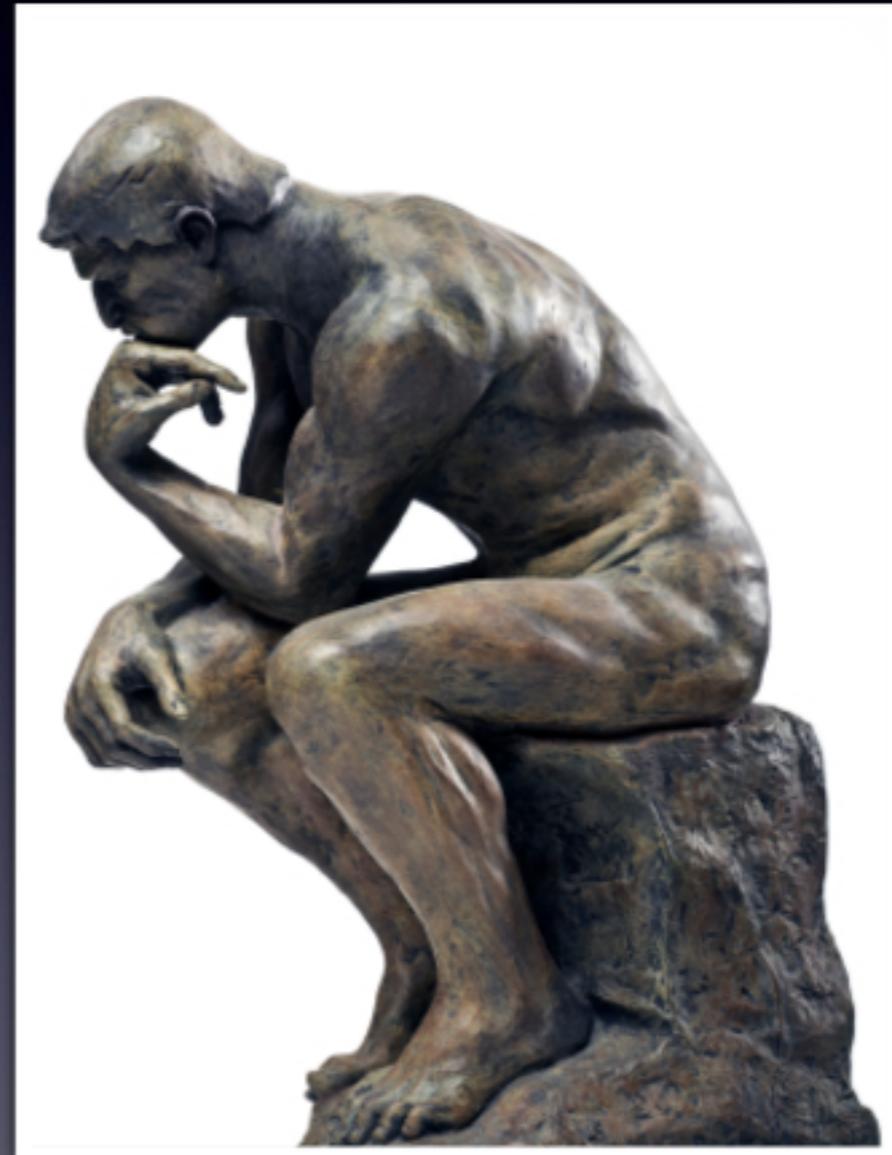


Designmönster

- Vad menas med designmönster?
- UML
- Exempel på designmönster: Singleton och Observer
- GoF
- MVC - Model-View-Controller
- RoR - Ruby on Rails

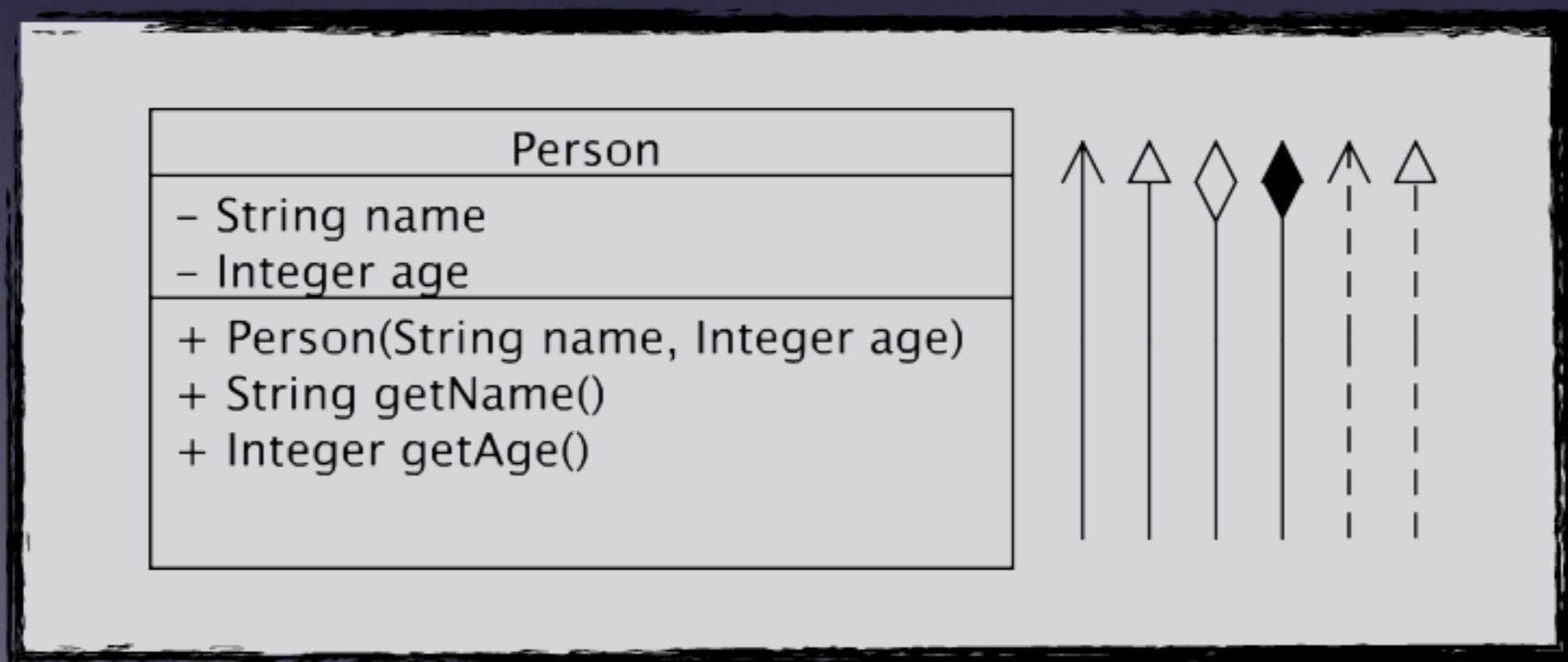
Vad är det?

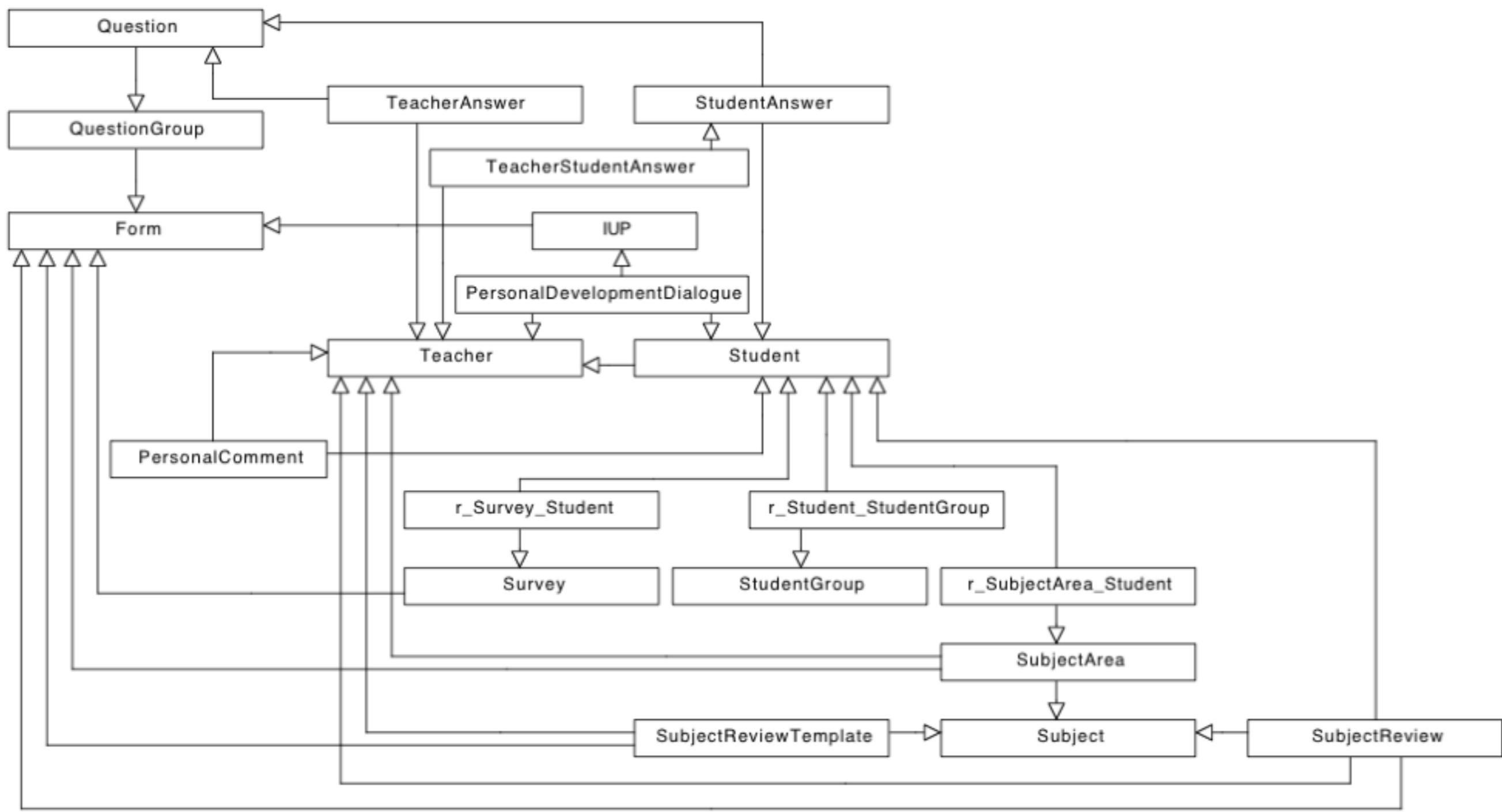
- Språkoberoende designtankar
- Lösningar på vanliga problem
- Flera går att kombinera
- Ger fin, väldoftande kod



UML

- Grafisk representation av mjukvarudesign
- Används vid dokumentation och planering





| Student |
|---|
| <ul style="list-style-type: none"> - id - username - givenName - familyName - password - mentor - email - phone - nationalIdentificationNumber |

| Teacher |
|---|
| <ul style="list-style-type: none"> - id - username - givenName - familyName - password - email - phone - nationalIdentificationNumber |

| Survey |
|--|
| <ul style="list-style-type: none"> - id - name - teacher - form - time - deleted |

| r_Survey_Student |
|--|
| <ul style="list-style-type: none"> - id - survey - student - answered - time - deleted |

| StudentAnswer |
|---|
| <ul style="list-style-type: none"> - id - question - time - student - answer |

| PersonalDevelopmentDialogue |
|---|
| <ul style="list-style-type: none"> - id - student - teacher - iup - done - time |

| StudentGroup |
|--|
| <ul style="list-style-type: none"> - id - name - type - externalId |

| r_Student_StudentGroup |
|---|
| <ul style="list-style-type: none"> - id - student - studentGroup - time - active |

| Subject |
|--|
| <ul style="list-style-type: none"> - id - name |

| TeacherAnswer |
|---|
| <ul style="list-style-type: none"> - id - question - time - teacher - answer |

| IUP |
|--|
| <ul style="list-style-type: none"> - id - plan - followedUp |

| SubjectReview |
|---|
| <ul style="list-style-type: none"> - id - student - teacher - subject - form - deleted - time - visible |

| r_SubjectArea_Student |
|---|
| <ul style="list-style-type: none"> - id - student - subjectArea - deleted |

| SubjectArea |
|---|
| <ul style="list-style-type: none"> - id - name - subject - teacher - form - startTime - endTime - PPText - PPFile - deleted |

| PersonalComment |
|--|
| <ul style="list-style-type: none"> - id - student - teacher - comment - time - deleted |

| Form |
|---|
| <ul style="list-style-type: none"> - id - title |

| QuestionGroup |
|---|
| <ul style="list-style-type: none"> - id - title - type - form |

| Question |
|---|
| <ul style="list-style-type: none"> - id - deleted - questionGroup - type - data - questionOrder |

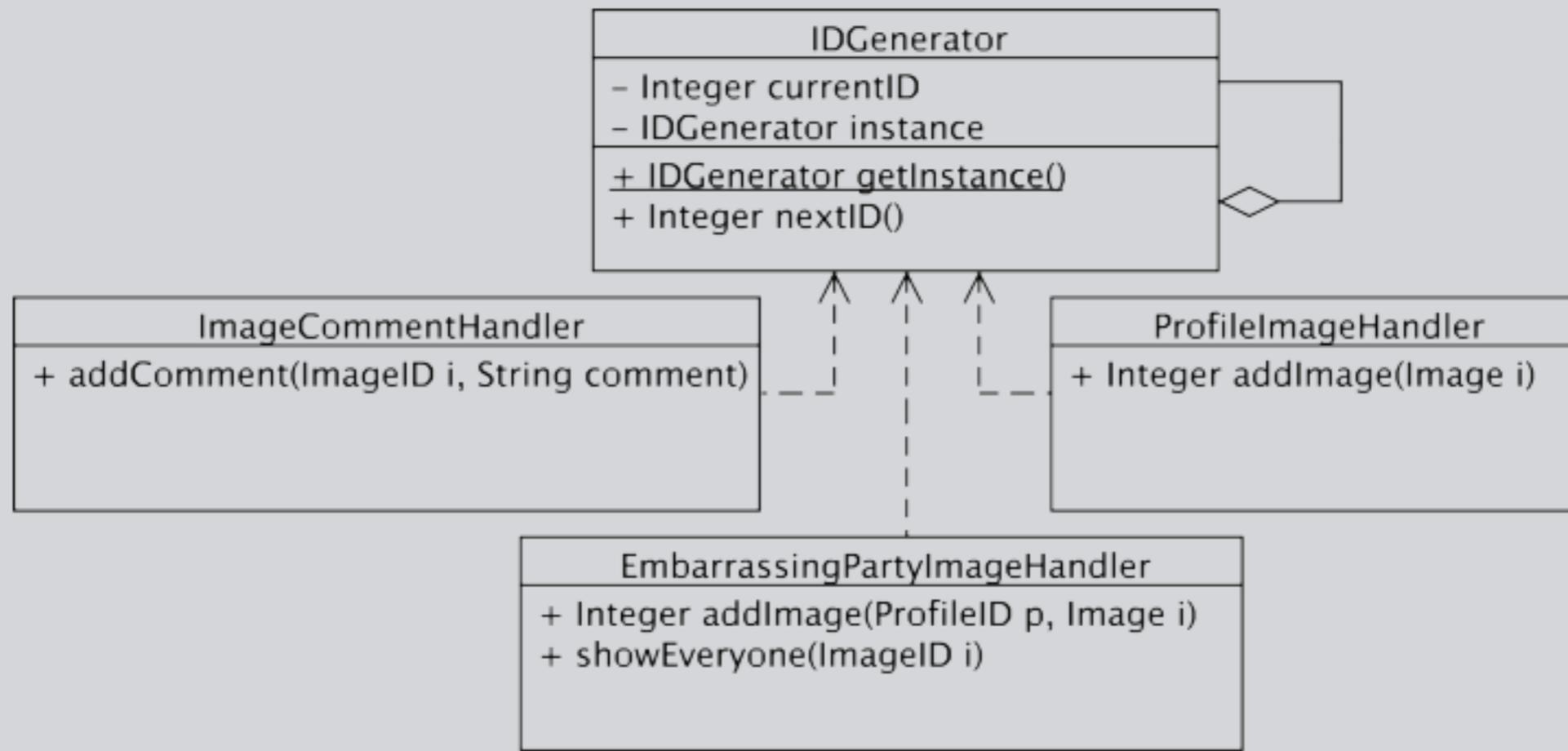
| TeacherStudentAnswer |
|--|
| <ul style="list-style-type: none"> - id - studentAnswer - time - teacher - answer |

| SubjectReviewTemplate |
|---|
| <ul style="list-style-type: none"> - id - name - teacher - time - deleted - subject - form |

Singleton

- Ibland är det bra om en sak är unik
- Exempel:

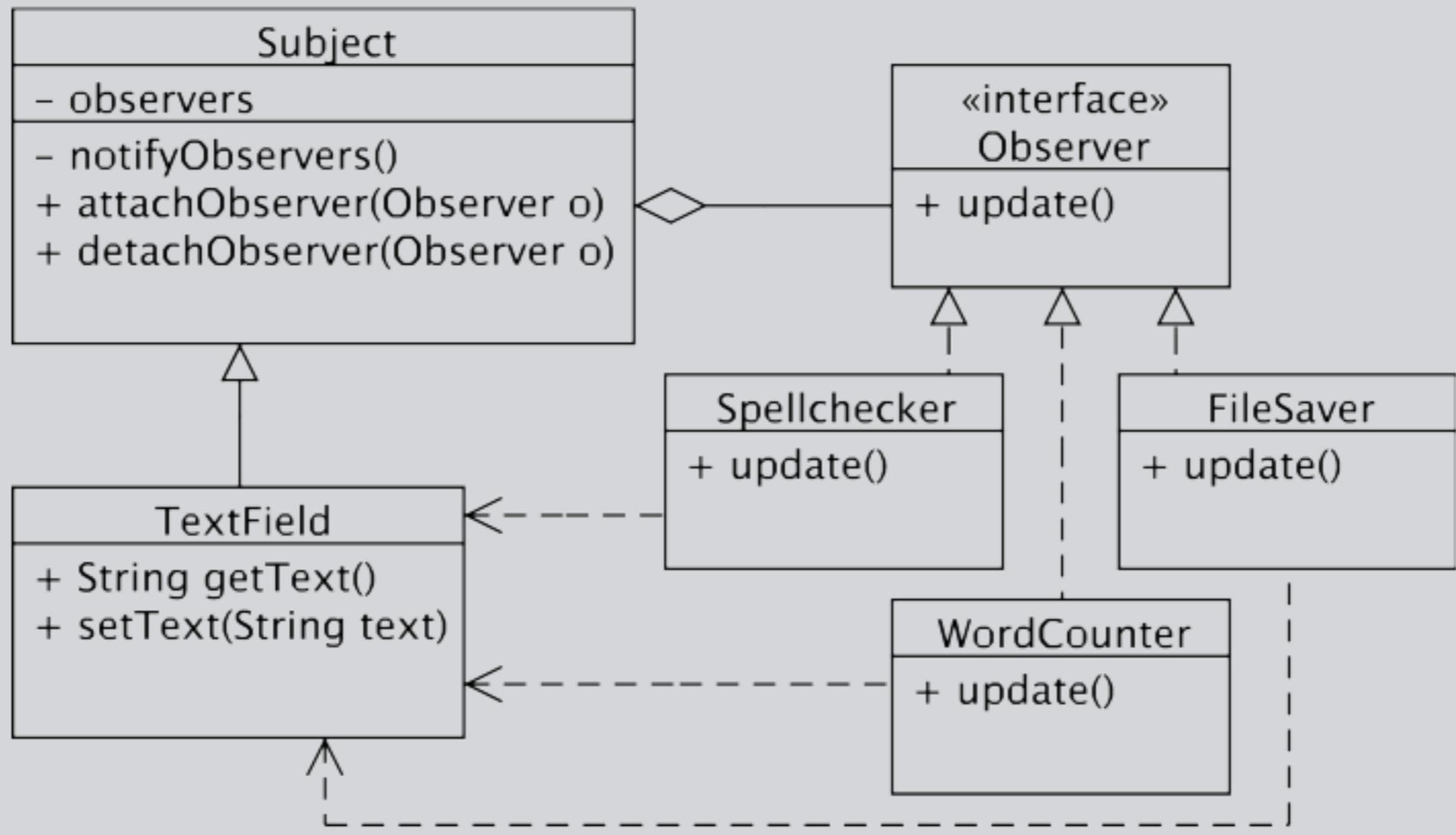
En stor hemsida låter folk ladda upp foton. Hemsidan drivs av en applikation som är igång hela tiden. Varje foto ska få ett eget id, en ständigt ökande siffra. Sifferutdelaren måste vara unik så att olika delar av applikationen inte krockar med varandra.



Observer

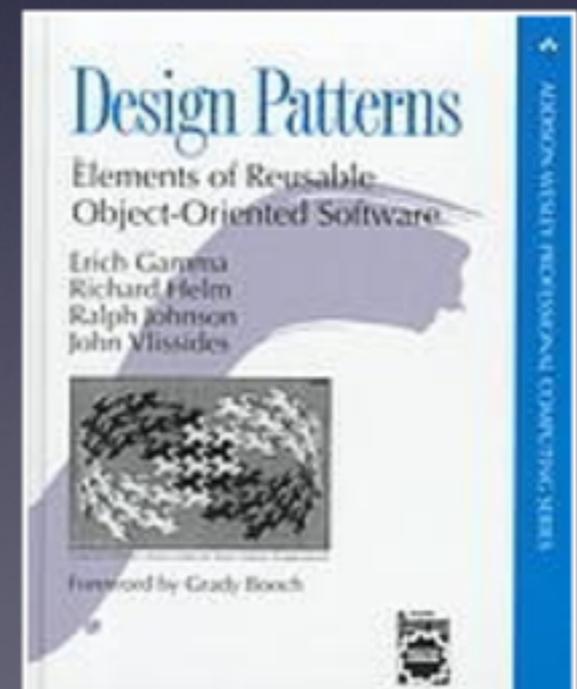
- Observer enkelriktar ett ett-till-många-beroende
- Exempel:

En texteditor har en yta där användaren ska skriva text. Bredvid textytan ska det synas hur många ord användaren skrivit och för varje ord som skrivs in ska det automatiskt rättstavas. Dokumentet ska också sparas automatiskt om texten ändras.



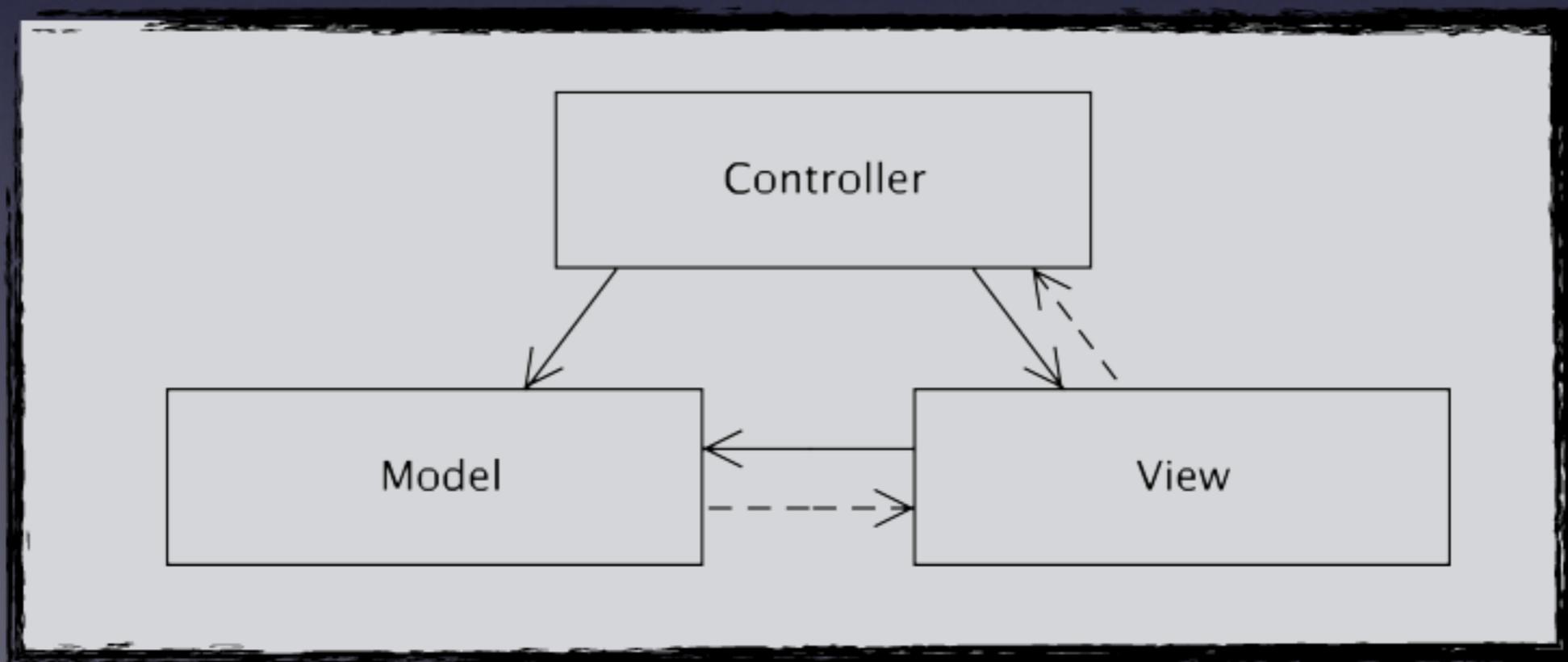
GoF

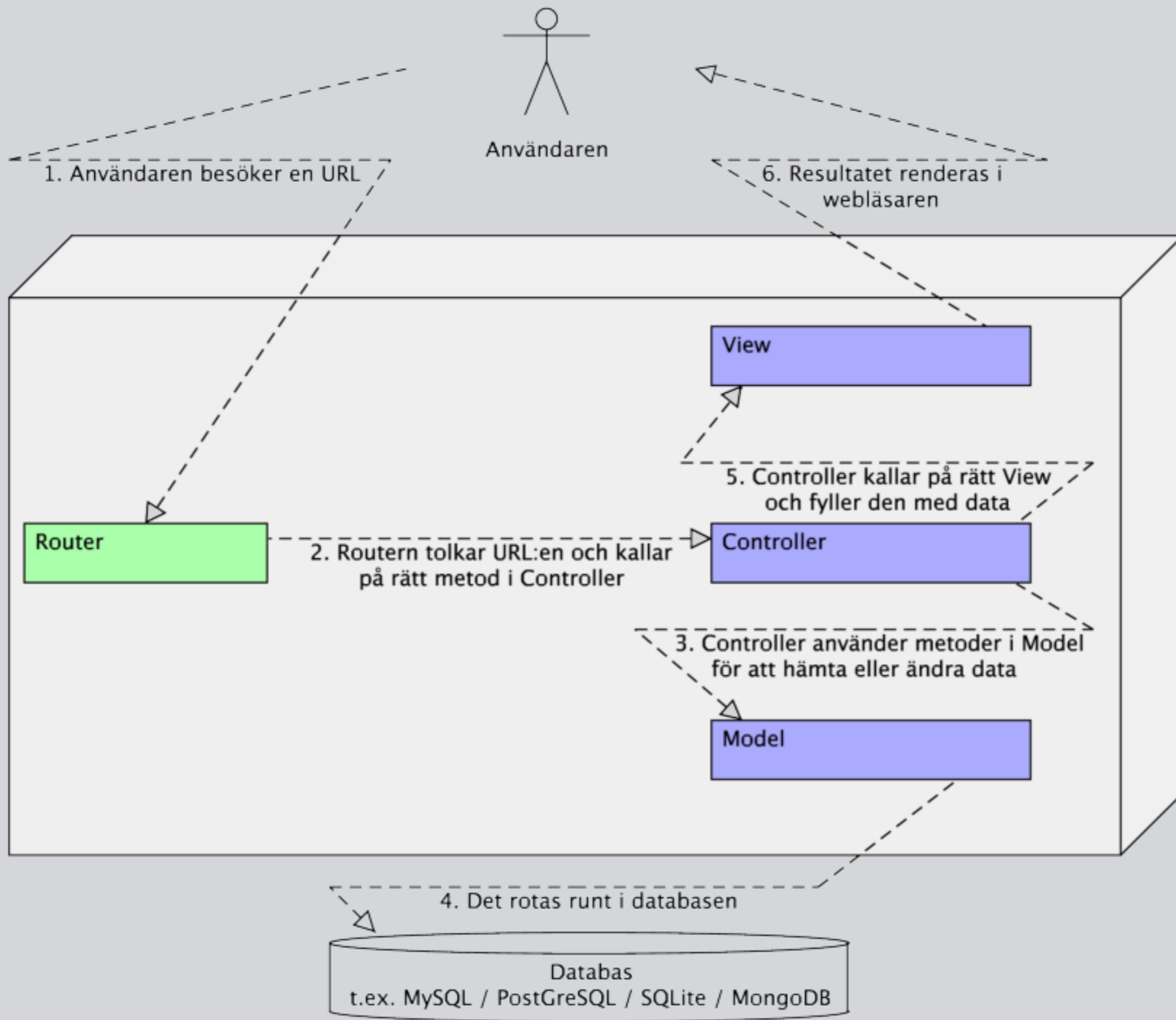
- Erich Gamma, Richard Helm, Ralph Johnson och John Vlissides
- Design Patterns: Elements of Reusable Object-Oriented Software
- 23 stycken lösningar på problem



MVC

- Separerar logik från utseende och data
- Väldigt användbart när man bygger grafiska applikationer, exempelvis hemsidor





Vilka använder MVC?

- Asp .NET MVC
- Ruby on Rails
- Java Spring MVC
- Qt
- Cocoa
- och många andra...

Singleton i PHP

Snyggt när man ska hantera en databas

```
<?php
class Database
{
    private static $instance = null;

    public static function get_instance()
    {
        if (self::$instance == null)
            self::$instance = new Database();

        return self::$instance;
    }

    private function Database()
    {
        .....
    }
}
?>
```

```
<?php
```

```
    $db = Database::get_instance();
```

```
?>
```

Det sparar resurser om man
behöver komma åt databasen
från många ställen.