

目录

- 1.获取监测配置参数
- 2.代码部署
 - 2.1 引入JS
 - 2.2 监控JS tar.init
 - 2.3 微信JS wx.config
 - 2.4 微信JS 分享设置
 - 2.5 自定义埋点
 - 2.6 在非微信环境下如何部署？
- 3.代码调试
- 4.部署完成
- 5.常见问题

1.获取监测配置参数

创建完监测任务后，便可获取到监测ID (tar_tid) ,监测token (tar_token)。

2.代码部署

部署四步骤：

1、 head中引入监测JS和微信JS → 2、 配置tar.init → 3、 配置wx.config → 4、 配置wx.ready

开发准备：

1) 开发者至少需要具备接入微信JSSDK分享接口的能力！

接入条件：认证订阅号/认证服务号

接入方法：参见 [接入微信分享](#)。

2) 开发者接入微信网页授权

接入条件：认证服务号

接入方法：参见 [接入微信授权](#)。

注意事项：

授权后重定向的回调地址redirect_uri，注意必须动态获取，不能 hardcode，且需要urlencode，否则授权过程中可能造成链接参数的丢失，导致监测数据不完整。

PHP、ASP、JSP等web语言动态获取页面完整URL方法：

- PHP

```
$redirect_url = 'http://' . $_SERVER['HTTP_HOST'] . $_SERVER['REQUEST_URI'];
```

- ASP

```
string redirect_url = Request.Url.ToString();
```

- JSP

```
String redirect_url = request.getScheme()+ "://" + request.getServerName()+request.getRequestURI()+"?"+request.getQueryString();
```

其他开发语言请参考对应编程语言API获取。

2.1 引入JS

第一步：在Html页面head头中引入监控JS

请注意，如果你的页面启用了https，务必引入<https://res.wx.qq.com/open/js/jweixin-1.2.0.js>，否则将无法在iOS9.0以上系统中成功使用微信JSSDK

```
<script type="text/javascript" src="//res.wx.qq.com/open/js/jweixin-1.2.0.js"></script>
<script type="text/javascript" src="//js.tarsocial.com/h5stat-2.1.1.js"></script>
```

2.2 监控JS tar.init

第二步：配置监控tar.init，需在wx.config之前完成

```
var tar_userinfo = []; //微信网页授权获取到的微信用户信息，默认为空；示例代码见下方。
tar.init({
  tar_debug:true //选填，debug模式：值为true开启、false为关闭，默认开启。
  tar_tid:"1xxxx", //必填，监测系统分配给此次监测活动的id
  tar_token:"xxxxxxxxxxxxxxxxxxxxx" //必填，监测系统分配给此次监测活动的token
  tar_userinfo:tar_userinfo
});
```

注意：需要将微信网页授权获取的信息传给tar_userinfo，示例如下：

```
<!--显性授权-->
var tar_userinfo = [{
  "openid":"OPENID",
  "nickname":"NICKNAME",
  "sex":"1",
  "province":"PROVINCE",
  "city":"CITY",
  "country":"COUNTRY",
  "headimgurl":"http://wx.qlogo.cn/mmopen/g3MonUZtNHkdmzicllibx6iaFqAc56vxLS",
  "unionid": "o6_bmasdasdsad6_2sgVt7hMZOPfL"
}];

<!--隐性授权-->
var tar_userinfo = [{
  "openid":"OPENID",
  "unionid": "o6_bmasdasdsad6_2sgVt7hMZOPfL"
}];
```

2.3 微信JS wx.config

```
wx.config({
  appId: "wxXXXXXXXXXX", //必填,公众号的唯一标识
  timestamp: "", //必填,生成签名的时间戳
  nonceStr: "", //必填,生成签名的随机串
  signature: "", //必填,签名,见 http://t.cn/RL24Fgw
  jsApiList: [
    "getNetworkType",
    "getLocation",
    "onMenuShareTimeline",
    "onMenuShareAppMessage",
    "onMenuShareQQ",
    "onMenuShareWeibo",
    "onMenuShareQZone" //如业务需求，可继续加入其他微信JS接口
  ]
});
```

2.4 微信JS 分享设置

```
wx.ready(function () {
  var shareData64 = {
    title: "这里填写分享标题",           //必填,分享标题
    desc: "这里填写分享描述",           //选填,分享描述
    imgUrl: "http://www.xxx.com/...", //选填,分享图链接
    link: "http://www.xxx.com/...",    //必填,支持直接填写location.href
    success: function () {
      // 用户确认分享后执行的回调函数
    },
    cancel: function () {
      // 用户取消分享后执行的回调函数
    }
  };
  wx.onMenuShareAppMessage(tar.shapeShareAppMessage(shareData64));
  wx.onMenuShareTimeline(tar.shapeShareTimeline(shareData64));});
```

2.5 自定义埋点

通过在页面元素中加上H5点击上报代码，来统计事件的点击次数。

- 1.数字列表项目统计页面中指定按钮或链接的点击量
- 2.统计需要点击效果的html标签的点击量

在需要点击效果的html标签中嵌入代码 `onclick="tar.hitTag('tagvalue')"`。示例代码：

```
tar.hitTag('tagvalue');//'tagvalue' 自定义，英文字母+数字
```

2.6 在非微信环境下如何部署？

在非微信环境下，代码部署与微信环境一致即可，监测JS会自动兼容其它平台（QQ、微博、QQ空间、PC等等）。

需要注意的是：

如果H5有接入微信网页授权，则需要动态判断H5是否是在微信浏览器内打开的。

不在微信内打开时需要关闭掉微信网页授权，否则页面将直接跳转至授权页面，导致无法浏览。

3.代码调试

开启Debug模式，在微信内打开活动网页，根据Alert信息查看监测是否部署正确，具体见“debug模式”。

```
tar_debug true, //选填，debug模式：值为true开启，默认关闭。
```

4.部署完成

代码部署正确后，即可开始统计监测数据，您可以登录系统，查看对应的统计报告。

5.常见问题

Q1、传播图中，所有或大部分节点均与活动源相连，传播关系仅有一层。

出现该种情况的原因是微信网页授权过程中丢失了一个名为tar_foid的参数，请仔细检查redirect_uri是否为动态获取。

解决方案：授权后重定向的回调地址redirect_uri，注意必须动态获取，不能hardcode，且需要urlencode，否则授权过程中可能造成链接参数的丢失，导致监测数据不完整。

Q2、监测JS部署完成，但是后台仅有PV、UV数据，没有分享数据或传播图数据为空。

请按照以下检查：

- 1.监测页面是否有JS报错
- 2.开启debug模式检查wx.config是否正确。
- 3.tar.init是否在wx.ready前面。
- 4.在wx.ready中是否有加入如下代码：

```
wx.onMenuShareAppMessage(tar.shapeShareAppMessage(shareData64));  
wx.onMenuShareTimeline(tar.shapeShareTimeline(shareData64));
```

Q3、H5接入了微信网页授权，但是后台没有昵称、头像等信息。

- 1、获取昵称、头像等数据需要使用显性授权。
- 2、确保获取到授权用户基本信息后，有传递给tar.init中的userinfo。参见 [2.2 监控JS tar.init](#)

Q4、为什么传播图中的节点之间的连线都是阅读原文及其他？

出现该种情况的原因是微信网页授权过程中丢失了一个名为from的参数，请仔细检查redirect_uri是否为动态获取。

解决方案：授权后重定向的回调地址redirect_uri，注意必须动态获取，不能hardcode，且需要urlencode，否则授权过程中可能造成链接参数的丢失，导致监测数据不完整。