment.htm | 商铺评价 |

# 10.标签调用

1.标签核心文件处理层，用于读取数据，并将结果送给不同的模板用于多样化的展示。存放于includes/label下面。文件名由　模块名\_label.php　组成。里面的function 名必须和文件名中的“模块名相对应。

如news\_label.php是文件名，里面的function名是news.

2.label模块模板表现层，每一个处理层会有不同的样式模板，用于展示从数据库里读出来的数据，放置在模板目录下的label下面，如果里面没有，会自动调用templates/default/label里面的相关文件。

３．如何调用。<{insert name="label" type='pro' temp='pro\_picprice' rec="1" limit=12}>

type为模块名。temp为模块调用的模板名。rec是推荐，limit是条数。显示全部在模板里完成，比如字数，是不是带图片宽度和高度，全由模板来完成，无需传参数。

固定参数名不变，所有标签都统一处理。：

name="label" 所有标签固定不变。

type模块名，现已包括  
brand,品牌

cat,类别

friendlink,友情连接

hottags,　热门ＴＡＧ  
news,　新闻

newscatnav,　新闻导航

newstag,　新闻标签

notice,　公告

pro,　产品

user。　用户

temp模板名。

rec推荐。

limit条数。

以上是现在可用参数，若要增加需再议，各开发人员不能擅自增加。

４．工作原理。

标签会首先调用insert\_function.php中的　insert\_label　的函数。这个函数用来分发调用不同的模块标签，进进数据库读数据和相关的一个逻辑处理。读回来数据据送给模板，连同模板一同完成表现层任务后将代码返回，返回的数据是已经被执行了的ＨＴＭＬ代码可以直接加入模板框加中应用显示。