

Object Oriented Modeling and Design

Design Example: Connecting the UI Layer to the Domain Layer

Remember, we put a controller object between the UI and domain layers to ensure low coupling.

However, in some cases, UI objects may send messages to domain objects directly.

For example, in the case of the enterItem message, we want the window to show the running total after each entry.

Solution 1: Add a getTotal method to the Register.

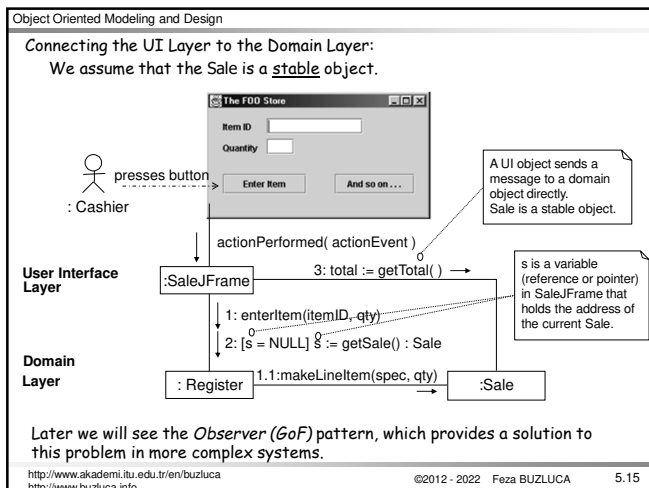
The UI sends the getTotal message to the Register, which delegates to the Sale. Then the Register gets the result from Sale and passes it to the UI layer. This provides low coupling but may overload the Register, making it less cohesive.

Solution 2: An object in the UI gets the reference of the current Sale object from the Register.

When the UI requires the total, it directly sends messages to the Sale. This design increases the coupling from the UI to the domain layer. However, coupling to the Sale is not a major problem if the Sale is a stable object. This makes the Register more cohesive.

http://www.akademi.itu.edu.tr/en/buzluca
http://www.buzluca.info

©2012 - 2022 Feza BUZLUCA 5.14



Object Oriented Modeling and Design

Design Example: Initializing the System

For most of the systems, it is necessary to write a "Start-up" use case that includes system operations related to the starting up of the application.

What should happen when we start the program?

Although the "start-up" use case is the earliest one to execute, we delay its design until after all other system operations have been considered.

Do the initialization design last.

In a start-up, we create an **initial domain object** (or a set of initial domain objects).

The initial domain object is responsible for

- the creation of its direct child domain objects (which must be created at the start up);
- ensuring the necessary visibility (connection) between related objects.

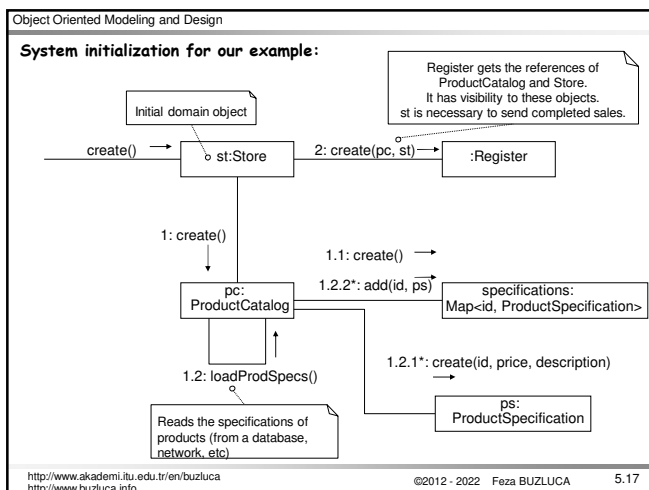
For example,

- Creating the Register and the ProductCatalog objects and establishing a connection between them.
- Establishing the connection between the UI and the controller object.

In our example, we chose the Store as the initial object.

http://www.akademi.itu.edu.tr/en/buzluca
http://www.buzluca.info

©2012 - 2022 Feza BUZLUCA 5.16



Object Oriented Modeling and Design

Initialization in Java:

```

public class Main // Java
{
    public static main( String[] args)
    {
        // Store is the initial domain object
        Store store = new Store();
        Register register = store.getRegister(); // register is created by Store
        ProcessSaleJFrame frame = new ProcessSaleJFrame(register); // Frame is connected // to Register
    }
}

```

Initialization in C++:

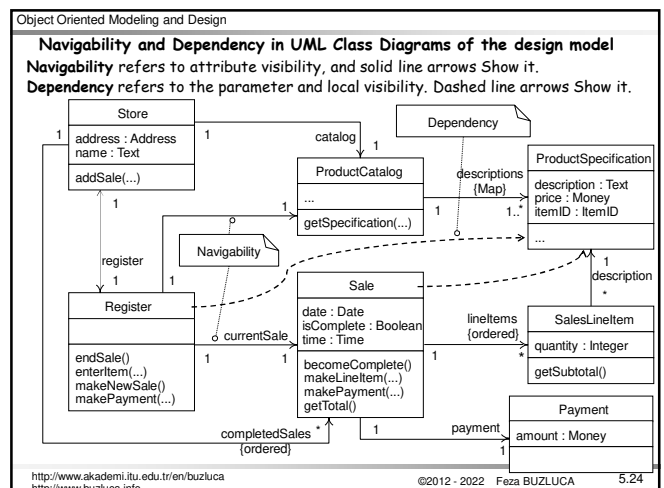
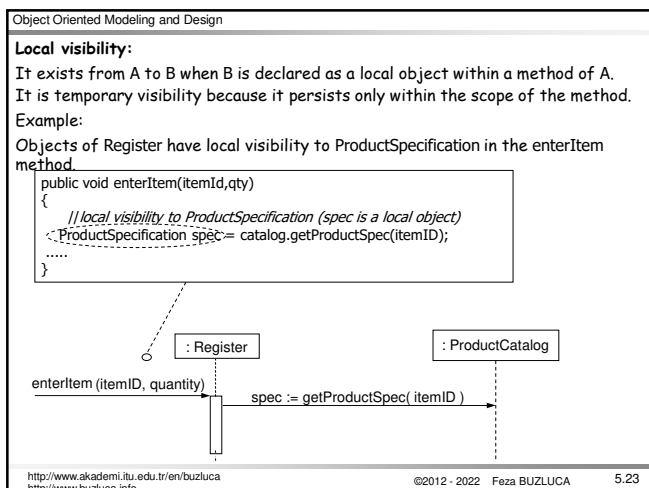
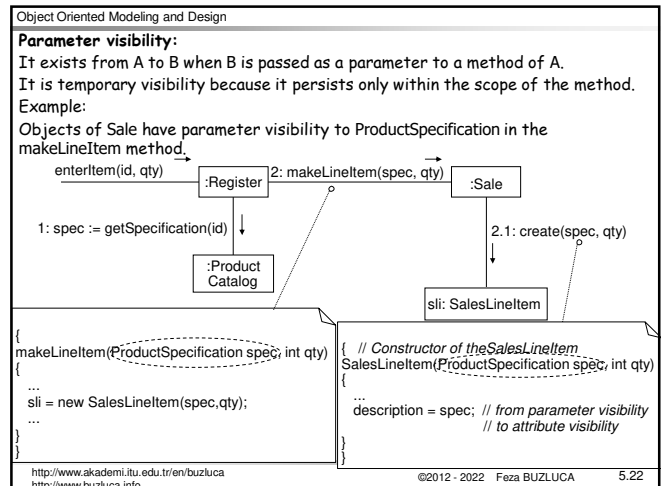
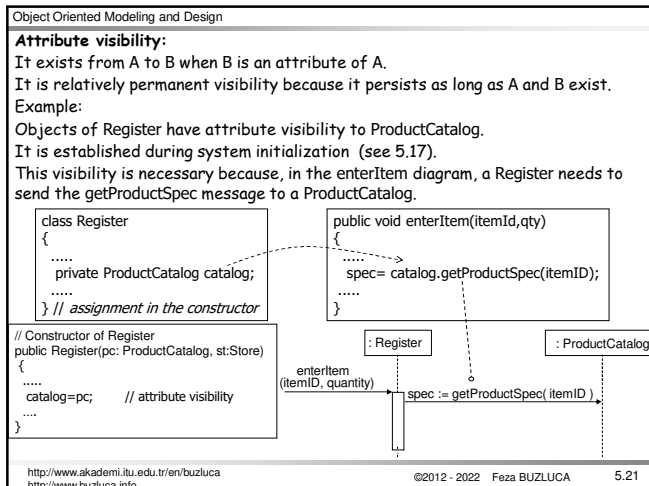
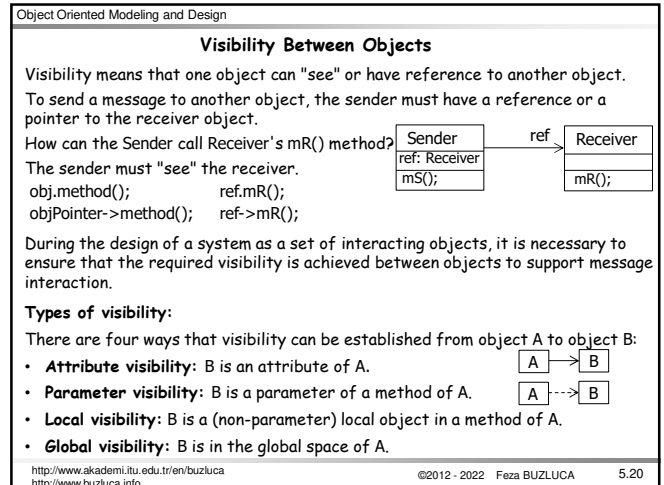
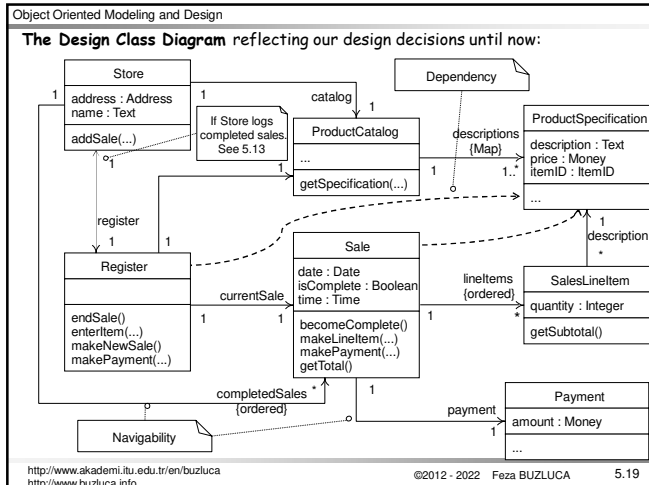
```

int main() // C++
{
    // Store is initial domain object
    Store store;
    Register *register = store.getRegister();
    ProcessSaleJFrame *frame = new ProcessSaleJFrame(register);
}

```

http://www.akademi.itu.edu.tr/en/buzluca
http://www.buzluca.info

©2012 - 2022 Feza BUZLUCA 5.18



Details of UML Class Diagrams

If necessary, access modifiers of the class members and data types may be shown in class diagrams.

In most cases, class diagrams are used to indicate the design decisions. Therefore, programming details are optional.

