inherits from `Posts` directly. `MessagePost` objects inherit everything from both superclasses and have the same fields and methods as before. Objects of class `EventPost` will inherit the `username` and `timestamp`, but not the comments.

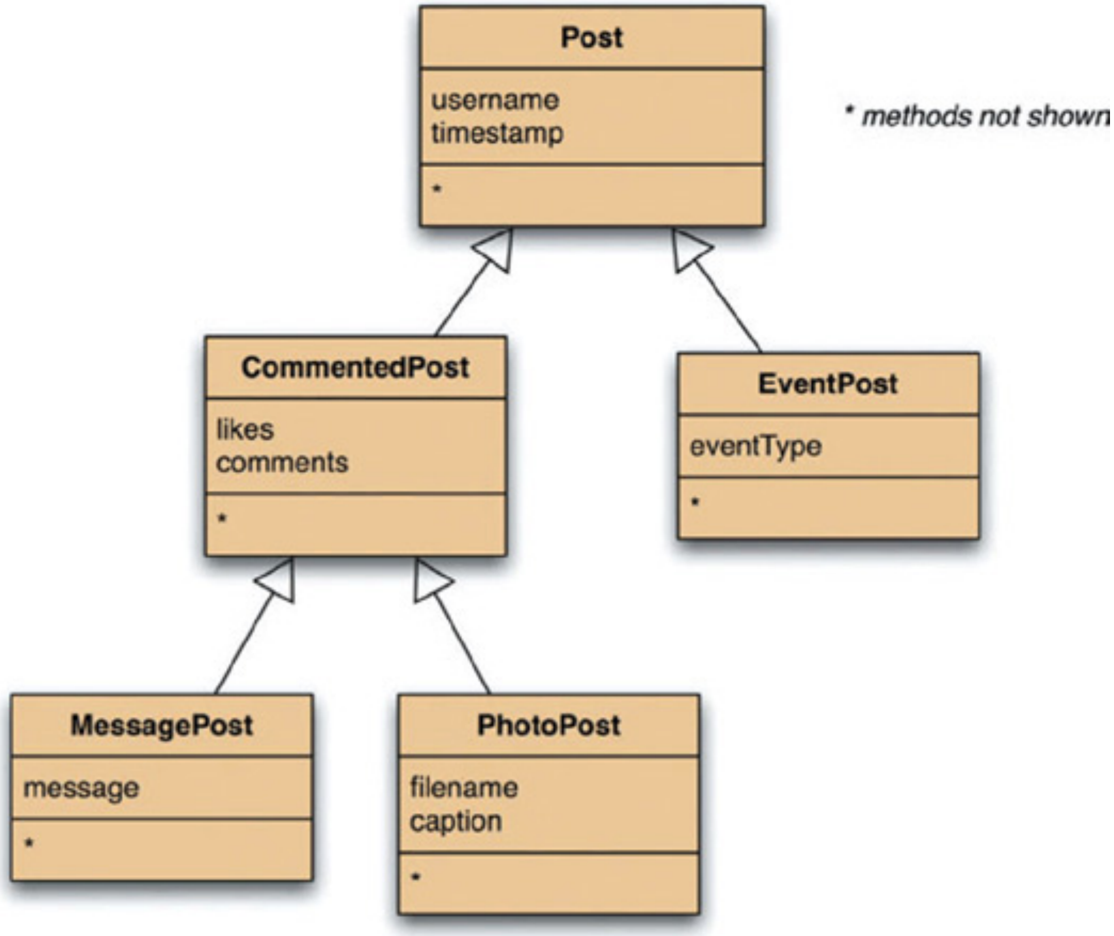

Figure 8.8 Adding more post types to *network*

This is a very common situation in designing class hierarchies. When the hierarchy does not seem to fit properly, we have to refactor the hierarchy.

Classes that are not intended to be used to create instances, but whose purpose is exclusively to serve as superclasses for other classes (such as `Post` and `CommentedPost`), are called *abstract classes*. We shall investigate this in more detail in **Chapter 10**.

Exercise 8.8

Open the *network-v2* project. Add a class for event posts to the project. Create some event-post objects and test that all methods work as expected.