CSC301

Observers, Adapters & Test Doubles

Observer/Observable Pattern

- Common design pattern
- Appeared in the Gang of Four book
  - Gamma, Helm, Johnson & Vlissides. **"Design Patterns: Elements of Reusable Object-Oriented Software"**, Addison-Wesley, 1994
  - Essentially, a collection of common problems and "recipes" for generic solutions

OBSERVER   Object Behavioral

Intent
Define a one-to-many dependency between objects so that when one object changes state, all its dependents are notified and updated automatically.

Also Known As
Dependents, Publish-Subscribe

Motivation
A common side-effect of partitioning a system into a collection of cooperating classes is the need to maintain consistency between related objects. You don't want to achieve consistency by making the classes tightly coupled, because that reduces their reusability.

For example, many graphical user interface toolkits separate the presentational aspects of the user interface from the underlying application data [KP88, LVC89, P*88, WGM88]. Classes defining application data and presentations can be reused independently. They can work together, too. Both a spreadsheet object and bar chart object can depict information in the same application data object using different presentations. The spreadsheet and the bar chart don't know about each other, thereby letting you reuse only the one you need. But they *behave* as though they do. When the user changes the information in the spreadsheet, the bar chart reflects the changes immediately, and vice versa.

Chat:

Anonymous143: What did he just say about xxx?

Anonymous232: xxx implies aaa so yyy

Anonymous32: wasnt yyy yyz?

Anonymous233: no yyy by the pythagorean theorem is bdsad

JohnSmith: How does A relate to B again?

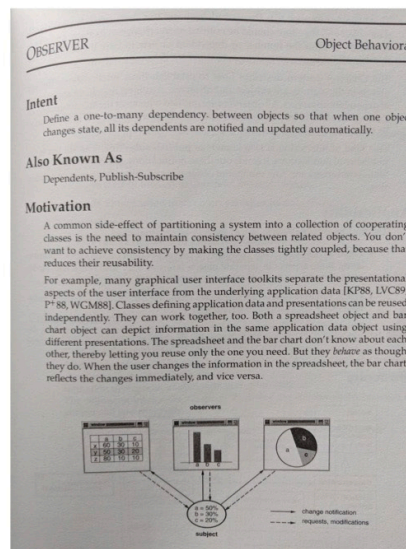Anonymous34: a is the runtiem of b

Jack: @Slide2 Do we need the gang of four book?

JohnSmith: No

Anonymous2: When did he say the test was gonna be?

☑ As Anonymous ☐ As Bobby

Next week!