

ALMA MATER STUDIORUM · UNIVERSITÀ DI
BOLOGNA

FACOLTÀ DI SCIENZE MATEMATICHE, FISICHE E NATURALI
Corso di Laurea Triennale in Informatica per il Management

**ANALISI E COMPARAZIONE DEI
SISTEMI OPERATIVI PER
DISPOSITIVI MOBILI**

Tesi di Laurea in Sistemi Operativi

Relatore:
Chiar.mo Prof.
DAVIDE SANGIORGI

Presentata da:
LUCA CAVINA

Sessione II
Anno accademico 2010 - 2011

*Questa tesi è dedicata ai miei nonni,
ai miei genitori, a mia sorella Erika
e alla mia fidanzata Margherita.*

Indice

Introduzione	i
1 Evoluzione dei dispositivi mobili	1
1.1 Dispositivi cellulari	1
1.1.1 Origini	1
1.1.2 Caratteristiche	3
1.1.3 Sistemi operativi	4
1.2 Dispositivi palmari	4
1.2.1 Origini	5
1.2.2 Caratteristiche	6
1.2.3 Sistemi operativi	7
1.3 Dispositivi Smartphone	9
1.3.1 Origini	10
1.3.2 Caratteristiche	11
1.3.3 Sistemi operativi	13
2 Sistemi operativi dei dispositivi smartphone	15
2.1 Android	15
2.1.1 Versioni	15
2.1.2 Caratteristiche tecniche	19
2.1.3 Applicazioni	21
2.1.4 Ambiente di sviluppo	23
2.2 iOS	26
2.2.1 Versioni	26

2.2.2	Caratteristiche tecniche	30
2.2.3	Applicazioni	33
2.2.4	Ambiente di sviluppo	33
2.3	Symbian OS	36
2.3.1	Versioni	36
2.3.2	Caratteristiche tecniche	37
2.3.3	Applicazioni	42
2.3.4	Ambiente di sviluppo	43
2.4	Windows Mobile	44
2.4.1	Versioni	45
2.4.2	Caratteristiche tecniche	47
2.4.3	Applicazioni	52
2.4.4	Ambiente di sviluppo	53
2.5	BlackBerry OS	54
2.5.1	Versioni	54
2.5.2	Caratteristiche tecniche	57
2.5.3	Applicazioni	60
2.5.4	Ambiente di sviluppo	60
3	Differenze tecniche tra i sistemi operativi	63
3.1	Facilità d'uso/Interfaccia utente	63
3.2	Caratteristiche	65
3.2.1	Multimedia	66
3.2.2	Navigazione Satellitare	68
3.2.3	Aggiornamenti	68
3.3	Performance	69
3.3.1	Multitasking	70
3.4	Applicazioni	72
3.5	Telefoni supportati e Hardware	74
3.6	Analisi delle differenze	77

4	Gestione della memoria	79
4.1	Gestione memoria Android	79
4.1.1	Gestione delle attività	79
4.1.2	Garbage Collector e gestione dei processi	82
4.1.3	Come usare la memoria in modo efficiente	85
4.1.4	Problemi di allocazione memoria	85
4.2	Gestione memoria iOS	87
4.2.1	Regole di base Gestione della memoria	88
4.2.2	Autorelase Pool	89
4.2.3	MRR (manual retain-relase)	91
4.2.4	Automatic Reference Counting (ARC)	93
4.2.5	Problemi relativi alla memoria	93
5	Analisi di mercato	95
5.1	Mercato mondiale	95
5.2	Mercato USA	99
5.3	Mercato europeo	102
5.4	Mercato italiano	104
5.5	Il mercato smartphone rispetto agli altri settori	108
	Conclusioni	111
	Bibliografia	119

Elenco delle figure

2.1	Architettura del sistema operativo Android	20
2.2	Architettura del sistema operativo Apple iOS	31
2.3	Architettura del sistema operativo Symbian	41
2.4	Architettura del sistema operativo Windows CE 6.0	48
4.1	Ciclo di vita di un'attività	80
4.2	Stati delle applicazioni	83
4.3	Esempio sul funzionamento dell'autorelase pool	90
4.4	Un tipico ciclo di vita di una variabile, quando il retaincount raggiunge il valore 0 viene automaticamente rilasciata	92
5.1	Situazione del mercato mondiale dei dati del secondo quadrimestre del 2011 rispetto allo stesso periodo nel 2010	96
5.2	Previsione del mercato mondiale dei sistemi operativi smart- phone fino al 2015	97
5.3	Previsione del mercato mondiale dei SO smartphone fino al 2015	98
5.4	Il SO desiderato in base alla tipologia di consumatore	99
5.5	Dati comScore aggiornati a luglio 2011	101
5.6	Dati comScore aggiornati ad agosto 2011	101
5.7	Dati comScore aggiornati a luglio 2011	102
5.8	Dati Kantar divisi per paesi aggiornati a luglio 2011	103
5.9	Il trend di crescita dell'Internet Mobile in Italia	105
5.10	La crescita degli smartphone in Italia e la previsione dell'im- minente sorpasso	105

5.11	La soddisfazione verso l'Application store	107
5.12	Analisi dei guadagni Samsung nel Q2 2011	109
5.13	Analisi dei guadagni Apple nel Q4 2011	110

Introduzione

I dispositivi mobili di ultima generazione, chiamati smartphone, stanno diventando i nuovi personal computer, il computer desktop non sta scomparendo, ma il mercato degli smartphone è in continua e rapida crescita. I dispositivi vengono utilizzati abitualmente come computer da sempre più persone e per sempre più scopi; sono generalmente più economici rispetto ai computer tradizionali, sono più convenienti per la loro portabilità, e spesso sono inoltre più utili con il contesto fornito dalla geolocalizzazione.

Una parte fondamentale di questa rapida evoluzione è da attribuire ai loro sistemi operativi. Questo elaborato ha lo scopo di documentare, l'analisi e la comparazione dei principali sistemi operativi per dispositivi mobili, concentrandosi su quelli per smartphone.

Verranno inizialmente introdotti i dispositivi mobili, parlando della loro evoluzione, dai dispositivi cellulari, passando per i palmari e per finire con gli smartphone; oltre ai passaggi storici verrà descritta l'evoluzione sia dal punto di vista delle caratteristiche che da quello dei sistemi operativi. Ci soffermeremo nel mondo degli smartphone dove verranno analizzati nel dettaglio i 5 principali sistemi operativi disponibili in commercio (Android, Apple iOS, BlackBerry OS, Symbian OS e Windows Mobile), nello specifico si parlerà delle ultime versioni rilasciate, descrivendo le novità e le nuove tecnologie aggiunte, la struttura e le caratteristiche tecniche adottate nel sistema operativo, le applicazioni supportate, l'ambiente di sviluppo per crearle e i linguaggi di programmazione supportati.

Successivamente, in luce all'analisi fatta precedentemente si illustrano le

differenze sostanziali tra i vari sistemi operativi, in termini di interfaccia utente, di caratteristiche come la gestione dei file multimediali o la navigazione satellitare, di performance entrando nello specifico parlando della gestione del multitasking, delle applicazioni soffermandosi anche sui loro store, e dell'hardware supportato. Verrà fatta l'analisi di tutte queste differenze cercando di capire, sia dal punto di vista del consumatore che da quello dello sviluppatore, quali possono essere tra questi i punti chiave che hanno fatto diventare prima iOS poi Android i leader indiscussi nel settore dei sistemi operativi mobili.

Arrivati a questo punto si entra ancora di più nel dettaglio e verranno mostrate le differenti gestioni della memoria da parte di Android e di iOS, per il sistema operativo open source si descrive il metodo utilizzato per gestire la memoria e le priorità che hanno i vari tipi di applicazioni e gli stati di processo che determinerà l'ordine in cui verranno uccisi i processi per recuperare le risorse necessarie; per il sistema di Apple si mostrano i tre metodi di gestione della memoria di un'applicazione forniti da Objective-C.

Infine nell'ultimo capitolo si analizzerà la situazione di mercato nel settore degli smartphone, basandosi sui dati delle società di analisi e ricerca come comScore, Nielsen e altri si descrive la situazione nel mercato mondiale, americano, europeo e infine in quello italiano; cercando di fare previsioni sugli andamenti strettamente futuri. Il capitolo si chiude con un ultimo paragrafo dove si parla dello sviluppo del mercato smartphone rispetto agli altri settori all'interno di un'azienda, prendendo come esempio gli ultimi dati di vendita Apple e Samsung.

Cenni teorici

Prima di cominciare l'analisi è necessario citare alcuni cenni teorici sul mondo dei sistemi operativi che troveremo in questo elaborato in modo da rendere più chiara la lettura.

Il **kernel** costituisce il nucleo di un sistema operativo; generalmente è un software avente il compito di controllare e gestire l'accesso all'hardware dei vari processi in esecuzione nel sistema operativo, assegnando ad ogni processo lo spazio in memoria, stabilendone un ordine di priorità, caricandone il codice in memoria ed eseguendolo. In particolare un **microkernel** offre un insieme ristretto di funzionalità, tra cui una gestione degli indirizzi a basso livello, l'amministrazione dei vari thread in esecuzione e un IPC (Inter-Process Communication) per la comunicazione fra i vari processi; si serve eventualmente di altre applicazioni per adempire ad altre richieste. I vantaggi del microkernel sono la portabilità, l'estensione, la facilità di modifica e l'affidabilità. Dato che possono esserne eseguiti simultaneamente più di un processo, il kernel ha anche la responsabilità di assegnare una porzione di tempo-macchina e di accesso all'hardware a ciascun programma (multitasking).

Possiamo quindi dire che il **multitasking (multiprocessualità)** è la funzionalità attraverso la quale un sistema operativo gestisce più processi contemporaneamente nello stesso processore. Esistono due principali tecniche di controllo di termine e pausa del multitasking:

- Senza sospensione dell'esecuzione (**cooperative**) - In questo caso i programmi cedono volontariamente il controllo al sistema operativo una volta terminata l'operazione in corso oppure passando allo stato di attesa. Questo metodo non ha bisogno di supporto hardware e si può implementare su ogni tipo di architettura, ma un singolo programma che si rifiuta di cedere il controllo, o che si ferma per qualche errore,

può bloccare l'intero computer in quanto il sistema operativo non ha modo di riprendere il controllo da solo .

- Con sospensione dell'esecuzione (**preemptive**) - In questo caso si interrompe un processo a prescindere dalla volontà del processo stesso, ciò avviene grazie a delle particolari strutture hardware integrate nel microprocessore (come il temporizzatore) che automatizzano il cambio di contesto (context switch): in questo lo scheduler interviene oltre che nei due casi visti per il multitasking cooperative, anche quando il processo passa dallo stato di esecuzione allo stato di pronto per essere eseguito; oppure quando passa dallo stato di attesa allo stato di pronto per essere eseguito. Un multitasking con prelazione (preemptive) non può quindi essere implementato se la piattaforma hardware non mette a disposizione gli strumenti necessari, ma in compenso, il cambio di contesto è molto più efficiente favorendo l'adozione di quanti di tempo regolari e una esecuzione più fluida dei vari processi. Il preemptive multitasking è stato adottato dalla maggior parte dei sistemi operativi moderni.

Per **multithreading** si intende invece un ulteriore livello di astrazione nella quale si suddivide un processo per ridurlo in elementi (thread) più facili da gestire, da eseguire comunque nello stesso tempo. Questa tecnica si distingue da quella alla base dei sistemi multiprocessore per il fatto che i singoli thread condividono lo stesso spazio d'indirizzamento, la stessa cache e lo stesso translation lookaside buffer. Il multithreading migliora le prestazioni dei programmi solamente quando questi sono stati sviluppati suddividendo il carico di lavoro su più thread che possono essere eseguiti in parallelo. Un sistema multithread è dotato di una singola unità di calcolo che si cerca di utilizzare al meglio eseguendo più thread nella stessa unità di calcolo.

Un **sistema operativo real-time** (abbreviato in RTOS) è un sistema operativo specializzato per il supporto di applicazioni software real-time, capace di accettare vincoli temporali. Questi sistemi vengono utilizzati tipicamente in

ambito industriale o comunque dove sia necessario ottenere una risposta dal sistema entro un tempo prefissato. Un sistema operativo real-time non deve essere necessariamente veloce: non è importante l'intervallo di tempo in cui il sistema operativo/applicativo deve reagire; l'importante è che risponda entro un tempo massimo pre-determinato. In altre parole il sistema deve essere prevedibile. In pratica un sistema real-time deve garantire che una elaborazione (o task) termini entro un dato vincolo temporale o scadenza (detta in gergo deadline). Per garantire questo è richiesto che la schedulazione delle operazioni sia fattibile. Il concetto di fattibilità di schedulazione è alla base della teoria dei sistemi real-time ed è quello che ci permette di dire se un insieme di task sia eseguibile o meno in funzione dei vincoli temporali dati. I task di un sistema real-time possono però essere:

- **Soft real-time** (vincolo debole) nella quale un task che non rispetta la sua scadenza provoca un danno non irreparabile al sistema. Il superamento della deadline produce un degrado delle prestazioni proporzionale al tempo di superamento.
- **Hard real-time** (vincolo forte) dove un task che nel caso superi temporalmente la sua deadline provoca un danno irreparabile al sistema.

Capitolo 1

Evoluzione dei dispositivi mobili

Un **sistema operativo per dispositivi mobili**, in inglese mobile OS, è un sistema operativo che controlla un dispositivo mobile con lo stesso principio con cui Mac OS, Linux o Windows controllano un desktop computer oppure un laptop. Tuttavia affronta problematiche legate alla natura del dispositivo mobile, più critiche rispetto ad un desktop o un laptop, per esempio: la limitatezza delle risorse (memoria,CPU), l'assenza di alimentazione esterna, differenti protocolli di trasferimento dati per l'accesso a internet (WiFi, GPRS, HSDPA...), nuovi metodi d'immissione (touchscreen, minitastiere) e le ridotte dimensioni del display.

I dispositivi mobili principali in ordine cronologico sono: i telefoni cellulari, i palmari e i più recenti smarthpone (unione delle funzioni dei due precedenti dispositivi) e tablet.

1.1 Dispositivi cellulari

1.1.1 Origini

Il primo cellulare è ufficialmente arrivato nel 1985, anche se nel 1973 l'ingegnere americano Martin Cooper, facente parte della Motorola, ha ef-

fettuato la prima chiamata dal cellulare utilizzando il prototipo Dyna-Tac 8000X, un dispositivo pesante 1130 grammi, senza display e senza ulteriori funzioni, se non quelle di parlare, ascoltare e comporre il numero. La batteria di questo prototipo aveva appena 35 minuti di autonomia e per ricaricarsi impiegava ben 10 ore; il Dyna-Tac 8000X è stato messo poi in vendita nel 1983 negli Stati Uniti e data la sua forma venne soprannominato “il mattone”. Dopo il successo del Dyna-Tac, nel 1989 Motorola introdusse il MicroTac, molto più piccolo e con “lo sportellino”; in Europa invece, i primi cellulari sono stati commercializzati a partire dal 1989 e nel 1991 fu commercializzato il primo telefono cellulare GSM, il sistema di telefonia mobile che è andato poi a sostituire il sistema TACS, che consentiva non solo di effettuare telefonate, ma di inviare e ricevere messaggi di testo. Nel 1992 nacquero pertanto gli sms, utilizzati dapprima dai soli operatori telefonici e nel 1997 siccome non era più sufficiente una sola linea per gestire tutto il traffico, fu creata una seconda banda, la 1800mHz dando vita dunque ai telefonini Dual Band.

Il cellulare sicuramente più famoso degli anni passati è sicuramente lo StarTac della Motorola, nato nel lontano 1996 e grazie al quale è stato coniato il termine “clamshell”, poichè l’apertura del cellulare ricordava quella di una conchiglia; questo design è stato parecchio imitato e ancora oggi viene utilizzato negli attuali dispositivi e riscuote un grande successo. Ma il vero antenato dei telefoni cellulari moderni è il Nokia 9000i, nato nel 1997, un dispositivo che univa le funzioni di telefonino e computer, caratterizzato da un processore derivante dagli Intel 386 e con 8Mb di memoria Ram; questo cellulare era una novità: infatti è stato il primo ad avere un display con orientamento orizzontale e la tastiera Qwerty. Il 9000i poteva inviare e ricevere fax ed sms e poteva collegarsi ad internet e inviare messaggi con una lunghezza massima di 160 caratteri. L’anno seguente, il 1998 è stato un anno rivoluzionario per i cellulari, essi infatti prima erano tutti dotati di antenna esterna che toglieva fascino ai dispositivi; gli ingegneri Nokia, in quell’anno disegnarono un’antenna piccola e piatta, che andava posizionata all’interno

del dispositivo, il risultato fu il Nokia 8810, il cui design piccolo e compatto è diventato poi basilare nei terminali dell'azienda, infatti al giorno d'oggi è raro trovare un cellulare con antenna esterna. Nel 1999 comparì il primo cellulare con tecnologia Wap il Nokia 7110 , che consentiva l'accesso ad internet.

1.1.2 Caratteristiche

Dalla sua comparsa, il telefono cellulare ha usato diversi sistemi di funzionamento principali, basati su differenti tecnologie e standard di comunicazione; si è passato dai sistemi analogici (160/450/900 MHz) a quelli digitali basati su standard GSM, GPRS, UMTS/EDGE e VSF-Spread OFDM (rispettivamente terza e quarta generazione ossia 3G e 4G). Mentre il passaggio dal segnale analogico a quello digitale ha permesso di implementare alla sola chiamata vocale l'uso dei messaggi di testo SMS , ora da pochi anni, l'UMTS, la terza generazione, permette l'utilizzo del telefono cellulare anche per inviare foto digitali, effettuare videotelefonate, registrare e visualizzazione videofilmati, navigare in Internet (nello speciale protocollo Wireless Application Protocol WAP), spedire e-mail e in alcuni modelli visionare il segnale TV di alcune emittenti televisive dedicate. Con l'evoluzione dei segnali radio si è affiancata una ancor più evidente evoluzione tecnologica del telefono cellulare; dagli schermi LCD monocromatici poi divenuti a colori, ora con tecnologia a LED e touch screen. La grafica sempre più definita, l'impiego di menù complessi fino all'uso delle icone rendendo, nel tempo, i telefoni cellulari sempre più simili a un Computer. Accessori come macchine fotografiche integrate in grado di fotografare con ottime risoluzioni, riprendere piccoli video e a seconda del modello e alla quantità di memoria disponibile la possibilità di registrare veri e propri filmati digitali. Visionare foto e video sempre più dettagliati, navigare in Internet su siti sempre più simili a quelli visionati da un PC, un'evoluzione senza sosta. Le suonerie, grazie a micro casse acustiche e a componenti audio incorporati nel telefono sempre più sofisticati, da monofoniche sono divenute polifoniche e poi stereo per arrivare all'audio virtual 3D. Oramai anche i cellulari sono dei veri e pro-

pri apparati multimediali in grado di far ascoltare stazioni radio, compilation di MP3, effettuare registrazioni ambientali, memorizzare l'audio delle nostre conversazioni telefoniche e il tutto con qualità audio di altissimo livello. Sono aumentate anche le porte e i metodi di connessione verso altri sistemi come ad esempio la trasmissione dati ad Infrarosso (IR), il Bluetooth e il wi-fi, la connessione tramite cavo USB, la possibilità di collegare cuffie, auricolari, microfoni e apparati vivavoce per auto o volendo un vero e proprio impianto HiFi. Alcuni modelli inoltre possono alloggiare schede di memoria aggiuntive Secure Digital (SD) per potenziare le prestazioni del telefono o perfino alloggiare e gestire 2 schede telefoniche SIM contemporaneamente quindi con due differenti numerazioni telefoniche sullo stesso telefono.

1.1.3 Sistemi operativi

I software che gestivano i telefoni delle passate generazioni erano molto rudimentali, non definibili sistemi operativi e piuttosto simili su tutti i modelli disponibili sul mercato. Ora nei cellulari di ultima generazione sono veri e propri Sistemi Operativi in grado di gestire i più svariati applicativi come browser per navigare in Internet, programmi di video scrittura e posta elettronica, giochi e suonerie scaricabili dalla rete ecc. Alcuni cellulari chiamati Smartphone offrono anche la possibilità di installare programmi complessi come per il fotoritocco, per il controllo del computer, per la protezione crittografica della conversazione (crypto phone) o altri innumerevoli applicativi.

1.2 Dispositivi palmari

Un computer palmare, spesso indicato in lingua inglese con l'acronimo PDA (Personal Digital Assistant), è un computer di dimensioni contenute, tali da essere portato sul palmo di una mano (da cui il nome), dotato di uno schermo tattile. Originariamente concepito come agenda elettronica (in inglese electronic organizer), o sistema non particolarmente evoluto dotato di

un orologio, di una calcolatrice, di un calendario, di una rubrica dei contatti, di una lista di impegni/attività e della possibilità di memorizzare note e appunti (anche vocali). L'insieme di queste attività (e anche altre) che il PDA organizza viene anche chiamato PIM, Person Information Manager.

1.2.1 Origini

Il concetto di palmare e il termine di personal digital assistant (PDA) fu coniato dal dirigente di Apple, John Sculley, nel 1992, in una conferenza stampa tenuta in occasione della mostra informatica Consumer Electronics Show di Las Vegas. Nel 1993 la Apple produsse il primo computer palmare: il MessagePad, questo palmare poteva contare su importanti prestazioni hardware in anticipo sui tempi (processore ARM¹ strongArm RISC² a 166Mhz, 8Mb ROM, 8Mb RAM espandibile, slot PCMCIA cardbus, infrarossi, audio in/out e schermo tattile da 5 pollici 320x480 a 16 bit), era molto all'avanguardia per l'epoca pure sul fronte software (dal riconoscimento della scrittura a quello vocale, dalla navigazione internet agli applicativi base), tuttavia non ottenne presso il pubblico il successo sperato, anche a causa del costo elevato, rimanendo ai margini del mercato fino a scomparire. Anni dopo, quando il mercato era ormai maturo, altri produttori si sono affacciati su questo settore proponendo palmari con caratteristiche generalmente inferiori al Newton originale, ma con un prezzo e dimensioni più contenute, ottenendo ampi riscontri economici. Dopo l'Apple, chi ha poi ampliato il mercato rendendo i PDA un oggetto non solo per "addetti ai lavori" è stata senza dubbio la Palm. Il primo Palm Pilot (del 1996), seguito dal Professional,

¹L'architettura ARM (Advanced RISC Machine) indica una famiglia di microprocessori RISC a 32-bit utilizzata in una moltitudine di sistemi embedded. Grazie alle sue caratteristiche di basso consumo l'architettura ARM domina il settore dei dispositivi mobili. Attualmente la famiglia ARM è una delle più diffuse architetture a 32 bit del mondo.

²RISC (Reduced Instruction Set Computer) indica un'architettura per microprocessori che predilige uno sviluppo semplice e lineare. Questa semplicità di progettazione permette di realizzare microprocessori in grado di eseguire il set di istruzioni in tempi minori rispetto a una classica architettura CISC.

hanno infatti fatto storia, il mercato americano si dimostrò molto recettivo verso questi nuovi dispositivi: innanzitutto per il prezzo abbordabile e l'ergonomicità, ma anche per il design, che lo rendeva uno status symbol. La Microsoft non si fece aspettare ed entrò sul mercato a metà degli anni '90 con la prima versione di Windows CE, successivamente i palmari targati Windows CE vennero chiamati PocketPC, sigla che indica un sistema operativo molto più convincente e sicuramente orientato alla multimedialità "da passeggio". Altri PDA molto famosi furono la serie Cassiopea della Casio oppure il Velo Compaq.

1.2.2 Caratteristiche

I palmari sono molto utilizzati per le email, per la scrittura di note e promemoria e prevedono quindi una piccola tastiera semplice da usare per l'introduzione di questi testi. Normalmente questi dispositivi sono dotati della capacità di collegarsi e sincronizzare dati con i personal computer, sia con un collegamento a infrarossi che con una connessione seriale, USB o Bluetooth; inoltre spesso è possibile caricare programmi appositamente sviluppati che permettono di aggiungervi le più diverse funzionalità: fogli elettronici, calcolatrici scientifiche, client di posta elettronica, MP3 e video player, giochi, ecc. Infine alcuni palmari integrano o possono collegarsi a dispositivi esterni (telefono cellulare, GPS) aumentandone le possibilità d'uso. Ultimamente i palmari stanno diventando sempre più potenti e accessoriati; alcuni modelli integrano in sé direttamente la connettività telefonica GSM / GPRS / EDGE / UMTS / HSDPA (trasformandosi in smartphone), e quindi sono in grado di fare anche da telefono cellulare in modo autonomo. Il maggiore limite che si riscontra, finora, è quello della memoria disponibile, che raramente supera i 128 MB, pur essendo estendibile in modo limitato con memory card. Per ovviare a questo inconveniente alcuni produttori hanno lanciato sul mercato dei dispositivi dotati di un hard disk interno (la cui capacità varia dai 2 GB agli 8 GB).

1.2.3 Sistemi operativi

I più diffusi sistemi operativi per computer palmari sono:

Palm OS, sviluppato da PalmSource, Inc. Dalla sua introduzione nel 1996, la piattaforma PalmOS ha definito il trend e le aspettative per i palmari. Grazie a questa piattaforma, i palmari da semplici agende elettroniche si sono evoluti in veri e propri computer. L'ultima versione impiegata da produttori di hardware del sistema operativo è Palm OS 5 (Garnet), che supporta microprocessori a 32 Bit ARM della Motorola, Intel e Texas Instruments. A differenza delle versioni precedenti, Palm OS 5 è multithread, ed ha il supporto nativo per il Wi-Fi, un sistema di crittografia a 128 Bit ed un supporto per schermi a risoluzione fino a 320 x 480 pixel; inoltre ha un miglior supporto multimediale per video e audio. Si sono susseguite diverse versioni, l'ultima è la 5.4.9 che migliora il supporto per dispositivi bluetooth con protocollo 1.2 e supporta, seppur in maniera limitata, il multitasking; con i palmari e smartphones Palm OS si possono riprodurre un video o file MP3 o eseguire alcune altre funzioni mentre vengono svolte attività. Nel 2008 è stata presentata una nuova versione di Palm OS, sotto il nome di ALP (Access Linux Platform) o Palm OS on Linux, sviluppato su un kernel Linux ottimizzato per dispositivi mobili.

Windows Mobile (Windows CE) di Microsoft per i suoi dispositivi Pocket PC. Secondo l'ideologia di Microsoft, il Pocket PC è un "dispositivo portatile che permette agli utenti di ricevere e inviare e-mail, gestire rubriche personali, gestire appuntamenti, riprodurre file multimediali, usare videogiochi, navigare in Internet e altro" ; qualsiasi dispositivo che viene classificato come Pocket PC deve attenersi a diversi requisiti:

- Deve eseguire il sistema operativo Microsoft Windows CE (versione PocketPC);
- Deve aver installato nella memoria ROM una serie di applicativi standard;

- Deve includere un touchscreen;
- Deve includere un pad direzionale o touchpad;
- Deve essere basato su una CPU ARM (versione 4 o compatibile) o processori Intel XScale (La prima generazione di Pocket PC era dotata di processori MIPS o SH3).

Windows CE (Windows Embedded Compact) è un sistema operativo real time sviluppato da Microsoft, a partire dal 1996, per dispositivi portatili (PDA, Palmari, Pocket PC), Smartphone e sistemi embedded. Come si intuisce dal nome, è un derivato della famiglia di sistemi operativi Windows, ma ha un kernel differente e non è quindi una semplice "riduzione". Le API e l'aspetto grafico sono comunque molto simili.

Il termine "Windows CE" è in realtà il nome tecnico con il quale viene indicata la piattaforma generale di sviluppo di questo sistema operativo. Essendo Windows CE sufficientemente modulare e flessibile, sono state sviluppate delle specifiche versioni per dispositivi differenti (oltre che per processori differenti). Tali specifiche versioni hanno diversi nomi "commerciali" come: MS Handheld 3.0 (e 3.1), MS Handheld 2000, Microsoft Pocket PC 2000 (e 2002), MS Smartphone 2002, MS Windows Mobile 2003, fino ai più recenti MS Windows Mobile 6.0, MS Windows Mobile 6.5 e Windows Phone 7. Tali varianti fanno tutte riferimento a specifiche evoluzioni della piattaforma di riferimento "Windows CE", passato dalla v1.0 alle ultime versioni utilizzate anche per gli smartphone.

GNU/Linux, utilizzato da Sharp per i suoi Zaurus. Gli Zaurus sono computer palmari costruiti dalla Sharp Corporation per il mercato giapponese; utilizzano un'implementazione del sistema operativo GNU-Linux e, recentemente, OpenBSD. Il primo modello è stato l'SL-5000D, sul quale girava Qtopia basato su Embedix Plus; gli ultimi modelli della serie Zaurus SL-CXXXX hanno uno schermo retro illuminato touch screen, una tastiera qwerty, uno slot CompactFlash PCMCIA, uno slot per l'espansione con memorie

SD ed una porta mini USB. Nel 2007 Sharp ha annunciato la decisione di sospendere la produzione.

Symbian (prima noto come **EPOC**). Symbian non è altro che una evoluzione del sistema operativo dei palmari Psion, denominato Epoc. Epoc è stato sviluppato da Psion verso la fine degli anni 80; era un sistema operativo assai evoluto, essendo a 16 bit, basato sui processori Intel della serie 8086, cioè gli stessi che pulsavano nei primi PC della IBM; è anche un vero multitasking preemptive, cosa che neanche le prime versioni di Windows erano. Epoc subisce una importante evoluzione nel 1997, quando venne presentata una nuova edizione, completamente diversa, concepita per lavorare con i processori ARM; questi processori ebbero un enorme successo, sia a causa della loro potenza e versatilità, data dall'architettura RISC a 32 bit, sia per il loro basso consumo, caratteristica importantissima per apparecchi alimentati a batteria. Le prime versioni di Symbian infatti erano praticamente identiche alle ultime versioni di Epoc, montate sui palmari Psion Serie 5 e Revo; l'unica aggiunta apportata consisteva nella gestione della porzione telefonica, assente in Epoc, dato che gli Psion erano palmari puri, non dei telefonini. Il sistema operativo Symbian, nelle sue ultime versioni, verrà discusso nel dettaglio più avanti.

A causa dell'evolversi del mercato di nicchia di questi apparecchi, le vecchie generazioni sono ormai scomparse per lasciare il passo ad apparecchi leggermente più evoluti come i netbook, i tablet o i più simili smartphone.

1.3 Dispositivi Smartphone

Uno **smartphone**, o in italiano telefonino intelligente, è un dispositivo portatile che abbina funzionalità di telefono cellulare a quelle di gestione di dati personali. Può derivare dall'evoluzione di un PDA a cui si aggiungono funzioni di telefono (per questo detti anche PDA-Phones) o viceversa, di un

telefono mobile a cui si aggiungono funzioni di PDA. La caratteristica più interessante degli smartphone è la possibilità di installarvi ulteriori applicazioni, che aggiungono nuove funzionalità; questi programmi possono essere sviluppati dal produttore dello smartphone, dallo stesso utilizzatore, o da terze parti.

1.3.1 Origini

Il primo smartphone apparso sul mercato si chiamava Simon, è stato progettato da IBM nel 1992 e presentato come prototipo, è stato poi messo sul mercato a partire dal 1993 e venduto dalla BellSouth; oltre ad essere un telefono cellulare, conteneva anche un calendario, rubrica, orologio mondiale, calcolatrice, blocco note, internet, e-mail, invio e ricezione fax, e giochi. Simon già all'epoca godeva di un grande vantaggio tecnologico essendo dotato di monitor touch screen con la possibilità di digitare i numeri con le dita o con una stilo inclusa nella confezione; con gli standard odierni, la qualità di Simon risulterebbe piuttosto bassa, ma per l'epoca era veramente un gioiello della tecnologia. La linea Communicator fu la prima della classe smartphone di Nokia iniziata nel 1996 con il modello Nokia 9000; questo smartphone è il risultato di uno sforzo di collaborazione di uno dei primi modelli PDA di Hewlett Packard in combinazione con uno dei migliori telefoni Nokia di quel tempo. Il modello successivo, il Nokia 9210, è stato probabilmente il primo vero smartphone con un sistema operativo, Nokia stessa però ha continuato a riferirsi ad esso come Communicator.

Nel mese di ottobre, 2001 Handspring ha annunciato il Palm OS smartphone Treo, utilizzando una tastiera completa che combina la navigazione web senza fili, e-mail, calendario e rubrica, utilizzando applicazioni di terze parti che potevano essere scaricate e sincronizzate con un computer. Nel 1999 nacque il marchio RIM Blackberry e nel 2002 fu lanciato il loro primo smartphone, il BlackBerry 5810, pensato per la categoria business, caratterizzato da tastiera Qwerty, con supporto Push Email e fu il primo smartphone ottimizzato per l'utilizzo wireless; con la quale raggiunse una base totale di

8 milioni di clienti nel giugno 2007, di cui tre quarti nel solo Nord America. Nel 2002 negli Stati Uniti, ci fu il lancio sul mercato del T-Mobile Sidekick altrettanto detto “Danger Hiptop”, uno smartphone innovativo che annunciava una nuova “forma” caratterizzato da un display LCD ampio e a scorrimento che nascondeva una tastiera QWERTY; la linea “Sidekick” persiste ancora oggi e ha influenzato parecchi dispositivi, questa tipologia di design ha inoltre aiutato la diffusione dei messaggi di testo.

Motorola nel 2004 stravolse l’atmosfera con il suo RAZR V3, un cellulare clamshell sottilissimo con un ampio display lcd a colori, tastiera marchiata, fotocamera e soprattutto capacità multimediali; più che le sue caratteristiche tecniche, questo smartphone raggiunse il successo grazie al suo design innovativo. Nel 2007 ecco arrivare sul mercato della telefonia mobile, Apple, con il suo iPhone, un dispositivo non solo pensato per “telefonare”, ma anche per giocare e per ascoltare la propria musica preferita. L’iPhone è stato il primo dispositivo a consentire l’accesso ai vari social network, come Facebook, Twitter e quant’altro; l’introduzione di questo dispositivo rivoluzionario nel mercato portò lo sviluppo degli smartphone allo stato attuale dove il mercato della telefonia mobile è forse quello più redditizio e sviluppato.

1.3.2 Caratteristiche

Mentre negli USA gli smartphone tendono ad essere PDA a cui si aggiungono funzioni di telefono, al contrario, in Europa e Giappone sono telefoni con aggiunta di funzioni di PDA. La loro forma e l’interfaccia con cui l’utente può utilizzare il dispositivo può essere o molto simile a un PDA, quindi con un display di dimensioni ampie sensibile al tocco (Touch Screen) dello stilo o delle dita che funziona da monitor, da interfaccia fisica per la gestione del software e da strumento di scrittura grazie all’uso della tastiera touch virtuale (alcuni modelli incorporano una tastiera integrata come opzione di scrittura aggiuntiva allo stilo); o molto simile a un cellulare (display di dimensioni ridotte che mostra funzioni controllabili attraverso il tastierino alfanumerico, il navigatore o joystick, un pulsante che incorpora tasti direzionali e di confer-

ma delle operazioni, altri tasti programmabili, frontali e laterali, che aprono le applicazioni o inizializzano determinate funzionalità. Indipendentemente dalla forma, questi dispositivi presentano le seguenti funzionalità:

- Organizzazione delle attività personali o PIM, Person Information Manager (calendario, lista delle attività e degli appuntamenti, rubrica dei contatti, schedario, note testuali con possibilità di riconoscimento della scrittura manuale, appunti vocali con possibilità di integrare la tecnologia di riconoscimento vocale, calcolatrice, orologio e sveglia, ecc.).
- Funzioni di personal computer (programmi quali videoscrittura, fogli di calcolo, data base, presentazioni, ecc.) e compatibilità con i più comuni formati di file come PDF e quelli della suite Office.
- Funzioni multimediali e intrattenimento (visualizzatore di filmati, lettore musicale di MP3, album di immagini statiche o animate, fotocamera digitale e videocamera digitale integrate, giochi, lettore di e-book, ecc.).
- Servizi di telecomunicazione tradizionali (telefono, fax).
- Servizi Internet, WAP, GPRS, EDGE, UMTS, HSDPA, HSUPA (navigazione nel Web con browser che renderizzano le pagine web in modo da poterle leggere senza dover scrollare orizzontalmente, e-mail, programmi di messaggia istantanea, Web TV, Web radio, fruizione della musica on line, invio e ricezione di SMS e MMS, suonerie polifoniche e loghi, supporto JAVA, giochi on line, chat, videochiamate, ecc.).
- Software per GPS (navigatore satellitare; mappe satellitari interattive).
- Connettività wireless WiFi, Bluetooth o IrDA con dispositivi di vario tipo (personal computer Windows OS o Mac OS, periferiche hardware, cellulari, ecc.).

1.3.3 Sistemi operativi

Negli smartphone, il sistema operativo è il software preinstallato che gestisce tutte le funzioni che lo smartphone è in grado di svolgere. Oggi i sistemi operativi più diffusi presenti sul mercato sono cinque: Android, Apple iOS, Symbian OS, Windows Mobile, Blackberry OS; nel capitolo seguente illustriamo le caratteristiche di ciascun SO.

Capitolo 2

Sistemi operativi dei dispositivi smartphone

2.1 Android

Android è un sistema operativo per dispositivi mobili sviluppato inizialmente da Android Inc. e acquisito da Google, il gigante di internet, nel 2005. Fondamentalmente Android non è sviluppato da zero, è infatti un sistema operativo che si basa su diverse versioni del kernel Linux, ciò contraddistingue questo sistema operativo dagli altri per la sua natura open source e per la sua versatilità, infatti può funzionare su qualsiasi dispositivo mobile. Il primo smartphone dotato di piattaforma Android è stato l'HTC Dream, presentato il 22 ottobre del 2008.

2.1.1 Versioni

Le versioni di Android sono:

- **Cupcake** (Android 1.5, basato sul kernel Linux 2.6.27);
- **Donut** (Android 1.6, basato sul kernel Linux 2.6.29);
- **Eclair** (Android versione 2 e 2.1, basata su Linux Kernel 2.6.29);

- **Froyo** (Android versione 2.2, basato sul kernel Linux 2.6.32);
- **Gingerbread** (Android versione 2.3, basato sul kernel Linux 2.6.35.7);
- **Honeycomb** (Android versione 3.0 per tablet);
- **Ice Cream Sandwich** (Android versione 4.0).

Android 2.2 Froyo è stato rilasciato nel maggio 2010 (Rev 1.0) ed è stato aggiornato nel luglio 2010 (Rev 2.0); anche se è stata una minor release, ci sono stati molti miglioramenti , si riporta in seguito le sue principali funzioni e caratteristiche:

- Kernel Linux 2.6.32
- Multitasking
- Integrazione del motore JavaScript Chrome V8 nel Browser
- Supporto avanzato per Microsoft Exchange
- Supporto Bluetooth e Wi-Fi
- Funzione hotspot Wi-Fi (consente di collegare fino a sei dispositivi allo smartphone come se fosse un router wireless a banda larga)
- Supporto per il caricamento dei file nei form dell'applicazione browser
- GIF animate supportate nel browser
- Adobe Flash 10.1 supportato
- Supporto per schermi ad alta risoluzione
- nuovi widget che consentono agli utenti di configurare i loro schermi con widget e collegamenti in modo efficiente.
- collegamenti dedicati per il browser, launcher App e Telefono sono stati forniti sulla schermata iniziale, gli utenti così possono accedere ai servizi da una qualsiasi delle 5 schermate Home.

- Protezione con password Alfanumerica o numerica per sbloccare il dispositivo.

Android 2.3 Gingerbread è stato pubblicato il 6 Dicembre 2010. Ci sono parecchi miglioramenti e nuove funzionalità incluse in Gingerbread, infatti Android 2.3 è una major release; supporta le seguenti caratteristiche in aggiunta a quelle precedenti:

- Kernel Linux 2.6.35
- Nuova interfaccia utente con nuovi temi (temi a risparmio energetico)
- Supportate dimensioni Extra Large dello schermo
- Comunicazione VoIP/SIP (per le chiamate audio e video. Se si ha a disposizione una buona rete 3G o Wi-Fi e un account SIP è possibile effettuare telefonate via Internet.)
- Supporto per la tecnologia NFC (Near Field Communication, che è un meccanismo di comunicazione ad alta velocità che opera su alte frequenze in distanze limitate di circa 10 cm).
- Supporto per riproduzione video WebM/VP8 e audio con codifica AAC
- Nuovi effetti audio, come riverbero, equalizzazione, virtualizzazione delle cuffie e bass boost
- la tastiera è stata ridisegnata
- Migliorata la funzionalità Copia e Incolla
- Migliorato il software multi-touch per la tastiera
- Miglioramenti di audio, grafica ed input per gli sviluppatori di giochi
- Nuovi sensori (e.s. giroscopio)
- Download manager migliorato

- Migliorata la gestione della batteria e il controllo delle applicazioni (Infatti Android 2.3 gestisce il sistema operativo e le applicazioni in background in modo migliore, chiudendo quelle non necessarie.)
- Supporto per più fotocamere

Android 4 Ice Cream Sandwich presentato il 19 Ottobre 2011 ha integrato numerose nuove funzionalità e tecnologie:

- Rinnovata l'interfaccia utente (azioni comuni più visibili, carattere nuovo ottimizzato per una migliore leggibilità, nuovi pulsanti virtuali);
- Multitasking più semplice e visivo per gli utenti con il pulsante "applicazioni recenti";
- Widget ridimensionabili, così gli utenti possono espanderli per visualizzare più contenuti o ridurli per risparmiare spazio;
- Inserimento del testo migliorato e controllo ortografico;
- Nuovo e potente motore di input vocale;
- Contatti e profili sono integrati attraverso applicazioni e social network per una facile accessibilità;
- Migliorata la funzionalità della fotocamera e photo editor nativo;
- Calendario integrato con tutte le altre applicazioni;
- Possibilità di scattare e condividere screenshot;
- Esperienza cloud;
- Navigazione web migliorata con la possibilità di gestire i segnalibri di google chrome del pc, possibilità di visualizzare la versione desktop del sito e lettura offline;

- Condivisione basata sulla tecnologia NFC (Android Beam consente alle persone di scambiare istantaneamente applicazioni preferite, contatti, musica, video. Basta appoggiare due dispositivi Android dotati di chip NFC, quindi toccare per inviare);
- Face Unlock (opzione per il lock-screen che consente agli utenti di sbloccare i dispositivi con i loro volti, si avvale della tecnologia di riconoscimento facciale per registrare un volto e riconoscerlo in un secondo momento, al momento dello sblocco del dispositivo).

2.1.2 Caratteristiche tecniche

Il sistema operativo è costituito da uno stack software (ovvero un set di sottosistemi software) che include un sistema operativo di base, basato sul kernel Linux, e che è composto da applicazioni Java che vengono eseguite su uno speciale framework, basato anch'esso su Java e orientato agli oggetti; vengono utilizzati i middleware ¹ per le comunicazioni e le applicazioni di base.

Sopra il kernel poggiano le librerie fondamentali, la piattaforma utilizza un set di librerie C e C++ come: il leggero database SQLite; la libreria SGL per la grafica bidimensionale; l'implementazione delle API OpenGL ES 2.0 per gestire l'accelerazione grafica 3D; Media Libraries per il supporto a diversi formati audio e video, oltre che alla manipolazione delle immagini; un motore grafico e di layout basato su WebKit; surface manager, un framework multimediale OpenCore e System C library, ovvero una libreria derivata da libc, la libreria standard Linux, adattata per i dispositivi mobili.

Le applicazioni, una volta scritte in linguaggio Java, vengono eseguite tramite la **Dalvik virtual machine**, una macchina virtuale adattata per

¹Con il termine middleware si intende un insieme di programmi informatici che fungono da intermediari tra diverse applicazioni e componenti software. L'utilizzo di uno strato software aggiuntivo, consente di ottenere un elevato livello di servizio per gli utenti, ed un elevato livello di astrazione per i programmatori.

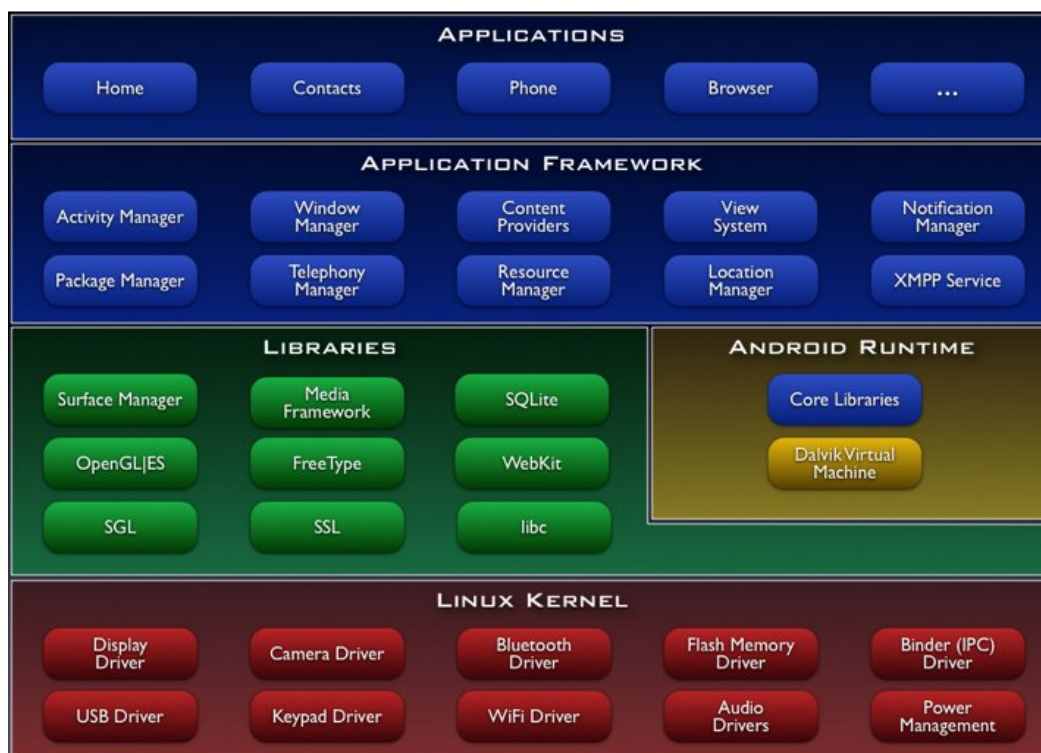


Figura 2.1: Architettura del sistema operativo Android

l'uso su dispositivi mobili dotata di compilatore just-in-time (JIT) ². La DVM è stata progettata da Dan Bornstein, dipendente Google, ed è uno dei componenti principali di Android; grazie ad un utilizzo intelligente dei registri di sistema permette una maggiore ottimizzazione della memoria in dispositivi con bassa capacità, consente di far girare diverse istanze della macchina virtuale contemporaneamente e nasconde al sistema operativo sottostante la gestione della memoria e dei thread. Ovviamente la DVM non esegue bytecode Java ma un qualcosa che si può ottenere da esso e che prende il nome

²Un compilatore just-in-time permette un tipo di compilazione, conosciuta anche come traduzione dinamica, con la quale è possibile aumentare le performance dei sistemi di programmazione che utilizzano il bytecode, traducendo il bytecode nel codice macchina nativo in fase di run-time. L'obiettivo finale dei sistemi JIT è di combinare i vantaggi della compilazione del bytecode a quelli della compilazione nativa, aumentando le prestazioni quasi al pari di una compilazione direttamente in linguaggio macchina.

di Dalvik bytecode. Le applicazioni per Android si sviluppano in Java per poi trasformare il bytecode Java in bytecode Dalvik, quindi su un dispositivo Android non girerà alcun bytecode Java ma un bytecode le cui specifiche sono descritte dal formato .dex (Dalvik EXecutable).

Nel penultimo strato dell'architettura è possibile rintracciare i gestori e le applicazioni di base del sistema; ci sono gestori per le risorse, per le applicazioni installate, per le telefonate, il file system e altro ancora: tutti componenti di cui difficilmente si può fare a meno.

Infine, sullo strato più alto dell'architettura, poggiano gli applicativi destinati all'utente finale. Molti, naturalmente, sono già inclusi con l'installazione di base: il browser ed il player multimediale sono dei facili esempi; in questo livello si inseriranno anche le applicazioni sviluppate da terze parti.

In sintesi, il sistema operativo Android è composto in tutto da 12 milioni di righe di codice che comprendono 3 milioni di righe di XML, 2,8 milioni di righe di C, 2,1 milioni di righe di Java e 1,75 milioni di righe di C++.

2.1.3 Applicazioni

Android è fornito di una serie di applicazioni preinstallate: un browser, basato su WebKit, una rubrica e un calendario. Quando si vuole aggiungere all'ambiente Android una funzionalità non presente, come ad esempio un software per l'ufficio, un videogioco o una immagine di sfondo (wallpaper), si usa cercarla nel Market e "installarla", ossia copiarla all'interno del dispositivo affinché sia sempre presente e utilizzabile.

Dalla versione 2.2 di Android è possibile installare una app, oltre che nella memoria interna del dispositivo, su una card esterna; questa nuova feature viene spesso chiamata dalla community "app2sd" (o apps2sd), derivata dai nomi dei primi esperimenti effettuati da programmatori indipendenti che, attraverso del software nativo (scripts in linguaggio shell), spostavano su card esterna sia i software installati che alcune cartelle di sistema, utilizzando poi dei link simbolici per i collegamenti e facendo anche delle copie di riserva per poter ripristinare la situazione precedente in sicurezza.

Per volere della stessa Google Inc., è stato allestito un sito di E-commerce destinato a creare un punto di incontro tra gli utilizzatori dei dispositivi che cercano software per Android e gli sviluppatori che vogliono diffonderli a pagamento o gratuitamente. Si chiama **Android Market**. Tutti o quasi i dispositivi Android hanno preinstallata una icona denominata "Market". Per accedere al Market, bisogna eseguire tale applicazione e registrarsi gratuitamente al servizio; la registrazione è sia per scopi di fatturazione che di prevenzione a eventuali abusi del servizio stesso. All'interno del Market si possono visionare le ultime novità oppure fare una ricerca di software o altro materiale (come ad esempio i wallpaper o le suonerie) per termine, tipologia (applicazione, videogioco, altro) o di costo (gratuito o a pagamento). Al fine di agevolare la diffusione del servizio, non è necessario depositare subito i dati contabili. Inoltre non è prevista la possibilità di addebitare l'acquisto su una scheda telefonica SIM, l'acquisto di un prodotto nel Market può essere effettuato tramite Google Checkout, con una carta di credito.

Il file APK

Il software viene solitamente distribuito sotto forma di pacchetto autoinstallante, quindi un file con estensione .APK . Questo non è altro che un file compresso, contenente il software (file con estensione .dex) le sue risorse (immagini, suoni ecc...) e alcuni file XML. L'utente medio non ha necessariamente bisogno di conoscere tale tipologia di file, dato che il dispositivo gestisce tutta la parte di installazione mediante web services come Android Market. All'interno di questo file c'è anche un certificato che permette l'installazione di un pacchetto .APK su un dispositivo Android se questo non è stato compromesso o revocato. Il certificato deve essere presente in qualsiasi pacchetto, altrimenti Android non installerà l'applicazione al suo interno, il certificato viene creato dallo sviluppatore dell'applicazione, che può scegliere di crearne uno di "debugging" (quindi a uso interno) o di "mercato" (per la distribuzione) e può deciderne la sua diffusione delle copie (libera o limitata). Il distributore (per esempio Android Market) ci aggiungerà poi una

sua chiave, che potrà successivamente revocare, se necessario. In caso di revoca, l'applicazione non è più installabile né eseguibile in nessun dispositivo Android.

Se uno sviluppatore indipendente vuole poter distribuire un suo software con pacchetto .APK , senza passare per un web service certificato, può autocrearsi il certificato; in tal caso, l'utente che installa l'applicazione riceverà un avviso che sta installando un software di questo tipo ("self-signed"); a questo punto potrà annullare l'installazione o farla proseguire a suo rischio.

2.1.4 Ambiente di sviluppo

Le applicazioni di Android sono sviluppate all'interno di un framework, ossia di una struttura dati specifica; gli sviluppatori hanno pieno accesso alle stesse framework API usate dalle applicazioni di base. L'architettura delle applicazioni è progettata per semplificare il riutilizzo dei componenti; ogni applicazione può rendere pubbliche le sue capacità e tutte le altre applicazioni possono quindi farne uso (sono soggette ai limiti imposti dalla sicurezza del framework). Questo stesso meccanismo consente all'utente di sostituire i componenti standard con versioni personalizzate. Alla base di ogni applicazione si trova un set di servizi e sistemi, tra cui:

- **System View:** un gruppo ricco ed estensibile di strumenti che possono essere usati per costruire un'applicazione. Contiene liste, caselle di testo, pulsanti e addirittura un browser web integrato;
- **Content Providers:** permettono alle applicazioni di accedere a dati da altre applicazioni, come i contatti, o di condividere i propri dati;
- **Resource Manager:** offre l'accesso a risorse non-code come strings localizzate, grafica, files di layout;
- **Notification Manager:** permette a tutte le applicazioni di mostrare avvisi personalizzati nella status bar;

- **Activity Manager:** gestisce il ciclo di vita delle applicazioni e fornisce un backstack di navigazione comune;

La struttura del framework è molto chiara se si utilizza l'ambiente di sviluppo (SDK, Software Development Kit), esso include gli strumenti di sviluppo, le librerie, un emulatore del dispositivo, la documentazione (in inglese), alcuni progetti di esempio, tutorial e altro. È installabile su qualsiasi computer che usi come sistema operativo Windows, Mac OS X (dalla versione 10.4.8) o Linux. L'IDE (Integrated Development Environment) ufficialmente supportato per lo sviluppo di applicazioni per Android è Eclipse, per cui è fornito un plug-in progettato per fornire un ambiente potente ed integrato in cui costruire le applicazioni. L'SDK è basato sul linguaggio di programmazione Java che fornisce una serie di API specifiche per mezzo del quale è possibile interagire con il sistema operativo Android, controllare l'hardware del dispositivo e lo sviluppo dell'interfaccia grafica.

Le applicazioni Android sono caratterizzate da una certa dualità: parti dinamiche scritte in Java e parti statiche scritte in XML. Tipico delle parti statiche possono essere quelle caratteristiche che non cambiano durante l'esecuzione dell'applicazione, come per esempio il colore dello sfondo; tipico delle parti dinamiche sono invece gli aspetti programmatici come per esempio la gestione degli eventi. Questa dualità è però solo apparente, durante l'esecuzione, infatti, la Dalvik Virtual Machine (DVM) esegue sempre un programma.

Per lo sviluppo delle applicazioni è disponibile una completa documentazione la quale, anche graficamente, riprende la struttura tipica della documentazione Java del sito Oracle. Ai fini della programmazione, il team di Android ha specificato nella documentazione ufficiale vari termini per definire i vari tipi di applicazioni:

Attività (activity) sono quelle applicazioni destinate a una interazione diretta con l'utente utilizzando lo schermo e i dispositivi di input messi a disposizione dallo smartphone. Un esempio sono i videogiochi, le applicazioni

per l'ufficio e i visualizzatori (reader) di E-book. Le attività vengono generalmente distribuite sotto forma di file .APK , vengono poi installate in delle cartelle nella memoria del dispositivo (o in una card estraibile), infine viene creata una icona per l'utente, che gli permetterà di eseguirla in qualsiasi momento. E' anche possibile disinstallare le attività mediante una utility integrata con Android.

Servizi (service) sono, al contrario, quelle applicazioni che per loro natura svolgono delle operazioni autonome e che vengono richiamati dalle attività al bisogno; gira in sottofondo e non interagisce direttamente con l'utente. Il sistema operativo fornisce alle applicazioni vari servizi già pronti all'uso, per ottenere l'accesso all'hardware o a risorse esterne (ad esempio dei web services di messaggistica). Un esempio di servizio è `com.android.inputmethod.latin`, ossia il componente che fa comparire la tastiera quando si seleziona (con i tasti o con un "tocco" sul touch-screen) un campo di input testuale. I servizi possono essere eseguiti o interrotti direttamente dall'utente, sebbene siano eventualità alquanto rare.

Frammento (fragment) è quella porzione di codice (quindi di applicazione) che gestisce la parte grafica, in base alle possibilità del dispositivo su cui è stato installato. Il problema dello sviluppatore è evidente quando si trova a dover sviluppare una applicazione che funzioni, ad esempio, sia su un tablet (generalmente questi hanno uno schermo touch-screen di grandi dimensioni) che su alcuni tipi di smartphone (che possono avere 2 schermi e non è detto che entrambi siano touch-screen). E' stato allora deciso, dal team di Android, di creare il concetto di fragments, ossia una classe tanto generica da permettere lo sviluppo di una applicazione con la parte grafica slegata a quella "decisionale", in modo da rendere agevole l'adattamento dell'applicazione alle varie situazioni. Il programmatore creerà vari frammenti della parte grafica e poi Android la ridisegnerà correttamente per il dispositivo in uso; l'alternativa per il programmatore sarebbe stato di scriversi da solo un framework che facesse lo stesso lavoro (e l'avrebbe dovuto fare per tutti i suoi progetti) oppure avrebbe dovuto creare più versioni, ognuna destinata a

una tipologia di dispositivo, quindi costringendolo a riscrivere diverse parti del proprio software.

Broadcast Receiver; viene utilizzato quando si intende intercettare un particolare evento, attraverso tutto il sistema. Ad esempio lo si può utilizzare se si desidera compiere un'azione quando si scatta una foto.

Il kernel e le librerie di base; questi componenti non sono sostituibili; al massimo sono aggiornabili alcune parti per correggere eventuali problemi di sicurezza. Quando viene rilasciata una nuova versione di Android, significa che alcune di queste parti sono state aggiornate o sostituite.

2.2 iOS

iOS (fino a giugno 2010 iPhone OS) è il sistema operativo sviluppato da Apple. È stato rilasciato nel mercato degli Stati Uniti nel giugno 2007. iOS è un sistema operativo proprietario di Apple e può essere eseguito solo nei dispositivi Apple (iPad, iPhone e iPod Touch), è stato originariamente sviluppato per iPhone e ora è utilizzato anche negli iPod, iPad e Apple TV.

2.2.1 Versioni

Il SO mobile di Apple è iniziato con la versione 2 e la versione attuale è Apple iOS 5.

Apple iOS 3 ha introdotto caratteristiche interessanti come:

- Funzione Copia e Incolla,
- indicazioni stradali a piedi nella mappa,
- funzioni aggiuntive per youtube (come il login, commenti, video rating),
- contatti modificabili con le chiamate recenti,
- registrazione di video in HD,
- cattura video trimmer,

- unzione SMS rinominata come messaggi,
- funzionalità MMS con l'invio di immagini, video e vCard,
- opzione “Trova il mio telefono” aggiunto a MobileMe,
- supporto abbonamento iCalender,
- miglioramenti in Safari,
- supporto HTML5,
- Tenere premuto per ottenere le opzioni apri, apri in nuova pagina e link copia,
- supporto linguistico migliorato,
- Tethering via USB, Bluetooth
- nuove applicazioni memo vocale.

La quarta release del sistema operativo, **Apple iOS 4.2.1**, pubblicata con iPhone 4 il 21 giugno 2010, ha aggiunto numerose funzioni quali:

- Multitasking (Questo è un metodo di condivisione di risorse di elaborazione comuni, come una CPU a più applicazioni.)
 - Audio di fondo – Si può ascoltare la musica durante la navigazione web, giocare, ecc
 - Voice over IP – Applicazione in grado di ricevere chiamate e continuare a parlare mentre si utilizzano altre applicazioni.
 - Posizione in background – Fornisce un modo efficiente per monitorare la posizione degli utenti quando si muovono. Questa è una grande caratteristica di social networking per identificare le posizioni amico. (Solo Se consentito dagli utenti)
 - Notifiche locali - Applicazione che segnala agli utenti eventi in programma e gli allarmi in background.

- Task finishing - l'Applicazione verrà eseguita in background e finirà il suo compito completamente, anche se l'utente abbandona l'applicazione. (Se utilizzo l'applicazione della posta elettronica per controllare le mail posso aprire intanto i messaggi per inviare un SMS mentre sto chiamando, e l'applicazione di posta potrà ancora ricevere o inviare e-mail).
- Commutazione rapida di applicazioni - Gli utenti possono passare da una qualsiasi applicazione a un'altra in modo che altre applicazioni restano in esecuzione in background fino a quando non si torna indietro.
- Airprint - rende semplice stampare e-mail, foto, pagine web e documenti direttamente dal dispositivo.
- IAD - Advertising su Mobile (piattaforma di pubblicità mobile per gli sviluppatori)
- Airplay - AirPlay consente lo streaming di file multimediali digitali in modalità wireless dall' iPhone alla Apple TV o a qualsiasi AirPlay abilitato; si possono guardare film e foto sul televisore widescreen e riprodurre musica attraverso i migliori altoparlanti della casa.
- Trova il mio iPhone - La funzione di MobileMe consente di individuare il dispositivo disperso e proteggere i propri dati. Questa funzione è ora su qualsiasi iPhone 4 con iOS 4.2. Una volta che lo si imposta, si può trovare il dispositivo perso su una mappa, visualizzare un messaggio sullo schermo, impostare a distanza un codice di blocco, e avviare una cancellazione remota per eliminare i vostri dati. E se alla fine si trova il vostro iPhone, è possibile ripristinare tutto dall' ultimo backup.
- Game Center - Ti permette di trovare amici per giocare a giochi multiplayer.
- Valorizzazione Tastiera e Directory - iOS 4.2 supporta 50 lingue.

- Messaggi di testo con tono - Assegnazione di 17 toni personalizzati per le persone in rubrica, in modo che quando si arriva sms senza guardare il testo è possibile identificare chi lo ha inviato.

Il 6 giugno 2011 è stata presentata al WWDC la quinta versione di iOS (iOS 5), uscita il 12 ottobre, con numerose nuove funzioni, tra cui:

- Gestione delle notifiche grazie al "Centro Notifiche" (le notifiche appaiono in modo istantaneo nella parte superiore dello schermo);
- Servizio iMessage, sistema di Instant Messaging gratuito per gli utenti dei dispositivi portatili Apple con iOS che siano connessi a Internet, usando la rete 3G o il WiFi. I contatti iMessage sono identificati in base all'ID (mail) o anche tramite il numero di telefono, se non si è connessi a internet verrà inviato un normale sms;
- Twitter incorporato, possibilità di condividere tweet anche da fotocamera, YouTube, browser Safari e mappe, possibilità di aggiunta posizione corrente al tweet grazie alla geolocalizzazione;
- Organizzazione automatica di riviste e giornali nella home, anche con gli abbonamenti ad essi;
- Fotocamera migliorata ed editing immagine;
- Migliorata la gestione delle mail;
- Mirroring AirPlay per 2 iPad e iPhone 4S , per visualizzare il contenuto del dispositivo anche su tv
- Nuove migliorie alla tastiera;
- Supporto iCloud (permette di sincronizzare e condividere i contenuti dei vostri dispositivi Apple su un server cloud).

2.2.2 Caratteristiche tecniche

iOS, come Mac OS X, è una derivazione di UNIX (famiglia BSD) e usa un microkernel XNU Mach basato sul sistema operativo open source Darwin OS. È un kernel ibrido, basato su un'unione del codice del microkernel Mach (uno dei primi microkernel creati e anche il più famoso) e del kernel monolitico FreeBSD (variante originaria di Unix sviluppata presso l'Università di Berkeley, ampiamente utilizzato in ambiti di ricerca, di produzione, in prodotti commerciali ed in apparecchi embedded) . Le funzioni primitive e i servizi fondamentali del kernel XNU si basano su microkernel Mach 3.0; Apple lo ha modificato ed esteso per raggiungere la funzionalità e gli obiettivi prestazionali di Mac OS X o di iOS. Il risultato è una combinazione dei vantaggi di Mach e BSD, la parte del microkernel Mach è responsabile del gestore della memoria, della comunicazione tra processi e del sistema input/output; permette inoltre protezione della memoria, preemptive multi-tasking e una gestione avanzata della memoria virtuale; la parte del kernel BSD gestisce gli utenti e i permessi, contiene lo stack di rete, offre un virtual file system (VFS) e osserva la compatibilità con le specifiche POSIX (famiglia degli standard definiti dall'IEEE denominati formalmente IEEE 1003).

iOS ha quattro livelli di astrazione: il Core OS layer, il Core Services layer, il Media layer ed il Cocoa Touch layer; il sistema operativo occupa meno di mezzo gigabyte della memoria interna del dispositivo.

Al livello più basso (**Core OS**) troviamo lo stack di derivazione BSD, quindi funzioni per IO, threading, gestione processi, sqlite etc.

Dal livello **Core Services** in poi troviamo tecnologie sviluppate da Apple, in particolare Core Foundation che , come indica il nome, è la base su cui sono sviluppate molte delle api di livello più alto. Il Core Services e Core OS, si occupano di offrire i servizi fondamentali del sistema operativo. Gli strati più in alto forniscono le astrazioni dei servizi dei livelli più in basso facilitando lo sviluppo delle applicazioni. L'accesso da parte degli sviluppatori ai livelli inferiori resta comunque garantito.

Il livello **Media** fornisce principalmente strumenti per lo sviluppo di grafi-

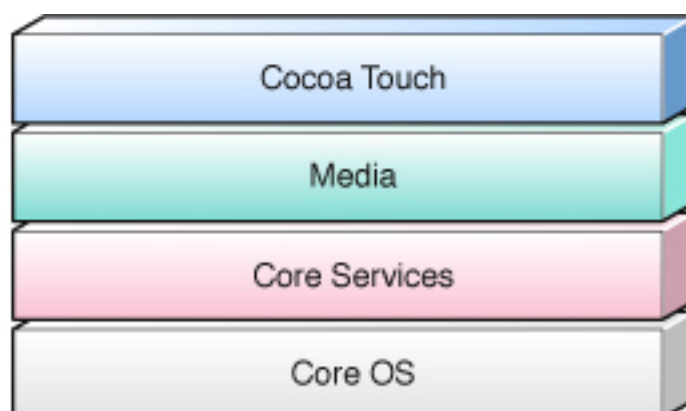


Figura 2.2: Architettura del sistema operativo Apple iOS

ca, per la gestione dell'audio e per la gestione del video. I framework OpenGL, ad esempio, rendono disponibili strumenti per il disegno 2D e 3D, o ancora, il framework Quartz Core, con le interfacce Core Animation, consente di progettare animazioni ed effetti visivi. Le funzionalità di basso livello, come detto, sono definite nello strato Core OS e il loro accesso è garantito dalla libreria LibSystem. Di seguito sono elencate sommariamente le funzionalità:

- Threading.
- Networking.
- Gestione del File-system.
- Standard I/O.
- Bonjour e servizi DNS.
- Allocazione della memoria.
- Math computation.

Al livello più alto troviamo lo strato **Cocoa Touch** si compone dei framework UIKit e foundation (UIKit.framework e Foundation.framework). Questi

framework rendono disponibili gli strumenti necessari per la realizzazione della grafica. UIKit gestisce tutto quello che riguarda l'interazione con l'utente o la visualizzazione, come le classi legate alla gestione del multitouch e alla renderizzazione su schermo, oltre ad una serie di widget grafici legati all'UI. Foundation gestisce le classi legate alla gestione/manipolazione/storage dei dati, wrapper OO per alcuni dei livelli più bassi. Cocoa Touch è utilizzato per l'implementazione delle seguenti caratteristiche:

- application Management.
- Supporto alla grafica.
- Supporto alla gestione di eventi.
- Gestione dell'UI.
- Supporto per testo e contenuti web.
- Gestione dell'Accelerometro.
- Gestione della fotocamera.

Il processore di iPhone ed iPod Touch è un RISC ARM, come il SoC ³ usato nell'iPad è allo stesso modo di architettura ARM Cortex, a differenza del processore x86(e prima PowerPC o MC680x0) che viene comunemente usato nella linea Macintosh; le soluzioni ARM sfruttano OpenGL ES 1.1 e OpenGL ES 2.0 renderizzate da un processore video PowerVR. Le applicazioni per Mac OS X non possono essere ufficialmente copiate e lanciate in dispositivi con iPhone OS ma necessitano di essere customizzate e compilate specificatamente per iPhone OS e per l'architettura ARM; tuttavia, il web browser integrato Safari web browser supporta le "web applications". La prima versione dei rilasci di iPhone OS è stata diffusa il 9 gennaio 2007, senza il supporto per l'SDK e la creazione di applicazioni di terze parti.

³SoC (System on a Chip) è un termine che viene usato per indicare i circuiti integrati che in un solo chip contengono un intero sistema, o meglio, oltre al processore centrale, integrano anche un chipset ed eventualmente altri controller come quello per la memoria RAM, la circuiteria input/output o il sotto sistema video.

2.2.3 Applicazioni

L' **App Store** è un servizio realizzato da Apple disponibile per iPhone, iPod touch e iPad che permette agli utenti di scaricare e acquistare applicazioni disponibili in iTunes Store; le applicazioni possono essere sia gratuite che a pagamento, e possono essere scaricate direttamente dal dispositivo o su un computer. Le applicazioni possono esclusivamente essere vendute tramite l'iTunes Store per Mac e Windows oppure tramite l'applicazione "App Store" presente su iPhone, iPod touch, e iPad. Il 10 luglio 2008, il CEO Steve Jobs ha dichiarato che l'App Store conteneva 500 applicazioni di terze parti per iPhone e iPod touch, e il 25% di queste erano gratis; le applicazioni vanno dall'intrattenimento all'istruzione. Il 7 luglio 2011 le applicazioni disponibili sono ben 425000 con 15 miliardi di download.

A differenza delle applicazioni native presenti su iPhone, iPod touch e iPad, le applicazioni scaricate dall'App Store possono essere rimosse. Per prevenire gli abusi, le applicazioni devono essere preventivamente approvate da Apple. Il processo di approvazione prevede che due operatori provino in modo indipendente l'applicazione al fine di fornire un giudizio obiettivo e imparziale; le applicazioni possono essere rifiutate per una serie di motivazioni di carattere tecnico/commerciale. Applicazioni malfunzionanti o che fanno un eccessivo utilizzo delle connessioni telefoniche sono vietate, altresì sono vietate applicazioni pornografiche o che entrino in competizione diretta con i prodotti Apple. Il gruppo di approvazione è composto da quaranta operatori e ad agosto 2009 tra applicazioni nuove e aggiornamenti sono state verificate più di 200.000 applicazioni. Il 95% delle applicazioni viene testato in meno di due settimane e il 20% delle applicazioni viene rifiutato per i motivi esposti sopra.

2.2.4 Ambiente di sviluppo

Il 17 ottobre 2007, Steve Jobs ha annunciato che un SDK (software development kit) sarebbe stato disponibile agli sviluppatori di terze parti in

febbraio 2008. L'SDK è stato rilasciato il 6 marzo 2008 e permette agli sviluppatori di creare applicazioni per iPhone e iPod touch, e testarle in un simulatore di iPhone; tuttavia il caricamento di una applicazione nei dispositivi è possibile solamente dopo aver pagato una tassa di iscrizione all'iPhone Developer Program. L'ambiente di sviluppo per l'iPhone SDK è Xcode, che è lo stesso che consente lo sviluppo di applicazioni per Mac OS X. I linguaggi supportati da Xcode sono L'Objective C, l'AppleScript, il C++, l'Objective C++ e Java. Inoltre, l'ambiente Cocoa è utilizzabile anche con altri programmi di sviluppo e utilizzando anche linguaggi come il Perl, il Python (grazie al bridge PyObjC) e Ruby (grazie a RubyCocoa).

Gli sviluppatori sono liberi di scegliere qualsiasi prezzo per le loro applicazioni che sono distribuite tramite App Store, per le quali riceveranno il 70% del ricavo. Essi possono anche optare per rilasciare l'applicazione gratis e non pagheranno nessun costo di rilascio o distribuzione, eccetto la tassa di sottoscrizione al programma developer. Dato che l'iPhone è basato su una variante dello stesso XNU kernel che si trova in Mac OS X, i tools usati per lo sviluppo sono basati su Xcode. L'SDK è diviso nei seguenti set:

- Cocoa Touch
 - Multi-touch eventi e controlli
 - Accelerometer supporto
 - View gerarchica
 - Localizzazione (i18n)
 - Camera supporto

- Media
 - OpenAL
 - Audio mixing e recording
 - Video playback
 - Image file formats

- Quartz
- Core Animation
- OpenGL ES
- Core Services
 - Networking
 - Embedded SQLite database
 - GeoLocation
 - Threads
- OS X Kernel
 - TCP/IP
 - Sockets
 - Power management
 - File system
 - Security

All'interno dell'SDK è contenuto l'iPhone Simulator, un programma usato per emulare il "look and feel" dell'iPhone nel desktop dello sviluppatore; originariamente chiamato Aspen Simulator, è stato rinominato con la beta 2 dell'SDK. Da notare che l'iPhone Simulator non è un emulatore ed esegue codice generato per un target x86. L'SDK richiede un Mac Intel con Mac OS X Leopard, altri sistemi operativi, inclusi Microsoft Windows e vecchie versioni di Mac OS X, non sono supportati.

Xcode lavora in congiunzione con Interface Builder (proveniente da NeXT), un tool grafico per realizzare interfacce grafiche; include inoltre GCC, che è in grado di compilare codice C, C++, Objective C/C++ e Java; supporta ovviamente i framework Cocoa e Carbon, oltre ad altri. Una delle caratteristiche tecnologicamente più avanzate di Xcode è che supporta la distribuzione in rete del lavoro di compilazione; usando Bonjour e Xgrid è in grado di

compilare un progetto su più computer riducendo i tempi. Supporta la compilazione incrementale, è in grado di compilare il codice mentre viene scritto, in modo da ridurre il tempo di compilazione.

2.3 Symbian OS

Symbian OS è il sistema operativo prodotto da Symbian Foundation. Il consorzio Symbian nasce nel 1998, da un accordo fra alcune aziende che portò alla nascita della Symbian Ltd, allo scopo di produrre un sistema operativo per telefoni cellulari di fascia alta. I fondatori di Symbian sono: Nokia, Psion, Ericsson, Panasonic, Samsung, Siemens e Sony; nel giugno del 2008 Nokia ha comunicato l'intenzione di rilevare le quote azionarie delle altre società al fine di divenire l'unico proprietario del sistema operativo, e una volta completata l'acquisizione di renderlo open source con la creazione di Symbian Foundation: essa sarà formata dai vecchi proprietari e aperta ad altri produttori, e si dovrà occupare di unificare tutte le interfacce in una nuova release del sistema operativo e gestirne l'apertura agli sviluppatori esterni.

2.3.1 Versioni

Le ultime versioni, con codice aperto sono:

Symbian 1, essendo la prima release uscita nell'ottobre del 2008, costituisce la base per la piattaforma. Incorpora Symbian OS e S60 5th Edition (che si basa su Symbian OS 9.4) e quindi non è stato reso disponibile in open source.

Symbian 2, è stato il primo titolo gratuito la versione di Symbian, che venne rilasciato il 1 giugno del 2010. Alcune porzioni sono concesse in licenza EPL, ma la maggior parte del codice sorgente è sotto la licenza proprietaria SFL e disponibile solo ai membri della Symbian Foundation.

Symbian 3, rilasciato a settembre 2010. ha introdotto nuove caratteristiche come:

- una nuova architettura grafica 2D e 3D (Con questa architettura, possono essere aggiunte nuove transizioni ed effetti per l'interfaccia utente);
- rinnovamento architettonico in ambito networking;
- miglioramenti dell'interfaccia utente (Il metodo "unico tocco" è stato applicata all'interfaccia touch così gli utenti non hanno più bisogno di toccare una prima volta per selezionare e poi toccare di nuovo per l'azione);
- supporto per display esterno tramite HDMI;
- La radio e il gioco sono stati migliorati;
- Introduce le tre schermate personalizzabili;

Symbian Anna rilasciato da Nokia nel mese di aprile 2011, include miglioramenti come ad esempio un nuovo browser, una tastiera virtuale con orientamento verticale, nuove icone e lo scorrimento homescreen.

Symbian Belle rilasciata il 24 agosto 2011, è l'ultima versione della piattaforma Symbian, aggiunge una barra di notifica, un' integrazione più della comunicazione near-field, widgets homescreen a forma libera, e sei schermi home invece dei precedenti tre, la funzionalità NFC (che permette di condividere contenuti tra due dispositivi con un semplice "tocco" e di connettersi con accessori come cuffie e auricolari) e una navigazione in Rete migliorata.

2.3.2 Caratteristiche tecniche

Symbian è stato sviluppato con tali caratteristiche:

Microkernel EKA2 (EPOC Kernel Architecture 2). L'EKA2 migliora la precedente versione 1 introducendo l'approccio real-time (attraverso cui le chiamate a libreria sono più veloci e di durata limitata) e utilizzando nuovi protocolli per l'interazione fra processi, compilatore e memoria. Con la tecnica real-time la CPU è abilitata a processare anche il signalling stack del telefono: nell' EKA1 tale insieme di protocolli per le comunicazioni di rete era

gestito da una CPU dedicata. Tale kernel fornisce un importante vantaggio nella salvaguardia del tempo dell'utente e garantisce l'integrità e la salvaguardia dei dati utente; inoltre permette ai dispositivi di contenere le proprie dimensioni, di essere più economici e di migliorare la propria efficienza in termini di consumi.

La scelta di tale kernel consente di accedere ad altre prerogative quali **Multitasking** e **Multithreading**; in particolare si parla di preemptive multitasking in Symbian. In un sistema mono processore, con tale strategia si dà la possibilità allo scheduler (applicazione del sistema operativo che si occupa della gestione dei processi in esecuzione nella CPU) di eseguire context switch. Così facendo si condivide il tempo di esecuzione della macchina, dando l'idea che i processi siano compiuti contemporaneamente, come in un sistema multi-core. In verità in Symbian solo ed esclusivamente un processo viene eseguito ad ogni istante, essendo dedicato ad architetture single core, ma il context switch è svolto ad alte frequenze, dando l'idea della contemporaneità. Il multithreading è infine gestito tramite ISR ⁴. Sempre per il fatto che il sistema è single-core vengono utilizzati due tipi di interruzione (Immediate Function Call, per rispondere immediatamente alla richiesta, e Deferred Function Call, risposta in differita) per compiere lo scambio dei thread in esecuzione.

Active Object: è una forma di multitasking cooperativo attraverso la quale un task fa una chiamata a sistema; esso poi restituisce subito il controllo al processo, senza aver ultimato la sua richiesta. Il chiamante è libero di compiere altre azioni prima di tornare il controllo al sistema operativo; quando la richiesta è completata quest'ultimo identifica il thread richiedente

⁴Interrupt Service Routine (ISR) è una funzione informatica, di tipo callback, che viene avviata dal sistema operativo in risposta a un interrupt. Affinché il meccanismo dell'interruzione funzioni correttamente, è necessario che tutte le azioni svolte dall'ISR siano trasparenti rispetto al programma interrotto, cioè che al termine venga ripristinato tutto come era prima dell'interrupt. Per fare ciò bisogna che la CPU, prima di mandare in esecuzione l'ISR, faccia una commutazione di contesto, ovvero salvi tutto quello che stava facendo (cioè il suo contesto attuale), ed alla fine dell'ISR lo ripristini com'era.

e lo riattiva, ripassandogli il controllo. Tale tecnica permette di risparmiare parecchie risorse del dispositivo.

Protezione della memoria: è la capacità di controllare gli accessi alla memoria da parte di un processo, impedendogli ad esempio l'uso di uno spazio allocato per un altro task. È un servizio particolarmente importante per quanto riguarda l'integrità e la salvaguardia dei dati.

È una struttura **basata su eventi**. Il sistema operativo risponde e determina il proprio comportamento in relazione agli eventi: tali eventi possono essere output di sensori, azioni da parte dell'utente, messaggi dai thread. Tale approccio è in contrasto con il batch programming, nella quale il flusso del programma è imposto dall'utente.

Approccio **Request-and-Callback** ai servizi e separazione fra interfaccia utente e motore delle applicazioni. Per quanto riguarda il primo protocollo si tratta di una procedura con la quale un processo richiede un servizio ad una libreria attraverso una chiamata di sistema, ad esempio; per quanto concerne il secondo, si traduce nell'impossibilità da parte dell'utente di accedere ai servizi di basso livello del motore attraverso la UI (User Interface). Entrambe le caratteristiche permettono ancora una volta un'accelerazione nell'esecuzione dei processi.

Ottimizzazione per la salvaguardia dell'energia: tutti i dispositivi che utilizzano tale sistema operativo montano batterie a bassa potenza, nella quale piccoli accorgimenti per la riduzione dei consumi sono fondamentali. Per raggiungere tale scopo si converte la CPU (Central Processing Unit) in modalità "basso consumo" quando non ci sono applicazioni in esecuzione e si gestisce la retroilluminazione del dispositivo, ad esempio, attraverso alcuni software; inoltre, si gestisce anche l'accensione e lo spegnimento dei vari sensori quali accelerometri o gps, i quali assorbono una non trascurabile percentuale dell'energia della batteria.

Sistema XIP e rientro in librerie condivise. La prima caratteristica è l'acronimo di Execute In Place: con tale metodo si intende eseguire

un programma direttamente dalla memoria di storage piuttosto che copiarla nella RAM. Per farlo ci si serve di alcuni accorgimenti, che sono:

- un'interfaccia simile tra CPU e disco di storage, o perlomeno un layer d'interfacciamento che sia sufficientemente rapido nelle operazioni di lettura/scrittura;
- un file system (qualora dovesse essere usato) che disponga di un'appropriata funziona di mappatura;
- programmi "linkati", affinché possano facilmente essere trovati in memoria; non devono però modificare i dati caricati da quest'ultima.

Per quanto riguarda il rientro in librerie condivise, si allude ad un direttiva secondo la quale una libreria utilizzata di recente può quasi sicuramente essere riutilizzata in breve tempo; è quindi necessario che i tempi di rientro in tale struttura siano rapidi. Entrambe le prerogative permettono una riduzione dei tempi di esecuzione dei processi.

Organizzazione MVC (Model-View-Controller): si tratta di un pattern architetturale per lo sviluppo di interfacce grafiche orientate ad oggetti, ossia un algoritmo di risoluzione di un problema ricorrente, in questo caso, la UI orientata ad oggetti. Si basa sulla divisione dei processi di un'applicazione fra Business Logic (applicativi che permettono l'interazione fra applicazione e utente), input per i processi e presentazione dei dati elaborati. Per implementare questa configurazione ci si avvale di tre entità, da cui poi ne scaturisce il nome: Model, cioè un insieme di processi che permettono l'elaborazione dati; View, che adatta i dati all'interazione con lo user; e il Controller, che si occupa degli input e li risolve con varie chiamate al "Model". Con tale paradigma è permesso scindere lo sviluppo di uno stesso software in tre fasi diverse, testando o correggendo ognuna separatamente. Tutto ciò consente un progresso nello sviluppo software per tale piattaforma.

Descrittori: si occupano di gestire e conferire un comportamento prevedibile ai dati di tipo Stringa. In un qualsiasi sistema operativo, Symbian com-

preso, i dati di tipo Stringa possono rappresentare un problema per la loro amministrazione e per i loro comportamenti inaspettati. Evitano ad esempio problemi di **buffer overflows**⁵, mettendo a disposizione una dozzina di classi per la manipolazione “sicura” di tali tipi di dati. Questo è un esempio determinante di protezione delle scarse risorse disponibili.

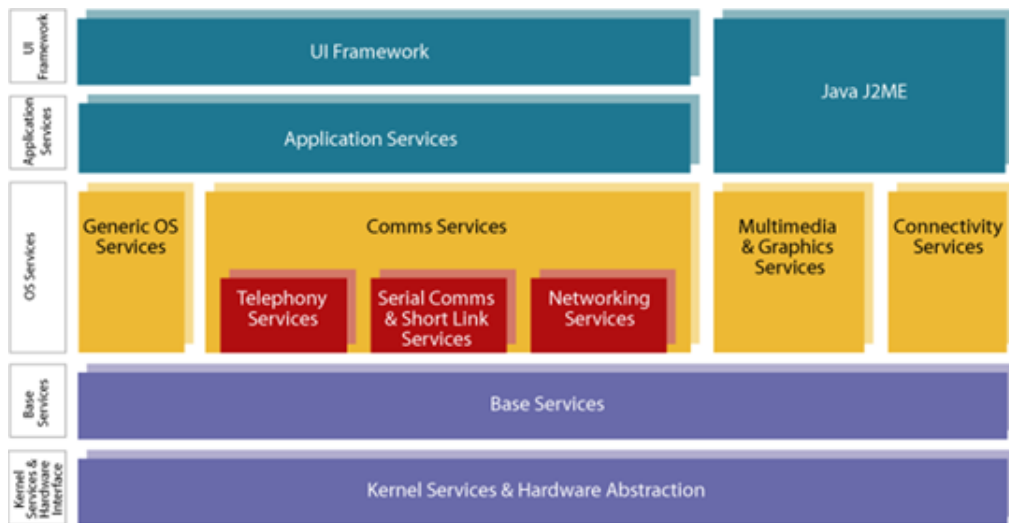


Figura 2.3: Architettura del sistema operativo Symbian

Cleanup Stack: provvede a ripulire lo stack dagli oggetti per la quale i metodi utilizzati hanno generato eccezioni. Essenzialmente, creando un oggetto, gli si alloca uno spazio in memoria, appunto nello stack degli oggetti. Quando si chiama un metodo su di esso, possono essere generate delle eccezioni per un suo comportamento scorretto, che provocano l'interruzione dell'esecuzione. Se lo stack dovesse rimanere carico di oggetti “inutili”, esso esaurirebbe il proprio spazio in poco tempo. Questo tipo di problematica

⁵Il buffer overflow è una vulnerabilità di sicurezza che può affliggere un programma software. Quando per errore o per malizia, vengono inviati più dati della capienza del buffer destinato a contenerli, i dati extra vanno a sovrascrivere le variabili interne del programma, o il suo stesso stack; come conseguenza di ciò, a seconda di cosa è stato sovrascritto e con quali valori, il programma può dare risultati errati o imprevedibili, bloccarsi, o bloccare il computer.

è chiamata **memory leak**. Tale classe si avvale di semplici metodi per la risoluzione di questo problema, migliorando ancora una volta l'utilizzo delle scarse risorse di cui dispone il dispositivo. Si deve tenere conto che nella maggior parte dei casi la memoria è di tipo Flash; non se ne può quindi disporre in grande quantità visti i limiti di tale tecnologia.

Demand Paging: è un'applicazione di memoria virtuale. Permette di gestire meglio il caricamento dati dalla memoria, imponendo che una disk page venga caricata in memoria solo se esiste un effettivo tentativo di accedere ad essa. Il risultato è che l'avvio del processo è velocizzato ed inoltre si evitano svariati errori del tipo page fault.

2.3.3 Applicazioni

Per Symbian OS è disponibile un discreto numero di programmi, sia gratuiti che a pagamento, il che ne fa un prodotto espandibile e personalizzabile. La piattaforma che permette di scaricare nuove applicazioni per i dispositivi symbian si chiama **Ovi Store**. L'Ovi Store è stato lanciato in tutto il mondo nel maggio 2009; qui, i clienti possono scaricare giochi per cellulari, applicazioni, video, immagini e toni di chiamata per i propri dispositivi Nokia. Dall'Ovi Store è possibile scaricare i seguenti contenuti:

- I consigliati da Nokia
- Applicazioni
- Giochi
- Contenuti Audio e Video
- I contenuti di Personalizzazione
- Infine è possibile vedere tutti i vari contenuti acquistati

Per gli sviluppatori, Nokia offre uno strumento self-service per portare i loro contenuti sul Ovi Store. I tipi di contenuto supportati comprendono: Java

ME, applicazioni Flash, widget, suonerie, sfondi, temi e molto altro ancora per Nokia Series 40 e S60 e Symbian 3. Nokia offre una quota del 70% del fatturato lordo, al netto dei rimborsi. Il numero giornaliero dei download ha raggiunto 9.000.000 nel mese di agosto 2011; nonostante la recente decisione di Nokia di utilizzare Windows Phone 7 come sistema operativo primario, Ovi Store sarà ancora disponibile per il presente e il futuro dei telefoni Symbian, mentre Ovi Store e Windows Phone 7 Marketplace verranno uniti sulla piattaforma WP 7. Ci sono 50.000 applicazioni a partire da aprile 2011. Le applicazioni per la versione 3.0 ossia symbian 3, sono compatibili, o parzialmente compatibili con Symbian 1.

2.3.4 Ambiente di sviluppo

Esistono svariati linguaggi per l'implementazione di software per Symbian, ma i principali sono due:

Symbian C++ - È il linguaggio nativo del sistema operativo Symbian, pur non rappresentando lo standard d'implementazione. Oltre alle librerie tipiche di C++, supporta anche lo standard Open C (software tool per l'analisi e il debug di applicazioni in C++), senza il bisogno di installare estensioni alla piattaforma grazie alle API dedicate. Per realizzare un'applicazione per Symbian in tale linguaggio sono necessari alcuni prerequisiti, da installare nel proprio PC:

- SDK (Software Development Kit), utile strumento per lo sviluppo e il test dell'applicazione in C++. Oltre a una svariata collezione di API, librerie, documentazione ed esempi, prevede anche un GCCE (Gnu C Compiler Embedded), un compilatore dedicato alla costruzione di programmi per dispositivi mobili.
- IDE (Integreated Development Enviroment) è letteralmente un ambiente di sviluppo integrato, costituito da una famiglia di software, destinato a facilitare la creazione dell'applicazione. Un esempio ne è Carbide C++, sistema basato su Eclipse che permette di utilizzare un'interfac-

cia semplice e user-friendly che segua il programmatore in ogni fase dello sviluppo.

- Compilatore, un programma dedicato generalmente alla traduzione di un programma sorgente, scritto in codice di alto livello, in un linguaggio binario.
- Command line tool, configurazioni attraverso la quale è possibile fare a meno dell'IDE e seguire lo sviluppo dell'applicazione direttamente da riga di comando, eseguendo particolari procedure in background.

Il risultato della creazione di un'applicazione definita in Symbian C++ è un pacchetto di files, caratterizzati dall'estensione .sis, destinati ad essere installati nel dispositivo.

Qt - Dal 2010, Symbian è passato da usare lo standard C++ (Symbian C++) al linguaggio Qt come l'SDK. Qt è una libreria multiplatforma che isola l'utente il più possibile dalle differenze dei vari sistemi operativi. Servendosi di Qt, l'utente può costruire un'applicazione una volta per tutte, utilizzandola poi su diversi desktop e sistemi operativi. Lo standard seguito da tale linguaggio deriva dal C++: ciò significa che per l'utente che già conosce tale sintassi è sufficiente approfondire pochi dettagli per poter sfruttare questa potente multiplatforma; può essere sviluppato sia con Carbide C++ che con Qt Creator.

I dispositivi Symbian possono anche essere programmati usando i linguaggi: Python, Java ME, Flash Lite, Ruby, .NET, Web Runtime Environment (WRT) e Standard C/C++.

2.4 Windows Mobile

Windows Mobile, è un sistema operativo compatto basato sulle API Win32 di Microsoft. Inizialmente sviluppato per i PDA, chiamati Pocket

PC, la sua ultima versione per smartphone è Windows Phone 7.5 (Mango). Windows Phone 7 o conosciuto come WP 7, è il sistema operativo per smartphone di Microsoft, che lo ha presentato al Mobile World Congress il 15 febbraio 2010. Si rivolge al mercato consumer (invece del mercato enterprise come il suo predecessore) eliminando molte delle funzionalità in dotazione di Windows Mobile.

2.4.1 Versioni

Il precedente sistema operativo per cellulari di Microsoft è Windows Mobile 6 (con aggiornamenti a Windows Mobile 6.1, 6.5 e 6.5.3). WP 7 è stato sviluppato completamente da zero ed è completamente differente da tutte le precedenti versioni di Windows Mobile; la compatibilità all'indietro con WP 7 è un problema, infatti gran parte dei telefoni attualmente in esecuzione Windows Mobile 6.x non sono aggiornabili a Windows Phone 7. Le sue caratteristiche principali sono:

- Supporta gli schermi capacitivi multitouch;
- Nuova interfaccia grafica molto simile a quella di Zune HD;
- Interfaccia utente di nuova concezione, con “tiles” (piastrelle) invece dei widget nella schermata iniziale (i Tile sono collegamenti ad applicazioni, funzioni oppure oggetti individuali dinamici e si aggiornano in tempo reale - per esempio, il tile di un account email mostrerà il numero di messaggi non letti, o un Tile può mostrare un aggiornamento costante del meteo o il numero dei messaggi non letti);
- Supporta 5 lingue inglese, francese, italiano, tedesco, spagnolo;
- Riunisce in una sola piattaforma i contenuti di Xbox Live (copn cui è anche possibile accedere simultaneamente su console e telefono e inviare e ricevere messaggi tra console e telefono) e Zune (applicazione che si occupa dell'intrattenimento e della sincronizzazione tra PC e telefono;

- L'esperienza multimediale del Windows Phone 7 è divisa in due distinti "hub", il Music + Videos hub, che si occupa della riproduzione della musica, dei video e podcasts, e permette agli utenti di accedere al Zune Marketplace per comprare musica o per affittarla, ed il Pictures hub, mostra le foto di Facebook e di Windows Live insieme alle foto effettuate dal telefono ; entrambi assomigliano nella grafica e nelle funzionalità allo Zune HD);
- Gestisce gli account di social network quali Facebook e Twitter;
- Possiede una nuova versione di Internet Explorer basata su Windows Internet Explorer 7 con alcuni elementi della versione 8;
- Contiene un'edizione di Mobile Office 2010, con Word, Excel, Powerpoint, OneNote e Sharepoint;

Microsoft a fine settembre 2011 ha rilasciato la nuova versione **Windows Phone 7.5 Mango**, tra le maggiori caratteristiche introdotte riportiamo:

- Multitasking (che consentirà di passare rapidamente da un'applicazione all'altra e di eseguirle in background. Questa funzione dovrebbe implicare anche un sistema di push notification, soprattutto per quelle applicazioni che si occupano di instant messaging);
- Internet Explorer 9 con supporto HTML5, CSS3 e Javascript (flash e Silverlight non sono ancora compatibili) e accelerazione hardware ;
- Aggiunta dei nuovi hub People (spazio dedicato ai contatti che aggrega e riordina tutte le informazioni relative a una persona, sia le informazioni più elementari come mail e numero di telefono, sia le attività sui social network e le fotografie) e Messaggi (integrazione tra tutti gli strumenti di messaging);
- Supporto a Twitter, LinkedIn e Facebook con l'integrazione direttamente nei contatti;

- Funzionalità di conversione della voce in testo e del testo in voce;
- Le Live Tile, che già alcune applicazioni supportano. Grazie ai "riquadri animati" sarà possibile ricevere dalle applicazioni informazioni in tempo reale, senza la necessità di aprirle;
- Estensione dei servizi Zune alle diverse piattaforme tramite cloud con Skydrive;
- Tethering WiFi che supporta fino a 5 dispositivi (attualmente ci sono anche alcune limitazioni hardware relative al tethering; Microsoft assicura che gli smartphone dotati di chip radio Broadcom 4329, dopo l'aggiornamento a Mango, potranno sfruttare il tethering, ma non quelli basati sul chip Broadcom 4325; inoltre la funzione può essere bloccata o resa a pagamento da alcuni operatori telefonici);
- L'accesso al marketplace per Windows Phone completamente basato sul Web, nel quale si potranno cercare le proprie app preferite senza confusione o intrusioni da parte di altri software dedicati, ad esempio, Zune.

2.4.2 Caratteristiche tecniche

Windows Phone 7 è basato sulla versione del kernel Windows Embedded CE 6.0 R3 con alcune caratteristiche prese in prestito da Windows Embedded Compact 7, rendendo così una soluzione ibrida. Una delle evoluzioni più grandi e più importanti rispetto alla versione precedente di Windows Mobile (che si basavano sul kernel Windows CE 5) è il fatto che il SO ora possa sfruttare appieno l'architettura CPU ARMv7 e anche l'introduzione di un'interfaccia utente con accelerazione GPU basata su una nuova versione di Mobile Direct3D.

È un sistema operativo **Hard Real Time** a 32 bit e modulare (700 componenti) pensato per sistemi embedded (no swap di memoria su disco:

memoria limitata da quella fisica); supporta il **multitasking** e gira su architetture di processori diversi, tra cui ARM, MIPS, x86 e SH4, operando nello spazio di indirizzamento virtuale di 4 gigabyte.

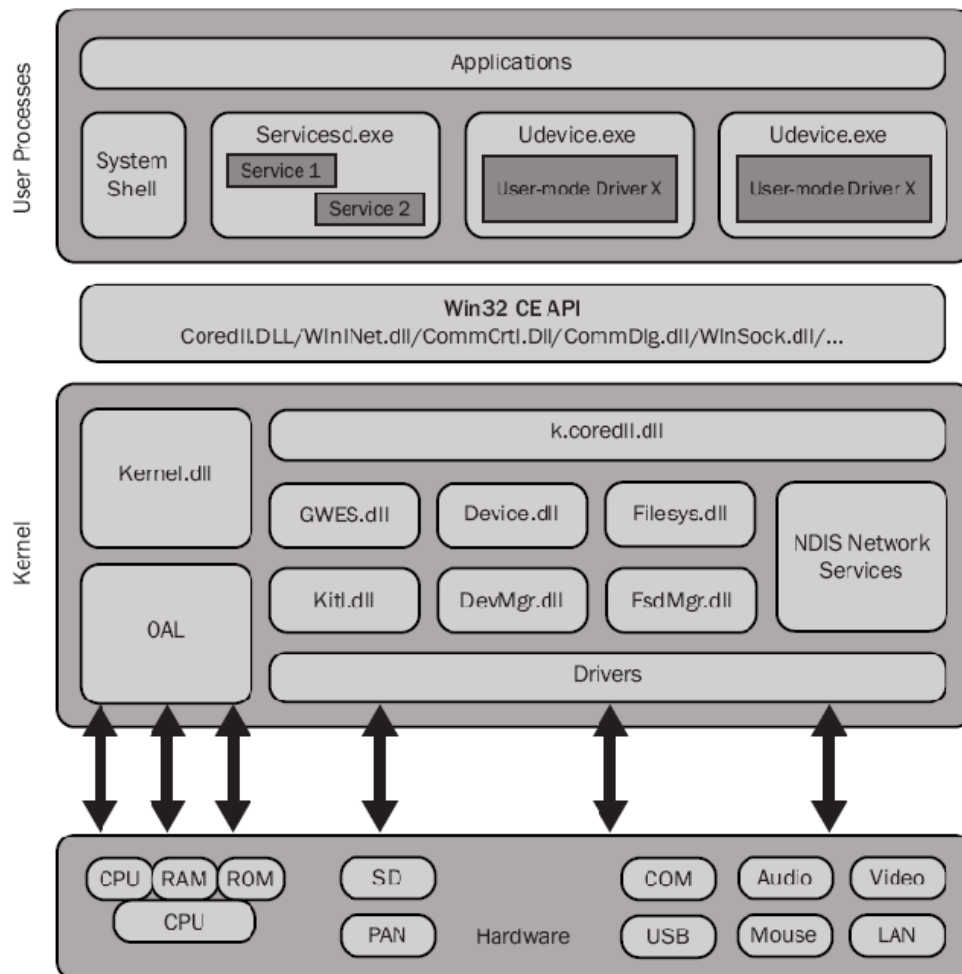


Figura 2.4: Architettura del sistema operativo Windows CE 6.0

Il **kernel** di sistema utilizza i 2 GB di memoria virtuale superiore (contro i 32MB della versione precedente), mentre i restanti vengono impiegati per il processo del utente attivo; supporta fino a 32768 processi utente (in crescita rispetto al limite della precedente versione di 32 processi), e tale numero effettivo viene limitato dalle risorse di sistema. I processi utente includono

speciali sottoprocessi che rendono le interfacce di programmazione (API) disponibili per le applicazioni utente; la shell del sistema rende l'interfaccia principale a disposizione del utente.

Il nucleo del sistema operativo è il processo Nk.exe, nel quale sono state caricate le librerie dinamiche responsabili di varie funzionalità di sistema; il sistema API è disponibile per tutte le applicazioni attraverso la libreria core.dll, che è collegata a tutti i moduli eseguibili del sistema operativo. Il kernel di Windows Embedded CE 6.0 interagisce con l'hardware attraverso la OAL (parte del sistema operativo che lo rende adattabile all'hardware, indipendentemente dalla piattaforma), che nasconde l'implementazione del processore ed esegue l'inizializzazione del hardware. La costruzione è basata sulla memoria virtuale, che viene fornita dal sistema operativo in modo flessibile ed efficace per gestire le limitate risorse di memoria fisica. L'architettura della memoria virtuale è una mappatura degli indirizzi di memoria virtuale in indirizzi fisici.

I file system sono molto simili ai sistemi UNIX: hanno una root unica, mount dei device con nome e non con lettere, RAM come ObjectStore; gestiscono supporti di memorizzazione e file di dimensioni maggiori (fino a 4 GB) e la crittografia dei media removibili. Tra le caratteristiche principali possiamo citare:

Silverlight per Windows Embedded. Il potere di Silverlight portato a Windows Embedded CE per creare applicazioni ricche e interfacce utente.

Internet Explorer integrato. Internet Explorer con multi-touch capacità di panning e zoom e un'interfaccia personalizzabile per ottimizzare l'esperienza di navigazione su dispositivi Touch. Plug-in per abilitare funzionalità di input naturali come il multi-touch a 4 punti, movimenti avanzati e complesse animazioni 3D.

Connection Manager. Infrastruttura tecnologica per gestire le interfacce di rete sul dispositivo.

Microsoft Office e PDF Viewer. Applicazioni per visualizzare i documenti Microsoft Office Word, PowerPoint, Excel e Adobe PDF sul dispositivo.

Connettività. Windows Embedded CE 6.0 R3 estende le funzionalità degli attuali dispositivi CE di collegarsi facilmente a una rete Internet e rimanere in contatto. I dispositivi hanno più metodi di connessione sia LAN, WiFi, GPRS, Bluetooth o USB. Il Connection Manager consente lo sviluppo di applicazioni di connessione molto semplici con una sola API per richiedere una connessione di rete. Anche un'applicazione può sottoscrivere gli eventi dello stato della connessione per aiutare a gestire la qualità della connessione.

Web Services on Devices API (WSDAPI). Costruire rapidamente applicazioni di servizi Web per interagire con i dispositivi collegati in rete. WSDAPI offre un ricco scambio di metadati e la messaggistica, oltre a un supporto integrato per la sicurezza e la gestione degli eventi.

Remote Desktop Protocol. Abilita un piccolo client basato su Windows Embedded CE connettendosi utilizzando l'ultima versione di Windows Server permettendo il controllo remoto.

Driver. Produzione di qualità dei driver delle periferiche aiutano a diminuire la quantità di modifiche necessarie per i driver di lavorare con hardware personalizzato; il supporto driver migliorato contribuisce a garantire una facile portabilità. I driver possono essere eseguiti in modalità kernel per le prestazioni o in modalità utente per la robustezza.

Caratteristiche di sicurezza. Funzioni di primo livello di sicurezza, conforme all' SDL (Security Development Lifecycle) e contribuisce a garantire che le applicazioni solo autenticate possano essere eseguite su un dispositivo embedded.

Microsoft ha deciso di predefinire delle specifiche hardware che gli OEM⁶ dovranno utilizzare per costruire il loro dispositivi WP7, una delle specifiche più importanti è l'uso di un singolo SoC (System on a Chip) la Qualcomm QSD8250 e la sua GPU Adreno 200. Ciò significa che, contrariamente alle versioni precedenti di Windows Mobile e ad Android, Microsoft vuole sviluppare e testare il sistema operativo per una sola singola architettura, riducendo notevolmente il rischio di frammentazione hardware nell'ecosistema e garantire che il prodotto sarà perfettamente ottimizzato ed eseguito il più velocemente possibile su ogni singolo dispositivo. Per dirla semplicemente: WP7 è il perfetto mix tra la vasta selezione di Android di fattori di forma e lo stretto controllo di Apple OS sull'integrazione hardware. Microsoft controlla essenzialmente quasi tutto questa volta, dalla risoluzione minima dello schermo, alla quantità di ram e la velocità dell'interfaccia e se un OEM non è conforme a queste linee guida non può lanciare un dispositivo sul mercato.

Windows Phone 7 utilizza una tecnologia multi-touch; l'interfaccia utente di default ha un tema scuro che prolunga la durata della batteria sui monitor OLED perché i pixel completamente neri non emettono luce. L'utente può comunque scegliere un tema chiaro al suo posto, e può anche scegliere tra tanti colori accesi.

Partnership con Nokia

L'11 febbraio del 2011, durante una conferenza stampa a Londra, l'amministratore delegato della Microsoft Steve Ballmer e l'amministratore delegato della Nokia Stephen Elop hanno annunciato una partnership tra le loro aziende nella quale Windows Phone diventerà il principale sistema operativo della Nokia. L'evento era principalmente focalizzato sulla creazione di "un nuovo ecosistema globale", affiancando competitor come Android e iOS. L'in-

⁶Original Equipment Manufacturer, in acronimo OEM, significa letteralmente produttore di apparecchiature iniziale, si utilizza nel contesto dei processi produttivi industriali. La società che ha originariamente prodotto un componente o un prodotto finito viene detta OEM.

tegrazione dei servizi Microsoft con i servizi della Nokia è stata annunciata, Bing sarà il motore di ricerca per i telefoni Nokia e sarà presente un'integrazione tra Nokia Maps con Bing Maps così come l'OVI della Nokia sarà integrato con il Windows Phone Marketplace. La partnership include "passaggi di fondi per royalties, il marketing e la condivisione", che la Microsoft ha annunciato successivamente essere, "misurato in miliardi di dollari".

2.4.3 Applicazioni

Il servizio offerto da Microsoft per permettere agli utenti di scaricare le applicazioni sviluppate da terze parti si chiama **Windows Phone Marketplace**; l'interfaccia utente è presentata in una "visione panoramica" dove l'utente può sfogliare le categorie e titoli, visualizzare gli elementi presenti, e ottenere i dettagli con le valutazioni, recensioni, screenshot e informazioni sui prezzi. Il Marketplace di Windows Phone è stato lanciato insieme a Windows Phone 7 nell'ottobre 2010 in alcuni paesi. Con il lancio di Mango (Windows Phone 7.5) l'accesso al marketplace è completamente basato sul Web, senza intrusioni da parte di altri software dedicati, ad esempio Zune. Gli sviluppatori registrati a Windows Phone 7 ed Xbox Live possono inserire e modificare le loro applicazioni per la piattaforma attraverso l'applicazione online App Hub. L'App Hub fornisce strumenti di sviluppo e di supporto per gli sviluppatori di applicazioni; le applicazioni inserite devono sottostare all'approvazione e la validazione di Microsoft. Il costo delle applicazioni approvate dipende dallo sviluppatore, ma a Microsoft spetta il 30% del guadagno (il restante 70% va agli sviluppatori). Microsoft paga gli sviluppatori solo al raggiungimento di determinati limiti, e trattiene il 30% di tasse per sviluppatori non statunitensi, a meno che non si siano già registrati con il servizio fiscale degli Stati Uniti. Microsoft paga soltanto sviluppatori provenienti da una lista di trenta paesi. Per far parte della lista ufficiale gli sviluppatori devono pagare una tassa annuale. Per far apparire un'applicazione sul Windows Phone Marketplace, l'applicazione deve essere inviata alla Microsoft per

l'approvazione. La Microsoft ha inoltre delineato i contenuti che non saranno approvati, tra questi i software a sfondo sessuale.

2.4.4 Ambiente di sviluppo

Applicazioni e giochi per WP7 sono basati su XNA o su specifiche versioni di Microsoft Silverlight.

Microsoft XNA (XNA is Not an Acronym) è un framework realizzato da Microsoft specifico per lo sviluppo di videogiochi e applicazioni su più piattaforme: PC, Xbox 360 e Windows Phone 7. Esso si occupa di fornire una serie di strumenti (sia utility che classi) atti a semplificare notevolmente lo sviluppo di videogiochi sia in 2D che in 3D. XNA ha l'abilità infatti di coprire molti degli aspetti che normalmente bisogna affrontare durante lo sviluppo di un videogioco, ad esempio offre un'implementazione integrata per la gestione della grafica, del suono e dell'input. Uno degli aspetti in cui XNA mostra tutta la sua potenza, è la rapidità con cui un progetto possa essere convertito e portato su uno qualsiasi degli ambienti supportati. Basato su DirectX 9 e .NET Framework, esso si sviluppa in C# con l'utilizzo dell'IDE Visual Studio. Un'altra caratteristica molto apprezzata di XNA è la sua possibilità di poter lavorare sia ad alto livello che a basso livello, a discrezione dello sviluppatore.

Microsoft offre il **Windows Phone SDK** (precedentemente denominato Windows Phone Developer Tools), un pacchetto software gratuito per sviluppare applicazioni per Windows Phone 7 che contiene:

- Visual Studio 2010 Express e Expression Blend per Windows Phone (versione di sviluppo per telefoni di applicazioni maggiori)
- Windows Phone Emulator
- Silverlight per Windows Phone
- Microsoft Expression Blend per Windows Phone
- XNA Game Studio 4.0

La nuova versione non modifica direttamente il framework di XNA, che rimane dunque alla versione 4.0. Ciò nonostante, le nuove feature del sistema operativo Windows Phone mettono a disposizione dello sviluppatore alcune interessanti possibilità, come: l'integrazione con Silverlight, nuovi sensori come la bussola e il giroscopio e nuove API per analizzare movimenti combinati (inclinazione, accelerazione e rotazione).

2.5 BlackBerry OS

BlackBerry OS è un sistema operativo mobile proprietario, sviluppato da Research In Motion per la sua linea di smartphone BlackBerry.

2.5.1 Versioni

BlackBerry OS 5.0 (introdotto nel 2009) presenta le seguenti funzionalità:

- Browser web BlackBerry aggiornato con un render accurato di AJAX, l'elaborazione del JavaScript più veloce, tra cui il supporto widget di Google Gears e SQLite per Widgets BlackBerry;
- Per il servizio Push si ha la gestione dei messaggi e-mail, la gestione delle cartelle e-mail, la possibilità di visualizzare, modificare, salvare le stesse, inviare e ricevere e-mail con allegati documenti da condivisioni di file remoti;
- Vengono introdotte molte animazioni assieme alla nuova interfaccia grafica;
- Calendario migliorato con visualizzazione degli allegati nelle voci del calendario;
- Possibilità di trovare, aprire, visualizzare, modificare, salvare o inviare per e-mail file JPEG, PDF, MS Word, MS Excel e MS PowerPoint direttamente dal telefono;

- Sincronizzazione dei contatti wireless in cartelle di contatti multipli e liste di distribuzione personali nel desktop degli utenti;
- Supporto per molte delle funzioni di Gmail come la ricerca, etichette, stelle, archiviare, visualizzare conversazione e spam report;
- Nuova gestione dei messaggi SMS, che rendono la lettura più esemplificata (in stile chat) e durante la scrittura delle stesse si ha a disposizione una nuova finestra per l'inserimento delle emoticon;
- BlackBerry Maps - Zoom disponibile per guardare da vicino, calcolo del percorso più veloce, la rilevazione automatica degli indirizzi (da e-mail o pagine web), Punto di interesse e la sua valutazione in mora mappe, navigazione, Geotagging foto;

Ad aprile 2010 RIM ha annunciato il nuovo **BlackBerry OS versione 6.0**, che è stato rilasciato nel terzo trimestre 2010; con questa nuova versione, si sono introdotte le seguenti modifiche:

- Migliorata la navigazione internet, sia sotto il punto di vista della compatibilità che della velocità di navigazione, grazie al passaggio al motore di render WebKit, (Navigazione a schede, feed RSS migliorati, vista Made Easy, ovvero lo Zoom dei contenuti è facile e perfetto sia con l'introduzione di multi touch nei modelli touch screen, sia nei modelli normali);
- Nuova interfaccia grafica (schermata iniziale personalizzata con possibilità di aggiungere altre voci di menu);
- L'introduzione di due aree di accesso rapido: una per gestire le connessioni, gli allarmi e le schermate di opzione; e l'altra per consentire l'accesso ai messaggi più recenti come email, SMS, BBM (Blackberry Messenger), telefonate, appuntamenti e le notifiche facebook e twitter;
- Ricerca universale (Universal Search) per effettuare ricerche all'interno del portatile così come ricerche web;

- Introduzione di menu popup sensibili al contesto operativo;
- Social Feeds, che permette di gestire in unica soluzione i contenuti provenienti da tutti i principali social network, tra cui anche posta elettronica e SMS;
- Media player avanzato;

Le voci di un sistema operativo serie 6.1 sono iniziate a circolare nel 2011, insieme alle immagini del nuovo software. Il 2 maggio 2011, RIM ha annunciato in seguito tale sistema come il **BlackBerry OS 7**. RIM ha annunciato che i dispositivi attuali non possono essere aggiornati con il BlackBerry OS 7 (Nessun supporto per dispositivi attuali) esso funzionerà sui dispositivi touch BlackBerry Bold; questo nuovo sistema introduce:

- nuovo browser che utilizza il compilatore Javascript JIT (Just in Time) in grado di migliorare la velocità di caricamento delle pagine; inoltre è stato esteso il supporto ad elementi aggiuntivi in HTML5 come i video codificati con tale standard;
- BlackBerry Balance. Grazie al quale è possibile tenere separate le informazioni legate alle proprie attività personali, da quelle concernenti il proprio lavoro. Le aziende potranno poi impostare delle specifiche policy di sicurezza tramite BlackBerry Enterprise Server 5.0.3 e BlackBerry;
- Enterprise Server Express. Impedisce, ad esempio la diffusione di informazioni concernenti il lavoro del dipendente tramite strumenti tipicamente afferenti alla sfera personale (Facebook, Twitter, etc);
- Funzioni di ricerca migliorate. La funzione Universal Search, già introdotta da BlackBerry OS 6 verrà ulteriormente migliorata con il supporto delle ricerche ad attivazione vocale;
- Supporto alla grafica Open GL 2.0. Già con il BlackBerry Bold 9900 si inizia ad apprezzare l'impegno di RIM finalizzato a potenziare il

comparto hardware dei dispositivi di prossima uscita. Per sfruttarlo al meglio, BlackBerry OS 7 consentirà agli sviluppatori di utilizzare le librerie grafiche Open GL 2.0;

- Realtà Aumentata e supporto alla tecnologia NFC. Verranno incluse nuove API che consentiranno agli sviluppatori di realizzare applicazioni di realtà aumentata di sfruttare il chip NFC integrato nei nuovi smartphone;
- Docs to go in versione premium. La suite per la redazione di documenti in formato Office disponibile con le funzioni complete sarà preinstallata nei BlackBerry 9900.

2.5.2 Caratteristiche tecniche

BlackBerry OS è sviluppato nel linguaggio di programmazione C ++ ed è programmato per svolgere più attività contemporaneamente, fornisce multitasking. Esso utilizza diversi dispositivi di input specializzati che sono stati adottati da RIM come la rotella di scorrimento (la trackball), il trackpad e touchscreen che sono presenti sul telefono.

Ha un kernel basato su Java e utilizza un'architettura ARM con un processore Intel XScale ARM e un set di istruzioni con architettura RISC; e utilizza dei registri a 16 x 32-bit. Il BlackBerry supporta il multitasking si possono quindi eseguire più applicazioni contemporaneamente svolgendo alcune attività in background.

La Java Virtual Machine si occupa della gestione della memoria sul BlackBerry per la maggior parte delle applicazioni di terze parti, a meno che venga utilizzata la specializzata MDS runtime. La JVM gestisce l'allocazione della memoria, lo scambio di dati tra la SRAM ⁷ e la Garbage Collection ⁸. Per

⁷La SRAM (Static Random Access Memory), è un tipo di RAM volatile che non necessita di refresh. I banchi di memorie SRAM consentono di mantenere le informazioni per un tempo teoricamente infinito, hanno bassi tempi di lettura e bassi consumi, specialmente in condizioni statiche.

⁸Per garbage collection si intende una modalità automatica di gestione della memoria,

affrontare la limitata capacità di memoria degli smartphone è in esecuzione un gestore speciale della poca memoria; quando la quantità di memoria libera scende al di sotto di una certa soglia il gestore di memoria tenta di liberare la memoria esistente. Applicazioni standard così come le applicazioni di terze parti dovrebbero funzionare con questa interfaccia, e cercare di cancellare i dati priorità bassa e media quando l'ascoltatore con poca memoria riceve un evento.

La piattaforma BlackBerry è forse più nota per il suo supporto nativo della posta elettronica aziendale attraverso MIDP 1.0⁹ e, più recentemente, un sottoinsieme di MIDP 2.0, che consente di completare l'attivazione wireless e la sincronizzazione con Microsoft Exchange, Lotus Domino o Novell GroupWise, per la sincronizzazione di e-mail, calendario, attività, note e contatti (necessità l'attivazione del BlackBerry Enterprise Server). Queste funzioni sono svolte da diversi sistemi:

Il **BlackBerry MDS** (Mobile Data System), è il sistema che permette l'accesso al **BES** (BlackBerry Enterprise Server). Si tratta di uno dei servizi più importanti messi a disposizione da BlackBerry, che consente ad un'azienda di sincronizzare i propri server di posta o di applicazioni con i cellulari in dotazione ai dipendenti, i quali possono ricevere nella propria casella di posta le email tramite un canale wireless sicuro. Oltre alla posta elettronica è possibile sincronizzare i dispositivi mobili con dati PIM personali e modiche a calendari, attività, rubriche. Il MDS è il canale utilizzato dal device per collegarsi al BES: nella prima parte, attraverso una qualsiasi connessione ad internet 3G o WiFi, si connette alla BlackBerry Infrastructure, anche

mediante la quale un sistema operativo, libera le porzioni di memoria che non dovranno più essere successivamente utilizzate dalle applicazioni. In altre parole, il garbage collector annoterà le aree di memoria che non sono più allocate da un processo attivo, e le libererà automaticamente.

⁹MIDP (Mobile Information Device Platform) in combinazione con CLDC (Connected Limited Device Configuration) costituiscono l'ambiente di runtime per le applicazioni Java, su dispositivi come PDA e telefoni cellulari. MIDP provvede alle funzionalità di base necessarie alle applicazioni mobili quali; l'interfaccia utente, la connessione di rete, la gestione dei dati sul sistema locale e la gestione del ciclo di vita del programma.

conosciuta come NOC (Network Operation Center). Da qui i dati vengono inviati al BES che è in genere protetto dall'azienda tramite un firewall. Tutta la comunicazione dal device al BES attraverso il MDS è protetta mediante cifrari a chiave privata. MDS è anche un proxy HTTP e TCP sicuro che permette ad applicazioni sviluppate da terze parti in Java ed al Browser BlackBerry di comunicare con le applicazioni ed i web server aziendali. Si noti che, una volta richiesta la connessione al server BES, il device seleziona automaticamente la rete wireless sulla quale fare viaggiare i dati, in genere quella che richiede il costo minore in termini di consumo energetico, ovvero la rete WiFi. Il passaggio attraverso il NOC è evitabile utilizzando un access point aziendale, ed eventualmente una rete VPN.

Se il servizio BES è rivolto a grandi aziende, per utenti consumer o piccoli gruppi RIM mette a disposizione il **BIS** (BlackBerry Internet Service), che consente di effettuare una sincronizzazione delle email senza un apposito server (e quindi senza i servizi annessi), ma facendo svolgere lo stesso ruolo ad un operatore telefonico che, previa registrazione dell'utente, accede tramite IMAP o POP3 alla user mailbox ed invia in modalità push le email ricevute al dispositivo BlackBerry dell'utente. Il percorso del BIS è analogo a quello del MDS, ma senza l'accesso al BES, ed a server di applicazioni aziendali. Il BIS risulta essere un servizio limitato, in quanto non offre le funzionalità di sicurezza del BES e di solito nemmeno la possibilità di sincronizzare altro se non email, dati i limiti dei protocolli POP3 e IMAP usati dall'operatore. Come per il MDS, la connessione wireless per le connessioni BIS viene selezionata dal dispositivo, dando la precedenza all'uso di reti WiFi.

Per quel che concerne le connessioni WAP a gestione è dipendente dal carrier di rete che trasmette i dati dai server Web al dispositivo senza utilizzare infrastrutture o servizi RIM e lo stesso si può dire delle connessioni TCP Direct. Per avere prestazioni migliori, conviene utilizzare connessioni TCP tramite reti WiFi, accedendo ad Internet da un access point qualunque o anche da uno aziendale per comunicare con applicazioni predisposte.

2.5.3 Applicazioni

Il market delle applicazioni per il sistema operativo di RIM si chiama **BlackBerry App World**, questo è il nome del negozio virtuale che si è da poco (settembre 2011) aggiornato alla versione 3.0.

Il servizio offre agli utenti di BlackBerry, un ambiente per navigare, scaricare e aggiornare le applicazioni sviluppate da terze parti; il servizio è andato online il 1 aprile 2009. Dei tre principali fornitori di applicazioni, è quello con più fatturato per app a \$ 9,166.67 rispetto ai \$ 6,480.00 dell' Apple App Store e ai \$ 1,200.00 dell' Android Market.

RIM annunciò inizialmente che il negozio sarebbe stato disponibile solo negli Stati Uniti, Regno Unito e in Canada, a partire dal 9 Settembre 2010, BlackBerry App World è stato reso disponibile in 113 Paesi e accetta pagamenti in tutti i 113 utilizzando una combinazione di PayPal, carta di credito e il supporto di fatturazione. Le applicazioni sono sia gratuite che a pagamento a partire da 0,99 \$ a 999,99 \$ USD; negli Stati Uniti le tasse di registrazione e presentazione dell'applicazione a carico sviluppatori sono attualmente pari a 0 \$. Il servizio è disponibile in inglese, francese, italiano, tedesco, spagnolo e portoghese brasiliano.

2.5.4 Ambiente di sviluppo

Gli sviluppatori che intendono scrivere applicazioni per i dispositivi BlackBerry hanno a disposizione da RIM essenzialmente due scelte: la prima è utilizzare un ambiente di sviluppo specifico creato ad hoc da RIM, il BlackBerry JDE (BlackBerry Java Development Environment). Permette di creare, importare ed esportare progetti BlackBerry, nonché di compilarli ed eseguirli o effettuarne il debug su dispositivi collegati al calcolatore; è inoltre possibile utilizzare il BlackBerry JDE per sviluppare applicazioni Java Micro Edition. In alternativa si possono utilizzare degli applicativi che simulano il comportamento dei dispositivi mobili. La compilazione avviene utilizzando il classico

Java Compiler, verificando nel contempo la compatibilità dell'applicazione con le API BlackBerry.

La seconda scelta possibile è quella di utilizzare Eclipse, un potente IDE per applicazioni in Java, ed in altri molteplici linguaggi di programmazione come C, C++, Python, PHP, ecc... Eclipse è plugin extensible e grazie a questa caratteristica RIM mette a disposizione un plugin per godere, anche in questo ambiente, degli stessi servizi presenti nel suo JDE nativo.

Il plug-in BlackBerry Java per Eclipse e il BlackBerry Java Development Environment supportano sia la versione 6 che la versione 7 del sistema operativo. Sono inoltre disponibili nuove API per sfruttare le caratteristiche hardware dei nuovi device ed incorporarle nelle applicazioni; fra le funzionalità più importanti citiamo:

- OpenGL ES 2.0 - sfrutta il nuovo chip grafico e permette di creare giochi dalla grafica complessa;
- Window Manager API - sfrutta anch'essa il nuovo chip grafico e la maggior potenza del device;
- Compass API - utilizza il nuovo magnetometro per determinare la posizione e l'orientamento del dispositivo (bussola digitale o magnetica). Se combinato con la Window Manager API permette lo sviluppo di applicazioni di realtà aumentata;
- NFC API - permette di simulare le virtual tag e di creare soluzioni complesse per i pagamenti e il commercio mobile con la nuova tecnologia NFC (Near Field Communications);
- API per l'accesso all' Interfaccia Utente;
- Le API relative alla ripresa video saranno aggiornate per supportare la registrazione in HD;
- Send Menu API - permette di condividere contenuti con i propri contatti come fanno le applicazioni native;

- Unified Search API - l'aggiornamento permette un'integrazione ancora più semplice con la Ricerca Universale;
- BlackBerry Maps API - l'aggiornamento semplifica il processo di visualizzazione dei dati su Mappe;
- Options API - si adegua al nuovo look and feel del sistema operativo e permette ad applicazioni di terze parti di aggiungere le proprie voci nelle opzioni relative allo schermo;
- Multimedia API - è stato aggiunto il controllo per il buffer del playback e la definizione del bitrate per la registrazione video;
- Barcode API - l'aggiornamento supporta Zxing 1.6 e permette la definizione dei decoder personalizzati.

Per l'effettivo debugging delle applicazioni vengono forniti diversi simulatori che riproducono, anche graficamente, le sembianze e le funzionalità dei dispositivi BlackBerry.

Capitolo 3

Differenze tecniche tra i sistemi operativi

Verranno ora analizzate le differenze tecniche dei vari sistemi operativi, cercando di capire quali tra queste sono determinanti per diventare i leader del mercato, sia considerando il punto di vista dell'utente che dello sviluppatore.

3.1 Facilità d'uso/Interfaccia utente

IOS è risultato essere il sistema operativo mobile più semplice, dove tutto è così chiaro ed evidente che chi l'ha usato per la prima volta, senza distinzione di età, riesce a capire il suo funzionamento, senza dover fare riferimento a un manuale. La ragione di questo è che l'interfaccia non presuppone che l'utente sappia come usarlo e per questo è possibile fare le operazioni di base senza alcun tipo di aiuto.

Il secondo nella linea di intuitività è **Android**; non ha lo stesso livello di semplicità di iOS, ma è comunque molto facile; purtroppo l'interfaccia Android cambia da telefono utilizzato, il che significa che se un utente poco esperto utilizza, per esempio, un telefono HTC Android, si sentirà perso quando prende in mano un telefono Samsung Android. Così, anche se Google

e gli OEM cercano di rendere l'interfaccia utente amichevole, il fatto che ci siano tanti tipi differenti di interfacce è destinata a lasciare un utente laico confuso. La facilità d'uso di Android aumenta anche grazie ai widget, che sono piccole icone interattive di diverse dimensioni in grado di eseguire tutte le funzioni (o specifiche funzioni) di un'applicazione particolare, per esempio, l'utente può aggiungere un widget e-mail, che mostra tutte le email ricevute; non tutte le applicazioni però supportano i widget.

Utilizzando le prime versioni di Symbian S60 5th Edition, si notava che l'interfaccia utente è stata progettata per i telefoni con tastiera e Nokia ha fatto poco per assicurarne l'utilizzabilità anche sui dispositivi touch. Non è stato così per **Symbian 3**, che ha fatto molti passi in avanti in termini di usabilità; le cose ora funzionano a dovere e non c'è più nessun dubbio se cliccando qualcosa serva per evidenziarla o per avviarla (come succedeva nella vecchia versione); non riscuote successo il modo in cui sono sparse le applicazioni attraverso il menu e la tastiera su schermo non è delle migliori. In generale però l'ultima versione di Symbian è abbastanza facile da usare, e a differenza di Android, non ci si deve preoccupare di layout di interfacce diverse su dispositivi diversi.

Windows Phone 7 non è il migliore quando si tratta di facilità d'uso, anche se il miglioramento con le precedenti versioni di windows mobile è notevole; ci sono alcune cose che non sono immediatamente evidenti, come ad esempio il modo in cui è necessario premere e tenere premuto su alcune voci per visualizzare le opzioni aggiuntive; poi c'è anche il comportamento bizzarro del pulsante di ricerca o il minuscolo tasto di chiamata / fine chiamata e la necessità di sbloccare lo schermo prima di poter ricevere una chiamata. L'interfaccia utente è completamente differente rispetto ad iOS e Android e l'idea dei riquadri "tiles" è sicuramente originale; risulta buona la tastiera, che però, è alla pari con la tastiera su Android Gingerbread.

3.2 Caratteristiche

Le caratteristiche non sono mai state un punto forte di **iOS**, ma nel corso degli anni Apple ha aggiunto un sacco di funzionalità al sistema operativo, come ad esempio la possibilità di installare applicazioni, il multitasking, copia-incolla, cartelle, ecc. iOS oggi lascia poco spazio per il reclamo, tuttavia, ci sono alcune cose che Apple deve ancora prendersi cura, come i trasferimenti di file Bluetooth, file manager, memoria di massa, widgets homescreen e radio FM per citarne alcuni.

Più grande vantaggio di **Android** su iOS è stata la funzionalità e con il rilascio di Android Gingerbread ha quasi tutte le funzionalità che si possa desiderare, se si tratta di multitasking, widgets, tethering con hotspot Wi-Fi o supporto Adobe Flash 10.1, Android copre tutti i campi. Anche se non supporta diversi formati video, è il più completo di tutte e quattro le piattaforme, in termini di caratteristiche.

Symbian 3 se la cava bene. Si possono trovare quasi tutte le caratteristiche che si hanno su Android, insieme ad alcune che non ha, come la radio FM, la connettività USB On-the-Go e il supporto ad un'ampia varietà di formati video. Caratteristiche come il multitasking, copia e incolla hanno sempre fatto parte di Symbian diversi anni fa e sono sempre stati eseguiti alla perfezione; Symbian 3 ha la maggior parte delle caratteristiche che si vorrebbe avere e in questo campo non ha nulla da invidiare ad Android o iOS.

Quello delle caratteristiche è invece un aspetto in cui **Windows Phone 7** fallisce miseramente. Per un sistema operativo lanciato nel 2010, Microsoft ha lasciato fuori alcune cose molto importanti (come il multitasking e il copia-incolla per il testo); anche se stanno incorporando la maggior parte delle funzionalità attraverso gli aggiornamenti, avrebbero dovuto farlo dal primo giorno; poteva essere scusabile lasciar fuori queste cose nel 2007, ma in questo caso Microsoft non ha scuse, considerando che sono nel settore smartphone molto prima del rilascio di Windows Phone 7. Ha alcune buone caratteristiche, come le homescreen tiles, il supporto con i servizi Xbox Live,

Windows Live, Bing, Zune Pass e l'integrazione di Office, ma non regge il confronto contro i rivali.

3.2.1 Multimedia

IOS si basa sul software di **iTunes** per trasferire file multimediali compatibili dal PC al dispositivo dell'utente; i file musicali saranno poi aperti tramite l'applicazione iPod, le immagini attraverso l'applicazione Foto e video tramite l'applicazione Video. iTunes creerà una libreria di quei file multimediali, che possono essere automaticamente sincronizzati con il dispositivo o aggiunti manualmente uno per uno; inoltre, gli utenti con Apple ID possono scaricare album completi, canzoni, podcast (che agiscono come una stazione radio ma con elenchi di riproduzione preimpostati che vengono aggiornati di tanto in tanto), audiolibri o addirittura i film da iTunes Store, sia sul dispositivo o direttamente al PC, che sarà successivamente trasferito al telefono. Possono essere scaricate ulteriori applicazioni sviluppate da terze parti dall' App Store in grado di offrire opzioni migliori sulla gestione dei media, come l'apertura di file non supportati, l'editing delle immagini e così via. Le fotocamere supportano anche geotagging (aggiunta di dati di identificazione geografica alle immagini in modo da collegarli in una posizione specifica per la condivisione delle informazioni).

I telefoni **Android**, a differenza degli iPhone, hanno il vantaggio di poter essere usati come un disco rigido esterno quando è collegato al PC, dove il trasferimento di tutti i tipi di file multimediali supportati può essere fatto semplicemente trascinando un file o una cartella e lasciarlo all'interno del telefono (o della scheda di memoria se installata). Il telefono cellulare riconosce automaticamente questi file e li apre nella rispettiva applicazione; il trasferimento delle foto in un' unica cartella permetterà al telefono di riconoscere questa cartella come un album e visualizzarla di conseguenza. Anche in questo caso, essendo un sistema operativo Open Source, si possono scaricare molte applicazioni dall' Android Marketplace che consentono all'utente di accedere a tutti i tipi di file e la possibilità di introdurre nuove funzionalità

come gli effetti speciali e la manipolazione di foto. Anche le fotocamere Android hanno opzioni geotagging diverse in base ai produttori, che aggiungono il loro software per trarre beneficio da esso.

Come nel caso di iOS e iTunes, **Windows Phone 7** si basa fortemente sul software **Zune** per il trasferimento dei file multimediali; il software eseguirà la scansione dei file multimediali nel disco fisso dell'utente e li importa in una libreria sotto la scheda "Collection", in cui l'utente sarà in grado di scegliere quale file vuole trasferire nel telefono. Gli utenti saranno in grado di trasferire musica, video, immagini e canali podcast. È importante notare che quando le foto vengono trasferite da un computer al telefono utilizzando il software Zune ottengono un ridimensionamento, che è considerato un aspetto negativo per molti, soprattutto per coloro che vogliono mantenere la qualità delle loro foto per la condivisione o semplicemente per visualizzarle. Le impostazioni della fotocamera sono piuttosto semplici e non includono nessun aggiustamento sofisticato, per il momento non è disponibile nessun software aggiuntivo della fotocamera che offre anche possibilità di geotagging.

Symbian 3 ha avuto un sviluppo nel reparto della galleria fotografica, infatti è la più veloce, e rileva eventuali nuove foto da poter copiare nella memoria di massa; la galleria carica le miniature in un baleno, il visualizzatore di immagini ha il pinch-to-zoom e lo swipe laterale che porta alla foto successiva in un istante; l'applicazione galleria è davvero tra le migliori del settore. L'editor di foto ti permette di fare alcune modifiche di base come rotazione, ridimensionamento e il ritaglio, così come l'aggiunta di effetti, cornici, testi, disegni, la riduzione occhi rossi e l'ottimizzazione fotografica. Il lettore musicale è molto ben organizzato, dispone di un equalizzatore con una buona gamma di impostazioni per gli stili musicali. Il lettore video supporta DivX, ed è il migliore, perché oggi non ci sono sistemi operativi mobili che includono il supporto nativo per una così ampia gamma di formati video; la riproduzione video è molto liscia e l'editor video ha due modalità: creare video con il tuo videoclip e fotografie, o creare una presentazione da solo foto.

Come nel caso dei telefoni Android, anche **Blackberry** può essere col-

legato al PC e riconosciuto come supporto di memorizzazione per trasferire facilmente i file; questo può essere fatto anche tramite l'applicazione **BlackBerry Media Sync** offerto da RIM, che permette agli utenti di trasferire facilmente i file ai loro dispositivi. Tutti i file multimediali si trovano nelle loro applicazioni native, che offrono caratteristiche di base per riprodurli e visualizzarli; l'equalizzatore offre funzionalità di base per il lettore musicale e una opzione "Media Boost" che migliora la qualità delle canzoni in riproduzione. La fotocamera sul Blackberry offre geotagging e una modalità di ripresa che esegue un ottimo lavoro quando si passa da una scena all'altra.

3.2.2 Navigazione Satellitare

Tutti i sistemi operativi offrono un sistema di navigazione GPS e A-GPS (Assisted GPS, dove il telefono utilizza la rete per triangolare una posizione più precisa e più velocemente) tuttavia, ogni sistema operativo utilizza un'applicazione di navigazione nativa diversa: **Android** e **iOS** utilizzano entrambi Google Maps, con la possibilità di scaricare applicazioni aggiuntive e installare mappe su memoria interna / scheda di memoria al fine di navigare con precisione, senza la necessità di connettersi a Internet. **Windows Phone 7** utilizza Bing Maps, un servizio fornito da Microsoft ed è l'unico sistema operativo che non permette l'uso di qualsiasi altra applicazione di navigazione. **Blackberry** usa Blackberry maps e per **Symbian 3**, l'applicazione mappe è il solito Ovi Maps, che dà la possibilità di scaricare le mappe quando si utilizza il WiFi che vengono poi utilizzate per la navigazione offline.

3.2.3 Aggiornamenti

Apple aggiunge ulteriori funzionalità ad ogni importante aggiornamento del firmware, ma il più delle volte questi sono limitati ai soli dispositivi più recenti, mentre i più vecchi ne ottengono solo una parte; le patch e bug fix sono rilasciati solo da Apple e il downgrade non è possibile ufficialmente.

WP7 ha la ricerca degli aggiornamenti software automatica; non supporta l'integrazione a monte, e gran parte dei telefoni attualmente in esecuzione con Windows Mobile 6.x non sono aggiornabili a Windows Phone 7.

Anche **Android** ha gli aggiornamenti automatici delle applicazioni; gli aggiornamenti (anche minor release) vengono rilasciati spesso e con regolarità; ci sono inoltre un sacco di versioni personalizzate da parte di sviluppatori esterni con correzioni immediate ai bug.

3.3 Performance

Quando **iOS** è uscito per la prima volta, ha entusiasmato la sua interfaccia fluida che ha funzionato perfettamente anche sul modesto hardware dell'iPhone di prima generazione; nel corso degli anni il sistema operativo è diventato più pesante e la prova di questo è stato il modo in cui l'iPhone 3G faccia fatica a girare con iOS 4.0; se però si prova lo stesso sistema operativo su un iPhone 4, si può notare un mondo di differenza; l'interfaccia utente è liscia e senza lentezze evidenti, anche quando si passa da più applicazioni, l'interfaccia utente mantiene la sua fluidità, senza esitazione.

Qualcosa di simile è stato osservato nel caso di **Android**; finché lo si utilizza con dell'hardware veloce, funziona bene, ma sui dispositivi più lenti tende a soffocare. Tuttavia, a differenza di iOS, anche se in esecuzione su hardware più veloce, Android non è mai perfettamente fluido, a volte si possono notare rallentamenti inattesi e inspiegabili, mentre si naviga attraverso l'interfaccia utente; Google ha inoltre aggiunto effetti grafici all'interfaccia utente, che però tendono ad impantanarsi sui dispositivi con hardware meno potente. Android non utilizza la GPU (Graphics Processing Unit) per visualizzare sullo schermo le immagini, il che significa che la CPU è sovraccarica, causando ulteriori rallentamenti. Con qualche ottimizzazione, Android può funzionare abbastanza bene anche su dispositivi più lenti; e non specifica alcun requisito minimo.

Uno dei maggiori punti di forza di **Symbian** è che è sempre stato un sistema operativo molto leggero che può essere eseguito anche da hardware più deboli; è per questo che tutti i telefoni Symbian hanno hardware meno impressionanti di quelli che siamo abituati a vedere sui dispositivi di fascia alta, ma questo è assolutamente un bene perchè anche su quei tipi di hardware il sistema operativo funziona perfettamente. Dato che il sistema operativo è così leggero, si elimina la necessità di utilizzare inutilmente l'hardware e consumare di più la batteria durante il processo; questo è il motivo per cui i telefoni Symbian hanno una maggiore durata della batteria tra gli smartphone. Nokia ha anche fatto buon uso della GPU per visualizzare tutte le immagini su schermo, lasciando la CPU libera di gestire altre attività.

Quando si tratta di interfaccia utente scorrevole, **Windows Phone 7** è imbattibile. Questo soprattutto perché è sempre situato su un hardware potente, ma anche perché il sistema operativo è ben ottimizzato per questa caratteristica. Purtroppo, la morbidezza è limitata solo alle applicazioni di default, le applicazioni di terze parti non vengono eseguite con gli stessi standard che Microsoft ha creato.

3.3.1 Multitasking

Windows Mobile, fin dalle sue prime implementazioni, utilizzava un particolare multitasking innovativo sotto alcuni punti di vista, il cui intento era rendere le cose molto più semplici e funzionali; se un processo rimane inutilizzato in memoria, il sistema, nel caso in cui avesse bisogno di un quantitativo maggiore di memoria, chiudeva automaticamente il processo, liberando quindi la memoria da poter essere utilizzata per altri processi, peccato che l'operazione di chiusura di un processo per liberare memoria richiedesse un quantitativo maggiore di memoria, causando lag, rallentamenti o addirittura blocchi del sistema.

Android, implementa un sistema di multitasking simile a Windows Mobile, ma in maniera molto più efficiente, e con algoritmi leggermente più complessi; infatti non chiude a prescindere gli ultimi processi inattivi per

liberare memoria, ma utilizza un insieme di regole per stabilire la priorità di chiusura di un processo: in questo modo non vengono terminati processi importanti o attualmente in esecuzione, ma vengono chiusi effettivamente quelli inutilizzati dal sistema. Android inoltre utilizza speciali servizi e background receivers, sviluppati per funzionare in ogni momento, se la memoria non viene mai esaurita del tutto, un processo continua a funzionare in background.

Diversa è la situazione di **iOS**, e la sua implementazione del multitasking; fino alla versione 3, infatti iOS era un sistema che in gergo si definisce multitasking, in realtà è errato definirlo in tale modo perché alcune applicazioni di sistema (come il lettore multimediale) avevano accesso alle funzionalità di multitasking del sistema; le applicazioni di terze parti invece però non avevano tale privilegio, quindi una volta avviate, venivano chiuse (tranne per alcune, che tuttavia potevano sfruttare l'agente background per le notifiche Push). Da una parte c'erano grossi vantaggi, in quanto un processo aveva accesso ad un enorme quantitativo di memoria e risorse della CPU, e raramente subiva rallentamenti, lo svantaggio però di tale sistema era comunque l'impossibilità di richiamare un processo inattivo. Con iOS 4, Apple ha implementato un nuovo sistema di multitasking; le applicazioni che richiedono particolari funzioni, possono interagire con il sistema per mantenere attive queste funzionalità finché non vengono stoppate dall'utente: queste funzionalità sono limitate all'audio in riproduzione/registrazione, geolocalizzazione, e funzioni VoIP/chiamate. Tutte le altre applicazioni che non richiedono tali funzionalità vengono gestite dal sistema: alla pressione del tasto Home, il processo viene ibernato o sospeso; pertanto non può avere accesso alle risorse della CPU, ma continua ad utilizzare quantitativi di memoria; nel caso in cui il sistema necessita di un quantitativo maggiore di memoria per eseguire un processo, chiuderà in automatico i processi inattivi più vecchi.

Windows Phone 7 sta seguendo la scia di iOS; le prime versioni infatti non permettevano alle applicazioni di terzi di accedere alla funzionalità di multitasking, a differenza invece di alcune applicazioni di sistema come mail, lettore multimediale, ecc.. il vantaggio, oltre a quelli descritti sopra per

iOS, consistono in un maggior risparmio energetico. Tuttavia con il major update di WP7, Mango, è stato introdotto pieno supporto al multitasking, un' implementazione che sta a metà fra quello di iOS 4 e quello di Android. Gli sviluppatori che vorranno che le loro applicazioni possano eseguire azioni in multitasking potranno avere accesso ai Background Agents. A differenza delle altre implementazioni del multitasking, dove i processi hanno libero accesso alle risorse di memoria e CPU, su Windows Phone 7 con i Background Agents, i processi potranno avere accesso al solo 10% della CPU e a 5 MB di memoria di RAM, nel caso in cui il telefono stia utilizzando la batteria; altri processi potranno essere eseguiti solo per 15 secondi ogni mezz'ora per avere accesso a funzionalità come la sincronizzazione di dati; ma la novità sta nel fatto che i Background Agents hanno meno limitazioni nel momento in cui il telefono sia collegato alimentato a corrente diretta, ovvero alla presa AC, o via USB al PC: in questo modo i processi possono avere maggior vantaggio, e sfruttare maggiori funzionalità del sistema, come scaricare un maggior quantitativo di dati.

La ragione per cui ognuno implementa il proprio sistema di multitasking, è trovare il giusto compromesso tra prestazioni e stabilità, e minori consumi della batteria; un consumo eccessivo di risorse di CPU e memoria infatti implica un maggior consumo (e spreco) di energia; inoltre deve essere implementato correttamente anche a livello di interfaccia grafica, in modo che gli utenti possano riprendere facilmente vecchi processi, o chiudere autonomamente quelli inattivi.

3.4 Applicazioni

Questo è un settore in cui **iOS** tira fuori un vantaggio enorme davanti a tutte le altre piattaforme. Ci sono milioni di applicazioni disponibili su App Store al momento in attesa di essere scaricate; premesso che a più della metà di loro non vale la pena darle una seconda occhiata, ci sono alcune applicazioni veramente brillanti. Infatti, la qualità generale di applicazioni

disponibili è la più alta tra tutte le piattaforme smartphone; alcune di queste applicazioni hanno veramente rivoluzionato il modo in cui usiamo i nostri smartphone e in un modo che nemmeno Apple avrebbe mai immaginato quando hanno costruito l'iPhone.

Sebbene **Android** stia rapidamente recuperando terreno con iOS in termini di numero di applicazioni, non si trovano applicazioni veramente interessanti a favore di Android. La maggior parte delle grandi applicazioni sono già disponibili su iOS e le restanti sono di Google Apps; ci sono poche grandi applicazioni o giochi che sono esclusivi di Android in questo momento. Le cose però potrebbero cambiare una volta che tutti si renderanno conto che Android è la piattaforma migliore per lo sviluppo, considerando che non ci sono restrizioni severe da seguire a differenza dell' App Store, la gente potrebbe fare un passo verso Android. Inoltre Android può consentire il download di applicazioni di terze parti direttamente da Internet o da qualsiasi altra app store (anche non ufficiale) e li installa sul telefono senza alcuna restrizione, cosa impossibile da fare ufficialmente su iOS.

Anche se **Symbian** ha una delle migliori librerie di applicazioni disponibili in termini puramente numerici, la mancanza di un negozio di applicazioni significava una maggiore difficoltà ad avere accesso ad esse. Ora che Nokia ha Ovi Store, le cose sembrano migliorare, anche se la piattaforma è abbastanza nuova, il negozio ha un buon numero di applicazioni disponibili e sta crescendo ad un ritmo costante. C'è da dire però che su Ovi Store si può solo avere applicazioni delle funzioni base e non c'è varietà di scelta come su iOS o Android; ad esempio se vuoi un client Twitter, c'è Gravity, se vuoi un'applicazione instant messaging, puoi usare Nimbuzz; anche se questo rende più facile la scelta, a volte si sente il bisogno di avere più applicazioni della stessa categoria da poter scegliere.

Windows Phone 7 ha il numero meno impressionante di applicazioni disponibili, il Windows Marketplace non inonda il mercato con grandi applicazioni come ci si poteva aspettare. Proprio come Ovi, ha tutte le applicazioni di base coperte, ma non c'è niente che non sia disponibile sulle

altre piattaforme, al momento; inoltre, le applicazioni e soprattutto giochi sembravano eccessivamente onerosi sul Marketplace rispetto all' App Store e all' Android Market, la stessa applicazione costerebbe 2-3 volte di più sul Marketplace per nessun motivo.

3.5 Telefoni supportati e Hardware

IOS funziona solo sui dispositivi Apple (iPhone, iPad, iPod Touch) quindi sia il dispositivo che il sistema operativo sono prodotti dalla stessa azienda. In sintesi l'hardware necessario per iOS sull' iPhone è:

- GSM Quad-band e supporto quad-band 3G HSDPA con 7,2 Mbps e HSUPA 5,76 Mbps
- schermo 3.5" 16M-color LED-backlit TFT , touch screen capacitivo da 640 x 960 px
- Vetro antigraffio anteriore e posteriore
- 1 GHz Apple A4 SoC, 512 MB di RAM
- 5 MP fotocamera auto-focus con flash LED e messa a fuoco
- Registrazione video 720p a 30fps
- Wi-Fi 802.11b/g/n
- GPS con A-GPS, bussola digitale
- 16/32GB di memorizzazione
- Sensore giroscopico Accelerometro, sensore di prossimità e tre assi
- Cancellazione del rumore attiva con un microfono dedicato secondario
- Audio jack Standard da 3.5 mm, Bluetooth stereo v2.1
- Fotocamera Secondaria frontale

A differenza della Apple, **Windows Phone 7** è fornito da diversi produttori come HTC, Samsung, LG e Dell. Per porsi su una piattaforma elevata e mantenere una esperienza coerente tra tutti i dispositivi, Microsoft ha imposto un elenco specifico e duro di requisiti hardware minimi per quei produttori:

- Schermo: capacitivo, 4-point multi-touch con risoluzione schermo WVGA 800 x 480;
- Processore: Min 1 GHz o superiore
- Grafica: DirectX9 con capacità di rendering GPU
- Memoria: 256 MB di RAM con almeno 8 GB di memoria Flash
- Fotocamera: 5 megapixel con un flash LED
- Tasti fisici: 6 pulsanti hardware dedicati - indietro, Start, Cerca, fotocamera, accensione/spegnimento e Volume Su e Giù
- Altro: Accelerometro con bussola, sensore di luce ambientale, sintonizzatore radio FM, sensore di prossimità e GPS assistito .

Il primo dispositivo con Windows Phone 7 è stato l' LG Optimus 7, successivamente sono usciti il Samsung Omnia 7, HTC 7 Mozart e HTC HD7.

Android è disponibile nella maggior parte degli smartphone di oggi (Samsung, HTC, Sony Ericson, Motorola, LG e Acer) in cui ognuno dei produttori ha dato il suo tocco personale al design e al look dei propri dispositivi come Google ha dato a loro una grande quantità di libertà di movimento per essere creativi. I telefoni Android possono essere in tutte le forme e le specifiche, con touch screen, tastiere e così via, e fino a oggi non ha mostrato alcun bisogno di alcun requisito minimo per essere disponibile su qualsiasi piattaforma specifica, anche se ovviamente più alte saranno le specifiche, migliore sarà l'esperienza. Tuttavia, gran parte dei telefoni Android condividono un comune denominatore in termini di pulsanti hardware (possono essere pulsanti capacitivi o i vecchi tasti fisici), dove troverete:

Un pulsante Home: Come dice il nome, che vi porta alla schermata iniziale del telefono. Inoltre, se tenuto per un po', si apre il menu multitasking per spostarsi all'interno delle applicazioni diverse aperte.

Un pulsante Menu: apre un menu diverso a seconda che le applicazioni attualmente in uso, come opzioni aggiuntive, impostazioni o dati applicazione generale (versione, dettagli ...).

Un pulsante Indietro: Chiude un'applicazione o da all'utente un passo indietro.

Un pulsante Trova: Può essere utilizzato per cercare sia per alcune informazioni specifiche sul telefono (persone, messaggi, file ...) o sul web. Se tenuto per un po', si apre il menu di comando vocale.

I telefoni che attualmente supportano la piattaforma **Symbian 3** sono il Nokia N8 e gli ultimi device Nokia C6, Nokia C7 e Nokia E7. Utilizzano una potenza minima del processore di 600 Mhz, una RAM minima di 512 Mb, fotocamera di fascia alta (almeno 5 MP, il N8 alloggia da solo un sensore della fotocamera da 12 MP), un touchscreen capacitivo che non ha nulla da invidiare da altri dispositivi touch.

I dispositivi **Blackberry** hanno avuto un aspetto distinto per quasi un decennio:

- Uno schermo di diverse dimensioni (di solito in 2"- 2.8" range);
- Una comoda tastiera QWERTY (o altri, a seconda della regione);
- Un bottone Chiamata e uno Fine;
- Un pulsante menu dove apre il relativo menu dell'applicazione aperta;
- Un pulsante Indietro per muoversi un passo indietro;

- Una Scrollball cliccabile (sostituito recentemente dalla trackpad ottica, sensibile al tocco) per lo scorrimento delle pagine e la selezione del menu;
- Un pulsante Fotocamera;
- Due pulsanti laterali personalizzabili (uno a sinistra, l'altro a destra), che può essere programmato per eseguire una funzione specifica dal comando vocale per lanciare un'applicazione;
- Un jack audio da 3.5mm;
- Volume rocker;

Tuttavia, con l'aumento della popolarità del touch screen tra i telefoni cellulari, Blackberry ha cercato di entrare nel gioco con il Blackberry Storm 9500 (anche il Storm2 9520), che ha introdotto un 3,25" touch screen con la tecnologia SurePress. Questa però non ha dimostrato di essere una tecnologia di successo rispetto a tutte le altre piattaforme presenti sul mercato. Un altro tentativo da parte di RIM nella categoria touch screen è stato fatto con il rilascio del Torch 9800, che combina un 3,2 " touch screen con una tastiera fisica slide-out. La maggior parte dei BlackBerry in questi giorni hanno un processore 528 MHz o 624 MHz uno.

3.6 Analisi delle differenze

Analizzando le differenze appena proposte possiamo notare come la facilità d'uso, il numero di funzionalità presenti ma soprattutto la maggior varietà di scelta delle applicazioni e la possibilità di avere un certo numero di app per ogni funzione, siano i fattori più importanti appartenenti ad Android e iOS che rappresentano i due sistemi operativi leader del settore.

Capitolo 4

Gestione della memoria

Ora, entriamo nel dettaglio e analizziamo alcuni aspetti caratteristici della gestione della memoria che differenziano i sistemi operativi Android e Apple iOS.

4.1 Gestione memoria Android

Android utilizza le librerie native open source C che hanno alimentato le macchine linux per anni; tutte le operazioni di base del sistema operativo come la gestione degli input/output, della memoria e così via sono gestite dal kernel nativo di Linux. Prima di spiegare come viene gestita la memoria e i suoi processi spieghiamo brevemente come vengono gestite le attività e i suoi stati.

4.1.1 Gestione delle attività

Le applicazioni Android si compongono di quattro mattoni fondamentali: le attività (activity), i servizi (service), i broadcast receiver e i content provider. Ogni applicazione è formata da uno o più di questi mattoni, non è detto che li contenga tutti, ad esempio potrebbe essere costituita da due attività e da un servizio, senza avere broadcast receiver né content provider. Nella stragrande maggioranza dei casi, però, le applicazioni comprendono

è in esecuzione e interagisce direttamente con l'utente, le altre, invece, sono ibernata e tenute nascoste in sottofondo, in modo da ridurre al minimo il consumo delle risorse di calcolo. L'utente, naturalmente, può ripristinare un'attività ibernata e riprenderla da dove l'aveva interrotta, riportandola in primo piano; l'attività dalla quale si sta allontanando, invece, sarà ibernata e mandata in sottofondo al posto di quella ripristinata.

Visto che le attività ibernata, in termini di risorse di calcolo, non consumano nulla, in Android il concetto di chiusura delle attività è secondario e tenuto nascosto all'utente. I casi in cui un'attività può terminare sono due:

- L'attività è ibernata e il sistema, arbitrariamente, decide che non è più utile e perciò la distrugge.
- Il sistema è a corto di memoria, e per recuperare spazio inizia a “uccidere” bruscamente le attività in sottofondo.

I differenti passaggi di stato di un'attività attraversano alcuni metodi della classe "Activity", le chiamate ai metodi eseguiti durante i passaggi di stato dell'attività sono:

- **onCreate()** Richiamato non appena l'attività viene creata.
- **onRestart()** Richiamato per segnalare che l'attività sta venendo riavviata dopo essere stata precedentemente arrestata.
- **onStart()** Richiamato per segnalare che l'attività sta per diventare visibile sullo schermo.
- **onResume()** Richiamato per segnalare che l'attività sta per iniziare l'interazione con l'utente.
- **onPause()** Richiamato per segnalare che l'attività non sta più interagendo con l'utente.
- **onStop()** Richiamato per segnalare che l'attività non è più visibile sullo schermo.

- **onDestroy()** Richiamato per segnalare che l'applicazione sta per essere terminata.

4.1.2 Garbage Collector e gestione dei processi

La gestione dei processi e della memoria di Android sono un po' insolite; come Java e .NET, Android utilizza il proprio runtime e la propria macchina virtuale per gestire la memoria delle applicazioni; a differenza di uno di questi frameworks, il runtime (tempo di esecuzione) Android gestisce anche le durate dei processi. Android garantisce la risposta delle applicazioni, arrestando e uccidendo i processi se necessario, per liberare risorse per le applicazioni con priorità più alta.

La Dalvik VM è stata ottimizzata per garantire che un dispositivo possa eseguire più istanze in modo efficiente; utilizza il kernel di Linux alla base del dispositivo per gestire le funzionalità di basso livello tra cui sicurezza, threading, gestione dei processi e la gestione della memoria. Tutto l'hardware Android e il servizio di accesso al sistema è gestito attraverso Dalvik come un livello intermedio, utilizzando una macchina virtuale per ospitare l'esecuzione delle applicazioni, gli sviluppatori hanno un livello di astrazione che assicura di non dover mai preoccuparsi di una implementazione hardware particolare.

Ogni applicazione gira in un processo separato all'interno di una propria istanza Dalvik; la Dalvik Virtual Machine implementa un metodo chiamato **Garbage Collector** liberando lo sviluppatore dal compito di gestire in prima persona la memoria e delega ogni responsabilità di gestione della memoria e dei processi alla fase di esecuzione di Android, che ferma e uccide i processi necessari per gestire le risorse. Dalvik e il runtime Android sono situati sulla cima di un kernel Linux che gestisce l'interazione tra hardware di basso livello (inclusi i driver) e la gestione della memoria, mentre un insieme di API fornisce l'accesso a tutti i sotto servizi, funzioni e hardware.

L'ordine in cui vengono uccisi i processi per recuperare risorse è determinata dalla priorità delle applicazioni ospitate; la priorità di un'applicazione è uguale alla sua più alta priorità dei suoi componenti; quando due applicazioni

hanno la stessa priorità, il processo che è stato con una priorità più bassa per più tempo sarà ucciso per primo; la priorità del processo è influenzata anche dalle dipendenze tra processi, se una applicazione ha una dipendenza da un servizio o da contenuti forniti da una seconda applicazione, l'applicazione secondaria avrà almeno pari priorità come l'applicazione che supporta.

Tutte le applicazioni Android rimangono in esecuzione e in memoria fino a quando il sistema ha bisogno delle sue risorse per altre applicazioni.

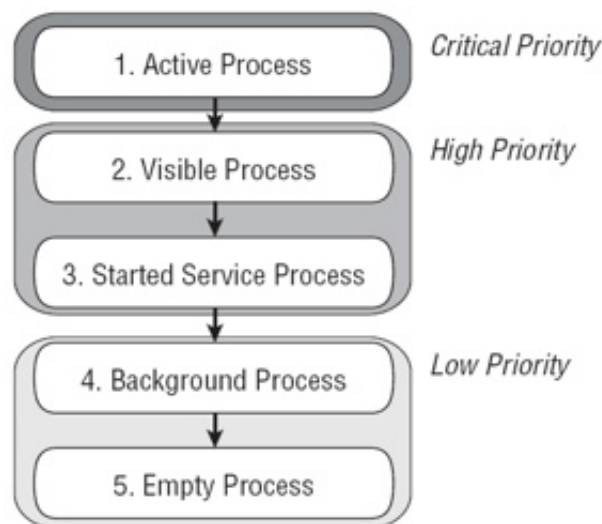


Figura 4.2: Stati delle applicazioni

È importante strutturare l'applicazione correttamente per garantire che la sua priorità sia appropriata al lavoro che sta facendo; se non lo è, l'applicazione potrebbe venire uccisa mentre è nel mezzo di qualcosa di importante. L'elenco che segue descrive ciascuno degli stati dell'applicazione mostrata in figura, spiega come lo Stato è determinato dai componenti dell'applicazione che lo compongono:

Processi Attivi (primo piano) i processi attivi sono quelle applicazioni che ospitano componenti che spesso interagiscono con l'utente. Questi sono i processi che Android cerca di mantenere reattivi bonificando le risorse, gen-

eralmente ci sono molti pochi processi di questo tipo e saranno uccisi solo come ultima risorsa. I processi attivi sono:

- Attività in uno stato attivo, cioè, attività in primo piano che rispondono agli eventi utente.
- Attività, Servizi o ricevitori broadcast che sono attualmente in esecuzione su un gestore di eventi OnReceive.
- I servizi che sono eseguiti su un gestore di eventi onStart, onCreate, o onDestroy .

Processi visibili. Processi visibili, ma inattivi, sono quelli che ospitano attività visibili. Come suggerisce il nome, le attività visibili sono visibili, ma non sono in primo piano o rispondono agli eventi generati dall'utente. Questo accade quando un'attività è solo parzialmente oscurata (da un'attività trasparente o non a schermo intero). Sono generalmente processi visibili pochissimo, e saranno uccise solo in circostanze estreme per consentire ai processi attivi di continuare.

Processi di servizio avviati. Processi che ospitano servizi che sono stati avviati. Servizi di assistenza in corso di elaborazione, che dovrebbero continuare senza un'interfaccia visibile. Poiché i servizi non interagiscono direttamente con l'utente, ricevono una priorità leggermente inferiore rispetto alle Attività visibili. Sono ancora considerati come processi di primo piano e non vengono uccisi a meno che le risorse sono necessarie per i processi attivi o visibili.

Processi background. Processi che ospitano attività che non sono visibili e che non hanno nessun servizio avviato, sono considerati i processi in background (in secondo piano). Ci sarà generalmente un gran numero di processi in background che Android ucciderà con un modello last-seen-first-killed (l'ultimo visto è il primo ad essere ucciso) per ottenere le risorse per i processi in primo piano.

Processi vuoti. Per migliorare le prestazioni complessive del sistema, Android spesso mantiene le applicazioni in memoria dopo che hanno raggiunto la fine del loro ciclo di lavoro. Android mantiene questa cache per migliorare il tempo di avvio delle applicazioni quando vengono rilanciate; questi processi vengono sistematicamente uccisi quando richiesto.

4.1.3 Come usare la memoria in modo efficiente

Android gestisce automaticamente le applicazioni aperte che sono in esecuzione in background, per cui ufficialmente non ci si dovrebbe preoccupare di questo. Questo significa che Android chiude le applicazioni quando il sistema ha bisogno di più memoria; tuttavia, la maggior parte degli utenti non sono molto soddisfatti di come viene gestita questa cosa, perché a volte lascia troppi processi in esecuzione che provoca lentezza nelle prestazioni di tutti i giorni e un maggiore consumo di batteria. Per rimediare a questo si può utilizzare un'applicazione sviluppata da terzi chiamata task killer/task manager che questo lavoro lo fa molto bene.

4.1.4 Problemi di allocazione memoria

Scrivere efficaci applicazioni mobili non è sempre semplice, in particolare, le applicazioni Android contano su una gestione automatica della memoria gestita dal Dalvik garbage collector, ma ciò non significa che è possibile ignorare la gestione della memoria, anzi a volte possono sorgere problemi di prestazioni se non si sta attenti con l'allocazione della memoria.

Alcuni problemi di utilizzo della memoria sono evidenti, per esempio, se l'applicazione ha perdite di memoria ogni volta che l'utente tocca lo schermo, probabilmente si attiva un `OutOfMemoryError` e alla fine la vostra applicazione crasha. Altri problemi sono più sottili, e possono peggiorare le prestazioni sia dell'applicazione che dell'intero sistema, per esempio, in un percorso performance-sensitive, come il codice della logica di un gioco, l'at-

tribuzione della memoria ha un prezzo; dopo troppe assegnazioni di memoria, il garbage collector entra in gioco e interrompe l'applicazione per liberare la memoria, la maggior parte delle volte, garbage collection agisce abbastanza velocemente per non farsi notare, tuttavia, se accade mentre si sta tentando di sconfiggere un nemico in un gioco, si può improvvisamente vedere un calo in termini di prestazioni/capacità di risposta dell'applicazione. Non è insolito per una garbage collection di prendere da 100 a 200 ms, per capire, un'animazione fluida ha bisogno di disegnare ogni fotogramma 16-33 ms, se l'animazione viene improvvisamente interrotta per 10 fotogrammi, possiamo essere certi che gli utenti lo noteranno.

La maggior parte del tempo, si verificano tanti garbage collection a causa di tonnellate di oggetti piccoli e di breve durata; ci sono alcuni tipi di garbage collector (come i Garbage Collector generazionali) che sono in grado di ottimizzare la raccolta di questi oggetti in modo che l'applicazione non venga interrotta troppo spesso; ma purtroppo il garbage collector di Android non è in grado di eseguire tali ottimizzazioni e la creazione di oggetti di breve durata in punti critici del codice può diventare molto costosa per l'applicazione.

Per aiutare ad evitare garbage collection frequenti, l'Android SDK fornisce due metodi principali per monitorare l'utilizzo della memoria in un'applicazione: Allocation Tracker e gli heap dumps (depositi di memoria).

L' **Allocation Tracker** è uno strumento fa parte della DDMS (Dalvik Debug Monitor Service) ; per ogni processo non solo viene mostrato che tipo di oggetto è stato assegnato, ma anche in quale thread, in quale classe, in quale file e in quale linea di codice. L'allocation tracker aiuterà a identificare i problemi importanti nel codice. È utile quando si vuole avere un'idea di che tipo di allocazione sta accadendo in un determinato periodo di tempo, ma non fornisce alcuna informazione sullo stato generale della memoria heap (alla quale non è assegnata a nessun impiego) dell'applicazione.

Gli **heap dumps** sono uno strumento di analisi della memoria più potente; un heap dump è un'istantanea della memoria inutilizzata da un'applicazione, che viene memorizzata in un formato binario chiamato HPROF. Dalvik utilizza un formato che è simile ma non identico al tool HPROF di Java. Per analizzare un heap dump, è possibile utilizzare uno strumento standard come Eclipse Memory Analyzer (MAT) che può aiutare a rintracciare le perdite di memoria nella vostra applicazione.

Nel runtime Dalvik, il programmatore non può esplicitamente allocare e liberare memoria, quindi non si può avere una vera perdita di memoria, come si ha nei linguaggi come C e C++. Una "perdita di memoria" nel codice è quando si mantiene un riferimento a un oggetto che non è più necessario. A volte un solo riferimento può impedire a un ampio insieme di oggetti dall'essere eliminato con il garbage collection.

4.2 Gestione memoria iOS

L'applicazione della gestione della memoria è il processo di allocazione della memoria durante l'esecuzione del programma, utilizzandola, e liberandola quando è terminata. Un programma ben scritto utilizza come memoria il meno possibile; in Objective-C, può anche essere visto come un modo di distribuire proprietà delle risorse di memoria limitate tra i molti pezzi di dati e codice.

Anche se la gestione della memoria è di solito considerata a livello di un singolo oggetto, bisogna gestire in realtà grafi di oggetti. Si vuole fare in modo di non avere più oggetti in memoria di quelli necessari. Objective-C fornisce tre metodi della gestione della memoria di un'applicazione:

- Nel metodo **garbage collection**, il sistema mantiene automaticamente traccia dei propri oggetti rispetto agli oggetti degli altri; è quindi automaticamente libera dagli oggetti che non sono più referenziati. Questo metodo però è supportato solo in ambiente runtime di Mac OS X, non

su iOS principalmente per motivi prestazionali e di occupazione delle risorse; viene quindi utilizzato il metodo **Autorelease Pool**, nato come necessità di fornire un sistema automatico di rilascio delle risorse non utilizzate.

- Un metodo, denominato **MRR (manual retain-release)** è esplicitamente quello di gestire la memoria per tenere traccia degli oggetti che si possiede. È implementato utilizzando un modello, noto come il conteggio di riferimento (reference counting), che la classe `NSObject` (del framework foundation) fornisce in congiunzione con l'ambiente di runtime; è una gestione manuale della memoria dove notificiamo all'autorelease pool che vogliamo utilizzare la variabile e che siamo responsabili del suo rilascio.
- Con la versione di iOS 5 è stato introdotto un nuovo sistema chiamato conteggio di riferimento automatico (**ARC, Automatic Reference Counting**) il sistema utilizza lo stesso sistema di reference counting del MRR, ma si inserisce il metodo di gestione della memoria all'interno del tempo di compilazione. Si è fortemente incoraggiati ad utilizzare ARC per i nuovi progetti; se si utilizza ARC, di solito non c'è nessun bisogno di capire l'implementazione sottostante, anche se in certi casi può essere utile.

4.2.1 Regole di base Gestione della memoria

Il modello di gestione della memoria si basa sulla proprietà degli oggetti. Durante lo sviluppo di qualunque applicazione per iOS bisogna tenere in mente una serie di regole per la gestione degli oggetti:

- Un oggetto può essere utilizzato da uno o più proprietari (altri oggetti);
- Quando un oggetto non ha proprietari, viene distrutto automaticamente senza alcun preavviso e in un momento imprecisato durante

l'esecuzione del vostro applicativo dall'Autorelease Pool, generalmente dopo l'uscita dal metodo in cui è stato creato;

- Per impedire la sua distruzione automatica si deve notificare l'Autorelease Pool che si è divenuti proprietari di tale oggetto (utilizzando il metodo `retain`);
- Se non si è più interessati a tale oggetto, si deve smettere di esserne proprietari (utilizzando i metodi `release`/`autorelease`) per consentirne una possibile distruzione da parte dell' Autorelease Pool;
- Il numero delle dichiarazioni di appartenenza (`retain`) deve essere bilanciato con quello dei rilasci (`release`) in modo da consentire il rilascio automatico;
- Il numero di dichiarazioni di utilizzo viene chiamato `retainCount` e assume superiori o uguali a 0, in caso assuma quest' ultimo verrà deallocato dall' Autorelease Pool;
- Quando si raggiunge una certa esperienza è opportuno utilizzare il più possibile `alloc/retain/release` invece di utilizzare `autorelease`, per motivi prestazionali, soprattutto all'interno di cicli con un numero consistente di iterazioni;
- Il sistema invia un avviso di risorse insufficienti (principalmente RAM) quando questa decresce raggiungendo un valore prossimo ad 1.5MB.

4.2.2 Autorelase Pool

Il funzionamento dell'autorelase pool è identificabile come un analizzatore di istanze di oggetti sulle quali è stato invocato il metodo `autorelease`, questo provvede a liberare la memoria, distruggere quelle istanze di oggetti non più necessari, in maniera automatica, analizzando una particolare proprietà chiamata `retainCount` (un variabile numerica presente in ogni istanza) quando troverà questa pari al valore 0. L'autorelase pool invoca sulla variabile un

metodo, che viene identificato come il suo distruttore, chiamato dealloc, nel quale si effettuano generalmente operazioni di rimozione di legami con altre variabili.

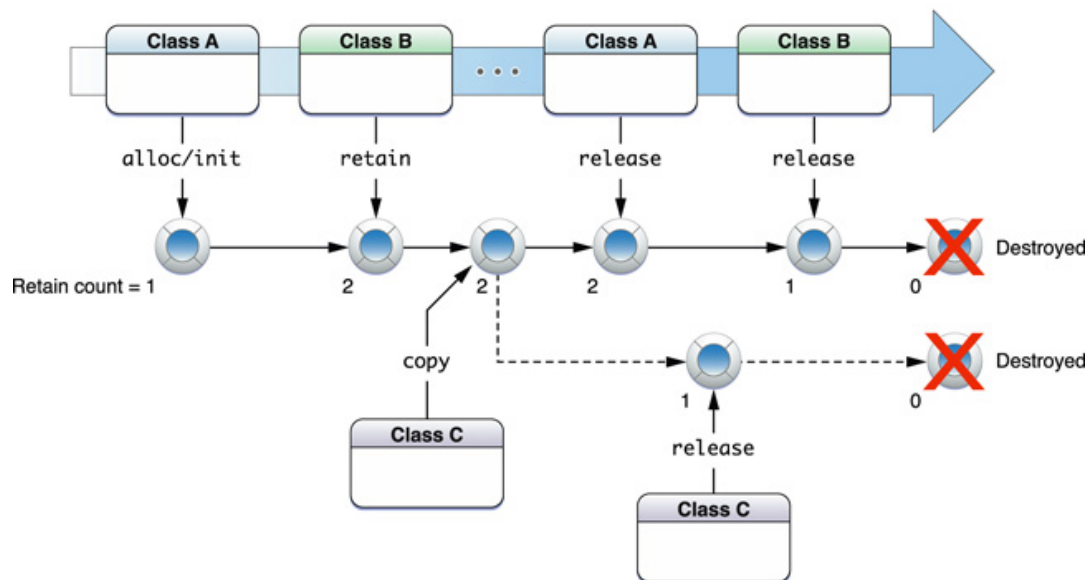


Figura 4.3: Esempio sul funzionamento dell'autorelease pool

Durante il ciclo di vita di una variabile, il retainCount viene quindi incrementato e decrementato, fino generalmente a terminare in un determinato istante t , in cui questo assumerà valore 0: non è detto che si raggiunga tale valore durante l'utilizzo del software, è ammissibile che una variabile abbia retainCount superiore a 0 per tutta la sua esistenza, e termini la sua vita alla chiusura dell'applicativo. L'operazione di rilascio automatico avviene in maniera non prevedibile e non è possibile quindi sapere quando viene effettuata in maniera automatica.

Quando invochiamo autorelease su un oggetto richiediamo il decremento del suo retainCount di un'unità in un momento futuro, se questo scenderà a 0 ci sarà il conseguente rilascio automatico.

In conclusione Autorelease Pool provvede in maniera automatica e trasparente a rilasciare la memoria allocata per un oggetto in un non precisato momento durante l'esecuzione del vostro applicativo. Ogni istanza su cui non

viene invocato il metodo `alloc` (ma anche `, allocWithZone:, copy, copyWithZone:, mutableCopy, mutableCopyWithZone, new` o `retain`), viene gestita da tale oggetto automaticamente. Per evitare tale rilascio automatico della memoria si utilizza la gestione manuale.

4.2.3 MRR (manual retain-release)

Esistono due modi per evitare che le proprie variabili vengano rimosse automaticamente dall'Autorelease Pool, utilizzare **retain** o **alloc**. Nel primo caso basta invocare tale metodo durante la creazione della variabile, dove viene invocato come detto in maniera trasparente il metodo autorelease, oppure successivamente, all'interno del metodo in cui questa viene creata. In questo modo notificiamo all'autorelease pool che vogliamo utilizzare tale variabile e che siamo in prima persona responsabili del suo rilascio, abbiamo ora il pieno controllo sul suo tempo di vita, se non rilasceremo in futuro tale variabile non sarà mai deallocata.

Associata a `retain` esiste la variabile interna a ogni oggetto che abbiamo detto chiamarsi `retainCount`; tale variabile è presente in tutti gli oggetti che discendono da `NSObject`; `retainCount` assume un valore numerico intero, è un contatore e indica il numero di oggetti che hanno chiamato il metodo `retain` su un determinato oggetto, hanno quindi dichiarato che la stanno utilizzando, e ne sono quindi co-proprietari. `retainCount` assume valore 1 quando la relativa variabile viene inizializzata; è stato introdotto il concetto di `retain` per evitare che tali variabili vengano deallocate automaticamente se una o più classi hanno dichiarato che la stanno utilizzando; in questo modo si riduce il rischio che avvengano crash in cascata dovuti alla deallocazione automatica effettuata dall'Autorelease Pool. Quando decidiamo di notificare all'Autorelease Pool che non vogliamo più utilizzare una variabile, non vogliamo quindi più esserne proprietari, invochiamo su di essa il metodo `release`, che decrementa di un'unità il suo `retainCount`: utilizziamo il comando `[nomevariabile release]`, in questo modo, se la nostra variabile aveva `retain count` pari a 1 scenderà a 0 e verrà rilasciata immediatamente: attenzione,

quindi, se viene deallocata non potrete più accedervi, e, nel caso lo faceste, causerete un crash del sistema.

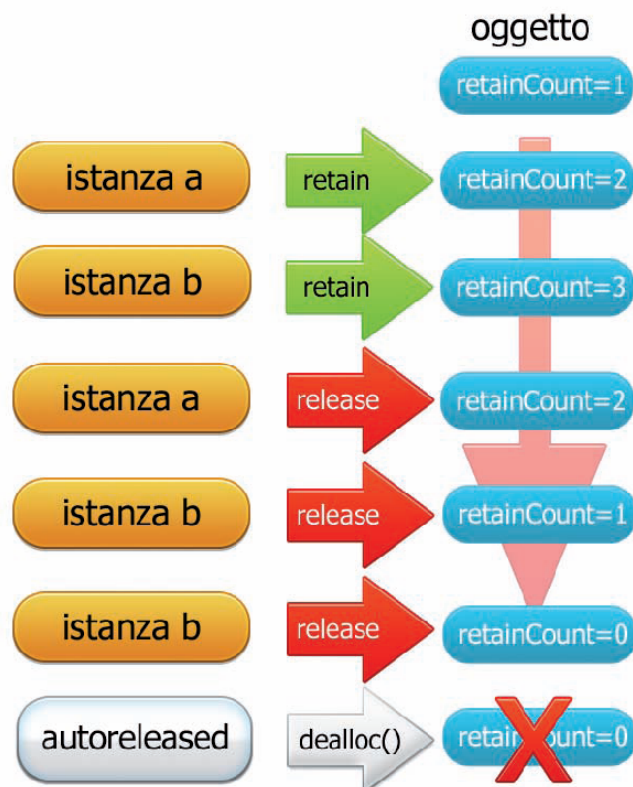


Figura 4.4: Un tipico ciclo di vita di una variabile, quando il `retainCount` raggiunge il valore 0 viene automaticamente rilasciata

Il secondo metodo per evitare che una variabile venga deallocata automaticamente, è quello di crearla utilizzando il metodo `alloc`, che imposta il suo `retainCount` a 1 (effettua quindi un `retain` automatico), e sarà quindi nostra cura decrementarlo utilizzando `release`/`autorelease` per consentire, poi, nel caso tale valore raggiunga valore 0, all'Autorelease Pool di invocare su di esso il metodo `dealloc`. Quando si usa `alloc`, si richiede che venga allocato lo spazio in memoria necessario per contenere tale istanza; per inizializzare il suo contenuto si invoca generalmente un metodo che inizia con il prefisso `init`, che completa, quindi, la fase di impostazione del suo stato interno.

4.2.4 Automatic Reference Counting (ARC)

L' Automatic Reference Counting (ARC) per Objective-C, mette la gestione della memoria nel lavoro del compilatore. Abilitando ARC con il nuovo compilatore Apple LLVM, non ci sarà più il bisogno di digitare `retain` o `release`, semplificando notevolmente il processo di sviluppo, riducendo al tempo stesso crash e perdite di memoria (memory leak). Il compilatore ha una conoscenza completa degli oggetti, e rilascia ogni oggetto nell'istante in cui non è più utilizzato, in modo che l'applicazione sia più veloce di prima.

4.2.5 Problemi relativi alla memoria

Ci sono tre tipi principali di problemi che derivano dalla non corretta gestione della memoria:

- Liberare o sovrascrivere i dati che sono ancora in uso, questo provoca corruzione della memoria, e in genere i risultati nell'applicazione sono crash, o peggio, i dati degli utenti vengono danneggiati.
- Invocare metodi o variabili di un'istanza non più disponibile, perchè deallocata (in seguito a un autorelease automatico oppure a un release manuale) genera un crash chiamato *Zombie*.
- Non liberare i dati che non sono più in uso provoca perdite di memoria, detti leaks (una perdita di memoria è dove la memoria allocata non viene liberata, anche se la variabile non verrà mai riutilizzata). Perdite causate dall'applicazione che utilizza una quantità di memoria sempre più crescente, che a sua volta può portare a prestazioni scarse del sistema o alla chiusura forzata dell'applicazione.

Capitolo 5

Analisi di mercato

Ora analizziamo le situazioni di mercato nel mondo degli smartphone, analizzando prima il mercato mondiale, poi quello degli Stati Uniti, quello europeo per poi concentrarci infine su quello italiano.

5.1 Mercato mondiale

Le vendite mondiali di dispositivi mobili per gli utenti finali sono state pari a 428,7 milioni di unità nel secondo trimestre del 2011, un incremento del 16,5 % rispetto al secondo trimestre del 2010, secondo la società di ricerca e analisi **Gartner, Inc.**

Le vendite di smartphone, aggiornate ad agosto 2011, sono aumentate del 74 per cento anno su anno e rappresentano il 25 per cento delle vendite complessive nel secondo trimestre del 2011, rispetto al 17 per cento nel secondo trimestre del 2010. Google e Apple sono i vincitori evidenti nell'ecosistema smartphone; la quota complessiva di iOS e Android nel mercato dei sistemi operativi per smartphone (OS) è raddoppiato a quasi il 62 per cento nel secondo trimestre del 2011, contro poco più del 31% nel corrispondente periodo del 2010 (vedi tabella sottostante). Gli analisti di Gartner hanno osservato che questi due sistemi operativi hanno l'utilizzabilità che i consumatori

vogliono, le applicazioni che desiderano, e un portafoglio di servizi sempre più ampio, fornito dal proprietario della piattaforma.

Table 2
Worldwide Smartphone Sales to End Users by Operating System in 2Q11
(Thousands of Units)

Operating System	2Q11	2Q11 Market Share	2Q10	2Q10 Market Share
	Units	(%)	Units	(%)
Android	46,775.9	43.4	10,652.7	17.2
Symbian	23,853.2	22.1	25,386.8	40.9
iOS	19,628.8	18.2	8,743.0	14.1
Research In Motion	12,652.3	11.7	11,628.8	18.7
Bada	2,055.8	1.9	577.0	0.9
Microsoft	1,723.8	1.6	3,058.8	4.9
Others	1,050.6	1.0	2,010.9	3.2
Total	107,740.4	100.0	62,058.1	100.0

Figura 5.1: Situazione del mercato mondiale dei dati del secondo quadrimestre del 2011 rispetto allo stesso periodo nel 2010 (Fonte: Gartner, Inc - agosto 2011)

Secondo quanto rilevato ad agosto dalla società di analisi **IDC**, la piattaforma Android si è affermata come la più diffusa con il 39% di market share, mentre a seguire vengono Symbian (con il 20%), iOS (con il 18%) e quindi Blackberry OS (14%) e Windows Mobile (4%). Il tutto a fronte di un numero di dispositivi venduti da maggio ad agosto 2011 che ammonta a 106 milioni di pezzi. I dati trovano una generale conferma dalle previsioni diffuse dagli analisti di Gartner, che registrano 108 milioni di smartphone venduti nel secondo quadrimestre 2011 ed una quota di mercato leggermente superiore per Android, dato al 43%, e inferiore per Windows Mobile, al 2%. La società **Canalys**, relativamente al secondo trimestre dell'anno, da Android addirittura al 50%, con il primato in ben 35 dei 56 paesi interessati dal rilevamento. Android, dopo aver sorpassato iOS, mantiene il primato e a spingerlo sono soprattutto i mercati emergenti, quello asiatico in particolare.

Le proiezioni sia di Gartner che di IDC per il mercato dei sistemi operativi per smartphone si spingono fino al 2015, indicando che il predominio di Android si rafforzerà ulteriormente: i primi dicono che Android diventerà il sistema operativo più popolare in tutto il mondo nel 2012, con uno share del 49,2 % per poi assestarsi al 48,8 % nel 2015, Gartner prevede che Apple iOS rimarrà la seconda piattaforma più grande in tutto il mondo fino al 2014, nonostante la sua quota calerà leggermente dopo il picco del 2011, dove la crescita del volume è ben al di sopra della media di mercato con una quota salita al 19,4% rispetto al 15,7% del 2010 (causato dall' aumento delle vendite nei principali mercati maturi come gli Stati Uniti e nell'Europa occidentale).

Table 1
Worldwide Mobile Communications Device Open OS Sales to End Users by OS (Thousands of Units)

OS	2010	2011	2012	2015
Symbian	111,577	89,930	32,666	661
Market Share (%)	37.6	19.2	5.2	0.1
Android	67,225	179,873	310,088	539,318
Market Share (%)	22.7	38.5	49.2	48.8
Research In Motion	47,452	62,600	79,335	122,864
Market Share (%)	16.0	13.4	12.6	11.1
iOS	46,598	90,560	118,848	189,924
Market Share (%)	15.7	19.4	18.9	17.2
Microsoft	12,378	26,346	68,156	215,998
Market Share (%)	4.2	5.6	10.8	19.5
Other Operating Systems	11,417.4	18,392.3	21,383.7	36,133.9
Market Share (%)	3.8	3.9	3.4	3.3
Total Market	296,647	467,701	630,476	1,104,898

Source: Gartner (April 2011)

Figura 5.2: Previsione del mercato mondiale dei sistemi operativi smartphone fino al 2015

Ciò riflette che Apple resterà interessata a mantenere i margini di profitto, piuttosto che perseguire una quota di mercato cambiando la sua strategia di

prezzo; questo continuerà a limitare l'adozione dei suoi dispositivi nei paesi emergenti. Analizzando più in dettaglio le stime della società di analisi si nota che Symbian era ancora il sistema operativo più diffuso fino all'anno scorso con uno share del 37,6% ma è ovviamente destinato gradualmente a sparire dalla classifica nel corso dei prossimi quattro anni: 19,2% (2011), 5,2% (2012) e 0,1% (2015). Le maggiori novità, per i prossimi anni, riguardano il sistema operativo di Microsoft che scalerà rapidamente la classifica grazie al matrimonio con Nokia; Windows Phone 7 a quota 4,2% nel 2010, crescerà dal 5,6% di quest'anno al 10,8% del 2012 fino a raggiungere il 19,5% nel 2015, diventando il terzo SO al mondo. BlackBerry, invece, è destinato a un lento declino, passando dal 13,4% di quest'anno al 11,1% nel 2015.

Le previsioni per il 2015 della società IDC mostrano uno scenario simile, dove si attribuisce ad Android una quota del 45,4%, mentre BlackBerry e iOS perderanno alcuni punti percentuali venendo scavalcati dal prorompente successo di Windows Phone di Microsoft (la nuova versione di Windows Mobile), forte dall'alleanza stretta con Nokia, che secondo gli analisti raggiungerà il 20%. Symbian invece scomparirà invece definitivamente dalla scena.

Worldwide Smartphone Operating System 2011 and 2015 Market Share and 2011-2015 CAGR

Operating System	2011 Market Share	2015 Market Share	2011-2015 CAGR
Android	39.5%	45.4%	23.8%
BlackBerry	14.9%	13.7%	17.1%
iOS	15.7%	15.3%	18.8%
Symbian	20.9%	0.2%	-65.0%
Windows Phone 7/Windows Mobile	5.5%	20.9%	67.1%
Others	3.5%	4.6%	28.0%
Total	100.0%	100.0%	19.6%

Source: IDC Worldwide Quarterly Mobile Phone Tracker, March 29, 2011

Figura 5.3: Previsione del mercato mondiale dei SO smartphone fino al 2015

Gli analisti si dichiarano fiduciosi che entro il 2015 Windows Phone dovrebbe scavalcare Blackberry e iOS, e questo proprio grazie agli smartphone finlandesi, passando dall'attuale 4% al 20%. Per ora i dati di Windows Phone sono poco incoraggianti (1,5 milioni di dispositivi venduti, con un calo del 52% sull'anno) ma c'è da dire che il nuovo sistema operativo a cui Nokia e Microsoft lavorano per sfidare Android sui dispositivi di basso costo, potrebbe determinare nuovi equilibri.

5.2 Mercato USA

Stando agli ultimi dati targati **Nielsen**, a luglio 2011, il 40% degli utenti mobile utilizzano uno smartphone, sono in crescita rispetto al 30% di un anno fa. Fra gli smartphone Android fa la parte del leone con il 40%, seguito da Apple iOS al 28% e RIM di Blackberry con il 19%.

Late adopters least sure of which operating system they want in their next smartphone

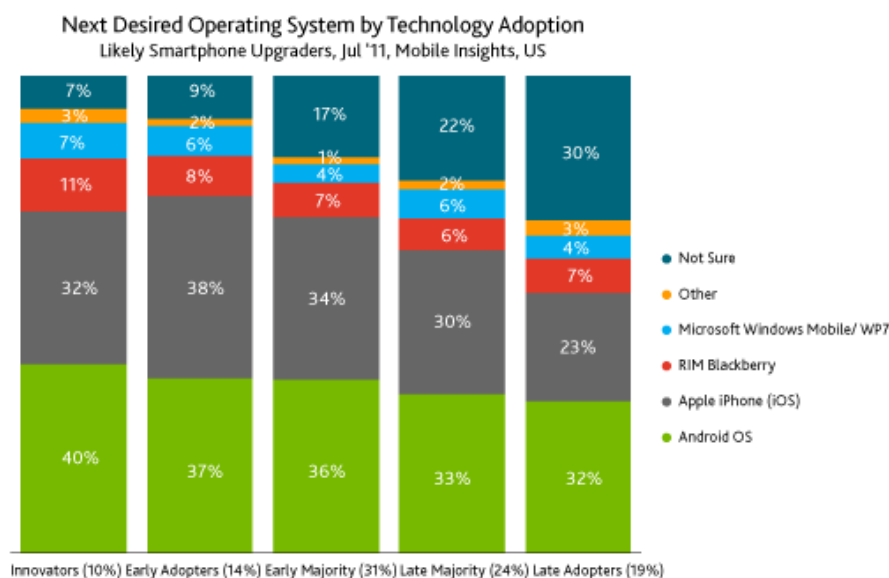


Figura 5.4: Il SO desiderato in base alla tipologia di consumatore

Analizzando attraverso il grafico, le attitudini degli utenti consumer verso le nuove tecnologie, possiamo osservare che sia gli innovators (anche detti: very early adopters) che gli early adopters (chi compra i dispositivi appena escono), ovvero persone con un ottimo grado di confidenza con le nuove tecnologie sono abbastanza convinti di ciò che tornerebbero ad utilizzare in futuro. I primi danno la maggior parte delle preferenze ad Android, con un 40%, mentre i secondi si affidano ad Apple, assegnandole un 38% contro il 37% di Android. È interessante notare l'assenza di RIM e Microsoft in questo grafico di Nielsen, fra il 6 e l'11%, BlackBerry non appare attraente per gli acquirenti. Ma il dato più significativo rimane quello dei late adopters, ovvero di chi si avvicina per la prima volta ad una nuova tecnologia; sebbene tendono a favorire Android al 32%, seguito dal 23% di Apple, c'è ancora un segmento molto ampio, il 30% degli intervistati, che si dichiara indeciso su quale sistema operativo adottare. In questa fascia potrebbe esserci Microsoft, con il suo Windows Phone 7.5 (non analizzato dato che a luglio 2011 non era ancora uscito). Sicuramente una grossa fetta di mercato è ancora indecisa e il sistema operativo che riuscirà a primeggiare in questa categoria avrà tutte le possibilità per diventare il leader nel settore.

Il rapporto di **comScore** sullo stato del mercato degli smartphone a stelle e strisce fa riferimento al trimestre conclusosi a luglio 2011, e su 82.200 persone che posseggono un dispositivo mobile di nuova generazione (+10% rispetto allo scorso trimestre), Google Android si classifica al primo posto dei sistemi operativi con il 41,8 % del mercato, in crescita di 5,4 punti percentuali. Apple ha rafforzato la propria posizione e si mantiene seconda con il 27,0 %, in crescita dell' 1,0 per cento rispetto allo stesso periodo di riferimento nel trimestre precedente; BlackBerry al terzo posto con 21,7 %, seguita da Microsoft (5,7 %) e Symbian (1,9 %).

Top Smartphone Platforms 3 Month Avg. Ending Jul. 2011 vs. 3 Month Avg. Ending Apr. 2011 Total U.S. Smartphone Subscribers Ages 13+ Source: comScore MobiLens			
	Share (%) of Smartphone Subscribers		
	Apr-11	Jul-11	Point Change
<i>Total Smartphone Subscribers</i>	100.0%	100.0%	N/A
Google	36.4%	41.8%	5.4
Apple	26.0%	27.0%	1.0
RIM	25.7%	21.7%	-4.0
Microsoft	6.7%	5.7%	-1.0
Symbian	2.3%	1.9%	-0.4

Figura 5.5: Dati comScore aggiornati a luglio 2011

I dati usciti il mese successivo, dimostrano il continuo successo del mercato smartphone, che continua ad aumentare del 10% portandosi a 84.5 milioni di persone che lo utilizzano. Invariata anche la situazione dei sistemi operativi, dove si può notare un ulteriore aumento di Android (dal 41.8 al 43.7 %) e iOS (dal 27.0 al 27.3 %), mantiene la stessa quota di mercato Microsoft mentre scendono sia BlackBerry (due punti percentuali in meno) che Symbian.

Top Smartphone Platforms 3 Month Avg. Ending Aug. 2011 vs. 3 Month Avg. Ending May 2011 Total U.S. Smartphone Subscribers Ages 13+ Source: comScore MobiLens			
	Share (%) of Smartphone Subscribers		
	May-11	Aug-11	Point Change
<i>Total Smartphone Subscribers</i>	100.0%	100.0%	N/A
Google	38.1%	43.7%	5.6
Apple	26.6%	27.3%	0.7
RIM	24.7%	19.7%	-5.0
Microsoft	5.8%	5.7%	-0.1
Symbian	2.1%	1.8%	-0.3

Figura 5.6: Dati comScore aggiornati ad agosto 2011

5.3 Mercato europeo

Un interessante rapporto di comScore è stato effettuato in 5 paesi della comunità europea, precisamente in Germania, Francia, Italia, Spagna e Regno Unito, la scoperta è stata che Android è cresciuto in un anno del 16,2% ed ora è la seconda piattaforma per telefoni smartphone più popolare nel vecchio continente.

Top Smartphone Platforms in EU5 by Share of Smartphone Users*			
3 Month Average Ending July 2011 vs. July 2010			
Total EU5 (DE, FR, IT, ES and UK) Mobile Subscribers, Age 13+			
Source: comScore MobiLens			
Smartphone Platform	Share (%) of EU5 Smartphone Users		
	Jul-10	Jul-11	Point Change
Total Smartphone Users	100.00%	100.0%	0.0
Symbian	53.9%	37.8%	-16.1
Google	6.0%	22.3%	16.2
Apple	19.0%	20.3%	1.2
RIM	8.0%	9.4%	1.5
Microsoft	11.5%	6.7%	-4.8

Figura 5.7: Dati comScore aggiornati a luglio 2011

Symbian possiede ancora il primo posto, anche se la decisione di Nokia di abbandonarlo come sistema operativo ha portato a un rapido declino della sua quota di mercato, scendendo di 16,1 punti percentuali rispetto al luglio 2010/2011; rimane davanti a rivali per il momento, ma con il 37,8% del mercato. La vasta gamma di telefoni Android disponibili sviluppati da diversi produttori, disponibili in diverse tipologie di prezzi accessibili, ha permesso al sistema operativo di emergere come il leader nel mercato degli smartphone europei, sorpassando in modo significativo la crescita del suo concorrente, Apple iOS, lo scorso anno. Android ha ottenuto un aumento di 16,2 punti percentuali in un anno, arrivando a una quota di mercato del 22,3%, contro solo il 6% dell'anno precedente; i produttori di HTC e Samsung hanno quasi due terzi del mercato dei dispositivi Android con HTC che ha il 34,6% e

Samsung al 31,7%. Apple e BlackBerry crescono leggermente dell'1,2% e dell'1,5%; Microsoft invece ha avuto un calo del 4,8% così da scendere dal terzo al quinto e ultimo posto dovute dalle vendite deludenti a seguito del lancio di Windows Phone 7.

Anche l'indagine condotta da Kantar Media, ha analizzato la situazione di mercato a luglio 2011 di 7 paesi, tra cui 5 paesi europei (UK, Germania, Francia, Spagna e Italia), i risultati mostrano invece Android come il sistema più diffuso.

KANTAR MEDIA | OPERATIONAL
COMTECH > Real Consumer Connection

OS (Operating System) Share - Smartphone Sales

	12 w/e 11 Jul 2010	12 w/e 10 Jul 2011	Change		12 w/e 11 Jul 2010	12 w/e 10 Jul 2011	Change
	%	%	%		%	%	%
GB	100.0%	100.0%	0.0	US	100.0%	100.0%	0.0
Symbian	31.7	7.6	-24.1	Symbian	9.9	0.6	-9.3
RIM	18.9	19.2	0.3	RIM	31.8	8.1	-23.7
iOS	25.3	20.2	-5.1	iOS	20.4	28.6	8.2
Windows Mobile 7	0.0	1.7	1.7	Windows Mobile 7	0.0	2.1	2.1
WinMobile	3.9	1.0	-2.9	WinMobile	8.8	1.2	-7.6
Android	18.6	48.8	30.2	Android	24.2	57.0	32.8
Bada	0.2	1.2	1.0	Bada	0.0	0.0	0.0
Other	1.5	0.3	-1.2	Other	4.8	2.3	-2.5
Germany	100.0%	100.0%	0.0	Australia	100.0%	100.0%	0.0
Symbian	52.1	19.4	-32.7	Symbian	45.8	17.3	-28.5
RIM	4.1	3.9	-0.2	RIM	9.3	1.6	-7.7
iOS	24.6	20	-4.6	iOS	32.9	39.2	6.3
Windows Mobile 7	0.0	7.1	7.1	Windows Mobile 7	0.0	2.6	2.6
WinMobile	11.1	1.1	-10.0	WinMobile	6.0	0.7	-5.3
Android	4.8	44.3	39.5	Android	4.7	38.0	33.3
Bada	0.5	3	2.5	Bada	0.0	0.3	0.3
Other	2.7	1.3	-1.4	Other	1.3	0.3	-1.0
France	100.0%	100.0%	0.0	Italy	100.0%	100.0%	0.0
Symbian	27.2	17.2	-10.0	Symbian	68.3	35.1	-33.2
RIM	11.7	9.1	-2.6	RIM	5.4	5.3	-0.1
iOS	33.1	20.3	-12.8	iOS	13.7	22.0	8.3
Windows Mobile 7	0.0	1.6	1.6	Windows Mobile 7	0.0	1.5	1.5
WinMobile	6.8	1.3	-5.5	WinMobile	9.8	3.9	-5.9
Android	19.7	41.0	21.3	Android	1.3	27.6	26.3
Bada	0.0	9.6	9.6	Bada	0.0	4.6	4.6
Other	1.5	0.0	-1.5	Other	1.6	0.0	-1.6
Spain	100.0%	100.0%	0.0	Spain	100.0%	100.0%	0.0
Symbian	73.7	30.1	-43.6	Symbian	73.7	30.1	-43.6
RIM	8.6	10.4	1.8	RIM	8.6	10.4	1.8
iOS	3.9	7.1	3.2	iOS	3.9	7.1	3.2
Windows Mobile 7	0.0	2.6	2.6	Windows Mobile 7	0.0	2.6	2.6
WinMobile	1.9	2.0	0.1	WinMobile	1.9	2.0	0.1
Android	10.7	47.9	37.2	Android	10.7	47.9	37.2
Bada	0.0	0.0	0.0	Bada	0.0	0.0	0.0
Other	1.1	0.0	-1.1	Other	1.1	0.0	-1.1

Figura 5.8: Dati Kantar divisi per paesi aggiornati a luglio 2011

Android si conferma leader in Inghilterra, Francia, Germania e Spagna con quote di mercato che si aggirano intorno al 46%. In Inghilterra Android ha la percentuale di mercato più alta con 48,8% , l'aumento maggiore in percentuale rispetto allo scorso anno però ce l'ha la Germania (+ 39% in confronto a luglio 2010). Il secondo sistema operativo rimane iOS con un 20% in tutti i paesi, tranne la Spagna dove ha solo il 7% e il secondo SO è Symbian (il quale ha paradossalmente anche il calo maggiore, ben 43,6%). Al terzo posto si alternano BlackBerry in Inghilterra (19,2%) e Spagna (10,4%); e Symbian in Germania (19,4%) e Francia (17,2%); mentre lo stato in cui Windows phone ha la quota maggiore è la Germania con il 7,1%.

Tra gli altri stati coinvolti nell'indagine da notare l'Australia dove Android è secondo al 38% e iOS è il sistema operativo più diffuso con il 39,2%. Non è stata volutamente analizzata la situazione italiana perchè verrà discussa nel paragrafo successivo.

5.4 Mercato italiano

L'analisi di Kantar, mostra la situazione italiana, che a differenza del resto dell'europa ha Symbian al primo posto dei sistemi operativi per smartphone; il sistema operativo di Nokia a Luglio 2011 aveva il 35,1% del mercato, la metà se si considera la quota del 68,3% dello stesso periodo del 2010, ma nonostante questo Symbian mantiene la leadership, tallonato da Android con il 27,6% (il sistema di Google in un anno è cresciuto di 26,3 punti percentuali); la terza piazza è per iOS dell' iPhone, forte del 22% del mercato e cresciuto in un anno di 8,3 punti. Al quarto posto c'è BlackBerry che, nonostante perda lo 0,1%, si mantiene con il 5,3% del mercato italiano.

I dati Nielsen sull'utilizzo del mobile in Italia nel primo trimestre del 2011 (dati mensili) rilevano una costante crescita del numero di individui che accedono ad internet dal proprio cellulare: 13 milioni a inizio 2011, il 34% in più rispetto allo stesso periodo del 2010 e oltre 5 milioni in più rispetto al primo trimestre 2009.

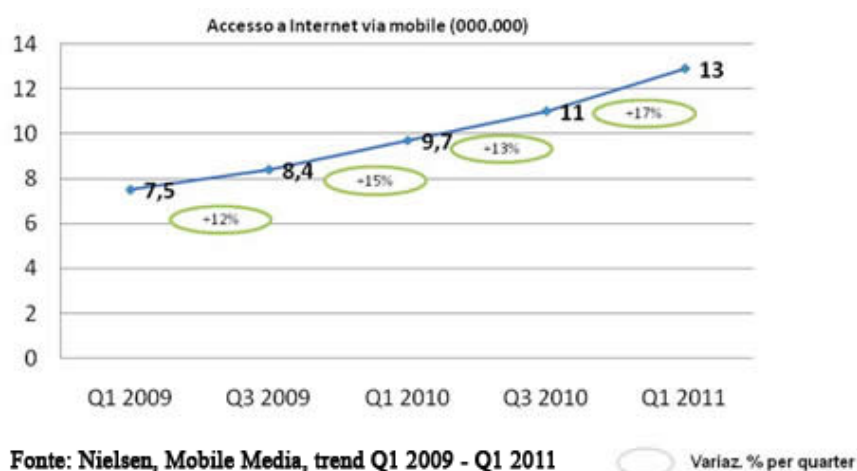


Figura 5.9: Il trend di crescita dell'Internet Mobile in Italia

Dalla rilevazione Nielsen appare evidente la correlazione tra la crescita degli accessi ad internet da cellulare e la crescente diffusione degli smartphone in Italia. I possessori di telefoni cellulari di ultima generazione hanno superato infatti i 20 milioni di individui, con una crescita del 52% rispetto allo stesso periodo del 2010. Il grafico sottostante evidenzia come, a questi tassi di crescita, il sorpasso degli smartphone sui telefoni tradizionali potrebbe avvenire entro la fine del 2011.

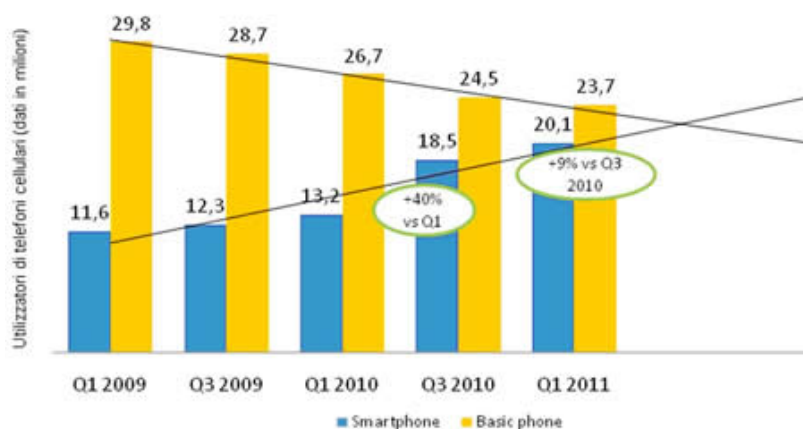


Figura 5.10: La crescita degli smartphone in Italia e la previsione dell'imminente sorpasso (Fonte: Nielsen, Mobile Media)

Nell'ultimo anno la diffusione di smartphone ha registrato un tasso di crescita senza precedenti: a inizio 2011 ci sono ben 7 milioni di possessori in più rispetto allo stesso periodo dello scorso anno, è interessante soprattutto notare che i nuovi utilizzatori di smartphone si distribuiscono anche in quelle fasce della popolazione tipicamente meno coinvolte e meno digitalizzate: crescono infatti le donne, che arrivano al 44% del totale (+5 punti percentuali) e aumenta l'età media dei possessori di smartphone, che passa dai 40,4 anni di inizio 2010 agli attuali 43,3 anni". Per quanto riguarda i sistemi operativi più diffusi, Symbian di Nokia mantiene un'ampia leadership ma negli ultimi sei mesi registra un calo che lo porta al 68% di quota di mercato. L'Apple iPhone OS continua a crescere (+29,6%) raggiungendo una quota dell'11%; ma il fenomeno più interessante che si sta registrando è la fortissima crescita di Android OS, che vede più che triplicare la propria quota di mercato in 6 mesi, passando dall'1,8% al 7,4%. Perdono invece terreno Windows Mobile, in attesa degli effetti della partnership con Nokia, e Blackberry OS, che si attestano entrambi sotto il 5% di quota.

Ma quali sono i fattori che influenzano maggiormente questi trend? Analizzando i driver per la scelta di uno smartphone, subito dopo gli elementi standard che guidano l'acquisto di qualsiasi cellulare (precedente esperienza con il brand, prezzo e facilità di utilizzo) e che pesano per il 45% circa, troviamo gli elementi che tipicamente caratterizzano gli smartphone, come la connessione wi-fi, il sistema operativo, la dimensione e facilità di lettura del display oltre al fatto che sia touchscreen e le applicazioni disponibili; questi fattori pesano per circa il 30% nella scelta dello smartphone, mentre il restante 25% è composto da elementi come la forma, il design, la fotocamera, la durata della batteria, la dimensione, etc. Analizzando invece la soddisfazione degli utenti troviamo una forte correlazione tra i trend di mercato e la soddisfazione relativa agli Application Store; risulta infatti che le applicazioni disponibili sono il vero punto di forza di Android e il principale driver della grande crescita registrata negli ultimi sei mesi da questo sistema operativo. L'81% dei possessori è infatti molto soddisfatto dell'Android Mar-

ket, contro il 70,5% di soddisfazione per l'Apple Store. Più basse le quote di OVI (l'Application Store di Symbian) e di Blackberry, con il 45,5% e il 37,3% rispettivamente.

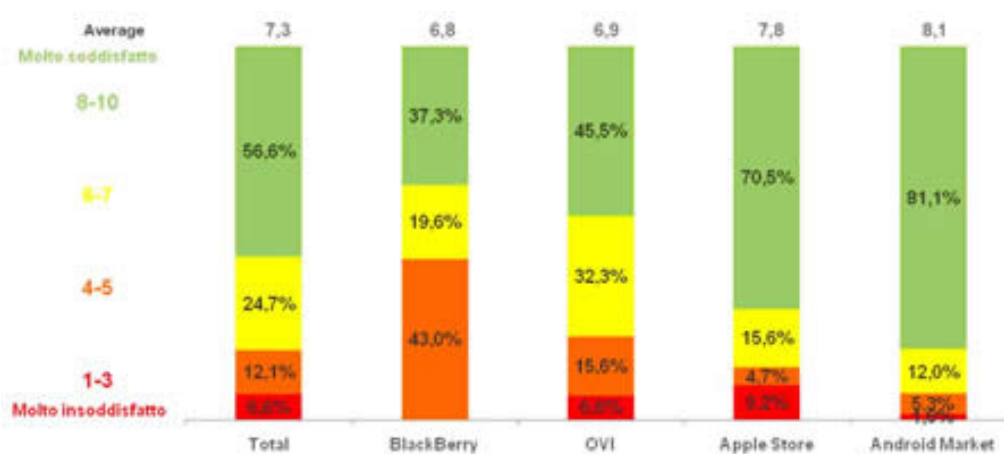


Figura 5.11: La soddisfazione verso l'Application store (Fonte: Nielsen, Mobile Media, Q1 2011)

L'utente di telefonia mobile si fa dunque sempre più evoluto, in particolare se in possesso di uno smartphone. Accede sempre più spesso a internet dal proprio cellulare, tanto che i navigatori da mobile su base mensile sono circa la metà di quelli che accedono da PC (26,6 milioni ad Aprile 2011 secondo i dati Audiweb powered by Nielsen), e fa scelte molto precise in base a bisogni sempre più specifici: applicazioni prima di tutto e in generale l'acquisto di un device che permetta un accesso facile e veloce a tutti i contenuti.

Le stesse cifre sono emerse anche durante ThinkMobile, il primo evento dedicato all'ecosistema mobile organizzato da Google in Italia; sono cifre che potrebbero essere utili ad aziende e sviluppatori per trasformare il successo del comparto mobile in opportunità di business. Il fenomeno Smartphone rappresenta un'opportunità di crescita importante per l'economia dell'intero Paese.

Secondo lo studio commissionato da Google a “Ipsos MediaCT” nel luglio 2011, lo smartphone è ormai centrale nella relazione per gli Italiani. Il 60% raramente si separa dal proprio smartphone e lo utilizza quasi ovunque: in casa (98%), fuori casa (88%), in ufficio (77%). Lo smartphone viene adoperato spesso anche in parallelo ad altri media: mentre si ascolta musica (47%), mentre si guarda la TV (32%) e, nel 30% dei casi, persino quando si naviga su Internet con un altro dispositivo.

Circa la metà degli intervistati (45%) lo utilizza per recuperare rapidamente informazioni sul web o per sfruttare utilmente pause e tempi morti (53%), come per esempio in attesa del bus o del treno. Lo smartphone è dunque entrato a pieno titolo nel vissuto quotidiano degli italiani.

La ricerca Ipsos MediaCT risponde anche alla domanda “Che rapporto lega gli italiani alle app?”, rivelando che gli italiani con lo smarphone possiedono in media 19 applicazioni, di cui 4 sono state acquistate. Sette apps sono usate mensilmente con una certa frequenza: il 39% degli intervistati ha intenzione di utilizzare le apps con maggiore frequenza nell’arco dei prossimi 12 mesi. Il 23% dei possessori di smartphone ha già comprato in mobilità. L’M-commerce in Italia, a differenza dell’ e-commerce non si discosta troppo dagli Stati Uniti (29%) e da Gran Bretagna e Germania (28%), anzi è superiore alla media francese (17%).

5.5 Il mercato smartphone rispetto agli altri settori

Finora abbiamo confrontato i vari sistemi operativi smartphone tra loro, ma all’interno di un’azienda quanto vale il mercato smartphone rispetto alle altre attività di settore? Cerchiamo di capirlo prendendo gli ultimi dati delle vendite di Apple (per l’ iPhone con iOS) e di Samsung (per i suoi smartphone che utilizzano principalmente Android).

Il settore delle telecomunicazioni guida la crescita dei ricavi nel secondo trimestre di **Samsung**, con vendite in aumento del 43 per cento rispetto

all'anno scorso (generando un utile di 1,5 miliardi di dollari), causato da una forte domanda di smartphone come il Samsung Galaxy S e il Samsung Galaxy S II. Il settore Digital Media & Elettrodomestici (che comprende TV, soluzioni informatiche e apparecchi digitali) ha visto aumentare le vendite rispetto al trimestre precedente, sostenuta dalla domanda per televisori a schermo piatto, ma rispetto all'anno precedente scende del 5%. Diminuiscono anche il settore dei semiconduttori (che comprende la produzione di memorie come i banki RAM ecc...) e dei Panel Display (ovvero i monitor, gli schermi dei vari notebook e delle TV). L'utile operativo è diminuito dall'anno scorso in tutti i settori, a parte quello degli smartphone che ha supportato la crisi e ha portato ad un aumento delle vendite generale del 4%.

Segment Information

Sales by Segment

(Unit: Trillion Won)

	2Q '11	Y-on-Y	2Q '10	1Q '11
Semiconductor	9.16	4%↓	9.53	9.18
- Memory	5.89	12%↓	6.71	5.87
DP	7.09	9%↓	7.76	6.51
- LCD	5.76	15%↓	6.80	5.27
Telecom	12.18	43%↑	8.52	10.64
- Mobile	11.69	45%↑	8.05	10.14
DM & A	14.07	5%↓	14.80	13.52
- VD	7.86	11%↓	8.85	7.68
Others	△3.06	-	△2.72	△2.86
Total	39.44	4%↑	37.89	36.99

* Sales of each segment include intersegment sales

Figura 5.12: Analisi dei guadagni Samsung nel Q2 2011 (fonte: Samsung Electronics - July 2011)

Analizzando le vendite **Apple**, che produce solo prodotti informatici (non come Samsung che ha molti più settori) possiamo notare come le vendite di iPhone e iPad (i punti 4 e 5) siano incredibilmente superiori rispetto alle vendite dei computer desktop (che include iMac, Mac mini, Mac Pro e Xserve) e dei computer portatili (che includono MacBook, MacBook Air e MacBook Pro). I risultati finanziari del quarto trimestre fiscale 2011 conclusosi il 24 Settembre 2011 hanno mostrato che i dispositivi mobili Apple quali iPhone e

iPad hanno avuto un incremento del 21 e del 166 % rispetto allo scorso anno, vendendo in totale più di 28 milioni di dispositivi producendo un fatturato di 17,8 miliardi di dollari (rispetto ai 28,27 totali); il settore dei computer fissi e portatili ha venduto 4,9 milioni di dispositivi per un guadagno di quasi 6,3 miliardi di dollari. Rispetto alla Samsung possiamo notare che la situazione di Apple è totalmente differente, dove il solo settore a calare rispetto all'anno scorso è quello degli iPod, la quale funzione è stata completamente rimpiazzata dagli iPhone, mentre tutti gli altri sono in crescita. Comunque anche in questo caso il settore degli smartphone e dei dispositivi mobili produce più del doppio del fatturato dei restanti settori.

Apple Inc.
Q4 2011 Unaudited Summary Data

Product Summary	Q3 2011		Q4 2010		Q4 2011		Year/Year Change	
	Units K	Revenue	Units K	Revenue	Units K	Revenue	Units	Revenue
		\$M		\$M		\$M		
Desktops (1)	1,155	\$1,580	1,242	\$1,676	1,278	\$1,687	3%	1%
Portables (2)	2,792	3,525	2,643	3,194	3,616	4,585	37%	44%
Subtotal CPUs	3,947	5,105	3,885	4,870	4,894	6,272	26%	29%
iPod	7,535	1,325	9,051	1,477	6,622	1,103	- 27%	- 25%
Other Music Related Products and Services (3)		1,571		1,243		1,678		35%
iPhone and Related Products and Services (4)	20,338	13,311	14,102	8,822	17,073	10,980	21%	24%
iPad and Related Products and Services (5)	9,246	6,046	4,188	2,792	11,123	6,868	166%	146%
Peripherals and Other Hardware (6)		517		477		640		34%
Software, Service and Other Sales (7)		696		662		729		10%
Total Apple		\$28,571		\$20,343		\$28,270		39%

Figura 5.13: Analisi dei guadagni Apple nel Q4 2011 (fonte: Apple Press Info)

Conclusioni

Secondo la legge formulata dall'ingegnere statunitense Gordon Bell nel 1972, chiamata la Legge di Bell sostiene che circa ogni 10 anni compare una nuova classe di computer più economici e piccoli, che creano nuove applicazioni, nuovi mercati e nuovi settori nell'industria. Quando i cosiddetti minicomputer apparvero per la prima volta nel 1970 e soppiantarono i mainframe sia per le dimensioni che per il prezzo, l'arrivo dei personal computer, i notebook, gli smartphone e i tablet sono stati la conferma di quanto profetizzato da Bell con la sua legge.

La situazione attuale infatti mostra che già ci sono più cellulari che computer collegati a Internet, stiamo quindi assistendo ad un cambiamento nel modo in cui le persone utilizzano i computer. Le applicazioni desktop che usiamo più di frequente si sono centrate attorno alla comunicazione, piuttosto che al compito più tradizionale dei personal computer di creare dei documenti; sia nel mondo del lavoro che nella nostra vita personale, e queste attività di comunicazione vengono tradotte efficacemente nelle applicazioni mobili.

Come lo smartphone guadagnerà un'adozione diffusa, i computer desktop saranno destinati agli specialisti e alle élite professionali. Sempre più attività di routine che si eseguono attualmente su un computer desktop o portatile, saremo in grado di realizzarle su uno smartphone, ma ancora più importante, le nuove applicazioni saranno in grado di soddisfare le esigenze di persone che non hanno mai utilizzato un computer oggi. Lo sviluppo del software si sposterà verso lo sviluppo mobile come la maggioranza delle per-

sone che usano i computer li utilizzerà indirettamente attraverso un telefono cellulare. Il centro di gravità dell'industria del software sarà destinato verso le applicazioni mobili.

Anche alla fine degli anni '90, lo sviluppo mobile era considerato un mercato caldo; la maggior parte dei primi dispositivi smartphone e PDA (come l'Apple Newton Message Pad, Pilot Palm, Palm e Pocket PC Windows) supportava l'installazione di applicazioni, ma il processo di installazione delle applicazioni risultava scomodo e la maggior parte degli utenti finali non aggiungevano applicazioni al telefono.

L'iPhone ha rivitalizzato il mondo dello sviluppo di applicazioni mobili; Apple ha creato un modo facile per l'acquisto e l'installazione di applicazioni di terze parti, e cosa più importante, ha promosso la capacità dei propri utenti e potenziali clienti. I sistemi operativi smartphone si innovano attivamente per stare al passo con i progressi nello sviluppo hardware con strumenti sempre migliori e API. Spesso le innovazioni più significative non sono puramente tecniche; l'App Store infatti ha ridotto gli ostacoli allo sviluppo di applicazioni, fornendo un facile accesso alla distribuzione; non c'è da sorprendersi che le persone sviluppino più applicazioni in cui vi sia un mercato accessibile e un buon canale di distribuzione.

Dopo aver analizzato i vari sistemi operativi nel dettaglio e la loro situazione sul mercato si è potuto constatare che dopo un periodo in cui iOS era il sistema operativo di riferimento, ora Android l'ha raggiunto e domina con esso il settore.

Nonostante i grandi miglioramenti di Symbian con le ultime versioni, per l'intero mercato la piattaforma risulta obsoleta, sia in termini di interfaccia che nei software integrati; si è cercato di colmare il gap con i rivali, ma le differenze continuano ad esserci. È funzionalmente inferiore a iOS 4 (ad esempio in termini di giochi, servizi sociali, interfaccia, funzionalità di posta elettronica, la quantità di software di terze parti) e ad Android (ad esempio in termini di interfaccia, rubrica e funzioni aggiuntive di calendario). Il numero di aggiornamenti in questa versione di Symbian non sono stati sufficienti a

qualificarsi per il suo nome e per ritornare il leader come un tempo, rimane la scelta di Nokia di utilizzarlo per i suoi dispositivi di fascia bassa (come la nuova linea di smartphone Nokia Asha), mentre per quelli di fascia alta (come i nuovi Nokia Lumia) utilizza Windows Phone 7.5.

Blackberry continua ad essere la soluzione più desiderata dalle aziende, per via della sua tecnologia che aiuta la gestione dei contatti e delle mail in tutta sicurezza, ma per il resto dei consumatori, è un sistema operativo privo di molte funzioni, anche se con la versione 7 del sistema operativo ha inserito molte caratteristiche (come la tecnologia NFC) per cercare di raggiungere il livello di Android e iOS, il sistema operativo di casa RIM rimane penalizzato da un prezzo eccessivo dei suoi dispositivi e da una disponibilità di applicazioni non paragonabile al mondo Android e iOS, rimane per un mercato di nicchia.

Windows Phone 7, può avere in linea teorica tutte le carte per arrivare alla pari con i due leader, ma l'uscita tardiva nel mercato degli smartphone affiancata a una forte restrizione hardware di Microsoft per gli OEM hanno penalizzato molto la sua diffusione, il quale lo si può trovare in pochi dispositivi; dato che Microsoft non produce propri dispositivi come iOS o BlackBerry, questa forte restrizione non è sembrata una mossa vincente, guardando soprattutto la filosofia opposta di Android che, nonostante il rischio di frammentazione, lo si può trovare installato nella maggior parte dei dispositivi mobili in commercio.

Come numero di smartphone in commercio anche iOS e Blackberry, avendo la propria produzione interna del dispositivo, da un lato offrono una maggiore ottimizzazione dell'hardware per il SO ma dall'altro offrono poche alternative d'acquisto; Symbian nonostante sia opensource viene ormai utilizzato solo nei dispositivi Nokia, quindi non può neanche sfruttare il vantaggio che contraddistingue invece Android.

Il fattore che sembra però essere il più importante dal punto di vista dei consumatori è la quantità di applicazioni disponibili, dove iOS e Android non hanno rivali, mentre WP7, BlackBerry e Symbian coprono tutte le fun-

zioni base, ma senza la varietà di scelta che sembra il valore aggiunto per i consumatori.

Per gli sviluppatori di applicazioni entrambi gli ecosistemi offrono vantaggi e svantaggi; se si sceglie di creare applicazioni per iOS qualsiasi sviluppatore, indipendente o azienda già affermata, deve acquistare una licenza annua, al prezzo di 99 \$ (79 euro per noi europei), senza la licenza non sarà possibile pubblicare le proprie creazioni sull' App Store; oltre a disporre della licenza ufficiale, per vedere pubblicate le proprie creazioni, ogni applicazione deve passare anche sotto il controllo di Apple, che decide se accettarle o meno. Le applicazioni infatti, devono rispettare il contratto del SDK, che prevede alcune limitazioni: è vietato, ad esempio, duplicare funzioni già presenti nel S.O. dell'iPhone (come l'invio di SMS e MMS), applicazioni mal funzionanti o che fanno un eccessivo utilizzo delle connessioni telefoniche, che entrino in competizione diretta con i prodotti Apple, oppure quelle con contenuto a sfondo sessuale o violento. Ciò offre un controllo di sicurezza molto rigido ma limita molto la libertà degli sviluppatori. Sviluppare applicazioni per Android, al contrario lascia piena libertà di sviluppo, senza nessuna restrizione, ciò pone molti rischi di distribuzione di applicazioni nocive, ma non pone limite allo sviluppo, in grado di creare applicazioni in certi casi anche migliori di quelle native, permettendo di migliorare i servizi offerti dal sistema operativo. Lo svantaggio di Android però consiste nella forte frammentazione del suo sistema operativo per i vari dispositivi mobili, varie versioni del SO all'interno di smartphone di marche diverse con risoluzioni e capacità diverse mettono in grande difficoltà gli sviluppatori costretti a testare le proprie applicazioni su vari dispositivi con caratteristiche molto diverse tra loro; la notizia recente di Sports Interactive, che farà uscire il suo gioco manageriale calcistico Football Manager 2012 solo su iOS per problemi di frammentazione nel mercato Android, conferma che sta diventando troppo difficile realizzare una versione che funzioni bene su tutti i differenti hardware che utilizza il sistema operativo di Google; se ci sono sei o sette versioni di Android, c'è la necessità di fare sei o sette tipi di test differenti, e ciò richiederebbe troppo tempo, il prob-

lema non è tanto lo sviluppo effettivo delle applicazioni, ma sta diventando testarli su tutti i formati a disposizione. Questo fattore sta diventando sempre più problematico per Android, gli sviluppatori infatti sostengono che la frammentazione è un vero e proprio ostacolo che impedisce il pieno sviluppo della piattaforma, rischiando di disincentivare i programmatori a sviluppare applicazioni per la loro piattaforma.

Diventare i leader nel settore significa avere la quota di mercato maggiore e quindi vendere più dispositivi rispetto ai rivali, il mercato occidentale ormai è maturo dato che la maggior parte della popolazione con una condizione economica medio-alta, possiede già un dispositivo mobile e difficilmente sarà intenzionato a cambiare il tipo di SO (chi ha comprato uno smartphone Android difficilmente tenderà a cambiare per iPhone o viceversa); per incrementare la propria fetta di mercato quindi bisogna mirare a una tipologia di clienti che ancora non hanno uno smartphone, una grande percentuale di potenziali clienti proviene dai paesi emergenti, per esempio il Sudamerica, la Cina e l'India, oppure dai consumatori occidentali in condizioni economiche inferiori. La principal analyst della società di analisi Gartner ha dichiarato che "Entro il 2015, il 67 per cento di tutti i dispositivi con sistemi operativi open avrà un prezzo medio di vendita di \$ 300 o inferiore, a dimostrazione che gli smartphone sono stati finalmente davvero democratizzati, il prezzo diminuirà e ci sarà un ulteriore vantaggio per i consumatori". La posizione di Android nella fascia alta del mercato rimarrà forte, ma la sua occasione più grande nel lungo periodo sarà quella negli smartphone a medio-basso costo, soprattutto nei mercati emergenti, cosa che Apple per la sua filosofia di mercato non potrà mai fare, producendo prodotti solo per una tipologia di cliente con condizioni economiche medio-alte.

Cercare di differenziare il proprio prodotto suddividendo i propri smartphone in due categorie di prezzo può essere la mossa vincente per Android, cercando di eliminare il più possibile il problema della frammentazione, magari portando tutti i dispositivi di fascia bassa a una versione di Android (per esempio Froyo) e aggiornare i dispositivi di fascia alta alla ultima versione.

Uno scenario non ancora affrontato in questo elaborato, ma che sta diventando sempre più possibile, è quello della convergenza tra i sistemi operativi per smartphone con quelli per computer. Questa transizione in parte pare già in atto, infatti come possiamo notare i sistemi operativi mobili stanno diventando sempre più potenti, mentre quelli per desktop sempre più leggeri, in più i due colossi Microsoft e Apple con le uscite dei loro nuovi sistemi operativi per desktop hanno mostrato grafiche e interfacce grafiche sempre più simili a quelle dei loro rispettivi smartphone. Windows ha presentato l'anteprima del nuovo Windows 8, e la novità più eclatante consiste nella nuova interfaccia denominata Metro che porterà sui PC desktop e sui notebook la grafica a piastrelle già vista su Windows Phone 7; Apple con l'ultimo aggiornamento del proprio sistema operativo per i propri computer (OS X Lion), ha introdotto sul desktop alcune schermate identiche a ciò che possiamo trovare sugli iPhone e gli iPad, insieme ad alcune gestive che permettono di sfogliare le schermate come fossero pagine. Apparentemente, dunque, Microsoft e Apple stanno andando nella stessa direzione: portare la grafica ed il modo di usare i dispositivi mobili, che sono sempre più usati, anche sul desktop. Gli scopi però sono opposti, Apple cerca di attrarre ai propri computer la clientela di iPhone e iPad, in modo da incrementare le vendite dei dispositivi fissi (portare i clienti mobile sul desktop); Microsoft invece che ha il dominio dei pc, inserendo l'interfaccia di Windows Phone in Windows 8 spera che gli utenti desktop, apprezzino la nuova interfaccia metro e si convincano ad acquistare gli smartphone che adotteranno Windows Phone (portare i clienti desktop sul mobile).

Questa convergenza, viene accelerata anche dall'implementazione del cloud computing, ora che i dati di un utente non devono essere necessariamente legati ad un particolare dispositivo, ci si aspetta che i dati siano accessibili in egual misura su dispositivi multipli e la memoria limitata degli smartphone diventerà un problema sempre minore. Non si tratta di una visione condivisa, ci sono ancora molti pareri che pensano che desktop e mobile resteranno divisi, sicuramente la situazione attuale presenta due mondi ancora ben dis-

tinti, e la situazione più probabile di un immediato futuro può essere quella di avere l'uso professionale sul desktop e quello personale su smartphone, ma se in un futuro più remoto avverrà questa convergenza, si ridurranno anche i costi di ricerca e sviluppo da parte dei produttori di sistemi operativi e si ridefiniranno nuovi equilibri.

Bibliografia

- [1] Sarah Allen, Vidal Grauper, Lee Lundrigan. *Pro Smartphone Cross-Platform Development*. Apress, 2010.
- [2] Carlo Pelliccia. *Android Programming*. PuntoInformatico Libri, Edizioni Master.
- [3] Andrea Leganza. *iPhone Programming*. PuntoInformatico Libri, Edizioni Master.
- [4] M. Carli. *Android - Guida per lo sviluppatore*. APOGEO, 2010.
- [5] Erica Sadun. *The iPhone Developer's Cookbook*. Addison-Wesley.
- [6] *iOS Human Interface Guidelines*. Apple Inc. 2010.
- [7] Andrea Busi. *Tutorial pratici per iphone SDK*.
- [8] Pietro Brunetti. *CocoaTouch Layer Memory Management guidelines*.
- [9] Stanislav Pavlov, Pavel Belevsky. *Windows Embedded CE 6.0 Fundamentals*. Microsoft.
- [10] Microsoft. *Microsoft developer center*. msdn.microsoft.com
- [11] Apple Inc. *iOS Developer Library*. <http://developer.apple.com/devcenter/ios/index.action>
- [12] Google Inc. *Android Developers*. <http://developer.android.com/index.html>
- [13] Nokia Corporation. *Nokia Developer*. <http://www.developer.nokia.com/>

- [14] Research In Motion. *BlackBerry Developer Zone*.
<http://us.blackberry.com/developers/>
- [15] Research In Motion. *BlackBerry Planet Web Support Site*.
<http://www.blackberryplanetbook.com/>
- [16] Tim Bray. *Memory Analysis for Android Applications*. Android Developers Blog, 24 March 2011.
- [17] Prasad Naik. *Smartphone OS Comparison*. <http://www.techtree.com>, 31 Jan 2011.
- [18] Azmi Sayadi. *The Ultimate Mobile OS Comparison*. <http://www.gigalb.com>, 09 Feb 2011.
- [19] Valeria Guastamacchia. *Storia ed evoluzione del telefono cellulare*. <http://www.cellularmagazine.it>, 11 Oct 2009.
- [20] Andrea Bai. *Gli italiani e lo smartphone: oltre 20 milioni di utenti*. <http://www.businessmagazine.it>, 26 Sept 2011.
- [21] Gartner, Inc. *Gartner Says Sales of Mobile Devices in Second Quarter of 2011 Grew 16.5 Percent Year-on-Year; Smartphone Sales Grew 74 Percent*. Egham, UK, August 11, 2011.
- [22] Gartner, Inc. *Gartner Says Android to Command Nearly Half of Worldwide Smartphone Operating System Market by Year-End 2012*. Egham, UK, April 7, 2011.
- [23] IDC Corporate USA. *IDC Forecasts Worldwide Smartphone Market to Grow by Nearly 50% in 2011*. FRAMINGHAM, Mass. March 29, 2011.
- [24] comScore, Inc. *comScore Reports July 2011 U.S. Mobile Subscriber Market Share*. RESTON, VA, August 30, 2011.
- [25] comScore, Inc. *comScore Reports August 2011 U.S. Mobile Subscriber Market Share*. RESTON, VA, October 5, 2011.

-
- [26] comScore, Inc. *Android Captures 2 Ranking Among Smartphone Platforms in EU5* . LONDON, UK, 13 September 2011.
 - [27] The Nielsen Company (Italy) S.r.l. *Continua a crescere la navigazione da mobile in Italia* . June 27, 2011.
 - [28] The Nielsen Company S.r.l. *40 Percent of U.S. Mobile Users Own Smartphones; 40 Percent are Android*. September 01, 2011.
 - [29] Apple Press Info. *Apple Reports Fourth Quarter Results*. Apple Inc, October 18, 2011.
 - [30] Samsung Financial Information. *Q2 2011 Earning release*. Samsung electronics, July 29, 2011.