Object orientation

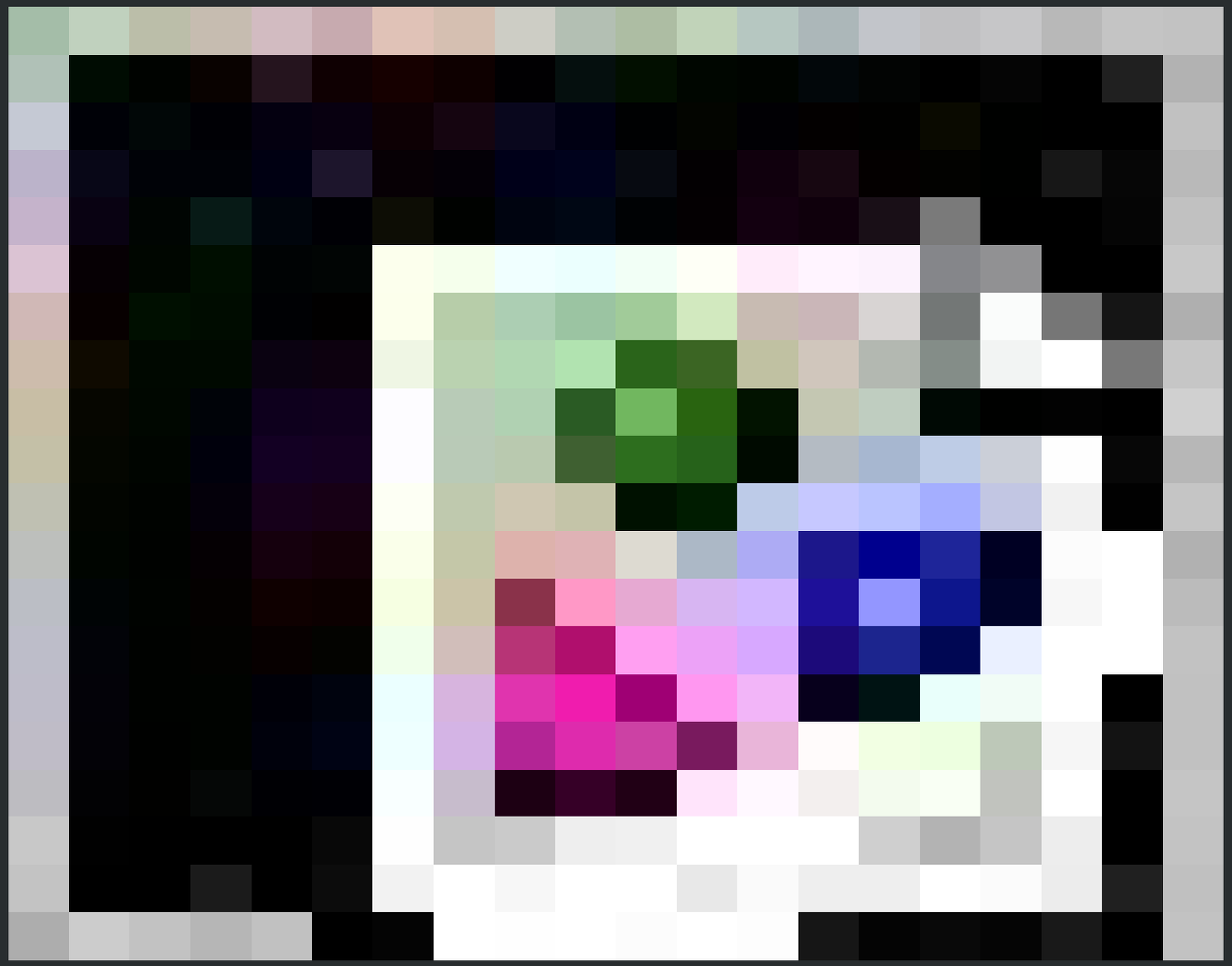
Alan Kay

Computing [...] it’s a place where you don't have to be a Ph.D. or anything else. It’s a place where you can still be an artisan. People are willing to pay you if you're any good at all, and you have plenty of time for screwing around.

What is Object Orientation?

Computer programs are designed by making them out of objects that interact with one another.

Classes are blueprints

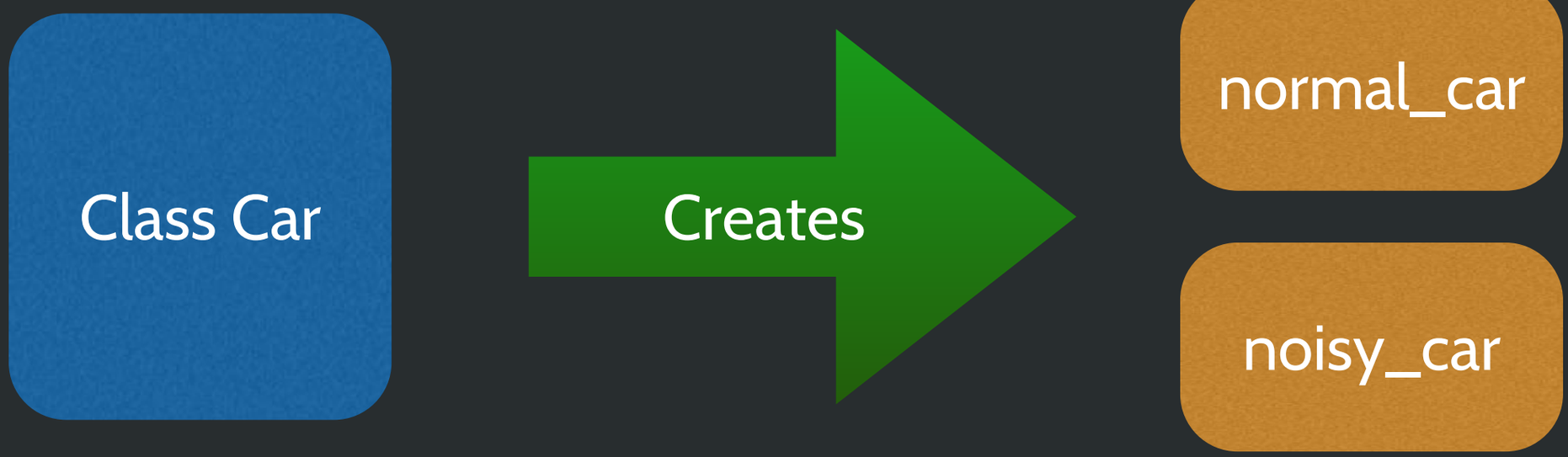


Exercise

Create a class that represents a Car. That car should have a method to make noise that prints “Broom” to the console.

Then create one car and call that method

Classes vs Instances



The constructor

- A special method called initialize

- It is executed for each instance we create

Instance variables

- Its name starts with the @ sign
- Each instance of a class has its own value for the instance variables

Exercise

Modify our Car class so we can have different sounds for different cars. Each car should have its own sound. Create two cars. One of the should make “Broom” and the other should make “BROOOOOOOM”.

Class variables

- Its name starts with the `@@` sign
- All instances of a class share the same value for this variable

Instance methods and class methods

- By default all methods are instance methods
- We can also have methods for a class if we prepend its name with self. during declaration

Exercise

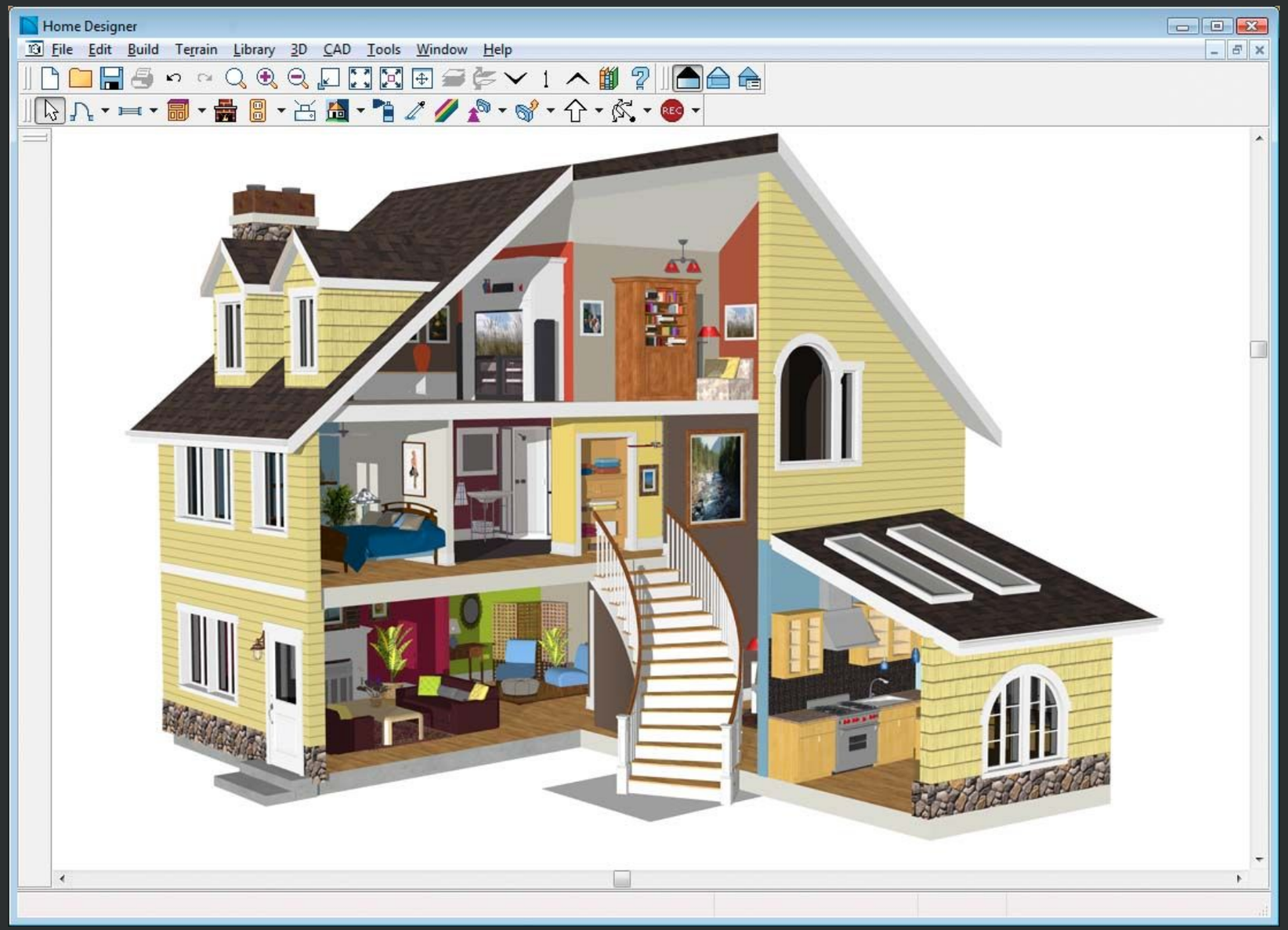
Create a class method called speed_control that asks the user the current speed and shows a message if the speed is above 100.

(resist the temptation to see the solution in the next slide)

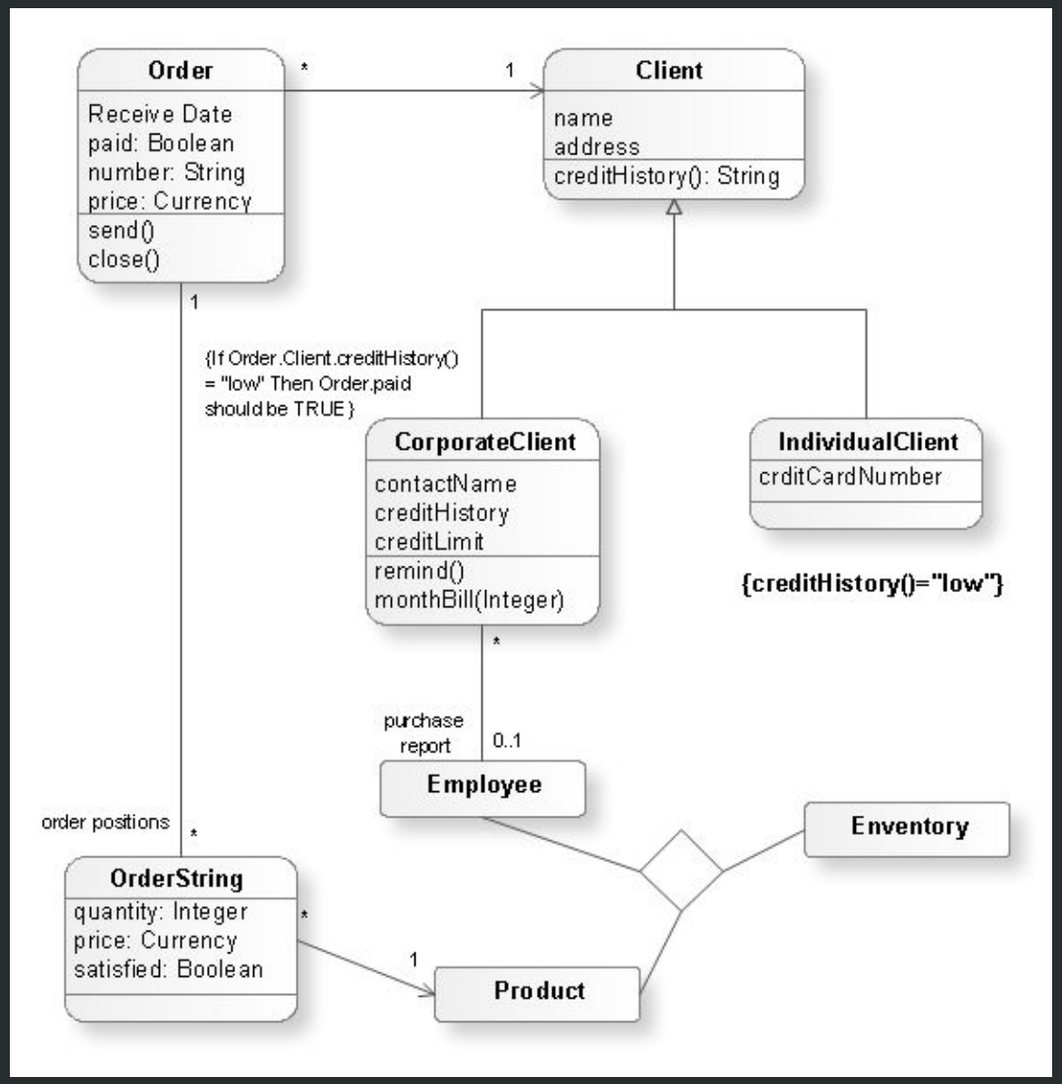
Solution

```
def self.speed_control
   puts "You just arrived to a speed control! Enter your current speed?"
   current_speed = gets.chomp
   if current_speed > 100
     puts "Woooohhhh slow down"
   end
end
```

Design


Home Designer
File Edit Build Terrain Library 3D CAD Tools Window Help

What are blueprints in software?



Encapsulation

- The important part of an object is the messages (methods) it exposes and the outcome they produce

- The objects are black boxes. We should not care on how the code is done but what it does.

- This is the most basic form of abstracting complexity

Exercise

A car needs to hold the names of the cities it visits.

A car should have a method cities that returns the array of names.

Exercise

The names of the cities should be stored in a file instead of an array

Encapsulation

The user of a class should not give a f**k about how it is implemented.

Inheritance

- It’s basing a class on another class so they share implementation

- All classes in Ruby inherit from the Object class

- We can overwrite behaviour of a class in the child class by replacing methods

Exercise

Extend car class with a racing car class whose noise will always be “BROOOM” no matter what

Attributes

- Ruby has syntactic sugar for exposing attributes of an object
  - attr_reader
  - attr_writer
  - attr_accessor
- The syntax for retrieving attributes is no different from calling a method

Exercise

Create a class Person that has 2 attributes. Name and age. The name of the person must be a read only attribute.

Both attributes must be set on initialization

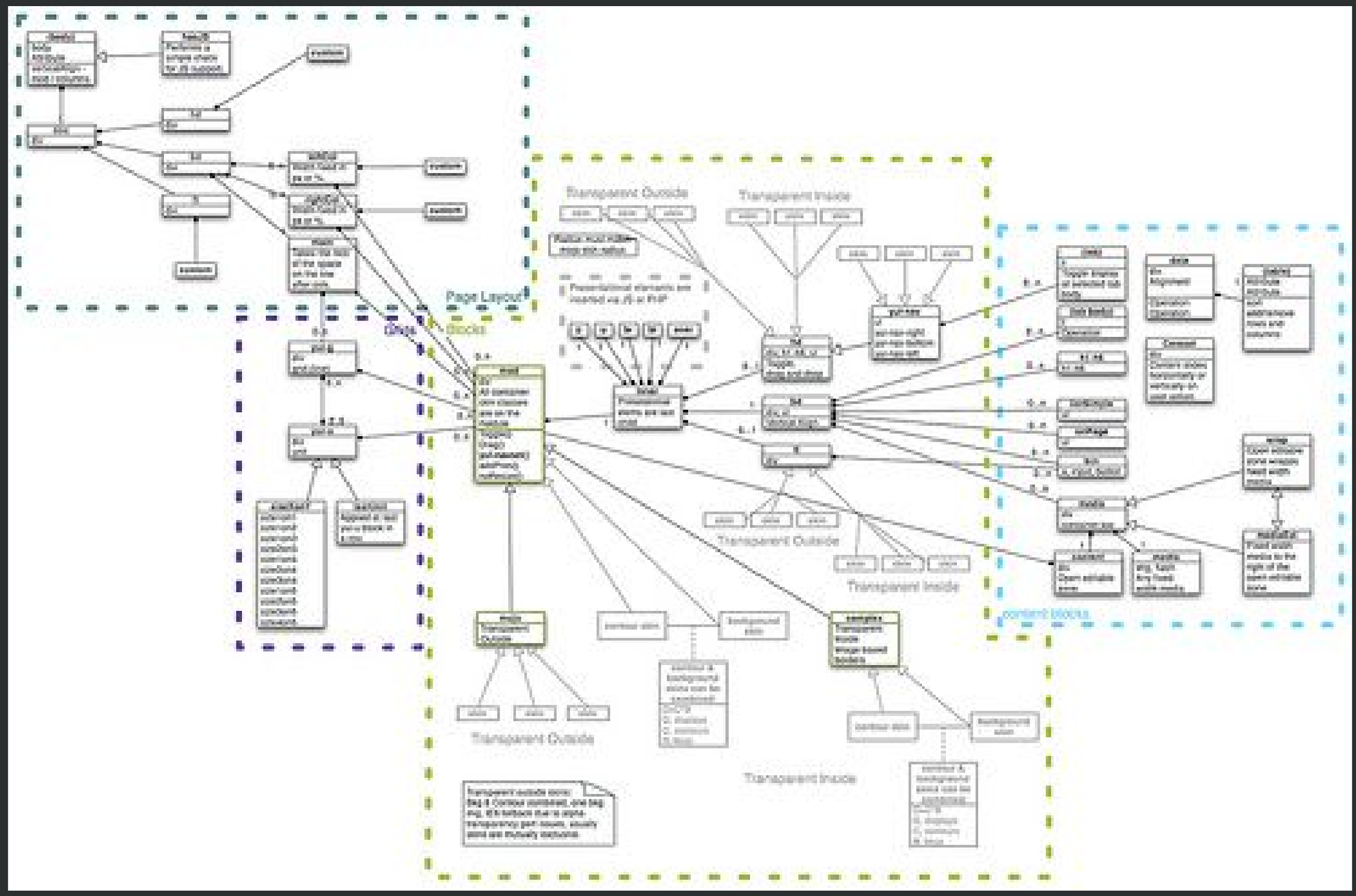
Procedural vs Object Oriented

Procedural programming, also called imperative programming, features an orderly flow of control through the program, starting at the top and continuing through the end.

An object-oriented language seeks to model something in the real world through the use of objects.

Evolutionary design

Big design up front



BDUF

==

Sedation of comfort

What we mean by this: do not plan the whole problem ahead

Solve the problem step by step