

ANDROID



Hauptseminar - Mobile Computing Platforms

Dennis Reuling

dennis.reuling@student.uni-siegen.de

Master Informatik Nebenfach Elektrotechnik

Inhaltsverzeichnis

1	Einleitung und Motivation	2
2	Grundlagen und Geschichte	3
3	Grundziele und Herausforderungen	4
3.1	Mobilität	5
3.2	Geschwindigkeit	5
3.3	Offenheit	6
3.4	Heterogenität	6
3.5	Sicherheit	6
4	Architektur	6
5	Möglichkeiten und Vergleich	10
5.1	Hardware	10
5.2	Software / Anwendungen	11
6	Reflektion und Ausblick	14

1 Einleitung und Motivation

Diese Seminararbeit behandelt das Betriebssystem *Android*. Es wird versucht dem Leser den aktuellen Stand des Betriebssystems sowie dessen Grundlagen näher zu bringen. Sie orientiert sich an vielen Quellen, die in die einzelnen Themenbereiche intensiver und gründlicher eingeht, als das hier auf den wenigen Seiten getan werden kann. Diese Seminararbeit bietet quasi einen Einstieg in das komplexe Thema der mobilen Betriebssysteme, genauer *Android*, geht jedoch nicht zu sehr in die Tiefe, sondern versucht durch die Breite einen Überblick zu verschaffen. Durch diesen Querschnitt kann der Leser selbst entscheiden, in welchem Themengebiet er sein Wissen vertieft.

Als erstes wird auf die Grundlagen des Betriebssystems eingegangen. Um einen gesamten Überblick von *Android* bieten zu können, wird in diesem Zuge auch die Geschichte und Meilensteine von *Android* behandelt. Daraufhin wird auf die Ziele und die damit verbundenen Herausforderungen für das Betriebssystem eingegangen. Die grundlegenden Definitionen wie beispielsweise die Architektur folgen im nächsten Kapitel. Darauf aufbauend werden detaillierter die Möglichkeiten des Betriebssystems dargestellt, auch im Vergleich zu anderen mobilen Betriebssystemen. Im Folgenden wird auf Anwendungen unter *Android* eingegangen. Als eine Reflektion kann das vorletzte Kapitel angesehen werden. Nachdem der Stand des Betriebssystems dargelegt wurde, soll ein Ausblick sowohl auf technische Möglichkeiten als auch auf die zukünftige Marktentwicklung gegeben werden.

Als Motivation für die Wahl dieses Themas dient die Aktualität sowie der Entwicklung von *Android*. Denn seit der raschen Verbreitung von Smartphones und mobiler Internet Flatrates gibt es einen riesigen Markt in dem Bereich *Mobile Computing*. Zusätzlich bieten Smartphones durch die ra-

sante Chipentwicklung ein so großes Leistungspotential, dass diese sogar Desktop PCs oder Laptops gefährlich werden können. Durch diese Kombination von moderner Soft- und Hardware bieten sich Möglichkeiten, die vor einigen Jahren undenkbar waren.

2 Grundlagen und Geschichte

Bei *Android* handelt es sich um ein Betriebssystem für mobile Geräte wie z.B. Smartphones oder Tablets. Es wurde im Jahr 2007 offiziell von der für Android gegründeten *Open Handset Alliance* vorgestellt, nachdem Google den Konzern Android Inc. 2005 aufkaufte [Wik01]. Dieser entwickelte bereits seit 2003 das Betriebssystem *Android*.

Die *Open Handset Alliance* besteht aus vielen namhaften Herstellern und Firmen wie Google, Intel, HTC oder T-Mobile. Ein Jahr später stießen weitere Hersteller hinzu. Der Gründungsgrund sowie das Ziel dieser „Allianz“ war es, eine Plattform für offene Standards im Bereich der mobilen Geräte zu schaffen.

Wie auch Apple mit *iOS* versuchte die OHA mit Android die neuen Hardwaremöglichkeiten von mobilen Geräten durch Software auszunutzen. Smartphones boten neue Sichtweisen und Funktionen durch Touchscreens oder GPS- und Lagesensoren. Einige Möglichkeiten, die durch Android geboten werden sollten, waren:

- Programme, die nachträglich aus dem Internet geladen werden konnten (Apps)
- Integration von Sozialen Netzwerken
- Integration von Google-Diensten
- Deutlich bessere Hardware (Display, Prozessor, ...) ausnutzen

Die erste Android Version, 1.0, erschien am 23.09.2008. Zu diesem Zeitpunkt gab es nur ein kompatibles Smartphone, das *HTC Dream*, welches auch oft als *Google Handy* betitelt wurde. Da das Entwicklerkit (SDK) bereits vor der Auslieferung dieses veröffentlicht wurde (wie auch in Zukunft bei jeder neuen Android Version), nahm die Entwicklung rasant zu:

Kurz nach dem Erscheinen war der Marktanteil von Android bei ca 2,8 Prozent. Durch die Möglichkeit der konsequenten Weiterentwicklung des Betriebssystems gab es mehr Hardware, welche kompatibel war sowie mehr Hardwarehersteller, die Android ab Werk auslieferten. Somit erreichte Android über mehrere Meilensteine wie den Versionen 1.5 oder 2.3 schließlich die heute aktuelle Version 4.x.

Heute (Stand 16. November 2011)[Wik01] sind über 200 Millionen Android Geräte im Umlauf und es werden täglich 700.000 aktiviert. Dabei gilt es aber zu erwähnen, dass dies nicht unbedingt Smartphones sind. Durch die Modularität von Android findet dieses sich heute in Tablets oder sogar Küchengeräten und Uhren. Mit diesen Zahlen besitzt Android über 50 Prozent des gesamten Marktanteils von mobilen Geräten, neben Konkurrenten wie Apple oder Microsoft.

Ein Grund für dieses rasante Wachstum besteht in der Unterstützung der Mitglieder der *Open Handset Alliance*, welche zum großen Teil namhafte Hersteller mobiler Geräte sind. Während Microsoft auf feste Partner und Verträge angewiesen ist, kann Android durch seine Offenheit auf praktisch allen mobilen Geräten eingesetzt werden. Somit können auch kleine Firmen bei neuen Produktideen Android einsetzen, wie z.B. einem Haushaltsgerät und müssen keine teuren und bindenden Lizenzen eingehen.

Somit lag der Erfolg Androids zum Großteil an der möglichst großen Verbreitung durch die Heterogenität der Geräte, auf denen dieses zum Einsatz kommt. Diese Möglichkeit wurde durch die definierten Grundziele von Android geschaffen, die im folgenden Abschnitt näher erläutert werden sollen.

3 Grundziele und Herausforderungen

Im Bereich des Mobile Computing treten Herausforderungen auf, die es in den klassischen Desktop-PCs nicht gibt. Dazu gehören begrenzte Hardwareressourcen wie z.B. Prozessor oder Arbeitsspeicher sowie eine lange Akkulaufzeit des Gerätes trotz ausreichender Leistung. Zusammenfassend ergeben sich folgende Grundziele bei der Entwicklung des mobilen Betriebssystems Android:

- Mobilität

- Geschwindigkeit
- Offenheit
- Heterogenität
- Sicherheit

die im Folgenden ausführlicher erläutert werden.

3.1 Mobilität

Im Gegensatz zu Desktops ist einer der wichtigsten Punkte bei der Entwicklung einer mobilen Plattform die Mobilität. Dazu gehört zum Einen die Akkulaufzeit, zum Anderen die Portabilität.

Die Akkulaufzeit wird durch den Stromverbrauch des Gerätes bestimmt, welcher wiederum von Prozessor, Display oder Grafikeinheit abhängt. Da bei Smartphones oder Tablets davon auszugehen ist, dass diese soviel wie möglich ohne Strom auskommen müssen, stellt dies ein sehr wichtiges, wenn nicht sogar das wichtigste Ziel dar.

Unter der Portabilität versteht man die Möglichkeit, ein Gerät unterwegs dabei haben zu können. In dem Fall von Android versucht man dieses Ziel mit Hilfe von Heterogenität zu erreichen. Jeder definiert etwas Anderes als „portabel genug“, so dass von Android verschiedene Displaygrößen unterstützt werden.

3.2 Geschwindigkeit

Auch wenn dieses Ziel gegensätzlich zur *Mobilität* erscheint, und auch zum Teil ist, spielt die Geschwindigkeit eine große Rolle für den Bereich *Mobile Computing*. Gerade seit der Verbreitung von Internet Flatrates verlangt der Markt nach leistungsfähigen Geräten, mit denen die neuen Möglichkeiten wie z.B. Webbrowsing oder Online-Gaming genutzt werden können. Daher sollten Anwendungen schnell und ressourcenarm ausgeführt werden können.

3.3 Offenheit

Eines der Ziele, bei denen sich Android am meisten von den anderen mobilen Betriebssystemen unterscheidet, ist die Offenheit. Es wurde bei der Entwicklung von Android darauf geachtet, dass möglichst große Teile der Software-Architektur quelloffen ist, d.h. dass diese von jedem einsehbar und veränderbar ist. Dadurch bieten sich Anpassungsmöglichkeiten für jede Einsatzmöglichkeit von Android und es treten keine Probleme durch Lizenzgebühren oder Ähnliches auf.

3.4 Heterogenität

Auch dieses Ziel versucht Android zu erreichen. Durch die Heterogenität kann dieses auf vielen verschiedenen Arten von mobilen Geräten laufen. Diese können sich in der Hardware unterscheiden, wie z.B. der Displaygröße, einer physischen Tastatur oder aber dem Prozessor. In Zusammenarbeit mit der Offenheit kann Android somit auf den verschiedensten Geräten laufen und kann für das jeweilige Gerät angepasst werden.

3.5 Sicherheit

Damit das Betriebssystem auf möglichst vielen Geräten verbreitet werden kann, spielt diese eine zentrale Rolle. Falls man wichtige Geräte (wie z.B. Notabschaltungen) mit Android ausstatten möchte, ist dies der wichtigste Punkt beim Einsatz. Bei der Sicherheit geht es darum, dass kein Schadcode ausgeführt werden kann, der Daten beschädigt oder entwendet.

4 Architektur

Um die im vorigen Abschnitt beschriebenen Ziele zu erreichen, wurde die Architektur mit diesen im Hinterkopf entwickelt. Diese gliedert sich wie in der folgenden Abbildung zu sehen:

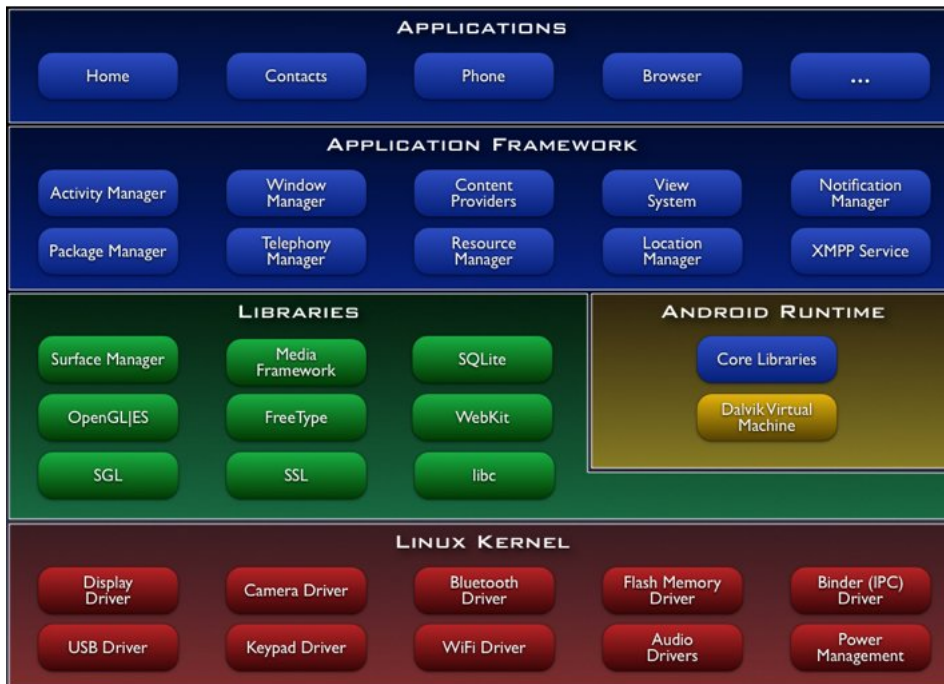


Abbildung 1: Die Android Architektur[Wik01]

Die Architektur besteht aus vier Schichten:

Die unterste Schicht, der *Linux Kernel*, bietet den direkten Hardware-Zugriff durch die Treiber. Da dieser sehr heterogen ist und eine gute *ARM* Unterstützung bietet, sowie komplett quelloffen ist, wurde sich für diesen entschieden. Die *ARM* Architektur ist dafür bekannt, trotz ausreichender Leistung das Augenmerk auf geringen Stromverbrauch zu legen, weshalb sie oft in mobilen Geräten vorkommt. Jedoch bietet der Linux Kernel auch Unterstützung für andere Architekturen, daher ist bei der Festlegung auf diesen keinerlei Bindung an eine bestimmte Hardware gegeben.

Die darüberliegende Schicht bietet die Bibliotheken an, welche unter Android benutzt werden. Erneut wurde viel Wert auf die Offenheit und Heterogenität gelegt. Die Bibliotheken stammen zum großen Teil aus der Open Source Community und entsprechen daher den gewünschten Anforderungen. Dazu zählen z.B. die *WebKit* Bibliothek für die Darstellung von Internet Seiten oder die *SSL* Bibliothek für die verschlüsselte Übertragung von

Daten.

In dieser Schicht findet sich auch die *Dalvik Virtual Machine*, welche für die Ausführung von Anwendungen zuständig ist. Jede Anwendung erhält von der virtuellen Maschine einen eigenen Interpreter und eine einmalige ID zur Identifizierung, was die Sicherheit und Zuverlässigkeit gewährleisten soll. Sie verwendet als Eingabe Dalvik-Bytecode, welcher aus Java-Bytecode kompiliert wird:

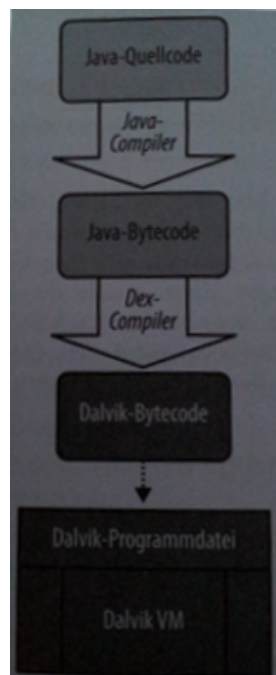


Abbildung 2: Dalvik Funktionsweise[GS11]

Die Verwendung einer eigens entwickelten virtuellen Maschine anstatt der üblichen *Java Virtual Machine* hat zum Einen den Grund der Anpassung für Android, zum Anderen lizenzbedingte Gründe¹.

Die genannten Bibliotheken und damit auch die Dalvik Virtual Machine werden durch die nächst höhere Schicht aufgerufen. In dieser findet sich das *Application Framework* oder auch oft als *API* bezeichnet. Diese ist sowohl dafür zuständig, die Verbindung zwischen den Bibliotheken und den

¹Dazu im nächsten Kapitel mehr

Anwendungen herzustellen, als auch die organisatorischen Dinge zu verwalten. Dazu gehört zum Beispiel die Ressourcenverwaltung oder die Darstellung auf dem Bildschirm.

In der obersten Schicht finden sich die Anwendungen, welche vom Benutzer ausgeführt werden. Diese greifen, wie zuvor erwähnt, über die verschiedenen Schichten auf die Hardware zu. Dabei gilt zu beachten, dass durch die Hierarchie der Schichten die Grundziele erreicht werden. Zum Beispiel werden beim Ausführen einer Anwendung die jeweiligen Sicherheitsmechanismen z.B. im Ressourcenmanager oder in der virtuellen Maschine Dalvik ausgeführt. In dieser Schicht programmieren die Anwendungsentwickler sogenannten *Apps*², und benutzen dabei das darunterliegende *API*.

Wie in [GS11] ausführlicher beschrieben, ist der gesamte Software-Teil, von der untersten bis zur obersten Schicht, in Android vollständig offen. Wie zuvor erwähnt, besteht die unterste Schicht aus einem Linux Kernel und die darüber liegende aus offenen Bibliotheken. Einige kleine Änderungen gibt es jedoch unter Android im Gegensatz zu Linux:

- GNU libc, die C-Bibliothek für Linux, wurde unter dem Namen *Bionic* neu geschrieben
- Kleinere Codeteile, wie z.B. Software für Hardware-Schnittstellen (Bluetooth, usw) sind nicht quelloffen(Herstellerspezifisch)

Diese Punkte hängen mit den Lizenzen des Codes zusammen. Google bzw die Open Handset Alliance versuchte bei der Android Entwicklung soviel wie möglich aus der Open Source Community zu übernehmen. Die *GNU libc* steht unter der *GPL* Lizenz, welche besagt, dass jegliche Änderungen des Codes an die Open Source Community zurückgegeben werden müssen. Da aber ein Hauptziel die weite Verbreitung auf möglichst vielen Geräten war, benötigte man als Partner Hardware- und Softwarefirmen. Diese bringen oft nur Unterstützung mit, wenn sie eigene Kreationen wie z.B. Anwendungen oder eine Oberfläche nicht mit Mitbewerbern teilen müssen, sondern sich dadurch von anderen abgrenzen und Gewinne erwirtschaften können. Die unter Android verwendete Lizenz ist die *Apache Software License 2* oder kurz *ASL 2*, welche alle Anforderungen erfüllt, da man den

²Dazu später mehr

Quelltext verändern und für seine Zwecke anpassen darf, aber diese Änderungen nicht an die Open Source Community zurückgeben muss.

5 Möglichkeiten und Vergleich

Durch die in den vorigen Abschnitten erwähnten Punkte geht *Android* einen anderen Weg als die Mitstreiter wie *iOS* oder *WindowsPhone*. Dadurch bieten sich gerade für kleine Firmen ganz neue Möglichkeiten, ein Standbein im *Mobile Computing* aufzubauen.

5.1 Hardware

Auch wenn das Hauptaugenmerk bei der Android Entwicklung für viele Firmen oder Anwender auf dem Softwarebereich liegt, sollen noch kurz die Eigenschaften von Android und Unterschiede zu der Konkurrenz beleuchtet werden.

Apple hat direkte Verträge mit Zulieferern und bietet die Hardware sowie die Software als ein Paket an, sodass man als Hardware-Entwickler keinerlei Möglichkeit hat, in den Apple Markt einzusteigen.

Bei Microsoft hat man als Hardware Hersteller nur die Möglichkeit, ein Handy mit Windows Phone auszustatten, wenn man Verträge mit Microsoft abschließt, die wiederum teuer und oft lange Laufzeiten haben. In diesem Markt können meist nur große Firmen bzw Hersteller mobiler Geräte wie z.B. Nokia Fuß fassen.

Bei Android hingegen gibt es keinerlei Einschränkung bei der Hardwarewahl. Durch die Verwendung des Linux Kernels wird eine sehr große Vielfalt an Hardware unterstützt und durch die Offenheit der Software kann hier jedes kleine Unternehmen mit wenig Startkapital seine Ideen umsetzen.

Ein gutes Beispiel für die vielen Möglichkeiten, die Android bietet, zeigt der Hersteller *Blue Sky* mit seiner Armbanduhr, auf welcher das Betriebssystem läuft:



Abbildung 3: I am Watch, Uhr mit Android[Wik05]

Mit dieser Armbanduhr hat man die Möglichkeit Musik zu hören oder aber neue Anwendungen herunterzuladen. Gedacht ist dieses mobile Gerät als Erweiterung eines Handys oder Smartphones, an welches es per Bluetooth gekoppelt ist und man somit telefonieren kann oder aber SMS schreiben.

5.2 Software / Anwendungen

Dieser Bereich ist für den Markt und viele Entwickler durch die Zugänglichkeit von Software bzw Programmierumgebungen sehr wichtig, da mit wenig Geldaufwand (meist deutlich weniger als für Hardware) bereits ein Produkt entworfen und entwickelt werden kann.

Bei der Software beginnen die Unterschiede zwischen den mobilen Plattformen bereits bei der Wahl der Programmiersprache sowie der Distribution von Anwendungen und den damit verbundenen Kosten, was für kleine Unternehmen einen großen Unterschied machen kann:

Möchte man eine Anwendung unter iOS programmieren, verwendet man die Programmiersprache *Objective C*, da dies von *Apple* so vorgegeben ist.

Um die entwickelten Anwendungen an die Kunden zu bringen ist eine Veröffentlichung im *App-Store* nötig. Apple hat das Monopol dieses Marktes, außerdem werden Anwendungen erst von Apple freigegeben, wenn diese einer technischen und inhaltlichen Prüfung unterzogen wurden. Programmierer von Anwendungen müssen zudem ein Geheimhaltungsabkommen und einen Mitgliedsbeitrag bezahlen, zusätzlich zu einer prozentualen Abgabe der Gewinne, die durch Verkäufe im App-Store entstanden sind[Wik04].

Eine Entwicklung unter WindowsPhone findet in *C#* oder in *VisualBasic* statt. In diesem Fall benötigt man keine Lizenz oder einen Mitgliedsbeitrag, muss aber auch hier im vorgegebenen Store seine Programme veröffentlichen und nach einer Prüfung freigeben lassen.

Android setzt im Gegensatz dazu auf folgende Punkte:

1. Eine Teilmenge von Java als Programmiersprache
2. Keine Lizenzgebühren, SDK frei verfügbar, keine Gebühren der Mitgliedschaft
3. Leicht integrierbar in die freie Entwicklungsumgebung Eclipse, damit unabhängig vom Betriebssystem
4. Software kann ohne vorhandene Android Hardware entwickelt werden
5. Kein vorgegebener App Store
6. Google's App Store basiert auf Nutzerbewertungen und führt keinerlei vorherige Zensurierung oder Prüfung von Anwendungen durch
7. Software auch direkt von Websites ohne Store installier- und verwendbar
8. Von Drittanbietern entwickelte Software ist der vorinstallierten (nativen) ebenbürtig, in Möglichkeiten sowie Rechten
9. Keinerlei Bindung an einen bestimmten Mobilfunkanbieter

Zu 1) soll noch erwähnt werden, dass für Android der größte Teil von Java verwendet werden kann, entfernt wurde z.B. die Benutzerbibliothek *Swing*

und durch Android-eigene Bibliotheken ersetzt. Diese wurden im Hinblick auf *Mobile Computing* entwickelt. In Android wird keine *JVM* zur Ausführung der Programme verwendet, sondern die *DVM*. Erneut war einer der Gründe dieser Entscheidung die fehlende offene Lizenz der virtuellen Maschine sowie die Anpassung an die mobilen Gegebenheiten.

Zu 5) und 6) gilt zu sagen, dass dies für kleinere Firmen und Entwicklerteams eine ganz andere Möglichkeit bietet, in den Markt einzusteigen. Sie können ohne Lizenzgebühren oder Angst vor Zensur eine Anwendung entwickeln und veröffentlichen. Auf der anderen Seite können über diesen Weg auch schlecht programmierte oder schädliche Anwendungen ihren Weg auf das Gerät unbedarfter Nutzer finden, die sich der Gefahren nicht bewusst sind. Um jedoch einen gewissen Schutz zu bieten, unabhängig von Zensur oder einem bestimmten Store, bietet Android folgenden Kompromiss:

- Jede Anwendung läuft in ihrer eigenen virtuellen Maschine
- Jeder Zugriff auf private Daten oder Netzwerkzugriff ist standardmäßig deaktiviert
- Falls eine Anwendung solche Rechte benötigt, wird dies bei der Installation angegeben und der Benutzer muss dem Zugriff zustimmen, ansonsten wird die Anwendung nicht installiert.
- Jegliche Zugriffsrechte sind statisch deklariert, sodass diese nicht nachträglich geändert werden können.

Punkt 8) ist ein großer Unterschied zu den anderen mobilen Betriebssystemen. Unter Android sind alle Anwendungen „gleich viel wert“. Das bedeutet, dass ein Anwendungsentwickler sogar eine eigene Anwendung zum Telefonieren oder für das Adressbuch programmieren kann, welche die originale ersetzt. Bei der Konkurrenz ist dies nicht möglich, da nachträglich installierte Anwendungen den originalen untergeordnet sind. Sie haben weniger Rechte und Möglichkeiten als die nativen Anwendungen.

Als einige Beispiele solcher Anwendungen können Email Programme wie *Gmail*, welches die Daten jederzeit mit einem GoogleAccount synchronisiert, Navigationsprogramme oder aber Programme zur Erstellung oder Bearbeitung von Dokumenten genannt werden. Dies ist nur ein kleiner Aus-

schnitt aus den Möglichkeiten, die mit heutigen Android Geräten zur Verfügung stehen.

6 Reflektion und Ausblick

Durch diese Seminararbeit sollte klar geworden sein, dass durch die heutigen technischen Möglichkeiten sowie der Massenproduktion von leistungsfähigen Geräten der Bereich *Mobile Computing* aktueller denn je ist. *Android* bietet durch seine Offenheit sehr viele Möglichkeiten für heutige und zukünftige Entwicklungen, sei es auf der Softwareseite oder der Hardwareseite. Bei Betrachtung der Verbreitung und Akzeptanz von Android kann man ein weiterhin starkes Wachstum erwarten.

Viele der heute üblichen Anwendungen oder Verwendungszwecke eines Desktop-PCs lassen sich, wie zuvor gesehen, bereits durch mobile Geräte mit Android ersetzen. Bereits jetzt verzeichnen Desktop-PCs einen starken prozentualen Verkaufsrückgang, was sich in der Zukunft weiter fortsetzen dürfte.

Das liegt zum Einen an der Weiterentwicklung von Android selbst, wozu z.B. mehr Schnittstellen wie *NFC* gehören, oder aber auch an mehr Einsatzmöglichkeiten und Integration von mobilen Geräten. Vorstellbar wäre zum Beispiel der Vorstoß in den Bereich *Spiele*, den man durch eine komplette Integration möglich machen könnte. Bei der rasanten Entwicklung liegt es durchaus im Rahmen des Möglichen, dass ein Smartphone mit genug Leistung ausgestattet ist, dass es alle Geräte für digitale Medien oder Interaktionen in sich vereint. Ob durch Kabel oder Wireless verbunden kann der Nutzer dann an seiner Tastatur über das Smartphone seine Tätigkeiten wie Briefe schreiben oder Ähnliches erledigen. Zusätzlich können über den Fernseher im Wohnzimmer Musik, Filme oder Fotos auf dem Android Gerät abgerufen werden. Auch wenn das zurzeit bereits teilweise möglich ist, ist noch ein Zusatz denkbar: Das Smartphone als Spielekonsole. Gerade bei dem heutigen Umschwung auf digitale Medien und Distribution, könnte eine Verbindung von stationärer und mobiler Unterhaltung in einem Gerät ein wichtiger Schritt des Mobile Computing sein. Somit hat der Nutzer alle Möglichkeiten zu spielen, ob unterwegs auf dem Smartphone selbst oder aber im Wohnzimmer auf einem großen Bildschirm mit Surround Sound.

Eine solche Synchronisation könnte auch im Geschäftsbereich Sinn machen:

Jeder Mitarbeiter besitzt ein Android Handy und benutzt dieses als Arbeitsrechner, im Büro und zuhause. Dadurch gehen keine Daten verloren, alle Daten und erledigte Arbeit sind immer „synchronisiert“. Auch wenn Laptops mit Dockingstationen dieselbe Funktionalität bieten, kann ein Smartphone den Vorteil der Verfügbarkeit bieten. Denn ein solches passt in jede Tasche und kann im Notfall auch in engen Räumen und im Gehen bedient werden, was ein nicht zu unterschätzender Vorteil bei der heutigen Informationsflut ist. Natürlich sind diese Möglichkeiten nicht auf ein Smartphone beschränkt, sondern erstrecken sich möglicherweise auf Tablets oder einer anderen Art der mobilen Geräte.

Also bleibt als abschließende Prognose zu sagen:

Die Mobile Computing Plattformen werden vermutlich die heutigen PCs für alle die Leute ersetzen, die nicht mehr als vom Smartphone oder Tablet zur Verfügung gestellten Ressourcen benötigen. Alle aktuellen Aufgaben wie z.B. Emails oder Dokumente können dann von überall aus ausgeführt werden und alle Daten wie z.B. Fotos sind auf einem Gerät, jedoch auf vielen Plattformen verfügbar und jederzeit abrufbar.

Die *Open Handset Alliance* hat sich als Ziel in der Zukunft gesetzt, *Apple* zu überholen und Android auf jedes mobile Gerät, von jedem Hersteller in jedem Netz zu bringen. Den bisherigen Zahlen und Entwicklungen zufolge ist dies nicht unrealistisch.

Literatur

[GS11] Gargenta, Mark; Schulten, Lars: Android Entwicklung, O'REILLY, Köln 2011

[Wik01] Wikipedia(EN): Android, Stand 20.12.2011, [http://en.wikipedia.org/wiki/Android_\(operating_system\)](http://en.wikipedia.org/wiki/Android_(operating_system))

[Wik02] Wikipedia(DE): Android, Stand 27.12.2011, [http://de.wikipedia.org/wiki/Android_\(Betriebssystem\)](http://de.wikipedia.org/wiki/Android_(Betriebssystem))

[Wik03] Wikipedia(DE): OSI-Modell, Stand 28.12.2011, <http://de.wikipedia.org/wiki/OSI-Modell>

[Wik04] Wikipedia(DE): Apple iOS, Stand 07.01.2012, http://de.wikipedia.org/wiki/Apple_iOS

[Wik05] Wikipedia(EN): I am Watch, Stand 09.01.2012, http://en.wikipedia.org/wiki/I%27m_Watch