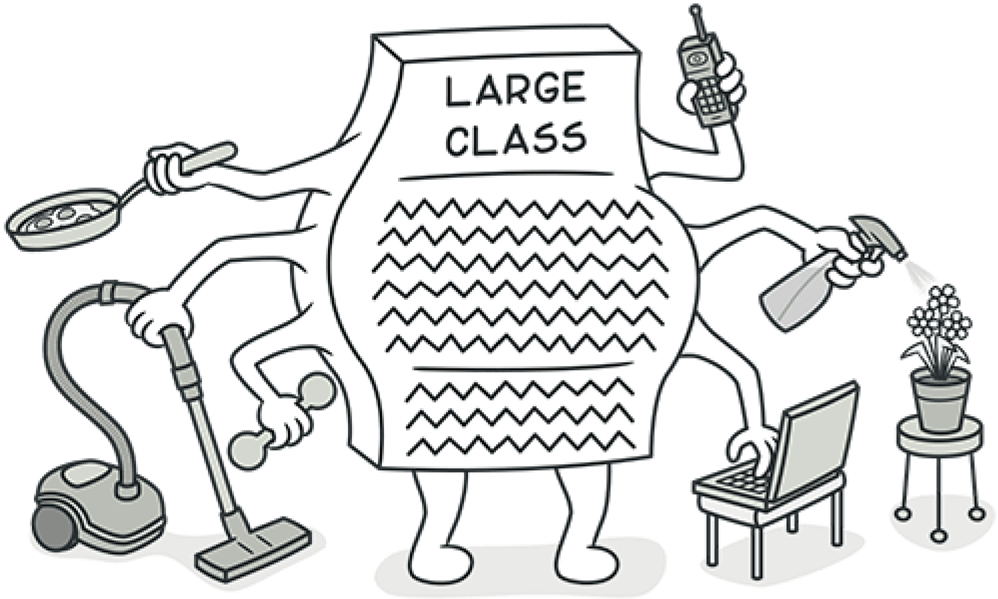
/ Refactoring / Code Smells / Bloaters

Large Class

Signs and Symptoms

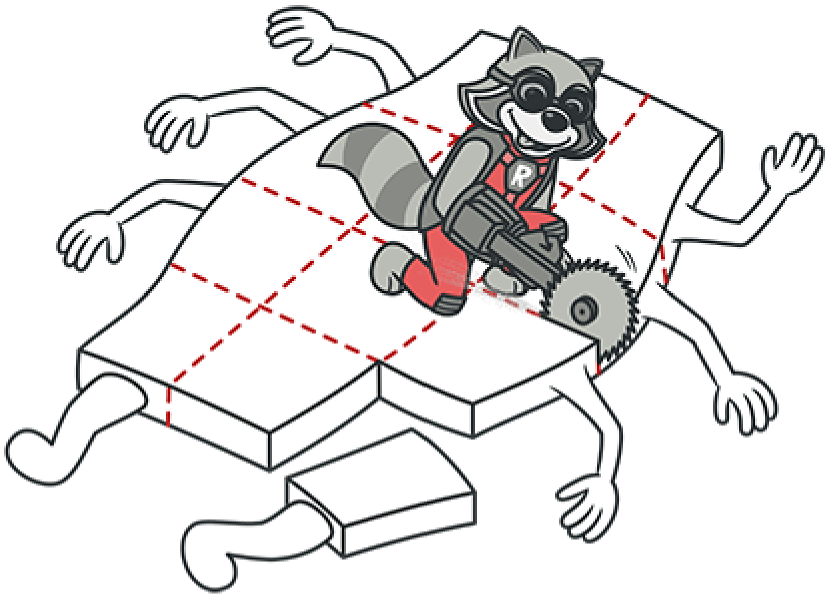
A class contains many fields/methods/lines of code.



Reasons for the Problem

Classes usually start small. But over time, they get bloated as the program grows.

As is the case with long methods as well, programmers usually find it mentally less taxing to place a new feature in an existing class than to create a new class for the feature.

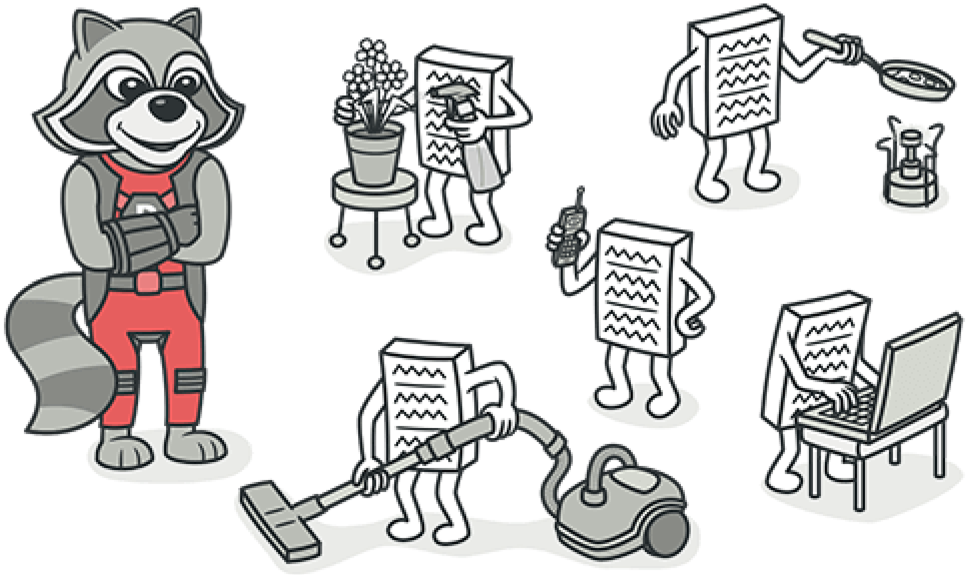


Treatment

When a class is wearing too many (functional) hats, think about splitting it up:

- **Extract Class** helps if part of the behavior of the large class can be spun off into a separate component.
- **Extract Subclass** helps if part of the behavior of the large class can be implemented in different ways or is used in rare cases.
- **Extract Interface** helps if it's necessary to have a list of the operations and behaviors that the client can use.

- If a large class is responsible for the graphical interface, you may try to move some of its data and behavior to a separate domain object. In doing so, it may be necessary to store copies of some data in two places and keep the data consistent. **Duplicate Observed Data** offers a way to do this.



Payoff

- Refactoring of these classes spares developers from needing to remember a large number of attributes for a class.

- In many cases, splitting large classes into parts avoids duplication of code and functionality.