sillyview Architecture

1. How Does it Work?

sillyview provides interfaces for models and views, some useful implementations, and some support for controllers.

By far, the most useful model is the VelocityModel which backs a view using HTML template files processed by the the Velocity Template Engine.

architecture diagram

1.1. Passing information between a View and an Application

Consider the diagram below, illustrating how information is passed between a Swing application and a view. The application creates a VelocityModel, and gives it a template to work with. It then populates the model with data. Then a view is created (likely a JPanelView or one of its subclasses). The view is populated by merging the model's data with the template. The view is connected to the application by registering a hyperlinkListener with it. Then the user clicks, types, and generally responds to the view. He fires a hyperlink event, probably by clicking a SUBMIT button.

architecture diagram

The hyperlink event sends information back to the application. The application parses this information, reacts to it, and, most likely, updates the model's data. The new data is (likely) merged with the template, giving the user a new display in the view reflecting the application's new state.

1.2. So, What's The Point?

The point is this: you separated (most of) the UI from the application, and specified it in HTML rather than Swing. This means that you can change your UI by editing a HTML template file rather than Swing code. Furthermore, you can provide multiple UI's (think skins) by providing a handful of templates from which the user can choose at runtime.

Finally, say you want to use the same program logic and user interface in a servlet. Well, the program logic can probably remain virtually unchanged. Use a StringBufferView instead of a

Copyright © 2004 T.J. Willis All rights reserved.

JPanelView, and return it as your ServletResponse. Instead of using a hyperlinkListener, parse the URL passed into your servlet, et voila, you have a servlet.

Copyright © 2004 T.J. Willis All rights reserved.