



简述移动端IM开发的那些坑：架构设计、通信协议和客户端



男人 · 2 个月前

1、前言

有过移动端开发经历的开发者都深有体会：移动端IM的开发，与传统PC端IM有很大的不同，尤其无线网络的不可靠性、移动端硬件设备资源的有限性等问题，导致一个完整的移动端IM架构设计和实现都充满着大量的挑战。本文将简述移动端IM最重要的架构设计和通信协议选择方面的坑点，希望为IM开发者同行带来些许启发。（本文同步发布于：[简述移动端IM开发的那些坑：架构设计、通信协议和客户端-IM开发/专项技术区](#)）

2、学习交流

- 即时通讯开发交流群：[215891622](#) [推荐]

- 移动端IM开发推荐文章：《[新手入门一篇就够：从零开发移动端IM](#)》

3、概述

移动互联网时代的来临促使我们所有的开发者都要从用户视角出发，基于某一特定场景来创建应用，满足用户需求。通常，在这些应用中，沟通环节都是必不可少的。这就要求创业者不仅要花时间和精力来琢磨用户在某一特定场景下有何痛点需求，琢磨如何解决这一需求，并且可

能还要花费更多的精力和时间来解决产品中“沟通”这一技术节点。

而要解决沟通问题，就需要一套IM系统（而且肯定要支持移动端）。做为IM开发者或即将成为IM开发者的技术人员，IM的价值和重要性不言自明。但从技术实现来说，这并不容易。当然，假设你有100个用户，什么都是容易的，但是假设你有了100万、1000万甚至1亿的用户，再简单的技术节点解决不好，都会成为灾难，何况IM系统（尤其是移动端的IM系统）还是存在许多技术难点和坑点的。

4、有关移动端IM通信协议的坑

其次，我们再看一下IM 协议如何选型。通常IM采取的协议有xmpp、mqtt、protobuf等数据通信私有协议，我们来逐一分析他们的优缺点。

1. XMPP协议：

优点：基于xml协议，容易理解，使用广泛，易于扩展。

缺点：流量大，在移动终端也耗电。交互过程复杂。多被pc时代的产品使用，不适合移动时代的IM产品，即使我们基于xmpp进行改进，简化握手过程，改进文件传输机制，但是它的基因决定了如何改进，他都不适合移动互联网时代的IM产品。就像凤姐无论怎么整容，也变成不了高圆圆一样。

2. MQTT协议：

优点：适配多平台。

缺点：协议简单，但是需要自己扩展好友，群组等功能。

3. 私有协议：

优点：随心所欲，自己定义，流量小。

缺点：工作量巨大，扩展性差，需要考虑全面。

4. Protobuf协议：

优点：非常小、非常快、非常简单，一条消息数据用Protobuf序列化后的大小是JSON的1/10、XML格式的1/20、是二进制序列化的1/10。

缺点：不能表示复杂的数据结构，但是对于IM来讲，已经足够。强烈推荐此协议。

补充1：强烈建议使用Protobuf，理由如下

灵活、高效：灵活（方便接口更新）、高效（效率经过google的优化，传输效率比普通的XML等高很多）；

易于使用：开发人员通过按照一定的语法定义结构化的消息格式，然后送给命令行工具，工具将自动生成相关的类，可以支持java、c++、python等语言环境。通过将这些类包含在项目中，可以很轻松的调用相关方法来完成业务消息的序列化与反序列化工作。

语言支持：原生支持c++、java、python等多达10余种语言。

补充2：Protobuf主要适用于

需要和其它系统做消息交换的，对消息大小很敏感的。那么protobuf适合了，它语言无关，消息空间相对xml和json等节省很多。

小数据的场合。如果你是大数据，用它并不适合。

项目语言是c++、java、python等，因为它们可以使用google的源生类库，序列化和反序列化的效率非常高。其它的语言需要第三方或者自己写，序列化和反序列化的效率不保证。

总体而言，Protobuf还是非常好用的，被很多开源系统用于数据通信的工具，在google也是核心的基础库。

（更多文章：《[强烈建议将Protobuf作为你的即时通讯应用数据传输格式](#)》、《[如何选择即时通讯应用的数据传输格式](#)》、《[理论联系实际：一套典型的IM通信协议设计详解](#)》）

5、移动端IM客户端的坑

最后，我们再来了解一下移动端有哪些难点需要解决。

1. 流量：

采取哪种协议、图片缩略图、附件的压缩三点决定了流量的大小。

2. 耗电：

(1)流量越小，耗电越低。(2)心跳策略，减少心跳次数，耗电量就会降低。

3. 心跳时长：

wifi，2G，3G，4G，移动、电信、联通,不同网络，不同运行商，NAT失效时间不一样，因此心跳的时间也就不一样。

4. 网络连接：

cmnet和cmwap下连接处理机制。

5. 网络不稳定：

移动端最大的特点就是网络不稳定，在不稳定的网络状态下，如何保证消息以最快的速度到达？如何避免重联风暴？这些既需要从整体架构考虑，也需要在移动端采取巧妙的策略加以避免。

（更多文章：[移动端IM开发需要面对的技术问题](#)）

6、移动端IM架构设计的坑

首先，来看移动端IM架构设计需要考虑的问题。

1. 连接器的设计：

连接器主要用来管理客户端的长连接。目前最好的连接器单台8G8核的服务器可以做到70万—100万的连接，而某些开发者只能做到4000左右的连接，相差好几个数量级。这里的奥妙在哪里呢？

2. 中间件的设计：

是否采用通讯中间件？通讯中间件的好处有哪些？如果不采用中间件，连接器和逻辑服务器的连接关系如何管理呢？

3. 逻辑服务器：

逻辑服务器通常简单一点，主要是根据业务逻辑进行最小粒度的划分即可。但是即便如此，还是有很多的开发把看似相关实则不相关的逻辑放在一起，如把鉴权和message服务器放在一起。

4. 状态服务器：

状态服务器主要管理用户在线、离线的相关状态，需要采取中心节点的方案，否则状态就会不同步。这里主要需要考虑状态服务器所对应的数据存储机制，如何进行写操作，如何进行读操作？以便最大的提高状态服务器的处理能力和响应速度。

5. 数据库的设计：

数据库的设计是最难的，也是做大的瓶颈。因为无论对于sql（关系型）数据库还是nosql（非关系型）数据库，都有读写处理的极限，那就需要考虑数据库如何分区（根据什么原则、什么操作、哪些用户访问哪个节点上的数据库）。同时又需要考虑每个原子操作（如登陆）需要读哪些库，写哪些库。只有这些指标明确了，你才能在假设有100万并发用户，100万条并发消息的情况下，准确评估服务端需要多少台服务器，如何部署。

6. 其他：

还有设备推送的处理，何种机制能够保证不丢消息，离线消息如何处理，等等。这些都是必备而又非常复杂的功能点和技术要求，都需要采取正确的架构和策略才能实现。

（更多文章：[IM架构篇 - 淘帖](#)）

7、结语

以上难点和坑点草草记录下来也不过千把字，但是真正要解决这些问题并达到生产应用标准，却要不知道花费多少日日夜夜、敲下多少行代码，恐怕也只有真正做过IM的开发者才有比较深刻的体会。（本文同步发布于：[简述移动端IM开发的那些坑：架构设计、通信协议和客户端-IM开发/专项技术区](#)）

附录：更多IM技术文章

[1] 网络编程基础资料：

《[TCP/IP详解 -第11章·UDP：用户数据报协议](#)》

《[TCP/IP详解 -第17章·TCP：传输控制协议](#)》

《[TCP/IP详解 -第18章·TCP连接的建立与终止](#)》

《TCP/IP详解 -第21章·TCP的超时与重传》

《理论经典：TCP协议的3次握手与4次挥手过程详解》

《理论联系实际：Wireshark抓包分析TCP 3次握手、4次挥手过程》

《计算机网络通讯协议关系图（中文珍藏版）》

《NAT详解：基本原理、穿越技术(P2P打洞)、端口老化等》

《UDP中一个包的大小最大能多大？》

《Java新一代网络编程模型AIO原理及Linux系统AIO介绍》

《NIO框架入门(三)：iOS与MINA2、Netty4的跨平台UDP双向通信实战》

《NIO框架入门(四)：Android与MINA2、Netty4的跨平台UDP双向通信实战》

>>更多同类文章

[2] 有关IM/推送的通信格式、协议的选择：

《为什么QQ用的是UDP协议而不是TCP协议？》

《移动端即时通讯协议选择：UDP还是TCP？》

《如何选择即时通讯应用的数据传输格式》

《强烈建议将Protobuf作为你的即时通讯应用数据传输格式》

《移动端IM开发需要面对的技术问题（含通信协议选择）》

《简述移动端IM开发的那些坑：架构设计、通信协议和客户端》

《理论联系实际：一套典型的IM通信协议设计详解》

《58到家实时消息系统的协议设计等技术实践分享》

>>更多同类文章

[3] 有关IM/推送的心跳保活处理：

《Android进程保活详解：一篇文章解决你的所有疑问》

《Android端消息推送总结：实现原理、心跳保活、遇到的问题等》

《为何基于TCP协议的移动端IM仍然需要心跳保活机制？》

《微信团队原创分享：Android版微信后台保活实战分享(进程保活篇)》

《微信团队原创分享：Android版微信后台保活实战分享(网络保活篇)》

《移动端IM实践：实现Android版微信的智能心跳机制》

《移动端IM实践：WhatsApp、Line、微信的心跳策略分析》

>>更多同类文章

[4] 有关WEB端即时通讯开发：

《新手入门贴：史上最全Web端即时通讯技术原理详解》

《Web端即时通讯技术盘点：短轮询、Comet、Websocket、SSE》

《SSE技术详解：一种全新的HTML5服务器推送事件技术》

《Comet技术详解：基于HTTP长连接的Web端实时通信技术》

《WebSocket详解（一）：初步认识WebSocket技术》

《socket.io实现消息推送的一点实践及思路》

>>更多同类文章

[5] 有关IM架构设计：

《浅谈IM系统的架构设计》

《简述移动端IM开发的那些坑：架构设计、通信协议和客户端》

《一套原创分布式即时通讯(IM)系统理论架构方案》

《从零到卓越：京东客服即时通讯系统的技术架构演进历程》

《蘑菇街即时通讯/IM服务器开发之架构选择》

《腾讯QQ1.4亿在线用户的技术挑战和架构演进之路PPT》

《微信技术总监谈架构：微信之道——大道至简(演讲全文)》

《如何解读《微信技术总监谈架构：微信之道——大道至简》》

《快速裂变：见证微信强大后台架构从0到1的演进历程（一）》

《17年的实践：腾讯海量产品的技术方法论》

>>更多同类文章

[6] 有关IM安全的文章：

《即时通讯安全篇（一）：正确地理解和使用Android端加密算法》

《即时通讯安全篇（二）：探讨组合加密算法在IM中的应用》

《即时通讯安全篇（三）：常用加解密算法与通讯安全讲解》

《即时通讯安全篇（四）：实例分析Android中密钥硬编码的风险》

《传输层安全协议SSL/TLS的Java平台实现简介和Demo演示》

《理论联系实际：一套典型的IM通信协议设计详解（含安全层设计）》

《微信新一代通信安全解决方案：基于TLS1.3的MMTLS详解》

《来自阿里OpenIM：打造安全可靠即时通讯服务的技术实践分享》

>>更多同类文章

[7] 有关实时音视频开发：

《即时通讯音视频开发（一）：视频编解码之理论概述》

《即时通讯音视频开发（二）：视频编解码之数字视频介绍》

《即时通讯音视频开发（三）：视频编解码之编码基础》

《即时通讯音视频开发（四）：视频编解码之预测技术介绍》

《即时通讯音视频开发（五）：认识主流视频编码技术H.264》

《即时通讯音视频开发（六）：如何开始音频编解码技术的学习》

《即时通讯音视频开发（七）：音频基础及编码原理入门》

《即时通讯音视频开发（八）：常见的实时语音通讯编码标准》

《即时通讯音视频开发（九）：实时语音通讯的回音及回音消除概述》

《即时通讯音视频开发（十）：实时语音通讯的回音消除技术详解》

《即时通讯音视频开发（十一）：实时语音通讯丢包补偿技术详解》

《即时通讯音视频开发（十二）：多人实时音视频聊天架构探讨》

《即时通讯音视频开发（十三）：实时视频编码H.264的特点与优势》

《即时通讯音视频开发（十四）：实时音视频数据传输协议介绍》

《即时通讯音视频开发（十五）：聊聊P2P与实时音视频的应用情况》

《即时通讯音视频开发（十六）：移动端实时音视频开发的几个建议》

《即时通讯音视频开发（十七）：视频编码H.264、V8的前世今生》

《简述开源实时音视频技术WebRTC的优缺点》

《良心分享：WebRTC 零基础开发者教程（中文）》

>>更多同类文章

[8] IM开发综合文章：

《移动端IM开发需要面对的技术问题》

《开发IM是自己设计协议用字节流好还是字符流好？》

《请问有人知道语音留言聊天的主流实现方式吗？》

《IM系统中如何保证消息的可靠投递（即QoS机制）》

《谈谈移动端 IM 开发中登录请求的优化》

《完全自己开发的IM该如何设计“失败重试”机制？》

《微信对网络影响的技术试验及分析（论文全文）》

《即时通讯系统的原理、技术和应用（技术论文）》

《开源IM工程“蘑菇街TeamTalk”的现状：一场有始无终的开源秀》

>>更多同类文章

[9] 开源移动端IM技术框架资料：

《开源移动端IM技术框架MobileIMSDK：快速入门》

《开源移动端IM技术框架MobileIMSDK：常见问题解答》

《开源移动端IM技术框架MobileIMSDK：压力测试报告》

>>更多同类文章

[10] 有关推送技术的文章：



首发于
即时通讯(IM)技术开发

关注专栏

写文章

登录

《Android端消息推送总结：实现原理、心跳保活、遇到的问题等》

《扫盲贴：认识MQTT通信协议》

《一个基于MQTT通信协议的完整Android推送Demo》

《求教android消息推送：GCM、XMPP、MQTT三种方案的优劣》

《移动端实时消息推送技术浅析》

《扫盲贴：浅谈iOS和Android后台实时消息推送的原理和区别》

《绝对干货：基于Netty实现海量接入的推送服务技术要点》

《移动端IM实践：谷歌消息推送服务(GCM)研究（来自微信）》

《为何微信、QQ这样的IM工具不使用GCM服务推送消息？》

>>更多同类文章

[11] 更多即时通讯技术好文分类：

淘帖 - 即时通讯网

作者：Jack Jiang (点击作者姓名进入Github)

出处：JackJiang的空间

交流：欢迎加入即时通讯开发交流群215891622

讨论：即时通讯网 - 即时通讯开发者社区!

Jack Jiang同时是【原创Java Swing外观工程BeautyEye】和【轻量级移动端即时通讯框架MobileIMSDK】的作者，可前往下载交流。

「真诚赞赏，手留余香」

赞赏

还没有人赞赏，快来当第一个赞赏的人吧！

即时通讯 (IM)

👍 3

🔗 分享 🚩 举报



文章被以下专栏收录



还没有评论

写下你的评论

推荐阅读

为什么说基于TCP的移动端IM仍然需要心跳保活？

1、前言很多人认为，TCP协议自身先天就有KeepAlive机制，为何基于它的通讯链接，仍然需要在应用层实现额外的心跳保活？本文将从移动端IM实践... [查看全文](#) >

男人 · 1 个月前

发表于 即时通讯(IM)技术开发

移动端IM系统的协议选型：UDP还是TCP？

1、前言对于有过网络编程经验的开发者来说，使用何种数据传输层协议来实现数据的通信，是个非常基础的问题，它涉及到你的第一行代码该如何编... [查看全文](#) >

男人 · 2 个月前

发表于 即时通讯(IM)技术开发

美国大学生如何看待12大社交网络？

编者按：本文来自微信公众号“峰瑞资本”（微信号：freesvc），授权知乎发布，转载请联系原作者。这可能是关于社交网络最好的一篇分析。这位 ... [查看全文](#) >

峰小瑞 · 13 天前 · 编辑精选

发表于 FREE Thinking

Live 内容回顾 —— 商业新视角：社会创业与社会企业

-上次 Live 结束之后我又出门了，所以内容回顾再次拖了很久，请各位见谅，由于出门在外，回家之后可能还会对这次回顾有些补充和修改。内容回... [查看全文](#) >

徐博聞 · 1 个月前 · 编辑精选

发表于 不又斋：公益专版