

UBB语法

UBB的解析

一个有效的UBB节点结构分为三个部分,示例如下:

```
[url href=http://www.guokr.com"]果壳网[/url]
```

- 起始标签: [url href=http://www.guokr.com]
- 内容块: 果壳网
- 结束标签: [/url]

其他关键部分:

- 标签名: /[a-zA-Z0-9]+/, 有大小写无关的英文字母与数字组成
- 属性: " href=http://www.guokr.com", 起始标签中去掉 "["、"]"和标签名[

1. 标签的特性

标签的特性有三个:

- canContains表示此标签可以包含的标签:字符串,例如:'bold,italic','*'
- canWrap是否可换行:bool值,表示标签内容块是否可以换行
- isBlock是否是块级元素:bool值,表示标签转换成HTML元素后该HTML元素是否

稍后再讲这三个特性的作用。

标签名	属性	canContains	canWrap	isBlock	样例	HTML
bold	无	'bold,italic,color,url'	false	false	[bold]加粗[/bold]	加粗
italic	无	'bold,italic,color,url'	false	false	[italic]斜体[/italic]	<i>加粗</i>
color	=#000	'bold,italic,color,url'	false	false	[color=blue]蓝色[/color]	蓝色
url	href=http://www.guokr.com/	'bold,italic,color,url,image'	false	false	[url href=http://www.guokr.com]果壳网[/url]	果壳网
image	无	"	false	false	[image]http://www.guokr.com/skins/3-logo.png[/image]	

标签名	属性	canContains	canWrap	isBlock	样例	HTML
video	无	"	false	false	[video]http://player.youku.com/player.php/sid/XNDMwNDEzMjc2/v.swf[video]	前端预览 : 后端生成 : <embed src="http://player.youku.com/player.php/sid/XNDMwNDEzMjc2/v.swf" type="application/x-shockwave-flash" allowscriptaccess="sameDomain" allowfullscreen="true" wmode="transparent" width="480" height="400">
flash	无	"	false	false	[flash]http://player.youku.com/player.php/sid/XNDMwNDEzMjc2/v.swf[flash]	前端预览 : 后端生成 : <embed src="http://player.youku.com/player.php/sid/XNDMwNDEzMjc2/v.swf" type="application/

标签名	属性	canContains	canWrap	isBlock	样例	HTML
						x-shockwave-flash" allowscriptaccess="sameDomain" allowfullscreen="true" wmode="transparent" width="480" height="400">
blockquote	无	'*	true	true	[blockquote]引用别人的评论 [/blockquote]	<blockquote>引用别人的评论 </blockquote>
ul	无	'*	true	true	[ul] 第一条 第二条 [/ul]	 第一条 第二条
ol	无	'*	true	true	[ol] 第一条 第二条 [/ol]	 第一条 第二条
ref	无	"	false	true	[ref]引用别人的评论[/ref] p	前端预览： <div class="gui-ubb-ref">http://www.guokr.com/article/176586/</div> 后端生成： <div class="gui-ubbv-ref-bd"> <div class="gui-ubbv-ref-bd">

1. 换行处理

换行使用
标签替换;

当canWrap为true时，标签的内容块可以换行。并且此标签转换成HTML之后一定是block元素。当canWrap为false时，标签的内容块不可以换行，并会进行容错处理（稍后再解释）。注意canWrap为false并不意味着该标签转换后的HTML元素一定不是block元素，例如ref标签的canWrap为false，但是转换后的HTML元素是DIV（block元素）。

当isBlock为true时，紧随标签之后的换行不会被转换成
。以此避免如下情况：

```

ubb为：
[ref]http://www.guokr.com/post/263028/[ref]
12

```

结果为：

引用 [馒头老妖](#) 的帖子

【冷犯罪】蝇头小利惹来深度邪恶：道德的防线啊，你为...

评论 16 | 推荐 2

近日,《赫芬顿邮报》、《枪口还冒烟哩》等媒体报导了一起颇有些无聊又可恨的案件:一名米国小伙,在购买了药品并使用后再重新装好包装退货,被店员识破并被捕。此案于上月月底(2012-06-28... [查看全文](#)

12

2. 转义

反斜杠“\”为转义符号,即“[”或“]”会被转义显示成“\[”或“\]”。

例如: `[bold]aaa[/bold]` 输出显示结果为 `[bold]aaa`

3. 容错处理

原则是:尽量保留开始标签,自动补全结束标签。结束标签不匹配则丢弃。

a. `canWrap`为`false`时,标签内容部分包含换行

解决方案,拆分并逐行自动补全。例如:

```
[bold]aaa
bbbb[/bold]
```

转换成:

```
[bold]aaa[/bold]
[bold]bbbb[/bold]
```

b. 标签不匹配情况

情况一: `[bold]aaaa`

结果: `[bold]aaaa[/bold]`

自动补全结束标签

情况二: `aaaa[/bold]`

结果: `aaaa`

抛弃多余的结束标签

情况三: `[bold][italic]aaaa[/bold]bbb[/italic]`

结果:

步骤一: `[bold][italic]aaaa[/italic]`

找到未匹配标签`bold`,自动补全`italic`结束标签

步骤二: `[bold][italic]aaaa[/italic][bold][italic]bbb`

`[/bold]`找到标签匹配,并将前一个自动比和的`[italic]`标签同时放到`[/bold]`.

步骤三: `[bold][italic]aaaa[/italic][bold][italic]bbb[/italic]`

`[/italic]`找到标签匹配

情况四: `[bold]aaaa[/bold][italic]ccc[/italic]bbb[/bold]`

结果: `[bold]aaaa[/bold][italic]ccc[/italic]bbb`

抛弃无法匹配的结束标签

情况五：`[bold][italic]aaaa[/bold]bbb`

结果：

步骤一：`[bold][italic]aaaa[/italic]`

找到未匹配标签**bold**，自动补全**italic**结束标签

步骤二：`[bold][italic]aaaa[/italic][[/bold][italic]bbb`

`[/bold]`找到标签匹配，并将前一个自动补全的**italic**标签同时放到`[/bold]`。

步骤三：`[bold][italic]aaaa[/italic][[/bold][italic]bbb[/italic]`

最后自动闭合剩余的

c. 与标签包含规则相悖

情况一：前一个标签不能包含后一个标签，但后一个标签可以包含前一个标签

解决方案，在自动补全前几个标签。例如：

```
[bold][italic][ul]
aaa
bbbb
[/ul]ccc[/italic][[/bold]
```

转换成：

```
[bold][[/bold][ul][bold][italic][[/italic][[/bold]
[bold][italic]aaa[/italic][[/bold]
[bold][italic]bbbb[/italic][[/bold]
[bold][italic][[/italic][[/bold][[/ul][bold][italic]ccc[/italic]
```

步骤一：`ul`开始标签优先级大于**bold**和**italic**开始标签，于是自动补全**bold**和**italic**

步骤二：**bold**和**italic**不支持内部换行，补全

步骤三：**bold**结束标签，找到已匹配标签**bold**，自动将前一个**bold**开始标签同时做为

情况二：标签不能互相包含

解决方案，在自动补全前几个标签。例如：

```
[bold][ref]aaa[/ref]ccc[/bold]
```

转换成：

```
[bold][[/bold][ref]aaa[/ref][bold]ccc[/bold]
```

步骤一：`ul`开始标签优先级大于**bold**开始标签，于是自动补全**bold**结束标签；

步骤二：**bold**结束标签，找到已匹配标签**bold**，自动将前一个**bold**开始标签同时做为