

从技术细节看美团的架构

夏华夏 极客时间 2016-01-23

很多人认为，电商都没有什么技术含量，电商没有什么门槛，入门的门槛并不高，电商很痛苦，需要不停地去扫街，不停地去拜访各个商家，要在用户和商家之间拉客接客。国内曾经出现的团购类网站有6400多家，到四年多以后的现在，美团已经是成为国内最大的本地生活服务平台，不管怎么说，现在美团在这些电商，至少团购类的电商里边是走的比较成功的，如果说电商真的是没有门槛，那难道说美团走到现在是因为幸运吗？

那必然不是因为运气，如果大家知道王兴，美团的创始人，他在这个行业内有个非常响亮的外号，叫国内史上最倒霉的连环创业者。因为他之前做过像校内网，做过饭否，最后都是因为一些莫名其妙的原因就没有做起来，或者发生了很多问题。**但是美团现在他做得非常好，那肯定不是因为运气。**其实在我们内部，很多同学也在做思考总结，我们希望找出一些比较好的东西能留下来，然后以后继续保持，在这其中分析来分析去，其中有一部分很重要的原因，就是我们技术团队的努力。今天与大家分享的，就是在技术团队中，不断追求极致努力的一小部分经验。

首先第一部分给大家介绍美团的**技术架构**，**架构是如何演变的**。第二部分讲一讲美团的**业务架构**，在业务方面如何做一些业务流程的优化。最后第三部分介绍**O2O技术**，如何实现线上和线下都用技术来做优化贯通的。

第一部分首先讲一讲**技术架构**，其实在初期的时候，美团的技术架构非常简单，的确在最初2010年、2011年的时候，技术是没有门槛的，任何一个人都可以写一个电商的网站。

技术架构：初期

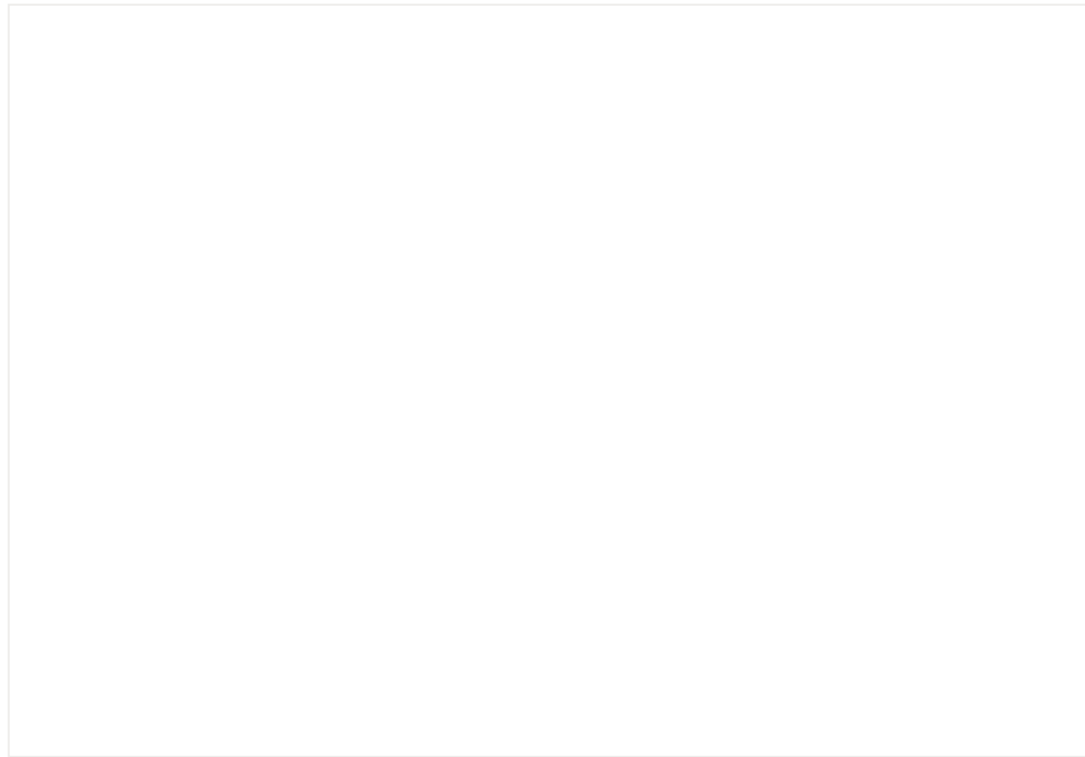


这就是一个最初期的架构，一个比较典型的LAMP架构，前端加上Apache/PHP，后端是MySQL，当然我们会有一些运维的工作在里面。可能大家如果自己写个网站的话，一开始都是这种架构，这种架构一开始也很好用。然后慢慢的，当业务量大了之后，我们发现整个系统的性能跟不上。那时候我们也只是做一些简单的优化就够了，比如说一开始我们是在前端，就是在Nginx和Apache之间加一些Varnish的缓存，然后在后端，我们可能用MemCached来减少MySQL的压力，这些都是缓存，整个架构还是没有太大的变化，还是一个优化了的LAMP架构。

技术架构：初期



然后到2011年的时候，我们开始做移动端，这时候架构还是没有太大的变化，只不过是Apache这种已有服务的API前面，又包了一层。就是我们在提供给PC端的同时，我们也包了一层移动的API，这样我们可以继续给手机端的用户提供服务。



这个时候其实也就是简单地把LAMP架构做了一点点扩展，但是已经可以支撑很多很多的用户，很多很多的容量了。我们在这种架构的前提下发展，直到我们想去做新的业务。美团一开始起步是个团购公司，后来我们去做一些新的，比如说酒店业务、电影业务，直到现在大家可能使用过的美团外卖的业务。当我们去做很多不同的业务的时候，我们发现做每一个业务似乎需要添加一些新的部分，这样一个部分、一个部分堆积，对很多技术的同学来说，这是不能容忍的，那我们怎么去改进它呢？我们希望把中间的很多的公共的东西，与业务无关的东西抽取出来，形成一些公共的技术的组件，这样可以为很多的不同的业务来使用，发展到现在，形成这样一个看起来稍微复杂的架构。

在最底层会有云平台，对内对外都有服务，会有云主机，会有云存储，也会有虚拟网络，包括一些负载均衡的东西。在云平台上面，我们会有一些基础的组件，这些基础组件跟业务的逻辑相隔比较远，它会有比如配置，队列中心，注册中心，包括一些SQL和NoSQL的存储等等，这些技术组件我们在所有的业务里都会使用，所以我们把它提取出来，作为我们的技术组件提供给业务能用。再往上，确实有一些东西是与业务结合比较多，比如说用户中心，支付，包括一些搜索，推荐，还包括，比如说我们会建立用户的一些地理位置的库，包括一些风险控制的东西，这些东西是与业务是有交互的。但是我们去分析之后发现在不同的业务里面，这些组件还是差不多的，所以我们也是把它抽象出来，现在叫业务组件，这些业务组件在所有的业务之间也是共用。

再往上才是我们各个不同的业务的，真正的比较独特的一些逻辑。在这些业务逻辑前面，是前端的接入，这个前端接入其实对不同的业务也是一样的，它会有前端的接入和转发，会有前端内容的过滤，就是一些防抓取，防攻击这样的内容过滤。比如说为了做用户访问

性能的优化，我们会做大量的各种各样内容的缓存，包括CDN也好，包括我们内部不同层次之间，包括一些验证码的服务。所以在这种架构下面，当我们再要去做一个新的业务的时候，我们就关注在中间业务逻辑这一块就可以了，这样可以很快地去拓展新的业务逻辑，而且每一个人，每一个团队，只关注真正最有价值的那一部分的软件的开发。那当然两边会有我们的，运维的工作，安全的工作，是在每一层都会涉及的。

但是整个这样一个逻辑发展到现在，我们是觉得最适合我们美团现在这个阶段的一套技术架构，那从一开始的最简单的LAMP，到现在可能我们分了很多很多个组件、很多很多层，这些架构看起来是非常非常不一样的。但是我们现在回想起来并不觉得说，原来的就不好，现在的就好。我们觉得在公司发展的不同的阶段，一开始就最适合那种最简单的情况，如果说我们一开始，比如说美团2010年成立的时候就上这种很复杂的架构的话，那可能我们2010年底才把软件开发完，那时候上线的时候，可能已经有五千多家团购网站在线上了，所以这是不切实际的。所以整体来说，我们觉得在整个技术架构的演变过程中，就是找当前真正能够满足我们业务需求的。

另外一个特点，大家也可以看到，在我们的整个的架构里面，大量应用了一些开源的东西，从最初的LAMP架构的时候，包括MySQL、Apache，到现在我们一些很复杂的架构里面，比如说搜索，现在会用到Lucene，会用到Solr，在云主机、云平台这一块，我们会用到比如说OpenStack的一些个组件，包括比如存储的Swift等等，用到很多的开源的东西。开源产品拿过来当然会加速我们的这种开发的周期，但是开源产品我们也不仅仅是单单把它拿过来，因为任何一个开源的产品，如果你要拿到一个比较复杂的业务里，你就会发现它不是那么匹配的，它总是有些边边角角，比如说要与系统的集成，或者很多开源产品，它在大规模的情况下，高并发的情况下，考虑地并不是那么周到。所以我们在开源的基础上做了大量的优化，一方面能让我们的整个系统能做更好地水平的扩展、系统的扩展、系统的优化，同时也让整个的用户体验能够更好。

总结下来，就是在技术架构方面，想跟大家分享这么几点，一个就是整个技术架构总是在不断地结合业务在不断地演化，还有就是至少从美团来说，我们是在开源软件的基础上，然后不断地做集成，不断地做优化，最后，软件开发的时候，不管是在对用户体验来说，还是对工程师自己的体验，我们总是在追求一些极致，这样的情况下，我们的技术架构就自然而然的在不断地演变了。

第二部分分享一些业务架构方面，我们做的优化。

这个图是一个比较复杂的图，我们也不去讲它的太多的细节，大概分析一下，上面这一块其实是刚才给大家看的，对用户访问端，它所涉及到的一些组件，一些部分。但是对于电商来说，其实它还有一个很复杂的生产系统，这个生产系统就是说我们怎么去跟生产商谈

单，谈完之后，我们怎么把这个单子录到线上，怎么去编辑，怎么去审核等等，这个单子的生产我们叫生产系统。

除此之外，还有整个公司的运营，一些市场的营销推广，我们怎么去拉动我们新的用户，怎么去拉动我们的新的商家等等，所以就涉及到很多的业务的模块。整个的这个框架，细节我们不关注，**但是第一感觉肯定是非常复杂，这个复杂的业务架构有一个什么后果呢？**一般来说，它会让整个流程非常复杂，当流程复杂了，那自然而然带来的整个效率低下，所以对于技术团队来说，我们一个努力就是在不断地去优化我们的业务架构，不断地让流程简单，让效率更高，那怎么来优化呢？

我们有一些自己总结出来的方法论，就是让复杂的事情简单化。一个很复杂的业务架构，我们希望对它做很多理解和梳理，梳理的过程中，我们就会发现一部分步骤其实是不需要的，可以省略的，这是一种简化；还有一个就是，当我们梳理完了，发现每一个步骤都需要的时候，我们会尽量地把一个复杂的东西拆成很多比较小块的，易于把控的一些东西，这就是一个把复杂东西简单化的一个过程。当把一个复杂的东西拆成了简单的小的东西之后，我们就容易地去对这个简单的模块，简单的功能进行标准化。

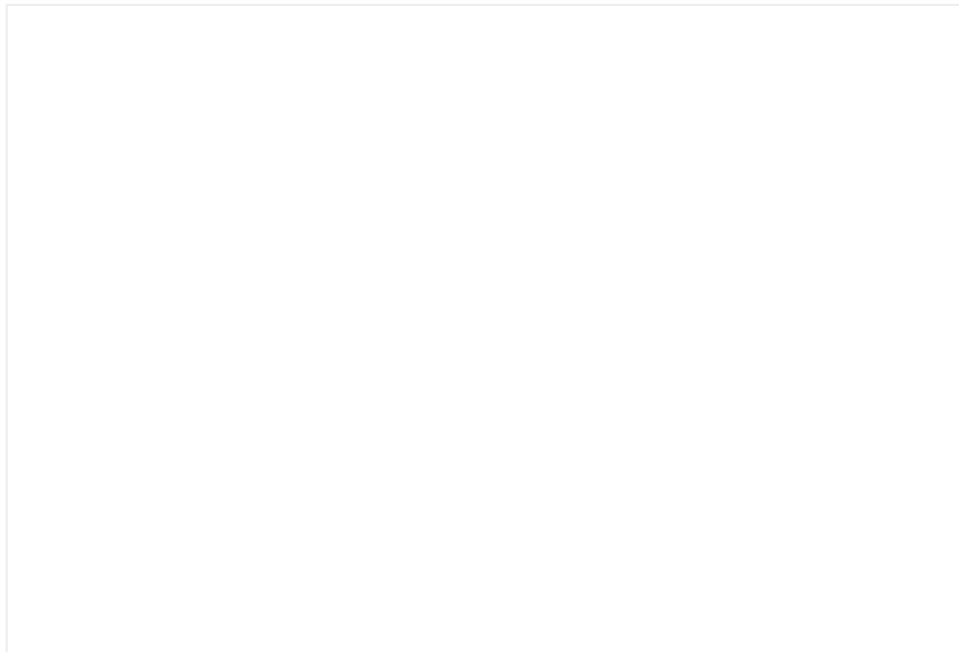
所谓的标准化，就是去制订一个标准，这个东西该怎么做，应该实现什么目的，做了之后我们怎么去衡量。所以这三个是非常重要的，就是我要去做什么，我怎么做，然后怎么去衡量。**如果把每一个简单的东西都处理好了之后，这个简单的东西就成了一个标准的东西，标准的东西在很多时候就比较容易去推广。**这就是标准化的过程，如果整个的标准比较完善了，那我们就希望把这个标准固化下来，固化下来就是说整个的工作就会变成一个很简单的流程。

招几个新的员工，然后给他们一个手册，告诉他们，照着这个手册一步一步，第一步做什么，第二步做什么，第三步做什么，这就是流程化的东西。如果发展到这个时候，其实复杂的东西已经可以比较高效地往下运作了。但是对于计算机来说，对于搞技术的同学来说，我们知道，其实计算机它最擅长的东西就是处理这种简单的流程，所以我们如果做到流程化，就有了一个自动化的基础，我们可以用计算机来把这些固化的流程完成，这样最终就把复杂的事情能够尽量地做到自动化。

我来举一个简单的例子，尤其是后面流程化和自动化这个东西，大家可能不是那么理解。就是我们在上单，所谓上单就是一个单子，比如一个餐馆售卖的东西，就是本地服务的一个产品，我们叫一个单子。上单的时候，我们的销售同学和商家谈了一个问题，最终要上到我们的整个网站里边。**我们今年上半年曾经做过一个很大的努力就是，上单的时候我们**

希望免审核、免写、免编辑，为什么要这么做呢？给大家介绍一下旧有的流程，在旧有的流程里边，销售团队可能从签订合同开始，还不算他一开始跟商家去谈私人关系，去一次一次的沟通，那时候可能要碰很多壁，即使是商家已经同意了要和你合作了，那销售的同学就可以和商家签订合同，从这个时候就进入我们生产流程，然后要到我们的审核团队去审核合同，看这个合同的价格，定价是不是合理，是不是偏高，或者偏低，因为偏高了损害用户的这个利益，偏低了之后美团要贴钱，所以要去审核，包括一些法律的东西，是不是合法，一些条款是不是合法，这是审核，如果审核不通过，要打回来，重新签，如果审核通过了，回到我们的编辑团队，那编辑的团队会干什么？

会把合同里的东西输成文档变成文字，变成一个文本的描述，然后还包括编辑的同学，摄影师的团队，他们会去每个商家去拍很多菜品的照片，或者商家门头的照片等等，还要把这些照片再去剪切，包括打上防伪的美团的水印等等，这些就是我们编辑团队原来要做得，他要把所有的这些东西原材料变成一个网上的单子，上了单子之后，先在一个系统里给商家看，这是我要给用户展示的东西，这样行不行？商家说可以，我们就可以最终给用户来卖了，那在这个整个的流程，从签订合同开始，到最后用户能够看到这个单子，这之间的时间是7到10天，这是一个非常非常长的时间，因为7到10天就可以对商家带来很多很多流量。我们现在每天的销售额是几亿人民币，如果我们每个单子都拖到7到10天的话是不可忍受的。



那技术团队就会想怎么去优化这个流程。我们其实做了几件事情，第一个就是说把业务流程所有的东西尽量在线化，比如说离线运行的东西。有的编辑的同学他本来是去签一个纸质的合同，纸质的合同寄到我们编审那边，编审的同学要去一条一条的读这个纸质的合同，这个是很难容忍的，这个是没办法提升的。我们首先就把很多的，比如说合同，尽量在线上来填合同，还有摄影师拍的照片，尽量直接传到网上，不要通过一个其他的渠道，

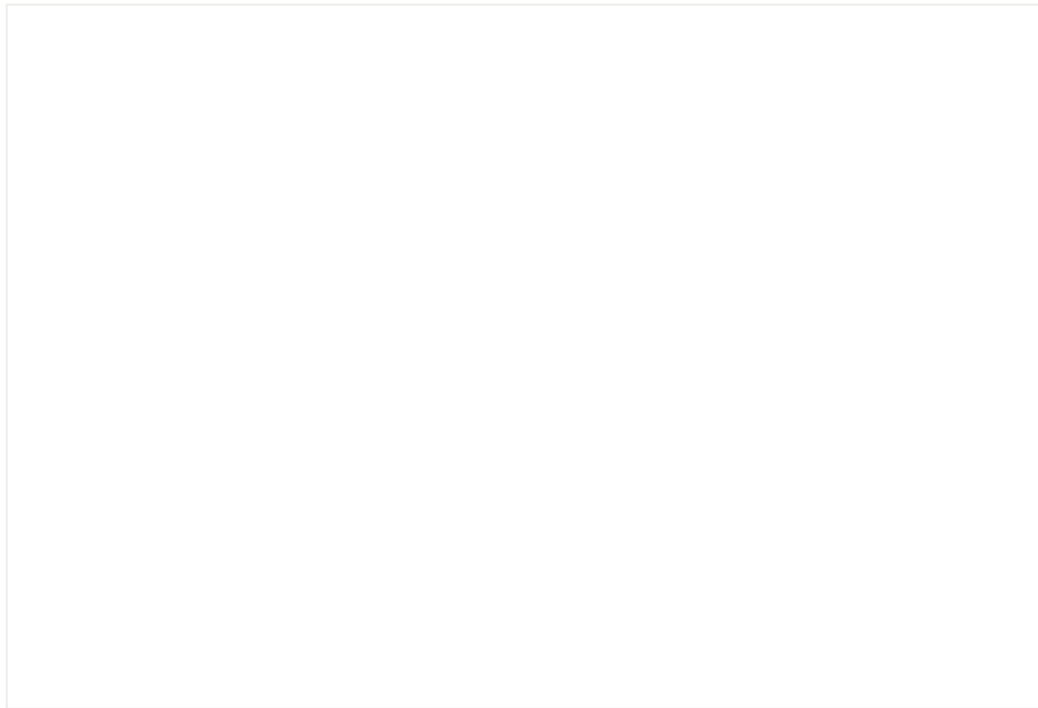
U盘等等来传。这样所有的东西在线上了以后，我们才有了所有用计算机处理的一个基础。

还有一个就是，我们希望所有的数据结构化。举一个例子，对于一个餐馆来说，我们可能往往会有很多的条款，比如说他这个餐馆是几点到几点营业，这个单子是几点到几点可以用，用的条件包括你可以用包间，或者不可以用包间，以及是否提供停车位，还有一些菜单的东西，这些东西在最初的时候，就是我们编辑的同学一条一条对着那个合同把它用手录成一大段文字，这样不是结构化的东西。我们结构化的努力，就是我们把每一项条款都变成我们数据里的一个结构化的单元，比如说你的这个单子在什么时间可用，星期几，几点几分到几点几分可以用，这个本身就是一个数据存储的项目。

当结构化之后，销售上单的时候，它就是一个表单，一个表格这么填写，最后生成的数据就不是一大段文字了，而是很多结构化的数据。这个数据有什么好处？比如说我们生成单子的时候，如果要改版，它很容易做一些改版，或者说我们商家要调整一个价格，就只把价格那个项目给商家来做调整就可以了。不用担心商家改的时候，把一些条款的其他东西改掉。然后还包括，比如说现在会在PC端和这个移动端同时显示同样的单子，其实因为显示器的差别，我们在PC端和移动端的显示肯定是不一样的。

只有当我们把它结构化之后，我们才能自动地匹配不同的显示环境，这就是结构化的一个好处。再比如说，我们会把一切可量化的东西量化，就是价格我们不希望输入一个字符串的几块钱，因为这个东西计算机是不容易去理解的，我们希望如果它是数字的，比方说价格，你就填一个价格，如果进价是多少，我卖价是多少，还有可使用时间，就用时间的格式来填，这样的好处就是我们的审核就变得非常容易。我们的审核团队，它根本不需要人工的去读这个东西，只要我们把规则制订好了，那计算机就可以把这个量化的东西一条一条地过一下。

所以当我们做了这些努力之后，我们新的流程变成什么样子了呢？销售团队同样还是要去填合同，但是他填写一个表格的数据，填写好后这个表格的数据，系统会自动地审查，自动地生成一个单子，然后立刻可以给商家做确认，商家确认之后，就可以上线给用户了。然后，做了这些努力之后，我们把中间的很多环节和步骤，包括人力都省掉了，我们不需要那么多编辑的同学，不需要那么多审核的同学，而且整个的流程会走得非常顺畅，非常快，便捷。



给大家看一下收益是什么呢？这是从今年1月份到9月份的一个数据，蓝线是我们每个月的上单量，从今年1月份，除了2月份，2月份因为是春节，整个的上单量有所下降，其余几个月份上单量一直是非常快速的往上涨的，一直到现在，每个月我们美团上的单子有40多万单不同的消费单，这种单子做一个假设，如果有一个吃喝玩乐的达人，他每天去吃一单美团的单子的，那40万单够他吃一千年。**假设我们这个单子能持续一千年，我们会提供非常非常丰富的单子给用户做选择，上这么多的单我们的单均成本，我们从原来的很高，现在已经降到了个位数。**如果我们没有做刚才那些努力的话，我们是根本不可能实现这么多的上单量，我们的成本也不可能降下来。因为我们把成本压的很低，就可以为商家提供更好的服务，也给用户提供更好的优惠，这就是技术给我们带来的一些优势。

最后第三点，给大家分享一下，在线上线下这两部分，技术都是可以去做的，我们说O2O，O2O是什么？就是Online+Offline，就是线下加线上。有一部分同学可能会有一些误解，可能技术只是在线上的，其实不是那样，技术它不分线上、线下，它在线上线下都是非常重要的，需要贯穿线上线下。我在这里给大家举两个例子，线上的就不举了，因为技术都是在优化线上的东西。给大家举两个例子，看我们怎么去用技术来优化线下的一些东西。

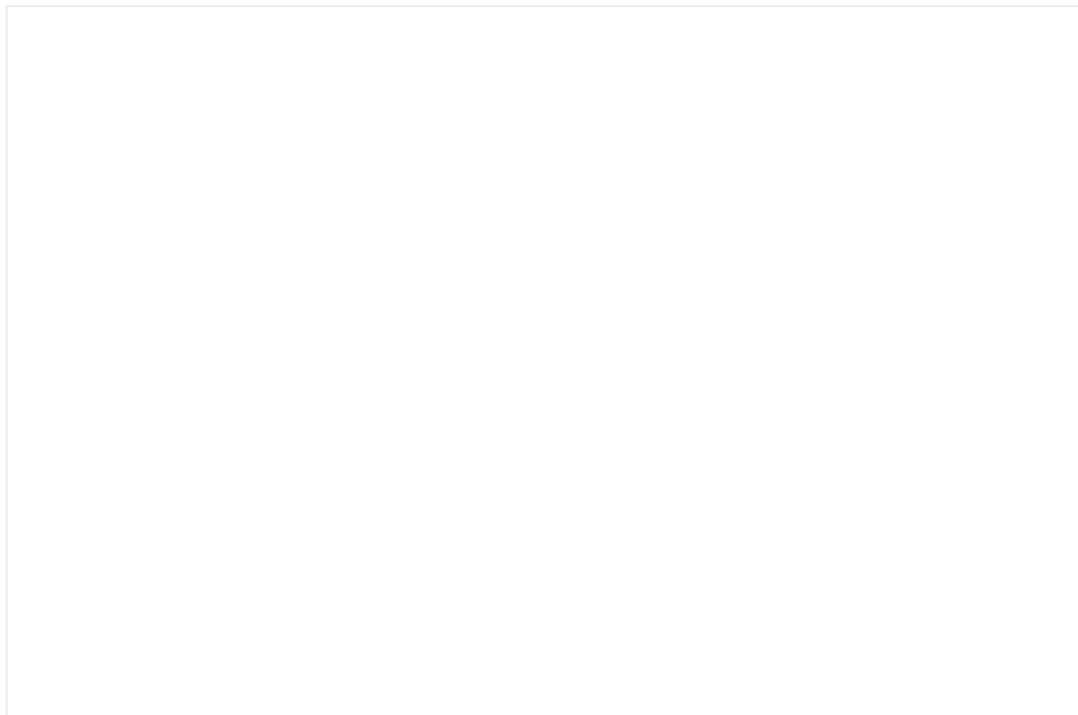


第一个就是外卖单子这个流程中，我们做了一个外卖打印机。先给大家介绍一下背景，美团外卖整个下单是怎么样子的？比如用户下单，在我们的网站上说，我要买一个砂锅饭，在哪个餐馆买，我要送到什么地方，我们就需要通知商家，通知商家的时候，要告诉他用户要买什么，他地址是什么，电话是什么，然后我们的商家的厨师去做菜，小二就要去送餐，用户完成消费。美团外卖是2013年的11月13号上线的，第一单的时候，最初的时候，我们怎么处理呢？那个时候真的很原始，当然我们一开始还是做了个手机的APP，用户在手机的APP上下单，我们会有美团的外卖的这个同学，包括一些客服的同学，也包括一些很多技术的同学也帮忙打电话，我们就打电话通知商家说，某某定了一个什么单子，然后他的地址是什么，电话是什么，告诉商家。

商家怎么办？就只能用这个纸笔记下来，交给厨师，说你去做饭，厨师做完了，把小纸条给配送的同学，配送的同学就去给用户来送东西，这个里面过程是非常痛苦的。当然那时候大家做的很有热情，非常喜欢打电话，看到我们有用户进来，一天从一单十几单到几百单，大家打电话打得很兴奋。但是很快发现承受不了，因为打一单，对美团的开销是说，一单哪怕只需要一分钟去给商家打电话，几百单的时候还行。一天一个人8小时工作，哪怕你持续不停的给商家打电话，那一分钟打一个也就四百多单，你可以帮用户定四百多单。但是当这个单子增长很快的时候，我们就发现我们人力跟不上了，技术的同学都去打电话了，技术同学受不了。商家也很痛苦，因为打电话的时候，很多的信息是说不准确的，他要记电话号码，如果记不清的话，他还要再打过来问刚才那个单子电话是什么，包括地址，有些地址字还比较难写。

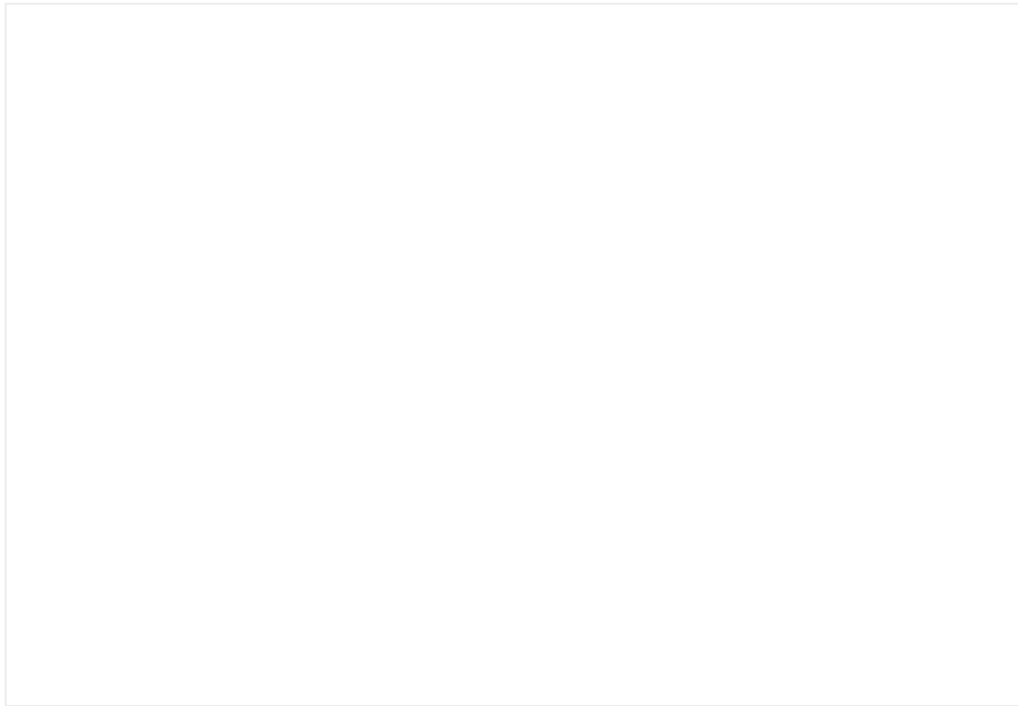


我们为了解放，首先我们解放我们自己，我们不给商家打电话了，我们给商家开发一个APP，每个商家只要安装了这个APP，他在手机上就可以接到通知。一旦有用户下单，APP上就会有通知说谁谁下了一个单，这时候老板根据APP上的信息，拿张纸，找个笔，把它写下来，然后给厨师。这个时候你会发现说，至少美团的同学这个工作就省下来了。但是商家的工作，他虽然抄得准确一点，不用从电话里抄，直接从APP上抄，但是还是要做大量的工作，包括这个小纸条传来传去，有的人写得笔迹不是那么清楚。



后来我们再进一步，我们APP接到一个打印机上，这个APP有可能是一个手机的APP，也有可能是电脑上的一个应用程序，这个APP连着打印机，一旦用户下单，打印机就会打出

一个小条来，他拿这个小条，这个信息就非常准确了。商家不用花时间去写了，这时候就可以给厨师去做菜，厨师做完了交给配送员去送菜，这个就比较方便了。后来我们还是发现这个也不是很好，一般这个APP总是在老板那里的，如果说好几个人拿着APP会出现问题，好几个人拿的话，比如老板和老板娘都拿着，他们可能都去打，一个定单可能打了两份；或者说，有时候说老板拿着这个APP，但是老板刚好不在，那他就打不了。有的店，他就不得不买一台电脑放在那，但是电脑对于很多小的外卖店，还是一笔额外的开销。因为现在手机上网的多，但是电脑上网的人已经很少了，还是对商家有很多不方便的地方。



今年5月份的时候，我们的硬件的团队，他们就说我们自己来做一个云打印机。所谓云打印机就是它自己可以联网，联网的时候有很多联网的方式，比如说我可以插一个手机卡，通过手机的网络来联网，或者我也可以通过WiFi来联网，联网之后，手机下单的时候，我们的美团后台的服务器会把这个单子的信息直接推送到这个云打印机上，这个云打印机就像一个POS机那样，是一个很小的设备，会自动地打印出单子的信息，这样就大大得解放了我们美团的同学和商家的同学。这是一个我们用技术来优化商家端这边的流程的一个例子。

接下来给大家举一个，我们怎么去用技术优化用户端的例子。我们在用户端也有很多线下的工作，其中一个工作就是用户运营。我们有一种需求叫“拉新”运营团队，他们的任务就是对一些已经注册了美团的帐号，但是过来逛了一圈，最后没买东西就走了，可能来逛了好几次，他还是没买，这时候美团就急了，美团急了怎么办呢？给你10块钱，你赶紧买一个吧。因为很多用户的确是这样，他没买可能是因为他不知道网上怎么支付。他支付流程没做通，或者说他可能就不太习惯，所以我们希望把这些用户转化成一个习惯于在网上消费的用户，让他体验一下，可能体验一下他觉得好，他可以以后接着买。给用户10块钱，

20块钱，做一些优惠活动，吸引用户完成首次的购买。那其实我们这边的花销是真金实银的，我们是给用户很多免费，对于美团来说**薄利多销，利很薄**，所以我们希望少花钱，多办事，这个钱能少给就少给，能不给就不给，运营的团队，就跟我们技术的同学聊，问这个事情能不能优化？

我们就去分析，先去分析这些用户到底是什么样的用户，发现用户有很多类，一类是有一些用户他虽然来逛了一圈，或者逛了几圈，他还没买，但是假以时日，可能再过几天，最后他还是会自动的转化，所以有一些用户的这种自动转化的可能性是比较高的。还有一些用户，他可能来了几次，他可能每次就是来逛逛，每次逛逛，就像逛街一样，虽然他不买，但是逛着也很爽，看着菜单他可能口水直流，也觉得挺爽。所以有些用户，就是你不给他刺激，他就完不成自动的转化。然后还有一些用户质量低，所谓质量低就是对美团的质量，我给他券的时候他就过来买个东西，比如我给他10块钱，他可以过来买个11块钱的东西，然后转身就跑了，我不给他券，他就半年不过来，美团又忍不住，又给他10块钱，他又过来看了看买了个9块钱的东西就走了。对这种用户不是说质量低，对美团来说当然我们希望尽量这种薅羊毛式的这种用户，他来当然我们也欢迎，但是不来我们也就不要去拉拢了。

还有一些用户就是，属于用户质量比较高的那些用户，如果我们一旦帮助他转化，就是越过了这个首次消费这个坎的话，他可能就成为一個很高频的，习惯于在美团消费的用户，我们叫高质量用户。对于美团来说，我们希望真正运营的对象，真正拉拢的对象就是在这些，我如果不给他刺激，他自己可能转化不了，然后同时如果他转化了，他对美团的销售额的贡献会比较大。这种用户，对其他三个象限，比如说第一象限里边，这部分用户我不需要给他发券了，发券就是对美团来说浪费钱了，我们叫浪费，当然也没浪费。对下面这两个象限呢，因为用户本身质量比较低，所以他来和不来至少对美团的最后的获利来说没有大的关系，那我们怎么去找这部分用户？这是运营的同学给技术团队提出了一个要求，那我们怎么去做呢？



做用户画像，在美团其实有大量的用户的数据。我们可以看到，他什么时候注册的美团，他从什么浏览器，从什么操作系统，然后包括说，他去看了哪些单子，浏览了什么单子，他做了什么搜索，他用的是什么手机，这些信息我们都有。有大量的数据，那我们从这些数据里面就给用户画像，我们会去判断用户的各种属性，这些属性有一些是，可能用户在注册的时候就会告诉我们的了，比如说性别，年龄，包括邮箱，这些可能本身他注册的时候就会告诉我们，还有一些是需要我们去分析，比如说这个是否结婚了，他的职业是什么，然后包括他的地理位置，地理位置当然比较好拿，因为很多，现在我们绝大部分用户是在手机上的，他一旦在手机上做周边的这种搜索，我们就可以拿到他的地理位置，地理位置的话，我们就可以知道，比如说他是一个学校的用户，是学生，还是说一个工业园区的一个白领，还是一个比如金融街的一个比较高大的一个这个金融界人士。邮箱也很重要，比如说我们发现用户是qq.com的这种用户，跟gmail.com的用户，他是两类不一样的用户，他所关注的东西，感兴趣的东西都不一样。

我们也会去分析用户的收入水平和消费水平，收入水平就是我们可能根据他的职业，根据他的年龄的工作年限去判断，他是在一个高收入，还是低收入，然后消费水平相对来说，有可能他虽然收入很高，但是他过来看的单子都是一些比如打折打得很狠，比如吃一顿饭人均20块钱这种也有。所以收入水平和消费水平我们是分开来判断，然后会判断这个用户是倾向于最优惠的价格，他还是讲求比较高的质量，就是订酒店的时候是99块钱的酒店，还是说899的五星级酒店等等。

我们会去做一些分析，我们根据用户他浏览的历史，他去看了哪些单子，他去做了什么搜索，我们就可以知道他的品类偏好，他到底是喜欢美食，还是喜欢去看电影，还是喜欢去

唱卡拉OK，这种娱乐，这些也可以判断出来，再加上一些正常的统计的数据，比如说用户搜索了多少次了，然后他去浏览了多少次，就是来逛街逛了多少次了，他是不是有收藏一些他比较喜欢的一些单子等等，这些很多很多的属性加起来，我们就建了用户画像的这种数据库，然后有了这些属性，那我们接到运营同学的需求的时候，我们就会对用我们的用户画像里的数据先做特征的选择，做特征的提取，甚至有一些属性，比如说年龄这个属性，我们会做很多特征的离散化，就是把一个属性离散成几百个，甚至几万个这种属性，放到我们的机器学习的模型里面去训练一些模型，最后我们用一些分类模型去，其实最后我们选的是SVM，就是支持项连机这种机器学习的算法，训练出的模型，它的准确度相对来说，看起来没有那么高，但是效果已经很明显，沃勒准确率是75%，判断用户是不是真的在左上角那个象限里，准确度75%，召回率能达到68%。有了这个技术支撑，最后我们运营的单位成本，比如说我们拉新运营的单位成本就可以降低35%，总共的拉新运营开支就节省了30%。

所以这个就是我们在技术，在线下的用户运营这边做的一个努力。大家可以看到刚才举了两个例子，一个是在商家端，一个是在用户端，其实不管怎么样，我们在线下技术可以做很多很多的优化，让我们线下的效率做很多的提升，那正是所有这些一个一个的优化，加起来就可以把美团的整个的生产效率提升上去，把我们的成本降下来，这样我们最后的获利就是美团越来越有竞争力。

今天给大家分享的一些小的例子，最后总结一下，把它总结成几点。第一点，就是的确我们的**技术架构是要随着业务的发展而不断地变化的**，不一定一个复杂的的一个完善的架构就是好的，而是看，我们现在的公司是处在什么阶段，如果是一个初创的公司，那大家还是要小步快跑，一个简单有效的方案就行了。第二我们是**随着业务发展，在一些开源软件的技术上不断的优化，业务流程我们也不断的做一些标准化，自动化，就是刚才给大家讲了我们四个原则，就是把复杂的东西简单化，简单的东西标准化，标准的東西流程化，流程的东西再自动化**，这是我们在业务流程方面的一个优化的原则。

第三个就是在技术上的，技术不仅仅对线上有用，在线下的每一个操作里，我们在不停地看我们能做一些什么样的优化。最后一句话就是，一个简单的事情，哪怕像美团这样，很多人觉得简单的事情要把它做到极致，就是真正做到极致，也会有足够多的这种技术挑战，足够高的门槛，所以我觉得现在很多O2O的创业的团队来说，他们做得事情看起来非常简单，但是只要大家不断地优化，不断地极致，不断地朝着极致去发展，你就可以在竞争中胜出。这个就是给大家分享的美团的技术团队，我们做的很多的努力。

编者按：本文是根据ArchSummit北京2015大会上美团网现任技术委员会主席夏华夏的演讲《从技术细节看美团架构》整理而成，感谢Emily的整理。

阅读原文