微博众筹的架构设计

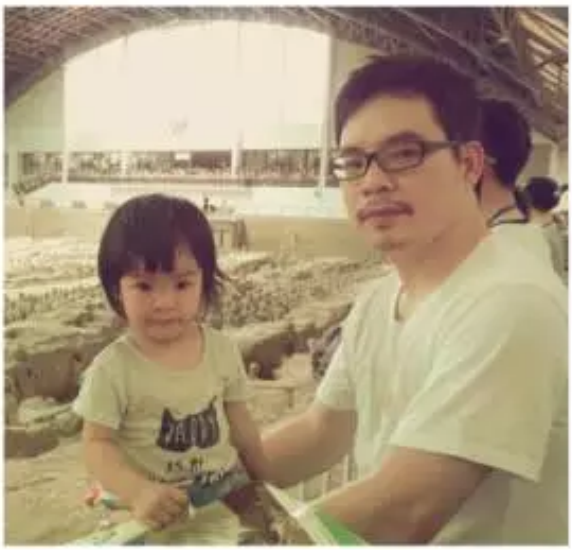
原创：陈杰 高可用架构 2016-06-22

导读：我们每一天都能感受到互联网金融的成长和进步，在 6 月 19 日，微博商业产品部联合天弘基金（余额宝），小米支付、还有创业公司付钱拉等金融技术团队策划了首届互联网金融系统沙龙，围绕在互联网金融过程中碰到核心技术架构、系统安全、数据一致性、业务开发模式等与业界进行分享及交流。本文是陈杰在本次沙龙的演讲，授权高可用架构首发。

陈杰，新浪微博资深系统架构师，毕业于清华大学化学系，从 2004 年开始做过测试、GIS 二次开发、游戏开发、搞过创业，2011 年底到新浪开始接触互联网，喜欢做程序员喜欢做的事。

互联网金融已经影响生活方方面面，我们可以拿着手机，不用银行卡、不用现金来体验新时代的衣食住行。互联网金融现在已经成为互联网巨头争相布局的一个领域，BAT、微博、小米都已经在发力金融。今年年初京东金融融资 66.5 亿人民币，4 月份微博金融发布微博众筹第一款产品。就在本月小米刚刚高调宣布进军民营银行，以获得中国银监会复批。

我今天讲的主题是微博众筹架构，我叫陈杰，2004 年清华化学系毕业，之后不小心进入 IT 这行。最开始做的是测试，后来慢慢接触一些 VB，做了两年的游戏开发，再后来跟朋友去创业，2011 年来到微博开始做互联网相关的一些事情。



```
define('NAME', '陈杰');
define('SCHOOL', '清华2004年化学系毕业');

$works = [
    ['2004', 'VB, C# .Net', 'GIS', '某公司'],
    ['2008', 'C# .Net', '游戏', '人人'],
    ['2010', 'C# .Net', '影院售票', '创业'],
    ['2011', 'PHP', '*', '微博'],
];

$tags = ['MTB' , '奶爸'];
```

本人有两个标签，第一我喜欢骑自行车，在创业之前那时候比较年轻，基本上每周会去两天香山骑车，另外一个标签是奶爸。

从业务产品到金融产品

今天我主要从三个变化去介绍微博众筹的架构，前面两个主要是介绍一下行业的背景。

从技术人员对产品的要求或是对技术上的深度，把它分成三个部分：第一是业务产品、第二是服务产品，第三就是金融产品，我今天讲的是围绕着三个点。

业务产品有什么要求？互联网时代如果慢了，可能就没有机会了。所以我们要做到快速迭代，**业务产品的要求就是快速迭代**。所以技术要能很简单的去实现，才能跟得上产品的步伐。

服务产品角度，有可用性、性能、扩展性等标签，这是互联网服务产品必须具备的一些特性。

金融产品也是我今天讲的重点，我会突出它对数据安全方面的要求，对数据一致性方面的要求，这是两方面对数据相关的比较重要的地方。

数据安全可以参看下图从一个网站拿到的数据，80% 的数据安全问题都是来自于内部人员的一些误操作或者是恶意操作。另外几种常见的情况，比如黑客入侵或者程序员写的一些 BUG，还有硬件上面一些稳定性的问题带来数据安全上的一些漏洞。

金融方面还有一个非常关注的点就是**数据一致性**。金融的数据可能有两种类型，比如可以**分为增值服务类和理财回报类**，像微博里面有微博红包，付费打赏，这都增值服务类。它可能对一致性的要求没有众筹或者微博支付那么高，但追求是最终一致，追求是服务上的性能，能达到数据最终一致就可以。

回顾一下刚才说的三个级别，主要从技术要求从低到高的一个说明。

- 业务产品需要简单实现快速迭代；
- 服务角度如果要打赢别的产品，必须要提高性能、可用性、伸缩性，在应对一些互联网访问峰值事件的时候能从容度过。
- 金融产品，今天主要讲的是安全和数据一致性。

从传统金融到互联网金融

从传统金融到现在的互联网金融环境有很多变化，互联网带来的好处会冲击哪些方面？我们怎么去解决互联网带来这些问题，或者是怎么能把互联网这些优势用到传统的金融行业上？

下面主要分四个方面去描述变化或者是机遇。

第一是服务规模和性质。银行周末通常都是不上班，所以它偶尔会系统维护不能用。但是对于互联网产品，这是不可以的。如果服务有这些问题，会丢掉很多用户。另外很常见的比如 12306 这样高并发系统，或者秒杀类似的活动会很多在互联网金融里面。但是对于传统的互联网，可能就用排队，或者打电话预约等方式来处理。

第二是业务模型，传统银行很多年前就按照很简单这种瀑布开发，比较保守方式去进行业务产品开发。但对于互联网，刚才一直强调如果慢或者没有创新，那可能就活不下去了。这是两个完全不一样的业务开发模式。

第三个是安全。比较直观就是网络环境，一个是比较私有，一个是公开的，这上面会带来安全上的很多问题和隐患。

第四个是成本，这是非常直观，也是老板们非常关注的一个问题。现在大家都提倡去 IOE 化，IOE 是 IBM 大型服务器、Oracle 企业级数据库，EMC 存储设备。我们未来都用云，或者是免费的软件，对于软硬件的成本有大幅度的降低。

这是刚才讲的互联网接入对传统金融带来一些机遇或者是问题，我们总结两部分。

首先是成本变低，互联网一些特性如高可用、高并发等，能降低企业成本。

其次带来的问题可能就是比较明显就是风险变高了，主要有三个方面。第一个是它的安全隐患增多了，因为它的公开，因为它的快速迭代，它的稳定性也会有所降低。另外一个比较严重的是它的影响范围会很大，比如数据泄露带来的影响可能是全部灾难性的。影响控制的力度也没有传统金融那么强的手段，我们讲的它的带来的问题就是高风险。

从新手众筹到网红众筹

讲了前面两个变化，主要是目的是介绍后面的微博众筹架构怎么去解决刚才带来高风险的问题，怎么提升刚才带来的那些好处?

标题是从新手众筹到网红众筹这个概念什么意思？我从三个点上说明。

一、经验。我们的开发以前都习惯于开发业务或者服务类型的产品，对于金融类型也有所接触，但是没有积累那么多的经验或者没有沉淀下来那么多的工具框架等，很好的应用在金融的系统上。从经验上来看我们是一个新手，会犯很多错误，会踩很多坑。

二、全新业务。微博众筹是一个崭新的业务，虽然在业内已经有很多众筹，我们想赶上这个脚步，让自己活下去。它需要不同于其他业务的一种方式去进行，所以它会存在很多无法预测，比如业务的规模或者业务的模型模式对开发人员来说是无法控制。所以不确定性带来很多问题，也带来很多机遇。

三、网红的影响。第三个点网红这个点什么意思？微博现在主流的说法是粉丝经济或者网红经济。前几天微博网红节，大家都有所了解。**张大奕一天在微博卖东西能卖一个亿，而且就她一个人**。如果这些网红用我们的众筹，那她带来的高并发或者它的模型完全是不一样的。对一个新的业务，这是一个非常大的挑战，所以今天题目就是从新手到网红的变化。

怎么去应对这些问题？我们还是从刚才说的第一节那三个点，业务产品到服务产品到金融产品，分别来看微博众筹怎么在这三个点上去解决刚才遇到的问题。

第一个就是业务上我们需要快速迭代，比如京东众筹现在有很多项目，我们能不能很快的打造出一个众筹简单的平台，能让我们也在这上面跑，抢占一点市场的额度，这是快速迭代的需求。

这是几年前用的 **SWIFT 框架**。它的问题是后端去套用 HTML 的模板，可能套过模板都知道，这不是很容易，非常耗时。我们要经常跟前端或者跟产品沟通页面的设计，标签的设计，样式哪里又错了等，各种问题非常耗时间。如果要快速实现产品，这是非常致命的问题。现在红包用的是这个框架，它的页面比较简单，因此不是非常大的问题。

我们就开发了第二套框架：**EPF 框架**。它解决的问题就是后端人员不用去管模板，所有模板在前端写 HTML，他们直接去套数据。大大减少了直接的沟通，熟练的人干熟练的事情。这是两年前的框架，现在奇秀说的打赏就是用这个框架。

第三个版本引用了 **React 前端框架**，带来的好处就是前端套用模板那部分也放到了更接近产品的那一层，也减少了另外一部分的沟通的工作。可以看到这个带来的好处就是目前众筹一周可以发一个版本。从框架上从最早的 SWIFT SMARTY 所有模板前端处理，再后

来直接使用 React。主要是沟通上优化，还有页面呈现中一些优化。这是业务侧对于快速迭代需求的响应。

服务类型的一些业务，怎么实现高可用？我们在这方面都做了什么事情或者提供什么工具能实现这样的目标?

有一个前端监控系统，可以监控所有页面，比如众筹某个页面的首包或者是首屏时间出了问题，我可以很快速的定位，也有报警相关的东西，很快定位到底是哪个接口和页面，联合后端的数据。

（点击图片可以全屏缩放）

比如上图是一个 H5 首页的数据，如果有某个平均响应时间比较高，我们怎么去定位？后台就提供了一个类似瀑布图这样的结构，一次请求里面到底是哪条语句或者哪个过程耗的时间比较长？可以通过这个工具去分析及优化。

高性能上我们提供一个工具，这个工具背后有一个 EPF 框架支撑，包括它这套日志收集系统完成相关目标。

主要大概是底层框架，有一个平台展现，有一套系统去收集这些数据。收集数据可能会包括第三方 API 或者内部 API 的一些性能，包括前端渲染，首屏首包黑屏的时间都会在这个系统里面去呈现。我们能很快去定位到性能出现的问题的地方然后去优化，这是在高性能这部分做的事情。

在高可用方面，比如常说的 99.99%。如果用时长定位可用衡量的指标的话，一年中出现指标只有不到一个小时。如果你超过一小时，服务不可能四个九，只有三个九。当你收到报警，还没有迅速定位到问题，可能就超过四个九的范围了。

监控的设计

如果要快速发现问题，它的手段就是监控。我们能知道所有软硬件的问题，所有业务健康状态是否 OK，第三方调用所有依赖资源，你可能会调第三方的接口，它们是不是正常的，这是要监控的范围。

第二列目标，就是我们的系统该实现什么样的功能，数据是实时，报警必须是非常及时的，能提供一些性能分析的工具，刚刚我们类似瀑布图那样的一些工具。我们具体实现就是调用所有记录，系统出问题我们能及时知道，然后定位问题。

降级的设计

还有另外一种手段就是降级，比如页面功能比较多，小的功能可能会引起整个流程不能使用，这样就可以考虑把不关键功能降级，以便解决一些功能很复杂的时候，一些非关键因素引起的一些问题。

在方案上，得先得有隔离，得知道哪些能降级，哪些不能降级，如果全降级业务跟死掉一样。肯定是部分业务的降级或者是业务的部分降级，能让一部分使用，这是降级的方案。

模式可能有自动的，但成本可能比较高，比如所有业务的健康状态自动进行标识。另外一种方式是手动，目前我们系统里面大部分都是用手动的这种方式。手动比如埋一个开关，然后通过后台去控制。

在微博众筹框架里面为什么有这么奇怪的插件，这是因为我们基于公司的动态平台，有可能经常更新，我们会做到自动扩容的伸缩性，经常加机器和减机器。我们的开关是放在本地的，如果用类似这种方式同步文件，有一些限制，这个方向无法实现。我们做到 IP 上报，直接对这台机器的开关进行操作。这是对高可用方面做的两个工具。

伸缩性

一个工具框架，最忌讳就是过度设计，为什么？首先业务是变化的，我们无法预测，你做这个设计有可能是有用，有可能是没用的，我们都是先简单实现。但是又不能扩展性很差，所以在对于可扩展性这方面，我们做了一些类似 ID 的设计，我们后续想做分秒，可以按照这个结构 ID 做相应的处理。

金融安全

金融上面主要关注安全和一致性。安全做了哪些事情？业内常提到分层，我们分层分了五个应用四个层，这五个应用中，后面两个微博支付跟支付宝是公司已有，公司内部做的是应用平台跟金融。分层做的主要是分层隔离，它可以做很多的手段。比如后台的 API，肯定是后台才能去访问。如果你不分层，这些是很难去做实现的。

所有模块之间都是通过接口调用，你不能跨模块访问数据库，数据库本身也是隔离的。

另外一个层就是平台层，会对着两个上层跟下层，都是通过接口的方式去访问。他只会访问自己的数据库，他也有管理层的一些接口，管理层接口本身从代码权限，从访问路径都是有限制的。

这几个层意思差不多，我刚刚画了复杂的图，这是复杂图里面每一层的细节。手段分层做的主要是隔离，颗粒够细可以做更细的权限控制，我们做隔离和权限控制的处理。

数据库之间本身也是拆分了，前后端应用平台是前端和后端，管理端和用户端都是隔离的，相互是不能够访问的，做到一些权限跟隔离上去实现数据安全的一部分。

分了那么多层，它复杂度肯定是提高的，我们在做系统时候需要权衡，要保证一些的特性，肯定要丢弃一些别的特性。这就是我们宁愿自己是一个金融系统，我们要保证数据安全性，所以需要通过分层来实现。

分层带来的问题可能是它们之间的数据同步。比如上面我们就拿微博众筹三个层来说，它有应用、平台跟金融，它之间会有数据，同一个数据在这三个层面都会有，都有状态。怎么做到这数据的一致性，也是我们目前做了很多系统去保证这个一致性。

大家经常会看到，我们用淘宝或者支付宝，你支付完了之后他会有一个支付等待页面，有一个同步还有异步回调的过程。他主要是下层通知上层，数据是否 OK。我们分了那么多层，这几层数据通知的通道，我们后台建了一个奥莱这样的系统去做通道，提高这几层之间信息通道的性能。后续大家对这个感兴趣可以找我们详细去了解。

因为刚才也提到在金融方面我们是一个比较新的团队，所以我们设定了很多的规范或者是军规这样一些细节。这可能对于数据库事务上的一些处理。设计上也会有一些相应的规范，中间这个大概介绍一下。状态就是我们所有的对象都应该有最终的状态，不能放在一个中间的状态不处理。所有异常准备都应该有一个工具或者是流程去修复它。

另外我们会有相应的版本和规划或者是迭代的计划，对金融层或者其他层，它的节奏和它的目标不一样，这是我们金融层的一些规划。可以看到我们重点核心的部分都是在做安全跟数据一致性上面的工作。

总结

总结一下我今天所讲，我们从业务服务跟金融这三个层级来看做的哪些事情。为了达到快速迭代，框架上的升级，主要目的就是减少沟通，提高工作效率。服务的话可以采用业内常见的方法，比如提供一些工具或者是有一些平台去保障，这是服务的一些特性。

对于安全和一致性，我们做了很多底层工具，业务开发的时候主要是去分层，能实现这样的一些特性。我们有一些规范跟原则，都会去遵守。我今天讲的就这些，谢谢。

Q&A

Q：问一个无关技术的问题，我看您本科是学化学，您是怎么样走到计算机这行？给我们的建议是什么？
陈杰：首先我个人比较喜欢计算机，虽然没有搞编程，高中时代开始接触计算机，偶尔也会玩一些小东西，可能没有真正的去编程。但是我对计算机有爱好，这是一个非常关键的地方。如果你不爱好这个事情，你做这个事情有可能是虚度光阴。另外一方面，这也是机缘巧合，本来我是去中石化的，因为分到的是新疆，太遥远了，最后选择了 IT。

Q：看上面提到做了很多分层，分层肯定有很多接口，这个接口实现方式是 HTTP 是吧？
陈杰：是

Q：您提到访问控制，这块是怎么做的？分哪几个维度，怎么去监控这块？
陈杰：最上层比如说我们从物理访问 IP 上做一些限制，只有内网特殊的机器才能访问。比如说类似管理端可能只有一两台机器能访问，管理端的权限非常大，这是一层。另外一层我们会限制一些类似访问来源这样一些地方，我们的接口都是有一些权限，除了业务上有一些权限，哪些信息可以访问，哪些信息部可以访问。

Q：我比较关心咱们服务这块扩容的问题，你比如分了很多层之后服务力度比较细，这时候如果涉及到一些突发流量这块咱们有什么考虑？
陈杰：可以看到我们比如说从业务上分了三层，最前端是应用的平台，用户最先访问的是应用。假如就跟高速公路或者管子，最上面管子跟最下面管子比，上面管子粗你才能访问的了。我们设定的假如说正常都能访问，我们在扩展性的时候，我们底层的服务都是可以平行的方式去设定。比如说对业务层，最上层的用户访问没有太多跟底层数据库或者金融产品去做交互。可能它对于一致性的要求，我采用最简单的缓存或者是 CDN 的方式，用户都能访问的页面。数据尽量所有单元是无状态，直接能很快的扩容，对这个数据的整个业务流程影响是比较小的，我们设定的目标都是一些无状态的一些单元化的东西。

Q：你刚才有很多的流程监控，从前端到后端包括数据库一整套。它这个监控工具是一个吗？还是每一个比如说浅淡、后端还有数据库都分别有，还是就是一个监控框架？

陈杰：我们每个端都会收集数据，汇总到我们的平台做监控，里面有很多工具配置怎么报警，怎么呈现这些数据。

Q：作为监控工具的话是咱们微博平台自己的还是用很多开源的一些东西？

陈杰：目前用的数据是 ELK，数据搜集就是用 Logstash 去存，Kibana 去展现，中间配很多小工具去实现，有一些开源，有一些没有。我们后续有一些开源计划，因为定制性比较强，所以还在排计划。

Q：你说到数据一致性的问题，刚才说的比较笼统，能不能详细说一下？

陈杰：简单来说我们如果是一个金融性，我们数据肯定要求一致性，看到多少钱就是多少钱，不可能一分钟之后告诉我多少，这是强一致性，比较典型的例子。另外比如说打赏，打赏的用户列表，那可能是最终一致性。后续奇秀也会涉及到这方面，它这个列表只要最后一样就行，作者看到谁打赏的。

Q：上个问题如果是有分布式的那种框架，如何保证这个？

陈杰：我明白你意思，比如说有两个模型，ACID 和 BASE，对很多开发人员一致性不同，对 ACID 就是数据库基本的概念。他要保证就是强一致性，它改的数据必须是改，你用分布式，可能他就是通过比如说主从的同步来说是最终一致，但不是强一致的。我们的金融做强一致，肯定有一些跟事物相关的处理。刚才提到另外一个概念，分布式事物，我们后续会往这方面去发展，现在做的最好是阿里。

Q：最后一个问题就是接口，或者说你们本身开放的 API 或者调用别人的 API，你们接口安全性包括加密怎么考虑？

陈杰：加密方式是这样，我们现在通信中的加密，对请求数据层我们可能会目前是采用最简单，加个签名。比如说你是第三方，我是服务提供方，我们之间非对称的方面，你给我的数据上面包含一个签名。这种方式实现数据的安全保证，你不可能篡改数据，因为我们之间是有一套签名去非对称的方式去校验。

后记：目前网红的经济下，微博众筹刚一上线就面临很大的成交量和业务访问量的压力。陈杰上文讲述了一下他是如何在业务迅速构建的过程去平衡它的整个系统的资金安全以及在整个系统的弹性，可用性，一致性方面的一些实践的经验。希望对同行的工作和项目有所借鉴。

想更多了解本期互联网金融系统沙龙内容，请关注「ArchNotes」微信公众号以阅读后续文章。转载请注明来自高可用架构及包含以下二维码。

高可用架构
改变互联网的构建方式

长按二维码 关注「高可用架构」公众号