

## Problem 7

### 1 The basic problem

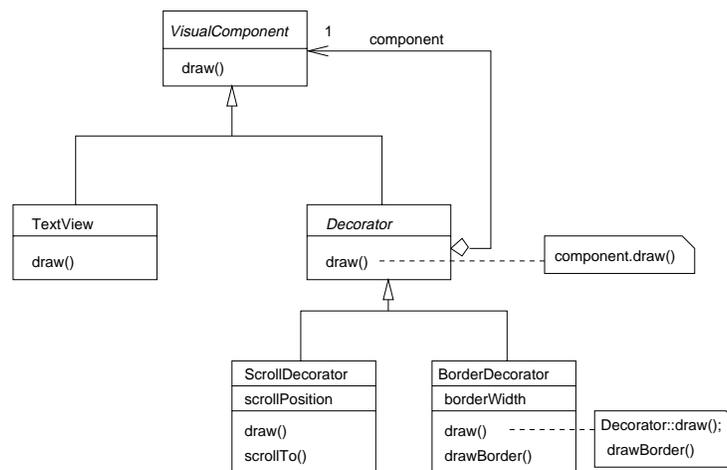
A designer needs to be able to add several different responsibilities to an existing object (which we'll call the component).

Consider, for example, a GUI toolkit in which there is already a plain `TextView` class with the responsibility to display a page of text. We may need to have some text windows which are scrollable (can respond to a `scroll()` message), some which have a visible border, and some which have both.

### 2 An initial solution...

### 3 A standard solution: DECORATOR

We enclose the component in a wrapper or *decorator* which implements (at least) the same interface as the original. The decorator maintains a reference to the component, and forwards messages to it. It may do extra work before or after forwarding the message, however; this is where the decorating behaviour is. A typical class diagram for the situation is:



### 4 When and why is this solution good?