

Test First User Interfaces

ThoughtWorks®



Owen Rogers
orogers@thoughtworks.com

Challenging UI Testing Myths

- It's too hard to test
- I'm a server-side programmer
- It's only look-and-feel
- Why bother?

Just Another Library

- Widgets are just objects
- Take control and drive them with tests
- However:
 - Inherently multithreaded
 - Not designed for testability
 - Steep learning curve



Why test FIRST?

- Test expose problems in our design
- Help realise simple design
- Activation energy
 - Hard to write the first test
 - Overcome the inertia

Evil Wizards

- Generate unmaintainable code
 - High code duplication
 - Tight coupling
- Encourage bad OO design
- Inhibit learning
- Challenge: try changing the colour of all buttons uniformly across your application

Reveal()

- Use tests to take control of your UI
- Display any form or component using a single command
- Temporarily present GUI for display and interaction

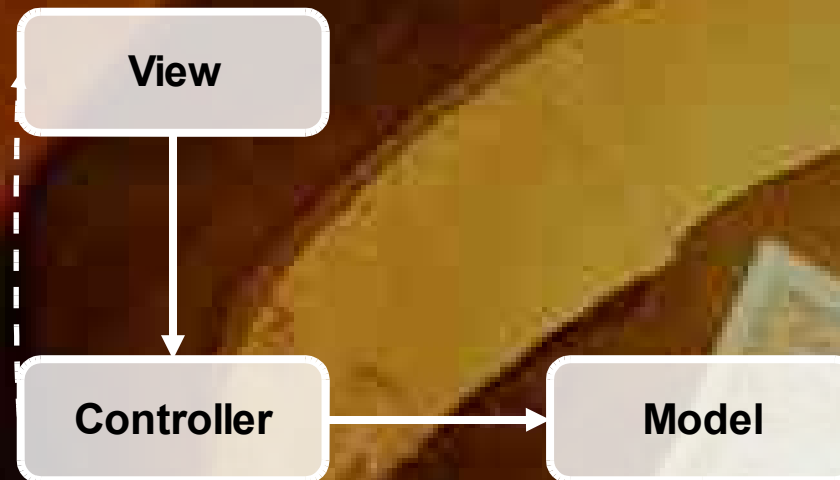
```
class GuiTestCase extends TestCase {
    protected void reveal(Component component) {
        JDialog jDialog = new JDialog();
        jDialog.setModal(true);
        jDialog.getContentPane().add(component);
        jDialog.pack();
        jDialog.setVisible(true);
    }
}

reveal(new SearchScreen());
```

The Humble Dialog Box

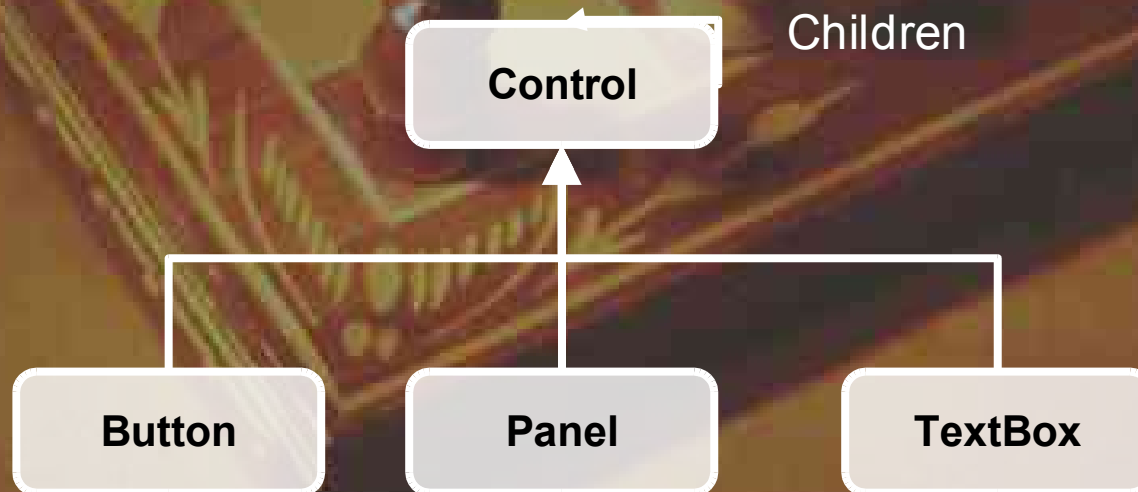
- Representation Layer
 - variant of Model-View-Controller
 - mirrors the functionality of the UI
 - caveat: completely decoupled from UI toolkit

```
SearchScreen screen = new SearchScreen(new SearchController());
```

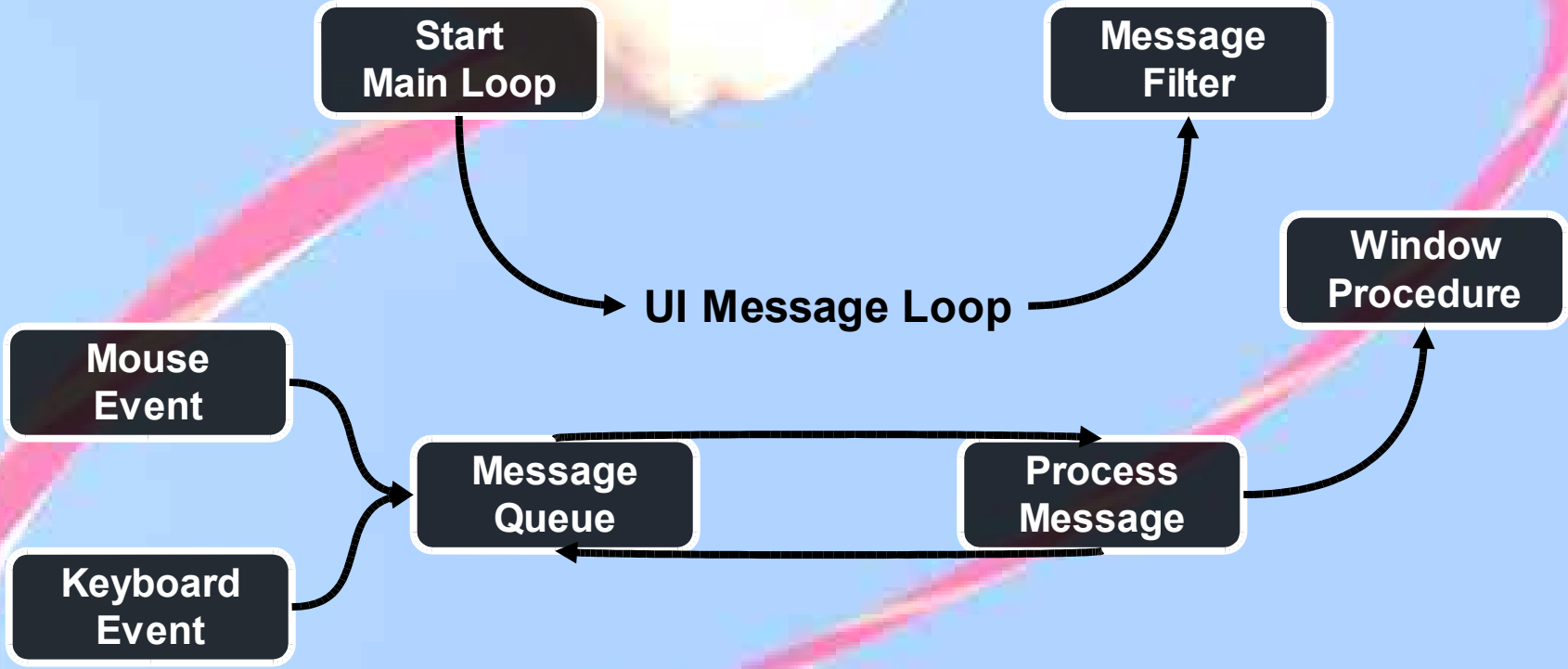
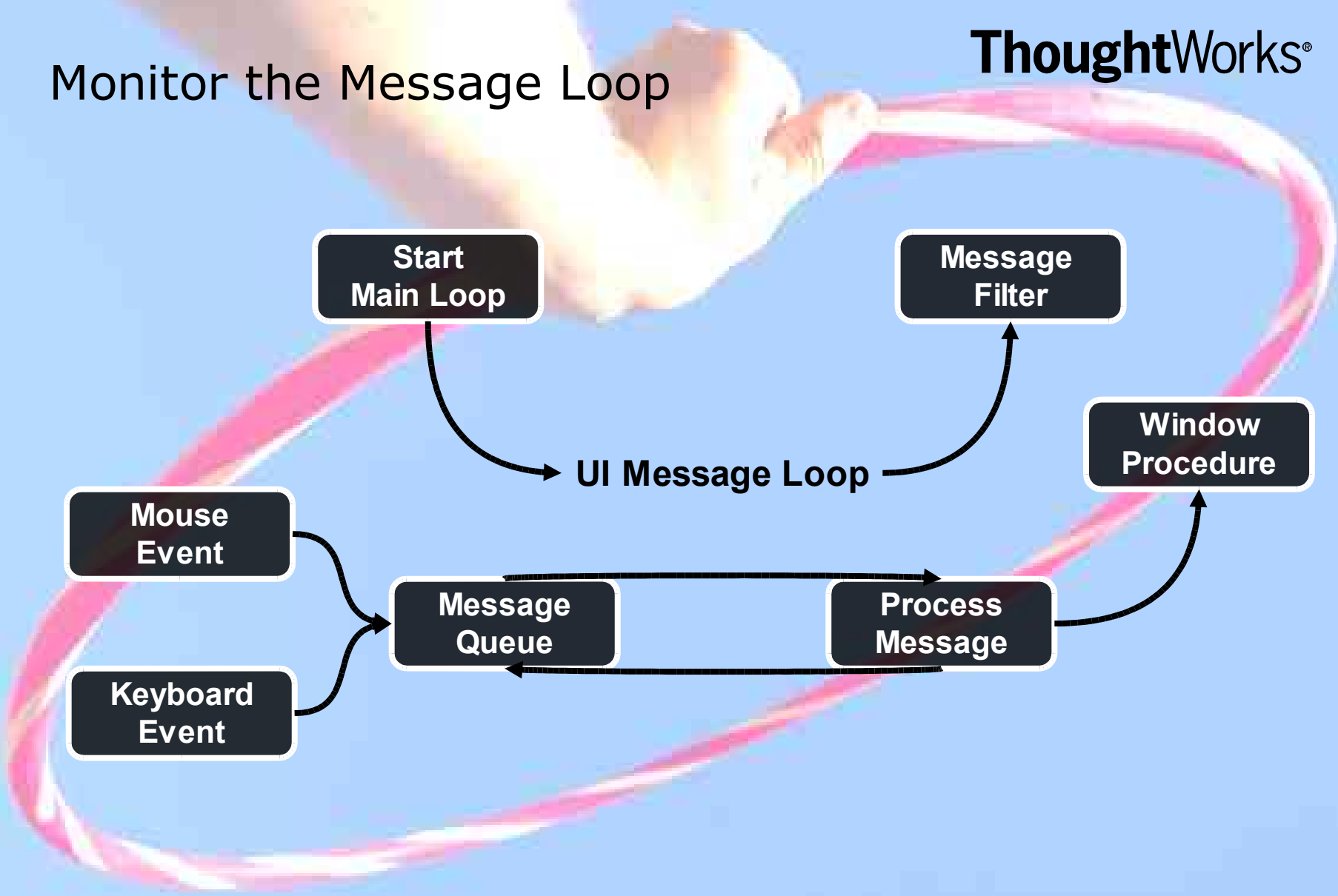


Learn the event model

- Attach trace event listeners to all controls
 - Navigate the control composite
 - Use reflection to find event handlers
 - Output raised event



Monitor the Message Loop



Reuse existing toolsets

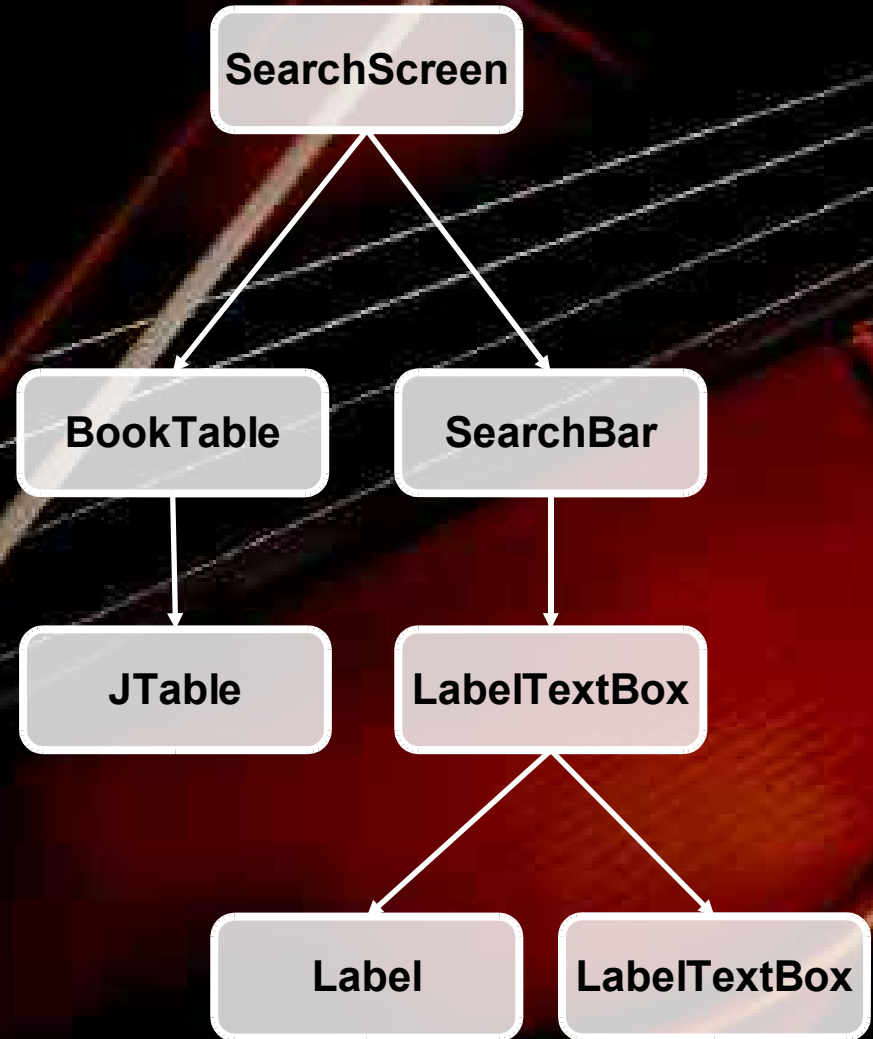
- Marathon
- Abbot
- NUnitForms



Design by composition

Search:

Title	Author	ISBN



Use widget factories

- Consolidate widget creation in factories
- Use constructor dependency injection to wire up widgets



Questions?

ThoughtWorks®

