

浙政钉h5&小程序应用采集开发手册

适用范围

采集流程

1、web（h5）引用SDK方式

基础埋点

用户信息埋点

2、小程序引用SDK方式

基础配置

用户信息埋点

3、上报日志API

record API说明

sendPV API说明

SDK埋点信息

SDK埋点业务规范

h5埋点注意事项：

适用范围

- 本文档适用于浙政钉业务web（h5）或小程序应用。
- 不适用App端native页面。

采集流程

埋点开发主要步骤如下：

- 采集首先需要引用SDK，SDK引用方式分为web（h5）和小程序两种方式；
- 完成SDK引用后，可设置用户信息进行用户信息埋点；
- 可通过上报日志API对埋点信息实现上报；

开发者可根据实际情况分别参考下文进行采集埋点开发。

1、web（h5）引用SDK方式

在页面head内引用applusJS

web (h5) 通用采集SDK：浙政钉钉交付对应专有云sdk地址，根据实际信息进行替换。

```
1 <script>
2   (function(w, d, s, q, i) {
3     w[q] = w[q] || [];
4     var f = d.getElementsByTagName(s)[0], j = d.createElement(s);
5     j.async = true;
6     j.id = 'beacon-aplus';
7     j.src = 'https://alidt.alicdn.com/alilog/mlog/aplus_cloud.js'
8   };
9   f.parentNode.insertBefore(j, f);
10  })(window, document, 'script', 'applus_queue');
11  // 如果是私有云部署还需要在上面那段JS后面紧接着添加日志域名埋点
12  // 通常私有云日志服务端域名类似于: quickapplus-web-api.xxx.com.cn , 具体域名见开发者后台配置信息
13  applus_queue.push({
14    action: 'applus.setMetaInfo',
15    arguments: ['applus-rhost-v', '替换为浙政钉采集域名']
16  });
17  applus_queue.push({
18    action: 'applus.setMetaInfo',
19    arguments: ['applus-rhost-g', '替换为浙政钉采集域名']
20  });
21  // 这个会落到app_key字段上
22  applus_queue.push({
23    action: 'applus.setMetaInfo',
24    arguments: ['appId', '替换为浙政钉appkey']
25  });
26 </script>
```

基础埋点

web (h5) 普通页面的PV日志是自动采集的，如果你的应用为单页应用，或者需要在PV日志里加上特定的埋点参数，可以这样埋点（如无这方面需求略过）：

```
1 // 单页应用 或 “单个页面”需异步补充PV日志参数还需进行如下埋点：
2 applus_queue.push({
3   action: 'applus.setMetaInfo',
```

```

4   arguments: ['aplus-waiting', 'MAN']
5 });
6 // 单页应用路由切换后 或 在异步获取到pv日志所需的参数后再执行sendPV:
7 aplus_queue.push({
8   'action': 'aplus.sendPV',
9   'arguments': [{
10     is_auto: false
11   }, {
12     // 自定义PV参数key-value键值对（只能是这种平铺的json，不能做多层嵌套），
    如：
13     x: 111,
14     y: 222
15   }]
16 })

```

说明：

如对sendPV这个API有疑问可以参考 [上报日志-sendPV API说明](#) 查阅其详细用法。

用户信息埋点

```

1 // 如采集用户信息是异步行为需要先执行这个BLOCK埋点
2 aplus_queue.push({
3   action: 'aplus.setMetaInfo',
4   arguments: ['_hold', 'BLOCK']
5 });
6
7 // 设置用户昵称
8 aplus_queue.push({
9   action: "aplus.setMetaInfo",
10  arguments: ["_user_nick", "当前会员用户昵称"]
11 });
12 // 设置用户ID
13 aplus_queue.push({
14   action: "aplus.setMetaInfo",
15   arguments: ["_user_id", "当前用户ID"]
16 });
17 aplus_queue.push({
18   action: "aplus.setMetaInfo",

```

```

19   arguments: ["_dev_id", "yourDeviceId"]
20 });
21
22 // 如采集用户信息是异步行为，需要先设置完用户信息后再执行这个START埋点
23 // 此时被block住的日志会携带上用户信息逐条发出
24 aplus_queue.push({
25   action: 'applus.setMetaInfo',
26   arguments: ['_hold', 'START']
27 });

```

注意事项：

如果需要采集用户信息，那么上述单页应用的sendPV调用也需要置于 `_hold:BLOCK` 和 `_hold:START` 配置之间。

2、小程序引用SDK方式

下载小程序通用采集SDK：<https://d.alicdn.com/alilog/mlog/aplus.js?id=202782072>

基础配置

1. 在小程序工程根目录中创建一个/Utils目录，再创建一个applus_mini_config.js
2. 在applus_mini_config.js做初始化配置：

```

1 // 如私有云日志服务端域名类似于：quickapplus-web-api.xxx.com.cn
2 {
3   'metaInfo': {
4     'appId': '替换为对应交付环境（分平台类型为Android/iOS/PC）的采集appkey',
5     // PV日志接收日志域名，必须按实际情况填写
6     'applus-rhost-v': '替换为对应交付环境的采集域名',
7     // 一般日志（如点击和曝光）接收日志域名
8     'applus-rhost-g': '替换为对应交付环境的采集域名',
9     // 开通自动采集点击事件时需要配置具体的clk_config
10  }
11 }

```

注意事项:

- 这里配置的applus-rhost-v、applus-rhost-g两个域名需要添加到小程序后台的request合法域名名单中

3. 将下载好的applus.js放在小程序工程目录/utils目录中
4. 在小程序app.js内引用applus_mini_config.js和applus.js:

```
1 import config from './utils/applus_mini_config.js'
2 var applus = require('./utils/applus.js')(config);
3
4 App({
5   applus
6 });
```

用户信息埋点

1. 在小程序获取到userInfo（用户ID、用户昵称）、设备ID时，插入以下代码:

```
1 var app = getApp();
2
3 // 如采集用户信息是异步行为需要先执行这个BLOCK埋点
4 app.applus.applus_queue.push({
5   action: 'applus.setMetaInfo',
6   arguments: ['_hold', 'BLOCK']
7 });
8
9 // _user_id为用户ID
10 app.applus.applus_queue.push({
11   action: "applus.setMetaInfo",
12   arguments: ["_user_id", "用户ID"]
13 });
14 // _user_nick为用户昵称
15 app.applus.applus_queue.push({
16   action: "applus.setMetaInfo",
17   arguments: ["_user_nick", "用户昵称"]
18 });
```

注意事项:

- 如果一开始就将_hold设置成了BLOCK，那么无论用户ID是否获取成功，最终都需要将_hold设置成START，否则日志都发不出去！

3、上报日志API

A+ 小程序和web（h5）采集SDK均采用指令队列机制来响应用户的请求，即用户无法直接调用SDK的功能性API，而是将对功能API的调用请求压入指令队列。SDK在完成加载/初始化后，从队列中读取指令顺序执行。

队列机制用于规避因JS加载异步而可能出现的API调用早于API就绪的调用异常，以及可能伴生的其他日志异常。

SDK对用户公开的API只有一个，即向指令队列推入日志指令。API定义如下：

```
1 aplus_queue.push({
2     'action': '$APIName',
3     'arguments': [$arguments]
4 })
```

其中，

- action参数为使用的API的名称，其入参为一个字符串，取值为枚举值，可使用的值如下（样例代码请参考前面章节）：
 - record
 - sendPV
- arguments参数为action中指定的API的入参，其入参为一个数组，数组内对象的顺序与该API接口定义的入参顺序一致。

record API说明

record方法将发送一条事件日志。其API定义如下：

```
1 // API格式
2 aplus_queue.push({
3     'action': 'aplus.record',
4     'arguments': [$trackerEventCode, $eventType, $eventParams, $method]
5 })
```

其中:

trackerEventCode为注册的事件编码. 当上报的事件为PV事件时, trackerEventCode可传空值或 'PageView' 常量

eventType 可取值及业务含义如下:

- 'EXP':自定义曝光事件
- 'CLK':自定义点击事件
- 'OTHER': 其他自定义事件

eventParams为本次事件中上报的事件参数. 其取值为一个JSON对象 (平铺的简单对象, 不能多层嵌套). JSON中的key不能是以下保留属性. 需要注意的是:

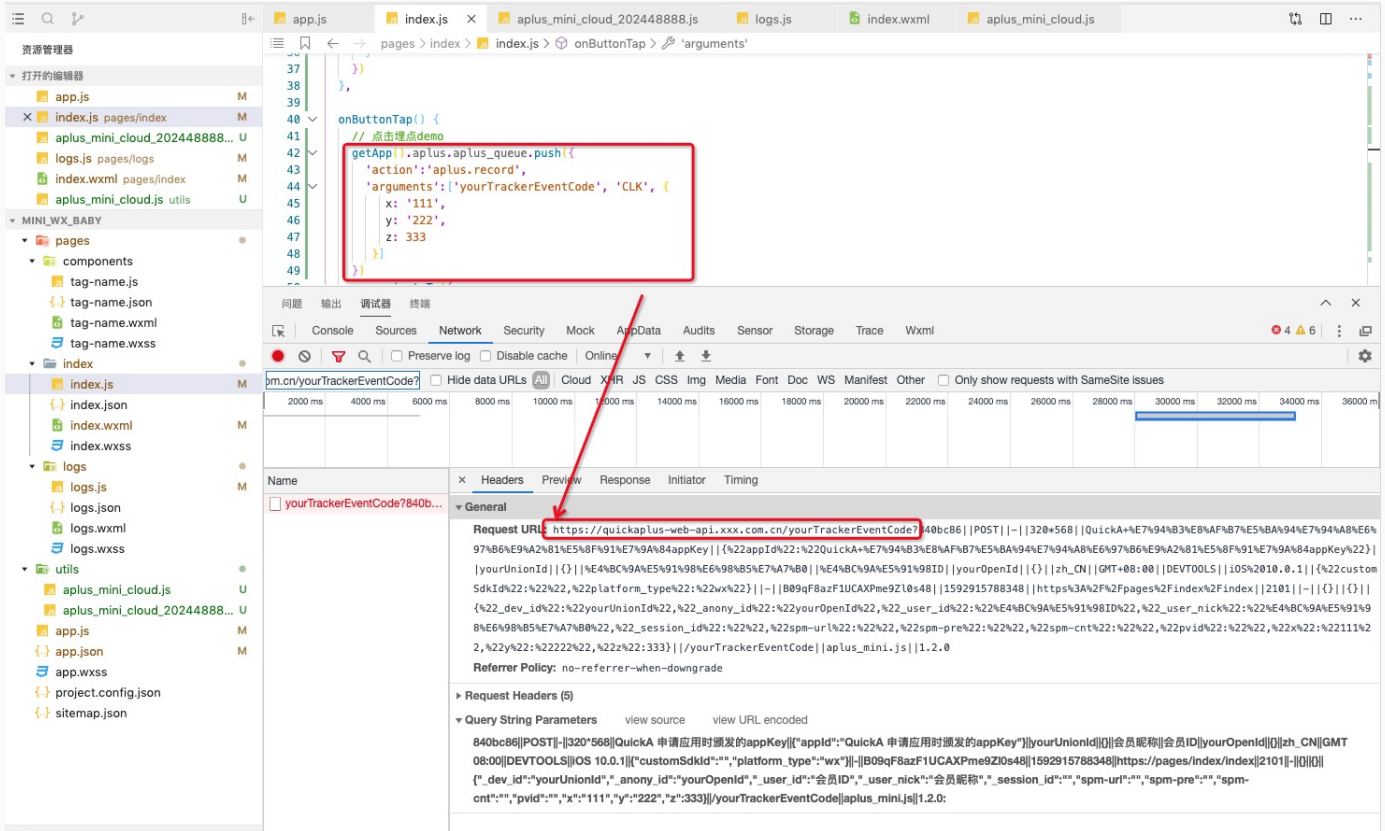
- 调用record API上报参数时, 该次赋值仅对该条事件有效
- 保留属性: uidaplus,spm-url,spm-pre,spm-cnt,pvid,_dev_id,_anony_id,_user_id,_user_nick,_session_id

method 可取值即业务含义如下:

- 'GET': 用get请求上报日志 (目前仅支持GET请求, 此参数不传默认值也是GET)

```
1 // 一个简单的demo
2 aplus_queue.push({
3   'action':'aplus.record',
4   'arguments':['yourTrackerEventCode', 'CLK', {
5     x: '111',
6     y: '222',
7     z: 333
8   }]
9 })
```

效果如:



sendPV API说明

sendPV方法将发送一条页面PV日志. 其API定义如下:

```
1 // API格式
2 aplus_queue.push({
3   'action': 'aplus.sendPV',
4   'arguments': [$pageEventConfig, $userdata]
5 })
```

其中:

pageEventConfig为页面事件的配置, 只需要写死 { is_auto: false } 即可

userdata为本次页面事件的扩展参数, 其取值为一个JSON对象 (平铺的简单对象, 不能多层嵌套)

```
1 // 一个简单的demo
2 aplus_queue.push({
3   'action': 'aplus.sendPV',
4   'arguments': [{
5     is_auto: false
```

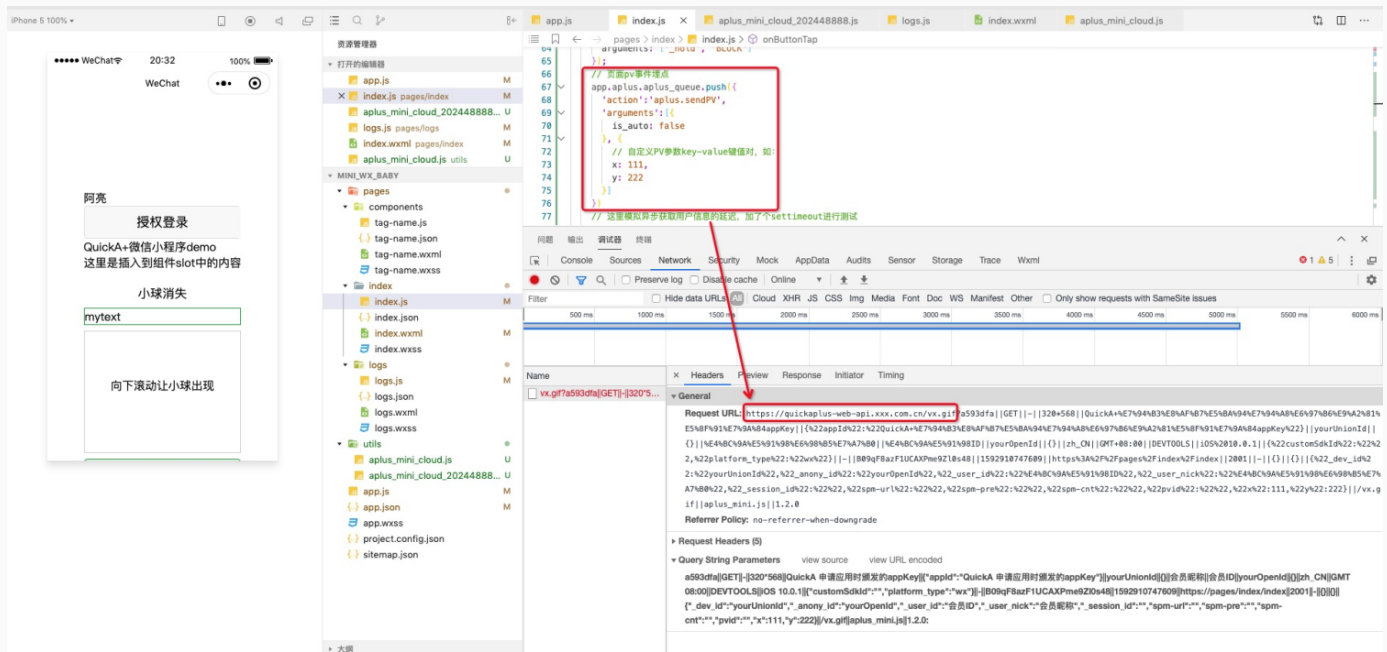


```

6    }, {
7      sapp_id: '需替换为对应小程序在开放平台app_base_info表主键',
8      sapp_name: '需替换为对应小程序在开放平台的应用标识',
9      // 自定义PV参数key-value键值对，以下内容必填，根据应用实际情况设置。
10     page_id: '页面ID，与page 参数配合使用，保证唯一性',
11     page_name: '页面中文名称',
12     page_url: '页面URL'
13   }]
14 })

```

效果如：



SDK埋点信息

此SDK采集的信息均会落入原始日志表，以下说明的是采集信息与表字段的关系

埋点信息	字段信息	自动采集	说明
_user_id	biz_user_id	N	用户ID
_user_nick	biz_user_nick	N	用户昵称
appId	app_key	N	QuickA+颁发的appKey
_anony_id	biz_anony_id	N	匿名用户ID
_session_id	biz_session_id	N	业务端维护的session id

event_code	event_id	N	页面事件：2001；点击事件：2101；曝光事件：2201；其他事件：19999
event_args	event_args	N	对应页面事件日志的userdata参数； 对应一般事件日志的eventParams参数，数据格式为平铺的JSONString，如{x:111,y:222}，不支持多层嵌套
logkey	arg1	N	对应页面事件日志的枚举值（小程序端："/vx.gif"、H5："/m.gif"、PC："/v.gif"） 对应普通事件日志的trackerEventCode
time	event_timestamp	Y	客户端上报日志的时间
sdk_info	sdk_info	Y	applusJS SDK信息，如 jsver=applus_mini.js&lver=1.1.15
url	page	Y	小程序当前URL，格式如https:// + currentPagePath
pre	pre	Y	来源页的pageUrl，小程序首页为空值，二级页面的来源是首页的url，以此类推
LANG	locale	Y	地区-语言信息
timeZone	timezone	Y	设备使用的时区
os	os	Y	操作系统 如："OS X"
os_version	os_version	Y	操作系统版本号/Build码，如："10_14_4"
resolution	resolution	Y	屏幕宽度*屏幕高度，如：" 1792*828"
UA	user_agent	Y	客户端UA信息
-	global_args	-	用户设置的全局参数，生命周期级别（暂不支持）
-	session_args	-	用户设置的Session级参数，跨Session清空（暂不支持）

注意事项：

- 标注自动采集为Y的属于采集SDK自动采集的范畴，不需要开发埋点！

SDK埋点业务规范

参数名称	参数说明	是否必填	示例（非实际值，仅供参考）	备注

app_key		应用唯一标识	Y	2190879*	开放平台提供
app_version		app版本号	Y	5.23.1	
arg1		用户扩展预留， 用于元素唯一表示	Y	UICompatibilityInputViewController	元素唯一标志
page		页面完整的url或者screen name	Y	UICompatibilityInputViewController	保证唯一性
ll_userid		长登录id	Y	8afac0cc5d36cf79015d40669b3d2442	
ll_usernick		长登录用户名称	Y	张**	
biz_user_id		业务端用于识别已登录用户的会员id(短登录id)	Y	8afac0cc5d36cf79015d40669b3d2442	
biz_user_nick		业务端用于识别已登录用户的会员标识(短登录nick)	Y	张**	
global_args	ding_tenantId	租户ID	Y	101*27	SaaS版多租户情况下填写
	gdLbs	GPS定位	Y	省#市#区#adcode#街道	可参考类似地理位置服务信息，采用坐标系与高德一致。 https://lbs.amap.com/api/android-location-sdk/locationsummary/
	long	经度	Y		
	lati	纬度	Y		
event_args	site_id	站点ID	N		
	sapp_id	应用开放平台注	Y		开放平台提供

		册的唯一ID			应用开放平台 注册的唯一标识（须保证唯一性） 目前使用的为 开放平台 app_base_info 表主键
sapp_name	应用开放平台应用名称	Y			
userId	用户ID，web端埋点	Y			
userName	用户名称	Y			
userType	用户类型，个人/企业	N			
clickName	点击元素中文名称	Y			
page_id	页面ID，与page参数配合使用，保证唯一性	Y			
page_url	页面URL				
page_name	页面中文名称	Y			
cardId	楼层ID	Y			
cardName	楼层名称	Y			
index	坑位，0开始	Y			
itemId	元素编号	Y			
itemTitle	元素标题	Y			
itemType	元素类型， MATTER：事项 APP：应用 MESSAGE：消息 NEWS：新闻	Y			

业务参数，页面、楼层、坑位及元素根据应用自身业务情况进行规划并进行埋点上报。

h5埋点注意事项：

- 1.埋点参数value值必须是字符串类型；
- 2.基础埋点（sendPV调用）数据需要置于需要置于 _hold:BLOCK 和 _hold:START配置之间。
- 3.埋点代码若通过js引用需要置于script标签中；
- 4.若通过js统一引用的话，A+埋点本身也引用了js，需要参考通过js再引用js的写法。可以参考：
<https://www.cnblogs.com/cxx8181602/p/9340678.html>