



Google Android

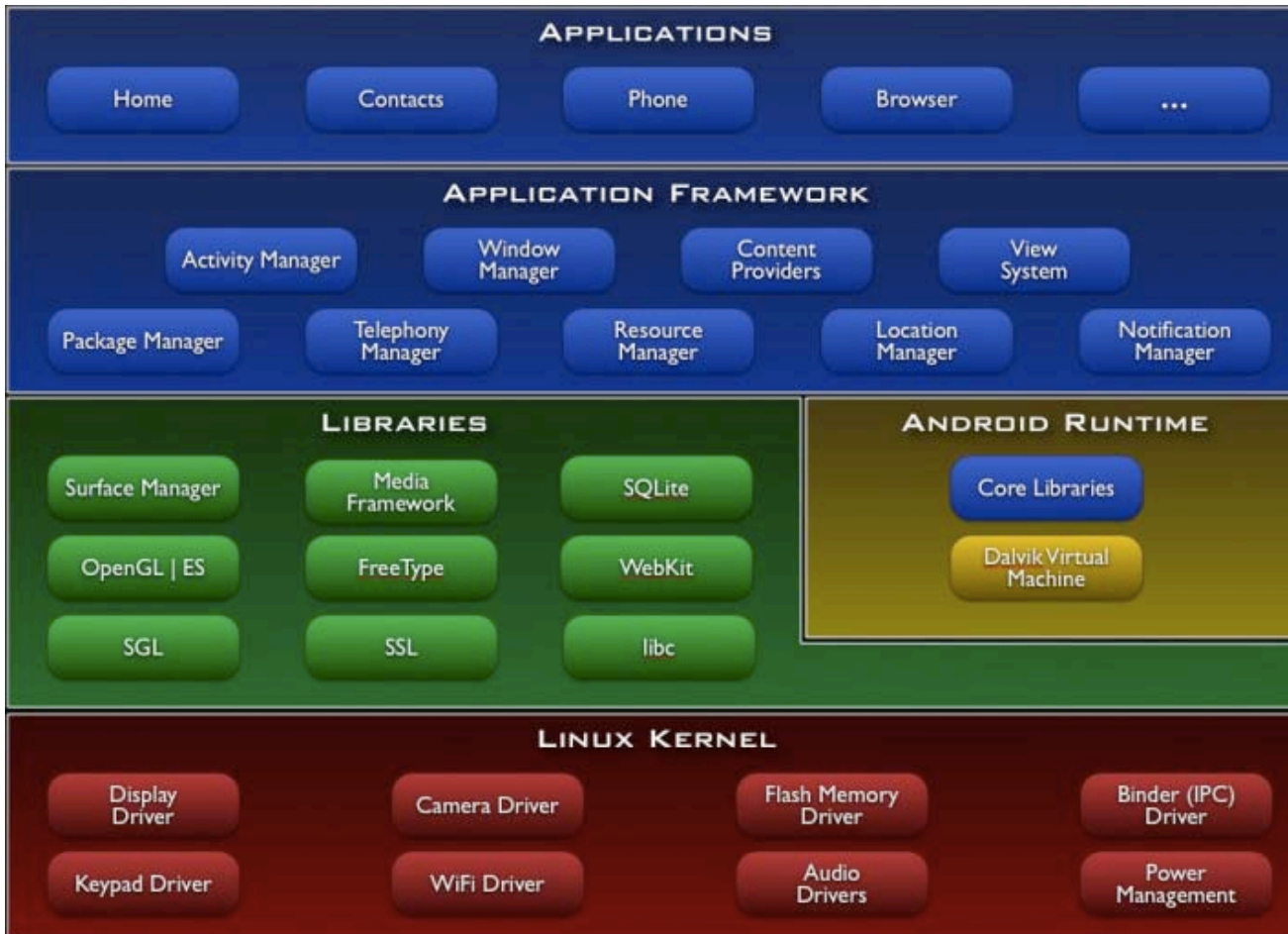
Überblick

- Geschichte
- Architektur
- SDK
- Anwendungen
- Komponenten einer Anwendung
- Beispiel
- Referenzen

Geschichte

- 5. November 2007: Google kündigt Android an
- Open Handset Alliance: Vereinigung von z. Z. 47 Netzbetreibern, Chip-, Telefon- und Software-Herstellern
- 22. Oktober 2008: T-Mobile G1 auf US-Markt verfügbar
- 7. Juli 2009: Ankündigung von Google OS

Architektur



Applications

■ Kern-Applikationen

- E-Mail Client
- SMS
- Kalender
- Maps
- Browser
- Kontakte
- ...

■ alle in Java geschrieben

Application Framework

- vollständiger Zugriff auf API, die auch den Core Applications zur Verfügung steht
- Wiederverwendung von Komponenten: Applikationen stellen Funktionalität zur Nutzung durch andere Applikationen zur Verfügung
- Views
- Content Providers
- Resource Manager
- Notification Manager
- Activity Manager

Libraries

- libc (BSD)
- Media Libraries (MPEG4, H.264, ...)
- LibWebCore
- 2D/3D
- FreeType
- SQLite

Android Runtime

- Subset von Apache Harmony; Grossteil der Core Java Libraries unterstützt
- Dalvik VM: register-basierte, auf minimalen Speicherbedarf optimierte VM; kein JIT Compiler



Linux Kernel

- Linux 2.6.x Kernel

SDK

- <http://developer.android.com>
- aktuell 1.6 R1
- Development Tools
 - Eclipse Plugin
 - Emulator, Debugger, ...
- Android Plattformen
- Beispielcode
- Dokumentation

Anwendungen I

- in Java geschrieben, Transformation des Java-Bytecode in Android-Bytecode im .dex Format
- gebündelt in einem Android Package .apk
- i. d. R. läuft jede Anwendung in einem eigenen Prozess, mit einer eigenen Instanz der Dalvik VM, unter einem eigenen Linux-User

Anwendungen II

■ Manifest Datei

- Java Package Name (Id der Anwendung)
- Komponenten-Deklaration
- Berechtigungen
- verwendete Libraries



Komponenten einer Anwendung

- Activity
- Service
- Broadcast Receiver
- Content Provider

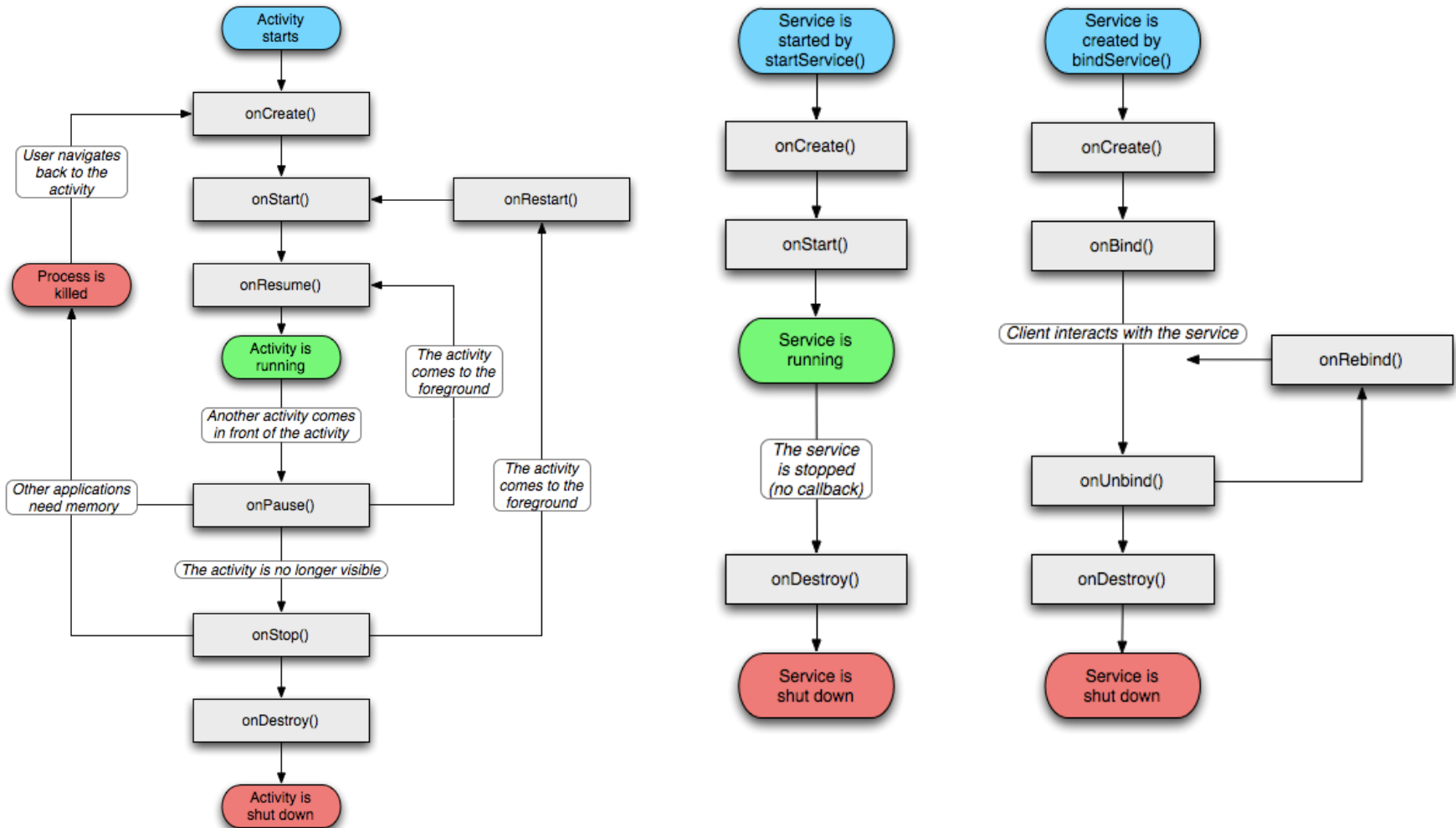
Activity

- User Interface für einen einzelnen Task
- z. B. IM Applikation:
 - Liste von Kontakten
 - Textmitteilung verfassen
 - Nachrichten-Archiv
 - Einstellungen
- grafischer Inhalt eines Fensters durch Hierarchie von Views festgelegt

Service

- entspricht einem Service (Daemon) eines Desktop- oder Server-Systems
- läuft im Hintergrund ab Instanziierung bis zum Ausschalten
- Kein User Interface
- z. B. MP3 Player

Lifecycles



Broadcast Receiver

- hört und reagiert auf systemweite Ereignisse, ausgelöst durch Android selber oder durch ein Programm
- z. B. „battery low“
- stellen selber keinen grafischen Inhalt dar, können aber Activities starten oder mittels NotificationManager Alerts senden

Content Provider

- speichert und lädt Daten und stellt sie Applikationen zur Verfügung
- vorgegebene Content-Provider für Basis-Datentypen
- z. B. Abfrage von Kontaktdaten mittels Query an die URI
„content://contacts/people“

Intent

- asynchrone Nachricht zur Aktivierung einer Komponente
- explizite Intents: werden an Instanz der benannten Zielkomponente gesendet
- implizite Intents: Android ermittelt die am besten geeignete Komponente

UI

- Definition einer Hierarchie von View und ViewGroup und dem Layout dieser Hierarchie in einer XML Layout Datei
- Android generiert Klasse mit Resource-Identifiern

Referenzen

- <http://developer.android.com>
- <http://www.openhandsetalliance.com>
- Android Application Development; Rogers, Lombardo, Mednieks, Meike; O'Reilly 2009