

Applets & Video Games

The Plan

❖ Applets

- ❑ Demo on making and running a simple applet from scratch
- ❑ Demo on making and running a simple application from scratch

❖ Video Games

- ❑ Measurements
- ❑ Frame rates
- ❑ Threads

Applets

- ❖ **Definition**
- ❖ **Differences from applications**
- ❖ **Download process**
- ❖ **Use of html**
- ❖ **Use of jar files**
- ❖ **Example**

Definition

From the Java 5.0 API

An applet is a small program that is intended not to be run on its own, but rather to be embedded inside another application.

The name Applet is derived from the name Application.

Applet vs. Application

Applets

- ❖ Run in web browser
- ❖ Often downloaded from untrusted site
- ❖ Restricted from file system access
- ❖ Restricted from outside network communication

Applications

- ❖ Run independently
- ❖ Typically obtained through trusted source
- ❖ Allow creation and modifications of files
- ❖ Allow outside network communication
- ❖ Must have a `main` method

Running an Applet

1. Load a web page with an `<applet>` tag embedded in the HTML
2. Load the compiled applet from website to local machine
3. Run the compiled applet on the local machine

HTML for Applet

```
<applet code="pong/Pong.class"  
  archive="pong.jar" width=200  
  height=200></applet>
```

- ❖ **code is the name of the class that extends JApplet**
- ❖ **archive is the name of the jar file containing all classes**

Jar files

Jar is short for Java Archive

- ❖ **Compresses files and directories into a single file**
- ❖ **Can be executed in compressed format**
- ❖ **Files and directories can be extracted**
- ❖ **Can contain any combination of source code, byte code, and other files**

Demo

Applet Demo

- ❖ **Make an applet in Eclipse**
- ❖ **Make the html in composer**
- ❖ **Save both to network drive**
- ❖ **View from the web**

Application Demo

- ❖ **Make an application in Eclipse**
- ❖ **Run in Eclipse**

Video Games

- ❖ **Simulation**
- ❖ **Measurement units**
- ❖ **Discrete/Continuous**
- ❖ **Monitor frame rate limitation**
- ❖ **Model frame rate limitation**
- ❖ **User interaction rate**
- ❖ **Threads overview**

Video Game as Simulation

Video games are simulations of the real world and worlds that do not exist. These simulations are built for our pleasure, but may serve other purposes as well.

Examples

- ❖ **Flight simulator**
- ❖ **Oregon trail**
- ❖ **Pinball**

Measurement Units

- ❖ Initial setup in coordinate system with origin at top left and (1, 1) at bottom right
 - ❑ Allows simple scaling to varying screen resolutions
 - ❑ Can be done hierarchically
- ❖ Distances in pixels
- ❖ Time in seconds
- ❖ Velocity in pixels/second

Discrete vs. Continuous Time

Discrete

- ❖ Used to approximate continuous
- ❖ Simple conceptually for good rough estimates
- ❖ Causes problems when modeling continuous functions with too coarse grain estimates

Continuous

- ❖ Requires abstract representation or infinite precision
- ❖ Requires analytical reasoning
- ❖ Conceptually difficult to model directly

Monitor Frame Rate

Why 75-85 Hz (Frames/second)?

Because we don't actually visually process continuously

Smooth fast movement?

Consider a rate of 1 pixel a second would take more than 8 seconds to move across the screen.

For fast movement there must be jumps in location.

Model Frame Rate

Model is continuous.

Frame rate is discrete.

The granularity of frame rate is too coarse for our continuous model.

How do we solve this problem? Two separate rates:

- ☐ **Monitor frame rate**
- ☐ **Model frame rate**

User Interaction Rate

Devices such as the keyboard and mouse must also be polled at regular intervals.

At what rate should they be polled?

Depends on:

- ❖ **Available compute power**
 - ❑ **In contention with monitor frame rate**
 - ❑ **In contention with model frame rate**
- ❖ **Which thread has priority**

How it happens “all at once”

Threads!

- ❖ Threads are like programs within programs
- ❖ Seem to run all at once, but typically share resources, primarily the processor
- ❖ Cost overhead for switching the thread running

Summary

- ❖ User interaction, model frame rate, and monitor frame rate all contend for the processor.
- ❖ Threads enable programs to behave as if several sub-programs (threads) were running at once.
- ❖ Continuous events can be modeled discretely.
- ❖ Careful selection of measurement units can simplify program modifications.
- ❖ Video games can be viewed as simulations.

Summary

Applets

- ❖ are like small applications run from a web browser.
- ❖ have security restrictions.
- ❖ require HTML code to be executed
- ❖ are downloaded from a remote site and executed locally
- ❖ use jar files to compress and combine all compiled code and supporting files