Outline

- Mobile phone operating systems:
  - Symbian OS
  - Google Android
  - BlackBerry
  - iOS
  - Windows Phone
Mobile Phone Subscribers

Mobile phone subscribers per 100 inhabitants 1997-2007
Operating System Requirements in Mobile Phones

- Low memory footprint
- Low dynamic memory usage
- Efficient power management
- Real-time support for communication and telephony protocols
- Device and data integrity
### Market share by OS

<table>
<thead>
<tr>
<th>Year</th>
<th>Symbian</th>
<th>Android</th>
<th>RIM</th>
<th>iOS</th>
<th>Microsoft</th>
<th>Other OSs</th>
</tr>
</thead>
<tbody>
<tr>
<td>2010</td>
<td>37.6%</td>
<td>22.7%</td>
<td>16.0%</td>
<td>15.7%</td>
<td>4.2%</td>
<td>3.8%</td>
</tr>
<tr>
<td>2009</td>
<td>46.9%</td>
<td>3.9%</td>
<td>19.9%</td>
<td>14.4%</td>
<td>8.7%</td>
<td>6.1%</td>
</tr>
<tr>
<td>2008</td>
<td>52.4%</td>
<td>0.5%</td>
<td>16.6%</td>
<td>8.2%</td>
<td>11.8%</td>
<td>10.5%</td>
</tr>
<tr>
<td>2007</td>
<td>63.5%</td>
<td>N/A</td>
<td>9.6%</td>
<td>2.7%</td>
<td>12.0%</td>
<td>12.1%</td>
</tr>
</tbody>
</table>

Source: Gartner
Symbian OS

- 32-bit operating system
- Ported to many flavors of ARM architecture
- Architecture requirements
  - Integrated memory management unit (MMU) and a cache, to operate in various privileged access modes
  - Interrupts handling and exceptions
Symbian OS System Services (1/2)
Symbian OS System Services (2/2)

- Networking (TCP/IP, PPP, TSL, SSL, IPSec, FTP)
- Communications (Bluetooth, IrDA, Obex)
- Security (DES, RSA, DSA, DH)
- Messaging (POP3, IMAP4, SMTP, SMS, BIO)
- Browsing (HTML, HTTPS, WAP, WML)
- Telephony (GSM, GPRS, fax)
- Graphics, multimedia (WAV, AU, WVE, JPEG, BMP, MBM, GIF)

- Access to services through a cliente-server model
- Servers run as unprivileged threads
- Framework keeps a record of objects in the system

Symbian OS top layer

- Application engines
  - Contacts
  - Agenda
  - Jotter (a document producing application)
  - Skeleton support for graphical user interface
Google Android Overview

- Software stack for mobile devices
  - Mobile operating system based on a modified Linux kernel (2.6)
  - Middleware
  - Key applications

Date
August 2005  Google acquires Android Inc.
2006  Rumors of Google handset
November 2007  Open Handset Alliance is formed
October 2008  Android goes open-source
October 22, 2008  HTC Dream released
October 2009  Android 2.0 released
May 2010  Android 2.2 released
December 2010  Android 2.3 released
February 2011  Android 3.0 released for tablets
Android Features

- Application framework enabling reuse and replacement of components
- Dalvik virtual machine optimized for mobile devices
- Integrated browser based on the open source WebKit engine
- Optimized graphics powered by a custom 2D graphics library; 3D graphics based on the OpenGL ES 1.0 specification
- SQLite for structured data storage
- Media support for common audio, video, and still image formats (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF)
- GSM Telephony
- Bluetooth, EDGE, 3G, and WiFi
- Camera, GPS, compass, and accelerometer
- Rich development environment including a device emulator, tools for debugging, memory and performance profiling, and a plugin for the Eclipse IDE

Android Architecture (1/3)
Applications

- Core set: email client, SMS program, calendar, maps, browser, contacts, and others
- Written using the Java programming language

Application Framework

- Same framework APIs used by the core applications
- Designed to simplify the reuse of components

Libraries

- Set of C/C++ libraries exposed through the Android application framework

Android Runtime

- Provides most of the functionality of the Java core libraries
- Every Android application runs in its own process with its own instance of the Dalvik VM
- A device can run multiple VMs efficiently
- VM is register-based, Java classes are transformed into .dex format using “dx” tool
- Dalvik VM relies on Linux kernel for underlying functionality such as threading and low-level memory management

Linux Kernel

- Core system services such as security, memory management, process management, network stack, and driver model
- Abstraction layer between the hardware and the rest of the software stack
BlackBerry Overview

- Smartphone devices developed by Research in Motion (RIM)
- Very popular in the corporate world
- Major selling point: instant, secure, mobile access to email.
- Started as a two way pager
BlackBerry OS Overview

- Proprietary mobile operating system
- Developed by Research In Motion for its BlackBerry line of smartphone handheld devices
- Provides multitasking
- Designed for use of input devices such as the track wheel, track ball, and track pad
- Provides support for Java MIDP 1.0 and WAP 1.2.
- Allows synchronization with almost everything
- Updates to the operating system may be automatically available from wireless carriers that support the BlackBerry over the air software loading (OTASL) service
- Third-party developers can write software using the available BlackBerry API classes, although applications that make use of certain functionality must be digitally signed.

BlackBerry OS Architecture
iOS Overview

- iPhone is released on June 29, 2007
- It’s operating system was iPhone OS, later renamed iOS
- iOS runs on Apple Inc.’s mobile devices such as the iPhone, iPod Touch, and iPad
- Updates are released through iTunes

<table>
<thead>
<tr>
<th>Version</th>
<th>Device</th>
<th>Date</th>
<th>Obs.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>iPhone</td>
<td>June 29, 2007</td>
<td></td>
</tr>
<tr>
<td>1.1</td>
<td>iPod Touch</td>
<td>September 14, 2007</td>
<td></td>
</tr>
<tr>
<td>2.0</td>
<td>iPhone 3G</td>
<td>July 11, 2008</td>
<td>App Store</td>
</tr>
<tr>
<td>3.0</td>
<td>iPhone 3GS</td>
<td>June 17, 2009</td>
<td>Copy and paste, MMS</td>
</tr>
<tr>
<td>3.2</td>
<td>iPad only</td>
<td>April 3, 2010</td>
<td></td>
</tr>
<tr>
<td>4.0</td>
<td>iPhone 4</td>
<td>June 21, 2010</td>
<td>Multitasking</td>
</tr>
<tr>
<td>4.3</td>
<td>iPad 2</td>
<td>March 9, 2011</td>
<td></td>
</tr>
</tbody>
</table>
## Cocoa Touch Layer

- Sits at the top of the iPhone OS stack
- Contains the frameworks that are most commonly used by iPhone application developers.
- Primarily written in Objective-C
- Is based on the standard Mac OS X Cocoa API
- Has been extended and modified to meet the needs of the iPhone.
- Provides the following frameworks for iPhone app development:
  - UIKit Framework (UIKit.framework)
  - Map Kit Framework (MapKit.framework)
  - Push Notification Service
  - Message UI Framework (MessageUI.framework)
  - Address Book UI Framework (AddressUI.framework)
  - Game Kit Framework (GameKit.framework)
Media Layer

- Provides audio, video, animation and graphics capabilities
- Provides the following frameworks:
  - Core Graphics Framework (CoreGraphics.framework)
  - Quartz Core Framework (QuartzCore.framework)
  - OpenGL ES framework (OpenGLES.framework)
  - iPhone Audio Support
  - AV Foundation framework (AVFoundation.framework)
  - Core Audio Frameworks (CoreAudio.framework, AudioToolbox.framework and AudioUnit.framework)
  - Open Audio Library (OpenAL)
  - Media Player framework (MediaPlayer.framework)

Core Services Layer

- Is the foundation to the above layers
- Consists of the following frameworks:
  - Address Book framework (AddressBook.framework)
  - Core Data Framework (CoreData.framework)
  - Core Foundation Framework (CoreFoundation.framework)
  - Foundation Framework (Foundation.framework)
  - Core Location Framework (CoreLocation.framework)
  - Store Kit Framework (StoreKit.framework)
  - SQLite library
Core OS Layer

- **Bottom layer of the iPhone OS stack**
- **Sits directly on top of the device hardware**
- **Provides a variety of services:**
  - CFNetwork Framework (CFNetwork.framework)
  - External Accessory framework (ExternalAccessory.framework)
  - Security Framework (Security.framework)
  - System (LibSystem) (memory management, file system handling and threads)

Applications development

- iPhone apps are developed using the iPhone SDK in conjunction with Apple’s Xcode development environment
- Xcode is an integrated development environment within which you will code, compile, test and debug your iPhone applications
- Xcode enables you to graphically design the user interface of your application
Windows Phone 7 Overview

- Windows Phone 7 is a mobile operating system developed by Microsoft
- Is the successor to its Windows Mobile platform
- Launched in Europe, Singapore, Australia and New Zealand on October 21, 2010, and in the US and Canada on November 8, 2010, Mexico on November 24, 2010, with Asia to follow in 2011
- Windows Phone become the primary smartphone operating system for Nokia
Device Requirements

- Capacitive, 4-point multi-touch screen with WVGA (480x800) resolution
- ARM v7 “Cortex/Scorpion” – Snapdragon QSD8X50, MSM7X30, and MSM8X55
- DirectX9 rendering-capable GPU
- 256 MB of RAM with at least 8 GB of Flash memory
- Accelerometer with compass, ambient light sensor, proximity sensor, Assisted GPS, and Gyroscope
- 5-megapixel camera with an LED flash
- FM radio tuner
- 6 dedicated hardware buttons – back, Start, search, 2-stage camera, power/sleep and Volume Up and Down

Windows Phone Architecture
**Windows Mobile Development**

- Applications developed in C# or Visual Basic

Two applications development platforms

- Silverlight
  - application framework for writing and running browser plug-ins or other rich Internet applications
  - features and purposes similar to those of Adobe Flash.

- XNA
  - based on the native implementation of .NET Compact Framework 2.0 for Xbox 360 development and .NET Framework 2.0 on Windows
  - specific to game development
Outline

- Mobile phone operating systems:
  - Symbian OS
  - Google Android
  - BlackBerry
  - iOS
  - Windows Phone

Next Class

- Sensor networks