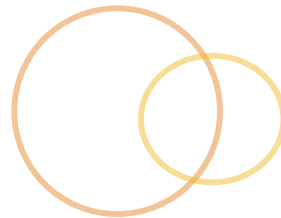
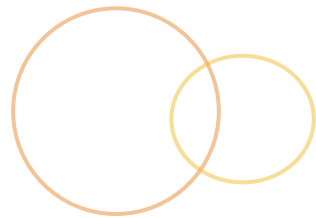
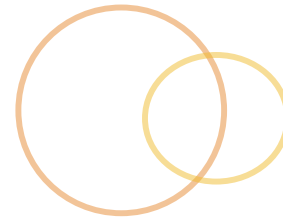
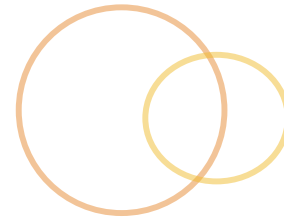


JSF Navigation

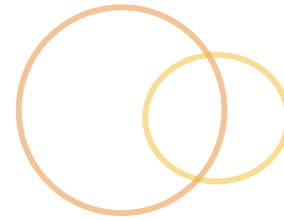


JSF Navigation



- Navigation between views controlled by FacesServlet
 - Typically initiated as a result of an action
 - Standard HTML links do NOT go through FacesServlet
- Navigation depends on navigation rules defined by:
 - A navigation case
 - The originating source
 - The originating action
 - The resulting outcome (of the action)
 - The path to the destination resource

Types of Navigation



- ◎ JSF supports three types of navigation:
 - ◎ Path-based
 - ◎ Action-based
 - ◎ Path and action based
- ◎ Can have multiple navigation cases per rule
 - ◎ Used to handle branching scenarios
 - ◎ Can be add complexity to configuration
- ◎ If an application has many navigation rules, could create a faces-navigation.xml file to aid in configuration of faces-config.xml

Path-based Navigation



- ⦿ Easy, similar to basic HTML linking
- ⦿ Focus on where the request came from
 - ⦿ Like the referrer header in HTTP
 - ⦿ No need to define an action for UICommand controls
- ⦿ Navigation is defined in terms of:
 - ⦿ `<from-view-id>sourcepath</from-view-id>`
 - ⦿ `<to-view-id>targetpath</to-view-id>`
- ⦿ Supports matching similar to servlet mappings
 - ⦿ `<from-view-id>/app/*</from-view-id>`
 - ⦿ `<to-view-id>targetpath</to-view-id>`

Action-based Navigation



- ⦿ Focus on what caused the navigation request
 - ⦿ Can be used to setup a “global” navigation rule in JSF
 - ⦿ Must define an action for UICommand controls
 - ⦿ Do not define <from-view-id>
- ⦿ Navigation is defined in terms of:
 - ⦿ <from-view-outcome>outcome</from-view-outcome>
 - ⦿ <to-view-id>targetpath</to-view-id>
- ⦿ If no matching <from-view-outcome> case is found, no navigation occurs

Path and Action-Based Navigation

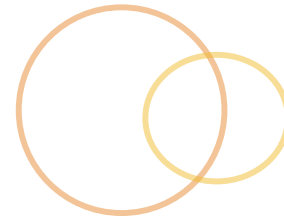


- ◎ Most specific type of navigation
 - ◎ Combines path-based rules with action-based rules
 - ◎ Provides most control, extremely explicit
 - ◎ Must define an action for UIControl controls
- ◎ Specify path-based rules
 - ◎ <from-view-id>
 - ◎ <to-view-id>
- ◎ Specify action-based rules
 - ◎ <from-outcome>
- ◎ If no matching case is found, no navigation occurs

Dynamic Navigation

Moving beyond Static Navigation Rules 

Dynamic Navigation



JSF FacesServlet can function as a:

- ◎ Controller

- ◎ Navigation rules are statically defined
- ◎ Action outcomes are hard-coded in UICommands
- ◎ Using Path-based, Action-based, or Path and Action-based

- ◎ Front Controller

- ◎ Performs core “controller” functionality
- ◎ Incorporates command pattern
- ◎ Command function bound in UICommand action AND <from-outcome>
- ◎ Command outcome dictates navigation

Command Pattern Design



- Command implementation is executed during the “Invoke Application” phase of JSF lifecycle
 - Command actions are different than ActionEvents
 - Intentionally isolated from UI components
 - Any method can function as a command action
- Two ways to implement command
 - As a method on a Java Bean
 - Since most commands rely on model data, encapsulate command in the model
 - Command will have direct access to model data
 - As a separate class
 - Allows commands to be reused
 - Command will not have direct access to model
 - Will have to retrieve model data from FacesContext

Command Pattern Implementation



- ⦿ Managed-bean requirements
 - ⦿ In managed bean, define a method that will perform command action
 - ⦿ Remember, method will be called before the render response phase and after the update values phase
 - ⦿ This means the bean is in a valid state
 - ⦿ The method should return a String value
 - ⦿ Value will be should match a <from-outcome> rule
 - ⦿ Consider using “success” and “failure”
 - ⦿ The command method does not need to follow JavaBeans rules
 - ⦿ The command should not accept arguments

Command Pattern Implementation [cont.]



- ⦿ JSF requirements

- ⦿ Bind command to UICommand control's action attribute
- ⦿ `<h:commandButton action="{mybean.command}" />`

- ⦿ Faces-config.xml requirements

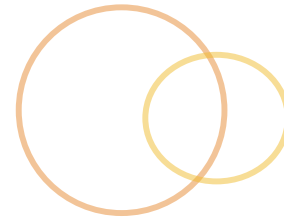
- ⦿ Bind command to `<from-action>`
- ⦿ `<from-action>{mybean.command}</from-action>`

About DevelopIntelligence



- ◎ Founded in 2003
- ◎ Provides outsourced services to learning organizations in area of software development
- ◎ Represents over 35 years of combined experience, enabling software development community through educational and performance services
- ◎ Represents over 50 years of combined software development experience
- ◎ Delivered training to over 40,000 developers worldwide

Areas of Expertise



◎ Instruction

- ◎ Java
- ◎ J2EE
- ◎ WebServices / SOA
- ◎ Web Application Development
- ◎ Database Development
- ◎ Open Source Frameworks
- ◎ Application Servers

◎ Courseware

- ◎ Java Application Development
- ◎ Java Web App Development
- ◎ Enterprise Java Development
- ◎ OOAD / UML
- ◎ IT Managerial
- ◎ Emerging Technologies and Frameworks



- ◎ For more information about our services, please contact us:
 - ◎ Kelby Zorgdrager
 - ◎ Kelby@DevelopIntelligence.com
 - ◎ 303-395-5340