



Das AndroidPITiden-Buch

Kleiner Androiden-Führer
für
Einsteiger und Fortgeschrittene
von
Andreas Itzchak Rehberg

[Version 25](#)

Dieses eBook unterliegt einer [Creative Commons Lizenz](#). Du darfst es also auf jeden Fall in unveränderter Form weitergeben – und bei Quellen-Nennung zitieren. Weitere Details findest Du im verlinkten Lizenz-Text, wenn Du auf das folgende Bild klickst:



Die jeweils aktuellste Version dieses eBooks findest Du [hier](#).

Cover-Design: AndroidPIT

Unter dem Namen **Das inoffizielle Android-Handbuch** (ISBN: [978-3-645-60141-2](#)) gibt es eine Print-Ausgabe dieses Buches vom Franzis-Verlag.

ANDREAS ITZCHAK REHBERG



Das AndroidPITiden-Buch



AndroidPIT
Fill up your mobile



INHALT

0. [Vorwort](#)
1. [Grundlagen für den Einsteiger](#)
 1. [Anwendungen verwalten](#) (Installieren, Aktualisieren, Bereinigen)
 2. [Apps organisieren](#)
 3. [Google Market Ergänzungen und Alternativen](#)
 4. [Datensicherung](#)
 5. [Konfiguration](#)
 6. [Zurücksetzen](#)
2. [Mit Android arbeiten](#)
 1. [Schaltzentrale: Home-Screen, Widgets & "Home Replacements"](#)
 2. [Steuerzentrale: Einstellungen und "Switches"](#)
 3. [Von Taskkillern und anderen bösen Buben](#)
 4. [Das Android-Gerät vom PC aus verwalten](#)
 5. [Datenaustausch mit dem PC](#)
3. [Sicherheit](#)
 1. [Was brauche ich wirklich?](#)
 2. [GMV](#)
 3. [Rundum-Sorglos-Pakete](#)
 4. [Anti-Virus und Anti-Malware](#)
 5. [Bei Diebstahl und Verlust](#)
 6. [Worauf Apps Zugriff haben](#)
4. [Apps machen das Phone smart](#)
 1. [Telefonieren](#)
 2. [Die Kosten im Blick und unter Kontrolle](#)
 3. [Nachrichten verschicken und empfangen](#)
 4. [Lektüre](#) (eBooks, News, Nachschlagen)
 5. [Schule & Studium](#)
 6. [Unterwegs](#) (Fahrpläne, Navigation, Shopping)
 7. [Gesundheit](#)
 8. [Büro, Office & Verwaltung](#)
 9. [Sensoren](#)
 10. [Augmented Reality](#)
 11. [Fernbedienen und Überwachen](#)
 12. [MultiMedia: Alles, was Krach macht](#)
 13. [Tools](#)
 14. [Automatisieren von Aufgaben](#)
5. [Tiefgehendes für Fortgeschrittene](#)
 1. [Der Super-User "root"](#)
 2. [Apps am automatischen Starten hindern](#)
 3. [Vorinstallierte Apps entfernen](#)
 4. [Tuning – Das Android-System auf Trab bringen](#)
 5. [Durststrecke – mehr aus dem Akku herausholen](#)
 6. [ROMs: Stock, Vendor, und Custom](#)
 7. [Ortsdaten-Cache einsehen und verwalten](#)
 8. [Zugriffe sperren: Firewalls & Permission-Blocker](#)
6. [Anhang](#)
 1. [Begriffserklärungen](#)
 2. [Häufig gestellte Fragen – und ihre Antworten](#)

3. [Google Permissions](#) – und was sie bedeuten
4. [APN-Einstellungen](#) ausgewählter Netzbetreiber
5. [Secret Codes](#) oder Magische Nummern
6. [Leistungsaufnahme verschiedener Komponenten](#)

VORWORT



Was ist das denn für ein komischer Name: "AndroidPITide"? Gar nichtLUSTIG nicht komisch. Ich bin ein AndroidPITide: Mitglied der Community bei [AndroidPIT](#). Du bist natürlich jederzeit willkommen, auch einer zu werden (so Du es nicht bereits bist). Alles klar?

Als Grundlage für die Inhalte stehen meine [App Reviews nach Einsatzzweck](#) bei [AndroidPIT](#), die ich hier sinnvoll zusammenzufassen versucht habe. Es handelt sich also bei diesem eBook eher um eine Übersicht, die den Einstieg erleichtern soll – weit mehr Details findest Du unter dem angegebenen Forums-Link. Dort sind die Informationen naturgemäß i.d.R. auch aktueller, vor allem was die Apps betrifft. Hier kommen ja ständig neue hinzu. Und außerdem kann ich nicht alle Benutzer-Erfahrungen hier aufnehmen. Hinzu kommt, dass Du im Forum auch [Deine Fragen stellen](#) (und auf Antworten hoffen) kannst – was mit einem eBook allein etwas schwierig sein dürfte.

Gedacht ist das Ganze also so, dass Du Dir mit diesem eBook einen Überblick verschaffst – und für die tiefer schürfenden Dinge auf das Forum zurück greifst. Die meisten Links hier führen Dich ohnehin zu AndroidPIT – und dort bist Du in guten Händen!

Und noch eins muss ich loswerden: Viele der hier kurz vorgestellten (oder auch nur genannten) Apps habe ich selbst nie getestet – etwa, weil ich nicht die Voraussetzungen dazu habe (ich nutze kein Facebook, und mein Phone auch nicht zum Spielen, um nur zwei Dinge zu nennen). Ein Grund mehr, auf die Erfahrungen der Community zurückzugreifen: Bei [AndroidPIT](#) bist Du jederzeit herzlich willkommen! Auch für Dein Feedback sowie Deine Fragen zu diesem kleinen eBook findet sich dort ein Plätzchen.

Zu guter Letzt noch ein kleiner technischer Hinweis: Sofern Dein eBook-Reader die StyleSheets korrekt unterstützt, erkennst Du zwei verschiedene Arten von Links an ihrer Textfarbe: Rote gehen "nach draußen" (öffnen also wahrscheinlich Deinen Web-Browser), während blaue "drinnen bleiben" (also nur Querverweise innerhalb dieses eBooks sind).

Doch nun: Viel Spaß bei der Lektüre!

GRUNDLAGEN FÜR DEN EINSTEIGER

Im ersten Teil dieses kleinen Handbuches geht es um die Grundlagen. Fortgeschrittenere Anwender können diesen also getrost überspringen – und gleich zum zweiten oder gar dritten Teil schreiten...

Die erste Inbetriebnahme und Grund-Einrichtung des Androiden erlaube ich mir an dieser Stelle zu überspringen: Zum Einen unterscheiden die sich je nach Hersteller ein wenig, zum Anderen liegt dem Gerät zumindest dafür in der Regel eine Kurzanleitung bei. Wer dennoch Starthilfe benötigt, findet sie z. B. in einem [Workshop bei Chip.DE](#).

Wie soll Steve Jobs am Ende seiner Vorstellung des ersten iPhone gesagt haben: "Ach ja, telefonieren kann man damit auch." Natürlich sind wir mit Android im "ganz anderen Lager" (laut Stevie in der Schmuddel-Ecke – aber wir wissen es natürlich besser). Dennoch gehe ich hier ähnlich vor, und klammere das Telefonieren zunächst aus. Stattdessen steige ich mit den Anwendungen ganz allgemein ein: Wie bekomme ich die auf meinen Androiden – und ggf. auch wieder runter? Wie organisiere ich sie, sodass ich mich auch nach der 50sten installierten App noch darin zurechtfinde? Und wie erstelle ich Sicherungen meiner Daten, falls es denn doch einmal "knallt"?

Anwendungen verwalten

Um folgende Themen geht es in diesem ersten Kapitel:

- Installieren von Apps
- Aktualisieren (neudeutsch: "Updaten") von Apps
- Apps Bereinigen (Cache löschen etc.)
- De-Installieren (Löschen) von Apps

Also im Prinzip der ganz normale "Lebens-Zyklus" einer App auf dem Androiden: Erst wird sie installiert (und benutzt), hin und wieder gibt es neuere Versionen, und schließlich – ist man ihrer überdrüssig geworden, oder hat etwas besseres gefunden – soll sie wieder vom Phone verschwinden. Zwischendurch gilt es u. U. den Cache zu bereinigen: Entweder, um Platz freizuschaukeln, oder um diverse kleine Probleme zu lösen. Für all diese Aufgaben gibt es verschiedene Ansätze.

Apps? APK-Datei?

Zu allererst gilt es noch ein paar Begriffe zu klären...

App nennt man die Anwendungen unter Android (und übrigens auch bei Apples iOS). Dieses Kürzel leitet sich nicht etwa von *Apple* ab, wie einige meinen, sondern vom Wort *Application* – jaja, der "Aküfi" (Abkürzungs-Fimmel) ist nicht ausschließlich © Germany.

Eine [APK-Datei](#) enthält in der Regel eine solche App, und ist so etwas wie ein "Installations-Archiv". Erhältlich im [Android Market](#), und oftmals auch auf den Webseiten der jeweiligen Entwickler – diese beiden Quellen können als relativ sicher gelten (sowohl was "Systemsicherheit" als auch was "Legalität" anbetrifft). Sie sind auch in der "freien Wildbahn" anzutreffen (etwa als Suchergebnis einer Google-Suche nach "<AppName>.apk"). Im letzteren Falle ist allerdings Vorsicht geboten: Zum einen sind derartige Angebote nicht selten illegal, zum anderen die hier auffindbaren APK-Dateien oftmals auch manipuliert (Stichwort: [DroidDream](#)).

Ergo: Das Futter für den Androiden sollte besser ausschließlich aus verlässlichen Quellen besorgt werden. Und die schauen wir uns jetzt an.

Bordmittel

Natürlich bringt Android passende Hausmittel für die genannten Aufgaben mit – irgendwie müssen ja die ersten "Früchtchen" in den Korb gelangen können.

Android Market

Über die *Market-App* (siehe Screenshot rechts) soll der Android-Jünger sich seine Apps besorgen. Die Fülle an Apps kann hier grob nach Rubriken durchblättert oder, so der Name der gesuchten App bekannt ist, auch gezielt durchsucht werden. Letzteres ist natürlich auch nach Stichworten möglich, die in Namen oder der Beschreibung der Apps vorkommen. Aufgrund der großen Anzahl an im Market verfügbaren Apps ist das Ergebnis aber nicht unbedingt immer befriedigend. Filtermöglichkeiten (etwa das Ausblenden unerwünschter Entwickler oder das Ausschließen bestimmter Begriffe) gibt es in der App leider nicht.



Etwas komfortabler wird das Ganze, wenn man [die Website des Markets](#) mit dem Browser am PC benutzt: Hier lassen sich viele der aus der "erweiterten Google-Suche" bekannten Tricks verwenden – etwa um mit dem Begriff vorangestellten "-" Zeichen Begriffe auszuschließen. So findet man Apps z. B. zum Thema Scuba-Diving (Sporttauchen) durch eine Suche nach "+scuba -log" (oder "+dive -log" – jeweils ohne die Anführungszeichen) Apps zum Thema Scuba-Diving, schließt jedoch Logbücher aus. Die Informationen lassen sich hier am größeren Bildschirm auch weit bequemer sichten. Ist die gesuchte App gefunden, kann sie überdies, sofern man mit seinem Google-Account angemeldet ist, mit einem einfachen Klick auf den Button "Installieren" auf den Androiden befördert werden: Schon wenige Sekunden später sieht man dort in der Regel den Download und schließlich auch den Installationsprozess starten. Sind mehrere Geräte mit dem selben Google-Account verknüpft, lässt sich das gewünschte Zielgerät natürlich auswählen. Auch filtert der Market automatisch die Apps aus, die mit dem Zielgerät nicht kompatibel sind (siehe Anhang: "[Warum finde ich die App im Market nicht?](#)").

Um nun aber die Nutzung des Markets auf dem Androiden zu verbessern bieten sich, abgesehen von "alternativen Market-Apps", auch diverse "Market-Ergänzer" an.

Anwendungen verwalten



"Anwendungen verwalten" (Screenshot links – zu finden unter Einstellungen→Anwendungen im Android-Menü) ist vor allem für eines bekannt: Es ist LAAAAHM! Zumindest unter Android < 2.2: Nach dem Aufruf auf meinem Wildfire konnte ich erst einmal Kaffee holen gehen. Wenn ich zurück war, hatte sich die Liste vielleicht aufgebaut. *Erst dann* konnte ich über "Menü->Filter" einstellen, dass ich nur die von mir selbst installierten Apps sehen will. Keine Möglichkeit, das als Default einzustellen.

Spätestens mit [Froyo](#) (Android 2.2) scheint sich das geändert zu haben. Die Seite ist jetzt in Tabs eingeteilt: Heruntergeladene (also selbst installierte; dieser Tab wird bei Aufruf von "Anwendungen verwalten" geladen), Alle, Auf SD-Karte, Ausgeführte. Die ersten drei Tabs sind ruckzuck geladen, der vierte braucht ein paar Sekunden und ist ein vollwertiger Task-Manager...

Ist die Liste aufgebaut, lassen sich Details zu den Anwendungen einsehen. Dies geschieht, indem die gewünschte App in der Liste kurz angetippt wird. Sodann offenbart sich: Wieviel Platz belegt die App selbst, wie viel ihre Daten. Wie viel Cache benutzt sie. Und so weiter. Von hier aus lässt sich dann z. B. auch der **Cache leeren** – oder die Anwendung deinstallieren (löschen).

Alternativen zur Cache-Bereinigung findest Du im Kapitel [Tuning](#).

Market-Alternativen

Derer gibt es viele: AndroitPIT, AppBrain, PDassi... Da fällt es schon bald schwer, über alle auf dem Laufenden zu bleiben, denn es kommen ja auch ständig neue hinzu. Daher kann diese Übersicht keinesfalls vollständig sein – vielmehr beschränke ich mich auf ein paar Beispiele, die mir besonders sinnvoll erscheinen.

Eines ganz zu Anfang: Auch wenn es durchaus Sinn machen kann, mit mehreren/verschiedenen dieser Alternativen parallel zu arbeiten empfiehlt es sich, Einkäufe immer an der gleichen Stelle zu machen. Sonst verliert man recht leicht den Überblick – und weiß etwa nach einer Neuinstallation oder dem Wechsel auf ein neues Gerät nicht mehr, aus welchem Market man nun die gekaufte App wieder bekommt, ohne sie nochmals bezahlen zu müssen.

AndroidPIT AppCenter



Bevor ich irgend etwas anderes hier erwähne, möchte ich auf die "Apps" bei AndroidPIT eingehen. Wem das Stöbern im Android Market mit der App zu unbequem ist, der sollte auf der AndroidPIT-Website nämlich unbedingt mal auf [Apps](#) klicken: Hier finden sich zu vielen Apps nicht nur Bewertungen aus dem Market und von den AndroidPITidern – sondern oftmals auch Testberichte, die die ganze App durchleuchten, und so schon vor der Installation einen genaueren Einblick erlauben. Nicht selten sind es sogar mehrere Testberichte pro App, die sich dann auf unterschiedliche Versionsstände beziehen – so bekommt man auch gleich noch ein Gefühl dafür, wie sich die App entwickelt hat. Und das findet man wirklich nicht überall! Zudem sind die Kerndaten so übersichtlich aufbereitet, wie sonst nirgendwo: Größe der *.apk Datei, geforderte Permissions (System-Berechtigungen), minimale unterstützte Android-Version...

Hinzu kommen – für immer mehr Apps – Links zum direkten Download. Dies ist besonders interessant für die "Geschädigten kleiner Displays" wie z. B. HTC Wildfire, die über die Market-App oft nur ein "Item not found" zu sehen bekommen.

Wer das direkt auf seinem Androiden tun möchte, der greift zum [AppCenter](#) (siehe linkes Bild). Auch hier besteht die Möglichkeit, nach Kategorien zu browsen, Filter einzusetzen, und mehr.

Und ein weiteres Plus bietet AndroidPIT in dieser Hinsicht: Mehr Flexibilität, wenn es ums Bezahlen geht. Klar geht auch hier die Kreditkarte, PayPal gab's auch schon, bevor es im "offiziellen Markt" eingeführt wurde. Und weitere Zahlungsmöglichkeiten kommen ständig dazu. Einziger Haken: Nicht jede Kauf-App lässt sich hier erwerben (dazu müssen die Entwickler sich mit AndroidPIT entsprechend einigen). Aber auch das werden täglich mehr.

Ach ja: Und so manche App, die Du evtl. mit Deinem "zu kleinen" Gerät im "offiziellen Market" gar nicht zu sehen bekommst, kannst Du hier mit Leichtigkeit finden und installieren. Und auch die Vorschläge von alternativen Apps sind nirgends so treffsicher wie hier – weil handverlesen und kontrolliert. Ganz zu schweigen von den Testberichten, die auch in der App einsehbar sind, und dem Zugang zum Forum...

AppBrain

Auch bei [AppBrain](#) kann sich die Kombination aus Website und App durchaus sehen lassen – auch wenn sie sich in Sachen Übersichtlichkeit keinesfalls mit AndroidPIT messen kann.

Es gibt einen (rudimentären) Filter (leider nur mit den dort sichtbaren festen Kriterien, siehe rechtes Bild), man kann Apps nach Kategorien durchstöbern, und die Ergebnisliste sortieren. Ergänzt wird dies von einer Liste mit Empfehlungen, die anhand der bereits installierten Apps ermittelt werden. Hier kann man "ungewünschte Artikel" auch entfernen, und bekommt dann wieder neue Vorschläge.

Gut gelöst ist auch das Update: Egal, aus welcher Quelle eine App installiert wurde – sofern sie im Google Market enthalten ist, wird sie auch von *AppBrain* gefunden. Nach der Synchronisation der Liste von auf dem Androiden installierten Apps mit der im eigenen AppBrain-Konto (der Login dort erfolgt mit dem eigenen Google-Konto) einmal auf den Button "Perform Installs" gedrückt, und ab die Luzie! wird alles in einem Rutsch gemacht. Naja, fast – eine kleine Mogelpackung ist es naturgemäß, schließlich muss der Telefon-Besitzer ja noch die "Permissions" abnicken. Und das erfolgt dann lustigerweise wieder in der originalen Google Market-App...

Ein weiteres Plus dieser App: Einzelne Apps lassen sich vom Update ausschließen. Hier erfolgt dann auch keine Benachrichtigung über verfügbare Updates mehr, und auch beim gerade beschriebenen Sammel-Update werden diese nicht angefasst. Auch können einzelne Updates einer App übersprungen werden – dann erfolgt eine neue Benachrichtigung für diese erst wieder beim nächsten Update. Beides Dinge, die man in Googles *Android Market* vergeblich sucht.

Und auch mit dieser App bekommt man die gewünschten Dinge auf den Androiden, die man mit der Market-App vielleicht nicht findet. Was natürlich nicht heißt, dass diese Apps dann auch funktionieren – denn eigentlich gibt es für das "nicht Finden" ja einen guten Grund, zumindest in der Theorie: Der Entwickler hat Kriterien (z. B. eine mindest-Bildschirm-Auflösung) angegeben, die das Gerät nicht erfüllt. Oder vergessen, eine Einschränkung aufzuheben...



PDassi

Der Vollständigkeit halber sei auch die App von [PDassi](#) an dieser Stelle kurz erwähnt. Diese bietet u.a. zusätzliche Bezahlmethoden wie z. B. Paypal, Bankeinzug oder Überweisung – für all jene, die entweder keine Kreditkarte haben oder diese nicht mit ihrem Google-Account verknüpfen möchten. Wobei an dieser Stelle auch darauf hingewiesen sei, dass man auch bei AndroidPIT.de (s.o.) alternative Zahlungsmöglichkeiten findet!

Alternative Verwaltung

So – jetzt haben wir alles mögliche installiert und in Gang gebracht. An dieser Stelle fragte schon Goethes Zauberlehrling:

*Ach, da kommt der Meister!
Herr, die Not ist groß!
Die ich rief, die Geister,
Werd ich nun nicht los.*

Wie bzw. wo also nun de-installieren? Die "Hausadresse" findet sich, wie bereits unter "Bordmittel" festgestellt, unter *Einstellungen* → *Anwendungen* → *Anwendungen verwalten* – und ist (bis Android 2.1) schnarchlangsam. Warum das so ist? Nun, bei jedem Aufruf werden hier alle Apps gescannt – auch die System-Apps, an denen "Otto Normaluser" eh nicht schrauben kann. Gebraucht wird das sicherlich hin und wieder (um z. B. den Market-Cache zu löschen) – aber in den meisten Fällen eher nicht. Deshalb gibt es auch die Möglichkeit, an jener Stelle über *Menü* → *Filter* auszuwählen, dass man nur die selbstinstallierten Apps sehen möchte. Das geht aber erst, nachdem alle Apps geladen sind – und die dumme Nuss merkt sich das nicht bis zum nächsten Mal...

Höchste Zeit also für die Ghost Busters, Krümel- oder besser: **AppMonster**:



Wie am linken Screenshots unschwer zu erkennen, handelt es sich hier um einen vollwertigen Software-Manager. Für 1,49Euro gibt es die Vollversion – nachdem man zunächst natürlich die gratis-Version ausgiebig testen kann. Einmal installiert, läuft die MonsterApp im Hintergrund (äh, nicht wirklich – genau genommen wird sie bei Bedarf vom Event-Manager aufgerufen; nämlich immer dann, wenn etwas neues installiert wurde). Wurde etwas neues installiert, schlägt das Monster zu – und macht sogleich ein Backup auf die SD-Karte. Kommt ein Update – schwupps, das gleiche. Wie, die neue Version tut nicht? AppMonster, hol mal die vorige raus! Kein Thema.

Whipe, Flash, neues ROM – na und? AppMonster installiert, "Batch Install" des aktuellen Backups – und schon sind alle Apps wieder da. Nagut, für die Daten braucht's dann schon ein richtiges Backup-Programm...

Was wollten wir eigentlich? Achso, loswerden wollten wir eine App – wie zu erwarten, findet sich dieser Punkt im jeweiligen Kontext-Menü (also lange auf den entsprechenden Eintrag "drücken"). Dazu muss die App natürlich erstmal aufgerufen werden – das dauert aber unter Android 2.1 nicht einmal halb so lange wie beim "Bordmittel" (ab 2.2 ist die Ladezeit vergleichbar), da sich AppMonster ausschließlich für die vom Benutzer installierten Apps interessiert – und die anderen brav in Ruhe sanften lässt (oder so).

Fazit: AppMonster ist eine App, die auf keinem Androiden fehlen sollte. Eigentlich sollten die "Distributoren" diese App bereits vorinstalliert mit ihren Geräten ausliefern!

Alternative UnInstaller

Neben den "vollwertigen Software-Verwaltern" gibt es dann auch noch die für das schnelle "Iiih – weg damit!" bei denjenigen, die täglich mehrere Apps "durchspielen". Als Beispiele seien da [Fast Uninstaller](#) und [Shake - Uninstall](#) genannt:

Beim *Fast Uninstaller* geht es um die schnelle De-Installation: Antippen – und weg ist die App. Klingt mir für meine Begriffe ein wenig riskant (hab es selbst nicht ausprobiert), tippt man mit seinen "Wurst-Fingern" doch schnell mal daneben. Sympatischer klingt mir da schon *Shake - Uninstall*: Das ist doch mal ein interessantes Konzept: Man aktiviert diesen Service – und dann schaut man sich all die Apps an, wo's einen ohnehin schüttelt. Fazit: Das Gerät ist anschließend gut aufgeräumt 🤖 Interessant wird es nur, wenn man seinen Homescreen verbockt hat – was wohl passiert, wenn es einen da schüttelt?

Apps aus "alternativen Quellen"

Die Voreinstellung eines Android-Smartphones besagt: "Du sollst keine anderen Quellen haben neben mir". Und "mir" meint natürlich den Google Market. Dahinter steht der Sicherheits-Gedanke: Apps sollten nur aus vertrauenswürdigen Quellen installiert werden. Und die einzige derartige, die Google hier definitiv kennt, ist nun einmal Google.

Hin und wieder will/muss man aber mal eine App aus "alternativen Quellen" installieren: Sei es, dass einem der Entwickler was zum "testen" zugeschickt hat ("Schau mal, ob das Dein Problem löst!"), oder eine App mit dem Browser heruntergeladen wurde, da das Phone es im Market nicht findet – oder, oder, oder... OK, die *.apk-Datei haben wir nun – aber wie die App installieren?

Klar kann man das *.apk einfach in den passenden Ordner von *AppMonster* (siehe oben unter "Alternative Verwaltung") packen, und es dann damit installieren. Einfacher machen es zahlreiche Datei-Manager, die dann beim Antippen einer solchen Datei den Installer aufrufen. Zu den beliebtesten Kandidaten hier zählen [Astro Dateimanager](#) und [ES Dateimanager](#) – beide gut erweiterbar, und z. B. auch für den Zugriff auf lokale Netzwerke via FTP oder Windows-Freigaben (AKA "SMB") geeignet.

Welche dieser Möglichkeiten man aber auch verwenden will: Immer kommt der Hinweis "Du darfst hier nicht rein!" – denn zuerst muss die Installation aus "Fremdquellen" generell einmal erlaubt werden. Mit einem kleinen Häkchen an der richtigen Stelle. Dieses findet sich unter *Einstellungen* → *Anwendungen*, und ist dort mit "Unbekannte Quellen" beschriftet.

Apps organisieren

Jetzt sind jede Menge Apps installiert und die Frage drängt sich auf: Wie soll man da den Durchblick behalten? Öffnet man den "Drawer" (also die Liste der auf dem Gerät verfügbaren Apps), ist die Liste recht lang. Und nicht unbedingt übersichtlich. Alle 87 Apps (oder wieviel auch immer) teilweise ohne jede erkennbare Ordnung (oder im besten Falle alphabetisch sortiert) in einem Ordner.

Zwanzig mal hin-und-her scrollen auf der Suche nach der zu startenden App ist nicht jedermanns Sache. Wie leicht passiert es, dass man ein wenig "zu kräftig schubst" – und schon scrollt die Liste in einem Wahnsinns-Speed vorbei. Oder man schubst "zu langsam" – und das dumme Teil meint, die gerade unter dem Daumen befindliche App starten zu müssen... Und das "vollklatschen" aller Desktops mit Icons für jedes App ist auch nicht unbedingt die wahre Lösung. Was also tun?

Apps Organizer und Folder Organizer

Zwei hilfreiche Kandidaten sind [Apps Organizer](#) und [Folder Organizer](#).



Beim Start von *Apps Organizer* scannt dieser zunächst alle installierten Apps, was ein paar Sekunden dauert. Anschließend kann man zunächst jeder App ein oder mehrere Label zuweisen. Nun ist es möglich, die Apps nach diesen Labels zu browsen (linkes Bild). Schon Mal ein Fortschritt – die zu durchsuchende Liste wird kürzer.

Aber das ist natürlich noch nicht alles – denn die App bietet auch passende Widgets (den Umgang mit diesen zeige ich noch ausführlich im Abschnitt [Home-Screen & Widgets](#)). Für jedes der erstellten Labels sowie für "Favoriten" (Apps können hier als solche definiert werden) lässt sich auf diese Art ein Icon auf den Home-Screens platzieren. Tippt man dieses an, öffnet sich ein Fenster, welches die zugehörigen Apps auflistet (siehe rechtes Bild). Je nach Bildschirmgröße (und -auflösung) sowie Anzahl der Apps mit dem zugehörigen Label ist nun oftmals gar kein Scrollen mehr nötig: Der Start einer App klappt jetzt also mit nur zwei Tapps! Na, das ist doch was!

Ganz ähnlich sieht es übrigens bei *Folder Organizer* aus (diese beiden Apps sind sich halt sehr ähnlich), der zusätzlich auch noch Transparenz und einiges anderes unterstützt:



Folder Organizer bezeichnet sich selbst als "the evolution of Apps Organizer". Es kann alles das, was Apps Organizer auch kann – und mehr: Nicht nur Apps können hier mit Labeln versehen werden, sondern auch Lesezeichen, Kontakte und "Shortcuts" (z. B. zu Systemeinstellungen). Wie auch bereits bei *Apps Organizer* werden hier auch Iconssets unterstützt, um die Folder mit den passenden Bildchen zu versehen.

Diese zusätzlichen Funktionen wollen dann aber auch entsprechende "Permissions" bekommen: So fordert die App u.a. Kontaktdaten lesen, Kontaktdaten schreiben, und Telefonnummern direkt anrufen.

Weitere Kandidaten

Natürlich gibt es noch eine ganze Reihe weiterer Kandidaten – die sich jedoch im Groben und Ganzen letztendlich weitgehend mit einer der beiden gerade vorgestellten Apps vergleichen lassen. In meinem zugehörigen Thread bei AndroidPIT ([Apps Organisieren](#)) stelle ich Euch diese näher vor. Außerdem bieten auch einige „alternative Launcher“ (siehe [Home Replacements](#)) die Möglichkeit, Ordner direkt im „App-Drawer“ anzulegen.

Bekannte Probleme

Einige der genannten Apps zeigen vielleicht das Problem, dass hin und wieder Icons vom Home-Screen verschwinden – oder unbrauchbar werden. In der Regel

hat der Anwender dann ein HTC-Gerät, und verwendet die "Sense" Oberfläche – die sich leider nicht so ganz an die Android-Standards bei Widgets hält. Ob sich das umgehen lässt, bereits Lösungen existieren, oder HTC eventuell nachgebessert hat, weiß ich nicht zu sagen; hier hilft nur, im Forum zu fragen bzw. bei den Entwicklern und/oder HTC nachzuhaken.

Google Market Ergänzungen und Alternativen

Da so manchem der Google Market unzureichend ist ("vor lauter Spam nix zu finden"), oder er auch so manches Mal einfach nur zickt ("meine Downloads starten nicht" etc.), habe ich ja bereits [zuvor](#) ein paar alternative Market-Apps benannt. Weiteren Möglichkeiten widmet sich dieses Kapitel.

Market Ergänzungen

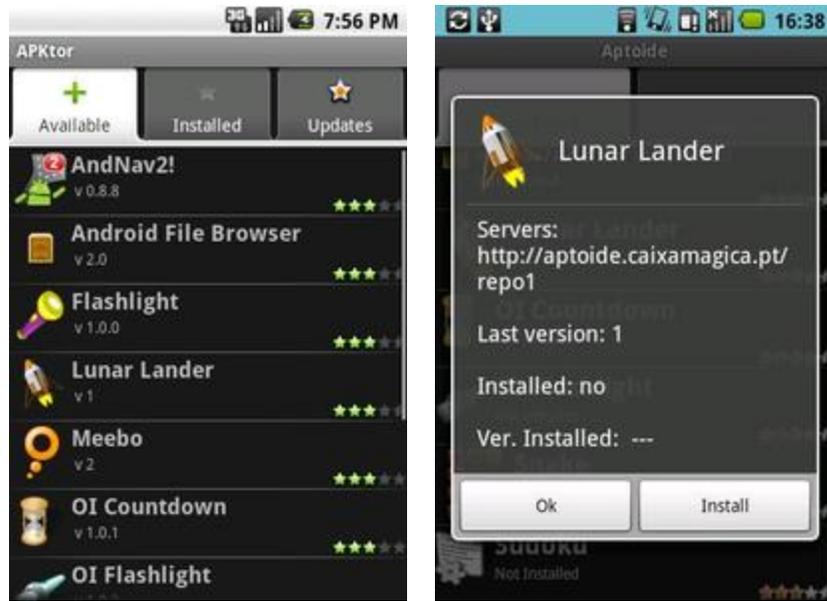
Wie lässt sich mehr aus dem "originalen" Google-Market herausholen? Da gibt es Apps, die sich in die Market-App integrieren – sowie solche, die sie von außen "manipulieren". Ist ein Androide z. B. mit "root" (siehe [Super-User "root"](#)) ausgestattet, lassen sich mit [MarketEnabler](#) die regionalen Beschränkungen aushebeln: Diese App gaukelt dem Market vor, man wäre mit einem ganz anderen Provider in einem ganz anderen Land unterwegs. Und mit AT&T in den Staaten dürfen "US only" Apps natürlich installiert werden 😊

Auch ohne "root" funktioniert der [Market Comments Reader](#), und zeigt die "versteckten" Kommentare an (Schon mal gewundert, wie eine App x* bei >0 Kommentaren kriegen kann, obwohl da kein Kommentar steht?).

Öffentliche Märkte

Linux-Anwender kennen "Software [Repositories](#)": Hier werden Software-Pakete bereitgehalten (und Updates für selbige gepflegt), welche die Entwickler selbst einstellen. Je nach Betreiber des jeweiligen Repositories sind Restriktionen für das Einstellen hier wenig bis gar nicht vorhanden. Der lesende Zugriff seitens der Anwender (für die Suche nach Software und deren Installation auf dem eigenen Rechner) ist entweder allen möglich ("public" bzw. "öffentliches" Repository), oder nur einem ausgewählten Personenkreis (z. B. firmenintern, oder für produktbezogene Entwicklung).

Einer der bekanntesten Repository-Typen ist [APT](#), das **A**dvanced **P**ackaging **T**ool – hauptsächlich bei [Debian](#) und dessen [Derivaten](#) im Einsatz. Gibt es aber auch für Androiden-Soft:



Mit Tools wie dem [APKtor](#) (linkes Bild) oder [Aptoide](#) (rechtes Bild – der Name ist ganz offensichtlich eine Kreuzung aus "APT" und "Androide") lässt sich auf derartige Repositories zugreifen. Da auch die Tools zur Pflege solcher Repositories existieren (und zwar als OpenSource Anwendungen), steht auch dem eigenen Repository (etwa für Entwickler, oder auch Firmen) nichts im Wege.

Topp Apps, Apps Sonderangebote, und ähnliches

Was ist gerade "In"? Wo gibt es die neuesten Killer-Features, die besten Apps, die Top-X Apps? Bei einer mindestens 6-stelligen Zahl verfügbarer Apps ist das mit den bislang aufgeführten Apps schwer festzustellen. Man sieht zwar die Anzahl der Bewertungen und evtl. auch Downloads – um aber zu wissen, was das für die aktuelle Version heißt (wer weiß, vielleicht ist der Support ja längst eingestellt?), müsste man den Trend verfolgen.

Und warum schreibe ich das? Natürlich gibt es auch dafür wiederum passendes Androiden-Futter: So stellt Adam Curry mit seiner bereits vom iPhone her bekannten App [Big App Show](#) regelmäßig Aktuelles vor. [AppAware](#) klemmt sich hinter die Trends und zeigt, welche Apps derzeit gerade besonders häufig installiert werden, Topp-Apps der Community sind, oder im eigenen geografischen Umfeld am häufigsten genutzt werden (wer mag, darf aus dieser App heraus auch die eigene App-Auswahl twittern). Ähnliches bieten auch [Appreciate](#) – mit Facebook-Support (wer's mag), [Fling](#), [Appsfire](#), und weitere.

Auf Schnäppchen-Jagt geht es zum Beispiel mit [Apps on Sale](#): Diese App spürt Sonderangebote auf (welche App wird gerade günstiger angeboten?).

Datensicherung

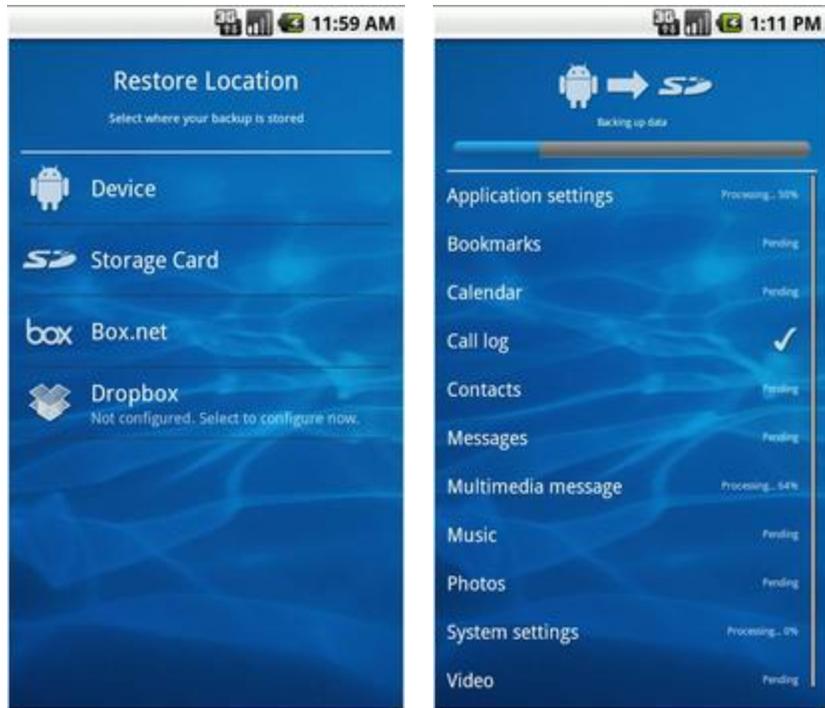
Jetzt ist klar, wie die Apps auf den Androiden (und auch wieder von selbigem herunter) kommen, wie man sie verwaltet und organisiert. Natürlich würde es völlig den Rahmen sprengen, die Funktionsweise aller möglichen Apps hier zu erklären – aber einen wichtigen Punkt gibt es noch: Backups. Vielleicht muss das Android-Gerät ja irgendwann auf Werkseinstellungen zurückgesetzt werden, ein neues Phone kommt ins Haus, oder ein Update "zerschießt" etwas! Die Apps lassen sich zur Not von Hand wieder zusammensuchen. Aber warum umständlich, wenn es auch einfacher geht? Darüber hinaus ist es gut zu wissen, dass die Daten dann wieder zur Hand sind – denn die lassen sich nicht so leicht wieder "irgendwo auftreiben".

AppMonster habe ich ja bereits unter [Anwendungen verwalten](#) kurz vorgestellt: Es sichert bei jeder Installation (und jedem Update) die jeweilige App. So lässt sich nicht nur zu einer beliebigen, bereits zuvor einmal installierten, Version einer App zurückkehren – sondern auch auf einen Rutsch die jeweils aktuelle Version jeder zuvor installierten App aufspielen. Dies macht zum Beispiel bei einem Gerätewechsel, aber auch nach einem Werksreset viel Sinn – vor allem, wenn nicht all Apps direkt über den Google Android Market installiert wurden. In diesem Falle ginge das nämlich auch über die Market-App – allerdings nur, solange die App auch noch im Market verfügbar ist.

Was aber ist mit Anwendungs-Daten? Was ist mit den Kurznachrichten, Telefonbüchern, und so weiter? Die Kontakte lassen sich noch aus der gleichnamigen App (Menü→Importieren/Exportieren) sichern bzw. wieder herstellen. Oder sie werden mit dem Google-Konto synchronisiert, was auch mit den Kalender-Einträgen geht. Was aber nicht unbedingt jeder will. Für alles andere steht gar kein Bordmittel bereit (warum eigentlich nicht?). Aber auch hier gibt es Abhilfe:

Allgemeine Backups

Wirklich *alles* vollständig sichern – das klappt mal wieder nur mit "root". Die Killer-App hierfür heißt [Titanium Backup](#), und ich werde im Abschnitt [Fortgeschrittenes](#) (Kapitel [Vorinstallierte Apps entfernen](#)) näher auf sie eingehen – denn diese App setzt "root" voraus. Doch auch für nicht gerootete Androiden gibt es durchaus brauchbare Lösungen.



Wie zum Beispiel [Sprite Backup](#) (beide Bilder). Die App verspricht eine vollständige Sicherung – und interessanterweise ist hier nirgendwo etwas von root zu lesen? Sicherungen können wahlweise auf der SD-Karte – oder aber auf externen Rechnern (z. B. via FTP) abgelegt werden. Auch Dropbox-Accounts lassen sich hier nutzen.

Diese beiden (und einige weitere) Apps sind somit grundlegende Rundum-Lösungen, die die meisten Sachen abdecken sollten. Etwas derartiges gehört eigentlich auf jedes Android-Gerät – erstaunlich, dass es dort nicht bereits vorinstalliert ist.

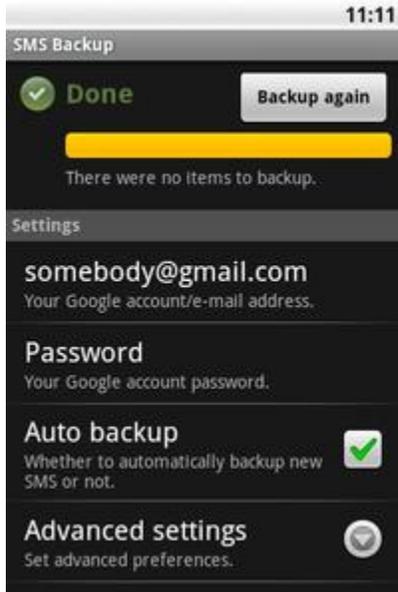
Daten-Backups auf die SD-Karte

Wer dedizierte Lösungen für spezielle Daten sucht, und letztere nicht irgendwo im Netz ablegen möchte, findet hier zahlreiche Lösungen. So lassen sich etwa Kurznachrichten (SMS) mit [SMS Backup & Restore](#), [TxtArchive SMS Backup](#), sowie [TextractLite MMS & SMS Backup](#) (letztere unterstützt auch MMS) auf die SD-Karte sichern. Für Anruflisten tun dies z. B. [Call Logs Backup & Restore](#) und [Backup Call History](#).

Um die Kontakte kümmert sich zum einen die Kontakte-App selbst (über Menü→Importieren/Exportieren), und zum anderen z. B. [UiA – Backup Contacts](#), Lesezeichen können mit [Bookmark Sort & Backup](#) & Co. gesichert werden. Und dann gibt es noch diverse Kombi-Lösungen, die sich wie z. B. [Mobile Backup II](#) um Kalender, Kontakte, SMS, und Anruflisten kümmern.

Ausführlichere Infos dazu finden sich in [diesem Forums-Beitrag](#).

Online-Backups



"Das Wetter. Heute ist es bewölkt." So in etwa begrüßte mich mein Androiden-Wecker ([Alarm Droid](#)) heute morgen. Und ja: Jetzt geht es um "Backups in die Cloud". Das heißt zum Einen: Die Daten sind fast immer von überall erreichbar. Zum Anderen heißt es aber auch: Sie landen auf fremden Servern. Abwägen muss das jeder für sich selbst.

Da wäre zunächst [SMS Backup](#) (Bild links) zu nennen, das Kurznachrichten in IMAP-Ordern bei Google Mail ablegt – was für viele die "Vertrauensfrage" sicher bereits beantwortet. Voraussetzung dafür ist natürlich, dass IMAP dort auch aktiviert ist – was man vom PC aus erledigen kann. Sodann sind auch automatische Backups möglich. Auch [SMS Backup +](#) tut dies – nutzt aber zusätzlich den Google Kalender als Datenablage, und kümmert sich außerdem auch gleich um MMS und Anruf-Listen. Doch während sich derart

gesicherte SMS auch wieder auf dem Androiden herstellen lassen, wird dies für MMS und Anruflisten (derzeit noch) nicht unterstützt.

Mit [mIQ Backup](#) ist auch ein Komplett-Backup hier mit von der Partie. Der Hersteller schreibt dazu: *Der einfachste Weg, um Backup-& verwalten Sie Ihr Handy! Mit mIQ, Ihre Kontakte, Texte, Kalendereinträge, Anrufe, Fotos, Videos & Voicemails sofort hochladen & als sie sind sicher backed-up auf einer privaten Web Account passieren.* Google Translate lässt grüßen.

Backups für spezielle Apps

Jaaa – und dann wären da noch die besonderen Spezialitäten. Böse, böse. Böse Vögel zum Beispiel: Hier lassen sich mit [AngryBirds Backup](#) die Daten sichern und wieder herstellen. Oder mit [Angry Birds Seasons Backup](#) Backup, Restore & Unlock für Rovio's Angry Birds -Seasons (inkl. Highscores) mit der SD-Karte verwalten.

Für Freunde des sozialen Netzwerkens gibt es u.a. [Facebook Backup Basic](#) (auch als gratis Trial oder, für ein paar Taler mehr, als ausgewachsene "Pro" Version). Hier läuft das Backup aber anders herum: Facebook-Bilder werden auf die SD-Karte geladen.

Als ganz nützlich bei einem anstehenden Wipe kann sich [APN Backup & Restore](#) erweisen. Diese App kümmert sich um die mobilen Internet- und MMS Zugänge. Also vor dem Wipe sichern, danach wieder herstellen. Statt alles wieder von Hand einzutippen.

Konfiguration

Bei Android lässt sich so einiges konfigurieren. Und mit jeder neuen Version kommen neue Dinge hinzu. Ich möchte hier jetzt nicht auf alles eingehen – doch einige zentrale Einstellungen finden sich hier erläutert.

Die folgenden Dinge sind alle in den "Einstellungen" von Android untergebracht. Wie man dorthin gelangt? Vom "[Home-Screen](#)" ausgegangen, geht es zunächst über die "Menü"-Taste ins Menü, und von hier in den Punkt "Einstellungen". Dann geht es entsprechend weiter, wie in den folgenden Abschnitten beschrieben...

WLAN

Klar, mit so einem Smartphone möchte man gern ins Netz. Und wenn man noch keinen vernünftigen Daten-Tarif gebucht hat: Was liegt da näher, als das heimische WLAN zu nutzen? Oder das bei Freunden und Verwandten? Zumal es in der Regel ja auch schneller ist als die [mobile Datenverbindung](#). Was also ist zu tun?

In den Einstellungen wählen wir den Punkt "Drahtlos & Netzwerke". Hier lässt sich WLAN schon erst einmal generell aktivieren (indem man das passende Häkchen setzt). Sodann tauchen wir in den Punkt "WLAN-Einstellungen" ab – und gelangen zu einem Bildschirm, der dem rechts dargestellten ähnelt.

Der erste Punkt entspricht dem generellen Aktivieren der WLAN-Funktionalität (wie auf der vorigen Seite). Ist WLAN aktiv, und mit einem Netzwerk verbunden, wird das hier auch angezeigt. Mit dem zweiten Punkt kann man sich "unterwegs" über verfügbare offene WLAN-Netzwerke informieren lassen. Wer jedoch vertrauliche Daten auf seinem Androiden hat, sollte mit solchen Netzen vorsichtig sein: Man weiß ja nie...

Darunter nun werden alle aktuell verfügbaren WLAN-Netze aufgelistet. Auch wird dargestellt ob (und wenn ja wie) diese verschlüsselt sind. Hier sollte also auch das "eigene" WLAN (bzw. das, in welches sich eingebucht werden soll) nun stehen. Einfach antippen, ggf. den Schlüssel (das "Verbindungs-Passwort") eingeben, und auf "Verbinden" tippen – und wenige Sekunden später sollte die Verbindung stehen.

Einmal eingegeben, merkt sich übrigens Android die Verbindungsdaten: Kommt man das nächste Mal bei aktiviertem WLAN in die Nähe dieses Netzes, erfolgt die Verbindung automatisch.



Mobiles Datennetz



Der "moderne Mensch" ist ja heutzutage permanent online. Unser WLAN können wir aber nicht überall hin mitnehmen. Was tun?

Die Antwort heißt: Einen passenden Daten-Tarif (Volumentarif oder Flatrate) buchen, und das "mobile Datennetz" konfigurieren! Ersteres gibt es beim Provider – und letzteres findet sich wieder unter "Drahtlos & Netzwerke". Bezeichnenderweise unter dem Punkt "Mobilfunknetze". Der führt dann zu einem Bildschirm, der dem links dargestellten stark ähneln dürfte.

Ein kurzer Blick auf das Bild dürfte auch bereits helfen, eine der größten Sorgen auszuräumen: Was ist mit meinen Daten-Kosten, wenn ich im Ausland bin? Ja was? Das hängt ganz davon ab, wie hier die Häkchen gesetzt sind. Standardmäßig sind die Häkchen bei "Roaming" *nicht* gesetzt (also wie auf dem Bild zu sehen). Im Ausland bzw. generell im Netz eines "Fremdanbieters" wird daher die Datenverbindung gar nicht erst aufgebaut. Daher sollten hier auch keine diesbezüglichen Kosten entstehen.

Ganz unten ist bei mir noch ein Häkchen gesetzt, welches die Datenverbindung auf "2G" beschränkt. Das ist zwar nicht so schnell wie 3G oder gar 4G – spart aber einiges an Energie, sodass ich mit einer Akku-Ladung länger auskomme. Für ein wenig E-Mail und Web ist das auch völlig ausreichend. Sollte ich tatsächlich einmal mehr Durchsatz benötigen, kann ich das jederzeit umschalten.

Woher weiß Android denn nun, wie es ins Internet kommt? Diese Einstellungen verbergen sich hinter den "Zugangspunkten". So manch [Custom-ROM](#) (wie z. B. [CyanogenMod](#)) ermittelt diese Konfiguration automatisch: Anhand der SIM-Karte erkennt es den Anbieter, und ordnet die entsprechenden Zugangsdaten aus seiner Datenbank zu. Wer dieses Glück nicht auf seiner Seite hat, findet die passenden Zugangsdaten jedoch hoffentlich im [Anhang](#) – andernfalls lassen sie sich beim Provider erfragen.

Tethering

An dieser Stelle folgt oft die Frage: "Ich habe da noch ein Tablet/Notebook/... Kann ich jetzt irgendwie die mobile Datenverbindung mit nutzen?" Vor Android 2.2 (aka "Froyo" oder "Frozen Yoghurt") hieß die Antwort eindeutig: Jein. Mit [root](#) und einer App wie [Wireless Tether](#) ließ sich dies erreichen. Ansonsten galt der übliche Spruch: "Ohne root sich nix tut".

Zum Glück hat sich das eindeutig geändert: Ab Froyo gehört Tethering zur "Standard-Ausrüstung", und ist natürlich auch noch bei Gingerbread mit an Bord (Screenshot rechts). Wer sicher gehen möchte, dass kein Dritter "mitsurft", kann das Netzwerk über USB weiterreichen. Einfacher geht es jedoch, wandelt man seinen Androiden in einen "mobilen Hotspot" um. Hierzu wird der zweite Punkt aktiviert, und die Details werden unter "WLAN-Hotspot-Einstellungen" eingetragen: Eine SSID (Name für den Zugangspunkt) kann nach Gusto vergeben, eine Verschlüsselung gewählt, und natürlich auch der zugehörige Schlüssel/Passwort hinterlegt werden. Und schon steht dem Surf-Vergnügen nichts mehr im Wege...

Wer immer noch ein wenig unsicher ist, hat vielleicht bemerkt: Da gibt es noch einen Punkt namens "Hilfe". Stimmt. Und da wird das Ganze auch nochmal erklärt – falls dieses Buch gerade mal nicht zur Hand ist...



Internet-Telefonie



Oh nein, den Netzanbietern hat das sicher nicht sonderlich gefallen. Aber dennoch: Seit Gingerbread gehören die [SIP](#)-Einstellungen zu den Bordmitteln. Wer noch kein Gingerbread auf seinem Androiden hat, muss dafür zu Drittanbieter-Software wie [SIPGate](#) greifen, kommt aber auch zum Ziel.

Versteckt sind diese Konfigurationsdaten unter den "Anrufeinstellungen" (also *nicht* unter "Drahtlos & Netzwerke"), und zwar ganz am Ende des Bildschirms (siehe Screenshot links). Von hier aus geht es über "Konten" in die Übersicht eingerichteter SIP-Konten – beim ersten Aufruf dürfte diese leer sein. Ein Button, beschriftet mit "Konto hinzufügen", wartet jedoch schon auf Betätigung. Die passenden Zugangsdaten gibt es beim VoIP-Anbieter. Essentiell sind Nutzernamen, Passwort und Server – etliche

optionale weitere Einstellungen wie Proxy u.a. stehen ebenfalls zur Verfügung.

Mittels einer Checkbox lässt sich ein SIP-Konto als "primär" festlegen – was aber nur bei mehreren Konten interessant ist. Über dieses Konto werden dann ausgehende Anrufe geführt.

Zurücksetzen

Zurück auf LOS! Was ist los? Wo ist LOS? Und was, bitte, wohin zurücksetzen?

Den "älteren Semestern" unter uns ist sicher die "Reset"-Taste am PC noch ein Begriff. So als Reißleine, Notbremse, letzte Ausflucht, wenn nichts mehr geht. Auch das ist eine Form von "Zurücksetzen". Unter Android gibt es da mehrere Rücksetz-Möglichkeiten, mit zum Teil recht unterschiedlichen Auswirkungen. Und daher auch recht unterschiedlichen Verwendungszwecken...

Softreset

Dieses "weiche zurücksetzen" lässt sich am ehesten mit dem "Affengriff" unter Windows (Strg-Alt-Delete) vergleichen. Nur dass die Tastenkombination, je nach Gerät, noch wesentlich abenteuerlicher ist. Bei HTC-Geräten z. B. üblicherweise das gleichzeitige Drücken der *Leiser*-Taste, der *Action*-Taste (Trackball), und des Einschaltknopfes. Möglichst ohne das Gerät dabei fallenzulassen...

Bewirken soll das Ganze dann ein "sanftes" Herunterfahren des Systems – üblicherweise wenn gar nichts anderes mehr funktioniert (sonst könnte man ja auch normal über das Menü abschalten).

Hardreset

Hierbei handelt es sich um das "Zurücksetzen auf Werkseinstellungen", was sich zum Beispiel über den genau so benannten Punkt unter Einstellungen→Datenschutzeinstellungen erreichen lässt. Dabei werden alle vom Anwender installierten Apps sowie sämtliche Einstellungen gelöscht – das Gerät ist somit wieder in einem "jungfräulichen" Zustand.

Wird es anschließend wieder angeschaltet, muss mit der Einrichtung ganz am Anfang begonnen werden – genau so, als hätte man das Gerät gerade zum ersten Mal aus der Originalverpackung geholt. Das ist auch einer der Gründe, für den diese Funktionalität benötigt wird: Wenn das Gerät verkauft/verschenkt/weitergegeben werden soll. Natürlich möglichst ohne private Datenspuren darauf zu hinterlassen.

Außerdem ist dies auch noch ein "Last Resort", wenn der Androide komplett verrückt spielt. Der Hersteller verlangt dies meist, um Probleme mit der Hardware ausschließen zu können: Löst ein *Hardreset* das Problem, ist die Hardware nämlich ganz offensichtlich unschuldig – es hat sich nur die Konfiguration verdreht.

Bevor man zu diesem Schritt greift, kann man noch folgendes versuchen:

Wipe des Dalvik-Cache

Dieser erzwingt die Neu-Übersetzung des Programmcodes aller installierten Apps (siehe [Dalvik](#) bei den [Begriffs-Erklärungen](#)). Das wäre ein Schritt, den man bei nicht behebbarem "ungewöhnlichen Verhalten" des Systems noch durchführen kann, ohne das ganze Gerät komplett auf Werkseinstellungen zurückzusetzen.

Android-Apps sind in Java geschrieben, und Java ist bekanntlich Plattform-unabhängig. Vereinfacht ausgedrückt, ist das ein Zwischending zwischen einem Skript wie einer Batch-Datei oder einem PHP-Skript, und einem kompilierten Programm. Vor der eigentlichen Ausführung muss da also noch eine Übersetzung in Maschinensprache stattfinden, die möglichst nah am verwendeten System ist. Bei Java nennt man dies "Byte-Code". Um die schmalen Ressourcen von mobilen Android-Geräten noch schonender zu nutzen, geht man bei Dalvik-VMs (so nennt sich die spezielle "Java-Variante" unter Android) noch einen Schritt weiter, und nutzt noch mehr Optimierungs-Möglichkeiten.

Damit dies nun nicht bei jedem Aufruf einer App geschehen muss (das wäre unerträglich langsam), macht Android das unmittelbar nach der Installation einer App – und legt den optimierten "Byte-Code" im sogenannten [Dalvik-Cache](#) ab. Wird dieser gelöscht, erzwingt dies lediglich eine Neu-Übersetzung (sagte ich ja schon) – die Anwendungsdaten und Einstellungen bleiben jedoch vollständig erhalten.

Dieser Schritt ist definitiv zu empfehlen, wenn ein (neues/anderes) [Custom-ROM](#) eingespielt werden soll. Bei "offiziellen Updates" sollte sich der Hersteller darum kümmern, sofern dies nötig ist.

MIT ANDROID ARBEITEN

Schaltzentrale: Home-Screen, Widgets & "Home Replacements"

Wenn es bei Android so etwas wie eine "Schaltzentrale" gibt, ist dies sicher am ehesten der **Homescreen**: Hier starten alle Aktivitäten. Das ist es, was der Anwender nach dem Start seines Androiden zu sehen bekommt – von hier startet er seine Apps – hier platziert er (so er dies tut) seine Übersichten wie aktuelle Kalender-Ereignisse, News-Feeds, und so weiter. Daher macht es durchaus Sinn, dass sich das erste Kapitel dieses Abschnittes zunächst diesem widmet.



Eigentlich sollte ich besser sagen: "diesen". Klar gibt es einen "Standard-Launcher" bzw. "Stock-Launcher" ("Launcher" ist ein anderes Wort für den Homescreen, welches obigen Sachverhalt betont: Dass man von hier alle Aktivitäten "launcht", also startet). Auf fast allen Geräten ist jedoch bereits eine Alternative installiert: Da wäre HTC mit ihrem *Sense* Launcher, Motorola mit der *MotoBlur* Oberfläche, etc. pp.. Und zahlreiche Alternativen sind im Market verfügbar – wie etwa der [Zeam Launcher](#) (linkes Bild), oder [Launcher Pro](#) (rechtes Bild). Jeder hat so seine Besonderheiten und Vorteile gegenüber den anderen – beispielsweise "scrollable Widgets" u.a. bei den beiden gerade genannten, dann wären da auf's Ressourcen-Schoneren getrimmte Launcher, minimalistische Launcher (sowie deren Gegenstücke) – und, und, und. Ein genauerer Überblick findet sich natürlich wieder im passenden [Forums-Thread](#).

Docking Bar

Das ist i.d.R. der Bereich "unten", in dem besonders häufig genutzte Funktionen verankert sind (auf obigen Screenshots auch gut zu erkennen). Bei einigen *Launchern* sind diese Aktionen "fest verdrahtet", und lassen sich nicht ändern/anpassen. Die Auswahl der Aktionen ist dabei für die Masse durchaus tauglich: Telefon ist immer dabei (das Gerät heißt ja auch "*SmartPhone*", und nicht "*MiniComputer*" – auch wenn die Grenzen da schwer zu definieren sind), dazu kommen meist Anrufliste und Kurznachrichten, sowie der *App-Drawer*.

Die meisten (mir bekannten) *Launcher* erlauben es jedoch zumindest, die Aktionen selbst auszuwählen. So lassen sich entsprechende "Icons" z. B. bei o.g. *Zeam Launcher* per Drag-and-Drop platzieren (und entfernen), auch die Reihenfolge lässt sich nachträglich ändern. *Launcher Pro* (und einige andere) gehen sogar noch weiter, und lassen den Benutzer an die grafische Ausgestaltung direkt heran. Wer also alles individuell gestalten möchte, kann dies durchaus tun!

App-Icons

Diese lassen sich in der Regel auf dem Launcher (s.o.), und generell auf den HomeScreens platzieren. Letzteres gilt auch für die *Shortcuts* und *Widgets* (siehe unten). Für alle drei ist das Standard-Vorgehen zur Platzierung, eine freie Stelle auf dem "Desktop" "lange zu drücken". Daraufhin öffnet sich ein Kontext-Menü und fragt nach, was es denn sein darf – wobei unsere drei Kandidaten, und ggf. (je nach Launcher) auch noch weitere Dinge zur Auswahl stehen können. Wieder entfernen lassen sie sich wiederum durch "langes Drücken" (diesmal auf dem Icon selbst) und anschließendes "ziehen" auf die sich öffnende (meist rote) Mülltonne.

Unsere *App-Icons* haben nun keine weitere Funktion, als die zugehörige App zu öffnen. Nicht viel, aber mehr braucht es ja oft auch nicht: Von zentraler Stelle die wichtigsten Dinge schnell starten, ohne sich erst durch den "Drawer" (die komplette Applikationsliste) wühlen zu müssen. Benötigt man doch einmal etwas spezielleres, kommen unsere anderen beiden Kandidaten zum Einsatz:

Shortcuts

Nomen est Omen, wie der Latiner sagt: Hier geht es um "Abkürzungen", die einige Apps anbieten. Was auf dem HomeScreen wie ein gewöhnliches (gerade eben beschriebenen) *App-Icon* aussieht, ist es auch – nur mit ein wenig Zusatz-Funktionalität. Es springt bei der zugehörigen App gleich zu einem bestimmten Bildschirm, oder löst eine bestimmte Aktion aus. Ein "klassisches Beispiel" wäre bei [Note Everything](#) zu finden: Die Startseite (mit den Übersichten) überspringen, und direkt eine neue Notiz öffnen. Oder bei den weiter oben unter [Apps Organisieren](#) genannten "Organizern" das Öffnen eines bestimmten Ordners. Bei diesen Dingen handelt es sich um *Shortcuts*.

Dazu muss gesagt werden, dass diese Shortcuts von den Apps selbst bereitgestellt werden müssen: Was die App nicht anbietet, steht da auch nicht zur Verfügung.

Widgets

Gleiches gilt auch für die Widgets: Grafische Elemente, die erweiterte Informationen zur Verfügung stellen – und optional auch noch als Shortcuts dienen können. Einige Beispiele dafür finden sich in den beiden obigen Screenshots:



Widgets von [DroidStats](#), die Informationen zu aktuellen Statistiken (hier: Telefonminuten und SMS) geben – und bei "Antippen" die App gleich auf der zugehörigen Detail-Seite öffnen (links).

Widgets von [Mini-Info](#), die über diverse System-Informationen auf dem Laufenden halten (rechts). Tippt man sie an, wird die App (ganz normal) gestartet.



Ein TaskManager-Widget, welches über freien Speicher sowie die Anzahl gerade laufender Prozesse informiert. Die bei Antippen ausgeführte Aktion ist konfigurierbar – etwa das Starten der App, oder Killen aller "black-listed" Apps. Übrigens: Auch die Uhr im obigen Screenshot ist ein Widget...

App-Drawer

Auch zu diesem zu guter Letzt noch ein paar Worte. Ich habe ihn ja bereits zuvor als die "unübersichtliche Lagerhalle von Icons installierter Apps" erwähnt (naja, nicht mit diesen Worten – aber so kommt es vielen und oft vor). Dem Hören-Sagen nach muss das nicht generell so sein. Es soll Launcher geben, die hier alternativen Implementierungen folgen, und Dinge wie "Reiter", "Unter-Ordner", "Kategorien", und ähnliches anbieten. Wer hier also gern ein wenig aufräumen würde, und einem "alternativen Launcher" nicht abgeneigt ist, sollte bei der Auswahl auch darauf achten. Womit er sich ggf. auch den unter [Apps Organisieren](#) genannten separaten "Organizer" erspart.

Wo wir gerade vom App-Drawer sprechen: Ab Android 4.0 („Ice Cream Sandwich“) findet sich in diesem ein zusätzlicher Reiter, der verfügbare Widgets auflistet. Somit hat man endlich eine Übersicht darüber, welche Widgets verfügbar sind. Auf den Home-Screen kann man selbige dann befördern, indem man sie ganz doll drückt: Der App-Drawer blendet sich dann aus, und man lässt das Widget sodann an der gewünschten Stelle einfach "fallen".

Steuerzentrale: Einstellungen und "Switches"

Haben wir den *Home-Screen* als "Schaltzentrale" bezeichnet – so ist der Ort, an dem die ganzen Systemeinstellungen getätigt werden, ja wohl die "Steuerzentrale". Und es gibt so einiges einzustellen bei Android, die Liste ist also nicht unbedingt kurz. Hinzu kommt, dass vieles "historisch gewachsen" ist – und somit manche Dinge an den verschiedensten Orten zu suchen sind, obwohl sie aus subjektiver Sicht eigentlich zusammen gehören...

Klar, es handelt sich bei aktuellen Android-Versionen schon um recht komplexe Systeme, wo man an vielen Schraubchen drehen können muss. Doch insbesondere für Neueinsteiger sind das meist zu viele (wobei genau die, die man gerne hätte, natürlich fehlen). Doch auch hier gibt es einige Apps, die für Erleichterung sorgen: Entweder, weil sie die Auswahl auf wesentliche (häufig benutzte) Punkte zusammenstauchen – oder, weil sie in spezifischen Bereichen zusätzliche Einstellungsmöglichkeiten schaffen. Zu beiden Themen lässt sich meine [Übersicht zu Einstellungen](#) im Forum etwas ausführlicher aus.

Mehr Übersicht, bitte!

Wer sich lieber auf Wesentliches beschränken möchte, greift am besten zu Apps wie [Quick Settings](#) (rechtes Bild):



Die App bietet die Möglichkeit, sowohl die Auswahl als auch die Reihenfolge der angezeigten Einstellungs-Punkte zu konfigurieren. Auf diese Weise lässt sich eine sehr personalisierte Konfigurations-Seite erstellen. Sehr spezielle Punkte (die in der Regel selten benötigt werden) stehen aber oft nicht zur Auswahl.



Noch etwas eingeschränkter in Sachen Konfigurierbarkeit ist [Easy Access Settings](#) (links): Hier hat man (meines Wissens) keinen Einfluss auf die angezeigten Punkte oder auch nur die Reihenfolge Ihrer Anordnung. Dafür beinhaltet die App zusätzlich einen App-Manager und auch einen Service-Manager. Verbesserungen an der Bedienbarkeit sind für eine zukünftige Version hier angekündigt.

Zusätzliche Einstellungen

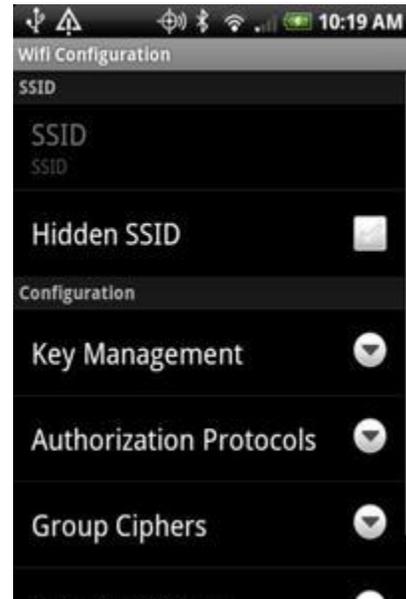
Während die einen es lieber kompakter hätten, gibt es da auch noch die Gruppe derer, denen die vorhandenen Konfigurationsmöglichkeiten nicht ausreichen. Auch dieser kann (in einem gewissen Rahmen) geholfen werden:



So schaltet etwa die App [Spare Parts](#) (links) eine ganze Reihe zusätzlicher Schalterchen frei. Sie ist die umfangreichste App in diesem Bereich, und auch die, die bereits am längsten im Market verfügbar ist. Leider wurde sie aber auch schon ein Weilchen nicht mehr aktualisiert – Nutzer aktuellerer Android-Versionen (Froyo und neuer) müssen also zunächst einmal prüfen, ob noch alles Bestehende passt – und können kaum darauf hoffen, dass neue Features bereits unterstützt werden.

Einige wenige zusätzliche Einstellungen bietet auch [Extra Phone Settings](#) (Bild oben rechts). Mit ihrer Hilfe lässt sich der Androide bei einem ausgehenden Anruf kurz zum Vibrieren bringen, sobald die Gegenseite "abgenommen" hat.

Zu guter Letzt sei noch eine besondere Spezialität kurz erwähnt: Der [Wifi Config Editor](#) (rechtes Bild) ermöglicht es, die Wifi-Einstellungen noch detaillierter vorzunehmen. Weit mehr Einstellungen, als der Standard-Dialog normalerweise anzeigt. Otto-Normal-Benutzer braucht das sicher kaum einmal – doch für manchen "Spezialisten" ist es sicher ein wertvolles Werkzeug.



Von Taskkillern und anderen bösen Buben

Oh ja, ich höre schon die Schreie: "Taskkiller gehören verboten! Android kann das selbst!". Und gleich aus der Gegenrichtung: "Taskkiller muss man haben, mein System läuft jetzt viel flüssiger!".

Wer hat nun Recht? Beide. Keiner. Denn hier gibt es kein einfaches Schwarz und Weiß. Sicher ist jedoch: Wer nicht weiß, wie man eine Spritze setzt, sollte sich nicht als Arzt ausgeben – das kann sonst gehörig in die Hose gehen. Ein guter Arzt weiß jedoch die genannte Spritze so einzusetzen, dass sie dem Patienten hilft. Ist jetzt die Spritze "böse"?

Man sollte also schon genau wissen, was man tut – und Taskkiller, Autostart-Helfer, & Co. können sehr nützlich sein. Wer dies nicht weiß, lässt besser die Finger weg!

Kurz zusammengefasst (ausführliche Erläuterungen finden sich, wie gewohnt, [in einem Forums-Thread](#)): Hier handelt es sich um ein sehr kontrovers diskutiertes Thema. Worin mir allerdings (fast) jeder zustimmen dürfte: Eingriffe ins System setzen eine gute Kenntnis desselben voraus.

Es ist korrekt, dass Android sich um die Speicherverwaltung selber kümmert. Dennoch haben Task-Manager / Task-Killer durchaus ihre Berechtigung – solange man weiß, was man da tut:

- Falsch: "ich will den Speicher freiräumen". Dafür ist der "OOM Killer" (direkt im Android-System integriert, näheres dazu im [Tuning-Kapitel](#)) zuständig.
- Richtig: "eine App hat sich aufgehängt, und blockiert [irgendwas]". Hier ist der Task-Killer angesagt – weil bis der "OOM-Killer" hier zuschlagen würde... Und ein Reboot ist nicht gerade die wünschenswerte Alternative.
- Richtig: "eine App läuft Amok" (Panik-Mode: Man erwischt gerade eine App dabei, wie sie alle persönlichen Daten inkl. Nackt-Fotos auf eine berüchtigte Website hochlädt...). Ohja: Abschießen! Oder gleich abschalten. Weil: Bis der OOM-Killer... genau, da ist es dann eh zu spät...

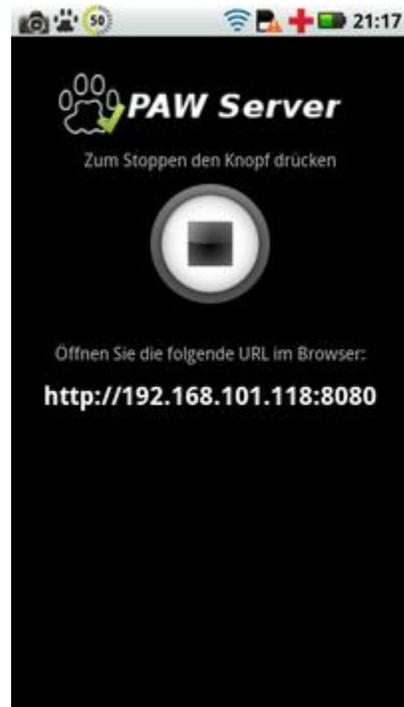
Das Android-Gerät vom PC aus verwalten



Auch die zierlichsten Frauenhände stoßen auf dem Androiden schnell an ihre Grenzen – und so richtig Spaß macht das auf den kleinen Bildschirmen dann nicht wirklich. Abgesehen davon, dass man immer erst suchen muss: Wo war diese Option doch jetzt gleich noch? Und oftmals schmerzlich eine "richtige Tastatur" vermisst. War da noch was? Genau, Inhalte sollen ja ebenfalls noch von A nach B und umgekehrt, also zwischen PC und "dem Kleinen" ausgetauscht werden...

Die beliebteste Lösung für dieses Problem heißt [MyPhoneExplorer](#) – benötigt aber auf PC-Seite ein Windows-Programm, und ist somit nur für Windows verfügbar. Hier vorgestelltes sollte aber möglichst für alle Anwender eine Option sein. Also greife ich nicht zum nächstbesten, sondern zum nächsten und besten:

[PAW Server](#) ist unabhängig von jedweder Plattform, was den PC betrifft. Auf selbigem wird nur ein Web-Browser benötigt – alles andere erledigt die App unter Android, wobei der Androide über WLAN bereitgestellt wird. PAW lässt sich dabei sehr sicher konfigurieren: Nicht nur, dass sich ein gutes Passwort wählen lässt – auch das sicherere HTTPS-Protokoll steht hier zur Verfügung. So kann man durchaus erwägen, im Bedarfsfall eine [Portfreigabe](#) am Internet-Router zu erstellen, um sich z. B. von einem Fachmann helfen zu lassen. Auf der anderen Seite ist es auch kein Problem, bei Freunden/Verwandten auf diese Weise auf seinen Liebling zuzugreifen: Es wird ja keine Zusatz-Software benötigt.



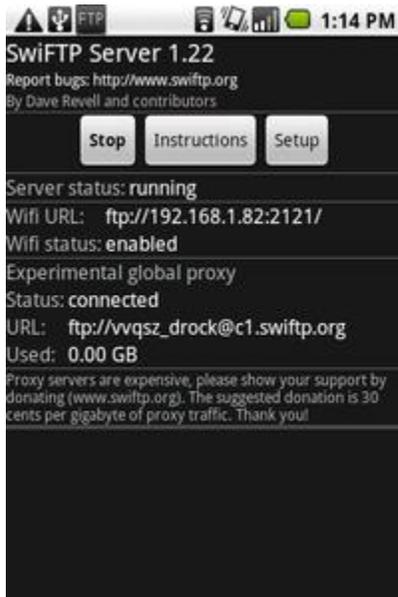
Wie auch beim eingangs genannten MyPhoneExplorer, lassen sich mit *PAW Server* Anruflisten, SMS, Kontakte etc. einsehen, Anrufe initialisieren, SMS schreiben... Und wenn der Hund sich "den Knochen" geschnappt und verschleppt hat, selbigen per Knopfdruck zum Klingeln bringen (den "Knochen", nicht den Hund!) um festzustellen, wo beide denn nun abgeblieben sind. Vorausgesetzt natürlich, die beiden haben beim Spielen nicht das WLAN-Signal verloren...

Natürlich ist auch ein Dateimanager enthalten. Fotos lassen sich ebenfalls durchstöbern (auf Wunsch sogar eines davon als neues Hintergrundbild festlegen), der Androide als Diktier- oder Vorlesegerät, Musik-Player oder auch WebCam nutzen, und vieles mehr.

Programmierern stehen darüber hinaus auch zahlreiche Schnittstellen zur Verfügung, mit denen sich die Funktionalität erweitern lässt – und umgekehrt können sie auch ihre Apps um Funktionalitäten des *PAW Servers* anreichern.

Der *PAW Server* ist also eine gute Wahl. Wer sich allerdings vor einer Entscheidung noch mögliche Alternativen anschauen möchte, findet dazu – wie immer – auch einen passenden [Thread im Forum](#).

Datenaustausch mit dem PC



Alles klar: USB-Kabel anschließen, und die Karte wird am PC freigegeben. Weiß doch jeder. – Ja, schon. Aber zum einen ist das umständlich, zum zweiten ist laut Murphy genau dann kein Kabel zur Hand, wenn man es bräuchte, und zum dritten ist das ja sowas von uncool und unzeitgemäß... Kurzum: Es gibt weit bequemerer, ohne Kabel. Wobei man dem Kabel natürlich zugute halten muss: Sichere Übertragung, und braucht keine Zusatz-Software...

Warum nicht auf dem Androiden Freigaben erstellen, und diese via WLAN nutzen? Sowas geht doch sogar unter Windows! Und was ist mit Android? Ja, auch bei Android. Auch da geht das. Einfach und ressourcenschonend als FTP-Server z. B. mit [SwiFTP](#) (Bild links): Das Installationspaket unter Android bringt keine 80kB auf die Waage, und der Zugriff funktioniert unter Windows, MacOS und Linux

gleichermaßen einfach: Browser öffnen, und die auf dem Android-Screen angezeigte URL eintippen. Schon lässt sich durch das Dateisystem navigieren. Tipp: Unter Windows in die Maske bei Start→Ausführen, bzw. unter Linux bei "Alt-F2" eingeben, das öffnet den Service dann im Explorer bzw. in Konqueror (KDE3) oder Dolphin (KDE4). Und wem das alles nicht zusagt, der greift auf dem PC halt zu Drittanbieter-Anwendungen wie z. B. [Filezilla](#). Dank vom Entwickler bereitgestellter Proxy-Funktion lässt sich mit *SwiFTP* sogar über das mobile Datennetz der Dienst bereitstellen.

Zu spartanisch? Wer es lieber grafisch mag, und auch Fotos und Videos, sowie die Musiksammlung anhand von Covern verwalten möchte, der kann auch zu [WebSharing](#) greifen. Bezahlt wird dieser zusätzliche Komfort nicht zuletzt in Kilo- oder besser Megabyte, und derer gleich zwei – so groß ist nämlich diese App. Da liegen Welten dazwischen.

Auch "echte" Windows-Freigaben sind, z. B. mit [Samba Filesharing](#) möglich. Nicht zu vergessen WebDAV mittels [DavDrive](#) vom Macher des bereits genannten *PAW Server*. Einzelheiten dazu können dem zugehörigen [Forums-Thread](#) entnommen werden.

SICHERHEIT

Was brauche ich wirklich?

Anti-Virus, Anti-Malware, Diebstahlschutz... Was braucht es eigentlich wirklich auf dem Androiden? Klar gibt es auch hier wieder für alles eine App – und natürlich auch eine passende [Übersicht im Forum](#). Das Wichtigste solltet man jedoch (hoffentlich) nicht all zu lange suchen müssen:

GMV

GMV sollte bereits im biologischen Speicher vorinstalliert sein. Leider wird es oft mit Worten wie "No risk, no fun!" deaktiviert – was dann meist unschöne Folgen hat. In der Regel taucht der/die Betroffene kurz darauf im Forum auf, und öffnet einen Thread mit dem aussagekräftigen Titel "HILFEEEE!" (aha, GMV noch immer deaktiviert).

GMV? Was ist das denn nun wieder? Oh-oh... Das sollte eigentlich jeder haben, zumindest ein wenig davon: **Gesunder Menschenverstand**. Hilft enorm. Auch gegen "Viren" und "Malware".

Seien wir doch mal ehrlich: Wie viele Viren gibt es wirklich für Android? Und wie kommen die aufs Gerät? Wie kommt Malware aufs Gerät? Indem man ohne nachzudenken auf alles klickt, was sich bewegt? Indem man eine "böse App" installiert? Die wichtigsten Regeln beachtet, kann so etwas eigentlich kaum passieren. Vor der Installation einer App sollte man sich z. B. folgende Fragen stellen:

- Ist die Quelle vertrauenswürdig?
 - Positiv-Beispiele: AndroidPIT-Market, AppCenter, Google-Market, Website des bekannten (!) Entwicklers
 - Negativ-Beispiele: Bei Rapidshare "gefunden", in einer Tauschbörse aufgetrieben, per eDonkey aus unbekannter Quelle gezogen...
- Sehen die Permissions vernünftig aus?
 - Positiv-Beispiele: Ein Webbrowser muss ins Web, eine SMS-App kann natürlich SMS lesen/schreiben/schicken und braucht ggf. auch (lesend) Zugriff aufs Adressbuch
 - Negativ-Beispiele: Eine Wallpaper-App braucht in der Regel keine Telefonnummern anrufen, ein Ballerspiel muss keine SMS senden.
 - Besondere Vorsicht: Apps, die auf persönliche Daten (Kontakte, Kalender, Nachrichten) zugreifen und gleichzeitig ins Internet wollen. Leider lässt sich bei letzterem (Internet) die Frage der Notwendigkeit nicht so einfach beantworten – es könnte auch einfach nur für Werbung-Laden gebraucht werden...
- Was sagen andere Nutzer zur App/zum Entwickler (Bewertungen, Forum)?
 - Auch hier wieder GMV aktivieren. Kommentare wie "Geil!", "Super", etc. sagen nicht wirklich etwas aus (da hat eher jemand bei deaktiviertem GMV einen Kommentar hinterlassen)
 - Gleiches gilt für manchen negativen Kommentar: Nicht gerade selten passiert es, dass jemand einfach zu blöd war. Oder die Anforderungen der App gar nicht verstanden hat.
 - Nicht alle Bewertungen beziehen sich wirklich auf die App. Die kann schließlich nix dafür, wenn der Market mal wieder klemmt, und daher der Download nicht funktioniert. Oder die HD-Video-App, die mindestens WVGA benötigt, mit dem Motorola Flipout (mini-Display) im Market nicht gefunden wird...
 - Ganz neue App? Noch keine Bewertungen? Im Zweifelsfall im Forum nachfragen, ob schon jemand die App kennt und etwas dazu sagen kann.

Natürlich können andere Apps aus der "Sicherheits-Abteilung" eine gute Ergänzung zu GMV sein. Insbesondere bei [Verlust des Gerätes](#) – denn dagegen macht auch GMV nicht immun...

Rundum-Sorglos-Pakete

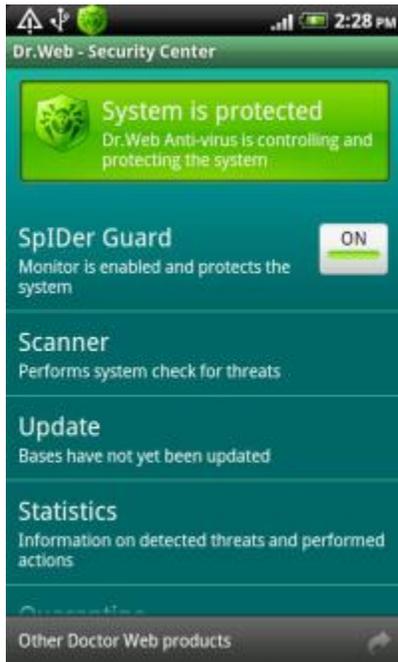
Das sind die Apps, die gleich alle Bereiche abdecken. Also [Anti-Virus](#), [Anti-Malware](#), und "[Diebstahlschutz](#)" in einem. Ein Beispiel dafür ist [NetQin Antivirus](#) (in verschiedenen Varianten, u.a. für verschiedene Android-Versionen bzw. mit unterschiedlichem Funktionsumfang und "Pricing", verfügbar; Bild rechts). Der Name lässt bereits auf die Haupt-Tätigkeit schließen: Das Vorgehen gegen Viren und Malware. Und zwar sowohl in Echtzeit (App läuft im Hintergrund), als auch "On Demand" ("auf Verlangen": Sie lässt sich also bei Bedarf veranlassen, das gesamte Gerät zu prüfen).

Geht das Gerät einmal verloren (d.h. es wurde entweder verlegt, oder ein Langfinger hat es "abgegriffen"), kann man z. B. einen lauten Alarm auslösen ("DIEBE! ICH BIN EIN GEKLAUTES HANDY!" – äh, ich weiß nicht, wie es mit der Auswahl des Sounds aussieht...). Oder aber in wilder Panik gleich alle Daten löschen und das Gerät sperren lassen. Sowa geht einfach per SMS mit dem entsprechenden "Codewort". Natürlich kann man auch erstmal seinen GMV aktivieren, und sich auf der Karte (Google Maps) zeigen lassen, wo sich der Androide gerade herumtreibt. Ob dafür irgendwo ein Account benötigt wird, und wie das genau funktioniert, fand ich leider nirgendwo beschrieben...

Hinzu kommen noch Tools zum Sichern persönlicher Daten, Memory-Booster, Task-Killer, Traffic-Monitor, File-Manager... (was soll das Ding eigentlich nicht machen?!?). Bei so vielen Features ist auch die Anzahl der geforderten Permissions entsprechend umfangreich...



Anti-Virus und Anti-Malware

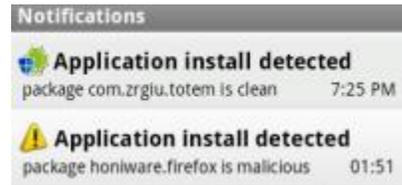


Viren und Malware (nein, hier sind jetzt keine Apps zum Malen gemeint – sondern bösartige, hinterhältige Apps wie Griechen, äh, Trojaner) lassen sich schwer trennen. Und da es von ersteren für Android nicht viele gibt, kümmert sich auch eine "reine Antivirus-App" wie selbstverständlich gleich mit um letztere...

Als reine Anti-Virus-App wäre hier sicher [Dr. Web Anti-virus Light](#) (Bild links) eine gute Empfehlung: Sparsam in Sachen Permissions, gratis im Markt verfügbar, beste Bewertungen.

Die gratis-Version scannt einfach auf "böse Dateien", und sperrt diese in die "Quarantäne". Hierbei scheint sowohl ein Echtzeit-Scan zu erfolgen – als auch die Möglichkeit zu einem "On-Demand-Scan" zu bestehen. Außerdem lässt sich noch einstellen, dass auch die SD-Karte bei jedem Einbinden geprüft werden soll. Die Vollversion bietet dazu auch eine Filterung eingehender Anrufe und SMS, inklusive Blacklist (z. B. für nervige Werbe-Anrufer und Spam-SMS).

Auch [Antivirus-Free](#) ist durchaus eine gute Alternative: Fast genau so gut bewertet, und absolut sauber, was die Permissions angeht, klinkt sich diese App offensichtlich in den System-Event für "App installiert" ein – und prüft sodann die neu installierte App auf "Schädlingsbefall". Eine entsprechende Notiz findet sich sodann in der "Notification Area" (Bild-Montage rechts): "Application install detected: package com.entwickler.appname is xxx". Wobei "xxx" dann entweder "clean" (sauber) oder "malicious" (schädlich) heißt.



Bei Diebstahl und Verlust



Eine App, die wirklich gegen Diebstahl und Verlust schützt, muss sicher erst noch erfunden werden. Apps in dieser Kategorie werden also i.d.R. erst dann aktiv, wenn das Kind bereits in den Brunnen gefallen ist. Nur ist es dann natürlich für eine Installation meist zu spät – darum sollte man sich also bereits im Vorfeld kümmern!

Zu empfehlen wäre hier u.a. [WatchDroid Pro](#) (Bild links) für nur anderthalb Euro, sofern eine "Stand-Alone Lösung" gewünscht ist.

Krach schlagen und SMS mit GPS-Daten verschicken geht auch bereits mit der gratis-Version, sodass man erst einmal in Ruhe testen kann. Auch diese begibt sich bereits in eine Art "Stealth Modus", sodass sie für einen "unberechtigten Abgreifer" (sprich: Dieb) nicht sofort offensichtlich erkennbar (und damit Ziel einer Löschung) ist. So richtig interessant wird es aber

erst mit der Pro-Version: Lock und Wipe stehen dann mit auf der Feature-Liste, und die App erkennt auch einen eventuellen SIM-Karten Wechsel – und verschickt in einem solchen Fall automatisch eine SMS an den hinterlegten Empfänger. Jaja, der Trend geht zum Zweit-Handy...

Worauf Apps Zugriff haben

Wer hat sich nicht schon einmal gefragt, was eigentlich bei der Installation einer neuen App aus dem Market der seltsame Hinweis sagen möchte: "Diese App darf auf folgendes zugreifen:" – gefolgt von einer teilweise recht langen Liste komischer Dinge? Nun: Der so Fragende ist hier genau richtig. Zu viele Benutzer ignorieren das nämlich einfach, ohne darüber nachzudenken. Und am Monatsende ist die Überraschung dann gelungen, wenn beim Blick auf die Mobilfunkrechnung die Frage aufkommt: "Moment – ist das jetzt der Betrag, oder die Kontonummer für die Überweisung? Wer hat denn da so viele Premium-SMS... und all die Anrufe bei 0900-*???"

Was also darf eine App? Oder, anders herum gefragt: Welche App darf denn ...? Auf beide Fragen gibt z. B. [RL Permissions](#) (Bild rechts) gute und aussagekräftige Antworten. Eine Ampel zeigt nämlich jeweils an, wie schwerwiegend der *potentielle* Schaden ist, der mit der jeweiligen Berechtigung angerichtet werden *könnte*. Was natürlich nicht heißt, dass die jeweilige App das auch tut – denn natürlich muss eine SMS-App SMS verschicken können, sonst macht sie ja nun wirklich wenig Sinn. Eine Wallpaper-App hingegen muss das nicht unbedingt.

Außerdem erklärt die App auch immer gleich, wofür die entsprechende Permission eigentlich gedacht ist. So hat man diese Information gleich im passenden Kontext. Eine kurze Übersicht mit ausgewählten Permissions sowie einer kurzen Beschreibung derselbigen findet sich übrigens auch [im Anhang](#). Und eine Liste alternativer Apps zum Thema, wie gewohnt, [im Forum](#).



APPS MACHEN DAS PHONE SMART

Wie viele gibt es da gerade? Ich schreibe hier besser gar keine Zahl – die würde ohnehin bereits nicht mehr stimmen, kaum dass ich auf "Speichern" drücke. Einigen wir uns auf "echt viele". So viele, dass man den Durchblick schnell verliert.

Wer nun meint, ich würde jetzt hier jede Menge Apps vorstellen: Weit gefehlt. Dann müsste ich wahrscheinlich im Wochen- oder doch zumindest Monatsrhythmus eine neue Version dieses eBooks veröffentlichen. Und das ist mir zu umständlich. Dem Leser wahrscheinlich auch, weil der dann immer schauen müsste, ob er auch die aktuellste Version hat – und wenn nicht, an welcher Stelle sich denn nun was geändert hat. Daher habe ich eine bessere Lösung:

Ja, genau – es ist wieder einmal AndroidPIT. Und ja, wieder das Forum. Denn hier pflege ich u.a. eine besondere Liste: [App-Reviews nach Einsatzzweck](#). Eine reine Link-Liste – wobei die Links wiederum auf Forums-Threads verweisen, in denen ich verschiedene Apps kurz vorstelle. Diese widmen sich immer jeweils einem bestimmten Thema, sodass sich mögliche Alternativen bereits im Vorfeld halbwegs vergleichen lassen. Und das sogar ohne sie vorher im Market selber suchen zu müssen.

Hinzu kommen Kommentare und Ergänzungen anderer Benutzer, Rückfragen und Antworten, sowie zusätzliche Links und Informationen. Und auch für den Fall, dass jemandem ein Themenbereich fehlt, ist gesorgt: In [diesem Thread](#) können neue Themen vorgeschlagen werden. Sobald ich dann wieder Zeit (und Lust) finde, kümmere ich mich dann darum...

Beispiele gefällig? Da wäre etwa der Bereich "Büro, Office, Verwaltung" mit Threads jeweils zu den Themen [Barcode Scanner & Generatoren](#), [Kalender](#), [Statistiken für Anrufe, SMS, MMS und Datenvolumen](#) u.a.m.; die "Lese-Ecke" mit den Themen [eBook-Reader](#), [Nachschlagen und Übersetzen](#) sowie [RSS-Reader](#). Weitere Rubriken umfassen Reise, Tools, Multimedia, Fernbedienungen... Einfach mal reinschauen!

Auszüge davon findest Du in den folgenden Kapiteln.

Telefonieren

Genau: Es ist kein Zufall, dass die Sektions-Überschrift hier heißt "machen das Phone smart" – und nicht "für das Smartphone". Eigentlich sollte der kleine Knochen ja ein Telefon mit Zusatzfunktionen sein. Auch wenn das bei einigen eher umgekehrt ist – oder sich zumindest die Waage hält. Doch definitiv sollte der Schwerpunkt sein: In Verbindung bleiben. Also das Wichtigste zuerst...

Telefon-Apps



Wie jetzt - braucht man dafür eine extra App? Nun: Genau genommen ist es ja eine App, mit der man das auf seinem Androiden von Anfang an gemacht hat. Auch wenn sie bereits vorinstalliert war, und zum Kernsystem gehört. Aber bei Android ist eigentlich alles, was man auf dem Bildschirm zu sehen bekommt (und alles, was irgendwelche Aktionen bearbeitet) eine App. Und da gibt es (fast) immer Alternativen; die Geschmäcker sind halt verschieden.

Die vorinstallierte Telefon-App hat mich von Anfang an genervt. Zum einen zu unübersichtlich: Die SIM-Kontakte ließen sich nicht ausblenden, jede Nummer wurde gefühlte zehn Mal wiederholt – für jede Aktion wie Anrufen, SMS schicken, separat. Auch wenn es recht wenig Sinn macht, eine SMS an eine Festnetz-Nummer zu schicken (vor allem, wenn zu dem Kontakt auch eine Mobilfunk-Nummer existiert). Und dann die "Fehlschüsse": Das Antippen eines Elements öffnet bei jeder anderen App die zugehörige Detailansicht. Die Telefon-App startete hingegen sofort einen Anruf. Wie oft das schon versehentlich losging – am besten mitten in der Nacht... Also musste dafür dringend eine Alternative her.

Die war zum Glück mit [Dialer One](#) (Bild links) schnell aufgetrieben. Diese App beseitigt alle vorgenannten "Pannen": Es wird übersichtlich, und sofern man nicht direkt auf das Telefon-Symbol am rechten Rand tippt, geht auch kein Telefonat los. Für die Aktionen gibt es ein Kontext-Menü. Schön auch die "automatische Suche": Beginnt man mit der Eingabe auf dem Ziffern-Feld, fängt die App im Hintergrund gleich zu suchen an: Ist das Eingegebene Bestandteil einer bereits gespeicherten Nummer? Oder entspricht es, in Buchstaben umgesetzt, dem Namen eines Kontaktes? Die gefundenen Treffer werden dann bereits während der Eingabe aufgelistet (siehe linkes Bild). Einzig zum Bearbeiten der Einträge wird auf die vorinstallierte Kontakte-App zurückgegriffen.

Für mich eine prima Lösung. Wem das noch nicht gefällt, der findet im Market auch weitere Alternativen – ähnliche, aber auch ganz andere...

Telefon-Widgets

Über *Shortcuts* und *Widgets* haben wir ja bereits im Zusammenhang mit dem [Home-Screen](#) gesprochen. Was aber sollen jetzt bitte "Telefon-Widgets" sein?

Klar, da hat Izzy wieder mal einen Begriff konstruiert. Da gibt es also Icons auf dem Home-Screen, die telefonieren können? Ja, so ungefähr. Unter [Apps organisieren](#) hatte ich ja bereits den *Folder Organizer* genannt, der die Permission zum Anrufen verlangt – genau hierfür: Es lassen sich damit nämlich auch *Shortcuts* zu Kontakten anlegen. Gibt es also Leute, die man öfter anruft, muss man deren "Kontakte" nicht erst lange im Adressbuch suchen – sondern legt gleich eine passende Verknüpfung auf dem Home-Screen ab. Auch manche Launcher (wie etwa der bereits genannte GO Launcher EX) bieten derartige Widgets an – was dann auch bei diesen die geforderte Anruf-Berechtigung erklären dürfte.

Werden die anderen Features von *Folder Organizer* nicht benötigt, und ist stattdessen eine Alternative gefragt? Der Markt hält davon etliche bereit. Viele davon heißen [Speed Dial](#) (nicht nur die verlinkte).

Die Kosten im Blick und unter Kontrolle

Ausführlicheres zu diesem Thema findet sich im Thread [Statistiken für Anrufe, SMS/MMS und Datenvolumen](#).

Alleskönner

Was – schon wieder das gesamte Kontingent an Freiminuten aufgebraucht, und der Monat ist gerade mal zur Hälfte vorbei? Oder einen Schock beim Blick auf die Rechnung bekommen, weil das Datenvolumen hoffnungslos überschritten wurde? "Das muss nicht sein!" Nein, weder Geschirrspülmittel noch Palmolive helfen hier, und Du badest auch nicht gerade Deine Hände darin. Aber mit [DroidStats](#) (Bild rechts) wäre das nicht so schnell passiert.



Wie am Screenshot gut zu erkennen, bietet hier die Übersichts-Seite nicht nur die nackten Daten feil – sondern auch eine Schätzung, wie das Ganze wohl am Ende des "Abrechnungszeitraumes" (also i.d.R. des laufenden Monats) aussieht, wenn so weitergemacht wird wie bisher. Und damit man dafür nicht immer erst die App aufmachen muss (das könnte man ja mal vergessen), lassen sich auch entsprechende Widgets auf den Home-Screens platzieren (siehe links).



Und das ist noch längst nicht alles: Will man wissen, mit wem man am meisten/längsten telefoniert bzw. SMS ausgetauscht hat, teilt *DroidStats* das ebenfalls mit. Und mehr. Kurz: Solange man keinen zu komplizierten Tarif hat, ist *DroidStats* hier die erste Wahl!

Alternativen? Doch einen recht komplexen Tarif, so mit Sonderkonditionen von..bis, wenn..dann, und was einem noch so kompliziertes einfallen könnte – und *DroidStats* in Sachen Konfiguration überfordert? Dann hilft vielleicht ein Blick auf [Call Meter 3G](#) (bzw. bei Angst vor dem Wörtchen "Beta" auf [Call Meter NG](#)). Die Konfiguration ist hier naturgemäß weit komplexer (so mancher Einsteiger dürfte da leicht überfordert sein) – aber Forum und Support funktionieren hier eben so gut wie bei *DroidStats*.

Telefonie-Spezialisten



Spezialtarife für verschiedene Netze? Etwa *Base* mit 30min E+, Festnetz-Flat und 50min in alle Netze – oder ähnliches? Dann möchte man natürlich gern vorher wissen, in welchem Netz sich die anzurufende Nummer befindet – und stets, wie viele Minuten in welches Netz bereits "verbraten" sind. Und sollte einen Blick auf [Zielnetz](#) (Bild links) werfen. Die App bietet recht ausführliche Statistiken – mittlerweile auch für SMS und Daten, sodass diese App jetzt eigentlich in das vorige Kapitel gehört. Mit dabei sind übrigens auch Widget und Warnungen – letztere etwa bei teuren Rufnummern.

Die wichtigsten Features kurz im Überblick:

Zielnetzabfrage per Rufnummer, aus dem Telefonbuch oder mit einem Klick für alle Kontakte
automatischer Abruf neuer Nummern (Info wird als Notiz im Telefonbuch gespeichert)

- akustische und optische Zielnetz-Info vor Anruf
- zahlreiche Statistiken.

Natürlich lassen sich die Informationen zu Flatrates und Minutenpaketen für die Statistiken konfigurieren.

Nicht verschweigen darf ich aber die für Zielnetz selbst anfallenden Kosten. Hier besteht die Wahl aus verschiedenen Tarifmodellen:

- OnDemand: 0.01 EUR pro Abfrage (15 Abfragen gratis)
- Flatrate: 3.99 EUR einmalig (Bezug über den Market)

Als Alternative zu *Zielnetz* wäre noch [Welches Netz](#) zu nennen. Der Funktionsumfang ist hier ähnlich (zusätzlich lassen sich hier noch Limits/Warnschwellen für Anrufe und SMS konfigurieren); laut Beschreibung im Market fallen hier jedoch keine "Abfrage-Kosten" an.

Daten-Spezialisten

Der Spezialist für die Datenverbindungen wurde ja bereits zuvor erwähnt, und heißt [3G Watchdog](#). Diese App überwacht die mobile Datenverbindung (3G/Edge/GPRS) und deren Trafficverbrauch, zeigt Benachrichtigungssymbol (Grün, Orange, Rot) in der Statuszeile und gibt eine detaillierte Übersicht zum Verbrauch. Zwei Widgets stehen auch zur Wahl.



Aber *3G Watchdog* kann noch mehr. Sicher: Es warnt vor und bei Erreichen der konfigurierten Limits – was für sich genommen schon eine gute Sache ist. So richtig interessant wird es, wenn außerdem die App [APNroid](#)

installiert ist: Kurz vor Erreichen des eingestellten Limits dreht *3G Watchdog* dann nämlich einfach den Hahn zu! Der Zugangspunkt (in der Android-Konfiguration) wird dazu von *APNroid* so verändert, dass er nicht mehr funktioniert. Und bevor jetzt Panik ausbricht: Selbstverständlich lässt sich diese Änderung rückgängig machen...

Gute Nachricht für alle Anwender (und weniger gute für das *3G-Watchdog*-Team): Android 4.0 macht diese App überflüssig. Denn hier ist die entsprechende Funktionalität bereits von Haus aus im System integriert: Eine Überwachung des Datenverbrauchs bis auf App-Ebene herab (sogar getrennt nach Verbrauch bei Vorder- und Hintergrundaktivität), mit Statistik-Graph, und einschließlich der Möglichkeit zum Festlegen eines Warn- und eines „harten“ Limits (siehe linkes Bild). Natürlich getrennt nach WLAN und mobiler Datenverbindung.



Nachrichten verschicken und empfangen

Zum "Stay-in-Touch" gehören heutzutage natürlich auch die diversen Nachrichten. Nein, nicht Twitter, Facebook & Co – ich rede von Kurznachrichten (SMS), Multi-Media Nachrichten (MMS) und "richtigen" Mails. Die erstgenannten gibt es schon seit den frühen Generationen der Mobiltelefone – aber da es mittlerweile auch die letzt genannten auf unser mobiles Allzweck-Gerät geschafft haben, sollen auch diese hier mit behandelt werden.

SMS & MMS

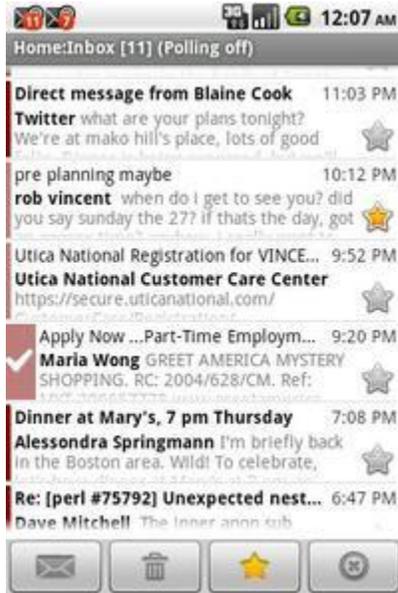
Wenn es ums "Texten" geht, scheint [Handcent SMS](#) (rechtes Bild) der absolute Favorit zu sein. Allein die Auflistung der "wichtigsten" (natürlich subjektiv) Features füllt auf der Market-Page eine ganze Bildschirmseite (ja, auch aufgrund der vielen Leerzeilen): SMS und MMS werden gleichermaßen bedient, Nachrichten können an Empfängergruppen verschickt werden. Backup und Restore von SMS und MMS. Mehrsprachig (16 Sprachen unterstützt). Mit zahlreichen Plugins erweiterbar. Die ganzen bunten Features lassen sich ja bereits problemlos anhand des Screenshots erkennen.

Damit sollte hierzu das Wichtigste gesagt sein – aber vielleicht sollte ich ja auch noch kurz die gut 4.5* bei über 300.000 (!) Bewertungen erwähnen?

Wie – trotzdem ein Alternativ-Vorschlag gewünscht? Naja, da wäre noch [chompSMS](#) zu nennen. Bei ca. 4.4* und ca. 100.000 Bewertungen kann es mit Handcent ja fast mithalten. Und was bietet diese App? Sie ist nicht ganz so "bunt", hat aber dafür Chat-Ansicht, Kontaktbilder, Quick Reply, Signaturen, Code-Sperre und auch ein Widget. Die Anwender meinen: Schaut gut aus, und ist umfangreich konfigurierbar. Besser? Schlechter? Das sind wieder sehr subjektive Entscheidungen, die jeder selbst treffen muss.



Mail



Auch die EMail ist aus unserem Alltag nicht mehr wegzudenken. Das hat man auch bei Android erkannt, und liefert eine passende App gleich mit. Welche das ist, ist aber teilweise auch noch Hersteller-spezifisch. Aber auch egal: Wer mit der bei sich installierten App eh unzufrieden ist, braucht sich um selbige ja auch nicht weiter kümmern. Halten wir also besser nach einer guten Alternative Ausschau!

Eine der bekanntesten und beliebtesten Apps hierfür ist [K-9 Mail](#) (linkes Bild). Unterstützt mehrere Accounts, auf Wunsch auch mit "gemeinsamer Inbox" (quasi als "Zusammenfassung" der einzelnen Postfach erkennt man dort an einer farblichen Markierung – wobei sich die Farben dafür natürlich wählen lassen). Als Protokolle werden sowohl POP3, IMAP4, als auch Exchange unterstützt; natürlich

lässt sich auch das Google Mail Konto einbinden. Bei IMAP wird auch "Push" unterstützt (d.h. der Mail-Server gibt der App Bescheid, wenn neue Mail da ist).

Natürlich ist dies nur ein Auszug aus dem Funktionsumfang – da könnte man noch weit mehr aufzählen. Zum Beispiel die Unterstützung für PGP (signieren/verschlüsseln von Mails), konfigurierbare Benachrichtigungen in der "Notification Area" (dem "Balken" ganz oben auf Deinem Screen, den man nach unten "aufziehen" kann) sowie per Audio, Shortcuts für den Home-Screen, Signatur-Unterstützung...

Da jetzt sicher wieder auf eine Alternative gewartet wird, habe ich natürlich auch hier etwas herausgesucht: [MailDroid](#) wäre da eine Option. Zumindest von der Beschreibung her klingen die Features vergleichbar – und auch die Market-Bewertungen scheinen das zu unterstützen. Bei Interesse also einfach mal einen Blick darauf werfen! Und ansonsten, Alternativ, einen Blick in die [passende Übersicht...](#)

Lektüre

eBook Reader

Bei den eBook-Readern ist [Moon+ Reader](#) (in der Pro-Version für ca. 3.50 Euro – es gibt auch eine [werbe-finanzierte Gratis-Version](#)) mein klarer Favorit. Sofern man auf DRM-behaftetes Material verzichten kann, kann ihm keiner das Wasser reichen:

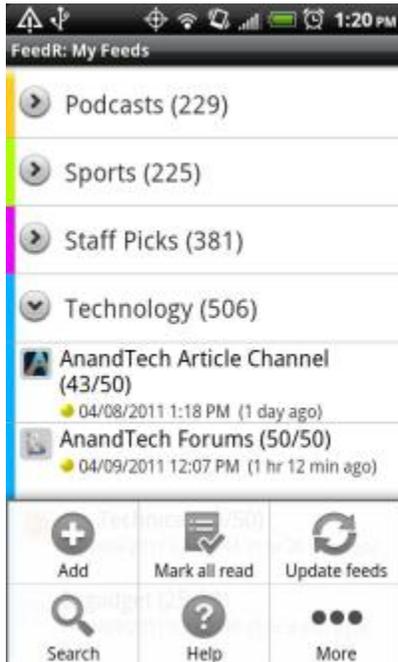
- Zugriff auf zahlreiche Online-Bibliotheken direkt aus der App (vorkonfigurierte wie z. B. [Izzys Bibliothek](#) mit über 4.000 gratis verfügbaren Büchern in deutscher Sprache, und eigene)
- Formate: txt, html, epub, umd, fb2, zip
- verschiedene Themes (u.a. "Tag" und "Nacht" – letzteres im rechten Screenshot)
- Unterstützung für Online und Offline Wörterbücher
- Highlighting, Annotations, Bookmarks, Share
- Scrolling, Vorlesen (Pro-Version)

Und damit sind nur die wichtigsten Funktionen kurz angerissen. Mehr Details gibt mein Testbericht (auf der App-Seite verlinkt). Und auch der [zugehörige Forums-Thread](#).

Alternativen? Gibt es nicht wirklich. Der [FBReader](#) ist noch recht verbreitet – unterstützt aber weniger Formate, und möchte auch noch auf Konto-Informationen zugreifen (sicher für's Einkufen von Büchern, welches er wohl unterstützt). [txtr](#) unterstützt neben ePub auch PDF – benötigt aber fünf mal soviel Platz für die Installation. Wiederum recht verbreitet und gut bedienbar ist [Aldiko](#), der auf manchen Geräten auch bereits vorinstalliert ist. Wie bereits geschrieben: Ein Blick in die Übersicht im Forum gibt mehr Informationen.



RSS Newsreader



Auch hier habe ich mit [FeedR](#) meinen Favoriten. Alle Feeds lassen sich wunderbar kategorisieren, wobei Kategorien als Ordner fungieren (wie auch im Screenshot zu sehen), und mit dem Google-Reader synchronisieren (so man es braucht). Die App lässt sich sehr umfangreich konfigurieren: So kann man z. B. festlegen, dass *FeedR* automatisch alle Stunde aktualisieren soll, sofern man in einem WLAN Netz eingebucht ist. Natürlich lassen sich auch einzelne (oder alle) Feeds jederzeit manuell aktualisieren.

Die Feed-Sammlung kanns man auch exportieren – oder die aus einem anderen Reader exportierte Einlesen, was einen Umstieg vereinfacht. Es gibt auch Sortierfunktionen für Feeds, Artikel und Ordner, sowie ein einfaches Widget.

Einziges Manko ist vielleicht, dass *FeedR* nur bis auf die Trailer-Ebene (den Vorspann vor dem Haupttext) die Inhalte selbst darstellt – für den eigentlichen Artikel wird der Browser aufgerufen.

Das jedoch wahlweise über einen "Mobilizer", damit es schneller geht. Achja: Und seit neuestem werden auch Podcasts unterstützt...

Alternativen? Ja, gibt es auch. Da wäre sicher zuerst [NewsRob](#) zu nennen. Und daneben gibt es noch eine ganze Reihe weiterer Kandidaten, von denen einige wieder in einer [Forums-Übersicht](#) aufgeführt und kurz vorgestellt sind.

Nachschlagewerke

Diese wurden ja bei den [eBook-Readern](#) schon einmal kurz erwähnt: Hier machen sie natürlich definitiv Sinn. Insbesondere bei der Lektüre fremdsprachlicher Texte. Was steht in diesem Bereich zur Verfügung?

Da wäre zunächst einmal mein Favorit: [Fora](#). Diese App eignet sich vorzüglich für nahezu alle Einsatzgebiete. Sie unterstützt zahlreiche Online-Wörterbücher, und kann auch Google Translate zum Übersetzen heranziehen. Darüber hinaus wird zur Offline-Nutzung (also ganz ohne Datenverbindung) das StarDict Format unterstützt, für das es zahlreiche Wörterbücher frei zum Downloaden gibt. Ein Blick in den [Testbericht](#) gibt weitere Details preis.

Als Alternative mit ähnlichem Funktionsumfang wäre an dieser Stelle [ColorDict](#) zu nennen. Darüber



hinaus gibt es aber noch eine ganze Reihe weiterer Kandidaten, die einen Blick wert sind – zu finden wieder einmal in einem meiner [Forums-Threads](#).

Schule & Studium

Auch Schülern und Studenten steht "Andy" hilfreich zur Seite: So passt z. B. der mit Formel- und Nachschlagewerken gefüllte Schulranzen früherer Tage heute bequem in die Jacken- oder Hosentasche.

Formelsammlungen und Übersichten

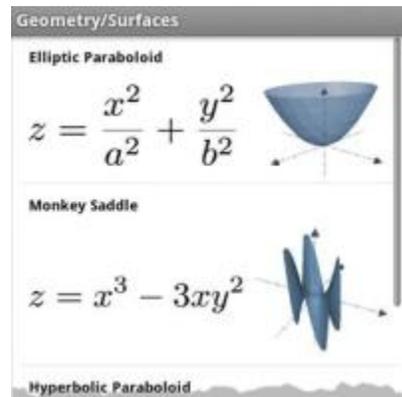


Da wäre als **das** Highlight zuerst die App [Merck PSE](#) zu nennen: Mit 4,9 von maximal möglichen 5 Punkten im Android-Market absolut topp bewertet, kann sie mit Fug und Recht hier als Vorzeige-App herhalten.

Merck's Periodensystem gibt Schülern, Chemiestudenten und Lehrenden die Möglichkeit, sich umfassend und interaktiv über die Elemente des Periodensystems zu informieren. Damit steht Interessierten ein mehrsprachiges Nachschlagewerk zur Verfügung, das komplexe Inhalte intuitiv erfahrbar macht. So wird die App im Market beschrieben.

Jede Menge Informationen stehen zu den einzelnen Elementen zur Verfügung. In verschiedensten Ansichten. So lässt sich über einen "Zeitregler" auch recht einfach feststellen, welche Elemente zum Zeitpunkt X bereits bekannt waren. Oder anzeigen, wer sie entdeckt hat. Oder, oder, oder – der Möglichkeiten sind hier viele. Für weitere Details empfiehlt sich ein Blick in den [Testbericht](#) sowie auf die [Projektseite](#) – wobei man sich bei letzterer nicht davon irritieren lassen sollte, dass hier laufend von irgend einem iPhone die Rede ist...

Und natürlich gibt es auch entsprechende Referenzen und Nachschlagewerke für andere Fächer wie etwa Physik oder Mathematik, wie z. B. [Math Ref](#) (Abbildung rechts), die für wenig Geld (etwa 0,75 Euro) eine große Menge Wissen in kompakter Form anbietet. Eine umfangreichere und detailliertere Übersicht findet sich wieder einmal im [Forum](#).



Nachschlagen und Übersetzen

Nachschlagewerke haben wir ja bereits [weiter oben](#) behandelt. In der Regel werden hier Nachschlagewerke zur Begriffserklärung und Wörterbücher zur Übersetzung innerhalb ein und derselben App behandelt – schließlich ist das Prinzip ja auch in beiden Fällen das gleiche: Einen Begriff nachschlagen, und die zugehörige Information anzeigen.

Vokabeln & FlashCards



Wer nun nicht ständig zu Nachschlagewerken und Wörterbüchern greifen möchte, muss sich die Begriffe einprägen. Und da hören wir schon die Stimme unserer "Vorfahren": *Wir haben das früher mit kleinen Zetteln in einer Streichholzschachtel gemacht!*

Ohja, das kenne ich aber auch noch – nur waren meine Streichholzschachteln aus Stabilitätsgründen schnell durch TicTac-Schachteln ersetzt. Das System ist geblieben: Damals stand auf der einen Seite des Zettels der Begriff, und die Bedeutung/Übersetzung auf der Rückseite. Heute nennt sich das "FlashCards", und statt umdrehen muss man antippen...

Die derzeit best bewertete, umfangreichste und noch dazu kostenlose App ist [AnyMemo](#) (linkes Bild). Sie bezeichnet sich selbst als *Vokabeltrainer mit dem adaptiven Lernalgorithmus* – was heißt, dass entweder die App sich dem Lernenden anpasst (was wohl gemeint ist), oder auch umgekehrt. Über 560 Datenbanken für Arabisch, Englisch, Chinesisch, Japanisch, Spanisch, Französisch uvm. stehen zur Verfügung – es lassen sich aber auch eigene erstellen: Import aus verschiedenen Formaten (u.a. auch CSV) wird unterstützt, Export ebenso (z. B. zur Datensicherung oder zur Weitergabe der eigenen Sammlung). *Keine Werbung*, verspricht der Entwickler auch bei der Gratis-Version. Dennoch: Der Preis der Pro-Version ist mit knapp 1,50 Euro mehr als gerechtfertigt – eigentlich sogar ein Schnäppchen, wenn alles so funktioniert wie beschrieben!

Wem die App nicht gefällt, oder wer sich zunächst nach Alternativen umsehen möchte – der wird, wie gewöhnlich, in einem speziellen [Forums-Thread](#) fündig.

Studentenfutter: Mensa-Pläne



Lernen macht hungrig. Also muss etwas zu essen her. Der Student lebt nicht vom Aldi allein, und auch – entgegen allen Vorurteilen – nicht vom Pizza-Bringdienst. Es gibt da so gewisse Einrichtungen, die

sich "Mensa" nennen. Und einige davon sollen tatsächlich gesunde Nahrung servieren...

Wie aber diese ausfindig machen? Zum Glück gibt es Android, und jede Menge Apps. Viele davon ortsbezogen (dafür bitte im entsprechenden [Forums-Thread](#) nachschauen). Aber es sind auch einige dabei, die eine ganz respektable Abdeckung bieten. Und so lässt sich der Speiseplan der Stamm-Mensa z. B. mit [Studentenfutter](#) als Widget direkt auf dem Homescreen platzieren (linkes Bild).

Dies beherrscht auch [MyMensa](#) (rechtes Bild), zumindest in der Pro-Version. Aber auch schon in der gratis "Lite" Version ist hier der Funktionsumfang beachtlich: Wo ist die nächste Mensa? Zumindest die nächste unterstützte sollte sich mit dieser App leicht ausfindig machen lassen. Was gibt es da? Die Frage wird definitiv beantwortet. Taugt das was? Ein Blick in die Bewertungen der Kommilitonen sollte das klären. Wie schaut das aus? Mit etwas Glück hat's schon jemand fotografiert und hochgeladen.

Die Liste unterstützter Orte und Mensen ist bereits beachtlich lang. Sollte doch noch ein Ort oder eine Mensa fehlen, so freut sich der Entwickler über eine kurze Rückmeldung. Damit die Liste noch länger werden kann.



Unterwegs

Da es hier um *Mobil*telefone geht, sind wir natürlich auch mobil. Die Warte- und Reisezeit in Bahn, Bus und Flieger haben wir uns bereits mit [Lektüre](#) verkürzt – aber wohin soll es eigentlich gehen? Und wie kommen wir dahin? Das hätten wir doch fast vergessen...

Fahrpläne



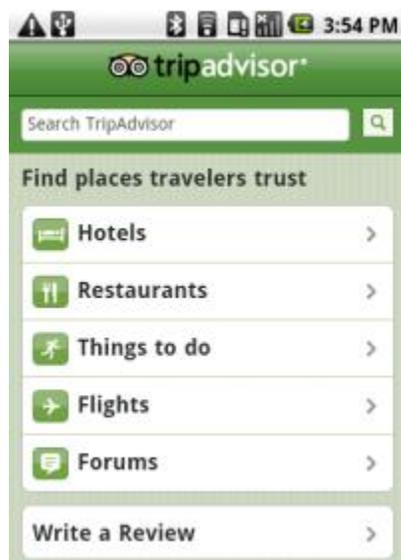
Öffi Haltestellen 1.5		
Marienburg Str.	M2	277m
M2 → S+U Alexanderplatz		in 7 min
Greifswalder Str./Danziger Str.	M10	495m
M10 → S Nordbahnhof		jetzt
Greifswalder Str./Danziger Str.	M10	495m
M10 → S+U Warschauer Str.		in 10 min
Greifswalder Str./Danziger Str.	M4	495m
M4 → Zingster Str.		in 2 min
Greifswalder Str./Danziger Str.	M4	495m
M4 → S Hackescher Markt		jetzt
Hufelandstr.	M4	422m
M4 → Zingster Str.		in 1 min
Hufelandstr.	M4	422m
M4 → S Hackescher Markt		in 1 min
M4 → S Hackescher Markt		in 13 min

Für den ÖPNV (Öffentlichen PersonenNahVerkehr) ist sicher **Öffi** (Bild links) der absolute und ungeschlagene Spitzenreiter unter den verfügbaren Apps. Nicht nur aufgrund seines Umfangs, sondern auch seiner Aktualität (ja, mehr als zwei Updates die Woche können manchmal schon ein wenig nerven). Aber wer irgendwo von A nach B möchte, liegt mit dieser App goldrichtig. Und zwar egal, ob in Berlin, München, Dresden, oder in Wien, Salzburg, Innsbruck, oder Graz, Basel... oder gar London, San Francisco, Melbourne oder Dubai (aha, daher die ständigen und vielen Updates). Öffi versorgt zielsicher mit Informationen wie nahegelegene Haltestellen (das Smartphone weiß ja, wo es ist - und dafür benötigt die App die Berechtigung für den Standort-Zugriff) inkl. Karte, den nächsten Abfahrtszeiten (inkl. etwaiger Verspätungen – hierfür und für die nächsten beiden Punkte wird der Internet-Zugriff benötigt), Verbindungen, und Netzplänen.

Warum Öffi auf die Kontakte zugreifen möchte? Damit es auch gezielt zu ihnen führen kann. Oder zu einem im Kalender eingetragenen Treffpunkt. Kalender schreiben? Klar doch, die Verbindung zum Termin. Laut Beschreibung kann es eine Verbindung auch per Mail an ausgewählte Kontakte verschicken – macht ja alles irgendwo Sinn, oder?

Alternativen? Nicht in dem Umfang von *Öffi*, aber klar gibt es sie. Unter anderem viele lokale Spezialitäten – wie etwa den [ZVV-Fahrplan](#) für Zürich und Umgebung. Wie gewohnt, finden sich in einer [Forums-Übersicht](#) wieder einmal ausführlichere Informationen.

Und was, wenn die Reise ein wenig weiter gehen soll? So von Stadt zu Stadt, wo die S-Bahn nicht mehr fährt? Dann wird z. B. zum [DB-Navigator](#) (rechtes Bild) gegriffen. Klar gibt es hier gewisse Überschneidungen: Diese App bietet Fahrpläne für DB Bahn, S-Bahn, U-Bahn und Bus von VRR, VRS, RMV, VRN, VBB, VGN, VGM, MVV und NVV, Frankreich (SNCF), Österreich (ÖBB), Schweiz (SBB) und viele weitere. Sie berücksichtigt nicht den privatisierten Nahverkehr – zeigt aber bei allen anderen Verbindungen etwaige Verspätungen auch mit an. Sogar buchen soll man mit dieser App können.



Was denn, noch nicht weit genug weg? Möchte da wer "die Fliege machen", und sucht nach Flugverbindung mit Hotel und allem, was so dazu gehört? Dem soll auch hier geholfen werden – z. B. mit dem [TripAdvisor](#) (Bild links). Wie der Screenshot schon zeigt, findet man hier nicht nur Flüge und Hotels, sondern auch gleich noch das passende Restaurant. Und vorhandene Sehenswürdigkeiten. Das Ganze auch gleich mit Bewertungen von Leuten, die schon da waren – und der Möglichkeit, selbst eine Bewertung zu hinterlassen. Ein Zugriff aufs Forum lässt sich da auch am vorletzten Menüpunkt erahnen.

Ein Kommentar weist bei dieser App noch auf einen Kniff hin: Man muss die App nicht unbedingt installieren – es gibt auch die zugehörige mobile Webseite, mit gleichem Funktionsumfang...

Und bevor ich es vergesse: Zu fliegenden Apps gibt es weitere Informationen in [diesem Forums-Thread](#).

Navigation

Mit den Öffentlichen Verkehrsmitteln kommen wir nun also klar. Wie aber sieht es mit Auto, Rad und zu Fuß aus? Damit beschäftigt sich [dieser Forums-Thread](#) – und das aktuelle Kapitel.

[Google Maps](#) ist hier sicher die bekannteste App, und auf den meisten Androiden bereits vorinstalliert. Das kleine Monster bietet eigentlich grundlegend alles, was zur Navigation benötigt wird: Kostenlose GPS-Navigation mit Sprachführung, Orte finden, Bewertungen, Empfehlungen – und bindet auch soziale Komponenten ein (Freunde auf der Karte sehen und bei Orten einchecken). Die Routenplanung eignet sich sowohl für die motorisierte als auch die unmotorisierte Fortbewegung.

Eine kleine Einschränkung könnte sein, dass man normalerweise dafür eine ständige Netzverbindung benötigt (es ist also eine sogenannte "offboard" Lösung, da nicht alles Material "onboard" ist). Aber auch das lässt sich umgehen, indem man den Karten-Cache vorher entsprechend befüllt. Google hat ja entsprechende Funktionalität bereits angekündigt – doch bis diese umgesetzt ist, kann dies zum Beispiel [Maps\(+\)](#) erledigen.

Natürlich gibt es auch hier wieder Alternativen, für die ich jedoch auf den genannten Forums-Thread verweise. Daneben gibt es aber auch noch Spezial-Lösungen, die hier zumindest kurz erwähnt werden sollten. Wie etwa [GPS Mate](#) und [OruxMaps](#) (Outdoor Navigation für Radler, Wanderer, Skifahrer, Segler und Piloten - sowie Geo-Caching), [GPS Compass Map](#) (erstellen eigener Tracks – also Routen-Erfassung mit anschließendem Nachschauen, wie man gelaufen/gefahren ist), [Ski Eagle GPS](#) (für Ski-Fahrer und Pisten-Fans), diverse Location-Sharing Apps, GPS Toolboxen, Speedometer, GPS Reminder (Wecker, die bei gewissen Koordinaten "klingeln": "Da ist die Post – jetzt gib endlich den Brief auf!" oder "Weindepot – da ist doch so ein Loch im Keller..."), und, und, und. Bei Interesse also wirklich mal einen Blick in den genannten Thread werfen.



Lokalkolorit



Was soll das denn jetzt sein? Wenn man reist, findet man viele lokale Gegebenheiten vor. Viele unterschiedliche. Touristen erkennt man hier häufig am (immer weniger um den Hals hängenden) Fotoapparat, am ständigen Knippsen, und oftmals auch am unkoordinierten Rascheln und Drehen des Stadtplans. Die Einheimischen ziehen höchstens einmal das Handy heraus – um zu schauen, wie spät es ist...

So wie der Typ da drüben. Hm, komisch: Der Kleidung nach ist der aber von ganz woanders. Scheint aber genau zu wissen, wo er hier was findet. Eben vertraut mit den lokalen Gegebenheiten. Und was hat er auf dem Handy geschaut? 10 Sekunden, und er wusste, wo die nächste Szene-Kneipe ist! Holla!

Man ahnt es schon: Da war eine App im Spiel. "Allways be a local" – mit diesem Slogan wirbt [Aloqa](#). Die verfügbaren Informationen sind in sogenannten "Kanälen" organisiert, wie etwa Theater, Restaurants, Krankenhäuser, u.a.m.. So

kann jeder die Kanäle abonnieren, die ihn interessieren – und die anderen ignorieren.

Den aktuellen Standort ermittelt *Aloqa* via GPS – wie oft die App das tun soll (angefangen von alle 5 Sekunden bis hin zu "gar nicht", also manuell – per Default immer dann, wenn man die App in den Vordergrund holt), lässt sich nach eigenen Bedürfnissen konfigurieren, damit man den Akkuverbrauch ein wenig unter Kontrolle halten kann. Pro Kanal kann man auch einstellen, ob nach neuen Inhalten gesucht – und wie man auf selbigen aufmerksam gemacht werden soll. Die Möglichkeiten umfassen hier: Nicht aktualisieren, "still und heimlich" aktualisieren, vibrieren bei neuen Inhalten, oder einen Alarmton wiedergeben.

Da die Informationen somit immer recht aktuell sind, sind es in der Tat nur wenige Klicks bis zur nächsten Szene-Bar. Oder zum McDonalds. Oder zum Schuhladen... Weltweit. Und wem das nicht reicht, der erstellt einfach [seinen eigenen Kanal](#)...

Wer's lieber spezifischer und lokaler haben möchte, kann natürlich alternativ auch zu anderen Apps greifen. Etwa zu [meinestadt.de](#)...

Shopping

Jetzt hat uns das Navi also in die Shopping-Meile geführt – und da stehen wir nun, und haben dieses tolle XYZ in den Händen. Taugt das was? Stimmt der Preis? Gibt es das vielleicht nebenan günstiger? "Ja, ja, nein", wird der Verkäufer sagen, und einen vom Pferd erzählen. Denken wir jedenfalls. Und die Chance besteht ja, immerhin will er was verkaufen. Aber müssen wir ihm deshalb blind vertrauen? Gibt es Alternativen? Auch für Android? – "Nein, ja, ja..."

Der Klassiker schlechthin in diesem Bereich nennt sich [Barcoo](#), ist gratis im Market erhältlich, und "scannt Dich glücklich". Äh – scannt? Na klar: Drehen wir doch mal das XYZ in der Hand, da ist bestimmt irgendwo ein Barcode drauf. Den scannen wir mit *Barcoo* ein – und sehen dann einen Bildschirm ähnlich dem Linken. Aha: Online also für diesen Preis. Und was es taugt, das könnten die Nutzer-Bewertungen aussagen. Gibt es Shops in der Nähe, die XYZ zu einem günstigen Preis anbieten, werden auch die angezeigt – sogar auf der Karte.

Sein volles Potential spielt Barcoo aber bei Lebensmitteln aus: Inhaltsstoffe werden hier ebenso aufgeführt wie die "Lebensmittel-Ampel", die unsere Politiker nach gründlichem Überdenken ja vielleicht 2057 einführen werden (oder auch nicht). Kurzum: Das wäre meine Empfehlung in diesem Bereich.

Dann wären da natürlich noch weitere Schnäppchen-Jäger-Apps, wie das allseits beliebte [Woabi](#) (Woanders Billiger). Oder Gutschein-Apps. Dinger, die auf Sonder-Aktionen ("Heute Friss-die-Hälfte zum doppelten Preis", oder umgekehrt?) hinweisen. Achtung – jetzt kommt der übliche Spruch: Einfach einen Blick in den passenden [Forums-Thread](#) werfen...



Gesundheit

Ernährung

Auf die "Ernährung" möchte ich hier in drei Schritten eingehen: Da wäre zuerst der Einkauf mit der Frage "Wo?", gefolgt von der Frage "Was ist drin in den Lebensmitteln?". Und schließlich die Frage: Was tun mit dem Einkauf?

Gesunder Einkauf



Was fällt einem zum Thema "gesunder Lebensmittel-Einkauf" als erstes ein? Klar: Bio. Und für Android? [Bio123](#). Wie viele derartige Apps, ist die Einsatz-Eignung regional verschieden – und hängt nicht zuletzt von der vorhandenen Datenbasis ab. Einen Versuch wert ist es jedoch allemal, und zumindest der Bereich München ist, den auf der App-Seite im Market verfügbaren Screenshots zufolge, hier recht gut abgedeckt (siehe auch linkes Bild).

Postleitzahl und Umkreis in km eingegeben, und schon kurz darauf zeigt sich eine Liste mit Fundstücken und ihrer Anzahl: Bioläden, Bistros, Bäckereien, Cafés... Die gewünschte Kategorie angetippt, und die Details werden offenbar: Wie heißt der Laden, und wie weit ist er entfernt? Jetzt den gewünschten Eintrag noch ausgewählt, und es gibt die Öffnungszeiten, Telefon, ggf. auch Website (mit der Möglichkeit, selbige im Browser zu öffnen) und EMail (kann mit der Mail-App geöffnet werden).

Und natürlich die Anschrift – mit der Möglichkeit, diese auch gleich auf der Karte anzuzeigen.

Als kleines Schmäckerl wird hier auch gleich zu Schritt 3 gesprungen: Bei Bio brauchen wir ja nicht zu schauen, was drin ist – Bio natürlich. Also geht es hier gleich direkt zu den Rezepten. Und wem das noch nicht schnell genug ist: Es stehen ja auch Bistros, Cafés und Restaurants in der Liste...

Was ist drin?

Nicht immer kann alles Bio sein. Für den Einen ist das preislich nicht drin, beim Nächsten gibt es einfach keinen Bioladen in akzeptabler Nähe, und der Dritte findet nicht alle benötigten Zutaten. Was also tun im "normalen Supermarkt"? Wie lässt sich da herausfinden, was drin ist?

Zunächst lässt sich da auf eine im Kapitel [Shopping](#) bereits genannte App zurückgreifen: Barcoo. Wie dort bereits genannt, soll sie ja hier im Bereich Lebensmittel ihr volles Potential ausspielen. Tut sie auch: Sie sagt nicht nur, wo es vielleicht günstigere Angebote gäbe – sondern zeigt die Lebensmittel-Ampel (rot-gelb-grün für viel/akzeptabel/wenig Zucker, Fett und Co), Bewertungen anderer Kunden, und oftmals auch Hintergründe und soziale Kompetenz des Herstellers.



Etwas weiter geht da der [das ist drin Scanner](#) (siehe Bild rechts), der besonders für Allergiker interessant sein dürfte: Diese App zeigt an, welche Allergieauslöser in der Packung mit drin stecken (zusätzlich zum gewünschten Lebensmittel). Und da nicht immer alle Zutaten bekannt sind, steht auch noch dabei, welche Allergieauslöser bekanntermaßen *nicht* drin stecken.

Damit verbleiben noch die kryptischen E-Zutaten, die einem immer die Haare zu Berge stehen lassen: Manche Lebensmittel scheinen ja fast ausschließlich aus solchen zu bestehen! Dank sei der chemischen Industrie: Im Zeitalter von Rinderwahn, Schweinepest, Vogelgrippe, Atomfisch und EHEC-Gemüse wüssten wir ja ohne sie gar nicht mehr, was wir überhaupt noch essen könnten... Achso, für die E-Nummern hat der *das ist drin Scanner* auch ein Register integriert, das "Inhaltsstoffe-Lexikon".

Rezepte

Bio-Rezepte hatten wir ja bereits beim gesunden Einkauf als Dreingabe. Für "normale Rezepte" gibt es sicher auch viele Apps, die noch mehr Megabyte im Speicher belegen, und zum Teil auch seltsame [Berechtigungen](#) verlangen. Daher erlaube ich mir an dieser Stelle einmal einen kleinen Kunstgriff:

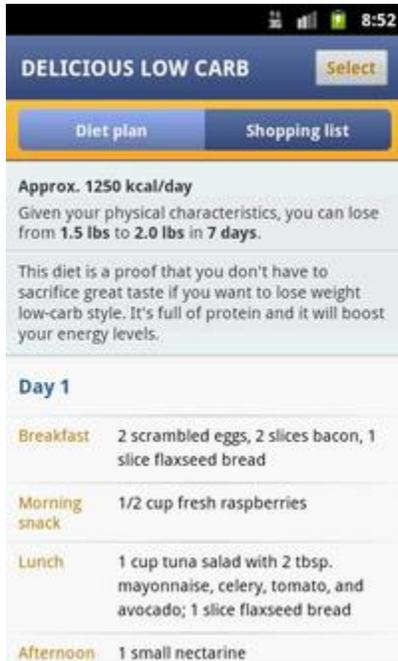
Wir greifen zum mobilen Browser, und legen hier ein Lesezeichen für <http://m.essen-und-trinken.de/> an. Und schon haben wir eine recht umfangreiche Rezeptsammlung zur Hand!

Wer sich doch lieber eine App installieren möchte, wird wieder einmal im [Forum](#) fündig...

Abnehmen: Weg mit den Pfunden!

Die Hose geht nicht mehr zu? Der Gürtel ist zu kurz? Oder muss gar schon eine Schubkarre her, um den Bierbauch zu transportieren? Höchste Zeit, den Pfunden den Kampf anzusagen! Und welche Apps unterstützen uns dabei? Die "große Übersicht" findet sich wieder einmal im [Forum](#) – die "Kompakt-Ausgabe" gibt es hier:

Diät



Keine Lust auf Sport? Vielleicht tut es ja auch eine reine Diät. Dabei kann [DietPoint](#) (Bild links) helfen:

Die App ist zwar (noch) nicht komplett "eingedeutscht" – doch die deutschen Bewertungen im Market lassen darauf schließen, dass sie auch hierzulande gut verwendbar ist. Hier lassen sich Diät-Pläne verwalten, und auch gleich in einer Tag-für-Tag Diät aufreihen, eine passende Einkaufliste lässt sich ebenfalls erstellen. Der zu erwartende Gewichtsverlust wird anhand der hinterlegten (und stets gepflegten) Daten berechnet. Mit dabei sind auch BMI sowie BMR Rechner, Tipps und Ratschläge – und die direkte Einbindung eines Forums zum Austausch mit anderen interessierten.

Alarmer weisen auf bevorstehende Mahlzeiten hin und fördern so eine dem Abnehm-Prozess und der Gesundheit förderliche Regelmäßigkeit. Da neben dem imperialen Maßsystem auch das metrische unterstützt wird, sollten Komplikationen in dieser Hinsicht minimiert sein. Kostenlos ist das Ganze obendrein – was gibt es also zu verlieren, außer den Pfunden?

Sport

Diät ist furchtbar? Sicher, aber vielleicht ja notwendig. Alternativen? Okay, die gibt es natürlich auch: (Mehr) Sport treiben!

Wer das für eine prima Sache hält, kennt wahrscheinlich bereits die App [CardioTrainer](#), und setzt sie auch bereits fleißig ein. Da kommt mein Hinweis vielleicht wie gerufen, dass es dafür ein Plugin namens [Abnehmen mit CardioTrainer](#) gibt.

Während CardioTrainer für das Training gedacht ist, hängt sich dieses Plugin dort direkt ein, um die verbrauchten Kalorien zu zählen. Zuerst wird das Ziel der Aktion festgelegt (wieviel man in welchem Zeitraum abnehmen möchte – realistische Werte, bitte 😊), und die geplanten Aktionen (welchen Sport, wie oft, ggf. weniger Kalorien aufnehmen?). Dann überwacht CardioTrainer die Ausführung – und zeigt schließlich die Resultate an.

Wer *CardioTrainer* noch nicht benutzt, sollte allerdings einen Blick auf zweierlei werfen: Zum Einen ist die App mit ca. 7MB nicht unbedingt klein, was bei manchem Gerät bereits eine große Hürde darstellen könnte. Zum Anderen müssen auch die von der App verlangten [Berechtigungen](#) abgewogen werden: Die Verbindung "Kontaktdaten lesen" im Zusammenhang mit "uneingeschränktem Internetzugriff" würde sicher nicht nur mir Bauchschmerzen bereiten! Der Entwickler schreibt zwar zur Erklärung: *Die Einträge aus dem Adressbuch werden ausschließlich dazu verwendet, um mit anderen CardioTrainer-Benutzern in Verbindung zu treten.* Aber vielleicht zieht er, wenn es oft genug gefordert wird, auch die Auslagerung dieser Funktionalität in ein Plugin in Erwägung...



BMI & Protokoll



Und dann waren da noch diejenigen, die das Thema auch so im Griff haben. Ohne Diät-App, ohne Sport-App. Aber es tut dennoch gut, den Erfolg "schwarz auf weiß" verfolgen zu können, oder? Und auch für diesen Fall ist mit Android gesorgt:

So wäre da zum Beispiel die App [Droid Weight](#). Nicht vom englischen Namen irritieren lassen: Es ist eine deutschsprachige App aus deutschen Landen. Sie speichert Gewicht und BMI, und stellt ersteres als Graph über bis zu 6 Monaten dar. Darüber hinaus lässt sich das Zielgewicht hinterlegen. Man kann sich sogar daran erinnern lassen, regelmäßig die Werte einzugeben. Dabei versteht die App sowohl metrische als auch imperiale Maße. Die Datenbank lässt sich auf die SD-Karte ex- und auch von dort wieder importieren. Ebenso kann sie komplett

zurückgesetzt (geleert) werden. Kurzum: Alles, was für Protokoll und Statistik nötig ist, hat *Droid Weight* mit an Bord.

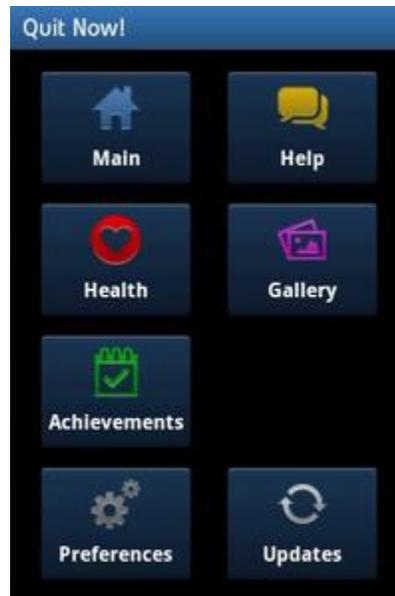
Rauchentwöhnung

Die große Übersicht findet sich, wie gewohnt, wieder im [Forum](#) – was besagt, dass Android auch hier wieder eine ganze Reihe von Alternativen anbietet.

Aus dieser Reihe möchte ich hier [QuitNow!](#) herausgreifen – die in diesem Bereich mit Abstand am Besten bewertete App.

Laut Kommentaren im Market ist die App gut lokalisiert – auch wenn Name und Screenshot in Englisch sind, sie ist ebenso des Deutschen mächtig. *QuitNow!* bietet einige Statistiken – so zum Beispiel rauchfreie Tage und gesparte Zigaretten (letztere auch in Bares umgerechnet). Während des gesamten Prozesses können die Auswirkungen auf insgesamt neun gesundheitliche Aspekte beobachtet werden – es wird ja nicht einfach ein Schalter umgelegt; der Körper muss die ganzen Gifte erst nach und nach abbauen, und sich entsprechend regenerieren. Es lässt sich also verfolgen, wie er sich langsam erholt: Etwa das nach 48 rauchfreien Stunden so einige Geschmacks-Sensoren wieder erwachen...

Auch ein Widget ist mit dabei. So hat man den Erfolgsstand auch dann vor Augen, wenn die App gerade nicht im Vordergrund läuft.





Wie: Das war jetzt nicht so ganz ernst gemeint mit dem Aufhören? Geselligkeit und so? Macht nix, dann tauschen wir die App einfach aus, und die Beschreibung ebenso:

Mit der [Mitrauchzentrale](#) lassen sich raucherfreundliche Lokalitäten aufspüren. Nicht etwa nur Kneipen! Die Liste umfasst Clubs, Bars, Cafés, Restaurants, öffentliche Plätze, Privat, Parks, Raucherecken und Sisha-Lounges.

Mit einem Login gibt es darüber hinaus auch diverse Community-Features: Wie viele Raucher sind gerade an der Location angemeldet? wie ist selbige bewertet? Selbst eine Bewertung abgeben ist natürlich ebenfalls möglich. Wo es "Raucherecken" gibt, zeigt die App auf der Karte an.

Für weitere Details lohnt sich auch ein Besuch des [Headquarters](#)...

Arzt und Apotheke

Arztsuche

Früher oder später trifft es jeden: Ein Arzt wird gebraucht. Entweder die Suche eines neuen Hausarztes nach dem Umzug, oder die Suche nach einem Spezialisten, den man sonst noch nie benötigt hatte. Klar: Die "Gelben Seiten" und auch andere Telefonbücher kennen Ärzte zuhauf. Doch man will ja schließlich nicht den erstbesten, sondern den ersten *und* besten: Gut soll er (oder sie) sein, natürlich vom Fach "etwas" verstehen – aber auch die "soziale Kompetenz" darf nicht zu kurz kommen. Alles Dinge, die uns das Telefonbuch nicht verrät.

Aber die [jameda Arztsuche](#) weiß an dieser Stelle weiter. Denn diese besteht nicht nur aus einem Telefonbuch mit Nummern – sondern auch aus einer Community, die ihre Bewertungen hinterlassen hat. Die folgen dem Schulnoten-Prinzip, wobei verschiedene Kriterien (u.a. Zufriedenheit, Vertrauensverhältnis, wurde sich Zeit genommen – oder war es eher eine "Massenabfertigung") separat ausgewiesen werden. Ein persönlicher Kommentar sagt schließlich etwas über die Dinge, die sich nicht in Zahlen fassen lassen.



Arzt, Tierarzt, Hebamme, Apotheke, Klinik, Augenoptiker oder auch Krankenkasse: Über 460.000 Adressen bundesweit sind in der Datenbank enthalten. Und Dank Umkreissuche findet man im Ernstfall auch den nächstgelegenen Arzt. Mit gerade einmal 300kB Größe eine App, die auf jedes Handy passt.

Apotheken



[Apotheken](#) ist die einzige offizielle Anwendung im Auftrag der deutschen Apothekerschaft für ganz Deutschland. Und als solche darf man sich ja wohl auf die von ihr gelieferten Informationen getrost verlassen.

Wenn man eine Apotheke sucht, dann in der Regel keine, die gerade geschlossen hat. Macht ja keinen Sinn. Also eine offene – Sonntag früh um ein Uhr dreißig. Ja und? Kein Problem: Notfall-Apotheken haben auch an Sonn- und Feiertagen geöffnet. Und sind auch nachts bereit. Und *Apotheken* kennt sie natürlich, als offizielle App der Apothekerschaft. Und weiß auch, welche gerade Notdienst hat.

Also alles kein Thema: Schon nach wenigen Klicks ist die richtige Apotheke gefunden, und kann bei Bedarf auch telefonisch kontaktiert werden (na, hat sie wirklich geöffnet? Ist das gewünschte Medikament da – oder sucht man besser die übernächste Apotheke?). Auch eine Anzeige auf der

Karte ist natürlich möglich. Inklusive Routenfunktion – wer jetzt immer noch nicht hingefunden hat, ruft besser ein Taxi...

Stopp einmal kurz: Und im Ausland? Gibt es da auch die "Deutsche Apothekerschaft" mit ihrer offiziellen App? Das vielleicht nicht, aber es gibt ja noch mehr Apps. Für diesen Fall wäre zum Beispiel [Apotheken-Sucher](#) einen Blick wert...

Medikamente

Fast jeder hat seine "Stamm-Medikamente", und sei es für die Reise-Apotheke: Aspirin für den Brummschädel, Iberogast für den Rumpel-Bauch, Voltaren für Verzerrungen & Co... Alles Sachen "für den Fall der Fälle", also nichts zeitkritisches. Oder Sachen, die man regelmäßig einnehmen muss: Im Moment noch genügend vorhanden, aber irgendwann braucht man wieder Nachschub.

Wenn es nicht akut ist, hat man Zeit zum Suchen nach dem besten Angebot. Und bei Dingen, die man immer wieder kauft, macht auch Kleinvieh mit der Zeit gehörig Mist. Und *wann* hat man die Zeit, so eine Suche durchzuführen? Genau: Wenn man ohnehin gerade zum Nichts-Tun verdonnert ist. Eine Stunde in der S-Bahn bietet sich da an – und die passende App auf dem Androiden:

MediPreis zum Beispiel. Wer das gesuchte Medikament gerade zur Hand hat, hält jetzt die Kamera seines Androiden auf den **Barcode**, bis es "Piep!" macht. Alle anderen geben brav den Namen in die Suchmaske ein. Und wenig später erscheint, eine Datenverbindung natürlich vorausgesetzt, eine Ergebnisliste – wie im rechten Bild zu sehen.

Keine Lust auf eine extra App? Oh... Dann tut es vielleicht auch ein Lesezeichen im mobilen Browser, welches auf handy.medipreis.de zeigt.



Notfall

110 & Co



Verhalten im Notfall

1. Schützen Sie sich selbst
2. Erste Hilfe leisten
3. Unfallstelle absichern
4. Hilfe rufen

Wichtige Fragen

1. **Wo** ist etwas geschehen?
2. **Was** ist geschehen?
3. **Wie** viele Verletzte?
4. **Welche** Verletzungen?
5. **Warten** auf Rückfragen!

Klar: Wem im Fall des Falles die Nummer 110 nicht mehr einfällt, der denkt auch nicht an eine auf dem Androiden installierte App. Doch kaum hat man die Nummer in der Hektik des Gefechts gewählt, geht das Stottern los: Wie sag ich's am Besten? Und was überhaupt? Welche Details sind wichtig?

Hier souffliert die [Mobile Notruf-App für Notfälle](#) (so der volle Name) mit den richtigen Stichworten – wie im linken Bild zu sehen. Frage 1 auch beantworten, selbst wenn es obsolet scheint: Natürlich hat das Smartphone im Hintergrund bereits die aktuelle Position per GPS ermittelt. Doch woher soll der CallCenter-Mitarbeiter am andern Ende der Leitung wissen, ob man selbst direkt am Ort des Geschehens ist – oder den Anruf aus "sicherer Entfernung" tätigt? "Ich sitze hier auf einer Bombe" ist wohl eher unwahrscheinlich...

Achja: Und dann wären da noch die Notruf-Nummern, die nicht jeder im Hinterkopf hat: Gift-Notruf? Frauenhaus? Oder, bei seelischen Notfällen: Telefon-Seelsorge? Die App kennt auch diese.

Mammi, ich muss mal!

Auch das ist ein "medizinischer Notfall" – gewissermaßen. Denn wenn jetzt nicht schnellstens reagiert wird, platzt am Ende die Blase. Oder das Kind wird von Mitte bis Unten ziemlich nass, und holt sich dadurch eine Erkältung. Beides nicht wirklich wünschenswert, oder?

Aber was tun – mitten in unbekanntem Terrain?

Zum Glück gibt es auch hier wieder eine tolle Android-App: [GoToilet](#) findet die passenden Örtchen. Und zwar weltweit! Sowohl die öffentlichen, als auch die von Cafés, Restaurants, oder Tankstellen (bei letzteren muss zur Rechtfertigung der hier getätigten Notdurfts-Verrichtung gegebenenfalls auch noch eine andere, kostenpflichtige Dienstleistung in Anspruch genommen werden).

Der Funktionsumfang beinhaltet nicht nur eine stumpfe Auflistung verfügbarer Notdurfts-Stätten (obwohl auch das, inklusive der Entfernung dorthin, Bestandteil ist). Auch auf der Karte können sie eingeblendet werden (siehe Bild rechts). Sofern Bildmaterial dazu bei Streetview vorhanden ist, lässt sich die



Umgebung des Wunsch-Ortes auch auf diese Weise vorab inspizieren. Mit etwas Glück gibt es sogar eine Bewertung – so dass unappetitliche Plätze gemieden werden können...

Büro, Office & Verwaltung

"Früher" sprach man von Büro-Gebäuden. Heute ist das Büro da, wo man gerade ist. Dummerweise auch nach Feierabend. Schauen wir uns also mal die Ausstattungs-Möglichkeiten an:

Barcodes



Barcodes sind aus unserem täglichen Leben ja nicht mehr wegzudenken – überall kleben die verschiedensten Fassungen drauf. Beim Thema [Shopping](#) sind wir ihnen ja bereits begegnet, und können jenen Teil (Produkt-Informationen) daher hier überspringen.

Am bekanntesten ist hier sicher der [Barcode-Scanner](#) (Bild links), den ja eigentlich jeder zweite auf seinem Androiden hat. Der erkennt z. B. die sogenannten QR-Codes, wie sie auch bei AndroidPIT benutzt werden: Draufhalten – "Piep" – und ab in den Market, auf die Seite der App, die man gerade "angepiept" hat. QR-Codes können verschiedenste Informationen enthalten: URLs (wie eben beschrieben), Adressen, kurze Texte, Termine (wäre nett, wenn die auf diversen Theater- und Kino-Plakaten mal Standard würden). Ein guter Reader öffnet dann jeweils die richtige App: Adressen lassen

sich so gleich der Kontaktliste zufügen, Termine in den Kalender eintragen, und so weiter. Praktische Sache das. Und die genannte App beherrscht das auch weitgehend.

Natürlich gibt es gerade in diesem Bereich eine ganze Reihe von weiteren Apps, wie z. B. [ixMAT](#), der besonders viele Formate kennen soll, [i-nigma](#), oder [lynkee](#). Für Details verweise ich daher wieder auf den [zugehörigen Forums-Thread](#).

Dort finden sich allerdings nicht nur die gerade beschriebenen Reader – sondern z. B. auch die Generatoren, mit denen man eigene Barcodes erstellen kann. Oder auch Apps wie [Shelves](#), mit denen sich ein eigenes Inventory aufbauen lässt (also quasi eine Artikelverwaltung gleich mit dabei). Aber auch Buch-Manager gibt es (zur Verwaltung der eigenen Bibliothek), oder Buch-Infos (zur Nutzung im Buchladen: Taugt das was? Referenzen?)...

Finanzen

Auch einen Überblick über sein Budget kann man mit Hilfe Deines Androiden behalten. Gerade in diesem Bereich stehen zahlreiche Apps zur Verfügung (für eine Übersicht siehe [diesen Forums-Thread](#)). Die Möglichkeiten reichen von der einfachen Erfassung von Ausgaben, über eine komplexe Budget-Verwaltung mit

Export und Reporting, bis hin zu Online-Diensten, die alles aufbereiten, was man ihnen per Foto schickt (hochlädt).

In den vorangegangenen Kapiteln klang ja bereits mehrfach an, dass ich eher konservativ bin, was persönliche Daten auf dem Smartphone allgemein und speziell auf fremden Servern und in der Cloud betrifft. Das gilt natürlich insbesondere für so sensible Dinge wie Finanzdaten. Daher sei an dieser Stelle nochmals auf eines deutlich hingewiesen: Das Risiko eines Missbrauchs ist in diesem Falle größer, als wenn sich selbige Daten lediglich auf dem stationären Rechner daheim (oder gar nur in Papierform im verschlossenen Schrank) befinden. Der Anwender muss daher für sich selbst entscheiden, ob er derart sensible Daten überhaupt auf seinem mobilen Gerät haben, oder gar der Cloud (und somit fremden Rechnern – einige Apps ermöglichen oder erfordern gar den Upload der entsprechenden Dokumente) anvertrauen möchte.

Eine recht gute Wahl bei den etwas umfangreicheren Apps scheint [CWMoney](#) (Bild rechts) zu sein. Hier handelt es sich um die in diesem Bereich am besten bewertete App. Laut Beschreibung werden mehrere Accounts sowie verschiedene Währungen unterstützt. Einträge können mit GPS-Stamps, Fotos oder auch Sprachaufnahmen angereichert werden. Es gibt hierarchische Strukturen, Kataloge, Filter, Reports, Torten-Graphen, und mehr. U.a. auch einen Daten-Export als XML bzw. CSV.



Allerdings hat das Ganze auch seinen Preis: Ca. 7 Euro werden für die Vollversion fällig. Wer's lieber gratis haben möchte, greift dann z. B. zu [Financisto](#). Ist fast genau so gut bewertet, aber vollständig Open Source.

Ist es interessanter, wie es praktisch auf dem Bank-Konto aussieht? Natürlich geht auch so "richtiges Homebanking" unter Android (die passende Übersicht im Forum findet sich [hier](#)). In Sachen Komfort und Sicherheit sind hier ganz klar die Produkte von [StarMoney](#) (die namensgleiche App ist links abgebildet) zu empfehlen. Je nachdem, ob es nur um ein Konto bei einer Sparkasse, oder mehrere Konten bei verschiedenen Banken geht, steht eine passende App gratis oder für ein bis vier Euro bereit.

Auch einige andere Banken stellen ihre eigene App bereit – Details dazu unter genanntem Link. Übrigens auch zum Thema "Börse" – sei es jetzt nur die Markt-Beobachtung, oder auch das "Broken" selbst...

Kalender

Oh, hier scheiden sich die Geister. "Die beste App" gibt es in dieser Kategorie nicht. Je nach Vorlieben und Bedürfnissen, gibt es hier immer mindestens zwei Kandidaten. Da fällt mir die Auswahl nicht leicht...

Also greife ich zuerst einmal den [Business Calendar](#) heraus. Der Name scheint ja bereits anzudeuten, wofür sich dieser besonders gut eignet. Unterstützt werden von der App sowohl der Google Kalender, Exchange, PC-Sync, und Facebook-Kalender – es besteht also die freie Wahl, wie öffentlich man gern sein möchte.

Die App lässt sich sehr angenehm bedienen. In der Übersicht (siehe Screenshot) lässt sich mit einem Slider (unten im Bild) frei einstellen, welchen Zeitraum man gern sehen möchte. Oder man nutzt die "Zwei-Finger-Geste" (auch als "Pitch-to-Zoom" bekannt), um den Zeitraum anzupassen. Kontext-sensitive Hilfe ist auch mit dabei.



Hm, alle Details können in der Monats-Ansicht sicher nicht angezeigt werden. Aber auch hier ist der Business Calendar clever: Termin antippen, und die Details erscheinen in einem extra Layer. Schön übersichtlich.

Als zweiten Kandidaten muss ich wohl zwangsläufig [Jorte](#) nennen – sonst bekomme ich Haue von der Community 😊. Also – linkes Bild, bitte!

Auch Jorte unterstützt sowohl Google, Exchange als auch PC-Sync Kalender. Wie bei der Größe (etwa 3MB Download) zu erwarten, kann man sich hier wirklich richtig austoben – alles mögliche lässt sich anpassen. Und da Jorte wirklich eine Unmenge an Widgets mitbringt, lässt sich die Widget-Auswahl des Homescreens ein wenig "handlicher" halten, indem man hier unerwünschtes einfach deaktiviert.

Eine wirklich gute Sache ist jedoch, dass Jorte die lokalen Feiertage importieren kann! Man muss in der Länderliste allerdings ein wenig suchen – das Sortierkriterium hat sich mir nicht erschlossen, sah eher etwas wild gewürfelt aus. Etwas kariert geschaut habe ich auch, dass es offensichtlich ein Land namens "Krawatte" gibt (da ist doch wohl nicht etwa Tie-Land gemeint?)... Leider sind die so importierten Feiertage offensichtlich nur in Jorte selbst zu sehen – oder ich habe etwas verpasst...

Für weitere Kandidaten sei wiederum auf den [entsprechenden Forums-Thread](#) verwiesen.

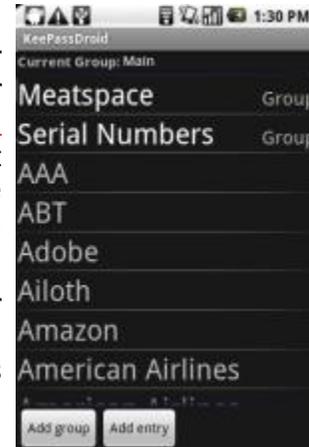
Und vielleicht gleich noch auf [einen weiteren](#), bei dem es um die Synchronisation der Kalenderdaten geht. Was? Achso, ja klar, geht auch über Google. Aber nicht jeder möchte seine privaten Daten auf fremde Server schicken. Und deshalb gibt es Apps wie [Caldav Sync](#) (sofern man einen passenden eigenen Webserver hat – bei Firmen ist das oftmals der Fall), [SyncEvolution](#) (zur Synchronisierung mit Evolution unter Linux), und andere.

Passwörter

Passwörter sollen möglichst sicher gespeichert werden. Apps dazu gibt es ja scheinbar wie Sand am Meer – sicher sind diese aber nicht unbedingt. Daher sollte bei der Auswahl unbedingt ein Blick in den [entsprechenden Forums-Thread](#) geworfen werden! Dort werden zwar nicht alle verfügbaren Apps ausführlich vorgestellt, doch die Liste im ersten Post hilft schon einmal, die "unsicheren Kandidaten" zu eliminieren.

Nach guter Tradition soll aber zumindest eine App hier kurz erwähnt werden. Sie nennt sich in diesem Fall [KeePassDroid](#). Sieht zugegeben etwas spartanisch aus (siehe Bild rechts), ist aber sicher. Und es gibt eine PC-Version, mit der die App sogar kompatibel ist. Vorausgesetzt, man führt nicht zeitgleich in beiden Installationen Änderungen durch, hat man seine Passwörter somit an beiden Stellen parat. Gut verschlüsselt, versteht sich: Sollte ein "Fremder" das Smartphone (oder auch nur die Passwort-Datei) in die Finger bekommen, beißt er sich beim "Knacken" die Zähne aus.

Zu den anderen Kandidaten (und es sind einige) kann ich nicht viel sagen – ich kann mir ja nicht alle angucken 😊

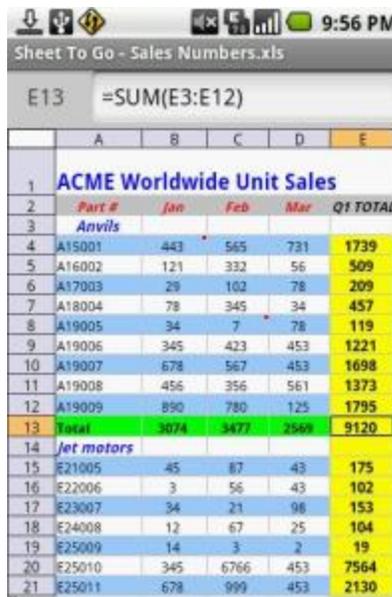


Office-Pakete

Mobiles Office? Kein Ding. Eine passende Übersicht gibt es in [diesem Forumsthread](#). Ein paar Stichproben natürlich wieder hier.

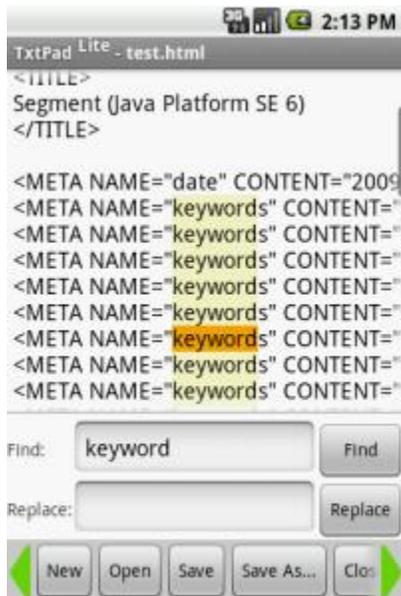
Allerdings gibt es eine traurige Nachricht gleich vorab: Ich konnte kein Office-Paket finden, das auch freie Formate (wie das OpenDocument Format, welches u.a. bei OpenOffice/LibreOffice zum Einsatz kommt) unterstützt. Alle sind voll und ganz auf Microsoft konzentriert...

Als prominenteste App wäre hier [Documents To Go](#) (rechtes Bild) zu nennen. Wie auch [Office Suite Pro](#) unterstützt die App Word, Excel und Powerpoint in den gängigen Versionen: Dokumente können geöffnet oder neu erstellt, bearbeitet, und natürlich wieder gespeichert werden. Darüber hinaus ist auch ein PDF-Viewer mit an Bord. Jeweils ungefähr 10 Euro kostet der Spaß – wobei es bei den kleineren Displays eher weniger Spaß machen wird. Aber dafür kann die jeweilige App nichts. Auf Tablets sieht es schon ein wenig anders aus...



The screenshot shows a spreadsheet titled "Sheet To Go - Sales Numbers.xls" with the formula bar displaying "=SUM(E3:E12)". The table below is titled "ACME Worldwide Unit Sales" and contains the following data:

Part #	Jan	Feb	Mar	Q1 TOTAL
Anvils				
A15001	443	565	731	1739
A16002	121	332	56	509
A17003	29	102	78	209
A18004	78	345	34	457
A19005	34	7	78	119
A19006	345	423	453	1221
A19007	678	567	453	1698
A19008	456	356	561	1373
A19009	890	780	125	1795
Total	3074	3477	2968	9120
Jet motors				
E21005	45	87	43	175
E22006	3	56	43	102
E23007	34	21	98	153
E24008	12	67	25	104
E25009	14	3	2	19
E25010	345	6766	453	7564
E25011	678	999	453	2130



Einziger Kandidat mit Unterstützung für ein offenes Format ist hier [Androffice](#). Wer sich nicht vom Wort "Office" irritieren lässt, findet in dieser App für ca. 7 Euro eine Tabellenkalkulation, die sowohl Excel als auch OpenDocument Spreadsheet unterstützt.

Für Spartaner und Entwickler sind sicher auch noch reine Text-Editoren, wie [TxtPad](#) (Bild links) interessant: Hiermit lassen sich reine ASCII-Texte bearbeiten. Das spart nebenbei auch Ressourcen, denn sowohl App als auch Dokumente benötigen weniger Platz.

Nicht verschweigen möchte ich an dieser Stelle die ganz speziellen Notizen-Apps, von denen ich besonders [Note Everything](#) hervorheben möchte.

Der Name ist wörtlich zu nehmen. Der Name ist Programm. Hiermit kann man wirklich alles notieren: Natürlich Textnotizen: Schnell ein paar Stichpunkte zum Vortrag, eh es wieder vergessen ist. Oder mal was skizzieren - kein Thema. Auch eine Foto-Notiz stellt Note Everything nicht vor ein Problem. Zu faul zum Schreiben? Dann diktier doch einfach was. Und schreib am Ende doch was dazu. Auch Widgets sind mit von der Partie: Für die schnelle Notiz zwischendurch...

Wem das nicht reicht, der greift für ca. 3 Euro zur Pro-Variante – und erhält zusätzlich Abhak-Listen, Video-Notizen, Foto-Notizen (oops), und mehr.

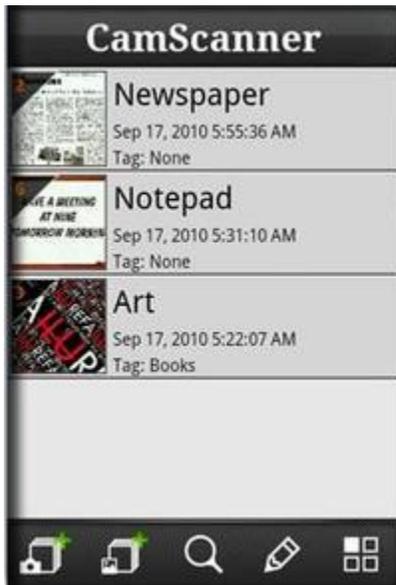
Natürlich gibt es auch hier wieder eine Reihe von Alternativen, wie z. B. [ColorNote](#), das auch bunte "Stickies" auf den Home-Screens ablegen kann. Einfach mal in den Forums-Thread schauen.

PDF-Dateien Anzeigen und Erstellen

PDF (das "portierbare Dokumenten-Format" oder, im O-Ton "portable document format") wurde 1993 von Adobe auf den Markt gebracht – und stellt seitdem einen gewissen Standard dar. Die Besonderheit dieses Dokumentenformates ist, dass der Inhalt überall gleich dargestellt werden soll: Ob unter Windows, Mac, oder Linux, auf dem Bildschirm oder gedruckt... Auf dem Androiden? Die Frage ist sicher nicht unberechtigt! Also habe ich ein wenig im Market gestöbert, ob ich auch zu diesem Thema passende Apps nennen kann.

Und: "Yes, we can!". Der unvermeidliche **Forums-Thread** gibt Dir eine ganze Liste an die Hand – gegliedert nach einfachen Betrachtern, Generatoren, und Tools. Klar sind da auch wieder die "Original-Produkte" von Adobe vertreten, auch wenn sie nicht unbedingt die Apps sind, die am besten abschneiden...

Bei den Betrachtern ist das vielmehr der **ezPDF Reader** (Bild rechts). Kostet zwar einen knappen Euro – ist aber nicht nur topp bewertet, sondern kann auch eine ganze Menge: Nicht nur das Anzeigen einer PDF-Datei. Das setze ich mal bei einem PDF-Reader als gegeben voraus. Mit dieser App lassen sich jedoch auch Markierungen anbringen, Texte unter- oder durchstreichen, Bereiche umrahmen (Rechteck oder auch Kreis/Ellipse), Notizen oder gar Freihand-Zeichnungen einfügen. Da bleibt eigentlich nichts mehr übrig, oder?



Wie jetzt – PDF-Dateien erstellen? Kommt ganz auf die Quelle an. Relativ viele Formate unterstützt hier natürlich das "Original", **Adobe CreatePDF**. Eine mit 250kB relativ kleine App – wie schafft die den Umgang mit MS Word (docx, doc), Excel (xlsx, xls), PowerPoint (pptx, ppt), Adobe Illustrator (ai), Photoshop (psd), InDesign (indd), Images – JPEG, BMP, PNG, GIF, TIFF, RTF, Text und WordPerfect, OpenOffice und StarOffice Dokumenten? Das kann ja unmöglich alles in der App stecken, auch wenn die mit über 7 Euro nicht gerade günstig ist. Die Antwort ist ganz einfach: Die App ist lediglich die "Benutzer-Oberfläche". Die Dokumente landen dann alle auf einem Adobe-Server: Dort findet sodann die Umwandlung statt. Wie, private/vertrauliche Dokumente auf fremde Server laden ist nicht drin? Das ist dumm, denn eine Alternative gibt es leider nicht...

Ganz interessant sind in diesem Zusammenhang aber auch "Hosentaschen-Kopierer" wie der im linken Bild dargestellte **CamScanner Phone PDF Creator**. Nein, diese Apps kopieren keine Hosentaschen – auf dem Smartphone installiert,

passen sie aber bequem in selbige. Und man hat sie quasi immer zur Hand. Eben mal schnell ein paar Seiten aus einem Buch in der Bibliothek, die man zu Hause nochmal genauer anschauen möchte? Kein Thema. Schnell gemacht – und auf der SD-Karte gespeichert, zu Google Docs hochgeladen, oder per Mail verschickt. Ist mit gut 2MB ein wenig größer als zuvor genannte App, arbeitet dafür aber auch mehr lokal. Und kostet nur die Hälfte.

Achja: Dann lassen sich natürlich auch noch Webseiten zum Offline-Lesen konvertieren. Hierfür bietet sich z. B. die App [UriToPDF](#) an. Mit ihren 100kB dient auch diese App natürlich nur als Frontend, welches die eigentliche Konvertierarbeit einem Webdienst überlässt – aber wenn die Quelle ohnehin öffentlich zugänglich ist, fällt dies weit weniger ins Gewicht, oder?

Zeiterfassung



Sicher nicht nur für Freiberufler interessant ist das Thema Zeiterfassung: Wieviel Zeit habe ich an welchem Projekt verbracht? [Xpert Timer](#) (Bild links) beantwortet mehr als nur diese Frage. Die Bedienung ist denkbar einfach, wie eine Stempeluhr: Bei Beginn der Tätigkeit auf Start, bei eventuellen Pausen auf Pause, und bei Feierabend auf Stop gedrückt. Natürlich müssen vorher einmal Kunde und Projekt erfasst sein – aber dann bekommt man neben zahlreichen Statistiken und Übersichten auch eine Stundenübersicht, die man auch direkt aus der App heraus verschicken kann. Stundensatz eingetragen? Dann zeigt sich auch gleich, wie es ums Finanzielle bestellt ist.

Wie beschrieben: Verschicken lassen sich die Reports direkt aus der App heraus. Aber auch einfach exportieren (als HTML oder CSV), und dann am PC weiter verarbeiten. A propos arbeiten: An einem Desktop-Client (leider nur für Windows) wird auch bereits fleißig gearbeitet. Und wenn ich mal mit diesem Buch fertig bin, arbeite ich vielleicht auch an meiner webbasierten Lösung weiter, sodass ich sie anderen zur Nutzung geben kann...

Fairerweise seien die Mitbewerber hier aber auch noch kurz erwähnt: Da wäre z. B. [Workaholic](#), dem man seine Arbeitsorte derart beibringen kann, dass er einen per Lokalisierung automatisch ein- und wieder ausstempelt. Oder [Time Tracker](#), der die Daten auch noch per Passwort schützen kann.

Wollte ich jetzt hier alle Alternativen aufzählen, würde es ein wenig lang. Also verweise ich wieder einmal auf den [passenden Forums-Thread](#) für die weitere Lektüre.

Sensoren

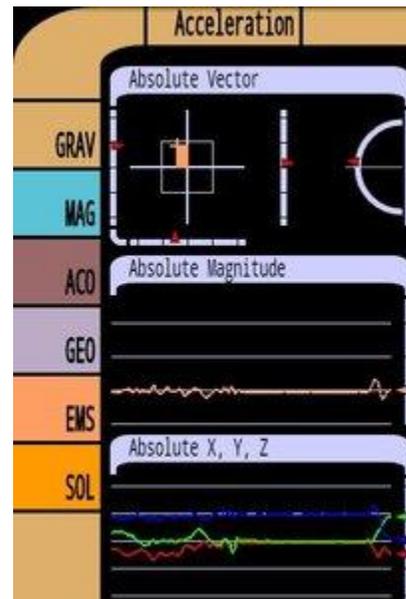
Es ist ja ein offenes Geheimnis, dass unsere kleinen Androiden mit Sensoren gespickt sind. Kaum jemand denkt darüber nach. Und wer weiß eigentlich im Detail, was da so beteiligt ist?

Entdeckerfieber geweckt? Dann lohnt sich ein Blick auf die App [Tricorder](#) (Bild rechts). Ja, sieht auf den ersten Blick nach einem Startrek-Spielzeug aus. Aber das hat es in sich. Am linken Rand finden sich die verfügbaren Sensoren: Gravity (Beschleunigungs-Sensor), Magnetfeld (Kompass), Akustik (Mikrofon), Geografisch (GPS), EMS (Elektro-Magnetisches Spektrum – also Funknetz), und schließlich Solaraktivitäten. Häh? Nein, für letzteres gibt es nicht wirklich einen Sensor – die Daten kommen übers Internet.

Die Anzeigen sind hier keinesfalls Fake, sondern spiegeln in der Tat die Sensoren-Daten wieder. Die Tricorder-App ist ein gutes Beispiel dafür, wie praktisches sich mit spielerischem verbinden lässt – weswegen ich es in meinem Buch auch unbedingt erwähnt haben wollte.

Leider hat Google den Tricorder auf Druck seitens Rechtsanwälten der CBS aus dem Markt entfernen müssen – der Name war hier weniger der Grund, Stein des Anstoßes war wohl vielmehr das Design. Nachzulesen ist diese Misere auf der [Projektseite](#). Als Ausweichmöglichkeit sei daher hier auf [My Sensors](#) verwiesen...

Wie lassen sich die Sensoren denn nun noch sinnvoll einsetzen? Da gibt es einige Möglichkeiten. Zum Beispiel als [Compass](#), oder mit [Bubble](#) als Wasserwaage. Diverse Logger und Monitore sind ebenfalls im Market verfügbar. Und (Geschicklichkeits-) Spiele. Und mehr. Details finden sich wieder einmal in einem [Forums-Thread](#)...



Augmented Reality



Hier wird die Realität erweitert – denn nichts anderes bedeutet die Übersetzung des Begriffs "Augmented Reality". Dazu werden mehrere Dinge gemischt: Das Kamera-Bild wird mit weiteren Informationen versehen. Meist mit Daten eines oder mehrerer [Sensoren](#). Oder mit Karten-Informationen. Oder weiteren Informationen zu auf dem Bild ersichtlichen Objekten. Oder einer Mischung mehrerer Komponenten...

Eine der einfacheren (aber dennoch wirkungsvollen) Varianten stellt hier [Compass Ball](#) dar: Nicht einmal 30kB Download erfordert dieses kleine Tool. Und prompt sitzt man im Kompass und schaut heraus auf die Umgebung. Nette Sache!

Etwas komplexer wird es da schon bei [Google Goggles](#). Hier ist *Augmented Reality*

eigentlich nur ein Teilaspekt der App, wie im rechten Screenshot zu sehen: Wo bin ich eigentlich, und was schaue ich da gerade an? Auf Wunsch kann *Goggles* entsprechende Informationen einblenden. Auch ohne aktiviertes GPS (wie hier im Bild); mit GPS sind die Informationen natürlich etwas genauer. Und manche scheint die App auch nur preiszugeben, wenn GPS aktiviert ist.



Wenn ich schon *Goggles* hier erwähne, dann möchte ich auch noch kurz einige weitere Features der App nennen – im übertragenen Sinne lassen sie sich ja alle in dieser Kategorie unterbringen: Man macht Fotos von realen Dingen – und *Goggles* sagt einem, was man da fotografiert hat: DVDs und Bücher (*Goggles* nennt Titel, Preis, und Erwerbsquellen), Logos, Kunstwerke (geniale Sache zum Angeben: Kurzes Foto machen und dann wissend tun, dass van Gogh dieses Gemälde namens ... im Jahre...), Barcodes (Produkt-Infos und Kaufangebote), Visitenkarten (Übernahme der Daten in die Kontaktliste), und mehr.

Damit ist das Thema aber noch lange nicht ausgeschöpft. Zu nennen wären da noch Apps wie [Wikitude](#) und [Mixare](#), die nicht nur mitteilen, was man unmittelbar sieht – sondern auch auf der Karte einblenden, was es in Blickrichtung (und in welcher Entfernung im vom eingestellten Radius) noch interessantes zu sehen gibt. Oder wenn es jemanden nicht auf der Erde hält: Mit [Google Sky Map](#) oder [Satellite AR](#) den Sternhimmel erkunden – was ist da gerade im Blickfeld? Sternbilder, Satelliten? Und wer ohnehin schon ein wenig Balla-Balla ist, kann auch gleich virtuelle (oder echte) Objekte jagen und abschießen (im Falle von echte á là Paintball) – genug Spielmaterial gibt es auch dafür. Nähere Informationen und mehr Details natürlich wieder im [passenden Forums-Thread](#)...

Fernbedienen und Überwachen

Kommen wir uns nicht alle hin und wieder etwas Fremdgesteuert vor? Und was fällt uns dazu bei unserem Androiden ein? Das logischste und naheliegendste ist, ihn als Fernsteuerung zu benutzen:

Den PC fernsteuern

Achso – dachte da jemand eigentlich an etwas anderes? Kommt auch noch, weiter unten... Aber zunächst schauen wir mal, wie wir unseren PC fremdsteuern können. Natürlich mit unserem Androiden. Und da gibt es Apps für alles Mögliche: Androide als Maus- oder Tastaturersatz, zur Bedienung von Powerpoint-Präsentationen, zur Steuerung verschiedener MultiMedia-Software wie Winamp, iTunes, VLC & Co. Auch Torrents im Blick behalten ist kein Problem.

Will man gar den gesamten PC fernsteuern, so ist auch das möglich. Auf den Mini-Displays so mancher Smartphones wird das aber sicher alles andere als bequem sein – und den meisten Tablets fehlt dafür die Netz-Verbindung... Aber es gibt hier zahlreiche Lösungen sowohl für Windows, Mac, Linux, als auch systemübergreifend.

Für letztgenanntes hat sich in letzter Zeit [Teamviewer](#) etabliert, und funktioniert sogar durch Firewalls hindurch. Hierfür installiert man auf den zu steuernden PCs den passenden Client. Und natürlich auf dem Androiden. Der Verbindungsaufbau erfolgt nun über einen Server von Teamviewer: Steckt der zu steuernde Rechner hinter einer Firewall, wird die Kommunikation vom Teamviewer-Server gemanagt. Andernfalls reicht dieser die Verbindung einfach durch.

Für den Privatgebrauch ist dies kostenlos – Firmen können entsprechende Lizenzen erwerben.

Wer bedenken hat, dass da der "Man-in-the-Middle" zu sehr mithorchen könnte, der greift halt zu einer der anderen Lösungen (über VNC oder RDP). Mit Firewall dazwischen wird es allerdings schwierig...

Und wem jetzt eine Liste möglicher Apps zu diesem Thema fehlt, der werfe bitte einen Blick auf [diesen Forums-Thread](#).



Multimedia-Geräte fernsteuern



Ohja, der Wust an Fernbedienungen auf dem Tisch oder Sofa. Und genau die, die man gerade benötigt, natürlich nicht dabei. All-in-Ones? Entweder zu teuer, oder nicht passend von der Belegung (welche Taste war's doch gleich nochmal?). Aber der Androide, der ist doch immer an der Frau (grabbel – oops... oder am Mann), kann man den nicht gleich... Aber klar doch, man kann!

Auch hier steht gleich wieder eine ganze Armee von Helferlein zur Verfügung. Aber auch wieder nicht für alles. Der große Haken: Die meisten gängigen Fernbedienungen funktionieren über Infrarot. Dafür haben "moderne Smartphones" aber weder Empfänger noch Sender (warum eigentlich nicht? Hallo, Herr Steller vom Hersteller?). Bleibt natürlich das IP-Netzwerk und daran angeschlossene IP-fähige Geräte, sofern man keinen passenden Adapter hat (oha, sowas gibbet also auch – in der Tat.). Dreamboxen können das – und die guten alten DM70x0 (aber auch neuere) kann man z. B.

mit [Controloid](#) bequem steuern. Und sieht schon vor dem Umschalten, was einen da erwartet (siehe Bild links). Voraussetzung ist hier lediglich Enigma1 oder 2. Im Zusammenspiel mit einem Streaming-fähigen Videoplayer (z. B. [VPlayer](#)) kann man damit auch direkt auf seinem Androiden das aktuelle Programm verfolgen (oder die Konserven abspielen).

Eine ganze Reihe weiterer Geräte lassen sich ähnlich fernbedienen: Etwa verschiedene Blu-ray-Player von LG und Sony, netzwerkfähige Receiver von Denon, Marantz und Yamaha, diverse TVs, und mehr. Wo sich dazu weitere Informationen finden, ist sicher nicht schwer zu erraten: Ja, auch hierfür gibt es einen [Thread im Forum](#)...

Hausautomation & Überwachung

Bei Multimedia ist natürlich noch lange nicht Schluss – wir können mehr! Auch das Licht zum Beispiel. Oder andere Dinge. Mit der richtigen Hardware und z. B. der App [EzControl](#) (Bild rechts) lässt sich so einiges steuern. Dazu braucht es allerdings in diesem Beispiel auch eine *EzControl XS1*, die verschiedene Hersteller und Standards unterstützt. Andere Apps wiederum unterstützen wieder andere Standards, wie etwa [KNXDroid](#) für EIB/KNX, oder [AutoHTN](#) für ZWave. Also für (fast) jeden etwas dabei.



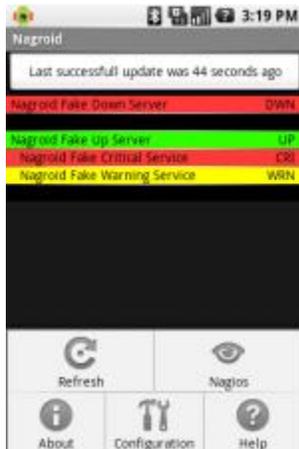
Und wer jetzt noch wissen will, wann man das Licht ausmachen muss, damit der Einbrecher auf die F...lurtüre (geschlossen) rennt, der greift zur passenden Video-Überwachung. Je nach Geldbeutel ist diese mit WebCams oder "richtig guten" IP-Cams ausgestattet. [IP Cam Viewer](#) unterstützt eine lange Liste von Kameras. [Jolicam](#) verspricht gar gleich, aus einer Webcam ein vollwertiges Surveillance-System zu machen. Und natürlich gibt es wiederum spezielle Apps für spezielle Kameras.

Nachdem der Einbrecher nun aufgelaufen ist, soll er vielleicht mit einer mobilen Kamera verfolgt werden? Mit [AndRovio](#) und der [dazugehörigen Hardware](#) ist auch das kein Problem.

Übrigens, nicht nur James Bond, sondern auch jeder Normalo kann sein Auto fernsteuern. Nein, nicht das kleine Spielzeug-Auto – das große. Zum Beispiel mit [CarLink](#) oder [OnStar](#). Die Apps gibt es gratis – das dazu passende Auto eher nicht...

So – und die Gesamtübersicht hierzu findet sich in [diesem Forums-Thread](#).

Server überwachen



Vom Haus zum Housing: Irgendwo steht der/stehen die Server, und keiner weiß, was auf ihnen eigentlich abgeht. Natürlich können wir auch diese überwachen. Und wenn es um Server-Überwachung geht, fällt uns natürlich als erstes Nagios ein. Genau dafür ist die App [Nagroid](#) gedacht. Alles grün? Prima, dann gibt es auch keine Probleme. Oder der "problematische Service" ist halt einfach noch nicht in Nagios eingebunden...

Nagios ist "zu fett" für den eigenen Bedarf – es gilt ja nur zu wissen, ob der Webserver läuft bzw. wann er Probleme hat? Dann wären vielleicht Apps à la [Site Alert Widget](#) oder der [HTTP Server Monitor](#) eine Alternative.

Weitere mögliche Apps sind in [diesem Forums-Thread](#) aufgeführt. Natürlich ist auch hier die Liste keinesfalls vollständig – wer weitere Apps kennt, kann sie dort aber jederzeit gern vorschlagen...

Anders herum: Den Androiden fernsteuern

Huch? Wo isser denn? Piep doch mal! Ja, das geht nicht nur mit dem neuen HTC-Sense Web Service. Das kann man auch haben, ohne seine Daten einem fremden Service anzuvertrauen. Auch wenn es dann wahrscheinlich nicht überall greift; aber meist hat man das "kleine Ding" ja eher im eigenen Zuhause verlegt...

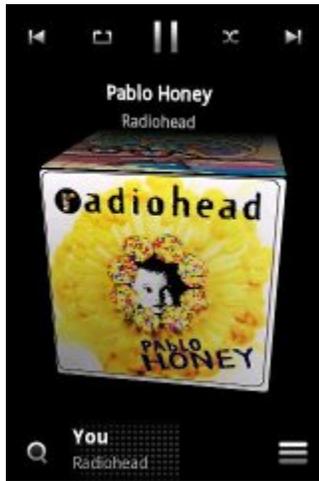
Das mächtigste Werkzeug in diesem Bereich ist sicher der [PAW Server](#). Ja, die App möchte einige Berechtigungen haben – aber sie muss schließlich auf all das zugreifen können, was gesteuert werden soll. Etwa SMS. Oder ein Foto machen. Oder irgendwelche Daten vom Phone zotteln. Alles kein Thema; sogar ein Plugin für Tasker und Locale wird bereitgestellt. Bei laufendem *PAW Server* verwaltet man sein Phone dann bequem aus dem Web-Browser heraus.

Als Alternative dazu wäre [MyPhoneExplorer](#) zu nennen – der allerdings ein Windows-System mit dort installiertem Desktop-Client voraussetzt. Dann kümmert er sich aber u.a. auch um den Datenabgleich mit Outlook, Thunderbird, Sunbird, Lotus Notes, Tobit Davis, Windows Kontakte, Windows Kalender und anderen, um Backups, das Verwalten von SMS-Nachrichten, Anruflisten, Dateien, Anwendungen...

Multimedia: Alles, was Krach macht

Zur Vielseitigkeit unserer kleinen Dauer-Begleiter gehört auch die Wiedergabe multimedialer Inhalte. Im allgemeinen Sprachgebrauch meint das: Audio und Video. Im übertragenen Sinne halt: Alles, was Krach macht. Also:

Musik: Jukeboxen und mehr



Dies ist wohl die gefragteste Gruppe: Warum noch einen MP3-Player zusätzlich mitschleppen? Allenfalls aufgrund der Akku-Laufzeit (sonst läuft am Ende nur noch der Träger und schnauft, während das restliche Equipment keinen Ton mehr von sich gibt).

Im "normalen Einsatz" ist 3 (sprich: Cubed, linkes Bild) sehr beliebt. Wo der Name herrühren mag, lässt sich dem Screenshot leicht entnehmen: Die Auswahl der Musikstücke erfolgt hier über einen Würfel. Senkrecht scrollt man durch die Alben, waagrecht geht es alphabetisch vor- bzw. zurück. Auch last.fm wird (laut Market-Kommentaren) unterstützt. Lädt Album-Art aus dem Internet, und bringt wohl auch einen Lockscreen-Ersatz mit.

Dann wären sicher noch die "Hands Free" Player für Auto und andere passende Plätze zu nennen (z. B. [Car Tunes](#)). Und Player, die mit der Zeit den Musikgeschmack des Hörers lernen, oder sich nach Farben bedienen lassen, Musik automatisch nach "Ähnlichkeit" verknüpfen. Oder solche, die man über den Bewegungssensor steuert (Schüttelt es einen bei einem Titel – schüttelt es den Androiden gleich mit, und weiter geht's zum nächsten Titel). Player mit Timer zum Einschlafen. Radios. Und, und, und. Sollten diese Themen jemanden interessieren, schaut er einfach wieder im [zugehörigen Forums-Thread](#) vorbei.

Video-Player

Mucke alleine reicht nicht – es soll auch auf dem Screen zucken? Kein Thema, auch Video-Player für Android sind nicht gerade dünn gesät. Ein richtiger Alleskönner in diesem Bereich ist der [MoboPlayer](#): Große Format-Vielfalt, Unterstützung für Untertitel und multiple Audio-Streams, Playlisten, Streaming aus dem Netz, sortieren, Thumbnails... Auf den ersten Blick scheint nichts zu fehlen. Auch eine spezielle Vorbereitung der Videos (Konvertierung) soll nicht nötig sein: Zum Einsatz kommt hier die [FFMpeg-Engine](#); alles, was die versteht, kann also abgespielt werden. Und das ist nicht gerade wenig...

Ebenfalls auf eine große Formatvielfalt greift [VitalPlayer Neon](#) zurück. Also, falls der eine nicht will, einfach den anderen probieren! Oder den ganz anderen: [No Video Player](#) erlaubt, das Bild einfach wegzulassen. Damit bleibt das Display aus (spart Akku), und man kann das Musik-Video ohne Bild genießen...



Habe ich jetzt etwas vergessen? Oh, vielleicht etwa die offizielle [YouTube App](#)? Iwo, die ist in den meisten Fällen ja ohnehin bereits vorinstalliert. Wenn nicht, findet sie sich natürlich im Market – und man kann damit das allseits bekannte Portal durchsuchen, sowie die Funde abspielen. Und mehr.

Mehr? Ja, das gibt es natürlich. Wie immer, im [passenden Forums-Thread](#).

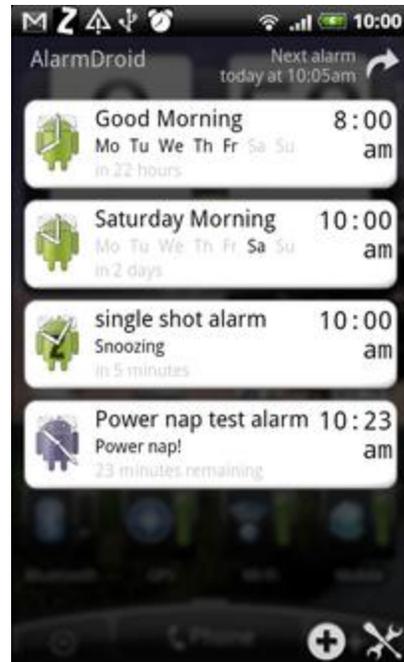
Wecker und Erinnerer



Das Klingeln des Aufzieh-Weckers ist "sowas von Out"? Naja, zumindest braucht der keinen Akku. Dafür kann man natürlich das Aufziehen mal vergessen haben... Also gut: Ja, auch der Androide kann wecken. Hat sogar ab Werk eine entsprechende App dabei. Zu öde? Da ist aber jemand anspruchsvoll! Macht nix, Hilfe gibt es trotzdem. Und zwar umfangreich. Wenn jemand in [diesem Forums-Thread](#) nicht fündig wird, sollte mich das doch stark wundern!

Neben "schnöden" allgemeinen Weckern gibt es hier eine ganze Reihe von Spezialitäten. So berücksichtigen Wecker wie [Sleep as an Droid](#) (Bild links) die Tatsache, dass wir nicht immer gleich tief schlafen, sondern in Phasen (Die meisten haben sicher schon einmal von der "Tiefschlaf-Phase" gehört: Wer da vom Wecker rausgerissen wird, steht meist "mit dem falschen Bein" auf). Diese Spezialisten warten also darauf, dass man in den "Halbschlaf" fällt – und wecken dann. Vielleicht 10 Minuten vor der Zeit, die eingestellt war; aber dennoch fühlt man sich ausgeruhter.

Das Gegenstück dazu sind die "Wellness Wecker", die uns abends sanft einlullen (z. B. mit Natursounds, oder einem selbst zusammengebrauten Mix). Ob sie einen dann morgens per Polizeisirene aus dem Bett werfen, habe ich nicht probiert... Aber vielleicht mag man dazu ja einen anderen Wecker nehmen, der ordentlich Radau macht – und damit erst aufhört, wenn eine knifflige Mathe-Aufgabe gelöst, eine Quiz-Frage beantwortet, oder (besonders ausgefallen: [Morning Routine](#)) das passende Produkt zu einem zuvor eingescannten Barcode gefunden wurde. Wie, das war jetzt die Milch – und die ist gerade alle, der Müll auch schon runtergebracht? Nachschub gibt es im Supermarkt. Und von da zurückgekehrt, den Barcode eingescannt, ist man sicher wach. Wer das nicht schafft, sondern den Task-Killer rausholt: Ja, auch wenn diese Aufgabe erledigt ist, ist man wach...



[AlarmDroid](#) mag zwar optisch eher ein wenig schlicht wirken (siehe Bild rechts), hat es aber durchaus in sich: Die App vereint das Beste aus den genannten Dingen, arbeitet zuverlässig, und nimmt noch einiges vorweg. Mich weckt sie jeden Morgen pünktlich mit einem persönlichen Gruß: "Guten Morgen, Izzy!". Es folgen Uhrzeit, aktuelles und erwartetes Wetter. Umdrehen des Androiden löst die "Snooze" Funktion aus – und 5 Minuten später geht das Ganze von vorn los. Bis ich den Androiden kräftig durchschüttel. Oder den Mini-Androiden auf dem Display mit meinen Wurstfingern erwische...

Alternativ ließe sich natürlich auch ein Internet-Radio-Stream abspielen. Oder das "Sanfte Wecken" (beginnt leise und wird immer lauter) zuschalten. Und die Mathe-Aufgabe integrieren. Wird alles von *AlarmDroid* unterstützt.

Nicht verrückt genug? Es gibt auch Wecker, bei denen man seine Freunde das Weck-Video bei Youtube raussuchen lässt. Kann eine schöne Überraschung sein. Oder möchte jemand mal zurück-brüllen, damit der Wecker Ruhe gibt? Haben wir auch. Für Leute im "Winterschlaf" gibt es sogar einen Wecker, der bei passendem Schnee auf der Piste losgeht.

A propos losgeht: Reisewecker sind was Feines, gelle? Und was, wenn Bus oder Zug wieder einmal Verspätung haben? Achso, das war bereits einkalkuliert, weil es die Regel ist... Nagut: Also was, wenn sie versehentlich mal pünktlich sind? Ohja, wir haben auch ortsbasierte Wecker. Die gehen nicht "wann" los, sondern "wo". Und man definiert statt der Uhrzeit Weckort und Radius. Sobald der Zug also 10km vorm Ziel ist, ist es soweit...

Tools

Einige Tools habe ich ja bereits vorgestellt – z. B. für die [Verwaltung](#) bzw. [Organisation](#) der Apps auf dem Androiden, oder für [Backups](#). Einige weitere sollen in diesem Kapitel folgen:

Dateimanager



Wo ist jetzt diese dumme Datei gelandet? Und wie bekomme ich mal eben die Datei von A nach B? Oder einfach weg? Datei-Manager gehören eigentlich zur Grundausstattung. Nur leider kommt bei Android nicht wirklich etwas brauchbares mit. Aber zum Glück gibt es da genügend im Market – wie z. B. der [ES Datei Explorer](#) (Bild links). Neben dem [Linda File Manager](#) und dem [Astro Datei Manager](#) gehört er zu den drei beliebtesten Apps in dieser Kategorie.

Und das nicht ohne Grund: Der *ES Datei Explorer* ist nicht nur intuitiv bedienbar, sondern kann auch gleich von Haus aus auf entfernte Dateisysteme (wie etwa den heimischen PC, oder auch einen FTP-Server) via SMB (alias Samba alias Windows-Freigabe) oder FTP zugreifen. Somit steht einem Datenaustausch nichts im Wege – auch wenn einmal kein USB-Kabel zur Hand ist. Im lokalen Netzwerk werden SMB-Freigaben auch automatisch gefunden – auch hier also wieder einfache Bedienung und Laientauglichkeit.

Auch mit ZIP-Archiven kann diese App von Haus aus etwas anfangen. Sogar ein kleiner Bildbetrachter sowie Video-Abspieler sind integriert. Und wem das noch nicht ausreicht, der findet auch Plugins für einen "Bookmark Manager" (Lesezeichen für Dateien, Verzeichnisse, etc. verwalten), einen "Sicherheits-Manager" (Apps mit Passwort schützen, Thread Detector, Gerät aus der Ferne sperren, Standort des Gerätes ermitteln) sowie einen "Task-Manager" (Task-Killer, Apps löschen – mit Widget).

Weitere und ähnliche Apps sind wieder im [zugehörigen Forums-Thread](#) aufgeführt und – teilweise – auch kurz näher beschrieben.

Tastaturen

Zumindest in diesem Bereich ist bei Android bereits ein brauchbares Bordwerkzeug dabei. Was natürlich nicht heißt, dass es nicht vielleicht besser ginge. Wobei: Was da besser ist, ist doch meist recht subjektiv – und so gibt es da auch die verschiedensten Ansätze für die Verbesserungen...



So gibt es mit [ThickButtons](#) (Bild rechts) eine Tastatur für Wurstfinger, die automatisch die wahrscheinlich passendsten Tasten (gemäß der vorigen Eingabe, also quasi Worteraten) vergrößert, damit man sie besser trifft. Finger noch dicker? Viele Tastaturen bieten auch "T9" an – tippen wie SiMSen in den "guten alten Zeiten".



Eine andere Möglichkeit der Texteingabe ist das "Swypen": Hier tippt man nicht jeden Buchstaben einzeln an, sondern "wischt" über die Tastatur, ohne abzusetzen. Und wäre es nicht schön, wenn man auch die lustigen Smileys immer gleich zur Hand hätte? Oder Funktionen zum Kopieren und Einfügen? [Ultra Keyboard](#) vereint alle diese Möglichkeiten in einer App, für den Preis von ca. 2 Euro.

Nostalgiker aus Palm-Zeiten greifen vielleicht lieber zu [Graffiti](#) (ja, das gibt es auch für Android!). Und wer es futuristischer mag, holt sich [Vlingo](#), und gibt seine Anweisungen verbal: "Vlingo! SMS an Peter: Komme heute später.", oder: "Vlingo, call Ingo!". Geht natürlich auch ohne Reim: "Vlingo: Suche nächste Sushi-Bar!". Vlingo führt dann (hoffentlich) die richtige Aktion durch: Schickt dem Peter die SMS, sucht die Nummer von Ingo raus und verbindet (telefonisch) mit ihm, und navigiert schließlich zur Sushi-Bar. Das sind nur einige Beispiele – da geht bestimmt noch mehr.

Mehr? Ja, wieder im [Forums-Thread...](#)

System-Info

Wer belegt da eigentlich schon wieder den ganzen Speicher? Und wer frisst die ganze CPU? Und wo ist die Bandbreite der Netzwerk-Verbindung abgeblieben? Diese Frage beantworten Apps wie [OS Monitor](#) (links) oder [SystemPanel](#) (rechts).



PID	Process Name	Mem
152	Telefon	28M
154	Kontakte	28M
2085	Internet	29M
258	K-9 Mail	29M
1736	Google Mail	30M
607	FeedR	33M
291	HTC Sense	39M

Die Details sind da recht unterschiedlich (und weitere in diesem Bereich verfügbare Apps mögen wieder eine andere Zusammenstellung bieten). *OS Monitor* bietet u.a. einen Task-Manager, in dem man laufende Prozesse nach Kriterien wie CPU- oder Speicherverbrauch sortieren (und bei Bedarf auch beenden) kann,

versorgt mit Informationen über vorhandene Netzwerk-Interfaces sowie offene Verbindungen (welche App und wohin – mit "whois" und Karten-Ansicht), und bietet auch Zugriff auf die System-Logs. *SystemPanel* hingegen eignet sich gut zum "Monitoring" – also zur Langzeit-Beobachtung der Verbraucher.

Natürlich sind das noch nicht alle Kandidaten dieser Kategorie: Da wären noch [Android System Info](#), welches allgemeine System-Informationen, TaskManager, AppManager, und ein farbiges SystemLog zur Verfügung stellt. Auch eine ganze Reihe von Widgets mit SystemInfos gibt es. Mehr Informationen dazu natürlich wieder im [entsprechenden Forums-Thread](#)...



Verschlüsselung

Vertrauliche Daten auf dem Smartphone sind heute sicher keine Seltenheit mehr. Was aber, wenn das Gerät in falsche Hände gerät? Wie sicher sind die Daten?

Wer also unbedingt sensible Dinge auf seinem Androiden hat, sollte sich vielleicht auch über Verschlüsselung derselben Gedanken machen. Apps wie [FilesCrypter](#) oder [Encryption Manager](#) sind in der Lage, sowohl einzelne Dateien als auch ganze Verzeichnisse zu Verschlüsseln. Ein gut gewähltes Passwort, kombiniert mit einer sicheren Verschlüsselungsmethode – das macht das "knacken" nahezu unmöglich.

Die App [Droid Crypt](#) bietet noch etliche zusätzliche Möglichkeiten: Etwa die Prüfung, ob irgendwo verschlüsselte Dateien auch noch unverschlüsselt vorliegen (das wäre gar nicht gut!). Oder die Möglichkeit, Dateien zusätzlich zum Verschlüsseln auch gleich zu Komprimieren (um Platz zu sparen). Alternativ zu Passwörtern lassen sich übrigens mit dieser App auch die Bewegungs-Sensoren nutzen: Schwenken und Schütteln als Passwort, das ist doch mal was anderes!

Wer hingegen gleich das ganze Gerät verschlüsseln möchte: Diese Möglichkeit bietet Android ab Version 4.0 von Haus aus...



Automatisieren von Aufgaben

Wozu hat man eigentlich einen Hosentaschen-Computer, wenn man dann doch jede Kleinigkeit selber machen muss? Und Mensch ist ja so vergesslich: Wieder einmal das Telefon auf dem Schreibtisch liegen lassen, zum Mittagessen gegangen, und die Kollegen hat das dauernde Klingeln "erfreut"? Oder vergessen, vor dem Starten des Navis GPS anzuschalten? Oder...



Was kann man also tun? Mein [Forums-Thread zum Thema](#) zeigt da etliche Möglichkeiten auf. Da gibt es einfache Apps für einfache Möglichkeiten (und einfache Leute) – und auch richtig komplexe Dinge, die schon ein wenig Einarbeitungszeit benötigen. Egal, was die Wünsche hier sind – dort sollte sich eine passende App finden.

Hat man einen sehr geregelten Tagesablauf, und möchte lediglich zeitgesteuert ein paar "kleine Dinge" erledigt haben – wie nachts in den Flugzeugmodus, morgens wieder an, und von 9-17 Uhr leise? Dann reicht eine zeitgesteuerte App wie [Timeriffic](#) oder [Android Audio Profile](#) völlig aus. Geht jemand oft ins Kino, aber zu unterschiedlichen Zeiten – und da soll der "kleine Quälgeist" gefälligst still sein? Dann greift dieser eher zu einer "ortsgesteuerten" App wie [Llama](#). Beides wird gebraucht, und vielleicht noch ein paar Aktionen mehr – doch zu kompliziert soll es auch nicht werden? Dann sind Apps wie [EasyProfiles](#) oder

[PhoneWeaver](#) vielleicht das Richtige.

Wer das Ganze aber richtig ausreizen will, greift zu Apps wie [Locale](#) oder besser noch [Tasker](#) (Bild links). Gerade bei letzterem kann man sich so richtig austoben – den Möglichkeiten sind hier (fast) keine Grenzen gesetzt: Bei Ankunft zu Hause das WLAN aktivieren. Um Mitternacht in den Flugmodus schalten, morgens um 7 wieder zurück, und dann auch gleich den Audio-Stream der Lieblings-Internet-Radio-Station (oder ein Random-MP3 von der Karte) auf die Ohren. Wenn der Kopfhörer angeschlossen wird, gleich den Musikplayer starten – und wenn die Navi-App gestartet wird, GPS anmachen.

Das waren noch die harmlosen Sachen. Wie wäre es damit: Während der Autofahrt eingehende Anrufe und/oder SMS automatisch beantworten lassen? Auch noch zu einfach. Anruf stummschalten, wenn das Handy auf das Display gelegt wird? Jaaa... Wifi Abschalten wenn Signal zu schwach? Geht auch. Automatische Freisprech-Einrichtung (Ton auf Lautsprecher legen, wenn Telefon nicht am Ohr)? Auch das.

Ich könnte noch eine ganze Weile so weiter machen. Alternativ kann aber auch in der "[Rezepte-Sammlung bei Android-Hilfe.DE](#) (Deutsch), oder in der [Profil-Liste des Tasker Wikis](#) (Englisch) nachgeschaut werden. Natürlich gibt es auch bei AndroidPIT zahlreiche Tasker-Threads, etwa mit [Vorschlägen für neue Profile](#). Mit *Tasker* wird uns da kaum jemals der Stoff ausgehen...

TIEFERGEHENDES FÜR FORTGESCHRITTENE

Nachdem sich die ersten beiden Teile dieses eBooks hauptsächlich an Einsteiger gerichtet haben, sollen auch die Fortgeschrittenen unter den Lesern nicht zu kurz kommen. Die hier behandelten Themen sind mit Sicherheit nichts für Neueinsteiger: Bevor Ihr Euch an die Umsetzung der "schweren Kost" macht, solltet Ihr mit Eurem Androiden-Gerät schon recht gut vertraut sein.

Dennoch heißt das nicht, dass diese jetzt das "Buch aus der Hand" legen müssen. Ich werde versuchen, möglichst allgemeinverständlich zu schreiben (auch auf die Gefahr hin, dass sich der eine oder andere mitlesende Profi ein wenig "verscheißert" vorkommen könnte). Dies verschafft zumindest einen Überblick über sich bietende Möglichkeiten. Und als Seiten-Effekt findet sich (insbesondere im [Tuning-Bereich](#)) sicher auch der eine oder andere hilfreiche Tipp für Neulinge.

Aber genug der Vorrede – kommen wir zum Thema. Oder besser zu den Themen:

Der Super-User "root"

Kauft man einen Windows-PC, gibt es auf diesem einen Account für den Benutzer "Administrator" – dem man bei der Ersteinrichtung ein Passwort verpasst. Installiert man Linux, heißt das Pendant "root" (bei einem Mac sicher ähnlich). Android basiert auf Linux – aber trotzdem gönnen uns die Hersteller den root-Zugang in der Regel nicht, sondern drohen: "Wer sich root-Zugang zu seinem Gerät verschafft, verwirkt damit den Garantieanspruch."

So, damit ist nun klar, um was es bei dem Wort "root" geht: Um den administrativen Zugang zum System, mit dem man alles (kaputt) machen kann. Naja, fast alles – die Hardware wohl eher nicht. Weshalb die Warnung mit der Garantie wohl letztendlich vor Gericht kaum haltbar sein dürfte, wenn man z. B. das Display wechseln lassen muss, oder der interne Speicher den Geist aufgibt (anders sieht es aus, wenn die CPU verglüht, weil man sie hoffnungslos übertaktet hat).

Also was nun: Braucht man den root-Zugang wirklich? Ja und nein. Wer mit seinem Gerät, dessen Funktionen, sowie der verwendeten Software bereits rundum zufrieden bist, alles so läuft, wie gewünscht, und "eigentlich" nichts vermisst – der braucht auch keinen root-Zugang. Er hat ja bereits alles, was er braucht. Hat man hingegen ein Problem, was sich ohne den root-Zugang nicht lösen lässt, sieht das schon anders aus: Je nachdem, wie schwer es einen trifft, neigt sich das Zünglein an der Waage immer mehr der Anzeige zu, die mit "mach mich root!" beschriftet ist.

Welche Vorteile sind es denn nun, die man mit einem root-Zugang erlangt – und welche Risiken sind damit ggf. verbunden?

Vorteile des root-Zugangs

Verschiedenste Einstellungen und Änderungen lassen sich ohne root-Zugang gar nicht vornehmen:

- [Anpassen der CPU Taktfrequenz](#) (siehe auch [Akkuleistung](#))
- [Entfernen/Deaktivieren vorinstallierter Apps](#) (Deaktivieren geht ab Android 4.0 auch ohne root)
- Bearbeiten der Start-Events (siehe [Apps am automatischen Starten hindern](#))
- Optimierung der Speicherverwaltung (siehe [Tuning](#))
- [Swap-Datei anlegen](#)
- automatische Datenbereinigung (Reste de-installierter Apps; siehe [Unnütze Apps raus!](#))
- App2SD bei Android < 2.2 (siehe [Tuning](#))
- Aufspielen alternativer Firmware (AKA "Custom ROM")
- Ändern der Systemschriftart(en)
- Erstellen eines wirklich vollständigen Backups Deines Android-Systems
- Einrichten einer Firewall ([DroidWall](#))

Diese Liste ist keinesfalls vollständig (natürlich auch nicht nach Relevanz sortiert – die wäre ohnehin wieder sehr subjektiv). Mit root hat man quasi überall Zugang – keine Ecke des Android-Systems bleibt verschlossen. Genau da liegt auch das

Risiko – aber da liegt es auch beim root-Zugang auf dem Linux PC, oder dem Administrator-Zugang beim Windows-PC:

Risiken des root-Zugangs

Die Risiken sind schnell mit einem Satz beschrieben: Falsch angewendet, kann man sich mit root-Zugang das System unbrauchbar machen. Im schlimmsten Fall verwandelt man gar seinen Androiden in einen Ziegelstein – wenn man z. B. ohne Sinn und Verstand die CPU hoffnungslos übertaktet, und diese schließlich den Hitzetod stirbt. Mit Wissen und Verstand eingesetzt, ist der root-Zugang ein mächtiges und nützliches Werkzeug. Quasi wie ein Autoschlüssel: Setzt sich der 8-jährige Steppke damit hinters Steuer... Womit wieder bewiesen ist, dass man uns für absolut unmündig hält...

Noch ein Wort zu vermeintlichen Risiken: "Wenn ich mein Phone gerootet habe, können dann alle Apps mit Superuser-Rechten jeden Mist machen?" Im Prinzip ja, aber... Da gibt es eine App, die nennt sich *SuperUser*. Die kommt mit jedem root-Zugang mit. Und an der müssen die Apps vorbei, die System-Rechte haben wollen. Die App lässt sie aber nicht so einfach durch: Es erscheint ein Pop-Up, welches man bestätigen muss: Darf/darf nicht, nur diesmal/immer. Also z. B. "Darf" "nur diesmal", "Darf nicht" "immer". Oder umgekehrt. Fazit: Im Prinzip kann jetzt jede App Mist bauen – aber nur, wenn der Anwender es ihr explizit erlaubt.

Wie bekomme ich root-Zugang?

Das jetzt so zu erklären, dass es für jeden gilt, führt ein wenig zu weit. Für diese Übersicht kurz zusammengefasst, gibt es da mehrere Möglichkeiten – und je nachdem, um welches Gerät es geht, greift davon eine, keine, oder mehrere.

Da ist zum einen "Software-root": Man lädt sich die passende App auf den Androiden, startet sie, und bestätigt: "Ja, ich will root!". Fertig. Toll: Mit so einem Gerät fühle ich mich absolut sicher. Wer sagt mir, dass eine andere App das nicht im Hintergrund tut, ohne mich zu fragen?

OK, auch die zweite Variante ist im Prinzip eine Art "Software-root" (schließlich geht es ja um Software-seitigen Zugang). Nur geht es hier nicht um eine "einfache App", sondern es ist schwieriger: Zunächst muss das USB-Debugging im Gerät aktiviert werden (explizierter Schritt, schwer von einer App auszuführen). Dann ist der Androide per USB-Kabel mit dem PC zu verbinden (unmöglich, dass das eine App im Hintergrund macht). Und schließlich muss man auf dem PC die "root-Software" starten, die über das Kabel auf das Android-Gerät zugreift. Die Schritte sind noch immer einfach und nachvollziehbar – aber hier habe ich keine Bedenken, dass das ohne mein Zutun passieren könnte.

Welche Variante jetzt für ein bestimmtes Gerät verfügbar ist, und welche Software dafür benötigt wird, recherchiert man am besten im Forum. Bei AndroidPIT gibt es gerätespezifische Foren (z. B. eines für das [Wildfire](#), eines für das [Desire](#), für das [Motorola Milestone](#), und so weiter). Jedes dieser Foren hat ein Unter-Forum für root-Fragen – dort finden sich die Informationen, die für das jeweilige Gerät zutreffend sind. Auch ein Blick in den [root Artikel des AndroidPIT-Wikis](#) kann sich für weitere Informationen als nützlich erweisen.

Laufen dann alle Apps mit root-Rechten?

Eine oft aufkommende Befürchtung – zum Glück unbegründet. Also die kurze Antwort: Nein, nicht ohne ausdrücklichen Wunsch des Anwenders.

Für eine detaillierte Antwort muss ich etwas tiefer greifen. Und wir müssen uns in Erinnerung rufen: Ein Android-System läuft ja mit Linux, also gelten hier auch entsprechende Richtlinien. Und jede App läuft darüber hinaus unter einem eigenen Benutzer. Auch *root* ist ein Benutzer, wenn auch ein ganz spezieller. Und wenn eine "normale App" etwas mit root-Rechten ausführen möchte, muss sie "root" dazu auffordern. Der Befehl dazu heißt **sudo**, was wir in unserem speziellen Kontext mit "SuperUser, DO ..." wiedergeben können.

Wenn eine App selbst unter "root" läuft, braucht sie auch kein "sudo". Das betrifft aber unter Android nur System-Apps, auf die der Anwender in der Regel keinen (direkten) Zugriff hat.

Läuft sie jedoch nicht unter "root" (und das ist bei Android die Regel: Jede App läuft, wie bereits gesagt, unter einem eigenen User), dann muss sie für Aktionen, die root-Rechte benötigen, root halt höflich bitten – und das tut sie, indem sie dem auszuführenden Befehl ein "sudo" voranstellt. Also "sudo <Befehl>". Derart geweckt, schaut der SuperUser in seiner "Datenbank" nach, ob die App denn sowas darf. Beim ersten Aufruf steht sie da noch nicht drin: Die Folge ist ein Popup der SuperUser-App "App xyz möchte etwas mit SuperUser-Rechten machen. Darf sie das?". Dazu zwei Buttons für "Ja" und "Nein", sowie eine "Checkbox", ob sich *SuperUser* diese Entscheidung für die Zukunft merken soll.

Bei jedem weiteren Aufruf findet der SuperUser die App in seiner Datenbank mit dem Vermerk "die darf das immer", und führt den Befehl direkt aus. Zur Sicherheit wird dieser Fakt jetzt nochmals als Hinweis eingeblendet (siehe Screenshot rechts). Die App wird dabei nicht gebremst, es ist auch keine Interaktion nötig. Daher sollte das in diesem Falle dann sogar vom Lockscreen aus funktionieren. Etwas störend ist das natürlich im Falle einer Screenshot-App, wie das Bild zeigt – da dieser Hinweis dann auf jedem Bild verewigt ist. Deshalb lässt er sich auch in den Einstellungen der *SuperUser*-App abschalten.

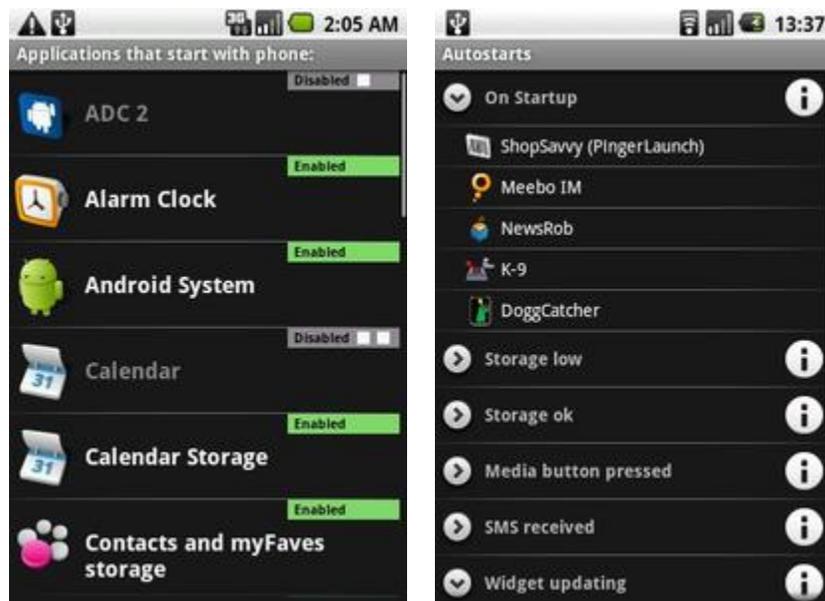


Apps am automatischen Starten hindern

Wer kennt das nicht: Man schaltet sein Handy ein, es fährt hoch, und ist eine gefühlte Ewigkeit später auch "betriebsbereit". Besonders üppig mit RAM ausgestattet sind unsere Androiden ja eher selten – und trotzdem tummeln sich schon zu diesem Zeitpunkt sackweise Apps in selbigem, die ich selten oder gar nie benötige: Flickr, FM-Radio, Google Maps, Peep... Wozu? Und wie kann ich das verhindern?

Hier soll es nun nicht um "aggressive Task-Killer" gehen, die (ausgenommen vielleicht einer Exclude-List) wild alles abschießen, was "peep" sagt (und nein, auch das Für-und-Wider derselben steht hier nicht zur Debatte). Stattdessen möchte ich Möglichkeiten nennen, gezielt die nicht (ständig) benötigten Apps an einem automatischen Start zu hindern (manchmal auch nachträglich, ooops).

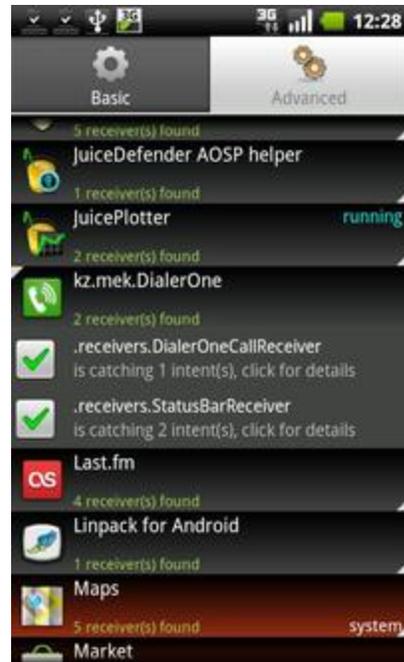
Für Details gleich an dieser Stelle den [Verweis zum zugehörigen Foren-Thread](#).



Ja, es gibt sie: Apps wie [Startup Auditor](#) (linkes Bild) und [AutoStarts](#) (rechts). Sie brauchen in der Regel [Root-Rechte](#), um ihre Tätigkeiten auszuführen. Und sie unterscheiden sich zum Teil stark – sowohl in ihrer Bedienbarkeit, Übersichtlichkeit, Wirksamkeit, als auch in der Art ihrer Vorgehensweise.

Dazu ein wenig Hintergrund-Information: Es ist nicht so, dass es da einen "Startup-Folder" gäbe. Vielmehr können sich Apps für "Events" registrieren, bei denen sie gern gestartet werden möchten. Der gewöhnlichste, der jedem sofort einfällt, nennt sich "boot completed" – unmittelbar, nachdem das System komplett hochgefahren ist. Aber das ist bei weitem nicht alles! Wer einmal mit o. g. *AutoStarts* sein System durchforstet, bekommt beim ersten Mal sicher Kulleraugen, wie viele solcher "Start-Rampen" es gibt. "USB-Kabel angesteckt" fällt einem vielleicht noch ein. Aber wer denkt sogleich an Dinge wie "eingehende SMS", "abgehender Anruf", "Speicher knapp"? Klar, jetzt fällt einem sicher auch "battery low" ein...

Je nachdem, welche dieser "Start-Rampen" unsere App nun also kennt, findet sie mehr oder weniger Kandidaten, die vom automatischen Starten abgehalten werden sollen. *AutoStarts* findet zum Beispiel sehr viele – *Startup Auditor* etwas weniger.



Und wie werden die Apps am Starten gehindert? Die meisten unserer "Verhinderer" warten einfach auf deren Auto-Start, und schießen die App dann über den Haufen. Anders *AutoStarts*: Hier wird die App quasi gleich von der Rampe genommen – und *AutoStarts* merkt sich App und zugehörige Rampe, um die Aktion ggf. später wieder rückgängig machen zu können. Das ist natürlich weit effektiver (und auch Ressourcen-schonender), birgt aber eine Gefahr: Sollte man *AutoStarts* einmal deinstallieren, ohne zuvor die Änderungen rückgängig gemacht zu haben – dann kann man sie gar nicht mehr rückgängig machen (es sei denn, man hat ein gutes Backup der App-Daten von *AutoStarts* – oder installiert die betroffene App einfach neu). Hat also alles seine Vor- und Nachteile.

Achtung: Wer die Apps nicht direkt "von der Rampe nimmt", sondern jeweils "nach dem Start abschießen lässt" (im Falle von Unsicherheit gilt letzteres), sollte anschließend prüfen, was dabei passiert. Bei einigen Apps (z. B. *Peep* oder *Aktien*) passiert es gern, dass sie nach dem "Abschuss" einfach wieder starten. Das artet dann in einen Kreislauf aus, der alles andere als Ressourcen-schonend ist! Es gibt allerdings eine App, die so etwas selbst erkennt: [Autorun Manager](#) (Bild rechts) markiert eine sich so verhaltene App als "Selbst-Restarter", sobald dieser Fall aufgetreten ist. Damit ist dann klar, dass sich diese nicht auf diese Weise am Starten hindern lässt...

Autorun Manager unterstützt übrigens beide Modi: Im "einfachen Modus" (kein root erforderlich) verhält sie sich wie *Startup Auditor*, und schießt die Apps nach dem Auto-Start einfach über den Haufen. Hier werden auch nur wenige Events berücksichtigt – also wahrscheinlich nicht alle Elemente erwischt. Im "Erweiterten Modus" (erfordert root) hingegen verhält sie sich wie *AutoStarts*, und "unregistriert" die jeweilige App vom jeweiligen Event. Hier muss man dann vor einer eventuellen De-Installation daran denken, **vorher** die ursprünglichen "Defaults" wieder herzustellen (geht allerdings einfach: "Rescue-Mode", und fertig).

Vorinstallierte Apps entfernen



Das kann echt nervig sein: Was hat mein Provider (bzw. der Telefon-Hersteller) da alles an Apps vorinstalliert, die "kein Mensch" braucht? Und wie werd ich "den Schrott" los? Jetzt kommt das böse Wort: "Ohne [root](#)? Gar nicht." Da wären wir also wieder...

Und mit root? Ja, da gibt es Möglichkeiten. Die bekannteste dürfte wohl [Titanium Backup](#) sein: Mit dieser App lässt sich jede ungewünschte App komplett vom System entfernen, wenn es denn sein soll. Wem das zu heikel ist, der hat auch eine Alternative: Einfrieren (was Android ab Version 4.0 auch von Haus aus anbietet). Damit taucht die App

in keiner Liste (außer hier bei *Titanium Backup*) mehr auf, wird nicht mehr (automatisch) gestartet – und kann dennoch jederzeit wieder "aufgetaut" werden.

Nebeneffekt der App – der Name lässt es erahnen: Man kann damit vollständige Backups machen. Von einzelnen Apps. Von deren Daten. Vom ganzen System. Und natürlich bei Bedarf Daten, Apps und System aus einem Backup zurückholen. Klasse Sache bei einem Geräte- oder ROM-Wechsel (aufpassen: Unterschiedliche Geräte/ROMs = unterschiedliche Systemdateien; hier nur die Apps und ggf. deren Daten wieder herstellen, und die System-Dinge nicht anfassen).

Wer da noch einfrieren und auftauen kann, und noch einiges mehr, ist [SuperBox](#), das sich selbst auch als "Schweitzer Taschenmesser" bezeichnet. Definitiv auch einen Blick wert. Kann nämlich auch den Cache aufräumen, und solche Dinge – die im nächsten Kapitel zur Sprache kommen...

Tuning – Das Android-System auf Trab bringen

...und nicht nur das: Auch mehr Stabilität kann ein durchaus willkommener Nebeneffekt dieser Maßnahmen sein.

Ein altbekannter Effekt: mit der Zeit wird unser Androide immer träger. Oder kommt uns das nur so vor? Doch spätestens dann, wenn jemand seinen "guten alten" Freund aus der ersten Generation neben einen Kameraden aus der aktuellen Mittel- oder gar Oberliga legt, liegt die Antwort auf der Hand: Ja klar, der neue ist definitiv schneller. Diesen großen Schritt werden wir mit unseren "alten kleinen Gurken" kaum vollständig bewältigen – aber wir können die Lücke kleiner machen. Und darum soll es hier gehen – in mehreren Schritten. Nicht jeder wird alle Schritte durchführen können/wollen (ich sage nur, da kommt Herr [Root](#) ins Spiel) – dennoch ist sicher für die meisten etwas passendes dabei. Auch, wenn das Gerät bereits neuer und schneller ist – ein wenig Finetuning kann ja nicht schaden, gelle?

Auch hier wieder, gleich zu Beginn, der Hinweis: Die ausführlichere Variante dieses Textes, ggf. mit weiteren Informationen und Benutzer-Diskussionen, findet sich im [zugehörigen Forums-Thread](#).

Schnellwaschgang

Bevor ich ein wenig in die Details einsteige, hier ein paar Schnelltipps für Ungeduldige – sozusagen das Wichtigste auf einen Blick (eine Übersicht über Verbrauchsdaten gibt es im [Anhang](#)):

Akku-Laufzeit verlängern

- WLAN komplett abschalten, wenn's nicht gebraucht wird. Das spart enorm! (das lässt sich auch [automatisieren](#))
- Wer kein mobiles High-Speed benötigt: 3G aus, das frisst auch ganz nett
- Dito ggf. für GPS (spart aber nur wenig, da es im Standby so gut wie nix frisst)
- Helligkeit des Displays weitmöglichst herunterregeln. "Aufdrehen" dann im Bedarfsfall.
- Live-Wallpaper und sonstige "animierte Dauerrenner": Wer drauf verzichten kann, sollte das tun!
- root und Modder: CPU nicht über-, sondern ggf. eher [untertakten](#). Ich weiß, das klingt nicht besonders "cool" – spart aber Akku.
- Apps, die nicht benötigt werden, [deinstallieren](#). Was nicht da ist, macht keinen Stress.
- Apps, die man *nur gelegentlich* braucht, müssen *nicht ständig* im Hintergrund laufen. Tun sie das doch, [verbietet man ihnen ggf. das Automatische Starten](#).
- Die Datensynchronisierung muss evtl. auch nicht ständig laufen. Bei vielen Apps, die eine solche benötigen (z. B. RSS-Reader, Mail-Apps) lassen sich die Intervalle entsprechend anpassen – oder gar den Abgleich auf Zeiträume mit WLAN-Empfang beschränken (das spart nebenbei gleich noch Datenvolumen)

Mehr Speed

- [RAM Bereinigen](#) (Android-interne Einstellungen optimieren)
- Eventuell [Swap-Space nutzen](#) (braucht root)
- Nur für bestimmte Situationen, wo es darauf ankommt: [CPU ggf. leicht übertakten](#). Frisst aber auch mehr Akku.
- Apps, die nicht benötigt werden, [deinstallieren](#). Was nicht da ist, macht keinen Stress.
- Die Datensynchronisierung muss evtl. auch nicht ständig laufen. Bei vielen Apps, die eine solche benötigen (z. B. RSS-Reader, Mail-Apps) lassen sich die Intervalle entsprechend anpassen – oder gar den Abgleich auf Zeiträume mit WLAN-Empfang beschränken (das spart nebenbei gleich noch Datenvolumen)

Mehr Platz im internen Speicher schaffen

- Apps, die nicht benötigt werden, [deinstallieren](#). Was nicht da ist, frisst kein Brot.
- Apps aus dem internen Speicher [auf die SD-Karte auslagern](#)
- [Cache bereinigen](#)

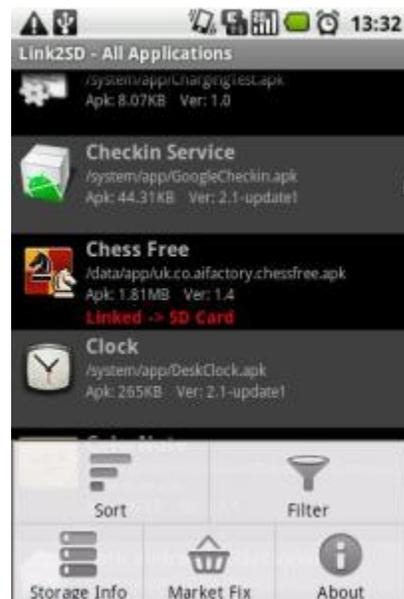
Apps auslagern

Normalerweise werden Apps im Telefonspeicher installiert – und der ist nicht immer unbedingt üppig ausgestattet. Ein paar "größere Knaller" installiert, und voll ist er: Vor der Installation der nächsten App muss man sich also entscheiden, auf welche bereits installierte App man hier verzichten kann...

Mit Froyo (Android 2.2) und neuer, lässt sich bereits von Haus aus [App2SD](#) nutzen – größere bzw. seltener benutzte Apps können so auf die SD-Karte ausgelagert werden. Damit hat man wieder freien internen Speicher – u.a. auch für die Apps, die kein *App2SD* unterstützen. Fertig – der Rest dieses Kapitels kann nun übersprungen werden!

Hm, noch hier? Also noch Android < 2.2 am Laufen, ja? Kein [root](#)? Naja, dann hat das Folgende eher informativen Charakter. Um die hier beschriebenen Möglichkeiten zu nutzen, muss der Androide zuvor gerootet werden...

Die App [Link2SD](#) (Bild rechts) ermöglicht das Auslagern von Apps auf die SD-Karte auch unter "älteren" Android-Versionen. Und zwar sogar "besser" als *Apps2SD*: Es wird eine separate [Partition](#) auf der Karte angelegt (äh, falsch: *der Anwender* muss diese Partition im ersten Schritt anlegen). Auf diese verschiebt man dann die gewünschten Apps, und am ursprünglichen Speicherort werden sogenannte "symbolische Links" auf die neue Location angelegt. Schließt man nun den Androiden per USB-Kabel an den PC an,



wird nur die andere Partition dorthin exportiert – und die ausgelagerten Apps bleiben weiterhin lokal verfügbar. Das ist bei *App2SD* nicht der Fall.

Was hierfür benötigt wird, und welche Schritte nötig sind, beschreibe ich ausführlich im Forum – und auch bei der App-Beschreibung finden sich die entsprechenden Informationen.

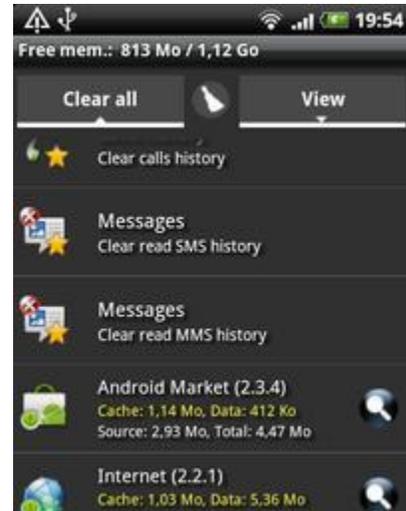
Cache bereinigen

Und weiter geht's mit der Bereinigung des internen Speichers. **Dieser Schritt benötigt kein root** – kann also von jedem genutzt werden.

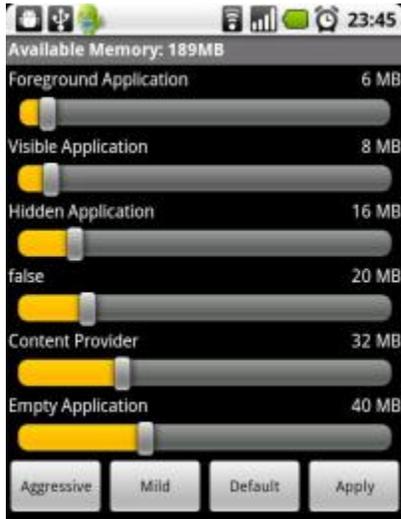
Cache ist schon eine feine Sache, und soll so manche Dinge ja schneller machen. Zum Beispiel, indem man Daten aus dem Internet im lokalen Dateisystem ablegt, damit man sie nicht jedes Mal neu laden muss, wenn sie wieder gebraucht werden. Der letzte Teilsatz ist aber genau der Knackpunkt: Braucht man sie überhaupt noch? Und was, wenn dummerweise mal gerade eine "kaputte Variante" im Cache landet? Oder der Platz im internen Speicher knapp wird?

Kurz: Wird der Cache gelöscht, geht dabei nix kaputt. Die Daten müssen im schlimmsten Fall neu (z. B. aus dem Internet) geladen, oder neu generiert werden. Dafür ist aber anschließend aufgeräumt – und so manches Problem nebenbei mit behoben.

Und wie erledigt man das? Mit Bordmitteln: In den Einstellungen in die Anwendungs-Verwaltung gehen, jeden Eintrag einzeln durchforsten (ob die App überhaupt Cache nutzt), und für jede betroffene App den Cache von Hand löschen. Da geht schon mal gern ein Stündchen bei drauf – und bequem ist es auch nicht gerade. Zum Glück gibt es zahlreiche Alternativen, wie z. B. [Quick App Manager](#) (rechts), [Quick Cache Cleaner](#), und weitere.



RAM bereinigen



Nachdem wir uns nun um den internen Speicher gekümmert haben, geht's ans Eingemachte: Das RAM ist fällig!

Wer aber nun an Task-Killer denkt – voll daneben. Kontrovers diskutiert, verteufelt, hochgejubelt... Aus meiner Sicht ist die Aufgabe eines Task-Killers nicht, freien Speicherplatz im RAM zu schaffen (das ist eher ein Nebeneffekt) – sondern vielmehr, "hängende" Apps zu beenden, die andernfalls z. B. Amok laufen, oder das System lahmlegen (z. B. mit hoher CPU-Last). Details finden sich weiter oben unter [Von Taskkillern und anderen bösen Buben](#).

Nächstes Argument: "Android kümmert sich doch selbst um die Speicherbereinigung". Jetzt kommen wir der Sache näher: Ja, das stimmt – die Frage ist nur: Wann? Wie oft? Und wie? Darum geht es in erster Linie: Die zugehörigen Einstellungen für diesen sogenannten "OOM-Killer" (**O**ut **O**f **M**emory **K**iller) anzupassen. Ihm beizubringen, wie viel freien Speicher wir benötigen – und wann er zuschlagen darf.

Auch hierfür gibt es wieder zahlreiche Helferlein. Da es sich um System-Einstellungen "unter der Haube" handelt, wird hier i.d.R. [root](#) benötigt. Eine rühmliche Ausnahme ist da [Auto Memory Manager](#) (Bild links), der es angeblich auch ohne schafft. Selbst getestet habe ich den [AutoKiller Memory Optimizer](#), der recht gut funktioniert.

Swapspace nutzen

Kommen wir zum nächsten Punkt dieser Reihe: Wenn das RAM nicht reicht, wie schaut es mit "virtuellem RAM" – also dem Auslagern auf einen anderen Datenträger – aus? **Hier brauchen wir definitiv wieder root.**

Klar: Wenn der Speicher im RAM knapp wird, kommt Android (genauer: o.g. OOM-Killer) daher und beendet "überflüssige" Prozesse. Geht ja auch nicht anders. Aber was in diesem Augenblick überflüssig ist, war das vielleicht vor 10 Minuten noch nicht – und wird u.U. in weiteren 10 Minuten doch wieder benötigt. Ja, es lässt sich neu starten – aber das ist naturgemäß nicht so schnell wie das "aufwecken" bestehender Strukturen.

Mehr RAM ist (in bestehender Hardware) schwer realisierbar – es lässt sich ja nicht, wie etwa beim PC, einfach ein zusätzlicher Speicher-Riegel einstecken. Somit kommen wir zur Auslagerung, dem sogenannten **Swapping**. Hierbei wird ein Bereich eines Datenträgers (gewöhnlich einer Festplatte – im Falle unserer Androiden muss dafür die SD-Karte herhalten) genutzt, um Teile des RAM (natürlich die gerade am wenigsten benötigten) dort temporär unterzubringen. Werden diese wieder benötigt, liest das System sie halt wieder ein. Das ist zwar nicht so schnell, als würde ausschließlich im RAM gearbeitet – aber i.d.R. immer noch schneller und besser, als hätte man diese Möglichkeit überhaupt nicht.

Wie realisieren wir das nun – sofern wir nicht ein Custom-ROM einsetzen, bei dem es bereits "fertig eingerichtet" ist? Auch hier greift uns wieder eine geniale App hilfreich unter die Arme: **Swapper** (Bild rechts). Die App kümmert sich in jeder Hinsicht um das Bereitstellen der Funktionalität (sofern der eingesetzte Betriebssystem-Kernel das unterstützt):

- Neuanlegen einer Swap-Datei beim Boot
- sauberes Abschalten des "Swapping", wenn der Datenträger per USB gemountet wird
- Neu-Aktivierung nach Lösen des USB-Mount
- gezieltes Einstellen der "Swap-Prioritäten"

Sind nur die wichtigsten Punkte – und jeder davon ist optional. Es lässt sich auch eine Swap-[Partition](#) nutzen (das erübrigt dann auch das Abschalten bei USB-Kontakt, sowie das spätere wieder Aktivieren). In der Regel ist eine Swap-Partition ein wenig schneller als eine Datei – aber aufgrund der begrenzten Schreibzyklen von SD-Karten hat man länger etwas von selbigen, wenn Swapper sich darum kümmert, die Datei bei jedem Neustart an anderer Stelle der Karte wieder anzulegen.

Nebenbei: Übertreiben sollte man es auch nicht, ein gutes Mittelmaß ist gefragt. Ist zuviel "Krams" ausgelagert, bremst das am Ende doch wieder. Gute Anfangswerte sind 32..64MB Swap sowie eine "Swappiness" von 10..20.

Unnütze Apps raus!

Das sollte eigentlich sowas von klar sein: Apps, die man überhaupt nicht mehr braucht, belegen unnütz Speicher – und können auch bei Nicht-Benutzung das System (leicht) ausbremsen. Also runter damit! Die notwendigen Details hierzu finden sich bereits weiter oben unter [Apps verwalten](#). Und was man nicht



deinstallieren kann/möchte, kann man ja zumindest noch [am automatischen Starten hindern](#).



Noch Leichen im Keller? Im Laufe seines Androiden-Lebens hat unser Gerät so manche App "einmal kurz gesehen", die dann wieder entfernt wurde – etwa weil wir etwas besseres gefunden haben, oder die App aus anderen Gründen nicht mehr benötigen. Nicht jede dieser Apps wurde restlos entfernt – manch eine App ließ noch ein paar "Leichen" zurück. Und dann wären da noch diverse "core dumps" von "Force Closes" ("Schließen erzwingen"), System-Logs, und mehr.

Eine neue App verspricht, sich genau darum zu kümmern: **SD Maid** (Bild links). Die App **braucht allerdings root**. Was sie tut: Sie durchsucht die typischen Verzeichnisse (`/data/data` und `/mnt/sdcard/Android/data` bzw. bei Samsung-Geräten `/dbdata/databases` für erstere Location) und vergleicht die dortigen Einträge mit der Liste tatsächlich installierter Apps. Auf diese Weise wird überflüssiges identifiziert, und kann entweder einzeln (Eintrag antippen) oder gleich in einem Rutsch ("Clean All") entfernt werden. Im Reiter "Clean System" lassen sich weitere Plätze und Dinge bereinigen – etwa System-Logs, Core-Dumps oder – huh? – auch Cache.

CPU Taktung anpassen

Wiedermal nur für gerootete Geräte.

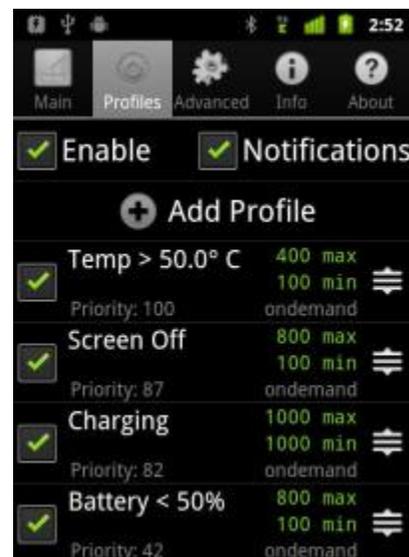
Hier gibt es generell eigentlich zwei Richtungen:

- Höhere Taktung → Schnellere Reaktion & Arbeitsgeschwindigkeit – aber auch der Akku wird so schneller leer
- Niedrigere Taktung → Langsamere Reaktion & Arbeitsgeschwindigkeit – aber der Akku hält (teilweise deutlich) länger durch

Eigentlich klar soweit – auch wenn die meisten beim Thema "CPU Taktung" nur/zuerst ans Übertakten denken, auch Untertakten kann seine Vorteile haben. Beim Übertakten sollte man überdies vorsichtig sein, dass man aus seinem Androiden nicht zuerst eine Warmhalteplatte, dann eine Kaffeemaschine, und schließlich einen Ziegelstein macht – nicht übertreiben!

Sinnvolle Szenarien beinhalten u.a.:

- Wenn das Gerät nicht benutzt wird (Bildschirm aus, Nachtschlafene Zeit, ...) → runter mit dem Takt
- Wenn das Gerät benutzt wird (Bildschirm an) → Auf "normalen" Takt schalten (was immer das für den einzelnen ist)
- Wenn eine bestimmte (hungrige) App im Vordergrund läuft → Hoch mit dem Takt (Vorsicht – nicht zu hoch)



Viele der CPU-Taktveränderer decken die ersten beiden Punkte ab. Der dritte Punkt ließe sich z. B. mit [Tasker](#) sicher realisieren.

Wiederum gibt es etliche Apps, die beim Drehen an der CPU-Schraube behilflich sein wollen. Die bekannteste ist sicherlich [SetCPU](#) (rechtes Bild), der bekannteste Mitbewerber dazu heißt [CPU tuner](#). Weitere Kandidaten sind im [Forums-Thread](#) benannt, oder finden sich auf den jeweiligen Seiten der Apps als Alternativen aufgeführt.

Durststrecke – mehr aus dem Akku herausholen

Wenn der Akku mit einer Ladung länger durchhalten soll, müssen wir den Energieverbrauch senken. Soweit klar. Also stellen sich die Fragen:

- Was verbraucht Energie?
- Was ist dafür verantwortlich?
- Wie können wir dem beikommen?

Schauen wir uns diese Fragen also einmal nacheinander an.

Was verbraucht Energie?

Gleich von vorn herein auf den Aspekt des "können wir dem beikommen" betrachtet, lassen sich die Verbraucher schnell auf wenige Punkte reduzieren:

- CPU
- Speicherzugriffe
- Netzwerkzugriffe
- "Peripherie-Geräte" (hier: Kamera)

Dafür verantwortlich sind natürlich die Apps, welche die Ressourcen beanspruchen – und natürlich nicht zuletzt der Benutzer, der einen gewissen Einfluss auf diese Apps hat. So ergeben sich einige Lösungsansätze schon von selbst:

Wie können wir dem beikommen?

Zum größten Teil sind diese Punkte bereits in den vorigen Kapiteln behandelt worden. Diese Kandidaten seien daher hier nur noch einmal kurz aufgezählt:

- Apps, die nicht benötigt werden: Weg damit! Was nicht da ist, kann auch nichts verbrauchen.
- Was man nur selten braucht, muss nicht ständig laufen – diese Apps sollte man also am [automatischen Starten hindern](#)!
- Wenn im RAM genügend freier Platz existiert (also der ["OOM-Killer" gut konfiguriert ist](#)), wird dieser als Cache genutzt – und dafür weniger auf die Speichermedien zugegriffen.
- Exzessive Nutzung von [Task-Killern](#) verbraucht zusätzliche Energie – da viele Apps gleich nach dem Kill vom System wieder nachgeladen werden
- Ist der Androide [gerootet](#), lässt sich [die CPU untertakten](#) (zumindest dann, wenn man sein Gerät ohnehin nicht benutzt).

Die meisten Dinge haben wir bereits besprochen. Ein paar jedoch noch nicht:

Speicherzugriffe: Ja, klar: mehr Cache mit sauberem Speicher – stand doch da schon. Richtig, aber das ist noch nicht alles. Die Frage ist auch: Was für eine SD-Karte steckt in meinem Gerät? Im Allgemeinen verbrauchen "ältere Karten" (also die langsameren Teile – erkennbar u.a. an ihrer kleineren "Klasse") mehr Strom als neuere, da die Lesezugriffe hier ja länger brauchen. Eine neue Karte verschafft uns also u.U. nicht nur mehr Speicher, sondern ist auch schneller – und der Akku hält länger. Steht also ohnehin ein Neukauf an, sollte man nicht zu

knauserig sein: Lieber einen Euro mehr ausgeben, und was vernünftiges holen! "Class-6" sollte es heutzutage schon sein.

Netzwerk-Zugriffe: Auch diese kosten Energie – manche mehr, manche weniger. 3G (UMTS) knabbert i.d.R. stärker am Akku als 2G (GPRS/EDGE). Wer also (wie ich) nicht unbedingt auf "HighSpeed" an dieser Stelle angewiesen ist, schaltet 3G einfach komplett ab! Die passenden Schalterchen finden sich in "Einstellungen→Drahtlos und Netzwerke→Mobile Netzwerke" (Netzwerkmodus: "Nur GSM"). Weitere Kandidaten sind Bluetooth und WLAN, die man bei Nichtgebrauch auch deaktivieren kann (siehe dazu unter [Helferlein](#)).

In die Rubrik *Netzwerk-Zugriffe* gehört übrigens auch der **mobile Datenabgleich** – auch hier kann man einschränken: Wetterinformationen müssen nicht alle fünf Minuten aktualisiert werden (hier genügen in der Regel 3-6 Stunden als Intervall). Bei RSS-Feeds kommt es auf die Ansprüche an; normalerweise ist ein stündlicher Abgleich aber ausreichend. Bei vielen RSS-Readern lässt sich das sogar noch auf die Zeiträume beschränken, in denen eine WLAN-Verbindung verfügbar ist; das spart dann nebenbei auch noch Datenvolumen. Bei Mails empfiehlt sich ein Anbieter (und eine App), die das "IMAP Idle" Protokoll unterstützen: Dann stößt der Server nämlich den Abgleich an, wenn neue Mail eintrifft – und man muss nicht alle paar Minuten unnötig nachschauen lassen ("Pollen").

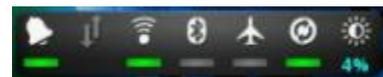
Kamera: Natürlich braucht auch diese "Saft", und nicht unbedingt wenig. Das betrifft nicht nur das Fotografieren, sondern auch die Barcode-Scanner. Von letzteren sind etliche dafür bekannt, dass sie auch nach "beenden" noch fleißig weiter nuckeln (was an einem Bug in der betreffenden Google-API zu liegen scheint). Hier könnte also ausnahmsweise ein [Task-Killer](#)-Einsatz gerechtfertigt sein. Oder aber die Wahl eines alternativen Barcode-Scanners.

Zu letzterem Fall gibt es sicher noch Parallelen, wo eine alternative App sparsamer wäre als die gerade eingesetzte. Um solche Kandidaten aufzuspüren, kann man z. B. die Statistiken unter "Einstellungen→Telefoninfo→Akku→Akkuverbrauch" nutzen, die besonders hungrige Apps auflistet. Aber aufpassen: Die Statistik zählt seit dem letzten Boot – lief Deine hungrige App also nur mal eben ein Minütchen, fällt sie hier (noch) nicht so ins Gewicht. Mehr Details liefert dann auch noch ein "Anruf" bei der [magischen Nummer](#) *##4636#*##*...

Helferlein

Nicht verschweigen möchte ich hier einige Helferlein, die dem Nutzer beim Energiesparen "unter die Arme greifen" können. Grob gesehen wären dies zwei Kategorien von Apps: Schnellumschalter (finden sich in [diesem Forumsthread](#) ausführlicher behandelt), und im Hintergrund laufende, auf "Events" reagierende Apps.

Schnellumschalter sind meist Widgets auf dem Home-Screen, mit denen man "toggeln" (also zwischen zwei Zuständen wechseln) kann. z. B. "WLAN an/aus", "Bluetooth an/aus", etc. Einige dieser Widgets stehen schon "von Haus aus" zur Verfügung – aber damit kann man sich den Home-Screen schnell "zupflastern", wenn mehrere benötigt werden. Daher gibt es auch Apps, die



mehrere Schalterchen sinnvoll vereinigen, und so platzsparend mehr anbieten können. Als Beispiele seien hier nur [Dazzle](#) (rechtes Bild) und [Extended Controls](#) genannt.

Hintergrund-Aufpasser steuern den Energieverbrauch "ereignisgesteuert", wobei der Anwender die Ereignisse meist selbst festlegt. Zum Beispiel: Wenn weniger als 20% Akkuleistung verbleibt, schalte Bluetooth und WLAN aus. Die bekannteste App in dieser Kategorie nennt sich [JuiceDefender](#) (Bild rechts); ein vergleichbarer Kandidat wäre [Green Power](#).

Nicht vergessen werden sollten an dieser Stelle auch spezielle Helferlein wie [APNroid](#) (mit dem man sein Telefon – temporär – daran hindern kann, Netzwerk-Verbindungen aufzubauen) oder [3G Watchdog](#) (passt nicht so ganz hier, da es eher über das Datenvolumen wacht – aber es nutzt [APNroid](#), wenn man möchte, um bei einem bestimmten Volumen "abzuschalten").

Sicher gibt es noch etliche mehr – und ich gebe auch frei heraus zu, nicht alle zu kennen. Aber ich hoffe, hier zumindest einen Einblick in die Möglichkeiten gegeben zu haben: Nun ist jedenfalls bekannt, wonach man Ausschau halten kann. Und zum Glück werden ja bei [AndroidPIT](#) zu (fast) jeder App ja eine Reihe vergleichbarer Apps genannt, sodass man sich da weiter "durchhangeln" kann.



Den Akku Kalibrieren



Wozu man sowas braucht? Android pflegt da eine spezielle Datei mit Akku-Statistiken. Auf den dort gespeicherten Daten beruht u.a. auch die Batterieanzeige: Wieviel ist noch übrig? Wenn man nun einen neuen Akku einbaut, werden die alten Werte unbrauchbar. Auch verliert ein Akku mit der Zeit an Kapazität – wie auch der Mensch, wenn er älter wird... In beiden Fällen macht es durchaus Sinn, die gesammelten Statistiken zurückzusetzen, und den Akku neu zu trainieren.

Was das noch bringt? Naja, spätestens wenn es passiert, dass die Anzeige bei 15% steht, der Androide aufgrund veralteter Statistiken hier aber "gelogen" hat, und sich das Teil einfach ausschaltet (weil 0%), hat man eine weitere Antwort. Noch eine fällt einem ein, wenn man nach der Kalibrierung weniger oft ans Ladekabel muss – weil die 15% nicht schon bei 50% angezeigt werden...

Also, kurz und gut: Wer meint, da kann was nicht mit rechten Dingen zugehen (und all die Dinge aus den vorhergehenden Kapiteln schon versucht hat) – eine Kalibrierung kann eigentlich nicht schaden.

Ein hilfreiches Tool dazu präsentieren die XDA Developers mit [Battery Calibration](#) (Bild links). Nach dem Start der App gibt diese genaue Anleitungen für das Vorgehen: Ladekabel anschließen und Akku vollaufen lassen wird empfohlen. Damit man nun nicht alle paar Minuten nachschauen muss, wie weit das Ganze gediehen ist, kann die App auch mit einem "Piep" benachrichtigen – wenn man nicht wie ich ein QVGA-mini-Display hat und somit übersieht, nach dem "Tapp" auf den "Piep" Button auch den links daneben für das "Wait" noch zu betätigen...

Akku voll? Dann geht's los: Per Druck auf den entsprechenden Button veranlasst man die App, die alten Akku-Statistiken zu löschen. Dem System bleibt nun keine Alternative mehr: Es muss neue erstellen. Womit die "Leichen" aus dem Keller wären. Empfohlen wird, den Akku nun einmal bis zur Erschöpfung zu leeren (am Besten komplett, unter 20% sollte es sein), und dann mit einem ununterbrochenen Ladezyklus wieder auf 100% hoch. Muss nicht sein, aber desto akkurater wird die Sache.

Alternative Apps? Aber klar doch, wie immer in einem speziellen [Forums-Thread](#)...

Wer saugt da meinen Akku leer?

Soso, irgendwer saugt da also plötzlich den Akku leer – und der Anwender hat keinen blassen Schimmer, wer oder was das sein könnte? Da gibt es zunächst ein paar Bordinstrumente für die Analyse. Zum Beispiel die Übersicht der "größten Verbraucher", die sich unter Menü→Telefoninfo→Akkuverbrauch finden. Und auch ein paar mehr Details, wenn man mal bei `*##4636##` "anruft".

Allerdings bieten beide Stellen nur einen groben Überblick über längere Zeiträume. Ein "Dauerläufer" sollte so zwar aufspürbar sein – doch manchmal hätten wir gern ein paar mehr Details zur Hand. Aber auch kein Thema: Einen passenden Kandidaten habe ich ja bereits unter [System-Info](#) benannt, das [SystemPanel App](#) (Bild rechts).

Für die detaillierte Analyse muss es allerdings in der Tat die Kaufversion sein – denn nur diese bietet das Monitoring im Hintergrund. Sobald sie für eine Weile Daten gesammelt hat, können diese dann ausgewertet werden. Wie das in etwa aussieht, zeigt der Screenshot am Beispiel eines bereits beendeten Google-Maps: Der obere Graph lässt erkennen, dass diese App etwa von 18 Uhr bis 20 Uhr lief. Ein Ladekabel war offensichtlich nicht angeschlossen (kein grüner Balken bei "Charging"), dafür wurde der Akku aber permanent entladen (dritter Graph von oben). Auch die CPU war derweil gut beschäftigt (unterster Graph),



obwohl das Gerät nicht selbst durchgehend aktiv genutzt wurde (blauer Graph). Während dieser zwei Stunden leerte sich der Akku von knapp 100% auf gut 50% – ohja, das hat gut geschluckt! Und der Schuldige am "plötzlich leeren Akku" ist mit ziemlicher Sicherheit identifiziert...

Wie jetzt: Alternativen? Na gut, auch hier gibt es natürlich einen passenden [Forums-Thread...](#)

ROMs: Stock, Vendor, und Custom

ROM – was ist das denn nun wieder? Klar, eine Abkürzung. **ROM** steht für **Read-Only Memory**. Also eigentlich Speicher, auf den nur lesend zugegriffen werden kann. Im Falle einer CD oder DVD ist das klar – im Falle unserer Androiden eigentlich glatt gelogen: Hier gehörte dann noch ein "m" davor, für "mostly" ("meistens"). Denn natürlich kann man da auch Schreiben: Irgendwie müssen die Updates ja da rein kommen...

Wie dumm von mir: Ich sollte erst einmal sagen, was da bei Android eigentlich drin ist. Nämlich das Betriebssystem, sowie die "Core Apps" (Google Market, Telefon & Co.). Manchmal auch noch mehr – kommt auf das ROM an. Da gibt es nämlich verschiedene: Stock, Vendor, Custom...

Stock ROM

Als "Stock" ROM bezeichnet man eigentlich das, was direkt aus der Google-Schmiede kommt. Wörtlich eigentlich aus dem "Lager". Also das "nackte Original", so wie es von Google eigentlich gedacht war. Naja, die meisten verstehen darunter auch (noch) das Folgende:

Vendor ROM

Das vom Hardware-Hersteller vorinstallierte ROM. Also das, was aus dem gleichen Lager kommt wie das Gerät – weshalb es meist auch als "Stock ROM" bezeichnet wird. Obwohl das so nicht korrekt ist: Es gibt kaum einen Geräte-Hersteller, der hier nicht noch seine Modifikationen einbaut. Ich denke da nur an HTC's Sense, Motorolas Moto-Blur, oder Samsungs TouchWiz Oberfläche. Oder an das ganze Gesocks an vorinstallierten Apps, die "Otto Normalbenutzer" gar nicht und "Super User" [auch nur mit Mühe](#) wieder los wird. Genau genommen fehlt jetzt gar noch eine Kategorie, denn auch die Provider spielen nochmals dran rum, und pappen ihren Brandy, pardon, ihr **Branding** noch oben drauf...

Während Updates von Google natürlich recht regelmäßig, und Updates vom Hersteller noch "relativ zügig" kommen, bremst das Branding natürlich ein weiteres Mal: Denn auch hier muss erst das Zusammenspiel getestet werden. So kommt ein Hersteller-Update in der Regel frühestens im Quartal nach dem Google-Update. Und das Update fürs gebrandete Gerät kann sich durchaus noch zusätzlich um ein halbes Jahr verspäten. Zu diesem Zeitpunkt ist unter Garantie schon wieder mindestens ein neues Stock-ROM-Update draußen...

Custom ROM

Kostümiert? Zugegeben, auch das gewissermaßen. Aber eigentlich steht "Custom" für "customized", also auf den Kunden (customer) angepasst. Im Gegensatz zum Hersteller, oder Provider, zu sehen. Hier stehen meist eingespielte Teams von Entwicklern dahinter – und ein neues Custom-ROM steht häufig schon kurz nach der Stock-Version (nicht selten auch mal vorher) zur Verfügung.

Das [AndroidPIT Wiki](#) schreibt zu diesem Thema:

Eine auf Open Source basierende Software ist zur Entwicklung durch Dritte freigegeben.

Somit haben Entwickler die Möglichkeit die original Software zu modifizieren, zu optimieren, Elemente hinzu zu fügen oder auch zu entfernen mit dem Ergebnis für den Verbraucher das bestmögliche Ergebnis zu bieten in den Bereichen Performance, Energie, Effizienz etc.

Womit auch bereits die zahlreichen Vorteile geklärt wären. Haken an der Sache: Um ein Custom ROM installieren zu können, muss der Androide [gerootet](#) sein. Das schafft Dir aber u.U. ein paar Probleme im eventuellen Garantiefall...

Die bekanntesten Custom-ROMs bzw. ihre Entwickler sind wahrscheinlich [MoDaCo](#) und [Cyanogen](#). Aber es gibt noch zahlreiche Spezial-ROMs, die speziell auf bestimmte Geräte zugeschnitten sind (wie etwa WildPuzzle und OpenFire im Falle des HTC Wildfire). Hier informiert man sich am besten bei AndroidPIT im root-Forum des betroffenen Gerätes.

Selbst installieren?

Kann man sich nun selbst ein ROM eigener Wahl installieren? Klar doch – denn außer vielleicht einem "guten Kumpel" nimmt einem das kaum jemand ab. Am wenigsten der Hersteller. Mit den richtigen Werkzeugen (wie z. B. dem [ROM Manager](#), Bild rechts) ist das auch recht problemlos bewerkstelligt – die App bietet u.a. eine Übersicht der für das jeweilige Gerät verfügbaren ROMs, lädt diese herunter, und führt durch den Installationsprozess. Einschließlich Backups, und was so dazu gehört.

Besonders vorbildlich ist das Ganze im Falle von [Cyanogen](#): Hier stellt das [Wiki](#) für jedes unterstützte Gerät eine Anleitung bereit. Wie das Ganze in der Praxis aussieht, und ob es wirklich so "schwierig" ist, wie es beim ersten Mal klingt – lässt sich übrigens für das Beispiel "HTC Wildfire bekommt Gingerbread" in [diesem Blog](#) nachlesen.

Wie bereits im vorigen Kapitel beschrieben, ist allerdings das [rooten](#) des Androiden eine Grundvoraussetzung, ohne die man kein Custom ROM installieren kann...



Ortsdaten-Cache einsehen (und verwalten)

Es geisterte ja gerade durch die Presse: Der böse Apfel-Konzern sammelt Bewegungsprofile seiner Kunden! Und speichert diese ein ganzes Jahr! Da liegt es natürlich nahe zu schauen, was Android in diesem Zusammenhang treibt.

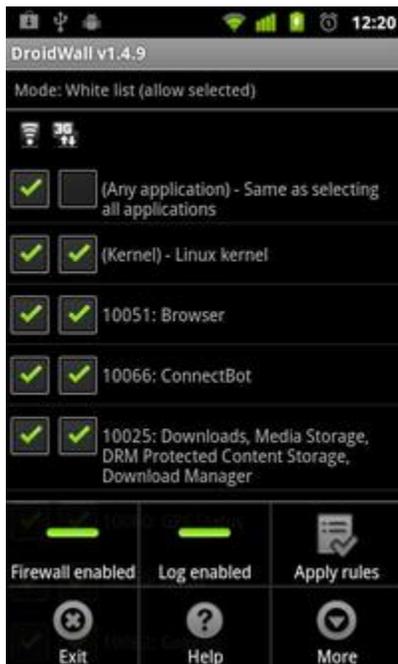
Und ja, auch hier wird "gesammelt". Das Ganze nennt sich "Cache". Laut Informationen des Entwicklers von [Android Location Cache Viewer](#) sind das die letzten 50 Mobilfunk-Zellen sowie die letzten 200 Wifi-Locations. Ein ganzes Stück weniger als Daten eines ganzen Jahres – auch wenn sich damit, zugegebenermaßen, durchaus auch bereits so einiges anfangen ließe.

Doch nun die gute Nachricht: Wie [Forums-Mitglied Frank feststellen durfte](#), bleibt der Cache komplett leer, wenn man das "Datensammeln" in den Netzwerk-Einstellungen einfach abschaltet (Huch? So einfach?). Und aus die Maus. Einziger Haken an der Sache: Der Androide benötigt dann beim ersten Aufruf einer "Location-based App" (Google Maps etc.) ein wenig länger, um die Position festzustellen. Und bei einigen ROMs (bzw. Android ab Version 2.2) schaltet man damit die Positions-Ermittlung über WLAN und Mobilfunknetz gleich ganz ab...

Wer nur mal eben kurz nachschauen wollte, wo er so gewesen ist – für den bietet sich (neben gerade benannter App) [Location Cache](#) (Bild rechts) an – sofern der Androide gerootet ist. Diese App zeigt die im Cache gespeichert Koordinaten als Liste oder auch auf der Karte an, bietet die Möglichkeit, die Daten im [GPX-Format](#) auf die SD-Karte zu exportieren, und kann überdies auch den Cache löschen sowie deaktivieren. Für letzteres benutzt er bestimmt das Schalterchen, welches Frank in den Einstellungen auch betätigt hat...



Zugriffe Sperren: Firewalls & Permission-Blocker



So mancher App möchte man kräftig auf die Finger hauen: Was will das Teil ständig im Internet? Dieser Teil der App-Funktionalität lässt sich leider auch nicht separat abschalten. Oder man weiß nicht, wann eine App ins Internet will, und wozu schon gar nicht. Nur die entsprechende Permission hat die App halt (jaja, das haben sie fast alle – in der Regel zum Laden von Werbung). Kurz und gut: Sie soll das nicht! Ist das hinzubekommen?

Na, wenn der Izzy das hier so dumm fragt, hat er bestimmt auch... Richtig. Eine mögliche Antwort ist im Bild zur Linken zu sehen und nennt sich [DroidWall](#). Dabei handelt es sich um ein FrontEnd für [iptables](#), welches im System integriert sein muss. Dies ist bei den vom Hersteller gelieferten Geräten von Haus aus leider nur selten der Fall, sodass die App in der Regel ein [Custom ROM](#) voraussetzt. Was [root](#) natürlich gleich impliziert...

Sind diese Voraussetzungen jedoch gegeben, hat man mit *DroidWall* ein gutes Werkzeug zur Hand: Entweder, man verbietet einzelnen Apps den Zugriff auf's Internet (erstellt also eine sogenannte "Blacklist" der "schwarzen Schafe") – oder man verbietet den Internet-Zugriff generell für alle Apps außer denen, die auf eine sogenannte "Whitelist" (die Apps mit der "weißen Weste") gesetzt wurden. Noch besser: Dies lässt sich auch getrennt nach "Netzwerk Interface" tun – was sich z. B. anbietet, wenn nur ein begrenztes Datenvolumen im Mobilfunk-Vertrag enthalten ist: So kann man einer App verbieten, die "mobile Datenleitung" zu benutzen – ihr aber gleichzeitig den Internet-Zugriff per WLAN erlauben.

Darüber hinaus lassen sich Zugriffe auch protokollieren. So weiß man anschließend zumindest, was "abging"...

Okay, prima – aber das betrifft ja nur Internet-Zugriffe. Klar, schon eine gute Sache: Eine App, die nicht ins Internet kommt, kann auch nicht meine privaten Daten dorthin übertragen. Aber wie schaut das mit den anderen Dingen aus? Zugriff auf meinen aktuellen Standort, meine SMS, meine Adressdaten?

Das sah bislang schlecht aus – aber zum Glück gibt es endlich erste gute Nachrichten: So enthalten aktuelle Versionen des [Custom-ROMs](#) von [Cyanogen](#) einen kompletten Permission-Blocker. Und wenn es nun dieses ROM für mein Gerät (noch) nicht gibt? Dafür gibt es jetzt [LBE Privacy Guard](#) (siehe rechtes Bild). Der passt im Hintergrund auf – und sobald eine App auf etwas "kritisches" zugreifen will, erhält der Benutzer eine entsprechende Warnung. Nun kann dieser entscheiden: Darf, darf nicht? Nur diesmal – oder ist es eine dauerhafte Entscheidung? Im letzteren Falle setzt man das entsprechende



Häkchen, und erhält in der gleichen Situation beim nächsten Mal keine Warnung mehr.

Achja: Versteht sich von selbst, dass eine solche App natürlich [root](#) voraussetzt...

ANHANG

Begriffserklärungen

Hier möchte ich einige Begriffe kurz erklären, da ich danach desöfteren gefragt wurde. Wie gewohnt, versuche ich mich dabei kurz zu fassen – und verweise für Details auf "externe Quellen".

2G

Gemeint ist hier die Datenübertragung der "zweiten Generation" (die erste ist bereits nicht mehr verfügbar). Hierzu gehören sowohl [GPRS](#) als auch [EDGE](#).

3G

Da dies ein Android-Handbuch ist, denken wir jetzt mal nicht an das iPhone 3G. Obwohl der Zusatz auch hier bedeutet: Dritte Generation. Gedacht ist in unserem Fall jedoch an die Datenübertragung mittels [UMTS](#) bzw. [CDMA](#).

ADB

Die **A**ndroid **D**ebug **B**ridge ist Bestandteil des Android-[SDK](#). Anders als der Name es nahelegt, ist ADB für mehr als nur das [Debuggen](#) gut. so lässt sich hiermit ein Android-Gerät steuern und kontrollieren, Dateien können übertragen, installiert oder auch gelöscht werden, und mehr.

Android

Hierfür zitiere ich einmal kurz das Wesentliche aus dem passenden [Wikipedia-Artikel](#):

Android ist ein Betriebssystem wie auch eine Software-Plattform für mobile Geräte wie Smartphones, Mobiltelefone, Netbooks und Tablets, die von der [Open Handset Alliance](#) entwickelt wird. Basis ist der Linux-Kernel 2.6. Android ist freie Software und quelloffen.

[...]

Als erstes Gerät mit Android als Betriebssystem kam am 22. Oktober 2008 das HTC Dream unter dem Namen T-Mobile G1 in den Vereinigten Staaten auf den Markt. Dass bereits dieses erste Gerät auf das Global Positioning System zugreifen konnte und mit Bewegungssensoren ausgestattet war, gehörte zum Konzept von Android.

Also grob zusammengefasst: Speziell für mobile Geräte – Linux unten drunter, "eine Art Java" obendrauf, und ganz zuoberst laufen unsere lieben Apps.

Android Versionen

Der kleine Andy (der grüne Roboter Androide) ist ein Süßer. Entsprechend hat Google auch die Namen der Android-Versionen recht süß gewählt. Kommen also entsprechende Kuchen-Varianten ins Spiel, zielt das i.d.R. auf eine Android-Version:

Version	Name	wichtigste Neuerungen
1.1		
1.5	Cupcake	AutoRotate, Bildschirmtastatur, Videos

Version	Name	wichtigste Neuerungen
1.6	Donut	VPN, Verbesserungen bei der Energieverbrauchssteuerung, TTS, Gesten
2.0	Eclair	Digitalzoom, Blitzlicht, Exchange, Bluetooth 2.1
2.1	Eclair	WebKit-Erweiterungen (HTML5, GeoLocation, u.a.)
2.2	Froyo (Frozen Yoghurt)	sparsamerer Kernel, mehr RAM nutzbar, Tethering , App2SD , Bluetooth-Sprachwahl
2.3	Gingerbread	DualCore, NFC , SIP
3.0	Honeycomb	Optimierungen für Tablets
3.1	Honeycomb	USB Host-Modus
3.2	Honeycomb	Optimierungen für Tablets
4.0	Ice Cream Sandwich	Data-Tracking, Screenshots

Wen die Unterschiede zwischen den einzelnen Versionen interessieren, der schaue doch einfach mal bei [Wikipedia](#) vorbei...

AndroidPIT

[AndroidPIT](#) ist eine [Community](#), in der sich Android-Nutzer austauschen. Wo sie aber auch in Blogs, Wiki, Foren und Testberichten fundierte Informationen erhalten. Kurzum: Wo sie eine Anlaufstelle für alle Fragen rund um Android finden.

Habe ich was falsches gesagt? Es ist *DIE* Android-Community. Zumindest in Deutschland. Nämlich die größte. Und (ja, das mag jetzt subjektiv sein) natürlich die beste. Aber was verbirgt sich hinter diesem seltsamen Namen?

Da steckt der Name "Android" drin, den wir ja gerade geklärt haben. Bleiben die drei Großbuchstaben am Ende: PIT. Was heißen die denn nun? **P**rogramme, **I**nformationen, **T**ests? **P**izza **I**m **T**iefkühlfach? (Unserem Apple-Stevie zufolge müsste das "P" ja sicher für "**P**orno" stehen – doch findet sich eh kein Regisseur, der einen solchen im **T**iefkühlfach dreht). Ich bin sicher, daraus ließe sich eine Preisfrage machen – und der Gewinner bekommt dieses Buch...

Na? Erraten? Ich helfe mal ein wenig nach: Denken wir mal an Autos. Ganz schnelle. F1 (Hilfe? Nee, Formel-1 meine ich). Und genau: "Pit stop", der Boxen-Stopp. Auftanken, und weiter geht's!

Damit wäre der Name geklärt. Weitere Informationen gibt es u.a. in der [FAQ](#).

API

Kurzform (oft gesprochen, wie man es schreibt – aber auch als Abkürzung buchstabiert) steht für **A**pplication **P**rogrammers **I**nterface. Gemeint ist hier eine definierte Schnittstelle, über die Informationen bezogen werden können. Die Android-API bietet auf diese Weise z. B. Informationen über Akkustand u.a.m.. So muss nicht jeder Programmierer das Rad neu erfinden.

APK-Datei

Die Abkürzung steht für **A**ndroid **P**ackage, und da drin befindet sich in der Regel eine App zur Installation unter Android. Da diese Apps ja in Java geschrieben sind, verwundert es sicher nicht, dass das APK Format eine "Abwandlung" des JAR (**J**ava **A**Rchive) ist, und sich somit mittels WinZip & Co. ein Blick in selbige werfen lässt...

Datei-Manager unter Android erkennen diese Packages natürlich, und bieten an, die enthaltene App zu installieren.

APN

Kürzel für **A**ccess **P**oint **N**ame (zu Deutsch: Name des Zugangspunkt). Gemeint ist in der Praxis mitnichten nur der Name, sondern vielmehr der komplette Datensatz. Siehe auch [mobiles Datennetz](#) für eine detailliertere Beschreibung, sowie [APNs](#) für eine nach Netzanbietern sortierte Liste von APN-Definitionen.

App

Kurzform für *Application*. Wird auch im Deutschen ("**Neudeutsch**": Applikation) verwendet, da "Anw" einfach blöd klingt. Denn nichts anderes bedeutet das englische Wort *Application*: Anwendung.

Im Zusammenhang mit Smartphones aller "Colour" (also Früchte wie auch KGMs, kleine grüne Männchen) hat sich die Kurzform "App" eingebürgert – "Application" (oder im Deutschen "Anwendung") wird hier eher selten verwendet.

App2SD

Das hat nix mit dem abendlichen "Jezz abba App ins Bett" zu tun – sondern vielmehr mit der Frage: "Wie kann ich mehr Apps installieren, als in den internen Speicher passen?". Dazu gibt es mehrere Ansätze, die unter dem Begriff "App2SD" zusammengefasst sind:

App2SD: Mit Froyo eingeführt. Hier werden Teile der App auf die (einzige) [Partition](#) der SD-Karte verschoben, wobei die App dies unterstützen muss. Mit Widgets klappt dies in der Regel nicht – hier kommt es zu Abstürzen, da die SD-Karten-Partition bei Anschluss an den PC via USB auf dem Androiden nicht mehr zur Verfügung steht.

App2SD+: Gibt es mit einigen Custom-ROMs. Hier wird das Widget-Problem dadurch umgangen, dass eine eigene Partition auf der SD-Karte verwendet wird. Android gibt bei USB-Anschluss lediglich die erste Partition frei, die zweite mit den Apps bleibt somit unangetastet. Wie bereits geschrieben: Benötigt [root](#) und [Custom-ROM](#), wobei auch nicht jedes Custom-ROM App2SD+ anbietet (Cyanogen zum Beispiel nicht).

Link2SD: Im Prinzip wie App2SD+ – nur bedarf es keines Custom-ROMs: Die App wird hier zunächst auf die zusätzliche Partition verschoben, und sodann ein sogenannter "symbolischer Link" dorthin im internen Speicher angelegt. Somit wird die App gefunden, als wäre sie im internen Speicher – obwohl sie ganz woanders steckt... Benötigt [root](#) und eine zweite Partition auf der SD-Karte.

Baseband

Oh, oder auch "Radio-ROM" bzw. "Radio-Image" wird es gern genannt. Das ist quasi die eigentliche Geräte-Firmware. Hat weniger direkt mit Android, als vielmehr mit der Hardware zu tun – und initialisiert letztere, so dass sie von ersterem genutzt werden kann. Also sowas ähnliches wie das BIOS beim PC. Und genau wie dieses, befindet es sich i.d.R. auf einem separaten Chip.

Das Teil bootet also die Hardware, und übergibt dann an den eigentlichen Bootloader, der sich dann um Android kümmert. Daher muss das Radio-

Image auch zum verwendeten [SPL](#) passen. Flasht man hier das falsche Image, hat man einen Ziegelstein (engl.: "Brick") oder Briefbeschwerer, aber kein brauchbares Telefon mehr...

Bootloader

Sozusagen der "zweite Teil" nach dem [Baseband](#) (daher auch "SPL" oder "Secondary Program Loader" genannt). Bleiben wir weiter bei den Hinke-Vergleichen, sind wir hier etwa im "Boot-Manager" (Lilo, Grub) gelandet (davor wäre noch der "MBR" oder "Master Boot Record" auf dem PC – das wäre hier der "IPL", der "Initial Program Loader" – der ist bei Androiden in Hardware gegossen, und daher nicht veränderbar).

Aber das wäre jetzt nur sehr grob und ungenau, denn hier steckt mehr drin: Der Android-Bootloader, sowie weitere Boot-Optionen wie das Recovery-Menü, Fastboot, u.a.m.

Brick

In der Regel das Lebensende eines Androiden – der dann nur noch als Briefbeschwerer o.ä. erhalten kann. Wörtlich heißt das zwar "Ziegelstein", aber das würde im Deutschen u.U. zu meilenweisen Verwechslungen mit gewissen Androiden aus dem Hause Motorola führen...

Was sich Google dabei dachte, als es die gleichnamige [Permission](#) einführte, sei der Fantasie anheim gestellt...

Wie verwandelt man einen Androiden nun in einen "Brick"? Dazu werde ich hier keine Schritt-für-Schritt-Anweisung geben (da wenig sinnvoll). Nur soviel sei gesagt: In etwa 95% aller Fälle hängt das mit dem [Flashen](#) eines zum Gerät inkompatiblen [RUU](#) zusammen. Vermeiden lässt sich solches also durch gründliches Lesen der Anleitungen und prüfen des "Zubehörs" **vor** dem "Brutzeln".

CDMA

[CDMA](#) (**C**ode **D**ivision **M**ultiple **A**ccess) ist ein Mobilfunkstandard der dritten Generation (3G), der primär in Amerika und Teilen von Asien und Afrika für den Betrieb von Mobilfunknetzen Anwendung findet. Die maximal möglichen Datenraten reichen hier fast bis an die 5 MBit/s.

CupCake

Eine Tasse Kuchen, klar. Oder doch eher eine [Android-Version](#)? Genau, nämlich 1.5.*.

Dalvik

Dafür muss ich ein klein wenig ausholen: Android besteht, vereinfacht gesagt, aus einem Linux-Kernel, auf dem eine spezielle Java-Version läuft. Letzteres ist die sogenannte *Dalvik VM* (wobei "VM" für "Virtual Machine" steht). Android Apps sind also in Java geschrieben.

Für die Ausführung der App wird der Java Code in einen sogenannten "Byte Code" übersetzt, der optimal auf die Hardware (und Android-Version) angepasst ist. Damit dies nicht bei jeder Ausführung der App geschehen muss, passiert diese "Übersetzung" unmittelbar nach der Installation der App – und der Byte-Code wird im sogenannten *Dalvik Cache* abgelegt. Da dies nach der Installation eines "neuen Systems" für alle Apps geschehen muss,

dauert auch der erste Start nach der Neuinstallation ein wenig länger (dafür geht die Ausführung der Apps nachher entsprechend schneller).

Bei der Installation eines neuen [ROMs](#) muss aus genannten Gründen (optimale Anpassung ans System) der *Dalvik Cache* neu aufgebaut werden. Dafür gibt es im [Recovery-Menü](#) einen extra Menüpunkt – aber das ist im entsprechenden Kapitel auch erklärt.

Debuggen

Wörtlich "entkäfern". Ein Computer-antiker Begriff, der noch aus einer Zeit stammt, in der "Programmierung" durch das Ziehen von Drähten, Stecken von Röhren und Löten von Leiterbahnen stattfand. Da war der "Bug" im "Programm" nämlich durchaus wörtlich zu nehmen – wenn ein verbrutzelter Käfer für einen Kurzschluss sorgte...

Die Käfer sind mittlerweile zu groß geworden (oder vielmehr die Chips zu klein), trotzdem haben sich beide Begriffe gehalten: "Bug" für einen Fehler im Programm, und "Debuggen" für die Suche nach und das Entfernen desselben.

Derivat

Eine Abspaltung (auch Fork; englisch fork = Gabel, üblicherweise als Maskulinum verwendet) ist in der Softwareentwicklung ein Entwicklungszweig nach der Aufspaltung eines Projektes in zwei oder mehr Folgeprojekte, wobei Teile des Quelltextes und seiner Historie kopiert werden und dann unabhängig von dem ursprünglichen Projekt weiterentwickelt werden. Mit Bezug auf das Urheberrecht wird auch von einem Derivat (derivativ, lat.: derivare = ableiten) gesprochen. ([Wikipedia](#)).

DLNA

Die [DLNA](#) (**D**igital **L**iving **N**etwork **A**lliance) ist eine internationale Vereinigung von Herstellern von Computern, Unterhaltungselektronik und Mobiltelefonen mit dem Ziel, die Interoperabilität von informationstechnischen Geräten unterschiedlicher Hersteller aus dem Bereich Heim- und Eigengebrauch sicherzustellen. (Wikipedia). Klingt nach viel ("Oh, da kann ich meine Waschmaschine, das Licht, den Rollladen...") – beschränkt sich meist jedoch auf die Steuerung von Bild- und Tonkonserven kompatibler Geräte (auf denen dann auch "DLNA" steht).

Donut

Die Lieblingsspeise eines gewissen Homer Simpson – aber was soll das hier? Na klar, süßes Backwerk, und daher folgerichtig eine [Android-Version](#) (1.6).

Downgrade

Kurz und schmerzlos: Das Gegenteil eines [Upgrades](#). Also ein (oder mehrere) Versionsschritt(e) rückwärts.

Warum man so etwas tun sollte/möchte? In der Regel natürlich gar nicht. Aber wenn man nach einem Upgrade wesentlich schlechter dran ist als davor, und auch kein [Update](#) zur Abhilfe in Sicht – dann bleibt einem nicht viel anderes übrig.

Ein weiterer Grund ist häufig, dass man sein Gerät [rooten](#) möchte, das aber mit der aktuellen Firmware nicht möglich ist...

DroidDream

DroidDream ist ein Trojaner, der eine Sicherheitslücke in Android-Versionen vor [Froyo](#) ausnutzt. Er sendet private Daten an einen Server im Netz und installiert eine "Hintertür" im Android-Gerät, durch die Code aus dem Netz nachgeladen werden kann.

Bekannt wurde dies Anfang März 2011: Etwa 50 infizierte Apps wurden im Google Market aufgespürt und aus diesem entfernt. Dies war auch einer der seltenen Fälle, dass Google von der Möglichkeit Gebrauch machte, diese Schadsoft aus der Ferne von betroffenen Android-Geräten zu löschen.

Weitere Informationen finden sich im Netz – u.a. [hier](#).

Eclair

Klingt nach einem süßen Backwerk – und lässt daher korrekt auf eine [Android-Version](#) (2.0./2.1.*) schließen.

EDGE

[EDGE](#) steht für **E**nhanced **D**ata Rates for **G**SM **E**volution. Als Erweiterung zu [GPRS](#) dient es der Datenübertragung, wobei es die maximal mögliche Datenrate auf 384 kbit/s mehr als verdoppelt. Dennoch gehört es noch zur Kategorie [2G](#). In der Statusleiste von Android-Geräten macht es sich durch ein "E" bemerkbar.

Fastboot

Der Name ist zunächst ein wenig irreführend – handelt es sich hier doch nicht um die Möglichkeit, das Android-Gerät schneller einsatzbereit zu haben. *Fastboot* hat eigentlich mit dem installierten Android-Betriebssystem nicht einmal direkt etwas zu tun...

Zu finden ist ein Fastboot-Eintrag gelegentlich im [Boot-Menü](#). Und gedacht ist es in erster Linie zum schnellen Bearbeiten von [Partitionen](#) via USB. Dazu wählt man am Android-Gerät diesen Punkt aus, und kann dann vom PC aus mit der entsprechenden Software passende Befehle absetzen – etwa um die Daten auf einer Partition zu löschen, mit einer Image-Datei zu überschreiben, oder schlicht das Gerät neu zu starten.

Flaschen

A: Behälter für Bier, Cola, Wein, u.a.m.

B: Taugenichtse, Tagediebe, Apfeldiebe...

C: Gesucht war bestimmt eher der Begriff [Flashen](#), gelle?

Flashen

Foto: Benutzung des Blitzlichts (engl.: flash)

Android: Den Androiden mit einem neuen [ROM](#) versehen, indem entweder ein reguläres [Update](#), oder ein Custom-ROM installiert wird. Der Name rührt daher, dass hier die Daten größtenteils im "internen Speicher", dem sogenannten "Flash Speicher", landen.

FOTA

Ein [OTA](#)-Update der Firmware. Auf einigen [Custom-ROMs](#) läuft auch ein Prozess namens *FOTA kill*, der eben selbiges (insbesondere seine Verfügbarkeits-Meldungen) verhindern soll. Die machen ja da auch keinen Sinn...

Froyo

Eigentlich *Frozen Yoghurt*: Eine [Android-Version](#) (2.2.*)

GingerBread

Eine [Android-Version](#) (2.3.*)

GingerBreak

In Anlehnung an den Namen [GingerBread](#) ist dies zum einen der Name eines Algorithmus als auch einer App zur Erlangung von [Root](#)-Rechten unter Android > 2.2.1.

GPRS

[GPRS](#) steht für **General Packet Radio Service**. Es dient der Datenübertragung, und ist aktuell wohl die langsamste Fassung davon. GPRS gehört in die Kategorie [2G](#), zusammen mit der Erweiterung [EDGE](#). Die maximal mögliche Datenrate beträgt bei *GPRS* 171,2 kbit/s. Zeigt die Statusleiste eines Androiden ein "G", muss sich dessen Besitzer hiermit begnügen...

Hardreset

Auch [Rücksetzen auf Werkseinstellungen](#) genannt: Wiederherstellung des Auslieferungszustandes.

HBoot

So nennt HTC seinen [Bootloader](#).

HSDPA

[HSDPA](#) (**H**igh **S**peed **D**ownlink **P**acket **A**ccess) ist eine Erweiterung des [3G](#) Mobilfunkstandards [UMTS](#), der Datenraten von bis zu 14 MBit/s ermöglicht. Wird manchmal auch gern als "3,5G" oder "3G+" bezeichnet, und macht sich in der Statusleiste eines Androiden durch ein "H" bemerkbar.

Ice Cream Sandwich

Hmmm, wieder etwas Süßes? Richtig: Eine [Android-Version](#), nämlich 4.0

Image

Die Fotos, die mit der Kamera Deines Androiden gemacht wurden, sind hier *nicht* gemeint. Die nennt man nämlich *Pictures*.

Ein **Image** nennt man das Speicher-Abbild einer [Partition](#) (für Windows-User: Das, was sich hinter einem Laufwerks-Buchstaben verbirgt; eine Partition kann schließlich wie ein eigenständiges Laufwerk behandelt werden). Im Zusammenhang mit Android werden Images häufig in folgenden Kontexten genannt:

- **[Nandroid-Backup](#)**: Hier werden Images von allen Partitionen erstellt
- **[Update.zip](#)**: Enthält nicht selten ein (oder mehrere) Image(s)
- **[Custom-ROMs](#)** kommen auch oft als Images daher
- **[Fastboot](#)** kann zum [Flashen](#) verschiedener Images verwendet werden

Kernel

Da steckt das Wort "Kern" drin, genau. Wenn wir hier vom "Kernel" sprechen, meinen wir den "Betriebssystem-Kern", den "[Linux-Kernel](#)". Das ist, vereinfacht ausgedrückt, eine Abstraktions-Schicht: Unten speziell an die

jeweilige Hardware angepasst, stellt der *Kernel* "oben" eine einheitliche Schnittstelle ([API](#)) für die Software zur Verfügung.

Bei Android läuft auf dem *Linux-Kernel* die [Dalvik-VM](#) (eigentlich je eine pro App), und in der *Dalvik-VM* sodann die [App](#).

Launcher

Der *Launcher* ist sozusagen die "Grafische Benutzeroberfläche" (GUI) von Android – das, was nach dem Entsperren des Bildschirms angezeigt wird. Anders als bei Windows (und eher ähnlich wie bei Linux) gibt es hier nicht *den* Launcher, sondern eine ganze Reihe von Alternativen: Angefangen von "besonders Ressourcen-schonend" bis hin zu "mit allen (un)möglichen Features". Mehr Details dazu finden sich im Kapitel zum [Home-Screen](#).

Nandroid Backup

Ein vollständiges System-Backup, welches sich z. B. aus dem [Recovery-Menü](#) heraus erstellen und auch wieder herstellen lässt. Hier werden nicht einzelne Dateien gesichert, sondern ein Abbild ("Image") des gesamten Systems wird angelegt. Es ist also ein "Alles-oder-Nichts": Die Wiederherstellung einzelner Dateien ist hier nicht vorgesehen (und auch nicht ohne weiteres möglich).

Insbesondere bevor man ein [Custom-ROM](#) einspielt, aber auch generell vor einem System-Update sollte ein Nandroid-Backup angelegt werden. Es ist natürlich auch sonst immer eine gute Idee, ein komplettes Backup zur Hand zu haben.

NFC

Nein, nicht **N**ashville **F**ried **C**hicken, sondern [Near Field Communication](#). Dient zum Ultra-Kurzstrecken-Datenaustausch (Reichweite also noch kleiner als bei Bluetooth), und soll u.a. zum "Bezahlen mit dem Handy" genutzt werden. Aber auch Dinge wie "Handy als Autoschlüssel" oder "Handy als Fahrkarte" etc. sind denkbar (und schon gedacht worden).

OTA

Da liegt was in der Luft... Denn OTA steht für "**O**ver **T**he **A**ir". Ja was denn? Beim Rundfunk ist es "On The Air" und heißt Musik. Bei Phil Collins "In The Air Tonight". Und bei Android ein "(komplettes) Over The Air Update" – also ein [Kotau](#), sozusagen. Die Frage wäre da nur, wer dabei der Kaiser ist...

Also, kurz gefasst: Beim OTA werden Software-Updates des Herstellers/Providers über das Funknetz des letzteren verteilt.

Partition

Eine Partition ist ein zusammenhängender Bereich auf einem Datenträger (unter Windows häufig mit einem Laufwerksbuchstaben verbunden).

Auf einem Android-System sind immer mehrere Partitionen in Benutzung, auch wenn nur der interne Speicher zur Verfügung steht (und keine SD-Karte eingesteckt ist): So ist das "/system" in der Regel nur lesend eingebunden (um Änderungen im Betrieb zu verhindern), während für [Apps](#) und Daten eine eigene Partition ("/data") bereitsteht.

Die SD-Karte beinhaltet meist nur eine (in der Regel unter "/sdcard" eingebundene) Partition. Es sind aber auch hier mehrere Partitionen möglich (und werden z. B. mit [App2SD+/Link2SD](#) auch verwendet) – wobei Android bei Anschluss an den PC via USB nur jeweils die erste Partition davon freigibt.

Weitere Details können z. B. bei [Wikipedia](#) nachgelesen werden.

Provider

In der Regel ist hier der "Netzanbieter" gemeint (also E-Plus, T-Mobile, A1 & Co).

Zum anderen könnte aber auch ein Dienst des Android-Systems gemeint sein: So stellt etwa der *Location Provider* über eine [API](#) den Apps die aktuelle Position bereit...

RAM

Diese drei Buchstaben stehen für **R**andom **A**ccess **M**emory – also Speicher, auf den man nach belieben an beliebiger Stelle zugreifen kann. Im Gegensatz nicht etwa zu [ROM](#), sondern zu Dingen wie Bandlaufwerken (jaja, so alt ist der Begriff schon), bei denen man sich erst mühsam vom Start zur gewünschten Position (linear) vortasten muss.

Sowohl auf PCs wie auch auf unseren Androiden ist damit meist der Arbeitsspeicher gemeint, in den die Programme/Apps zur Ausführung geladen werden. Üblicherweise ist dies der Bereich, der generell zu klein ist... oder von dem man halt nie genug haben kann...

Recovery Menü

Ein separater Bereich des Boot-Menüs (siehe [Bootloader](#)), aus dem heraus verschiedene Operationen wie [Nandroid-Backup](#) oder auch das Bereinigen des [Dalvik-Caches](#) möglich sind.

In das *Recovery Menü* gelangt man in der Regel durch eine spezielle Tasten-Kombination beim Einschalten. Diese ist aber zumindest von Hersteller zu Hersteller unterschiedlich. Bei HTC ist es üblicherweise das Halten der "Leiser-Taste" während des Einschaltens. Bei Motorolas Milestone muss die Kamera-Taste beim Einschalten gedrückt gehalten, und anschließend die "Lauter-Taste" betätigt werden. Und so weiter. Bei Bedarf also am besten im Forum erlesen/erfragen.

Einfacher geht es mit dem bereits im Kapitel [Custom-ROM](#) genannten *ROM Manager*: Diese App erlaubt auch einen direkten Boot ins Recovery-Menü...

Repository

Ein Repository (engl. für Lager, Depot, Quellen oder Archiv), auch Repositorium, ist ein verwaltetes Verzeichnis zur Speicherung und Beschreibung von digitalen Objekten. Bei den verwalteten Objekten kann es sich beispielsweise um Programme (Software-Repository), Publikationen (Dokumentenserver), Datenmodelle (Metadaten-Repository) oder betriebswirtschaftliche Verfahren handeln. Häufig beinhaltet ein Repository auch Funktionen zur Versionsverwaltung der verwalteten Objekte. ([Wikipedia](#))

Reset

Je nachdem: Siehe [Softreset](#), [Hardreset](#), sowie [Wipe](#).

ROM

Richtig **o**ffensichtlicher **M**ist ist diese **r**eal **o**fferierte **M**ehrdeutigkeit: Manchmal hat man den Eindruck, er wurde nur zur Verwirrung der Massen eingeführt. Wer einmal den Namen für eine Android-Komponente nicht kennt,

sagt einfach "ROM". Klingt, als wüsste man voll Bescheid – und die Chance, dass das auch noch Sinn ergibt, ist verdammt hoch...

Aber im Ernst: Worum geht es hier? Eigentlich steht der Begriff "ROM" für **Read Only Memory** – also Speicher, auf den ausschließlich lesend zugegriffen werden kann. Ja, richtig: So wie bei CD-ROM, da steckt das ja auch drin. Nur bei Android, da kann das alles mögliche sein. Nicht selten sachlich falsch – aber wen kümmert's? Schauen wir uns also die einzelnen Bedeutungen einmal an:

Systemspeicher: Teile des Android-Systems werden in der Tat "nur lesend" eingebunden. Unter anderem eine Schutzmaßnahme, um Veränderungen zu erschweren (damit wir die dusseligen Apps, mit denen uns die Hersteller/Provider "beglücken", nicht einfach löschen können). So heißt es z. B. in den Spezifikationen des HTC Wildfire: "384 MB RAM; 512 MB ROM". Nonsens(e): Ein Blick hinter die Kulissen offenbart, dass nur 250MB read-only (/system) eingebunden sind. Die restlichen 250MB "ROM" stehen zur Installation von Anwendungen zur Verfügung. Read-only? Mitnichten. De facto kann der ganze Bereich jederzeit schreibbar gemacht werden, sofern man [root](#) hat. Also eher irreführend – richtiger müsste es hier heißen: "interner Speicher", oder – in Abgrenzung vom [RAM](#) – "interner Flash-Speicher".

Das System selbst: Um die Verwirrung komplett zu machen, wird auch hier gern von "ROMs" gesprochen (siehe [Custom ROMs](#)). Das hat schon Tradition: Auch bei älteren Spiele-Konsolen sprach man davon, "ein ROM zu laden". Hier war es aber "seinerzeit" tatsächlich eine Cartridge – also ein Speicher-Chip, den man an das Gerät ansteckte. Später kamen dann die Emulatoren, welche die "antiken Geräte" auf moderner Hardware emulieren können. Und hier kommt diese "Cartridge" natürlich in Form einer Datei daher. Den Begriff "ein ROM laden" hat man beibehalten. Und schließlich weiter übertragen...

ROM Kitchen

ROM-Knast? Nein, gemeint ist der englische Begriff "Küche" (also wenn man schon "Knast" hat). In einem *ROM Kitchen* kann der Anwender sich sein eigenes [ROM](#) kochen. Natürlich setzt dies [root](#) voraus (sonst kann man das fertige "Gericht" nicht servieren/installieren), und ist etwas Gerätespezifisches.

Neben "vollständigen ROM-Küchen" gibt es auch noch welche, die sich auf Elemente der Gestaltung beschränken, wie z. B. das [UOI](#) (Ultimate Online Kitchen, Beschreibung und Video findet sich bei den [XDA-Developers](#)). Hier tauscht man lediglich einzelne Elemente (Batterie-Anzeige in der Statusbar, Icons, etc.) eines bereits installierten ROMs aus. Natürlich braucht auch dieses [root](#), da sonst kein Austausch von Systemdateien möglich ist.

root

Aus Herstellersicht: Die Wurzel allen Übels. Objektiv betrachtet: Der Administrator (auch "SuperUser") eines Linux-Systems. Der darf alles, und kann alles (kaputtmachen auch, ja). Näheres ist im Kapitel [root](#) ausführlicher beschrieben.

RUU

Radio Unit Update: Eine Art [update.zip](#) für das [Radio-Image](#), also ein "Firmware-Upgrade".

ROM Upgrade Utility: Wie der Name bereits sagt, ein Utility zum Upgrade des [ROM](#), welches entweder vom Hersteller oder von Drittanbietern zur Verfügung gestellt und vom PC aus installiert wird.

Wenn nicht ganz klar ist, was von beidem gemeint ist, ist es in der Regel das erste – wobei das durchaus mit dem zweiten identisch sein kann, da die Begriffe oftmals gleichbedeutend verwendet werden. Was mancherorts als "ROM Upgrade Utility" bezeichnet wird, ist nämlich nichts anderes als das Update des Radio-Images...

SDK

Das SDK ist die Grundlage für die Android App Entwicklung und liegt für die jeweilige Version von Android vor. Es ist aber nicht nur dafür verwendbar, sondern bringt auch einige brauchbare Tools wie [Fastboot](#), den [Dalvik](#) Debug Monitor sowie [ADB](#) mit.

Ergänzt man das Ganze noch um [Eclipse](#), kann es mit der Entwicklung von [Apps](#) losgehen!

SIP

Das Kürzel SIP steht für [Session Initiation Protocol](#). In unserem Zusammenhang kann es quasi als Alias für "Internet-Telefonie" bzw. "IP-Telefonie" betrachtet werden, da es bei dieser zum Einsatz kommt.

S-OFF

Mit S-OFF (kurz für Security off) ist der volle Zugriff (lesen und schreiben) auf die System-[Partition](#) während des Betriebs des Android-Gerätes gemeint. Üblich ist dieses "secu_flag" nur bei HTC, somit kann man dieses bei nicht-HTC-Geräten ignorieren.

Softreset

Ein "weiches" herunterfahren des Systems, wenn nichts mehr geht – vergleichbar mit Strg-Alt-Entf am PC. Siehe [Weiches Zurücksetzen](#) für Details.

Tethering

Die Leinen los! Ja, so kam es einigen vor, als das mit *Froyo* zur Standard-Funktionalität wurde: Mittels [Tethering](#) lässt sich die mobile Internet-Verbindung mit anderen Geräten teilen.

UMTS

[UMTS](#) (**U**niversal **M**obile **T**elecommunications **S**ystem) ist ein Mobilfunkstandard der dritten Generation, und wird deshalb umgangssprachlich auch einfach als [3G](#) genannt. Seine maximale Datenrate liegt – wie übrigens auch die der [2G](#)-Erweiterung [EDGE](#) – bei 384kBit/s. Schneller wird es mit der Erweiterung [HSDPA](#).

Unroot

Rückgängigmachen des sogenannten "rootens" (siehe [root](#)), also das System wieder vom root-Zugang befreien (und in den Hersteller-konformen Zustand zurücküberführen). Dies geschieht in der Regel durch das [Flashen](#) eines [RUU](#) (sauberste Variante) – wobei mittlerweile auch etliche rooting-Tools eine entsprechende "Undo-Funktion" anbieten.

Update

Das Ersetzen etwas Bestehenden durch etwas Neuere. Meist handelt es sich hier um eine Fehlerbereinigung von System oder Apps, aber auch neue/zusätzliche Features können damit einhergehen.

Update.Zip

Ganz offensichtlich eine Datei. Und ebenso offensichtlich will diese etwas aktualisieren – nur was?

Es handelt sich hier um ein "[flashbares](#) Update". Offizielle Firmware-Updates kommen meist unter diesem Namen daher. Und da das System eine solche Datei, so sie im Wurzel-Verzeichnis der SD-Karte liegt, als ein solches betrachtet, lässt sich auf diese Weise auch so einiges anderes ins System mogeln. Das nutzt z. B. [Titanium Backup](#) aus, wenn es ein update.zip erstellt.

Einspielen lässt es sich zum Beispiel über das [Bootmenü](#).

Upgrade

Ist mit einem [Update](#) ein Versionsprung verbunden (etwa von Eclair auf Froyo, oder von Froyo auf Gingerbread, oder gleich von Eclair auf Gingerbread), spricht man von einem Upgrade.

VoIP

Abkürzung für **V**oice **o**ver **I**P, oder auch "Internet-Telefonie". Aber so wie man heute simst (SMS schickt), googelt (mit Google sucht), und skypeet (mit Skype chattet oder telefoniert), so voipt man auch.

Wipe

Wörtlich übersetzt: Löschen. Das "was" ist hier allerdings die Frage. Und da kommt es darauf an, wen man fragt bzw. wie man den Wipe initialisiert.

Hardreset: Eine Form des Wipe ist mit diesem gleichbedeutend, denn sie löscht lediglich die "/data" [Partition](#). Das ist der Bereich, in dem die selbst installierten Apps sowie die Daten abgelegt werden.

Dalvik-Cache: Mit dem Wipe/Löschen des Dalvik-Caches erzwingt man eine Neu-Übersetzung des Programmcodes aller Apps. Damit geht kein Datenverlust einher: Lediglich der nächste Gerätestart dauert etwas länger.

Komplett-Wipe: Den gibt es in verschiedenen Versionen des [Recovery-Menüs](#). Sollte nur ausgeführt werden, wenn man auch wirklich weiß, was man da tut: Der löscht nämlich alle Daten von allen Partitionen, auch von "/system". Danach geht dann nichts mehr – es lässt sich lediglich ein neues [ROM](#) einspielen.

Häufig gestellte Fragen – und ihre Antworten

Google Account

Wozu brauche ich einen Google-Account?

Klar kann man seinen Androiden auch ohne Google-Account benutzen. Bei vielen Tablets hat man zunächst nicht einmal eine andere Wahl. Doch manche Dinge funktionieren einfach nicht ohne. Zum Beispiel der Google Market, über den die meisten Apps kommen. Auch die Synchronisation von Kontakten und Terminen ist an einen Google-Account gebunden – zumindest, wenn man die Bordmittel benutzen möchte.

Kann ich einen Account mit mehreren Geräten nutzen?

Das ist problemlos möglich. Und das Schöne dabei: Auch die mit dem Account gekauften Apps lassen sich auf all diesen Geräten parallel nutzen, denn der Kauf ist ja an den Account gebunden.

Darüber hinaus zeigt der neue [Web-Markt](#) bei jedem App an, mit welchem Gerät selbige kompatibel ist – und gibt dem Anwender dann die Möglichkeit, sie dort zu installieren. Auch Kalender und Kontakte werden dann auf allen mit diesem Account verknüpften Geräten synchronisiert (so man diese Funktion aktiviert hat). Kurzum: Die Sache scheint geradezu dafür ausgelegt.

Aufpassen sollte man allerdings mit Datensicherungen (auch diese lassen sich ja auf den Google-Servern speichern) – hier bin ich nicht sicher, wie diese auseinandergehalten werden.

Wie kann ich meinen Google-Account ändern?

Offensichtlich soll der Anwender dies nicht tun – denn die entsprechende Stelle in der Konfiguration lässt es nicht zu. Natürlich klappt das nach einer "Werksrückstellung", da dann ohnehin alles neu eingerichtet werden muss. Das ist aber nicht immer gewünscht: Auch alle Apps müssen dann nämlich neu installiert und konfiguriert werden.

Mit einem kleinen Trick geht es aber auch ohne "Werksrückstellung": Vom Home-Screen aus geht man dazu in die Maske "Einstellungen→Anwendungen→Anwendungen verwalten", und wählt nacheinander folgende Anwendungen an, um deren Daten und Caches zu löschen: GMail, Google Apps, GTalk (Google Talk), GTalk Services, IM und evtl. noch Google Storage (auch "Speicherplatz bei Google Mail" genannt; was nicht da ist, überspringt man halt). Jetzt hat der Androide seine "Kontoverbindung" vergessen, und man kann diese wieder neu einrichten – mit den gewünschten "anderen Daten", oder indem man einen neuen Account erstellt.

Google Market

Ich finde den Market auf meinem Gerät nicht!

Dies ist oftmals insbesondere bei Tablets "normal": Ein Android-Gerät muss bestimmte Voraussetzungen erfüllen, damit es mit den Google-Apps ausgeliefert werden darf (z. B. muss es über eine Kamera und ein GPS-Modul verfügen). Das heißt jetzt aber nicht, dass "Hopfen und Malz verloren" sind: Eine Suche in den einschlägigen Foren (z. B. im für das fragliche Gerät zuständige Hardware-spezifische Unterforum bei AndroidPIT), oder auch bei Google mit den Begriffen "*Google Apps*" <Gerätename>, führen nicht selten zu einer Lösung. Diese heißt dann in der Regel: Die gapps.apk händisch herunterladen, und auf dem Gerät installieren.

Warum finde ich die App im Market nicht?

Hier ist die Antwort ähnlich zur vorigen – nur dass es diesmal einen direkten Bezug zur App hat. Entweder hat der Entwickler was vergessen anzugeben (das ist nicht gerade selten), oder aber das fragliche Android-Gerät wird tatsächlich nicht unterstützt: Zu kleines Display (auf einem Motorola Flipout HD-Videos abspielen? Macht bestimmt Spaß), zu schwacher Prozessor (CPU-intensive Spiele wie Asphalt überfordern so manches kleine Gerät), oder die installierte Android-Version ist zu niedrig. Eine Anfrage beim Entwickler schafft hier häufig Abhilfe, sofern letztere möglich ist.

Hilfe, der Market spinnt!

Sowas tut der in der Tat: Downloads starten nicht oder werden nicht abgeschlossen, Installationen schlagen fehl, oder der Market wird erst gar nicht erreicht (Anmelde-Probleme). In der Regel liegt das in der Tat am Market bzw. den Google Servern, daher lohnt sich in solchen Fällen ein Blick auf deren [Monitor](#), das so genannte "Dashboard". Steht dort alles auf „grün“, zur Sicherheit noch im Forum vorbeigeschaut (bei AndroidPIT gibt es da auch eine Sektion für Market-Fragen), ob gerade wieder eine Epidemie ausgebrochen ist (und auch andere betroffen sind).

Ist auch das nicht der Fall, ist ein lokales Problem naheliegend. Jetzt gibt es mehrere Möglichkeiten, die einfach in dieser Reihenfolge abgearbeitet werden können. Nach jedem Schritt ist natürlich zu Prüfen, ob das Problem damit schon beseitigt ist – dann kann man sich die folgenden Schritte nämlich sparen:

- Im Android-Market auf die "Downloads" Seite gehen, lange (mindestens ein bis zwei Sekunden) auf die Fortschrittsanzeige "drücken", und aus dem sich nach dem Loslassen öffnenden Menü "Download abbrechen" auswählen. Dann erneut versuchen, die App zu installieren.
- Ein Geräteneustart (Herunterfahren, Ausschalten, evtl. zur Sicherheit noch den Akku rausnehmen, kurz warten, ggf. Akku wieder rein, neu Starten.
- Datum, Uhrzeit und Zeitzone manuell einstellen (kann später wieder auf Automatik zurückgestellt werden)

- Google Talk starten, sich dort abmelden, es wieder starten (und dann beenden)
- Unter "Einstellungen→Anwendungen→Anwendungen verwalten" den Market-Cache leeren
- An gleicher Stelle: "Stoppen erzwingen", dann "Daten löschen" (damit sind aber auch die Einstellungen für Auto-Updates zurückgesetzt).
- Letzter Ausweg: Zurücksetzen auf Werkseinstellungen

Unbekannter Fehler: -18

Dieser Fehler tritt nur im Zusammenhang mit App2SD auf, wenn eine zu aktualisierende App auf der SD-Karte installiert ist (oder eine neue dort installiert werden soll). Die einfachste Lösung ist hier: App wieder in den internen Speicher zurück verschieben, das Update erneut versuchen (sollte jetzt funktionieren), und anschließend optional wieder auf die SD-Karte damit.

Schlägt das Update noch immer fehl, oder handelt es sich um eine neu zu installierende App, kann auch folgendes helfen: Androiden per USB-Kabel an den PC anschließen, und die Karte als Laufwerk freigeben. Damit steht sie dem Android-System nicht mehr zur Verfügung, was eine Installation in den internen Speicher erzwingt.

Eine letzte Möglichkeit ist mit Vorsicht zu genießen: Mit einem Datei-Manager auf der SD-Karte den Ordner ".android_secure" (mit Punkt vorne!) suchen, und die darin befindliche Datei namens "smdl2tmp1.asec" löschen, dann die Installation bzw. das Update noch einmal versuchen.

Medien

Wie kann ich Dateien in der Mediengalerie ausblenden?

Einzelne Verzeichnisse (einschließlich deren Unterverzeichnisse) lassen sich "ausblenden", indem man in ihnen eine Datei mit dem Namen .nomedia (also mit Punkt vorne) anlegt. Dies lässt sich am einfachsten mit einem Dateimanager bewerkstelligen. Damit werden diese Verzeichnisse vom Medien-Scan ausgeschlossen – und ihre Inhalte somit in der Galerie nicht mehr angezeigt.

Wie kann ich eigene Töne als Klingelzeichen, Benachrichtigung, oder für den Wecker nutzen?

Dafür gibt es spezielle Verzeichnisse auf der SD-Karte:

Verzeichnis	Verwendung
alarms	Alarm-Töne
notifications	Benachrichtigungstöne
ringtones	Klingeltöne
ui	Tastatur-Klick-sounds etc.

Wo genau diese anzulegen sind, unterscheidet sich offensichtlich bei verschiedenen Android-Geräten. Der Wahrscheinlichkeit nach geordnet, sollte dies an folgenden Stellen geschehen:

- /sdcard/media/audio
- /sdcard/media
- /sdcard

In den in der Tabelle genannten Unterverzeichnissen werden die gewünschten Sound-Dateien dann platziert. Damit das System sie auch findet, muss der Medien-Scanner sie aber zunächst erfasst haben – also nicht wundern, wenn sie nicht sofort auftauchen!

Eine sichere Möglichkeit, einen Medien-Scan anzustoßen, besteht im Neustart des Androiden: Nach jedem Booten wird der Scanner nämlich automatisch ausgeführt. Wer nicht so lange warten will, lädt sich das kleine Tool [SDRescan](#) aus dem Market, und stößt den Scan damit manuell an. Anschließend sollten sich die neuen "Töne" als Klingelzeichen usw. auswählen lassen.

Wie kann ich meine SD-Karte wechseln?

Banale Antwort: Alte Karte raus, neue rein. Aber die Frage gilt sicher in erster Linie der Tatsache, dass man die darauf enthaltenen Daten auch gern auf die neue, größere Karte mitnehmen möchte. Auch das stellt aber kein Problem dar – solange auf der Karte nur eine einzige Partition ist (wer das jetzt von seiner Karte nicht weiß, bei dem ist dies mit ziemlicher Sicherheit der Fall).

Das Vorgehen ist in etwa folgendes:

1. Über "Einstellungen→SD-Karte und Telefonspeicher" zunächst die SD-Karte trennen (alternativ: Das Telefon abschalten)
2. Die "alte" SD-Karte entnehmen, und mit einem Kartenleser am PC anschließen
3. Sämtliche Daten in ein leeres Verzeichnis auf der Festplatte kopieren
4. Die Karte wieder sauber vom PC trennen, und die "neue" Karte mit dem Kartenleser anschließen
5. Die Daten wieder 1:1 auf die neue Karte kopieren
6. Karte sauber vom PC trennen, dem Kartenleser entnehmen, und in den Androiden einlegen.

Wer ganz auf "Nummer sicher" gehen möchte, macht das an einem Linux-PC (z. B. mit einer Live-CD wie Knoppix).

Fertig. Testen ob alles klappt. Das sollte es in 99% aller Fälle – und falls nicht, hat man bis zu einer Problemlösung ja noch immer die "intakte alte" Karte...

Weiteres

Ich habe mein Entsperr-Muster/Passwort vergessen!

Besteht gerade eine Netzwerk-Verbindung über 3G/2G (WLAN genügt eventuell nicht), gibt es noch Hoffnung: Einfach die Adresse des primären Google-Accounts (benutzername@googlemail.com – wobei "benutzername" natürlich entsprechend zu ersetzen ist) und das dazugehörige Passwort eingeben. Andernfalls (oder falls das fehlschlägt) bleibt nur noch ein Zurücksetzen auf Werkseinstellungen...

Ich fahre ins Ausland / Grenzgebiet. Wie vermeide ich Roaming-Kosten?

Die einfachste Möglichkeit ist natürlich, den Flugzeugmodus zu aktivieren – und ggf. bei Bedarf WLAN zu nutzen. Nicht immer ist dies aber das, was einem vorschwebt. Für das Datennetz gibt es einen einfachen Schalter in den Einstellungen unter "Drahtlos & Netzwerk": Hier lässt sich das Roaming für die Datendienste deaktivieren (per Default ist dies in der Regel auch die Voreinstellung). Darüber hinaus kann man hier auch den Netzbetreiber fest voreinstellen (manuell auswählen, anstatt automatisch wählen zu lassen). Somit bucht sich das Telefon auch für Gespräche nicht in ein fremdes Netz ein.

Wie kann ich ohne root Screenshots vom Android Handy erstellen?

Dies lässt sich am einfachsten bewerkstelligen, indem man eines der im Kapitel "[Das Android-Gerät vom PC aus verwalten](#)" benutzt, wie etwa den *PAW Server*. Screenshots erstellt man dann einfach vom PC aus.

Google Permissions – und was sie bedeuten

Normalerweise sieht man eine Kurzbeschreibung der Permission (z. B. bei der Installation einer App). Die technische Bezeichnung taucht selten im Klartext für den Anwender auf – man kann aber z. B. auch einen Blick auf das [Manifest](#) werfen, und da stehen sie im Klartext.

Nun werde ich aber hier nicht alle Permissions auflisten, das wäre einfach zu viel. Die findet man bei Interesse im [Entwickler-Handbuch](#) (allerdings auf Englisch). Ein paar ausgewählte, die doch häufiger einmal auftauchen könnten, möchte ich aber hier kurz erklären:

Permission	Erklärung
ACCESS_COARSE_LOCATION	<i>Ungefährer (netzwerkbasierter) Standort.</i> Hier kommt kein GPS zum Einsatz, sondern die Informationen von Funkmasten (Cell-ID) sowie WLANs
ACCESS_FINE_LOCATION	<i>Genauer (GPS-) Standort.</i> Exakte, per GPS ermittelte Standortdaten.
ACCESS_MOCK_LOCATION	<i>Falsche Standortquellen für Testzwecke.</i> Fake Location (falsche Standortdaten vom System anfordern). Für Testzwecke gedacht (z.B. im Emulator); wird aber scheinbar auch benötigt, um ein externes GPS Gerät nutzen zu können. Hier geht es nicht darum, einer anderen App eine Fake-Location unterzujubeln – das ginge allenfalls in Verbindung mit INSTALL_LOCATION_PROVIDER.
ACCESS_NETWORK_STATE	<i>Netzwerkstatus anzeigen.</i> Informationen über Netzwerke (besteht eine Verbindung, und wenn ja zu welchem Netzwerk?)
ACCESS_SURFACE_FLINGER	Zugriff auf die API des "Surface Flinger" (Teil des Medien-Frameworks unter Android: Stellt einen systemweiten "Oberflächen-Kompositor" bereit, der sich um das Rendering in Framebuffer-Devices kümmert - also Grafik, Grafik-Beschleunigung und so).
ACCESS_WIFI_STATE	<i>WLAN-Status anzeigen.</i> Informationen über WiFi-Netzwerke (besteht eine Verbindung, und wenn ja zu welchem Netzwerk? Welche Netzwerke sind verfügbar?)
ACCOUNT_MANAGER	<i>Als Konto-Manager fungieren.</i> App darf mit Konto-Authentifizierern interagieren. (für Systemanwendungen reserviert).
AUTHENTICATE_ACCOUNTS	<i>Als Kontoauthentifizierer fungieren.</i> Konto-Authentifizierungsfunktionen verwenden, Konten erstellen, Abrufen und Einstellen der zugehörigen Passwörter.
BIND_APPWIDGET	<i>Widgets auswählen.</i> Erlaubt einer App dem AppWidget-Service mitzuteilen, welche App

Permission	Erklärung
	auf die AppWidget-Daten zugreifen darf. Mit dieser Berechtigung kann anderen Anwendungen Zugriff auf persönliche Daten gewährt werden. Laut API-Referenz sollten nur sehr wenige Apps diese Permission benötigen.
BLUETOOTH	<i>Bluetooth-Verbindungen herstellen.</i> Zugriff auf bereits "authorisierte" Bluetooth-Geräte
BLUETOOTH_ADMIN	<i>Bluetooth-Verwaltung.</i> Bluetooth-Geräte "autorisieren" (also "pairen" und so). Eine mit dieser Permission ausgestattete App darf selbständig Bluetooth-Verbindungen aufbauen und etablieren – auch zu "wildfremden" Geräten.
CALL_PHONE	<i>Telefonnummern direkt anrufen.</i> Anruf ohne Bestätigung durch den Anwender tätigen. Wird z. B. für Kontakt-Widgets benötigt, wenn ein "Tapp" auf selbige direkt einen Anruf auslösen soll – macht aber bei einer Mal-App herzlich wenig Sinn. Gilt nicht für Notruf-Nummern.
CALL_PRIVILEGED	<i>Alle Telefonnummern direkt anrufen.</i> Wie CALL_PHONE, aber inklusive Notruf-Nummern ("Hallo, Polizei – Anwender ist gerade in Bank eingebrochen... Bitte Fußboden reparieren...")
CAMERA	<i>Fotos aufnehmen.</i> Vollzugriff auf die Kamera. Nebenwirkung: Diese App lässt sich nicht auf Geräten installieren, die über keine Kamera verfügen. Die API-Referenz schreibt sinngemäß: "Wenn die App auch ohne Kamera bedienbar ist, diese Permission nicht anfordern." Da sei die Frage erlaubt: Braucht man sie dann überhaupt, wenn es auch ohne geht?
CHANGE_CONFIGURATION	<i>UI-Einstellungen ändern.</i> Änderungen an der Umgebung durchführen. API-Ref nennt als Beispiel "Locale", also Ländereinstellungen wie Währung und Zeitformat. Beschreibung sehr vage.
CHANGE_NETWORK_STATE	<i>Netzwerkonnktivität ändern.</i> Netzwerk-Status ändern (also z.B. Verbindung trennen)
CHANGE_WIFI_MULTICAST_STATE	<i>WLAN-Multicast-Empfang zulassen.</i> WiFi MultiCast aktivieren. Damit können Datenpakete an mehrere Empfänger zeitgleich verschickt werden, ohne dass dies zusätzliche Bandbreite erfordert. Macht z.B. Sinn bei einem Streaming-Server, der mehrere Clients bedient ("Radio").

Permission	Erklärung
	Gleichzeitig ermöglicht dies auch den Empfang von Netzwerk-Paketen, die nicht an das eigene Gerät gerichtet sind (Netzwerk-Sniffer).
CHANGE_WIFI_STATE	<i>WLAN-Status ändern.</i> CHANGE_NETWORK_STATE für WiFi. Kann auch Änderungen an konfigurierten WLAN-Netzen vornehmen.
CLEAR_APP_CACHE	<i>Alle Cache-Daten der Anwendung löschen.</i> Cache beliebiger/aller Anwendungen leeren
CLEAR_APP_USER_DATA	<i>Alle Cache-Daten der Anwendung löschen.</i> Benutzerdaten beliebiger/aller Apps löschen (siehe Einstellungen→Anwendungen verwalten, der Button "Daten löschen" bei jeder Anwendung) – richtiger wäre also <i>Anwendungsdaten</i> löschen oder CLEAR_APP_DATA, das "User" verwirrt hier ein wenig.
DELETE_CACHE_FILES	(einzelne) Dateien aus dem Cache löschen
DELETE_PACKAGES	<i>Anwendungen löschen.</i> Apps entfernen/löschen/deinstallieren/wegmachen
DEVICE_POWER	Tiefgreifender (low-level) Eingriff in die Energieverwaltung (Power Management). Hat nicht direkt etwas mit "Power Off" zu tun, könnte aber bei Missbrauch durchaus zu selbigem führen...
DIAGNOSTIC	<i>Lese-/Schreibberechtigung für zu Diagnosegruppe gehörige Elemente.</i> Lese- und Schreibzugriff auf "diagnostic resources" – die API-Referenz beschreibt leider nichts genaueres.
DISABLE_KEYGUARD	<i>Tastensperre deaktivieren.</i> Tastensperre (inkl. deren Passwort-Schutz) deaktivieren, sodass der Bildschirm nicht mehr automatisch gesperrt wird. Sinnvoll z.B. bei Video-Apps und insbesondere bei Navis – und bei eingehenden Telefonaten.
EXPAND_STATUS_BAR	<i>Statusleiste ein-/ausblenden.</i> Status-Bar (Notification?) erweitern/kollabieren. Wohl die Lite-Version von STATUS_BAR
GET_ACCOUNTS	<i>Bekannte Konten suchen.</i> Liste konfigurierter Accounts abrufen (nur die Accounts, nicht die Zugangsdaten selber). Mit dieser Permission kann lediglich festgestellt werden, welche Accounts existieren.
GET_TASKS	<i>Laufende Anwendungen abrufen.</i> Informationen über laufende Anwendungen abrufen. Wird natürlich von Task-Managern und -Killern, aber auch von Akku-Statistik-

Permission	Erklärung
	Apps benötigt. "Böse Apps" können dies nutzen um auszukundschaften, wo sich lohnende Daten zum Klauen finden lassen.
INJECT_EVENTS	<i>Tasten und Steuerungstasten drücken.</i> "Generieren" und "ausführen" bestimmter Events, wie z.B. Benutzer-Eingaben. Die App kann also vermutlich andere Apps "fernbedienen".
INSTALL_LOCATION_PROVIDER	App will selber Ortsdaten bereitstellen ("Du bist jetzt <i>hier</i> "). Woher sie die nehmen will? Naja, vielleicht von einem Bluetooth-GPS o.ä.
INSTALL_PACKAGES	<i>Anwendungen direkt installieren.</i> Andere Apps installieren. Kann OK sein (App-Manager), muss aber nicht (Wallpaper etc. wollen vielleicht eher Schadsoft nachladen, wenn sie diese Permission anfordern)
INTERNET	<i>Uneingeschränkter Internetzugang.</i> Öffnen von Netzwerk-Sockets. Die App kann also beliebige Internet-Verbindungen herstellen. Wird von allen Apps gebraucht, die Werbung anzeigen wollen.
KILL_BACKGROUND_PROCESSES	<i>alle Anwendungen im Hintergrund schließen.</i> Hintergrund-Prozesse "töten", also beenden. Dabei kann es sich um die eigenen Prozesse handeln (was dem Anwender die Möglichkeit gibt, das Programm tatsächlich zu beenden, statt es nur in den Hintergrund zu schieben) – es können aber eben so gut fremde Prozesse beendet werden. I.d.R. handelt es sich dann um einen Task-Manager oder Task-Killer .
MANAGE_ACCOUNTS	<i>Als Konto-Manager fungieren.</i> Accounts/ Zugangsdaten <i>verwalten</i> – also auch verändern. Die Doku ist leider wieder einmal sehr vage – aber hier würden bei mir die Alarmglocken läuten.
MODIFY_PHONE_STATE	<i>Telefonstatus ändern.</i> Status der Telefonie anpassen: Power, MMI-Codes (z.B. Rufumleitung [de]aktivieren, Rufnummernübermittlung ein/ausschalten) etc. – jedoch nicht Anrufe tätigen. Allerdings kann das Netzwerk (zu einem anderen Anbieter, Roaming) gewechselt oder die Mobilfunkverbindung ein- bzw. ausgeschaltet werden, ohne dass der Benutzer davon informiert wird.
MOUNT_FORMAT_FILESYSTEMS	<i>Externen Speicher formatieren.</i> Externe Dateisysteme (SD-Karten etc.) <i>formatieren</i> (Vorsicht! Nach dem Formatieren ist das

Permission	Erklärung
	entsprechende Dateisystem leer, die (vorher) darauf befindlichen Daten sind weg!). Nix für Wallpaper, Spiele, etc.!
MOUNT_UNMOUNT_FILESYSTEMS	<i>Dateisysteme bereitstellen oder Bereitstellung aufheben.</i> Dateisysteme ein- und ausbinden. Toll für externe Festplatten am Telefon – gilt aber ebenso für SD-Karten.
PROCESS_OUTGOING_CALLS	<i>Abgehende Anrufe abfangen.</i> Ausgehende Anrufe beobachten, verändern oder abbrechen. Hm, könnte das einen Anruf bei der Mailbox ins Ausland weiterleiten? A propos Mailbox: Eine entsprechende Permission für eingehende Anrufe konnte ich bislang nicht entdecken.
READ_CALENDAR	<i>Kalenderdaten lesen.</i> Sollte klar sein: Alle Termine können damit gelesen werden.
READ_CONTACTS	<i>Kontaktdaten lesen.</i> Damit ist das Adressbuch fällig.
READ_FRAME_BUFFER	Zugriff auf die Frame-Buffer Daten (vereinfacht gesagt: Den Inhalt des Bildschirms). Erlaubt u. a. das Erstellen von Screenshots. Da stellt sich die Frage: Warum gibt es dann keine App zur Erstellung von Screenshots, die keine root-Rechte benötigt?
READ_HISTORY_BOOKMARKS	Erlaubt lesenden Zugriff auf Lesezeichen und Browserverlauf (Chronik).
READ_LOGS	<i>System-Protokolldateien lesen.</i> Lesender Zugriff auf die Log-Dateien des Systems. Hier werden allgemeine Informationen zu durchgeführten Aktionen gespeichert, i. d. R. jedoch keine vertraulichen Informationen (es sei denn, ein Programmierer hat etwas verbockt).
READ_OWNER_DATA	<i>Eigentümerdaten lesen.</i> Auslesen der auf dem Gerät gespeicherten Eigentümerdaten.
READ_PHONE_STATE	<i>Telefonstatus lesen und identifizieren.</i> Zugriff auf die Telefonfunktionen des Gerätes. Eine Anwendung mit dieser Berechtigung kann unter anderem die Telefon- und Seriennummer dieses Telefons ermitteln und feststellen, ob ein Anruf aktiv ist oder mit welcher Nummer der Anrufer verbunden ist. Ein Media-Player kann so z.B. bei eingehenden Anrufen automatisch auf "Pause" schalten. Bei Werbung (z.B. AdMob) wird dies häufig zum Auslesen der IMEI/IMSI genutzt um festzustellen, welche Werbung auf dem Gerät bereits angezeigt wurde (eindeutige Identifizierung, Tracking). Apps,

Permission	Erklärung
	die auch für Android 1.6 und früher kompatibel sein sollen, wird diese Permission automatisch gesetzt.
READ_SECURE_SETTINGS	Lesezugriff auf Systemeinstellungen (u. a. Umschalter für die mobile Datenverbindung). Eigentlich dem System vorbehalten.
READ_SMS	<i>SMS oder MMS lesen.</i> Damit lassen sich bereits gespeicherte Kurznachrichten lesen. Darunter können natürlich auch vertrauliche Informationen sein...
READ_SYNC_SETTINGS	<i>Synchronisierungseinstellungen lesen.</i> Lesezugriff auf die Einstellungen der Synchronisation – etwa um festzustellen, ob selbige für Kontakte aktiviert ist. Ein gutes Zeichen: Die App möchte vielleicht wissen, ob der Anwender eine Datensynchronisation im Hintergrund erlaubt, und sich (hoffentlich) entsprechend verhalten.
READ_SYNC_STATS	<i>Synchronisierungsstatistiken lesen.</i> Beispielsweise den Verlauf bereits durchgeführter Synchronisationen einsehen.
REBOOT	Erlaubt den Neustart des Gerätes. Wie der Name es bereits andeutet: Diese App kann mal eben einen Reboot veranlassen.
RECEIVE_BOOT_COMPLETED	<i>Automatisch nach dem Booten starten.</i> App möchte benachrichtigt werden, wenn der Bootvorgang abgeschlossen ist. I.d.R. heißt das: Sie möchte nach dem Booten automatisch gestartet werden.
RECEIVE_MMS, RECEIVE_SMS	<i>MMS empfangen, SMS empfangen.</i> Eingehenden MMS/SMS abfangen – da möchte wohl jemand mitlesen. Kann aber durchaus OK sein, wenn die App auf MMS/SMS reagieren soll. Auf der anderen Seite kann man damit eingehende Nachrichten auch "im Nirvana" verschwinden lassen...
RECORD_AUDIO	<i>Audio aufnehmen.</i> Tonaufnahmen erstellen. Das kann sowohl für ein "Diktaphon" genutzt werden – als auch zum Mitschneiden von Telefonaten.
RESTART_PACKAGES	<i>Anwendungen neu starten.</i> Laufende Apps neu starten. Wird z. B. verwendet, um von selbigen ein Backup erstellen zu können. (In der API-Referenz mittlerweile auf "deprecated" gesetzt, sollte also in neueren Versionen nicht mehr genutzt werden)
SEND_SMS	<i>Kurznachrichten senden.</i> Und zwar ohne Zutun des Benutzers, auch an richtig teure Premium-Dienste (womit klar ist, wozu "böse

Permission	Erklärung
	Apps" das gern hätten). Es gibt aber auch "gute" Gründe für diese Permission: Natürlich die SMS-Apps, aber teilweise auch In-App-Käufe, die nicht über Google Checkout abgewickelt werden.
SET_ACTIVITY_WATCHER	Die Ausführung von Systemaktivitäten beobachten. Wird meist für Debugging benutzt. Wenn es also keine Beta ist, hat der Entwickler vielleicht nur vergessen, das wieder raus zu nehmen.
SET_ALWAYS_FINISH	App kann sich selber <i>beenden</i> – also wirklich beenden, nicht nur in den Hintergrund gehen.
SET_ANIMATION_SCALE	<i>Allgemeine Animationsgeschwindigkeit einstellen.</i> Anpassung der Animationsgeschwindigkeit (schnellere/langsamere Animationen).
SET_PREFERRED_APPLICATIONS	App kann jeder Aktivität eine Default-App zuweisen (etwa den Browser zum Öffnen einer URL). In neueren Android-Versionen ohne Auswirkung.
STATUS_BAR	Kann die Status-Bar (Notification?) öffnen, schließen, und ausblenden. Meist will die App wohl letzteres, um einen "Vollbild-Modus" zu ermöglichen.
SUBSCRIBED_FEEDS_READ	<i>Abonnierte Feeds lesen.</i> Abrufen von Details zu den derzeitig synchronisierten Feeds.
SUBSCRIBED_FEEDS_WRITE	<i>Abonnierte Feeds schreiben.</i> Änderungen an kürzlich synchronisierten Feeds vornehmen.
SYSTEM_ALERT_WINDOW	<i>Warnungen auf Systemebene anzeigen.</i> Fenster mit Systemwarnungen einblenden. Eine böswillige App kann so den gesamten Bildschirm blockieren. Sollte eigentlich nicht benutzt werden, da dies für System-Meldungen gedacht ist. Erlaubt das anzeigen von "Alert Windows", d. h. Nachrichtenfenstern, die immer im Vordergrund angezeigt werden.
USE_CREDENTIALS	<i>Authentifizierungsinformationen eines Kontos verwenden.</i> Möchte die konfigurierten Zugangsdaten verwenden. Das heißt nicht unbedingt, dass es sie "zu sehen bekommt" – aber die App kann sich quasi "im Namen des Anwenders" anmelden.
USE_SIP	App kann Internet-Telefonie nutzen (SIP ist das Protokoll dafür)
VIBRATE	<i>Vibrationsalarm steuern.</i> Wird gern genutzt, um beispielsweise auf die Beendigung (oder auch den Start) einer Aktivität hinzuweisen –

Permission	Erklärung
	oder generell, um die Aufmerksamkeit des Anwenders zu wecken. Laute Töne sind ja nicht immer erwünscht. „Vibrieren“ heißt: Lass das Gerät zittern und summen...
WAKE_LOCK	<i>Standby-Modus deaktivieren.</i> App kann das System daran hindern, einen Ruhezustand einzunehmen (also den Bildschirm zu dimmen, die CPU "schlafen" zu lassen, etc.) Wäre doch blöd, wenn die Navi-App läuft und plötzlich der Bildschirm ausgeht.
WRITE_APN_SETTINGS	<i>Einstellungen für Zugriffspunktname schreiben.</i> App kann die Zugangsdaten zum Internet etc. (siehe APN) verändern. Meist geht es der App nur darum, den Namen des APN zu ändern – um die Verwendung des mobilen Datenverkehrs zu steuern (Beispiel: APNDroid).
WRITE_CALENDAR	<i>Kalenderdaten schreiben.</i> Diese Permission erlaubt lediglich den Schreib-, nicht aber den Lesezugriff auf den Kalender. Die damit versehene App kann also Termine hinzufügen, nicht aber lesen oder ändern.
WRITE_CONTACTS	<i>Kontaktdaten schreiben.</i> Wie
WRITE_HISTORY_BOOKMARKS	WRITE_CALENDAR, nur in Bezug auf die Kontaktdaten bzw. Browser-History (Chronik) und Lesezeichen.
WRITE_EXTERNAL_STORAGE	App darf beliebige Daten auf der (externen) SD-Karte lesen, schreiben, verändern und auch löschen – prinzipiell auch die Daten anderer Apps. Diese Permission ist aber beispielsweise essentiell für diverse Backup- und Kamera-Apps, die natürlich Daten auf der Karte manipulieren müssen.
WRITE_OWNER_DATA	<i>Eigentümerdaten schreiben.</i> Schreiben/verändern der auf dem Gerät gespeicherten Eigentümerdaten.
WRITE_SECURE_SETTINGS	<i>Allgemeine Systemeinstellungen ändern.</i>
WRITE_SETTINGS	Lesen und Schreiben von Systemeinstellungen. "Secure Settings" können nur von Systemanwendungen (also solchen, die ins "ROM" integriert wurden) angefordert werden.
WRITE_SYNC_SETTINGS	<i>Synchronisierungseinstellungen schreiben.</i> Schreibzugriff auf die Einstellungen der Synchronisation. Eine mit dieser Permission ausgestattete App kann u. a. die Synchronisation von Kontakten und Kalendern aktivieren bzw. deaktivieren.

Permission	Erklärung
com.android.vending.BILLING	In-App Payment (in der App integrierte Bezahldienste, die über den Market abgewickelt werden)

Weniger gebräuchliche sowie offensichtlich klingende Permissions (was heißt wohl [BRICK](#)? Ja, genau: Phone in Sachen Brauchbarkeit mit einem Ziegelstein vergleichbar zu machen, also unbrauchbar. Nicht lachen – diese Permission gibt es wirklich! So, jetzt lachen...) habe ich hier ausgelassen; die müssen also bei Bedarf unter eingangs genanntem Link selber nachgeschlagen werden. Oder man wirft einen Blick auf die App [AllPermissions](#) (bei AndroidPIT – aus dem Google-Market wurde sie entfernt). Wie der Name bereits suggeriert, handelt es sich hier um eine dummy-App, die alle (unter Android 2.1 verfügbaren) Permissions verlangt. Da AndroidPIT hier bei "Mouse Over" kurze Erklärungen anzeigt, lernt man dabei auch einiges.

APN-Einstellungen ausgewählter Netzbetreiber

APN steht für **A**ccess **P**oint **N**ame (Name des Zugangspunktes). Hier sind also die Zugangspunkte gemeint, die den Androiden ins Internet bringen – oder erlauben, MMS zu verschicken. Diese Zugangspunkte sind, verständlicherweise, Anbieter-spezifisch. Eine kleine Auswahl (entnommen aus dem [AndroidPIT Wiki](#) und von [Android-Hilfe.DE](#), unserem sogenannten "Parallel-Universum"; einen Blick Wert ist auch die [APNList von CyanogenMod](#), da gut gepflegt) finden sich hier:

Netzanbieter	Einstellungen
Deutschland	
Aldi Talk	Name: Tagesflat APN: tagesflat.eplus.de Username: eplus Passwort: gprs <i>Die 30-Tages-flat muss erst bei 1155 (Konto-Hotline) gebucht werden:</i> Name: Monatsflat APN: internet.eplus.de Username: eplus Passwort: gprs
Alice (O2-Netz GPRS, UMTS und MMS)	Name: Alice GPRS APN: internet.partner1 Proxy: nicht festgelegt Port: nicht festgelegt Nutzernamen: nicht festgelegt Passwort: nicht festgelegt Server: nicht festgelegt MMSC: http://10.81.0.7:8002 MMS-Proxy: 82.113.100.41 MMS-Port: 8080 MCC: 262 MNC: 07 APN-Typ: nicht festgelegt
Blau DE	Name: blau DE MMC: 262 MNC: 05 APN: internet.eplus.de Username: blau Passwort: blau APN-Typ: default,supl <i>MMS:</i> Name: blau DE MMS MCC: 262 MNC: 05 APN: mms.eplus.de Username: mms Passwort: eplus

Netzanbieter	Einstellungen
	MMSC: http://mms/eplus MMS proxy: 212.23.97.153 mmsport: 5080 APN-Typ: mms
e-plus	Name: Eplus Internet APN: internet.eplus.de Username: eplus Passwort: eplus MCC: 262 MNC: 03 APN type: default <i>Für MMS:</i> Name: Eplus MMS APN: mms.eplus.de Username: mms Passwort: eplus MMSC: http://mms/eplus MMS proxy: 212.023.097.153 MMS port: 5080 MCC: 262 MNC: 03 APN type: mms <i>WAP:</i> Name: E-Plus WAP GPRS MMC: 262 MNC: 03 APN: wap.eplus.de Proxy: 212.23.97.9 Port: 8080 APN-Type: default,supl,mms
netzclub	Name: netzclub APN: pinternet.interkom.de MCC: 262 MNC: 07 APN Typ: default <i>Für MMS:</i> Name: netzclub MMS APN: pinternet.interkom.de MMSC: http://10.81.0.7:8002 MMS Proxy: 82.113.100.5 MMS Port: 8080 MCC: 262 MNC: 07 APN Typ: mms
02	Name: o2-de (frei definierbar) APN: pinternet.interkom.de

Netzanbieter	Einstellungen
	<p>Proxy: nicht festgelegt Port: nicht festgelegt Nutzername: nicht festgelegt Passwort: nicht festgelegt Server: nicht festgelegt MMSC: http://10.81.0.7:8002 MMS-Proxy: 82.113.100.6 MMS-Port: 8080 MCC: 262 MNC: 07 APN-Typ: internet + mms Authentifizierungstyp: None (ggf. nicht festgelegt testen!)</p> <p><i>Alternativ (O2 Prepaid):</i> Name o2-de (frei definierbar) APN internet Proxy Nicht festgelegt Port Nicht festgelegt Nutzername Nicht festgelegt Passwort: Nicht festgelegt Server Nicht festgelegt MMSC http://10.81.0.7:8002 MMS-Proxy 82.113.100.5 MMS-Port 8080 MMS-Protokoll WAP 2.0 MCC 262 MNC 07 Authentifizierungstyp CHAP APN-Typ Nicht festgelegt</p>
Simyo	<p>Name: simyo Internet APN: internet.eplus.de username: simyo passwort: simyo MCC: 262 MNC: 03 APN Typ: default</p> <p><i>Für MMS:</i> Name: simyo MMS APN: mms.eplus.de username: simyo passwort: simyo MMSC: http://mms/eplus MMS Proxy: 212.023.097.153 MMS Port: 5080 MCC: 262 MNC: 03 APN Typ: mms</p>

Netzanbieter	Einstellungen
Solomo	Name: solomo.de Internet MMC: 262 MNC: 03 APN: internet.vistream.net Username: web Password: web APN-Typ: default,supl <i>MMS:</i> Name: solomo.de MMS MMC: 262 MNC: 03 APN: mms.vistream.net Username: mms Password: mms MMSC: http://172.30.66.40:20080 MMS proxy: 172.31.43.21 mmsport: 8080 APN-Typ: mms
Tchibo (O2-Netz)	Name: Tchibo APN: webmobil1 proxy: port: username: password: server: MMSC: http://10.81.0.7:8002 MMS Proxy: 82.113.100.8 MMS Port: 8080 MCC: 262 MNC: 07 APN Typ:
T-Mobile	APN: internet.t-mobile Benutzername: t-mobile Passwort: tm Authentifizierungstyp: PAP <i>MMS:</i> Name: T-Mobile MMS APN: internet.t-mobile Nutzername: t-mobile Passwort: tm MMSC: http://mms.t-mobile.de/servlets/mms MMS-Proxy: 172.28.23.131 MMS-Port: 8008 Authentifizierungstyp: PAP APN-Typ: mms
Vodafone	APN: web.vodafone.de primärer DNS: 139.7.30.125

Netzanbieter	Einstellungen
	sekundärer DNS: 139.7.30.126 Benutzername: - Passwort: - <i>MMS:</i> Name: Vodafone DE-MMS MMC: 262 MNC: 04 APN: event.vodafone.de MMSC: http://139.7.24.1/servlets/mms MMS proxy: 139.7.29.17 mmsport: 80 APN-Typ: mms
Österreich	
A1	Name: A1.net APN: A1.net Benutzer: ppp@A1plus.at Kennwort: ppp MCC: 232 MNC: 01 APN-Typ: Internet <i>Für MMS:</i> Name: free.A1.net APN: free.A1.net MMSC: http://mmsc.a1.net Proxy: 194.48.124.71 Port: 8001 APN-Typ: MMS
bob	APN: bob.at Benutzername: data@bob.at Passwort: ppp
Drei	APN: drei.at Benutzername: leer Passwort: leer MNC: 10
One Orange	APN: web.one.at Benutzername: web Passwort: web MNC: 05
Telekom-at	name: gprsinternet apn: *99# <i>Alle anderen Felder bleiben leer</i>
tele.ring	Name: tele.ring web APN: web Benutzer: web@telering.at Kennwort: web MCC: 232 MNC: 07

Netzanbieter	Einstellungen
Yesss!	APN: web.yesss.at Benutzername: leer Passwort: leer MNC: 12
Schweiz	
Orange-CH	MMC: 228 MNC: 3 APN: internet <i>MMS:</i> MMC: 228 MNC: 03 APN: mms MMSC: http://192.168.151.3:8002 MMSProxy: 192.168.151.002 MMSPort: 8080 APN-Typ: mms
Swisscom	Name: Swisscom GPRS APN: gprs.swisscom.ch username: gprs passwort: gprs MCC: 228 MNC: 01 APN Typ: default <i>Für MMS:</i> Name: Swisscom MMS APN: event.swisscom.ch MMSC: http://mms.natel.ch:8079 MMS Proxy: 192.168.210.2 MMS Port: 8080 MCC: 228 MNC: 01 APN Typ: mms

Secret Codes oder Magische Nummern

Klar kann man mit einem Telefon telefonieren. Dazu gibt man eine Ziffernfolge ein, und drückt die Taste für "Abheben". Was aber passiert, wenn man noch ein paar Sonderzeichen hinzufügt?

Disclaimer: Es kann sein, dass diese Codes nicht auf allen Geräten funktionieren. Sofern sie nicht im Handbuch des Gerätes aufgeführt sind, mag das durchaus seine Gründe haben: So kann der eine oder andere Punkt, unsachgemäß angewendet, u.U. Schäden verursachen. Ich übernehme keinerlei Garantien dafür, dass folgende Codes a) funktionieren, b) das tun, was da steht, oder c) "folgenfrei" genutzt werden können. Insbesondere übernehme ich keinerlei Verantwortung für etwaige negative Folgen! Für etwaige positive Folgen stelle ich natürlich gern mein Bankkonto zur Verfügung...

Es kann sein, dass nach Eingabe des einen oder anderen Codes nichts passiert. Es kann aber auch sein, dass dies nur so scheint (bei meinen Tests fand ich anschließend – am nächsten Tag – einige der "magischen Nummern" in der "Verbraucherliste" wieder). Es können hier durchaus Hintergrund-Dienste gestartet oder aber "versteckte Klassen/Menüs" in Apps freigeschaltet werden...

Code	Bedeutung
##0##*	Test des LCD-Displays
*#0011#	GSM-Infos
*#0228#	ausführliche Infos zum Barreriestatus
##0283##	Loopback-Test
**05*<PUK Code>*<neue PIN>*<neue PIN zur Bestätigung>#	Entsperren des Telefons aus dem Notruf-Modus
##0588##	Test des Annäherungs-Sensors
*#06#	IMEI
##0673##	Einer der beiden Codes führt zu einem Audio-Test
##0289##	
*#0782#	RTC (Real-Time-Clock) auslesen und anzeigen
##0842##	Test für Vibrator (oh ja!) und Hintergrund-Beleuchtung
##1111##	FTA Software Version
##1234##	Firmware-Info
*135#	Eigene Rufnummer anzeigen
##1472365##	ein kurzer GPS-Test (und Einstellungen)
##1575##	ein weiterer GPS-Test
##197328640##	Service-Menü mit verschiedenen Test-Möglichkeiten
##2222##	FTA Hardware Version
*#2263#	Bandnutzung
##232331##	BlueTooth-Test
##232337##	MAC-Adresse des BlueTooth-Interfaces
##232338##	MAC-Adresse des WLAN-Interfaces anzeigen
##232339##	
##526##	Einer dieser Codes führt zu den WLAN-Tests...
##528##	
##2663##	Touch-Screen Version anzeigen
##2664##	Touch-Screen Test
*##273283*255*663282*##*	angeblich für eine schnelle Sicherung der Medien-Dateien (Fotos, Videos..) gut. Von wo und nach wo? Habe ich nicht probiert...
*2767*3855#	Factory Format (Wipe). Vorsicht damit!!!
#31#<Rufnummer>	Mit unterdrückter Rufnummer anrufen
##3264##	RAM Test / RAM-Version anzeigen
##3424##	HTC Function Test
##34971539##	Kamera-Menü mit folgenden Punkten: Update mit Firmware aus Bild (keinesfalls! Sonst futsch!); Update mit Firmware von SD-Karte; Versions-Info; Update-Counter

Code	Bedeutung
*##44336##	Herstellungszeitpunkt und Laufende Nummer
*##4636##	Öffnet ein Menü, aus dem sich wählen lässt: Telefon-Infos, Akku-Infos, Akku-History, Verbrauchsstatistiken.
*##4986*2650468##	Diverse Hardware-Infos
*##564##	QXDM (Qualcomm eXtensible Diagnostic Monitor) Logging FrontEnd
*##7262626##	Ein Bett im Kornfeld... äh, Feld-Test
*##7269##	Standard Device-Logging (Device [logcat?], AT-Befehle, Kernel [dmesg?] -- HTC?)
*#7465625#	Netz- und Subnetzsperrung, Simlock, Service Provider und Corporate Lock (Galaxy-S?)
*##7594##	Verhalten des Einschalt-Knopfes ändern (z. B. direktes Abschalten ohne Menü)
*##7780##	Zurücksetzen auf Werkseinstellungen
*##8255##	GoogleTalk Service Überwachung
*##8350##	Logging der Anrufe deaktivieren
*##8351##	Logging der Anrufe aktivieren
*#9090#	Service-Modus UART/USB
*##9696##	FTP Test

Weitere spezielle magische Nummern gibt es für Geräte von...

- ...[LG](#)
- ...[Motorola](#)
- ...[Samsung](#)
- ...[Samsung](#) (SGS)
- ...[Samsung](#) (SamFirmware)

Leistungsaufnahme verschiedener Komponenten

Heise hat für seinen Artikel [Energiesparplan](#) ein Motorola Milestone angepasst und ausführlich getestet, was welche Komponente so verbraucht. Und wie man das einschränken kann (das war jetzt eine klare Empfehlung, den Artikel zu lesen!). Die Daten davon werden einerseits ein "war ja klar", aber bei einigen Daten auch ein "oh, das hätte ich jetzt nicht gedacht" hervorrufen:

Betriebszustand	zusätzliche Leistungsaufnahme
Videoaufnahme ¹	1557 mW
UMTS Upload	1410 mW
UMTS Download	1349 mW
EGDE Upload	1179 mW
WLAN Download	1158 mW
Video abspielen (fullscreen) ¹	1135 mW
UMTS-Telefonat	983 mW
Kamera ¹	934 mW
EGDE Download	853 mW
Bluetooth empfangen	751 mW
Display (höchste Stufe)	730 mW
GPS Suche	550 mW
GSM-Telefonat	511 mW
Bluetooth senden	487 mW
WLAN Upload	479 mW
Display (niedrigste Stufe)	310 mW
MP3 abspielen	160 mW
UMTS Standby	18,3 mW
GSM/EDGE Standby	11,6 mW
WLAN Standby	7,8 mW
Bluetooth Standby	2,8 mW
GPS Standby	0,4 mW

¹ Leistungsaufnahme des Displays bereits abgezogen

Für die Gesamt-Leistungsaufnahme muss natürlich noch die Grundlast hinzugerechnet werden (was das Gerät verbraucht, wenn alles in der Tabelle genannte abgeschaltet ist; also "Flugmodus"). Das wären ganze fette 6,4mW. Wird das Gerät also des Nachts in diesen versetzt, spart das bereits enorm:

Betriebszustand	zusätzliche Leistungsaufnahme
Flugmodus	6,4 mW
GSM Bereitschaft	18 mW
GSM Bereitschaft + WLAN Standby	25,8 mW
GSM Bereitschaft + WLAN Standby + Bluetooth Standby	28,6 mW
UMTS Bereitschaft	24,7 mW
UMTS Bereitschaft + WLAN Standby	32,5 mW
UMTS Bereitschaft + WLAN Standby + Bluetooth Standby	35,3 mW

Die Werte sind natürlich alle spezifisch für o.g. Motorola Milestone, und können auf anderen Geräten abweichen. Die Größenordnungen sollten aber zumindest ähnlich sein.

Nur so nochmal gesagt: Selbst wenn WLAN etc. alles aus sind, und nur Telefonate (und SMS) noch durchkommen (den Datenverkehr also mal ganz außen vorgelassen), reduziert sich der Verbrauch im Flugmodus auf ein Drittel (GSM) oder gar ein Viertel (UMTS). Einmal 6 Stunden angenommen (von 0 Uhr bis 6 Uhr), hält der Akku damit (theoretisch) für bis zu gut 2 Stunden länger. Natürlich wieder nur, wenn man dann anschließend auch nichts damit macht – also wieder so ein "Laborwert". Trotzdem kann man sich leicht ausrechnen: Sechs Stunden Flugmodus (statt GSM Bereitschaft) schaffen Raum für zusätzliche ca. 8 Minuten GSM Telefonat...