**一、简介**

新用户关注微信公众平台，将产生一个订阅事件，即**subscribe事件**，默认代码中没有对这一事件进行相应回复处理。

在新用户关注公众平台后，可能想知道该平台提供了哪些功能，以及怎样使用该平台，通俗一点讲就是该平台的“使用说明书”。

本文将详细讲述对subscribe事件的处理过程，回复相应信息提升交互性。

**二、思路分析**

微信目前提供了五种消息类型，分别为：

* 文本消息（text）；
* 图片消息（image）；
* 地理位置消息（location）；
* 链接消息（link）；
* 事件推送（event）；

接收到消息后，首先需要对消息类型做出判断，然后再针对不同类型的消息做出处理。在事件推送中，事件类型又分为三种，subscribe（订阅）、unsubscribe（取消订阅）、CLICK（自定义菜单点击事件），还需要再加一次判断；判断为subscribe事件后，根据设定好的欢迎消息，回复给用户。

**三、判断消息类型**

[复制代码](javascript:void(0);)

$postObj = simplexml\_load\_string($postStr, 'SimpleXMLElement', LIBXML\_NOCDATA);

$RX\_TYPE = trim($postObj->MsgType);

switch($RX\_TYPE)

{

case "text":

$resultStr = $this->handleText($postObj);

break;

case "event":

$resultStr = $this->handleEvent($postObj);

break;

default:

$resultStr = "Unknow msg type: ".$RX\_TYPE;

break;

}

[复制代码](javascript:void(0);)

**说明：**

$RX\_TYPE = trim($postObj->MsgType); 　　得到消息类型；

case "text":  
　　$resultStr = $this->handleText($postObj); 　　使用handleText() 函数处理文本消息；

case "event":  
　　$resultStr = $this->handleEvent($postObj); 　　使用handleEvent() 函数处理事件推送；

**四、判断事件类型**

[复制代码](javascript:void(0);)

switch ($object->Event)

{

case "subscribe":

$contentStr = "感谢您关注【卓锦苏州】"."\n"."微信号：zhuojinsz"."\n"."卓越锦绣，名城苏州，我们为您提供苏州本地生活指南，苏州相关信息查询，做最好的苏州微信平台。"."\n"."目前平台功能如下："."\n"."【1】 查天气，如输入：苏州天气"."\n"."【2】 查公交，如输入：苏州公交178"."\n"."【3】 翻译，如输入：翻译I love you"."\n"."【4】 苏州信息查询，如输入：苏州观前街"."\n"."更多内容，敬请期待...";

break;

default :

$contentStr = "Unknow Event: ".$object->Event;

break;

}

[复制代码](javascript:void(0);)

**说明：**

如果是subscribe事件，设定回复内容为“感谢您关注【卓锦苏州】...”；

**五、完整代码**

[复制代码](javascript:void(0);)

<?php

/\*\*

\* wechat php test

\*/

//define your token

define("TOKEN", "zhuojin");

$wechatObj = new wechatCallbackapiTest();

$wechatObj->responseMsg();

//$wechatObj->valid();

class wechatCallbackapiTest

{

/\*public function valid()

{

$echoStr = $\_GET["echostr"];

//valid signature , option

if($this->checkSignature()){

echo $echoStr;

exit;

}

}\*/

public function responseMsg()

{

//get post data, May be due to the different environments

$postStr = $GLOBALS["HTTP\_RAW\_POST\_DATA"];

//extract post data

if (!empty($postStr)){

$postObj = simplexml\_load\_string($postStr, 'SimpleXMLElement', LIBXML\_NOCDATA);

$RX\_TYPE = trim($postObj->MsgType);

switch($RX\_TYPE)

{

case "text":

$resultStr = $this->handleText($postObj);

break;

case "event":

$resultStr = $this->handleEvent($postObj);

break;

default:

$resultStr = "Unknow msg type: ".$RX\_TYPE;

break;

}

echo $resultStr;

}else {

echo "";

exit;

}

}

public function handleText($postObj)

{

$fromUsername = $postObj->FromUserName;

$toUsername = $postObj->ToUserName;

$keyword = trim($postObj->Content);

$time = time();

$textTpl = "<xml>

<ToUserName><![CDATA[%s]]></ToUserName>

<FromUserName><![CDATA[%s]]></FromUserName>

<CreateTime>%s</CreateTime>

<MsgType><![CDATA[%s]]></MsgType>

<Content><![CDATA[%s]]></Content>

<FuncFlag>0</FuncFlag>

</xml>";

if(!empty( $keyword ))

{

$msgType = "text";

$contentStr = "Welcome to wechat world!";

$resultStr = sprintf($textTpl, $fromUsername, $toUsername, $time, $msgType, $contentStr);

echo $resultStr;

}else{

echo "Input something...";

}

}

public function handleEvent($object)

{

$contentStr = "";

switch ($object->Event)

{

case "subscribe":

$contentStr = "感谢您关注【卓锦苏州】"."\n"."微信号：zhuojinsz"."\n"."卓越锦绣，名城苏州，我们为您提供苏州本地生活指南，苏州相关信息查询，做最好的苏州微信平台。"."\n"."目前平台功能如下："."\n"."【1】 查天气，如输入：苏州天气"."\n"."【2】 查公交，如输入：苏州公交178"."\n"."【3】 翻译，如输入：翻译I love you"."\n"."【4】 苏州信息查询，如输入：苏州观前街"."\n"."更多内容，敬请期待...";

break;

default :

$contentStr = "Unknow Event: ".$object->Event;

break;

}

$resultStr = $this->responseText($object, $contentStr);

return $resultStr;

}

public function responseText($object, $content, $flag=0)

{

$textTpl = "<xml>

<ToUserName><![CDATA[%s]]></ToUserName>

<FromUserName><![CDATA[%s]]></FromUserName>

<CreateTime>%s</CreateTime>

<MsgType><![CDATA[text]]></MsgType>

<Content><![CDATA[%s]]></Content>

<FuncFlag>%d</FuncFlag>

</xml>";

$resultStr = sprintf($textTpl, $object->FromUserName, $object->ToUserName, time(), $content, $flag);

return $resultStr;

}

private function checkSignature()

{

$signature = $\_GET["signature"];

$timestamp = $\_GET["timestamp"];

$nonce = $\_GET["nonce"];

$token = TOKEN;

$tmpArr = array($token, $timestamp, $nonce);

sort($tmpArr);

$tmpStr = implode( $tmpArr );

$tmpStr = sha1( $tmpStr );

if( $tmpStr == $signature ){

return true;

}else{

return false;

}

}

}

?>