

Frameworks

15-413: Introduction to Software Engineering

Jonathan Aldrich

With thanks to George Fairbanks



Hello, World in C



```
#include <stdio.h>
main() {
    printf ("Hello World!\n");
}
```

What context does the programmer need to know?

- When the programmer chooses to call printf() it will behave according to its specification

17 October 2005

Hello, World as Java Applet



```
import java.applet.Applet;
import java.awt.Graphics;
public class HelloWorld extends Applet {
    public void paint(Graphics g) {
        g.drawString("Hello world!", 50, 25);
    }
}
```

- What context does the programmer need to know?
 - By subclassing Applet, his applet will be run at the right time (i.e. no "main" method defined in the language)
 - The paint method needs to be implemented because the applet runner will use it as a callback
- This is not your father's Oldsmobile

17 October 2005

Frameworks



- Definition: "The skeleton of an application that can be customized by an application developer" – Ralph Johnson
- Characteristics
 - OO world: subclass parts of the framework
 - Specialized to a class of software
 - Hollywood model: "Don't call us..."

17 October 2005

Frameworks are Everywhere



- Java
 - Applets, Servlets, EJBs, Eclipse, NetBeans
- C/C++
 - Microsoft Foundation Classes, KDE, Gnome, .NET(?)
- Objective-C
 - Cocoa, Carbon (?)

17 October 2005

Framework Benefits



- Code reuse
 - Probably even more valuable than libraries
- High productivity
 - A few lines go a long way
 - Once you know the framework!
- Standardization
 - Platform look and feel
 - Integration between components

17 October 2005

Framework Challenges



- How to do X?
 - Must subclass C, override method m...
 - May need a sequence of method calls to get to the right object
 - getViewSite().getActionBars().getToolBarManager().add(newAction)
 - How to find this sequence?
- Distribution
 - Must coordinate several extension points
 - E.g. to add a context menu in Eclipse, you have to create a MenuManager in your view and a MenuListener that populates the menu when the user right-clicks
 - It's not obvious that you need the MenuListener (you could write reasonable-looking code without it), but if you forget it then things might not work
- Constraints
 - Must invoke super() when overriding
 - Can't call method X/Y from here
 - E.g. Swing threading goes both ways
 - Must handle certain events
 - App mysteriously fails if you forget
 - Make calls in sequence
 - You have to know when your methods are called
 - E.g. createPartControl in Eclipse for View setup

17 October 2005

Coping with Frameworks



- Read books
 - Good for an introduction
 - Good for reference
 - Too tedious to read beginning to end
- Read/copy source code
 - In practice, this is what people do most often
 - Can be hard to find code that is relevant; use search engines and book indexes
 - Danger that the code you find may be inappropriate for your problem
- Wizards
 - Magic code that works the first time
 - Limited to special circumstances
 - Still must learn the library if you want to modify it
- Read the framework code
 - Usually hopeless except for very simple cases

17 October 2005

Frameworks vs. Patterns



- Frameworks are more concrete
 - Actual code for a particular domain
 - Design patterns aren't code; they are a pattern that can be applied in code
- Frameworks are larger
 - May contain multiple patterns
- Frameworks are specialized to a domain
 - Design patterns apply in any domain

17 October 2005