



多厂商推送透传消息带通知使用文档 (restful)

日期: 2020年9月28日

变更记录

序号	变更时间	变更人	变更内容	SDK版本
0	2018-01-26	李居启	新建文档	
1	2019-02-14	陈云飞	去掉payload接口设置	
2	2019-05-21	陈云飞	添加厂商推送内容限制说明	
3	2019-05-21	陈云飞	添加intent推送设置注意点	
4	2020-07-15	陈云飞	添加厂商通知扩展字段使用说明	
5	2020-09-09	陈云飞	添加小米、OPPO富文本消息设置说明	
6	2020-09-28	陈云飞	添加华为富文本消息设置说明	

目录

- 1. 概述
- 2. 消息下发逻辑
- 3. 使用方式
 - 3.1 公共参数【必选】
 - 3.2 options通知扩展功能参数【可选】
 - 3.2.1 华为
 - 3.2.1.1角标
 - 3.2.1.2 富文本消息
 - 3.2.2 小米
 - 3.2.2.1 消息分类
 - 3.2.2.2 富文本消息
 - 3.2.3 OPPO
 - 3.2.3.1 公、私信
 - 3.2.3.2 富文本消息
 - 3.2.4 VIVO
 - 3.2.4.1 消息分类
- 4. 功能代码示例
- 5. 附录：Andriod开发工程师生成intent参考
 - 5.1 生成Intent
 - 5.2 设置 package (必选)
 - 5.3 设置 component (必选)
 - 5.4 设置 launchFlags (不起作用)
 - 5.5 设置参数
 - 5.6 设置action
 - 5.7 生成 intent uri
- 附件1：Android厂商推送消息流程图

前言

本文档包含了来自浙江每日互动网络科技股份有限公司的机密技术和商业信息，这些信息作为浙江每日互动网络科技股份有限公司专用。接受本文档即视为表示同意对其内容保密，并且未经浙江每日互动网络科技股份有限公司书面许可，不得复制、泄露或散布本文档的全部或部分内容。

本文档及其描述的内容及产品的著作权和版权受有关法律保护，对本文档内容的任何形式的非法复制，泄露或散布，将导致承担相应的法律责任。

浙江每日互动网络科技股份有限公司保留在不另行通知的情况下修改本文档的权利，并保留对本文档内容的全部解释权。

This document contains confidential technical and commercial information of Zhejiang Merit Technology Co., Ltd. and it can only be used for partners of Zhejiang Merit Technology Co., Ltd. for the purposes of related projects. Any contents shall not be reproduced or transmitted in any form or means without the document permission of Zhejiang Merit Technology Co., Ltd.

This document and the product it describes are protected by copyrights according to the applicable laws. Information in this document is subject to change without notice and describes only the product defined in the introduction of this documentation.

Zhejiang Merit Technology Co., Ltd. will, if necessary, explain issues, which may not be covered in this document.

1. 概述

由于第三方厂商和个推对特定推送参数支持不统一，个推为了给客户提供一致的推送服务，对推送参数做了特殊规定，解决接入第三方推送遇到的兼容问题，降低接入成本。

2. 消息下发逻辑

目前第三方厂商的通知到达率显著高于第三方透传的到达率。所以厂商渠道也建议使用通知，。消息默认下发策略：**SDK在线状态，走个推渠道下发透传消息，SDK离线状态，走厂商渠道下发通知消息，流程图请参见 附件1**。该文档不再介绍个推渠道的参数设置，着重介绍下第三方厂商渠道的通知消息功能使用。[restApi V2文档链接](#)。

3. 使用方式

push_channel 厂商通道消息内容，设置push_channel/android/ups/notification对象的title, content, intent等参数。

3.1 公共参数【必选】

成员和方法名	类型	必填	说明
title	String	是	通知栏标题（长度建议取最小集） 小米：title长度限制50字 华为：title长度限制40字 魅族：title长度限制32字 OPPO：title长度限制32字 VIVO：title长度限制 40 英文字符
body	String	是	通知栏内容（长度建议取最小集） 小米：content长度限制128字 华为：content长度小于1024字 魅族：content长度限制100字 OPPO：content长度限制200字 VIVO：content长度限制 100 个英文字符
click_type	String	是	点击通知后续动作, 目前支持5种后续动作, intent：打开应用内特定页面， url：打开网页地址， payload：启动应用加自定义消息内容， startapp：打开应用首页， none：纯通知，无后续动作
intent	String	否	长度不大于2048字节（vivo限制1024字节，建议不要超过该限制，以免请求vivo失败），通知带intent传递参数推荐使用，intent最好由Android开发工程师生成，生成方式见附录5

示例：

```
{
```

```

"request_id": "12333333332222333",
  "settings": {
    "ttl": 3600000
  },
  "audience": {
    "cid": [
      "98efe7d710e5d717be458f8d84136f64"
    ]
  },
  "push_message": {.....略},
  "push_channel": {
    "android": {
      "ups": {
        "notification": {
          "title": "厂商通知标题",
          "body": "厂商通知内容",
          "click_type": "intent",
          "intent": "intent串"
        }
      }
    }
  }
}

```

3.2 options通知扩展功能参数【可选】

成员和方法名	类型	必填	说明
constraint	String	否	扩展内容对应厂商通道设置如：HW,MZ,...
key	String	否	各厂商渠道消息相关的扩展字段，如： hw角标add_num， vv的消息分类classification， xm的channel。等等
value	String or int	否	value的设置根据key值决定,例如： hw角标设置：key设为add_num，value：1（原来的角标值+1），等等

3.2.1 华为

3.2.1.1角标

角标设置需要App端配合使用 [接口类说明-16 设置华为角标](#)。

成员和方法名	value 值类型	必填	说明
constraint	---	是	HW
key, value	String	是	key: /message/android/notification/badge/class, value: 应用入口Activity路径名称
	int	是	key: /message/android/notification/badge/add_num, value: 应用角标累加数字非应用角标实际显示数字。为大于0小于100的整数。
	int	是	key: /message/android/notification/badge/set_num, value: 角标设置数字。值大于等于0小于100的整数, 如果set_num与add_num同时存在时, 以set_num为准。

示例:

```
"options": [
  {
    "constraint": "HW",
    "key": "/message/android/notification/badge/class",
    "value": "应用入口Activity路径名称"
  }, {
    "constraint": "HW",
    "key": "/message/android/notification/badge/add_num",
    "value": 1
  }, {
    "constraint": "HW",
    "key": "/message/android/notification/badge/set_num",
    "value": 2
  }
]
```

注意事项:

1. 点击启动图标或通知栏系统并不会清理角标数, 需应用在端侧通过角标API去清理角标;
2. add_num: EMUI版本8.0.0 (及以上), 推送服务App版本8.0.0 (及以上) 支持;
3. set_num: EMUI版本10.0.0 (及以上), 推送服务App版本10.1.0 (及以上) 支持。

3.2.1.2 富文本消息

功能	成员和方法名	value 值类型	必填	说明
公共参数	constraint	---	是	HW
通知小图	key, value	String	是	key: /message/android/notification/image, value: 请写入对应图标https地址。URL使用的协议必须是HTTPS协议，取值样例：https://example.com/image.png。
通知长文本	key, value	int	是	key: /message/android/notification/style, value: 1。通知栏样式1，为长文本。
		String	是	key: /message/android/notification/big_title, value: 通知title内容。设置big_title后通知栏展示时，使用big_title而不用title。
		String	是	key: /message/android/notification/big_body, value: 通知body内容。设置big_body后通知栏展示时，使用big_body而不用body。

示例：

```
"options": [  
  {  
    "constraint": "HW",  
    "key": "/message/android/notification/image",  
    "value": "公网可以访问的https图标链接"  
  }, {  
    "constraint": "HW",  
    "key": "/message/android/notification/style",  
    "value": 1  
  }, {  
    "constraint": "HW",  
    "key": "/message/android/notification/big_title",  
    "value": "big_title"  
  }, {  
    "constraint": "HW",  
    "key": "/message/android/notification/big_body",  
    "value": "big_body"  
  }  
]
```

注意事项：1.通知长文本：EMUI版本9.1.0（及以上），推送服务App版本9.1.1（及以上）支持。

3.2.2 小米

3.2.2.1 消息分类

小米推送的消息通道分为“普通消息”（默认）和“通知消息”两类，默认下发普通消息。普通消息单日可推送数量有限制，通知消息不限。通知消息申请请参见[小米推送消息限制说明 FAQ4](#)。

成员和方法名	value值类型	必填	说明
constraint	---	是	XM
key, value	String	是	key: channel, value: 填写小米平台申请的渠道id

示例:

```
"options":[
  {
    "constraint":"XM",
    "key":"channel",
    "value":"Default"
  }
]
```

3.2.2.2 富文本消息

先调用小米接口，上传图片，获取图片url。java[4.4.1 大图/大文本/Large icon](#)， restAPI[11.1 大图、大文本/上传大图API](#)。

功能	成员和方法名	value值类型	必填	说明
公共参数	constraint	---	是	XM
通知小图	key, value	String	是	key: notification_large_icon_uri, value: 填写接口返回的图片链接。Large icon可以出现在大图版和多字版消息中，显示在右边，通知小图。图片要求：尺寸必须为120×120px，文件小于200KB，PNG/JPG/JPEG格式。
通知大文本	key, value	String	是	key: notification_style_type, value: 1。 <u>大文本和大图二选一</u> 。
				key: notification_bigTxt, value: 文本内容。支持展示多行文字；最多128字，可以通过\n换行。
通知大图	key, value	String	是	key: notification_style_type, value: 2。 <u>大文本和大图二选一</u> 。
				key: notification_bigPic_uri, value: 填写接口返回的图片链接。图片要求：固定876x324px，小于1M，PNG/JPG/JPEG格式。

示例:

```
"options":[
  {
    "constraint":"XM",
    "key":"notification_large_icon_uri",
    "value":"xxxxxxxxx"
```

```

    },
    {
      "constraint": "XM",
      "key": "notification_style_type",
      "value": "1"
    },
    {
      "constraint": "XM",
      "key": "notification_bigTxt",
      "value": "xxxxxxxxxxxxxxxxxxxxxxxxxx"
    },
    {
      "constraint": "XM",
      "key": "notification_style_type",
      "value": "2"
    },
    {
      "constraint": "XM",
      "key": "notification_bigPic_uri",
      "value": "xxxxxxxxxxxxxxxxxxxxxxxxxx"
    }
  ]
]

```

注意事项:

- 1.富文本消息仅在MIUI10及以上版本支持，在低版本和非MIUI系统上，消息将按照普通消息的样式展示；
- 2.国内版MIUI系统中，仅在MIUI12及以上版本支持large icon，MIUI12以下版本不会展示；
- 3.发送大图消息时，如果因为网络原因图片下载失败后将不会展示图片，而是按照普通消息样式展示。

3.2.3 OPPO

3.2.3.1 公、私信

OPush平台上所有通道分为“公信”（默认）、“私信”两类，默认下发公信消息。公信消息单日可推送数量有限制，私信消息不限（仅限单个用户）。私信消息申请请参见[OPPO PUSH通道升级公测邀请](#)。

成员和方法名	value值类型	必填	说明
constraint	---	是	OP
key, value	String	是	key: channel, value: 填写OPPO平台登记的渠道id

示例:

```

"options":[
  {
    "constraint":"OP",
    "key":"channel",
    "value":"Default"
  }
]

```

注意事项:

- 1.该接口需要联系个推技术人员申请开通才可以使用；
- 2.OPPO私信消息仅支持单推。

3.2.3.2 富文本消息

先调用OPPO接口，上传图片，获取图标url。restAPI“[图片上传](#)”。

功能	成员和方法名	value 值类型	必填	说明
公共参数	constraint	---	是	OP
通知小图	key, value	String	是	key: small_picture_id, value: 填写接口返回的图标id。图片要求：尺寸144*144 px，文件大小为50k以内,格式为PNG/JPG/JPEG。
通知长文本	key, value	int	是	key: style, value: 2。长文本和大图二选一。
		String	是	复用通知content接口，长文本样式（style 为 2）限制字数128个以内。
通知大图	key, value	int	是	key: style, value: 3。长文本和大图二选一。
		String	是	key: big_picture_id, value: 填写接口返回的图标id。大图样式（style 为 3）限制通知content字数50以内。图片要求：尺寸876*324 px,文件大小1M以内，格式为PNG/JPG/JPEG。

示例：

```
"options":[
  {
    "constraint":"OP",
    "key":"small_picture_id",
    "value":"xxxxxxxxxxxxxx"
  },
  {
    "constraint":"OP",
    "key":"style",
    "value":2
  },
  {
    "constraint":"OP",
    "key":"style",
    "value":3
  },
  {
    "constraint":"OP",
    "key":"big_picture_id",
    "value":"xxxxxxxxxxxxxx"
  }
]
```

```
}  
]
```

注意事项：

- 1.通知小图标、长文本、大图不支持单推，单推请求会返回无权限错误；
- 2.长文本样式（ColorOS版本>5.0可用，通知栏第一条消息可展示全部内容，非第一条消息只展示一行内容）；
- 3.大图样式（ColorOS版本>5.0可用，通知栏第一条消息展示大图，非第一条消息不显示大图，推送方式仅支持广播，且不支持定速功能）。

3.2.4 VIVO

3.2.4.1 消息分类

vivo消息分类功能将推送消息类型分为运营消息和系统消息，默认下发运营消息。运营消息单用户单应用单日接收条数上限为5条，系统消息不限。系统消息功能不用申请，可以直接使用，如特殊情况需额外提升系统消息量级，请参见[vivo推送消息分类功能说明](#)。

成员和方法名	value值类型	必填	说明
constraint	---	是	VV
key, value	int	是	key: classification, value: 0 代表运营消息，1代表系统消息，不填默认为0

示例：

```
"options": [  
  {  
    "constraint": "vv",  
    "key": "classification",  
    "value": 0  
  }  
]
```

4. 功能代码示例

请求数据示例如下，鉴权和其它接口说明参见个推官网文档 [restAPI V2服务端文档](#)。

```
{  
  "request_id": "12333333332222333",  
  "settings": {  
    "ttl": 3600000  
  },  
  "audience": {  
    "cid": [  
      "98efe7d710e5d717be458f8d84136f64"  
    ]  
  },  
  "push_message": { .....略 },  
  "push_channel": {  
    "android": {  
      "ups": {  
        "notification": {
```

```
"title": "厂商通知标题",  
"body": "厂商通知内容",  
"click_type": "intent",  
"intent": "intent串",  
"options": [  
    {  
        "constraint": "HW",  
        "key": "/message/android/notification/badge/add_num",  
        "value": 1  
    },  
    {  
        "constraint": "HW",  
        "key": "/message/android/notification/badge/class",  
        "value": "com.getui.demo.GetuiSdkDemoActivity"  
    }  
]  
}  
}  
}
```

5. 附录：Andriod开发工程师生成intent参考

Android开发工程师生成intent，可以在Android代码中定义一个Intent，然后通过Intent.toUri(Intent.URI INTENT SCHEME)获取对应的Intent Uri。

5.1 生成Intent

```
Intent intent = new Intent();
```

5.2 设置 package (必选)

如果设置了package，则表示目标应用必须是package所对应的应用。

```
intent.setPackage("com.getui.demo");
```

5.3 设置 component (必选)

component为目标页面的准确activity路径，component中包含应用包名和activity类的全路径。

```
intent.setComponent(new  
ComponentName("com.getui.demo","com.getui.demo.TestActivity"));
```

5.4 设置 launchFlags (不起作用)

launcherFlags 可指定在启动activity时的启动方式，个推sdk以及厂商sdk在启动activity时会自动设置为FLAG_ACTIVITY_NEW_TASK，所以该设置没有实际意义。

5.5 设置参数

可通过intent.putExtra方法设置一系列的参数。

```
intent.putExtra("stringType", "string");
intent.putExtra("booleanType", true);
intent.putExtra("doubleType", 1.0);
intent.putExtra("floatType", 1.0f);
intent.putExtra("longType", 1L);
intent.putExtra("intType", 1);
// 还有其他类型的参数可以设置，通过查找`putExtra`方法的重载方法即可
```

注：由于oppo只支持String类型的参数，设置其它类型参数，App端接收以后，会强转为String类型，所以建议参数统一设置为String类型。参数可以分开设置，也可以合并为一个json格式数据。

5.6 设置action

由于oppo sdk仅支持通过action的方式打开activity，所以若您的应用需要支持oppo推送（打开指定页面），您需要在intent中设置action，并在AndroidManifest.xml配置对应的action，如下：

```
/设置action
intent.setAction("android.intent.action.oppopush");
```

AndroidManifest.xml配置如下：

```
<!-- android:exported="true" 必须配置，否则部分机型上无法弹通知，影响消息的跳转-->
<activity android:name="com.getui.demo.TestActivity"
    android:exported="true">
    <intent-filter>
        <!-- category 为必须设置项 设置为 android.intent.category.DEFAULT 即可-->
        <category android:name="android.intent.category.DEFAULT" />
        <action android:name="android.intent.action.oppopush" />
    </intent-filter>
</activity>
```

5.7 生成 intent uri

```
Log.d(TAG, intent.toUri(Intent.URI_INTENT_SCHEME));
```

打印结果为：

```
intent:#Intent;launchFlags=0x10000000;action=android.intent.action.oppopush;package=com.getui.demo;component=com.getui.demo/com.getui.demo.TestActivity;floatType=1.0;longType=1;booleanType=true;stringType=string;doubleType=1.0;intType=1;end
```

将生成的intent uri交给服务端开发人员，在对应模板的setIntent方法中设置即可。

注：1、S大写是string类型，s小写是short类型，B大写是boolean类型，b小写是byte类型，其它均小写即可；

2、设置的参数如有特殊字符，比如Scheme链接含有#符号，建议urlencode编码处理，参数在经过Android系统接收后，会自动进行解码处理（经测试，intent参数里面有 % 符号，客户端接收参数会乱码；intent参数里面有 # 和 ; 符号，会影响跳转，urlencode编码处理后正常。）；

3、如果遇到点击第二条通知，页面不能正常跳转的情况，建议设置launchFlags为0x04000000测试，这个一般是activity启动模式的问题，参数会在activity的onCreate或onNewIntent方法回调。

附件1: Android厂商推送消息流程图

