

MVC vs. MVP

Czech Technical University
Faculty of Electrical Engineering

Obsah

- 1) Co to je MVC?
- 2) Historie
- 3) Současnost
- 4) MVC
- 5) MVP
- 6) Alternativy k MVC
- 7) Závěr

1. Co to je MVC?

- Architektura **MVC** dělí aplikaci na **3 logické části** tak, aby je šlo **upravovat samostatně** a dopad změn byl na ostatní části co nejmenší. Tyto tři části jsou Model, View a Controller. **Model** reprezentuje data a business logiku aplikace, **View** zobrazuje uživatelské rozhraní a **Controller** má na starosti tok událostí v aplikaci a obecně aplikační logiku.

2. Historie

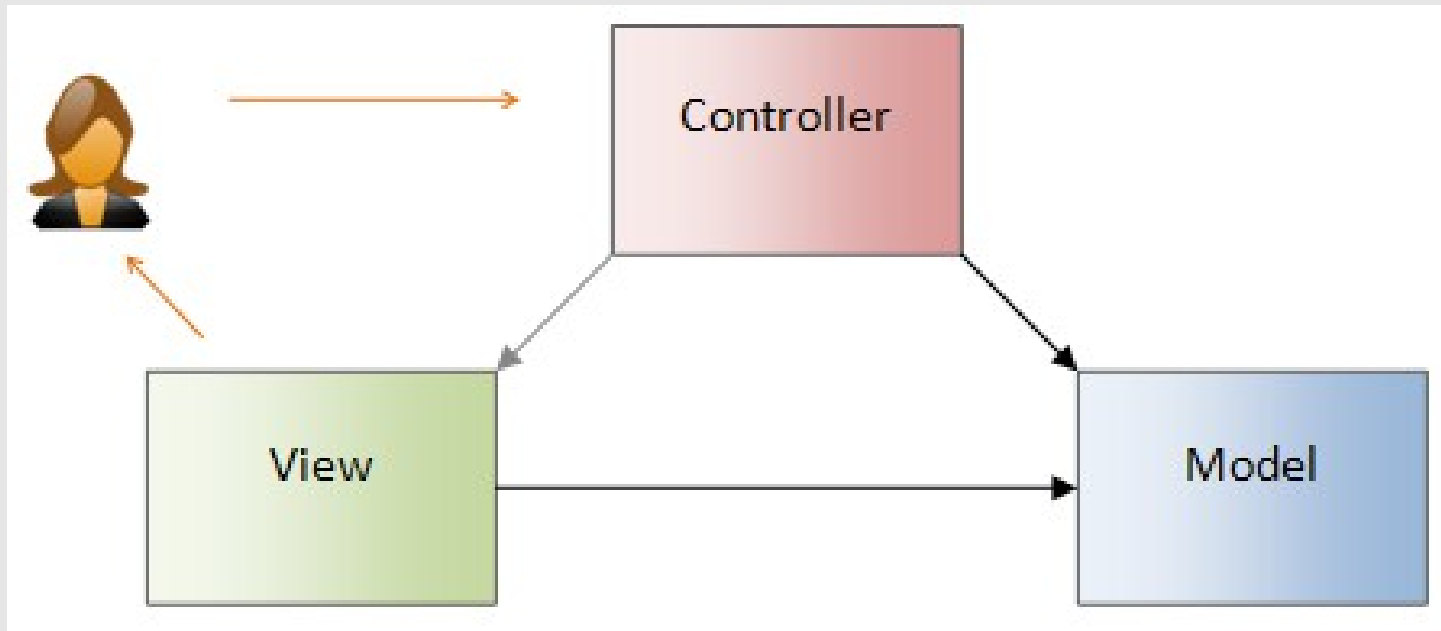
- počátky již v 70. letech
- desktop aplikace – objevuje se **MVP** (Model, View, Presenter)
- rozdělení View na **UI** komponenty – validace
- 90. léta příchod webu – **MVC**

3. Současnost

- tři druhy webových aplikací:
 - plně **AJAX**ové (**Gmail**)
 - klasická webová aplikace (**HTML**)
 - klasická webová aplikace + **AJAX** (nejčastější)
- MVC frameworky
 - JSF, Java Swing, Windows Forms, Flex, ASP.NET Web Forms, Nette, Django, Ruby on Rails, ...

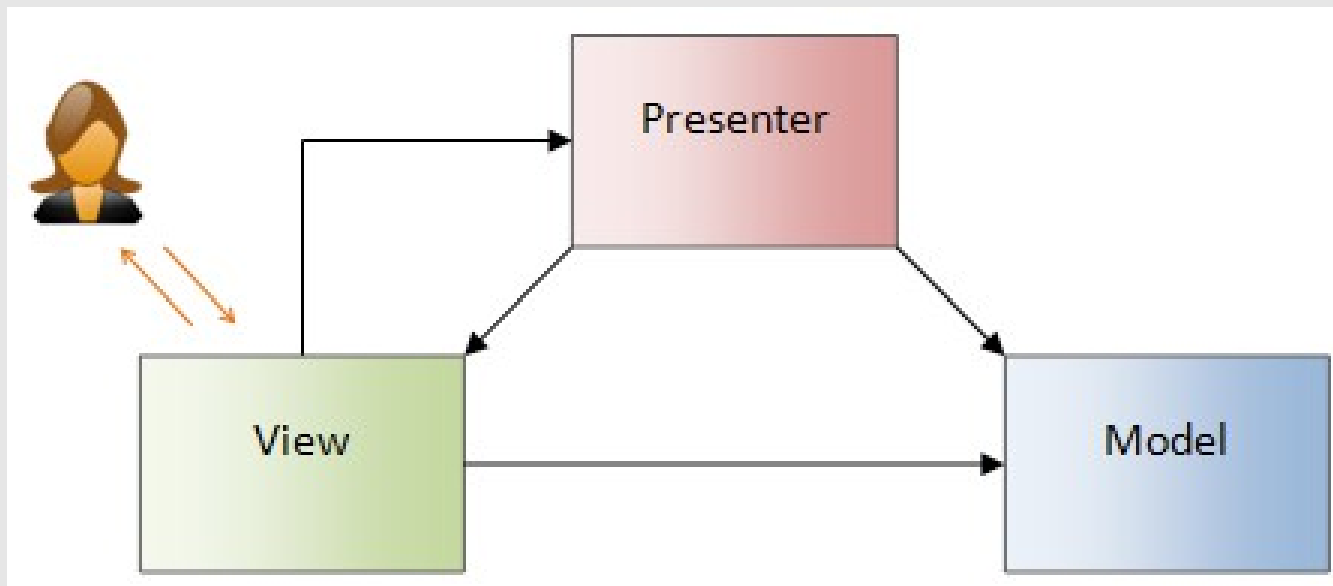
4. Model-View-Controller

→ Controller – validace, zobrazení View



5. Model-View-Presenter

- View obsahuje komponenty – validace
- Presenter – zobrazení View

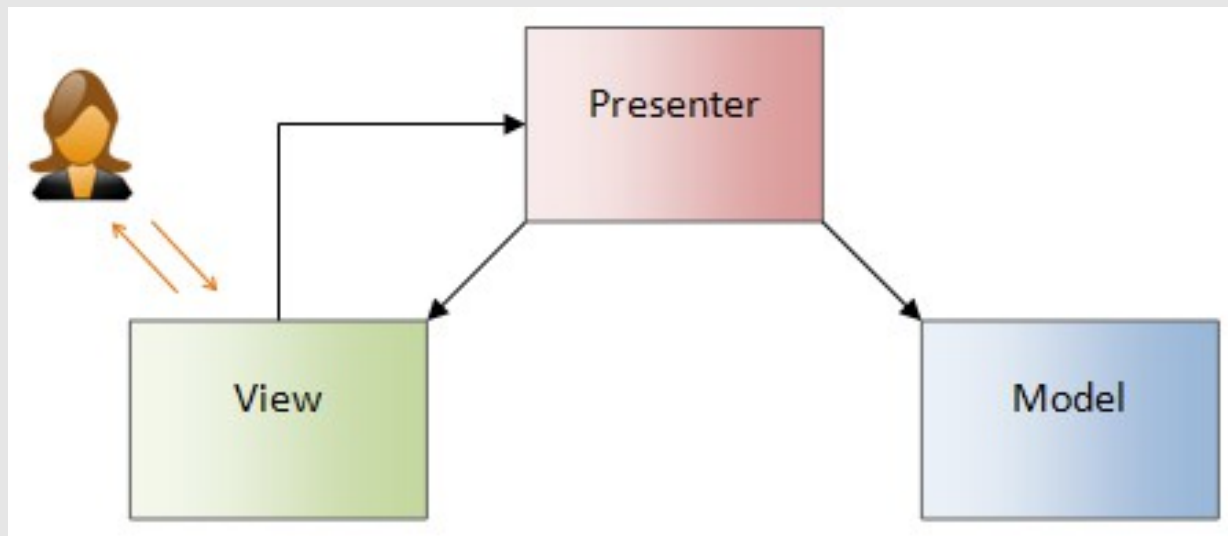


5. MVP – Supervising Controller

- v tomto případě controller = presenter!
- původní implementace MVP
- View a Presenter často implementují vzor Observer nebo data binding

5. MVP - Passive View

- View je hloupý, přímo řízen Presenterem
- lepší testovatelnost
- celá logika je v Presenteru



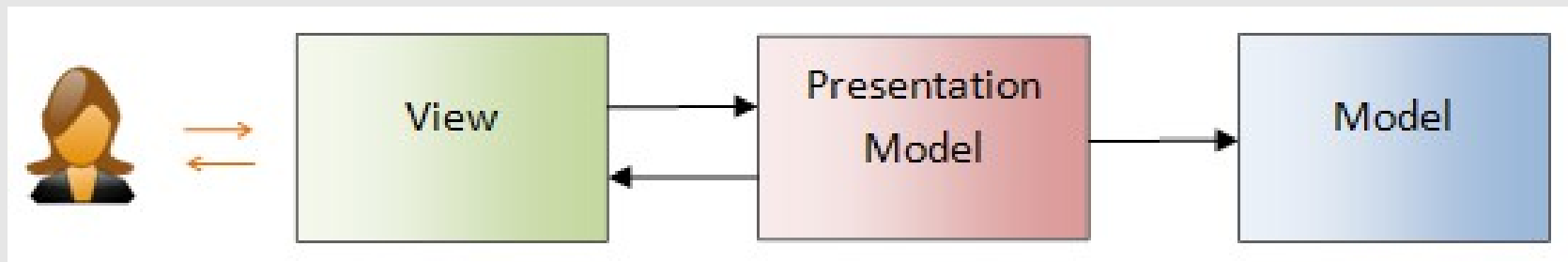
6. Alternativy k MVC

→ Autonomous View

- kombinace UI a logiky
- většinou při “drag&drop” vývoji
- hodí se výlučně pro malé projekty!

6. Alternativy k MVC

- Presentation-Model (Model-View-ViewModel)
 - nepatří přímo do MVC
 - komunikace přes Presentation-Model
 - Presentation Model implementuje **Adapter**



7. Závěr

- MVC je myšlenka
- nesnažit se slepě implementovat MVC
- konkrétní implementaci MVC vybrat podle použité technologie, druhu aplikace, ...
- MVC často implementován více vzory:
 - Observer, Command, Strategy, ...

Zdroje

→ Martin Fowler

→ <http://martinfowler.com/eaaDev/uiArchs.html>

→ <http://martinfowler.com/eaaDev/SupervisingPresenter.html>

→ Zdroják

→ <http://zdrojak.root.cz/serialy/mvc-a-dalsi-prezentacni-vzory/>

Děkuji za Vaši pozornost.