CS193k, Stanford
Spring, 2000-01

Handout #18
Nick Parlante

Java Future

Client Side Java

Applets

Largely failed

.Jar file applications

Package .class files into a jar file

Double click on the .jar runs the application (on windows anyway)

Code does not run in a "sandbox"

Java Web Start

http://java.sun.com/products/javawebstart/

Released now

Client installs the JWS loader on their machine once (like installing Acrobat)

Package in a .jar

Write a .jnlp file that describes the package -- where's one I made for a DiceMachine class in a DiceMachine.jar file (made by tweaking the Sun sample)...

```
<?xml version="1.0" encoding="utf-8"?>
     <!-- JNLP File for SwingSet2 Demo Application -->
     <jnlp
       spec="1.0+"
       codebase="http://www.stanford.edu/~nick/dice/"
       href="dice.jnlp">
       <information>
         <title>Dice Application</title>
         <vendor>Sun Microsystems, Inc.</vendor>
         <homepage href="docs/help.html"/>
         <description>SwingSet2 Demo Application</description>
         <description kind="short">A demo of the capabilities of the Swing Graphical
User Interface.</description>

         <!-- <icon href="images/swingset2.jpg"/>   -->
         <offline-allowed/>
       </information>
       <security>
<!--  we are not signed, so we don't request all perms  <all-permissions/>   -->
       </security>
       <resources>
         <j2se version="1.2"/>
         <jar href="DiceMachine.jar"/>
       </resources>
       <application-desc main-class="DiceMachine"/>
     </jnlp>
```

Unsigned code runs in a sandbox

The client just downloads the .jnlp file which points to enough info for the client to download and run the java code.

Can run with or without a net connection once downloaded.

Can check for updates automatically

The point: You send someone just a URL, and they can just click it to run the program on their machine.

Will JWS Catch On?

Like Flash catching on -- chicken-and-egg problem that works best if many clients have it pre-installed.

This will be hard since Microsoft controls the dominant OS and browser, and Microsoft hates Java

Enterprises love it internally -- easy way to distribute and update software

1.4 Features...

http://java.sun.com/j2se/1.4/docs/relnotes/features.html

New IO - NIO

Non-blocking IO for sockets

vs. the old 1-thread-per-socket model

New buffering system

Interruption should now work to un-block IO

Regular Expressions

Similar to Perl -- match and extract char regions

Syntax is more structured than in Perl -- more steps, more syntax

```
    // Pattern used to parse lines
...
    private static Pattern linePattern  = Pattern.compile(".*\r?\n");
        CharBuffer cb = ...;
...
        Matcher lm = linePattern.matcher(cb);   // Line matcher
        while (lm.find()) {
            CharSequence cs = lm.group();
            System.out.println(cs);
        }
```

Swing

FocusManager -- keyboard typing

Which component is currently "focused" -- getting keystrokes

New centralized handling of focus -- acquire/release/refuse-to-release focus

Image drawing slowed down significantly in 1.2 vs. 1.1, since the handling of images was made more abstract. (Perhaps another example of Sun preferring "elegance" a little too much over practical issues.)

A big issue for Swing apps, especially when using X Windows

1.4 gets some of the speed back

Slow startup time is still an issue in my mind, but every year computers get faster, so maybe in a year or two it won't matter.

Old Serialization

Design -- how to you serialize off a Java class?
Old serialization: write out its ivars
Problem: what if the class changes impl?

New, XML "Persistence"

http://java.sun.com/j2se/1.4/docs/guide/beans/index.html
Only serialize state that is accessible through public get/set methods (the "bean" view of an object)
This is the technology that underlies the new GUI/Bean/XML layout editor technology
Be smart about constructor defaults...
To serialize Foo f...
1. Construct Foo s;
2. Compute what setXXX() messages are necessary so that s looks like f.
3. Record the arguments for the ctor/setXXX sequence -- that **is** the persistent form of f
Advantages: totally independent of implementation. In fact you could serialize as Foo, and then read back into a different class, say Bar, so long as Bar had the same public ctor/get/set semantics as Foo.

Assert

http://java.sun.com/j2se/1.4/docs/guide/lang/assert.html
assert i==0;
Throws AssertionError
Can be turned on and off by class or package at CT or RT
Never put code that needs to run in an Assert, since the assert may be effectively deleted in some cases, but the program should still work

JavaDoc

Rules of thumb
Use verb forms that end in "s" -- reads, builds, rotates..
Don't mention the actor "this method reads" "this class represents" -- just say "reads ..."

```
// Piece.java

import java.awt.*;
import java.util.*;

/**
 An immutable representation of a tetris piece in a particular rotation.
 Each piece is defined by the blocks that make up its body.

 @author Nick Parlante
 @version   1.0, Mar 1, 2001
*/
public final class Piece {
   /**
    Returns a piece that is 90 degrees counter-clockwise
    rotated from the receiver.
   */
   public Piece nextRotation() {
      return next;
   }

   /**
    Returns an array containing the first rotation of
    each of the 7 standard tetris pieces.
    The next (counterclockwise) rotation can be obtained
    from each piece with the {@link #nextRotation()} message.
    In this way, the client can iterate through all the rotations
    until eventually getting back to the first rotation.
   */
   public static Piece[] getPieces() {
```

MIDP

Mobile Information Device Profile

http://java.sun.com/products/midp/

http://java.sun.com/products/midp/palmOS.html -- Palm midp beta

Subset of Java for small devices -- not as heavyweight as Swing

Allow you to write small apps that work on Palm, Windows CE, ...

Also, Connected Limited Device Configuration -- CLDC -- phones, etc.

Many vendors are excited about the "small device" space -- a new frontier vs. the desktop

Java Generics (not in 1.4)

http://developer.java.sun.com/developer/earlyAccess/adding_generics/

Compile time types

The RT is the same -- really it's checking the type every time, but you don't have to put the cast in at CT

Cleans up the code and finds some erors at CT, which are now masked by all the casting

```
// Suppose Foo responds to the bar() message
ArrayList<Foo> list;
Foo f = ...
list.add(f);...
...
Iterator<Foo> it = list.iterator();
while(it.hasNext()) {
   it.next().bar();  // NOTE: no cast required, it.next() has correct CT type
   ...
}
```

Java Dynasty?

Age of interpreted languages -- CPU time vs programmer time

Portability

Java Network effects

Good tools, everyone learns it in school, good books .... for any new project, Java has a lot of built-in network effect advantages

20 Years in the future -- still coding in Java?