

VAJ GUI Builder (VCE)

- The VCE thus far
 - static GUIs
 - What's missing?
 - data!
 - Design goal:
 - Separate GUI , pgm logic!
 - *VCE variables*
 - Now, *Patterns* !
 - MVC, Factory....
 - Patterns in the VCE -- MVC
- Reading Budd Ch 14, 15
 - Asst LaunchAuction
 - due Thu Nov 15 10am
 - demo in lab
 - Asst Case Study
 - Design Review – Mon or Wed
 - Implementation Plan....
 - Q&A -- Tuesday 2-5
 - Due Tuesday, Week 10 2pm

Static GUIs in VCE

buttons and beans....

- What makes a class a bean? -- Public, persistent
- Bean features -- Properties, Event Sets, Methods
- To create a visual class
 1. Start the AddClass Smart Guide (CreateClass button)
 2. Enter class name, java.awt.Frame or ...swing.JFrame as superclass
 3. Add import statements for packages, java.awt.*, etc.
- To visually compose the class
 1. Double-click frame's title bar; on property sheet, type frame title
 2. Select beans from the palette and drop them onto frame, setting label properties, bean names. Save often.
 3. Switch to hierarchy tab, to edit some methods -- constructor for classes, e.g., Add, Remove.
- Other SmartGuides : **CreateApplet, CreateApplication**

*Connection***

represents a listener for events from an event source

- Event-to-method (ex: remove-all)
 1. Choose a source bean and event; select a target bean & method
 2. Connect->actionPerformed ... remove buttonclick
 3. Target - the List component ... Connectable Features ... removeAll()
 4. Is the line dashed? ... Need parameters! ...
 - constant value -- set parameters in the connection property
 - event data -- check pass event data ...
 - the event object passed as 1st parameter
 - create connection from another method, property --
Connect->parameterName
 - Parameter-from-Method (of a bean)
 - Parameter-from-Property
 - Parameter-from-Code

Static GUIs in VCE

Layout Management....

First, a Word to the Wise

GUI Design vs. GUI Implementation

- The VCE is a _____ tool.

First, a Word to the Wise

GUI Design vs. GUI Implementation

- The VCE is a GUI Implementation tool.
 - Don't *hack together* a GUI....
- 1. Draw the GUI ...graph paper helps.
- 2. Annotate the drawing with sizing behavior
- 3. Determine the layout design
 - Group components into containers
 - Write a tree-form description of the layout design

VCE *variables*

- Use a variable instead of an instance of a class. You can draw connections to it, but you can **change instance** -- use variables as sockets to which data values are plugged
- ExR: a bean describing the relationship between people....
person, address, mom, dad.... pp. 419-425
 - *variables* provide access to a property of a property
 - Tear-off properties automate this! pp 425-427
- To access a value (primitive or class) as a different type
 - create a variable of the desired type & draw a property to property connection from the old to the new
- Variables as plugs for external classes
 - ExR: *morph* instance of person to variable of type Person
pp430-435

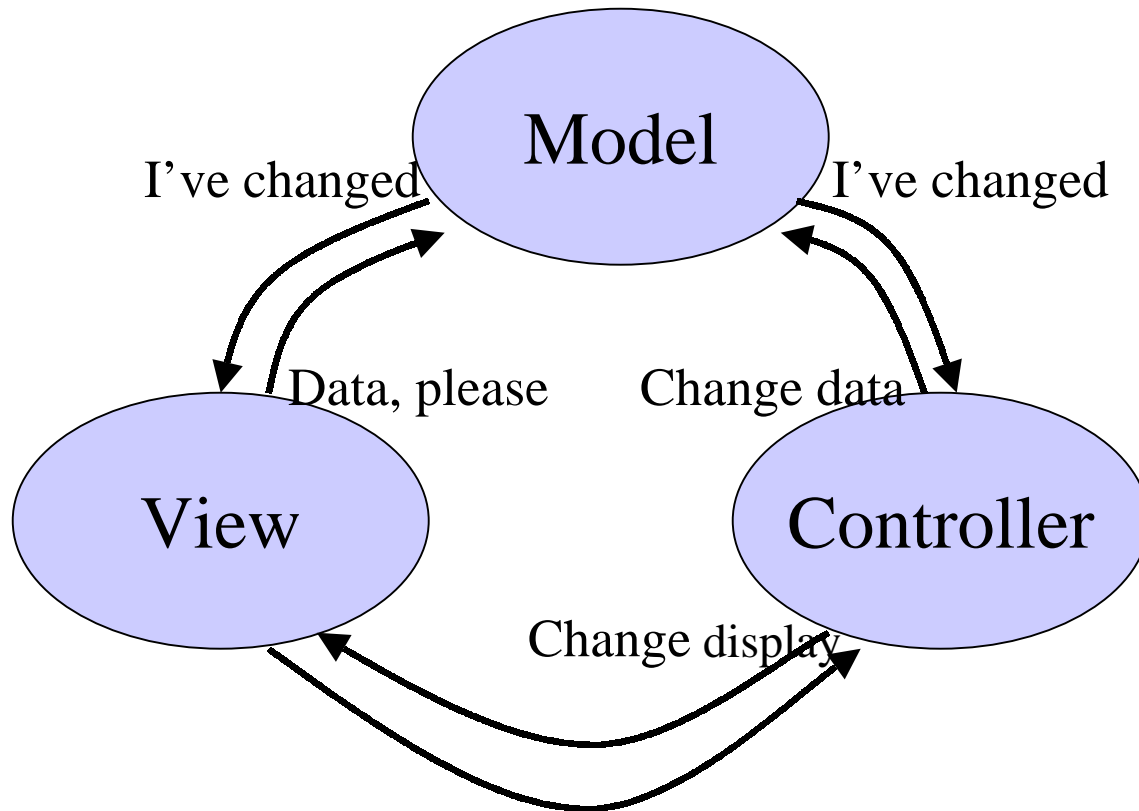
Patterns

Elements of Reusable OO Software, Gamma et al

- Proven methods for solving common problems
 - design reuse
 - common vocabulary & education
- Strategies for VCE coding
 - Less is more -- connections turned spaghetti....
 - don't create the entire application!
 - Code is OK -- **If** it is visible (use event-to-code connection)
 - Break down into sub-GUIs for maintenance
 - Separation is key -- the MVC
 - Patterns for GUI design
 - factorization, trigger, funnel, enabler, visual assertion, conditional fork....

MVC

Smalltalk, 1970s



MVC

Delegates -- combining View and Controller

