

Model-View-Controller

Dylan Lane McDonald

CNM STEMulus Center
Web Development with PHP

November 11, 2015

Outline

1 Design Patterns

2 Model-View-Controller

Design Patterns

Definition

A **design pattern** is a reusable and accepted solution to a particular software engineering problem. Design patterns are not directly translatable into code. Instead, they provide a template that can be followed while implementing a particular solution.

Design Patterns

Definition

A **design pattern** is a reusable and accepted solution to a particular software engineering problem. Design patterns are not directly translatable into code. Instead, they provide a template that can be followed while implementing a particular solution.

That is, a design pattern can be thought more as a roadmap to a particular solution. Design patterns are a core principle of modern object oriented design and widely used and recognized in both academia and industry.

There are several books on design patterns that form the foundation of today's web frameworks. [1, 2, 3]

Commonly Used Design Patterns

Many design patterns will be encountered in this class:

- **Data Access Object:** Core design pattern for all MySQL-enabled classes
- **Iterator:** Provide a way to access elements one at a time
- **Observer:** An object, maintains a list of its observers, and notifies them automatically of any state changes, usually by calling one of their methods
- **Lazy Initialization:** Loading an object from a database on demand
- **Model-View-Controller:** Divides software into three interconnected parts, so as to separate how data is manipulated and presented to the user

The most important of which is **Model-View-Controller**, which is covered in more detail in the next slides.

Model-View-Controller

Definition

Model-View-Controller is a design pattern for user interfaces. It clearly delineates the front and back end of a software solution by putting the front end in the view and the back end in the model & controller.

Model-View-Controller

Definition

Model-View-Controller is a design pattern for user interfaces. It clearly delineates the front and back end of a software solution by putting the front end in the view and the back end in the model & controller.

Model-View-Controller's main advantage is to separate the frontend (View) component from the backend (Model) component and have the two meet in the middle (Controller) component. Following this design pattern will give rise to cleaner and better organized code.

Model-View-Controller

The three components of Model-View-Controller:

- 1 **Model:** the data being represented. This is typically implemented as an object representing a database row.
- 2 **View:** the screen the user sees. This usually is HTML output of the model.
- 3 **Controller:** the mechanism that allows for the manipulation of the model. This is usually input form that allows the user to modify or add to the model.

Model-View-Controller is a commonly deployed design pattern in web programming. In fact, it is the entire basis for many web frameworks such as:

- Java Enterprise Edition
- Laravel
- Ruby on Rails

... and many more

Model-View-Controller

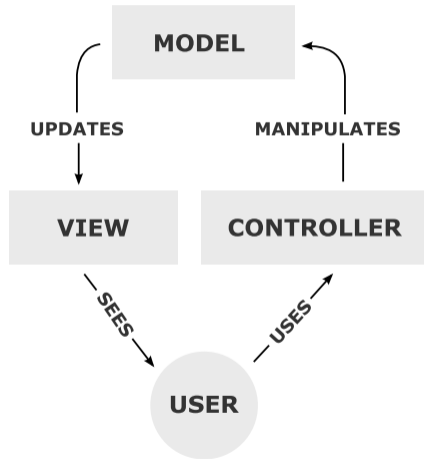





Figure 1: Model-View-Controller

Further Reading on Design Patterns

-  Steve McConnell.
Code Complete.
Microsoft Press, Redmond, WA, 1993.
-  Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides.
Design Patterns: Elements of Reusable Object-Oriented Software.
Addison-Wesley Professional, Boston, MA, 1995.
-  Deepak Alur, Dan Malks, and John Crupi.
Core J2EE Patterns: Best Practices and Design Strategies.
Prentice Hall, Upper Saddle River, NJ, 2003.