HCI Summary

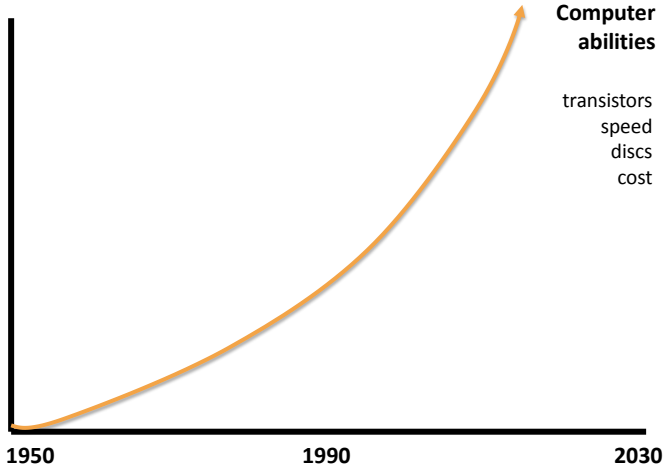
- Introduction to Human-Computer Interaction
- Useful vs. Useable
- Graphical User Interface (GUI)
- Java GUI (Swing)

By the end of this lecture, you will be able to describe what human-computer interaction is.

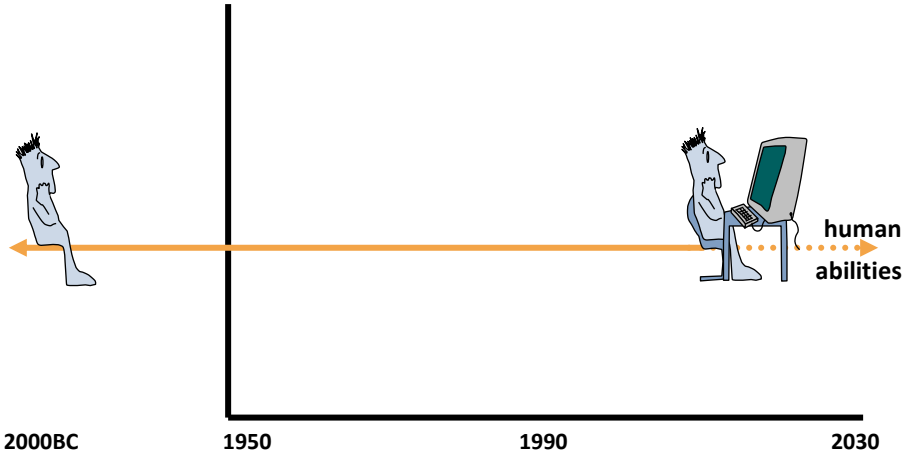
You will also be able to distinguish between usable and useful programs.

You will also be able to describe what an observer design pattern is.

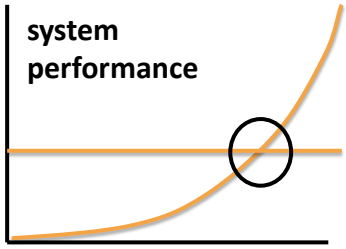
Moore's Law



Psychology

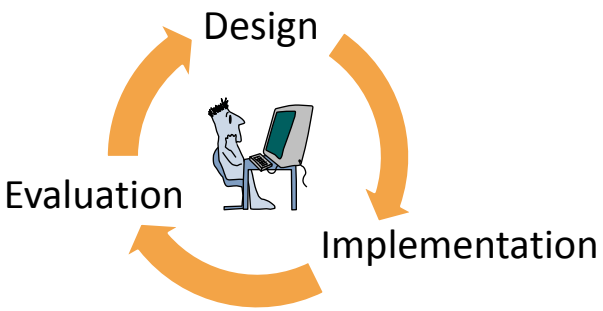


Where is the bottleneck?



Human Computer Interaction

- A discipline concerned with the



- of interactive computing systems for human use

Discussion: contrast this to our discussion about knowledge and understanding.

Where do the problems we analyse come from?

Discussion: How would you evaluate a program that you created?

What would you evaluate?

Useful vs. Useable

- Useful:
  - refers to function or purpose
- Useable:
  - refers to human ability

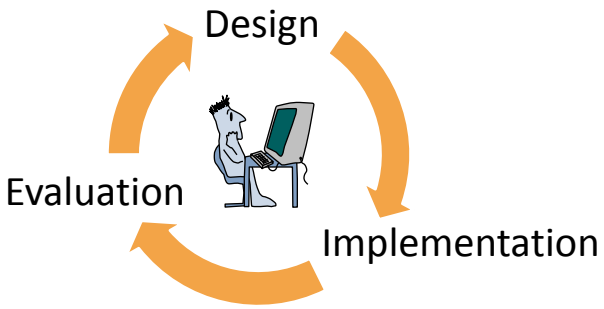
Exercise: What's an example of something that is useful, but not useable.

Exercise: What's an example of something that is useable, but not useful.

Question: which of these two qualities is important when you make a program?

Challenge: How do you make your program both useful and useable?

Iterative Design



Discussion: How would using an iterative design cycle affect how you write your code?

What would make it easier to iterate?

Graphical User Interface (GUI)

- Most programs you've written so far use the command line
- Most programming languages have a way to create GUIs (sometimes through a separate library)

Discussion: If you were to design a program that uses a GUI, how would it differ from what you've written so far?

How would you handle a mouse click? or when someone presses a button?

Observer Design Pattern

"The observer pattern ... is a software design pattern in which an object, called the subject, maintains a list of its dependents, called observers, and notifies them automatically of any state changes, usually by calling one of their methods. It is mainly used to implement distributed event handling systems."

"Observer Pattern," *Wikipedia: The Free Encyclopedia*

Often referred to as: The Event Model

Java Swing

- Tutorial: http://java.sun.com/docs/books/tutorial/uiswing/
- Use inheritance to create buttons, menus, tabs, etc.
- Use an event model to handle mouse clicks, keyboard/button presses, etc.

Example: GUI in Java

```
import javax.swing.*;

public class SimpleGUIProgram extends JFrame
{
    public SimpleGUIProgram()
    {
        super("A GUI!");
    }

    public static void main(String[] args)
    {
        SimpleGUIProgram program = new SimpleGUIProgram();
        program.setVisible(true);
    }
}
```

Example (cont'd)

```
public SimpleGUIProgram()
{
    super("A GUI!");
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

    JButton button = new JButton("Okay");
    button.addActionListener(this);

    JTextArea textArea = new JTextArea(70, 100);

    JScrollPane scrollPane = new JScrollPane(textArea);
    scrollPane.setPreferredSize(new Dimension(640, 480));

    JPanel mainArea = new JPanel();
    mainArea.add(scrollPane);
    mainArea.add(button);

    setContentPane(mainArea);
    pack();
}
```


Example (cont'd)

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

public class SimpleGUIProgram extends JFrame
                                    implements ActionListener
{
    // ...

    public void actionPerformed(ActionEvent event)
    {
        JOptionPane.showMessageDialog(this,
                                        "You pressed okay!");
    }

    // ...
}
```

Demo

Discussion: What kind of events would you have to consider in the demo you just saw?

HCI Summary

- Introduction to Human-Computer Interaction
- Useful vs. Useable
- Graphical User Interface (GUI)
- Java GUI (Swing)

Next Class

- I will be absent: Christopher Collins will provide the lecture
- Design Patterns (Part 2)