

Week 2: Observer pattern

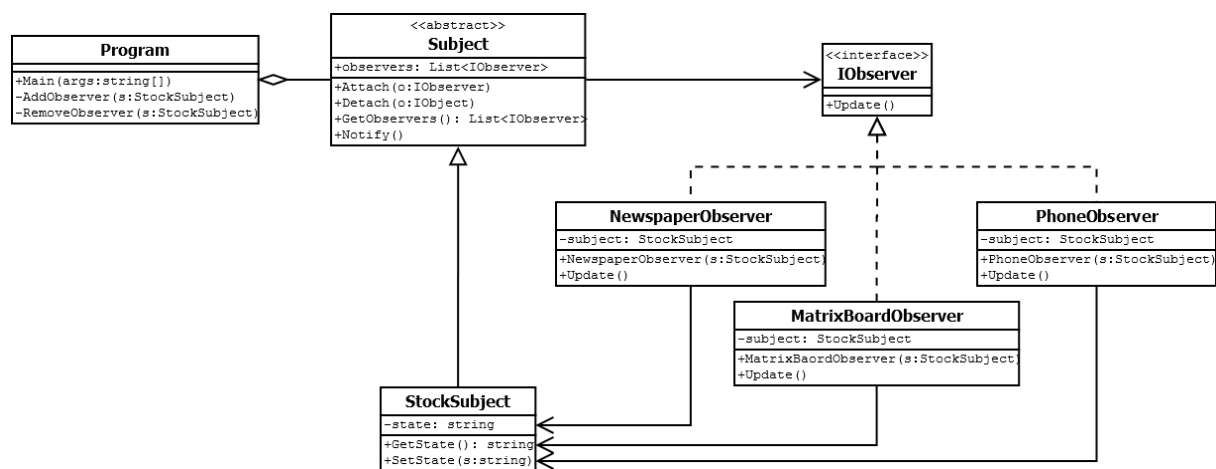
Minh-Triet Diep - Lars Jaeqx - T61

Het patroon

We hebben een abstracte klasse Subject, die de StockSubject klasse gebruikt. Hierin wordt een lijst bijgehouden van alle observers. Deze klasse heeft een Notify methode die wordt aangeroepen als de data door de server is veranderd. Deze methode laat alle observers een Update methode aanroepen die dan de nieuwe data van de server ophaalt.

Ook hebben we een IObservable interface, deze wordt geïmplementeerd door alle verschillende observers, dit zijn bijvoorbeeld een krant, een matrixbord en een telefoon. Hierin wordt een Update methode die van het subject de GetState methode aanroept.

UML



Pull versie van Observer.

Test run results

Standaard heeft de applicatie 3 observers, een telefoon, een krant en een matrixboard.
Wanneer de toets "1" wordt ingedrukt, wordt een observer van willekeurig type toegevoegd.
Wanneer de toets "2" wordt ingedrukt, wordt een willekeurige observer verwijderd.

```
Telefoon zegt: Tick: 0
Matrix zegt:  Tick: 0
Krant zegt:   Tick: 0
Telefoon zegt: Tick: 1
Matrix zegt:  Tick: 1
Krant zegt:   Tick: 1
Telefoon zegt: Tick: 2
Matrix zegt:  Tick: 2
Krant zegt:   Tick: 2
Telefoon zegt: Tick: 3
Matrix zegt:  Tick: 3
Krant zegt:   Tick: 3
Telefoon zegt: Tick: 4
```

1. Default run without addition or removal of observers

```
Matrix zegt:  Tick: 52
Krant zegt:   Tick: 52
Krant zegt:   Tick: 52
Telefoon zegt: Tick: 52
Krant zegt:   Tick: 52
Telefoon zegt: Tick: 52
Telefoon zegt: Tick: 52
Krant zegt:   Tick: 52
Krant zegt:   Tick: 52
Matrix zegt:  Tick: 52
Krant zegt:   Tick: 52
Telefoon zegt: Tick: 52
Krant zegt:   Tick: 52
```

2. Run met het toevoegen van veel willekeurige observers

```
Press ESC to exit, 1 to add an observer, 2 to remove an observer
Telefoon zegt: Tick: 0
Matrix zegt:  Tick: 0
Krant zegt:   Tick: 0
Telefoon zegt: Tick: 1
Matrix zegt:  Tick: 1
Krant zegt:   Tick: 1
Telefoon zegt: Tick: 2
Krant zegt:   Tick: 2
Krant zegt:   Tick: 3
doe ff nie niks verwijderen
```

3. Verwijderen van observers.

Voor en nadelen

Reusability

- + Elke nieuwe observer kan aan een andere subject worden gekoppeld, op deze manier kunnen er verschillende subjects worden gemaakt en kan de observer kiezen op welke hij subscribed.

Maintainability

- + Het subject hoeft niet aangepast te worden als er nieuwe observers worden toegevoegd.
- + Minimale afhankelijkheid tussen subject en observer.

Extensibility

- + Je kunt observers op elk moment toevoegen of verwijderen.
- Omdat observers niet van elkaars bestaan afweten kan er niet onderling gecommuniceerd worden.

Other

- Alle data moet altijd opnieuw worden opgehaald, bij veel data is dit niet efficiënt. (pull)

Reflectie

Door de hogere mate van abstractie was het dit keer wat meer moeite om het patroon geïmplementeerd te krijgen. De ISubject interface is hier voornamelijk schuldig voor, maar het heeft wel gezorgd dat het toevoegen en verwijderen van observers makkelijk te doen was in het hoofdprogramma.