

织梦 CMS 模板制作手册

版本: 20120312

编写: 天涯

上海卓卓网络科技有限公司

Copyright ©2002 - 2012 DesDev INC.All rights reserved.

目录

目录.....	2
版权申明.....	6
前言.....	7
简介.....	7
主题.....	7
发行方法.....	7
手册 Web 站点.....	7
最后.....	8
第一章 织梦模板设计师要求.....	9
1.1 设计师定义.....	9
1.2 设计师级别划分及要求.....	9
第二章 模板制作流程.....	10
第三章 模板文件及目录结构.....	11
3.1 模板目录.....	11
3.2 模板文件与功能说明.....	11
3.2.1 概念，设计和使用模板.....	12
3.2.2 命名规则.....	12
3.3 其它模板说明.....	13
第四章 模板引擎介绍.....	14
4.1 静态模板引擎.....	14
4.1.1 核心文件.....	14
4.1.2 功能描述.....	14
4.1.3 模板语法.....	14
4.2 动态模板引擎.....	15

4.2.1 核心文件.....	15
4.2.2 功能描述.....	16
4.2.3 使用方法.....	16
4.2.4 标记语法.....	16
4.3 模板标签.....	19
4.3.1 标签介绍.....	19
4.3.2 标签的分类.....	19
第六章 标签参考.....	20
6.1 全局标签.....	20
arclist 文档列表.....	20
arclistsg 独立单表模型文档列表.....	23
arcpagelist arclist 列表分页.....	24
autochannel 指定栏目.....	25
cattree 树形类目.....	26
channel 获取栏目列表标签.....	26
channelartlist 当前频道的下级栏目的内容列表.....	29
feedback 评论内容.....	32
field 内容变量.....	34
flink 友情链接.....	35
flinktype 友情链接类型.....	36
global 全局配置变量标签.....	37
hotwords 搜索关键词.....	37
json JSON 数据调用.....	38
loop 万能调用标签.....	41
memberlist 会员列表.....	42
myad 广告.....	44
mynews 站内新闻.....	46
mytag 自定义宏标签.....	48
php php 标签.....	49
softmsg 软件下载提示信息.....	50

sonchannel 子栏目调用.....	52
sql SQL 查询标签.....	53
tag TAG 调用标签.....	55
type 指定栏目调用标签.....	56
vote 投票标签.....	57
6.2 列表标签.....	57
list 列表数据标签.....	57
pagelist 列表分页标签.....	59
6.3 内容标签.....	60
likearticle 相关文章.....	60
pagebreak 内容分页标签.....	61
pagetitle 分页标题.....	63
prenext 获取上一篇、下一篇内容.....	65
fieldlist 当前模型内容页字段列表.....	66
6.4 模块标签.....	68
bookcontentlist 连载图书最新内容调用.....	68
booklist 连载图书调用.....	69
group 圈子调用标签.....	70
groupthread 圈子主题调用标签.....	71
6.5 其他标签.....	72
infoguide 分类信息的地区与小分类搜索.....	72
infolink 分类信息地区与类型快捷链接.....	73
第七章 模板扩展.....	75
7.1 标签扩展.....	75
7.1.1 定义标签.....	75
7.1.2 标签文件内容结构.....	75
7.1.3 属性处理.....	76
7.1.4 底层模板处理.....	77
7.2 常用函数.....	78
cn_substr.....	78

Html2Text.....	79
Text2Html.....	79
MyDate.....	80
GetDateTimeMk.....	81
FloorTime.....	82
HtmlReplace.....	82
RemoveXSS.....	83

版权申明

发布本资料须遵守开放出版许可协议 1.0 或者更新版本；

未经版权所有者明确授权，禁止发行本文档及其被实质上修改的版本；

未经版权所有者事先授权，禁止将此作品及其衍生作品以标准（纸质）书籍形式发行；

如果有兴趣再发行或再版本手册的全部或部分内容，不论修改过与否，或者有任何问题，请联系版权所有者 [tianya\(AT\)desdev.cn](mailto:tianya(AT)desdev.cn)。

对 DedeCMS 及本手册有任何疑问或者建议，请进入官方论坛 <http://bbs.dedecms.com> 发布相关讨论。并在此感谢所有参与到 DedeCMS 项目中的朋友。

有关 DedeCMS 项目及本文档的最新资料，请及时访问官方主站 <http://www.dedecms.com>。

本文档及其描述的内容受有关法律的版权保护，对本文档内容的任何形式的非法复制，泄露或散布，将导致相应的法律责任。

前言

简介

本手册是织梦官方推出，用于介绍织梦内容管理系统（以下简称：DedeCMS）模板的制作，为织梦模板设计师提供一份快速检索的操作文档，全面介绍了织梦的模板引擎、标签及常用函数，相信会大大提高模板设计师的制作效率。

DedeCMS 的模板引擎采用了官方自行研发的 Dede 模板引擎，类 XML 结构的标签简单易懂，便于在网页编辑器中进行编辑。同时标签齐全，应用广泛，这为基于 DedeCMS 建设开发的站点提供了一个良好的基础，只需要懂 HTML 就可以搭建您想要的站点，这也是 DedeCMS 风靡的一个重要因素。

主题

手册将会从以下几个部分进行重点的说明：

- 织梦模板设计师要求
- 模板制作流程
- 模板文件及目录结构
- 模板引擎介绍
- 模板标签
- 模板扩展

发行方法

手册包含在线阅读、PDF、CHM 几种发行方式，您可以登录织梦帮助中心来获取最新版本。

手册 Web 站点

您可以通过手册 Web 站点同官方取得交流，以及分享您在制作过程中的心得。

帮助中心: <http://help.dedecms.com>

织梦技术论坛: <http://bbs.dedecms.com>

最后

编写这本手册, 我们不希望他是一个人的项目, 希望他有更多热爱织梦、热爱中国开源事业的朋友参与进来。同时也需要对使用、支持 DedeCMS 的每一位用户提供的帮助、贡献、意见、评论、精神支持以及耐心表示感谢。

第一章 织梦模板设计师要求

本章节介绍织梦模板制作人员所需要的基本条件、所需相关技术。便于企业、单位招聘选择适合自身发展的从业人员及对员工进行定位、培训。

1.1 设计师定义

能够设计、制作 DedeCMS 程序模板的人员。

1.2 设计师级别划分及要求

这里根据建站需求，我们将织梦模板设计师进行一个大致的划分：

初级：能够对模板进行修改，例如：元素布局、调用内容等。需要掌握基本的 HTML，图形图像处理技术，同时需要对织梦的常用标签有大致地了解。

中级：能够根据设计稿输出页面、嵌套标签，并能够灵活调用 DedeCMS 内容。需要熟练掌握 Div+CSS、Javascript，DedeCMS 的模板标签用法，同时需要有一些基本的 PHP、MySQL 尝试。

高级：能够独立设计网站界面，并能够输出页面、嵌套标签，同时能够根据自己需求扩展编写自己需要的标签以达到站点所需效果。需要熟练掌握图形图像网页设计相关技术、PHP、Javascitpt（jQuery）、织梦模板引擎。

第二章 模板制作流程

织梦模板制作需要经过以下几个流程：

- ① 设计页面效果图，一般为 PSD 或者 PNG 格式的原图；
- ② 将页面效果图输出为 HTML 格式，后缀名为 “.htm”；
- ③ 根据页面内容调用需求生成或者编写标签所需代码；
- ④ 嵌套标签代码到输出页面对应位置；
- ⑤ 测试调试模板文件，保证调用和设计效果一致；
- ⑥ 将模板标签、文件上传到模板目录；
- ⑦ 在系统设定模板目录，重新生成全站；

第三章 模板文件及目录结构

3.1 模板目录

系统的模板目录在系统根目录下的 `templets` 内，下面是模板目录的文件目录结构。

```
/templets.....
├──default..... 默认模板目录
│   ├──images..... 模板图片目录
│   │   ├──mood.....
│   │   └──photo.....
│   ├──js..... 模板 JS 脚本目录
│   └──style..... 模板 CSS 样式目录
├──lurd..... LURD 系统模板
├──plus..... 插件模板目录
├──system..... 系统底层模板目录
└──wap..... WAP 模块模板目录
```

3.2 模板文件与功能说明

DedeCMS 系统的模板是非固定的，用户可以在新建栏目时可以自行选择栏目模板，官方仅提供最基本的默认模板，即是内置系统模型各个模板，由于 DedeCMS 支持自定义频道模型，用户自定义新频道模型后，还需要按该模型的字段设计一套新的模板。

此外，DedeCMS 也支持使用风格的形式使用模板，默认风格是 `default`，它表示系统默认使用 `{cmspath}/templets/default` 这个文件夹的模板，如果你下载了一套新的模板，你可以不必要删除 `default` 原有的文件，把下载的模板文件夹命名你想要的风格名称，如 `style2` 等，然后在后台修改了默认的模板风格名称为 `style2`，那系统将使用 `{cmspath}/templets/style2` 这文件当作默认模板，但是若你手工指定了栏目模板的位置，则后台参数风格的定义无效。

3.2.1 概念，设计和使用模板

板块（封面）模板

指网站主页或比较重要的栏目封面频道使用的模板，一般用“index_识别 ID.htm”命名，此外，用户单独定义的单个页面或自定义标记，也可选是否支持板块模板标记，如果支持，系统会用板块模板标记引擎去解析后才输出内容或生成特定的文件。

列表模板

指网站某个栏目的所有文章列表的模板，一般用“list_识别 ID.htm”命名。

档案模板：

表示文档查看页的模板，如文章模板，一般用“article_识别 ID.htm”命名。

其它模板：

一般系统常规包含的模板有：主页模板、搜索模板、RSS、JS 编译功能模板等，此外用户也可以自定义一个模板创建为任意文件。

3.2.2 命名规则

模板保存位置

模板目录：{cmspath}/templates/样式名称（英文，默认为 default，其中 system 为系统底层模板，plus 为插件使用的模板）/具体功能模板文件}

模板文件命名规范

- ① index_识别 ID.htm：表示板块（栏目封面）模板；
- ② list_识别 ID.htm：表示栏目列表模板；
- ③ article_识别 ID.htm：表示内容查看页（文档模板，包括专题查看页）；
- ④ search.htm：搜索结果列表模板；
- ⑤ index.htm：主页模板；

注：[识别 ID]可以在“频道模型管理”的地方获得，当然，你也可以在“频道模型管理”的地方确定某个频道

的模板命名。

例：list_image.htm 表示是就是内容类型为图片集的栏目默认列表模板。

article_article.htm 表示的是文章查看页模板。

3.3 其它模板说明

默认底层模板

位置：{cmspath}/templets/system

功能：在没有指定标记的默认底层模板的时候，系统将自动调用这个文件夹的相应文件作为底层模板。

插件目录模板

位置：{cmspath}/templets/plus

功能：评论、友情链接、RSS 地图等模板。

会员后台模板

位置：{cmspath}/member/templets

功能：会员后台的模板。

系统后台模板

位置：{cmspath}/dede/templets

功能：系统后台管理的模板

第四章 模板引擎介绍

DedeCMS 从 V5 开始采用了解析式引擎与编译式引擎并存的模式，由于在生成 HTML 时，解析式引擎拥有巨大的优势，但对于动态浏览的互动性质的页面，编译式引擎更实用高效，织梦 CMS 采用双引擎并存的模式，在保持标签风格一致性的同时，也保证将来开发更多互动模块时有更好的性能。

4.1 静态模板引擎

4.1.1 核心文件

include/dedetag.class.php

4.1.2 功能描述

DedeCMS V5.3 及之前版本使用的主要的模板类，它是解析式模板类，并非编译式的。

注：编译式是通过获得标签位置进行内容替换，编译式是直接解析为 PHP 代码，二次执行。

4.1.3 模板语法

织梦模板引擎是一种使用 XML 名字空间形式的模板解析器，使用织梦解析器解析模板的最大好处是可以轻松的制定标记的属性，感觉上就像在用 HTML 一样，使模板代码十分直观灵活，新版的织梦模板引擎不单能实现模板的解析还能分析模板里错误的标记。

织梦模板引擎的代码样式

```
{dede:标记名称 属性=值'/'}
```

```
{dede:标记名称 属性=值'} {/dede:标记名称}
```

```
{dede:标记名称 属性=值'}自定义样式模板(InnerText){/dede:标记名称}
```

注：如果使用带底层模板的标记，必须严格用 {dede:标记名称 属性=值'} {/dede:标记名称} 这种格式，否则会报错。

内置系统标记

① **global** 标记，表示获取一个外部变量，除了数据库密码之外，能调用系统的任何配置参数，形式为：

```
{dede:global name='变量名称'} {/dede:global}
```

Copyright ©2002 - 2012 DesDev INC. All rights reserved.

或

```
{dede:global name='变量名称'/'}
```

其中变量名称不能加\$符号，如变量\$cfg_cmspath，应该写成{dede:global name='cfg_cmspath'/'}。

② **foreach** 用来输出一个数组，形式为：

```
{dede:foreach array='数组名称'}[field:key/] [field:value/]{/dede:foreach}
```

③ **include** 引入一个文件，形式为：

```
{dede:include file='文件名称' ismake='是否为 dede 板块模板(yes/no)'/}
```

对文件的搜索路径为顺序为：绝对路径、include 文件夹，CMS 安装目录，CMS 主模板目录

自定义函数使用

```
{dede:标记名称 属性='值' function='youfunction("参数一","参数二","@me")'/'}
```

其中 @me 用于表示当前标记的值，其它参数由你的函数决定是否存在，例如：

```
{dede:field name='pubdate' function='strftime("%Y-%m-%d %H:%M:%S","@me")'/'}
```

织梦标记允许有限的编程扩展

格式为：

```
{dede:tagname runphp='yes'}
```

```
    $aaa = @me;
```

```
    @me = "123456";
```

```
{/dede:tagname}
```

@me 表示这个标记本身的价值，因此标记内编程是不能使用 **echo** 之类的语句的，只能把所有返回值传递给 @me。

此外由于程序代码占用了底层模板 InnerText 的内容，因此需编程的标记只能使用默认的 InnerText。

4.2 动态模板引擎

4.2.1 核心文件

include/dedetemplate.class.php

4.2.2 功能描述

用于非核心模块的动态页面或列表页的模板解析，如：member/content_list.php，通常是在

datalistcp.class.php 中使用，这个类在动态运行的情况下，由于本身是把模板编译成 PHP 的，因此性能上会优于旧的解析类，这个方法将在未来版本中作为通用的方式。

4.2.3 使用方法

```
$tpl = new DedeTemplate(
```

模板对象实例名称，通常是'tpl'，

[模板存放目录（生成缓存时会存放在这个目录），

include 语法默认引用目录]

```
);
```

通常情况下参数二和参数三是不必要的，如：

```
$tpl = new DedeTemplate('tpl');
```

如果在类文件中调用，应该加上设置：

```
$this->tpl->SetObject($this);
```

在一些块调用中默认将使用当前类的成员函数。

```
$tpl->LoadTemplate(模板的物理路径);
```

如果模板中带有 {dede:config name=" value="}/}

可以在载入模板后，通过 `$tpl->GetConfig($name)` 获得这些变量的值。

显示页面或保存页面为文件

```
$tpl->Display();
```

```
$tpl->SaveTo(物理绝对路径的文件名);
```

4.2.4 标记语法

标记通用特性

① 短标记

```
{dede:tagname.name/}
```

等同于

```
{dede:tagname name=""/}
```

② 块标记


```
{dede:tagname}
```

循环代码

```
{/dede:tagname}
```

标记的具体语法及对应的 PHP 代码

① 配置变量

```
{dede:config name=" value="}/}
```

配置变量可以在载入模板后通过 `$tpl->GetConfig($name)` 获得，仅作为配置，不在模板中显示。

② 短标记

```
{dede:global.name/}
```

 外部变量 等同于 `<?php echo $name; ?>`

```
{dede:var.name/}
```

 var 数组 等同于 `<?php echo $_vars['name']; ?>`

```
{dede:field.name/}
```

 field 数组 等同于 `<?php echo $fields['name']; ?>`

```
{dede:cfg.name/}
```

 系统配置变量 等同于 `<?php echo $cfg_name; ?>`

考虑到大多数情况下都会在函数或类中调用模板，因此 `$_vars`、`$fields` 数组必须声明为 `global` 数组，否则模板引擎无法获得它的值从而导致产生错误。

③ 自由调用块标记

```
{tag:blockname bind='GetArcList' bindtype='class'}
```

循环代码

```
{/tag:blockname}
```

必要属性：

bind 数据源来源函数

bindtype 函数类型，默认是 `class` 可选为 `sub`

rtype 返回结果类型，默认是 `array`，可选项为 `string`

自定义函数格式必须为 `function(array $atts,object $refObj, array $fields);`

在没有指定 `bind` 绑定的函数的情况下，默认指向 `MakePublicTag($atts,$tpl->refObj,$fields)` 统一管理。

④ 固定块标记

1. datalist

从绑定类成员函数 `GetArcList` 中获取数组并输出

```
{dede:datalist}
```

循环代码

```
{/dede:datalist}
```

遍历一个二维数组，数据源是固定的，只适用用类调用。

等同于

```
{tag:blockname bind='GetArcList' bindtype='class' rstyle='arrayu'}
```

循环代码

```
{/tag:blockname}
```

2.label

从绑定函数中获取字符串值并输出

等同于 {tag:blockname bind='func' bindtype='sub' rstyle='string'/}

3.pagelist

从绑定类成员函数 GetPageList 中获取字符串值并输出

等同于 {tag:blockname bind='GetPageList' bindtype='class' rstyle='string'/}

4.include

```
{dede:include file=""}
```

```
{dede:include filename=""}
```

5.php

```
{dede:php
```

php 代码

```
/}
```

或

```
{dede:php}
```

php 代码

```
{/dede:php}
```

6.If

仅支持 if , else , else 直接用 {else} 表示,但不支持 {else if} 这样的语法 , 一般建议模板中不要使用太复杂的条件语法, 如果确实有需要, 可以直接使用 php 语法。

```
{dede:if 条件} a-block {else} b-block {/dede:if}
```

条件中允许使用 var.name 、 global.name 、 field.name、 cfg.name 表示相应的变量。

如:

```
{dede:if field.id>10 }
```

```
{/dede:if}
```

7. 遍历一个 array 数组

```
{dede:array.name}

    {dede:key/} = {dede:value/}

{/dede:array}
```

各种语法的具体编译后的代码，可查看 dedetemplate.class.php 的 function CompilerOneTag(&\$cTag)。

4.3 模板标签

4.3.1 标签介绍

从上面的章节中我们可以获知，织梦模板引擎分为编译式和解析式，系统的标签同样也有这样的划分。我们在系统涉及到内容管理及生成 HTML 的地方大多采用了解析式的模板标签，一些互动部分的功能，如果使用了织梦编译式的模板引擎，也会涉及到编译式的标签，在模板制作和日常的站点维护中，我们大多接触的是解析式的模板标签，也是本章标签介绍的重点。

解析式标签所在的目录为系统根目录下 `/include/taglib` 中，编译式涉及到的标签文件存在放根目录下 `/include/tpplib` 中，这部分的扩展内容会在随后的章节中详细介绍。

4.3.2 标签的分类

每个标签都会有他的作用域，上文中我们可以了解系统文档页面可以分为：封面、列表、内容几个部分，每个页面的模板则会涉及到不同的标签，所以标签也有了他的作用域，我们在模板制作过程中主要根据模板的作用域来对模板进行如下划分：

全局标签

可以在前台文档任意页面使用的模板标记，例如：arclist、channel、sql、loop 等。

列表标签

仅在模板 *_list.htm 中可以使用的标签，例如：list、pagelist。

内容标签

仅在模板 *_article.htm 中可以使用的模板标记，例如：likearticle、pagebreak 等。

模块标签

仅在特定的模块、插件中使用的模板标记


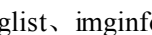



第六章 标签参考

6.1 全局标签

arclist 文档列表

(DedeCMS > 3.0, DedeCMS 4, DedeCMS 5)

名称: arclist

功能: 用于获取系统主从表模型（如：文章、软件、图集、商品等）的一系列文档，也称自由列表块标记，其中、、、、 都是由该标记所定义的不同属性延伸出来的别名标记。

语法:

```
{dede:arclist  flag='h' typeid=" row=" col=" titlelen=" infolen=" imgwidth=" imgheight=" listtype=" orderby=" keyword="
limit='0,1'}

    <a href="[field:arcurl/]">[field:title/]</a>

{/dede:arclist}
```

文件:

\include\taglib\arclist.lib.php

function lib_arclist(&\$ctag,&\$refObj)

默认底层模板:

imglist: part_imglist.htm

imginfolist: part_imginfolist.htm

arclist: part_arclist.htm

参数:

col=" 分多少列显示（默认为单列），5.3 版中本属性可以通过多种方式进行多行显示

如果 col='1'要多列显示的可用 div+css 实现

以下为通过 div+css 实现多列的范例:

```
<style type=text/css>

    div{width:400px;float:left;}

</style>

{dede:arclist row='10' titlelen='24' orderby='pubdate' idlist="" col='2'}
```

```
•[field:textlink/][field:pubdate function=MyDate('m-d',@me)/]<br/>
{/dede:arclist}
```

当 col>1 将使用原来的 table 多列方式显示

row='10' 返回文档列表总数

typeid="" 栏目 ID,在列表模板和档案模板中一般不需要指定,在首页模板中允许用","分开表示多个栏目;

getall='1' 在没有指定这属性的情况下,在栏目页、文章页模板,不会获取以","分开的多个栏目的下级子类

titelen = '30' 标题长度 等同于 titlelength

infolen='160' 表示内容简介长度 等同于 infolength (只控制 infos, 并不控制 description 的内容)

imgwidth='120' 缩略图宽度

imgheight='90' 缩略图高度

listtype='all' 栏目类型 image 含有缩略图 commend 推荐

orderby='sortrank' 文档排序方式

- orderby='hot' 或 orderby='click' 表示按点击数排列
- orderby='sortrank' 或 orderby='pubdate' 按出版时间排列
- orderby='near'
- orderby='lastpost' 按最后评论时间
- orderby='scores' 按得分排序
- orderby='id' 按文章 ID 排序
- orderby='rand' 随机获得指定条件的文档列表

keyword="" 含有指定关键字的文档列表,多个关键字用","分

innertext= " 单条记录样式

aid="" 指定文档 ID

idlist="" 提取特定文档 (文档 ID)

channelid 频道 ID

limit='起始 ID,记录数' (起始 ID 从 0 开始) 表示限定的记录范围 (如: limit='1,2' 表示从 ID 为 1 的记录开始,取 2 条记录)

flag = 'h' 自定义属性值: 头条[h]推荐[c]图片[p]幻灯[f]滚动[s]跳转[j]图文[a]加粗[b]

noflag = " 同 flag, 但这里是表示不包含这些属性

orderway='desc' 值为 desc 或 asc, 指定排序方式是降序还是顺向排序,默认为降序

subday='天数' 表示在多少天以内的文档

底层字段：

ID(同 id),typeid,sortrank,flag,ismake,channel,arcrank,click,money,title,shorttitle,color,writer,
source,litpic(同 picname),pubdate,senddate,mid, lastpost,scores,goodpost,badpost,notpost,
description(同 infos),filename, image, imglink, fulltitle, textlink, typelink,plusurl, memberurl, templeturl,
stime(pubdate 的"0000-00-00"格式)

其中：

```
textlink = <a href='arcurl'>title</a>

typelink = <a href='typeurl'>typename</a>

imglink=<a href='arcurl'><img src='picname' border='0' width='imgwidth' height='imgheight'></a>

image = <img src='picname' border='0' width='imgwidth' height='imgheight' alt=' titile' >
```

注：字段调用方法[field:varname/]，同样适用于其他标签

如：

```
{dede:arclist infolen='100'}

[field:textlink/]

<br>

[field:infos/]

<br>

{/dede:arclist}
```

底层模板里的 Field 实现也是织梦标记的一种形式，因此支持使用 PHP 语法，Function 扩展等功能。

如： 给当天发布的内容加上 (new) 标志

```
[field:senddate runphp='yes']

    $ntime = time();

    $oneday = 3600 * 24;

    if(($ntime - @me)<$oneday) @me = "<font color='red'>(new)</font>";

    else @me = "";

[/field:senddate]
```

arclistsg 独立单表模型文档列表

(DedeCMS > 5.3)

名称: arclistsg

功能: 类似 arclist 标签, 获取指定单表模型 (例如: 分类信息), 指定栏目, 指定排序及呈现样式的一系列文档

语法:

```
{dede:arclistsg  flag='h' typeid=" row=" col=" titlelen=" orderway=" key word=" limit='0,1'}  
  
    <a href="[field:arcurl/]">[field:title/]</a>  
  
{/dede:arclistsg}
```

文件:

\include\taglib\arclistsg.lib.php

function lib_arclistsg(&\$tag,&\$refObj)

默认底层模板:

part_arclistsg.htm

参数:

col=" 分多少列显示 (默认为单列), 5.3 版中本属性无效, 要多列显示的可用 div+css 实现

row='10' 返回文档列表总数

typeid=" 栏目 ID, 在列表模板和档案模板中一般不需要指定, 在封面模板中允许用 "," 分开表示多个栏目;

titlelen = '30' 标题长度 等同于 titlelength

orderway='desc'或'asc' 排序方向

keyword= 含有指定关键字的文档列表, 多个关键字用 "," 分

innertext = "[field:title/]" 单条记录样式 (innertext 是放在标签之间的代码)

arcid=" 指定文档 ID

idlist =" 提取特定文档 (文档 ID)

channelid = " 频道 ID

limit=" 起始, 结束 表示限定的记录范围 (如: limit='1,2')

flag = 'h' 自定义属性值: 头条[h]推荐[c]图片[p]幻灯[f]滚动[s]跳转[j]图文[a]加粗[b]

subday='天数' 表示在多少天以内的文档

底层字段:

ID(同 id),typeid, channel, arcrank, mid, click, title, senddate, flag, litpic(同 picname), userip, lastpost, scores, goodpost, badpost, textlink

范例:

这个标签用于调用单表模型的内容, 在 V5.3 系统中加入了单表模型的概念, 脱离了以前的主从表的数据表关联结构, 一般我们在添加内容模型的时候会选择是否为单表模型还是主从表模型。

我们可以进入系统后台[核心]-[内容模型管理], 查看系统现有的内容模型, 在系统默认环境下, 只有分类信息是单表模型, 一般添加单表模型, 模型的 id 号都小于-1, 即分类信息模型 id 号为-8。

在使用这个标签的时候, 使用方法同 arclist 大致相同, 我们来通过以下的标签内容来调用分类信息的内容:

```
{dede:arclistsg channelid='-8' limit='0,2'}  
  
    <a href="[field:arcurl/]">[field:title/]</a> <br/>  
  
{/dede:arclistsg}
```

这个标签就是调用分类信息的内容, 从 id 为 0 开始调用 2 条记录, 在系统后台的[全局标签测试]中显示以下的内容:

进行状态:

模板代码:

```
{dede:arclistsg channelid='-8' limit='0,2'}  
<a href='[field:arcurl/]'>[field:title/]</a> <br/>  
{/dede:arclistsg}
```

结果:

[网页设计-诚聘网页美工](#)
[大量高价收购二手电脑, 二手显示](#)

arcpagelist arclist 列表分页

(DedeCMS 5.6)

名称: arcpagelist

功能: 通过制定 arclist 的 pagesize 及 tagid 属性, 配合 arcpagelist 标签进行内容当前分页中显示, DedeCMS 5.8 中将取消这个标签的使用

语法:

```
<ul class="c1 ico2">

{dede:arclist flag='c' titlelen=42 row='16' tagid='dedecms' pagesize='8'}

    <li class='dotline'><a href="[field:arcurl/]">[field:title/]</a></li>{/dede:arclist}

</ul>

<div class="c_page">{dede:arcpagelist tagid='dedecms'/}</div>
```

文件:

\include\taglib\arcpagelist.lib.php

lib_arcpagelist(&\$tag, &\$refObj)

默认底层模板:

无

参数:

tagid="" 对应 arclist 的标签名称

底层字段:

无

autochannel 指定栏目

(> DedeCMS 4, DedeCMS 5)

名称: autochannel

功能: 指定排序位置的单个栏目的链接

语法:

```
{dede:autochannel partsort='2' typeid=1}

<a href="[field:typelink/]">[field:typename/]</a>

{/dede:autochannel}
```

文件:

\include\taglib\autochannel.lib.php

function lib_autochannel(&\$tag, &\$refObj)

默认底册模板:

part_autochannel.htm

参数:

partsort = '0' 栏目所在的排序位置

typeid='0' 获取单个栏目的顶级栏目

底层字段:

id,typelink,typedir,isdefault,defaultname,ispart,namerule2,typename,moresite,siteurl,sitepath,listman

cattree 树形类目

(>DedeCMS 5.3)

名称: cattree

功能: 树形类目标签

语法:

```
{dede:cattree typeid=" catid=" showall="/}
```

文件:

\include\taglib\cattree.lib.php

lib_cattree(&\$ctag, &\$refObj)

默认底层模板: 无

参数:

typeid="" 顶级树 id

catid=0 上级栏目 id

showall="" 在空或不存在时, 强制用产品模型 id; 如果是 yes 则显示整个语言区栏目树; 为其它数字则是这个数字的模型的 id

底层字段:

无

channel 获取栏目列表标签

(>DedeCMS 3, DedeCMS 4, DedeCMS 5)

名称: channel

功能: 织梦常用标记, 通常用于网站顶部以获取站点栏目信息, 方便网站会员分类浏览整站信息

语法:

```
{dede:channel type='top' row='8' currentstyle="<li><a href='~typelink~' class='thisclass'>~typename~</a> </li>"}
<li><a href="[field:typelink/]">[field:typename/]</a> </li>
{/dede:channel}
```

文件:

\include\taglib\channel.lib.php

function lib_channel(&\$tag,&\$refObj)

默认底层模板:

channel_list.htm

参数:

typeid = '0' 栏目 ID

reid = '0' 上级栏目 ID

row = '100' 调用栏目数

col = '1' 分多少列显示（默认为单列）

type = 'son | sun' son 表示下级栏目,self 表示同级栏目,top 顶级栏目

currentstyle = " 应用样式

底层字段:

ID(同 id),typeid,typelink,typename,typeurl,typedir(仅表示栏目的网址) 例:

```
{dede:channel type='top'} <a href="[field:typelink /]">[field:typename/]</a> {/dede:channel}
```

currentstyle 中支持~rel~、~id~、~typelink~、~typename~

注: 在没有指定 typeid 的情况下, type 标记与模板的环境有关, 如, 模板生成到栏目一, 那么 type='son' 就表示栏目一的所有子类

范例:

这个标签是全局常用标记, 主要用于显示页面的栏目分类, 我们可以查看默认模板\templets\default\head.htm 中的相关代码:

```
{dede:channel type='top' currentstyle="<li class='thisclass'><a href='~typelink~'>~typename~</a> </li>"}
<li><a href="[field:typeurl/]">[field:typename/]</a></li>{/dede:channel}
```

这里的栏目可以通过后台进行设置, 栏目显示顺序按照排序的高低进行, 我们可以在系统后台[核心]-[栏目管理]中进行栏目设置:

网站栏目管理		增加顶级栏目	批量增加栏目	更新栏目缓存	更新栏目HTML
提示：可使用右键菜单进行操作。 [展开全部]					
<input type="checkbox"/>	新闻资讯[ID:1]	预览 内容 增加子类 更改 移动 删除		0	
<input type="checkbox"/>	• 织梦动态[ID:9] (文档:19)	预览 内容 增加子类 更改 移动 删除		0	
<input type="checkbox"/>	• 行业动态[ID:10] (文档:44)	预览 内容 增加子类 更改 移动 删除		1	
<input type="checkbox"/>	软件下载[ID:2] (文档:8)	预览 内容 增加子类 更改 移动 删除		1	
<input type="checkbox"/>	图片展示[ID:3] (文档:4)	预览 内容 增加子类 更改 移动 删除		2	
<input type="checkbox"/>	分类信息[ID:5] (文档:4)	预览 内容 增加子类 更改 移动 删除		3	
<input type="checkbox"/>	图书展示[ID:8] (文档:4)	预览 内容 增加子类 更改 移动 删除		4	
<input type="checkbox"/>	商品展示[ID:4] (文档:8)	预览 内容 增加子类 更改 移动 删除		5	
<input type="checkbox"/>	专题[ID:6]	预览 更改 移动 删除		6	
<input type="checkbox"/>	在线留言[ID:7]	预览 更改 移动 删除		999	
<input type="button" value="更新排序"/> <input type="button" value="获取JS"/> <input type="button" value="更新栏目HTML"/> <input type="button" value="更新文档HTML"/>					

设置完成我们可以在首页顶部看到相关的效果



当然这里涉及到一个顶级栏目和子级栏目显示的问题，我们可以在模板中使用相应的标签进行调用，我们来看一个封面的页面内容：



我们查看这个文章封面\templates\default\index_article.htm 的模板标签

```
<ul>

{dede:channel type='son' currentstyle="<li class='thisclass'><a href='~typelink~'><span>~typename~</span></a> </li>"}

    <li><a href='[field:typelink/]'><span>[field:typename/]</span></a></li>{/dede:channel}

</ul>
```

这里就使用了 type='son'这个属性用来显示子栏目，currentstyle 则代表当前选中项的呈现样式。

很多用户希望顶部导航连接便于 SEO，需要去除超链接中的“index.html”，我们可以在这里对标签进行一个修改：

```
{dede:channel type='top'}

<li><a href='[field:typelink function='str_replace("index.html","",@me)'/]'>[field:typename/]</a></li>{/dede:channel}
```

channelartlist 当前频道的下级栏目的内容列表

(>DedeCMS，DedeCMS 4，DedeCMS 5)

名称：channelartlist

功能：获取当前频道的下级栏目的内容列表标签

语法：

```
{dede:channelartlist row=6}

<dl>
```

```
<dt><a href='{dede:field name='typeurl'}'>{dede:field name='typename'}</a></dt>

<dd>

{dede:arclist titlelen='42' row='10'}    <ul class='autod'>

    <li><a href="[field:arcurl /]">[field:title /]</a></li>

    <li>([field:pubdate function="MyDate('m-d',@me)"/])</li>

</ul>

{/dede:arclist}

</dl>

{/dede:channelartlist}
```

文件:

\include\taglib\channelartlist.lib.php

function lib_channelartlist(&\$tag,&\$refObj)

默认底层模板:

part_channelartlist.htm

参数:

typeid = '0' 频道 ID,多个请用","分开, 如果调用 1 个栏目可以使用 typeid='5,0'来调用。分隔 row = '20' 获取的栏目返回值 其他说明: 除了宏标记外, channelArtlist 是唯一一个可以直接嵌套其它标记的标记, 不过仅限于嵌套 {dede:type} {/dede:type} 和 {dede:arclist} {/dede:arclist} 两个标记。

底层字段:

包含 {dede:type} {/dede:type} 及 {dede:arclist} {/dede:arclist} 下面所有底层模板变量。

范例:

这个标签是系统中不多的一个支持嵌套的标签, 这个标签通常使用在首页 (含封面首页), 用于输出一组栏目内容列表, 我们可以看到默认模板首页:



这部分的内容就是通过这个标签进行显示的，我们可以查看首页模板\templets\default\index.htm

```
{dede:channelartlist}
```

```
<dl class="tbody">
```

```
<dt><strong><a href="{dede:field name='typeurl'}/">{dede:field name='typename'}/</a></strong><span class="more"><a
```

```
href="{dede:field name='typeurl'}/">更多...</a></span></dt>
```

```
<dd>
```

```
<ul class="d1 ico3">
```

```
{dede:arclist titelen='60' row='8'}
```

```
<li><span class="date">[field:pubdate function="MyDate('m-d',@me)"/]</span><a href="[field:arcurl /]">[field:title /]</a></li>
```

```
{/dede:arclist}
```

```
</ul>
```

```
</dd>
```

```
</dl>
```

```
{/dede:channelartlist}
```

feedback 评论内容

(>DedeCMS 4, DedeCMS 5)

名称: feedback

功能: 用于调用最新评论

语法:

```
{dede:feedback}

<ul>

<li class='fbtitle'>[field:username function="(@me=='guest'? '游客': @me)"/] 对 [field:title/] 的评论: </li>

<li class='fbmsg'><a href="plus/feedback.php?aid=[field:aid/]" class='fbmsg'>[field:msg /]</a></li>

</ul>

{/dede:feedback}
```

文件:

文件\include>taglib\feedback.lib.php

function lib_feedback(&\$ctag,&\$refObj)

默认底层模板:

tag_feedback.htm

参数:

row='12' 调用评论条数

titlen='24' 标题长度

infolen='100' 评论长度

底层字段:

id,aid,typeid,username,arctitle,ip,iscache,dtime,mid,bad,good,ftype,face,msg,title

范例:

这个标签主要调用系统的会员评论信息，我们在默认模板首页可以查看到相关的内容：



我们可以查看首页的模板\templets\default\index.htm 有如下代码:

```
{dede:feedback row='5' titlelen='24' infolen='80'}

<li> <small><a href="#" class="username">[field:username function="(@me=='guest' ? '游客' : @me)"/]</a> 评论 <a href="[field:global.cfg_phpurl]/feedback.php?aid=[field:aid]" class="title">[field:title]/</a></small>

<p>[field:msg]/</p>

</li>

{/dede:feedback}
```

这里调用的是会员最新评论的内容,当然我们也可以在系统后台的[核心]-[评论管理]中对现有的评论进行审核和编辑:



在[系统]-[系统基本参数] 的”互动设置“中也有评论的相关设置选项，可以控制会员评论。

field 内容变量

名称: field

功能: 用于获取特定栏目或者档案的字段值及常用的环境变量值

语法:

{dede:field name='字段名'/} 或者 {dede:field.字段名/}

文件:

所有视图类中

默认底层模板:

无

参数:

板块模板: phpurl,indexurl,indexname,templeturl,memberurl,specurl

列表模板: position,title,phpurl,templeturl,memberurl,indexurl,indexname,specurl,栏目表 dede_arctype 的所有字段其中 position 为 “栏目一 > 栏目二” 这样形式的链接，title 则为这种形式的标题

文档模板: position,phpurl,templeturl,memberurl,indexurl,indexname,specurl,id(同 ID,aid)

文档所有内容字段: 档案#@__archives 表 和 附加表的所有字段。

flink 友情链接

(>=DedeCMS 3, DedeCMS 4, DedeCMS 5)

名称: flink

功能: 用于获取友情链接, 在 DedeCMS 系统中已经默认集成友情链接插件, 所以无需其他安装即可使用该标签。

语法:

```
{dede:flink row='24'/{}
```

文件:

\include\taglib\flink.lib.php

```
function lib_flink(&$ctag,&$refObj)
```

默认底层模板:

```
<li>[field:link /]</li>
```

参数:

type='textall' 链接类型

row=24 链接数量

titlelen=24 站点标题文字的长度

linktype=1 链接位置内页

typeid=0 友情链接类型, 默认为所有类型, 可以在系统后台[模块]-[友情链接]中的“网站类型管理”中查看。

类型 ID	网站名称	状态
1	综合网站	<input checked="" type="checkbox"/> 保留
2	娱乐类	<input checked="" type="checkbox"/> 保留
3	教育类	<input checked="" type="checkbox"/> 保留
4	计算机类	<input checked="" type="checkbox"/> 保留
5	电子商务	<input checked="" type="checkbox"/> 保留
6	网上信息	<input checked="" type="checkbox"/> 保留
7	论坛类	<input checked="" type="checkbox"/> 保留
8	其它类型	<input checked="" type="checkbox"/> 保留
新增一个网站类型:		
	<input type="text"/>	<input checked="" type="checkbox"/> 新增
<input type="button" value="确定"/>		

底层字段:

url,webname,logo,link

flinktype 友情链接类型

(>= DedecMS 5.7)

名称: flinktype

功能: 用于获取友情链接类型

语法:

```
{dede:flinktype type="dedecms"}

    <a href="#" _for="flink_[field:id/]">[field:typename/]</a>

{/dede:flinktype}
```

文件:

\include\taglib\flinktype.lib.php

function lib_flinktype(&\$ctag,&\$refObj)

默认底层模板:

```
<li>[field:typename /]</li>
```

参数:

row=24 链接类型数量

tidelen=24 链接文字的长度

底层字段:

id,typename

范例:

```
{dede:flinktype type="dedecms"}

    <a href="#" _for="flink_[field:id/]">[field:typename/]</a>

{/dede:flinktype}
```

同时 flinktype 可以同 flink 标签进行嵌套使用,用于循环生成分类标签,类似 channelarclist 标签。

```
{dede:flinktype type="dedecms"}

<dd id="flink_[field:id/]">

    <ul class="f5">

        {dede:flink/}
```

```
</ul>

</dd>

{/dede:flinktype}
```

global 全局配置变量标签

(>=DedeCMS 3, DedeCMS 4, DedeCMS 5)

名称: global

功能: 获取系统全局配置变量内容

语法:

```
{dede:global.cfg_webname/}
```

[field:global.cfg_webname/] 用于底层模板中

文件:

系统模板引擎

默认底层模板:

无

参数:

无

底层字段:

无

hotwords 搜索关键词

(>=DedeCMS 3, DedeCMS 4, DedeCMS 5)

名称: hotwords

功能: 获取网站搜索的热门关键字

语法:

```
{dede:hotwords num=6 subday=365 maxlength=16/}
```

文件:

\include>taglib\hotwords.lib.php

function lib_hotwords(&\$ctag,&\$refObj)

默认底层模板：

无

参数：

num=6 关键词数目

subday=365 调用天数以内的关键词

maxlength=16 关键词最大长度

底层字段：

无

json JSON 数据调用

(>=DedeCMS 5.7)

名称： json

功能： 调用某个远程连接的 json 接口，方便同远程站点之间进行通信来调取内容

语法：

```
{dede:json url='http://yoursite/json.php' cache=300}

    [field:id/]-[field:title/]<br/>

{/dede:json}
```

文件：

\include\taglib\json.lib.php

function lib_json(&\$ctag,&\$refObj)

默认底层模板：

无

参数：

url=" 远程 json 接口地址

注：所传递的数据需要是 UTF-8 编码格式，否则可能会出现乱码。

cache=300 缓存时间，0 为不缓存

底层字段：

json 传递数组字段

范例：

我们首先编写一个简单的服务器端 API，将其存放在系统根目录/api 下，由于系统是 GB2312 编码，所以服务端编写的时候我们进行了一些编码转换的处理，保证服务端访问的编码是 UTF-8 即可。

```
<?php

$cfg_NotPrintHead = false;

header("Content-Type: text/html; charset=utf-8");

include_once (dirname(__FILE__)."/../include/common.inc.php");

error_reporting(E_ALL || ~E_NOTICE);

require_once(DEDEINC.'/json.class.php');

$reval = array();

$db->SetQuery("SELECT id,title FROM `#@__archives` ORDER BY id DESC LIMIT 0,10");

$db->Execute('me');

while ($row = $db->GetArray('me')) {

    $row['title'] = gb2utf8($row['title']);

    $reval[] = $row;

}

$json = new Services_JSON(SERVICES_JSON_SUPPRESS_ERRORS);

echo $json->encode($reval);

?>
```

通过地址 <http://www.dede.com/api/article.php> 我们可以访问到刚才编写的服务端文件。

如果您是想获取单条记录，可以构造如下的数组：

```
$row[] = $dsq->GetOne("SELECT * FROM #__archives WHERE id=88");
```

然后将这个数组通过 JSON 格式化进行处理。

loop 万能调用标签

(>=DedeCMS 3, DedeCMS 4, DedeCMS 5)

名称：loop

功能：调用任意表的数据标签

语法：

```
{dede:loop table='dede_archives' sort="" row='4' if=""}  
<a href=[field:arcurl/]>[field:title/]</a>  
{/dede:loop}
```

文件：

\include\taglib\loop.lib.php

function lib_loop(&\$tag,&\$refObj)

默认底层模板：

无

参数：

table="" 数据表名称

tablename="" 同 table

row=8 调用数据的行数

sort="" 用于排序的字段

if="" 查询的条件

ifcase="" 同 if

orderway='desc' 排序方式，desc 降序、asc 升序

底层字段：

这个标记的底层模板变量即是被查询表的所有字段

范例：

我们来查询用户 id 为 1 的用户发布的文档，可以用下面的标签来表示：

```
{dede:loop table='#@__archives' sort='mid' row='4' if='mid=1'}
<a href=[field:arcurl/]>[field:title/]</a>([field:mid/])<br />
{/dede:loop}
```

查询结果如下：

进行状态：

模板代码：

```
{dede:loop table='#@__archives' sort='mid' row='4' if='mid=1'}
<a href='[field:arcurl/]'>[field:title/]</a>([field:mid/])<br />
{/dede:loop}
```

结果：

[学习HTML 4.0事件属性\(1\)](#)
[初学者：XHTML常用标签的基本应用指南\(1\)](#)
[网页中应用表格爱好者\(1\)](#)
[初学者很好的参考：HTML标签详尽讲解\(1\)](#)

loop 标签同时可以查询跟 DedeCMS 处在同一个数据库的数据表，如果您的论坛或者商城程序安装在同一数据库，也可以使用 loop 标签进行调用。

memberlist 会员列表

(>=DedeCMS 4, DedeCMS 5)

名称：memberlist

功能：调用会员列表

语法：

```
{dede:memberlist orderby='scores' row='20'}

    <a href=" ../member/index.php?uid={dede:field.userid /}">{dede:field.userid /}</a>

    <span>{dede:field.scores /}</span>

{/dede:memberlist}
```

文件：

\include\taglib\memberlist.lib.php

function lib_memberlist(&\$ctag, &\$refObj)

默认底层模板：

Copyright ©2002 - 2012 DesDev INC.All rights reserved.

memberlist.htm

参数:

row = '6' 调用数目

iscommend = '0' 是否为推荐会员

orderby = 'logintime' 按登陆时间排序 money 按金钱排序 scores 按积分排序

signlen=50 签名长度

底层字段:

#@__member 所有字段及 spacename(空间名称)、sign(用户签名)

范例:

用于调用会员相关的列表信息，我们可以在默认模板中查看\templets\default\index.htm

```
<div class="latestlogin"> <strong> 最近登陆的会员</strong>

<ul class="e7">

{dede:memberlist row=6 signlen=30}

<li><a href="[field:spaceurl/]" target="_blank">[field:uname/]</a></li>

{/dede:memberlist}

</ul>

</div>

<!-- /latestlogin -->
```

我们可以查看网站首页部分:

且在这一版本中进一步加强了模块的概念
同的模型组合 ...[查看全文]

DeDe V

功能盘点之：

功能盘点之：

更新，清晰

- 中移动利用奥运契机建WiMAX网
- 软件安装服务
- somebody 's
- 高通卖专利起家终成3G时代霸主
- 央行提取个人信用信息 只针对先
- Sprint Q1亏损2.11亿美元 全年



我的留言

发表文章

访客记录

我的收藏

好友管理

查找好友

[会员中心](#) | [资料](#) | [空间](#) | [退出登录](#)

最近登陆的会员



及时雨



天涯



木耳



木林森



IT柏拉图

当然这里也可以通过指定 orderby 的属性来调用不同形式排序的内容。

myad 广告

(>=DedeCMS 3, DedeCMS 4, DedeCMS 5)

名称：myad

功能：调用广告内容

语法：

```
{dede:myad typeid="" name=""/}
```

文件：

\include\taglib\myad.lib.php

```
function lib_myad(&$ctag, &$refObj)
```

默认底层模板：

无

参数：

name="" 标记名称，该项是必须的属性，以下 2、3 是可选属性

ismake='no' 默认是 no 表示设定的纯 HTML 代码， yes 表示含板块标记的代码

typeid=0 表示所属栏目的 ID，默认为 0，表示所有栏目通用的显示内容，在列表和文档模板中，typeid 默认是这个列表或文档本身的栏目 ID

底层字段：

无

范例：

在系统后台可以添加一个广告



广告位标识就是广告标签的 id，那么可以使用：

```
<div class="banner">{dede:myad name='innerTopBanner1'}</div>
```

来调用出对应的广告位内容。



可以直接在广告管理页面，点击“代码”选项来获取对应的代码。

◇广告管理::获取JS

说明：如果嵌入的是织梦CMS广告标签，那么将会解析成标签中的内容到页面，广告更改后需要更新页面。如果不希望重新生成所有页面，则直接调用JS代码即可。

织梦CMS标签调用代码：

```
{dede:myad name='indexTopBanner1' /}
```

以下为选定广告的JS调用代码：

```
<script src='http://www.dede.com/plus/ad_js.php?aid=1' language='javascript'></script>
```

预览：



mynews 站内新闻

(>=DedeCMS 3, DedeCMS 4, DedeCMS 5)

名称：mynews

功能：站内新闻

语法：

```
{dede:mynews row=" " titlelen=""}
<b>[field:title /]</b>
发布人： [field:writer /]
时间： [field:senddate function='MyDate("Y-m-d H:i:s","@me")']/]
<p>[field:body /]</p>
{/dede:mynews}
```

文件：

\include\taglib\mynews.lib.php

function lib_mynews(&\$tag,&\$refObj)

默认底层模板：

mynews.htm

参数：

Copyright ©2002 - 2012 DesDev INC.All rights reserved.

row=1 调用站内新闻数

titlelen=24 新闻标题长度

底层字段:

aid,typeid,title,writer,senddate,body

范例:

系统默认已经安装了站内新闻的插件，我们直接在后台[模块]-[站内新闻]选项卡中来添加站内新闻。



我们使用下面的标签:

```
{dede:mynews row='2' titlelen=''}
<b>[field:title /]</b>
发布人: [field:writer /]
时间: [field:senddate function='MyDate("Y-m-d H:i:s","@me")']/<br />
{/dede:mynews}
```

测试后标签调用信息内容如下:

进行状态:

模板代码:

```
[dede:mynews row='2' titlelen='']/]
<b>[field:title /]</b>
发布人: [field:writer /]
时间: [field:senddate function='MyDate("Y-n-d H:i:s","@ne")' /]<br />
[/dede:mynews]
```

结果:

DedeCMS V5.7SP1更新包现提供版主、技术联盟成员下载测试 发布人: admin
织梦团队2011年国庆节放假通知 发布人: admin 时间: 2012-03-02 09:16:09

mytag 自定义宏标签

(>=DedeCMS 3, DedeCMS 4, DedeCMS 5)

名称: mytag

功能: 自定义宏标记调用标签

语法:

```
{dede:mytag typeid='0' name='/'}
```

文件:

\include\taglib\mytag.lib.php

```
function lib_mytag(&$ctag, &$refObj)
```

默认底层模板:

无

参数:

name="" 标记名称, 该项是必须的属性, 以下 2、3 是可选属性

ismake='no' 默认是 no 表示设定的纯 HTML 代码, yes 表示含板块标记的代码

typeid=0 表示所属栏目的 ID, 默认为 0, 表示所有栏目通用的显示内容, 在列表和文档模板中, typeid

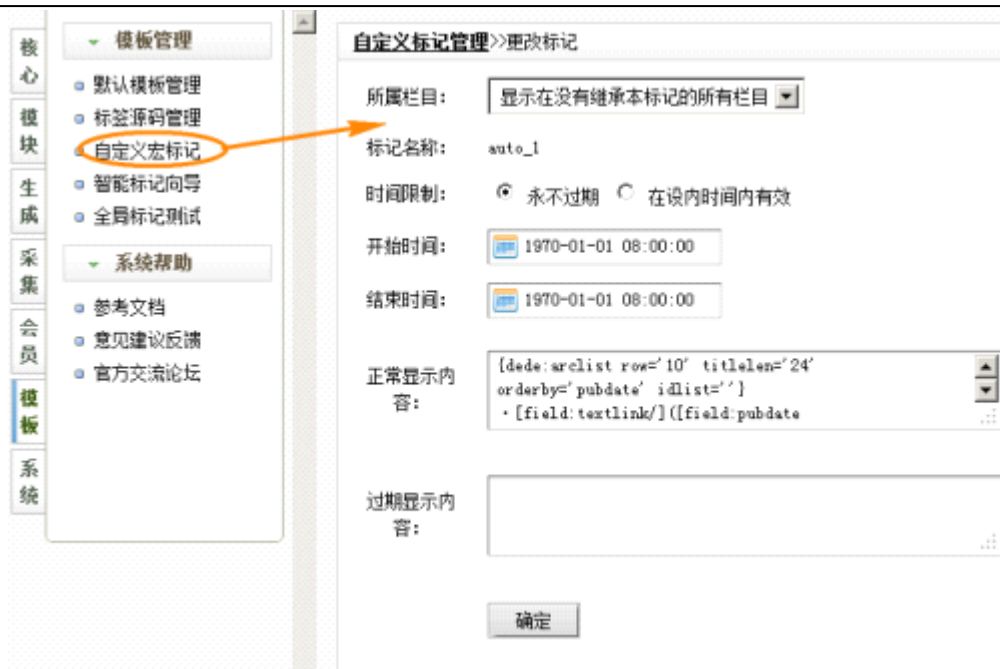
默认是这个列表或文档本身的栏目 ID

底层字段:

无

范例:

可以在系统后台[模板]-[自定义宏标记]中增加一个内容, 这个内容可以是一段普通的 HTML, 也可以是织梦的某个标签块。



这样我们可以直接使用标签：

{dede:mytag name='auto_1' ismake='yes'}/}来调用上面的内容。

进行状态：

模板代码：

```
{dede:mytag name='auto_1' ismake='yes'}/}
```

结果：

- [bbbbbbbbbbbbbbbbbb](#) (03-02)
- [sdfsdfsdfsds](#) (03-02)
- [微软发布Windows 8消费者](#) (03-02)
- [微软发布Windows 8消费者](#) (03-02)
- [省的烦随碟附送多风生多范](#) (02-29)
- [aaaaaaaaaaaaaaaaaaaaa](#) (02-28)
- [bbbbbbbbbb](#) (02-27)
- [sdfsdfs](#) (02-27)
- [asdasdasds](#) (02-27)

php php 标签

(>=DedeCMS 5.2)

名称： php

功能： 执行 PHP 代码

语法：

```
{dede:php}  
  
$a = "dede";  
  
echo $a;  
  
{/dede:php}
```

文件：

\include\taglib\php.lib.php

function lib_php(&\$ctag, &\$refObj)

默认底层模板：

无

参数：

无

底层字段：

无

范例：

进行状态：

模板代码：

```
{dede:php}  
$a = "dede";  
echo $a;  
{/dede:php}
```

结果：

dede

注：PHP 标签由于采用直接执行 php 代码，在使用的时候需要注意其安全性。

softmsg 软件下载提示信息

(>=DedeCMS 5.3)

名称: softmsg

功能: 调用下载说明

语法:

```
{dede:softmsg /}
```

文件:

```
\include\taglib\softmsg.lib.php
```

```
function lib_softmsg(&$tag,&$refObj)
```

默认底层模板:

无

参数:

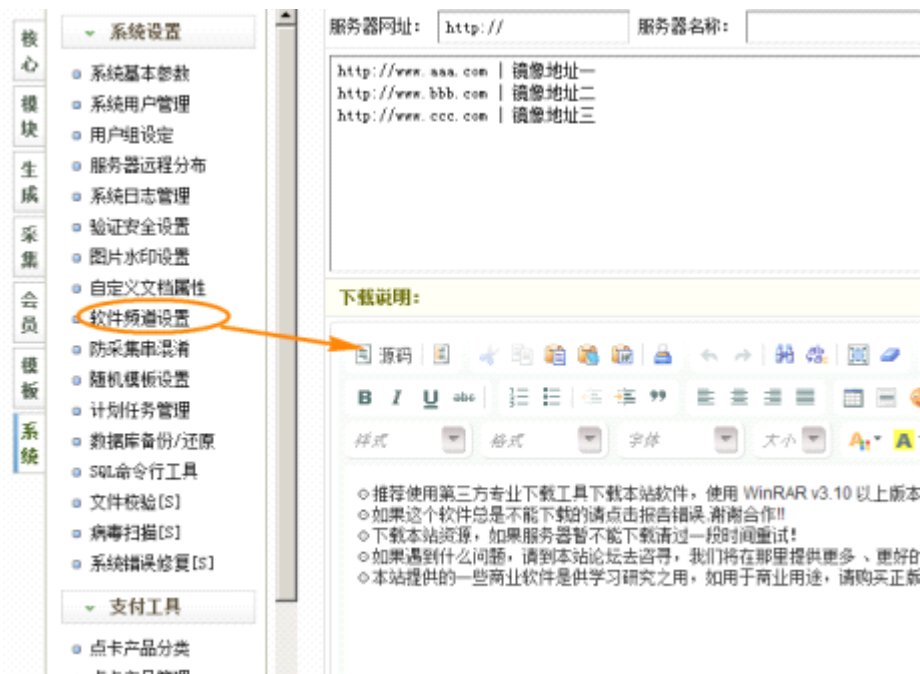
无

底层字段:

无

范例:

在系统后台[系统]-[软件频道设置]中配置下载说明:



使用标签就可以调用这里的信息。

进行状态:

模板代码:

```
{dede:softmsg /}
```

结果:

- 推荐使用第三方专业下载工具下载本站软件, 使用 WinRAR v3
- 如果这个软件总是不能下载的请点击报告错误, 谢谢合作!!
- 下载本站资源, 如果服务器暂不能下载请过一段时间重试!
- 如果遇到什么问题, 请到本站论坛去咨寻, 我们将在那里提供
- 本站提供的一些商业软件是供学习研究之用, 如用于商业用途

sonchannel 子栏目调用

(>=DedeCMS 3, DedeCMS 4, DedeCMS 5)

名称: sonchannel

功能: 子栏目调用标签

语法:

```
{dede:sonchannel}  
<a href=[field:typeurl/]>[field:typename/]</a>  
{/dede:sonchannel}
```

文件:

\include>taglib\sonchannel.lib.php

function lib_sonchannel(&\$ctag,&\$refObj)

默认底层模板:

无

参数:

row=100 返回数目

col=1 默认单列显示

nosonmsg="" 没有指定 ID 子栏目显示的信息内容

底层字段:

id,typename,typedir,isdefault,ispart,defaultname,namerule2,moresite,siteurl,sitepath,listman

范例：

这个标签使用的时候需要注意要在某个特定栏目下才能够正常被调用。

主用标记指定应用在网站主页、单独页面、频道页面使用的半独立的模板标记，在列表记），但是环境变量限定为文章或列表所在的栏目，如果您要测试的标记是在列表或文章中各标记的具体含义和用途，请在模板标记参考一章查阅。

输入要测试的局部代码：

```
{dede:sonchannel}
<a href=' [field: typeurl/]' >[field: typename/]</a>
{/dede:sonchannel}
```

环境变量：

进行状态：

模板代码：

```
{dede:sonchannel}
<a href=' [field: typeurl/]' >[field: typename/]</a>
{/dede:sonchannel}
```

结果：

[HTML](#) [DIV&CSS](#) [Javascript/Ajax](#) [Dreamweaver](#)

sql SQL 查询标签

(>=DedeCMS 3, DedeCMS 4, DedeCMS 5)

名称：sql

功能：用于获取 MySQL 数据库内容的标签

语法：

```
{dede:sql sql=" appname='default'}
[field:title/]
{/dede}
```

文件：

```
\include\taglib\sql.lib.php
```

```
function lib_sql(&$ctag,&$refObj)
```

默认底层模板:

无

参数:

sql="" 需要查询的 sql 语句

appname='default' 这个参数在 DedeCMS 5.7 中有效, 可以调用其他服务器或者数据库上的信息, 默认为 default, 即当前系统所运行的数据库, 可在 data/tag/sql.inc.php 中扩展, 在 DedeCMS 5.8 中用户可以通过后台[模板]-[SQL 标签应用管理]来配置应用。



底层字段:

所有 SQL 语句查询出来的字段

范例:

我们通过下列标签来调用 phpwind 论坛的会员信息。

```
{dede:sql sql='SELECT uid,username,email,memberid FROM `phpwind87`.pw_members ORDER BY uid desc'
appname='phpwind'}

[field:uid/]-[field:username/]<br />

{/dede:sql}
```

标签调试结果内容如下:

进行状态:

模板代码:

```
{dede:sql sql='SELECT uid,username,email,memberid FROM `phpwir`  
[field:uid/][field:username/]<br />  
{/dede:sql}
```

结果:

```
4test  
3ff  
2asd  
1admin
```

注：在 DedeCMS 5.8 中可以直接使用标签生成器来生成 SQL 标签。

tag TAG 调用标签

(>=DedeCMS 4, DedeCMS 5)

名称: tag

功能: TAG 调用

语法:

```
{dede:tag sort='new' getall='0'}  
<a href=[field:link/]>[field:tag /]</a>  
{/dede:tag}
```

文件:

\include\taglib\tag.lib.php

function lib_tag(&\$ctag,&\$refObj)

默认底层模板:

tag_one.htm

参数:

row=30 调用条数

sort='new' 排序方式 month, rand, week, hot, total

底层字段:

keyword,tag,link,highlight,

type 指定栏目调用标签

(>=DedeCMS 3, DedeCMS 4, DedeCMS 5)

名称: type

功能: 获取指定栏目链接

语法:

```
{dede:type}

<a href="[field:typelink /]">[field:typename /]</a>

{/dede:type}
```

文件:

\include\taglib\type.lib.php

function lib_type(&\$ctag,&\$refObj)

默认底层模板:

part_type_list.htm

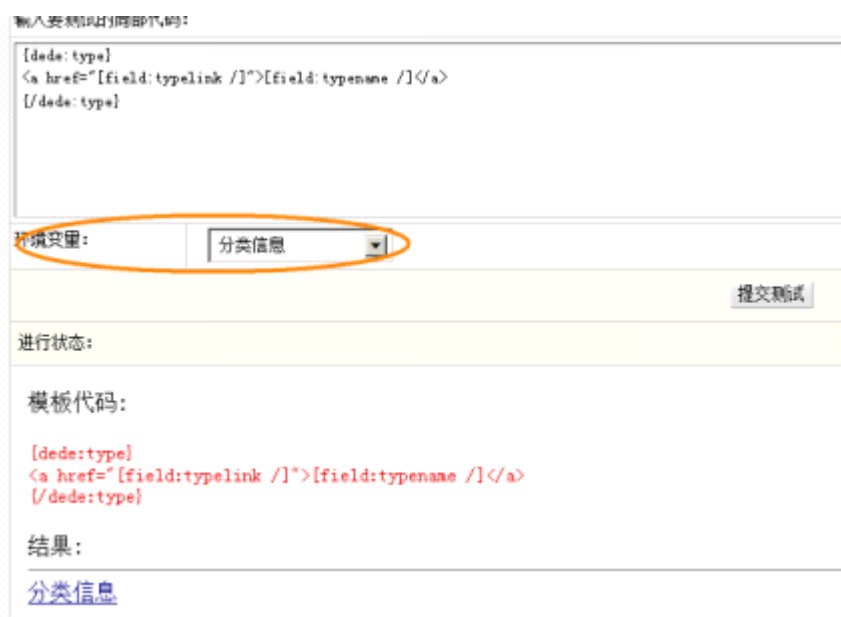
参数:

typeid=0 指定栏目 ID, 如果没有指定则获取当前页面下的环境变量

底层字段:

id,typelink,typename,typedir,isdefault,ispart,defaultname,namerule2,moresite,siteurl,sitepath,listman

范例:



The screenshot shows a web application interface. At the top, there is a text input field with the placeholder text "输入要测试的源代码:". Below this is a code editor containing the following code:

```
[dede: type]
<a href="[field:typelink /]">[field:typename /]</a>
[/dede: type]
```

Below the code editor, there is a section with a label "环境变量:" and a dropdown menu. The dropdown menu is currently set to "分类信息". This section is highlighted with an orange oval. To the right of the dropdown menu is a button labeled "提交测试".

Below this section, there is a label "进行状态:". Underneath, there is a label "模板代码:" followed by the same code as in the code editor:

```
[dede: type]
<a href="[field:typelink /]">[field:typename /]</a>
[/dede: type]
```

Below the template code, there is a label "结果:". At the bottom, there is a link labeled "分类信息".

vote 投票标签

(>=DedeCMS 3, DedeCMS 4, DedeCMS 5)

名称: vote

功能: 用于获取一组投票表单，系统默认安装了投票插件，可在[模块]-[辅助插件]-[投票模块]中来添加或者获取投票标签

语法:

```
{dede:vote id=" lineheight='22' tablewidth='100%' titlebgcolor='#EDEDE2' titlebackground=" tablebgcolor='#FFFFFF'}  
{/dede}
```

文件:

\include\taglib\vote.lib.php

function lib_vote(&\$ctag,&\$refObj)

默认底层模板:

无

参数:

id=0 数字，当前投票 ID

lineheight=24 表格高度

tablewidth='100%' 表格宽度

titlebgcolor='#EDEDE2' 投票标题背景色

titlebackground='' 标题背景图

tablebg='#FFFFFF' 投票表格背景色

底层字段:

无

6.2 列表标签

list 列表数据标签

(>=DedeCMS 3, DedeCMS 4, DedeCMS 5)

名称: list

功能: 获取列表模板中的列表内容

语法:

```
{dede:list col=" titlelen="
infolen=" imgwidth=" imgheight=" orderby=" pagesize="} {/dede:list}
```

文件:

\include\arc.listview.class.php

\include\arc.splistview.class.php

```
function GetArcList($limitstart=0,$row=10,$col=1,$titlelen=30,$infolen=250,
$imgwidth=120,$imgheight=90,$listtype="all",$orderby="default",$innertext="",
$stablewidth="100",$ismake=1,$
orderWay='desc')
```

默认底层模板:

list_fulllist.htm

参数:

col=1 内容列数

titlelen=30 标题长度

infolen=250 内容摘要长度

imgwidth=120 缩略图宽

imgheight=90 缩略图高

orderby='default' 排序方式，有效的排序方式有 senddate、pubdate、id、click、lastpost、postnum，默认为 sortrank

pagesize=20 分页大小

orderway='desc' 排序方式

底层字段:

ID(同 id),title,iscommend,color,typeid,ismake,description(同 info),postnum,lastpost,shorttitle
pubdate,senddate,arcrank,click,litpic(同 picname),typedir,typename,
arcurl(同 filename),typeurl,stime(pubdate 的"0000-00-00"格式),
textlink,typelink,imglink,image

范例:

```
{dede:list pagesize='10'}
```

```
<li> [field:array runphp='yes']@me = (empty(@me['litpic']) ? "" : "<a href='{@me['arcurl']}' class='preview'><img src='{@me['litpic']}' /></a>"); [/field:array]

[<b>[field:typelink/]</b>] <a href="[field:arcurl]" class="title">[field:title/]</a> <span class="info"> <small> 日期 :
</small>[field:pubdate function="GetDateTimeMK(@me)"/] <small> 点 击 : </small>[field:click/] <small> 好 评 :
</small>[field:scores/] </span>

<p class="intro"> [field:description/]... </p>

</li>

{/dede:list}
```

list 标签同时也适用单表模型文档列表。

注：list 标记除了支持档案表的基本模板变量外，还支持附加表的字段，你可以在模型管理中知道附加表支持列表使用的字段有哪些。

pagelist 列表分页标签

(>=DedeCMS 3, DedeCMS 4, DedeCMS 5)

名称：pagelist

功能：表示分页页码列表

语法：

```
{dede:pagelist listsize='3' listitem='/'}
```

文件：

```
\include\arc.listview.class.php
```

```
\include\arc.sglistview.class.php
```

```
function GetPageListDM($list_len,$listitem="index,end,pre,next,pageno")
```

```
function GetPageListST($list_len,$listitem="index,end,pre,next,pageno")
```

默认底层模板：

无

参数：

listsize=3 表示 [1][2][3] 这些项的长度 x 2

listitem='index,pre,pageno,next,end,option' 表示页码样式，可以把下面的值叠加

index 首页

pre 上一页

pageno 页码

next 下一页

end 末页

option 下拉跳转框

底层字段:

无

范例:

```
{dede:pagelist listsize='3' listitem='index pre pageno next end option'/'}
```

注: pagelist 标签是同 list 一同使用来调用列表数据的。

6.3 内容标签

likearticle 相关文档

名称: likearticle

功能: 通过关键词自动关连文档, 这个标签随着数据量的增加可能会比较影响检索性能, 不建议大数据站点使用

语法:

```
{dede:likearticle row=" col=" titlelen=" infolen="}  
<a href=[field:arcurl/]>[field:title/]</a>  
{/dede:likearticle}
```

文件:

\include\taglib\likearticle.lib.php

function lib_likearticle(&\$ctag,&\$refObj)

默认底层模板:

part_arclist.htm

参数:

row=12 调用行数, 默认 12 条

titlelen=28 标题关键词字数

infoLen=150 描述信息字符大小

col=1 分多少列显示（默认为单列），5.3 版中本属性可以通过多种方式进行多行显示

tablewidth=100 表格宽度

mytypeid=0 手工指定要限定的栏目 id，用,分开表示多个

byabs=0 是否绝对排序

imgwidth=120 缩略图宽度，默认 120

imgheight=90 缩略图高度，默认 90

底层字段：

ID(同 id),typeid, channel, arcrank, mid, click, title, fulltitle, textlink, info, filename, typeurl, stime,senddate, litpic(同 picname)

pagebreak 内容分页标签

名称：pagebreak

功能：表示文档的分页链接列表

语法：

```
{dede:pagebreak/}
```

文件：

```
\include\arc.archives.class.php
```

```
function GetPagebreakDM($totalPage, $nowPage, $aid)
```

```
function GetPagebreak($totalPage, $nowPage, $aid)
```

默认底层模板：

无

参数：

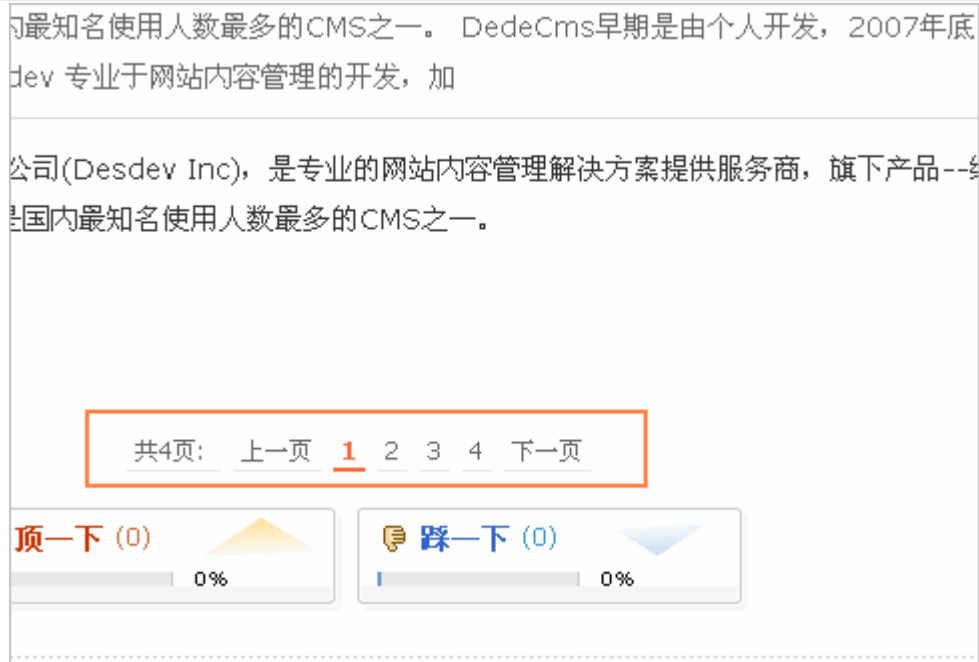
无

底层字段：

无

范例：

这个标签用于内容分页部分，我们可以查看内容页：



可以通过模板\templets\default\article_article.htm 查看到以下的内容

```
<div class="dede_pages">
<ul class="pagelist">
{dede:pagebreak/}
</ul>
</div><!-- /pages -->
```

这里就是专门用于生成内容分页的标记，当然我们可以查看生成后的源文件

```
<div class="dede_pages">
<ul class="pagelist">
<li><a> 共 4 页 : </a></li><li><a href="#"> 上 一 页 </a></li><li class="thisclass"><a href="#">1</a></li><li><a href='91_2.html'>2</a></li><li><a href='91_3.html'>3</a></li><li><a href='91_4.html'>4</a></li><li><a href='91_2.html'>下一页</a></li>
</ul>
</div><!-- /pages -->
```

我们可以通过指定 css 样式来修改这个分页的样式。

pagetitle 分页标题

名称: pagetitle

功能: 表示获取文档的分页标题

语法:

```
{dede:pagetitle style='select'/}
```

文件:

```
\include\arc.archives.class.php
```

```
function GetPageTitlesDM($styleName, $pageNo)
```

```
function GetPageTitlesST($styleName, $pageNo)
```

默认底层模板:

无

参数:

style="" 表示分页标题的展示样式

select 表示下拉框、link 表示文字直接链接

底层字段:

无

范例:

这个标签一般用于内容页面，我们这里以普通文章的内容为例，我们首先修改内容页模板 \templets\default\article_article.htm，我们看到以下内容：

```
<div class="content">
    {dede:field.body/}<br />
    分页标题: {dede:pagetitle/}
</div><!-- /content -->
```

保存这个分页标题的内容，然后我们去编辑一篇内容：

文章主栏目：

文章副栏目：

文章内容：

附加选项： ☒ 下载远程图片和资源 ☒ 提取第一个图片为缩略图

样式 格式 字体 大小

上海织梦信息科技有限公司(Desdev Inc)，是专业的网站内容管理解决方案提供商，使用人数最多的CMS之一。

#p#第一部分内容标题#e#

DedeCms早期是由个人开发，2007年底，在上海正式成立公司，Desdev 专业于网站内需求、使用习惯等理解十分透彻，自从成立公司后，不断招聘研发人员，为我们向广大用户

#n#第二部分内容标题#e#

DedeCms居于PHP+MySQL的技术开发，支持多种服务器平台，从2004年开始发布第一以简单、健壮、灵活、开源几大特点占领了国内CMS的大部份市场，目前已经有超过二十发。

#p#第三部分内容标题#e#

我们对产品追求的终极目标是：想到即可做到，做到要做最好。

编辑完成之后发布内容进行预览：

上海织梦信息科技有限公司(Desdev Inc)，是专业的网站内容管理解决方案提供管理系统(DedeCms)是国内最知名使用人数最多的CMS之一。 DedeCms早期：在上海正式成立公司，Desdev 专业于网站内容管理的开发，加

上海织梦信息科技有限公司(Desdev Inc)，是专业的网站内容管理解决方案内容管理系统(DedeCms)是国内最知名使用人数最多的CMS之一。

1、第一部分内容标题
1、第一部分内容标题
2、第二部分内容标题
3、第三部分内容标题

共4页： 上一页 1 2 3 4 下一页

顶一下 (0)
0%

踩一下 (0)
0%

一篇： [Web2.0网站：因为有文化所以做不大？](#)

 收藏

这样我们就将这个标签应用起来，当然这个标签还可以配合自定义函数灵活运用在很多地方。

prenext 获取上一篇、下一篇内容

名称: prenext

功能: 获取当前文档上一篇、下一篇内容

语法:

```
{dede:prenext/}
```

文件:

```
\include\arc.archives.class.php
```

```
function GetPreNext($gtype="")
```

默认底层模板:

无

参数:

get 获取连接类型 (5.x 以后版本), get="" 同 DedeCMS 4 显示横排的上下页连接, get='pre' 上一篇的连接

get='next' 下一篇连接

注: 此标记默认为横向排列, 如果你想用竖向排列, 可以用

```
{dede:prenext function="str_replace(' 下一篇: '<'.br!'/>下一篇: ',@me)"}表示
```

底层字段:

无

范例:

用于显示内容页的上一页、下一页的相关文档



我们可以在内容页模板页面中查看相关代码, 这里以文章内容页模板\templets\default\article_article.htm 为

例，我们查看相关代码

```
<div class="context">

<ul>

<li>{dede:prenext get='pre'/}</li>

<li>{dede:prenext get='next'/}</li>

</ul>

</div><!-- /context -->
```

当然我们也可以通过自定义的 function 来对这个标签进行处理，以配合网站模板的制作。

fieldlist 当前模型内容页字段列表

名称：fieldlist

功能：获取当前模型内容页字段列表

语法：

```
{dede:fieldlist}

<table width="98%" border="0" cellpadding="1" cellspacing="1" bgcolor="#BED1AB">

<tr>

<td height="24" bgcolor="#FAFDEE"><strong>[field:name]/</strong></td>

</tr>

<tr>

<td height="24" bgcolor="#FFFFFF">[field:value]/</td>

</tr>

</table>

<div style="font-size:2px;line-height:100%">&nbsp;</div>

{/dede:fieldlist}
```

文件：

\include\arc.archives.class.php

默认底层模板：

tag_fieldlist.htm

参数：

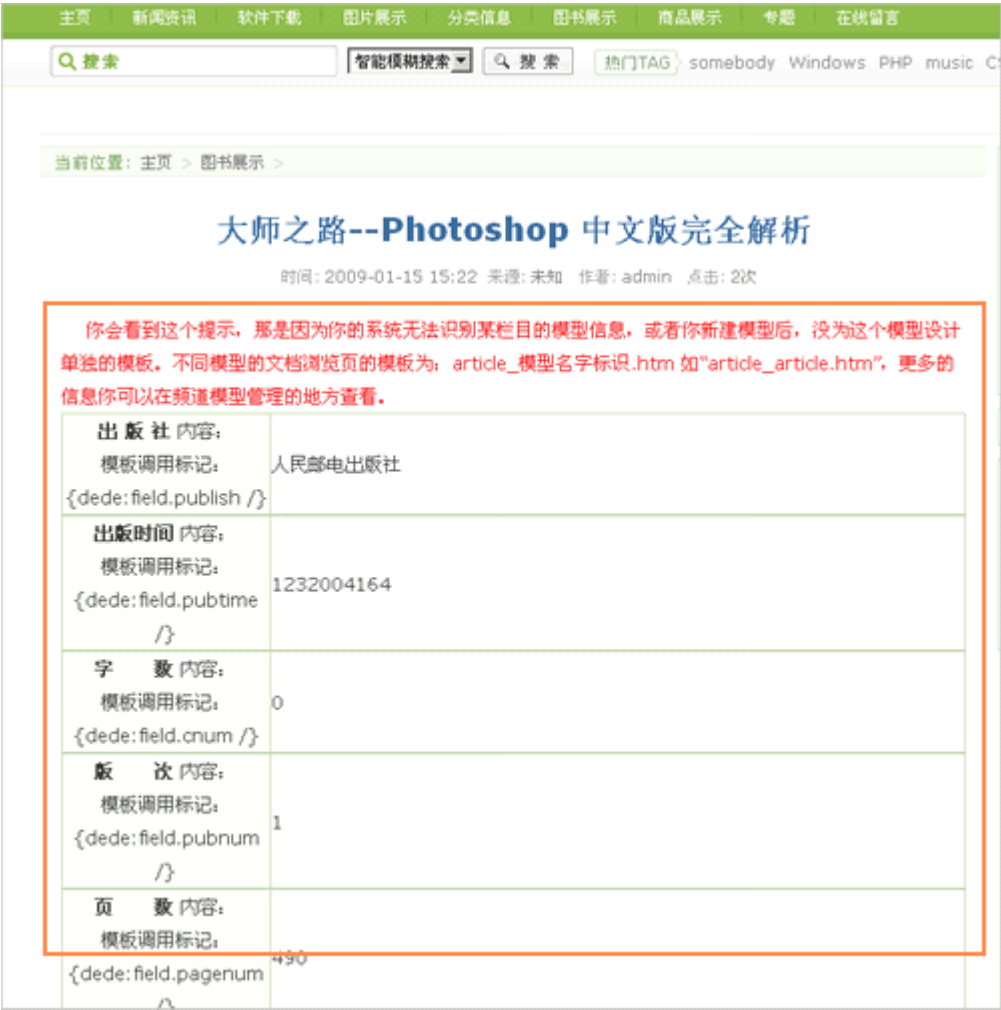
无

底层字段:

name,tagname,value

范例:

可参看\templets\default\article_default.htm 中相关字段



6.4 模块标签

bookcontentlist 连载图书最新内容调用

(>=DedeCMS 3, DedeCMS 4, DedeCMS 5)

名称: bookcontentlist

功能: 连载图书最新内容调用

语法:

```
{dede:bookcontentlist row='12' booktype='-1' orderby='lastpost' author="" keyword=""}

<table width="100%" border="0" cellspacing="2" cellpadding="2">

<tr>

<td width=40%>

[[field:cataloglink/]] [field:booklink/]</td>

<td width=40%>[field:contentlink/]</td>

<td width=20%>[field:lastpost function="GetDateMk(@me)"/]</td>

</tr>

</table>

{/dede:bookcontentlist}
```

文件：

\include>taglib\bookcontentlist.lib.php

function lib_bookcontentlist(&\$ctag, &\$refObj)

默认底层模板：

bookcontentlist.htm

参数：

row=12 调用记录条数

booktype=-1 图书类型，0 图书、1 漫画，默认全部

titelen=30 标题长度

orderby=lastpost 排序类型，当按排序类型为 commend 表示推荐图书

author="" 作者

keyword="" 关键字

底层字段：

contenttitle,contentid,contenturl,dmbookurl,bookurl,catalogurl,cataloglink,booklink,contentlink,imglink,ischeck,booktypename

booklist 连载图书调用

(>=DedeCMS 3, DedeCMS 4, DedeCMS 5)

名称：bookcontentlist

功能： 连载图书调用

语法：

```
{dede:booklist row='12' booktype='-1' orderby='lastpost' author="" keyword=""}

<a href="[field:bookurl /]">[field:bookname /]</a><br />

{/dede:booklist}
```

文件：

\include\taglib\booklist.lib.php

function lib_booklist(&\$ctag, &\$refObj, \$getcontent=0)

默认底层模板：

booklist.htm

参数：

row=12 调用记录条数

booktype=-1 图书类型，0 图书、1 漫画，默认全部

titlelen=30 标题长度

orderby=lastpost 排序类型，当按排序类型为 commend 表示推荐图书

author="" 作者

keyword="" 关键字

底层字段：

contenttitle,contentid,contenturl,dmbookurl,bookurl,catalogurl,cataloglink,booklink,contentlink,imglink,ischeck,booktypename

group 圈子调用标签

名称： group

功能： 调用系统圈子信息

语法：

```
{dede:group row='6' orderby='threads' titlelen='30'}

<li>

<span></span>

<span><a href="[field:url/]" title="[field:groupname/]" target="_blank">[field:groupname/]</a></span>
```

```
</li>

{/dede:group}
```

文件：

\include\taglib\group.lib.php

function lib_group(&\$ctag,&\$refObj)

默认底层模板：

groups.htm

参数：

row=6 调用条数

orderby=threads 排列顺序（默认是主题数）

tidelen=30 圈子名称最大长度

底层字段：

groupimg,groupid,groupname,url,icon

groupthread 圈子主题调用标签

名称： group

功能： 调用系统圈子模块主题

语法：

```
{dede:groupthread gid='0' orderby='dateline' orderway='desc' row='12' tidelen='30'}

<li>

  <a href="[field:groupurl/]" target="_blank">[[field:groupname function="cn_substr(@me,10)"/]]</a>

  <a href="[field:url/]" title="[field:subject/]" target="_blank">[field:subject/]</a>([field:lastpost function="GetDateMk('@me')"/])

</li>

{/dede:groupthread}
```

文件：

\include\taglib\groupthread.lib.php

function lib_groupthread(&\$ctag,&\$refObj)

默认底层模板：

groupthreads.htm

参数:

gid=0 圈子 id, 默认为全部

orderby='dateline' 排序, 默认为时间

orderway='desc' 排序方向, 默认是最新主题在前

row=12 记录数

tidelen=30 主题标题(subject)最大长度

底层字段:

subject,url,groupurl,gid,tid,lastpost,groupname

6.5 其他标签

infoguide 分类信息的地区与小分类搜索

(>DedeCMS 4, DedeCMS 5)

名称: infoguide

功能: 获取分类信息的地区和小分类搜索的表单, 本标签需要在分类信息模型下使用

语法:

```
{dede:infoguide}

    <form name='infoguide' method='get' action='/plus/list.php'>

        [field:nativeplace /]

        [field:infotype /]

        <input type='hidden' name='tid' value='[field:typeid /]' />

        <input type='hidden' name='channelid' value='-8' />

        <span class='infosearchtxt'>关键字: </span><span><input type='text' name='key word' value="" class="ipt-txt" /></span>

        <input type='submit' value='搜索信息' class="btn-2" style='cursor:pointer' />

        &nbsp;

        <input type='button' value='发布信息' class="btn-2"

        onclick="location='/member/archives_sg_add.php?channelid=-8';" style='cursor:pointer' />

    </form>

{/dede:infoguide}
```

文件：

\include\taglib\infoguide.lib.php

function lib_infoguide(&\$ctag,&\$refObj)

默认底层模板：

info_guide.htm

参数：


无

底层字段：

nativeplace,infotype,typeid

范例：

可以查看分类信息页面模板 list_info.htm 文件。



The image shows a web form titled "信息搜索" (Information Search). It contains two rows of dropdown menus for "地区" (Region) and "类型" (Type), each with a "请选择.." (Please select..) prompt. Below these is a text input field for "关键字" (Keywords). At the bottom are two buttons: "搜索信息" (Search Information) and "发布信息" (Post Information).

infolink 分类信息地区与类型快捷链接

(>DedeCMS 4, DedeCMS 5)

名称： infolink

功能： 获取分类信息地区与类型的快捷链接，仅在分类信息模型下能够使用

语法：

```
{dede:infolink}
<table id='dedeinfolink'>
<tr>
    <td colspan='3' class='iftitle'>
        信息附加条件：
```



```

        </td>
</tr>
<tr>
    <td width=50>&nbsp;地 区: </td>
    <td align='center' nowrap='yes'>
        <div>[field:linkallplace/]</div>
    </td>
    <td>
        [field:nativeplace /]&nbsp;
    </td>
</tr>
<tr><td colspan='3' class='spline'>&nbsp;</td></tr>
<tr>
    <td>&nbsp;类 型: </td>
    <td align='center' nowrap='yes'>
        <div>[field:linkalltype/]</div>
    </td>
    <td>
        [field:infotype /]&nbsp;
    </td>
</tr>
<tr><td colspan='3' class='spline'>&nbsp;</td></tr>
</table>
{/dede:infolink}

```

文件:

\include\taglib\infolink.lib.php

function lib_infolink(&\$ctag,&\$refObj)

默认底层模板:

info_link.htm

参数:

无

底层字段:

nativeplace,infotype,typeid,channelid,linkallplace,linkalltype

范例:

信息附加条件:

北京市 天津市 河北省 山西省 内蒙古自治区 辽宁省 吉林省 黑龙江省 上海市 江苏省 浙江省 安徽省 福建省 江西省 山东省 河南省 湖北省 湖南省 广东省 广西壮族自治区 海南省 重庆市 四川省 贵州省 云南省 西藏自治区 陕西省 甘肃省 青海省 宁夏回族自治区 新疆维吾尔自治区 台湾省 香港特别行政区 澳门特别行政区

地区:

东城区 西城区 崇文区 宣武区 朝阳区 丰台区 石景山区 海淀区 门头沟区 房山区 通州区 顺义区 昌平区 大兴区 怀柔区 平谷区 密云县 延庆县

类型: 商品 租房 交友 招聘 求职 票务 服务 培训

第七章 模板扩展

7.1 标签扩展

7.1.1 定义标签

在解析式模板引擎中我们需要定义全局标签，在上面的文档说明中已经知道在系统根目录下 /include/taglib 中进行扩展，标签的文件名应该是：标签.lib.php，例如 arclist 所对应的标签名称则为：arclist.lib.php。

我们创建一个测试的标签 {dede:test name='tianya'/}，那就需要创建一个名为 test.lib.php 的文件。

7.1.2 标签文件内容结构

标签内容的文件结构如下，为一个函数

```
<?php
if(!defined('DEDEINC')) exit("Request Error!");

function lib_test(&$ctag,&$refObj)
{
    global $dsq,$senvs;
```

```
//属性处理

$attlist="row|12,title|24";

FillAttsDefault($ctag->CAttribute->Items,$attlist);

extract($ctag->CAttribute->Items, EXTR_SKIP);

$revalue = "";

//你需编写的代码，不能用 echo 之类语法，把最终返回值传给$revalue

//-----

$revalue = 'Hello Word!';

//-----

return $revalue;

}

?>
```

这里 `function lib_test(&$ctag,&$refObj)` 就是我们标签的扩展函数，所有标签生成的内容都是由这个函数解析后获得的。

7.1.3 属性处理

函数中 `$attlist="row|12,title|24"` 为系统默认属性的参数，例如我们上面 `test` 标签需要一个默认的属性 `name`，并且 `name` 为空，则需要更改为：

```
$attlist="name|"
```

这里符号 “,” 用来隔开各个属性，符号 “|” 用来分隔变量的名称和值，这里 “`name|`” 则代表默认参数 `name` 为空。

我们在使用参数的时候，直接使用 `$name` 就可以在函数中进行调用了，例如：

```
function lib_test(&$ctag,&$refObj)

{

    global $dsq,$envs;
```

```
//属性处理

$attlist="name|";

FillAttsDefault($ctag->CAttribute->Items,$attlist);

extract($ctag->CAttribute->Items, EXTR_SKIP);

$revalue = '我的名字:'.$name;

return $revalue;

}

?>
```

我们在系统后台标签测试器中进行测试:

进行状态:

模板代码:

```
{dede:test name='tianya' /}
```

结果:

我的名字:tianya

7.1.4 底层模板处理

一般的标签会涉及到底层模板, 例如: arclist、channel 等, 我们来扩展下我们上面标签的功能:

```
{dede:test name='tianya' /}

我的名字叫:[field:name/]

{dede:test}
```

这里“我的名字叫:[field:name/]”则为我们要处理的底层模板, [field:name/]为我们标签属性中的 name。

这里我们需要了解 lib_test(&\$ctag,&\$refObj)函数中\$ctag 参数, 这个参数就是我们解析式模板引擎获取的当前标签内容, 其中这个对象下有一个 GetInnerText()方法能够获取当前标签的底层模板。

```
<?php

if(!defined('DEDEINC')) exit("Request Error!");

function lib_test(&$ctag,&$refObj)

{

    global $dsq,$senvs;
```

```
//属性处理

$attlist="name|";

FillAttsDefault($ctag->CAttribute->Items,$attlist);

extract($ctag->CAttribute->Items, EXTR_SKIP);

$innertext= $ctag->GetInnerText(); // 获取底层模板

$revalue = str_replace("[field:name/]", $name, $innertext); // 对标签进行解析

return $revalue;

}

?>
```

这样我们就完成了一个简单的对底层模板解析的处理。

进行状态:

模板代码:

```
{dede:test name='tianya'}
我的名字叫:[field:name/]
{/dede:test}
<BR/>
{dede:test name='tianya'}
[field:name/]是我的名字
{/dede:test}
```

结果:

我的名字叫:tianya
tianya是我的名字

标签其他一些开发技巧可参考/include/taglib 其他标签文件。

7.2 常用函数

cn_substr

(>=DedeCMS 2, DedeCMS 3, DedeCMS 4, DedeCMS 5)

函数名: cn_substr

功能: 截取中文字符串

参数:

\$str='' 需要截取的字符串

\$slen='' 需要截取的长度，单字节截取模式，一个中文占用 2 个字节

\$startdd=0 截取开始位置标记处，默认为 0 从头开始

返回值：

返回截取后的字符串，错误返回空字符串

范例：

```
{field:title function='cn_substr(@me,10)'/}
```

表示将会截取 title 中 5 个中文字符内容。

Html2Text

(>=DedeCMS 2, DedeCMS 3, DedeCMS 4, DedeCMS 5)

函数名：Html2Text

功能：HTML 代码转换为 Text 文本

参数：

\$str='' 需要转换的代码

\$r=0 如果\$r=0 直接返回内容,否则需要使用反斜线引用字符串

返回值：

转换后的 Text 文本

范例：

```
{field:description function='Html2Text(@me)'/}
```

一般页面 meta 中描述需要保证是 text 文本，所以可以采用这一函数来进行格式化。

Text2Html

(>=DedeCMS 2, DedeCMS 3, DedeCMS 4, DedeCMS 5)

函数名：Text2Html

功能：将 Text 转换为 HTML 代码格式，也可以使用 PHP 原生态函数 htmlspecialchars_decode 来处理

参数：

\$txt='' 需要格式化的 Text 代码

返回值：

转换后的 Html 代码

范例：

```
{field:body function=Text2Html(@me)'/}
```

MyDate

(>=DedeCMS 2, DedeCMS 3, DedeCMS 4, DedeCMS 5)

函数名：MyDate

功能：将 Unix 时间戳转换为格式化的日期

参数：

\$format='Y-m-d H:i:s' 日期转换后的格式

format 字符	说明	返回值例子
日	---	---
d	月份中的第几天，有前导零的 2 位数字	01 到 31
D	星期中的第几天，文本表示，3 个字母	Mon 到 Sun
j	月份中的第几天，没有前导零	1 到 31
l(“L”的小写字母)	星期几，完整的文本格式	Sunday 到 Saturday
N	ISO-8601 格式数字表示的星期中的第几天 (PHP 5.1.0 新加)	1 (表示星期一) 到 7 (表示星期天)
S	每月天数后面的英文后缀，2 个字符	st, nd, rd 或者 th。可以和 j 一起用
w	星期中的第几天，数字表示	0 (表示星期天) 到 6 (表示星期六)
z	年份中的第几天	0 到 366
星期	---	---
W	ISO-8601 格式年份中的第几周，每周从星期一开始 (PHP 4.1.0 新加的)	例如：42 (当年的第 42 周)
月	---	---
F	月份，完整的文本格式，例如 January 或者 March	January 到 December
m	数字表示的月份，有前导零	01 到 12
M	三个字母缩写表示的月份	Jan 到 Dec
n	数字表示的月份，没有前导零	1 到 12
t	给定月份所应有的天数	28 到 31
年	---	---
L	是否为闰年	如果是闰年为 1，否则为 0
o	ISO-8601 格式年份数字。这和 Y 的值相同，只除了如果 ISO 的星期数 (W) 属于前一年	Examples: 1999 or 2003

	或下一年，则用那一年。(PHP 5.1.0 新加)	
Y	4 位数字完整表示的年份	例如: 1999 或 2003
y	2 位数字表示的年份	例如: 99 或 03
时间	---	---
a	小写的上午和下午值	am 或 pm
A	大写的上午和下午值	AM 或 PM
B	Swatch Internet 标准时	000 到 999
g	小时, 12 小时格式, 没有前导零	1 到 12
G	小时, 24 小时格式, 没有前导零	0 到 23
h	小时, 12 小时格式, 有前导零	01 到 12
H	小时, 24 小时格式, 有前导零	00 到 23
i	有前导零的分钟数	00 到 59>
s	秒数, 有前导零	00 到 59>
时区	---	---
e	时区标识 (PHP 5.1.0 新加)	例 如 : UTC , GMT , Atlantic/Azores
I	是否为夏令时	如果是夏令时为 1, 否则为 0
O	与格林威治时间相差的小时数	例如: +0200
P	与格林威治时间 (GMT) 的差别, 小时和分钟之间有冒号分隔 (PHP 5.1.3 新加)	例如: +02:00
T	本机所在的时区	例如: EST, MDT (【译者注】在 Windows 下为完整文本格式, 例如 “Eastern Standard Time”, 中文版会显示“中国标准时间”)。
Z	时差偏移量的秒数。UTC 西边的时区偏移量总是负的, UTC 东边的时区偏移量总是正的。	-43200 到 43200
完整的日期 / 时间	---	---
c	ISO 8601 格式的日期 (PHP 5 新加)	2004-02-12T15:19:21+00:00
r	RFC 822 格式的日期	例 如 : Thu, 21 Dec 2000 16:01:07 +0200
U	从 Unix 纪元 (January 1 1970 00:00:00 GMT) 开始至今的秒数	参见 time()

\$timest=0 时间基准

返回值:

转换后的日期格式

范例:

```
{field:pubdate function=MyDate('Y-m-d',@me)'/}'
```


GetDateTimeMk

(>=DedeCMS 2, DedeCMS 3, DedeCMS 4, DedeCMS 5)

函数名: GetDateTimeMk

功能: 返回格式化(Y-m-d H:i:s)的时间, 类似 MyDate 函数, 但返回格式固定

参数:

\$mktime='' 需要转换的日期

返回值:

格式化以后的时间

范例:

```
{field:senddate function=GetDateTimeMk(@me)'/}
```

FloorTime

(>=DedeCMS 5.5)

函数名: FloorTime

功能: 将时间转换为距离现在的精确时间

参数:

\$seconds='' 日期时间戳

返回值:

格式化以后的返回时间

范例:

```
{field:senddate function=FloorTime(@me)'/}
```

HtmlReplace

(>=DedeCMS 5.5)

函数名: HtmlReplace

功能: 去除 html 中不规则内容字符

参数:

\$str='' 需要处理的字符串

\$rptype=0 返回类型

\$rptype = 0 表示仅替换 html 标记

\$rptype = 1 表示替换 html 标记同时去除连续空白字符

\$rptype = 2 表示替换 html 标记同时去除所有空白字符

\$rptype = -1 表示仅替换 html 危险的标记

返回值：

格式化后的字符串

范例：

```
{field:senddate function=FloorTime(@me)'/}
```

RemoveXSS

(>=DedeCMS 5.5)

函数名： RemoveXSS

功能： 去除 XSS 恶意代码

参数：

\$val=" 需要格式化的内容

返回值：

格式化后的字符串

范例：

```
{field:keywords function=RemoveXSS(@me)'/}
```