

百度云推送 + Android Demo

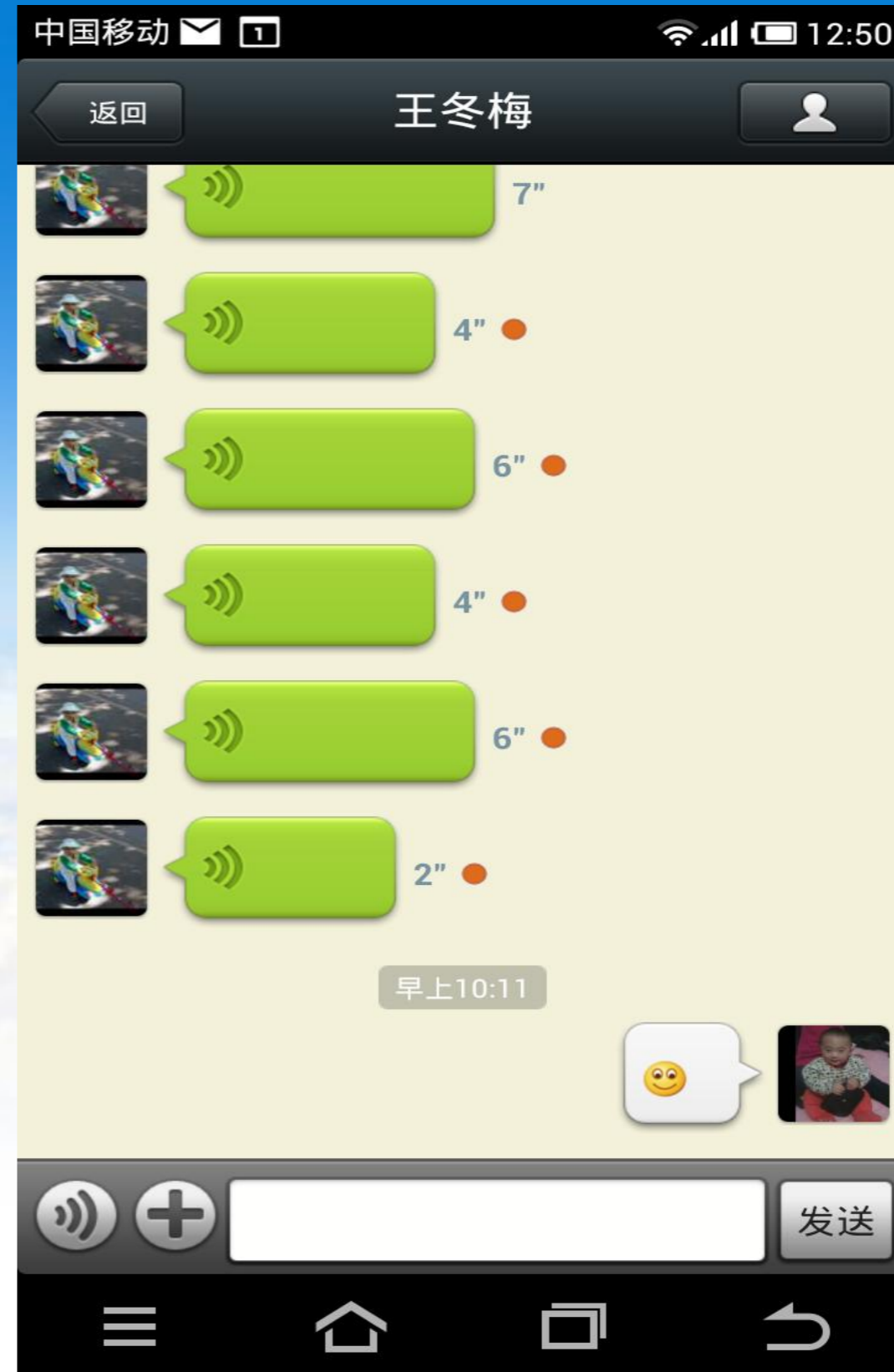


目录

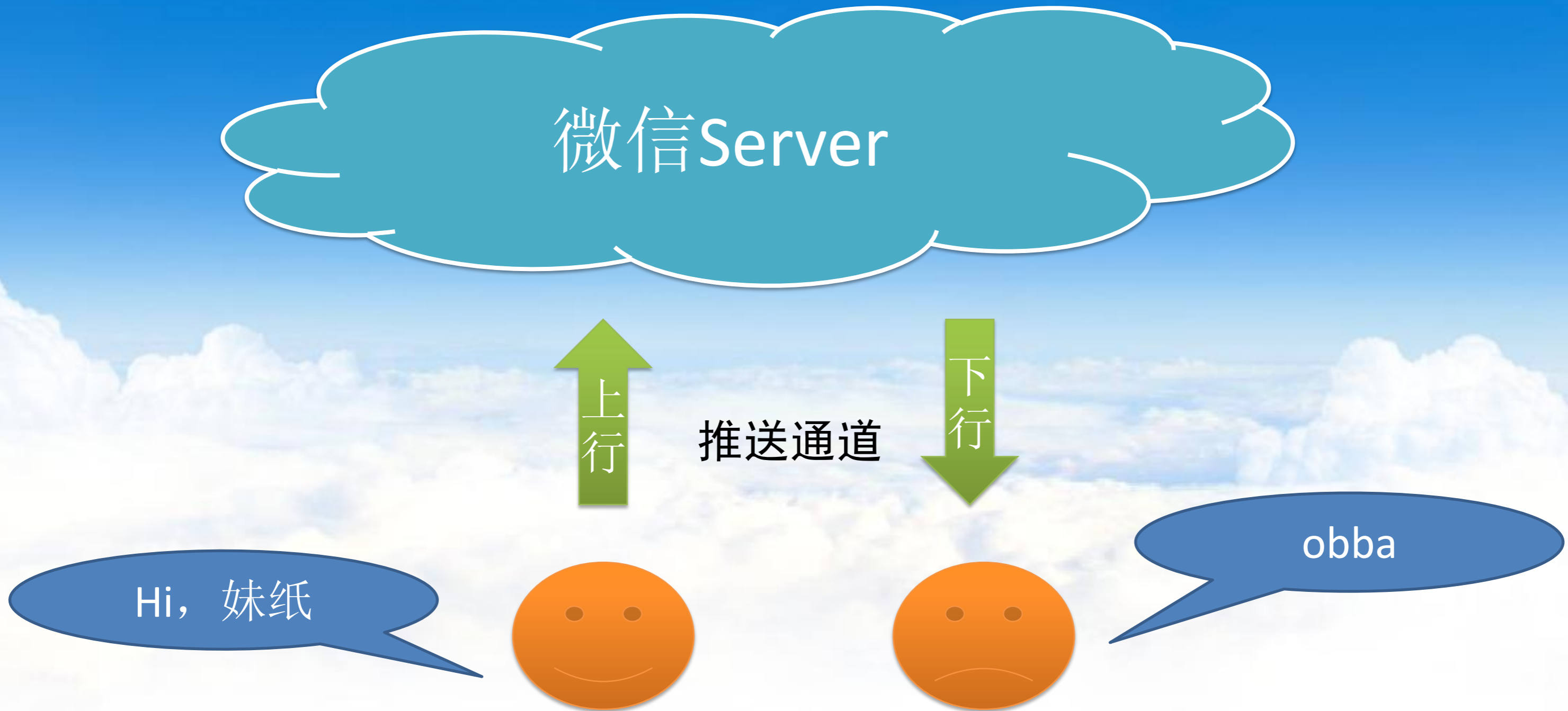
- **什么是云推送**
- **开发者为什么需要云推送**
- **百度云推送的优势**
- **如何使用百度云推送
(Anroid Demo)**



云推送是什么



微信-----信息推送



云推送服务 ---- 信息快递服务



开发者/运营

消息

云推送

消息

用户

云推送的价值

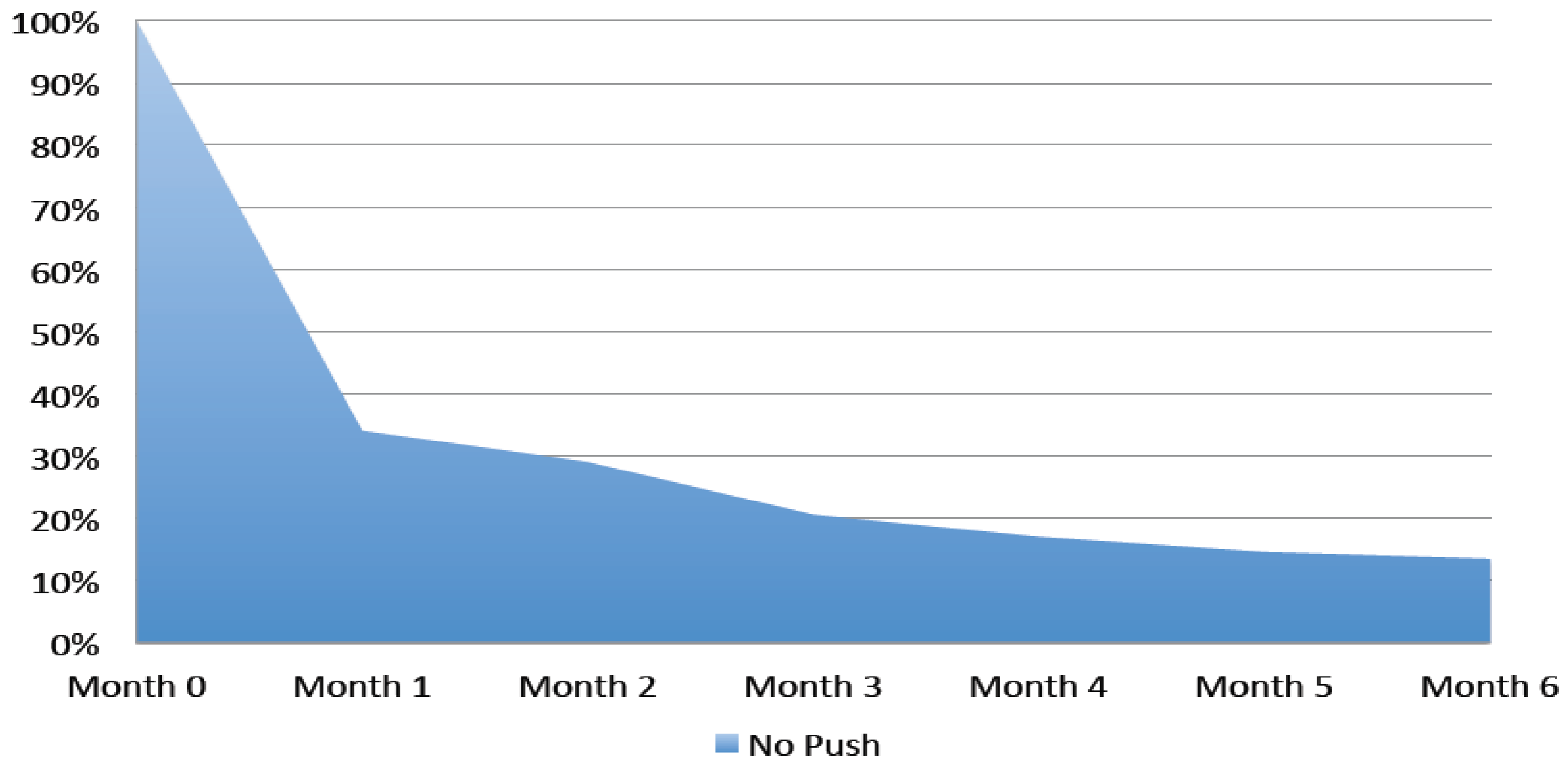
Cost

Real-time

Scalability

Activity

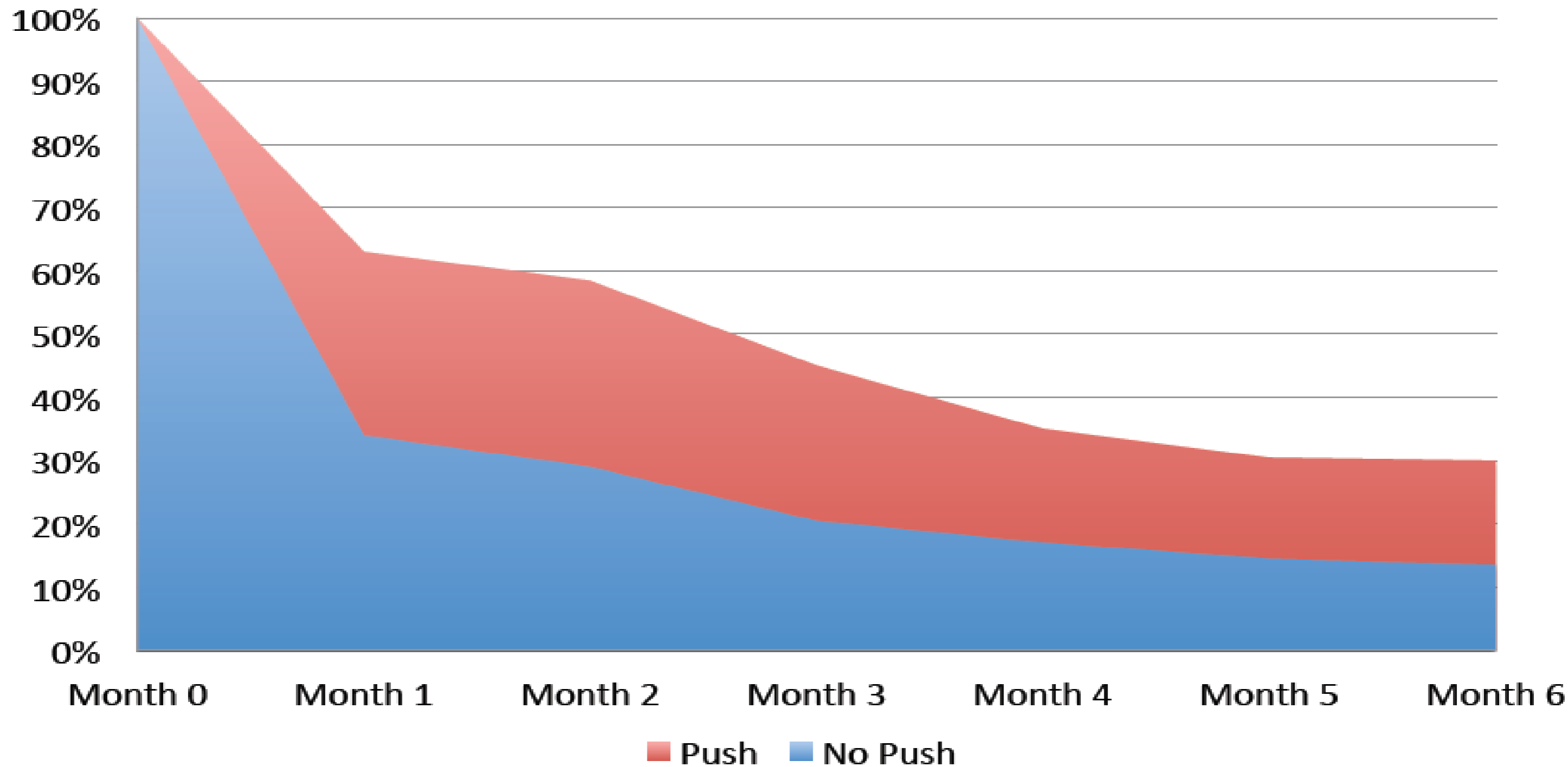
6个月留存率



数据来自Urban



Push通知提高留存率



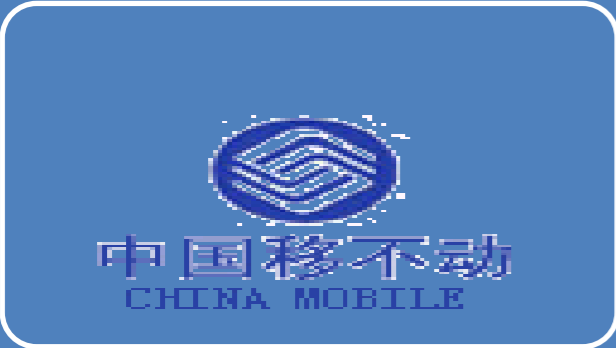
数据来自Urban



消息通道对比



1) HTTP (定时轮询)



2) 短信信道 (WapPush)



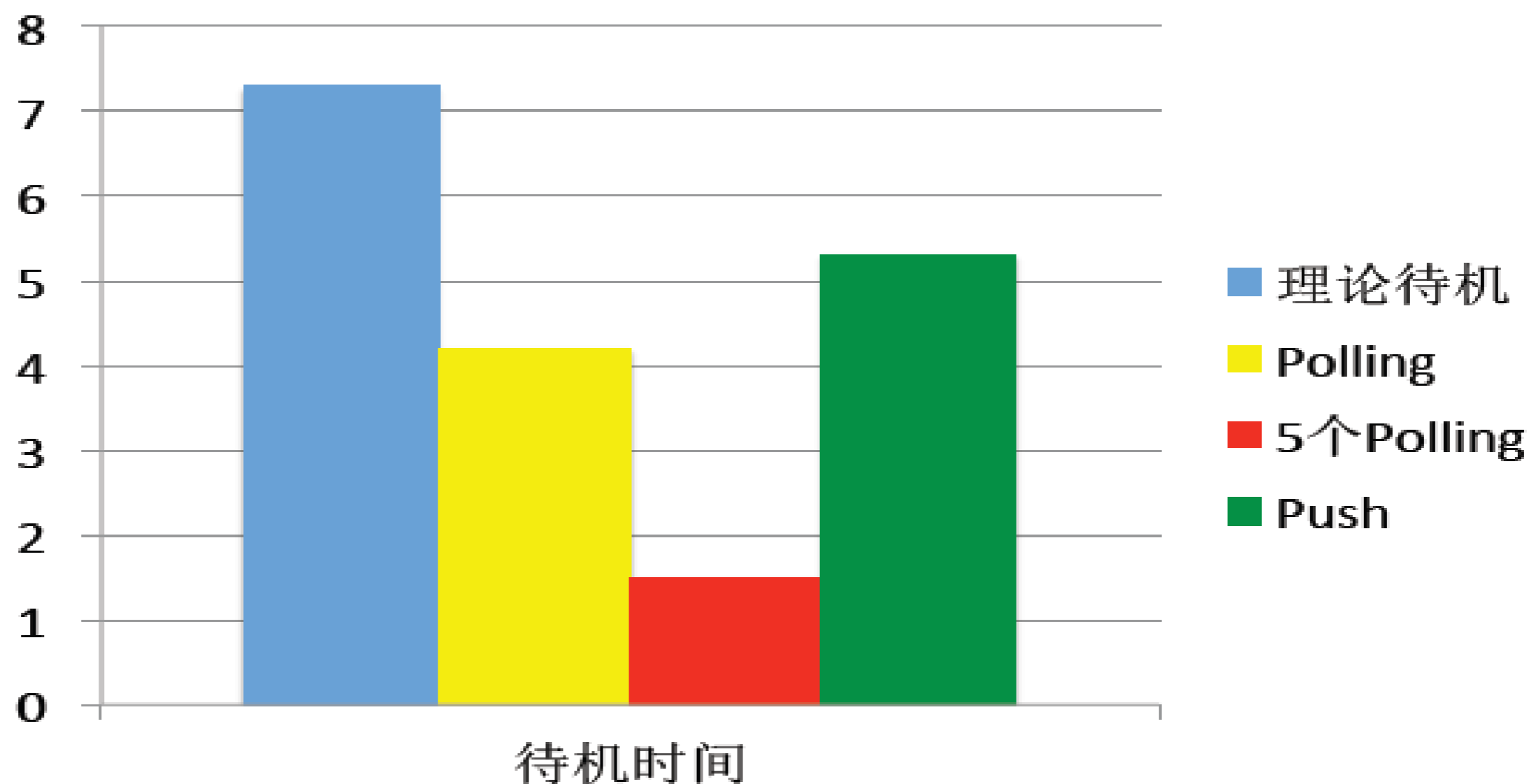
3) TCP (Socket长连接)

Push is better

- 无论有没有数据，都需查询
- 额外的电量和带宽

| | Push | Polling |
|--------|------|---------|
| 延迟 | 低 | 高 |
| 流量 | 低 | 高 |
| 电量 | 低 | 高 |
| 实现的复杂度 | 高 | 低 |

手机耗电情况

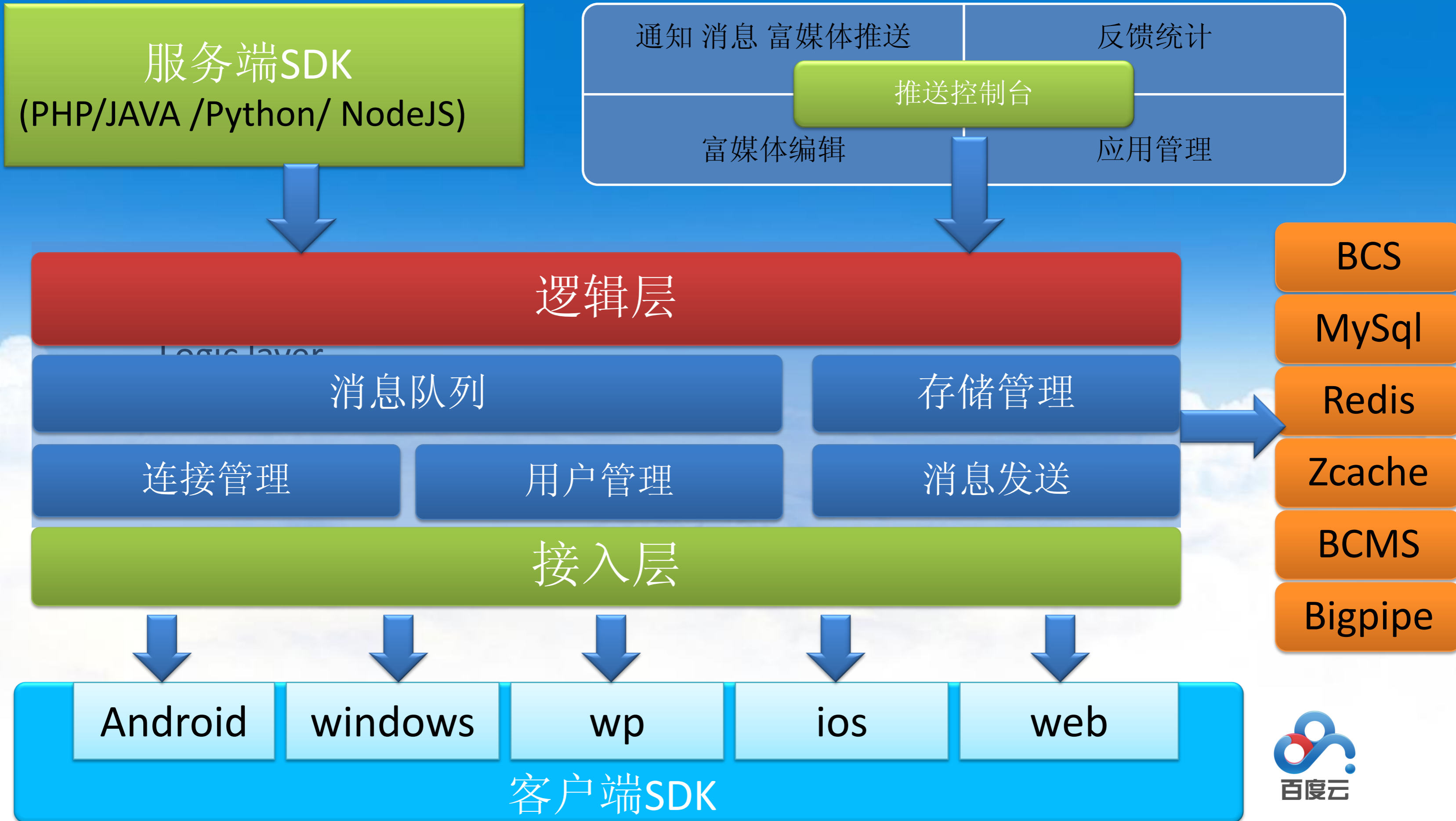


- 电池1400 mAh的手机，底电流8mA

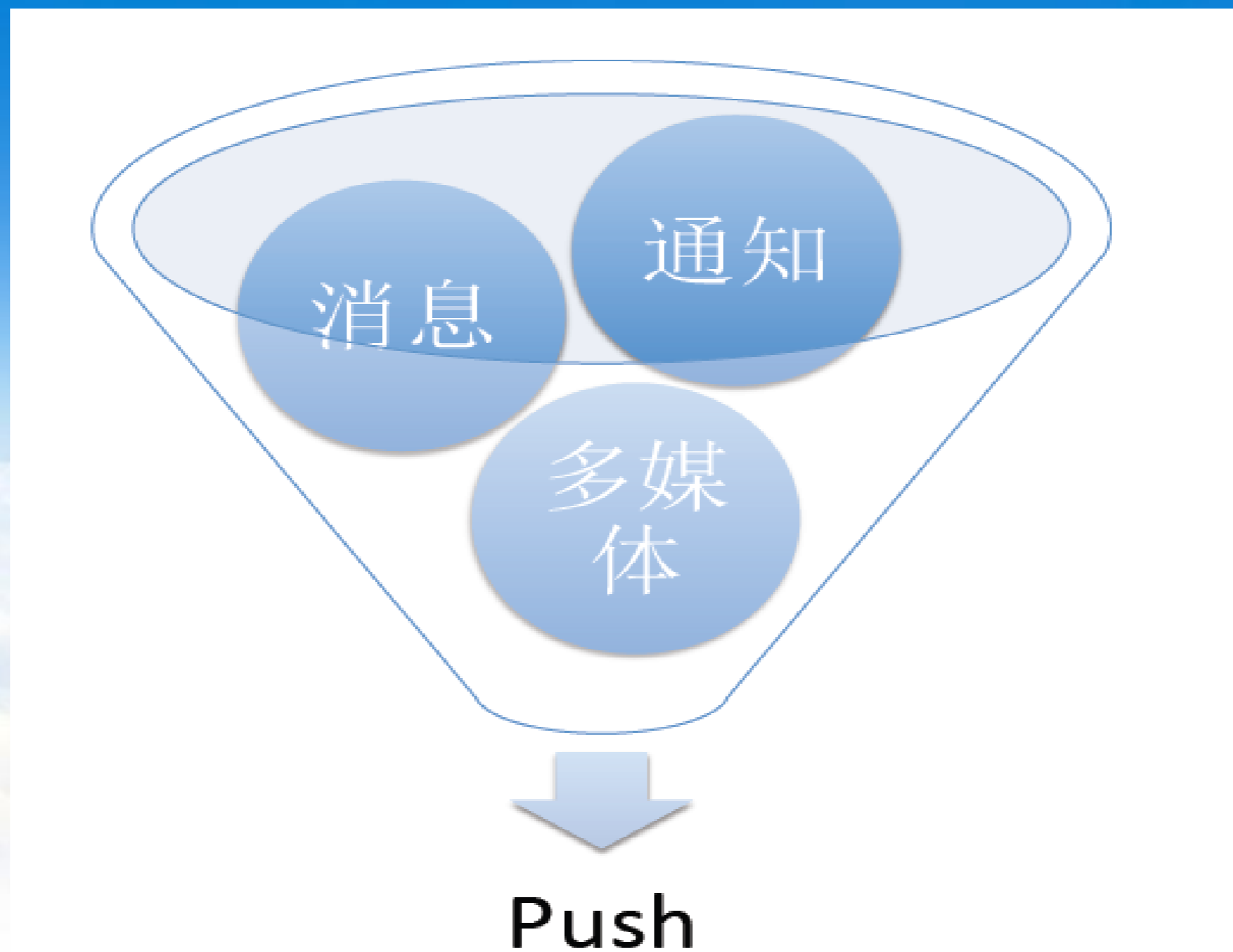
自己实现Push的成本

- 以某 IM 应用为参考，假设：
- – 100万并发
- 开发成本
 - – 6人
 - – 半年
- 运维成本
 - – 服务器：5节点以上
 - – 带宽：5000每月

Push 系统架构



百度云推送 — 能push什么？

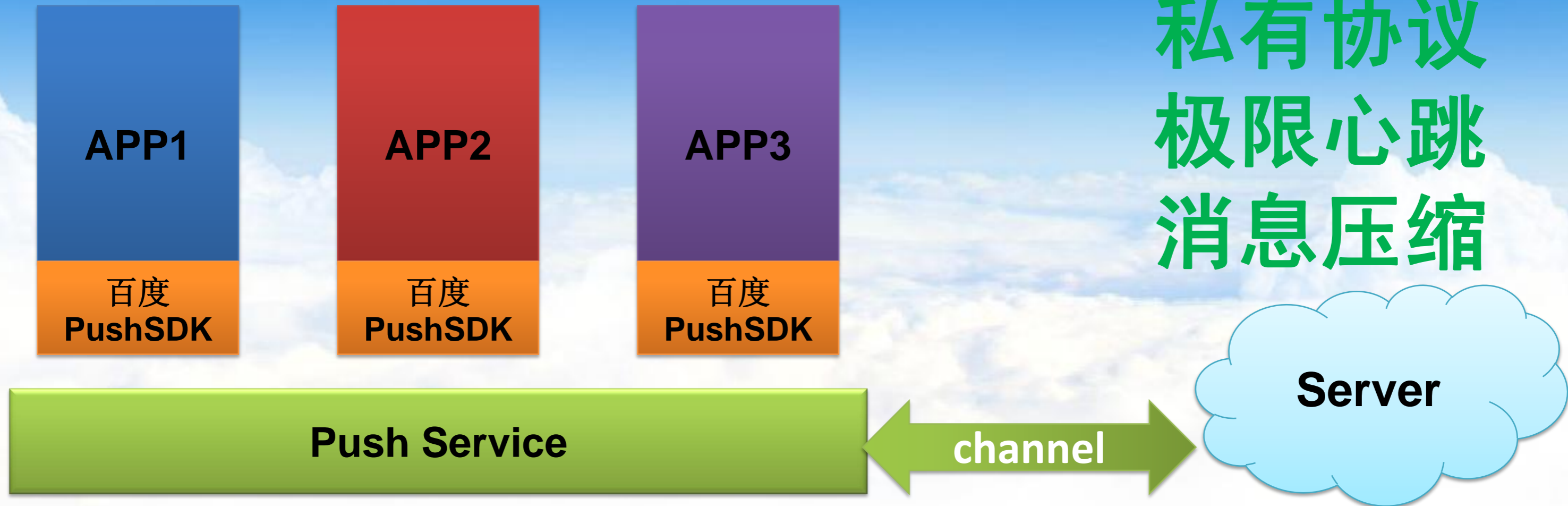


百度云推送 — 推送流程



省电省流量

单服务
单通道
私有协议
极限心跳
消息压缩



百度的优势

技术

强大的服务器资源，千万级消息并发

单通道和极限心跳，更省电省流量

跨平台：Android/iOS/Web/PC/WP7

功能强大的推送后台

丰富的开发资源

市场

百度的技术品牌

已有的亿级用户规模

免费

内部产品使用情况



百度地图



百度



百度音乐



百度新闻



百度云



百度视频



百度知道



百度手机助手



Demo show

Demo

Push (云推送)

如何使用推送后台

The screenshot shows the Baidu Developer Center interface for managing push notifications. The top navigation bar includes '首页', '服务体系', '行业报告', '管理中心', '帮助文档', and '客服支持'. The main content area is titled '推送消息' and includes a sidebar with navigation options like '基本信息', '渠道管理', '云环境(BAE)', '云推送(Push)', and '消息记录'. The '消息记录' section displays a table of sent messages with columns for '发送时间', '消息内容', '发送状态', '用户范围', and '操作'.

| 发送时间 | 消息内容 | 发送状态 | 用户范围 | 操作 |
|---------------------|---------|------|------|-------|
| 2013/04/07 17:49:40 | 0000000 | 已发送 | 所有人 | 重发 删除 |
| 2013/04/07 17:47:38 | 0000000 | 已发送 | 所有人 | 重发 删除 |
| 2013/04/07 17:44:08 | 0000000 | 已发送 | 所有人 | 重发 删除 |
| 2013/04/07 17:42:32 | 0000000 | 已发送 | 所有人 | 重发 删除 |
| 2013/04/07 17:41:28 | 0000000 | 已发送 | 所有人 | 重发 删除 |
| 2013/04/07 14:20:00 | 0000000 | 已发送 | 所有人 | 重发 删除 |

<http://developer.baidu.com/push/list>



如何使用PushSDK

- 前提：获取了access_token
- 集成SDK到你的APP，并定义PushMessageReceiver
- 调用

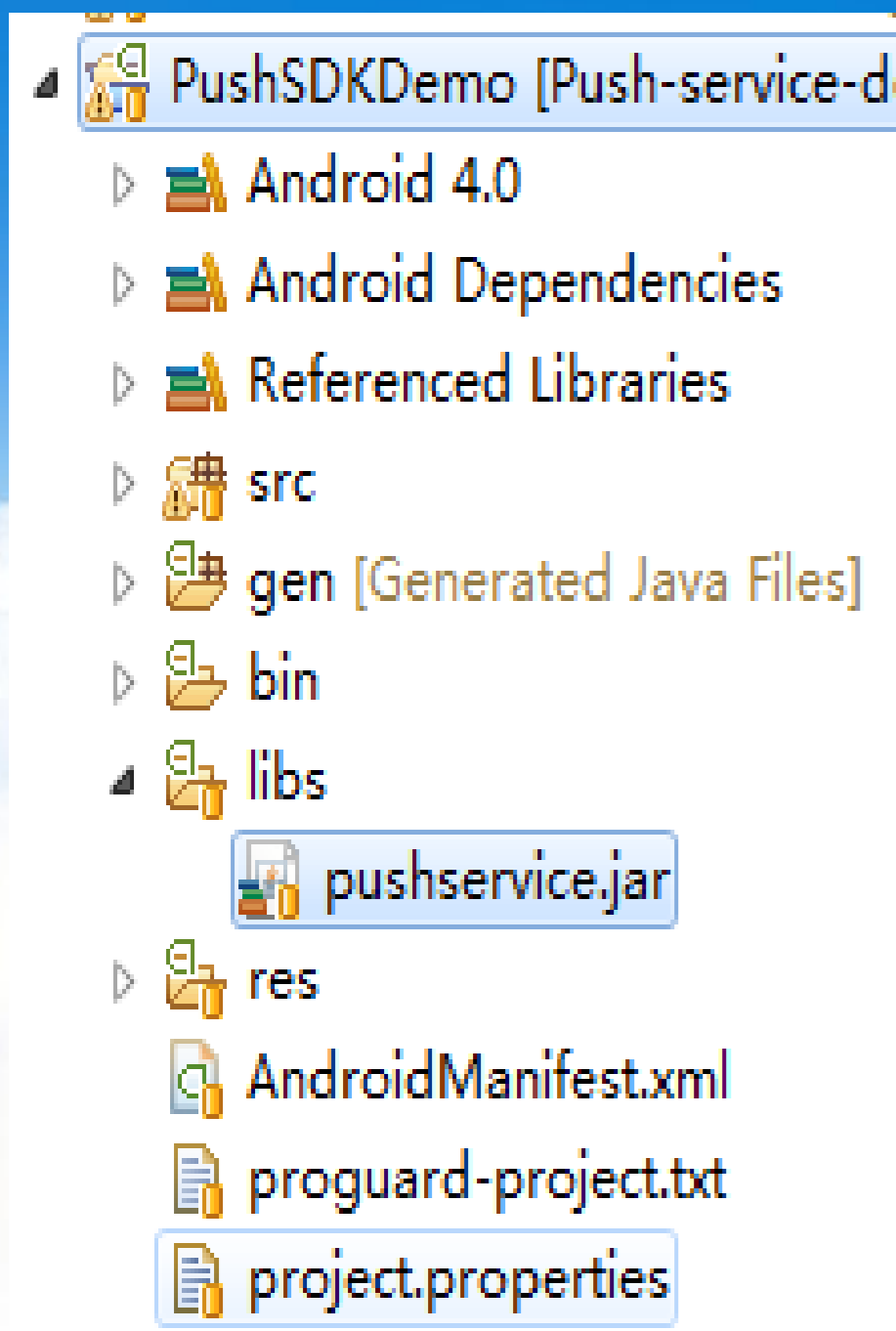
//启动push推送服务并完成app绑定

```
PushManager.startWork(Context,  
String:ACCESS_TOKEN);
```

- 即可在Receiver中接收到push消息
- 更多细节请关注developer.baidu.com(云推送篇)



PushSDK使用示例



```
initButton.setOnClickListener(new View.OnClickListener() {  
    public void onClick(View v) {  
        //启动push推送服务并完成app绑定  
        PushManager.startWork(v.getContext(), DEFAULT_ACCESS_TOKEN);  
    }  
});
```

一行代码, 轻松集成!

```
public class PushMessageReceiver extends BroadcastReceiver {  
    @Override  
    public void onReceive(final Context context, Intent intent) {  
  
        if (intent.getAction().equals(PushConstants.ACTION_MESSAGE)) {  
            //push 消息  
        } else if (intent.getAction().equals(PushConstants.ACTION_RECEIVE)) {  
            //rest api回调  
        }  
    }  
}
```

PushSDK使用示例

- 配置manifest.xml文件

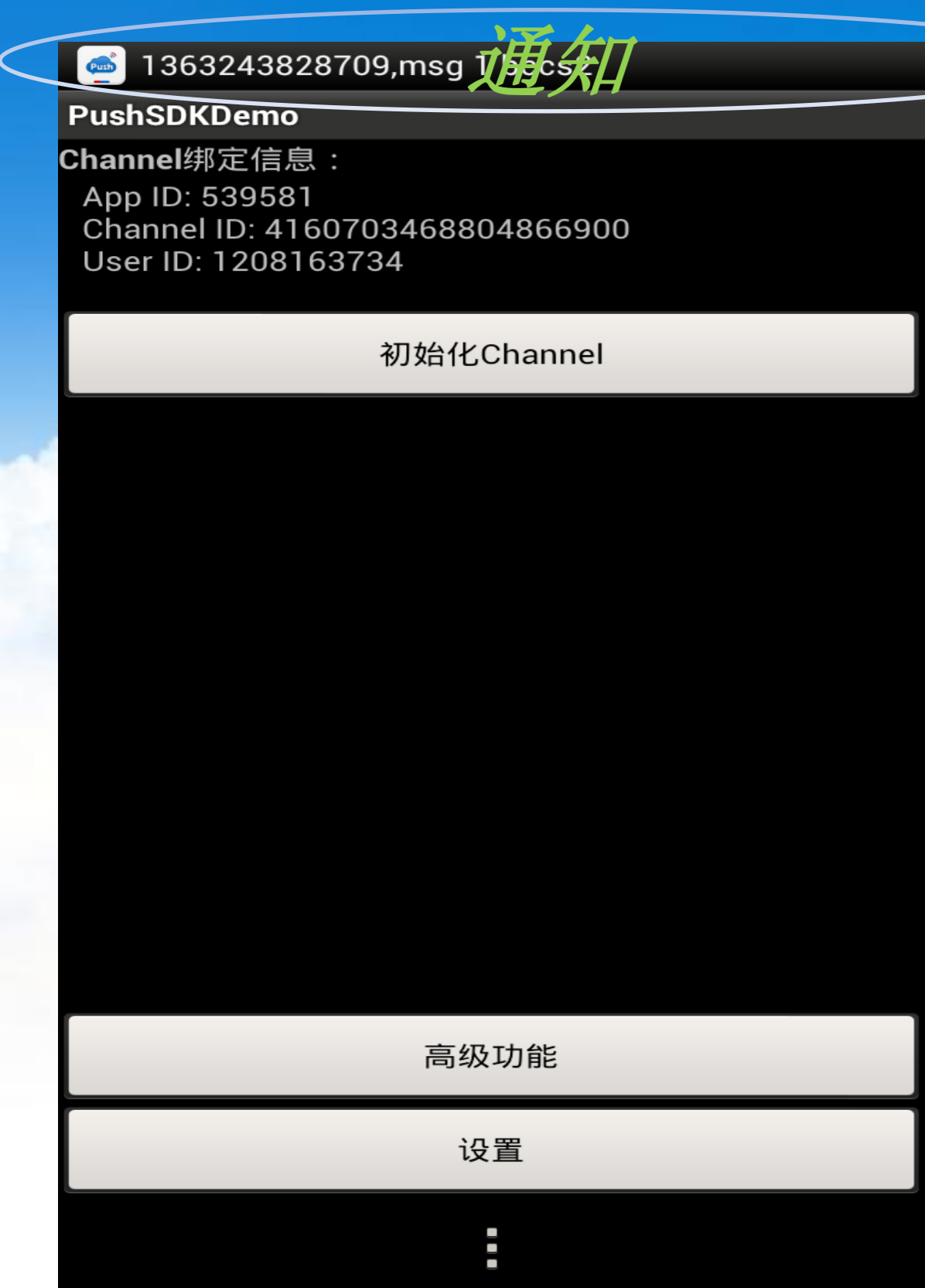
```
<!-- push service -->
<receiver android:name="com.baidu.android.pushservice.PushServiceReceiver"
    android:process=":bdservice_v1">
    <intent-filter>
        <action android:name="android.intent.action.BOOT_COMPLETED" />
        <action android:name="android.net.conn.CONNECTIVITY_CHANGE" />
        <action android:name="com.baidu.android.pushservice.action.notification.SHOW" />
        <action android:name="com.baidu.android.pushservice.action.media.CLICK" />
    </intent-filter>
</receiver>

<receiver android:name="com.baidu.android.pushservice.RegistratonReceiver"
    android:process=":bdservice_v1">
    <intent-filter>
        <action android:name="com.baidu.android.pushservice.action.METHOD" />
        <action android:name="com.baidu.android.pushservice.action.BIND_SYNC" />
    </intent-filter>
    <intent-filter>
        <action android:name="android.intent.action.PACKAGE_REMOVED"/>
        <data android:scheme="package" />
    </intent-filter>
</receiver>

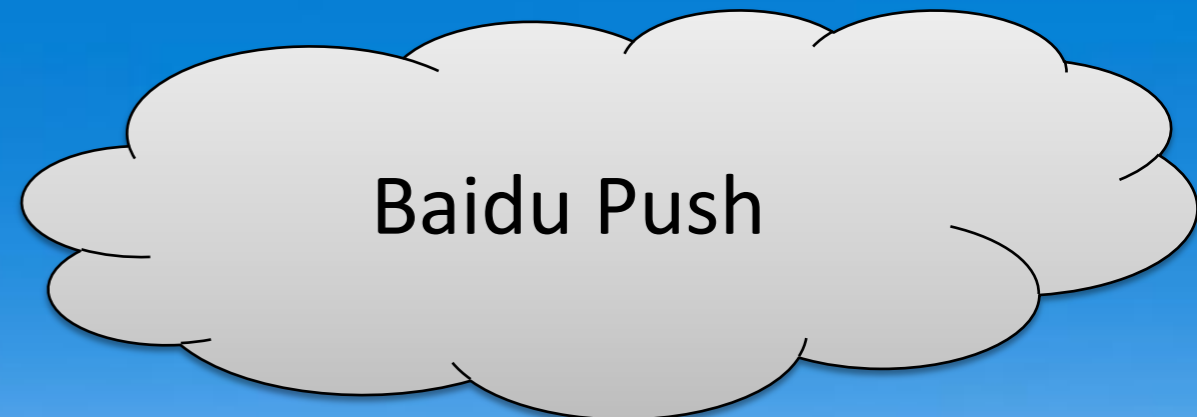
<service
    android:name="com.baidu.android.pushservice.PushService"
    android:exported="true"
    android:process=":bdservice_v1" />

<!-- push service end -->
```

附录- 一个直观的Push应用



← Channel



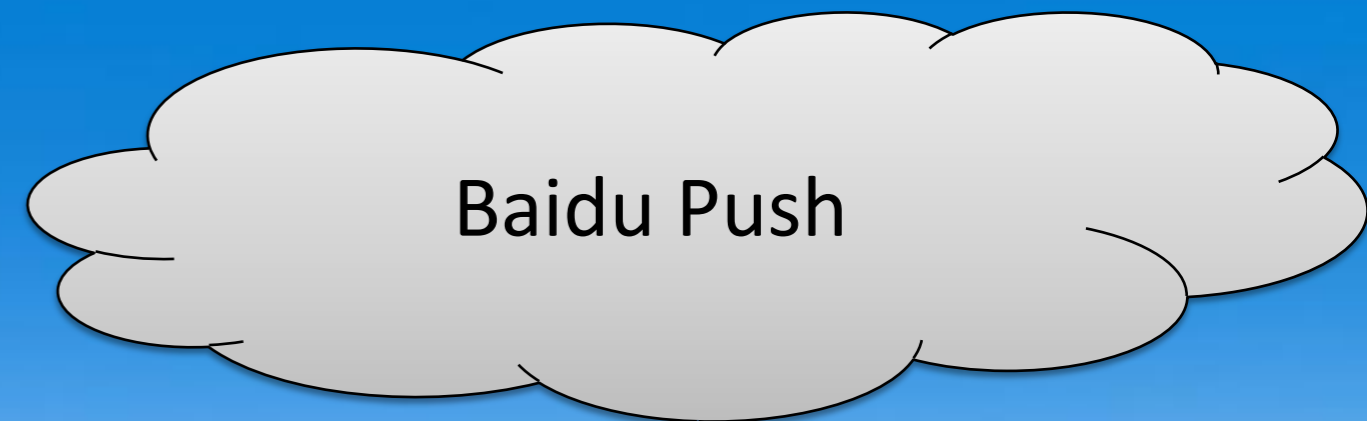
②

```
[work@db1-sumeru-multi-vedio01.vm.baidu.com channel_phpsdk]$ ./test_push_N_msg.zx.sh 1 bccs2
$3 =
Array
(
    [user_type] => 2
    [device_type] => 3
    [appid] => 539581
)
user id = 1208163734; channel id = 4160703468804866900
!!!!!!!!!!!!!!Array
(
    [zhangxin] => 1363243828709,msg 1 bccs2
)
SUCC, test_pushMessage OK!!!!
result: Array
(
    [request_id] => 1672029011
)
```


附录- 一个直观的Push应用



Channel



```
[work@dbl-sumeru-multi-vedio01.vm.baidu.com channel_phpsdk]$ ./test_push_N_msg.zx.sh 1 bccs2
$3 =
Array
(
    [user_type] => 2
    [device_type] => 3
    [appid] => 539581
)
user id = 1208163734; channel id = 4160703468804866900
!!!!!!!!!!!!!!Array
(
    [zhangxin] => 1363243828709,msg 1 bccs2
)
SUCC, test_pushMessage OK!!!!
result: Array
(
    [request_id] => 1672029011
)
```

开发者遇到问题怎么办?

联系我们

欢迎大家和我们联系

[channel-
user@baidu.com](mailto:channel-user@baidu.com)



dev_support@baidu.co
m@百度开发者中心



THANKS!

