

Sintesi dei requisiti

- Simulare “condizioni di mercato”
 - Alcune persone [*classe*] acquistano un prodotto
 - Il prodotto [*classe*] ha un certo prezzo [*attributo*], che può variare nel tempo
 - Le persone effettuano operazioni tipo prelievo, deposito [*metodi*] con la loro banca [*classe*]
 - Ogni operazione fatta in banca deve essere registrata [*aspetto*] e la banca guadagna su certe operazioni
 - Le persone traggono dei “benefici” [*attributo*] quando effettuano certe operazioni

Ing. E. Tramontana - AO design - 30-Nov-06 1

```
public class Person {
// attributi ...
public void show() {
    System.out.println("Persona: "+name);
    System.out.println("- happiness: "+happiness);
}

public void getMoney() {
    Bank b = new Bank( (int) Math.random()*3+1 );
    pocketMoney += b.withdraw();
}

public void getBeer() {
    happiness++; pocketMoney -= 8;
}

public String getBankName() {
    return BankName;
}

public void decrease(int amount) {
    balance -= amount;
}
}
```

Ing. E. Tramontana - AO design - 30-Nov-06 3

Design + codice

- Classi
 - Persona
 - Banca
 - Test
- Aspetto
 - Logger (riuso dell’aspetto già visto)

Ing. E. Tramontana - AO design - 30-Nov-06 2

```
public class Bank {
    private String name;
    static int totDeposit = 1000;

    public int withdraw(int amount) {
        if (totDeposit > amount) {
            totDeposit -= amount;
            // TODO: bisogna aggiornare il conto del cliente
            return amount;
        }
        return 0;
    }

    public void put(int amount) {
        if (amount > 0)
            totDepositBank += amount;
        // TODO: bisogna aggiornare il conto del cliente
    }

    public int getBalance() {
        return totDeposit;
    }
}
```

Ing. E. Tramontana - AO design - 30-Nov-06 4

```
// Aggiorniamo il conto cliente tramite aspetto
```

```
public aspect UpdatePerson {  
  
    // cattura l'operazione ed il contesto  
    pointcut trapWithdraw(Person pers, Bank b, int am) :  
        call(public int Bank.withdraw(int am)) &&  
        this(pers) &&  
        target(b) &&  
        args(am);  
  
    // usa i dati del contesto per aggiornare il conto cliente  
    after(Person pers, Bank b, int am) : trapWithdraw(pers, b, am) {  
        if (b.getBalance() > am)  
            pers.decrease(am);  
    }  
}
```

Ing. E. Tramontana - AO design - 30-Nov-06 5

```
public aspect UpdatePerson {  
  
    pointcut trapWithdraw(Person pers, Bank b, int am) :  
        call(public int Bank.withdraw(int am)) &&  
        this(pers) &&  
        target(b) &&  
        args(am);  
  
    after(Person pers, Bank b, int am) : trapWithdraw(pers, b, am) {  
        if (b.getBalance() > am)  
            pers.decrease(am);  
  
        // sottraggo un certo ammontare dal conto cliente  
        if (pers.myBank.compareTo(b.name) != 0)  
            pers.decrease(2);  
    }  
}
```

Ing. E. Tramontana - AO design - 30-Nov-06 7

Ulteriori requisiti

- Sviluppatore
 - Che significa la banca guadagna su certe operazioni? Quanto guadagna?
- Cliente
 - Sarebbe utile associare ad ogni singola operazione fatta dai clienti un costo
 - Variabile in base al tipo di operazione ed in base all'agenzia usata per l'operazione
 - Ovvero persone che prelevano e non sono clienti della banca pagano più dei clienti
 - Ai clienti che vanno in overdraft sarà applicata una penale in forma di interesse bancario

Ing. E. Tramontana - AO design - 30-Nov-06 6

call() this() target()

- call(join point) && this(Person)
 - Il join point specificato deve essere incontrato dentro un oggetto di tipo Person
- call(join point) && this(obj)
 - La variabile obj conterrà il riferimento all'oggetto che sta facendo la chiamata e tale riferimento sarà passato all'advice
- call(join point) && target(Person)
 - Il join point sarà selezionato quando il join point è associato con una istanza di tipo Person
- call(join point) && target(obj)
 - La variabile obj conterrà il riferimento all'oggetto destinatario della chiamata e tale riferimento sarà passato all'advice
- Dove: Person è un tipo e obj è una variabile

Ing. E. Tramontana - AO design - 30-Nov-06 8