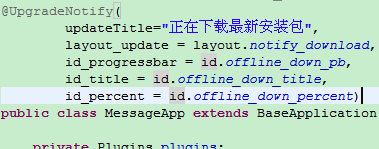
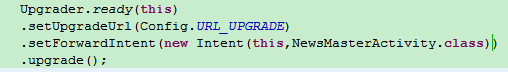
升级SDK文档

1. 使用说明
2. 设置通知栏样式

若程序设置了Application,则可以使用下面方式进行注入:



1. 最小更新配置:



其中UpgradeUrl表示升级接口地址。

forwardIntent表示更新结束后的前进Intent。

最后的upgrade()将检查参数启动更新.

1. 默认的升级策略:

升级包含Ground和Atmo升级。Ground升级为apk升级,atmo为配置升级。默认执行策略如下:

Ground强制升级:弹出强制升级提示框。选择是否确认下载后关闭程序。

Ground建议升级：弹出建议升级提示框。选择是否确认下载后执行forwardIntent跳转。

若在升级过程中发生：请求接口/解析接口失败、执行升级策略失败，将会执行forwardIntent跳转。

升级下载完毕后,程序中会弹出对话框来询问用户是否立即安装更新.

1. 扩展点:
2. UpgradeParser接口:



实现UpgradeParser接口,可以解析升级接口内容,返回升级指示。（强制/建议/不升级）。其中两个Version参数分别是本地的Ground版本以及Atmo版本。

1. UpgradeHandler接口:



实现UpgradeHanlder接口可以改变默认的升级策略。

handle:方法通过result(强制/建议/不升级)的提示,来响应升级。

handleError:在hanlde出现异常或者请求解析接口产生异常时处理.

1. Callback接口:



实现Callback接口可以完成Atmo下载完成的后续动作.(解压,更新配置等)

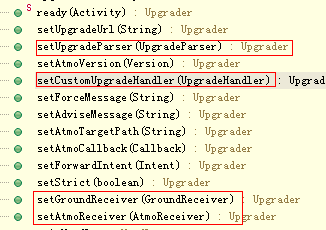
1. 实现自己的Ground/Atmo升级监听器:

只需继承GroundReceiver/AtmoReceiver.重写如下方法即可.



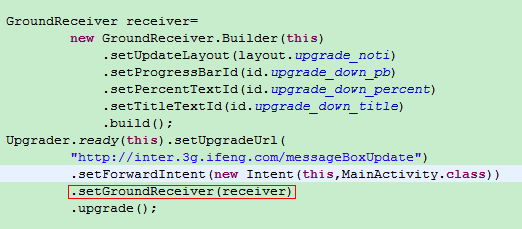
1. 注入扩展点:

通过Upgrader的如下配置实现.



1. FAQ
2. Q:因为某些原因,我不能使用Application来注入升级更新样式,怎么办?

A: 若程序没有Application,则需要手动构造通知UI更新。方法如下:



1. Q:因为某些原因,我没有欢迎界面.我需要在非欢迎界面完成无跳转动作升级,怎么办?

A:使用如下配置即可工作.



1. BUG修复以及扩展:

修复BUG后建议跑一遍test包下的TestCase。可以使用test包中的SplashActivity模拟正式环境进行测试验证.

跑通后使用compile.bat来完成编译sdk工作.编译会去除test包,并会在dist目录生成upgrade\_latest.jar和upgrade\_latest\_source.jar。