

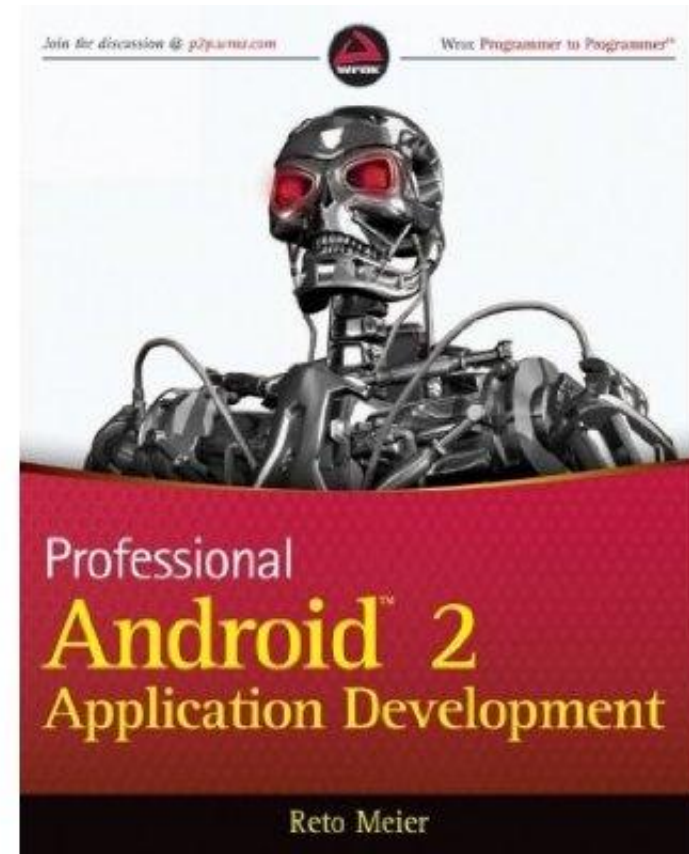
Android-Entwicklung



Michael „sECuRE” Stapelberg

Disclaimer

~~Folgen~~



Überblick

1 Die Android-Plattform

2 Entwicklungsumgebung

3 Grundlagen

Layout, Activity, Manifest, Listener, Intents

4 Framework

Adapter, Datenhaltung, Netzzugriff, Concurrency, GeoLocation



1

Die Android-Plattform

Android

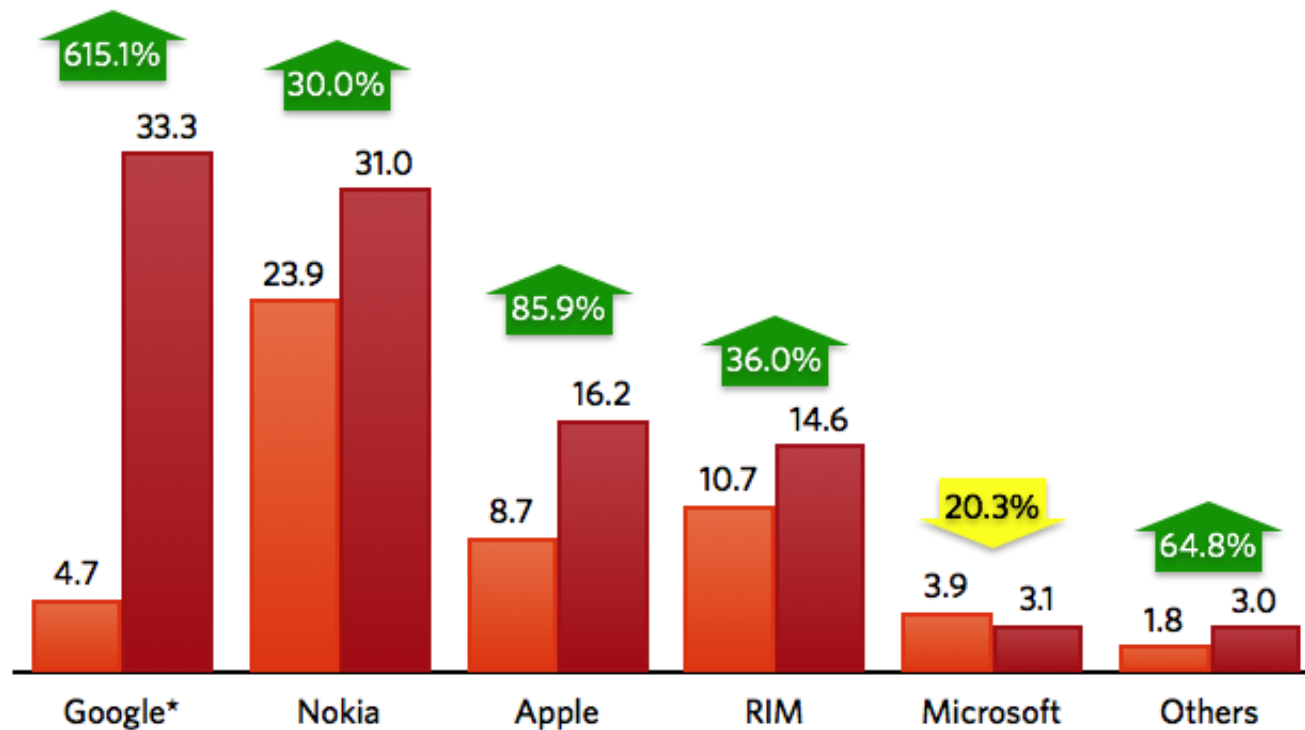
Linux-basiert. Smartphones, Tablets, Netbooks, ...

Java

Verbreitung

Worldwide smartphone market share
Millions of units shipped

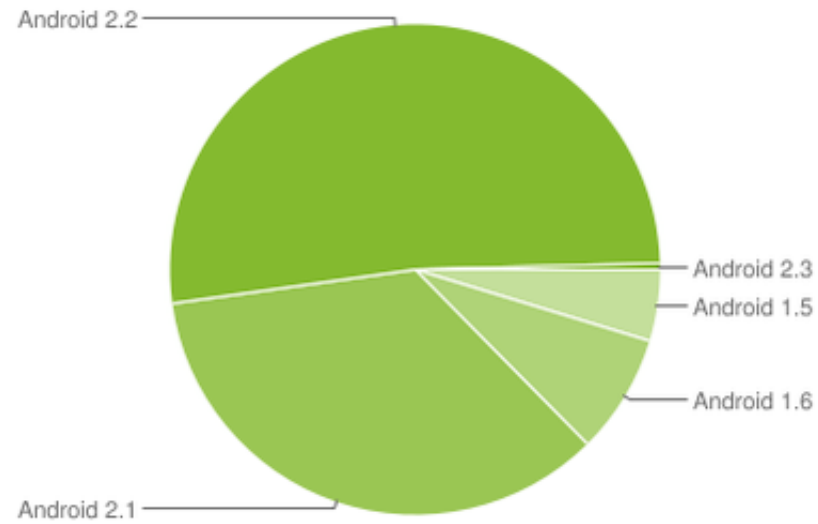
4Q 2009
4Q 2010



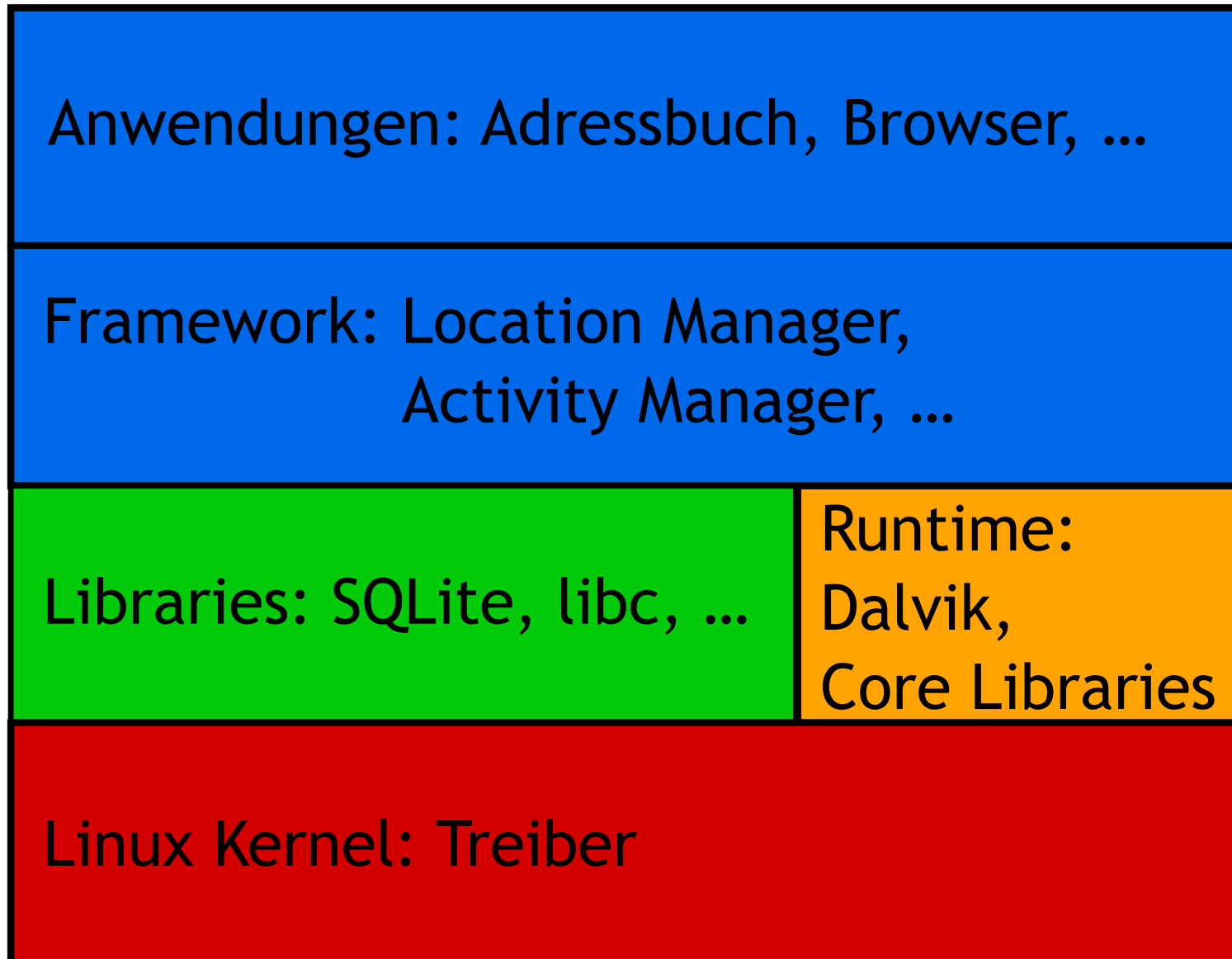
* Google numbers include Android and OMS, Tapas platform variants




Versionen



Android-Überblick





1

jetzt klar:
Wichtigste Aspekte
der Android-Plattform



2

Entwicklungsumgebung

Entwicklungsumgebung

Eclipse!

ADB (Android Debug Bridge)

Emulator



2

jetzt klar:
Entwicklungsumgebung



3

Grundlagen

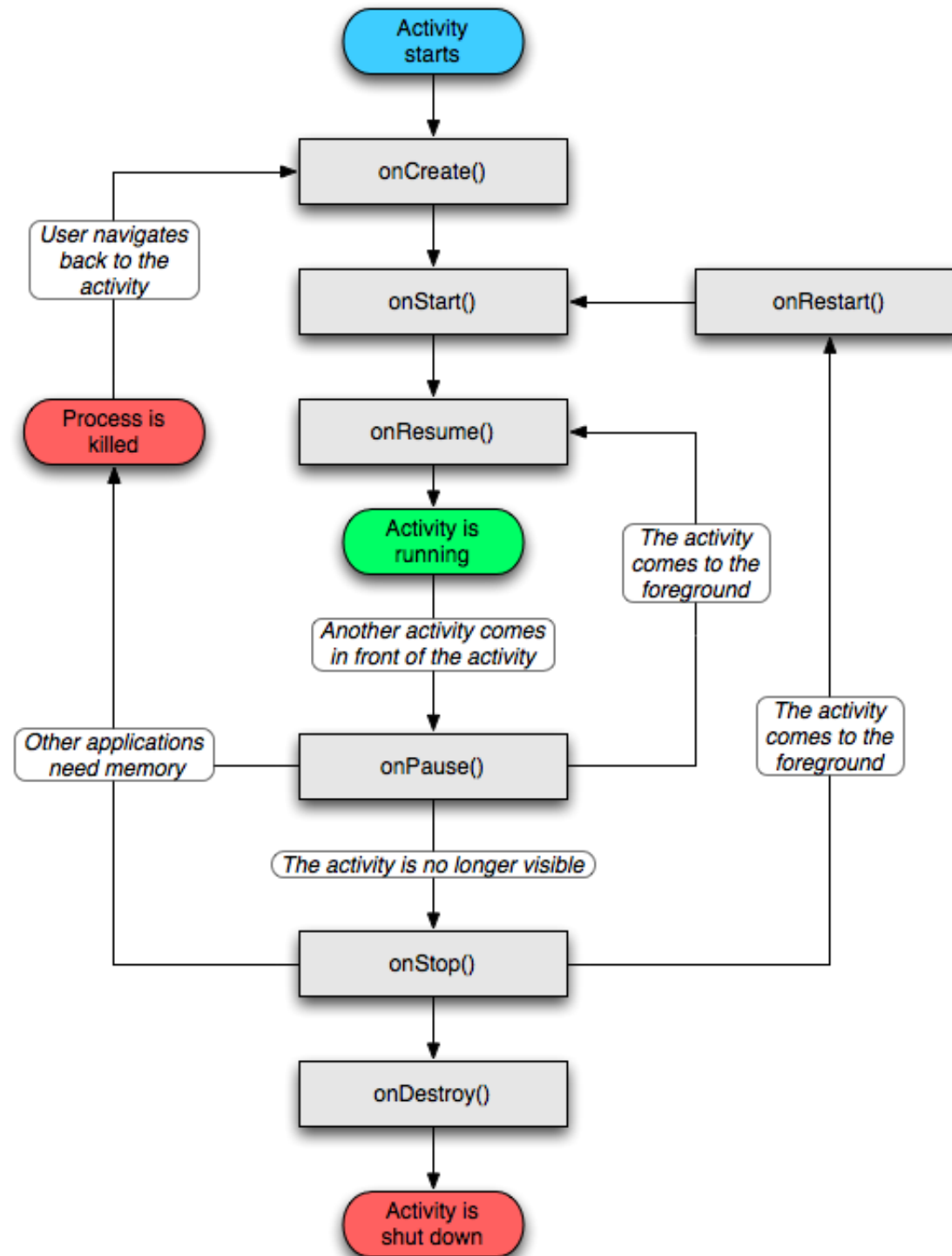
Activities

eine „Bildschirm-Ansicht“ für einen Zweck
(z.B. E-Mail anzeigen, E-Mail schreiben, ...)

genau eine zugehörige Klasse

fremde Activities

Lifecycle einer Activity



Layouts

Android-Eigenentwicklung

XML-Dateien in res/layouts/

als Activity oder Teil (z.B. Liste)

Views oder **ViewGroups**

Button

LinearLayout

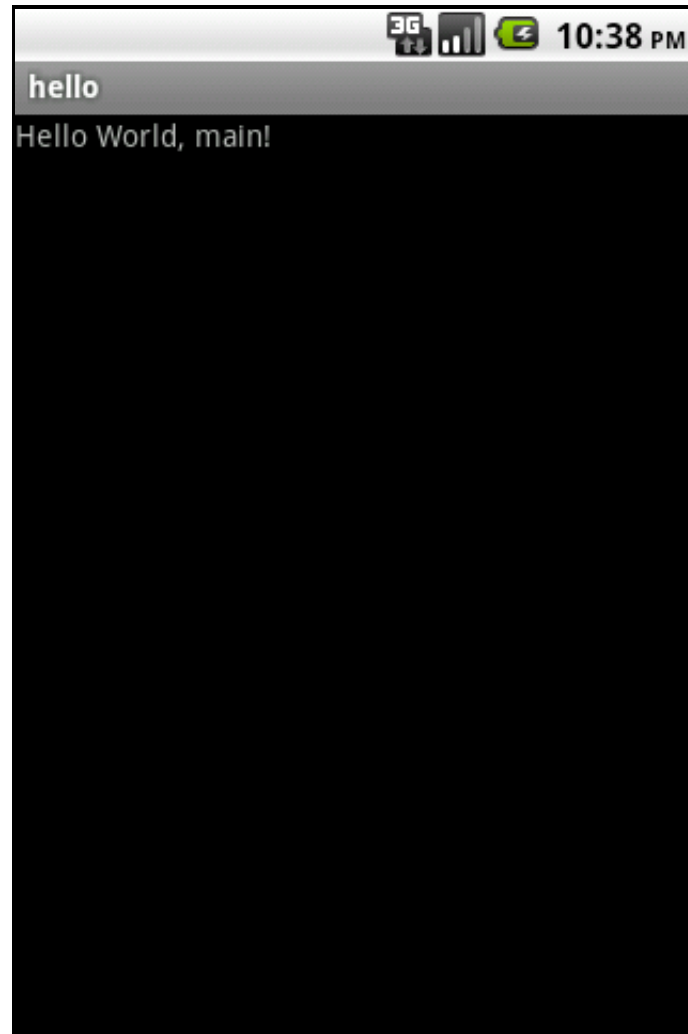
TextView

RelativeLayout

ImageView

EditText

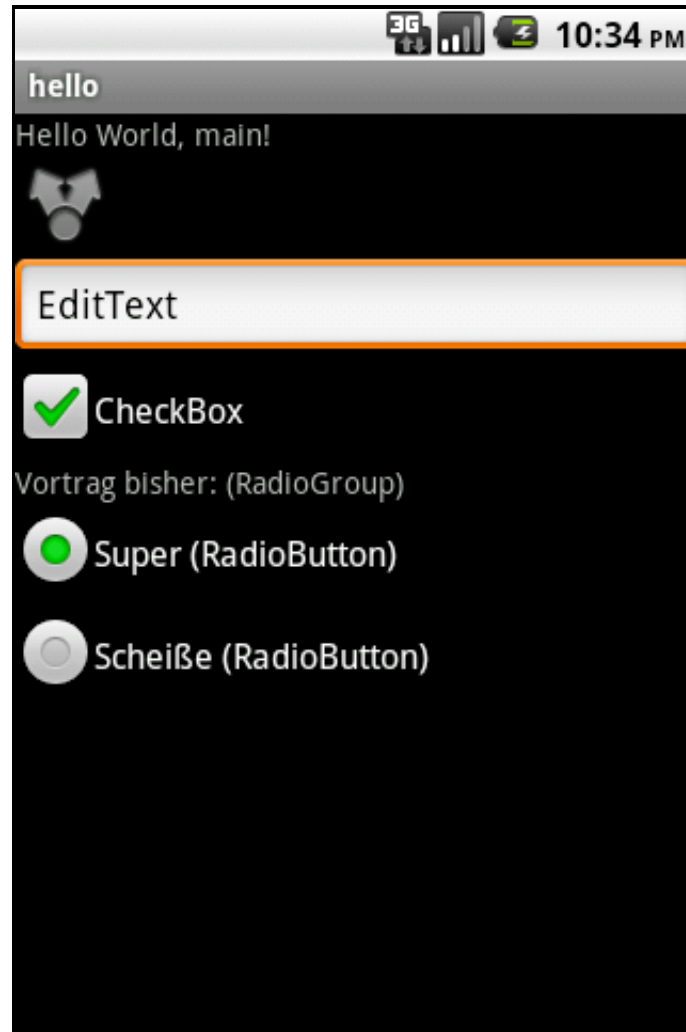
Hello world



Hello world (XML)

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

Views (Beispiele)



Siehe Resources → Tutorials → Hello Views

Listener

```
final TextView text =  
    (TextView)findViewById(R.id.ohai);  
  
text.setOnClickListener(new OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        Log.d("MeineApp", "Geklickt.");  
        text.setText("kthxbye");  
    }  
});
```

Activity

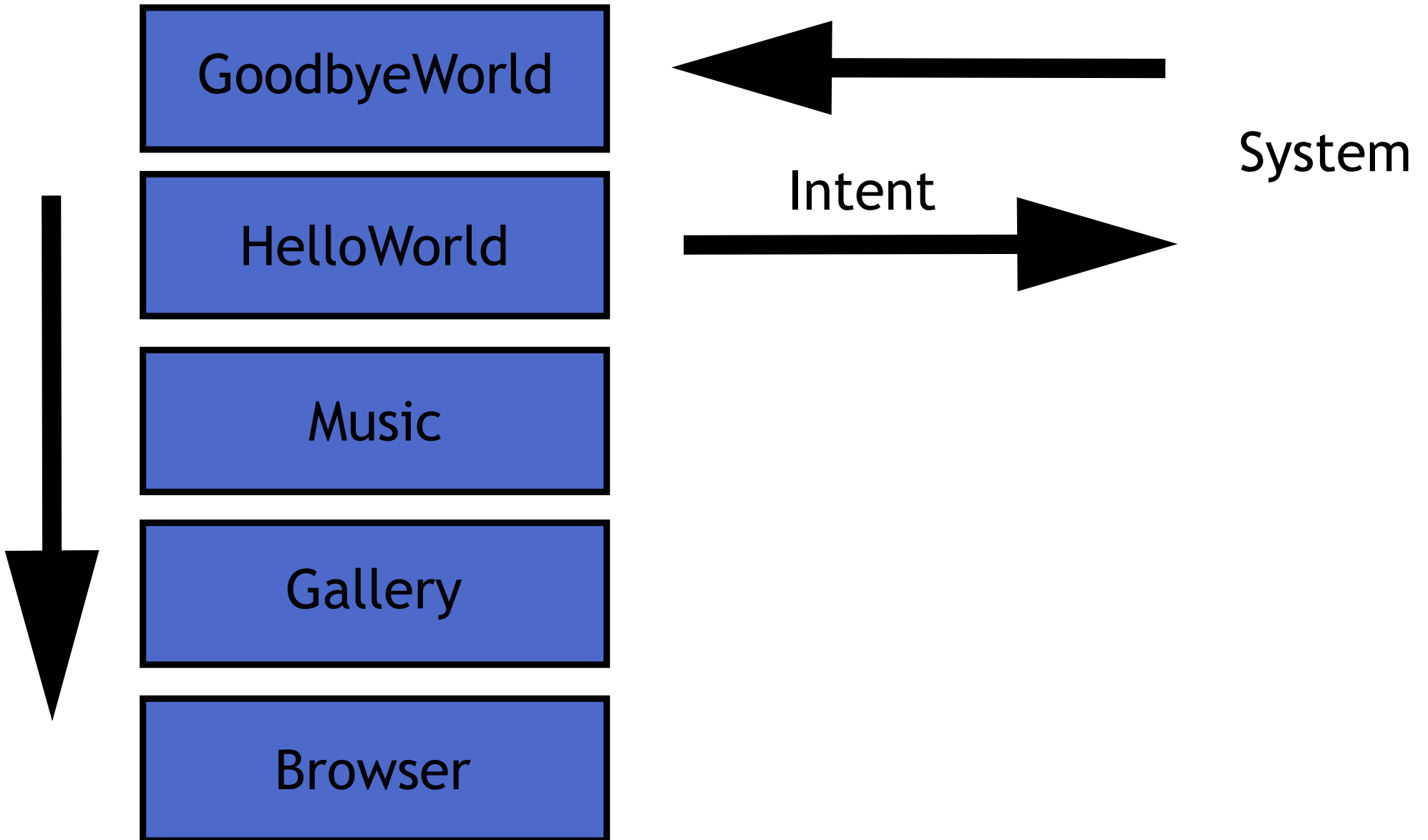
```
package org.rzl.hello;

import /* ... */;

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

        /* Listener einrichten */
    }
}
```

Intents



Intents (Third-Party)

```
String url = "http://raumzeitlabor.de/";  
Intent i = new Intent(Intent.ACTION_VIEW);  
i.setData(Uri.parse(url));  
startActivity(i);
```

Intents (App)

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


Manifest


Metadaten der App

enthaltene Activities

Permissions

Manifest (XML)

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="org.rzl.hello"
    android:versionCode="1"
    android:versionName="1.0">
<application android:icon="@drawable/icon"
    android:label="@string/app_name">
    <activity android:name=".main" android:label="@string/app_name">
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />
            <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
    </activity>
    <activity android:name=".GoodbyeWorld" android:label="Bye!" />
</application>
<uses-sdk android:minSdkVersion="4" />
<uses-permission android:name="android.permission.INTERNET" />
</manifest>
```



3

jetzt klar:
Grundlagen



4

Framework

Adapter

ListView fragt Adapter nach Daten

ArrayAdapter für Listen (ggf. überladen)

vorhandene Views neu befüllen

ArrayAdapter

```
setListAdapter(new MessagesAdapter(db.getMessages()));
```

```
class MessagesAdapter extends ArrayAdapter<String> {
    MessagesAdapter(ArrayList<String> entries) {
        super(Messages.this, R.layout.messages_item, entries);
    }
    @Override
    public View getView(int position, View row, ViewGroup parent) {
        TextView text;
        if (row == null) {
            LayoutInflater inflater = getLayoutInflater();
            row = inflater.inflate(R.layout.message, parent, false);

            text = (TextView)row.findViewById(R.id.message_text);
            row.setTag(R.id.message_text, text);
        } else text = (TextView)row.getTag(R.id.message_text);
        text.setText(getItem(position));
        return row;
    }
}
```

Datenhaltung

Dateien

SharedPreferences

SQLite

alles nur von der App lesbar

SharedPreferences

Datentypen: int, string, boolean, float, long

schreibt in XML-Datei

Bonus: PreferenceActivity

SharedPreferences

```
SharedPreferences settings = getSharedPreferences(PREFS_NAME, 0);  
boolean silent = settings.getBoolean("silentMode", false);
```

```
SharedPreferences.Editor editor = settings.edit();  
editor.putBoolean("silentMode", mSilentMode);
```

```
/* Commit, unbedingt aufrufen! */  
editor.commit();
```

SQLite

Datenbank, SQL-Schnittstelle

bester Weg für viele Daten

embedded; kein volles DBMS

SQLite (öffnen)

```
public class DBHelper extends SQLiteOpenHelper {  
  
    DBHelper(Context context) {  
        super(context, "rzldb", null, 1);  
    }  
  
    @Override  
    public void onCreate(SQLiteDatabase db) {  
        db.execSQL("CREATE TABLE foo (bar INTEGER)");  
    }  
}
```

SQLite (schreiben)

```
SQLiteDatabase db = helper.getWritableDatabase();
SQLiteStatement stmt = db.compileStatement(
    "INSERT INTO TABLE foo (bar) VALUES (?)");
stmt.bindLong(1, 23425);
stmt.execute();
```

SQLite (lesen)

```
int result;
```

```
SQLiteDatabase db = helper.getReadableDatabase();  
Cursor c = db.rawQuery("SELECT * FROM foo", NULL);  
if (c.moveToFirst())  
    result = c.getInt(0);  
if (c != null && !c.isClosed()) {  
    c.close();  
}  
  
/* result benutzen */
```

SimpleCursorAdapter

```
Cursor c = db.getMessages();

startManagingCursor(c);

SimpleCursorAdapter adapter =
    new SimpleCursorAdapter(
        this,
        R.layout.messages_item,
        c,
        new String[] { "Text" },
        new int[] {
            R.id.message_text
        }
    );

setListAdapter(adapter);
```

Netzzugriff

HTTP + JSON

AndroidHttpClient (HttpClient)

in separatem Thread! (→ AsyncTask)

Netzzugriff

```
InputStream uploadValue(String Messwert) throws Exception {
    HttpPost req = new HttpPost("http://mein.server/foo");

    req.setEntity(new ByteArrayEntity(Messwert.getBytes()));
    req.addHeader("Accept-Encoding", "gzip");

    AndroidHttpClient client = AndroidHttpClient.newInstance("");
    HttpResponse response = client.execute(req);
    StatusLine statusLine = response.getStatusLine();
    if (statusLine.getStatusCode() != 200) {
        Log.e("MyApp", "HTTP Error: " + statusLine.toString());
        throw new Exception("HTTP Error");
    }
    HttpEntity e = response.getEntity();
    InputStream s = AndroidHttpClient.getUngzippedContent(e);
    return new BufferedReader(new InputStreamReader(s), 8192).readLine();
}
```


AsyncTask

```
class CreateUserTask extends AsyncTask<String, Void, String> {
    private ProgressDialog dialog;
    @Override
    protected void onPreExecute() {
        dialog = new ProgressDialog(main.this);
        dialog.setTitle("Downloading the internet");
        dialog.show();
    }
    @Override
    protected String doInBackground(String... params) {
        uploadValue("fnord");
        return null;
    }
    @Override
    protected void onPostExecute(String errorMessage) {
        dialog.dismiss();
    }
}
```

Geolocation

GPS, Triangulation, WLAN

Manifest: ACCESS_COARSE_LOCATION (Netz)
oder ACCESS_FINE_LOCATION (GPS)

Koordinaten selbst beurteilen

möglichst kurz Location anfordern

GeoLocation verfügbar?

```
String provider =
    Settings.Secure.getString(
        getResolver(),
        Settings.Secure.LOCATION_PROVIDERS_ALLOWED);

if (provider != null && !provider.equals("")) {
    Log.v("MyApp", "Location providers: " + provider);
} else {
    Log.e("MyApp", "No location providers");
    startActivity(new Intent(Settings.ACTION_LOCATION_SOURCE_SETTINGS));
}
```

GeoLocation abonnieren

```
lm = (LocationManager) getSystemService(Context.LOCATION_SERVICE);  
  
lm.requestLocationUpdates(LocationManager.GPS_PROVIDER, 5 * 1000,  
    0, newLoc);  
lm.requestLocationUpdates(LocationManager.NETWORK_PROVIDER, 5 * 1000,  
    0, newLoc);  
  
private LocationListener newLoc = new LocationListener() {  
    public void onLocationChanged(Location location) {  
        Log.d("MyApp", "newLocation: lat " + location.getLatitude() +  
            ", lon " + location.getLongitude());  
    }  
};
```

Dev Guide → Location and Maps → Obtaining User Location

Fertig! Danke!

<http://d.android.com/>

<http://www.stackoverflow.com/> → [android] foo

google → android foo