



## Android Geek Night 3.0

Per Nymann Jørgensen

[pnj@trifork.com](mailto:pnj@trifork.com)

Niels Sthen Hansen

[nsh@trifork.com](mailto:nsh@trifork.com)

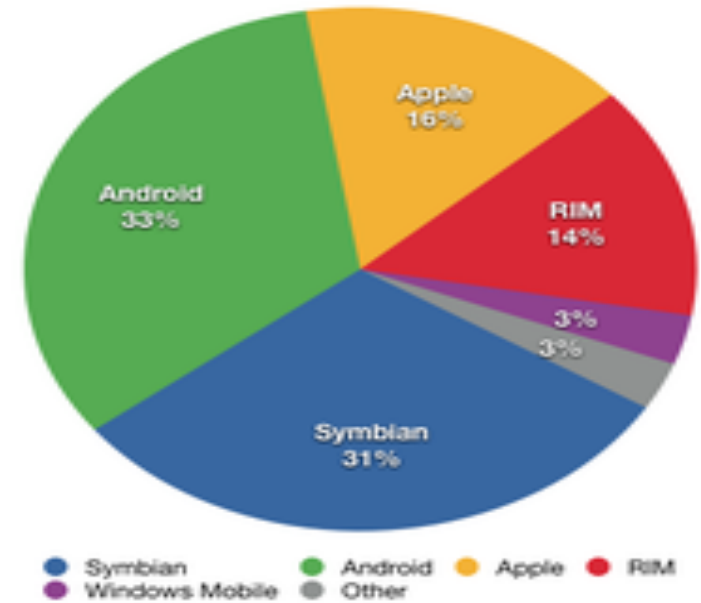
# Android Geek Night 3.0



- Android at a glance
- New features in Gingerbread & Honeycomb
- Demos & Code

# Android 101

- Operating system targeting mobile devices/Tables devices
- Linux based - with additions
- Open source under the Apache License
- Allows development in Java
  - Or Scala, JRuby, Groovy ..
- Two new versions just came out..



Share of worldwide 2010 Q4 smartphone sales to end users by operating system, according to [Canalys](#).<sup>[35]</sup>

# Android 101 - Dalvik VM



- Virtual machine developed by Google for mobile devices
- Uses the Dalvik Executable (.dex) format
- 
- Designed for limited processing power and memory
- 
- Register-based architecture
  - as opposed to stack machine Java VMs
- Class library based on Apache Harmony
  - No AWT, Swing
  - No Java ME

# Android 101 - SDK



- Android libraries
- The SDK and AVD manager, for maintaining the SDK components and creating virtual devices
- LogCat to capture logs from running device
- DDMS – Dalvik Debug Monitor
- Tools to convert Java .class files to Dalvik bytecode and create installable .apk files
- Plugin for Eclipse - Android Development Tools (ADT)

# Android 101 - Components



- Activity
  - GUI
  -
- Service
  - non-GUI
- Broadcast Receiver
  - Events
- Content Provider
  - Exposing data/content across applications

An Android application can be seen as a collection of components.

# Android API 10 New stuff



## ● New Sensors / New Sensor APIs

- Gyroscope
  - □ Rotation vector
- Acceleration
  - Linear acceleration (acceleration without gravity)
  - Gravity (gravity without acceleration)
- Barometer (air pressure)

# Android API 10 New stuff

## ● NFC

- Short range wireless communication.
- Do not require discovery or pairing
- Supported mode as of 2.3.3 (reader/writer/P2P limited)
- Enable application like Mobile ticketing (dare we say rejsekort), Smart poster, etc.
- Possible to handle TAG (API 10)
  - Foreground activity
  - ACTION\_NDEF\_DISCOVERED
  - ACTION\_TECH\_DISCOVERED
  - ACTION\_TAG\_DISCOVERED





# Android API 10 New stuff



- SIP Based VoIP
  - Enable application like video conferencing, streaming multimedia distribution, instant messaging, presence information, file transfer and online gaming.
- Multiple camera support (Facetime™ the next big thing ;-)
  - Possible to query the device for camera info;
    - Frontfacing/backfacing
    - Number of cameras

# Android API 10 New stuff



- Download manager
  - Take care of long running HTTP downloads
    - Retries after failure
    - Changing in connectivity
    - Reboot of the device
    - Nice addition to the framework
- UI
  - Overscroll
  - Support for touch filtering
  - Improved event management
  - Improved motion events
  - Text selection controls

# Android API 10 New stuff



- Xlarge screens
  - Actually size 7 - 10 inches
- AlarmClock Provider
  - Set an alarm using an intent
- Storage OBB (Opaque Binary Blob)
  - Used for storing big binary assets (Ex. Texture for games)
  - We will see the usage for this in the future.
- NativeActivity
  - Possible to make an Activity using C code:-)
  - Using NDK
    - `void android_main(struct android_app* state) { ...`
  - Do we say games again... all that nice C code :-)
- Location and Worksource (More battery hopefully!)

# Android API 11 (Honeycomb)



- The API for tablets.
- Will this release come for the phones ?
  - Some sort of merging will happen!
  - 2.4 maybe or 4.0 might be the next release!
  - Honeycomb have API 10 features as well.
  - Fragments in earlier api Android Compatibility Pack
- Not a complete walk trough of all the great stuff :-)



# Android API 11 new stuff

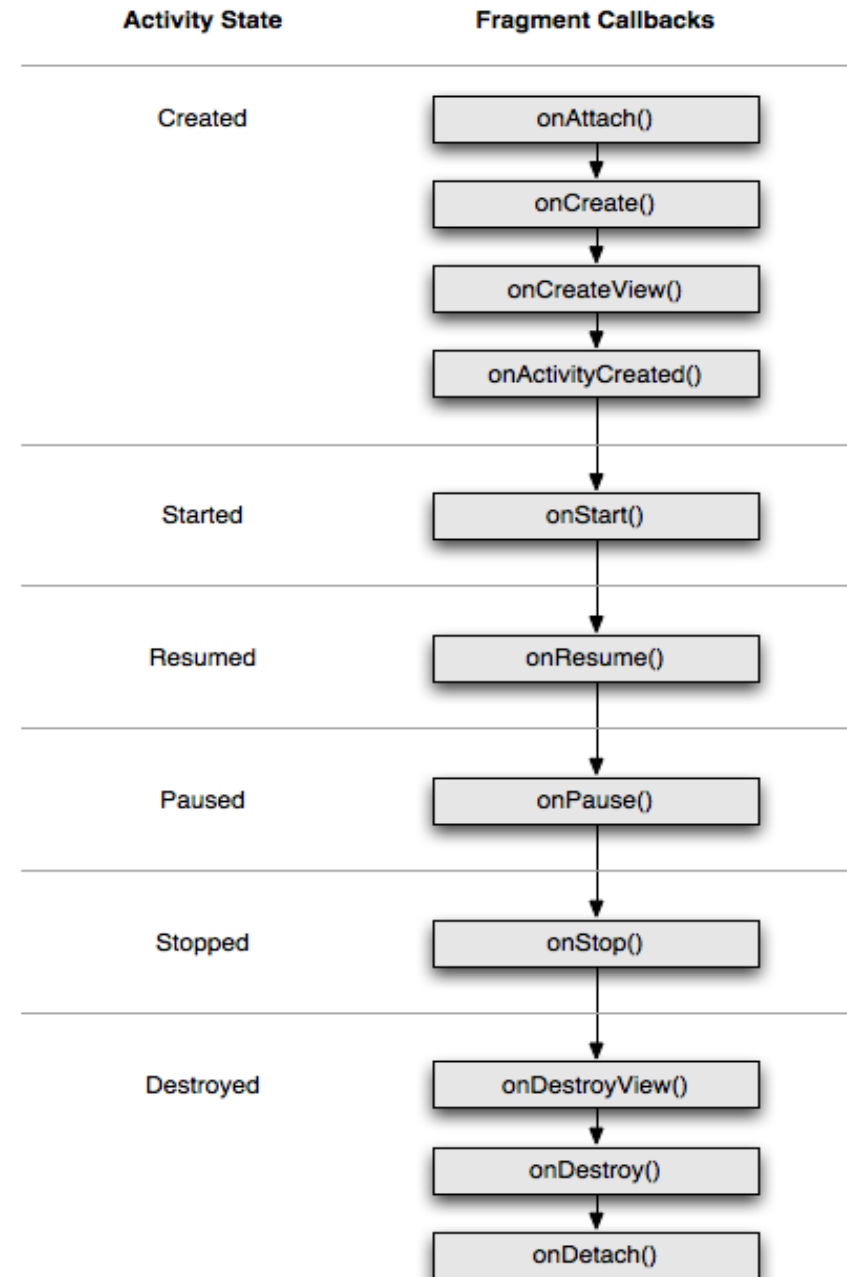


- Fragments
  - Self contained component with own lifecycle (and UI).
  - We don't declare in manifest.
  - Can be manipulated inside an activity.
  - Used when you want alternative formats, depending on screen size, orientation etc.
  - In the fragment you have access to the parent activity.
  - We don't have to build activities, but can use fragments as reusable parts in our activities.
  - A lot of Activity maybe replaced with Fragments.
  - Can be declared in xml layout as well.

```
<fragment class="com.trifork.TitlesFragment"  
android:id="@+id/titles" android:layout_weight="1"  
android:layout_width="0px" android:layout_height="match_parent" /  
>
```

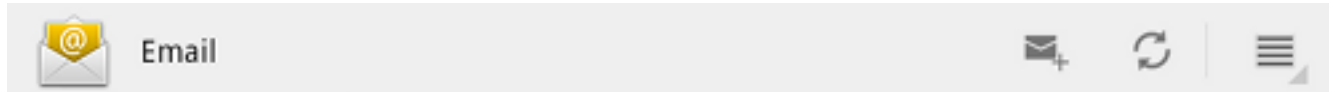
# Android API 11 new stuff

- Fragment lifecycle and its connection to activity
- You need a 'Anchor' in the activity.
- Can be put into a backstack 'living' in the activity.
- Breadcrumb support.

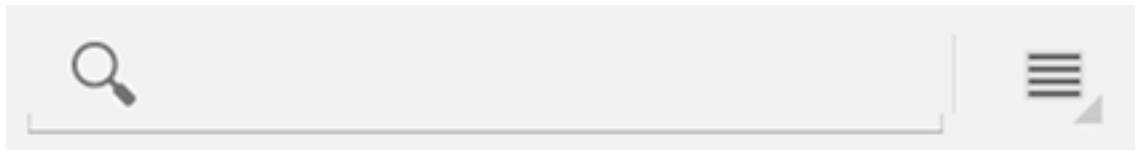


# Android API 11 new stuff

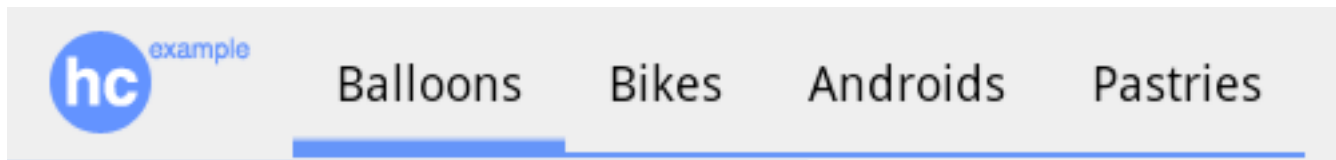
- New Statusbar (top of the application)



- Direct access to menu items.
- New items is added as normal android
- Menu items will overflow



- Interactive action views



- Home behavior/Navigation behavior
- Tabs to the statusbar (Using fragments)

# Android API 11 new stuff



- Renderscript
  - High performance 3D rendering/mat. computation
  - Written in C (C99)
  - No need to target specific device
    - Renderscript is converted to intermediate bytecode and compiled into machinecode JIT on the device
  - Running as Native code on the device
    - Can be running on the GPU/CPU.
    - Using a renderscript runtime (Which choice PU)
    - NDK can not be used!
    - Standard C lib. is not guaranteed!.



# Android API 11 new stuff



- New Animation API
  - No longer view centered, but property centered
- System Clipboard
  - Not just text!
- Loaders
  - Loading asynchronous data
  - Monitor change
    - Possible to avoid reloading to often
- Drag and drop
  - We had that before, but has been simplified!.
- New Widgets
  - CalendarView, NumberPicker, PopupMenu, StackView and more.

# Code samples



- Scala
- Proguard
- Roboguice
- Test Projects
- Animation 2D and 3D
- Fragments
- C2DM
- EWP Synchronizing / Account Manager

# Scala on Android



From time to time you hear people say that Java will be the new COBOL

- **C** ompletely
- **O** bsolete
- **B** usiness-
- **O** riented
- **L** anguage

# Scala on Android



In fact, some claim that Java *already is* a new COBOL due to outdated syntax and limited programming paradigms

Unfair or not, this criticism has fueled the proliferation of a number of new, JVM-based languages

# Scala on Android



Scala is ..

- .. a multi-paradigm programming language
  - Object-oriented
  - Functional
- .. fully compatible with Java
  - Compiles to Java bytecode, so Scala can use Java classes and vice versa
- .. statically typed
  - Allows for static analysis
  - Rich & concise type system with light-weight syntax
  - Like dynamic languages, existing classes can be extended
- .. shipped with some nice APIs
  - Collections, concurrency etc.

How would that look like on Android ?

# Scala on Android



```
public class MyActivity extends Activity {  
  
    public void onCreate(Bundle savedInstanceState) {  
        // call super, set content view  
  
        TextView txtView = (TextView)findViewById(R.id.myTxt);  
        txtView.setText("Other text");  
        txtView.setOnClickListener(new OnClickListener() {  
            public void onClick() { /* handle click event */ }  
        });  
        doSomeFancyStuff(txtView);  
  
        // do other stuff  
        ...  
    }  
}
```

Lots of boilerplate stuff

Not nice having to cast

# Scala on Android



```
class MyActivity extends Activity with Trait1, .. {  
  
    override def onCreate(savedInstanceState:Bundle ) =  
    {  
        // call super, set content view  
  
        val txtView = findViewById[TextView](R.id.myTxt);  
        txtView.setText("Other text");  
        txtView.setOnClickListener( (x) => /* handle click */ );  
        txtView.doSomeFancyStuff();  
  
        // do other stuff  
        ...  
    }  
}
```

Extend class with traits, -e.  
g. typed findViewById()  
Functions as parameters  
Add methods to any class..

# Scala on Android



.. and some reasons why you may **NOT** want to use Scala on Android

- Larger .apk files
  - Classes must be shrunk before .dex'ing
- Longer build-deploy round-trip
- Scala Traits are modelled with Java abstract classes and interfaces
  - Goes against Android guidelines



# More on Proguard



- Shrinks
- Optimizes
- Obfuscates

Creates a smaller .apk file which is (somewhat) harder to reverse engineer.

Standard part of the SDK since revision 8

# Roboguice



- Light-weight dependence injection framework for Android
- Based on Google Guice
- A way to beautify your Android apps while still using Java

# A Few Words on Testing



- You can use regular junit tests on non-android specific parts of your code
- Instantiating your own components in a junit test will as a general rule **not** work
- Android offers an Instrumentation test framework based on junit
  - On virtual devices
  - On real devices
  - The application to be tested is deployed to the device - essentially integration testing

# A Few Words on Testing



- Use the Monkey to exercise your GUI!

```
adb shell monkey -p com.trifork 100
```

- Will generate 100 pseudo-random UI events
- Also tools for test scripting, screen capture etc.

# Graphics Overview



- 2D Graphics
- Animations
  - Views
  - New Honeycomb animation API
- OpenGL

# 2D Graphics



- Android comes with a collection of Drawables
  - ShapeDrawable
  - BitmapDrawable
  - etc.
  
- Grab the View's Canvas in the onDraw() method

# Animations



- android.view.Animation
- Provides a standard library of animations
  - Translate (move!)
  - Rotate
  - etc.
- Use directly in code or specify in XML
- Works specifically on *Views*
- Change the *appearance* - not the actual properties

# Animations in Honeycomb



- Property-centric - not *View*-centric. You can animate **anything**
  - Properties on Drawables
  - In fact, properties on any object even if it is not UI-related
- The *View*\_class now has new properties to support this
  - `setRotationX()`, `setRotationY()`, `setTranslationX()`, `setAlpha()`, etc.



# Animations in Honeycomb



```
// Fade myShape to complete transparency
ObjectAnimator anim =
    ObjectAnimator.ofInt(myShape, "alpha", 255, 0);
anim.setDuration(4000);
anim.start();
```

# 3D - OpenGL



- OpenGL ES 1.1 since 1.6, OpenGL ES 2.0 since 2.2
- GLSurfaceView
- The views renderer gets access to a GLXX object
- Very easy to use - if you know your OpenGL....

# Fragment sample



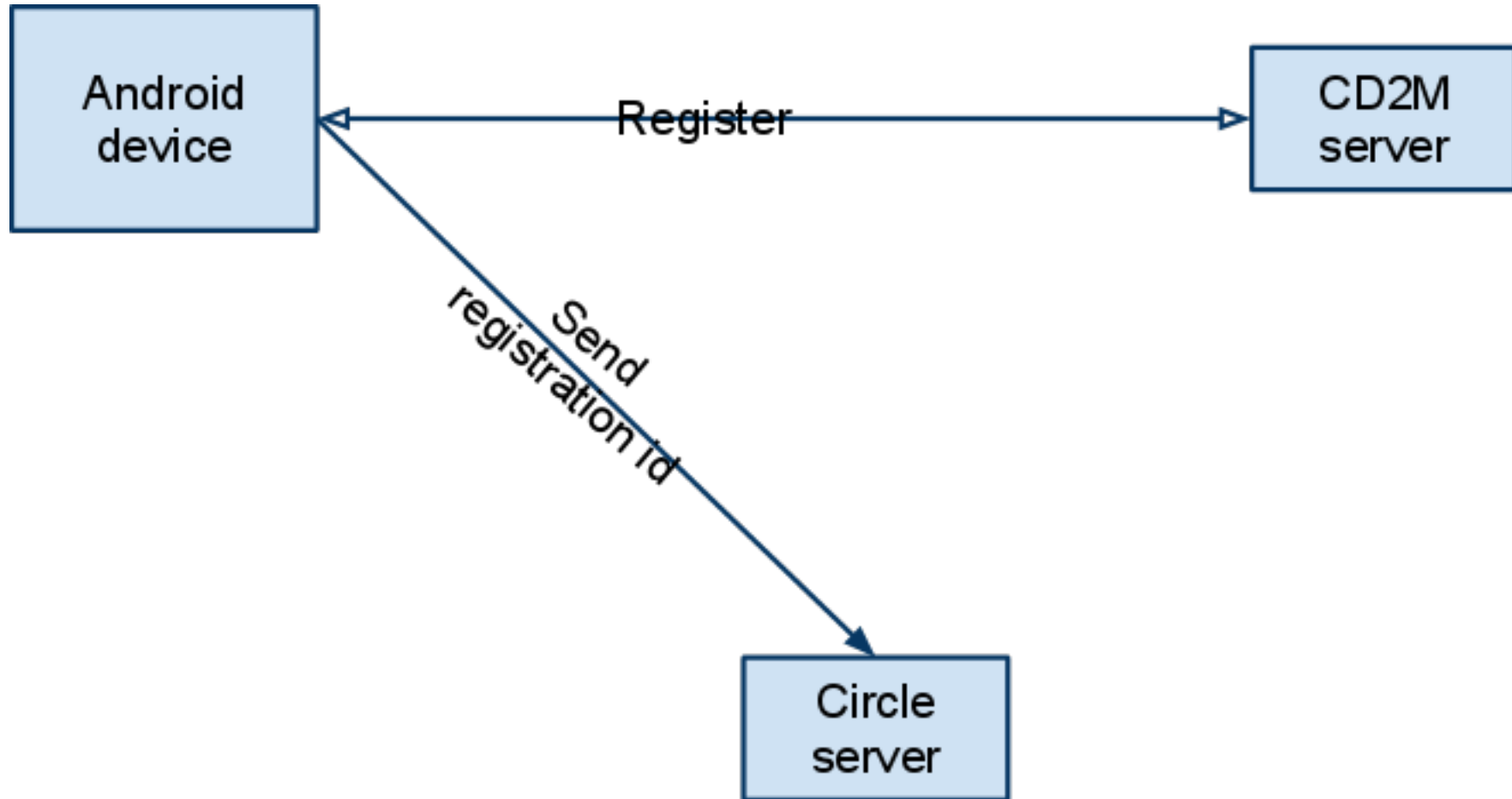
- Simple example showing the usage of fragments
- One presentation when in landscape another one when in portrait, using the same fragments.

# C2DM

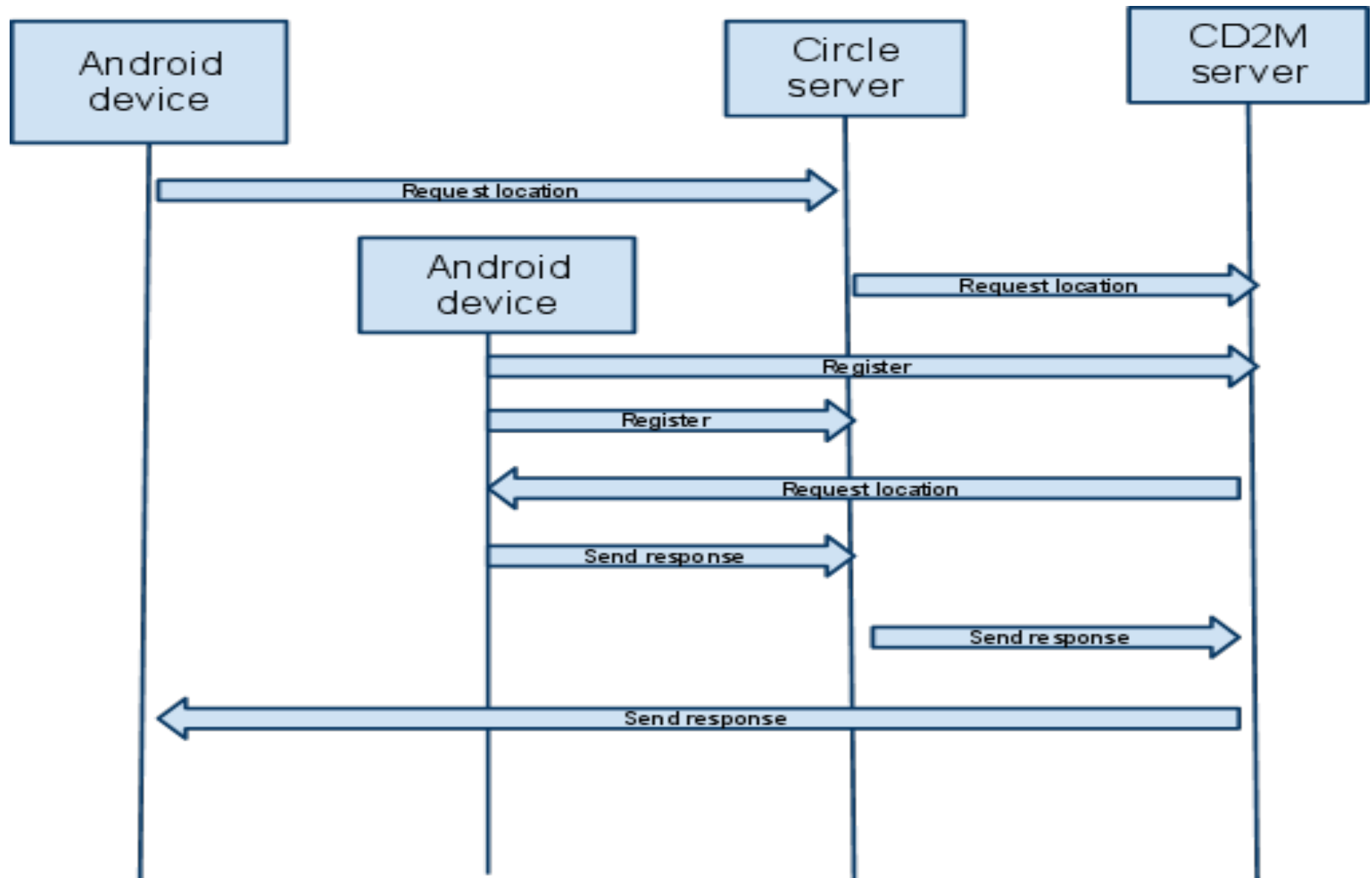
- Sending message from servers to android devices
- C2DM service is hosted at google
  - C2DM handling all aspect of queuing and delivery to the target application
- Still google Labs, so u need to sign up.
- Limited to short message from the you server to the device (1024 bytes)
- There is a limit to number of message that can be aggregated for a device.
- Know examples is Google Chrome to Phone.

# C2DM (example)

## Registration process



# C2DM (example)



# EWP Synchronizer



- What is EWP?
- We want to have some of the data from EWP available on the devices (this example is contact information for the employees)
- So the application on the devices should store credentials to access an account in EWP, and then synchronize these to the device.
- This is similar to what happens when we are synchronizing with our contacts on the google account.

# EWP Synchronizer



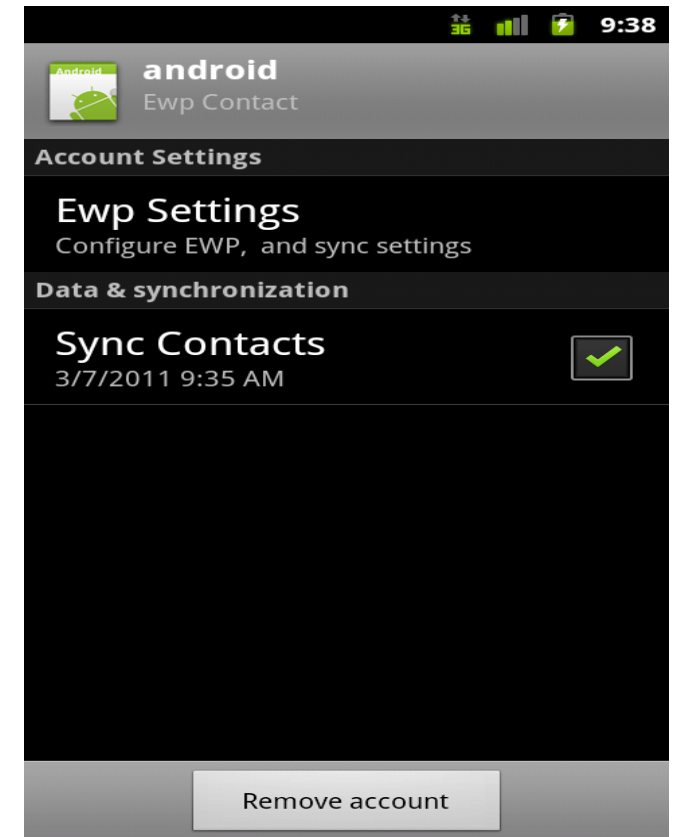
- Synchronizer API in Android, Old stuff :-) API level 5.
- Account Manager
- Sync manager



# EWP Synchronizer



- Sync API in Android, Old stuff :-) API level 5.
- Account API
  - Add new account. Intent filter **android.accounts**. **AccountAuthenticator**.
  - Account manager will callback on the authenticator service.
  - We will be able to add and manage our ewp account.



# EWP Synchronizer



- Sync is also a service called by the android base system.
- Defined in Androidmanifest.xml with intent [android.content.SyncAdapter](#)
- Reference to metadata

## syncadapter

```
<syncadapter xmlns:android="http://schemas.android.com/apk/res/android"  
    android:contentAuthority="com.android.contacts"  
    android:accountType="com.trifork.ewp.android.account"  
    android:supportsUploading="false"  

```

/>

- Connection between Content Provider and account type

# EWP Synchronizer



- The sync manager will bind to our service and perform sync.

```
public class SyncAdapter extends AbstractThreadedSyncAdapter {  
    ....  
  
    @Override  
    public void onPerformSync(Account account, Bundle extras, String authority, ContentProviderClient  
    provider, SyncResult syncResult) {  
        ...  
    }  
}
```

- Get the data from the server and store this locally.
- Could support 2 way sync. (ours don't)



- Please fill out the evaluation form



- GOTO CPH - <http://gotocon.com/cph-2011/>
- Android Tutorial 15. april in Århus.
- Android Tutorial 9. May on GOTO CPH.