GUIS: Graphical User Interfaces

Their mouse had a mean time between failure of … a week … it would jam up irreparably, or ... jam up on the table-- ... It had a flimsy cord whose wires would break. Steve Jobs: "... Xerox says it can't be built for < $400, I want a $10 mouse that will never fail and can be mass produced, because it's going to be the primary interface of the computer ..."

... Dean Hovey ... came back, "I've got some good and some bad news. Good news: we've got a new project with Apple. Bad news: I told Steve we'd design a mouse for 10 bucks."

... year later ... we … filed … and were granted a patent, on the electro-mechanical-optical mouse of today; ... we ended up ... [making] the mouse as invisible to people as it is today.

Steve Sachs interview on first computer with GUI: Apple Lisa (~$10K in 1982).
http://library.stanford.edu/mac/primary/interviews/sachs/trans.html

GUI (Graphical User Interface)

- Provides a friendly interface between user and program
- Allows event-driven or reactive programming: The program reacts to events such as button clicks, mouse movement, keyboard input
- Often is multi-threaded: Different threads of execution can be going on simultaneously

Two aspects to making a GUI:

1. Placing components (buttons, text, etc.) in it. TODAY
2. Listening/responding to events Next Lecture

GUI (Graphical User Interface)

- Provides a friendly interface between user and program
- Allows event-driven or reactive programming: The program reacts to events such as button clicks, mouse movement, keyboard input
- Often is multi-threaded: Different threads of execution can be going on simultaneously

We use Java's two packages for doing GUIs:

•**AWT** (Abstract or Awful Window Toolkit) —first one

•Swing —a newer one, which builds on AWT as much as possible

•JavaFX —completely new! Much more functionality, flexibility, but far too complicated to teach in CS2110.

We use Swing (AWT makes an appearance occasionally)

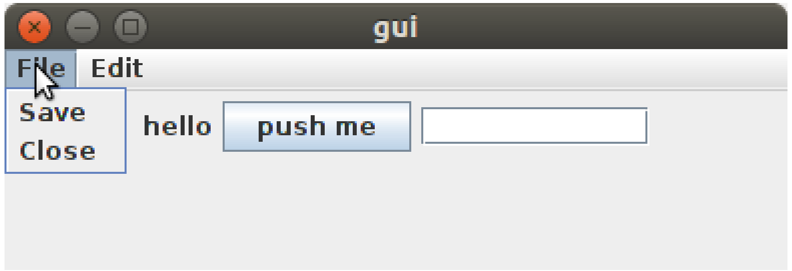
Notes on javax.swing vs. java.awt

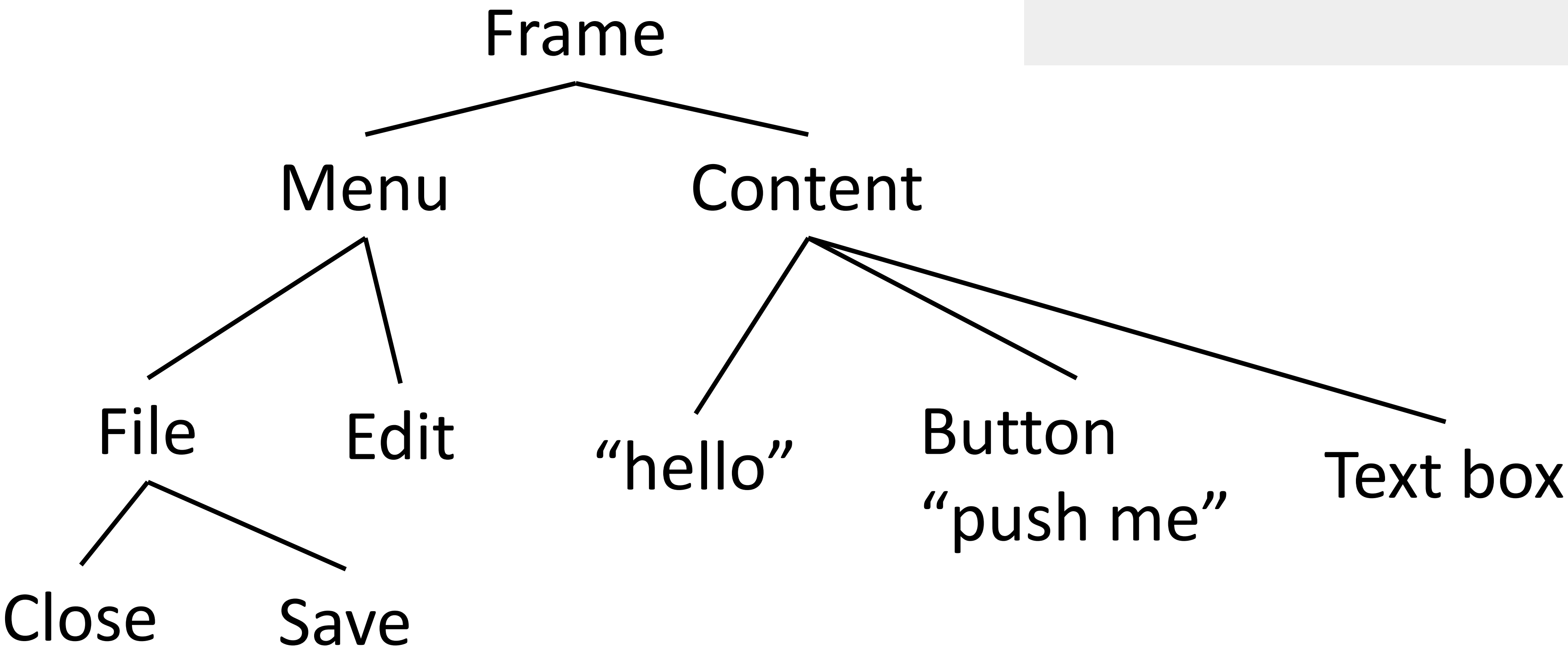
- java.awt is old; javax.swing is new
    - use swing wherever possible

- swing reuses some awt components
    - e.g. Dimension, LayoutManager, …

- swing component names start with J
    - e.g. JFrame,JButton instead of Frame, Button

GUIs are trees





Frame

Menu

Content

File

Edit

“hello”

Button
“push me”

Text box

Close

Save

Note: can parse trees…

Class JFrame

JFrame object: associated with a window on your monitor.

Generally, a GUI is a JFrame object with various components placed in it

Some methods in a JFrame object

hide() show() setVisible(boolean)

getX() getY() (coordinates of top-left point)

getWidth() getHeight() setLocation(int, int)

getTitle() setTitle(String)

getLocation() setLocation(int, int)

Over 100 methods in a JFrame object!

Class JFrame is in package javax.swing

Components

Component: Something that can be placed in a GUI window. Some useful concrete subclasses:

JButton:	Clickable button
JLabel:	Line of text
JTextField:	Field into which the user can type
JTextArea:	Many-row field into which user can type
JPanel:	Used for graphics; to contain other components
JCheckBox:	Checkable box with a title
JComboBox:	Menu of items, one of which can be checked
JRadioButton:	Same functionality as JCheckBox

Container:	Can contain other components
Box:	Can contain other components

Component Class Hierarchy

Component: Something that can be placed in a GUI window. These are the basic ones used in GUIs:

- Component
  - Container
    - JComponent
      - JButton
      - JToggleButton
        - JCheckBox
        - JRadioButton
      - JLabel
      - …

Note: just like

- Expr
  - Int
  - Sum
  - Product
  - …

Many operations are recursive tree traversals:

- paint, pack, show, handle events, …

Layouts

Containers have LayoutManagers
 - control layout of children

Examples:
 - FlowLayout: place children one after the other, wrap
 - BoxLayout: place everything in one row or column
 - BorderLayout: split container into 5 children
 - GridBagLayout, SpringLayout, …

Layouts may require extra parameter to add
 - e.g. f.add(c, BorderLayout.NORTH)).

Custom components

It’s easy to make your own components
 - just extend an existing class (e.g. JPanel)

Can override paintComponent(Graphics g)
 - Graphics object has methods like “drawLine”, …