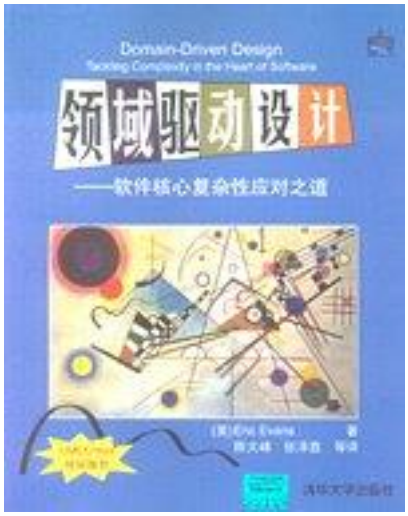
领域驱动设计



领域驱动设计_下载链接1_

著者:[美] Eric Evans

出版者:人民邮电出版社

出版时间:2016-6-1

装帧:平装

isbn:9787115376756

本书是领域驱动设计方面的经典之作，修订版更是对之前出版的中文版进行了全面的修订和完善。

全书围绕着设计和开发实践，结合若干真实的项目案例，向读者阐述如何在真实的软件开发中应用领域驱动设计。书中给出了领域驱动设计的系统化方法，并将人们普遍接受的一些实践综合到一起，融入了作者的见解和经验，展现了一些可扩展的设计新实践、已验证过的技术以及便于应对复杂领域的软件项目开发的基本原则。

作者介绍:

Eric Evans “领域驱动设计之父”，世界杰出软件建模专家。他创建了Domain Language公司，致力于帮助公司机构创建与业务紧密相关的软件。他在世界各地宣讲领域驱动设计（Domain-Driven Design，DDD）的思想，开设课程，参加会议，接受专访，拥有大批的追随者。从20世

纪80年代开始，他就以设计师和程序员的双重身份参与过许多大型面向对象系统的设计和开发，涉及各种复杂的业务和技术领域。同时，他还培训和指导过许多开发团队开展极限编程实践。

目录: 目 录

领域驱动设计_下载链接1_

标签

领域模型

软件工程

软件架构

DDD

设计模式

计算机

架构

领域驱动设计

评论

经典之作。边看边哭，一边感动于作者的切中肯綮，一边被译者的生搬硬造气哭。不推荐读！上豆瓣看了下这三位译者翻译过的书，简直博学多才。

这一套方法论是建立在面向对象 单元测试 重构 设计模式等等之上的
对于当下各互联网厂中的项目来说
很多都成长不到能够用这些招式来加以改善的阶段吧

领域驱动设计的思想是非常有价值的，但是本书成书太老，例子也是大多数人不熟悉的会计和货运领域，所以很难搞懂应该如何把领域建模应用到实际开发中。我觉得要写好领域驱动设计这个主题，好的例子真的非常重要。

翻译有点烂，观念有点旧，只能看其思想

常读常新

这种书就是当你还不会编程时候读不懂，会编程时候不需要读。总之就是什么时候读都不会提高自己水平。不花点时间读又不会知道不好。

本书与之前读过的其他技术书差异很大，它的主题是针对大型复杂行业或领域软件开发的一些思想。书中重点提出了很多概念，这些概念大多都很抽象，所以读起来会比较吃力。这本书也需要经常回味，一次性读完的话很多东西也吃不透。
这本书也凝结了作者在软件设计中的很多经验，所以很多话都可以仔细思考推敲，对我们平时开发中遇到的问题也做了很多思考或总结，值得经常拿出来翻一翻。
当然本书里面多少思想适用于我们目前的工作，多少适用于互联网微服务架构下的开发，也是值得我们思考的问题。

大二寒假从图书馆借来这本书，只翻了一下，然后打了一个寒假游戏。今天终于把债还清了，说实话，这本书很晦涩。

很多词都不翻译，不能认同。

翻译就不说了，美国回来的大厂同事表达没见过ddd

翻译真的懒，应该去看其他高评价版本的。浪费了时间，少了收获

诘屈聱牙

同重构一类书有点相似，如果不会的人很可能完全看不懂，如果会的人往往会发现之前通过一些观察和实践，已经用到了书里一些方法。单纯对于本书，前面比较好读，后面的模型过于复杂了，看不太下去。

没啥可说的只有学习、体会、实践！！
看完了，虽然后悔这么晚才看完，但其实就是早点看估计也是看不懂的，至少现在看能知道人家讲的什么事情，解决的是什么问题。周末的时候详细写个书评吧。

十年前东西，有些不知不觉在用了；边界上下方对微服务有借鉴作用，其它业务领域了解不深

感觉很抽象很难读懂，决定弃疗，开始读《实现领域驱动设计》

讲的东西比较抽象，应该是经典的东西，但是看上去总感觉隔阂：有的地方熟悉，有的地方感觉价值不那么大，不知道是不是翻译的问题。

完全没办法读下去的一本书，对于DDD来说，或者说对于设计来说，采用一个通俗易懂的例子太重要了，我大部分精力基本都耗费在了理解业务知识以及陌生的"英文转悠名词"上。再者就是对于互联网领域而言，也许真的不太适用。

说不清这东西是否有用，那就算没用吧。分层，模块化不讲我也知道呀，至于业务驱动编码，谁家还能技术驱动业务？

用model理解并固化domain knowledge，这一点有点像「建筑师的图解思考」。layered architecture更像是建筑解构，每一层都有它专注的任务，正如围护层与结构层等等的关系。 layered architecture与smart UI（react/vue）互斥。layerd architeture一般划分成用户界面层、应用层、领域层、基础框架层。domain model里的「构造块」：entity/value object/service/module/aggregate/factory/repository

---
领域驱动设计_下载链接1_

书评

《领域驱动设计》一书是领域模型领域的代表作，被很多牛人推荐，其中的概念还需要在思考和实践中逐步理解。书中描述的一些现象有些与我们类似，比如越来越多的领域规则被嵌入到查询代码中，或者直接就不见了。领域逻辑跑到查询代码和客户代码中去了，而实体和值对象变成了纯粹...

不要过于关注书中描述的具体技术、设计方法
领域模型贯穿概念模型、逻辑和物理设计模型，贯穿需求采集、分析、设计、实现，到测试部署这一整个开发过程，应该注意从整体角度来理解领域驱动的思想
需求采集时与业务专家的沟通已经开始领域模型的建模工作； 对需求深入的分析整...

原版四星，中文版三星，知识有些陈旧、翻译差、阅读耗时、收益不成比率。
此书翻译比较差，一般情况是不符合中文的表达习惯，很多句子要读几遍才能明白，翻译差点的段落连机器翻译的质量都达不到。由于翻译质量差，所以可能需要花双倍以上的时间来阅读这本书，很多时候都会纠结...

首先说一下我是如何接触这本书的吧。我已经记不起是第一次听说领域驱动是在什么时候了，不过我只记得是在看一本别的架构方面的书时提及到这本书，我顺手在amazon上查了一下，有很多人在推荐这本书。出于对技术的追求，我有立刻把这本书买回家细细研读一下的冲动，于是我上网上...

在书中提到 value object 具有不变性和代数的计算封闭的性质。
在最近的几个开发项目中，大量地使用了 value object 。当使用 c/c++ 来开发的时候，使用 value object
可以减轻内存管理的负担。能带来这种便利的正是因为 value object 具有不变性。value object 一经构造就不...

从当今角度看，很多概念都有了大发展，日常工作中接触到的思想都不谋而合，甚至已经远远超越了作者当年的思想。但是作为领域设计的开篇著作，仍然有很好的阅读价值。
全篇最核心的概念是，人类的记忆力思考力限制，会将一个大型系统耦合复杂化。为了更好的理解及团队成员的合作...

刚开始是冲着这个书的副标题来的，软件核心复杂性应对之道，主标题并没有太在意。最后看了不到一半吧，零散着跳读的。翻译问题很大！！！
进书便开始和我说模型的事，又是分层又是画图，看了几章发现，弄了半天不就是一个UML图吗。 这书给我感觉是一本教 不懂业务只懂编程的程...

看了对于此书的短评，把这本书看成是一本“正确的废话”的人我想不在少数，10年前我看此书也是一样的感觉，10年后微服务大火，很多人又把“领域驱动设计”挂在嘴边，此时我再看此书确实感觉自己看懂了，我想这其中的奥秘其实就在“领域驱动设计”这六个字里。让我给大家仔细分...

我是一个所谓前端er，但我觉得对领域的概念对所谓的前端er们而言也非常重要。特别是中后台的业务前端在不需要实现界面操作的前提下，了解业务的实现非常重要。
这本书里讲了很多的"道"，例如团队协作，开发人员对待需求的态度。
我觉得这本书适合想要了解业务实现的开发人员，...

我是一个所谓前端er，但我觉得对领域的概念对所谓的前端er们而言也非常重要。特别是中后台的业务前端在不需要实现界面操作的前提下，了解业务的实现非常重要。
这本书里讲了很多的"道"，例如团队协作，开发人员对待需求的态度。
我觉得这本书适合想要了解业务实现的开发人员，...

我是一个所谓前端er，但我觉得对领域的概念对所谓的前端er们而言也非常重要。特别是中后台的业务前端在不需要实现界面操作的前提下，了解业务的实现非常重要。
这本书里讲了很多的"道"，例如团队协作，开发人员对待需求的态度。
我觉得这本书适合想要了解业务实现的开发人员，...

该书作者显然拥有大量的设计、编码实践。而且看的出，还是敏捷的拥护者。
难能可贵的是，该书的翻译质量还是很高的。很多地方直接使用英文原文，而不是搞个蹩脚的中文翻译来打乱你的阅读节奏。
只是有部分举例可能因为需要具有业务背景知识才好理解，所以自己感觉没能特别掌握...

原版内容应该不错，但翻译得不好，这可能是国内技术类图书翻译的通病。
以阅读翻译后的吃力劲，去看原版可能效果更好。也许是译者英文看多了，对汉语的语序也变得“英语化”了，有些简单的语言逻辑，被翻译之后，反而变得更生涩难懂。
但愿那些从事翻译的人在精通计算机专业...

Google翻译还是有道翻译的。。
弄明白后想竖个中指，那么简单的概念，翻译的那么复杂。
Google翻译还是有道翻译的。。
弄明白后想竖个中指，那么简单的概念，翻译的那么复杂。

软件最有价值部分是它的领域模型部分。软件开发应该围绕这个核心进行组织，这是领域驱动设计的核心理念。
这本书有价值的地方甚多，值得反复细细揣摩，书中最重要观点，摘录如下：
1.软件开发复杂性的根本原因是问题领域本身错综复杂，控制复杂性的关键是有一个好

的领域模型...

领域驱动设计_下载链接1_