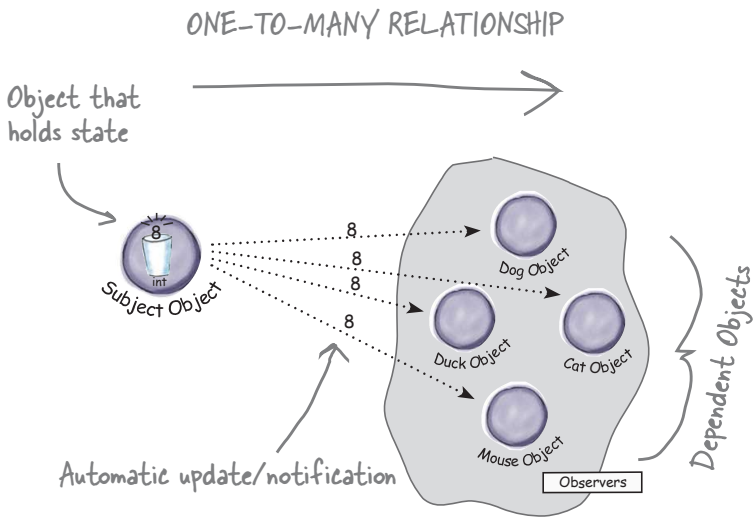
The Observer Pattern defined

A newspaper subscription, with its publisher and subscribers, is a good way to visualize the pattern.

In the real world, however, you'll typically see the Observer Pattern defined like this:

> **The Observer Pattern** defines a one-to-many dependency between objects so that when one object changes state, all of its dependents are notified and updated automatically.

Let's relate this definition to how we've been thinking about the pattern:



The subject and observers define the one-to-many relationship. We have *one subject*, who notifies *many observers* when something in the subject changes. The observers *are dependent* on the subject—when the subject's state changes, the observers are notified.

As you'll discover, there are a few different ways to implement the Observer Pattern, but most revolve around a class design that includes Subject and Observer interfaces.

The Observer Pattern defines a one-to-many relationship between a set of objects.

When the state of one object changes, all of its dependents are notified.