

深拓网站信息管理系统

STWMS V1.0

用户手册

开发公司：云南诺玛阿美文化传播有限公司

开发者：谢春平

开发时间：2013.04-2013.11

官方网站：<http://www.stwms.cn>

目 录

目 录.....	1
一、 简 介.....	4
二、 功能特色.....	5
(一) MVC 三层模式设计.....	5
(二) 个性化定制管理系统界面.....	5
(三) 模板标签设计.....	5
(四) 全面支持页面静态化.....	5
(五) 自动缓存机制.....	5
(六) 多种数据库底层支持.....	6
(七) 管理系统核心数据分离式存储.....	6
(八) 权限机制.....	6
(九) 三级审核机制.....	6
(十) 管理系统菜单全面定制.....	6
(十一) 支持表单类型自定义.....	7
(十二) 强大的插件机制.....	7
三、 功能模块介绍.....	8
(一) 内容管理.....	8
(二) 栏目管理.....	8
(三) 附件管理.....	8

(四) 模型管理.....	8
(五) 推荐位管理.....	9
(六) 类别管理.....	9
(七) 用户管理.....	9
(八) 角色管理.....	9
(九) 角色菜单管理.....	9
(十) URL 规则管理.....	10
(十一) 表单类型管理.....	10
(十二) 系统菜单管理.....	10
(十三) 插件管理.....	10
(十四) 系统设置.....	10
四、 操作指南.....	12
(一) 安装.....	12
(二) 开始使用.....	12
(三) 数据管理.....	20
(四) 页面静态化.....	22
(五) 用户及角色管理.....	23
(六) 系统设置.....	27
(七) 扩展功能.....	30
(八) 插件管理.....	34
五、 模板及标签使用.....	35
(一) 模板语法规则.....	35

(二) 标签使用.....	37
(三) 系统常量.....	45
(四) 页面变量.....	46
(五) 重要全局函数.....	49
六、 二次开发.....	52
(一) 系统文件目录结构分析.....	52
(二) 数据库结构分析.....	53
(三) 二次开发流程.....	54
(四) 入口程序.....	55
(五) 系统类库和主函数调用.....	55
(六) 配置文件调用.....	56
(七) 控制器类开发规范.....	57
(八) 常见问题及使用技巧说明.....	59
七、 插件开发.....	63
(一) 编写第一个插件.....	63
(二) 文件部署.....	65
(三) 入口程序.....	65
(四) 编码规范.....	66
(五) 打包发布.....	66
八、 第三方手册.....	67
九、 版权说明.....	70

一、 简介

深拓网站信息管理系统 v1.0（简称 stwms v1.0），采用 PHP 语言开发，运用 OOP（面向对象）方式进行基础运行框架搭建。系统采用 MVC 三层模式设计，底层支持多种数据库类型（包括 mysql、mssql、access、sqlite2、sqlite3），系统框架易于代码维护、二次开发、功能和插件扩展。为个人网站、政府企业门户、商务平台网站、网络数据库应用系统开发提供快速安全的解决方案。

二、 功能特色

（一） MVC 三层模式设计

系统按照模型层、视图层和控制层，分离逻辑业务处理和数据显示层，为与底层数据交换提供一种更快捷的解决方案。

（二） 个性化定制管理系统界面

管理系统界面按照需求可以全面自定义，界面与内核程序分离，管理界面实现“一键式”切换。

（三） 模板标签设计

前台界面采用模板化标签设计，使设计人员和开发人员分工协作，提升工作效率。模板修改后系统自动编译模板并缓存，同时支持标签数据缓存。

（四） 全面支持页面静态化

页面静态化支持手动和自动模式，同时支持栏目和分类静态化、信息列表和内容分页静态化和自定义静态化规则。

（五） 自动缓存机制

采用独特缓存机制，首次使用自动获取数据并缓存，更新数据库

后系统自动更新缓存，更换网站域名或数据后无需手动更新缓存。

(六) 多种数据库底层支持

全面支持MySQL、MSSQL、ACCESS、SQLIE2和SQLITE3数据库，提供多种数据库解决方案，满足不同服务器数据库环境需求。用户可以在同一应用中同时使用不同类型的数据库，根据数据访问需求进行数据库分离式布局，全面提升数据库处理效率。

(七) 管理系统核心数据分离式存储

核心数据采用XML数据存储和访问，并生成生成缓存，避免频繁访问数据库，全面提升系统性能和安全性。

(八) 权限机制

系统管理用户根据角色不同，分别可以设置不同的管理权限，包括操作权限、栏目权限和插件权限。

(九) 三级审核机制

栏目内容支持三级审核机制（包括初审、复审和终审）。不同用户根据栏目审核层级设置，对栏目下数据内容具有不同的审核权限，需要审核的栏目只有经过设置级别的审核才可发布。

(十) 管理系统菜单全面定制

不同管理用户角色可以拥有自定义操作菜单。按照设定的操作管

理权限，系统自动生成对应菜单，生成的菜单可以调整和组合，全面实现个性化定制。

(十一) 支持表单类型自定义

系统内置**20**种表单类型，包括图片上传、文件上传、编辑器等功能，表单中集成输入数据验证和处理，同时支持用户自定义表单类型。

(十二) 强大的插件机制

系统除核心功能外，外围功能（如留言板、会员系统等）以插件的形式实现，插件支持“一键式”安装和卸载。同时用户可以安装自主开发的插件和在线安装插件。

三、 功能模块介绍

（一） 内容管理

包括内容检索、排序、次序显示、添加、编辑、删除、预览、审核、推送和生成静态化，各种功能根据栏目设置和用户角色权限提供相应的操作功能。

（二） 栏目管理

包括栏目排序、添加、编辑、删除，栏目有一般栏目、单页栏目、外链栏目三种类型。一般栏目应用于一般模型，具有分页和详细页；单页栏目应用于单一的页面，采用系统单页模型或一般模型；外链栏目是外链地址栏目。

（三） 附件管理

用户上传文件后，不同类别附件实现统一管理。包括附件地址更新、检索、删除、预览、统计和下载。

（四） 模型管理

模型对应于数据库系统，根据底层数据库操作 API 实现与实体数据库结构和数据同步，无差别支持多种数据库。包括模型添加、修改、删除、预览以及模型字段排序、添加、字段编辑和删除。

(五) 推荐位管理

将内容根据推荐位设置推送到不同的页面，推荐位操作包括排序、添加、编辑、删除以及推荐位数据管理。

(六) 类别管理

每个类别下面有多种分类，系统提供类别和分类的分别管理。类别管理包括类别的添加、删除、修改以及栏目应用，类别下分类管理包括排序、添加、删除和修改。

(七) 用户管理

包括排序、添加、删除和修改，用户审核权限设置。

(八) 角色管理

包括角色添加、修改、删除、权限设置和菜单设置，角色权限设置包括操作权限、栏目权限和插件权限，根据所设置的权限生成相应的角色菜单。

(九) 角色菜单管理

不同登录角色拥有完全不同的菜单，角色菜单全面自定义，角色菜单根据设置的角色权限设置，包括角色菜单的排序、移动、添加、修改、删除和重新生成。

(十) URL 规则管理

URL 规则用于页面静态化地址规则，包括规则的添加、修改和删除，栏目和详细页面 URL 规则有不同的变量，用户可以根据指定的变量自动设计地址规则，实现个性化访问地址。

(十一) 表单类型管理

表单类型即是用户添加和编辑界面操作控件，各种表单类型实现不同的功能，目前系统内置 20 种表单类型，同时用户可以添加、编辑和删除自定义表单类型。

(十二) 系统菜单管理

系统菜单为系统内置菜单，也即是超级管理员登录后使用的菜单，每个菜单分别对应于操作的控制器及方法名，系统菜单管理操作包括排序、添加、编辑和删除。

(十三) 插件管理

基于实现开放式插件功能设计，用户可以安装自主开发的插件或者在线安装插件，插件安装成功后会在系统插件主菜单下生成操作菜单，安装后的插件可以禁用、卸载和管理。

(十四) 系统设置

采用三种不同层级的设置，包括常规设置（网站设置）、基础设

置和高级设置。常规设置包括网站基础信息、SEO 及附件相关；基础设置包括操作界面、邮件及连接设置；高级设置包括系统管理入口、密钥、系统界面、数据库及安全配置。管理员可以根据需求分配给不同角色以操作权限。

四、 操作指南

（一） 安装

1. 开始安装前需要配置服务器 PHP5.0 以上运行环境，然后根据需求安装相关类型的数据库（具体安装过程请参考相关资料）；
2. 将程序安装包上传至网站根目录，同时设置 stwms/data、stwms/___cache___/、stwms/plugin/和 ufs/目录为可读写；
3. 在浏览器地址栏输入：http://你的域名/index.php?admin01，使用默认帐号和密码（帐号：admin，密码：admin）登陆系统（登陆系统后强烈建议修改帐号和密码）；
4. 首次登陆系统，系统自动检测服务器运行环境，并根据提示选择并配置网站可用的数据库类型、安装系统基础模型；
5. 根据需要可以在“插件管理”菜单中安装功能插件。

（二） 开始使用

推荐流程：整体策划 → 建立模型 → 前台模板制作 → 创建栏目 → 创建分类 → 创建推荐位 → 内容添加 → 内容预览。

1. 整体策划

根据建站需求，对网站功能及栏目进行整体策划，同时设计数据库表结构及字段类型。

2. 建立模型

操作流程：模型管理 → 添加模型（通常选择一般模型，包括一些系统字段，方便后期功能调用）→ 字段管理 → 添加字段。

模型添加图解如下：

The screenshot shows the 'Add Model' form with the following fields and annotations:

- 表名:** test (Input field) [✓ 输入正确]
- 名称:** 测试 (Input field) [✓ 输入正确]
- 描述:** (Text area)
- 应用到栏目:** ☒ 是 ☐ 否 (Radio buttons) [Annotation: 必须应用到栏目，在栏目才能选择此模型]
- 窗口设置:** 宽: 400 px 高: 200 px (Input fields) [Annotation: 用于设置表单窗口大小]
- 是否可用:** ☒ 是 ☐ 否 (Radio buttons)
- 模型类别:** ☒ 一般模型 ☐ 自定义模型 ☐ 系统模型 (Radio buttons)
- 说明:** 只有模型应用到栏目后，新建或修改栏目时才能选择模型；窗口尺寸不设置使用默认值；一般模型包括系统默认字段，自定义模型不包括系统默认字段 (Text area)
- Buttons:** 添加 (Add), 关闭 (Close)

字段添加图解如下（以分类表单类型字段为例）：

1)、基础信息设置

The screenshot shows the 'Add Field' form for 'Classification' type with the following fields and annotations:

- 字段类型:** 分类 (Dropdown menu) [✓ 输入正确] [Annotation: 根据字段的用途选择字段类别，目前有系统提供 20 种字段类别。]
- 字段名:** (Input field) [Annotation: 数字和下划线组成，并且仅能字母开头，不以下划线结尾]
- 字段别名:** (Input field) [Annotation: 填写字段名称，按规范填写。此字段名映射到数据库中表字段名]
- 字段提示:** (Text area) [Annotation: 在字段别名下方作为表单输入提示]
- Buttons:** 添加 (Add), 关闭 (Close)

2)、字段属性配置

用户添加和编辑界面字段的相关属性，不同字段设置不同，可以个性化定制显示效果

相关参数：
设置表单相关属性

在内容管理，列表显示
状态下字段显示方式

管理界面设置：
台管理界面显示方式

建立数据表及字段相
关参数

允许类别个数 只可以填入数字，同时可以使用多个类别，为0或空不限制

是否允许即时编辑 ☐ 是 ☒ 否 即时编辑允许在添加或编辑信息的同时编辑分类

是否显示至列表 ☒ 是 ☐ 否 是否排序 ☐ 是 ☒ 否 字符数 宽度 px 居

是否启用检索 ☐ 是 ☒ 否

是否允许索引 ☒ 是 ☐ 否 字段索引能够加快检索速度，但会降低添加和更新效率

是否唯一 ☐ 是 ☒ 否

是否为NOT NULL ☐ 是 ☒ 否

3)、样式、校验及权限设置

表单数据校验设置

表单样式名：

定义表单的CSS样式或类名

字符长度取值范围：

提交时检测数据长度范围是否符合要求，如果不想限制长度请留空

最小值：

最大值：

数据校验正则：

正则校验表单提交的数据合法性，如果不想校验数据请留空

常用

数据校验未通过的提示信息：

内置字段：

添加和编辑字段值，值由系统设定

☐ 是 ☒ 否

系统字段：

系统字段由系统创建，无法编辑和删除

☐ 是 ☒ 否

此功能仅为系统创始人提供，请谨慎使用

限制用户组：

选择的用户组无法设置此字段值

☐ 超级管理员 ☐ 网站管理员 ☐ 网站运营 ☐ 网站编辑

字段权限设置

3. 前台模板制作

操作流程：设计制作静态 PSD 页面 → 切片 → 使用系统标签制作模板。

模板命名规则：

- 1). 首页命名：index.html
- 2). 栏目页面命名：category_*.html;

- 3). 列表页面命名: list_*.html;
- 4). 详细页面命名: show_*.html 或 page_*.html,
- 5). 单页命名: page_*.html。

4. 创建栏目

操作流程: 栏目管理 → 添加栏目 → 选择模型 (模型确定表单内容) → 审核功能 → URL 设置 → 模板设置 → SEO 设置 (栏目搜索引擎优化)。

栏目创建图解如下 (以一般栏目为例):

1)、基本选项

基本选项URL设置模板设置SEO设置

选择模型: 图片 ☒ 输入正确

上级栏目: 图片 ☒

栏目名称: 新闻图片 ☒

英文目录: pictrue ☒

栏目图片:

描述:

表单方式: ☐ 内置 ☒ 对话框 ☐ 新窗口

在导航显示: ☒ 是 ☐ 否

内容审核: ☒ 无 ☐ 一级 ☐ 二级 ☐ 三级

此英文目录同时用于静态化生成目录

此栏目添加和编辑信息弹出的窗口模式, 大小在模型管理中配置

配置栏目的内容审核层级, 如无, 则添加的内容直接发布

2)、URL 设置

基本选项

URL 设置

模板设置

SEO 设置

当选择栏目页或者内容页为静态化时，生成静态页面的 URL 规则，规则配置在详见相关操作

栏目页：☒ 动态 ☐ 静态化

内容页：☐ 动态 ☒ 静态化

动态简化： ☒ 输入正确

内容页静态化规则：

目录型详细页(news/china/show/1_2.html)

自动静态化：☐ 是 ☒ 否 内容变动自动更新到静态页，较耗系统资源

3)、模板设置

选择风格：

720china

频道页模板：

请选择模板

列表页模板：

list_TimeQuanjing.html

详情页模板：

show_photos.html

应用到子栏目：☐ 是 ☒ 否

若无频道页模板，则使用列表页模板应用于频道页

4)、SEO 设置

SEO标题
栏目标题

图片的记忆-720°中国全景网-720china.cn

SEO关键词
多个用“,” 隔开

图片的记忆

SEO描述
栏目简介

时间的记忆

说明：SEO 标题、关键字和描述分别应用于模板页面 head 之间 title、meta 中 keywords 和 meta 中 description，页面分别变量为：{\$SEO[title]}、{\$SEO[keyword]}和{\$SEO[description]}

5. 创建分类（如需分类）

应用说明：如需应用栏目分类需要在建立模型时添加分类字段（typeid）,每个栏目可以应用不同的类别，但只能属于类别下的一个分类。

操作流程：类别管理 → 添加类别 → 选择需要分类的栏目 → 分类管理 → 添加分类。

分类创建图解如下：

1)、创建类别

名称：

描述：

时间，地点，事件，人物

说明：需要在栏目模型中添加分类字段，此栏目才会出现在右边，选择后此类别才能应用选择的栏目。

全选

栏目名称

首页

☐

— 首页背景图片

☐

— 首页图片切换

☐

— 全景头部背景图片

☐

— 论坛推荐

必须选择栏目，类别才会在该栏目中应用

2)、管理分类

ID	[顺序]名称 (点击名称进入编辑模式)	操作
30	0 长街宴	描述 删除
31	0 奥运会 添加子分类	描述 删除
32	0 国庆节	描述 删除
33	0 车展	描述 删除

添加一级分类

点击查看或编辑此分类的描述信息

点击分类名称进入编辑模式，点击“添加子分类”添加分类下子分类

6. 创建推荐位（如需推荐位）

应用说明：推荐位作用于模型或者栏目，独立于数据库实体数据存在，可以放在模板页面任何位置。

操作流程：推荐位管理 → 添加推荐位 → 选择应用模型和栏目（只有对应的栏目下该推荐位才会显示） → 相关参数设置 → 推荐位数据管理。

推荐位创建图解如下：

1)、创建推荐位界面

名称: 首页_图片推荐 ✔ 输入正确

模型: 图片 ▼

栏目: 时间轴 ▼

最大数据量: 6 ✔ 输入正确

溢出替换: 是 ☒ 否 ☐ 注: 溢出替换, 会导致超出限定数量的数据替换最先的数据

说明: 此处如同时未选择模型与栏目, 则该位为全站通用; 如选择模型未选择栏目, 则该推荐位在该模型下通用。

2)、管理推荐位数据

排序	ID	标题	模型	栏目	管理操作
3	6	网上展厅	全景	网上展厅	预览 移除
2	5	元阳哈尼梯田	全景	时间轴 (其他)	预览 移除
1	4	五大中心房屋出售	全景	房产家居	预览 移除
排序					

7. 内容添加和编辑

特别说明: 添加或编辑内容时, 参照数据输入提示, 严格按照规定数据格式输入, 输入的数据会经过表单验证。

操作流程: 管理内容 → 选择添加内容的栏目 → 点击“添加内容”按钮 → 填写内容 → 推荐位选择 (若需要) → 分类选择 (若需要) → 添加完成后保存。

数据添加或编辑图解 (此处为范例, 具体表单界面根据自定义设置不同):

提示：系统提供 20 种不同类型的表单，各种表单可以根据需要自定义显示方式，表单类型介绍详见模型管理。

8. 内容预览

添加或编辑后的内容可预览，查看发布效果，方便随时修改调整。

（三） 数据管理

1. 内容管理

内容管理显示方式分为列表显示或缩略图显示模式，用户可以自主选择显示模式；可以对内容进行检索，设置特定检索字段，包括按日期检索、是否推荐检索和按关键字检索，同时可以对检索结果排序；用户可用操作包括添加、修改、回收站（标记删除，可以恢复）、彻底删除、排序、重新审核（需要用户具有审核权限且栏目开启审核流）、内容推送（将数据内容复制到同模型栏目中）。

1)、列表显示模式（图解）

The screenshot shows the content management interface in list view mode. Red boxes and arrows highlight various components:

- 数据检索区** (Data Search Area): Located at the top left, containing a search bar and filters.
- 显示模式切换** (Display Mode Switch): Located at the top right, showing icons for list and thumbnail views.
- 标题栏及排序** (Title Bar and Sorting): Located at the top of the table, showing columns for sorting.
- 数据展示区** (Data Display Area): The main table area showing a list of content items.
- 批量操作区** (Batch Operation Area): Located at the bottom left, containing a '全选' (Select All) checkbox and a '排序' (Sort) dropdown.
- 单条信息管理** (Single Item Information Management): Located at the bottom right, showing actions for each item: '预览' (Preview), '编辑' (Edit), and '删除' (Delete).

排序	ID值	标题	点击量	添加时间	管理操作
<input type="checkbox"/>	0	26 元阳哈尼梯田	0	2013-10-11 11:38:11	预览 编辑 删除
<input type="checkbox"/>	0	24 五大中心	0	2013-10-11 14:53:48	预览 编辑 删除
<input type="checkbox"/>	0	23 网上展厅	0	2013-10-11 14:54:44	预览 编辑 删除
<input type="checkbox"/>	0	1 红河广场	2	2013-10-11 11:07:18	预览 编辑 删除
<input type="checkbox"/>	0	25 元阳哈尼梯田	0	2013-10-11 14:51:25	预览 编辑 删除
<input type="checkbox"/>	0	10 元阳哈尼梯田	0	2013-10-11 11:21:56	预览 编辑 删除

2)、缩略图显示模式



2. 内容审核

如果栏目需要审核，添加的内容会首先存入内容审核流中，根据栏目审核设置，包括初审、复审和终审，只有经过最后一级审核之后才会在前端显示出来。

操作界面（图解）：

<input type="checkbox"/>	ID值	栏目ID	标题	栏目名称	模型名称	当前状态	审核流	管理操作
<input type="checkbox"/>	25	8	元阳哈尼梯田	国家地理	全景	未审核	3级	预览 编辑 删除 永久删除
<input type="checkbox"/>	23	8	网上展厅	国家地理	全景	未审核	3级	预览 编辑 删除 永久删除
<input type="checkbox"/>	10	8	元阳哈尼梯田	国家地理	全景	未审核	3级	预览 编辑 删除 永久删除
全选	<div>初审 复审 终审</div>			三级审核操作				

3. 附件管理

所有由系统控件上传的文件都会进入附件管理中心统一管理，可以检索附件、切换显示模式、预览或下载、删除附件和地址批量更新。

操作界面（图解）：



提示：地址更新功能在附件上传目录修改后执行此操作，详见后面操作技巧说明。

（四） 页面静态化

1. 生成首页

将首页生成 HTML 静态页面，保存到站点根目录下。

2. 生成栏目页

将栏目页生成 HTML 静态页面，按照栏目页 URL 规则保存到相应目录，同时支持将分类也静态化，静态化过程中可生成分页；

操作界面（图解）：



提示：只有在栏目管理“URL 设置”中将栏目页 URL 类型设置为“静态化”，在此处才会出现该栏目名称，详细操作见栏目管理。

3. 生成内容页

将详细页生成 HTML 静态页面，按照栏目页 URL 规则保存到相应目录，静态化过程中如详细页中存在分页符，将会生成相应分页；

操作界面（图解）：



提示：只有在栏目管理“URL 设置”中将栏目页 URL 类型设置为“静态化”，在此处才会出现该栏目名称，详细操作参见栏目管理。

（五）用户及角色管理

1. 角色管理

用户角色为登录系统的角色，根据角色不同，可以配置不同的权限，权限分为操作权限、栏目权限和插件权限，然后系统根据配置权限生成此角色的操作菜单（生成后可以手动调整，进行个性化定制），此类角色用户登录后使用配置的菜单。

1)、角色管理操作界面（图解）：

排序	ID	角色名称	角色描述	管理操作
0	1	超级管理员	可以管理网站所有信息，拥有最大权限。	权限设置 栏目权限 插件权限 菜单管理 查看用户 修改 删除
0	3	网站管理员	负责管理网站所有内容。	权限设置 栏目权限 插件权限 菜单管理 查看用户 修改 删除
0	4	网站运营	负责网站运营	权限设置 栏目权限 插件权限 菜单管理 查看用户 修改 删除
1	2	网站编辑	负责网站内容的编辑工作。	权限设置 栏目权限 插件权限 菜单管理 查看用户 修改 删除

排序

2)、权限配置（图解）

a. 操作权限设置（点击权限设置进入）

☒ 站点内容

☒ 用户管理

☒ 系统配置

☒ 扩展功能

- ☐ URL规则管理
- ☐ 表单类型管理
- ☐ 系统菜单管理
- ☒ 插件管理

☒ 系统插件

选择操作权限对应菜单

点击保存后设置生效

保存

关闭

b. 栏目权限设置（点击栏目权限进入）

全选		栏目名称	查看	排序	添加	删除	修改	回收站	推送
<input type="checkbox"/>	图片		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/>	—记忆馆藏		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/>	—时间轴		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/>	全景		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/>	—国家地理		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/>	—风景区		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/>	—网上展厅		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/>	—房产家居		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
									<button>保存</button>
									<button>关闭</button>

c. 插件权限设置（点击插件权限进入）

插件操作菜单由插件中定义，按照插件时自动安装菜单

说明： 插件权限的设置需要开启“插件管理”权限。

3)、角色菜单管理

a. 角色菜单界面图解（点击角色管理->菜单管理进入，见下页）

选择	排序	ID	菜单名称	操作地址	管理操作
<input type="checkbox"/>	0	1	站点内容	—	添加子菜单 修改 删除 隐藏
<input type="checkbox"/>	1	30	└─ 管理内容	/index.php?admin01&c=content&a=init	添加子菜单 修改 删除 隐藏
<input type="checkbox"/>	0	155	└─	/index.php?admin01&c=content&a=manage&catid=	添加子菜单 修改 删除 隐藏
<input type="checkbox"/>	0	158	└─ 首页	/index.php?admin01&c=content&a=manage&catid=3	添加子菜单 修改 删除 隐藏
<input type="checkbox"/>	0	159	└─ 首页背景图片	/index.php?admin01&c=content&a=manage&catid=4	添加子菜单 修改 删除 隐藏
<input type="checkbox"/>	0	160	└─ 首页图片切换	/index.php?admin01&c=content&a=manage&catid=5	添加子菜单 修改 删除 隐藏
<input type="checkbox"/>	0	161	└─ 图片	/index.php?admin01&c=content&a=manage&catid=6	添加子菜单 修改 删除 隐藏
<input type="checkbox"/>	0	31	└─ 栏目管理	/index.php?admin01&c=category&a=init	添加子菜单 修改 删除 隐藏
<input type="checkbox"/>	0	34	└─ 生成首页	/index.php?admin01&c=create_html&a=public_index	添加子菜单 修改 删除 隐藏
<input type="checkbox"/>	0	35	└─ 生成栏目页	/index.php?admin01&c=create_html&a=category	添加子菜单 修改 删除 隐藏
<input type="checkbox"/>	0	36	└─ 生成内容页	/index.php?admin01&c=create_html&a=show	添加子菜单 修改 删除 隐藏
<input type="checkbox"/>	0	38	└─ 附件管理	/index.php?admin01&c=attachment&a=init	添加子菜单 修改 删除 隐藏
<input type="checkbox"/>	0	57	└─ 模型管理	/index.php?admin01&c=model&a=init	添加子菜单 修改 删除 隐藏
<input type="checkbox"/>	0	3	用户管理	—	添加子菜单 修改 删除 隐藏
<input type="checkbox"/>	2	17	└─ 用户角色管理		添加子菜单 修改 删除 隐藏
<input type="checkbox"/>	1	12	└─ 系统用户管理	/index.php?admin01&c=user&a=init	添加子菜单 修改 删除 隐藏
移动到 删除 排序 重新生成					

点击将会删除现有个性化菜单设置，然后根据配置的权限重新生成菜单

b. 添加或编辑菜单界面（图解）

上级菜单：

└─ 管理内容

菜单名称：

选择可用菜单

返回修改

类别：

自定义菜单

内部菜单

外部链接

配置：

类名

方法名

参数

显示图标：

是否显示菜单：

是

否

提供三种类型自定义菜单，可以使用系统内部菜单，然后修改菜单名称

菜单显示的图标，不同管理界面中图标各不相同

2. 用户管理

包括修改登陆系统的账号和密码，添加、删除用户、修改密码和配置用户内容审核权限功能。

第 26/71 页

1)、用户管理界面截图

排序	ID	用户名称	所属角色	最近登录IP	最近登录时间	E-mail	真实姓名	状态	管理操作
0	1	admin	超级管理员	127.0.0.3	2013-11-23 22:51:52	402085437@qq.com	蓝天	✓	修改 删除
0	2	master	网站管理员	127.0.0.5	2013-11-14 11:36:01	xiaomings@163.com	李明	✓	修改 删除
0	3	editor	网站编辑	127.0.0.5	2013-11-14 10:49:54	editor@nomaml.com	行者	✓	修改 删除
排序									

2)、用户信息添加或修改（图解）

用户名称:	<input type="text" value="master"/>	请输入用户名称
用户密码:	<input type="password"/>	密码为空，则不修改密码！
重复密码:	<input type="password"/>	请重复输入密码
真实姓名:	<input type="text" value="李明"/>	请输入用户真实姓名
E-mail:	<input type="text" value="xiaomings@163.com"/>	
是否锁定用户:	<input type="radio"/> 是 <input checked="" type="radio"/> 否	用户锁定后将无法登陆
用户角色:	<div>网站管理员</div>	设置用户的角色
审核权限:	<input type="radio"/> 无 <input type="radio"/> 初审 <input checked="" type="radio"/> 复审 <input type="radio"/> 终审	需要开启内容审核权限

设置用户对内容审核的权限级别

（六） 系统设置

系统按照设置对运行影响深度共分为三种层级：网站设置、基本设置和高级设置。网站设置为常规网站参数设置，基本设置包含对模板、用户界面及操作方式的设置，高级设置用于设置系统后台界面、安全性等。超级管理员可以根据需要将制定的设计权限赋予相应的用户角色，从而实现不同用户层级对系统不同的深度的设置，避免用户由于不确定操作对系统运行稳定性和安全性的影响。

1. 网站设置

1)、基本设置截图

网站名称:	<input type="text" value="720°中国全景网"/>	
网站域名:	<input type="text"/>	提示: 以http://开头
ICP备案号:	<input type="text" value="滇ICP备13002786号"/>	
统计代码:	<div><div></div></div>	
邮箱:	<input type="text"/>	
网站QQ:	<div><div>715400404技术支持</div></div>	

2)、SEO 设置 (同栏目 SEO 设置, 略)

3)、附件设置 (图解)

此处设置为全局设置, 表单中相关设置会覆盖设置, 表单中上传类型为全局类型的子集。

允许同时上传文件数量:	<input type="text" value="20"/>	单次上传文件允许的最大数量, 为0或空则不限制
允许上传附件大小:	<input type="text" value="20800"/>	KB
允许上传附件类型:	<input type="text" value="doc docx rar zip jpg jpeg png gif bmp xls xlsx ppt pptx swf"/>	

附件MD5值校验: ☒ 启用 ☐ 关闭 对上传的附件进行MD5校验, 防止重复上传

对上传的文件进行 MD5 校验, 重复值不上传, 从而节省服务器空间

GD库功能检测: 支持

上传图片缩放尺寸 宽 px 高 px 为0或空则不限制

注意: 若设置此值, 所有上传的图片会此值进行缩小处理

水印模式设置

☐ 关闭 ☒ 图片 ☐ 文字 水印预览

上传图片水印处理参数, 图片水印和文字水印有不同的设置选项, 具体参照系统运行界面

宽	<input type="text" value="300"/>	px	高	<input type="text" value="300"/>	px
<input type="text" value="mark.png"/>		上传图片		路径: {UPLOAD_PATH}watermark/	
水印透明度	<input type="text" value="85"/>	请设置为0-100之间的数字, 0代表完全透明, 100代表不透明			
JPEG 水印质量	<input type="text" value="80"/>	水印质量请设置为0-100之间的数字, 决定 jpg 格式图片的质量			
水印位置	<input type="radio"/> 顶部居左	<input type="radio"/> 顶部居中	<input type="radio"/> 顶部居右		
	<input type="radio"/> 中部居中	<input type="radio"/> 中部居左	<input type="radio"/> 中部居右		
	<input type="radio"/> 底部居左	<input type="radio"/> 底部居中	<input checked="" type="radio"/> 底部居右		

2. 基本配置

1)、基本设置（图解）

前台界面默认模板:	<div>720china default</div>	提示：前台界面模板位于 {stwmS}/template/styles/目录
前台默认静态文件路径:	<div>720china/</div>	前台访问静态文件路径，如果为相
静态文件生成路径:	<div>720china/</div>	当需要生成静态文件时， <div>静态文件的生成路径，相对于网站根目录</div>
异步显示栏目层级:	<div>0</div>	默认为0,即直接显示栏目列表。当栏目层级较多时，可设置首次加载
启用页面Gzip压缩:	<div><input type="radio"/> 是 <input checked="" type="radio"/> 否</div>	
开启附件状态统计:	<div><input checked="" type="radio"/> 是 <input type="radio"/> 否</div>	记录附件使用状态，删除文章时同步删除附件。注意: 本

2)、邮件发送配置（略）

3)、连接登录配置（略）

3. 高级配置

1)、管理配置（图解）

系统入口:	<div>index.php</div>	提示：系统唯一入口URL，相对于站点根目录,前面不包括"/"字符 <div>此为进入管理系统登录配置，不要随便修改，否则无法进入</div>
管理密钥:	<div>admin01</div>	提示：此密钥不要随便修改，否则会导致后台管理无法进入
管理系统风格:	<div>ldark</div>	提示：网站风格位于 {stwmS}/template/admin/目录，静态文件位于 /statics/admin/目录

2)、数据库配置（图解）

数据库连接类型: mysql 提示：使用中的系统请勿随意更换数据库类型，更换类型后需要重新安装模型，原有数据会被清空

数据源: localhost,3306 请参照范例填写

用户名: root 数据库连接用户名

密 码: root 数据库连接密码

数据库名称: china720

表前缀: 720_

是否支持永久连接: ☐ 是 ☒ 否

测试连接

此设置用于配置系统全局数据源，文件型数据库请确认数据库文件具有写入权限，非文件型数据库配置后点击“连接测试”测试数据库连接是否正常

说明：目前系统底层支持的数据库类型为 mysql、mssql、access、sqlite2 和 sqlite3；其中 mysql 和 mssql 类型为 mysql 和 mssql 时，是升级后的连接方式，支持事务处理。

3)、安全配置（图解）

启用后台管理操作日志 ☐ 是 ☒ 否

保存错误日志 ☐ 是 ☒ 否

错误日志预警大小 20 MB

后台最大登陆失败次数 8

登陆失败后重试间隔 600 秒

此处为当用户登录系统失败后，可以重试的次数和超过重试次数后需要等候多少秒后才能再次重试

（七） 扩展功能

1. 表单类型管理

系统内置了 20 种可以直接使用的表单类型，用户也可以添加和编辑自己的表单类型（如果非开发用户，不熟悉系统代码，建议不要添加自定义表单）。

1)、系统表单类型

ID	类型	名称	描述	数据库类型
1	author	作者	为此篇网文的作者字段	字符型
2	box	选项	提供各种选项，包括单选或多选	字符型
3	catid	单栏目	单栏目编号，发布的栏目号不可修改	小整型
4	catids	多栏目	可以发布到多个栏目，需要用户有推送权限	字符型
5	datetime	日期时间	日期时间类型	整型
6	downfiles	文件上传	可以上传一个或多个文件	字符型
7	editor	编辑器	网站编辑器字段	文本型
8	float	浮点型	用于常用货币计算以及需要小数计算的字段	浮点型
9	image	图片	单张图片上传	字符型
10	images	多图上传	多张图片上传	文本型
11	integer	带符号整数	整型数值，用于基本数据的类型	整型
12	integeru	不带符号整数	不带符号整数值	整型
13	keyword	关键字	用于meta关键字	字符型
14	smallint	小整型	用于记录各种状态值，可以为负值	小整型
15	text	单行文本	用于各种标题等，小型文本存储	字符型
16	textarea	多行文本	多行文本输入	文本型
17	title	标题	用于标题字段的存储,可以设置标题中词语的样式	字符型
18	classid	分类	存储系统分类ID信息	字符型
19	posid	推荐位	模型的推荐位	小整型
20	description	摘要	用于meta描述信息	字符型

2)、表单添加或编辑图解

表单的添加或编辑包括参数配置、添加模板、编辑模板、表单代码、输入控制、输出控制和更新控制：其中添加模板、编辑模板为原始代码，用于表单属性参数设置；表单代码、输入控制、输出控制和更新控制为类方法代码，保存后整合到相关的处理类中。

参数配置

添加模板

编辑模板

表单代码

输入控制

输出控制

更新控制

类型: float 输入正确

名称: 浮点型 输入正确

描述: 用于常用货币计算以及需要小数计算的字段

数据库类型: 浮点型

默认值:

类型设置:

不带符号: ☐ 是 ☒ 否

允许索引: ☒ 是 ☐ 否

是否可用: ☒ 是 ☐ 否

数据库字段类型映射设置，分别对各种数据库类型底层进行统一封装

2. URL 规则管理

定义页面静态化后，页面 URL 地址规则，也是静态文件在服务器端的生成路径。用户可以自定义 URL 规则，编辑和删除现有规则，URL 规则使用系统提供的变量访问，确定唯一文件地址路径，确保多栏目中多条信息彼此不相冲突。

操作界面（图解）

名称: 目录型列表页 输入正确

URL示例: news/china/10.html

URL规则: {\$pdir}{\$cdir}/index.html|{\$pdir}{\$cdir}/{\$page}.html

应用类型: ☒ 列表页 ☐ 详细页

是否可用: ☒ 是 ☐ 否

可用变量: 父栏目路径: {\$pdir}, 栏目路径: {\$cdir}, 父栏目ID: {\$pid}, 栏目ID: {\$cid}, 分类

根据选择的类别不同，在栏目管理->URL 设置->静态化设置中可以选择在此处设置的 URL 规则（具体操作步骤参照 I 栏目管理部分）

1)、列表页规则可用变量

父栏目路径: {\$pdir}, 栏目路径: {\$cdir}, 父栏目 ID: {\$pid}, 栏目 ID: {\$cid}, 分类

ID: { \$typeid }, 分页: { \$page } 。

2)、详细页规则可用变量

父栏目路径: { \$pdir }, 栏目路径: { \$cdir }, 父栏目 ID: { \$pid }, 栏目 ID: { \$cid }
年: { \$year } 月: { \$month }, 日: { \$day }, ID: { \$id }, 分页: { \$page } 。

3. 系统菜单管理

系统菜单为超级管理员登录后使用的菜单,也是系统可以使用的所有菜单,与操作的控制器和方法相关联。

1)、管理界面部分 (图解)

排序	ID	菜单名称	类型名	方法名	管理操作
0	1	站点内容	-	-	添加子菜单 修改 删除
5	42	└─ 审核内容	content	audit	添加子菜单 修改 删除
1	30	└─ 管理内容	content	init	添加子菜单 修改 删除
8	46	└─ 点击关闭子菜单	content	move	添加子菜单 修改 删除
7	45	└─ 系统菜单	content	push	添加子菜单 修改 删除
6	44	└─ 系统菜单	content	listorder	添加子菜单 修改 删除
1	43	└─ 系统菜单	content	uninstall	添加子菜单 修改 删除
0	127	└─ 禁用	plugin	disabled	添加子菜单 修改 删除
0	153	系统插件	-	-	添加子菜单 修改 删除
0	154	└─ 留言板	plugin	admin	添加子菜单 修改 删除
0	155	└─ 评论	plugin	admin	添加子菜单 修改 删除

系统菜单管理操作: 点击“添加子菜单”为当前前菜单添加子菜单, 点击“修改”修改当前子菜单; 点击“删除”则删除对应菜单

2)、添加或编辑菜单界面 (图解)

上级菜单: └ 插件管理

菜单名称:

类型名: plugin

方法名: disabled

附加参数:

显示图标:

是否显示菜单: ☐ 是 ☒ 否

类型名为控制器目录下除 api 和 admin 控制器外所有的文件名称，方法名为控制器中管理方法名（即是 m_前缀方法名称）

（八） 插件管理

插件管理中心提供统一的插件管理，可以安装本地开发的插件和在线安装插件。

1)、插件管理中心截图

ID	名称	别名	作者	安装目录	是否可用	管理操作
14	comment	评论	蓝天之云	comment_v1	√	禁用 卸载 进入管理
13	message	留言板	蓝天之云	message_v1	√	禁用 卸载 进入管理

2)、本地安装插件（图解）

强制安装: 是 ☐ 否 ☒ 若已经插件安装目录存在，强制安装会修改安装目录后安装

选择插件 请选择插件后，系统会自动安装，安装过程中请不要关闭此窗口！

说明: 安装包压缩格式为zip格式，为保证插件的正常运行，请保证插件按规定开发

点击“选择插件”，从电脑中选择开发的插件包（zip 格式），选择后系统会自动上传并按照插件，安装过程会有进度提示

3)、在线安装插件（略）

说明：为保证系统安全性和插件能够正常运行，请仔细检查插件来源；若为插件开发用户，请参考“插件开发”部分技术资料。

五、 模板及标签使用

（一） 模板语法规则

1. 变量表示

`{ $name }` 被解析成 `<?php echo $name; ?>`，表示显示变量 `$name` 的值，其中的“`name`”由英文字母、数字和下划线组成首字母必须是英文字母或者下划线。

2. 常量表示

`{ name }` 被解析成 `<?php echo name; ?>`，表示显示常量 `name` 的值，其中的“`name`”由英文字母、数字和下划线组成首字母必须是英文字母或者下划线。

3. 条件判断

`{ if * } * { else } * { else } * { /if }` 或者 `{ if * } * { /if }`，其中 `{ if * }` 中的 `*` 就是此判断语句的条件表达式，符合 `php` 的表达式。

4. 循环

`{loop $a $b} * {/loop}` 或者 `{loop $a $b $c} * {/loop}` , `{loop $a $b} * {/loop}` 被解析成 `<?php if(is_array($a)) foreach($a AS $b) { ?> * <?php } ?>` ; 而 `{loop $a $b $c} * {/loop}` 则被解析成 `<?php if(is_array($a)) foreach($a AS $b=>$c) { ?> * <?php } ?>`

5. 模板包含

`{template 'file name'}` 例如: `{template 'index'}` 表示包含位于默认风格模板目录下 `index.html` 文件;

`{template 'file name','dir name'}` 例如: `{template 'header','common'}` 表示包含位于默认风格模板目录下 `common` 目录中 `header.html` 文件;

`{template 'file name','dir name','style name'}` 例如: `{template 'header','common','default'}` 表示包含位于 `default` 风格模板目录下 `common` 目录中 `header.html` 文件。

6. 自增、自减

例如: `{php $i=10;} {loop $a $b} {$i++} {/loop}`。

包括: 自减 `{$i--}` `--$i` , 自增 `{$i++}` `++$i`。

7. 单行 PHP 解析语法

`{php $i=1;}` 解析为: `<?php $i=1;?>`

`{php $string = date('Y-m-d');}` 解析为: `<?php $string = date('Y-m-d');?>`

建议 if 和 loop 语法应该写在 html 注释符之间，这样可以避免在编辑工具中难看，例如：

这样写 if

```
<!--{if $a>$b}--> <!--{else}--><!--{/if}-->
```

这样写 loop

```
<!--{loop $arr $key $val}--> <!--{/loop}-->
```

提示：如果您不习惯这套语法，也可以直接在模板中书写 php 代码，例如<?php echo 'Hello World! ';?>

（二） 标签使用

使用标签是获取数据库中数据的方式之一，能够方便和快速地获取后台管理系统中的数据，并显示到页面中。

标签包括模块类标签和工具类标签：模块类标签与提供调用模块数据接口，根据调用动作的不同，结合相应的参数，实现模块数据的快速获取；工具类标签为系统提供的一些工具箱,目前包括 PLUGIN 插件工具、JSON 获取工具、XML 获取工具、GET 数据库 SQL 查询工具。

模块类标签和工具类标签与系统公共类及函数协作完成部分实用功能，如：数据分页、图片缩略图、文字字数剪切、数组转换、数据统计、获取各种栏目信息等。

1. PC 标签分析

系统标签{pc:}{/pc}，必须是以{pc}开头，并以{/pc}结尾的代码片段，用于获取系统数据。

1.1、获取数据：

```
{pc:content action="操作参数名" 参数名 1="参数值 1" 参数名 2="参数值 2" ...}  
{/pc}
```

- a. 系统标签中{pc:}冒号之后跟随的为模块名，如：{pc:content}；
- b. action="操作参数名"，系统为每一个模型都为其 PC 标签定义了调用的参数。其中有一些调用参数是系统保留的参数，其对所有的 PC 标签有效的；
- c. 同时系统为不同的模板页面保留了可以直接调用的公共参数。

1.2、显示数据：

{loop}{/loop}标签，用户循环输出 PC 标签获取到的数据，以{loop}标签开头，以{/loop}标签结束。每条数据将执行一次循环。

示例：

```
{loop $data $key $val}  
    <A href="{ $val[url]}">{ $val[title]}</A><BR>  
{/loop}
```

PC 标签保留参数（公共参数）

变量	默认值	说 明
action	null	本参数的值表示为操作事件，模型类 PC 标签必须使用包含本参数，以说明要进行的操作
cache	0	缓存存储时间（单位秒）

num	null	获取记录的条数，最后会被模板引擎处理成 limit 传送到处理函数中
page	null	当前分页参数
pagesize	null	分页数量
return	data	返回数据变量名称
start	0	Limit 开始的数值

2. 内容模块标签

模块名：**content**

操作：action= “可用操作” 例如：action= “lists”

操作名	说 明
lists	内容数据列表
position	内容推荐位列表
hits	内容数据点击排行榜
classify	分类数据

2.1、操作：lists 标签处理函数，获取数据

可用参数

参数名	默认值	说 明
catid	0	栏目 ID,必须填写
typeid	0	标签所属分类 ID,可以不用设置（默认为 0，即获取顶级分类）、为数值或\$typeid 变量（此种方式会随着提交的 typeid 参数值变化而改变父级分类）
where	null	查询条件，默认为空，即查询 catid 栏目下的所有数据，格式如：“id>10” 或 “id>10 and name like '%li%’”
num	null	数量的限制，数字，默认为空获取所有内容
start	0	开始条数，数字，从第一条开始

order	id desc	排序，格式如：“listorder desc”或“listorder desc,id asc”
format	1	是否进行数据输出格式化，设置为 0 直接调用数据库中的原始数据
thumb	null	只获取具有缩略图的数据，如果返回的字段不包括“thumb”或“thumb”为空，则从所有数据中搜索
images	0	是否强制获取所有图片，将返回“allimages”字段，此字段包含了所有图片的数组

示例：

```
{pc:content action="lists" catid="2" order="id DESC" page="$page" num="4" start="3"
thumb="1"}
{loop $data $key $val}
<li> <a href="{ $val['url']}">{ $val['title']}</a></li>
{/loop}
{/pc}
```

2.2、操作：position 标签处理函数，获取推荐位数据

可用参数

参数名	默认值	说 明
posid	null	推荐位 ID,必须填写
limit	0	数量的限制，默认为空获取所有推荐内容，格式为：“5”或“2,5”，分别表示获取 5 条或从第三条开始获取五条
thumb	0	只获取具有缩略图的数据，如果返回的字段不包括“thumb”或“thumb”为空，则从数据所有数据中搜索
format	1	是否进行数据输出格式化，设置为 0 直接调用数据库中的原始数据

示例：

```
{pc:content action="position" posid="2" num="6" order="listorder desc"}

<ul>

    {loop $data $r}

        <li><h4><a href="{ $r[url]}"

title="{ $r[title][value]}">[ {catInfo($r[catid],'name')} ]{chinesesubstr($r[title][value],0,8)}</a></h
4></li>

        {/loop}

</ul>

{/pc}
```

2.3、操作：hits 标签处理函数，用户获取排行榜

可用参数

参数名	默认值	说明
catid	null	栏目 ID,默认为空，即调用所有栏目的数据，多个栏目之间用逗号隔开，格式如：“12,25”
day	null	调用多少天内的排行，默认为空即调用所有的排行
limit	null	数量的限制，默认为空获取所有推荐内容，格式如：“5”或“2,5”，分别表示获取 5 条或从第三条开始获取五条
order	views desc	“views desc”（本月排行- monthviews desc 、本周排行 - weekviews desc、今日排行 - dayviews desc）
format	1	是否进行数据输出格式化，设置为 0 直接调用数据库中的原始数据

示例：

```
{pc:content action="hits" num="10" catid="7" order="views desc"}

<ul>

{loop $data $r}

<li><a href="{ $r[url]}" title="{ $r[places]}">{chinesesubstr($r[title][value],0,16)}</a></li>
```

```
{/loop}  
</ul>  
{/pc}
```

2.4、操作：classify 标签处理函数，输出分类和数据

可用参数

参数名	默认值	说 明
classid	null	标签所属类别 ID,必须填写
typeid	0	标签所属分类 ID,可以不用设置（默认为 0，即获取顶级分类）、为数值或\$typeid 变量（此种方式会随着提交的 typeid 参数值变化而改变父级分类）

示例：

```
{pc:content action="classify" classid="2" typeid="2" }  
<ul>  
{loop $data $r}  
<li><a href="{ $r[url] }" title="{ $r[name] }"></a></li>  
{/loop}  
</ul>  
{/pc}
```

3. 工具标签

3.1、plugin 插件

参数形式

参数名	默认值	说 明
name	null	调用插件名称，为系统唯一可用的插件名
action	null	调用插件方法，为插件内部定义的方法名
return	data	获取调用方法有返回值（如果有）

示例：

```
{pc:plugin name="message" action="list" return="data"}
```

```
<ul>
```

```
{loop $data $key $val}
```

```
<li><a href="{ $val[url] }">{ $val['title'] }</a></li>
```

```
{/loop}
```

```
</ul>
```

```
{/pc}
```

调用 message 插件并将获取的数据输出。

3.2、json 获取工具

参数形式

参数名	默认值	说 明
url	null	json 数据源 URL 地址
return	data	获取调用方法有返回值（如果有）

示例：

```
{pc:json url="http://www.abc.com/json.php" cache="86400"}
```

```
<ul>
```

```
{loop $data $key $val}
```

```
<li><a href="{ $val[url] }">{ $val['title'] }</a></li>
```

```
{/loop}
```

```
</ul>
```

```
{/pc}
```

3.3、xml 获取工具

参数形式

参数名	默认值	说 明
url	null	xml 数据源 URL 地址
return	data	获取调用方法有返回值（如果有）

示例：

```
{pc:xml url="http://www.abc.com/json.php" cache="86400"}  
  
<ul>  
  
{loop $data $key $val}  
  
<li><a href="{ $val[url] }">{ $val['title']}</a></li>  
  
{/loop}  
  
</ul>  
  
{/pc}
```

3.4、get 数据库查询工具

参数形式

参数名	默认值	说 明
sql	null	要执行的 SQL 语句
page	0	分页，通过变量把当前的分页传给 PC 标签进行处理
dbsource	null	数据源，当你通过系统后台的数据源模块配置过数据源时，可把数据源名填写到这里，系统会去对应的数据源来读取数据。如果要读取本系统的数据请留空
return	data	返回数据的变量名称

示例：

```
{pc:get sql="select * from `db_member`" cache="3600" page="$page" dbsource="sqlite3"  
return="data"}  
  
<ul>  
  
{loop $data $key $val}  
  
{ $val[username] }<br />  
  
{/loop}
```


{ \$pages }

{ /pc }

(三) 系统常量

1. 基础常量

SYS_START_TIME 系统开始时间

SYS_TIME 当前时间戳

SITE_PROTOCOL 主机协议

SITE_PORT 主机端口

2. 客户端访问常量

SITE_URL 当前访问的主机名

ROOT_URL 当前网站根目录 URL

UPLOAD_URL 附件访问 URL

STATIC_URL 静态文件 URL

SYS_PLUGIN_URL 系统第三方插件 URL

ADMIN_STATIC_URL 管理系统静态文件 URL

STYLE_URL 前端静态文件 URL

3. 服务端常量

CORE_PATH base.php 所在目录路径，即内核路径

ROOT_PATH 网站根目录路径

CACHE_PATH 缓存路径

UPLOAD_PATH 上传路径

STATIC_PATH 静态文件路径

SYS_PLUGIN_PATH 系统第三方插件路径

ADMIN_TEMPLATE_PATH 管理系统模板路径

ADMIN_STATIC_PATH 管理系统静态文件路径

(四) 页面变量

在不同类型的模板里面，包括首页、列表页、内容详细页有可以直接使用的系统变量，这些变量值由系统根据特定栏目预先设置。

使用方法： {\$变量} ，例如： {\$catid} 直接输出栏目 ID。

备 注： 部分变量名相同，但是在不同类型的页面中输出的值代表的含义不同,数组变量需要配合数组标签使用。

1. 首页公共变量

变量	数据类型	说明
\$SEO	array	页面 SEO 信息数组 包含 site_title, title, keyword,description 字段。使用方式 {\$SEO['keyword']}
\$SITE	array	站点全局设置信息
\$categorys	array	所有栏目信息

2. 列表页面公共变量

变量	数据类型	说 明
\$SEO	array	页面 SEO 信息数组 包含 site_title, title, keyword,description 字段。使用方式 {\$SEO['keyword']}

\$SITE	array	站点全局设置信息
\$categorys	array	所有栏目信息
\$category	array	当前栏目信息
\$catid	int	当前栏目 ID
\$typeid	int	当前分类 ID(有分类时存在)
\$page	int	当前页面号
\$arrparentids	array	当前栏目所有父辈栏目 ID 数组
\$arrchildids	array	当前栏目所有子栏目 ID
\$top_parentid	array	当前栏目祖辈栏目 ID
\$url	string	当前页面 url
\$template	string	当前模板名称
\$_rs	array	可供使用的当前页面字段原始值数组(单页栏目特有)
\$sub_title	string	分页子标题(有分页符时存在, 单页栏目特有)
\$pagenumber	int	分页总数(有分页符时存在, 单页栏目特有)
\$pageurls	array	分页所有 URL(有分页符时存在, 单页栏目特有)
\$pages	string	生成的分页字符串(无分页符时为空, 单页栏目特有)
\$page	int	当前页号(有分页符时存在, 单页栏目特有)

3. 详细页面公共变量

变量	数据类型	说 明
\$SEO	array	页面 SEO 信息数组 包含 site_title, title, keyword,description 字段。使用方式 { \$SEO['keyword'] }
\$SITE	array	站点全局设置信息
\$categorys	array	所有栏目信息
\$category	array	当前栏目信息

\$id	int	当前页面 ID
\$catid	int	当前栏目 ID
\$model	string	当前页面模型
\$page	int	当前页面号
\$arrparentids	array	当前栏目所有父辈栏目 ID 数组
\$arrchildids	array	当前栏目所有子栏目 ID
\$top_parentid	int	当前栏目祖辈栏目 ID
\$url	string	当前页面 url
\$template	string	当前模板名称
\$_rs	array	可供使用的当前页面字段原始值数组（单页栏目特有）
\$sub_title	string	分页子标题（有分页符时存在）
\$pagenumber	int	分页总数（有分页符时存在）
\$pageurls	array	分页所有 URL（有分页符时存在）
\$pages	string	生成的分页字符串（无分页符时为空）
\$page	int	当前页号（有分页符时存在）
\$previous_page	array	前一条信息数组
\$next_page	array	后一条信息数组

（五）重要全局函数

全局公共函数不仅用于系统开发，还可以用于页面模板处理、方便简洁地实现某些特定功能，比如图片剪切、缩略图、字数剪切、获取栏目信息等各种适用功能。

更多公共函数：stwms/lib/function/global.func.php

生成缩略图	
函数	<code>thumb(\$imgUrl,\$maxWidth=0,\$maxHeight=0,\$cutType=0,\$defaultImg='nopic.gif',\$isGetRemot=0)</code>
参数	(图片地址,生成的宽度,生成的高度,是否自动剪裁,默认图片,是否获取远程图片)
返回	图片地址
使用	<code>{Thumb(图片地址,200,50,1)}</code>
截取从 0 开始的任意长度的字符串	
函数	<code>cut_str(\$sourcestr, \$cutlength=80, \$addfoot=true)</code>
参数	(待截取的字符串,截取长度,是否添加"..."在末尾,)
返回	截取后的字符串
使用	<code>{cut_str('456215wsde ',60)}</code>
将字符串转换为数组	
函数	<code>string2array(\$data)</code>
参数	(字符串)
返回	array
使用	<code>{string2array(\$data)}</code>
处理界面模板	

函数	template(\$template = 'index',\$type = "", \$style = "")
参数	(文件名, 类型名, 文件目录)
返回	模板路径
使用	{template 'header' ,'文件夹名' }
包含管理模板	
函数	admin_tpl(\$c=",\$a=")
参数	(类型名, 方法名)
返回	管理模板路径
使用	<?php include admin_tpl('head');?>
显示消息,采用网页返回	
函数	showmessage(\$msg, \$url_forward = 'goback', \$ms = 1250, \$dialog = "", \$returnjs = "")
参数	(消息字符串,返回 URL,延迟毫秒数,JS 中 alert 字符串,执行的 JS 字符串)
返回	null
使用	略
将数组转换为 JSON 字符串（兼容中文）	
函数	make_json(\$array)
参数	(要转换的数组, 是否也应用到 key 上)
返回	转换得到的 json 字符串
使用	略
安全过滤函数（防止注入）	
函数	safe_replace(\$string)
参数	(原始字符串或数组)
返回	处理后的字符串
使用	略
获取当前页面完整 URL 地址	
函数	get_url(\$type 方法类型)
参数	(URL 类别)

	\$type 取值及返回值意义 0->绝对地址, 1->相对地址, 2->不带参数绝对地址, 3->不带参数相对地址
返回	绝对地址 URL
使用	略
获取栏目信息	
函数	catInfo(\$para,\$ky=NULL,\$base=true)
参数	(栏目 ID 或栏目信息数组,获取键名称,是否调用基本设置默认 true)
返回	栏目信息
使用	catInfo(12, ' name ')
获取子栏目	
函数	subcat(\$pid = NULL, \$type = NULL,\$self = 0,\$isBase=false)
参数	(父目录名称,栏目类型,是否包括本身,是否调用栏目详情) 栏目类型 0->一般栏目; 1->单页栏目; 2->外链栏目
返回	子栏目数组
使用	略
获取缓存	
函数	getcache(\$name,\$cDir,\$dType='file',\$addtions=NULL,\$autoCache=1)
参数	(缓存名称,缓存目录,缓存类型,自动缓存参数,是否自动缓存)
返回	缓存数据或文件路径
使用	如果缓存不存在, 尝试调用缓存名称对应的控制器的_fresh 方法, 重新生成缓存并返回
获取所有父栏目字段名	
函数	get_parents(\$id,&\$arr,\$type='id',\$spilter=',',\$is_self=false)
参数	(栏目 id,引用的栏目数组,获取字段名,分隔符,是否包括本栏目)
返回	字符串

六、 二次开发

(一) 系统文件目录结构分析

- | — statics 静态文件
 - | — admin 管理界面静态文件
 - | — default 默认样式目录
 - | — 其它样式目录
 - | — common 共用静态文件
 - | — css 共用 CSS 文件目录
 - | — images 共用图片文件目录
 - | — js 共用 JS 文件目录
 - | — plugins 第三方插件目录
 - | — others 其它共用文件目录
 - | — styles 前端静态文件文件
 - | — default 前端默认样式目录
 - | —前端其它样式目录
 - | — plugin 插件静态文件
 - | — message （示例：留言板插件）
 - | —其它插件样式
- | — stwms 主框架目录

- | — `__cache__` 系统缓存目录
 - | — `config` 系统配置文件目录
 - | — `data` 系统数据目录
 - | — `lib` 框架主类库、主函数目录
 - | — `module` 系统控制器目录
 - | — `plugin` 系统插件目录
 - | — `template` 模板目录
 - | — `base.php` 系统引导文件
-
- | — `ufs` 上传文件目录
 - | — `index.php` 程序主入口

(二) 数据库结构分析

系统数据库采用分离式布局，分为内核数据库和外围数据库

1. 内核数据库

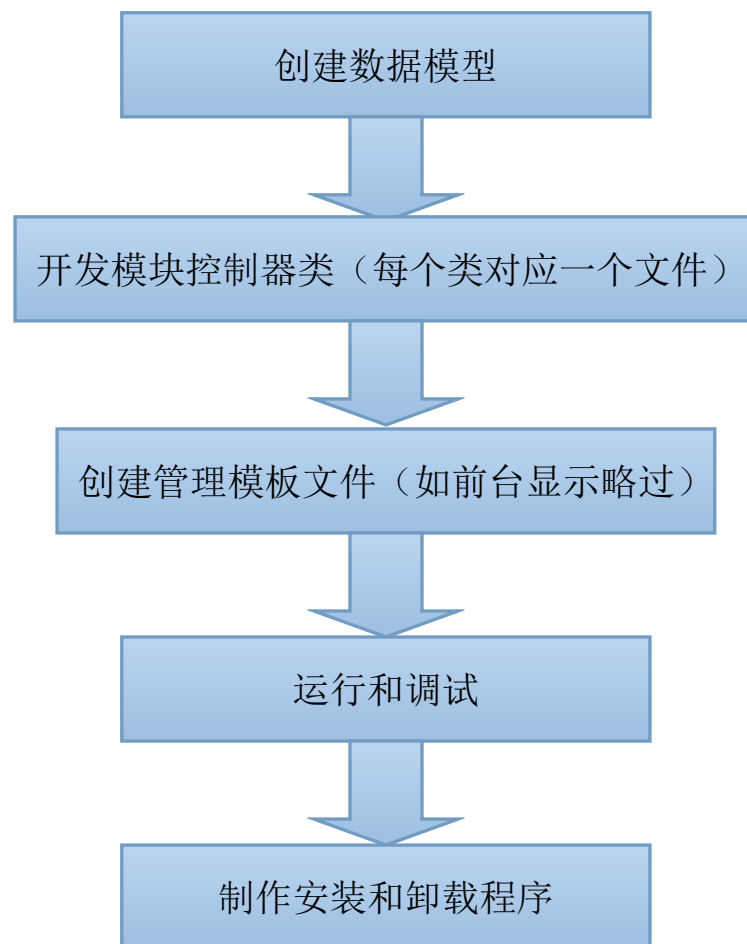
内核数据库为 XML 文件型数据库，用于存储轻量级数据，由系统专有数据库 API 统一管理，用于存储管理用户、栏目信息、分类信息、推荐位、系统表单、插件、URL 规则、插件、模型、菜单，每种数据库对应一个数据库文件。

内核数据表结构（略）。

2. 外围数据库

目前支持的数据库为 MySQL、MsSQL、ACCESS、SQLITE2、SQLITE3 数据库，数据库表结构根据模型确定。外围数据库用于存储重量级数据，内核模型映射外围数据库实体数据，并同步更新。内核模型包括系统模型、一般模型和自定义模型。系统安装时自动安装已经存在的可用模型。

（三）二次开发流程



(四) 入口程序

STWMS 是采用 MVC 设计模式开发, 基于模块和操作的方式进行访问, 采用单一入口模式进行项目部署和访问, 无论访问任何一个模块或者功能, 只有一个统一的入口。入口程序是在前期处理用户请求的引导程序。它是唯一一个可以被最终用户可以直接请求运行的。

stwms v1.0 的入口程序包含如下几行:

//index.php 文件

```
define('ROOT_PATH',dirname(__FILE__).DIRECTORY_SEPARATOR);  
include 'stwms/base.php';  
load::app();
```

这段代码首先加载了 stwms 框架的引导文件 base.php, 然后它根据指定的配置文件建立了一个 Web 应用实例并运行。

(五) 系统类库和主函数调用

1. 命名规范

系统类库位于系统的 stwms/lib/class/ 目录下面, 函数库文件名为 *.class.php。

系统函数库位于系统的 stwms/lib/function/ 目录下面, 函数库文件名为 *.func.php, 其中 global.func.php 为框架中默认加载, global.func.php 中函数可直接使用。

2. 系统类库调用

/** 参数说明:

string \$classname 类名,
integer \$initialize 是否初始化,默认初始化
alls \$para 传递给初始化的类构造函数的参数, 默认参数为空
string \$path 路径, 默认路径为空
string \$type 类型, 默认为".class"

*/

```
public static function cls($classname,$initialize =  
1,$para=NULL,$path=",$type='.class');
```

调用方法:

```
$imgObj=load::cls('image',1,1);//调用 image 类并传递参数初始化  
$res=$imgObj->thumbImg($oldimg_path, $IMG_PATH.$newimgurl,  
$maxWidth, $maxHeight,$cutType,$forceExec);//执行 image 类 thumbImg 方法  
处理图片
```

3. 系统函数库调用

/** 参数说明:

string \$func 函数库名,
string \$path 路径

*/

```
public static function func($func,$path=");
```

调用方法:

```
load::func('dir');//加载目录处理函数
```

(六) 配置文件调用

配置文件位于 stwms/config/目录下, 文件名为*.cfg.php, 配置文

件包括系统配置和用户定义的配置，使用 `load::cfg()` 方法加载配置文件。

/ 参数说明：**

`string $file` 配置文件名，
`string $key` 要获取的配置键值，
`string $default` 默认配置。当获取配置项目失败时该值发生作用，
`boolean $reload` 强制重新加载

***/**

`public static function cfg($file, $key = "", $default = "", $reload = false)`

调用方法：

`$mysqlCfg=load::cfg('database','mysql');` //加载 mysql 数据库配置，返回配置数组

(七) 控制器类开发规范

控制器类为用户访问直接调用的类库，默认访问的路由配置 `stwms/config/route.cfg.php` 中控制器。

如浏览器地址栏输入（具体地址根据实际服务器配置地址）：

`http://127.0.0.3/index.php?admin01&c=model&a=init`

则访问 `model` 控制器的 `init` 方法，其中 `init` 方法为管理方法，`admini01` 为系统登录密钥。

控制器类的方法有四种类型：

1. 管理方法

以“`m_`”为方法前缀，并声明为 `public` 的方法，如 `public function`

m_manage(), 访问此方法需要用户登录, 处 admin 和 api 控制器中管理方法外, 其它控制器中的管理方法访问还需要特定的权限;

2. 前端访问方法

即用户直接在浏览器地址栏中访问的方法, 不需要管理权限, 此类方法无前缀, 并声明为 public。

如在地址栏中输入 (具体地址根据实际服务器配置地址):

`http://127.0.0.3/index.php?c=content&a=lists&catid=7`

则访问 content 控制器中 list 方法, 并在此方法中接受 catid 参数

3. 数据交换方法

此类方法用于控制器之间数据交换, 以 “_” (下划线) 为前缀, 需要声明为 public, 按照使用位置不同分为一般方法、标签方法和缓存更新方法:

一般方法: 用于控制器中一般数据交换;

标签方法: 应用于前端模板标签的处理, 以 “_tag_” 作为方法名前缀;

缓存更新方法: 命名为 “_fresh”, 可以传递参数给此方法, 调用缓存时若缓存不存在, 则调用该缓存的控制器中的此方法。

控制器之间调用示例:

```
$menus=load::module('rolemenu')->_get_menu($roleid,$type);
```

//加载并初始化 rolemenu 控制器, 传递参数并调用 _get_menu 方法, 将返回的结果赋值给 \$menus 变量。

4. 私有方法

此类方法声明为 `private`，为控制器私有，通常为控制器提供操作工具的方法。

(八) 常见问题及使用技巧说明

1. 模板直读

模板可以不经过程序设置，直接经过 PHP 编译后显示输出，但与程序相关的公共变量无效。

使用示例：

`http://127.0.0.3/index.php?page_service.html`

将访问默认前端模板根目录下 `page_service.html` 模板。

也可以使用目录：

`http://127.0.0.3/index.php/about/page_service.html`

访问默认前端模板根目录下 `about` 目录中 `page_service.html` 模板文件。

2. 静态文件分离部署

静态文件分为三类，分别为管理系统静态文件、第三方插件静态和前端静态文件，可以将三类静态文件部署到其它服务器上，而主服务器只放置主框架文件。

部署方法如下：

1)、系统静态文件部署：位于“`statics/admin/`目录下”，在 `application` 类中 `ADMIN_STATIC_URL` 常量定义前端访问路径；

2)、第三方插件静态文件部署：位于“`statics/common/plugins/`”目录下，引导文件 `base.php` 中 `SYS_PLUGIN_URL` 常量定义前端

访问路径。

3)、前端静态文件部署：位于“statics/styles/”目录下，在管理系统中“系统配置”→“基础配置”→“基本配置”中：

前台默认静态文件路径： 前台访问静态文件路径

3. 系统框架文件分离部署

主框架位于“stwms/”目录下，可以将此目录放置到非 WEB 服务器站点根目录下，并设置相应子目录的读写权限，具体目录权限设置详见安装过程。

4. 网站更换域名后迁移

系统采用 URL 动态生成，即是从数据库中查询数据的同时根据 URL 规则生成地址。更换域名后，地址自动同步更新。但如果修改上传文件路径，默认为“/ufs/”目录，需要在管理系统手动更新地址。

更新方法：“站点内容”→“附件管理”→“地址更新”

输入原上传地址，并点击“确定”，程序会自动用现有上传地址替换数据库中所有原始地址。

5. 动态地址简化

即是将动态的 URL 地址进行简化处理，简化后的地址接近伪静态，更利于网站 SEO。

如原始地址：`http://127.0.0.3/index.php?c=content&a=lists&catid=7`

经过简化后：`http://127.0.0.3/about/`

简化方法：“站点内容” —> “栏目管理” —> “基本选项” —> “URL 设置”，如下

栏目页：	<input checked="" type="radio"/> 动态	<input type="radio"/> 静态化
内容页：	<input checked="" type="radio"/> 动态	<input type="radio"/> 静态化
动态简化：	<input type="text" value="about/"/>	<input checked="" type="checkbox"/> 输入正确

6. 将静态页面生成到根目录

登录管理系统：“系统配置” —> “基础配置” —> “基本配置”

静态文件生成路径：	<input type="text" value="/"/>	当需要生成静态文件时，生
-----------	--------------------------------	--------------

输入“/”或者留空，则生成静态页面到网站根目录。

7. 获取数据库对象

在系统引导文件 `base.php` 中的同时提供直接加载数据库对象的方法，开发人员可以运用此方法，传入相应的配置参数，直接获取数据库对象。各种数据库采用统一方法加载，数据库按照统一标准方法操作。

1). 使用方法一：

```
$mcount=load::db()->count('member');
```

//将加载默认数据库配置，并连接数据库，返回数据库对象。此处调用数据库对象的 `count` 方法，统计 `member` 表的记录总数并赋值给 `$mcount` 变量。

2). 使用方式二:

```
$dbObj=load::db('mysql');
```

//加载系统配置中 mysql 数据库

3). 使用方式三:

```
$dbObj= load::db('xml','formtype','core',3);
```

//加载系统内核数据库，此数据库路径为 stwms/ data/ core/

formtype.php。第四个参数为“3”表示每次加载都会产生新的数据库对象，通常应用于内核数据库。

4). 使用方式四:

```
$dbObj= load::db(array(
```

```
    'DSN' => 'localhost,3306', //数据库地址和端口
```

```
    'user' => 'root', //登录用户名
```

```
    'pass' => 'root', //登录密码
```

```
    'dbname' => 'china720', //默认操作的数据库名称
```

```
    'pconnect' => false, //是否启用永久连接
```

```
    'pre' => '720_', //数据库表前缀
```

```
));
```

//此种方法方式加载用户自定义的数据库配置

七、 插件开发

(一) 编写第一个插件

1. 新建一个目录，命名为“message”，此目录作为插件项目所在目录；
2. 在 message 目录下新建“config.php”文件，此文件为插件配置文件，将以下内容保存至此文件中：

```
<?php
return array(
    'name'=>'message', //插件唯一名称
    'alias'=>'留言板', //插件别名
    'install_dir'=>'message_v1', //自定义安装目录，安装时如存在此目录会自动切换目录
    'author'=>'蓝天之云', //插件作者
    'copyright'=>'版权所有 2013', //版权信息
    'entry'=>'init', //定义插件入口程序
    'menu'=> array( //显示菜单名称定义
        'show'=>'显示',
        'add'=>'添加',
    )
);
?>
```

3. 在 message 目录下新建“plugin_message.php”文件，此文件为插件主类文件，将以下内容保存至此文件中：

```
<?php
class plugin_message{
public function __construct(){ //构造方法，在此处可以执行初始化工作
}
public function m_init(){ //管理系统入口函数
    echo 'init';
}
public function m_show(){ //菜单名为“显示”的管理方法
```



```

}
public function m_add(){ //菜单名为“添加”的管理方法
}
public function index(){ //外部方法，可以直接访问
    echo 'index';
}
public function _tag_hits(){ //插件标签方法
    return $this->getHits();
}
private function getHits(){
    return 1;
}
}
}
?>

```

4. 在 message 目录下建立 statics 目录，此目录用于存放插件静态文件（若无静态文件可以省略此步骤）。
5. 将 message 打包成 “message.zip” 文件。
6. 安装插件：进入管理系统，插件管理 —> 安装插件 —> 选择插件（安装图解详见插件管理），系统自动安装，安装完成后，此插件就可以使用了。
7. 插件权限配置（配置图解详见角色管理）；
8. 插件使用：
 - a. 访问管理方法：在插件管理中点击“进入管理”，也可以点击配置的插件菜单进入管理；
 - b. 外部访问：http://127.0.0.2/index.php? plugin_c=message，访问 message 插件 index 默认方法；
 - c. 标签内调用：

```

{pc:plugin name=”message” action=”hits” return=”data”}

{$data}

```

{/pc}

调用插件的 hits 标签方法,并输出返回值。

(二) 文件部署

插件安装后分别将静态文件和服务端文件拷贝到相应目录，
以范例插件 message 为例说明：

1. 安装前的目录结构

- | — message 插件项目目录
 - | — config.php 配置文件
 - | — plugin_message.php 插件主类文件
 - | — statics 插件静态文件
 - | — 用户自定义文件

2. 安装后服务器端目录结构

- | — statics 系统静态文件目录
 - | — plugin 系统插件静态文件目录
 - | — message_v1 message 插件静态文件目录
- | — stwms 系统静态文件目录
 - | — plugin 系统插件目录
 - | — message_v1 message 插件目录
 - | — config.php 配置文件
 - | — plugin_message.php 插件主类文件
 - | —用户自定义文件

说明：插件安装目录为配置文件中配置的安装目录，如果此目录在系统插件目录下已经存在，则系统会自动更改安装目录；安装过程中将插件目录下命名为 statics 的目录下文件及子目录移动到插件所在的静态目录。

(三) 入口程序

插件的入口方法在配置文件中定义，键名称为“entry”，为管理插件入口，为该插件主要的管理操作界面，通过此入口可以访问其它管理方法。

(四) 编码规范

1. 配置文件：配置文件命名为“config.php”，文件内容为 PHP 数组（模板详见范例程序）；
2. 主类文件：主类文件命名为“plugin_插件名.php”，主类名为“plugin_插件名”，跟主类文件名称一致；
3. 方法命名：插件方法类型跟控制器方法相同，也分为管理方法、前端访问方法和标签方法，其它类型方法用户可自定义（命名规则详见“控制器类开发规范”）。
4. 静态文件：静态文件放置于命名为“statics”的目录下。

(五) 打包发布

将开发后的插件项目打包成 zip 压缩文件格式，然后安装到自己网站系统中，或者将插件发布到 STWMS 官方插件中心。

八、 第三方手册

第三方手册是 STWMS 中使用的第三方开源项目相关使用手册和 API 文档，方便用户在此基础上进行二次开发。部分源代码在 STWMS 中做了优化，增强了应用功能，详情参考源代码。

(一) JQuery 手册

STWMS v1.0 中使用的 js 框架为 jquery1.7 版

官方网站地址: <http://jquery.com/>

中文版在线手册: <http://www.php100.com/manual/jquery/>

什么是 jquery?

Jquery 是继 prototype 之后又一个优秀的 JavaScript 框架。它是轻量级的 js 库(1.7 版压缩后只有 67k) ,它兼容 CSS3,还兼容各种浏览器 (IE 6.0+, FF 1.5+, Safari 2.0+, Opera 9.0+)。jQuery 使用户能更方便地处理 HTML documents、events、实现动画效果，并且方便地为网站提供 AJAX 交互。jQuery 还有一个比较大的优势是，它的文档说明很全，而且各种应用也说得很详细，同时还有许多成熟的插件可供选择。jQuery 能够使用户的 html 页保持代码和 html 内容分离，也就是说，不用再在 html 里面插入一堆 js 来调用命令了，只需定义 id 即可。

jquery 文件在 STCMS V1.0 中放置在哪里?

web 目录\statics\common\js\jquery.js

(二) artDialog 手册

STCMS V1.0 中使用的弹窗效果框架为 artDialog4.1.7

项目地址: <http://code.google.com/p/artdialog/>

下载地址: <http://code.google.com/p/artdialog/downloads/list>

什么是 artDialog?

artDialog 是一个基于 javascript 编写的对话框组件，精致的界面与易用的接口是它的优势。

artDialog 项目文件在 STCMS V1.0 中放置在哪里?

web 目录\statics\common\plugins\artDialog\

(三) formValidator 手册

STCMS V1.0 中使用的表单验证框架为 formValidator 4.0

项目地址: <http://www.cnblogs.com/wzmaodong/archive/2011/07/15/2106947.html>

在线手册: <http://www.yhuan.com/formValidator4.1.1/doc/index.html>

什么是 formValidator?

jQuery formValidator 表单验证插件，它是基于 jQuery 类库，实现了 js 脚本于页面 html 代码的分离。你可以划分多个校验组，每个组的校验都是互不影响。对一个表单对象，你只需要写一行代码就可以轻松实现无数种（理论上）脚本控制。目前支持 5 种大的校验方式，分别是：inputValidator（针对 input、textarea、select 控件的字符长度、值范围、选择个数的控制）、compareValidator（提供 2 个对象的比较，目前可以比较字符串和数值型）、ajaxValidator（通过 ajax 到服务器上做数据校验）、regexValidator（提供可扩展的正则表达式库）、functionValidator（可使用外部函数来做校验）。

formValidator 项目文件在 STCMS V1.0 中放置在哪里?

web 目录\statics\common\plugins\formvalidator\

(四) calendar 手册

STCMS V1.0 中使用的日期选择器 Dynarch Calendar version 1.9

官方网站地址: <http://www.dynarch.com/jscal/>

什么是 Dynarch Calendar?

Dynarch.com 开发的一个日历控件，她功能十分健全，内置多种皮肤，

支持多种语言，同时提供了一个配置程序，使用非常方便。

Dynarch Calendar 项目文件在 **STCMS V1.0** 中放置在哪里？

web 目录\statics\common\plugins\calendar\

(五) swfupload 手册

STCMS V1.0 中使用的上传工具为 SWFUpload v2.2

项目地址: <http://code.google.com/p/swfupload/>

什么是 SWFUpload？

SWFUpload 是一个客户端文件上传工具，最初由 Vinterwebb.se 开发，它通过整合 Flash 与 JavaScript 技术为 WEB 开发者提供了一个具有丰富功能继而超越传统<input type="file" />标签的文件上传模式。

SWFUpload 项目文件在 **STCMS V1.0** 中放置在哪里？

web 目录\statics\common\plugins\swfupload\

(六) CKEDITOR

STCMS V1.0 中使用的编辑器为 ckeditor 4.1 standard 版

官方网站: <http://ckeditor.com/>

在线手册: http://docs.cksource.com/ckeditor_api/

什么是 CKEditor？

CKEditor 是新一代的 FCKeditor，是一个重新开发的版本。CKEditor 是全球最优秀的网页在线文字编辑器之一，因其惊人的性能与可扩展性而广泛的被运用于各大网站。

CKEditor 项目文件在 **STCMS V1.0** 中放置在哪里？

web 目录\statics\common\plugins\editor\

九、 版权说明

STWMS 系统内核源代码为开源项目，用户可以在此框架基础上自由修改和开发自己的应用程序，STWMS 官方网站不定期提供内核代码更新，用户根据需要升级系统内核。

系统管理 UI 界面及功能插件版权归开发公司或个人所有，STWMS 官方网站提供界面模板及插件发布平台。

官方网站：<http://www.stwms.cn>