Object oriented design - SOLID

Overview

- How to build complex software?
  - Structured programming
    - gotos considered harmful
  - Object oriented programming (OOP)
- Language support alone isn't enough
  - (or even strictly necessary)
  - How we use the tools matters!
- Objected oriented design (OOD)
  - How do we design software that is easy to modify, extend and maintain?

Overview

- Robert C. Martin "Uncle Bob"
  - Author:
    - Clean Code: A Handbook of Agile Software Craftsmanship
    - Agile Software Development, Principles, Patterns, and Practices
    - Agile Principles, Patterns, and Practices in C#
    - The Clean Coder: A Code of Conduct for Professional Programmers
    - …

http://butunclebob.com/ArticleS.UncleBob.PrinciplesOfOod

SOLID

- How to manage dependencies in code?
  - The first five principles - SOLID

SRP	Single responsibility principle
OCP	Open/closed principle
LSP	Liskov substitution principle
ISP	Interface segregation principle
DIP	Dependency inversion principle

SOLID

Software Development is not a Jenga game

http://lostechies.com/derickbailey/2009/02/11/solid-development-principles-in-motivational-pictures/

Single Responsibility Principle (SRP)

"A class should have only one reason to change."

- Strive for cohesion
  - Functional relatedness of the elements of a module
  - The forces that cause a module, or a class, to change

SRP	Single responsibility principle
OCP	Open/closed principle
LSP	Liskov substitution principle
ISP	Interface segregation principle
DIP	Dependency inversion principle

SINGLE RESPONSIBILITY PRINCIPLE

Just Because You Can, Doesn't Mean You Should

http://lostechies.com/derickbailey/2009/02/11/solid-development-principles-in-motivational-pictures/

Open/closed principle (OCP)

"Classes should be open for extension,
but closed for modification."

- Principle underlying common heuristics:
  - All instance variable should be private
  - Avoid global variables
- Design modules that *never change*

SRP	Single responsibility principle
OCP	Open/closed principle
LSP	Liskov substitution principle
ISP	Interface segregation principle
DIP	Dependency inversion principle

OPEN CLOSED PRINCIPLE

Open Chest Surgery Is Not Needed When Putting On A Coat

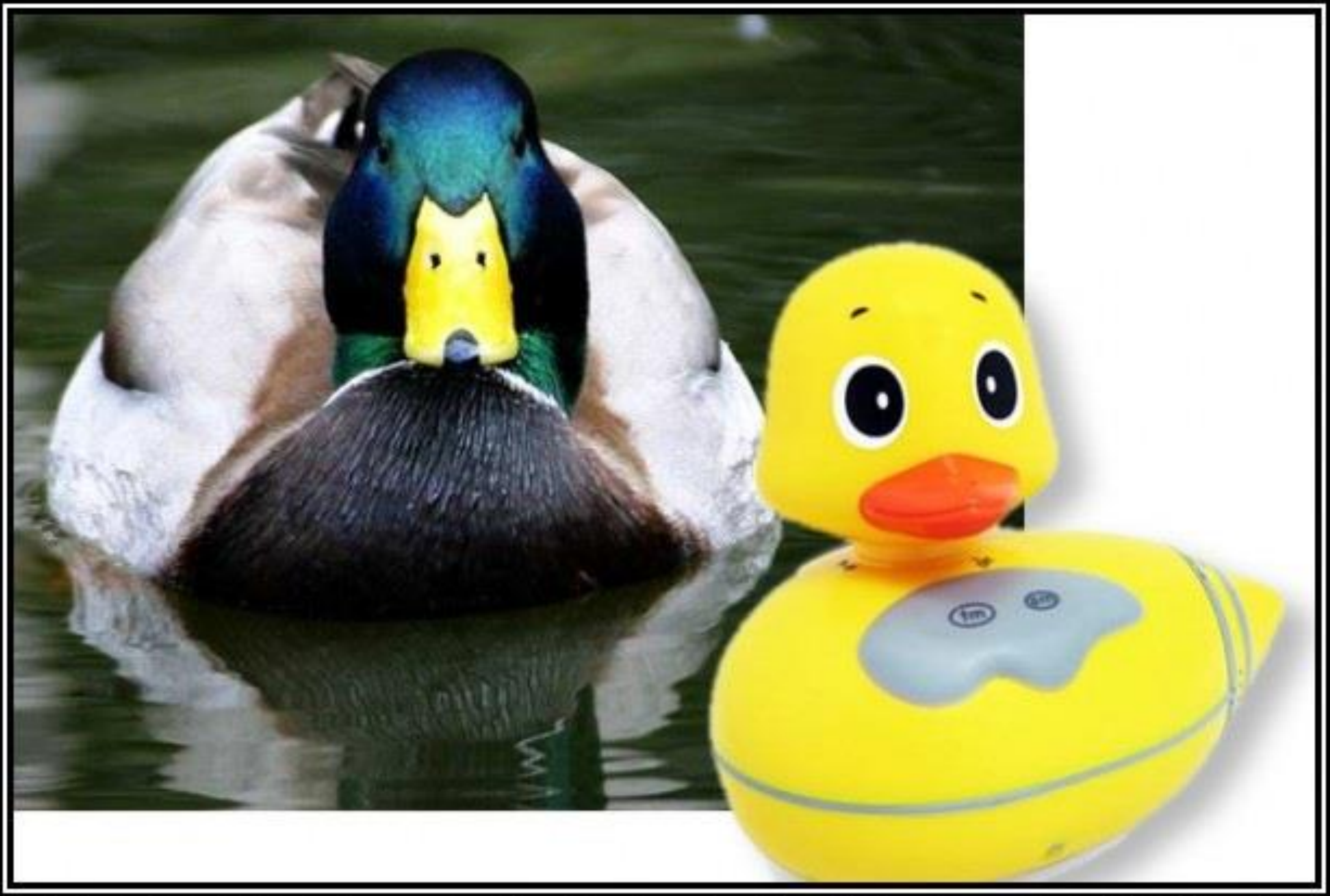
http://lostechies.com/derickbailey/2009/02/11/solid-development-principles-in-motivational-pictures/

Liskov Substitution principle

> "Derived classes must be substitutable for their base classes."

- The base class should not have to know about all of its derivatives
  - This would violate OCP
  - e.g. subclassing Square from Rectangle

SRP	Single responsibility principle
OCP	Open/closed principle
LSP	Liskov substitution principle
ISP	Interface segregation principle
DIP	Dependency inversion principle



LISKOV SUBSTITUTION PRINCIPLE

If It Looks Like A Duck, Quacks Like A Duck, But Needs Batteries - You Probably Have The Wrong Abstraction

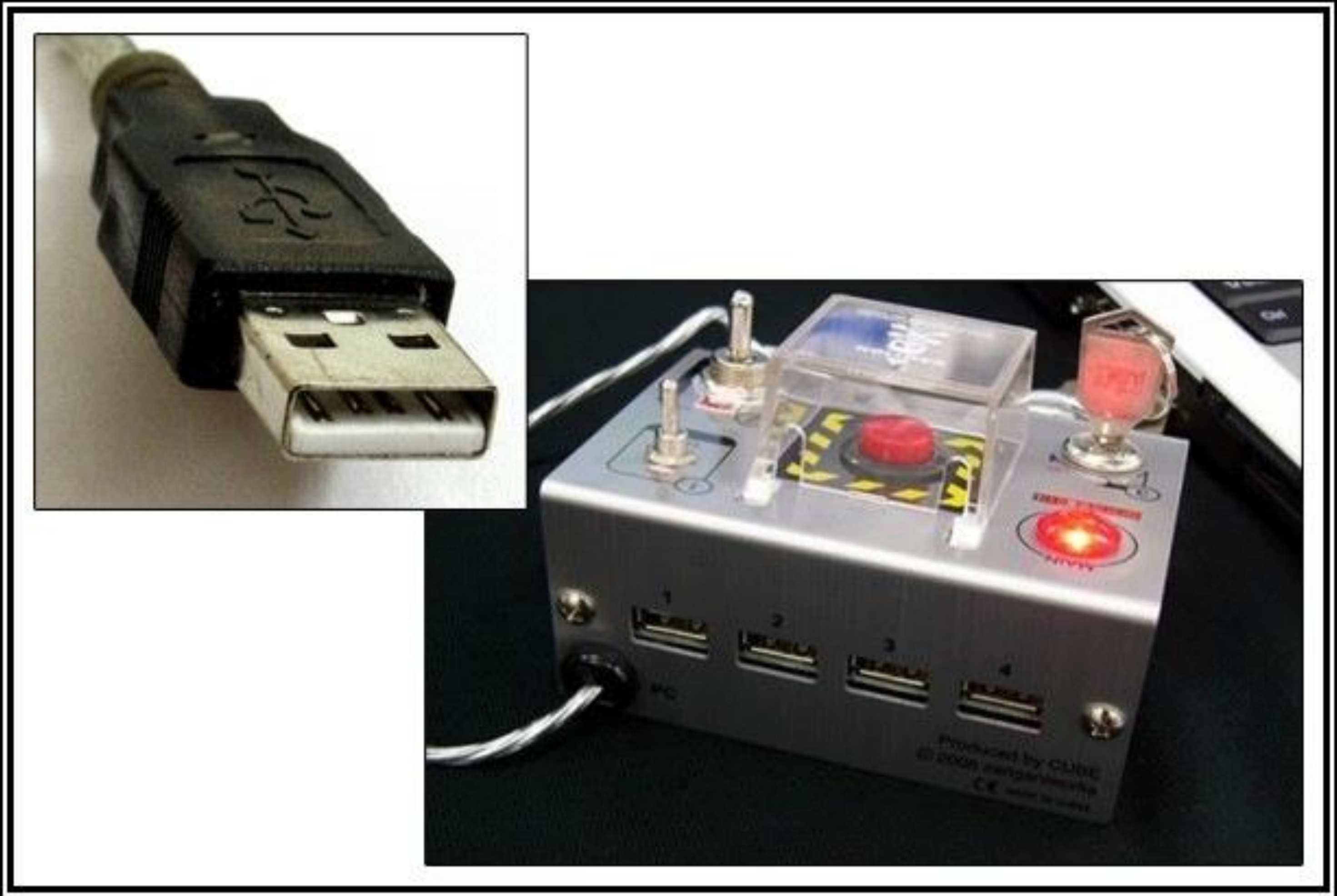
http://lostechies.com/derickbailey/2009/02/11/solid-development-principles-in-motivational-pictures/

Interface segregation principle (ISP)

> "Clients should not be forced to depend upon interface that they do not use."

- We don't want "fat" or "polluted" interfaces
  - Interfaces that are not specific to a single client
  - Leads to inadvertent coupling between clients that would otherwise be isolated

SRP	Single responsibility principle
OCP	Open/closed principle
LSP	Liskov substitution principle
ISP	Interface segregation principle
DIP	Dependency inversion principle



INTERFACE SEGREGATION PRINCIPLE

You Want Me To Plug This In, Where?

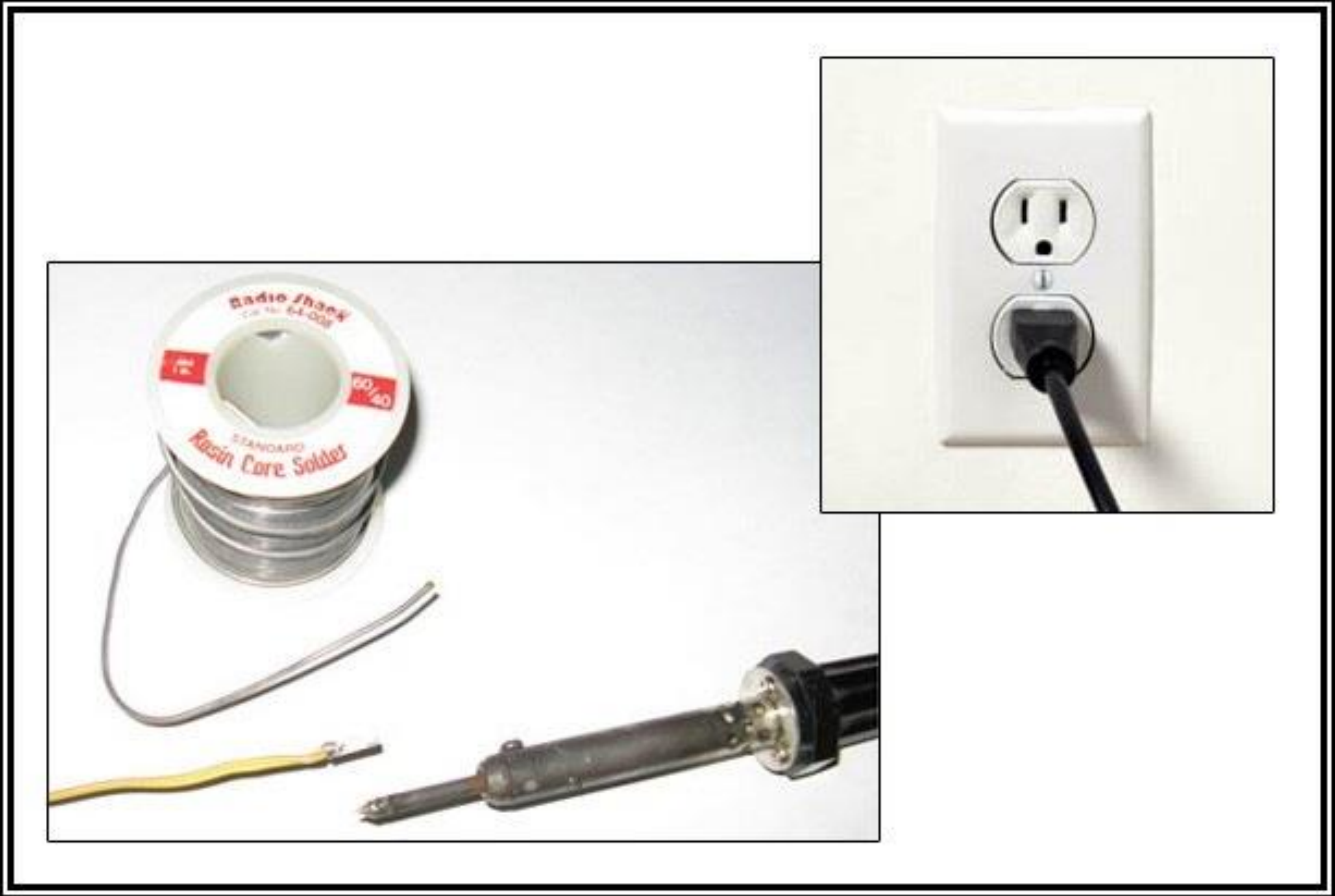
http://lostechies.com/derickbailey/2009/02/11/solid-development-principles-in-motivational-pictures/

Dependency inversion principle (DIP)

> "Depend on abstractions. Do not depend upon concrete classes."

- High-level components should not depend on low-level components
  - Both should depend on abstractions
    - Abstract classes should not depend upon concrete classes
    - Concrete classes should depend on abstract classes

SRP	Single responsibility principle
OCP	Open/closed principle
LSP	Liskov substitution principle
ISP	Interface segregation principle
DIP	Dependency inversion principle



DEPENDENCY INVERSION PRINCIPLE

Would You Solder A Lamp Directly To The Electrical Wiring In A Wall?

http://lostechies.com/derickbailey/2009/02/11/solid-development-principles-in-motivational-pictures/