# MVC Pattern

Simanta Mitra

- EXAMPLE SWING CODE TO SHOW GRAPH

what we want to do

- add two new views easily that change when data in table changes.

- add a new way to make table changes (like say keyboard entry).

- maybe even allow a view to get data from different source easily (say a text file or say a database).

What this pattern is about

- Observer pattern had the Subject and the Observer objects.

- MVC pattern has a Model, one or more Views, and one or more Controllers.
  - Model is in charge of storing/managing data
  - View is in charge of displaying data
  - Controller is in charge of handling user events and controlling Model and View operations.
  - **Many UI applications are organized this way.**
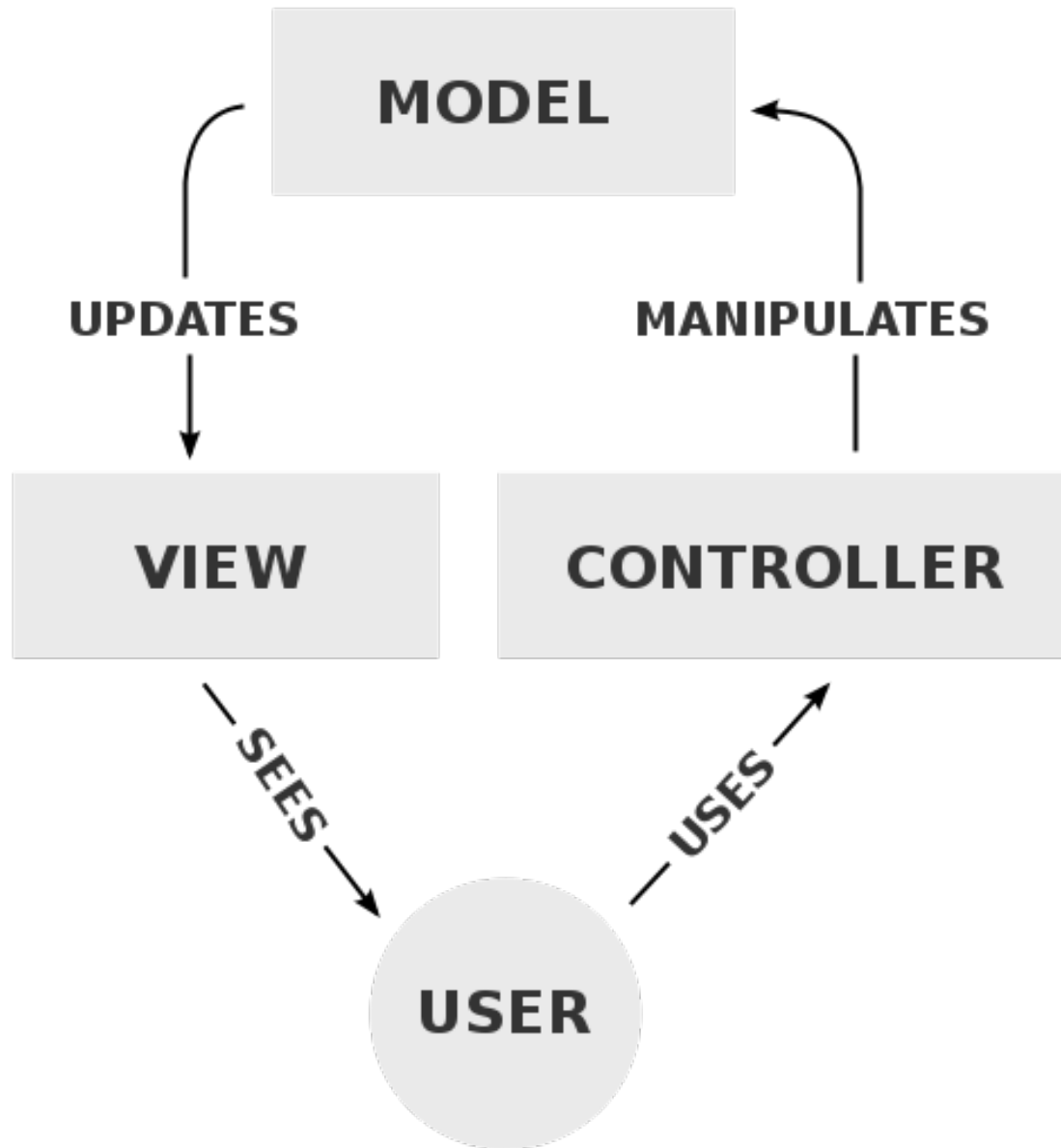
Model Class responsibilities

- Store application data

- Model does not know about view or controller. There is no reference to View or Controller in Model code

- provide interface for **making changes (set)** to the data - this is used by the **controller**

- notifies **views** of changes made to data
  - by firing event

- provides interface for **providing access (get)** to data - this is used by **views**

View Class Responsibilities

- Draws or represents the model (by getting data from the model)

- Responds to changes to model events by registering handlers for such changes

- View is loosely coupled with Model.

Controller Class Responsibilities

- Registers for (and acts on) user events

- updates model

- decides application behavior on user actions. For example – if it is a game – what should happen next?

- Controller knows about model and view.

Example of operation

1. User entry/command

2. Controller may decide to open a new view

3. Controller may also decide to update the model value (and do other processing)

4. Model will fire change event

5. This will cause views (handlers) to update themselves

BENEFITS

- Decouple view and logic from data

- separation into modules allowing for:
  - separate development and testing
  - views can be added/changed/removed easily
  - additional controllers can be added easily

- A model can be used with multiple views

- A view can be used for other models too – in that case view and model must be separated by an interface.