

工作坊 | 领域驱动设计中的事件建模

张逸 逸言 2014-09-15



这两日，我参加了Implementing Domain Driven Design一书作者Vaughn Vernon组织的IDDD Workshop。培训中，Vernon带领我们针对Domain Event进行了一次建模工作坊。



在领域驱动设计中，Domain Event变得越来越重要。在Implementing Domain Driven Design这本书中，Vaughn Vernon甚至将Domain Event上升到了一等公民的地位。

那么，Domain Event到底是什么？Domain Event即领域事件，是指领域中发生的事实（facts）。当满足某个条件时，某个发起者就会触发事件产生。因而在对事件建模时，我们可以关注统一语言中如下的关键词汇：

- “当...”
- “如果发生...”
- “当...的时候，请通知我”
- “发生...时”

“事件为事实（fact）”这一描述让我对“事件”本身有了更准确的认识。它让我想起两篇发表在InfoQ上的文章。一篇文章为《Datomic的架构》。文中提到：“信息是一组事实（facts），事实是指一些已经发生的事情。鉴于任何人都无法改变过去，这也意味着数据库将累积这些事实，而非原地进行更新。虽然过去可以遗忘，但却是不能改变的。”同

理推之，若事件即事实，那么它也是不可改变的。对于这些历史发生的“事实”，我们需要“立此存照”——于是，这就引出了Event Store或者Event Sourcing。我会在后续的文章深入分析Event Store与Event Sourcing。

另一篇文章是徐昊的《运用四色建模法进行领域分析》。文中表达了类似的思想：“任何的业务事件都会以某种数据的形式留下足迹。我们对于事件的追溯可以通过对数据的追溯来完成。……你无法回到从前去看看到底发生了什么，但是却可以在单据的基础上，一定程度的还原当时事情发生的场景。当我们把这些数据的足迹按照时间顺序排列起来，我们几乎可以清晰的推测出这个在过往的一段时间内到底发生了那些事情。”

在四色建模分析法中，徐昊认为应该将“时标性对象（moment-interval）”作为建模的起点。我在这里并不是要介绍四色建模法，这个话题留待以后再讲。我试图阐释的观点在于，如果事件与时间相关，那么在对事件进行建模时，就可以针对业务场景确定一条时间线，并通过分析业务状态的各种变迁，得到我们希望获得的事件模型。这正是这个工作坊的切入点。

整个事件建模的活动可以分为四个步骤：

1. 选定某个自己熟悉的领域，然后针对时间线去寻找那些用过去时态表现的事件；找到这些事件后，用黄色即时贴写出事件名称，形式如：OrderFilled。
2. 针对每个事件，对触发事件的Command对象进行建模，并用绿色即时贴写出Command的名称。对Command对象进行建模并非单纯地为了寻找Command对象，而是为了更深一步地验证之前建模的事件模型。在思考触发事件的对象时，我们可能会发现一些遗漏又或者多余的事件。
3. 判断这些事件应该属于哪个聚合对象，找到它们，然后写在紫色即时贴上。
4. 对这些识别出来的Event、Command、Aggregate进行分类，判断它们到底属于哪个Bounded Context，并用红色即时贴标出。



这种Workshop不仅只针对培训，它更应该运用到团队进行领域驱动设计的过程中。这也正是我一直在提倡的所谓“可视化设计”。可视化设计并非一个噱头，更不是为了美观好看，而是希望以直观简单的形式展现设计思路，尤其需要让整个团队成员都能以协作互动的形式参与到这个设计过程中。群策群力，头脑风暴，如此方能获得更好的设计方案，并以这种团队行为的方式完成知识的共享与传递。其实，“架构”究竟是什么，不就是一种软件设计的知识吗？设计架构，重要在于交流，在于知识传递，而不仅仅是严谨完整的解决方案文档。

逸言

「逸言」是张逸的个人空间，主要关注软件开发领域。空间既有业界时评，亦有技术感悟，偶而抒发人生感怀、社会反思又或文学雅趣。