Developing web applications for PDAs and Cellphones

Mobile Station Execution Environment (MExE)
- MExE is a standard for defining various levels of wireless communication
- These levels are called *classmarks*
- Mobile devices can support more than one classmark
- Current Defined classmarks include:
  - Classmark 1: Wireless Application Protocol (WAP)
  - Classmark 2: PersonalJava & JavaPhone (*includes classmark 1*)
  - Classmark 3: J2ME, CLDC & MIDP

WAP (Wireless Application Protocol)
- What is WAP?
  - It is a set of protocols for wireless data transmission
  - It strives to be an equivalent of what the internet is for the wired world
  - Designed to address issues specific to mobile commerce
Issues WAP Addresses

- The limitations of wireless communication
  - Intermittent network connectivity
  - Large latency
- Providing an easy to use interface for mobile devices
  - The screens are small
  - Limited input types
- Security concerns of transmitting wireless data
- Limited power and computational resources of mobile devices

WAP Gateway

- The WAP gateway provides a solution to the limitations of the wireless networks
- Mobile users connect to the gateway and any communications they request will be passed through the gateway
- It provides a access point to the rest of the web
- Takes over many cpu intensive operations for the mobile device
- Handles encryption and decryption as different (less strenuous) encryption protocols are used for mobile devices

Wireless Application Protocols

- WAP is based on:
  - WAE (Wireless Application Environment)
    - WML (Wireless Markup Language)
    - WMLScript (the javascript equivalent for WML)
    - WTAI/WTA (Wireless Telephony Application Interface)
  - WSP (Wireless Session Protocol)
  - WTP (Wireless Transaction Protocol)
  - WTLS (Wireless Transport Layer Security)
  - WDP (Wireless Datagram Protocol)

Wireless Markup Language

- WML is XML based
  - Follows strict xml formatting rules
- Replaces HTML in the WAP browser
- WML sites are called decks and are made up of cards
- The cards are the pages that the users views
Wireless Markup Language

<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN" "http://www.wapforum.org/DTD/wml_1.1.xml">

<wml>
  <card id="card1"> ...... </card>
  <card id="card2"> ...... </card>
  ......
  <card id="card3"> ...... </card>
</wml>

- WML pages have their own DTD
- A deck is the WML document
  - Enclosed in <wml> ... </wml>
- A card is the WML document
  - Enclosed in <wml> ... </wml>

Variables defined in a one card are accessible from all cards

Some other tags:
- <!--, <head>, <meta>, <template>, <br>, <p>, <table>, <td>, <tr>, <b>, <big>, <em>, <a>, <anchor>, <img>, <do>, <onevent>, <postfield>, <go>, <timer>, <input>, <optgroup>, <option>, <select>, <setvar>
- These are not all of the tags but as you can see many are the same as in HTML

WMLScript

- It is a lightweight version of JavaScript
- Runs client side
- Is compiled at a WAP gateway and sent to the device in byte code
- Not embedded in WML document
- Stored in a WMLS file (ie script.wmls)
- Includes Dialogs, Float, Lang, String, URL and WMLBrower function Libraries

Wireless Session Protocol

- Session level protocol
- It is the HTTP equivalent for WAP
- WSP is much more efficient than HTTP due to the bandwidth constraints of mobile devices
- Uses binary data vs text data
**Wireless Transaction Protocol**

- Provides both reliable and unreliable data transport
- Similar to TCP
- Built to minimize network traffic and be most effective over a wireless link

**Wireless Transaction Layer Security**

- Security layer protocol
- Similar to SSL (Secure Socket Layer)
- Provides authentication and encryption functionality
- Like SSL it is only used when needed

**Wireless Datagram Protocol**

- Similar to UDP (User Datagram Protocol)
- Lowest layer WAP protocol
- Lower layers are normally device dependent and some common interfaces are PPP (Point-to-Point Protocol), SMS (Short Messaging Service), and GPRS (General Packet Radio System)

**WAP and the rest of the web**

- Fewer that 1% of sites are written in WML
- Google has implemented a tool which converts regular HTML websites to WML for WAP browsers
- This allows WAP users access to the web content not specifically designed for mobile devices
**Classmark 2: JavaPhone**

Java Telephony API and its components

**What is JavaPhone?**

- It was created by the creators of Java, Sun Microsystems
- It is a Java like API for creating applications on cellular telephones
- Object Oriented

**What is it used for?**

- JavaPhone is used to develop applications for use on a variety of mobile devices
- It can also be used by telephone service providers to provide the connectivity to dynamic information services

**How does it work?**

- Much like regular Java, JavaPhone relies on an underlying virtual machine, or application environment which allows the telephone OS to seem transparent to the programmer.
- For JavaPhone this environment is either PersonalJava or EmbbededJava depending on the cellular hardware
Communications

- Additional packages are available for communications
- The Communication API, which allows for communication with serial and parallel ports
- The SSL package for use by applications to communicate securely communication over TCP/IP sockets

Summary

- JavaPhone is an API that can be used to develop applications for mobile devices in the same way that applications are developed for a computer.
- It allows for the development of programs to access the web as well as other internet protocols (FTP, SMTP, etc.)
- It also provides portability in an industry where many different hardware and OS implementations exist

Classmark 3: J2ME Environment

Java 2 Platform

MIDP – Mobile Information Device Profile
CLDC – Connected Limited Device Configuration
KVM – K Virtual Machine
Java 2 Micro Edition

- Java Virtual Machine ported onto a wide range of Operating Systems
- Growing processing power
  - More use of portable java
  - Less use of lower, more efficient languages
- J2EE JVM equivalent for the Growing processing power
  - More use of portable java
  - Less use of lower, more efficient languages
- J2ME is the K Virtual Machine (KVM)

J2ME Continued

Targeted for
- Small, Standalone Devices
- Connectable Consumer Devices
- Cellular phones
- Personal Digital Assistants

Two Primary Components
- Configurations
- Profiles

Configuration

Defines Minimum Java Platform for
- A range of devices
- Requirements for memory and power
- Minimum Java Libraries
- Virtual Machine Capabilities

CLDC
(Connected Limited Device Configuration)

Specifies Java Environment For
- Mobile Phones
- Personal Digital Organizers

Devices
- Limited Memory, 128 kb - 512 kb
- Intermittent, low bandwidth network link
- Constrained UI’s, small screens
- Most often wireless

K Virtual Machine (KVM)
- JVM designed for resource constrained devices
- CLDC runs on top of KVM
Profile

A collection of Java technology based APIs that add on to a Configuration to provide capabilities for a device type and extends its specific demands

MIDP

(Mobile Information Device Profile)

Specifies Java Environment For
- Mobile Phones
- Personal Digital Organizers

Addresses issues such as
- User Interface
- Store, Manage persistent local data
- Networking
- Application Model

Runs on top of the CLDC

Developed by a group of 20 companies representing the wireless industry

J2ME Wireless Device Stack

- Yellow Pages, Bus schedule, Tickets, Games
- UI, HTTP 1.1 network ...
- Java lang, Java lang.microedition...
- Threads, no Floats
- Risc processor
- Rom, Flash Ram

J2ME Devices
J2ME Software

- Active Desktop
  Access your home desktop from your mobile phone

- Web Viewer
  Supports HTML forms, image maps, GIFs, and JPEGs

- UEMail
  Mail client that communicates with any POP3/SMTP server

- Street Fighter
  The most famous arcade fighting game, with basic features

All this software is available as shareware at
http://j2me-software.com/repository/index.jsp

J2ME Development Tools

- They are available now
- Most are free
- They are all you need to create mobile Java applications
- References
  http://java.sun.com/j2me/
  http://java.sun.com/products/cldc/
  http://java.sun.com/products/midp/

J2ME Code

```java
import javax.microedition.midlet.*;
import javax.microedition.io.*;
import javax.microedition.lcdui.*;
import java.io.*;

public class HttpMIDlet extends MIDlet implements CommandListener {},
```
Testing Applications

- Available Virtual Mobile Device

Smart Cards

- tamper-resistant devices, usually containing an embedded 8-bit microprocessor
- utilize Application Protocol Data Units (APDU) to communicate with host devices through PIN codes and cryptographic keys
- Many different kinds including SIM, Java Card, UIM, WIM and S@T
- Used in cell phones, PDAs, satellite receivers, credit cards, etc.

SIM Cards

- Subscriber Identity Module
- Consist of microprocessor, ROM, EEPROM memory, volatile RAM, and serial I/O interface
- Software usually consists of an OS, file system, and application programs
- SIM cards rely on their GSM handset for battery and clock
**The SIM Toolkit (STK)**

- API used for securely loading applications onto SIM cards
- Allows mobile operator to create/provide services by loading them in the SIM without changing anything in the GSM handset. A convenient way of doing this is through Short Message Service (SMS).
- Once loaded, applications may be triggered through events such as incoming/outgoing calls, call duration, and/or location of the mobile device. Control software in the SIM monitors these events and reports them via SMS to a network based application server.

**WIM Cards**

- WAP Identity Module
- Used in conjunction with a SIM card
- Introduced with WAP spec 1.2 (Wireless Application Protocol)
- Provides end-to-end security for WAP applications, improving on the limitations of spec 1.1
- Allows the user to store certificates and digital signatures