Designing Solutions for the Pervasive Computing Environments

Rahul Banerjee
CS & IS Group
Birla Institute of Technology & Science, Pilani, India
Email: rahul@bits-pilani.ac.in
Home: http://www.bits-pilani.ac.in/~rahul

A brief recap of concepts of relevance

- The *Pervasive Computing* brings paradigm shift in computing by making *computing elements embedded, distributed and designed in such ways in common objects around us that users do not have to be aware of computers in action.*
- Technologies for Pervasive / Ubiquitous Computing include :
  - *Display technologies,*
  - *Storage technologies*
  - *Low-Power technologies*
  - *Network Communication and Access technologies, and*
  - *Ubiquitous Software System technologies (Operating Systems and Application software included)*

Operating Systems for Pervasive Computing Environments

A brief recap of Operating System concepts of relevance

- Types of Operating Systems
  - Classification based on location of functionalities:
    - Centralised OSes
    - Networked OSes
    - Distributed OSes
  - Classification based on Kernel / Core Styling
    - Monolithic Kernel based OSes
    - Microkernel based OSes
    - Exokernel based OSes
  - Classification based on hardware form-factor and scope
    - Server-class OSes (with and without real-time support)
    - Workstation-class OSes (with and without real-time support)
    - Embedded OSes (with and without real-time support)

Identifying Requirements of Operating Systems for Pervasive Computing Systems

- Identify classes of applications to be run atop the target OS
- Estimate the exact set of corresponding functionalities to be supported at the lower levels
- Identify additional performance and security-specific constraints that may be required to be satisfied
- Identify the hardware architectures over which the solution is expected to be built

- Identify the availability of ready-to-use device drivers for the devices expected to be supported
- Weigh the effects of various trade-offs at the OS-level to affect the targeted class / classes of applications

Select Operating Systems used in pervasive computing environments:

- **Symbian OS**
- Variants of **Microsoft Windows**
  - Windows XP Embedded
  - WinCE.net
- Variants of **Linux**
  - ARM-Linux
  - BlueCat Embedded Linux
  - RT-Linux
- **PalmOS**
- **QNX Neutrino**
- Variants of **JavaOS**
  - JavaOS
  - Java for Business
- **BeOS**

Key features of Symbian OS v8.0

- Set of application engines for PIM, messaging, browsing; object exchanging (e,g. vCalendar & vCard);
- Integrated APIs for data management, text, clipboard and graphics
- Multithreaded kernel with real-time support
- Support for a range of CPU architectures, peripherals, memory types
- Support for a range of messaging systems including multimedia messaging (MMS), SMS; internet mail using POP3, IMAP4, SMTP etc. with attachments
- Support for audio / video recording / playback / streaming; image conversion; voice recognition etc.
- Key features of Symbian OS v8.0
- Support for protocols including TCP/IP (dual mode IPv4/IPv6) and WAP; infrared (IrDA), Bluetooth, USB;
- Support for multihoming capabilities and link layer Quality-of-Service (QoS) on GPRS/UMTS networks
- Support for 3G along with GSM circuit switched voice and data and packet-based data (GPRS and EGPRS); CDMA circuit switched voice, data and packet-based data (IS -95, cdma2000 1x, and WCDMA);
- Extensible telephony subsystem APIs
- Support for Internationalisation through the Unicode Standard version 3.0
- Over-the-air (OTA) data synchronization support using SyncML; PC-based synchronization over serial, Bluetooth, Infrared and USB; a PC Connectivity framework providing the ability to transfer files and synchronize PIM data
- Support for device management

- Support for security through full encryption and certificate management, secure protocols (HTTPS, SSL and TLS), WIM framework and certificate-based application installation
- Support for content development options for C++, Java (J2ME) WAP etc.;
- Support for variety of user inputs – generic input mechanism supporting full keyboard, 0-9*# (numeric mobile phone keypad), voice, handwriting recognition and predictive text input.

Generations of Wireless Communication Networking Standards

- First Generation Global Mobile Radio standard : 1G
  - Only voice, No data
- Second Generation Global Mobile Radio standard : 2G
  - GSM: 9.6 Kbps <circuit switched voice / data>
  - Enhanced Second Generation Global Mobile Radio standard : 2.5G
    - GSM-GPRS <combination of circuit and packet switched voice / data>
    - GPRS-136: <100Kbps <packet switched>
- Third Generation Global Mobile Radio standard: 3G
  - CDMA2000, =< 2Mbps <packet switched voice / data>
- Fourth Generation Global Mobile Radio standard : 4G (near future)
  - ? 20-40 Mbps <packet switched voice / data>

Inside GSM

Inside the GSM Network Subsystem

- MSC (Mobile Services Switching Center) acts like a normal switching node and provides the connection to the fixed networks (such as the PSTN or ISDN).
- HLR (Home Location Register) contains information of each subscriber registered in the corresponding GSM network, along with the current location of the mobile. There is logically one HLR per GSM network
- VLR (Visitor Location Register) contains selected information from the HLR, necessary for call control and provision of the subscribed services and each mobile currently located in the geographical area controlled by the VLR.
- EIR (The Equipment Identity Register) is a database that contains a list of all valid mobile equipment on the network,
- AuC (The Authentication Center) is a protected database:secret key of SIM

GSM and GPRS

- GSM uses TDMA/FDMA to share the limited radio spectrum wherein the FDMA part divides frequency of the not more than 25 MHz B/W into 124 carrier frequencies spaced 200 kHz apart.; and Each of these carrier frequencies is then divided in time, using a TDMA scheme.
- GSM is a circuit-switched digital network.

GSM and GPRS

- SGSN (the Serving GPRS Support Node) keeps track of the location of the mobile within its service area and send/receive packets from the mobile , passing them on, or receiving them from the GGSN

- GGSN (Gateway GPRS Support Node) converts the GSM packets into other packet protocols (e.g.IP / X.25) and sends them out into another network.

A bit more on GPRS and GSM

- GPRS users can share the resource (Radio link) which is used only when users are actually sending or receiving data.
- GPRS is based on GMSK which is a modulation technique known as Gaussian minimum-shift keying. It can support a theoretical upper limit of speed up to 171.2kbps as against the GSM 's 9.6Kbps.
- In GPRS, a channel that is 200kHz wide, is divided into 8 separate data streams, each carrying maximum 20kbps(14.4kbps typical) whereas in GSM we use only one channel.

A possible scenario

- Consider a situation wherein you have been asked to suggest a simple pervasive computing infrastructure design that could should support the following features:
  - Identification and location-tracking of all staff and students on campus
  - Textual / Voice / Multimedia Messaging between people as per need
  - Capability to use the high-end computing stations on campus for compute-intensive jobs
  - Basic security while allowing mobility
  - a. Choice of the default networking protocols / schemes at lower and higher layers
  - b. Mechanism for location management / awareness
  - d. Mechanism for route optimization for needs like simple textual data transfer, voice transfers, short text messages, MMS, large data packet transfers
  - e. Choice of the minimum hardware configuration with respect to
    1. processing features
    2. memory size & memory technology
    4. Secondary storage capacity & technology
    6. Choice for power system & Management
    7. Choice of frequency bands for transmission & reception
    8. Choice of modulation / demodulation and / or coding / decoding, if any
    9. User interface design features
  - f. Choice of Operating systems
  - g. Choice of Application Software

Your solution should address the following possible questions!

- What solution would you suggest that would be viable (technically and economically both) under these situations?
- How shall you evaluate your solution?
- What should you know before you begin?
- How to approach the problem for finding a possible solution?
- Can there be multiple acceptable solutions?

Possible solution approaches

- Top-down approach
- Bottom-up approach
- Hybrid approach
- Ad-hoc / one-off approach

Each approach may involve usual steps of analysis, streamlining of specifications, identification of constraints, issues and choices; and finally, validation through formal or informal methods.

Processor technologies

- Intel's SpeedStep processor technology
- Transmeta's Crusoe processor technology
- Motorola's Dragon Ball processor technology
- Intel's StrongARM processor technology
- Intel's X-Scale processor technology

Interfacing technologies

- Navigation technologies
- Haptic interfacing technologies
- On-screen / Touch-panel technologies
- Voice interfacing technologies
- Video-interfacing technologies
- Handwriting-based interfacing technologies
- Hybrid interfacing technologies

What should the Infrastructure provide?

- Pervasive Computing Infrastructure has to comprise of computing elements, communicating elements, sensors, actuators, and interface devices.
- Computation to be available widely and freely (not free of cost).
- Intermittent connectivity has to be a supported feature due to physical limitations pertaining to power, cost, bandwidth and network congestion.
- Bluetooth and other choices address small-distance networking issues and allow intermittent connection.
- The infrastructure has to offer seamless connectivity to the devices / entities / services.
- It has to support placement and location of uniquely identifiable "information tags / trackable tags" to all devices / entities in the Pervasive Computing environment.
- User's environment must be able to be aware of the user's context.