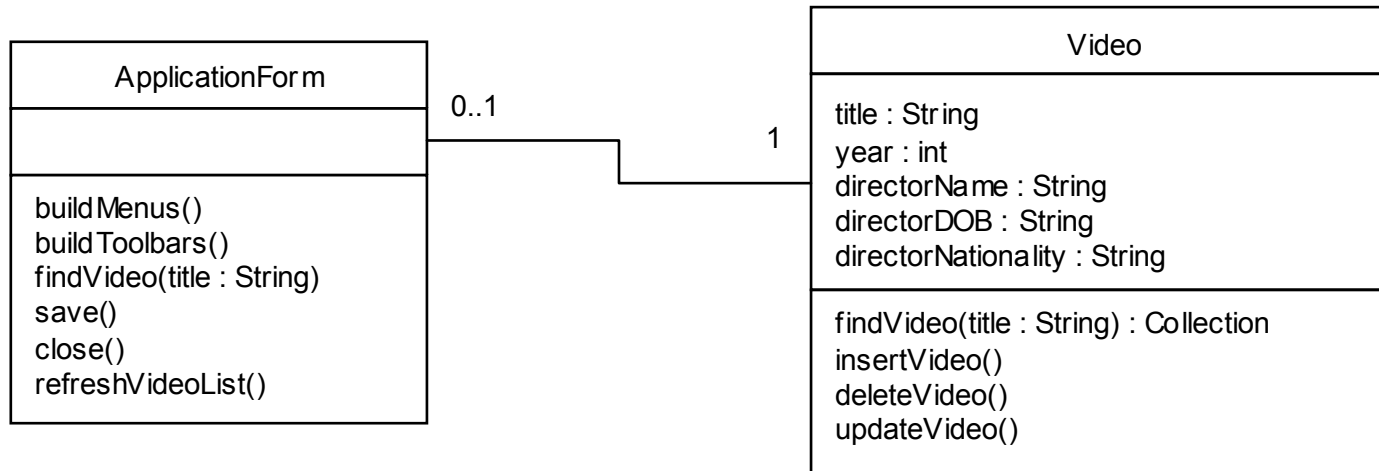


# OO Design Principles & Metrics

Exercises

# Single Responsibility

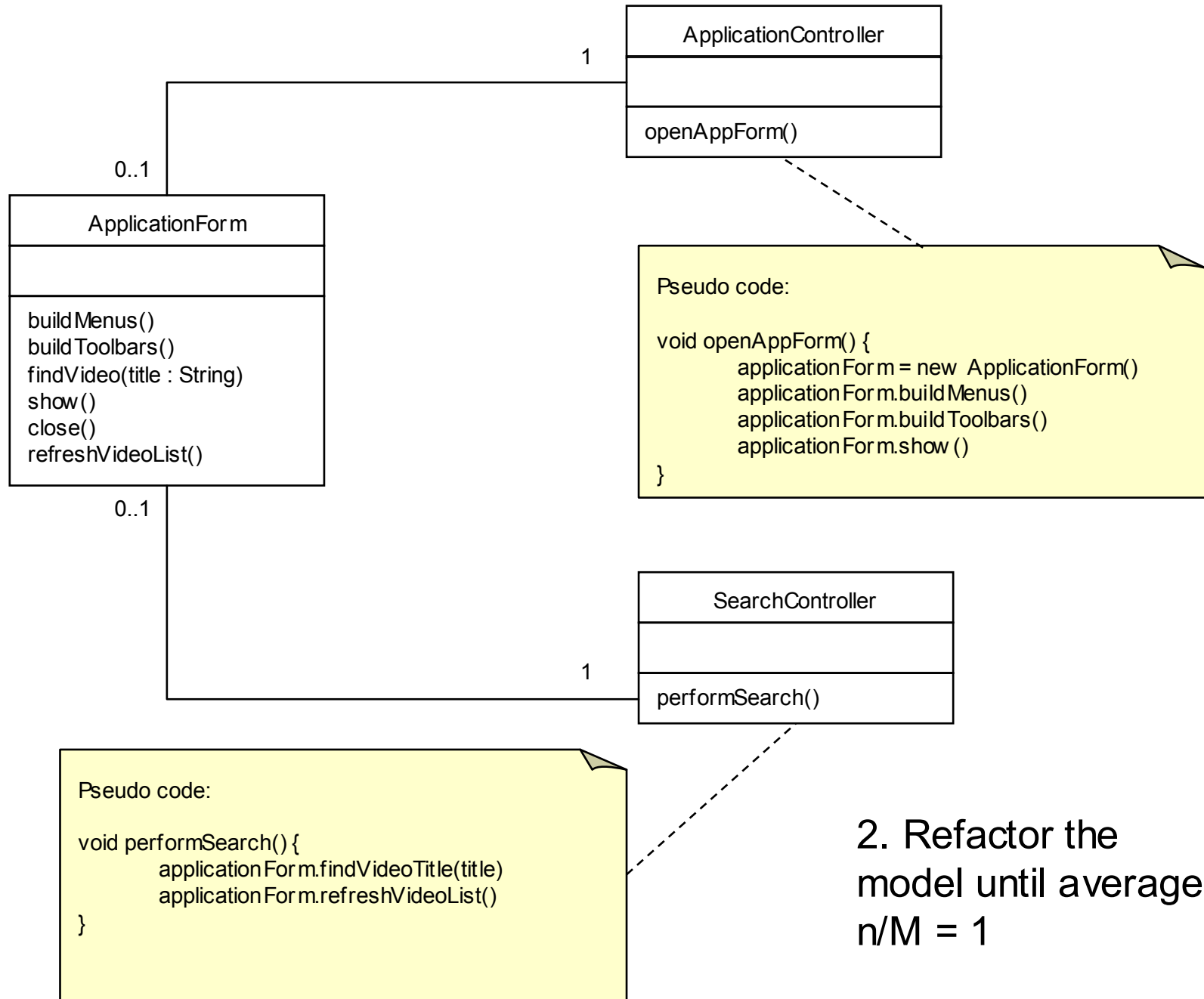
1. Calculate the responsibilities / class for the following model



2. Refactor the model until responsibilities / class = 1

# Interface Segregation

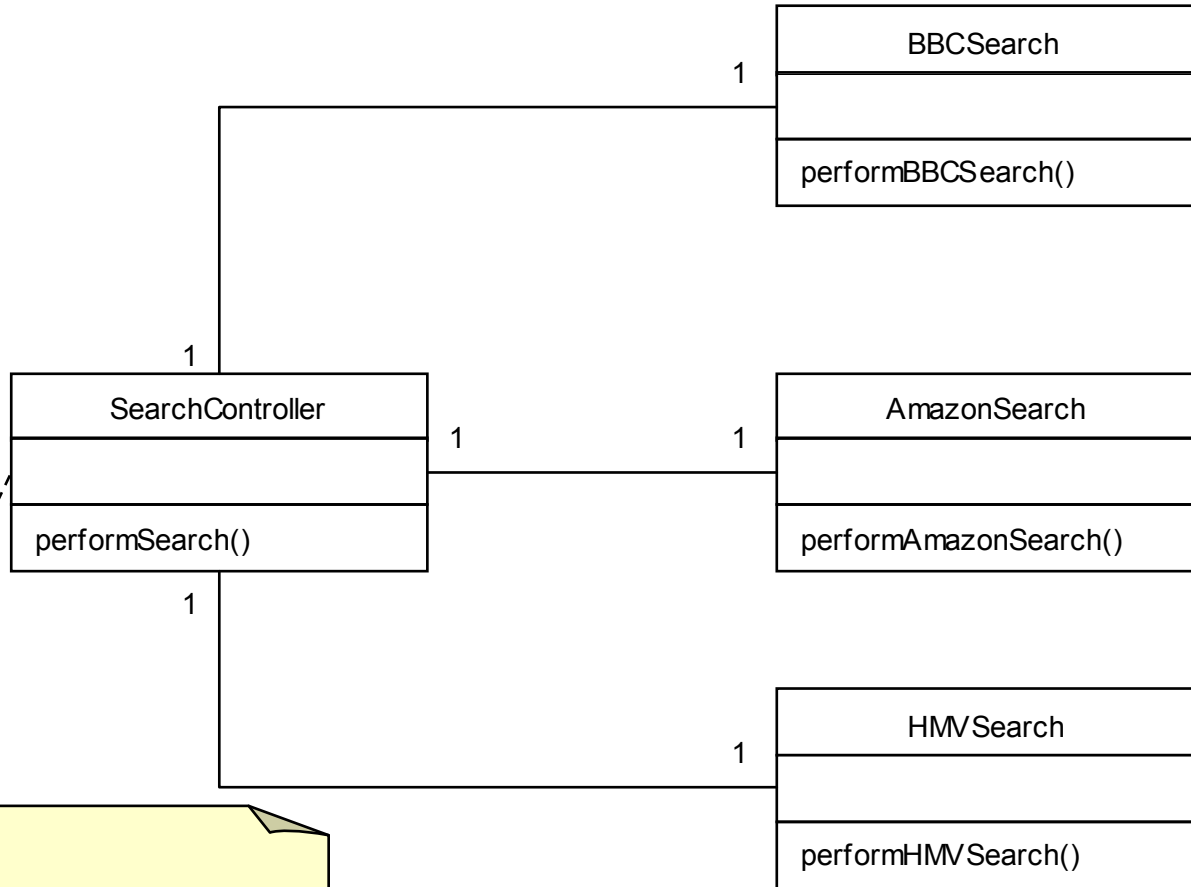
# 1. Calculate average n/M for the ApplicationForm class



2. Refactor the model until average  $n/M = 1$

# Dependency Inversion

# 1. Calculate dependencies on abstractions / total dependencies for this model



Pseudo code:

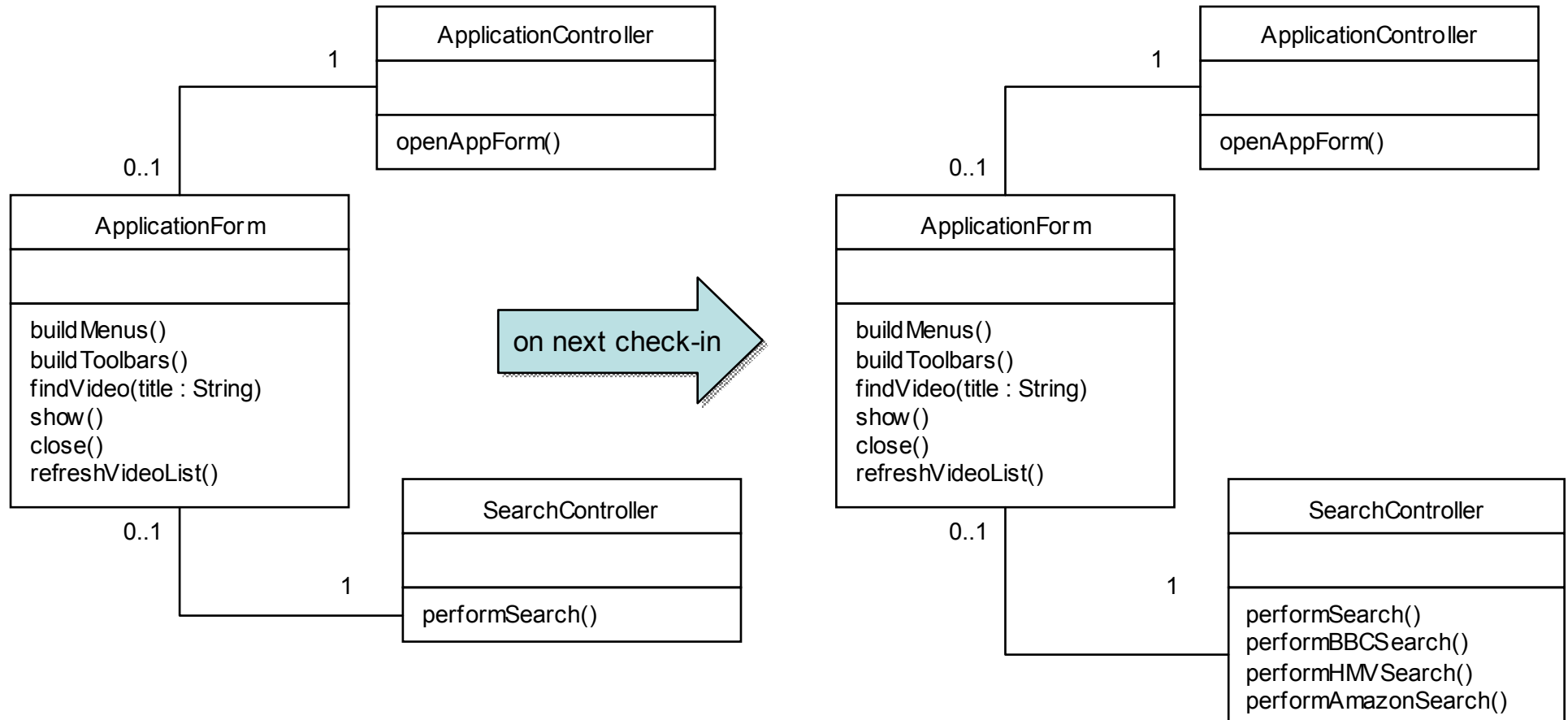
```
void performSearch() {
    results = new ArrayList()
    results.add(bbcSearch.performBBCSearch())
    results.add(amazonSearch.performAmazonSearch())
    results.add(hmvSearch.performHMVSearch())
}
```

# 2. Refactor the model until dependencies on abstractions / total dependencies = 1

Open-Closed



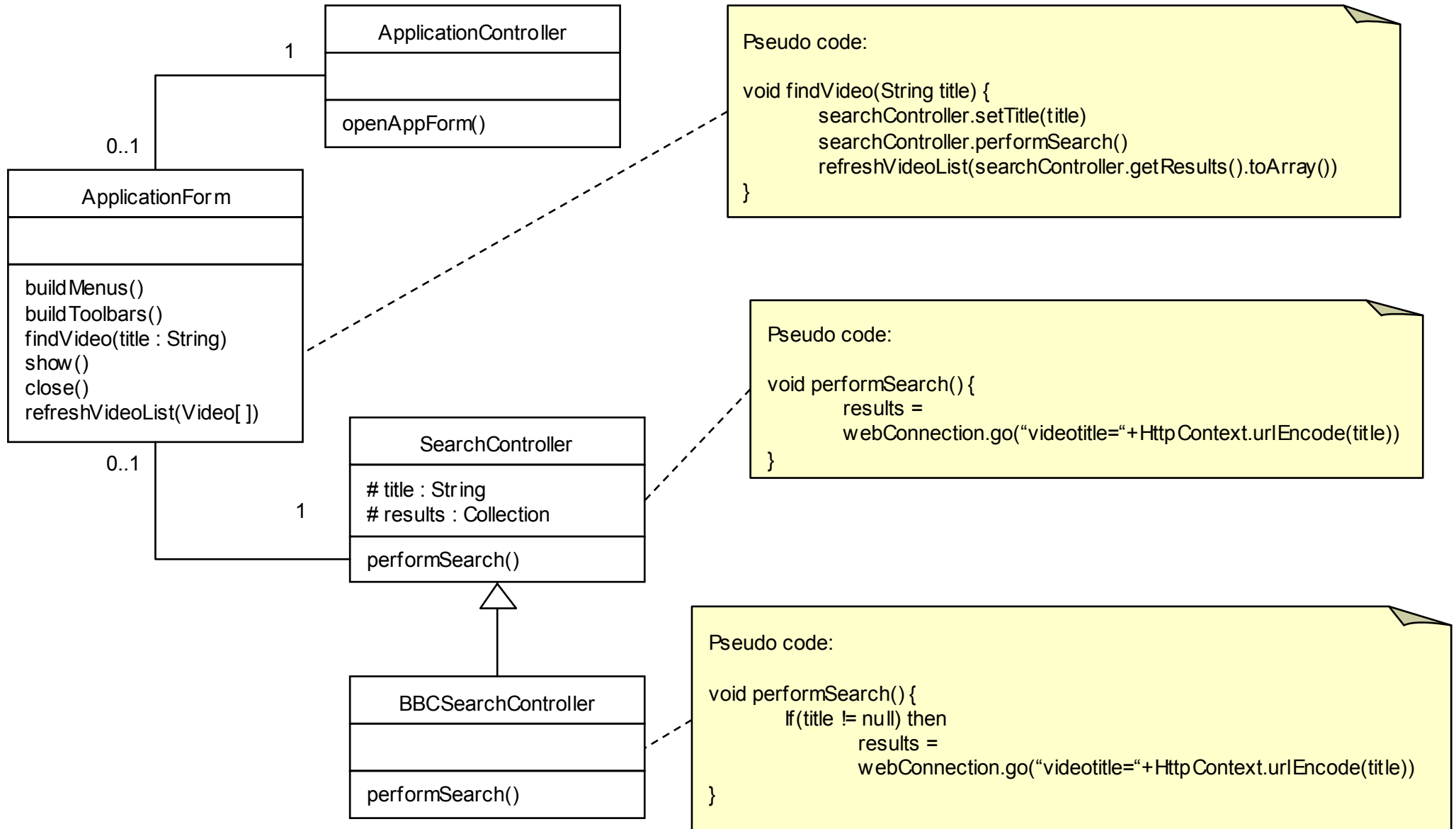
1. Calculate classes extended and not modified / classes extended and/or modified for this pair of models



2. How could this change have been done so that classes extended and not modified / classes extended and/or modified = 1?

# Liskov Substitution

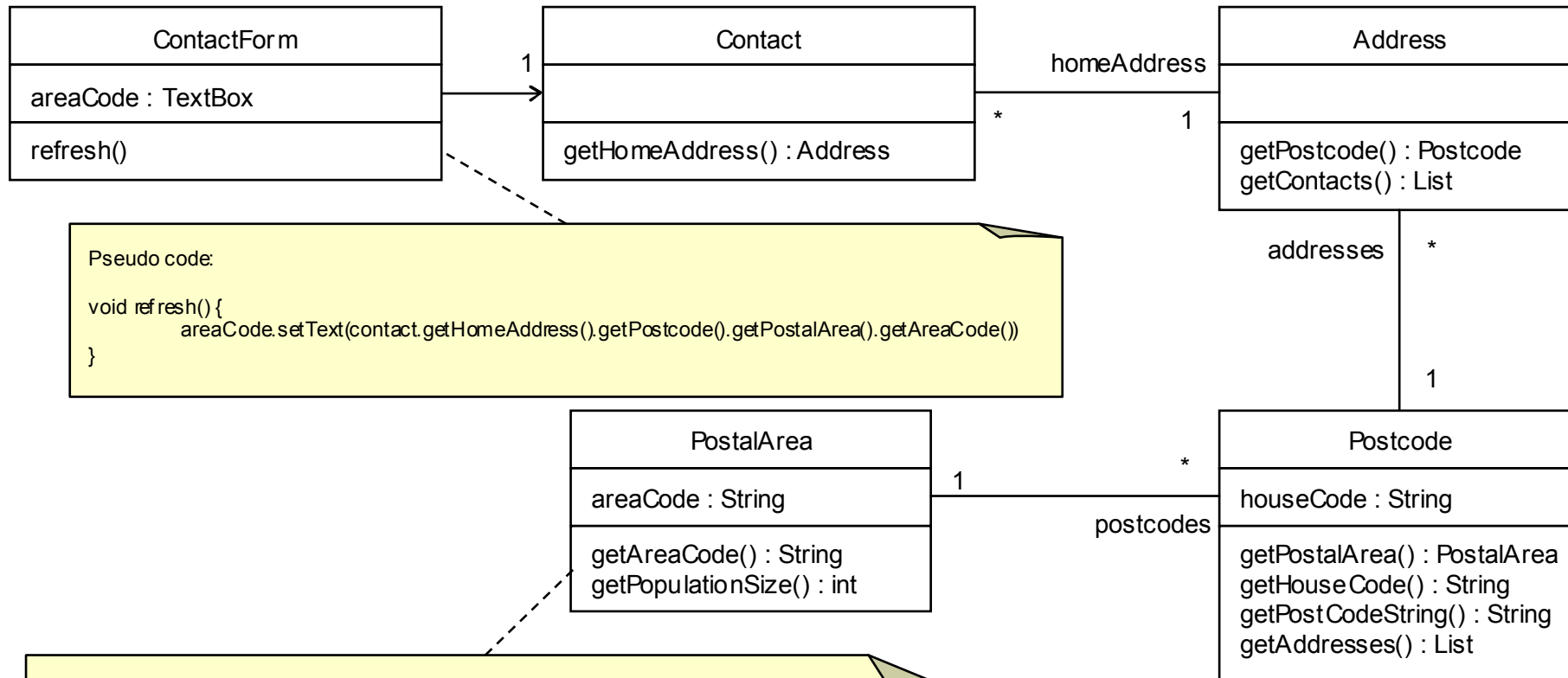
# 1. What is wrong with this model?



2. Refactor the model so that the client won't break when BBCSearchController is substituted for SearchController

# Law of Demeter

# 1. Calculate the average depth of navigation for this model



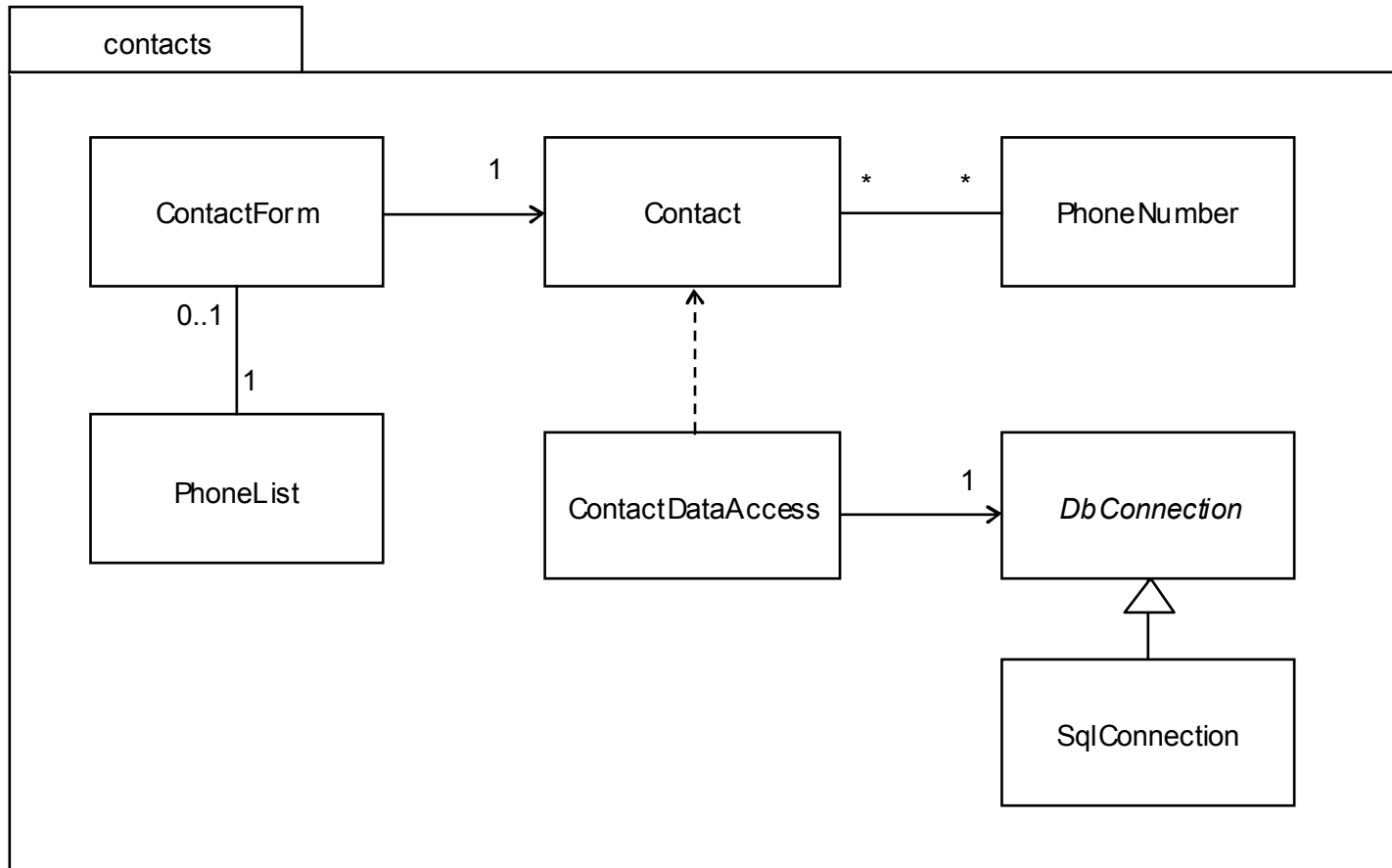
```
Pseudo code:
void refresh() {
    areaCode.setText(contact.getHomeAddress().getPostcode().getPostalArea().getAreaCode())
}
```

```
Pseudo code:
int getPopulationSize() {
    int population = 0
    for each Postcode postcode in postcodes {
        for each Address address in postcode.getAddresses() {
            for each Contact contact in address.getContacts() {
                population++;
            }
        }
    }
    return population;
}
```

2. Refactor the model so that the average depth of navigation is 1

# Package Cohesion

# 1. Calculate the cohesion of this package

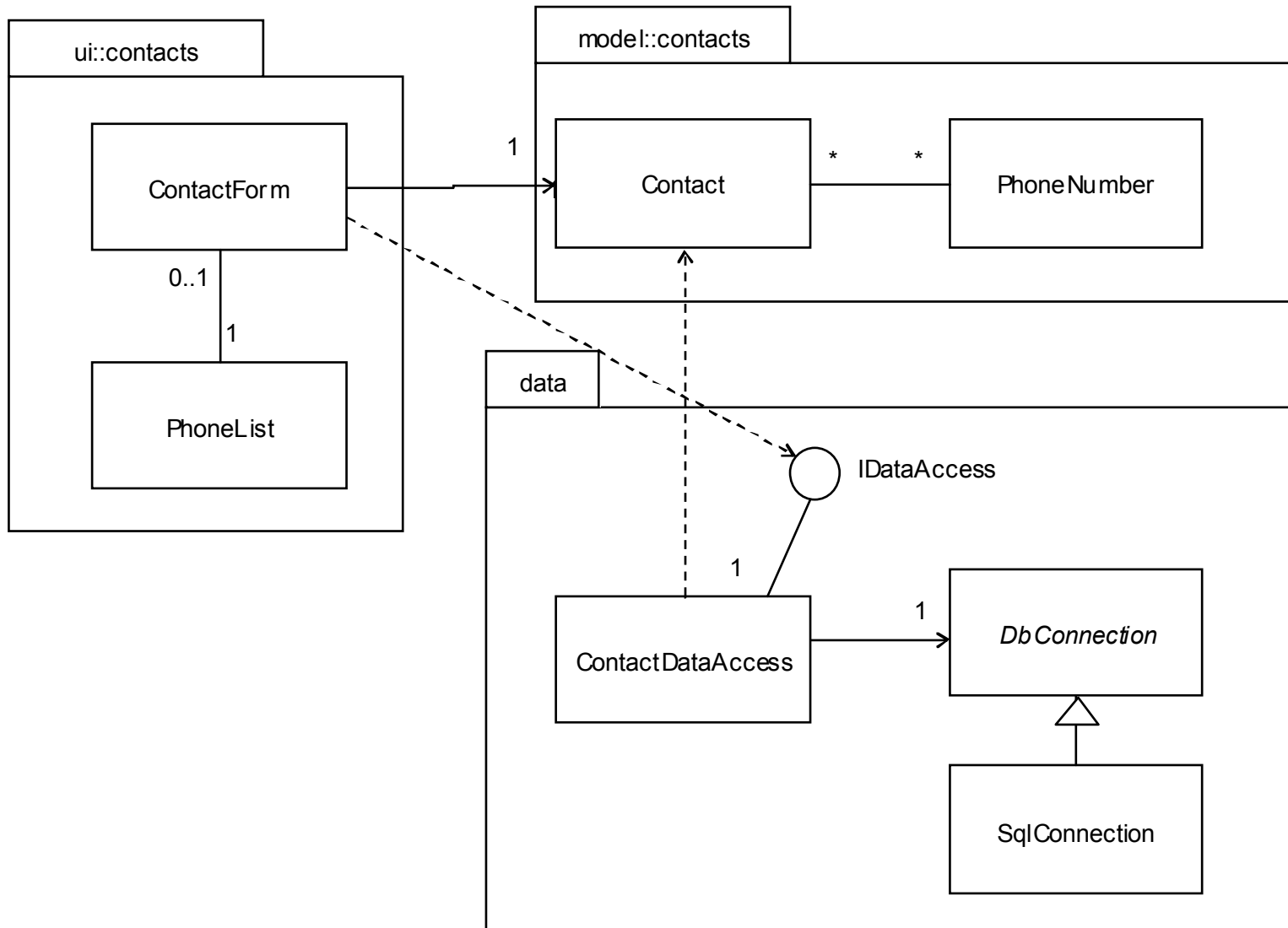


# 2. Refactor this model to make the package(s) more cohesive

# Package Coupling



1. Calculate the abstractness and instability of each package
2. Calculate the normalised distance from the main sequence of each package



3. Refactor the model to reduce the distance from the main sequence for every package where  $D > 0$