

# 余额宝技术架构及演进

原创：陈雨 高可用架构 2016-07-14

导读：余额宝开启了划时代的意义，开启了全民理财时代。上个月微博商业产品部联合天弘基金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技术架构问题与业界展开分享及交流。本文是陈雨在沙龙上的演讲，授权高可用架构首发。



陈雨，具有 8 年的软件研发和技术管理工作经验，专注于互联网金融、云计算、大数据等领域的发展动态和创新，目前在天弘基金负责基金注册登记系统架构和研发工作。

余额宝总结起来包括这样几个属性，第一它是一个传统的货币基金，但它把 T + 0 做到极致，另外他管理大量的用户资产。同时他具备极简的用户体验，符合互联网精神。我们在网页、支付宝 APP 或者其他途径能快速方便的进行基金申赎，它的应用渠道也非常多和广。

可以说从余额宝开始，真正的进入一个全民理财的时代，接下来给大家分享一下几个数字。余额宝用户数可以说达到了接近于 1/4 国人数量，日交易峰值可以达到两亿笔，最大并发数可以达到每秒五千笔。截止 2016 年上一季度公开披露信息，规模已经达到六千亿元以上。



从余额宝的创新来说可以从两个方面去讲它，一是业务上的创新，他对 T + 0 发挥到极致，是现金管理工具，是底层帐户。还有就是嵌入式直销，把货币基金嫁接到支付宝上去。当时来讲应该是一个在行业内是具有非常大的一个开创意义的一件事情。

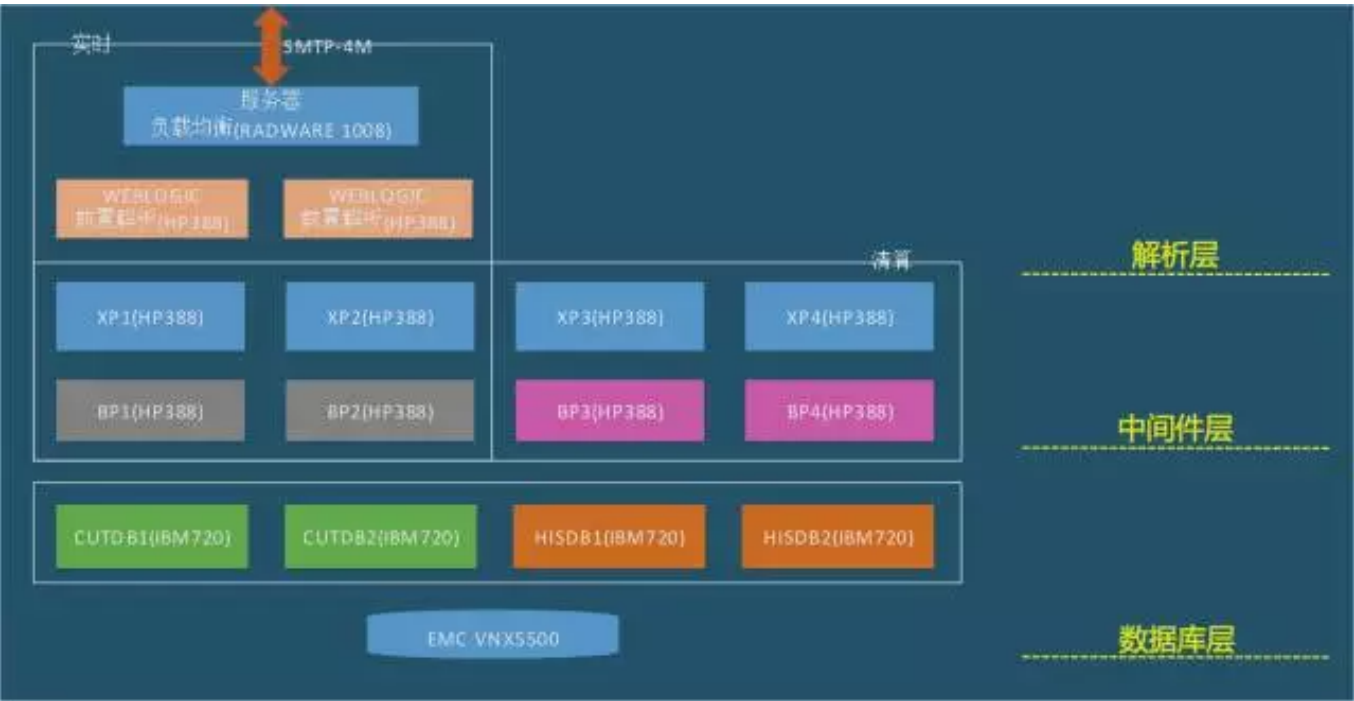
技术上创新是今天重点要说的事情：

- 1. **基金直销和 TA 清算的整合。**传统的基金系统直销和清算是分开。直销系统每天要把数据以文件形式导入清算系统里去。这件事情我们做了很大的改进，这么大体量数据来说，每天导入导出这个数据不可想象，在这里做了一个直销和 TA 融合，后面我会有一个详细的介绍。
- 2. **交易的简化，**监管大的框架下，满足监管要求的基础上，我们对交易逻辑做了很大的一个简化。
- 3. **余额宝是核心业务在云上运行的系统。**这是余额宝技术方面的创新。

## 架构演进历史

### 一期 IOE 架构

下面介绍下一期的架构，很明显看到就是传统的 IOE 架构。底层存储是 EMC 存储。中间层就是采用小型机，其中 KCXP 和 KCBP 是金证公司的消息中间件和业务中间件。往上前端是前置解析是用的 WebLogic，负载均衡用的硬件负载均衡。



这个架构对它的定位满足需求首先是支持千万级用户，传统基金销售模式是走代销机构的方式，投资基金用户也是以理财为目的。所以每天可能处理的帐户的开户可能也就是几万

到几十万的规模。由于余额宝对接是支付宝，支付宝有庞大的用户群，在用户规模上要达到千万级，这是当时对需求的定位。

第二点就是刚才提到把直销系统和 TA 清算系统做了融合，在数据库层面是共享的，避免数据再做一次导出和导入，对清算也节省了很多时间。

另外一点是传统基金的互联网化。传统基金只需要做到系统的 5 × 8 可用性，对接支付宝以后，要做 7 × 24 小时可用性。

2013 年 6 月，一期系统如期上线，业务规模远远超出我们想象。运营和运维人员反馈清算时间太长，基本上要从凌晨开始到早上八点，每天都是这样，我们感受到巨大的压力。另外当年要参加支付宝这边的双 11 活动，以当时的系统处理能力来讲，肯定是做不到的。

二期云端架构

基于这些原因，需要对一期的系统做优化，怎么优化？二期架构用一个词概括就是上云，充分利用云计算的计算能力，包括云计算对存储的处理能力。



整个架构进行了水平拆分。前面一期架构实际上就是一路的处理，到了二期把它分成多路。

从数据库层面分成多个 RDS（阿里云一款基于MySQL的关系型数据库产品）。另外一个就是去Oracle，很多利用数据库存储过程计算的部分，移到计算单元完成。

第三点是把直销和 TA 再次在计算资源层面分离。余额宝系统的数据处理，包括实时处理和批量处理。过去在一期架构的时候发现清算时，数据库负荷非常高，严重影响实时请求体验。所以在上云之后，在计算资源这块再次对它进行了分离，主要目的是提升客户体验。上云之后，当然充分利用了云计算的优势，其中很主要一个优势就是可扩展性。

## 水平拆分

接下来详细介绍一下是怎么来做水平拆分。

**第一点如何来分，以什么维度来分？最后确定以用户维度**，这样最终处理时间与用户交易的均衡程度有关。确定以用户维度进行拆分之后，确定哪些点来进行拆分，同样还是从用户角度出发，帐户、交易、份额、份额明细、份额变动等等。对于历史表直接合到仓库里去了，因为每日清算完之后，当日数据直接把它归档掉。

拆分之后，涉及到这样一个问题，TA 系统因为还要与周边的系统进行交互，**交互的接口同样还是文件**，数据导入需要先把文件拆成多份，再把每一份导入 TA，数据导出时系统要导出多份文件，再合并为一份。

## 总控

拆分最大的难点是在总控节点的处理，刚才说了 worker 节点能够保持松耦合，但仍需要通过总控节点进行统一协调，保持事务一致性。

最后数据核对阶段，也是要由总控汇总节点上的数据，按照清算规则对数据进行核对。还有很重要的收益分配部分，采用两个阶段来做，第一阶段由总控节点分配到每个节点上去。，然后在节点范围分配到用户粒度。

下图是上云前后指标上的一个对比，上云前基本上核心清算工作要做八个小时，上云之后在千秒以内可以完成。所以二期上云以后，IT 终于可以喘口气。目前来讲应对春节、双 11、国庆长假等场景，系统都能稳定应对这些。



(点击图片查看大图)

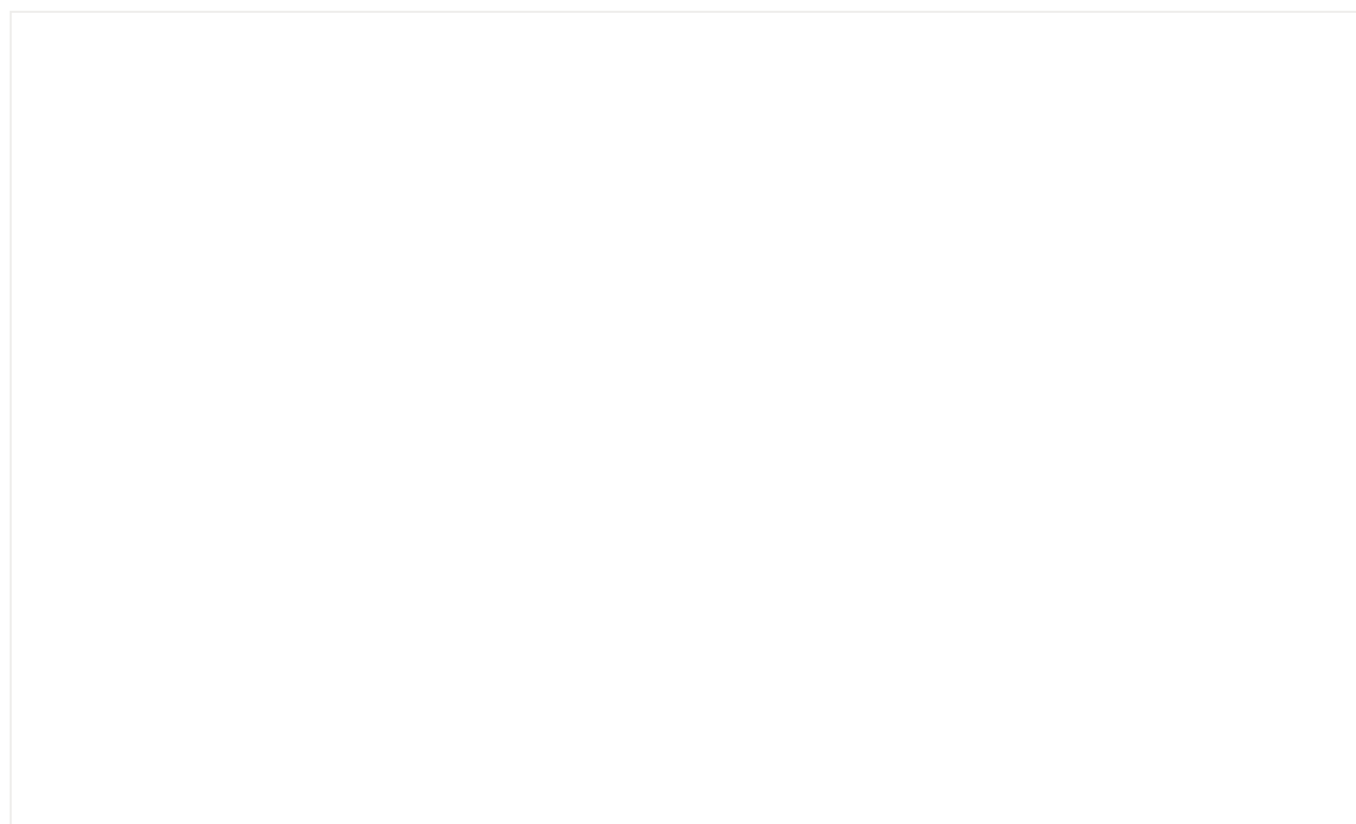
这是上云前后投入产出对比情况，传统的 IOE 架构特点成本很高，硬件成本给企业带来的压力非常大，云计算的好处就是在成本上是可以做到很细的，并且方便按需增加，这是一个非常大的成本上的优势。过去投入四百万只能支持一千万的帐户的量级，现在在投入上可能只是增长一倍，支持用户数已经远远不止一倍了。



数据架构

二期架构可以满足核心交易之后，还要考虑余额宝目前这么大的数据量，怎么把这个数据用好。

近一年来很多工作都是考虑数据后处理这块。其中数据来源于业务数据、日志数据和其他数据。我们推进数据仓库的建设和数据的产出。工具方面我们有很多自主开发的，同时也采用了阿里采云间，以及其他外采工具，具体支撑业务包括生产数据分析、资金预测、数据监控、运营支持，合规风控支持等等。开篇也提到了金融系统数据安全是重中之重，所以这块我们也会有相关的数据安全方面的管理。



## 二期架构的问题

二期架构解决很多问题，但并不是尽善尽美，总结一下还是有几个可以提高的点：

- **耦合**。首先计算和数据的耦合还是存在的。这实际上是对系统的扩展是不利的。另外，单个计算节点上，在业务上还是存在耦合，我们很多业务上的东西还是存在拆分的可能。
- **数据流转**，我们现在数据库层面也是分布式，所以数据的抽取、同步和流转会遇到很多现实的问题。
- **运维**。在运维方面除了遇到的传统分布式系统的运维遇到的一些难题之外，我们还在业务层面的运维也会遇到一些现实问题。

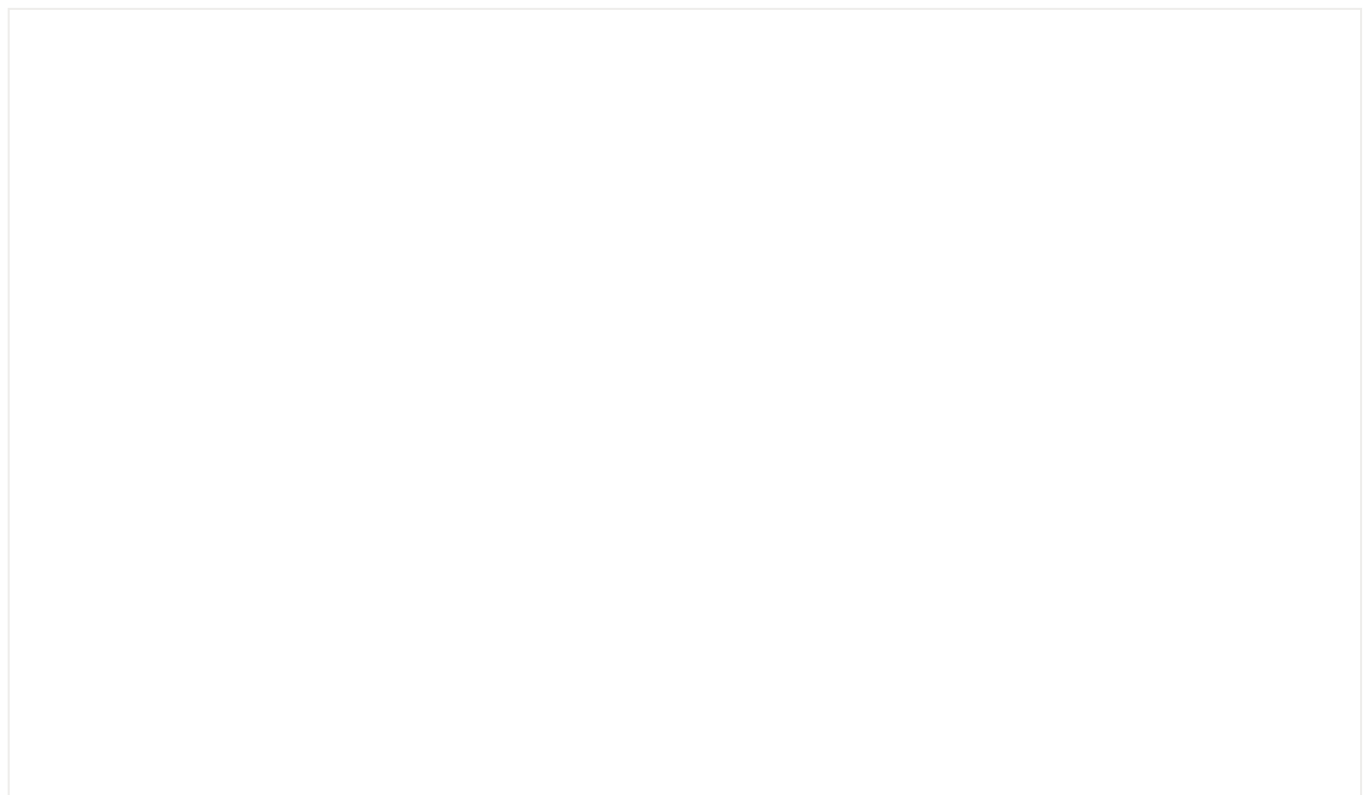
## 未来演进思考

对系统未来演进思考，主要分这么几个方面。

1. 从大的方面来讲是全局通盘考虑。我们要把核心和辅助系统通盘考虑，降低数据的冗余，降低数据维护成本。
2. 数据方面要用多不同的存储来解决不同场景的需求，还有刚才提到计算和存储的彻底解耦，做到计算和存储的独立可扩展。
3. 计算方面尽量做到业务上的拆分和轻量化，化繁为简，拆分之后把应用服务化。

### 数据驱动

我们系统的演进，数据量由单一小量向大量多类转变，同时应用种类从以交易为主到交易、分析和挖掘多种类并存。另外实时性要求也有变化，新的业务模式有时候要求实时或者准实时给用户呈现结果。



对业务来说对不同数据应用采用不同的存储。

- 比如对于在线交易，可以采用经过阿里支付宝验证过的 OB，专门用于解决金融级的分布式关系数据库的解决方案；
- 对于批量结算，可以继续沿用多年来在余额宝已经用的很娴熟的 RDS 集群。
- 对于 2T 到 PB 级的小数仓可以用 PetaData，解决以年度为单位的数据存储。

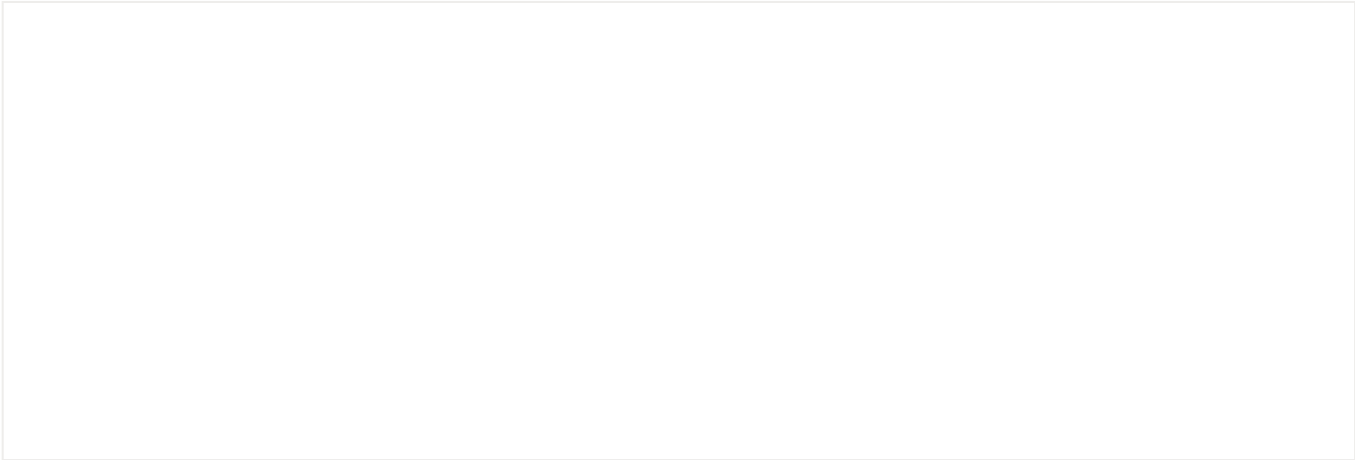
- 对于大规模的批量计算，数据仓库这块，我们直接就用 ODPS。
- 对大表存储可采用 OTS。
- 对于分析型、挖掘类需求可采用列存数据库。

服务化

关于拆分和服务化治理，后面考虑做的事情是充分利用阿里云的 PaaS 平台技术，把我们大应用拆分为简单的可横向扩展的小应用。



在服务的调用上，每个服务同时是服务提供方也是服务调用方，由 PaaS 平台的中间件统一管理服务。对我们来说是更多考虑如何基于中间件把业务来做好。服务化改造之后肯定会涉及到服务之间的调用。同步调用，可以直接走服务化的接口。



异步调用



**异步调用主要靠消息中间件。**金融系统对消息中间件的可靠性要求非常高，这块我们还是沿用传统思路，并不想采用开源解决方案去填那些坑，更多考虑采用成熟金融级消息中间件来做这件事情。



下面是一个总图，中间 EDAS 是统一企业级服务化解决方案，然后通过 DTS 解决数据实时同步的问题，采用 CDP 解决离线数据同步的问题。在数据应用上可以满足很多的需求，比如采集系统或者报表展示或者是用户短信的推送等等，这就是我们对整个未来的架构演进思考。

## Q&A

**提问：都切到云上，数据安全上怎么考虑？**

陈雨：之前讲到金融要求是私有云，我们是在阿里金融云上，并不是在公有云上，可理解为物理上是隔离的。

**提问：接口交互的技术是文件，文件的完整性和一致性如何保证的？你们自己要处理它吗？为什么要用文件的方式？**

陈雨：我们对接是支付宝，文件的正确性和准确性由支付宝保证。我们需要对大文件按节点数拆分成小文件，然后并行处理。接口必须用文件方式，金融行业很多系统对接最后要走文件接口，文件是用来对帐的准确性保障，实时不是那么可靠。

**提问：说到计算和数据耦合，输入输出解开，具体大体上是怎么实施它？**

陈雨：RDS 来是单机数据库产品，通过分布式中间件 DRDS 或其他解决方案，以实现计算节点像使用单机数据库一样使用数据库集群。

**提问：咱们有基于用户纬度拆分，主要是什么原因导致我们要这么拆，基于用户纬度拆分，有没有比较坑的地方或者我们怎么避免它？**

陈雨：基于用户的拆分，一方面签约协议号是跟支付宝的接口，还有一个考虑是以用户为维度的查询需求相对多。当然其他非用户纬度查询就费点事了。

**提问：我是互联网金融从业者，刚才您提到我们余额宝系统，有清算系统是吧。不知道清算是有内部清算和外部清算，我们这边清算是怎么做的？比如说内部清算是指交易明细和你的帐户余额之间的比对。你外部清算可能是你本地的数据和银行数据之间的比对。**

陈雨：我所说的清算是你所说的第一种。每天做一次内部比对，计算用户的份额和收益。

**提问：**之前也用过其他的消息中间件，你刚才提到成熟的消息中间件不是开源，我们其他从业者不能用到是吧？

陈雨：这涉及到一个生态圈的问题，如果进入阿里云的生态圈，可充分享用云计算资源。如果确实是在生态圈之外，可选择它的对应开源版本。开源版本在版本更替上或者服务方面，跟阿里云上存在一定的差别。

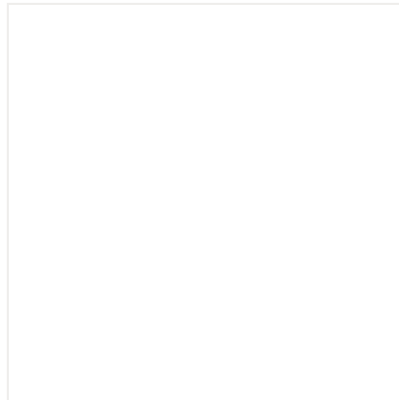
## 相关阅读

点击链接阅读相关文章

- [互联网金融系统技术沙龙:微博众筹架构设计](#)
- [互联网金融系统技术沙龙:小米风控实践](#)

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