**[C#开发微信门户及应用(8)-微信门户应用管理系统功能介绍](http://www.cnblogs.com/wuhuacong/p/3733189.html)**

最近对微信接口进行深入的研究，通过把底层接口一步步进行封装后，逐步升级到自动化配置、自动化应答，以及后台处理界面的优化和完善上，力求搭建一个较为完善、适用的微信门户应用管理系统。

微信门户应用管理系统，采用基于MVC+EasyUI的路线，由于多数域名服务器上都只能支持.NET4.0，所以以MVC3，C#4.0作为开发基础，基本上能够部署在任何.NET服务器上。

在微信门户系统里面，实现下面这些功能操作：

1）实现菜单的动态配置及更新到服务器上；

2）动态定义事件和响应消息，实现对不同行业，不同需求的菜单动作响应；

3）动态的应答指令配置处理，实现整套应答链的消息处理；

4）获取订阅用户和用户分组信息，并可以实现用户分组信息的维护等操作；

5）管理并更新多媒体文件、图文消息等内容，方便为客户推送消息做准备。

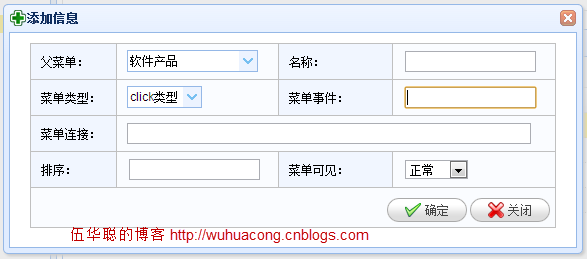
6）使用向选定订阅用户或者分组进行消息的群发功能。

**1、微信菜单管理**

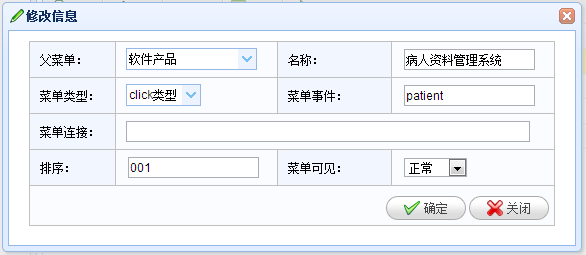
在系统中管理菜单，并通过把菜单提交到服务器上，实现菜单的动态配置和生成，能够为我们系统适应各种的需要，实现灵活的处理。



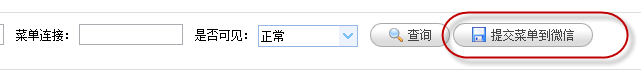
微信菜单的添加界面如下所示。



微信菜单的修改界面如下所示



微信菜单定义是存储在数据库里面，如果需要提交到微信服务器上并生效，则需要调用微信API接口进行处理，我在页面的Controller控制器里增加一个提交到服务器的处理方法。



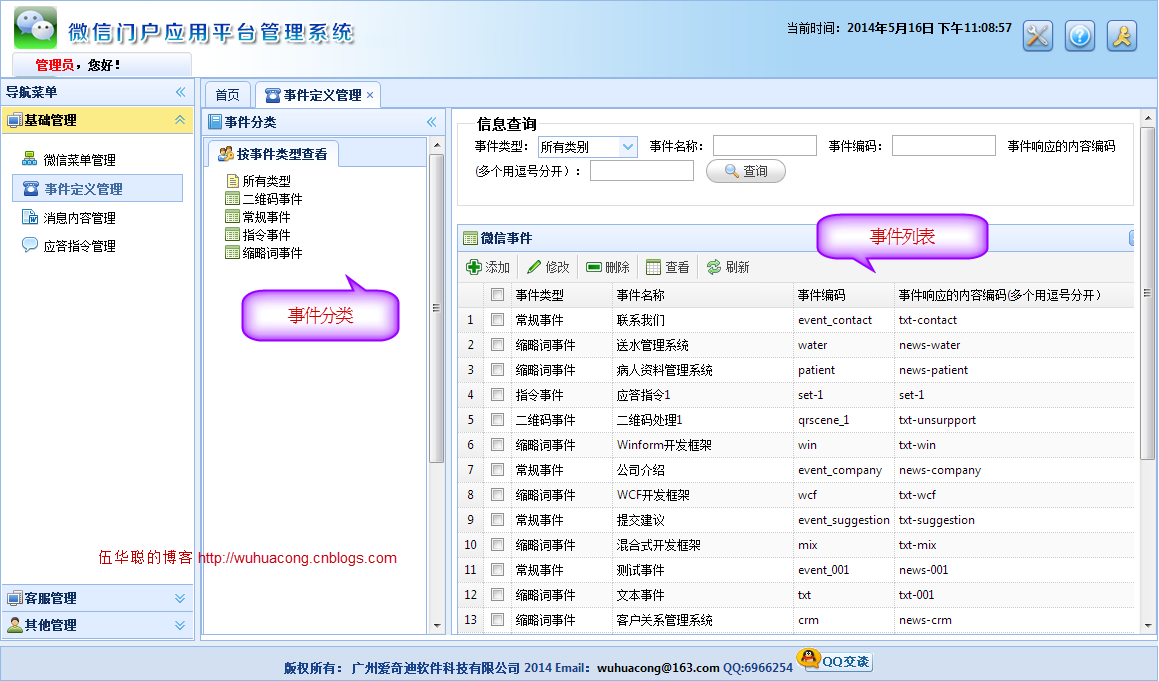
在微信服务账号的门户上，菜单的表现效果如下所示。



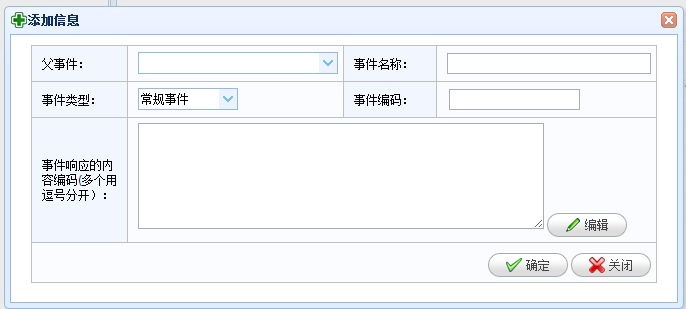
**2、菜单事件的处理**

对于动态生成的菜单，大多数情况下是用作Click的方式，也就是需要定义每个菜单的事件响应操作，我们使用微信的话，可以了解到，微信的处理事件，一般可以响应用户文本消息、图片消息、图文消息等内容，常规下，一般使用文本消息或者图文消息居多。

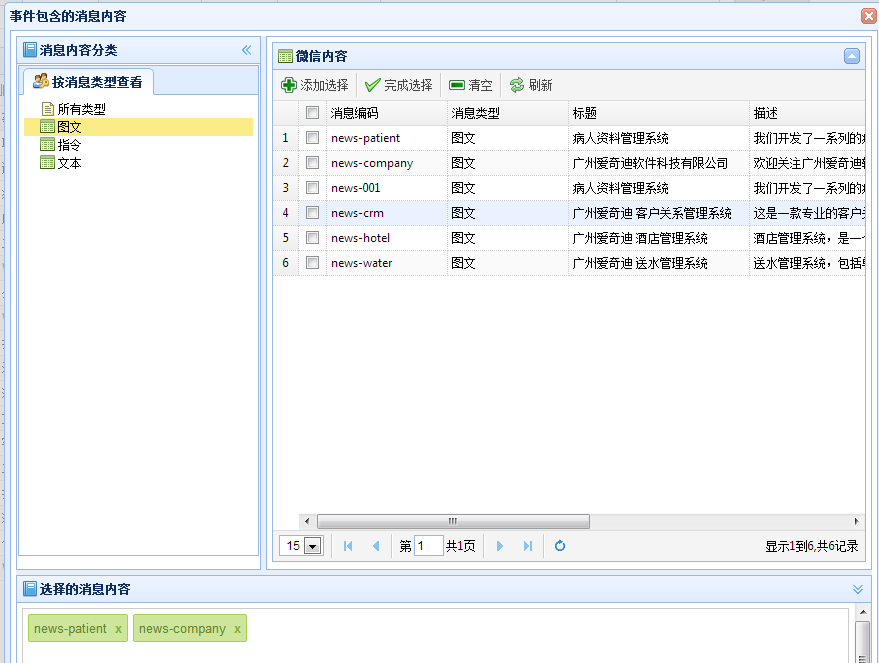
为了进一步实现响应内容的重用，我们把菜单的事件定义和内容定义进行分开管理，事件定义可以使用多个文本消息，也可以使用多个图文消息进行组合，这样可以实现更加灵活的使用环境。



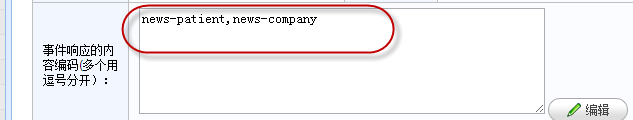
添加事件定义如下所示



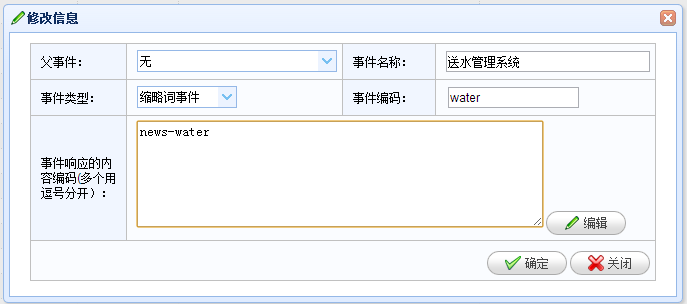
事件的响应内容编码，可以选择输入或者从“编辑”按钮中选择，当选择“编辑”按钮进行选择的时候，系统弹出一个对话框供用户对事件的响应内容编码选择。



完成选择后，回到原来的新增界面，将会看到返回的记录就是我们选择的记录。

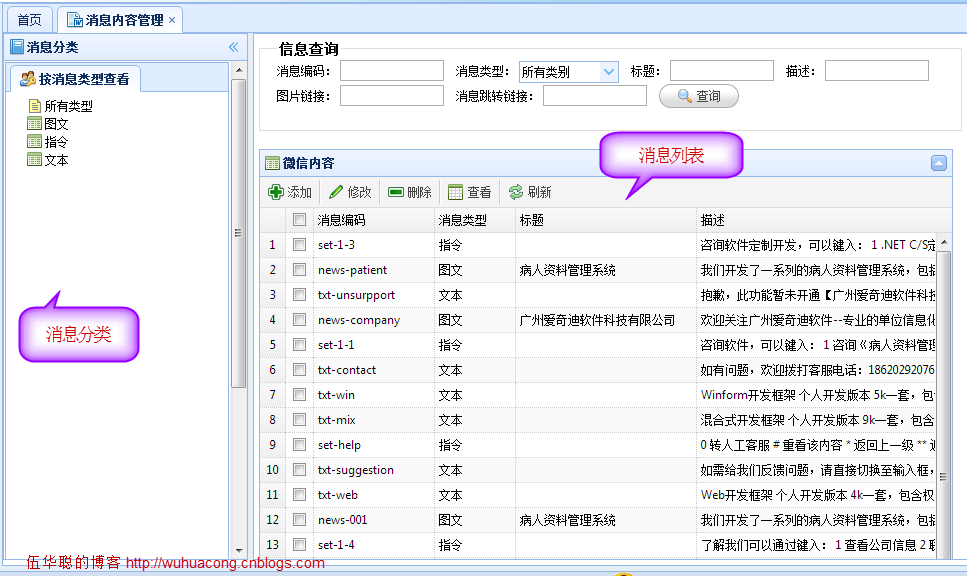


微信事件的编辑界面如下所示，类似新增界面的内容。



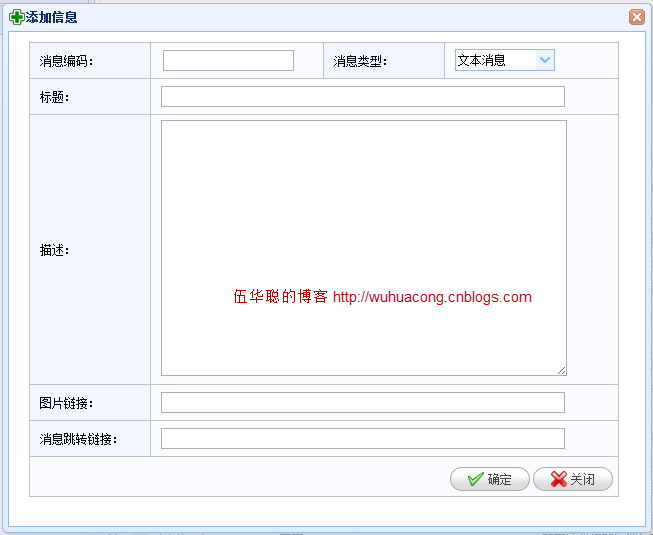
**3、微信消息内容管理**

 上面说到，菜单的事件通过关联事件编码进行处理，而事件本身可以组合多个消息内容，因此消息内容是响应客户操作的最小单元，它们可以是一条文本消息、图文消息，也可以是多条消息的组合（同类型的话）。

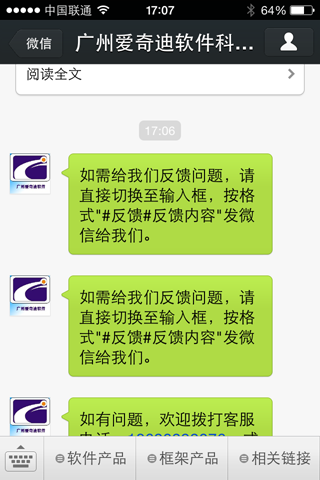


为了方便管理，我把消息分为了图文、指令、文本类型，如果需要，还可以根据需要把它细化为其他类型的消息。

消息内容的添加界面如下所示。



文本消息的手机上界面效果如下所示。



这里不管是文本消息还是图文消息，我们统一以图文消息的定义来定义消息，如果是文本消息，我们只需要获取描述内容作为消息的主体即可。

图文消息的编辑界面如下所示，主要就是填写完整的内容和图片，以及页面详细的链接即可。

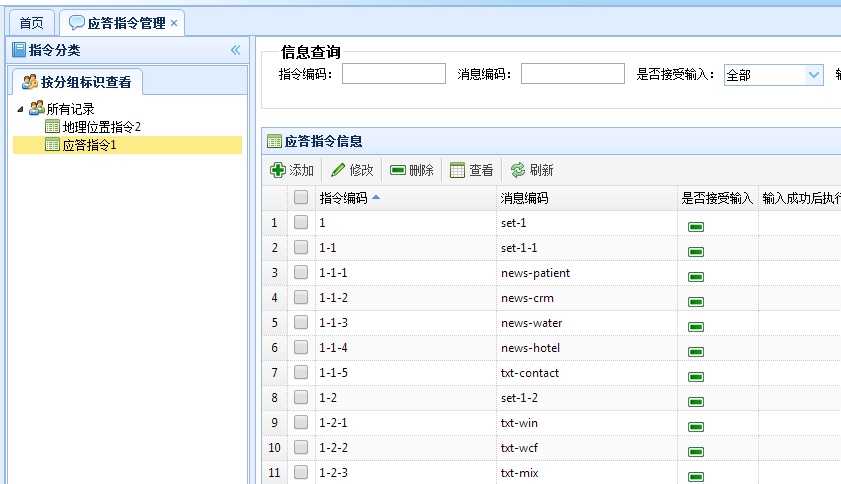


上面的这个客户关系管理系统的消息，在手机上显示的界面效果如下所示，单击链接，可以切换到消息跳转链接地址的。



**4、应答指令的维护**

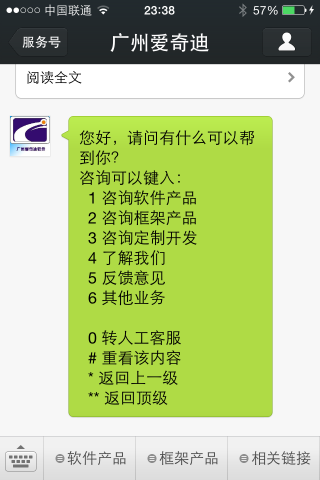
应答指令的维护，有点类似于事件的管理，主要就是定义一些用到的指令，方便构建应答系统的响应链，从而实现一步步的操作指令。



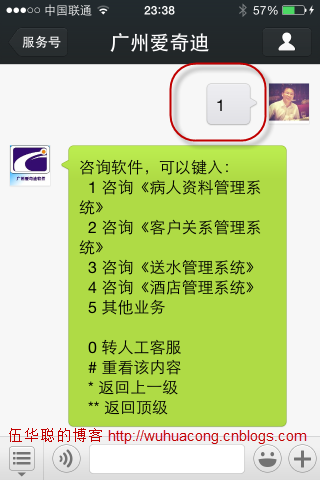
在后台设置好应答指令后，系统就能根据应答指令链进行处理了。首先我们需要提供一个进入应答链的提示界面，如下所示。



但我们在菜单选择应答系统后，系统返回一个文本提示界面，如下所示。



这个界面里面提示了一些按键，包括几个固定的按键和一些业务按键，输入简单的1~6可以对选择进行响应。

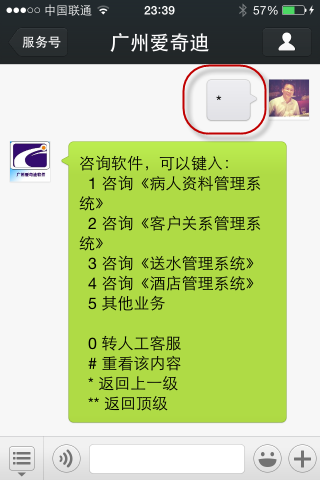


我们看到上面的界面，输入指令1后，系统进入下一层的应答指令，然后又列出几个可供输入的按键和内容提示。

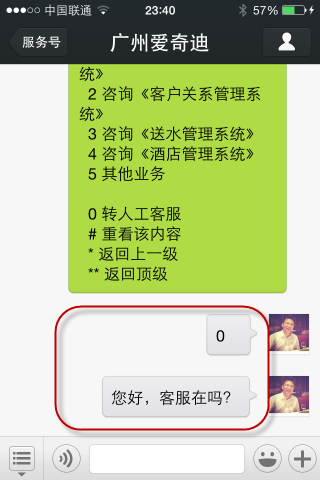
当我们继续输入业务按键1后，响应的是一个图文消息，也是关于按键的详细说明。



这个时候，我们也还可以输入\*号按键，返回上一级菜单的。



输入0则转入了客服对话模式，后续您发的任何消息，将会转发到多客服系统里面了。

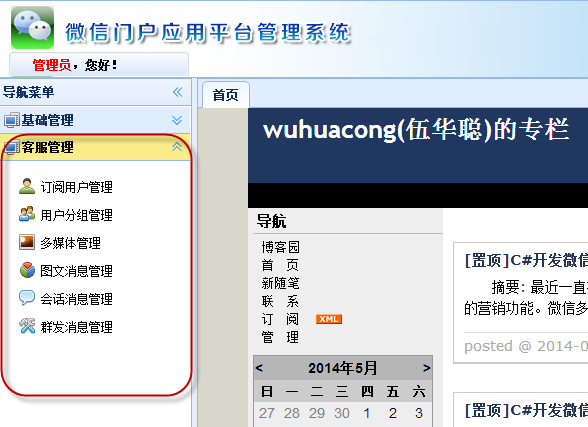


当用户发送消息后，客服助手就能及时收到消息并处理和客户的应答了。

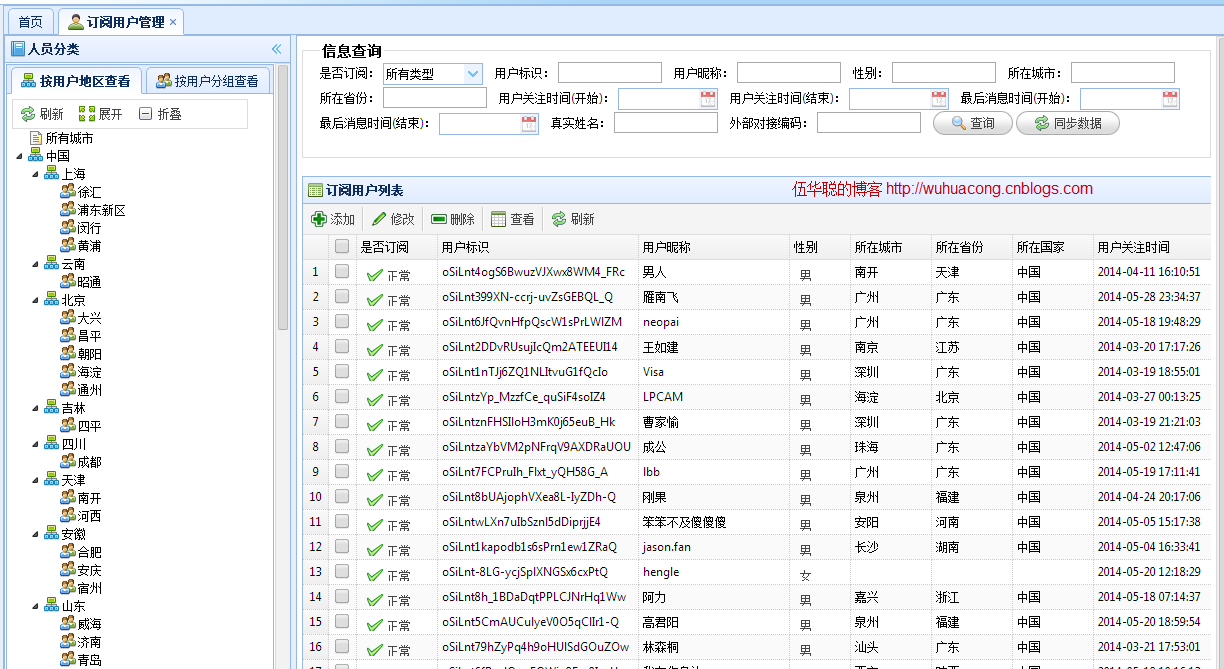


**5、订阅用户管理**

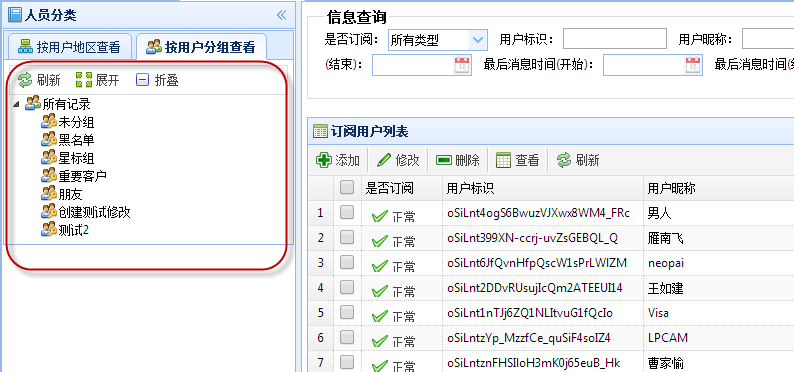
为了更有效管理订阅用户以及分组信息，我们可以从微信服务器上获取相关的信息，供我们了解关注的用户信息，也可以为后续的群发消息做准备。



订阅用户的管理如下所示，默认可以通过用户的地区进行查看，地区根据：国家-省份-城市这样的级别进行展开。单击同步数据，可以把服务器上的用户数据下载到本地进行更新或者写入。



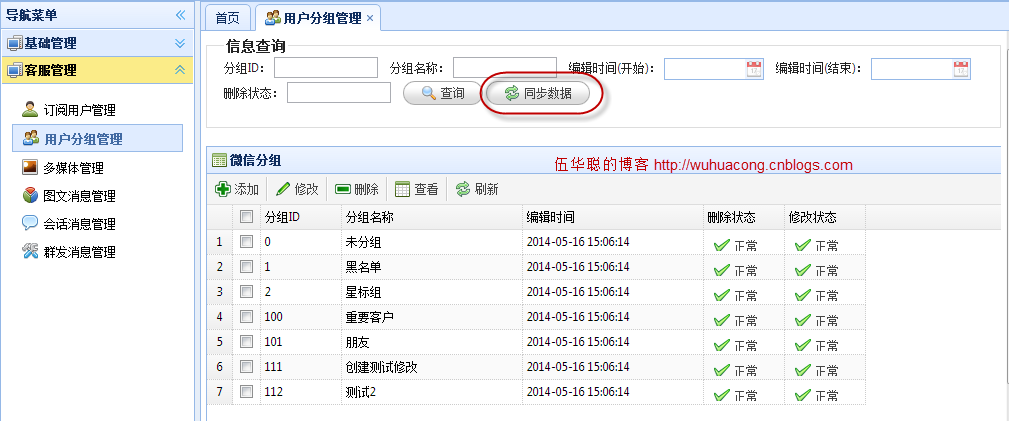
订阅用户，还可以根据分组进行查看



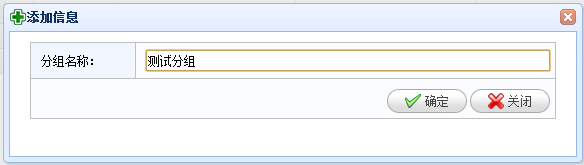
双击可以查看订阅用户信息，查看订阅用户的详细信息界面如下所示。



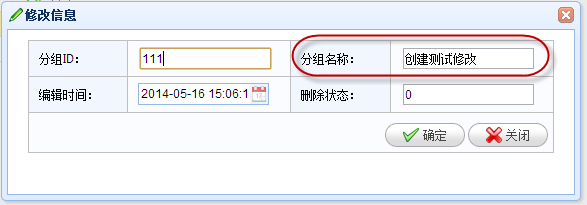
**7、用户分组管理**



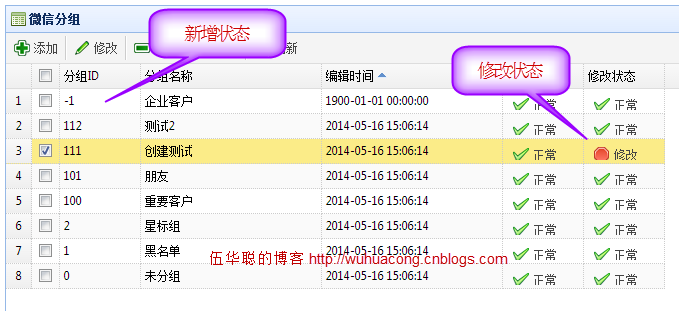
创建分组的界面如下所示。



编辑分组信息界面如下所示。



当对分组进行编辑保存后，系统会记住那些修改过的，同步的时候，把本地新增的内容，在服务器上创建分组；把修改的的分组名称，在服务器上进行修改，然后进行同步列表处理。

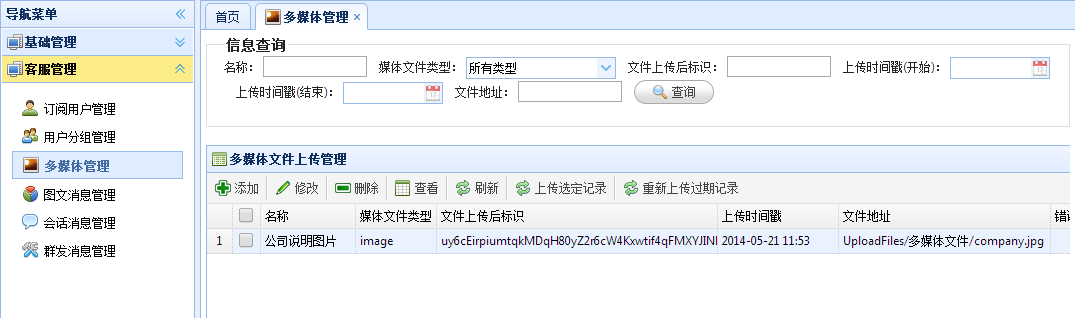


**8、多媒体管理**

多媒体管理是指把本地文件上传到微信服务器上进行保存，方便信息的发送等操作。微信要求，某些信息，必须是先上传到服务器上，然后才能使用它的媒体ID进行发送的。

文件成功上传到服务器后，在列表里面的“文件上传标识，就是一串BASE64的编码数据，同时有一个上传的时间戳（因为微信服务器只保留了3天的媒体数据，超过期限的数据会被自动删除。

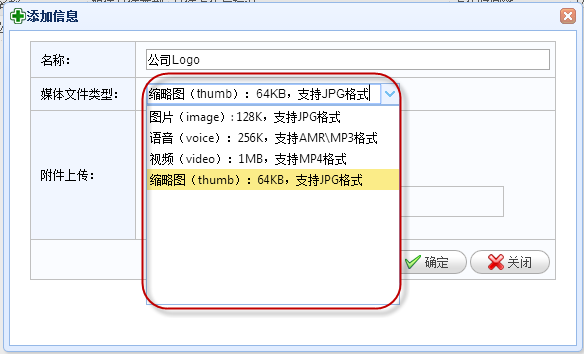
同时，在列表的上面，有两个重要的功能：上传选定的记录，重新上传过期的记录。方便我们对自己多媒体文件的重新更新操作。



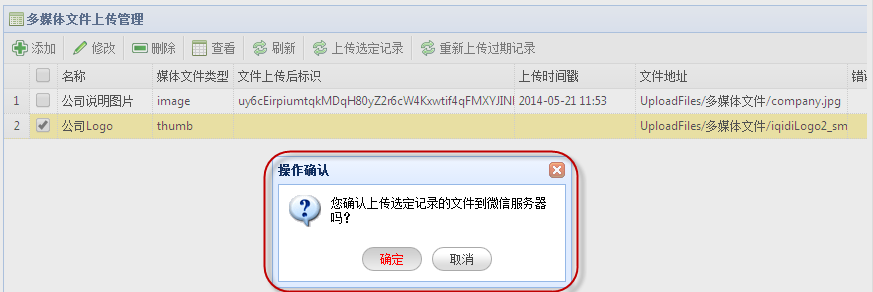
添加界面操作如下所示，其中引入了附件上传的控件进行文件的操作，非常方便。同时上传成功的文件，会在列表中列出。



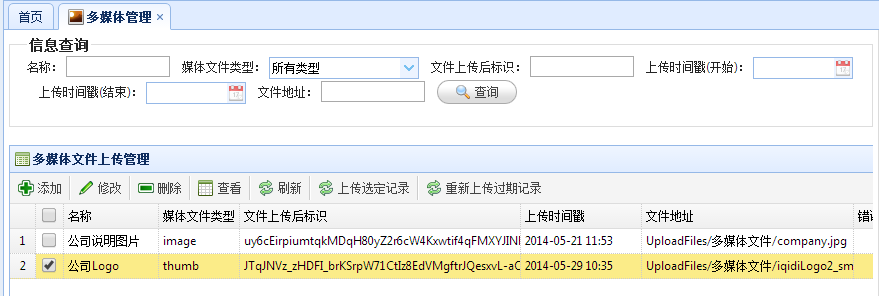
多媒体文件可以是下面几种方式：图片、语音、视频、缩略图。



保存后的数据记录，文件上传标识和时间戳都是空的，我们如果要使用，必须把他们上传到微信的服务器上，然后根据它的MediaId进行信息的发送，上传选定的记录操作界面如下所示。



多媒体文件顺利上传后，记录的信息如下所示。



**9、图文消息处理**

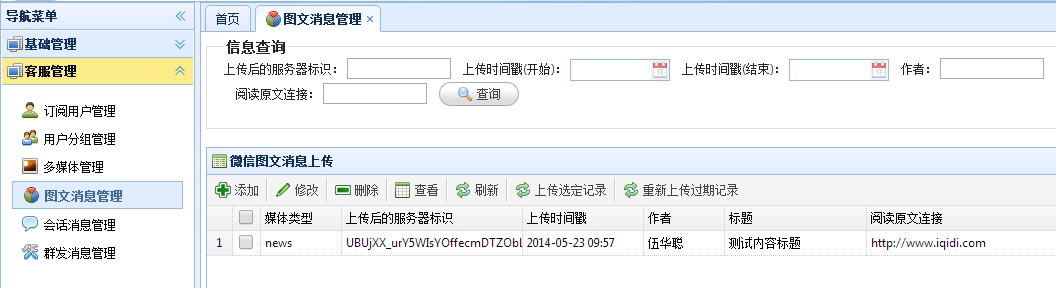
图文消息分为单图文消息和多图文消息两种，单图文消息如下所示。



多图文消息如下所示：



和多媒体数据管理一样，图文消息也是通过同样的方式进行管理，先上传到服务器，然后在进行消息的发送操作，多媒体消息一样有时间方面的限制要求，具体在我们的微信门户平台里面管理界面如下所示。



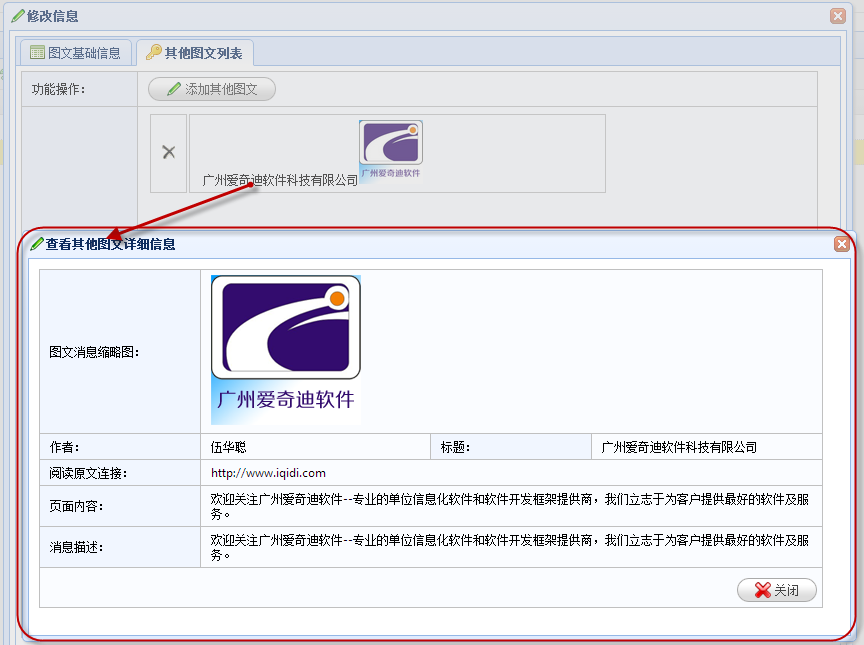
添加图文消息界面如下所示，保存后，可以在编辑界面中的“其他图文列表”里面，继续添加多图文的消息内容。



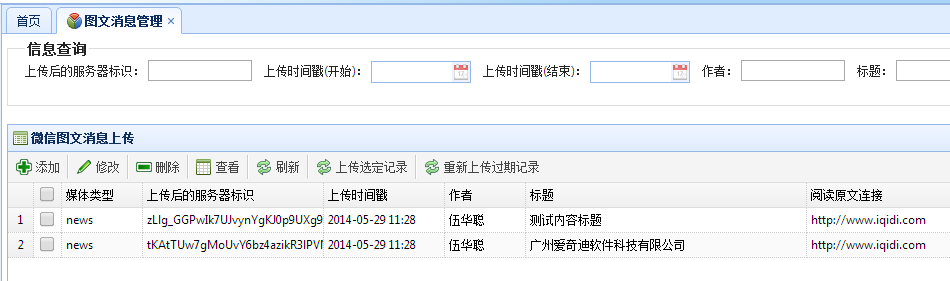
在添加界面中，选择图文消息的缩略图，都是通过选定指定的，已经上传到服务器上图片或者缩略图资源才可以的。



添加后的多图文列表，可以进行查看管理。

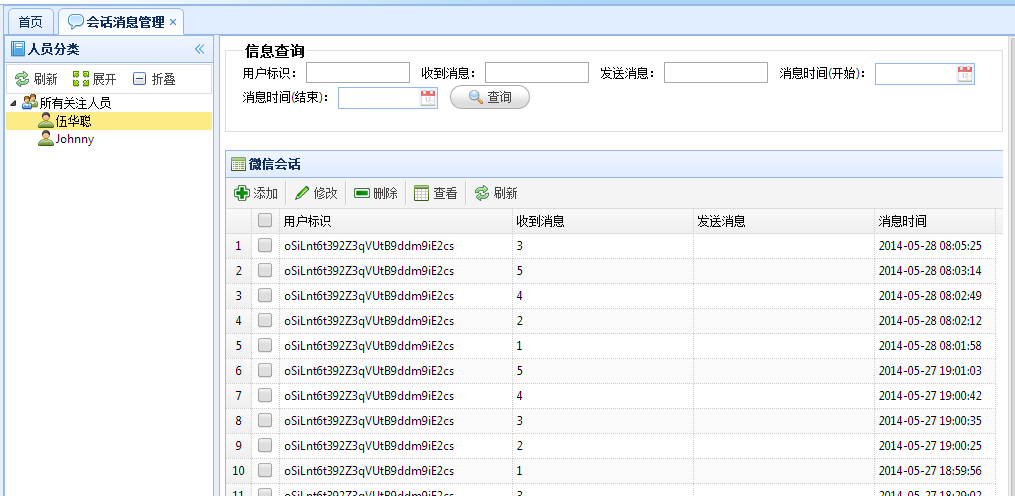


保存记录后，然后继续上传，上传后的记录界面如下所示，成功后返回一个上传后的服务器标识和时间戳，否则提示错误。



**10、会话消息管理**

为了方便记录客户的输入和发送信息，我们在微信门户管理平台里面记录用户的输入数据，具体会话消息管理界面如下所示。



我们可以双击最近48小时内的任何一条记录，可以给关注的客户进行消息的发送操作，如果消息发送成功，用户在手机的微信账号里面就能收到相关的发送消息了。



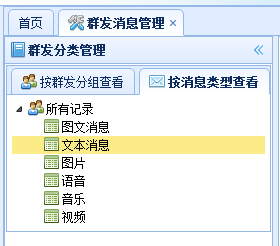
**11、群发消息管理**

为了对客户进行相应的营销操作，有时候我们需要对指定的群主或者人员进行消息的群发，让客户经常性的了解我们产品的信息和活动。

由于群发消息，除了文本消息，可以直接编辑发送外，其他数据，必须要求是上传到服务器的多媒体文件或者图文消息内容，因此前面的多媒体管理和图文消息管理，就是主要为了群发消息的目的引入的。有了上面的多媒体和多图文信息，我们从平台里面选择记录即可进行发送，从而省却麻烦的连带工作，实现高效的信息群发操作。



群发的消息，可以按群发分组进行查看，也可以按照消息类型进行查看，使得我们管理起来根据方便。



添加图文消息，可以选择文本消息、图文消息、图片消息等内容，根据不同的内容，界面提供不同的选择操作。

消息的群发类型分为两种，一种是根据分组，那么从平台里面选择对应的分组即可；一种是根据用户的OpenID进行发送，提供给用户输入。主要的操作界面如下所示。

