

Velkommen

Android 2.x CodeJam



Om oplægsholderen

- Jacob Nordfalk
 - nordfalk@lundogbendsen.dk
- Instruktør for Lund&Bendsen
- Ekstern lektor på IHK
- Underviser i Java SE, Java EE, Webserverprogrammering, OOAD, Android
- Forfatter af <http://javabog.dk>
- Udvikler til J2SE, J2EE, Android
- Har et par apps på Android Market
 - Teoriprøven, AndroidElementer, ...



Om Lund&Bendsen

- Formidling af Java er vores kernekompetence.
- Undervisning, mentoring, konsulent-ydelser mm.
- Specielt for større danske virksomheder
 - CSC, IBM, Steria, Logica, BEC, Novo, PFA, TDC, Post Danmark, SDC, DSB...
- Hjemmeside: www.lundogbendsen.dk
- Kontakt: info@lundogbendsen.dk
- Tlf: 33 861 861



Dagens program

- Oplæg: I gang med Android
- Første udfordring
- Oplæg: Elementer i Android
- Jamsession
- Afrunding

AndroidElementer

- Små, afgrænsede og (relativt) let forståelige eksempler på hvordan man programmerer til Google Android
 - <http://bit.ly/kWjDsg>



I gang med Android

- Værktøjer: JDK, Android SDK, Eclipse, Android Eclipse-plugin
- Android i forhold til java
- Aktiviteter
- Views og viewgroups
- Deklarative designs (design i Eclipse)
- Android manifestet

Hvad er Android?

- En softwareplatform og et styresystem til mobile enheder
- Udviklet af Google og senere Open Handset Alliance (OHA)
- Baseret på Linux-kernen
- Applikationer skrives i Java - platformens primære sprog
- Annonceret den 5. november 2007 med grundlæggelsen af Open Handset Alliance

Open Handset Alliance

- Et konsortium af selskaber, med formål at:
 - fremme åbne standarder for mobile enheder
 - sænke omkostningerne ved at udvikle og distribuere mobile enheder og tjenester



Open Source Licens

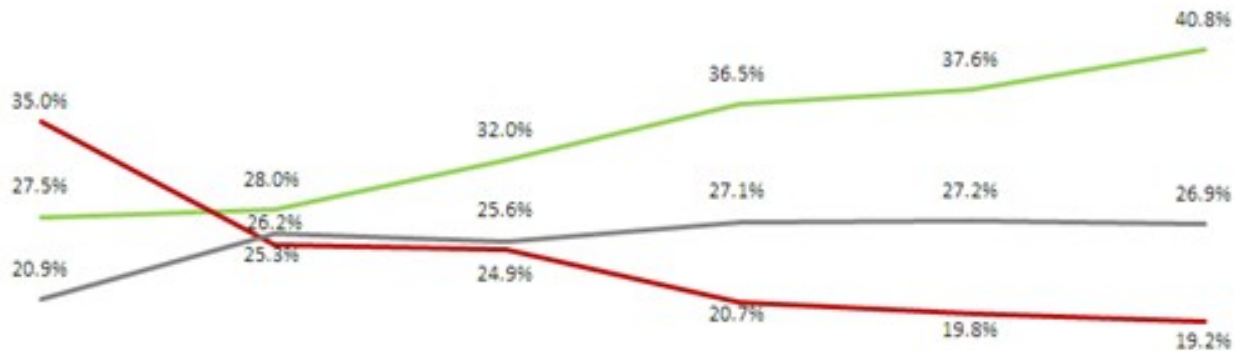
- Android er under Apache Software License
 - Open source licens - der er adgang til kildekoden
 - ASL kræver ikke at evt ændringer offentliggøres
- Enhver telefonfabrikant eller (kapabel) privatperson kan installere det
- Enhver udvikler kan bruge det
- Kildekode på <http://source.android.com>
- SDK på <http://developer.android.com>

Markedsandel

U.S. Smartphone Operating System Share - 6 Mo. Recent Acquirers

Adult Smartphone Consumers, Jun - Nov 2010

— Android — RIM BlackBerry OS — Apple iOS



June 2010 (n=1425)

July 2010 (n=1414)

Aug 2010 (n=1744)

Sep 2010 (n=1802)

Oct 2010 (n=1846)

Nov 2010 (n=1632)

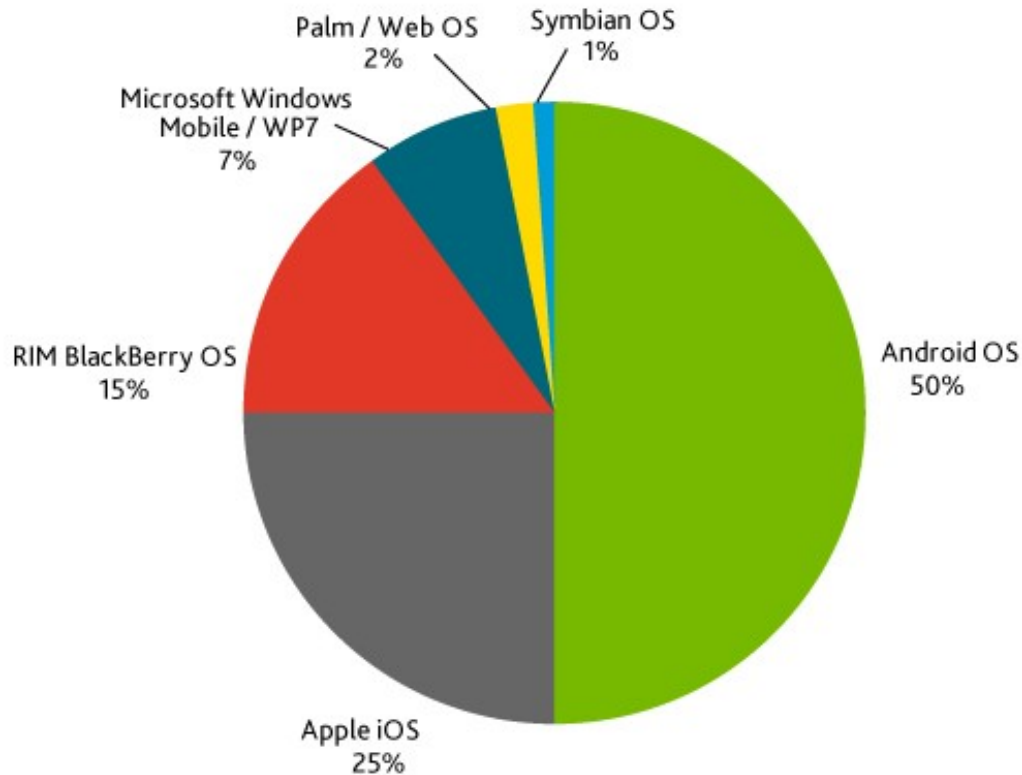
The Nielsen Company

<http://blog.nielsen.com/>

Markedsandel

Smartphone market share - recent acquirers

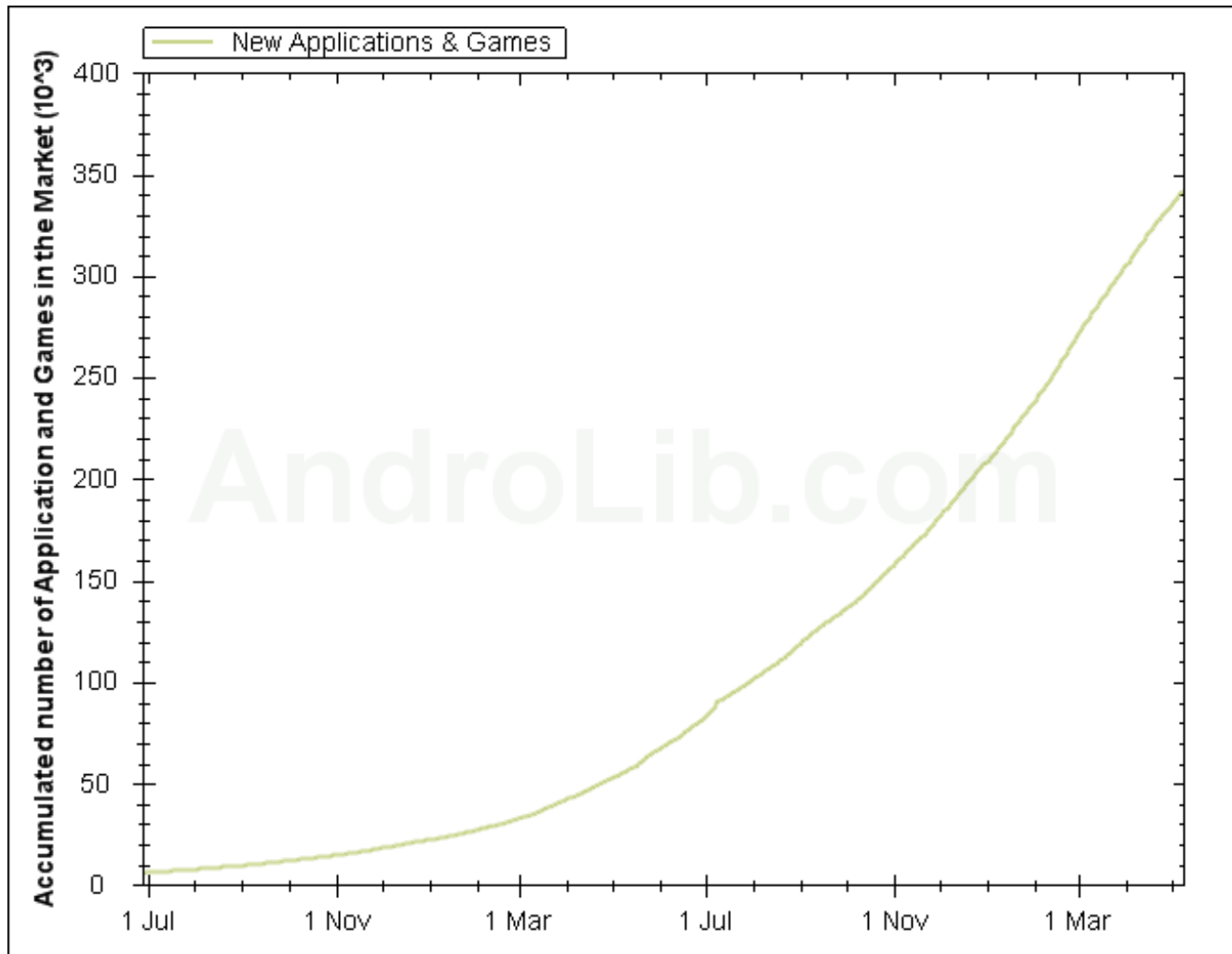
March '11, Nielsen Mobile Insights, National



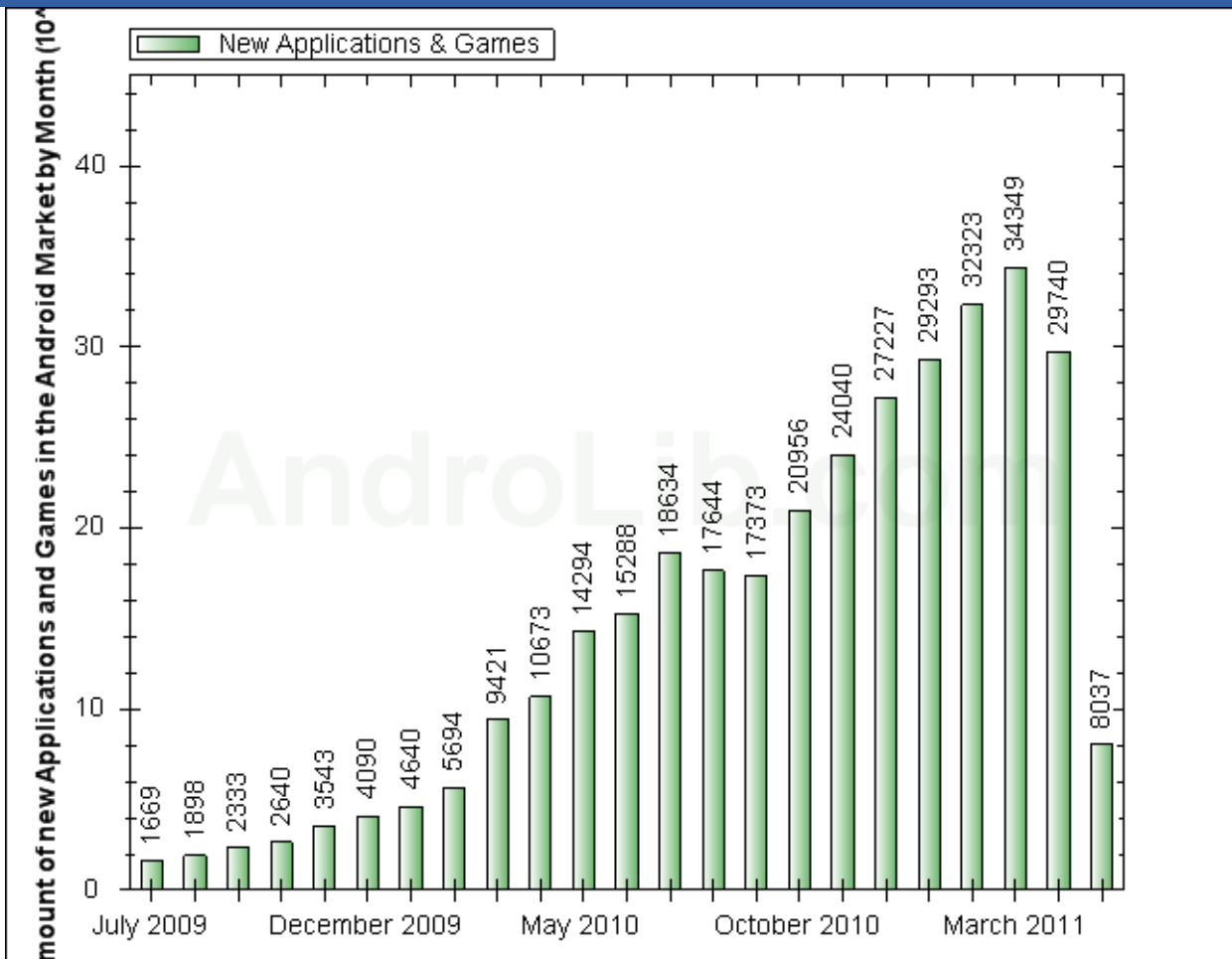
Source: The Nielsen Company.

nielsen

Over 1/4 mio apps



Antal nye apps pr måned



Kilde: <http://www.androlib.com/>

Se <http://www.androlib.com/appstats.aspx> for løbende statistik

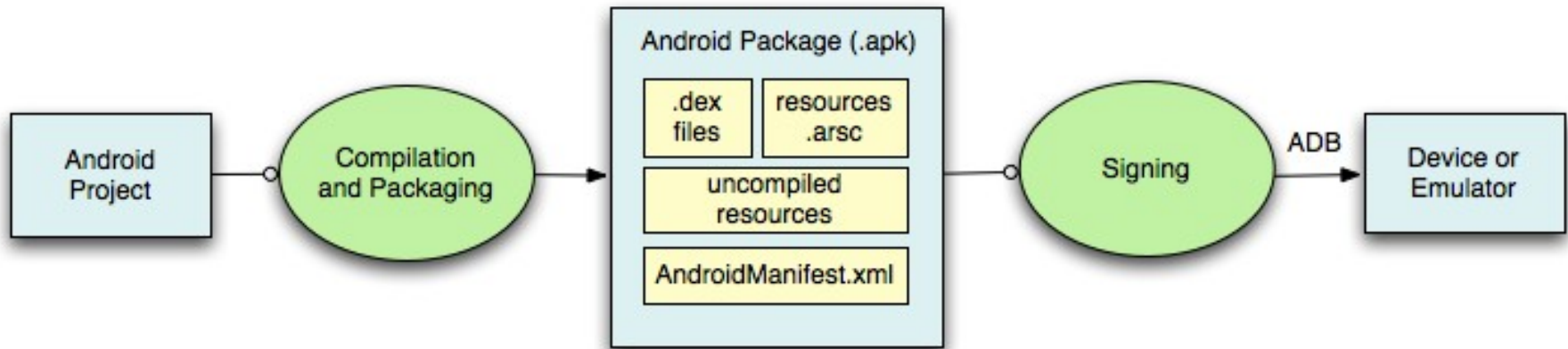
Udviklingsværktøjer

- Android SDK
 - Klassebibliotek (Android og... Javabibliotekerne)
 - Kommandolinjeværktøjer
 - adb – Android Debug Bridge
 - Emulatorer (AVD - Android Virtual Devices)
 - Dokumentation, eksempler, ...
- Udviklingsmiljøer
 - Eclipse plugin (open source, Google)
 - Motorola MOTODEV Studio for Android
 - Netbeans plugin (nbandroid - open source)
 - IntelliJ plugin (lukket)

Android i forhold til java

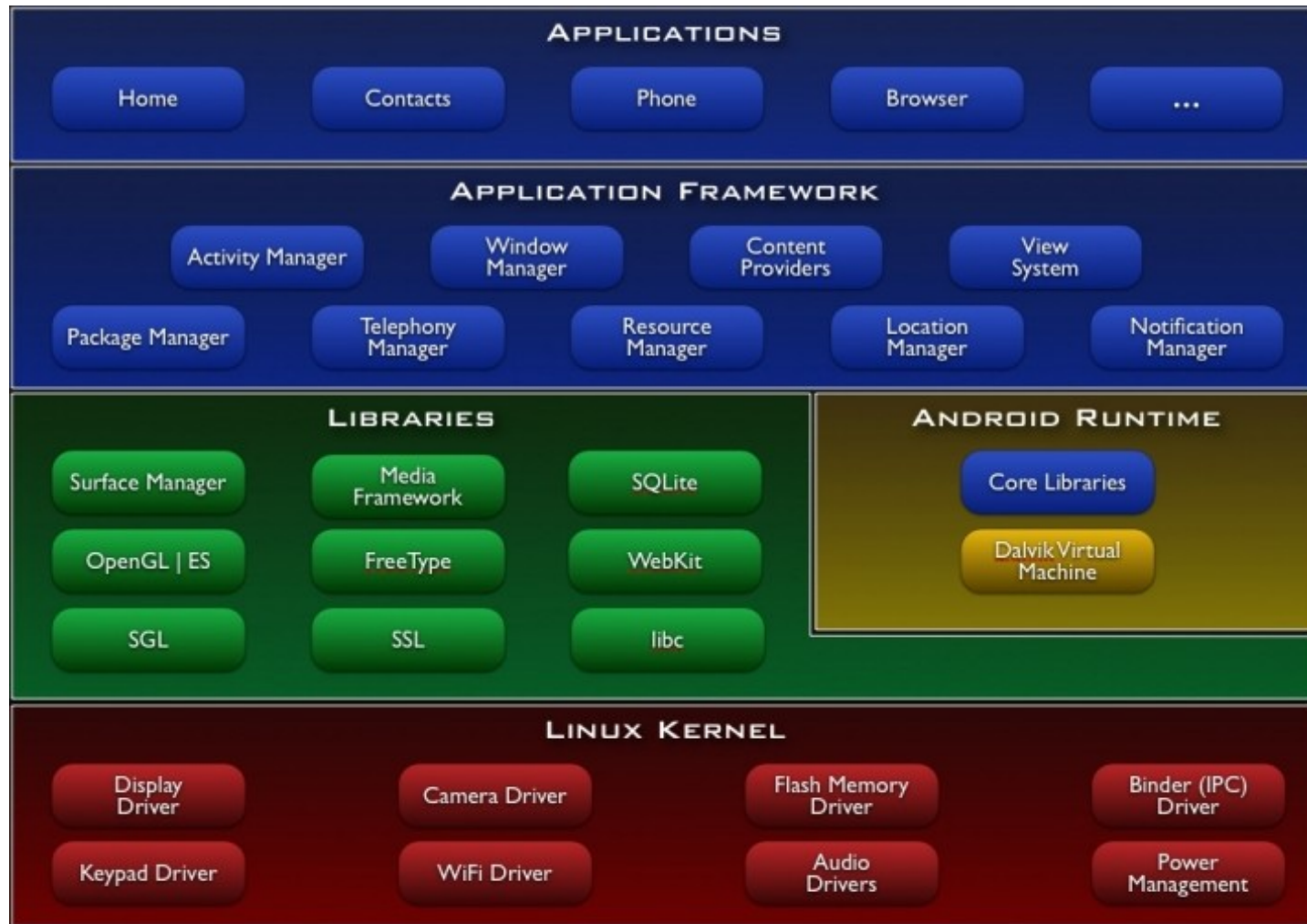
- Android har eget Java-bibliotek
 - brugergrænseflader: AWT/Swing er væk, i stedet er android.view.*
 - alternativ implementation (Apache Harmony) af standard Java
- .class-filer pakkes om til Androids format
 - Androids virtuelle maskine (Dalvik) kører .dex-filer, ikke Java .class-filer
- Den konverterede bytekodet bliver lagt i en arkivfil (.apk) sammen med programmets andre resurser (f.eks. billeder).
 - Det er .apk-filen som brugere kan installere på deres mobile enheder, enten ved at hente filen direkte eller via f.eks. Android marked.
- Resurser tilgås via static int-variable i klassen R
- Aktiviteter (skærm billeder) skal erklæres i AndroidManifest.xml
- Platform styrer skift mellem aktiviteter (via Intents)

Øversættelse og kørsel



- De binære .class-filer pakkes om til Androids eget format (.dex)
 - _ Fylder ca halvdelen - ukomprimeret
 - _ Androids virtuelle maskine (Dalvik) kan kun køre .dex-filer, ikke .class-filer
- Den konverterede bytekodde bliver lagt i en arkivfil med endelsen .apk, sammen med programmets andre resurser (billeder, lyd, data).
 - _ Det er .apk-filen som brugere kan installere på deres mobile enheder, enten ved at hente filen direkte eller via f.eks. Android marked.

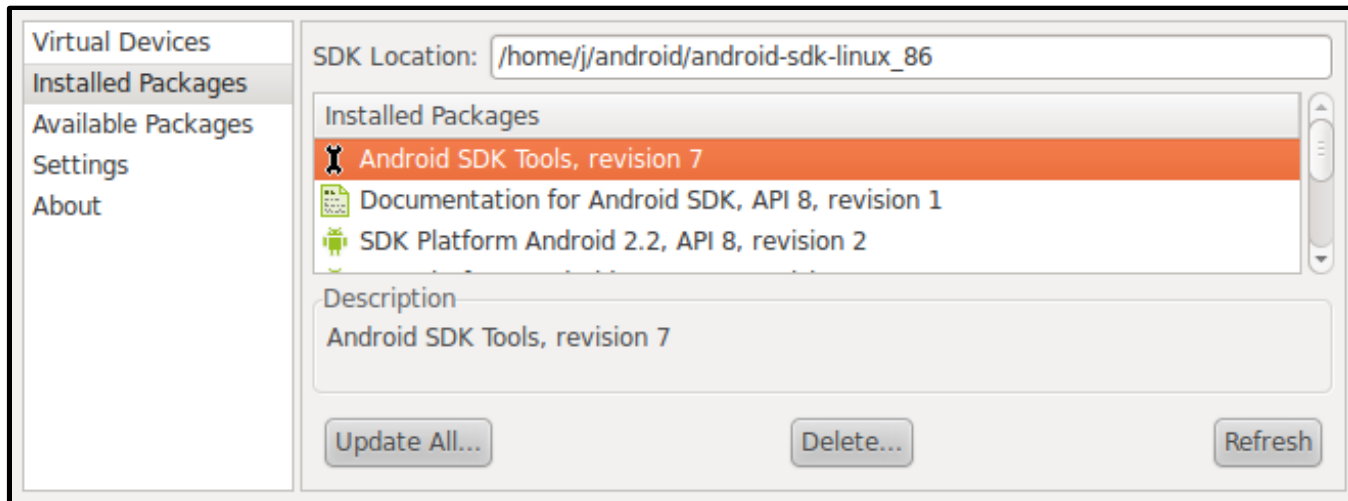
Arkitektur



<http://developer.android.com/guide/basics/what-is-android.html>

Android SDK

- Se <http://developer.android.com/sdk/>
 - Linux/Mac: Start kommandoen 'android'
 - Windows: Start 'SDK Manager.exe'
 - Eclipse: Window > Android SDK and AVD Manager



Android Virtual Devices

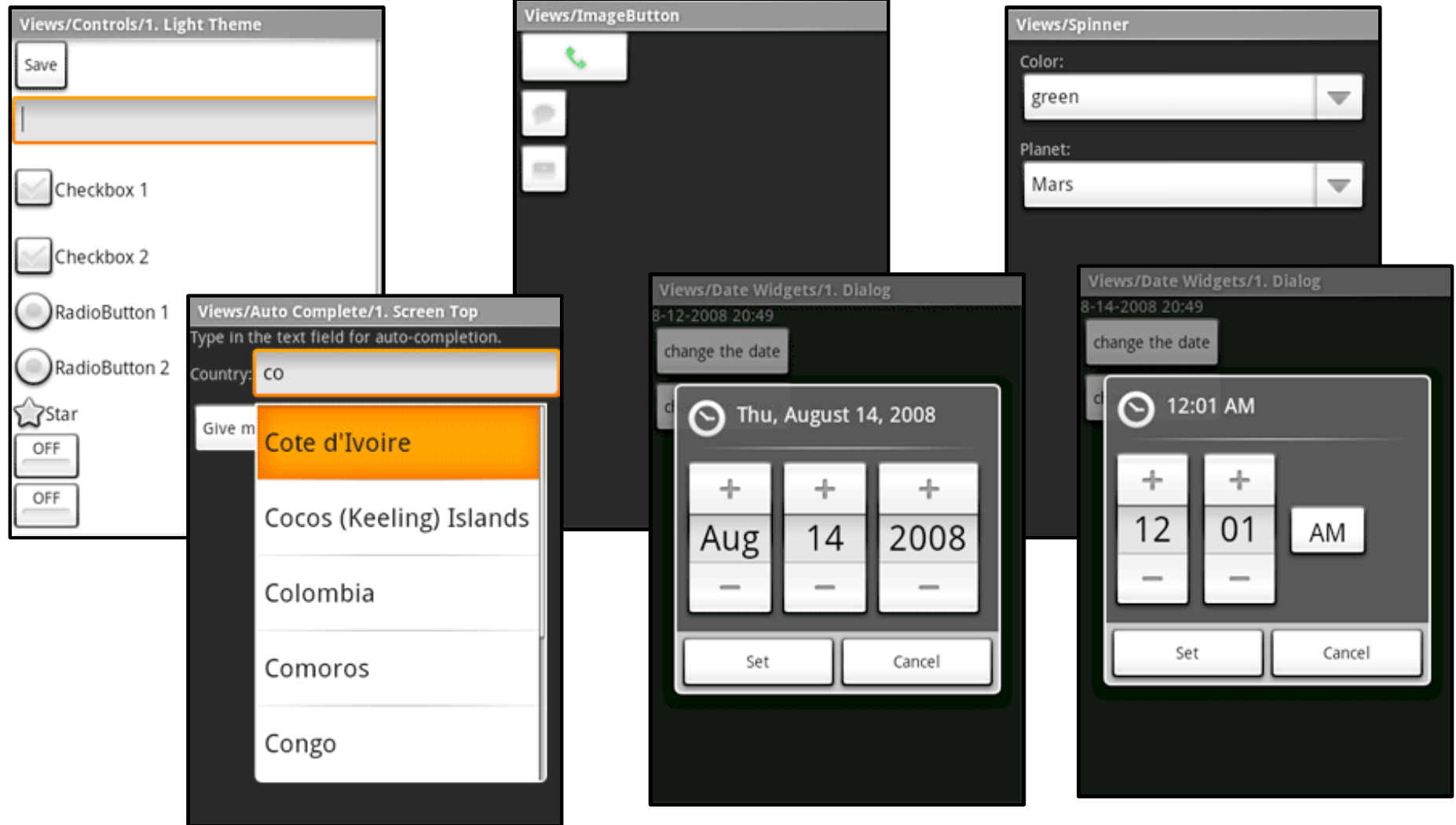
The screenshot displays the Android Studio AVD Manager interface. The main window is titled 'Name: Android' and 'Target: Android 2.2 - API Level 8'. The 'SD Card' section has 'Size' selected with a 'MiB' dropdown and a 'Browse...' button. The 'Skin' section has 'Built-in: Default (HVGA)' selected. The 'Hardware' section shows a table with 'Property' and 'Value' columns, containing 'Abstracted LCD density' with a value of '160'. At the bottom, there are 'Create AVD' and 'Cancel' buttons.

Overlaid on the bottom left is a 'Virtual Devices' window showing a list of existing devices:

AVD Name	Target Name	Platform	API Level
✓ Android1.6	Android 1.6	1.6	4
✓ and22_sd	Android 2.2	2.2	8
✓ and22_g_sd	Google APIs (Google Inc.)	2.2	8

Below the table, there are three buttons: 'New...', 'Delete...', and 'Repair...'. A legend at the bottom explains the status icons: a green checkmark for a valid device, a green checkmark with a document icon for a repairable device, and a red X for a device that failed to load.

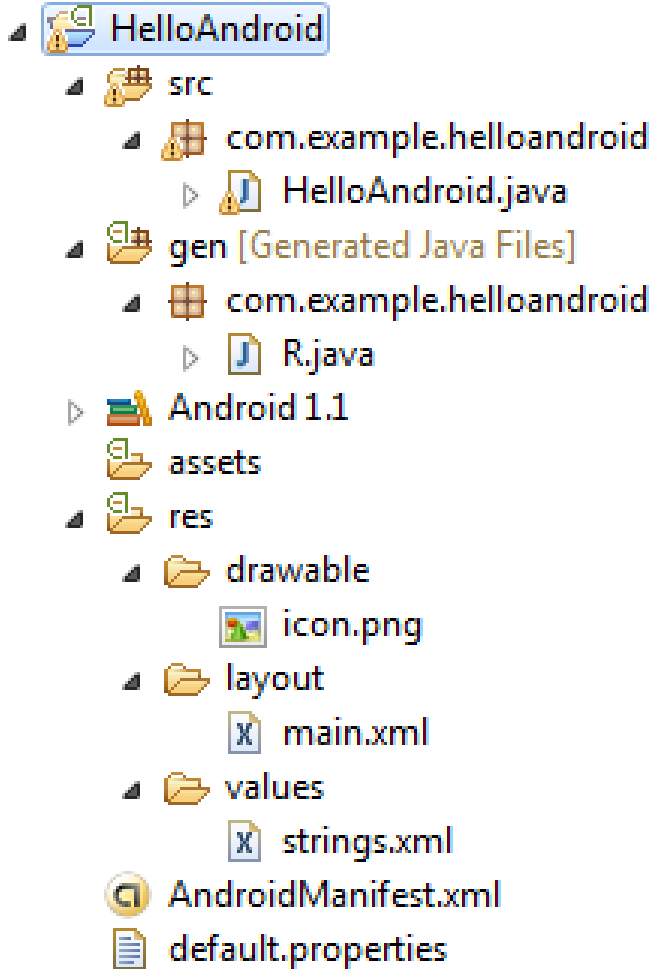
Eksempler på views



Views

- Button, CheckBox, RadioButton, ImageButton, TextView, EditText, Spinner, ListView, ImageView, WebView, VideoView
- Mindre brugte:
 - ToggleButton, AutoCompleteTextView, SurfaceView, GLSurfaceView, AnalogClock, DigitalClock, KeyboardView, Chronometer, DatePicker, TimePicker, ProgressBar, RatingBar, SeekBar, VideoView, ExpandableListView, ViewFlipper, ZoomButton

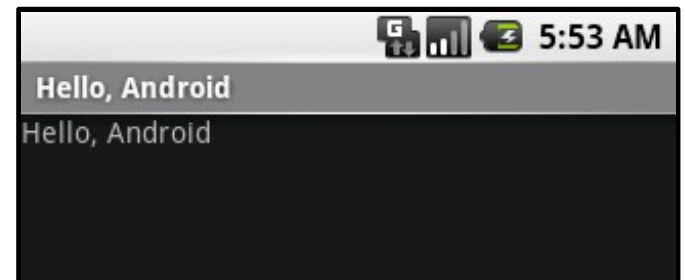
Programmatisk brugerfl.



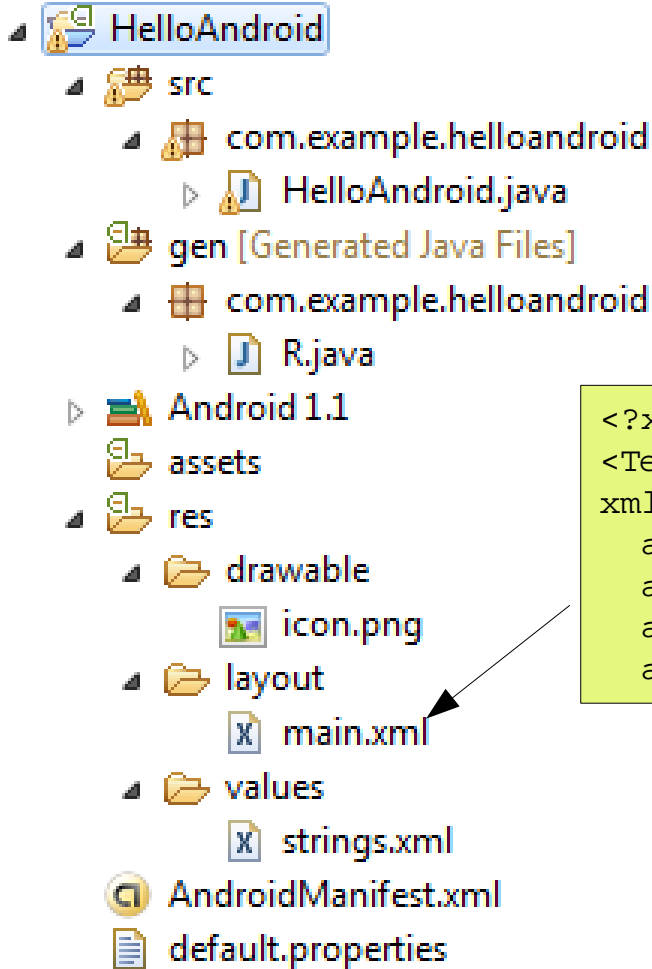
```
package com.example.helloandroid;

import android.app.Activity;
import android.os.Bundle;
import android.widget.TextView;

public class HelloAndroid extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        TextView tv = new TextView(this);
        tv.setText("Hello, Android");
        setContentView(tv);
    }
}
```

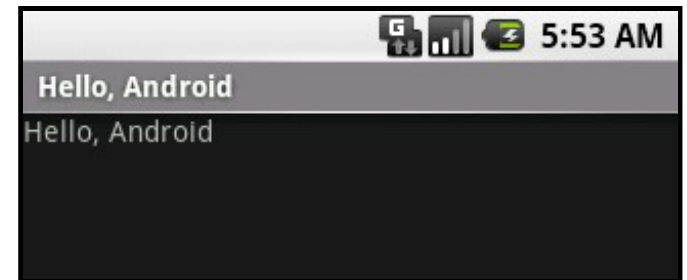


Deklarativ brugerflade

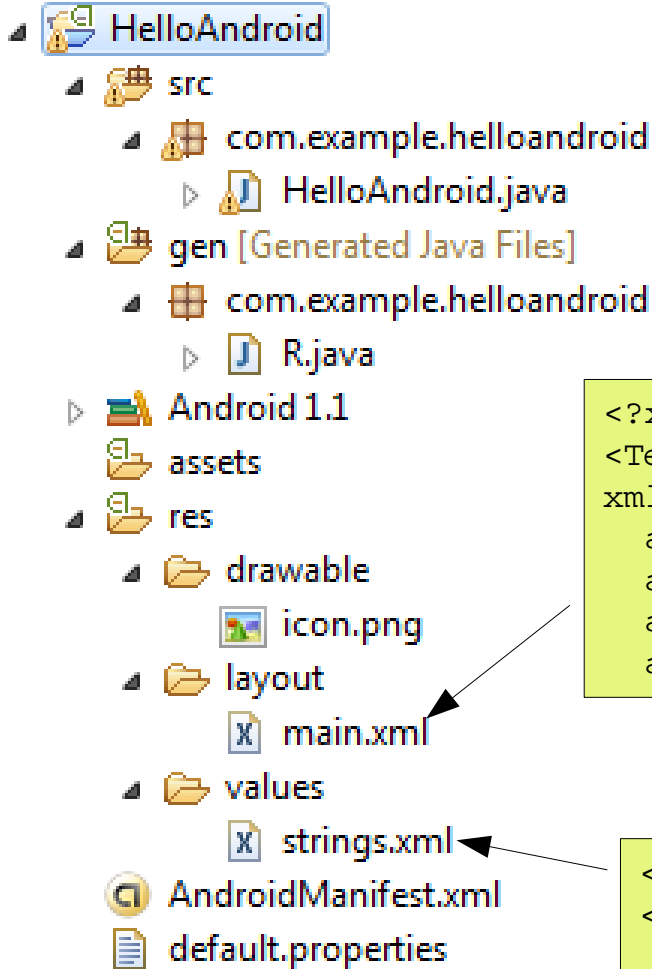


```
public class HelloAndroid extends Activity {  
    @Override  
    public void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.main);  
    }  
}
```

```
<?xml version="1.0" encoding="utf-8"?>  
<TextView  
    xmlns:android="http://schemas.android.com/apk/res/android"  
    android:id="@+id/textview"  
    android:layout_width="fill_parent"  
    android:layout_height="fill_parent"  
    android:text="Hello, Android" />
```



- med krydshenvisninger

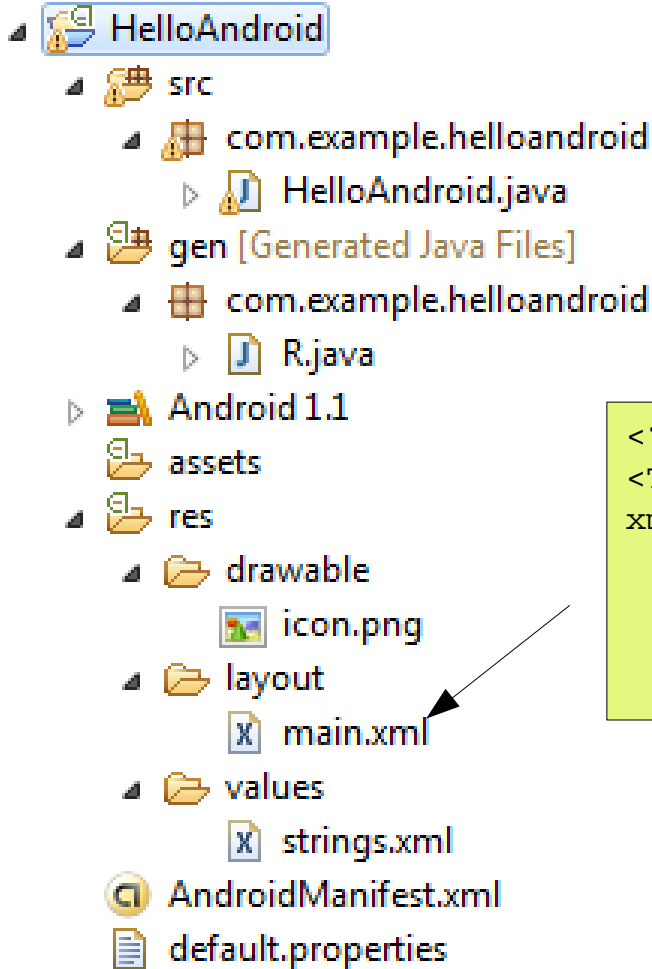


```
public class HelloAndroid extends Activity {
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
    }
}
```

```
<?xml version="1.0" encoding="utf-8"?>
<TextView
xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/textview"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:text="@string/hello" />
```

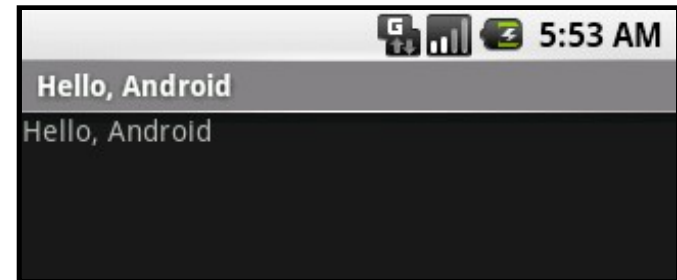
```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string name="hello">Hello, Android</string>
</resources>
```

Opslag med R-filen



```
public class HelloAndroid extends Activity {
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
        TextView tv = (TextView)
            findViewById(R.id.textview);
    }
}
```

```
<?xml version="1.0" encoding="utf-8"?>
<TextView
xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/textview"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:text="Hello, Android"/>
```



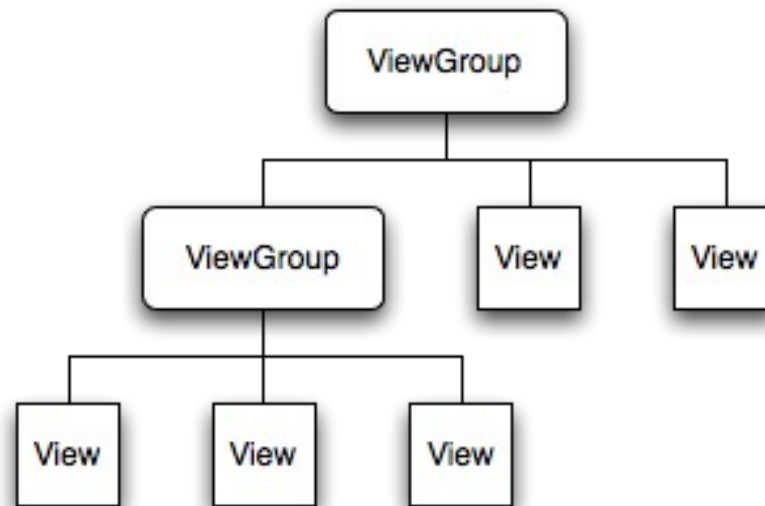
Lytte efter hændelser

```
public class HelloAndroid extends Activity implements OnClickListener {
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
        TextView tv = (TextView) findViewById(R.id.textview);
        tv.setOnClickListener(this);
    }

    public void onClick(View viewet) {
        // evt.: if (viewet.getId() == R.id.textview) { ... }
        ...
        TextView tv = (TextView) findViewById(R.id.textview);
        tv.setText(Html.fromHtml("Det var <b>flot</b>."));
    }
}
```

Grupper af views

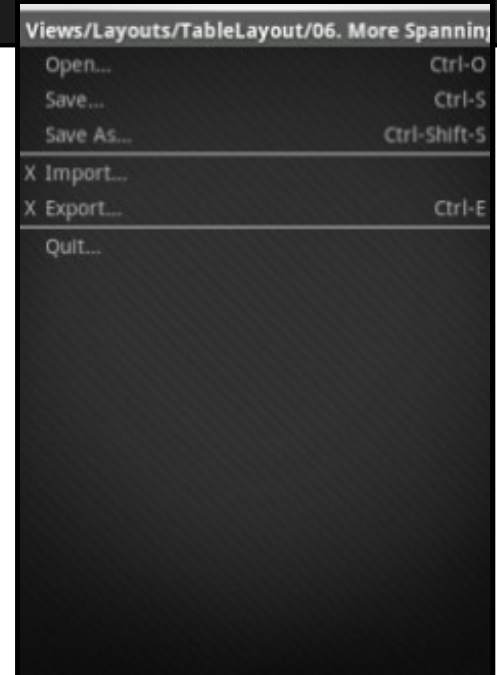
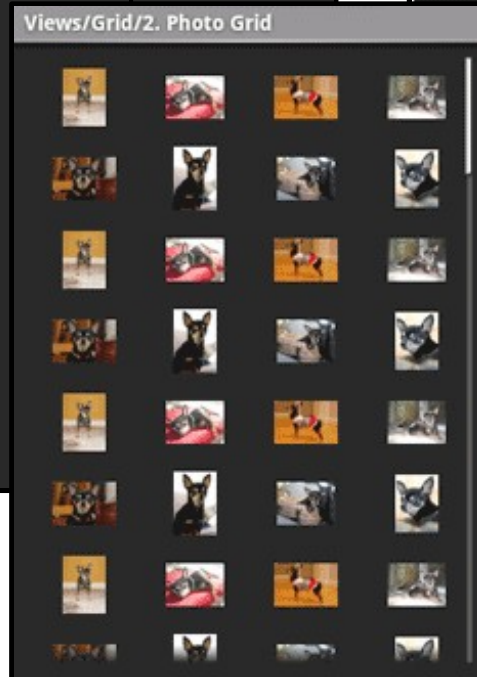
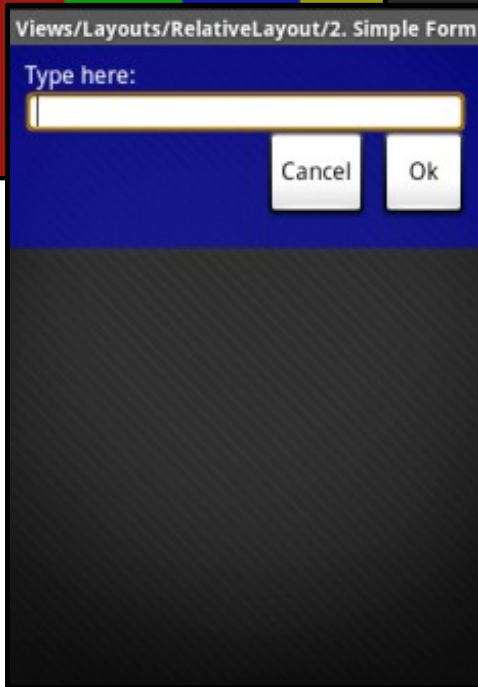
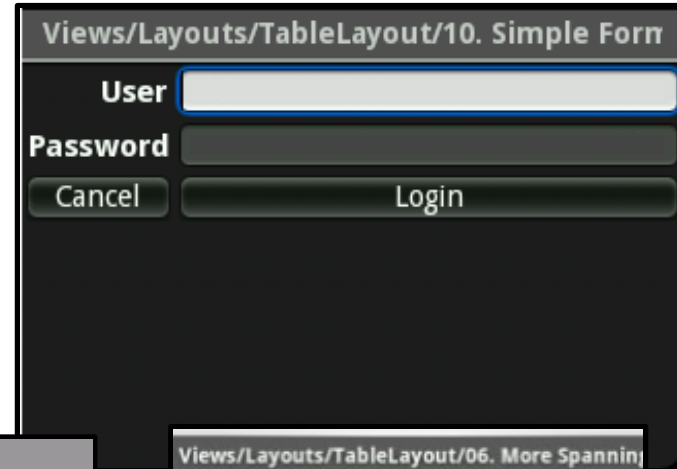
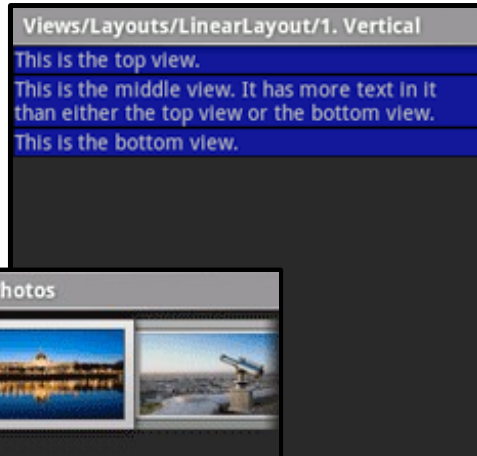
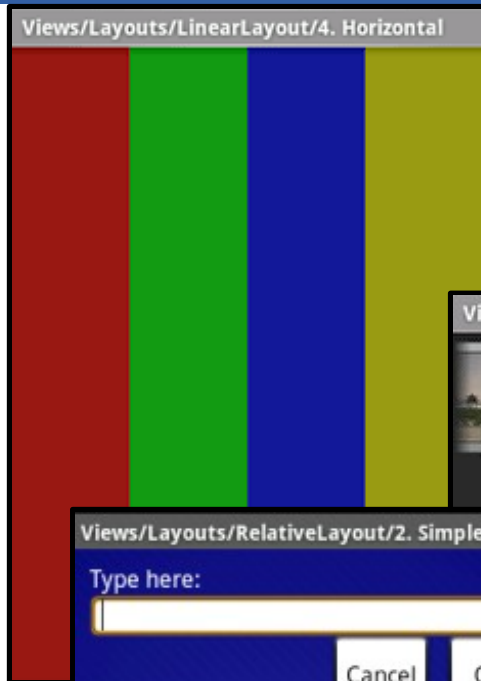
- View \approx java.awt.Component
- ViewGroup \approx java.awt.Container + layout



Viewgroups

- LinearLayout, TableLayout, TableRow, FrameLayout, RelativeLayout, ScrollView
 - Andre: AbsoluteLayout, TabHost, TabWidget, Gallery, GridView, HorizontalScrollView, ImageSwitcher, ListView, MediaController, RadioGroup, TextSwitcher, ViewFlipper
- Se oversigt på <http://droiddraw.org>

Viewgroups



Viewgroups

- **LinearLayout**
 - Lægger elementerne på en vandret række
 - Lodret layout fås med XML `android:orientation="vertical"`
- **TableLayout**
 - Hvert element kommer i sin egen række
 - Brug `LinearLayout` eller `TableRow` til rækker
 - `TableRow` justerer elementernes så de står i kolonner
- **RelativeLayout**
 - Positionerer elementer relativt i forhold til hinanden eller i forhold til det omgivende view

Viewgroups

- ScrollView
 - Kan kun indeholde ét element. Laver en lodret scrollbar hvis viewet er højere end skærmen.
 - Indeholder typisk et TableLayout med underelementer
- FrameLayout
 - Hvert view lægges i et nyt 'lag' oven over det forrige
 - Kan f.eks. bruges til at lægge billede bagved et layout
- AbsoluteLayout
 - Absolut positionering (med x- og y-koordinater og bredde og højde i pixels) frarådes

Viewgroups

- GridView
 - Viser elementer i et rektangulært regelmæssigt gitter
- Gallery
 - Viser et billedegalleri
- TabHost
 - Viser et sæt faneblade (TabWidget) og faneindhold

Deklarativ GUI

```
<ScrollView>
  <TableLayout>
    <TableRow>
      <TextView android:text="Vejret for" />
      <EditText android:text="2500"
        android:singleLine="true"
        android:textColor="blue"
        android:id="@+id/postnrEditText" />
    </TableRow>
    <Button android:text="OK"
      android:id="@+id/okKnap" />
    <Button android:text="Annuller!"
      android:id="@+id/annullerKnap" />
    <WebView android:id="@+id/webView"
      android:layout_height="300px" />
  </TableLayout>
</ScrollView>
```

```
// Javakode i onCreate():
setContentView(R.id.mitlayout); // Hvis layout ligger i res/layout/mitlayout.xml
EditText postnrEditText = (EditText findViewById(R.id.postnrEditText);
Button okKnap= (Button) findViewById(R.id.okKnap);
Button annullerKnap = (Button) findViewById(R.id.annullerKnap);
WebView webView = (WebView) findViewById(R.id.webView);
```



Programmatisk GUI

```
TableLayout tableLayout=new TableLayout(this);
TableRow tableRow = new TableRow(this);
TextView textView = new TextView(this);
textView.setText("Vejret for ");
tableRow.addView(textView);
postnrEditText = new EditText(this);
postnrEditText.setText("2500");
postnrEditText.setSingleLine(true);
postnrEditText.setTextColor(Color.BLUE);
tableRow.addView(postnrEditText);
tableLayout.addView(tableRow);

Button okKnap=new Button(this);
okKnap.setText("OK");
tableLayout.addView(okKnap);

Button annullerKnap = new Button(this);
annullerKnap.setText("Annuller!");
tableLayout.addView(annullerKnap);

WebView webView = new WebView(this);
webView.loadUrl("http://javabog.dk");
tableLayout.addView(webView);
webView.getLayoutParams().height=300;

ScrollView scrollView = new ScrollView(this);
scrollView.addView(tableLayout);

setContentView(scrollView); // vis scrollView'et
```



AndroidManifest.xml

- Bygges ind i Android-pakken (.apk-filen)
- Alle app'ens komponenter skal erklæres i manifestet
- Navngiver biblioteker som app anvender
- Erklærer tilladelser som app'en behøver

AndroidManifest.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="mit.pakkenavn">

    <uses-permission android:name="android.permission.VIBRATE"/>
    <uses-permission android:name="android.permission.INTERNET"/>

    <application>

        <!-- aktivitet der er synlig på startskærmen -->
        <activity android:name="mit.pakkenavn.MinKlasse" android:label="Navn">
            <intent-filter>
                <action android:name="android.intent.action.MAIN"/>
                <category android:name="android.intent.category.LAUNCHER"/>
            </intent-filter>
        </activity>

        <!-- interne aktiviteter -->
        <activity android:name="mit.pakkenavn.MinKlasse2" android:label="Navn2"/>
        <activity android:name="mit.pakkenavn.MinKlasse3" android:label="Navn3"/>

        <!-- pakkenavn kan udelades hvis det er det samme som app'ens -->
        <activity android:name=".MinKlasse4" android:label="Navn4"/>
    </application>
</manifest>
```

Skifte aktivitet

- Det er systemet der styrer aktiviteterne
 - Husk at alle aktiviteter skal være defineret i AndroidManifest.xml
- Du kan bede om at skifte aktivitet med et intent ('URI på steroider')

```
Intent i = new Intent(this, MinAkt2.class)
this.startActivity(i);
```

Sende data med intent

- Nøgle-værdipar kan sendes med intent

`i.putExtra(String nøgle, værdi)`

```
// Afsender
Intent i = new Intent(this, MinAkt2.class)
i.putExtra("navn", "Jacob")
i.putExtra("alder", 40)
this.startActivity(i);
```

- Aflæses i ny aktivitet med

`getIntent().getExtras().getString(nøgle);`

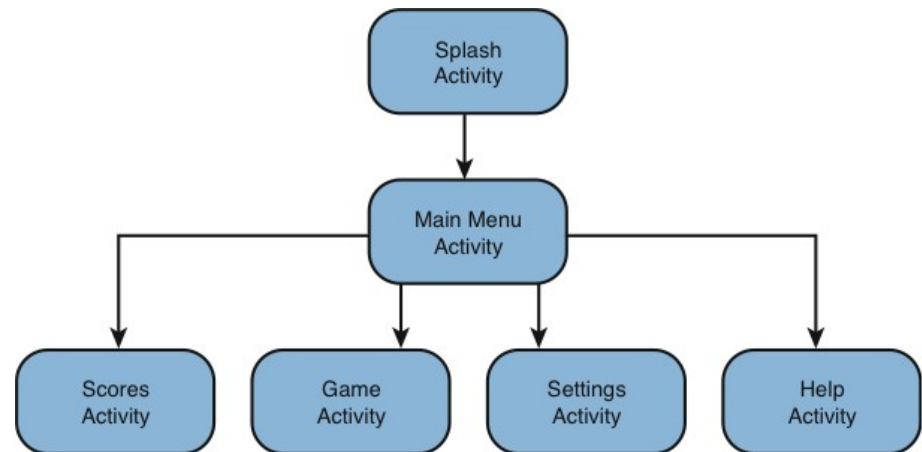
```
// Modtager (MinAkt2)
Intent i = getIntent();
String navn = i.getExtras().getString("navn");
int alder = i.getExtras().getInt("alder");
```

Dagens program

- Oplæg: I gang med Android
- Første udfordring**
- Oplæg: Elementer i Android
- Jamsession
- Afrunding

Første udfordring

- Tegn en oversigt over aktiviteter i jeres app
 - Opret nyt projekt med 3 af aktiviteterne
 - Det er en god idé at lade deres navne ende med f.eks. '_akt' eller lignende
 - Lav knapper der går fra en aktivitet til de andre



Dagens program

- Oplæg: I gang med Android
- Første udfordring
- Oplæg: Elementer i Android
- Jamsession
- Afrunding

Et views egenskaber

- Viewets egne egenskaber hedder `android:<navn>`
 - om hvordan viewet ser ud inden i og hvordan tingene placeres inde i viewet
 - ændrer ikke på det omsluttende layout

```
<TextView
  android:text="0:27 minutter tilbage"
  android:textColor="#FFFFFFFF"
  android:focusable="false"
  android:gravity="right"
/>
```

Views layoutegenskaber

- Det omsluttende layouts egenskaber hedder `android:layout_<navn>`
 - Disse egenskaber handler alle om hvordan viewet placeres og strækkes i det omsluttende layout
 - Dvs de bestemmer viewets plads og størrelse

```
<TextView
  android:text="0:27 minutter tilbage"
  android:textColor="#FFFFFFFF"
  android:focusable="false"
  android:gravity="right"
  android:layout_gravity="right"
  android:layout_width="wrap_content"
  android:layout_height="wrap_content"/>
```

Information til omsluttende
layout

Views layoutegenskaber

- Bredde og højde
 - Egenskaberne `android:layout_width`, `android:layout_height`, f.eks.
 - `android:layout_height="10px"` - 10 pixels (billedpunkter)
 - `android:layout_height="10mm"` - 10 millimeter
 - `android:layout_height="10dp"` - 10 density-independent pixels
 - _ højopløselige skærme har 1dp=2px, på lavopl er 1dp=px ("`10dip`" kan også bruges)
 - `android:layout_height="fill_parent"` (også kaldet "`match_parent`")
 - _ Viewet skal være lige så stort som det omsluttende layout
 - "`wrap_content`": Viewet skal være så småt som muligt
- Hvis der er plads til overs i det omsluttende layout
 - `android:layout_weight` giver vægten som hvert view skal strækkes
 - _ f.eks 1/4-del og 3/4-del: `android:layout_weight="1"` og `android:layout_weight="3"`
 - `android:layout_gravity` giver hvor hvert view skal placeres i forhold til den strukne plads
 - _ f.eks. `android:layout_gravity="right"`, "`center`" etc

WebView

- Et View som alle andre
 - Kan indlejres i en ViewGroup, såsom LinearLayout
 - Kan erklæres i XML eller i Java
- Vise URL
 - `webView.loadUrl("http://javabog.dk");`
 - `webView.loadUrl("file:///android_asset/index.html");`
- JavaScript kan aktiveres
 - `webView.getSettings().setJavaScriptEnabled(true);`
- JavaScript-funktioner kan aktiveres fra Java
 - `webView.loadUrl("javascript:alert('Hej')");`
- Java-objekt kan få kaldt metoder fra JavaScript
 - `MinKlasse min = new MinKlasse();`
 - `min.metode(); // fra Java`
 - `webView.addJavascriptInterface(min, "min");`
- Fra Javascript:
 - ``

Lyd - mediaplayer

- Læg fil i res/raw, f.eks res/raw/musik.mp3
- Husk at frigive resurser

```
MediaPlayer mp;

protected void onStart() {
    mp = MediaPlayer.create(this, R.raw.musik);
    mp.start();
}

protected void onStop() {
    mp.stop();
    mp.release();
    mp = null;
}
```

Video - VideoView

```
// overfør først videoen til telefonens SDKort.  
// F.eks. med kommandoen:  
// adb push samplevideo.3gp /sdcard/samplevideo.3gp  
  
public class VideoTest extends Activity {  
  
    public void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
  
        VideoView video = new VideoView(this);  
        this.setContentview(video);  
  
        video.setVideoPath("/sdcard/samplevideo.3gp" );  
        video.start();  
    }  
}
```

Egne views

- View - en grafisk komponent på skærmen
 - Lav en nedarving fra View
 - Definér metoden onDraw(Canvas c)
 - Android kalder onDraw() når viewet skal tegnes på skærmen

```
class GrafikView extends View {  
  
    public GrafikView(Activity a) { super(a); }  
  
    protected void onDraw(Canvas c) {  
        Paint tekstStregtype = new Paint();  
        tekstStregtype.setColor(Color.GREEN);  
        tekstStregtype.setTextSize(24);  
        c.drawText("Hej verden", 0, 20, tekstStregtype);  
    }  
}
```

Elementer i grafiktegnning

- View - en grafisk komponent på skærmen
 - Android kalder metoden `onDraw(Canvas c)`
- Canvas - lærredet
 - Repræsenterer objektet der tegnes 'på'
 - Du kan skalere, rotere, translaterere det
 - Øverste venstre hjørne er (0,0)
- Paint - stregtypen
 - Farve, gennemsigtighed, antialias, dither, stregtype, udfyldning, interpolering ved kurveknæk, afslutning ved ende af linje

Grafiktegning

- Farve - og gennemsigtighed
 - Sættes på Paint-objektet
 - Repræsenteres som et heltal (int på) 4 byte
 - Hver byte repræsenterer en farvekanal
 - gennemsigtighed (0=gennemsigtig, 255=uigennemsigtig)
 - rød kanal
 - grøn kanal
 - blå kanal
 - Farven 'uigennemsigtig blå' kan beskrives på 4 måder:
 - Konstant: `Color.BLUE`
 - Decimal: $255 * 256 * 256 * 256 + 255$
 - Ved hjælp af hjælpemetode: `Color.argb(255, 0, 0, 255)`
 - Hexadecimalt: `0xff0000ff`

Menuer

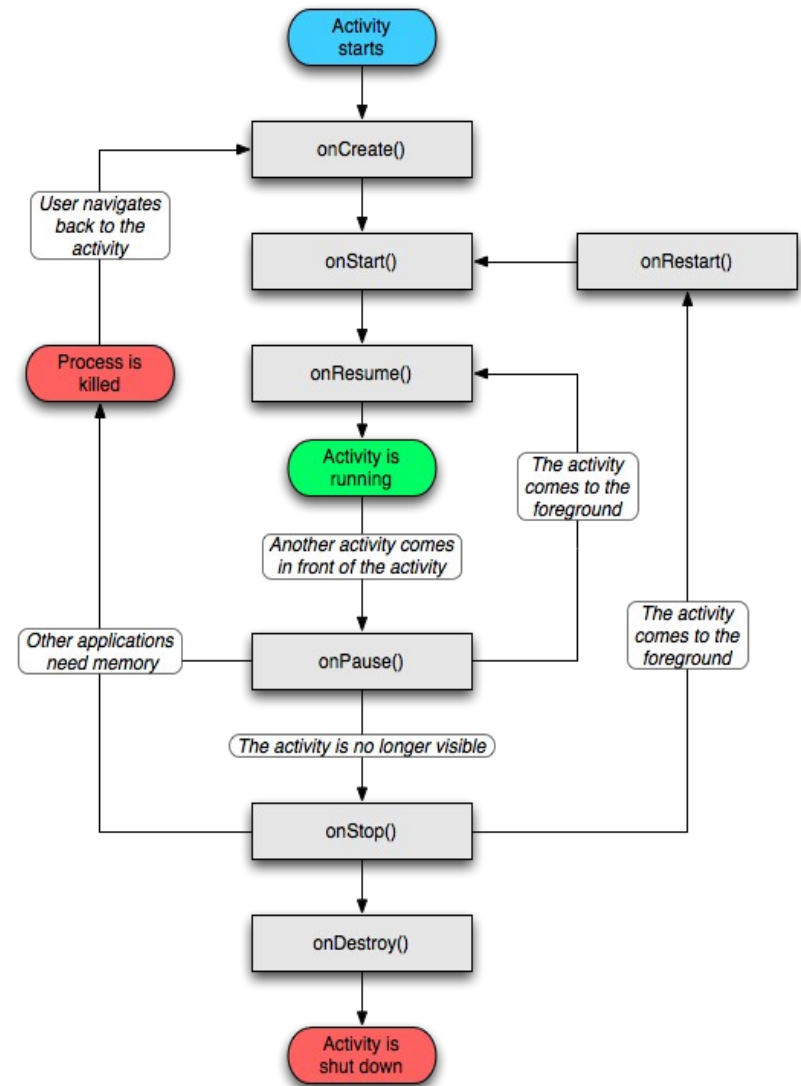
```
public class Vejrtaktivitet extends Activity
{
    ...

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        super.onCreateOptionsMenu(menu);
        menu.add(0, 43, 0, "Indstillinger");
        menu.add(0, 45, 0, "Vælg by");
        menu.add(0, 47, 0, "Vejrkort");
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        switch (item.getItemId()) {
            case 43:
                startActivity(new Intent(this, Indstillinger.class));
                return true;
            case 45:
                visVælgByDialog();
                return true;
            case 47:
                startActivity(new Intent(this, VejrkortAktivitet.class));
                return true;
        }
        return false;
    }
}
```

En aktivitets livscyklus

- En aktivitet kan være
 - Oprettet
 - onCreate()
 - onDestroy()
 - Synlig
 - onStart()
 - onStop()
 - Forrest
 - onResume()
 - onPause()



Livscyklusmetoderne

```
public class LivscyklusAktivitet extends Activity {
    private static final String TAG = "Livscyklus";

    protected void onCreate(Bundle savedInstanceState)
    {
        Log.d(TAG, "onCreate() "+savedInstanceState);
        super.onCreate(savedInstanceState);
    }

    protected void onStart()
    { Log.d(TAG, "onStart()"); super.onStart(); }

    protected void onRestart()
    { Log.d(TAG, "onRestart()"); super.onRestart(); }

    protected void onResume()
    { Log.d(TAG, "onResume()"); super.onResume(); }

    protected void onPause()
    { Log.d(TAG, "onPause()"); super.onPause(); }

    protected void onStop()
    { Log.d(TAG, "onStop()"); super.onStop(); }

    protected void onDestroy()
    { Log.d(TAG, "onDestroy()"); super.onDestroy(); }
```

Når skærmen vendes

- Når skærmen vendes eller tastaturet bliver skubbet ind siger man der er sket en ændring i telefonens konfiguration
 - Så skal komponenterne have ændret størrelse
 - Evt skal et andet layout vises (layout-land)
 - Evt skærmtastatur skal vises/skjules
- Normalt vil systemet oprette et nyt aktivitets-objekt oprettes med de nye skærmdimensioner etc.
 - Dvs det gamle aktivitets-objekt med dine data bliver kasseret!
 - Systemet sørger dog for at overføre data fra alle views med et id til den nye aktivitet
- Du har som udvikler to muligheder:
 - Slå det fra (så et nyt aktivitets-objekt ikke bliver oprettet)
 - Så skal du håndtere ændringen i konfiguration manuelt
 - Gemme de nødvendige data så brugeren kan fortsætte fra hvor han slap

Gemme data

- Filer
 - Applikationens private filområde - File privatMappe = getFilesDir()
 - Applikationens private cacheområde - File cacheMappe = getCacheDir()
 - _ Kan blive slettet af systemet når der mangler plads
 - Hukommelseskortet - getExternalStorageDirectory()
 - _ Gem private filer i /Android/data/<pakkenavn>/files/
 - _ Gem cachefiler i /Android/data/<pakkenavn>/cache/
 - _ Fælles filer i roden eller i Music/, Movies/, Download/ etc
 - _ Fra API 8 (Android 2.2) findes getExternalFilesDir(), getExternalCacheDir() etc.
- I database på telefonen
 - _ SQLite
- Over nettet, på (egen) server
 - _ fra API 8: understøttelse for backup til 'skyen' (f.eks. Googles egne servere)
- Indstillinger
 - _ Også 'indstillinger' brugeren aldrig ser

PreferenceManager

- Ofte den nemmeste løsning
- Forudsætter ikke at brugeren faktisk ser nogle indstillinger

- Læsning

```
PreferenceManager.getDefaultSharedPreferences(this)  
    .getBoolean("brugPosVedStart", true);
```

- Skrivning

```
PreferenceManager.getDefaultSharedPreferences(this)  
    .edit().putBoolean("brugPosVedStart", true).commit();
```

Indstillinger

- Opret res/xml/indstillinger.xml

```
<?xml version="1.0" encoding="utf-8"?>
<PreferenceScreen
  xmlns:android="http://schemas.android.com/apk/res/android">

  <CheckBoxPreference
    android:key="brugPosVedStart"
    android:title="Brug stedbestemmelse ved start"
    android:summary="Vis vejrudsigt for aktuel position ved
    android:defaultValue="true" />

  <CheckBoxPreference
    android:key="ventPåPos"
    android:title="Vent på stedbestemmelse"
    android:summary="Vent på en frisk positionsbestemmelse (
bruges sidst seneste position til vejrudsigt)"
    android:defaultValue="false" />

  <EditTextPreference
    android:key="foretrukkenPostNr"
    android:title="Foretrukken postnummer"
    android:summary="Dit foretrukne postnummer"/>
  <PreferenceScreen android:title="En underskærm">
  ...
```



Aktivitet til indstillinger

- Opret arving fra PreferenceActivity
 - husk at opdatere AndroidManifest.xml

```
public class Indstillinger extends PreferenceActivity {  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        addPreferencesFromResource(R.xml.indstillinger);  
    }  
}
```

SQLite i en nøddeskal

```
// Oprettelse af database
SQLiteDatabase db = SQLiteDatabase.openOrCreateDatabase ("database.db", null)

// Oprette tabel - foregår via SQL
db.execSQL("CREATE TABLE kunder (_id INTEGER PRIMARY KEY AUTOINCREMENT, navn TEXT NOT
NULL, kredit INTEGER);");

// Oprette en række
ContentValues række = new ContentValues();
række.put("navn", "Jacob Nordfalk");
række.put("kredit", 500);
db.insert("kunder", null, række);

// Søgning
String[] kolonner = { "_id", "navn", "kredit" };
String valg = "kredit > 100"; // WHERE
String sortering = "kredit ASC"; // ORDER BY

Cursor cursor = db.query("kunder", kolonner, valg, null, null, null, sortering);

while (cursor.moveToNext()) {
    long id = cursor.getLong(0);
    String navn = cursor.getString(1);
    int kredit = cursor.getInt(2);
}
```

SQLiteOpenHelper

- Cursor bruges ofte til at vise lister
 - Men optager resurser så længe den er åben
 - Kan bringes til at følge en aktivitets livscyklus
`aktivitet.startManagingCursor(cursor);`
- Androids SQLiteOpenHelper
 - Hjælper med at oprette og administrere database
 - Versionering/opgradering
 - Skifte mellem skrivebeskyttet og ikke-skrivebeskyttede instanser
 - Følge aktivitets livscyklus

Content Providers

- Lader andre forespørge på data
- At bruge en content provider føles næsten som at bruge en SQLite-database
- Opslag af en content provider foregår gennem en URI (Uniform Resource Identifier):
 - content://authority/path
 - content://authority/path/id
 - authority = pakkenavn, f.eks. dk.nordfalk.teori
 - path = tabelnavn eller lignende
 - /id = indeks på individuelt element - (primær)nøgle

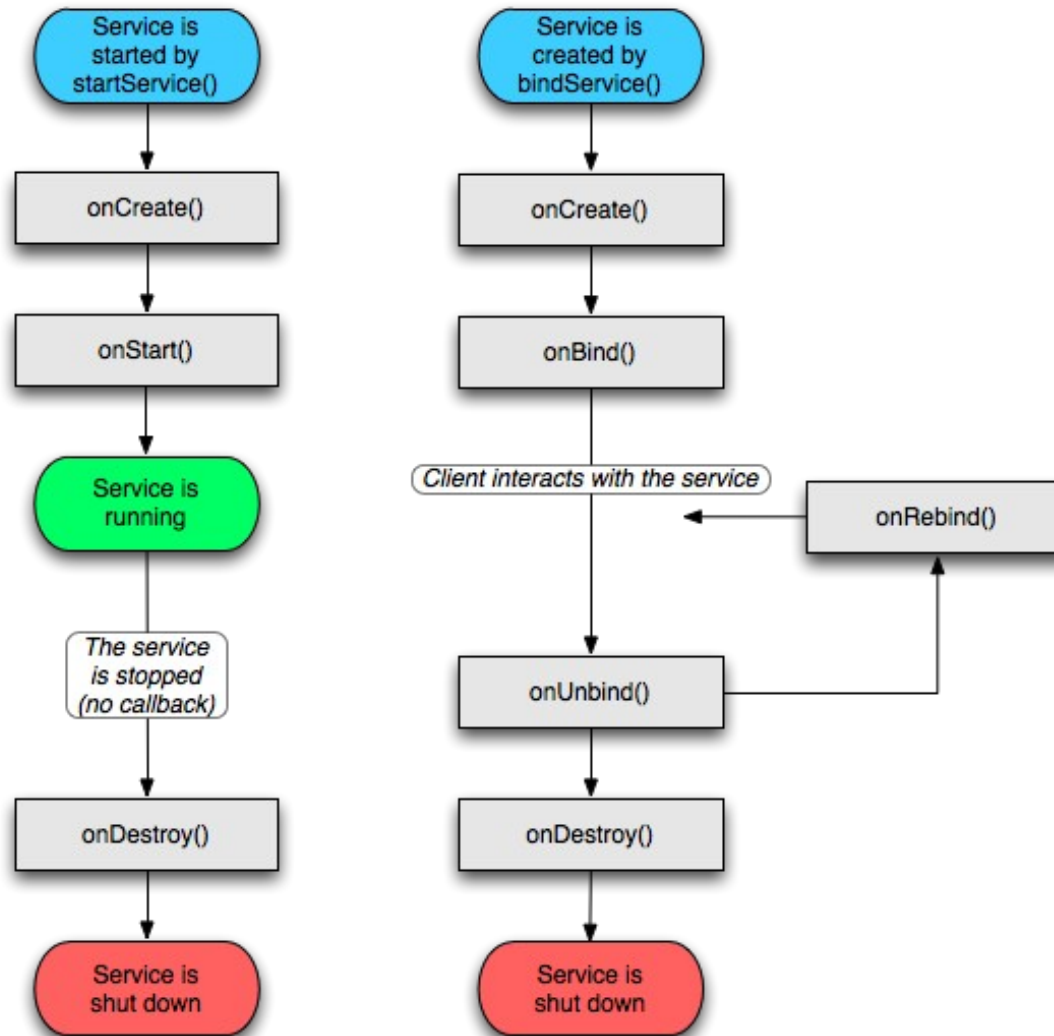
Service

- En service er en baggrundsaktivitet ('aktivitet uden en GUI')
 - Arver fra Context og ContextWrapper ligesom Activity
 - Har de samme muligheder som en aktivitet for at aflytte sensorer, starte intents etc
 - Skal erklæres i AndroidManifest.xml (med <service> tag)

```
<service android:name=".SmsService">  
  <intent-filter>  
    <action android:name="android.intent.action.BOOT_COMPLETED"/>  
  </intent-filter>  
</service>
```

- ligesom der er en startActivity(intent) er der en startService(intent)
- Der er 'garanti' for at systemet ikke lukker den
- Den kan kontaktes løbende
 - bindService (Intent service, ServiceConnection conn, int flags)
 - unbindService(ServiceConnection conn)
 - stopService (Intent service)

Livscyklus for en service



Broadcast Receiver

- Har kun én metode
 - void onReceive(Context curContext, Intent broadcastMsg)
- Metoden kaldes af processens GUI-tråd
 - Langvarige opgaver bør udføres i baggrundstråd
- Når metoden afslutter kan processen risikere at blive slået ihjel
 - Løsning: Start en service
- Kan erklæres i AndroidManifest.xml

```
<receiver android:name=".SmsReceiver">  
  <intent-filter>  
    <action android:name="android.provider.Telephony.SMS_RECEIVED"/>  
  </intent-filter>  
</receiver>
```

- Kan også registreres dynamisk med registerReceiver() og framelles med unregisterReceiver().

Nyttige tastaturgenveje

- Emulator:
 - Tilbageknappen - ESC
 - Skift orientering (vend skærm) - Ctrl-F11
 - Fuldskærm - Alt-retur
 - MENU-tast - F2 eller PgUp
 - Hjem-knap - Home
 - Skift mellem tidligere programmer - Hold Home nede 2 sek
 - Flytte fokus - Piletaster
 - Klikke - Retur

Dagens program

- Oplæg: I gang med Android
- Første udfordring
- Oplæg: Elementer i Android
- Jamsession**
- Afrunding

Jamsession - udfordringer

- Lav en aktivitet med et indtastningsfelt og en knap
- Lav en aktivitet, som aktiverer en anden aktivitet
- Vis en toast-besked til brugeren
- Lav en eventyrfortælling til Android (<http://javabog.dk/OOP/kapitel3.jsp#afsn3.5.2>)
- Afspil en lyd
- Lav en service
- Vis en notification til brugeren
- Gem noget data i SharedPreferences (og hent det igen)
- Vis en webside (f.eks. www.google.com)
- Lav en app, der parser noget XML og/eller JSON
- Kør noget kode asynkront
- Hent et nummer fra telefonbogen
- Send en SMS
- Vis en liste med 7-tabellen op til 1000
- Vis et kort (Google Maps)