# ColorOS桌面角标功能使用介绍

1. **调试准备工作：**

在ColorOS中，桌面角标有白名单控制，所以需要调试的机型接收到新的名单，以开启对应应用的角标开关。

1. 准备具备桌面角标功能的机器：

**Android6.0及以上+ColorOS3.0**及以上，推荐**R9S、R11**等比较新的机型。

1. 进入拨号盘，输入 \*#06# ，提供机器的**IMEI**号；提供“关于手机”中，版本相关信息。
2. 提供调试APK的**包名、主类名（配置了桌面图标的类）**，或者直接发个apk过来。
3. 待我方告知已推送名单后，在测试机型上进行如下操作：

① 手机正常联网，有安装需要适配角标的应用

② 启动拨号盘，输入\*##\*8110#，进入工程模式，开启OTA测试开关，直接按返回键退出

③ 再次 \*##\*8110#，进入工程模式，不用操作，退出

④ 进入设置-通知与状态栏-通知管理-“待适配的应用”，看看有没有角标的选项开关。如果有开关，则代表该机器已支持应用显示角标，若无开关，请联系我方一起排查。

**二、适配方案介绍**

整个流程简介：

1. 应用程序调用ContentProvider 的接口来通知应用程序角标数据变化，接口如下：

getContentResolver().call(Uri.parse("content://com.android.badge/badge"),"setAppBadgeCount", null, **extras**);

1）参数**extras** 类型为**Bundle**，初始化如下

a.设置未读数量extras.putInt("app\_badge\_count",count);// count最大设置99，大于99默认显示99+

b.设置显示角标的应用extras.[putString](http://opengrok.scm.adc.com:8080/source/s?defs=putString&project=R7_Master3.0)("app\_badge\_packageName", [packageName](http://opengrok.scm.adc.com:8080/source/s?defs=packageName&project=R7_Master3.0))；这项可以不设置，桌面会去获取调用方的包名

2. ContentProvider 更新应用程序角标数据，并通知系统桌面应用；

3. 系统桌面应用对应用程序图标进行角标更新

**对应用程序来说，只需要关注第1步**，即调用接口设置角标数量。

参考代码如下：

|  |
| --- |
| private int mCurrentTotalCount = -1;  //**badgeCount**即为要设置的未读数据。  public void setAppBadgeCount(int **badgeCount**) {  if (mCurrentTotalCount == badgeCount) {  return; //如果当前数据没发生变化，尽量不要重复设置  }  mCurrentTotalCount = badgeCount;  try {  Bundle extras = new Bundle();  extras.putInt("app\_badge\_count",**badgeCount**);  getContentResolver().call(  Uri.parse("content://com.android.badge/badge"),  "setAppBadgeCount", null, extras);  } catch (Exception e) {  Log.e(TAG, "Write unread number FAILED!!! e = " + e);  }  } |

**特别注意：**该方法会操作桌面数据库，且会引起桌面界面刷新，一定要在真正需要更新数据的时候调用。角标的上限是99+（实际数目>=100后），建议应用程序判断一下当前数据，不要重复设置大于99的数据。

英文版：

Below is the introduction and ways to support badge feature.

 Please do remember to send us the modified version APP for a further test.

Please feel free to contact me if you have any doubt or questions.

The icon badge feature is like this, when new messages comes in there will be a badaged number on the app icon showing the number of unread messages.



To achieve this we need your team to add some code to APP, and here is how:

When the APP save the unread message numbers, please invoke the follow function

---------------------------------------------------

private int **mCurrentTotalCount**= -1;  
  
/\*\*  
 \* Set the count of the subscript to OPPO Launcher.  
 \*  
 \* @param badgeCount   
 \*/  
public void **setAppBadgeCount**(int badgeCount) {  
    if (mCurrentTotalCount == badgeCount) {  
        return;  
    }  
    mCurrentTotalCount = badgeCount;  
    try {  
        Bundle extras = new Bundle();  
        extras.putInt("app\_badge\_count",badgeCount);  
        getContentResolver().call(  
            Uri.parse("<content://com.android.badge/badge>"),  
                "setAppBadgeCount", null, extras);  
    } catch (Exception e) {  
        Log.e(TAG, "Write unread number FAILED!!! e = " + e);  
    }  
}

And  you can get the Manufacture by  
public String **getManufacturer**() {   
    return SystemProperties.get("ro.product.manufacturer");   
}