The Model-View-Controller Architecture

Marcos Didonet Del Fabro
marcos.ddf _at_ inf.ufpr.br

C3SL
Universidade Federal do Parana'

Context

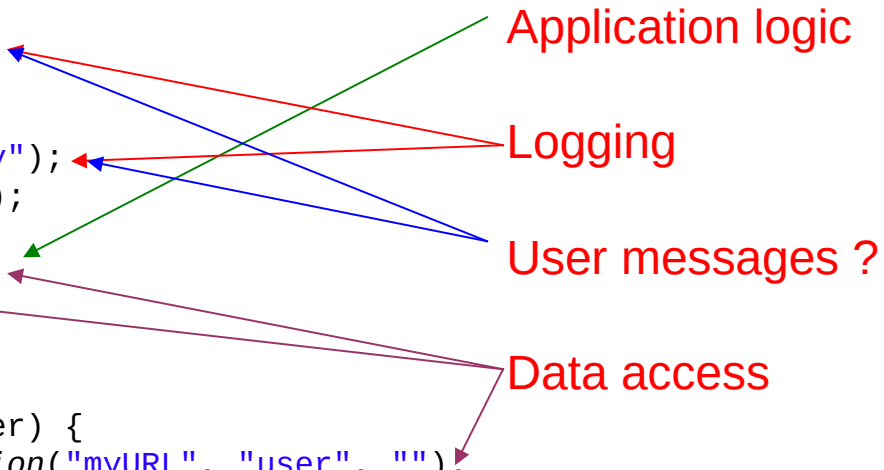
- Software development architectures
  - One layer
    - Single matchine setting
    - One large module
    - User identification, verification, data access, business logic
    - Direct access to the application using a terminal
  - Two layers
    - Multiple machine setting
    - Collaborative development
    - Separation of the data layer
    - Installation of application on clients

Context

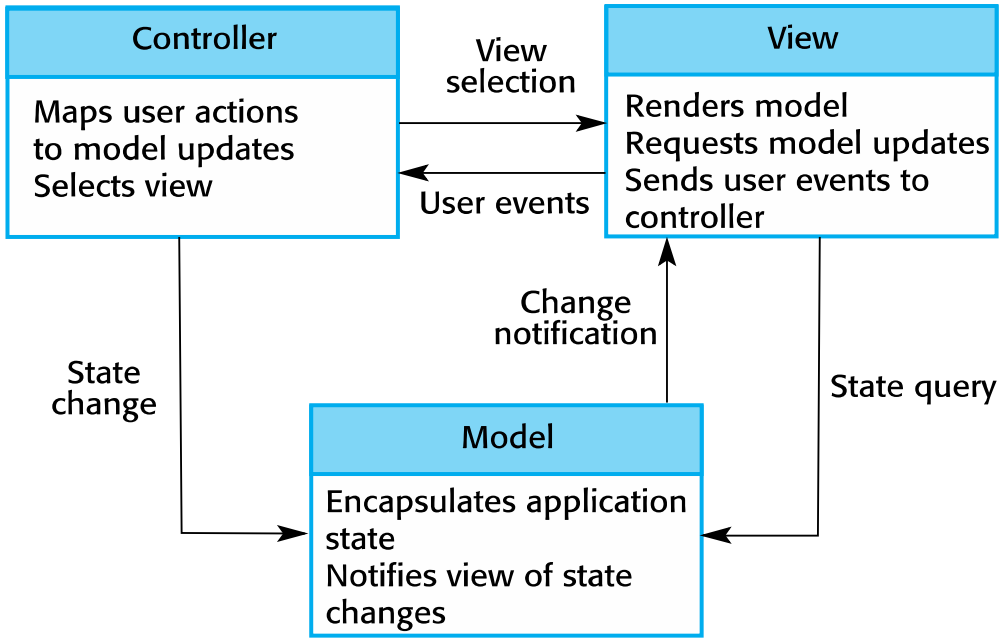
- N (three) layers
  - Business logic
  - User interface
  - Data access
- *Preferrably* Web-based deployement
  - Installation-free

Context: all-in-one

```
public class Account {
void transfer(Account fromAcc, Account toAcc, int amount, User user, Logger logger)
                                              throws Exception {
if (! checkUserPermission(user)){
    logger.info("User has no permission.");
} else {
    if (fromAcc.getBalance() < amount) {
      logger.info("Insufficient Funds, sorry");
      throw new InsufficientFundsException();
    }
    fromAcc.withdraw(amount, user, logger);
    toAcc.deposit(amount, user, logger);
  }
}
private boolean checkUserPermission(User user) {
  Connection con = DriverManager.getConnection("myURL", "user", "");
  Statement select = con.createStatement();
  ResultSet result = select.executeQuery
                    (“SELECT user from users where user = ? And password = ?");
  return false;
}
```



MVC components



MVC – Model-View-Controller

- MVC is not new
  - Smalltalk-80
- Three componets architecture
  - Model
    - Data modeling and business logic
    - Ex.: *Create new Account in the database*
  - View
    - User interface
    - Ex.: HTML page
  - Controller
    - Control the application flow
    - The *Go-Between*
    - Ex.: *When the user clicks on submit button, check the user ID and call the model method*

But it is NOT a 3-tier architecture!

MVC frameworks

- Utilization of patterns
  - Pre-defined folders
    - settings/
    - models/
    - views/
    - controllers/
    - dependencies/
    - framework/
  - CRUD applications
    - Create, Read, Update, Delete
- Generation of MVC code !!

Some web MVC frameworks

- Java/Type script
  - Angular
  - React
  - Bootstrap
  - Loopback
- Java
  - Spring MVC
  - JSF
  - Struts 2
  - Vaaddin
- Ruby
  - Ruby on Rails
  - Nitro
- .Net
  - ASP .Net MVC
  - MonoRail
  - Spring Framework.Net
- PHP
  - Agavi
  - Drupal
  - Joomla!
- Python
  - Django
  - Pyramid
  - TurboGears

Much more available !! Based on the principle of loosely coupled modules/components !