

UI Design Patterns

for the Web and Desktop

Scott Reed & Llewellyn Falco



UI Horror



Brief Survey of UI Patterns

- **MVC**
- **MVP**
- **MVVM**
- **Presentation Model**
- **Passive View**
- **Supervising Controller**
- **What do you call it if you don't use a pattern?**



The Greedy View

- **Anti-Pattern**
- **What happens by default**
- **Problems**
 - **Hard to maintain/extend**
 - **Hard to test/debug**
 - **No reuse of logic**



Greedy Demo



Separated Presentation

- **Supermodel UI Patterns**
 - **All three are separated presentation**
 - **MVC (Controller)**
 - **MVP (Presenter)**
 - **MVVM ViewModel (Presentation Model)**



Introducing

**The
View**



"Look at me"



Introducing

The Model



*"I know
everything"*



Introducing

The Controller



"I get stuff done"



MV-Poo

- **M is for Model (data)**
- **V is for View (actual display)**
- **Poo is everything else**

- **Differences in the patterns aren't that important**
- **As long as separated presentation is used**

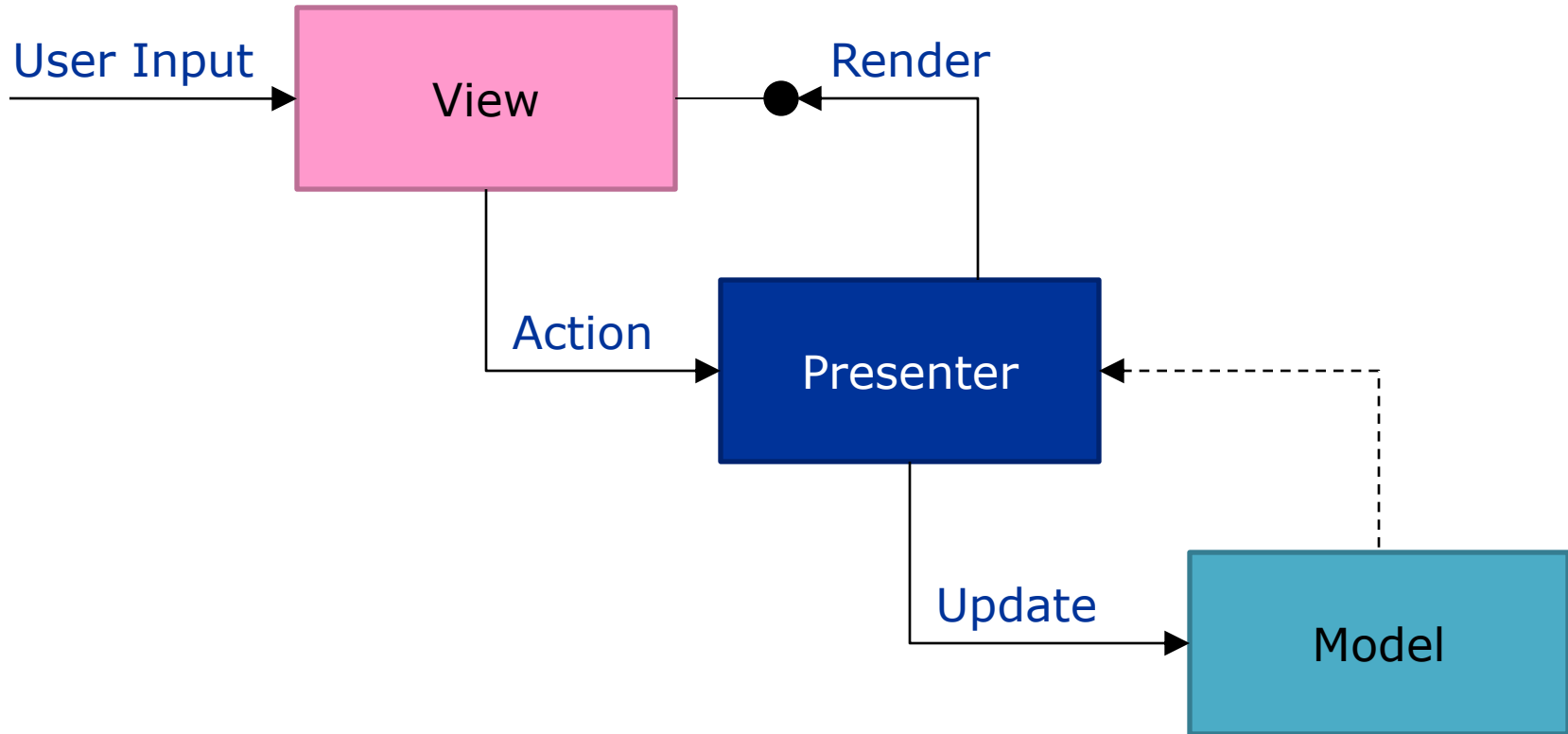


MVP (Presenter)

- **View gets the request first, passes to presenter**
- **Two variations**
 - **Passive view - presenter calls view through interface**
 - **Supervising controller - presenter fires events and view accesses model directly**



MVP (Passive View)

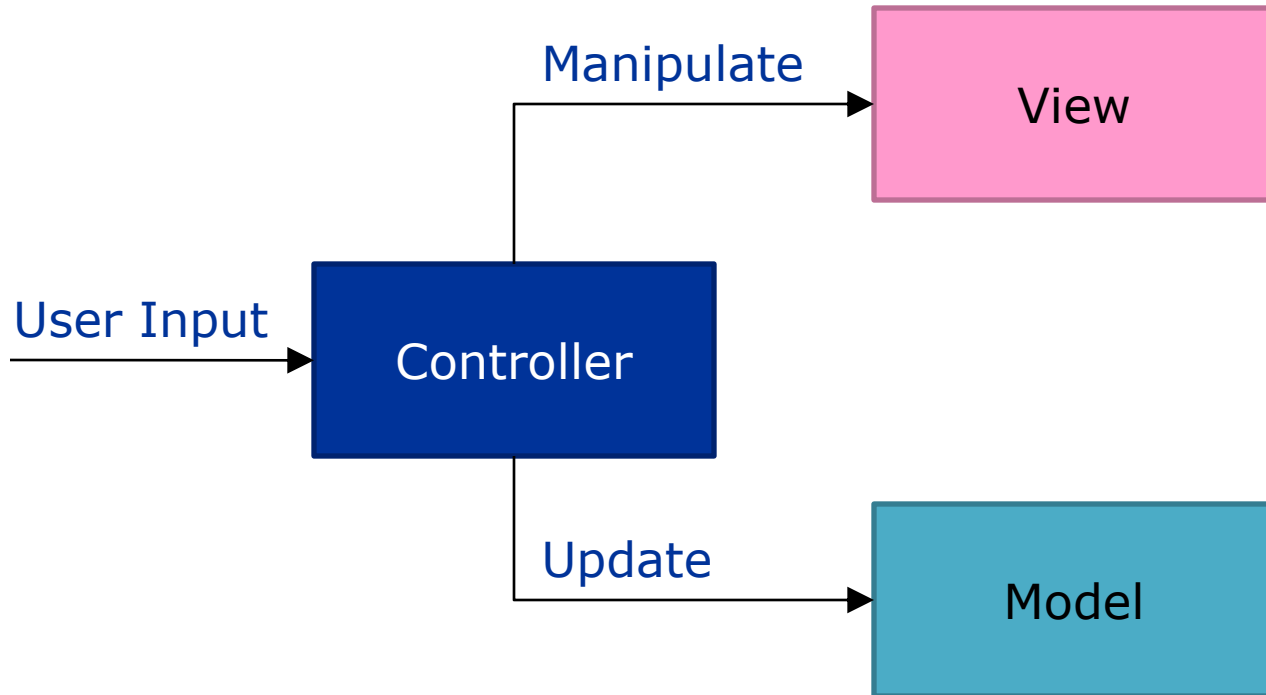


MVC (Controller)

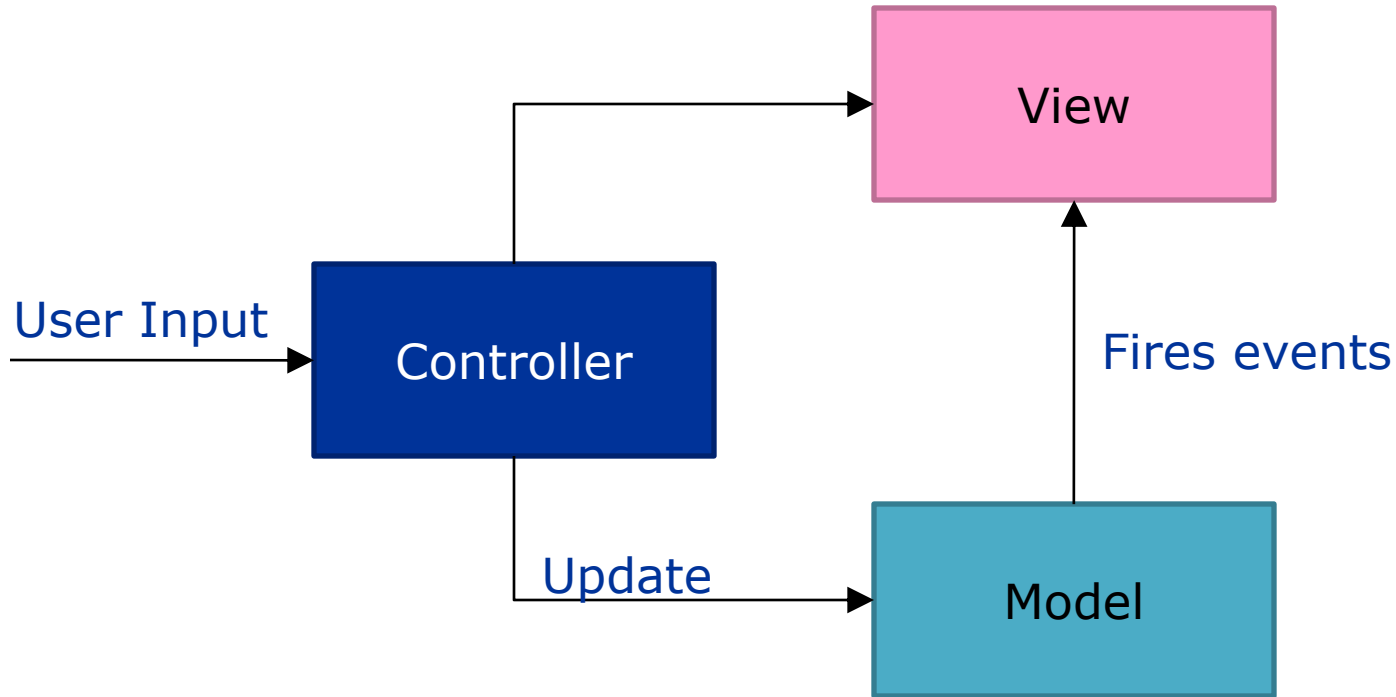
- **Controller gets the requests first**
- **Initializes/manipulates the model**
- **Chooses the appropriate view**
- **Two variations**
 - **Passive model (Web MVC)**
 - **Active model (Observers)**



MVC (passive model)



MVC (active model)

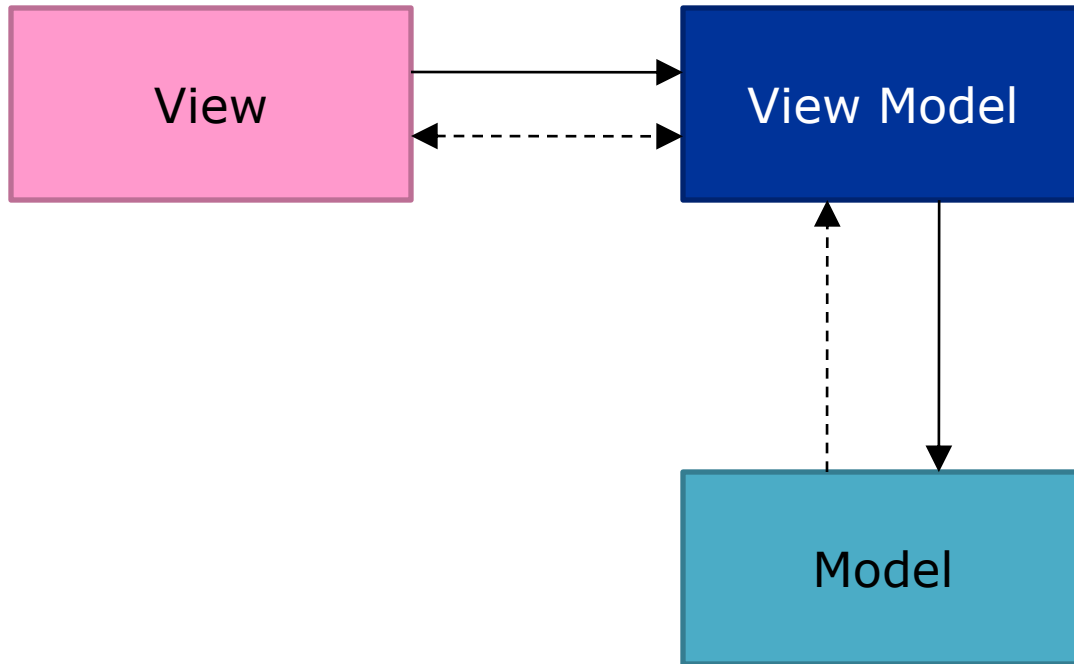


MVVM

- **View Model or Presentation Model**
 - **Tailored for designer/developer interaction**
 - **Requires powerful data binding mechanisms**
 - **Often used when model is not changeable**



MVVM (View Model)



Other Patterns

- **Page Controller**
 - Type of controller that picks the model and the view
- **Front Controller**
 - How the ASP.NET pipeline works - all requests through a single “handler” which can be decorated with modules that calls the handler for the particular page
- **Template View**
 - Type of view associated with aspx files



Other Patterns (part 2)

- **Transform View**
 - If the model is XML, and there is an associated stylesheet to change it to rendered HTML that is a transforming view
- **Two step view**
 - Similar to Master pages, where the specifics are filled in during a second step
- **Application Controller**
 - If the page interaction logic is duplicated across several controllers, then move the logic into an application controller that each controller calls.



From Greedy to Goodness

- **First move all the code in events to separate methods**
- **Try consolidating the logic into one method if possible**
- **Then move those methods into a separate file**
- **Then eliminate the UI namespaces from the other file**



Separation and MVVM

- **In XAML knowing when to stop is easier**
- **There is no view code at all!**
- **All code is found in the view model**



Thank You

Scott Reed

scott@brainhzsoftware.com

Llewellyn Falco

info@spunlabs.com



References

- **Authoritative source:**

<http://martinfowler.com/eaDev/uiArchs.html>

- **Other interesting reading:**

- <http://haacked.com/archive/2008/06/16/everything-you-wanted-to-know-about-mvc-and-mvp-but.aspx>

- <http://blogs.msdn.com/erwinvandervalk/archive/2009/08/14/the-difference-between-model-view-viewmodel-and-other-separated-presentation-patterns.aspx>

Note here the author is confused, MVVM *is* the same as a Presentation Model

