

CHAPTER 01

# MOBILE DEVICES

## 1. Introduction:

The Mobile phone has become a necessity in our lives; it is always with us, it is always on. It has become so much more than just a phone; it has become an extension of who we are. As the mobile phone development continues its astonishing progress, we are now experiencing a fusion of technology into a mobile device, not only being able to make phone calls, but also including radio, music player, megapixel camera, camcorder and internet connection to name a few properties.

In this chapter we will give a short overview of today's Smartphones their constraints and their limitations.

## 2. Characteristics of mobile phones :

Mobile phones can be basically distinguished into two categories: *feature phones* and *smartphones*.

A feature phone represents the standard cellular phone with all of its functionality, e.g. calendar, messaging or often support for Java Platform Micro Edition (Java ME) software.

A Smartphone whereas mostly unifies functionalities of a cellphone, a PDA, an audio player, a digital camera, a GPS receiver and a PC. It often uses PC-like *QWERTY* keyboards in order to increase typing speed and sometimes PDA-like pen displays for improved data and command handling. One significant characteristic is the ability of installing proprietary third party application. This presumes the existence of a publicly available Software Developing Kit (SDK). Smartphones use various techniques for creating wireless connections, which represents their most essential purpose:

- GSM is used for voice calls and services like SMS.
- GPRS provides voice and packet data communication.
- W-CDMA (UMTS) is able to transport data at higher speed than GSM.

Additionally, the devices provide Bluetooth (BT), Wireless LAN (WLAN) or IrD support for shorter range wireless connectivity. Using one of these connections, a user is able to make phone calls, use an internet browser, play multi-player games, or read emails.

Furthermore, the smartphone can be seen as the first platform for *pervasive computing* [1], where interesting areas of application are pointed out by Roussos et al. [2]:

- Information service endpoint, e.g. applied as navigational assistance or location based services.
- Remote controllers for devices like television or HiFi station.
- Pervasive network hubs to provide wide area connectivity, e.g. for wearable systems that need to communicate in order to transmit health related data.
- ID tokens in order to store information used to provide accountability.

### 3. Operating Systems for Smartphones :

A mobile operating system (mobile OS) is the operating system that controls a smartphone, tablet, PDA, or other mobile device. Modern mobile operating systems combine the features of a personal computer operating system with touchscreen, cellular, Bluetooth, Wi-Fi, GPS mobile navigation, camera, video camera, speech recognition, voice recorder, music player, Near field communication, personal digital assistant (PDA), and other features.

#### 3.1. BlackBerry OS from RIM :

Research In Motion (RIM) provides proprietary operating systems for its own Black-Berry devices. The target customers are business managers as the included techniques and services are intended to support their daily needs [3]. Although RIM only addresses a small customer group, they have got a recognizable share with their OS in the smart mobile device market.

Since only proprietary and restricted Java application are allowed on BlackBerry devices at the moment, only few official documents can be found about palm operating system or application security. Most security information addresses certain parts relating to the BlackBerry PUSH technology, which represents a mechanism for instantly bringing email messages to mobile devices.

### 3.2. Windows Phone from Microsoft :

The Windows Mobile operating system is based on Windows CE and was developed for mobile devices like Pocket PCs, PDAs, smartphones, and embedded systems, e.g. smart fridges [5]. In 2003 there were three different version of Windows Mobile: Windows Mobile 2003 for Pocket PC, Windows Mobile 2003 for Pocket PC Phone Edition and Windows Mobile 2003 for Smartphones. The version Windows Mobile 7.0, called “Magneto”, was released in the year 2005. It uses the .NET Compact Framework 1.0 SP2 to enable .Net applications and bases on Windows CE 5.0. Windows Mobile Supports WIFI, GPS, PC (Windows) synchronization.

Windows Mobile security addresses three major approaches: *security roles*, *security policies*, and *application signing*. Security roles define users or groups having predefined rights on a device. The most privileged role allows changing security policies, which are rules permitting certain actions on the device, e.g. installing and running unsigned applications<sup>[5]</sup>. Basically, Windows Mobile software is signed in order to permit access to sensitive APIs.

### 3.3. Symbian :

Symbian Limited is a software producing company located in London, UK. It is owned by Ericsson (15.6%), Nokia (47.9%), Panasonic (10.5%), Samsung (4.5%), Siemens (8.4%) and Sony Ericsson (13.1%). Symbian core business is developing and licensing Symbian OS, an operating system for mobile devices which has evolved from former Psion’s Epoc. The Symbian OS licensees represent over three quarters of mobile phone shipments globally [4]. Smartphone manufacturers that license Symbian OS are Arima, Ben Q, Fujitsu, Lenovo, LG Electronics, Motorola, Mitsubishi, Nokia, Panasonic, Samsung, Sharp and Sony Ericsson.

Symbian OS introduces three security concepts, which are *capabilities*, *installation file signing*, and *data-caging*. Capabilities limit access to sensitive APIs. There are basically three levels of limitation where on the highest level full device and network access is granted to the corresponding application<sup>[3]</sup>. These limitation levels are defined by certificates that are used to sign Symbian OS Installation System (.SIS) files. Without a valid signing, it is not possible to install application on Symbian OS devices. Data caging extends this approach as it limits access to the file system. Depending on the limitation coming from the

certificate, application can only write to certain areas, like the application folder, user data folder, or system folder<sup>[4]</sup>.

### **3.4. Android from Google :**

Google Android is a package of software that includes an operating system, middleware and basic applications. The Android system is built upon the Linux 2.6 Kernel and supports most of original provided functionalities. Android treats every application equal, which means on the one hand that a developer is able to replace every single android program. On the other hand it means that if you develop an android application it can be run on any Android device, only being limited by the provided functionalities.

The security mechanisms that can be found in Google Android basically are based on the same mechanisms that can be found in the Linux system. Especially access control, e.g. user and group IDs, is managed where every installed application gets its own user ID with its specific permissions. These permissions allow finer-grained access adjustment for processes using certain functionalities, e.g. sending SMS messages or dialing a phone call.

### **3.5. iOS from Apple :**

The Apple iPhone<sup>[1]</sup> is a very interesting device that can be classified as smartphone since Steve Jobs from Apple announced a SDK for it<sup>[2]</sup>. The device runs a modified version of Mac OS X and includes several applications, e.g. the Safari browser, a music player, and digital camera. As long as the SDK is not officially released, only little can be said about security approaches used in this version of the OS. It is imaginable that they are similar to the ones of the original version with a focus on mobility related measures.

## **4. Software Development for Smartphones :**

Developing, building, and testing smartphone software requires tools which are often included in a SDK or integrated development environment (IDE). These software packages will be briefly introduced in this section.

## 4.1. The SDK :

A Software Developing Kit (SDK) is a collection of software that gives a software developer the ability to create and deploy software for a certain framework, platform, operating system, programming language or hardware. Most SDKs can be downloaded for free from the internet. Current examples are the *Symbian S60 3rd SDK* <sup>[6]</sup>, the *Windows Mobile 5.0 SDK* <sup>[7]</sup>, or the *SUNWireless Toolkit (WTK)* <sup>[8]</sup>. Most SDKs are delivered with a software emulator.

## 4.2. The (Software-) Emulator :

A software emulator gives a developer the ability to run and test software on his computer though is developed for other systems, e.g. Symbian OS devices. This can reduce costs, as prototypes can already be programmed and run without buying a real device. On the other hand, the emulator often does not support all functionalities of a real device. This can lead to serious problems. Unlike a simulator, which reproduces programs behavior, an emulation attempts to reproduce the same states the original devices would enter at several points, but regarding current SDKs, the so called software emulators *simulate* connections, interface and functionality through mapping e.g. simulator Bluetooth port to PCs serial port.

Comparable software is available with the VMware products <sup>[9]</sup>. It enables parallel installation and simultaneous running of different operating systems like Windows or Linux.

## 4.3. The IDE :

The integrated development environment is very similar to SDKs, it often combines most of the tools to be able to write, compile, build and debug software. The main difference is, that today's IDEs integrate all tools into one single tool that has Graphical User Interface (GUI) especially developed for certain platforms. All needed actions then can be done through the GUI, which often speeds up development. Examples for IDEs for mobile devices are MS Visual Studio, Metroworks Codewarrior, Nokia Carbide, and Eclipse.

## **5. Conclusion :**

The mobile phones are getting more feature rich, the phone users want to share content on their mobile phones with their friends, and to acquire multimedia content available on the other phones in the wireless network.

This content is mostly self produced (e.g. pictures and videos) but maybe also professionally created (e.g. movie trailers, ring tones, and application programs).

In the next chapter we present the Peer-To-Peer technology that allows people to share their documents with their friends and other users conveniently.