Using Java to Internet-enable Embedded Systems

Prabal Dutta
NetEnabled.com
1275 Kinnear Road
Columbus OH 43212
TEL: (614) 675-3713
E-mail: prabal@NetEnabled.com

Presentation Overview

• The Market
• Benefits and Drawbacks of Web-Based Management
• Beyond the Embedded Web Server
• Implications for ISP’s
• Java
• Distributed Computing
• Java Everywhere?
The Market

- According to IDC, 20% of Internet access devices will be non-PCs by 2000 with 89 million non-PC devices connected to the web by 2001
- Wessels, Arnold & Henderson predict that the embedded Internet software industry will be a $5.6 billion market by 2001
- Automation Research believes that 30% of industrial automation products will be Internet-enabled by 2003
- According to IDC and Ariad, there will be 1 billion non-PC devices connected to the Internet by 2010
Benefits of Web-Based Management

• The user interface travels with the device
• Web browser is widely available and free
• Developing HTML is easier than custom GUI
• Simple and familiar user interface

Drawbacks of Web-Based Management

• The user interface is static
• HTTP performance is not suitable for near real-time control
• Typically requires a TCP/IP stack, Ethernet (or PPP), and a real-time operating system
• HTTP is a client/server protocol without support for asynchronous events
Beyond the Embedded Web Server

• An International Network Services survey reports:
  – 77% of respondents say Web-Based Management is important
  – 69% of respondents indicated that they were not satisfied

• What’s wrong?
  – NIH
  – Real-time device management
  – Application data is key, not EWS
  – Current protocols glorified
  – Legacy protocols ignored
  – C embedded in HTML and vice-versa
Beyond the Embedded Web Server

• How to fix it?
  – Abstract the device
  – Allow integration with legacy systems
  – Don’t mix presentation and representation code
The ISP Business Model

- Maximize subscribers (market share)
- Increase revenues by offering value-added services
- Lower costs (ISP support costs are significant; those who offer no support have prices that are discounted by 60%)

Using Java to Internet-enable Embedded Systems

The Old Subscribers

Using Java to Internet-enable Embedded Systems
The ISP Business Opportunity

- What if we could bring subscribers by the thousands?
- What if they signed Service Level Agreements?
- What if they never switched to a competing ISP?
- What if they never called to complain about configuration problems?
- What if they only called a few times per day, and only for a few minutes per call?

Does such a subscriber exist?!?

The New Subscribers

Using Java to Internet-enable Embedded Systems
Dial-Up Models

The Old Model

The New Model

Communications Protocols for Data Acquisition

Communications Protocols Used to Capture Measurement Data

Source: Keithley 1998 Measurement Needs Tracking Survey

Using Java to Internet-enable Embedded Systems
Where to put the Java?

- Devices without an Embedded JVM:
  - Home and Office Control Systems (Security, AC, Heat)
  - Industrial Automation and Process Control
  - Vending and other Remote Inventory Management
  - Retail and Industrial Measurement Instruments
  - Virtually any other Legacy System

- Devices with an Embedded JVM:
  - Palm Pilot w/ KVM
  - Dallas Semiconductor TINI
  - MC68HC11 with RealTimeJava
  - J2ME or E-Java Vendors: HP, Sun, Insignia, NewMonics

Connecting non-Java Devices to IP Networks

- Legacy Devices
- New Devices
Legacy Systems without JVM

- New Terminology
  - NetObject - a device or embedded system (conceptually similar to an object from object-oriented programming).
  - NetClient - a client interface that allows users to access, configure and manage NetObjects (can be practically any client platform or device).
  - NetRouter - the server software that provides routing services between two NetObjects or a NetObject and a NetClient.

Using Java to Internet-enable Embedded Systems
Legacy Systems without JVM

- 8- and 16-bit embedded systems
- TCP/IP too large for direct port
- Royalties too high for many sensitive price-points

Legacy Systems without JVM

- Make sure there is a well-defined communications interface (RS-232, RS-485, CAN, etc.)
- Create an object model of the device:
  - Fields
  - Methods
  - Events
- Generate a driver from this description
- Create any local support applications
- Extend using distributed object technology (RMI, CORBA, Jini)
Legacy Systems without JVM

- X10 home controller example
  - X10 commands are of the following form:
    - `<house code><unit code>`
    - `<house code><function code>`
  - House Codes range from A-P
  - Unit Codes range from 1-16
  - Function Codes include on, off, dim, bright, all off, lights on, etc.

Using Java to Internet-enable Embedded Systems
Legacy Systems without JVM

- Create driver (Java Communications API)
  - The javax.comm package is a Java 1.1. standard extension
  - Allows access to RS-232 serial ports
  - Allows access to IEEE-1284 parallel ports
  - Eventually may support USB and Firewire

Palm Pilot with KVM

- KVM uses approximately 64K of memory
- Class Libraries use another 64K of memory
- Packages
  - java.lang - Core Java classes for implementing the language
  - java.io - Supports system input and output through data streams and serialization
  - java.net - Provides classes for implementing networking applications
  - com.sun.kjava - Palm Pilot specific classes for Java One
Palm Pilot with KVM

• Package java.lang contains
  – Interfaces
    • Runnable
  – Classes
    • Class, Object, Runtime, String, StringBuffer, Thread, Throwable
  – Exceptions
    • Exception, IllegalAccessException, IllegalArgumentException, IndexOutOfBoundsException, NullPointerException, RuntimeException
  – Errors
    • Error
Palm Pilot with KVM

• Package java.net contains
  – Interfaces
    • none
  – Classes
    • Socket
  – Exceptions
    • SocketException
  – Errors
    • none

• Package com.sun.kjava contains
  – Interfaces
    • DialogOwner, ScrollOwner
  – Classes
    • Bitmap, Button, Caret, CheckBox, Database, Dialog, Graphics, HelpDisplay, IntVector, List, RadioButton, RadioGroup, ScrollTextBox, Slider, Spotlet, TextBox, TextField, Trigonometric, ValueSelector, VerticalScrollbar
  – Exceptions
    • SocketException
  – Errors
    • none
Palm Pilot with KVM

• Programming
  – Spotlet (not Applet) is the base class that provides callbacks for event handling
  – Event Handlers
    • penDown(int x, int y), penUp(int x, int y), penMove(int x, int y),
      keyDown(int key), beamReceive(byte[] data), beamSend(byte[] data),
  – Extend Spotlet
  – Override the specific event handlers
  – Use the Spotlet.register(int eventOptions) to gain focus

Using Java to Internet-enable Embedded Systems
Palm Pilot with KVM

• Sample Code continued

```java
public SimpleSpotlet()
{
    exitButton = new Button("Exit", 139, 145);
    testButton = new Button("Test", 100, 100);
    g.clearScreen();
    g.drawString("SimpleSpotlet", 5, 45, g.PLAIN);
    exitButton.paint();
    testButton.paint();
}

continued...
```

Using Java to Internet-enable Embedded Systems
Distributed Computing

- CORBA
- RMI
- Jini
- COM+
- RPC

Java Everywhere?

- Java 2 Enterprise Edition
  - Web, Application and Database Servers
  - Distributed Systems and Transactions
- Java 2 Standard Edition
  - Applets
  - Applications
- Java 2 Micro Edition
  - Rings
  - Chips
  - KVM
The Embedded Internet

NetEnabled.com

Prabal Dutta
NetEnabled.com
1275 Kinnear Road
Columbus OH 43212
TEL: (614) 675-3713