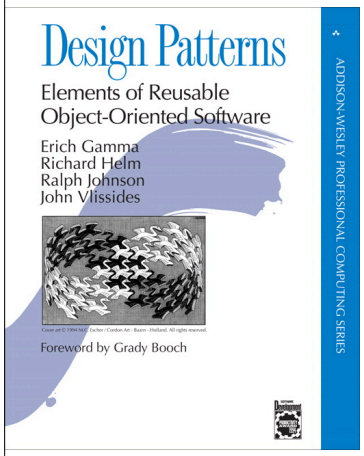
The elements of this language are entities called patterns. Each pattern describes a problem that occurs over and over again in our environment, and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice.

Christopher Alexander

Design pattern: Gang of Four

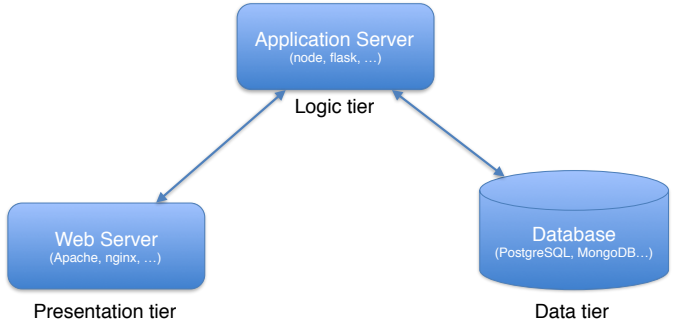


- Creational
  - Ways to create objects
- Structural
  - Ways to combine/compose objects
- Behavioral
  - Ways to communicate between objects

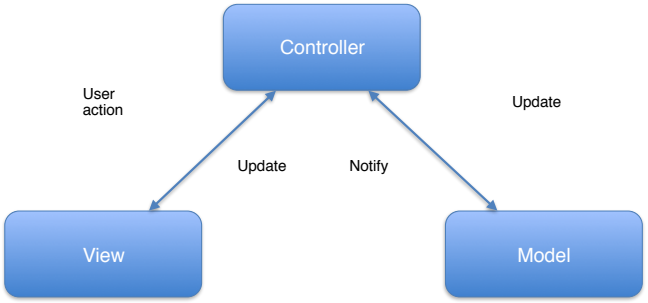
Anti-patterns: Signs you are getting it wrong…

- Viscosity
  - Easier to do a hack than do the “Right Thing”
- Immobility
  - Can’t DRY out functionality
- Needless repetition
- Needless repetition
- Needless repetition
- Needless complexity from generality

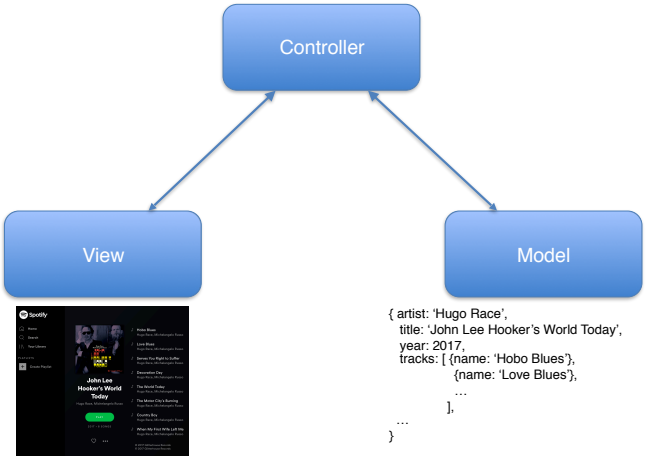
Design pattern: Three tier architecture



Design pattern: Model View Controller



Design pattern: Model View Controller



1 Which of the following is a benefit of the Model-View-Controller (MVC) design pattern?

HOW'D WE DO? **0/0** students answered

A MVC supports multi-user access and updating of the model data, with different views for each user

B Ensures there's a one-to-one mapping from each model to a single view

C Provides a window into the model and controller for debugging purposes

D Like most design patterns, it results in more concise code

▼ SHOW EXPLANATION

Frameworks/Libraries

ANGULAR

React

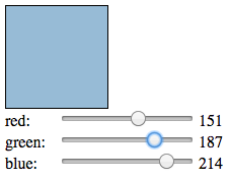
BACKBONE.JS

Vue.js

RAILS

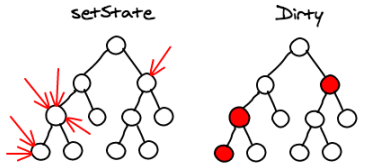
ember

Frameworks



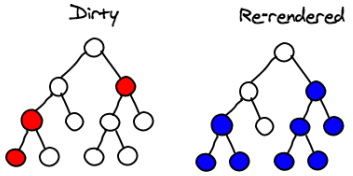
- Event based (e.g., Backbone)
  - Changing the data triggers an event
  - Views register event handlers
- Two-way binding (e.g. Angular)
  - Assigning to a value propagates to dependent components and vice versa
- Efficient re-rendering (e.g. React)
  - Re-render all subcomponents when data changes

Frameworks: React



Batched updates

Sub-tree re-rendering



https://calendar.perfplanet.com/2013/diff/