Web Presentation Patterns (view)

SWEN-343

From Fowler, *Patterns of Enterprise Application Architecture*

Software Engineering

Rochester Institute of Technology

View Patterns

Concerns

How to construct a view (a web page) in response to application processing?

How to generate the detailed HTML stream to send in response to the HTTP request?

Patterns

Transform View and Template View patterns

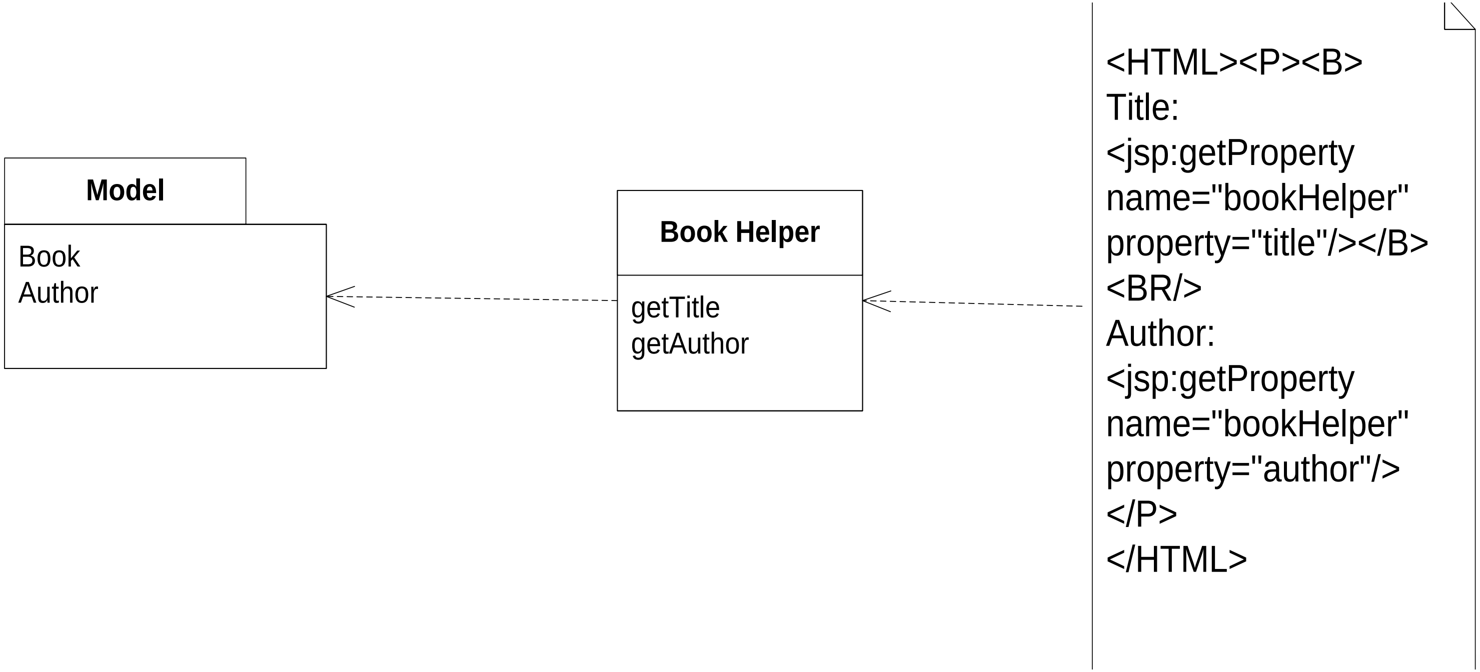
Do the view HTML creation in one-step or two

Two-Step View pattern

Template View and Transform View

- Template View
  - Write the presentation in the structure of the page
  - Embed markers to indicate where dynamically created content (such as domain layer results) needs to go
- Transform View
  - Use XSLT stylesheets to transform XML data returned from the model into XHTML streams

Template View



Model

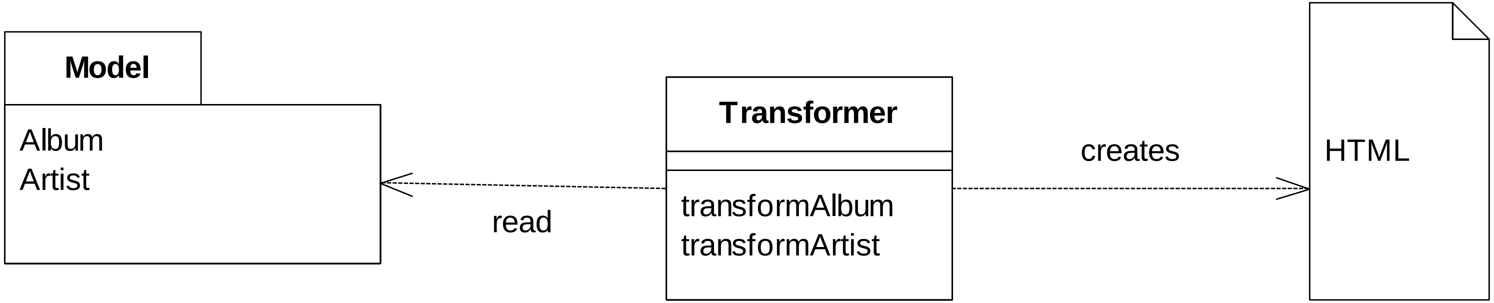
Book
Author

Book Helper

getTitle
getAuthor

```
<HTML><P><B>
Title:
<jsp:getProperty
name="bookHelper"
property="title"/></B>
<BR/>
Author:
<jsp:getProperty
name="bookHelper"
property="author"/>
</P>
</HTML>
```

Transform View



Caution

Template View is supported by “Server pages”-style technologies such as JSP, ASP, PHP

- Can embed full programming language horsepower in the page

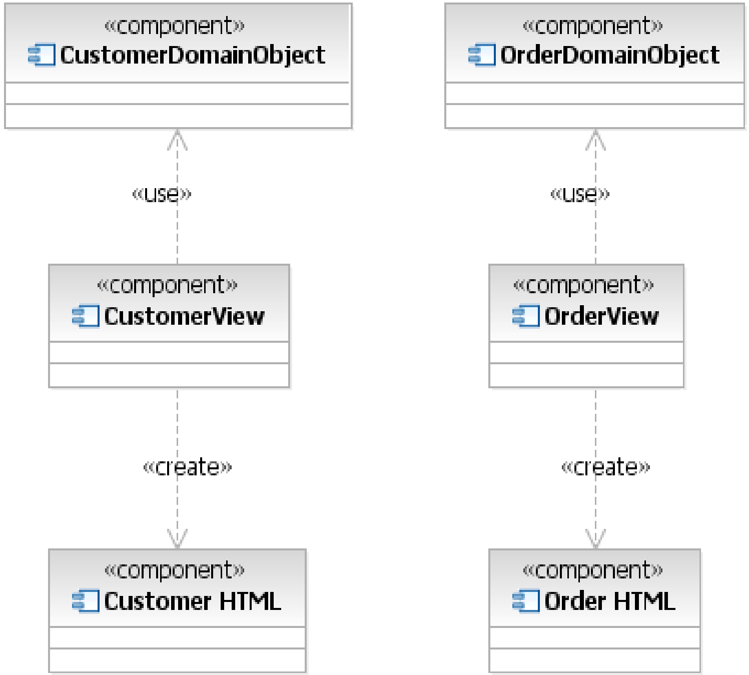
Authoring tools make it is too easy to embed application logic in the page

- Results in an unmaintainable mess
- Be very disciplined to keep application logic out of the page—use a helper object, instead

Single Stage View

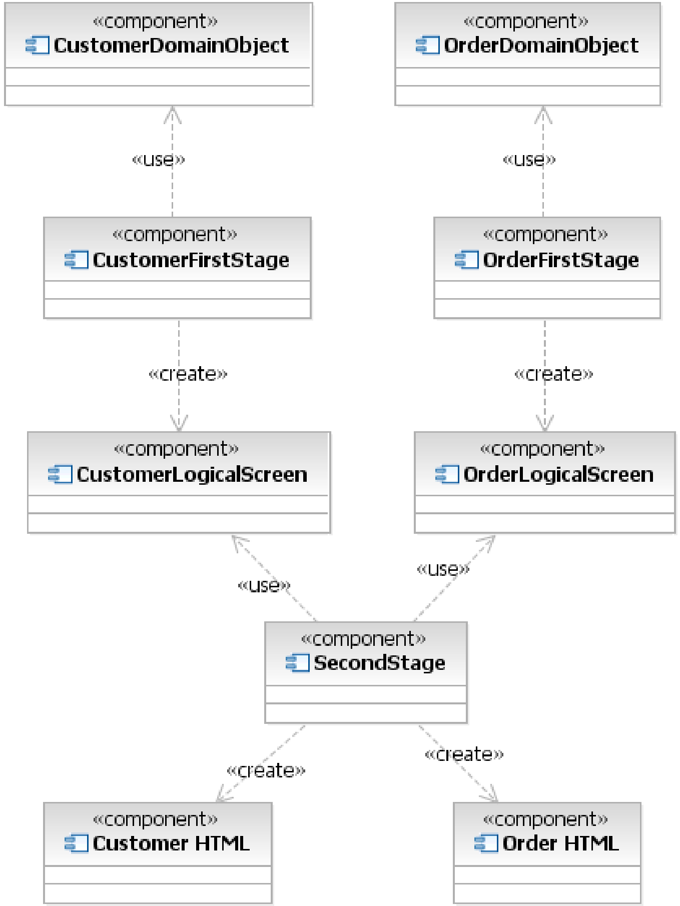
Single stage: One view component (mostly) for each application screen

- View component takes domain data and renders it in HTML

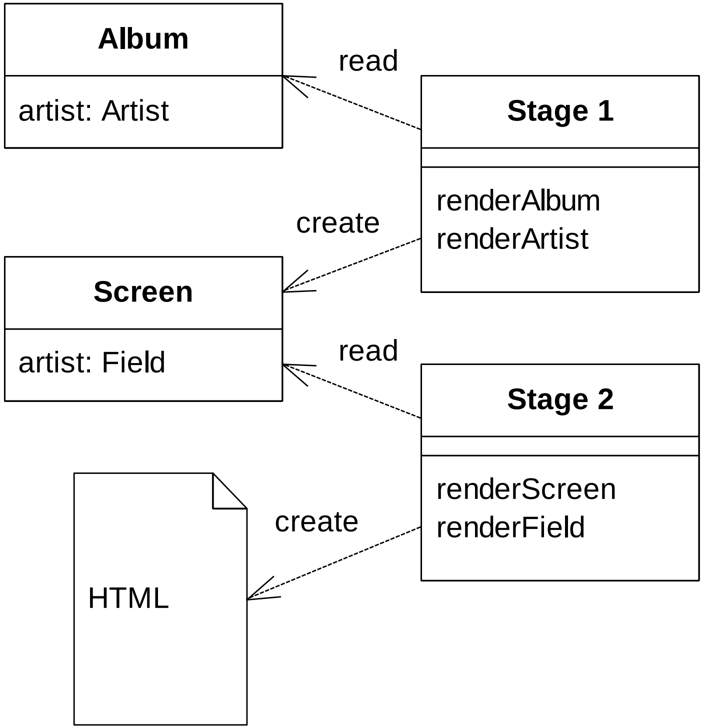


Two-Stage View

- First stage produces a logical screen from domain data
  - One first-stage view for each screen
- Second stage renders the logical screen in HTML
  - One second-stage view for whole application
  - Items common on all pages (branding, logos, navigation bars, etc.) are in one place
  - Requires consistent layout across screens
  - Requires consistent web browser (won’t work well on both a cell phone and desktop)



Two Stage View



Choosing a View

- Template View
  - Follows natural editing notions
  - Supports non-programmer editors of HTML
  - Needs discipline to avoid scriptlets
- Transform View
  - Can test without web server
  - Works well with XML (XSLT)
- Two Step View
  - Easy to make global appearance changes
  - Easy to support multiple appearances
  - Complex

Conclusions

- It is imperative to separate user interface (view, control) from application (model)
    - Especially when the application behavior is executed in the domain layer (as it should be!)
- Input Controller patterns and Application Controller pattern help manage complicated user session flow control and selecting and delegating to appropriate model behavior
    - Domain layer may have additional controllers to manage the flow of application logic, but these are independent of any presentation-oriented controllers
- View patterns help organize the construction of HTML response pages
- Read the pattern documentation in Fowler for detailed examples with code in Java and C#