

CS 635 Advanced Object-Oriented Design & Programming
Spring Semester, 2009
Doc 4 Code Smells
Feb 3, 2009

Copyright ©, All rights reserved. 2009 SDSU & Roger Whitney, 5500
Campanile Drive, San Diego, CA 92182-7700 USA. OpenContent (<http://www.opencontent.org/opl.shtml>) license defines the copyright on this document.

References

Refactoring: Improving the Design of Existing Code, Fowler
Chapter 3

Duplicate Code

Duplicate Code

Duplicate Code

Duplicate Code

Duplicate Code

Duplicate Code

Duplicate Code

Duplicate Code

Long Method - Large Class

The average method size should be less than 8 lines of code (LOC) for Smalltalk and 24 LOC for C++

The average number of methods per class should be less than 20

The average number of fields per class should be less than 6.

The class hierarchy nesting level should be less than 6

The average number of comment lines per method should be greater than 1

Long Parameter List

Divergent Change

One class is changed in different ways for different reasons

ShotGun Surgery

When you have to make a kind of change you have to make a lot of little changes in different locations

Feature Envy

A method seems more interested in a class other than the one it is in.

Data Clumps

Same three or four data items together in lots of places

Primitive Obsession

Using primitive types instead of creating small classes

Switch Statements

How do you program without them?

Lazy Class

Class that is not doing enough to pay for itself

Data Class

Class with just fields and setter/getter methods

Data classes are like children.

They are okay as a starting point, but to participate as a grownup object, they need to take some responsibility

Inappropriate Intimacy

Classes that spend too much time delving into other classes private parts

Message Chains

```
location = rat.getRoom().getMaze().getLocation()
```

Negative Slope

```
if (foo) {  
    if (bar) {  
        if (cat = dog) {  
            if (rat < 10) {  
                ...  
            }  
        }  
    }  
}
```

Temporary Field

Field is only used in certain circumstances

Common case

- field is only used by an algorithm

- Don't want to pass around long parameter list

- Make parameter a field

Refused Bequest

Subclass does not want to support all the methods of parent class

Subclass should support the interface of the parent class