

02 | 业务架构：作为开发，你真的了解业务吗？

2020-02-24 王庆友

架构实战案例解析

[进入课程 >](#)



讲述：王庆友

时长 14:54 大小 11.95M



你好，我是王庆友，今天我们一起聊聊业务架构。

作为开发人员，我们平常讨论比较多的是技术层面的东西，比如 Spring 框架、Redis 缓存、MySQL 数据库等等，我们喜欢讨论这些，是因为纯技术的东西比较通用，和业务相关性不大，沟通起来比较方便。

但你要知道，一个项目能否成功落地，首先需要的是把业务分析做到位，至于选用什么技术来实现，这是我们第二位才去考虑的因素。**从架构角度看，业务架构是源头，然后才是技术架构。**所以，作为专栏的第二讲，今天我们就从业务架构开始说起。

在软件开发的过程中，你肯定知道需求分析是怎么回事，但不一定知道业务架构设计是怎么回事；你也肯定清楚需要产品经理这个角色，但不一定清楚有时还需要业务架构师这个角色。关于需求分析和业务架构设计，相信你经常会有以下几个疑问：

1. **业务架构师和产品经理有什么区别？**

2. **需求分析和业务架构设计有什么区别，业务架构到底有什么用？**

我们知道，项目的开发都是从收集业务需求开始的，原始的需求一般来自于最终用户。但是，每个用户其实只清楚自己所负责的那部分，因此这些原始需求往往是零散和碎片化的，特别是当一个业务流程跨多个部门的时候，更没有一个人能够说清楚这个业务的全貌。

所以说，仅仅基于这些原始的需求来指导开发是远远不够的，这时，就需要产品经理和架构师介入进来，填补这段空白。

接下来，我们就一起看下，产品经理和架构师在这个过程中都会做些什么，他们是如何帮助业务落地的。

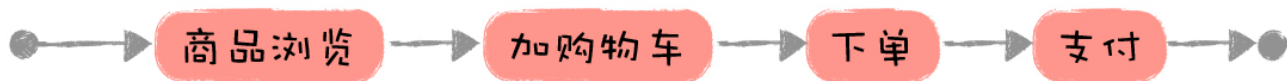
产品经理的职责

简单来说，产品经理的职责就是：**告诉用户，系统长什么样子；告诉开发，他要实现什么功能。**

产品经理首先会收集用户的原始需求，然后，将它们梳理成一个个业务流程，每个业务流程由多个业务步骤组成。一个业务步骤包含三部分的内容：输入、输出和业务功能。

比方说，一个典型的交易流程，它包含商品浏览、商品加购物车、下单、支付等步骤。其中，下单步骤的输入，就是订单的各种信息，下单的功能，就是整合这些信息，创建一个具体的订单，而下单的输出结果，就是新创建的订单。

购物流程



需求梳理好后，产品经理会把每个步骤具体化为页面原型。在原型中，会以直观的方式给出各个步骤的输入或输出，以及用户的操作过程，最后再把这些页面串起来，形成一个业务流程。

你可以看到，经过产品经理的工作，大量零散的原始需求经过梳理和关联，变成一系列有序的业务流程，以及流程里面的业务步骤（业务步骤也称之为业务节点），然后产品经理把这一系列的业务流程和业务节点以用户界面的方式定义出来，总的来说，产品经理定义了系统的外表。

这些产出对于用户了解系统长什么样子，应该如何使用这个系统，以及系统是否满足他们的需求来说，是足够的，但对于开发者来说还远远不够，因为他们需要能进一步看到系统的内部结构。

而这一步，就是业务架构师要做的事情。

业务架构师的职责

在这之前，我们不妨先思考下，如果是按照产品的输出，直接以业务流程的角度来构建系统，会是什么样子呢？

如果按照这个思路，我们将为每个业务流程搭建一个对应的系统模块，然后业务流程中的每个业务步骤，将对应系统模块中的一个接口，包括它的功能、输入和输出。

就拿前面的购物流程来说，我们设计一个购物流程模块，里面包含商品查询、添加购物车、下单和支付接口，来分别对应流程里的 4 个业务步骤。

以这样的方式构建系统，表面上看起来，业务和系统的映射好像非常简单，但在实际中，落地的难度非常很大。因为只是这样一个小小的购物流程模块，就要同时涉及商品、购物车、

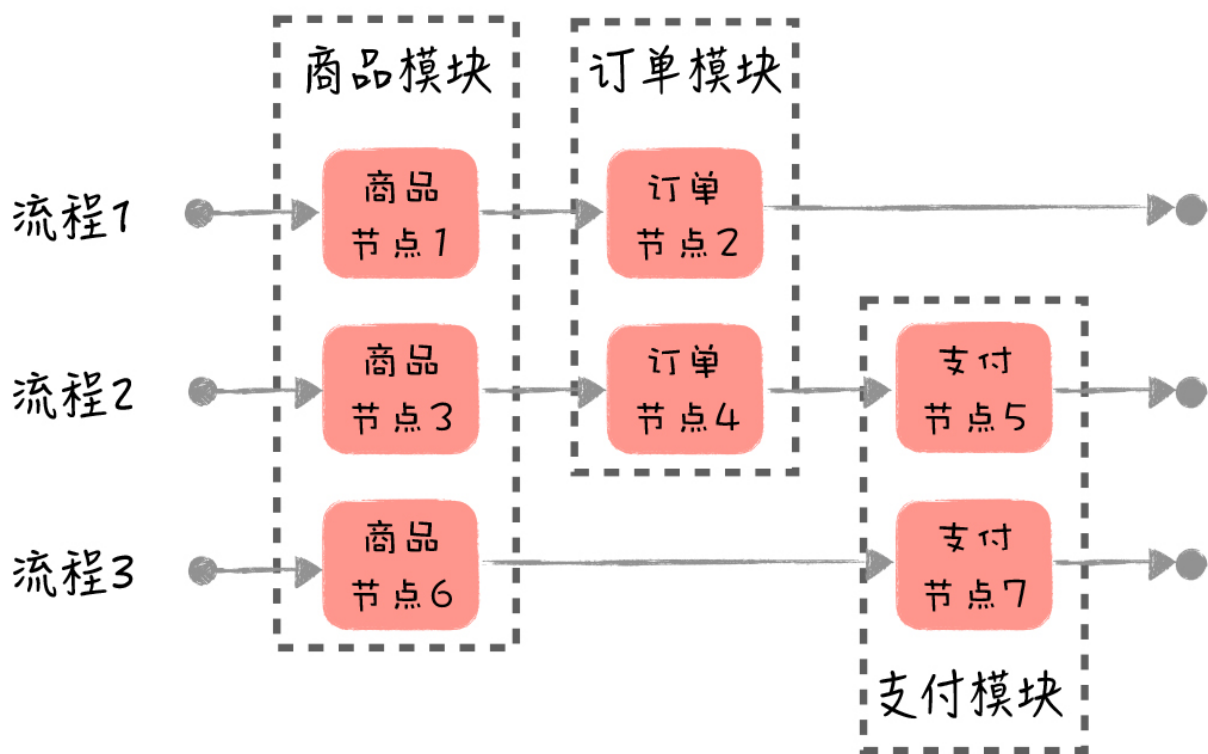
下单和支付四个业务，模块的开发要同时非常清楚这四部分的数据模型和业务逻辑。

同样的道理，系统里的其他模块也是包含多个业务领域的内容，如果一个业务领域的需求发生了变化，比如说，订单要增加一个新的状态，那么所有涉及该订单的模块都要知道这个变化，并要做出相应的调整。这就要求，每个开发者都是全知全能的，对所有业务都了如指掌，我们知道，这是不可能的。

每个业务都有其本身的专业性，比如订单业务、商品业务、支付业务，它们的数据模型和业务逻辑都相当复杂，构成了一个个相对独立的业务领域。如果我们是按照业务流程来划分系统模块，结果是把不同业务混在了一个模块里，所以，这种模块划分的方式并没有降低总的业务复杂度。

我们可以换一种做法，先把所有的业务流程拆散，这样得到了一堆业务节点；然后把业务节点进行归类，相关的业务节点放在同一个系统模块里。判断节点是否相关，主要看它们是否属于同一个业务领域，比如一个订单查询的节点，和订单创建的节点，它们都属于订单域，那么这些节点都可以归属到同一个订单模块里。

下图就清楚地表示出了系统模块按业务流程拆分，和按业务域拆分的不同。



如果按照业务流程来拆分系统模块，那么，有多少业务流程，就有多少个系统模块，这个对应关系比较直接，但实现起来很困难。

如果按照业务域来拆分，有多少业务领域，就有多个系统模块，流程中的业务节点按照业务域的不同，可以划分到不同的系统模块。

在实际业务场景中，一个业务节点可能会涉及不同业务领域的功能。比如说，一个下单节点，会涉及到获取商品信息、获取用户信息、扣库存、下订单等多个业务功能，那么你就可以进一步分解这个节点的功能，把不同的功能分到对应的业务域和系统模块。

基于业务域，构建了系统模块后，我们就可以按照这样的方式还原整个业务流程，比如上面的购物流程例子，我们就可以这样还原它：

购物流程 = 商品模块. 商品搜索 + 购物车模块. 添加商品 + 订单模块. 创建订单 + 支付模块. 支付

如果你把这个定义画成序列图，就很直观和容易理解，也比较符合开发人员思维，系统实现起来非常容易。通过这种系统模块之间的不同功能组合，我们很容易给出各个业务流程的定义。

所以，对业务架构师来说，TA 的工作，就是把业务流程和节点打散，按照业务域的维度来划分系统模块，并定义这些模块之间的关系，最终形成一个高度结构化的模块体系。这样，开发人员既容易理解业务，又方便落地系统。

现在，我们就可以回答文章开头的问题了，**产品经理和业务架构师的工作既有区别又有联系，简单地说，产品经理定义了系统的外观，满足了用户；业务架构师在此基础上，进一步定义了系统的内部模块结构，满足了开发人员。**

当然，满足现有的业务需求，保证系统顺利落地，这只是业务架构的最基本目标，业务架构的意义远不止于此，它有一系列更高的目标，下面，我就逐一为你展开介绍。

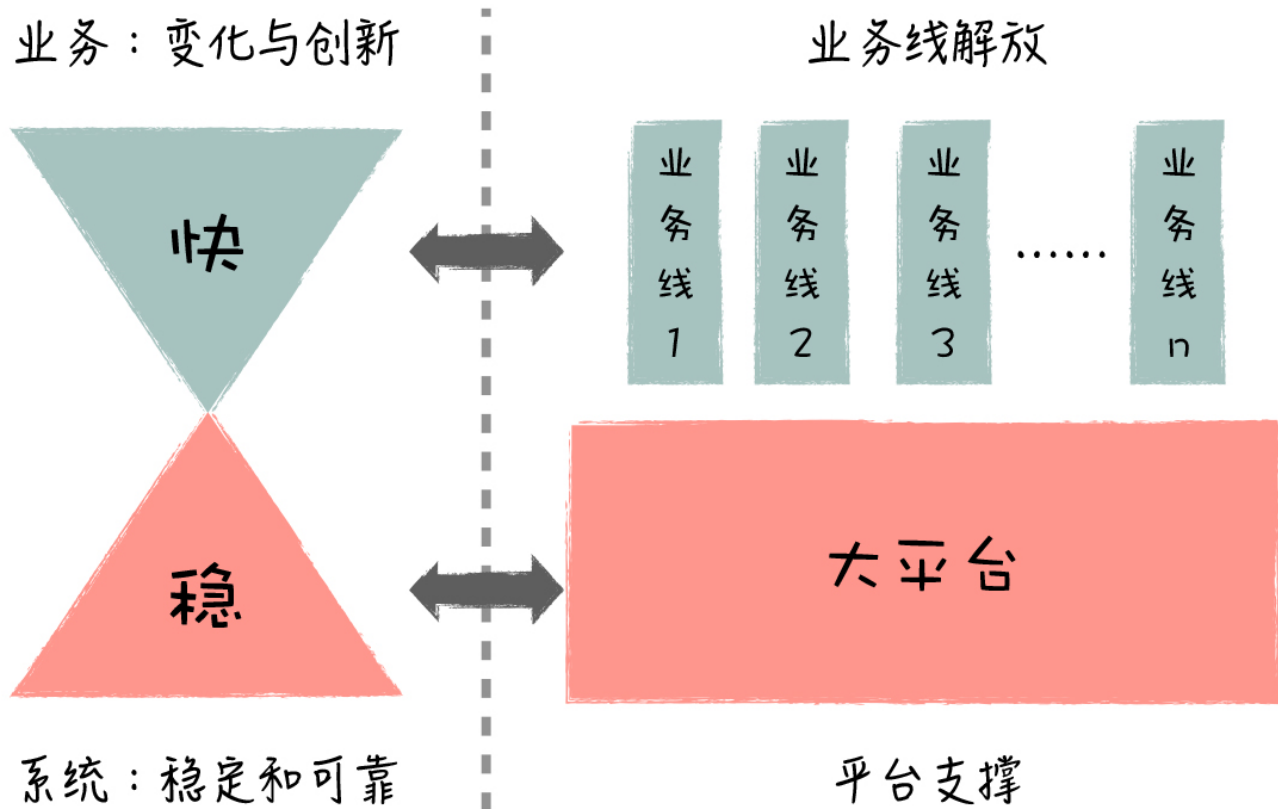
架构目标之一：业务的可扩展

第一个目标是业务的可扩展，我们都知道，业务需求是不断变化的，不断创新是业务的内在要求。而对于系统来说，它的要求却是相对稳定，尽量避免大的调整。

那么，我们如何才能实现业务的快速变化和系统的相对稳定呢？

这也是业务架构要重点解决的问题，具体地讲，业务架构设计要能支持打造一个柔性系统，通过提供良好的业务扩展性，允许业务不断调整和快速生长。

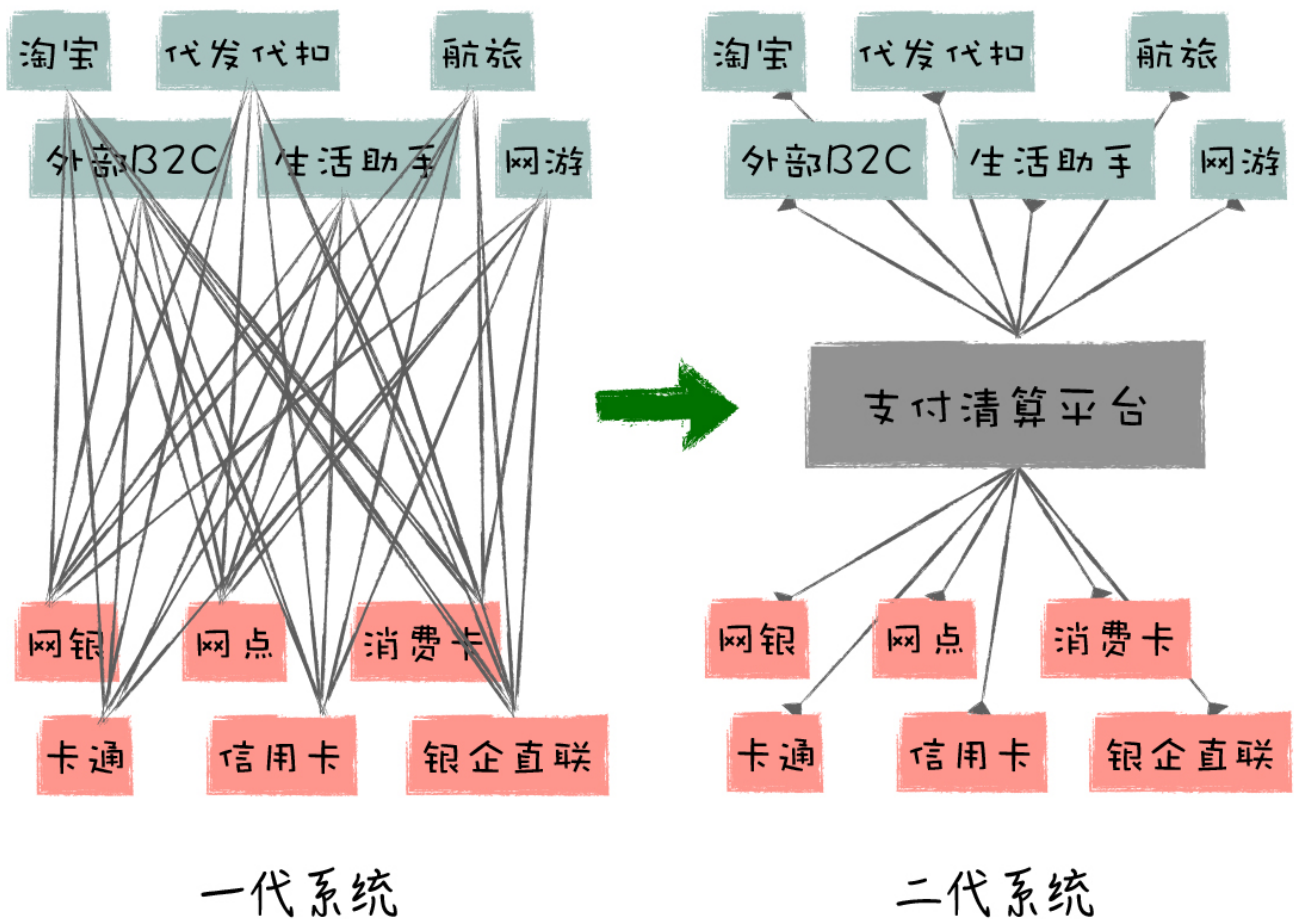
可以看到下图中，左边部分就比较形象地展示了业务和系统的不同特点：**业务的主题是变化和创新**，**系统的主题是稳定和可靠**。



在右边图中，我们通过通过巧妙的业务架构设计，很好地解决了业务和系统之间的矛盾。

这里，我们把业务平台和业务线剥离开，让业务平台封装基础通用的功能，这样，它就变得相当地稳定；让各个业务线包含自己的个性化需求，业务线只依赖业务平台，业务线彼此之间互相独立，可以自由变化。这样的业务架构设计，就同时保证了系统的相对稳定和业务的快速创新。

为了帮助你更好地理解业务架构的扩展性，这里，我给出了支付宝的业务架构变化过程。



在支付宝一代的业务架构中，前台的业务和后台的业务直接耦合，形成了多对多的网状结构，如果修改一个后台业务线，就会影响到很多前台业务线；如果增加一条新的前台业务线，需要同时和很多后台业务线对接，这样的架构无疑是对业务的扩展非常不利的。

而在支付宝二代业务架构中，你会发现，他们在前后台业务线之间，构建了独立的支付清算平台，从而实现了前台业务和后台业务的解耦。

在这里，不管前台业务，还是后台业务，都只需要对接中间的支付清算平台，把系统的变化收敛到一个点，而业务线之间相互不影响，这样的方式，自然可以很好地支持业务扩展。

好了，这里我们说完了业务架构的可扩展目标，接着再说说业务架构的另一个目标：可复用。

架构目标之二：业务的可复用

你肯定会有这样的体验：一个项目过来，你和伙伴们一起加班加点、紧赶慢赶，总算把它成功落地了。结果这时候又有另一个类似的项目过来，你们又要按照同样的方式，重新吃一遍

苦，结果就是开发不满意，项目经理不满意，老板也不满意。

对于类似的业务需求，如果原来做的工作可以大量复用的话，这是非常理想的结果，无论对于开发效率和开发质量的提升都很有帮助。

当然，能不能复用，能在多大程度上复用，这和业务架构设计很有关系，也是业务架构设计的重要目标之一。

那么，业务架构设计如何实现业务的可复用呢？

你可以试想一下，在业务架构设计中，如果只是简单地基于业务流程来定义系统模块，这个系统模块就要和业务流程严格对应。我们知道，业务流程对应业务场景，而业务场景是经常变化或是定制的，这就导致系统模块也是经常变化和定制的，那么，这样的系统模块就很难在不同业务场景中复用。

如果我们按照业务域来划分业务，把业务流程中的节点拆分到各个业务域，按照业务域构造系统模块，这样的复用效果会如何呢？

我们都知道，业务域是相对固定的，它有明确的数据模型和业务规则，这样一来，系统模块也就比较固定和通用，也就具备比较好的复用基础。

但要想实现高复用，业务架构对系统模块的定义，还有更多的要求。

首先，模块的职责定位要非常清晰。对于模块来说，在定位范围内的职责要全部涵盖到，而不在这个范围的职责全部不要。

其次，模块的数据模型和接口设计要保证通用。架构师需要归纳业务场景，通过抽象提炼，形成通用化的设计，以此来满足多个类似场景的需求。

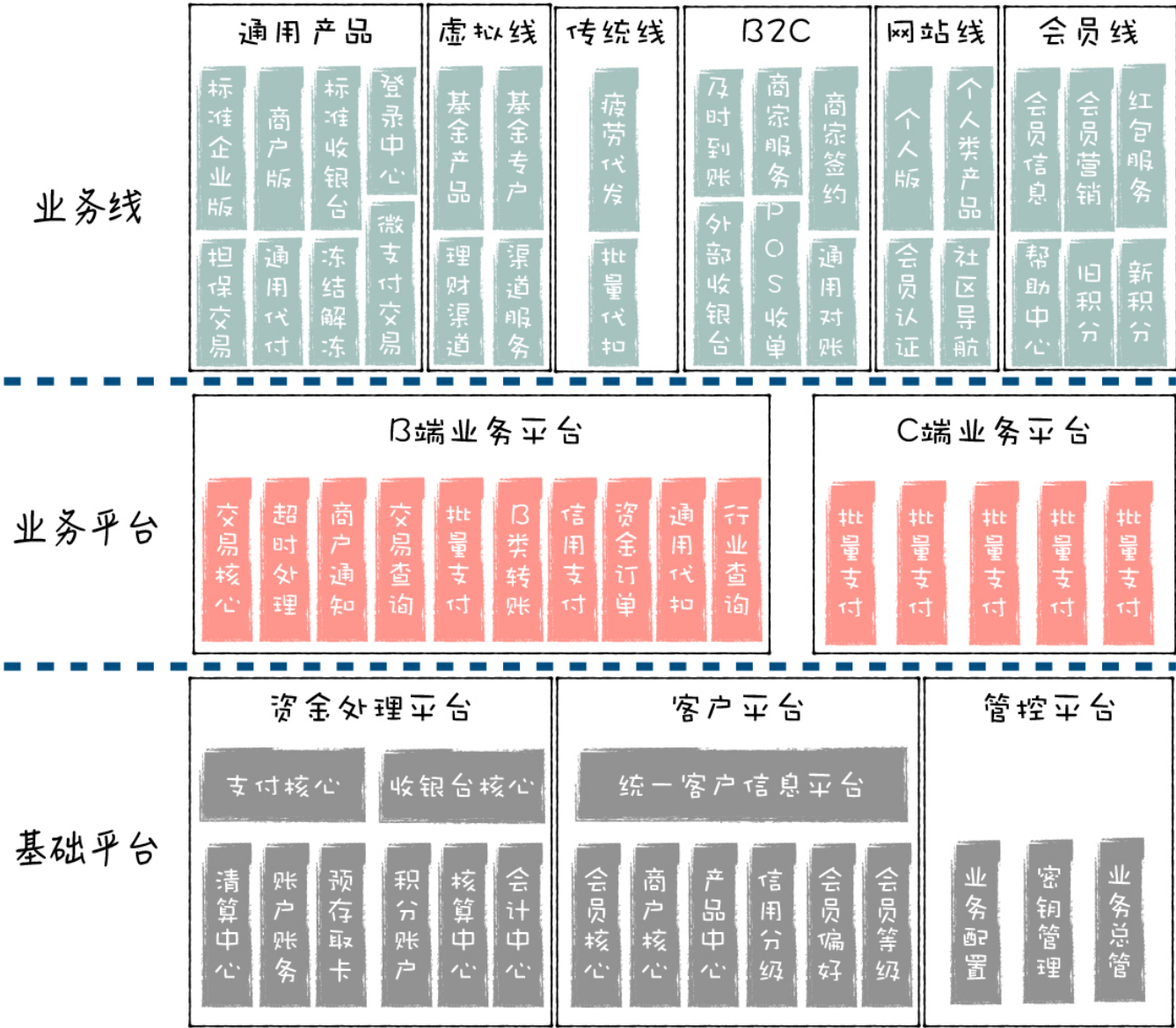
小提示：清晰的模块定位和通用化设计，是模块能够复用的内在要求。

最后，实现模块的高复用，还需要做好业务的层次划分。我们知道，越是底层的业务，它就相对更固定。举个例子，同样是订单业务域，对于底层订单的增删改查功能，不同类型的订单都是一样的，但对于上层的订单生命周期管理，外卖订单和堂食订单可能就不一样。

所以，在做高复用设计时，我们可以尝试把一个业务域按照层次拆分得更细，比如，把订单模块拆分为多个上层订单模块和一个基础订单模块，这样，基础订单模块对于所有类型的订单，都能够提供复用。

就拿当前非常流行的微服务架构来说，很多公司在微服务的基础上，通过服务分层，进一步落地了共享服务体系和中台架构，这些都是业务架构复用能力的体现。

下面是一个三方支付平台的业务架构图，你可以看下，在一个实际的业务架构中，模块是怎么划分的，架构的可扩展和高复用是如何体现的。



总结

今天，我带你了解了产品经理和业务架构师的不同职责，产品经理是站在用户的角度进行需求分析，而业务架构师是站在开发者的角度定义系统内部结构。通过今天的讲解，你应该对

业务架构也有了更清楚的认识。

除了满足当前的业务需求外，业务架构师还需要面向未来，实现业务的可扩展和高复用两大目标，我也大致介绍了架构师实现这些目标的思路。在接下来的文章里，我还会针对这两大目标，结合实际案例，具体讲解如何实现它们，让你能更加深入地理解业务架构设计，并可以在工作中学会去运用这些手段。

最后，给你留一道思考题：产品经理和业务架构师都是分析业务，产品经理为什么不能兼业务架构师的角色？他们的能力模型有什么区别？

欢迎在留言区和我互动，我会第一时间给你反馈。如果觉得有收获，也欢迎你把这篇文章分享给你的朋友。感谢收听，我们下期再见。

点击参与 

20年架构老兵邀你一起
打卡，带你进阶资深架构师



扫一扫参与小程序话题



新版升级：点击「 请朋友读」，20位好友免费读，邀请订阅更有**现金**奖励。

© 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。

上一篇 01 | 架构的本质：如何打造一个有序的系统？

精选留言 (13)

 写留言



孙泽勇 置顶



2020-02-24

整理输出内容: <https://www.processon.com/view/link/5e51378ce4b0c037b5f9d1e3>; 关于文中思考题, 我想业务架构师还是需要技术背景的, 按照概述篇讲解, 最起码写的一手好代码, 而产品就不需要了, 除非这个产品是开发转过来的, 产品能搞出清晰的prd应该就很棒了。产品和架构负责的是同等重要的事情, 若这个人nb就另当别论了。

展开 ▾

作者回复: 整理得很用心, 赞一个。业务架构师需要文武双全, 既懂业务, 懂技术, 懂技术和业务怎么结合。

1



Din 置顶

2020-02-24

总结:

1. 进行微服务拆分时, 需要根据业务领域来确定服务边界 (DDD)
2. 当业务越来越复杂, 前台业务场景越来越多, 为了保证系统的服务的扩展性和可复用, 在微服务的基础上, 可以通过服务分层, 进一步落地中台架构

...

展开 ▾

作者回复: 你这个理解得很好, 就微服务来说, 确定需要划分为几个服务, 各个服务边界是什么, 有了这么多的应用和服务后, 系统怎么分层比较好, 这个是业务架构要做的, 产品经理是搞不定这些的。

至于选用什么微服务框架, 是dubbo还是spring cloud, 这是技术架构层面的事。

1



Rory

2020-02-24

产品经理更多是聚集商业目标

业务架构更多是行业或者领域专家

产品经理根据商业, 用户群体, 建立用户画像, 业务场景, 以及满足业务的功能

业务架构师则根据业务 进行领域建模等 梳理各业务直接的关系等

展开 ▾

1



知莫问

2020-02-24

产品经理侧重于对现实世界的理解，业务架构偏向于对软件世界的理解。



1



约书亚

2020-02-25

思考题正是我想问。按照文章描述：

“先把所有的业务流程拆散，这样得到了一堆业务节点；然后把业务节点进行归类，相关的业务节点放在同一个系统模块里。判断节点是否相关，主要看它们是否属于同一个业务领域...”

为什么产品经理做不了这项工作呢？看起来只要按照“名词”划分就好了.....

展开 ∨



snowball

2020-02-25

老师好，请问产品经理梳理的业务流程和图2(右侧业务支撑)中的业务线的概念有什么区别呢？如果没区别，那么多个业务线在拆分业务域的时候，有共同的业务域时，该业务域对应的代码模块在图2业务支撑的什么位置呢

展开 ∨

作者回复: 业务线是个大的概念，比如美团有外卖业务线，打车业务线，酒店业务线等，一个业务线包含多个业务流程，比如外卖订单有它的流程，堂食订单有它的流程。如果多个业务线共享业务域，比如订单是通用的，那这个业务域可以下沉，变成一个共享的基础业务域，不属于那个业务线。



1



可乐

2020-02-25

老师，文中说到的序列化大对象不是很清楚，它是指把一个值对象集序列化后存储在实体数据库的一个字段中么？

展开 ∨

作者回复: 这篇文章有讲到序列化吗？有点不记得。

当你在java中调用一个远程方法时，需要把内存中的java对象转化为网络传输需要的01字节流，这个是序列化过程，在接收端，需要把01字节流转换为内存对象，这个叫反序列化。网上文章有很多介绍，你可以具体看下。





Alex

2020-02-25

产品经理聚焦用户，以用户需求为中心打造用户体验。业务架构师面向实现，保证内部结构合理稳定，保证扩展性，提高复用性。

展开 ▾



怀朔

2020-02-24

合理的抽象 对业务有前瞻性！！ 对技术架构师纬度又不一样。。



每天晒白牙

2020-02-24

真的有业务架构师这个职位吗？我们这边是产品经理提完需求，我们开发自己梳理，自己设计，然后开发。组内有个架构师，但他应该是技术架构师，还真没接触过业务架构师

作者回复: 实际上你说的架构师兼了业务架构师和技术架构师职责。而且更多偏业务一些。结合业务，划分服务边界，确定应用和服务之间的调用序列图，这些都是业务架构师做的事情。当然最后也会确定技术选型，以及如何部署等。纯粹的技术架构师是设计技术组件，在基础架构团队里面。当公司比较小，研发经理兼架构师角色，公司大到一定程度，业务架构师和技术架构师就会有专门的角色。



秋天

2020-02-24

这里面所说的业务架构师，也就是通常意义上咱们口中说的架构师把？换句话说，就是高级开发工程师 要有架构师的思想，可以自己抽象业务模块，用来做系统架构的分析

作者回复: 架构分业务决策和技术决策，通常职位设置上，没有分那么清楚，实际的架构师这两方面都会做，甚至有时没有专门的架构师职位，像你说的开发骨干会或者技术leader也要做这个事情。



Kăfkă²⁰²⁰

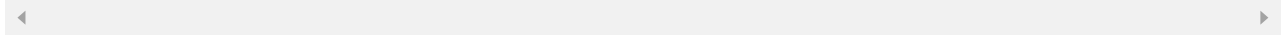
2020-02-24

产品经理需要的能力主要偏向用户和商业模式，所以常说要有同理心、洞察人性、懂得用

户心理和行为、了解市场。而业务架构师要能把业务划分出领域，业务由领域中不同的点串联成流程，便于团队组织用最小代价实现，需要有强的分析能力，找到系统中的变与不变

展开 ▾

作者回复: 说的很好。



Jeff.Smile

2020-02-24

产品经理站在客户角度梳理需求，业务架构师站在开发者角度划分领域形成模块。

