

---

# Project Overview

CSE 403, Spring 2003  
Software Engineering

<http://www.cs.washington.edu/education/courses/403/03sp/>

# Readings and References

---

- References
  - » What will people pay for? Dan Bricklin.
  - » Accessing a whole new world via multimedia phones. Dan Gillmor.
  - » J2ME Data Sheet from Sun
- Links to all of these are on the "otherlinks" web page

# Initial project motivation

---

- Dan Bricklin: What will people pay for?
  - » "You keep reading stories about how cellphones will be used for checking stock quotes and making trades, buying stuff, and other eCommerce. It seems business plans are based on people paying for such stuff. I think that eCommerce is not where things will go."
  - » *"People like to interact with people they care about. The interactions are often simple, but personally important. They are willing to pay money for this."*

<http://www.bricklin.com/peoplepay.htm>

# Opportunity

---

- The cell phone / mobile industry is in the state that the PC industry was in 15-20 years ago
  - » nobody knows what the killer apps will be, although simple messaging (aka email) is big
  - » there is still a relatively competitive environment with multiple operating systems and suppliers
  - » the existing players are not necessarily the ones who will dominate or have the big ideas
  - » the technology opens a wide range of new possibilities that have never existed before

# *mobile means personal*

---

- Mobile devices are small, portable, and personal
- Business support
  - » contact list, daily calendar, sales support
- Personal interest
  - » ringtones, games
  - » personal information management
  - » connectivity
    - web sites, text messaging (SMS), image messaging (MMS)
    - always-on information access is *very* nice! Google ...

# Connectivity

---

- Cell networks are growing in speed
- Data types are already a rich environment and the phones are rapidly growing more capable
  - » larger color screens
  - » sound capture and playback
  - » video capture and playback
- Wireless Connectivity
  - » *all media at all times in all places*



# Our project domain

---

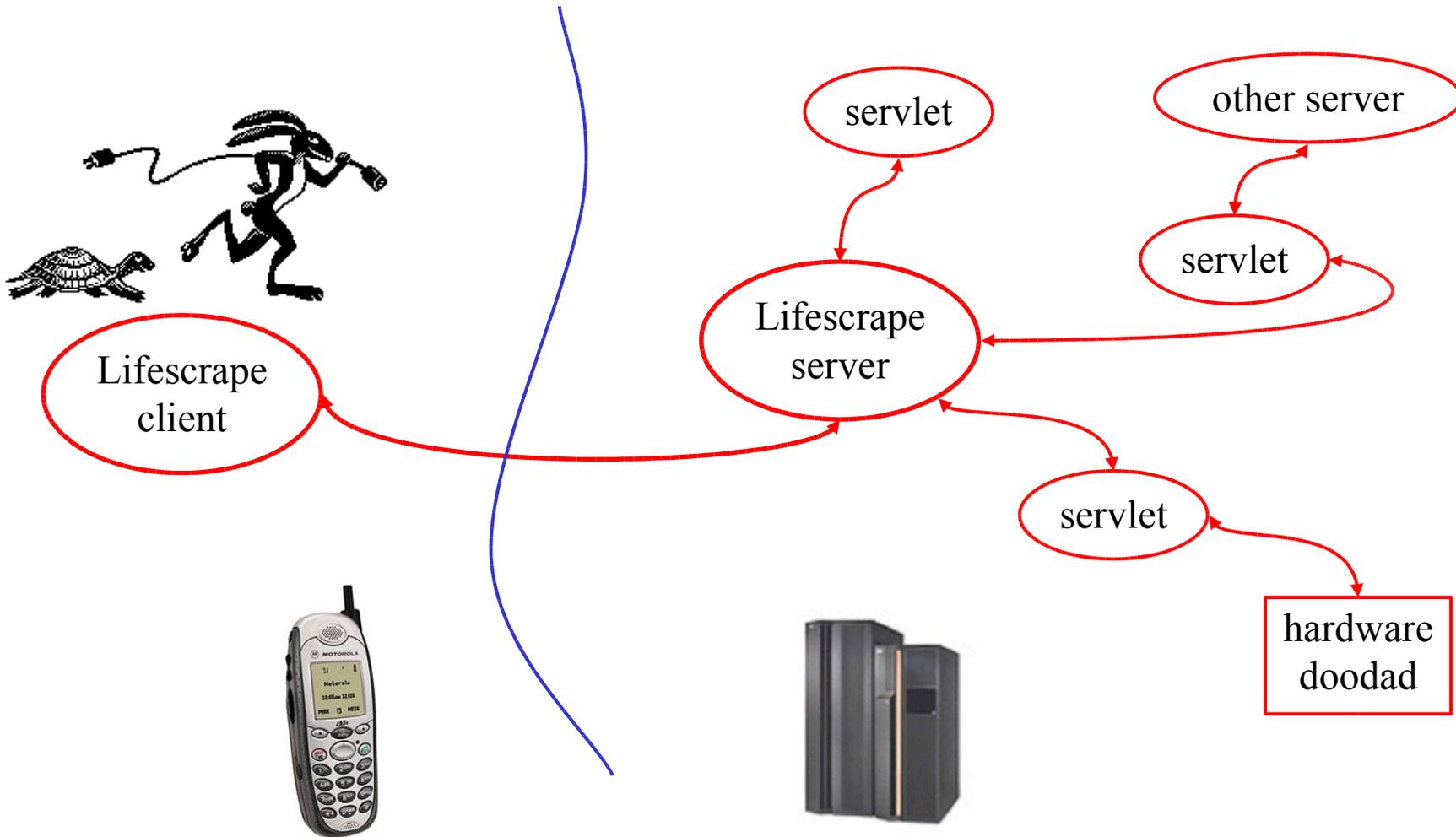
- We will focus on mobile clients, simple servers, and network links
- Lifescrape, the application
  - » Custom client midlet on the cell phone manages interaction with the user: commands and display
  - » Server can manage information itself or get it from any network aware source
  - » Servlet packages the information in concise form for transmission to/from cell phone upon request

# What sort of information?

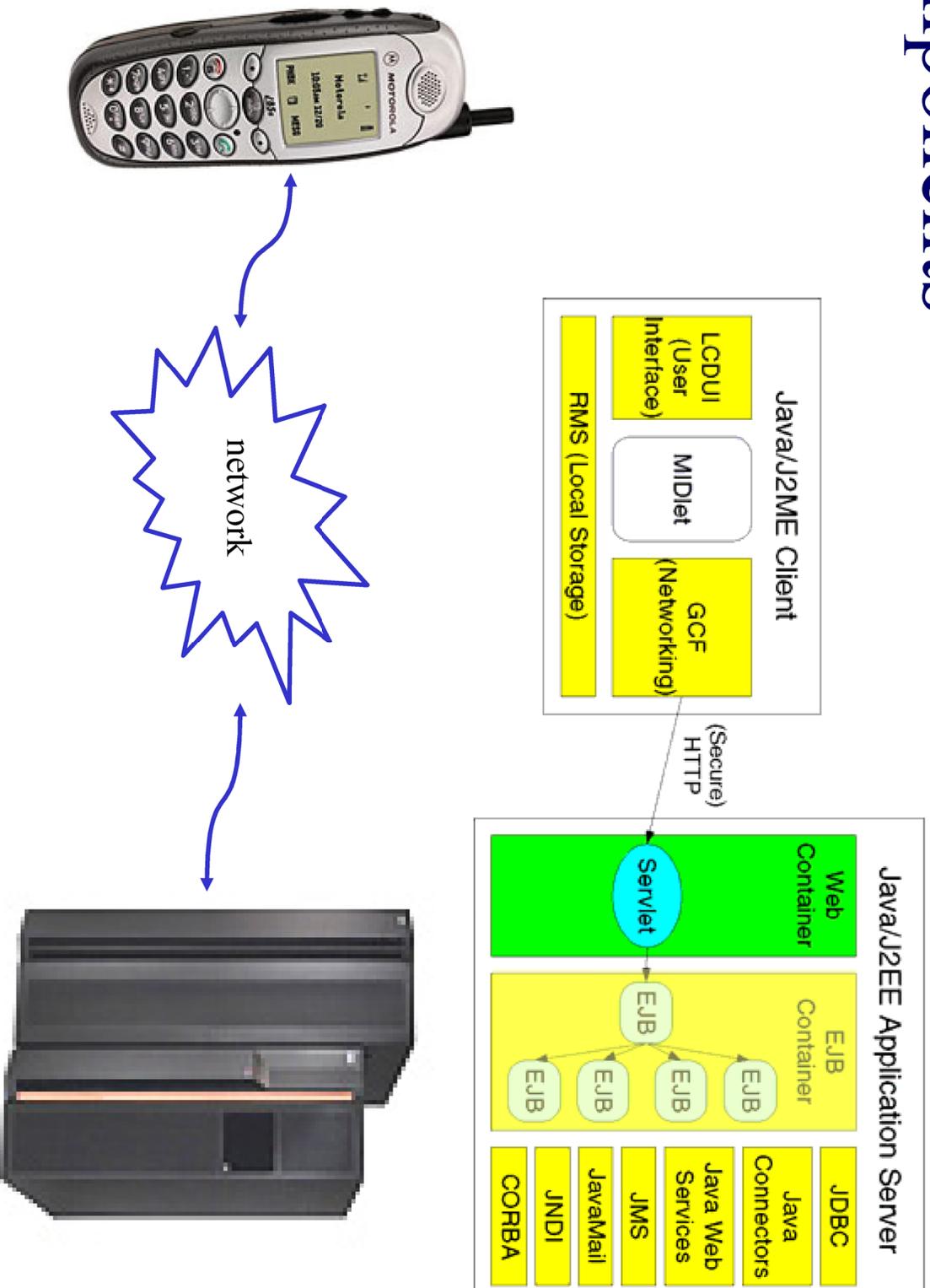
---

- Anything you want to know that can somehow be manipulated by a network aware application
  - » online game events and status, virtual agents
  - » transportation events (next bus, commute time)
  - » friends on-line (AIM, Unix finger)
  - » device state/control (laundry machine status, coke machine, coffee maker, room occupied?)
  - » video/audio snapshots (BBC news, sports, ...)
- Where is all this ultimately heading?
  - » Gillmor: *Ask a teenager. She'll be first to figure it out.*

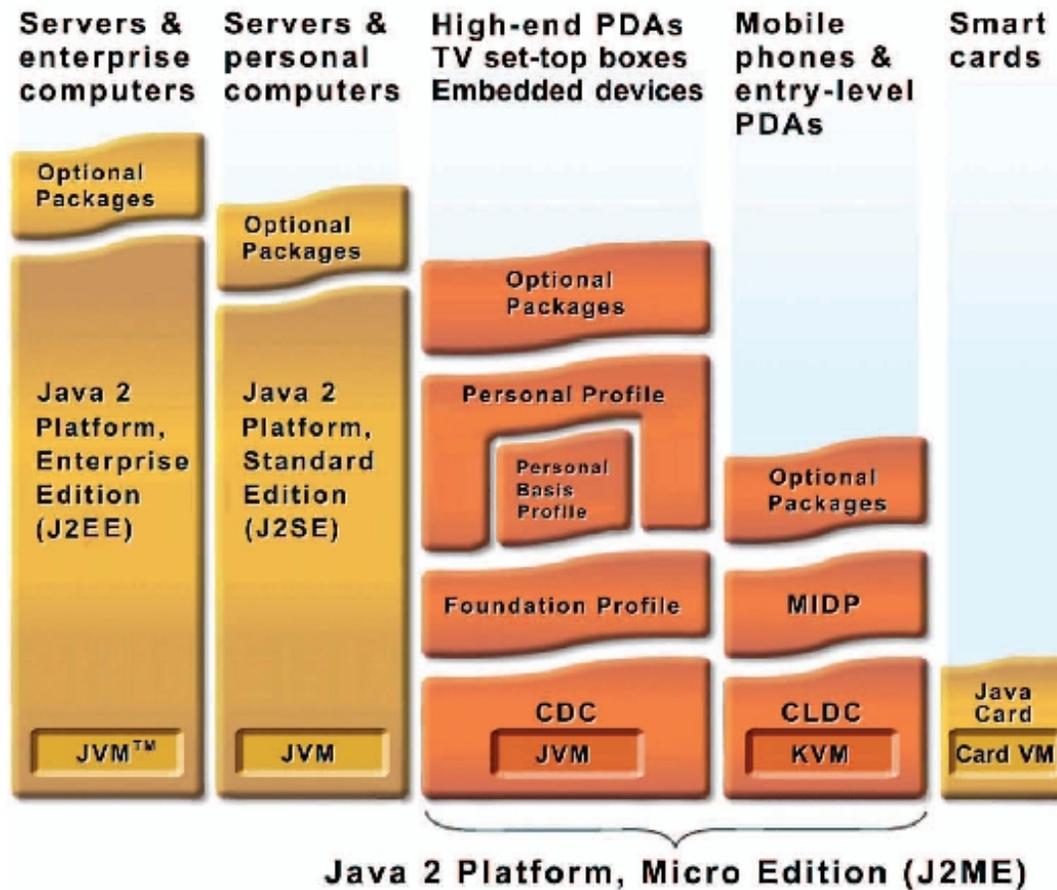
# Architecture bubbles



# Software Components



# Java editions - huge to tiny



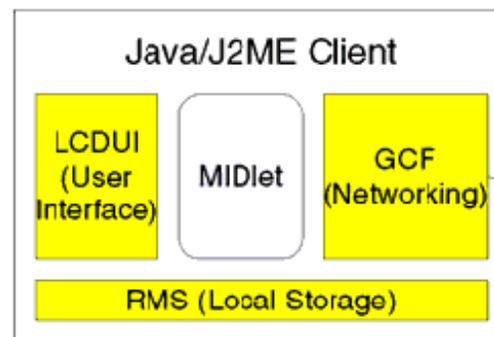
# J2ME terminology

---

- J2ME architecture
  - » configurations, profiles, and optional packages
- *Configuration* is a virtual machine and a minimal set of class libraries
  - » Connected Limited Device Configuration (CLDC)
- *Profiles* define the application life cycle model, user interface, device interface
  - » Mobile Information Device Profile (MIDP)
- *Optional packages* add features
  - » Mobile Media API (MMAPI)

# Java 2 MicroEdition (J2ME) packages

`javax.microedition.midlet.*`  
manage midlet life cycle



(Secure)  
HTTP

`javax.microedition.io.*`  
manage connections

`javax.microedition.lcdui.*`  
control screen display and listen for  
user commands

`javax.microedition.rms.*`  
manage simple record oriented data  
storage

# Smaller class libraries than J2SE

---

- The class libraries in J2ME are much smaller than they are in J2SE (the standard desktop edition of Java)
- Devices are small in memory and horsepower
  - » fewer classes reduces footprint significantly
- Display is limited
  - » no overlapping windows, one application at a time
- [API documentation](#) is located in Wireless Toolkit installation directory – *browser button*

# J2ME development tools

---

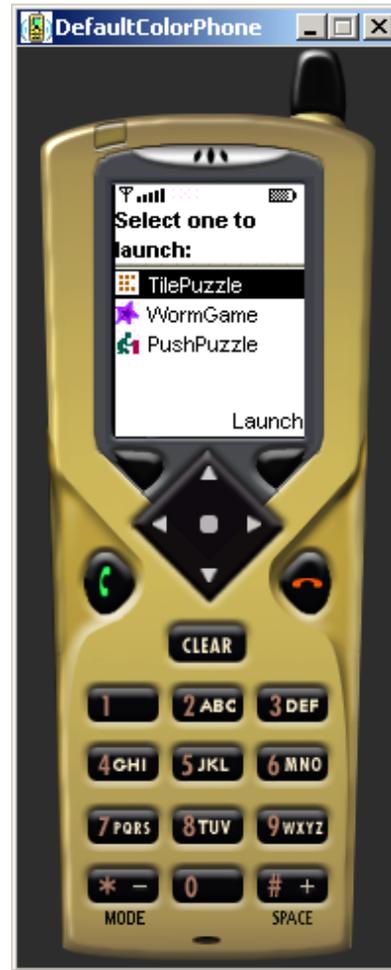
- Primary tool is Wireless Toolkit 1.0.4
- It's a little bit simplified, but it provides the ability to compile, verify, and package midlets
  - » uses J2SE compiler to compile sources
  - » special system class libraries for J2ME
  - » preverify and package midlet and update jad file
- User manual included in the installation
  - » read it over to get an introduction to the process

# run WTK with Ant or KToolBar

---

- KToolBar is a minimal development environment for developing MIDlet suites.
  - » From the KToolBar, you can:
    - Create a new project or open an existing one
    - Build, run, and debug your MIDlet
    - Package your project files
    - Modify the attributes of your MIDlet suite
- Ant is a general purpose build tool like make
  - » you can run WTK emulators from Ant
  - » you can run Ant from jEdit

# Write once, run in a bunch of places



# sample ant build file

---

build.xml

This Ant buildfile is used to build a Java midlet to run on a MIDP device (eg, a cell phone). The tasks assume that the Wireless Toolkit has been installed and that the WTK directory is defined in the `wtk.home` property. Also, the name of the directory that this build file is in is used as the value of `midlet.name` for this midlet suite unless `midlet.name` is set elsewhere.

Main targets:

<code>build</code>	Compile and package the midlet suite.
<code>buildjavadoc</code>	Create the documentation for this MIDlet suite
<code>clean</code>	Delete existing build and distribution directories
<code>dist</code>	Create distribution zip files
<code>init</code>	Initialize
<code>run</code>	Run the midlet suite in the specified emulator

Default target: `build`

# sample ant build file

---

The tasks assume that the directory structure is

```
../midlet.name/  
  build.xml  
  build.properties  
  build.number  
  src/  
  doc/  
  build/  
    classes  
    javadoc  
    bin  
  dist/
```

The build.properties file must define the correct value for wtk.home.

The build.number file contains a build number that is incremented automatically every time the build target is executed.

The java sources and the java application descriptor (jad) file are kept in the src directory. Package source directories are subdirectories of src.

The doc directory holds non-javadoc files like release notes.

The build directory is a working directory used to build the midlet.

The dist directory is the destination of the distribution files.

The build and dist directories are recreated after a clean.

---

# ant clean and build

---

```
C:\demo\spma>ant build
```

```
Buildfile: build.xml
```

```
init:
```

```
clean:
```

```
  [delete] Deleting directory C:\demo\spma\build
```

```
  [delete] Deleting directory C:\demo\spma\dist
```

```
build:
```

```
  [mkdir] Created dir: C:\demo\spma\build
```

```
  [mkdir] Created dir: C:\demo\spma\build\classes
```

```
  [wtkbuild] Compiling 1 source file to C:\demo\spma\build\classes
```

```
  [mkdir] Created dir: C:\demo\spma\build\bin
```

```
  [wtkpackage] Building jar: C:\demo\spma\build\bin\spma.jar
```

```
  [wtkpackage] Preverifying C:\demo\spma\build\bin\spma.jar
```

```
  [wtkpackage] Updating JAD file C:\demo\spma\src\spma.jad
```

```
    [copy] Copying 1 file to C:\demo\spma\build\bin
```

```
BUILD SUCCESSFUL
```

```
Total time: 10 seconds
```

```
C:\demo\spma>
```

# ant run

```
C:\demo\spma>ant -Demulator=DefaultColorPhone run
```

```
Buildfile: build.xml
```

```
init:
```

```
run:
```

```
[wtkrun] Running C:\demo\spma\build\bin\spma.jad in DefaultColorPhone  
[wtkrun] Execution completed successfully  
[wtkrun] 87525 bytecodes executed  
[wtkrun] 9 thread switches  
[wtkrun] 319 classes in the system (including system classes)  
[wtkrun] 672 dynamic objects allocated (24648 bytes)  
[wtkrun] 4 garbage collections (12976 bytes collected)  
[wtkrun] Total heap size 1048576 bytes (currently 1034856 bytes free)
```

```
BUILD SUCCESSFUL
```

```
Total time: 1 minute 54 seconds
```

```
C:\demo\spma>
```

