

# 开篇词 | 从今天起，换种方式学存储

李玥 · 后端存储实战课



你好，我是李玥，《[🔗消息队列高手课](#)》专栏的作者，目前在京东任职架构师。这是我在极客时间上的第二门课程，很高兴在这里遇见你。

在十多年的开发者职业生涯中，我的从业经历应该算是比较丰富的。在传统 IT 行业，做过非常多的企业级 ToB 的系统；转战互联网后，我又曾经带领创业团队体会过从 0 到 1 创业的艰辛，见证过互联网高速增长的高光时刻，也经历过京东数年大促的洗礼。

在工作过程中，我接触过很多系统。不同的系统，它的业务不一样，有做社交的，有做电商的，还有做内容的。系统的规模也不一样，有很小规模的系统，也有像 BAT 这样的巨无霸系统。在构建这些系统时，都会面临五花八门的问题。但总结下来我发现一个“神奇”的规律：

**凡是那些特别难解决的、让你付出巨大代价的，或者是损失惨重的技术问题，几乎都可以归为存储系统的问题。**

这个规律其实并不神奇，它是有原因的。

你可以想一下，我们开发的各种业务系统，几乎都是 MIS 系统，中文叫“管理信息系统”，有的大学还有这个专业。管理信息系统这个词的含义就是字面的意思：管理信息的系统。这里面的信息是什么？对，其实就是数据。不管你的系统业务是什么样的，最终都要落到对信息的管理，或者通俗点儿说，你系统的业务功能最终的结果就是数据。

只要这个数据是正确的，剩下的都是小问题。数据错了、丢了，甚至数据处理不及时，这些都是损失惨重的大问题。

所以用于承载数据的存储系统就显得非常重要，如果说，你能构建一个安全可靠、快速稳定的存储系统，基于这个基础之上构建的业务系统，显然就让人放心多了。

所以说，**存储是系统中最核心、最重要、最关键的组成部分，没有之一。**

## 你要关注存储的哪些特点？

我们常用的存储系统有很多，有单机的也有分布式的，有数据库、文件系统，还有一些介于二者之间的，种类非常多。无论是什么样的存储，比如 MySQL、Redis、ElasticSearch 等等，它们都有几个共同的特点。

**第一个特点是难用。**怎么难用呢？对于应用程序来说，存储无非就是帮我们安全可靠地保存数据，在我需要的时候能快速读出来也就可以了。很遗憾，几乎没有存储系统能满足这么简单的要求。

有一个非常形象的比喻：我开着车去商场购物，到了停车场发现这个停车场不能存车，只能存零件。我必须自己把车拆了，然后把这些零件分门别类打上标签，存放到停车场货架上，走的时候自己再把零件取出来把车组装起来。

听起来很可笑是吧，你想想咱们用的这些存储系统，不就是这样吗？我们应用程序里管理的数据都是对象是吧？你告诉我哪个存储系统能存对象？没有吧？

拿 MySQL 举例，我要存取对象，必须把对象转换成 MySQL 表中的行，还得写 SQL 语句才能存取。是不是很难用？难用你还不得不用，并且还得把它给用好了，这里面有很多的方法、技巧和实践经验需要学习掌握。

**第二个特点是慢。**最近几年的技术圈，分布式存储这块儿非常繁荣，你可以看到过一段时间就有一个新的数据库诞生了，不管它们功能怎么样，无一例外，都说自己的有多快，性能多好，顺便把像 MySQL 这样的老家伙拉出来，做个性能对比测试，吊打一遍。

“一个人炫耀什么，说明内心缺少什么”，这个道理放到技术圈同样适用，不断有新的存储刷新性能记录，恰恰说明了存储系统性能不能让人满意。

一个经过良好优化过的业务系统，它的性能瓶颈一定是存储。从性能角度上来说，存储系统就是整个系统中最短的那块儿板子，存储系统有多慢，你的整个系统就多慢。如何优化存储性能，从而让整个系统运转如飞，这里面同样有很多的方法、技巧和经验需要掌握。

**第三个特点是杂。**存储这块儿不像其他的成熟的技术领域，基本上都是一两种方案包打天下，比如 Java 开发，基本上就被 Spring 统治了，再比如 Web 容器，静态用 Nginx，动态 Tomcat。但存储这块儿却不是这样的，就拿真正广泛应用在生产系统中的存储来说，你看有多少种？

MySQL、Redis、ElasticSearch、HBase、Hive、MongoDB、RocksDB、CockroachDB 等等，这些存储还真就是谁都替代不了谁，每一种都有它擅长的地方，有它适用的场景，当然也有很突出的短板。如何根据业务系统的特点，选择合适的存储来构建我们的系统，这也是需要学习和掌握的。

## 学习存储的最佳姿势是什么样的？

既然存储有“难用、慢、杂”这几大特点，学习起来就更需要注意方法。如何来学最高效呢？我认为是实战，从问题入手。

存储涉及到很多理论知识和概念，比如各种数据结构、哈希、树以及它们的时间复杂度等等，这些内容往往都是偏数学范畴的一些知识，学起来不容易理解和记忆。并且，理论和实践之间往往存在着非常大的鸿沟，往往是“懂了一堆道理，却还是写不好代码”。

所以，我在极客时间上开了第二个专栏讲存储，我们只讲实践中大家都会遇到的问题，讲这些问题的解决方法，同时在这里面贯穿一些知识和原理。通过这样的学习方式，既可以快速地帮你解决实际问题，同时还能提升你的技术能力。

在接下来的课程中，我会带你一起，**从 0 到 1，从小到大，以电商作为场景，讲解不同规模的存储系统应该如何构建。**

每一节课，我们一起解决一两个实战的问题，比如：为什么明明数据量和访问量不大，MySQL 还是很慢？数据库宕机了怎么办？

为什么选择电商系统来讲呢？因为我熟啊！开个玩笑，当然不只是因为我做过几年电商系统。其实，很多培训学校、各种技术论坛都特别喜欢讲电商系统。因为电商这个系统，特别有代表性，特别适合作为案例来研究和学习。

**首先，电商系统覆盖面足够广泛。**特别是在互联网行业，你会发现几乎所有的互联网公司都在做两件事情：电商和社交。

**其次，用电商系统作为案例，直接就能学以致用。**即使你面对的业务和电商关系不大，因为电商的系统足够复杂，你在其他业务中可能遇到的技术问题，大多数在电商系统中基本都会遇到，一样有借鉴的意义。另外，电商这个业务领域对所有人来说都很熟悉，拿它作为案例基本上不需要再讲解业务知识，我们可以快速地专注于技术问题本身。

即使是同样一个电商系统，不同的规模，它需要解决的问题也不一样。不少做技术的同学崇尚于海量数据和高并发，认为只有大厂那些高并发、海量数据的核心或者是底层存储系统，才是真正“有技术含量”的系统，能胜任这样系统的开发者，才是真正的技术大牛。这其实是一个技术认知误区。为什么这么说？

因为，并不是规模小的系统就简单，只有大规模的系统才有难度。

创业团队需要快速低成本把系统完整地实现出来，好快速验证它的商业模式；处于高速增长期的团队，它面临的问题是业务高速增长和不断变化，相应的，也要对系统不停地进行升级改造来适应变化，并且要在变化的过程中确保稳定；业务规模足够大的一些大厂，它需要解决的是如何应对高并发、海量数据这些问题。

所以说，不同规模的系统，在技术上没有高低贵贱之分，它们的建设目标不一样，面临的挑战不一样，需要解决的问题也不一样，对于存储系统的选择、架构设计也是不一样的。

所以，我们的课程设计就是按照系统的发展过程，分成了创业篇、高速增长篇和海量数据篇这三个部分。

在创业篇，我们重点解决从 0 到 1 的问题；比如：如何低成本高质量地快速构建一个小规模的订单存储系统。

在高速增长篇，我们关注在高速变化的过程中，你的系统一定会遇到的一些共通问题，以及该如何应对这些问题。比如说，如何从单机的存储系统逐步演进为分布式存储系统；如何在线平滑的扩容我们的存储系统。

在海量数据篇，我们重点解决高并发、海量数据情况下的存储系统该如何设计的问题。比如，海量的埋点数据该怎么存储；如何在各种数据库之间实时迁移和同步海量数据，等等。

通过学习这门课程，你将收获的不仅是案例中那些解决具体问题的方法，在电商系统架构、对存储系统的认知以及存储系统的设计能力这几个方面，都会有所提升。

更重要的是，通过案例来学习常用的数据库和存储系统的使用和实现，可以总结出存储系统最通用、本质的技术原理。掌握了存储系统的本质之后，不仅会让你在面试时更加从容，而且会让你对存储的理解上升一个层次，从“知道怎么用”，升级为“知道为什么这么用”，最终做到活学活用。

在极客时间的一段新旅程即将开始，在开始正式学习之前，我还想再说一些我的想法。技术的发展让使用技术变得越来越简单，但是作为有理想有情怀开发者，不能让技术把我们的变得越来越“简单”。我很开心又能同各位同学一起持续地丰富自己，也希望你能不负时光，认真对待这段学习之旅。

在学习过程中，你都可以在留言区提问，我看到会第一时间给你回复。另外，像《消息队列高手课》专栏一样，我也希望你可以在学习之初立一个 Flag。有了目标指引，持之以恒地探索，成功必不会偏航。

好，开始学习吧！

© 版权归极客邦科技所有，未经许可不得传播售卖。页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

## 精选留言 (41)



业余草

2020-02-26

求代码，求实战，少讲理论。。。

共 14 条评论 >

👍 66



业余爱好者

2020-02-26

后端一般是个分布式系统。计算是分布式的，所以需要有负载均衡组件。存储是分布式的，所以有各种分库分表的库或中间件，以及比较热门的分布式数据库。可以看出，存储比计算要复杂，也是后端的重点与难点。

搞技术的一般都比较抵触业务，但是技术是为业务服务的。两者也并不对立。甚至可以说，技术从某种程度上说也是一种业务，是各种业务的共性部分。因为不管什么系统本质上就是在计算，然后积累数据。技术就是"元业务"。学习就要学习这些通用的，底层的技术。

电商是经济化，电子化时代非常重要的业务，满足人类的经济需求。社交满足政治需求。极客时间满足文化需求。电商业务设计的领域以及技术是比较多的，业务上涉及支付，库存管理，物流，技术上，单点登录，缓存，消息队列，负载均衡。所以比较适合用于作为学习技术的背景。

消息队列课虽然买的早，但是没跟上节奏，可能是因为工作中没有应用场景。

这次的后端实战课一定要跟上节奏，以点带面，以练带学，拿下后端存储。（顺便要把消息队列课程中没跟上的内容补上）电商涉及的相关业务也要顺便弄个明白。

作者回复: 🍌🍌🍌

共 3 条评论 >

👍 48



何妨

2020-02-27

业务代码好不好写，全看数据库设计存储设计的好不好，停车场的例子太真实了



👍 20



FATMAN89

2020-02-26

老师这次的大头照再也不像黄渤了，像吴彦祖了，嘿嘿



👍 11



梨还羽

2020-03-05

Elasticsearch不就是存对象的吗？

作者回复: 是的，已经非常接近应用中的对象了，这也是ES越来越受欢迎的原因之一吧



👍 8



Fchen

2020-02-27

「不同规模的系统，在技术上没有高低贵贱之分」是的，没有贵贱之分，但人家就喜欢招聘做过高并发处理过海量数据的人



👍 7



aoe

2020-02-27

老师，MongoDB可以看做是直接存取对象吗？不用把车拆成零件。

作者回复: 单个对象来看，别说还真是。集合和对象嵌套这种复杂的情况，还得咱自己拆装。

共 3 条评论 >

👍 6



L

2020-03-01

浏览了课程大纲，这些都是我工作中遇到的问题，公司里没有人能指点一二，自己的知识储备有限，总是百思不得其解，终于等到这门课程，期待自己打开新思路。

作者回复: 希望你通过学习能有所收获。



👍 4



良木

2020-02-26

按时打卡学习，这次老师的照片是极客MM按吴彦祖P的啦～！



3



约书亚

2020-02-27

这一次的画风很好，亲切，幽默，希望未来都以这种方式行文。架构师都比较严肃，不好玩。



2



小伟

2020-02-26

跟着李老师的《消息队列高手课》一起搞定了消息队列，对李老师非常敬佩，满篇干货，妙语连珠，特别是尾篇对于学习方法的把握让我受益匪浅。

非常期待李老师的存储专栏，更想有更多机会能和李老师交流。一起参悟科技的本质，领略科技的乐趣。加油💪！



2



南山

2020-02-27

看到开篇，就心动了



1



Spring coming

2020-02-26

主要想长长见识



1



静水流深

2020-02-26

终于又看到老师了。



1



FM微言送

2020-02-26

必须买啊，上一个老师讲的非常好👍👍👍。必须支持。





leslie

2020-02-26

继续跟着老师学习：第一季消息队列学完，第二季继续学习；希望同样能有所斩获。



不记年

2020-02-26

希望老师把各个阶段的各种技术原型，决策都详细讲下～



2023-07-03 来自上海

测试同学来学习



ifelse 

2022-12-01 来自浙江

存储是系统中最核心、最重要、最关键的组成部分，没有之一。--记下来



ifelse 

2022-12-01 来自浙江

学习打卡

