Problem 8

1 The basic problem

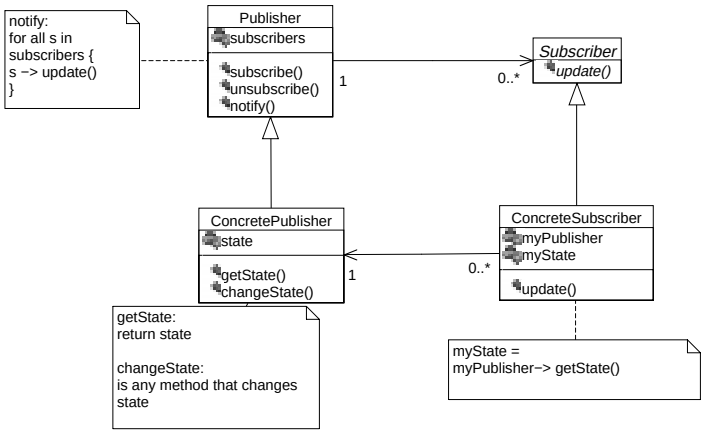
When one object (which we'll call the subject) changes state, one or more other objects (which we'll call the dependents) need to know about it so that their states remain consistent.

Consider, for example, a system with a (decoupled) user interface. Elements of the user interface may need to display an up to date version of the state of an object inside the system (a customer's balance, or whatever).

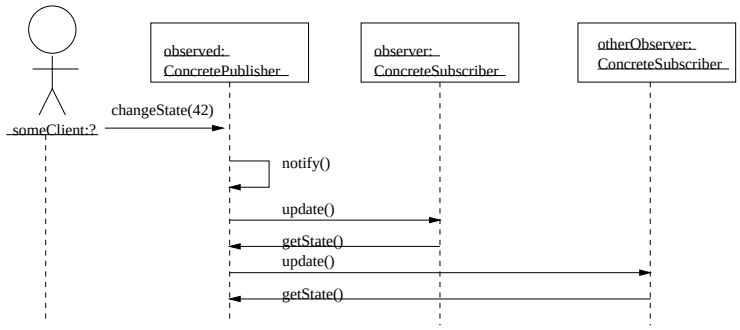
2 An initial solution...

3 A standard solution: OBSERVER

The subject (shown here as ConcretePublisher) inherits from an abstract class Publisher which maintains a collection of subscribers to this subject; that is, objects which should be notified when the state changes.



In the code of the subject (the ConcretePublisher), when the state is changed the subject sends itself the message notify. The code for acting on this message is in the Publisher class; it notifies all the current subscribers.



4 When and why is this solution good?