THE UNIVERSITY OF AUCKLAND

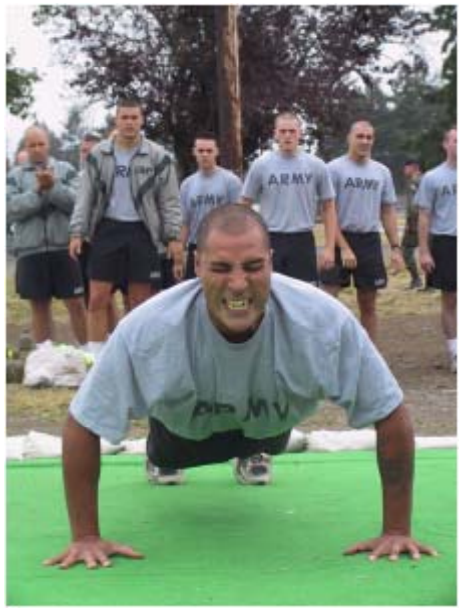
COMPSCI 230

Software Design and Construction

Revision

2013-06-07

Exam Preparation



"There is no such thing as tough.
There is trained and untrained.
Now which are you? "

What's in the Exam?

Don't Panic!

Christof's part contains:

- 10 multiple choice questions
- 1 code writing question
- 1 text short-answer question (2-3 sentences)

How to prepare for it?

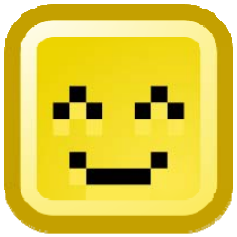
1. Watch the lecture videos: http://www.cs.auckland.ac.nz/~lutteroth/teachings/COMPSCI230/
2. Read the lecture slides
3. Answer the quiz questions at the end of each lecture
4. Try to think like a lecturer:
   - What concepts are important?
   - How could one ask about them?

THE UNIVERSITY OF AUCKLAND

Should I Use A Framework?

Advantages

- **Reuse** can save cost and time
- Higher level of **abstraction**
  - Less low-level work
  - Easier to understand
    (if the framework has a good API)
- Reduced **maintenance** cost
  (if the framework is maintained by someone else)



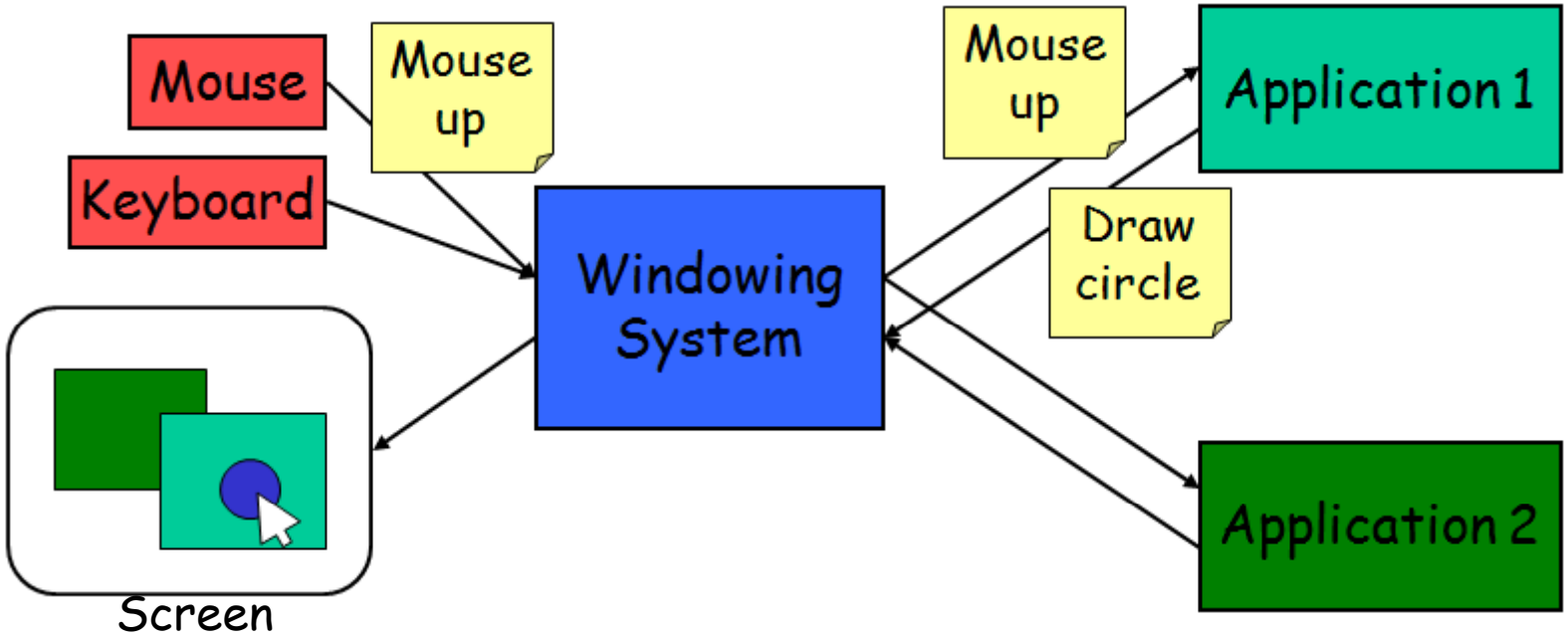
Disadvantages

- Can lead to code **bloat**
  - Framework may contain lots of unused code
  - May need to use several frameworks
- Cost of **learning** a framework (needs to be amortised by reuse)
- **Licensing** cost (for commercial frameworks)
- Risk of **vendor lock-in**

THE UNIVERSITY OF AUCKLAND

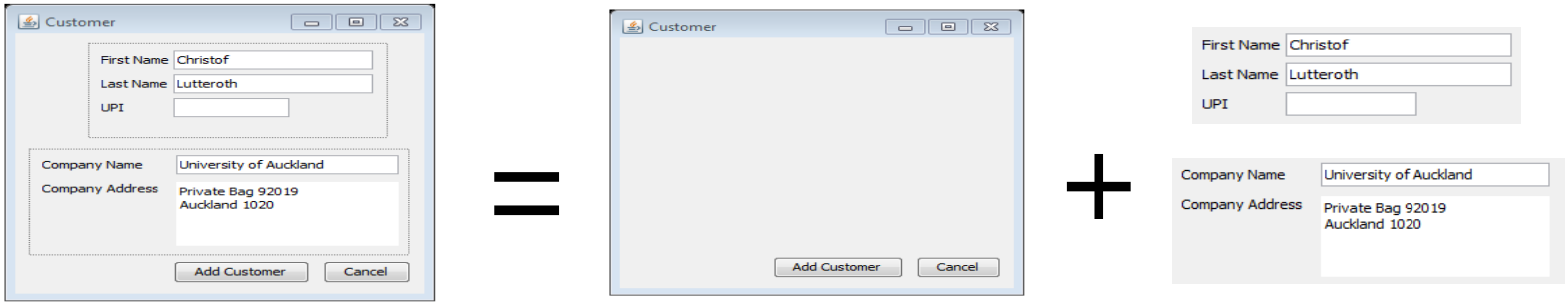
Windowing System

- Manages **input** and **output devices**:
  graphics cards, screens, mice, keyboards
- Sends **input events** from input devices to apps
- Receives and processes **drawing commands** from apps
- Often able to talk to remote applications: send input events and receive drawing commands over the network

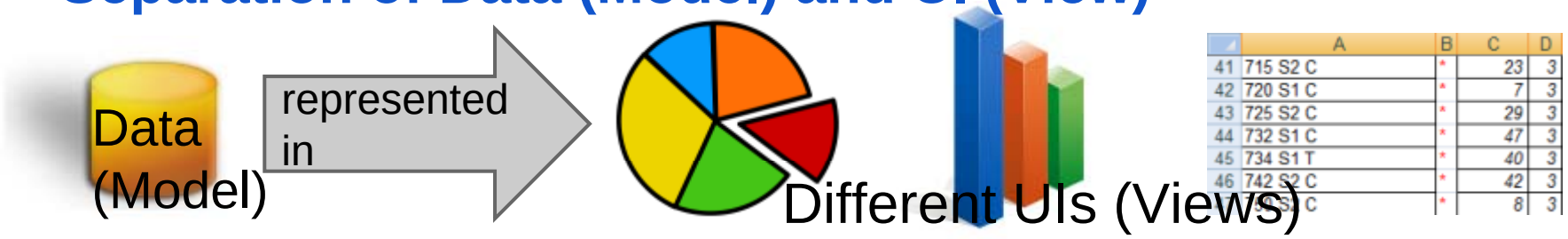


Recap: Separation of Concerns

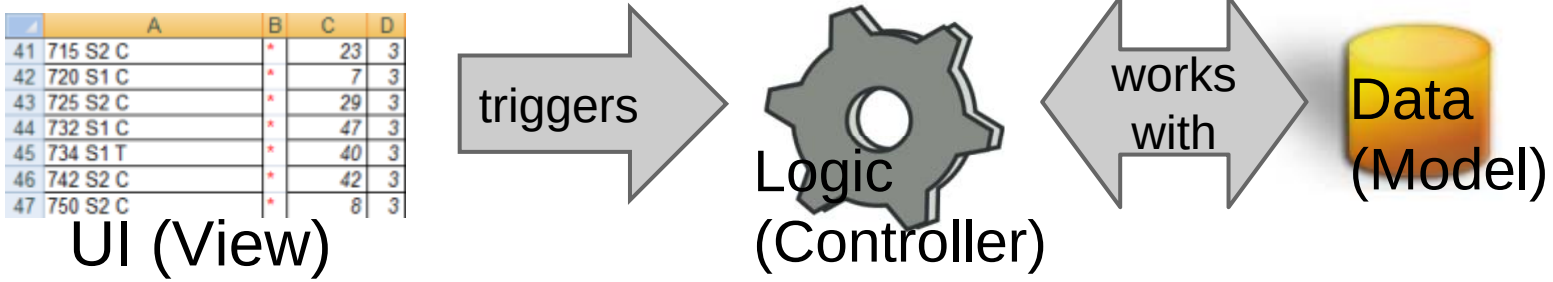
Hierarchical UI Decomposition



Separation of Data (Model) and UI (View)



Separation of Logic (Controller) and UI (View)



List Model Example

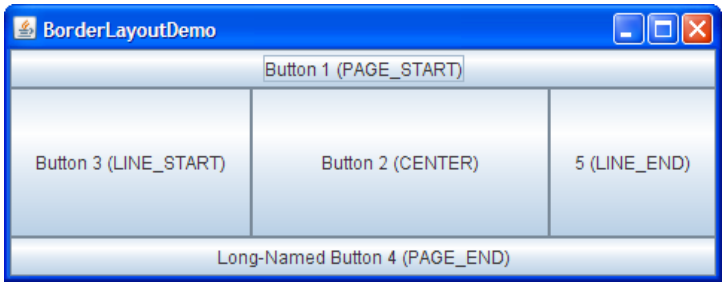
```
listModel = new DefaultListModel();
listModel.addElement("Alan Sommerer");
list = new JList(listModel);
…
hireButton.addActionListener(new ActionListener(){
  void actionPerformed(ActionEvent e) {
    listModel.addElement(nameField.getText());
  }});

fireButton.addActionListener(new ActionListener(){
  void actionPerformed(ActionEvent e) {
    int index = list.getSelectedIndex();
    listModel.remove(index);
  }});
…
```

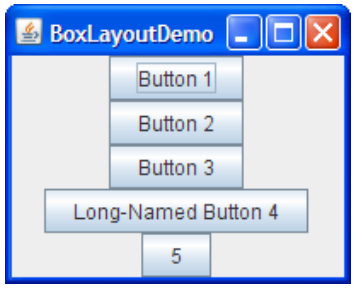
Full source code at:http://docs.oracle.com/javase/tutorial/uiswing/components/list.html

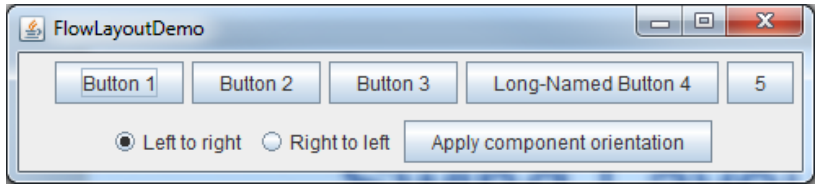
Swing Layout Managers

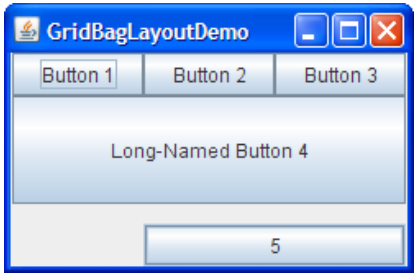
Layout managers implement the **`LayoutManager`** interface

- Has a method **`void layoutContainer(Container parent)`** that does all the layout work
- Is called by the container whenever it is changed / resized





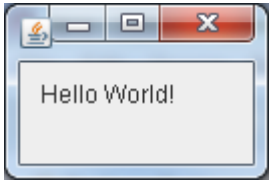




Creating a Custom Widget

1. Create new class that extends **`JPanel`**
2. Override **`paintComponent(Graphics g)`** with custom drawing code
   - Make sure to honor the **`width`** and **`height`** of the widget
   - Possibly call **`super.paint(g)`** to draw the superclass widget (e.g. unicolored background)
3. Override **`getPreferredSize()`** to return the right preferred size for your widget

```
class MyPanel extends JPanel {
    public void paintComponent(Graphics g) {
        super.paintComponent(g);
        g.drawString("Hello World!",10,20);
    }

    public Dimension getPreferredSize() {
        return new Dimension(100,50);
}  }
```



Good luck for the exam!!!

In the middle of difficulty
lies opportunity.
(Albert Einstein)