**Observer（观察者）**

意图

定义对象间的一种一对多的依赖关系，当一个对象的状态发生改变时，所有依赖于它的对象都得到通知并被自动更新。

动机

将一个系统分割成一系列相互协作的类有一个常见的副作用：需要维护相关对象间的一致性。我们不希望为了维持一致性而使各类紧密耦合，因为这样降低了它们的可重用性。

例如，许多图形用户界面工具箱将用户应用的界面表示与底下的应用数据分离。定义应用数据的类和负责界面表示的类可以各自独立地复用。当然它们也可以一起工作。一个表格对象和一个柱状图对象可使用不同的表示形式描述同一个应用数据对象的信息。表格对象和柱状图对象互相并不知道对方的存在，这样使你可以根据需要单独复用表格或柱状图。但在这里它们表现的似乎互相知道。当用户改变表格中的信息时，柱状图能立即反应这一变化，反过来也是如此。

这一行为意味着表格对象和棒状图对象都依赖于数据对象，因此数据对象的任何状态改变都应立即通知它们。同样也没有理由将依赖于该数据对象的对象的数目限定为两个，对相同的数据可以有任意数目的不同用户界面。

Observer模式描述了如何建立这种关系。这一模式中的关键对象是目标和观察者。一个目标可以有任意数目的依赖他的观察者。一旦目标的状态发生改变，所有的观察者得到通知。作为对这个通知的响应，每个观察者都将查询目标以使其状态与目标的状态同步。

这种交互也称为发布—订阅。目标是通知的发布者。他发出通知时并不需要知道谁是它的观察者。可以有任意数目的观察者订阅并接收通知。

结构图

