PATRÓ FACTORIA

Mario Vila Durán

Enginyeria del software

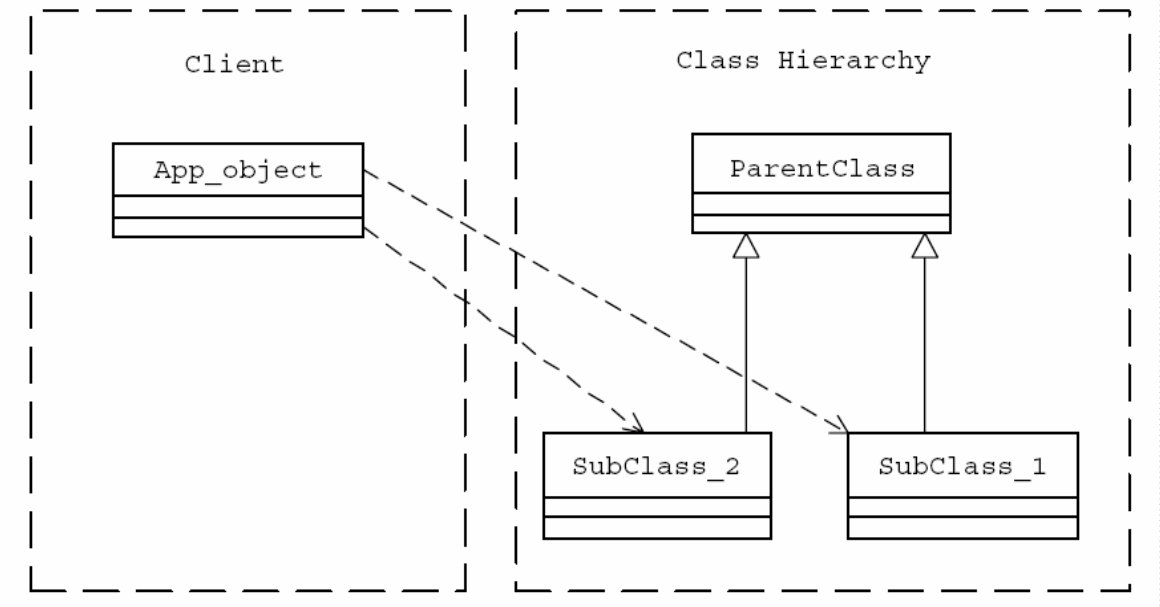
Definició

- S'encarrega de definir una interfície per crear objectes, però permet que la subclasse decideixi quina classe instanciar en temps d'execució.
- Es basa en que en determinats moments una aplicació, tot i conèixer a quin tipus de classe a d'accedir, pot no conèixer a quina subclasses concretament.

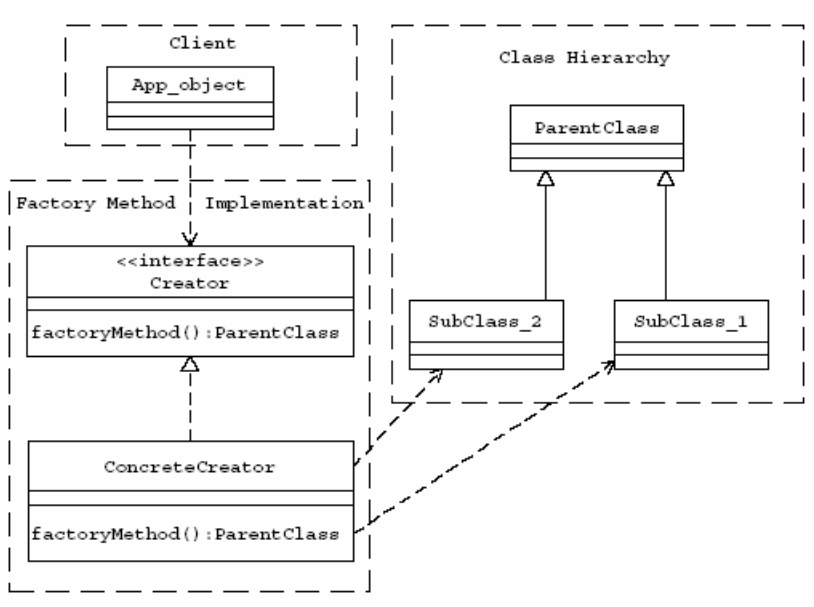
Principis

- Encapsular el que varia
- Complir el principi OCP
- Complir el principi DIP

Sense patró factoria



Amb patró factoria



EXEMPLE I

- Consisteix en una aplicació encarregada de tractar logs de diferents formes.
- Es defineix una interfície Logger.

```
public interface Logger {
  public void log(String msg);
}
```

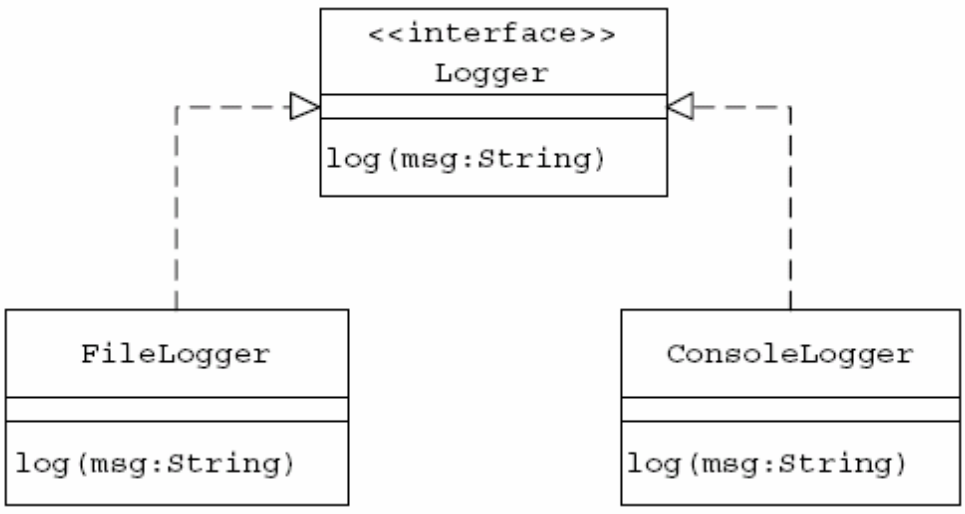
EXEMPLE II

- En general els missatges entrants poden tenir diferents formats d'output. En aquest exemple es tenen dos en compte

Implementer	*Functionality*
`FileLogger`	Stores incoming messages to a log file
`ConsoleLogger`	Displays incoming messages on the screen

EXEMPLE III

- Obtenim la següent estructura d'herència



EXEMPLE IV

- Implementació del mètode log

```
public class FileLogger implements Logger {
  public void log(String msg) {
    FileUtil futil = new FileUtil();
    futil.writeToFile("log.txt", msg, true, true);
  }
}
public class ConsoleLogger implements Logger {
  public void log(String msg) {
    System.out.println(msg);
  }
}
```

EXEMPLE V

- Depenent d'un valor contingut a FileLogging un objecte Logger ha d'utilitzar un mètode log o un altre.

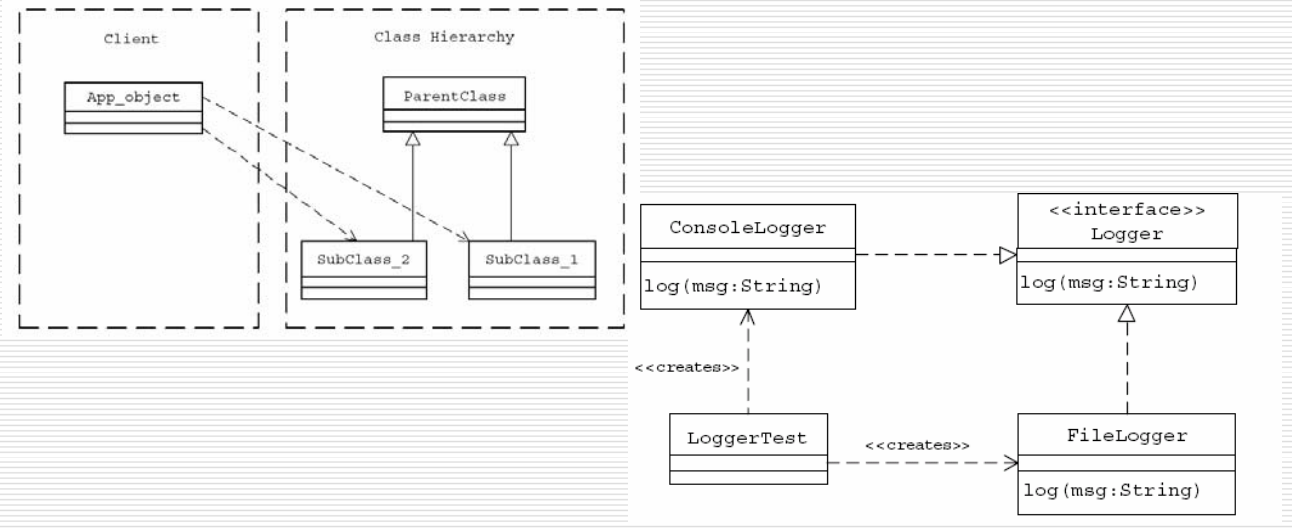
- En aquest exemple FileLogging=OFF significaria que vol mostrar el missatge per pantalla i si es ON el vol guardar en un arxiu.

EXEMPLE VI

- També es disposa d'una aplicació LoggerTest que ha d'utilitzar aquesta estructura.

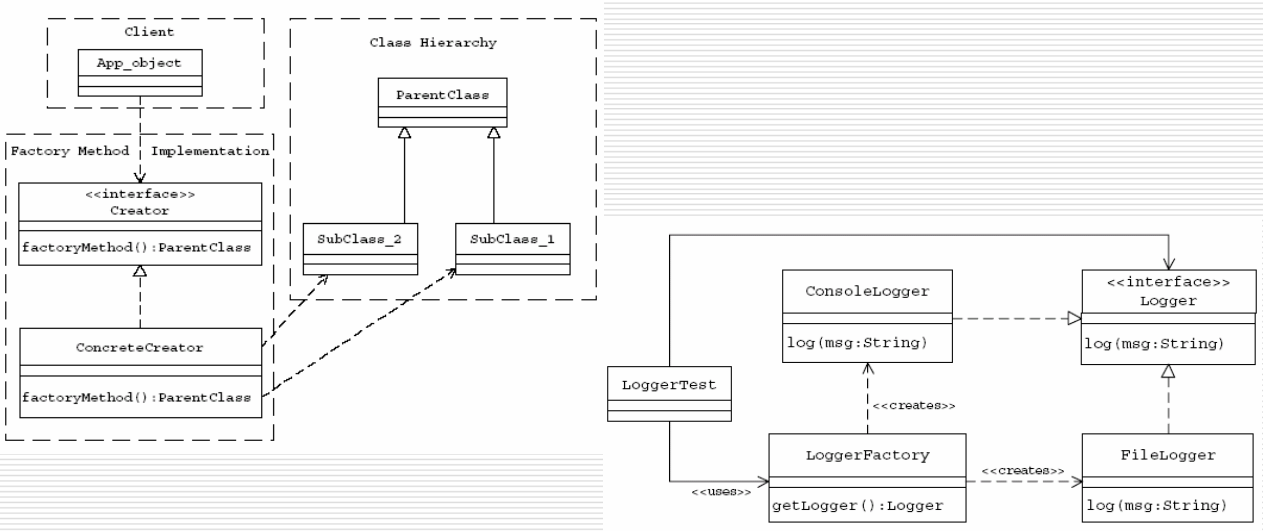
EXEMPLE VII

- Primer disseny sense patró factoria



EXEMPLE IX

- Segon disseny, amb patró factoria



EXEMPLE X

- Codi loggerTest

```
public class LoggerTest {
  public static void main(String[] args) {
    LoggerFactory factory = new LoggerFactory();
    Logger logger = factory.getLogger();
    logger.log("A Message to Log");
  }
}
```

EXEMPLE XI

- Codi LoggerFactory

```
public class LoggerFactory {
  public boolean isFileLoggingEnabled() {
    Properties p = new Properties();
    try {
      p.load(ClassLoader.getSystemResourceAsStream(
        "Logger.properties"));
      String fileLoggingValue =
        p.getProperty("FileLogging");
      if (fileLoggingValue.equalsIgnoreCase("ON") == true)
        return true;
      else
        return false;
    } catch (IOException e) {
      return false;
    }
  }
```

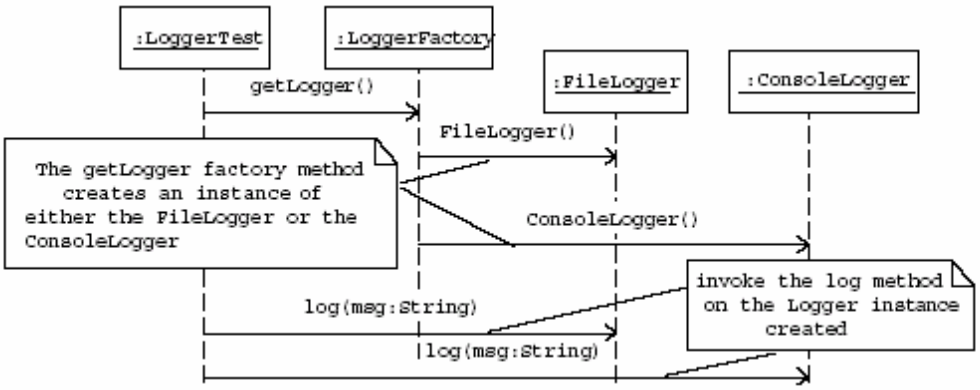
EXEMPLE XII

- Codi LoggerFactory

```
  public Logger getLogger() {
    if (isFileLoggingEnabled()) {
      return new FileLogger();
    } else {
      return new ConsoleLogger();
    }
  }
}
```

EXEMPLE XIII

- Diagrama de seqüència



PATRÓ FACTORIA

Mario Vila Durán

Enginyeria del software