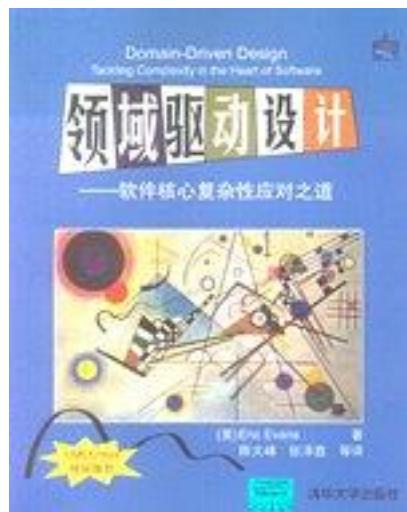


领域驱动设计



[领域驱动设计 下载链接1](#)

著者:Eric Evans

出版者:人民邮电出版社

出版时间:2010-4

装帧:平装

isbn:9787115224071

《领域驱动设计:软件核心复杂性应对之道(英文版)》是领域驱动设计领域的经典之作。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最佳实践综合到一起,融入了作者的见解和经验,展现了一些可扩展的设计最佳实践、经验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。

《领域驱动设计:软件核心复杂性应对之道(英文版)》适合各层次的面向对象软件开发人员、系统分析员阅读。

作者介绍:

Eric Evans世界著名软件建模专家,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。他在全球各地宣讲领域驱动设计的思想,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:

[领域驱动设计](#) [下载链接1](#)

标签

领域驱动

领域驱动设计

编程

领域模型

软件设计

软件开发

design

需求交互

评论

读到一半多的时候还觉得这本书太浅，废话太多。读完后半部分，才发觉这些理念帮助我对领域模型的认识得以升华。读完此书前，有意无意地做到了其中的70%，期待下一个项目对这30%新的认识进行实践。

经典，不解释。

[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...

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...

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...

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...

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...

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...

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...

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](#)