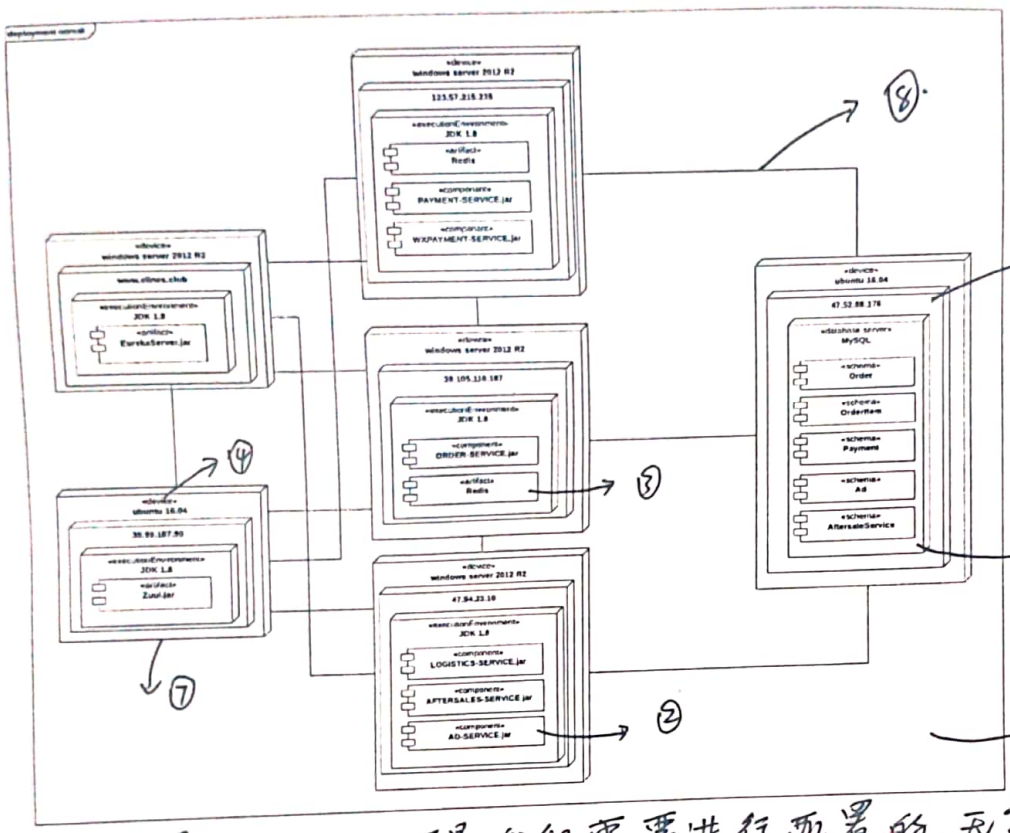


① 服务器的IP地址不是我们需要进行配置的，无需等用环境结点来表示。

② jar包应该是一个人工完成的 artifact，而不是一个组件。

③ redis 是我们配置的一个环境，应该用环境节点来表示。

④ device 设备所标识的应该是一个服务器名称，而不是一个操作系统名称。

⑤ MySQL 环境下应该是运行一个测试需用的 sql 文件，而不是是写为所有 schema.

⑥ 该图缺少客户机，无法体现测试用例对系统的~~的~~测试。

⑦ 网关所在的服务器也要与数据库服务器相连来获取权限信息。

⑧ 服务器与数据库之间是 TCP/IP 协议，其~~它~~的是网络通信 http/https 协议，图中没有体现。

③ 图一的状态和动作，这一点也

④ 在状态机图中

~~三，该图的错误~~

① Controller 层

该是 Comment

这个时候不应

Commen Service

二，我认为第一张图比较合理。

① 首先根据对象标准组定义的订单状态，图二明显是少了好多个状态的。在未发货前都可以取消订单这一点图一将取消的三个状态~~用大括号~~包起来，这一设计是比图二要~~好一些~~清晰明了的。

② 由于在数据库设计的时候已经有了 be deleted 字段，所以删除在这订单状态机图中不应成为一个状态，图二这点较为不合理。

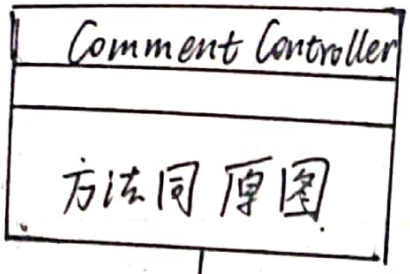
③ 图一的状态转变既有动作又有监护条件，而图二都是只有动作，这一点也是图一较为合理的。

④ 在状态机图中图二有~~好~~个结束点，这也是一个不合理的地方。

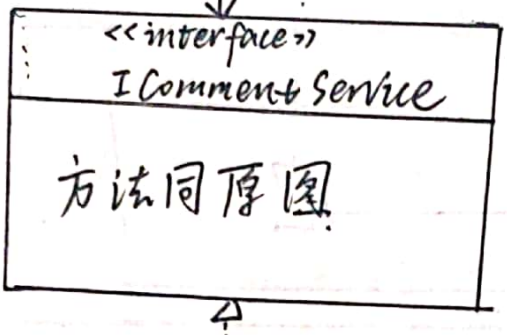
~~三，该图的错误~~

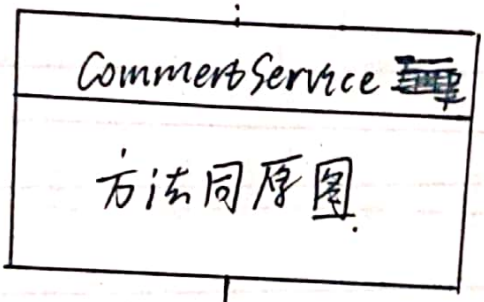
三、

Controller.

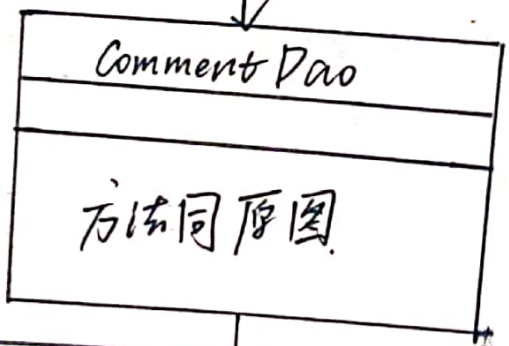


Service.

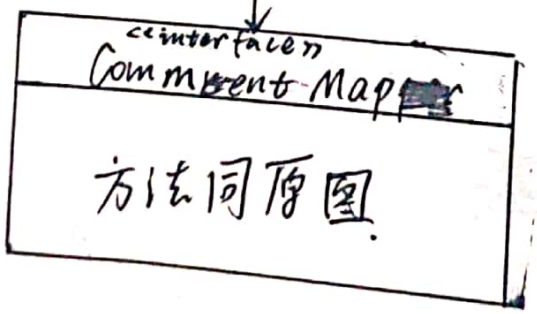




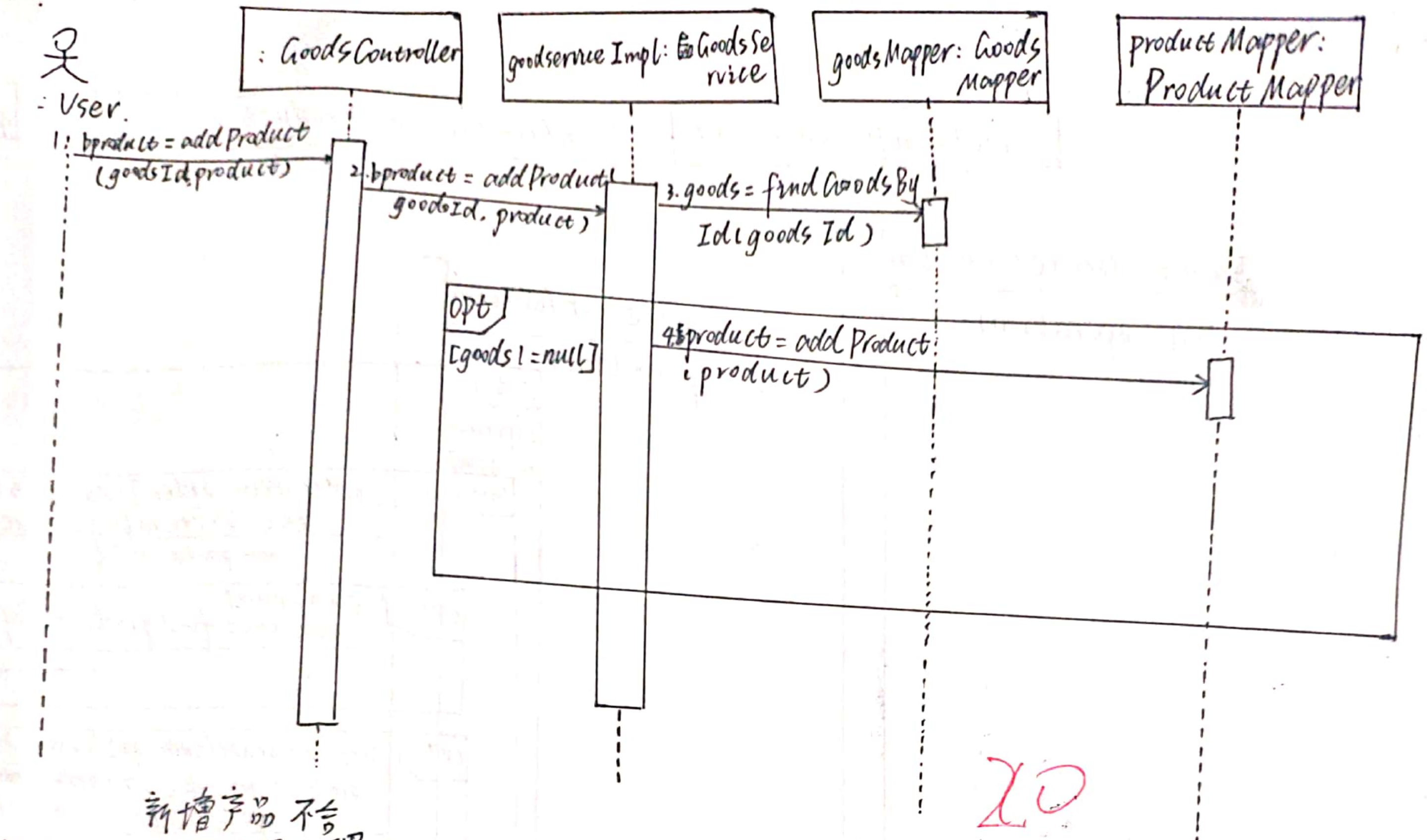
DAO



MAPPER.



四.



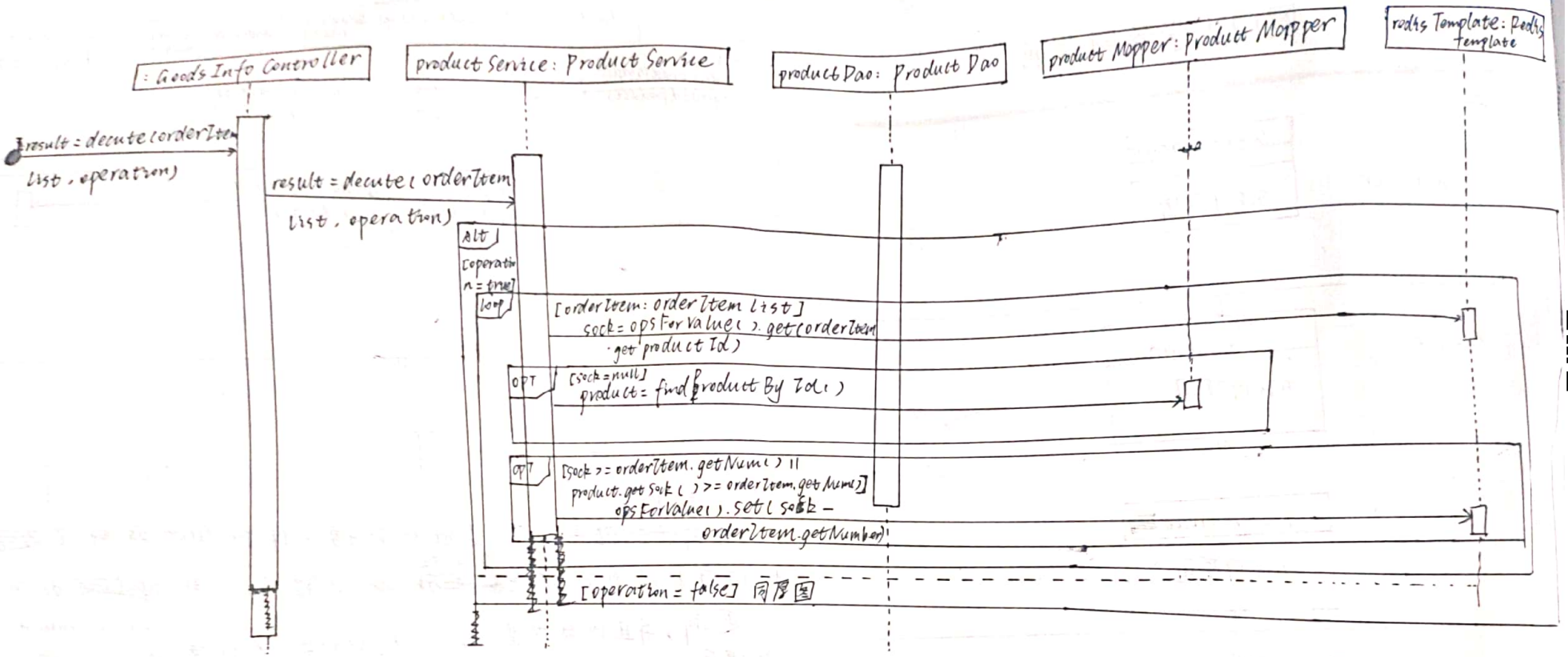
20

① 该设计新增产品不合理，5、13是因为新增一件product的时候在数据库中没有新增库存字段的功能，所以用updateProduct的函数去重新更新，并且用的也是product.stock，为何要多此一举，多次访问数据库。

② 3.goods是在service层中获得，所以opt框不应把controller包含进去。

③ Dao层和Mapper冗余了，没有必要存在Dao层。

五、



~~关于operation的判断属于逻辑判断，应该放在~~

① Dao层主要是做数据的持久化，数据封装等，将太多逻辑判断放DAO不合理，我认为应放在service层。 20

② 若是redis中找不到该product，去内存中找，找到之后需要用product来判断是否~~需要~~可以扣库存

③ 若是增加库存的操作，没有无法增加库存的情况，都能增加，无判断框。

六.

合理之处：①该图的 Controller 层是做了~~[illegible]~~ 与外部信息的处理转发，符合控制器原则。并且 order 是由 controller 进来的，最后处理完由 controller 传出。

② 符合信息专家原则：职责都分配到了拥有该信息的类上。

③ 利用 strategy 来保护变化，若以后有新的计算折扣方法易于扩展改变，也是策略设计模式。

不合理之处：

① ~~是~~ order 是在 service 层有了的信息，独立为一个对象而且没有创建的前提是不合理的，应该是 Service 中的自我调用。

② 策略 strategy 同样应该是创建之后才会有的，不符合创建者原则。