第 8 章 下沉复杂性

本章介绍了有关如何创建更深的类的另一种思考方式。假设您正在开发一个新模块，并且发现了一个不可避免的复杂性。那么，是应该让模块的使用者处理复杂性，还是应该在模块内部处理复杂性？如果复杂性与模块提供的功能有关，则第二个答案通常是正确的答案。大多数模块的使用者会多于其开发人员，因此麻烦开发人员比麻烦使用者更好。作为模块开发人员，您应该努力使模块使用者的生活尽可能轻松，即使这对您来说意味着额外的工作。表达此想法的另一种方式是，**让模块的接口简单比让其实现简单更为重要。**

作为开发人员，很容易以相反的方式行事：解决简单的问题，然后将困难的问题推给其他人。如果出现不确定如何处理的情况，最简单的方法是抛出异常并让调用者处理它。如果不确定要实施什么策略，则可以定义一些配置参数来控制该策略，然后由系统管理员自行确定最佳策略。

这样的方法短期内会使您的生活更轻松，但它们会加剧复杂性，因为许多人都必须处理一个问题，而不仅仅是一个人。例如，如果一个类抛出异常，则该类的每个调用者都必须处理该异常。如果一个类暴露配置参数，则每个系统管理员在每次安装中都必须学习如何设置它们。

8.1 示例：编辑器文本类

考虑为图像界面文本编辑器管理文件文本的类，这在 第 6 章 和 第 7 章 中讨论过。该类提供了将文件从磁盘读入内存、查询和修改文件在内存中的副本以及将修改后的版本写回磁盘的方法。当学生们要实现这个类时，许多人选择了面向行的接口，该接口具有读取、插入和删除整行文本的方法。这导致了类实现起来很简单，但也为更高层级的软件带来了复杂性。在用户界面的层级，很少涉及整行操作。例如，击键会导致在现有行中插入单个字符；复制或删除选择的区域可能同时修改几个不同的行。使用面向行的文本接口，更高层级的用户界面在实现时必须自行拆分和连接行。

面向字符的界面（如 第 6.3 节 中所述）下沉了复杂性。用户界面软件现在可以插入和删除任意范围的文本，而无需拆分和连接行，因此变得更加简单。但是文本类的实现可能会变得更加复杂：如果内部将文本表示为行的集合，则必须拆分和连接行以实现面向字符的操作。但这种方法更好，因为它在文本类中封装了拆分和连接的复杂性，从而降低了系统的整体复杂性。

8.2 示例：配置参数

配置参数是上升复杂性而不是下沉复杂性的一个示例。类可以暴露一些控制其行为的参数，而不是在内部确定特定的行为，例如高速缓存的大小或在放弃之前重试请求的次数。该类的使用者必须为参数指定适当的值。在当今的系统中，配置参数已变得非常流行，有些系统有数百个配置参数。

配置参数的拥护者认为配置参数不错，因为它们允许用户根据他们的特定要求和工作负载来调整系统。在某些情况下，低层级的基础结构代码很难知道要应用的最佳策略，而用户则对其领域更加熟悉。例如，用户可能知道某些请求比其他请求更紧迫，因此用户为这些请求指定更高的优先级是有意义的。在这种情况下，配置参数可以在更广泛的领域中带来更好的性能。

但是，暴露配置参数还提供了“偷懒的机会”：将参数该如何配置的重要问题传递给它的使用者。在多数情况下，用户或管理员很难或无法确定正确的参数值。在其他情况下，可以通过在系统实现中进行一些额外的工作来自动确定正确的值。设想一个必须处理丢失数据包的网络协议。如果它发送了请求但在一定时间内未收到响应，则重新发送该请求。确定重试间隔的一种方法是引入配置参数。但是，传输协议可以通过测量成功请求的响应时间，并将该响应时间的倍数用于重试间隔，自己计算出一个合理的值。这种方法下沉了复杂性，使其用户不必自行找出合适的重试间隔。它还具有动态计算重试间隔的优点，那么，当操作条件发生变化时，它将自动调整参数值。相反，配置参数很容易就过时了。

因此，您应尽可能避免使用配置参数。在暴露配置参数之前，请问自己：“用户（或更高层级的模块）能比我们确定一个更好的参数值吗？”当您创建配置参数时，请确认是否可以提供合理的默认值，以便用户仅需在特殊情况下提供这个值。理想情况下，每个模块都应当彻底解决问题，而配置参数使得解决方案不完整，从而增加了系统复杂性。

8.3 做过头了

下沉复杂性时要谨慎处理；这个想法很容易做过头。一种极端的方法是将整个应用程序的所有功能归为一个类，这显然没有意义。如果（a）被下沉的复杂性与该类的现有功能密切相关，（b）下沉复杂性将导致应用程序中其他地方的简化，(c) 下沉复杂性将简化类的接口，则下沉复杂性最有意义。请记住，目标是最大程度地降低整体的系统复杂性。

第 6 章介绍了学生们如何在文本类中定义一些反映用户界面的方法，例如实现退格键功能的方法。这似乎很好，因为它可以下沉复杂性。但是，将用户界面的知识添加到文本类中并不会大大简化高层级的代码，并且用户界面的知识与文本类的核心功能无关。在这种情况下，下沉复杂性只会导致信息泄露。

8.4 结论

在开发模块时，为了减少用户的痛苦，要找机会给自己多吃一点苦。