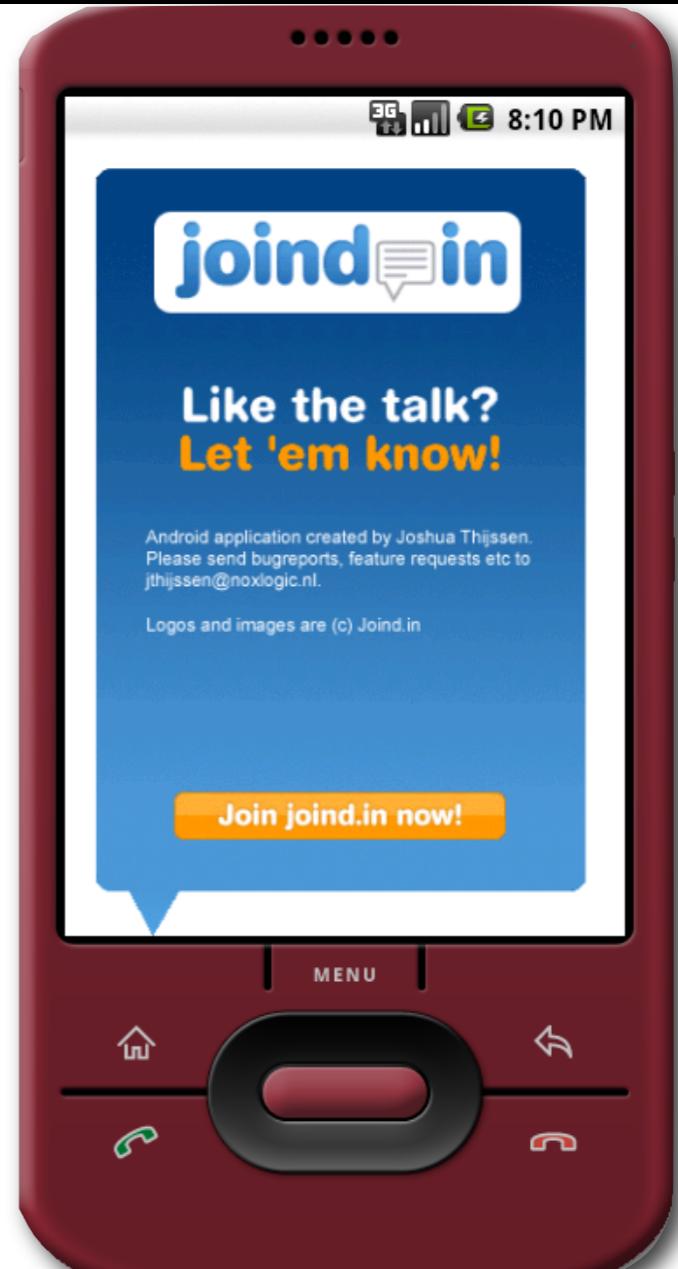


The Joind.in Android App

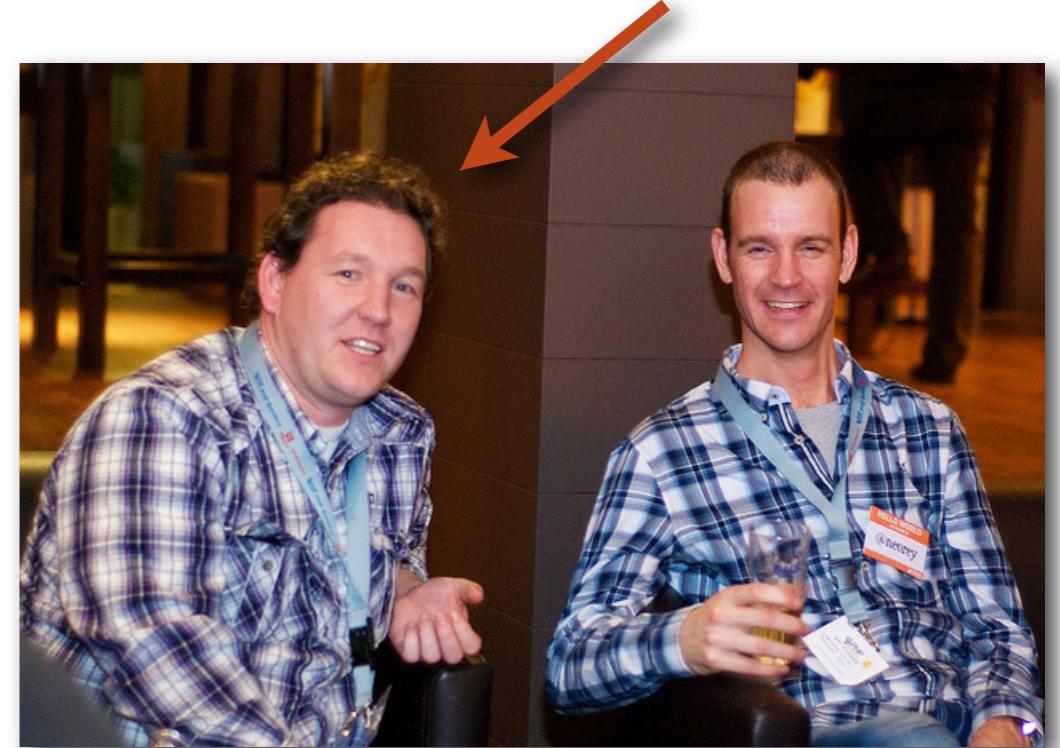
A story about Android programming, API compatibility and the lessons learned...



#4developers - 4 april 2011
Warsaw - Poland

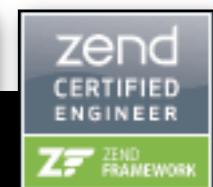
ENRISE

Who am I?



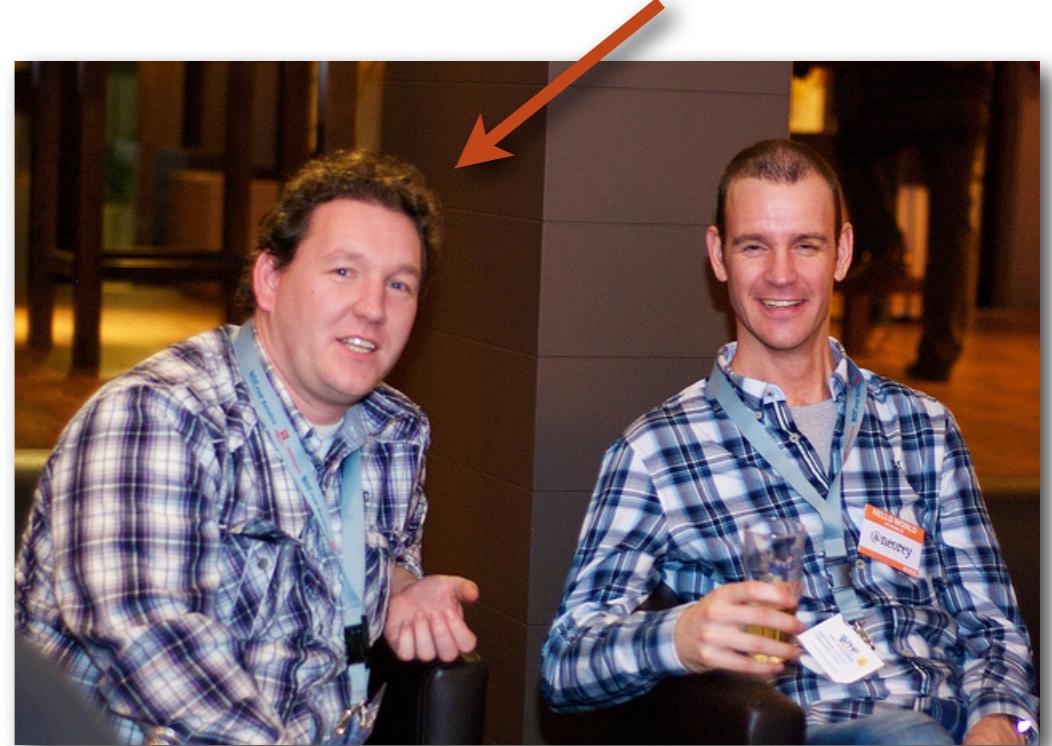
Who am I?

Joshua Thijssen (32)



Who am I?

Joshua Thijssen (32)
Senior Software Engineer @ Enrise

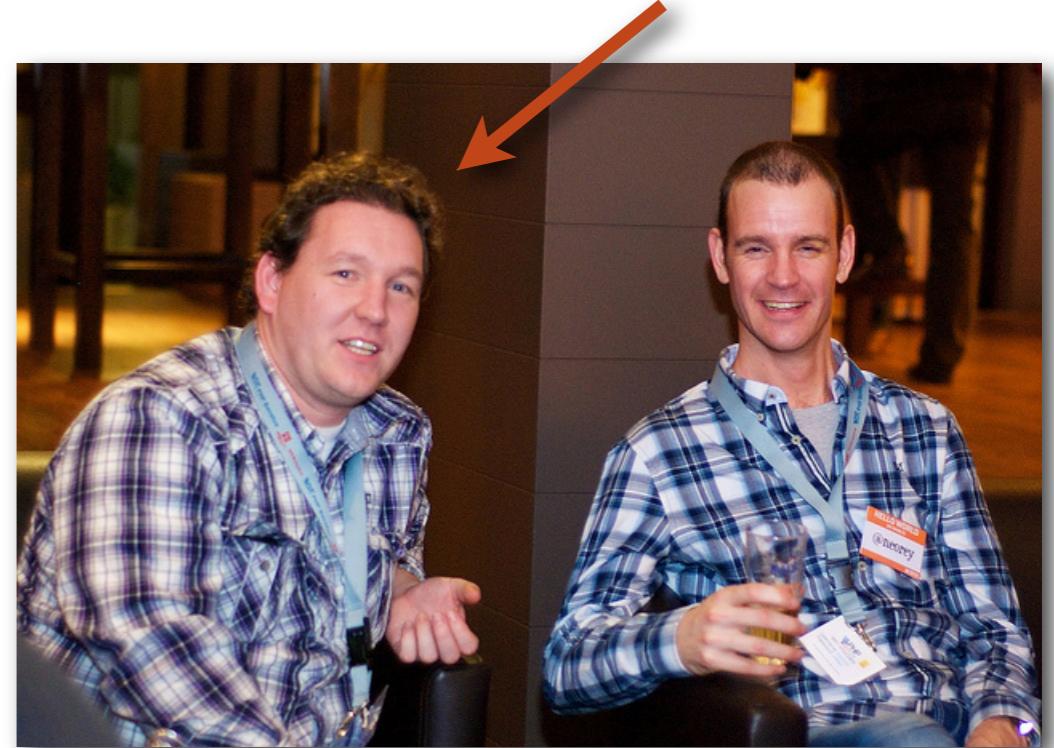


Who am I?

Joshua Thijssen (32)

Senior Software Engineer @ Enrise

Development in PHP, Python, Perl,
C, Java....



Who am I?

Joshua Thijssen (32)

Senior Software Engineer @ Enrise

Development in PHP, Python, Perl,
C, Java....

Blogs: <http://www.adayinthelifeof.nl>

<http://www.enrise.com/blog>



Who am I?

Joshua Thijssen (32)

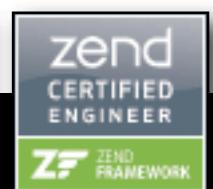
Senior Software Engineer @ Enrise

Development in PHP, Python, Perl,
C, Java....

Blogs: <http://www.adayinthelifeof.nl>

<http://www.enrise.com/blog>

Email: joshua@enrise.com



Who am I?

Joshua Thijssen (32)

Senior Software Engineer @ Enrise

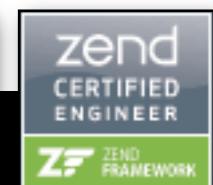
Development in PHP, Python, Perl,
C, Java....

Blogs: <http://www.adayinthelifeof.nl>
<http://www.enrise.com/blog>

Email: joshua@enrise.com

Twitter: [@jaytaph](https://twitter.com/jaytaph)

Identica: [jaytaph](https://identi.ca/jaytaph)



5 What is this talk about?

- ▶ QUESTIONS? RAISE YOUR HAND OR YELL LOUD

What is this talk about?

- The story about the joind.in website.

► QUESTIONS? RAISE YOUR HAND OR YELL LOUD



What is this talk about?

- The story about the joind.in website.
 - The story about the joind.in Android app.
-
- ▶ QUESTIONS? RAISE YOUR HAND OR YELL LOUD



What is this talk about?

- The story about the joind.in website.
 - The story about the joind.in Android app.
 - The story about mobile API connectivity.
-
- ▶ QUESTIONS? RAISE YOUR HAND OR YELL LOUD



What is this talk about?

- The story about the joind.in website.
 - The story about the joind.in Android app.
 - The story about mobile API connectivity.
 - The story about android programming.
-
- QUESTIONS? RAISE YOUR HAND OR YELL LOUD

The story about joind.in

Like the talk? Let 'em know!

Joind.in provides the missing link between the people attending a conference and the ones that presented.

Attendees can post their comments directly to each of the talks they attended, giving the speaker direct feedback on how they did and what they can do to improve.

Joind.in also has something to offer the speakers - you can track your record across the conferences and see how changes in your talk might have made a difference in your ratings.

[Join joind.in now!](#)





The story about joind.in (1)

- Launched by Chris Cornutt (phpdeveloper.org) in 2008
- Adopted by Lorna Mitchell and others.
- <https://github.com/joindin/joind.in>
- Large in the PHP community and gaining ground in others as well.



The story about joind.in (2)

- A website for conference organizers and attendees.
- Find information and give feedback on conference talks (like this one).
- Helps speakers to improve their talks.
- Over 10K registered users and over 30K page views per month.



The story about joind.in (3)

- What events are currently happening?
- Which talks are hot (or not)?
- Should I see talk X or Y?
- feedback = improvement
- your future conference planner

The story about the joind.in android app

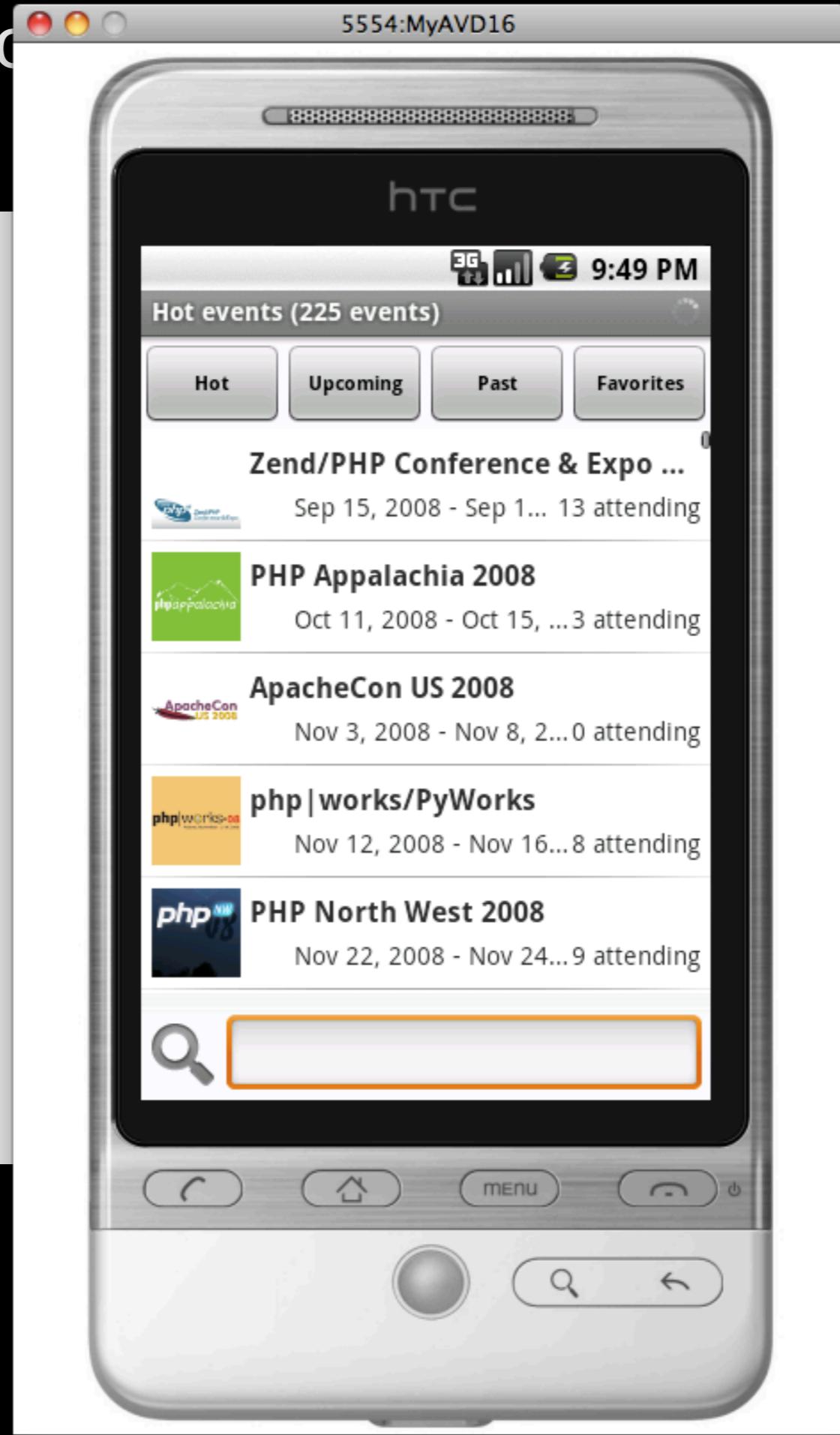


The story about the joind.in android app (1)



The story about app (1)

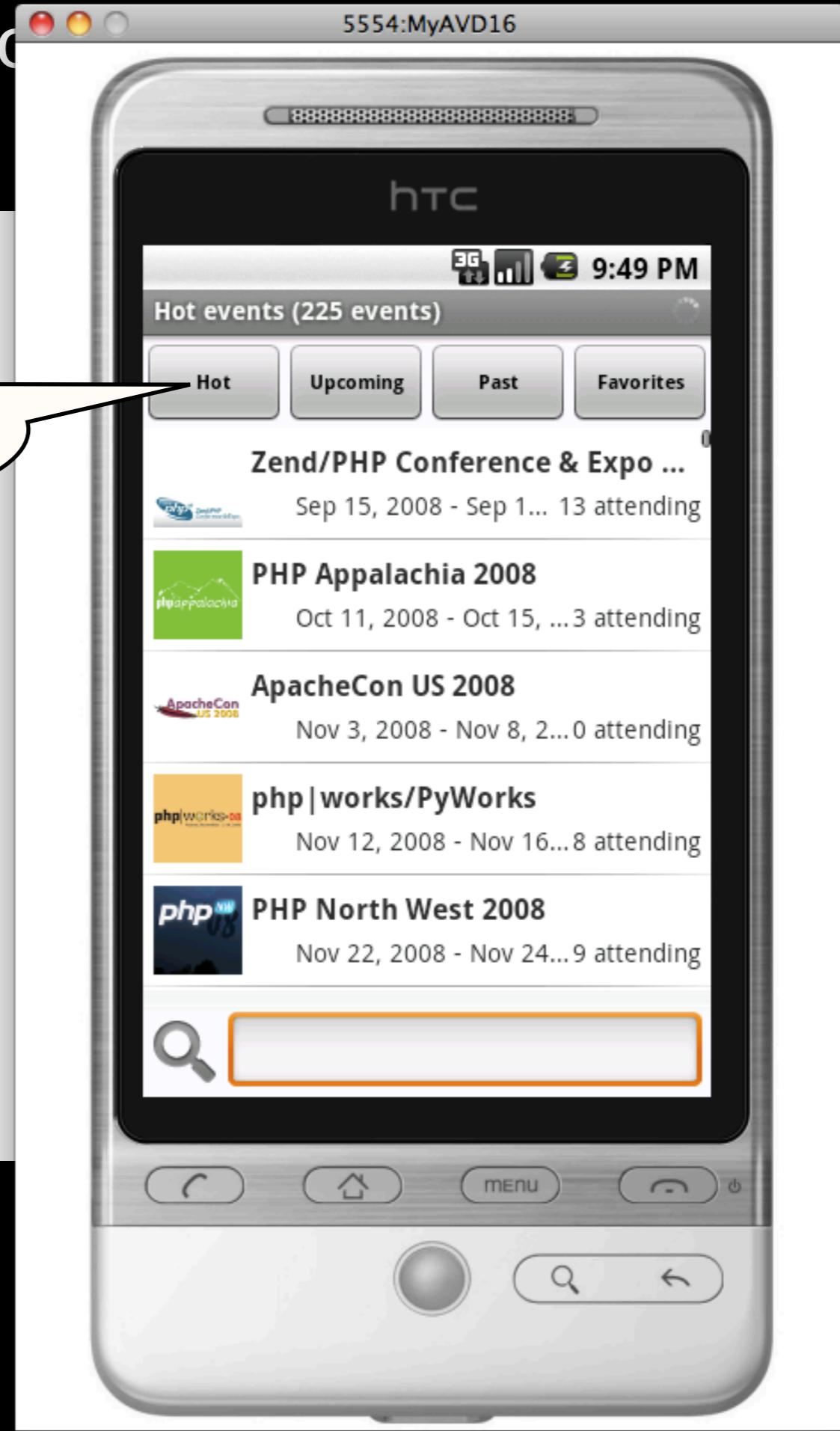
app (1)





The story about app (1)

Simple buttons

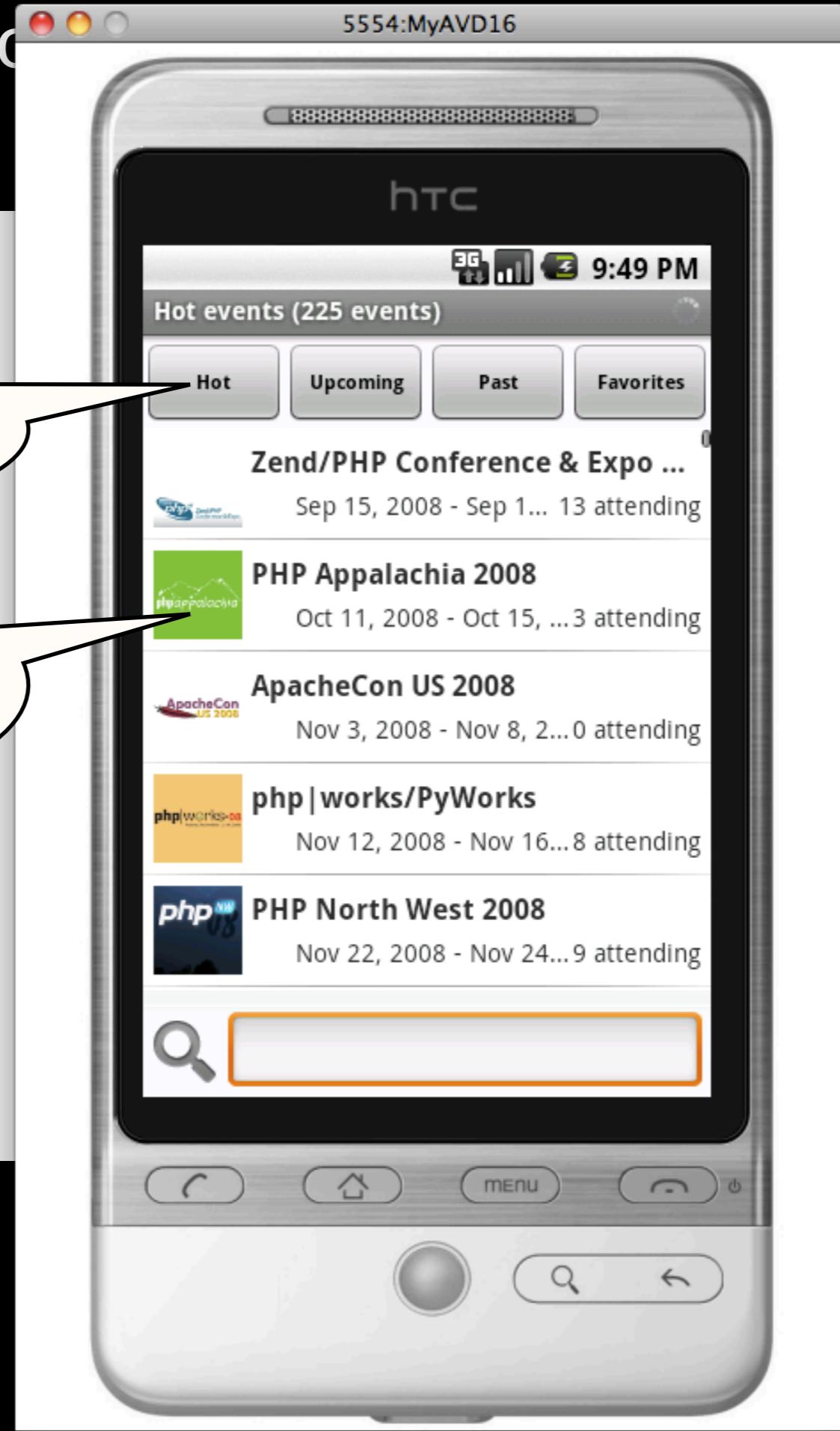




The story about app (1)

Simple buttons

Loaded in
separate thread





The story about

5554:MyAVD16

app (1)





The story about my app (1)



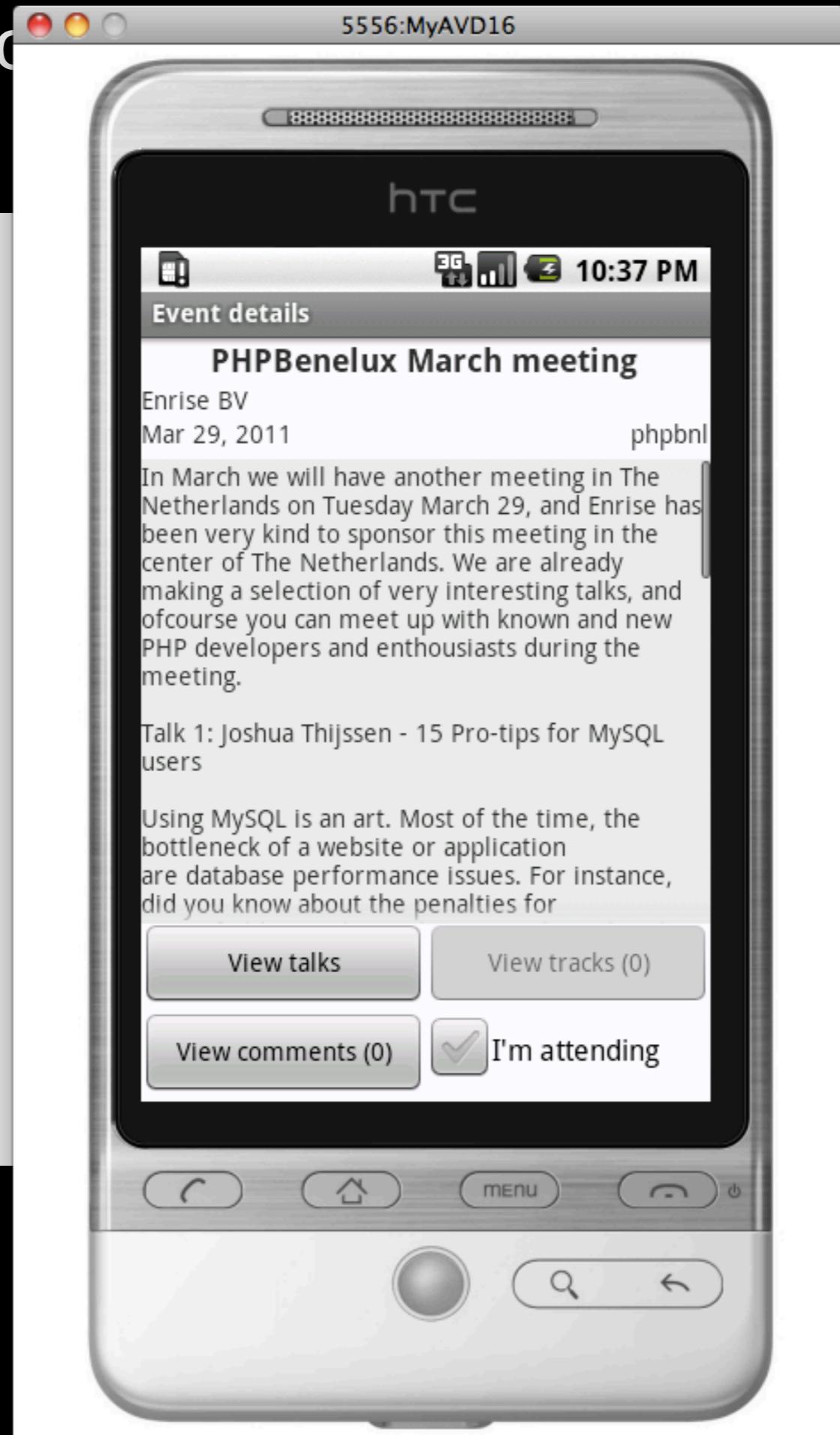
The story about the joind.in android app (1)

The story about the joind.in android app (2)



The story about my app (2)

app (2)



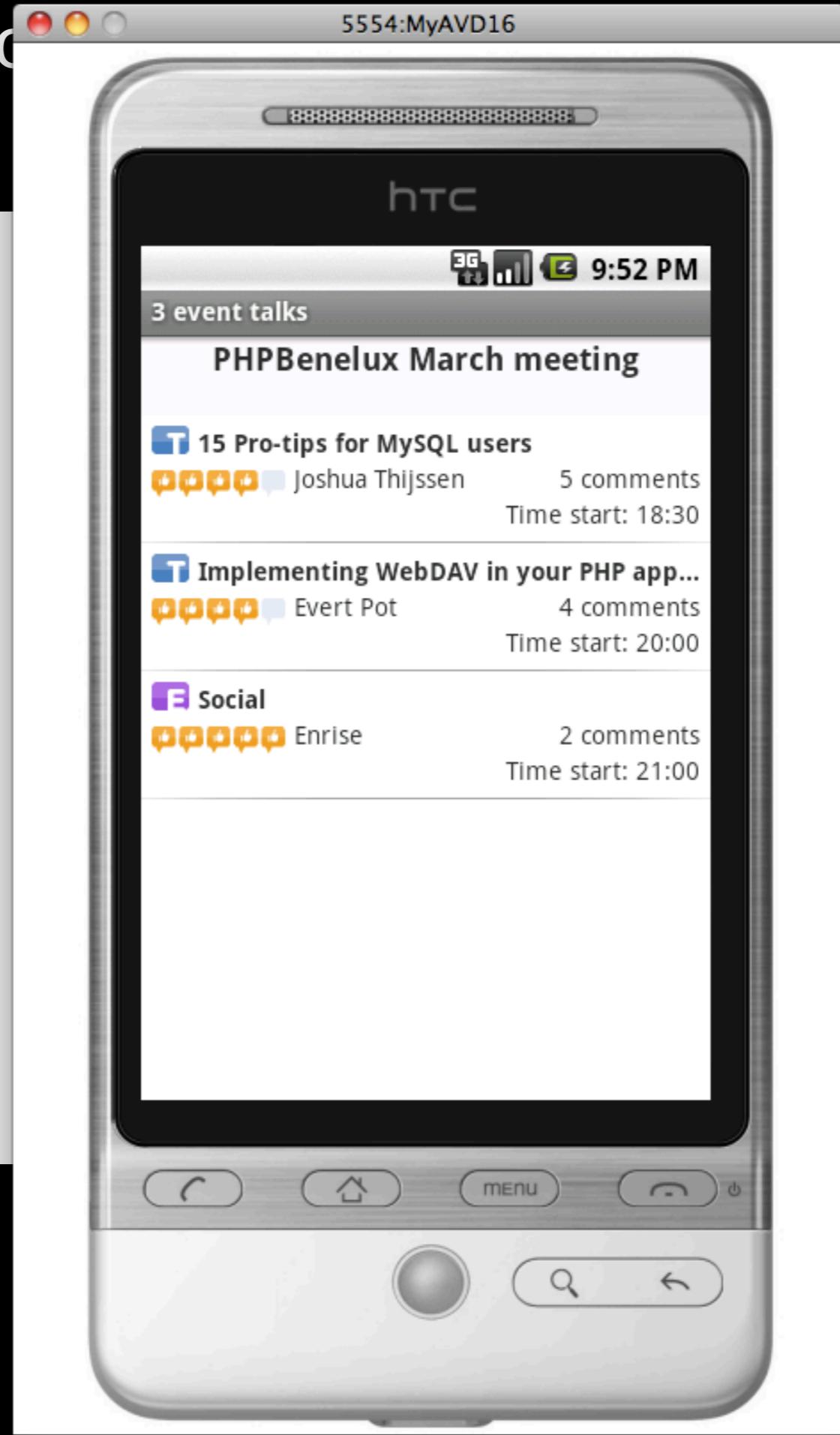
The story about the joind.in android app (2)

The story about the joind.in android app (3)



The story about the app (3)

app (3)



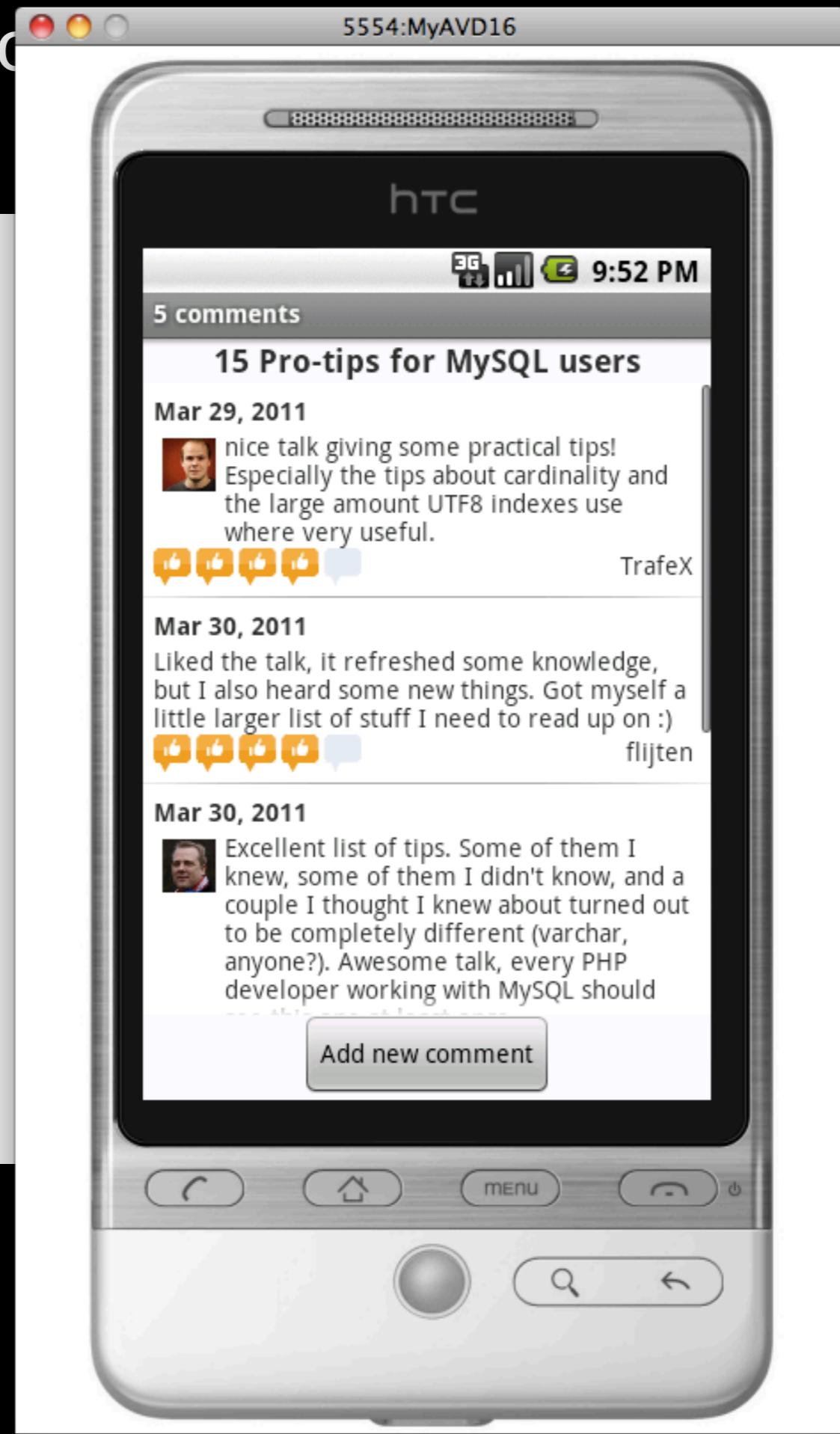
The story about the joind.in android app (3)

The story about the joind.in android app (4)



The story about the app (4)

app (4)

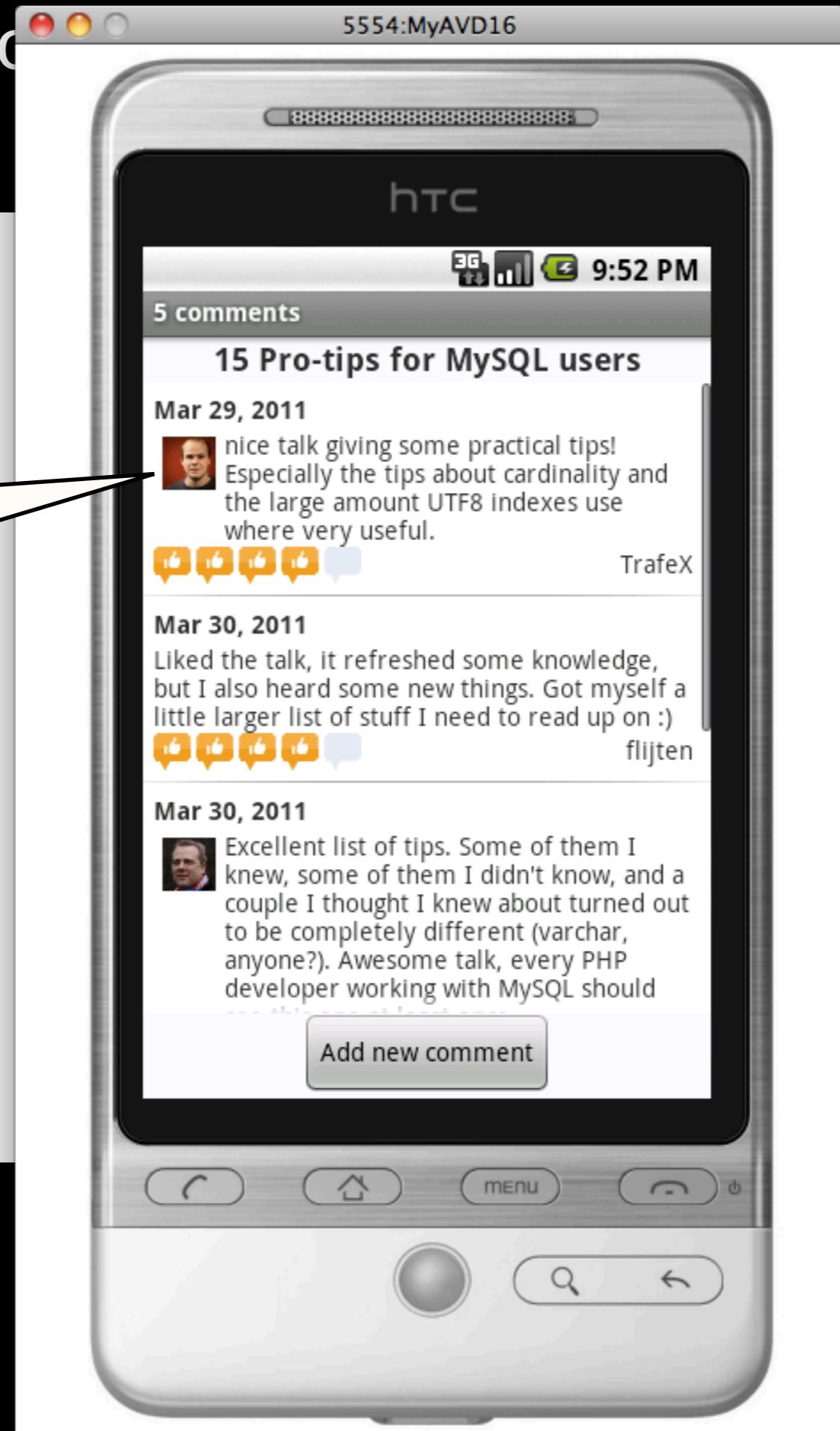




The story about the app (4)

app (4)

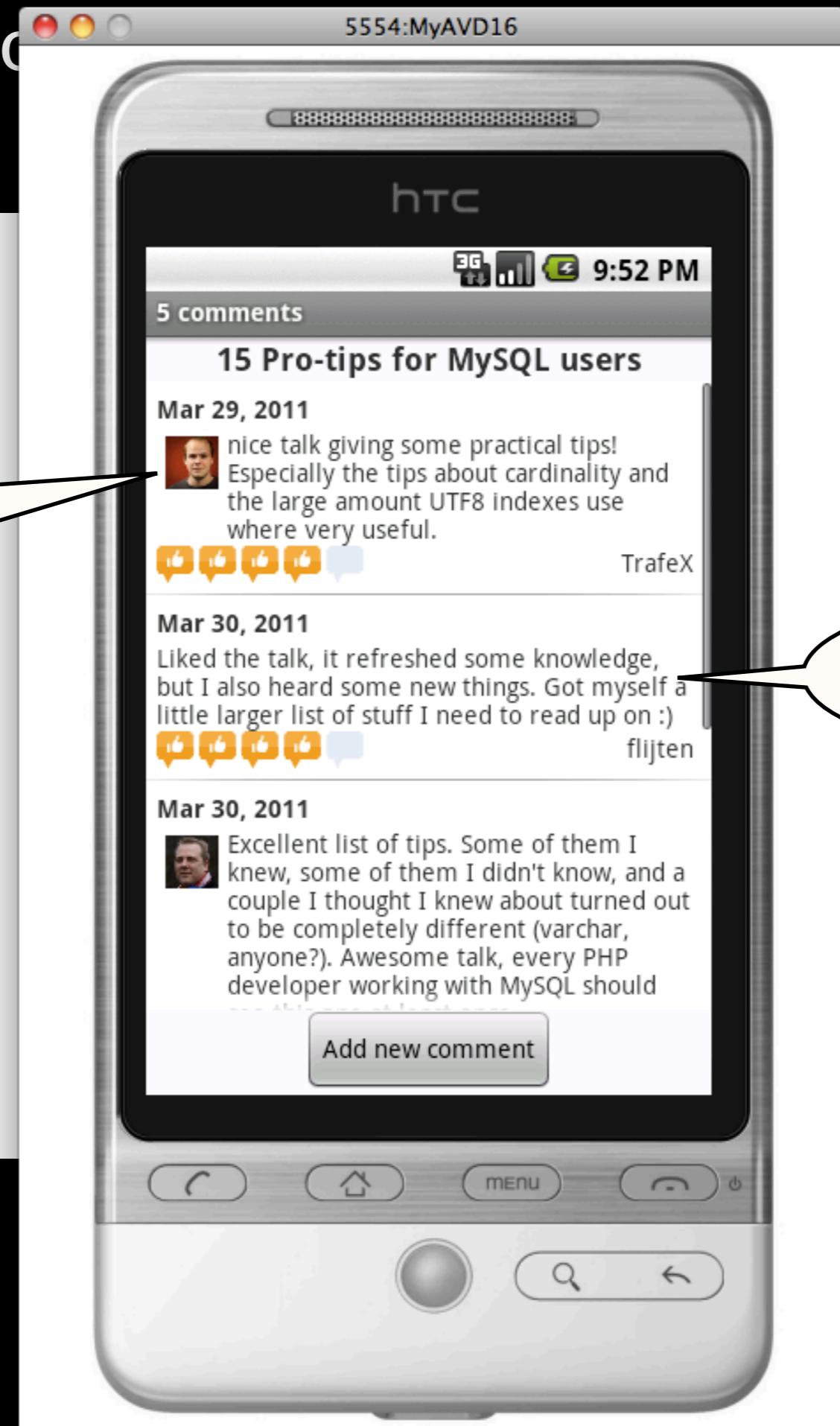
Loaded from
gravatar /
separate thread





The story about the app (4)

app (4)



The story about the joind.in android app (4)



The story about the joind.in android app (5)

- First (serious) android project
- Open source
- <http://github.com/jaytaph>
- Biggest concern now & then:
using the API.

Mobile API connectivity



<http://www.flickr.com/photos/stevewilhelm/2591525747/sizes/l/in/photostream/>



Mobile API connectivity (1)

CRM

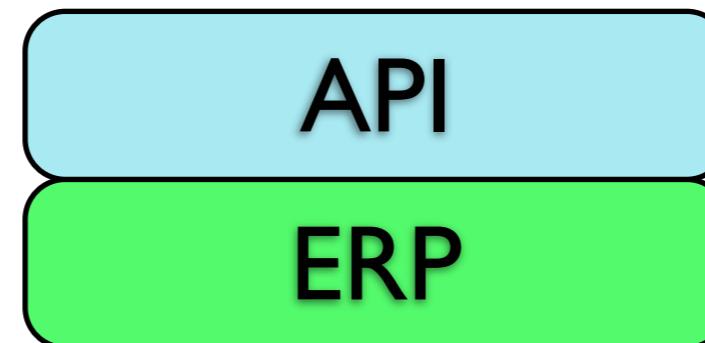
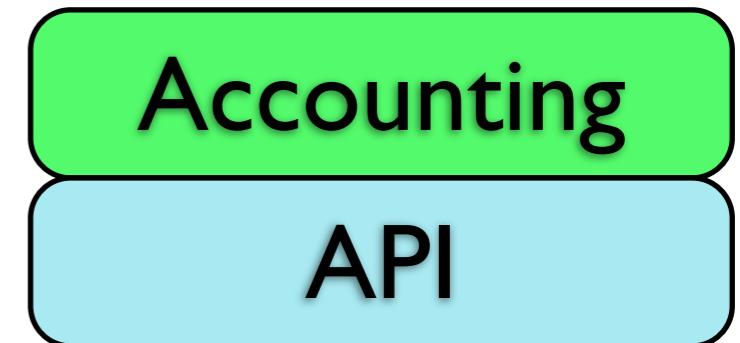
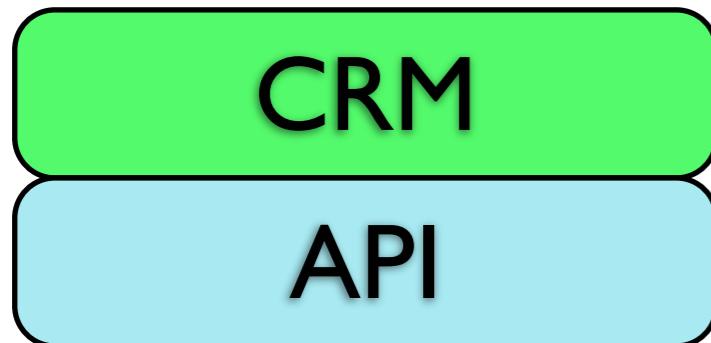
Accounting

ERP

- ▶ ISOLATED APPLICATIONS



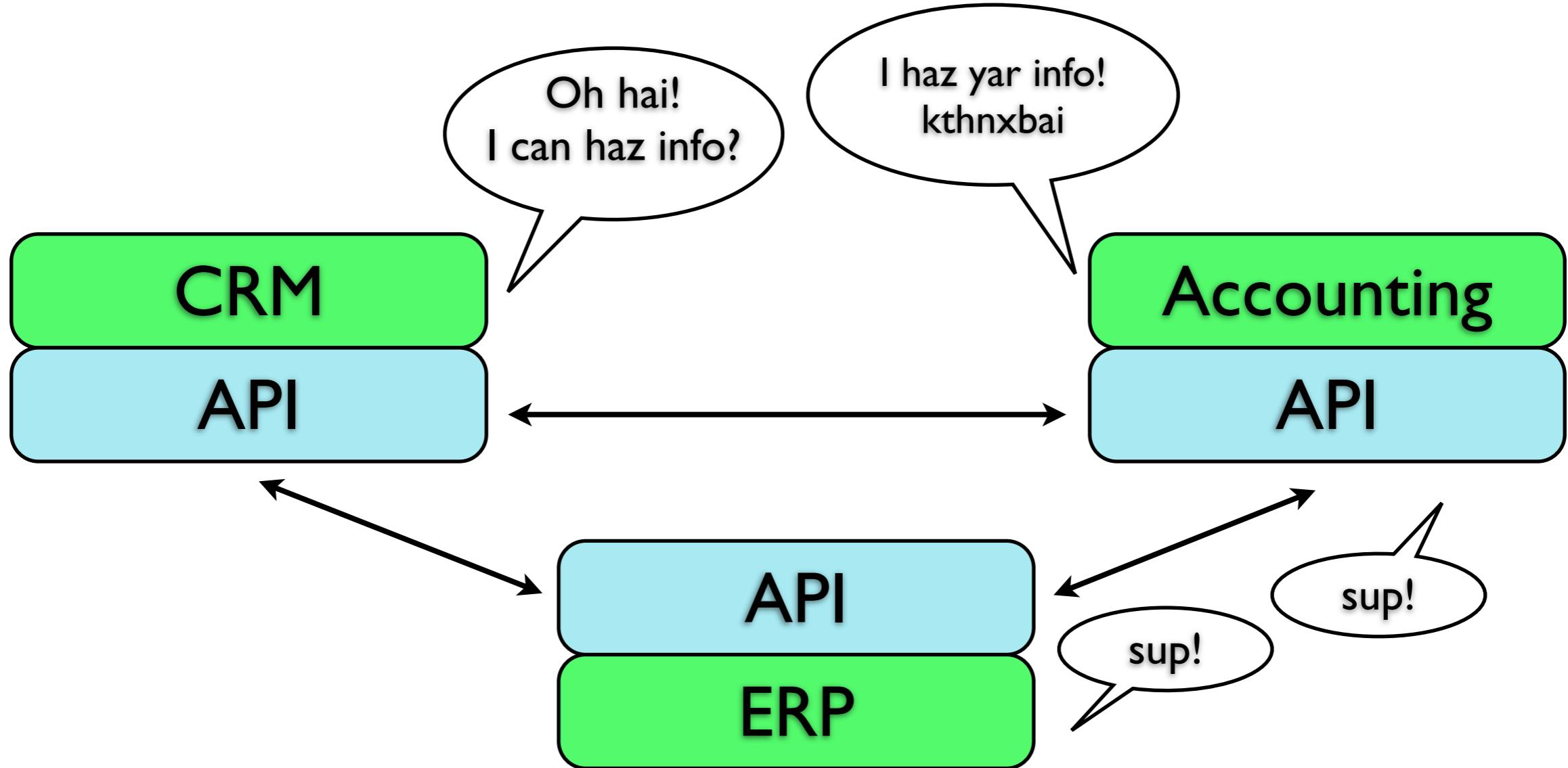
Mobile API connectivity (1)



- ▶ API'S ON TOP OF APPLICATIONS



Mobile API connectivity (1)



- ▶ LET'S START CHATTING



Mobile API connectivity (2)

Twitter

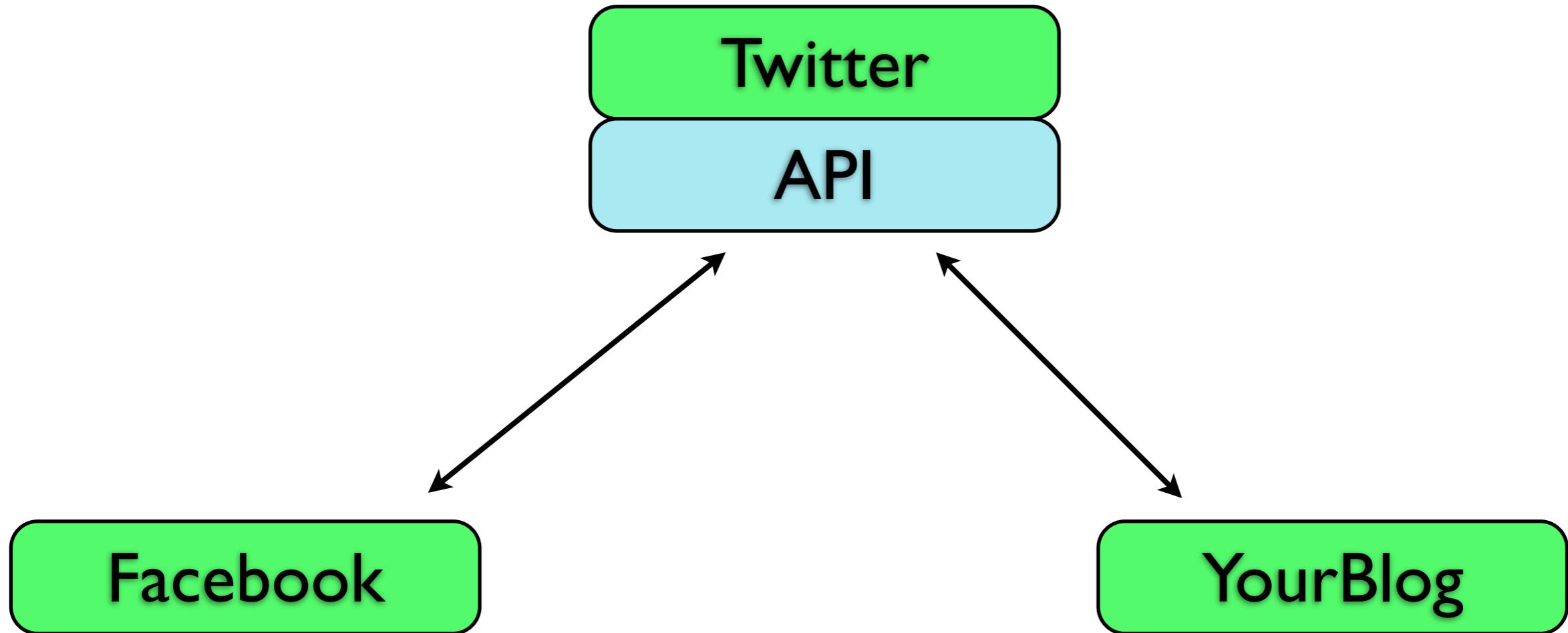
API

Facebook

YourBlog



Mobile API connectivity (2)





Mobile API connectivity (2)



JayTaph

Don't thank me yet.. there will be
some serious API-bashing..
@dzuelke-style :p

11 minutes ago via TweetDeck

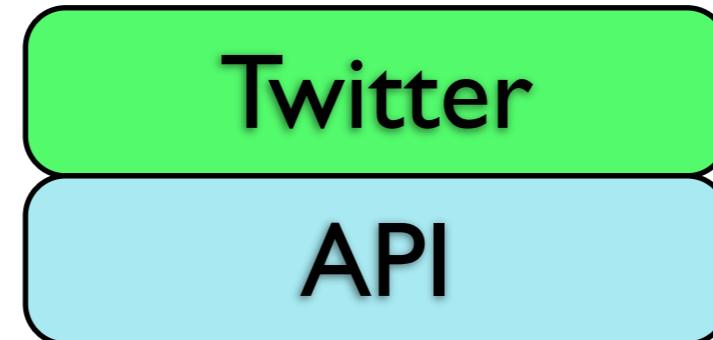
Did a 3km lap. Wasn't feeling like it, but the sun was shining!
:-)
about 9 hours ago via TweetDeck

Did a 4km run. Pace was not too bad but a bit windy.
5:08 PM Mar 5th via TweetDeck

@irule25 Da's wel heul lang geleden :-)
12:14 PM Mar 5th via TweetDeck in reply to irule25

@irule25 Plaatsen op www.gratisoptehalen.nl? :-)
11:29 AM Mar 5th via TweetDeck in reply to irule25

@_joris Cool.. will check it out..
11:22 AM Mar 5th via TweetDeck in reply to _joris



Current tweets

Don't thank me yet.. there will be
some serious API-bashing..
@dzuelke-style :p
11 minutes ago

Did a 3km lap. Wasn't feeling like it,
but the sun was shining! :-)
about 9 hours ago

Did a 4km run. Pace was not too bad
but a bit windy.
4:08 PM yesterday

@irule25 Da's wel heul lang geleden
:-)
11:14 AM yesterday

@irule25 Plaatsen op
www.gratisoptehalen.nl? :-)
10:29 AM yesterday

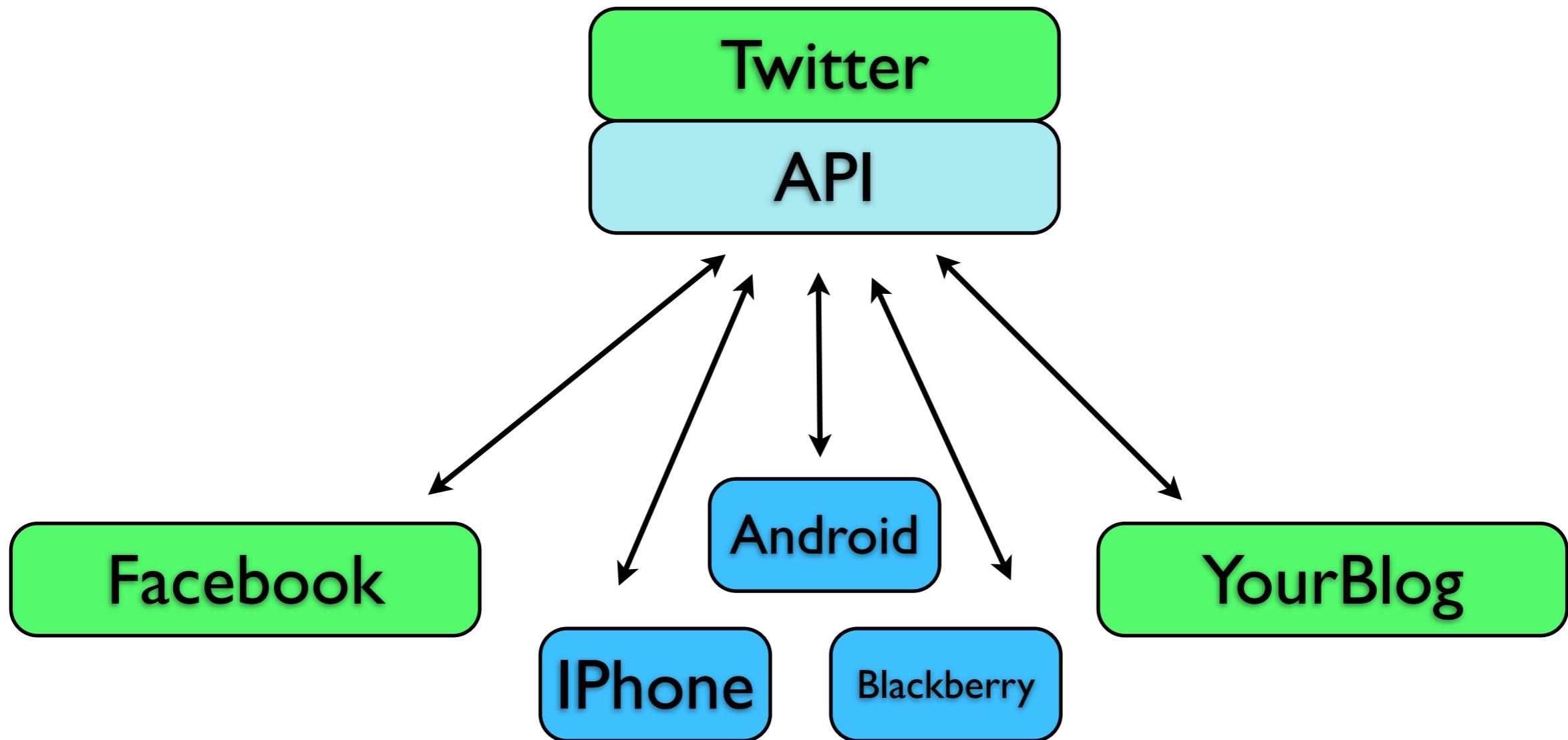
Facebook

YourBlog

- ▶ USING OTHER APPLICATION'S DATA



Mobile API connectivity (2)



- ▶ ONE TO MANY



Mobile API connectivity (3)





Mobile API connectivity (4)

Is your application **REALLY** ready for mobile connectivity?



5 Mobile API connectivity (5)

- Mobile bandwidth is expensive.
- Connectivity is slow and unreliable.
- Client side processing is harder.

► MOBILE FACTS

5 Mobile API connectivity (5)

- Not giving client what they want.
- Updating your API will be harder.
- BC breaks are a no-go.

► MOBILE FACTS



Mobile API connectivity (6)

- ▶ PROBLEMS WITH THE JOIND.IN API

5 Mobile API connectivity (6)

■ Authentication

- ▶ PROBLEMS WITH THE JOIND.IN API

Mobile API connectivity (6)

- Authentication
- Too much data being returned
(>300Kb per request)

► PROBLEMS WITH THE JOIND.IN API

7 Mobile API connectivity (6)

- Authentication
- Too much data being returned
(>300Kb per request)
- No pagination (want first 10 talks, get 200+)

► PROBLEMS WITH THE JOIND.IN API

Mobile API connectivity (6)

- Authentication
- Too much data being returned
(>300Kb per request)
- No pagination (want first 10 talks, get 200+)
- Duplicate data (tracks, talks, speaker info)

► PROBLEMS WITH THE JOIND.IN API

Mobile API connectivity (6)

- Authentication
 - Too much data being returned (>300Kb per request)
 - No pagination (want first 10 talks, get 200+)
 - Duplicate data (tracks, talks, speaker info)
 - Occasional BC breaks
- PROBLEMS WITH THE JOIND.IN API

7 Mobile API connectivity (7)



This will all change in the v2.0 API release.
(hopefully)



Writing (native) Android applications



```
29     super.onCreate(savedInstanceState);
30
31     // Set layout
32     setContentView(R.layout.eventtracks);
33
34     // Get event ID from the intent scratch board
35     try {
36         this.eventJSON = new JSONObject(getIntent().getStringExtra("eventJSON"));
37     } catch (JSONException e) {
38         android.util.Log.e("JoinInApp", "No event passed to activity", e);
39     }
40
41     // Set all the event information
42     TextView t;
43     t = (TextView) this.findViewById(R.id.EventTracksCaption);
44     t.setText (this.eventJSON.optString("event_name"));
45
46     // Initialize track list
47     ArrayList<JSONObject> m_tracks = new ArrayList<JSONObject>();
48     m_trackAdapter = new JITTrackAdapter(this, R.layout.trackrow, m_tracks);
49     ListView tracklist =(ListView)findViewById(R.id.ListViewEventTracks);
50     tracklist.setAdapter(m_trackAdapter);
51
52     // Add listview listener so when we click on an talk, we can display details
53     tracklist.setOnItemClickListener(new OnItemClickListener() {
54         public void onItemClick(AdapterView<?>parent, View view, int pos, long id) {
55             // Open event details with additional eventTrack data.
56             Intent myIntent = new Intent ();
57             myIntent.setClass(getApplicationContext(), EventTalks.class);
58             myIntent.putExtra("eventJSON", getIntent().getStringExtra("eventJSON"));
59             myIntent.putExtra("eventTrack", parent.getAdapter().getItem(pos).toString());
60             startActivity(myIntent);
61         }
62     });
63 }
```



Writing (native) Android applications (1)

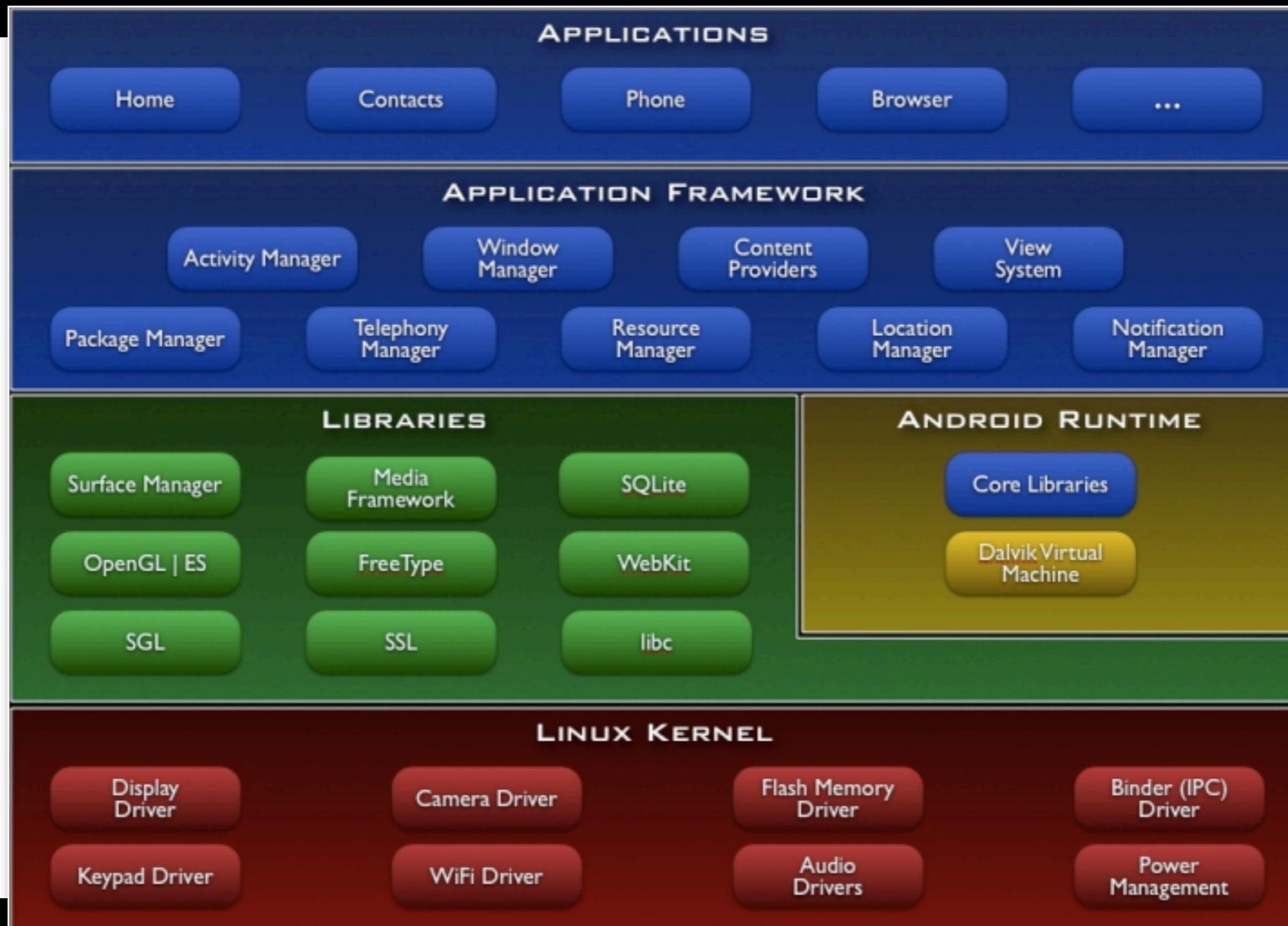
5 Writing (native) Android applications (1)

✓ Native (java)

✗ Non-native (php4android, html5/js)



Writing (native) Android applications (2)

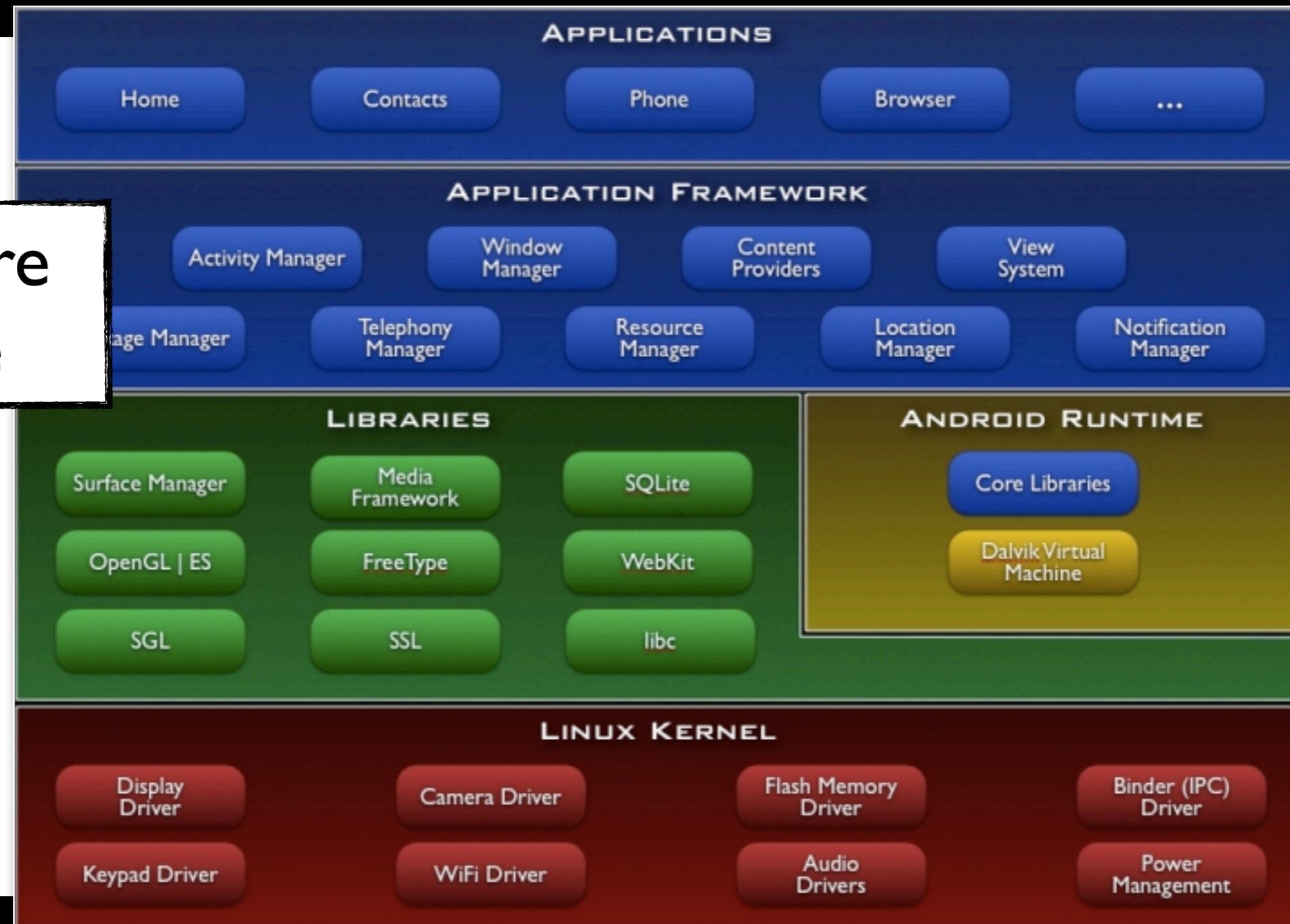


<http://developer.android.com/guide/basics/what-is-android.html>



Writing (native) Android applications (2)

You are
here



Writing (native) Android applications (3)

- Activities
- Layouts
- Intents

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

Activities (1)

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

Activities (1)

- provides a screen with which a user can interact.

Activities (1)

- provides a screen with which a user can interact.
- activities are “stacked”.

Activities (2)

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

Activities (2)

- activity 1 starts

Activities (2)

- activity 1 starts
- activity 2 starts & activity 1 pauzes

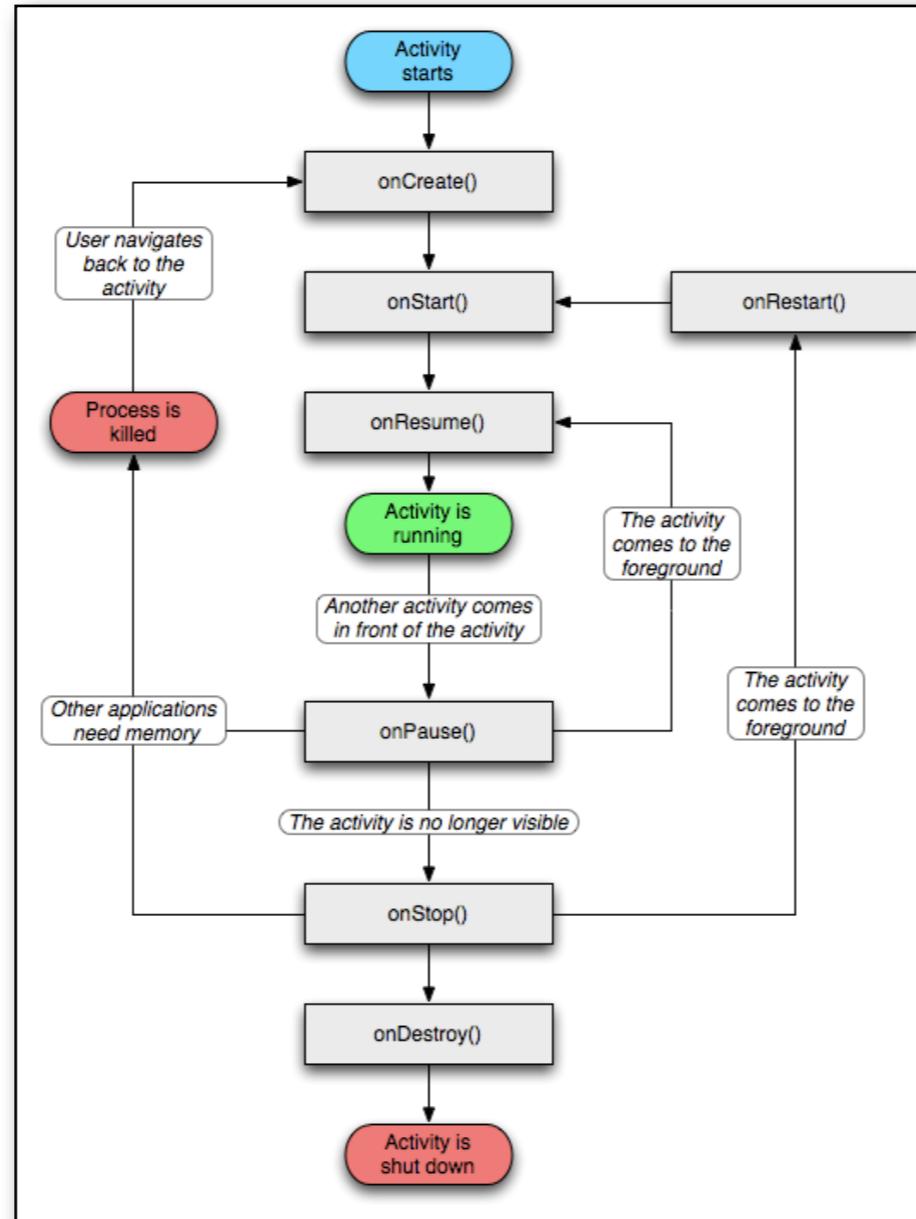
Activities (2)

- activity 1 starts
 - activity 2 starts & activity 1 pauzes
 - activity 2 stops

Activities (2)

- activity 1 starts
 - activity 2 starts & activity 1 pauzes
 - activity 2 stops
- activity 1 resumes

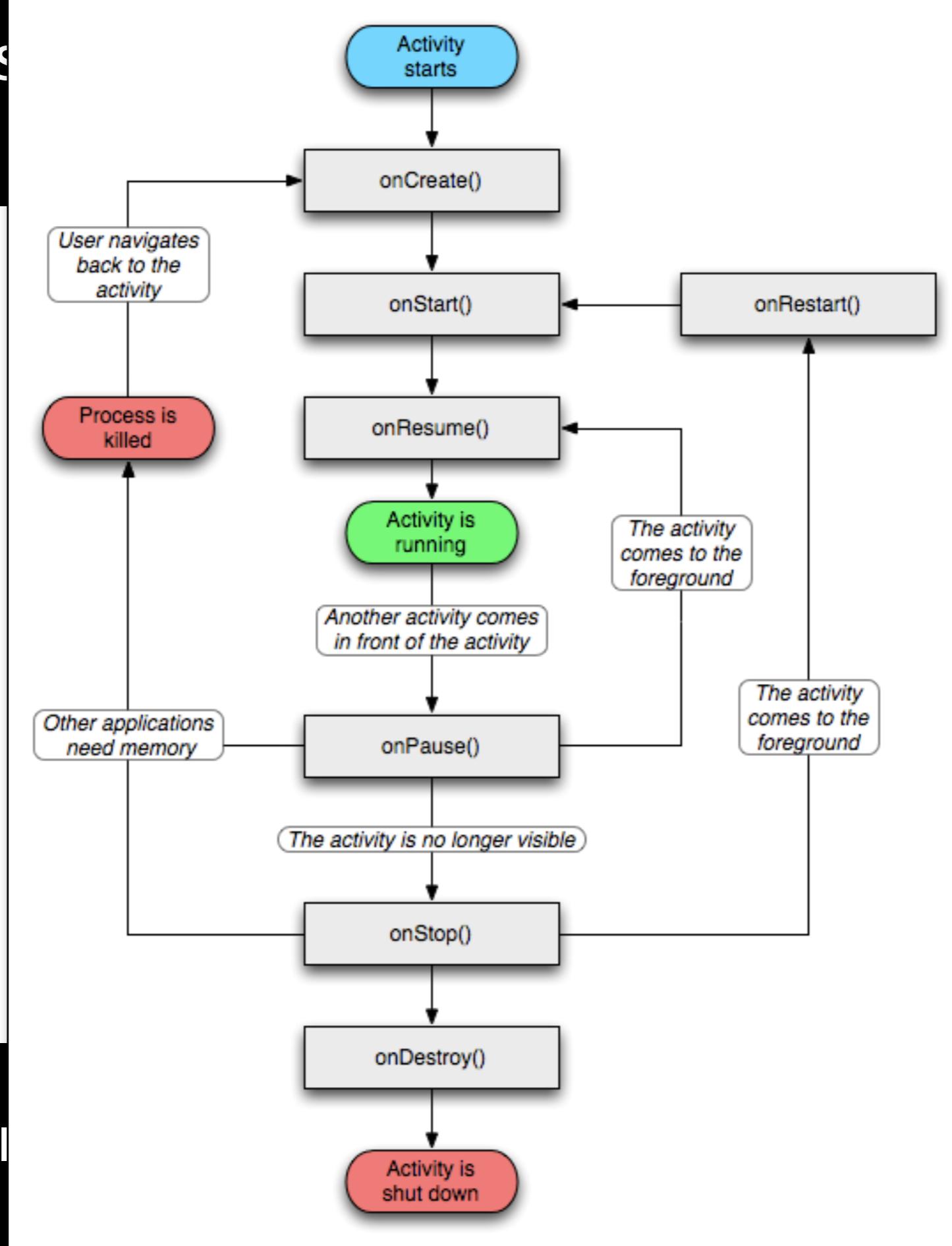
7 Activities (3)



► ACTIVITY LIFE CYCLE

7 Activities

► ACTIVITY LIFE CYCLE



Writing (native) Android applications (3)

- Activities
- Layouts
- Intents

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

7 Layouts (1)

7 Layouts (1)

- Get stuff onto your display

7 Layouts (1)

- Get stuff onto your display
- Created as XML resources

Layouts (1)

- Get stuff onto your display
- Created as XML resources
- “inflated” by android (costly)

7 Layouts (2)

- LinearLayout
- TableLayout
- FrameLayout
- RelativeLayout

Single column or row

<table></table>

<placeholder>

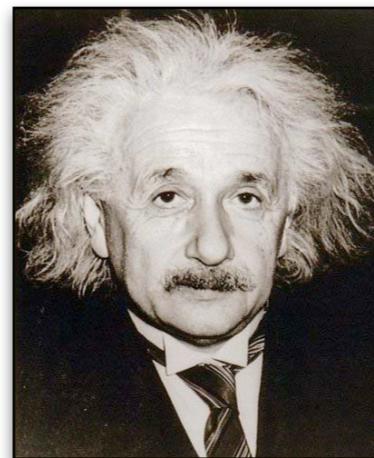
7 Layouts (2)

- LinearLayout
- TableLayout
- FrameLayout
- RelativeLayout

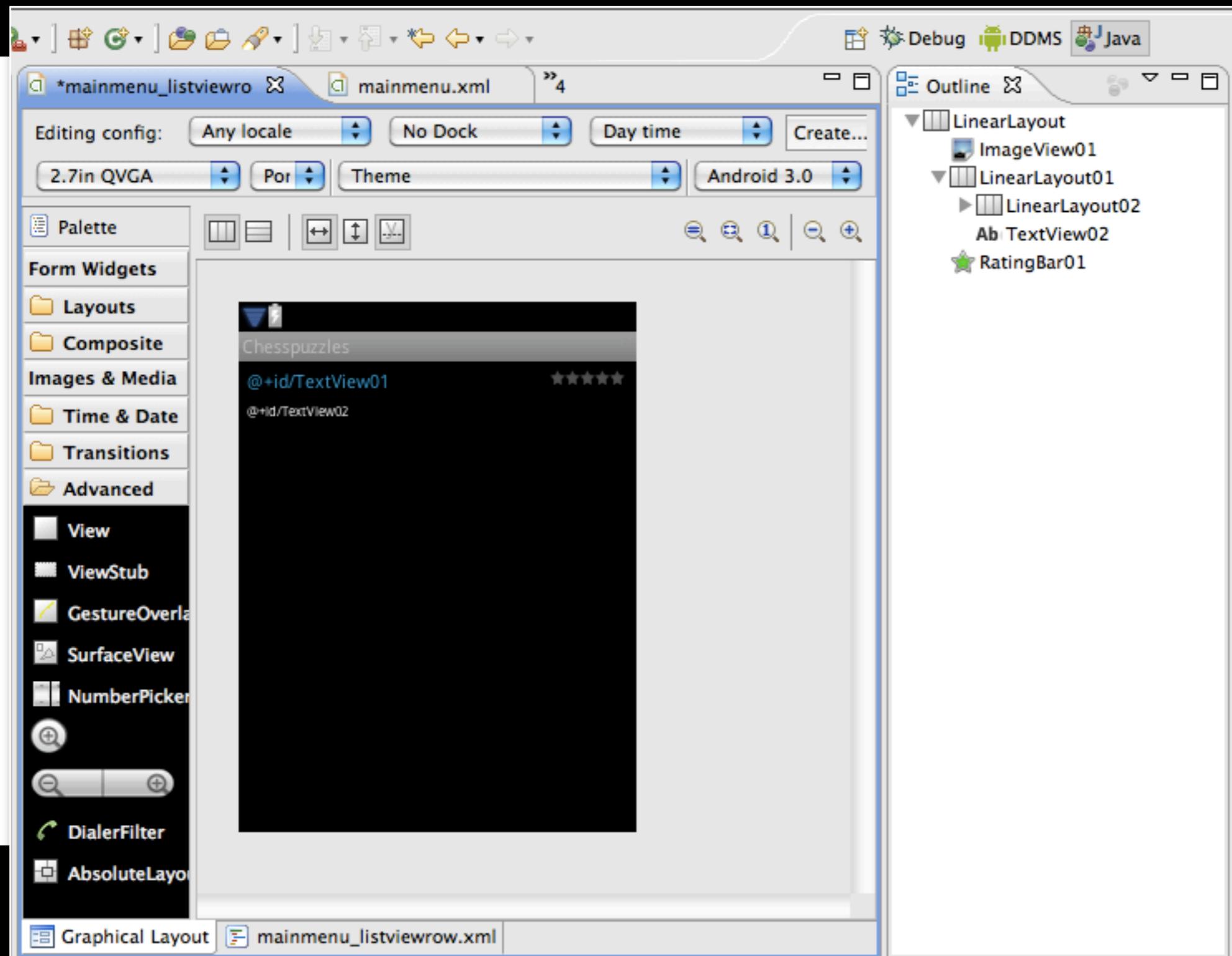
Single column or row

<table></table>

<placeholder>



7 Layouts (3)

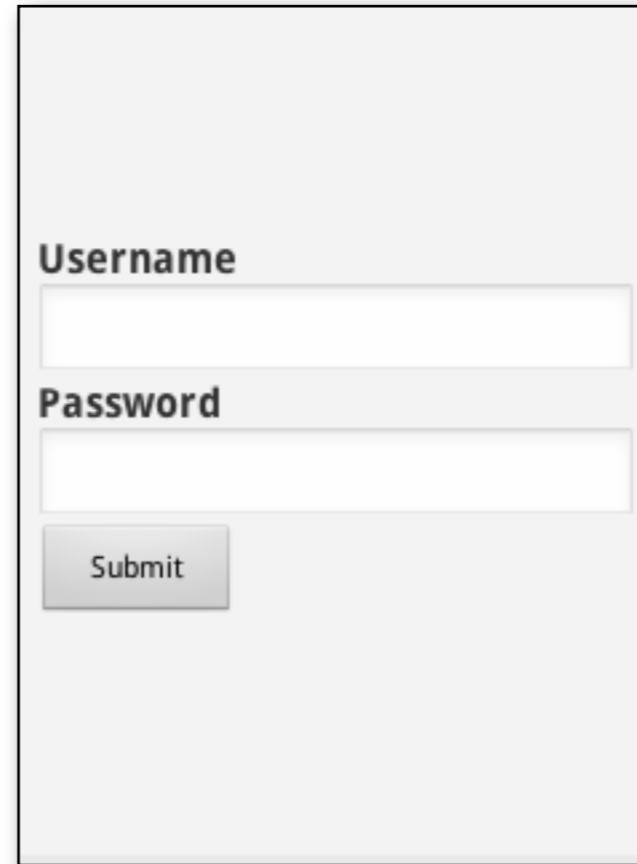


7 Layouts (4)

XML

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:alwaysDrawnWithCache="true">
    <TableLayout
        android:id="@+id/tableLayout1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center_vertical"
        android:layout_margin="10dip">
        <TableRow
            android:id="@+id/tableRow5"
            android:layout_width="wrap_content"
            android:layout_height="match_parent">
            <TextView
                android:layout_height="wrap_content"
                android:layout_width="wrap_content"
                android:textStyle="bold"
                android:id="@+id/TextView01"
                android:textSize="20dip"
                android:text="Username"></TextView>
        </TableRow>
    ....
```

Light theme



Dark (normal) theme



∂ DEFINING ANDROID LAYOUTS

Writing (native) Android applications (3)

- Activities
- Layouts
- Intents

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

Intents (1)

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

Intents (1)

- “message queue” between different android components.

Intents (1)

- “message queue” between different android components.
- Explicit & implicit intents

Intents (2)

5 Intents (2)

Start a new activity (explicit):

```
Intent myIntent = new Intent();  
myIntent.setClass(class.this, com.another.class);  
myIntent.putExtra("Data", "somedata");  
startActivity(myIntent);
```

5 Intents (2)

Start a new activity (explicit):

```
Intent myIntent = new Intent();  
myIntent.setClass(class.this, com.another.class);  
myIntent.putExtra("Data", "somedata");  
startActivity(myIntent);
```

Dial a number (implicit):

```
Intent intent = new Intent(Intent.ACTION_DIAL, Uri.parse("tel:123-456789"));  
startActivity(intent);
```

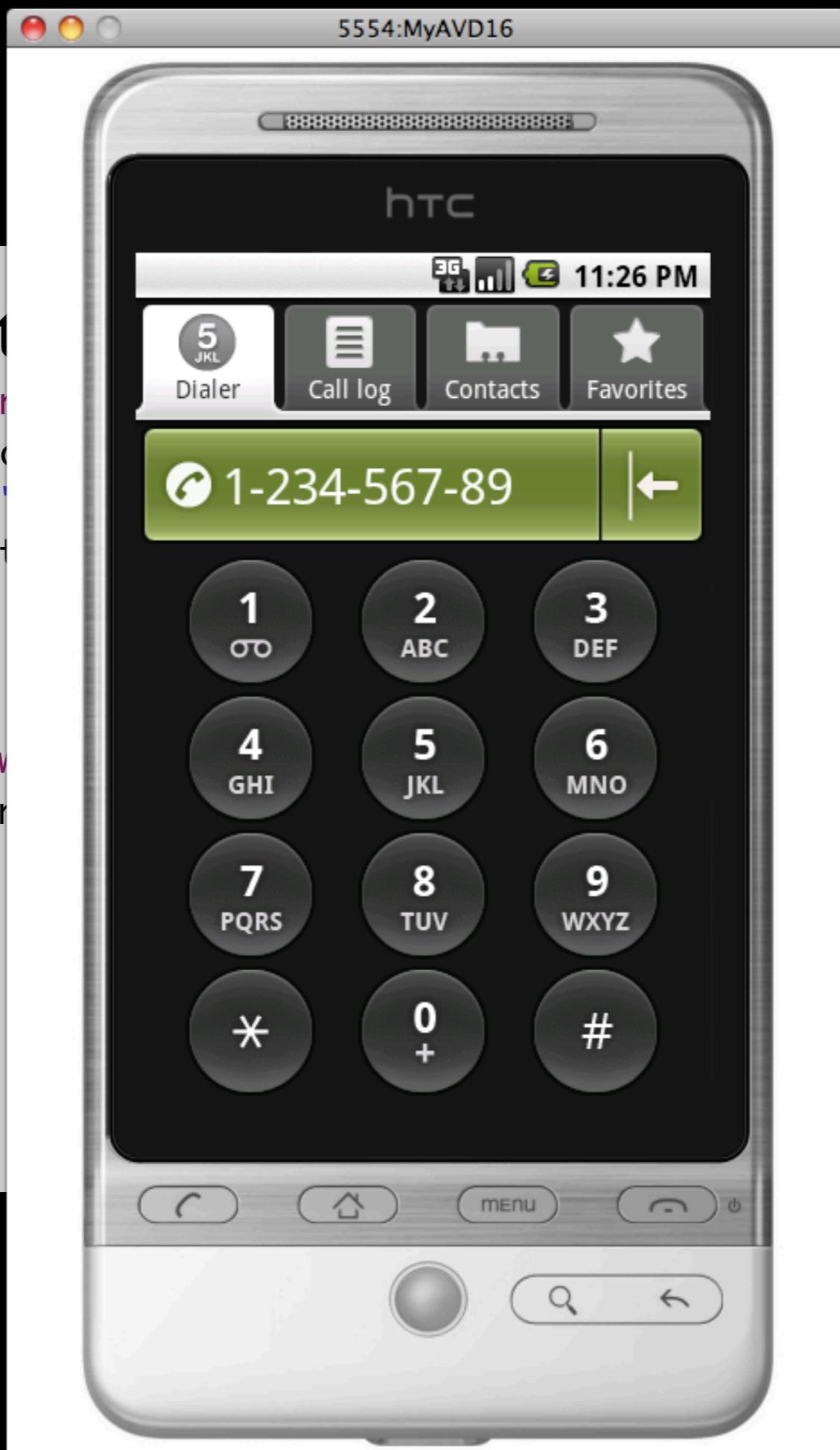
7 Intents (2)

Start a new activity

```
Intent myIntent = new Intent("com.android.dialer.Dialer");  
myIntent.setClass(getApplicationContext(), DialerActivity.class);  
myIntent.putExtra("number", "1-234-567-89");  
startActivity(myIntent);
```

Dial a number

```
Intent intent = new Intent(Intent.ACTION_DIAL);  
intent.setData(Uri.parse("tel:123-456789"));  
startActivity(intent);
```



'tel:123-456789"));

5 Intents (2)

Start a new activity (explicit):

```
Intent myIntent = new Intent();  
myIntent.setClass(class.this, com.another.class);  
myIntent.putExtra("Data", "somedata");  
startActivity(myIntent);
```

Dial a number (implicit):

```
Intent intent = new Intent(Intent.ACTION_DIAL, Uri.parse("tel:123-456789"));  
startActivity(intent);
```

Intents (3)

5 Intents (3)

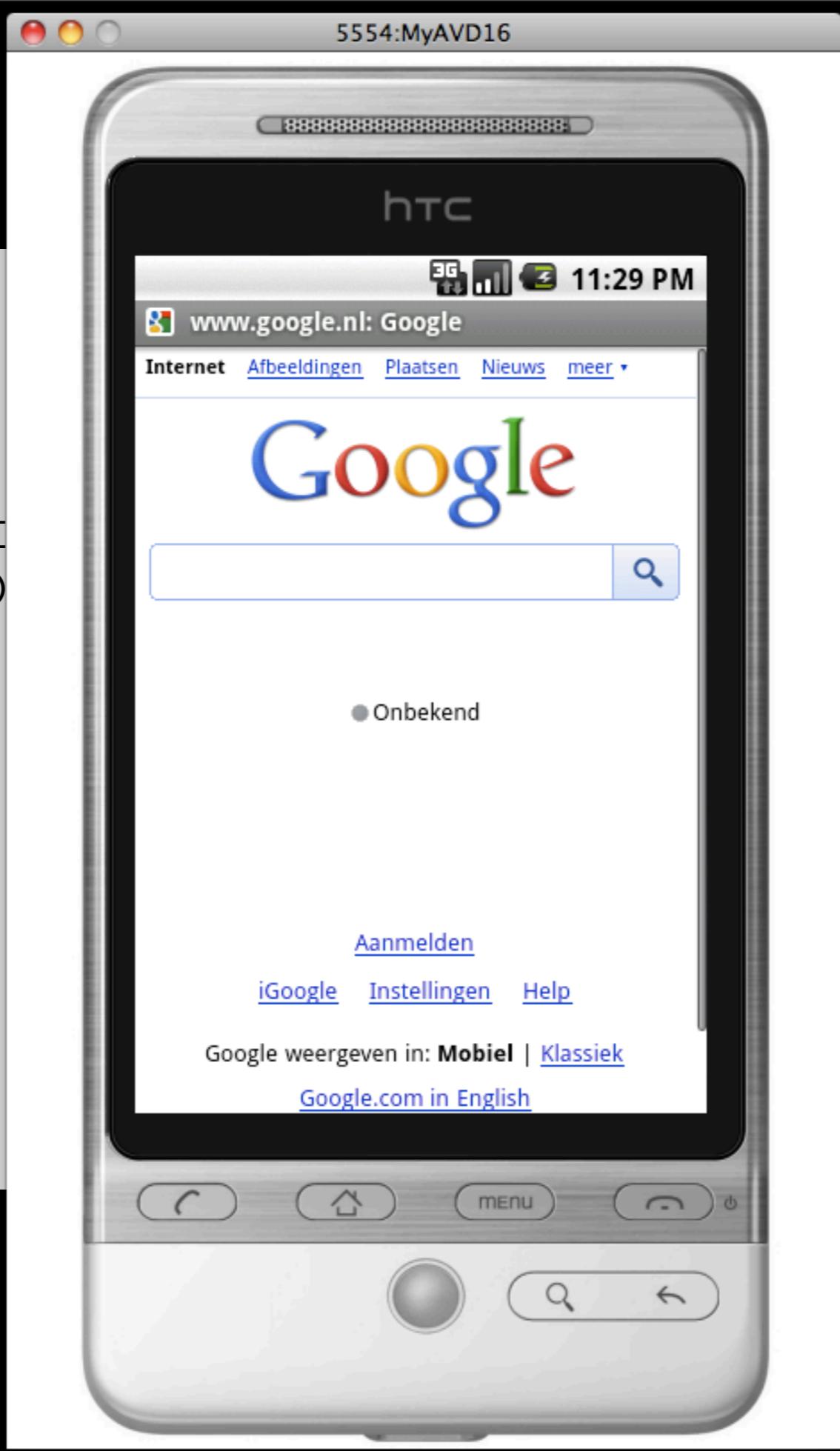
Visit a website (implicit):

```
Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse("http://www.google.com"));
startActivity(intent);
```

7 Intents (3)

Visit a website

```
Intent intent = new Intent();
startActivity(intent)
```



```
tp://www.google.com" ));
```

5 Intents (3)

Visit a website (implicit):

```
Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse("http://www.google.com"));
startActivity(intent);
```

5 Intents (3)

Visit a website (implicit):

```
Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse("http://www.google.com"));
startActivity(intent);
```

Open Google Maps (implicit):

```
Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse("geo:52.257651,21.058044?z=17"));
startActivity(intent);
```

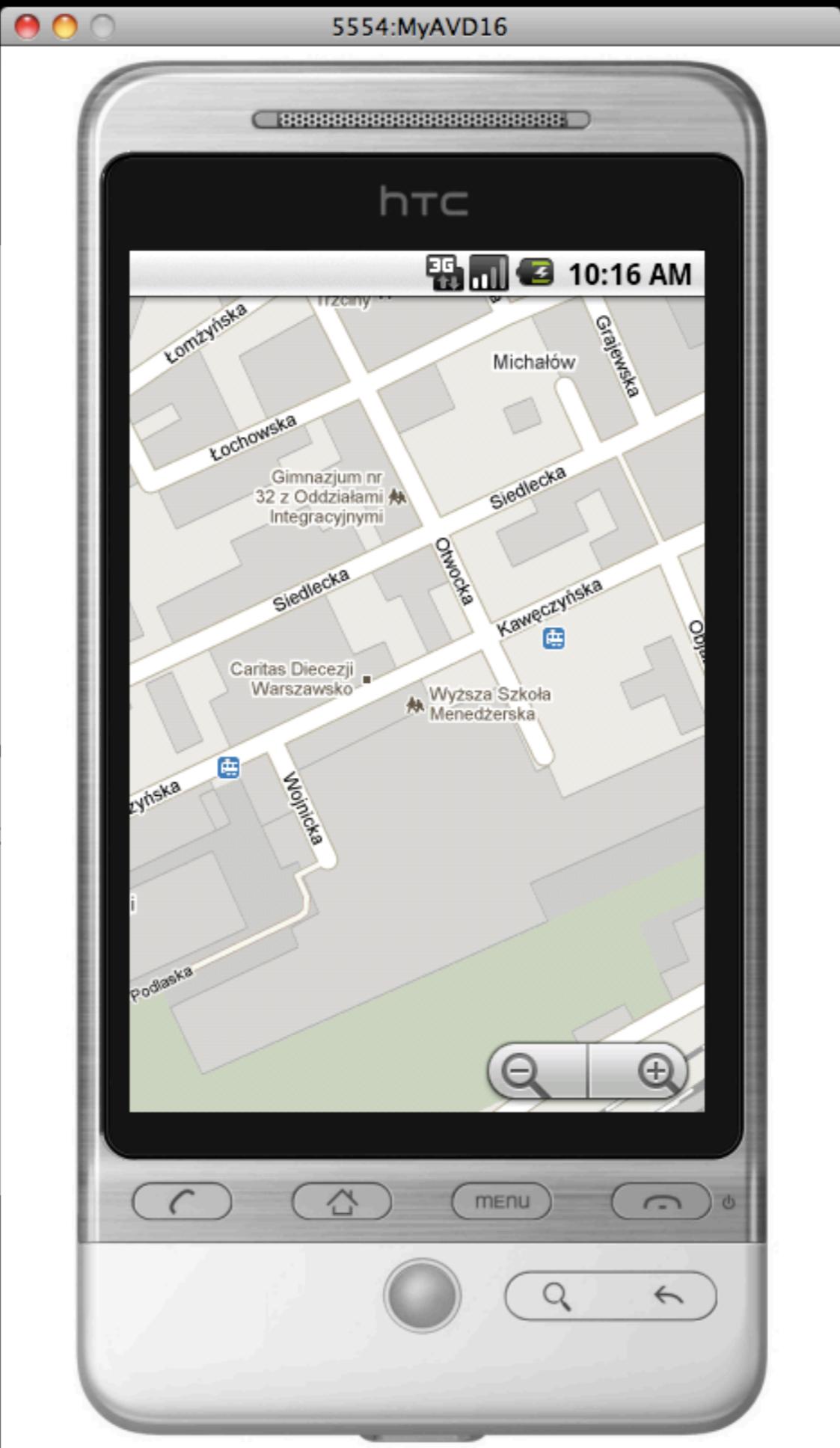
7 Intents (3)

Visit a website

```
Intent intent = new Intent()  
startActivity(intent)
```

Open Google Map

```
Intent intent = new Intent()  
startActivity(intent);
```



[tp://www.google.com" \)\);](http://www.google.com)

[52.257651,21.058044?z=17" \)\);](http://52.257651,21.058044?z=17)

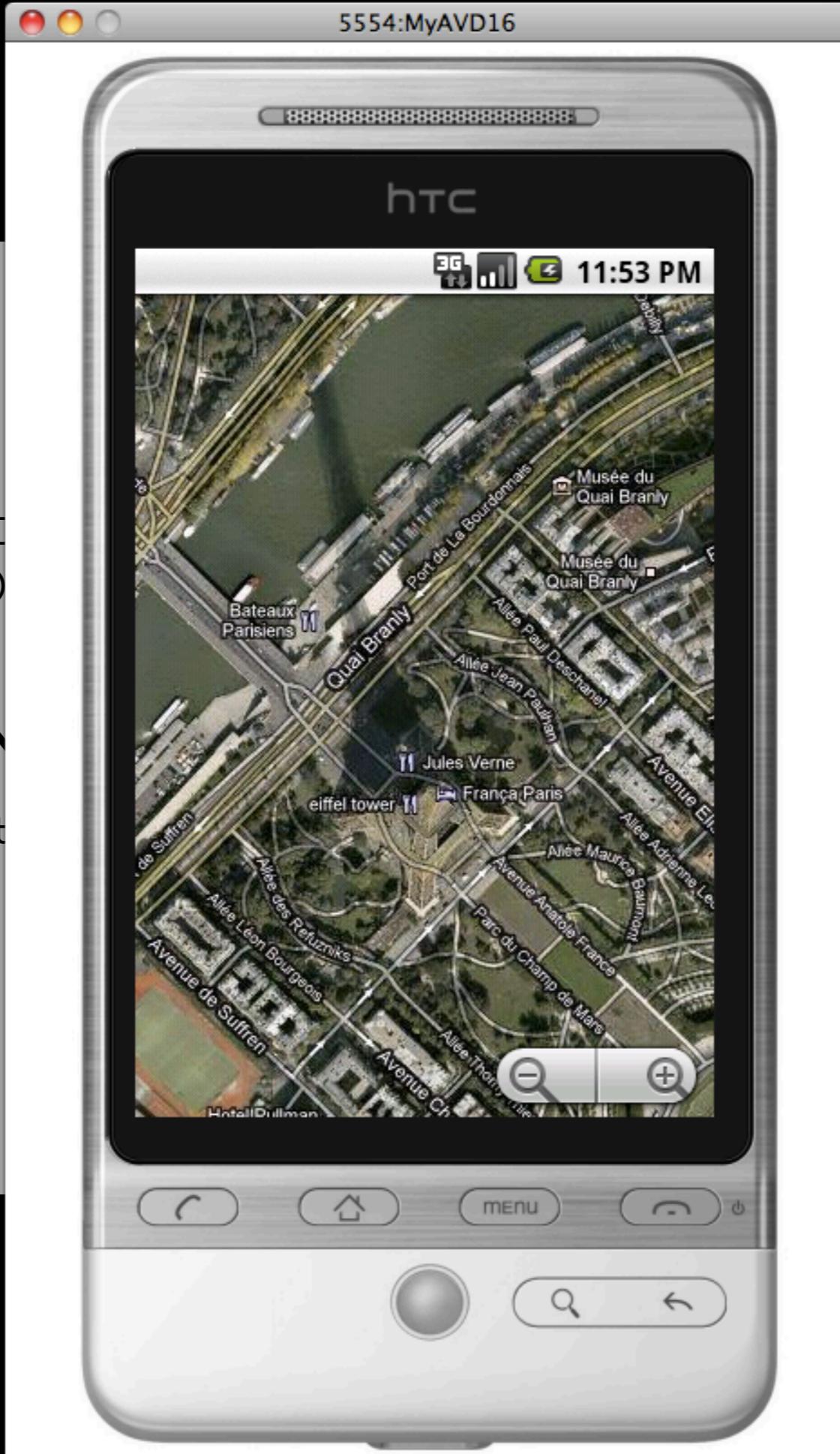
7 Intents (3)

Visit a website

```
Intent intent = new Intent();
startActivity(intent);
```

Open Google Map

```
Intent intent = new Intent();
startActivity(intent);
```



[tp://www.google.com"\(\)\);](http://www.google.com)

[52.257651,21.058044?z=17"\(\)\);](http://52.257651,21.058044?z=17)

7 Nine-patch (1)

Nine-patch (1)

Google Search SafeSearch moderate ▾

Go to Google Home 1,000 results (0.45 seconds) Advanced search

Everything Images Videos News Shopping More

Any size Large Medium Icon Larger than... Exactly...

Any type Face Photo Clip art Line drawing

Any color Full color Black and white

█ █ █ █ █ █

<http://www.aoeale.com/webho?hl=en>

The image shows a Google search results page for the query "nine-patch". The search bar at the top contains "nine-patch". Below it, there are several search filters: "Everything" (selected), "Images" (highlighted in blue), "Videos", "News", "Shopping", and "More". Under "Images", there are filters for "Any size" (Large, Medium, Icon, Larger than..., Exactly...), "Any type" (Face, Photo, Clip art, Line drawing), and "Any color" (Full color, Black and white). A color palette at the bottom allows users to filter by specific colors. The main content area displays a grid of 20 images related to nine-patch quilts. The first row contains 10 images, and the second row contains 10 more, with a caption "Page 2" above the second row. The images show various designs and patterns for nine-patch quilts, including traditional 3x3 grid patterns and more complex variations.

Nine-patch (1)

Google Search SafeSearch moderate ▾

Go to Google Home 1,000 results (0.45 seconds) Advanced search

Everything

Images

Videos

News

Shopping

More

Any size

Large

Medium

Icon

Larger than...

Exactly...

Any type

Face

Photo

Clip art

Line drawing

Any color

Full color

Black and white

<http://www.aoeale.com/webho?hl=en>

The image shows a Google search results page for the query "nine-patch". The results are displayed in two main sections: the first section contains 12 thumbnail images of different quilt patterns, and the second section, labeled "Page 2", contains another 8 thumbnail images. The thumbnails include various designs such as the classic nine-patch grid, floral patterns, and more complex geometric designs. The left sidebar of the search interface is visible, showing navigation options like "Everything", "Images" (which is selected), and "SafeSearch" settings.

7 Nine-patch (2)

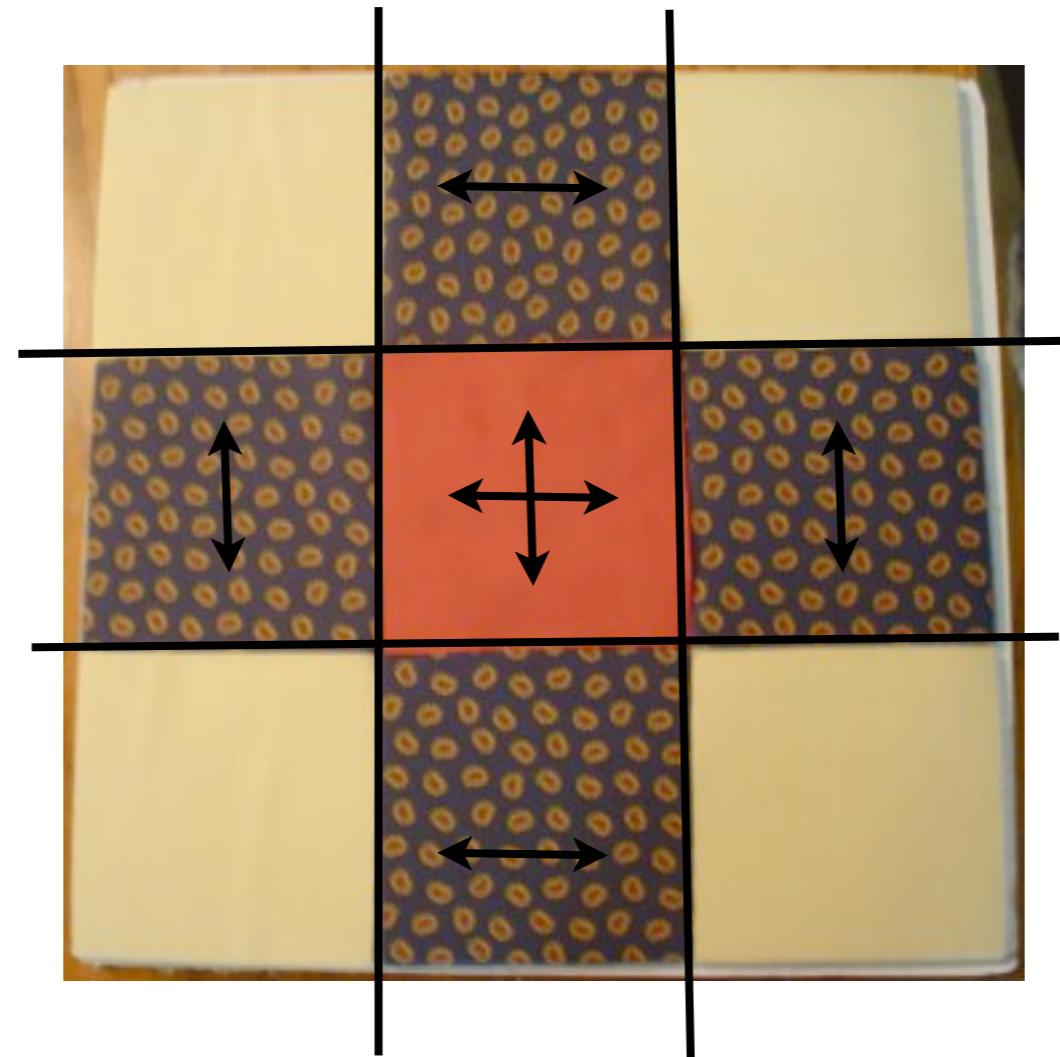


<http://www.pookiescreations.com/images/lrg%209%20patch%20kit%20005.jpg>

Nine-patch (2)



Nine-patch (3)

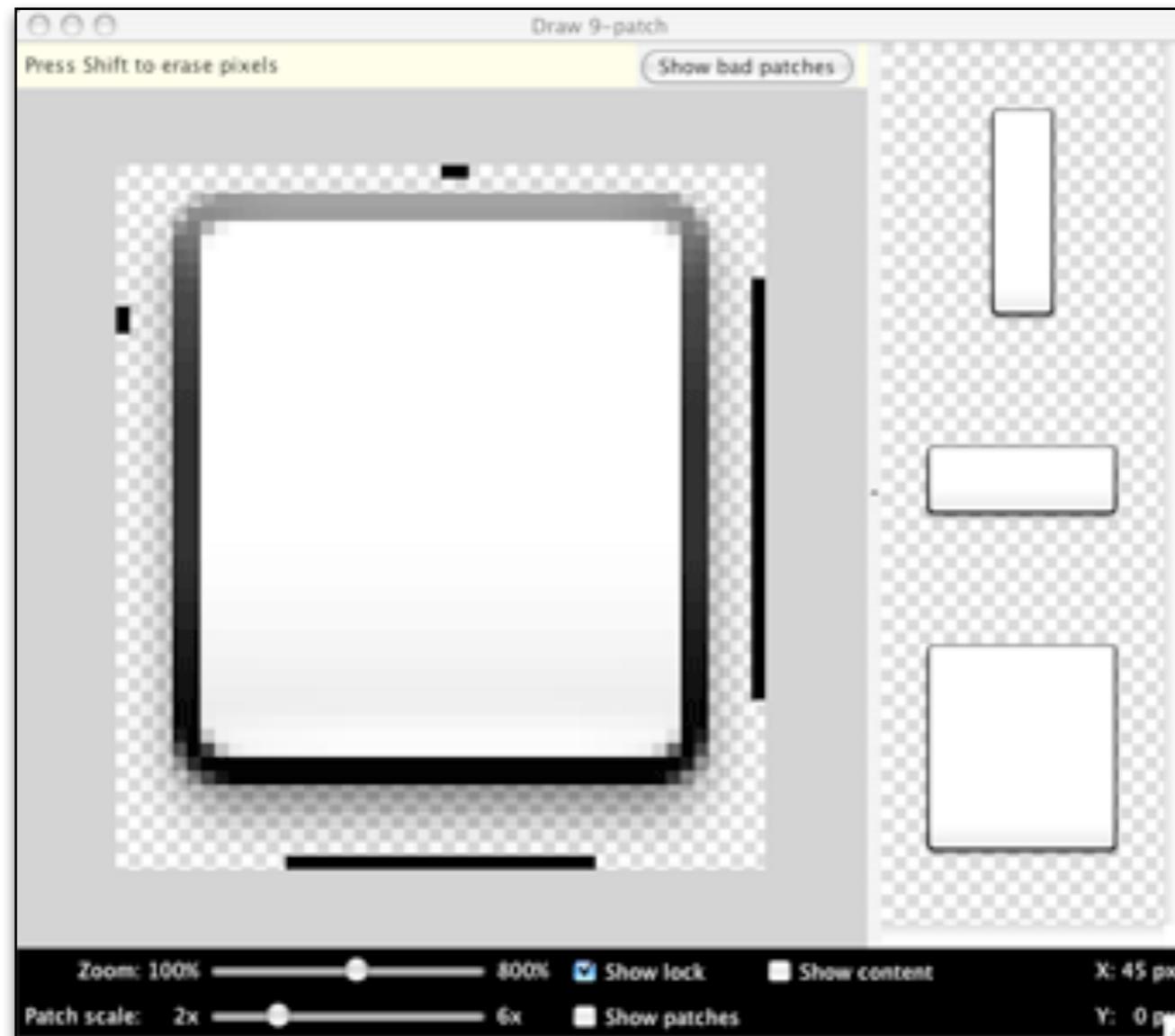


7 Nine-patch (4)



- ▶ ALMOST IMPOSSIBLE TO TELL THAT THIS IS PHOTOSHOPPED

Nine-patch (5)



► A BETTER EXAMPLE

<http://developer.android.com/images/draw9patch-norm.png>



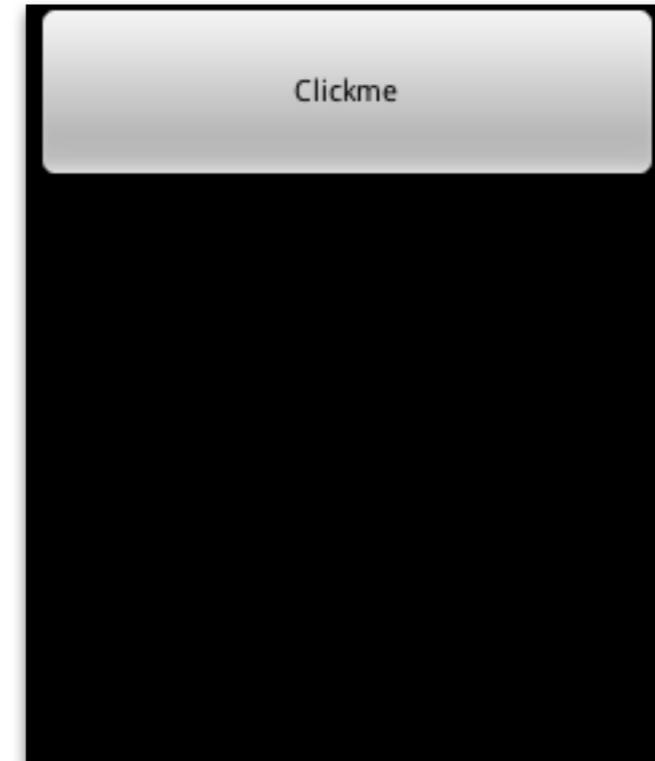
<http://www.flickr.com/photos/oskay/472097903/>



Code examples (1)

res/layout/main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent">
    <Button
        android:text="Clickme"
        android:id="@+id/button1"
        android:layout_height="wrap_content"
        android:layout_width="fill_parent"></Button>
</LinearLayout>
```



- ▶ DEFINE LAYOUT

Code examples (2)

```
package com.fourdevelopers;

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

public class main extends Activity {

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
    }
}
```

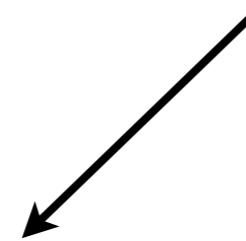
- ▶ HELLO WORLD



Code examples (2)

```
package com.fourdevelopers;  
  
import android.app.Activity;  
import android.os.Bundle;  
  
public class main extends Activity {  
  
    public void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.main);  
    }  
}
```

Activity



- ▶ HELLO WORLD



Code examples (2)

```
package com.fourdevelopers;

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

public class main extends Activity {

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
    }
}
```

Activity

Saved state after
onSaveInstanceState()

- ▶ HELLO WORLD

7 Code examples (2)

```
package com.fourdevelopers;  
  
import android.app.Activity;  
import android.os.Bundle;  
  
public class main extends Activity {  
  
    public void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.main);  
    }  
}
```

Activity

Saved state after
onSaveInstanceState()

Layout XML to display

- ▶ HELLO WORLD



Code examples (3)

```
package com.fourdevelopers;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.Toast;

public class main extends Activity implements OnClickListener {

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        Button button = (Button)findViewById(R.id.button1);
        button.setOnClickListener(this);
    }

    public void onClick(View v) {
        Toast.makeText(this, "Button clicked", Toast.LENGTH_SHORT).show();
    }
}
```

- ▶ REMEMBER: TMTOWTDI!



Code examples (3)

```
package com.fourdevelopers;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.Toast;

public class main extends Activity implements OnClickListener {

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        Button button = (Button)findViewById(R.id.button1);
        button.setOnClickListener(this);
    }

    public void onClick(View v) {
        Toast.makeText(this, "Button clicked", Toast.LENGTH_SHORT).show();
    }
}
```

Imports



- ▶ REMEMBER: TMTOWTDI!



Code examples (3)

```
package com.fourdevelopers;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.Toast;

public class main extends Activity implements OnClickListener {

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        Button button = (Button)findViewById(R.id.button1);
        button.setOnClickListener(this);
    }

    public void onClick(View v) {
        Toast.makeText(this, "Button clicked", Toast.LENGTH_SHORT).show();
    }
}
```

Imports

Implement click listener

- ▶ REMEMBER: TMTOWTDI!

Code examples (3)

```
package com.fourdevelopers;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.Toast;

public class main extends Activity implements OnClickListener {

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        Button button = (Button)findViewById(R.id.button1);
        button.setOnClickListener(this);
    }

    public void onClick(View v) {
        Toast.makeText(this, "Button clicked", Toast.LENGTH_SHORT).show();
    }
}
```

Imports

Implement click listener

Connect click listener to button

- ▶ REMEMBER: TMTOWTDI!

Code examples (3)

```
package com.fourdevelopers;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.Toast;

public class main extends Activity implements OnClickListener {

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        Button button = (Button)findViewById(R.id.button1);
        button.setOnClickListener(this);
    }

    public void onClick(View v) {
        Toast.makeText(this, "Button clicked", Toast.LENGTH_SHORT).show();
    }
}
```

Imports

Implement click listener

Connect click listener to button

Called when “something” is clicked

- ▶ REMEMBER: TMTOWTDI!

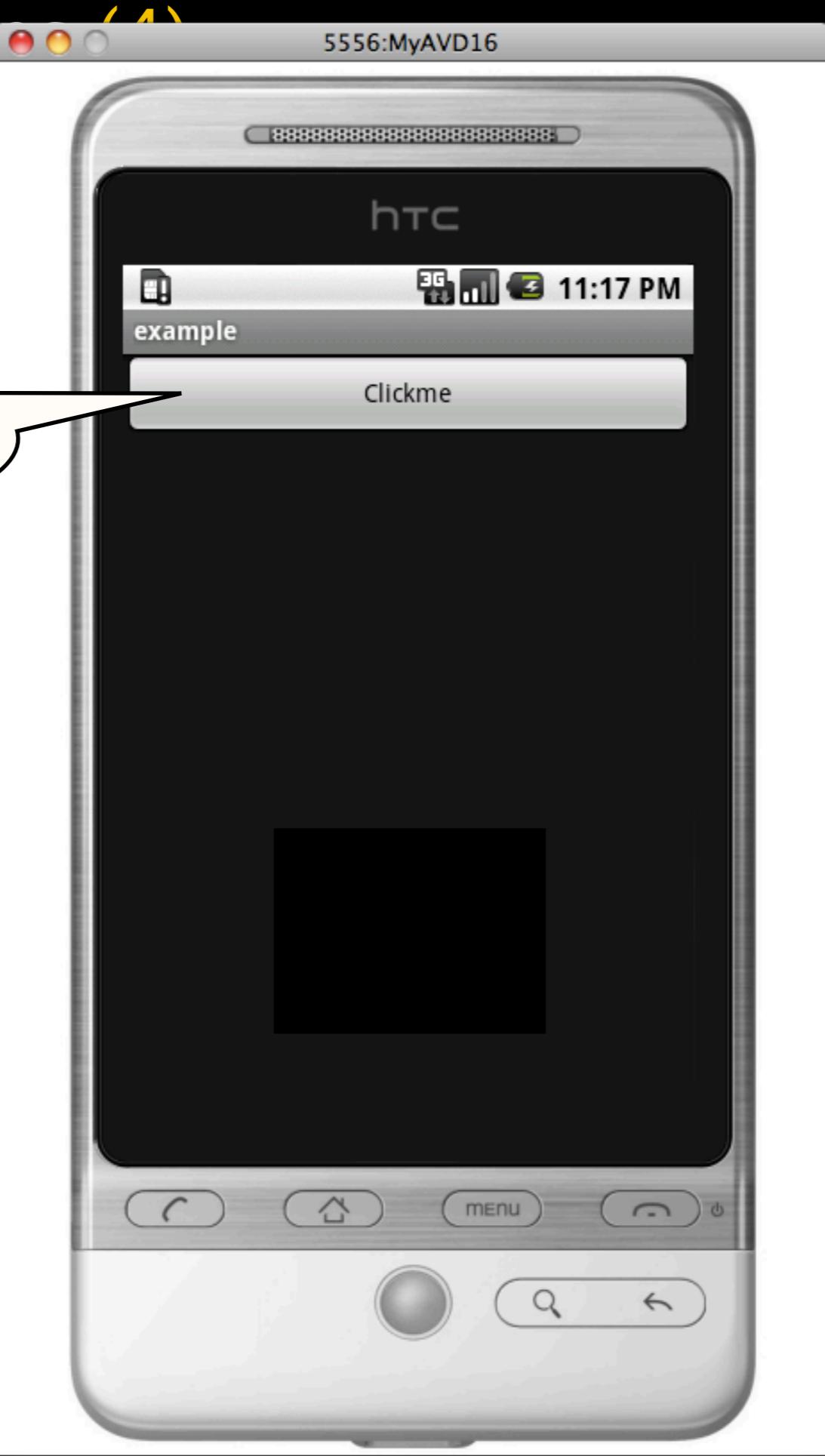


Code example



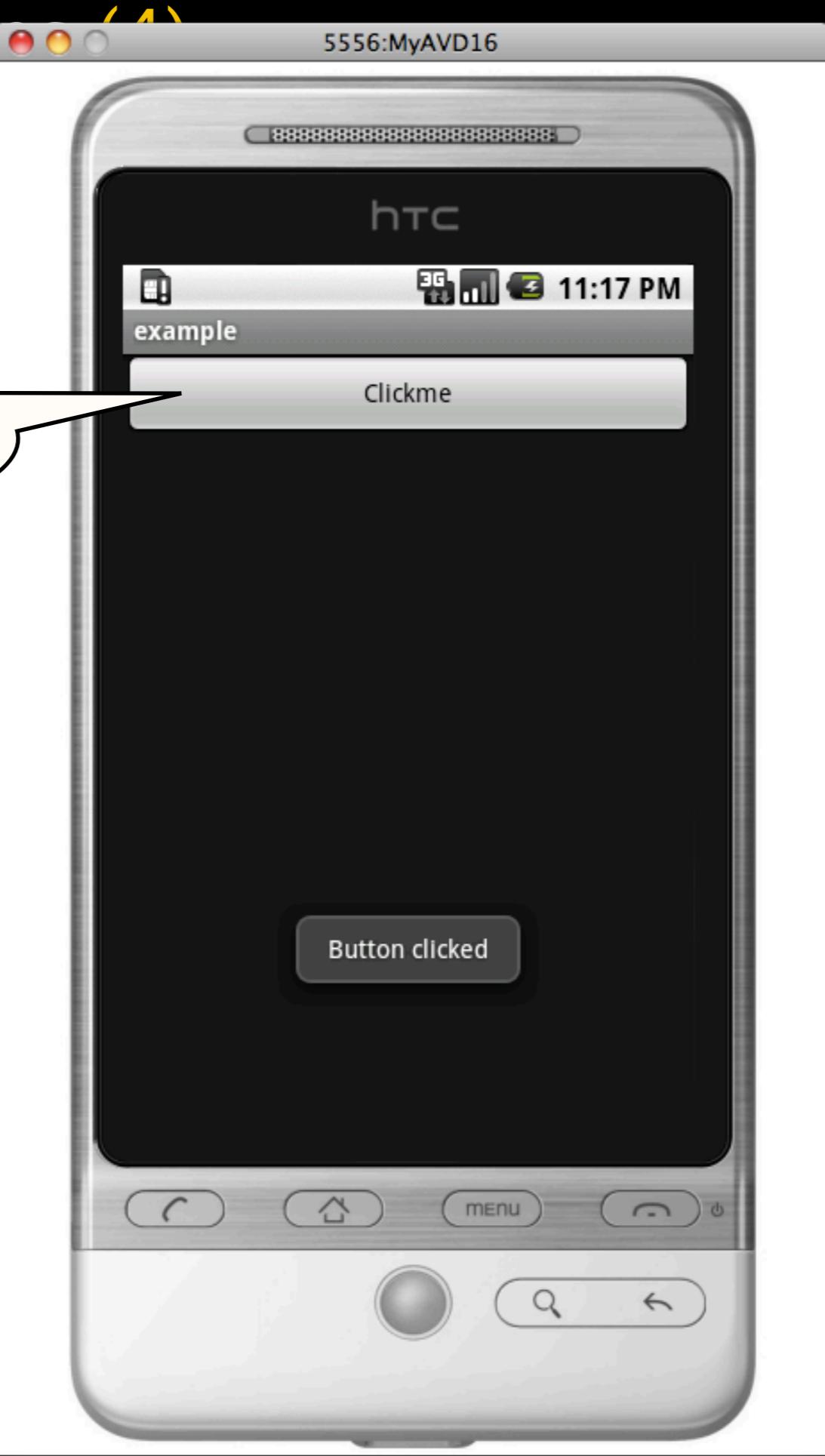


Code example



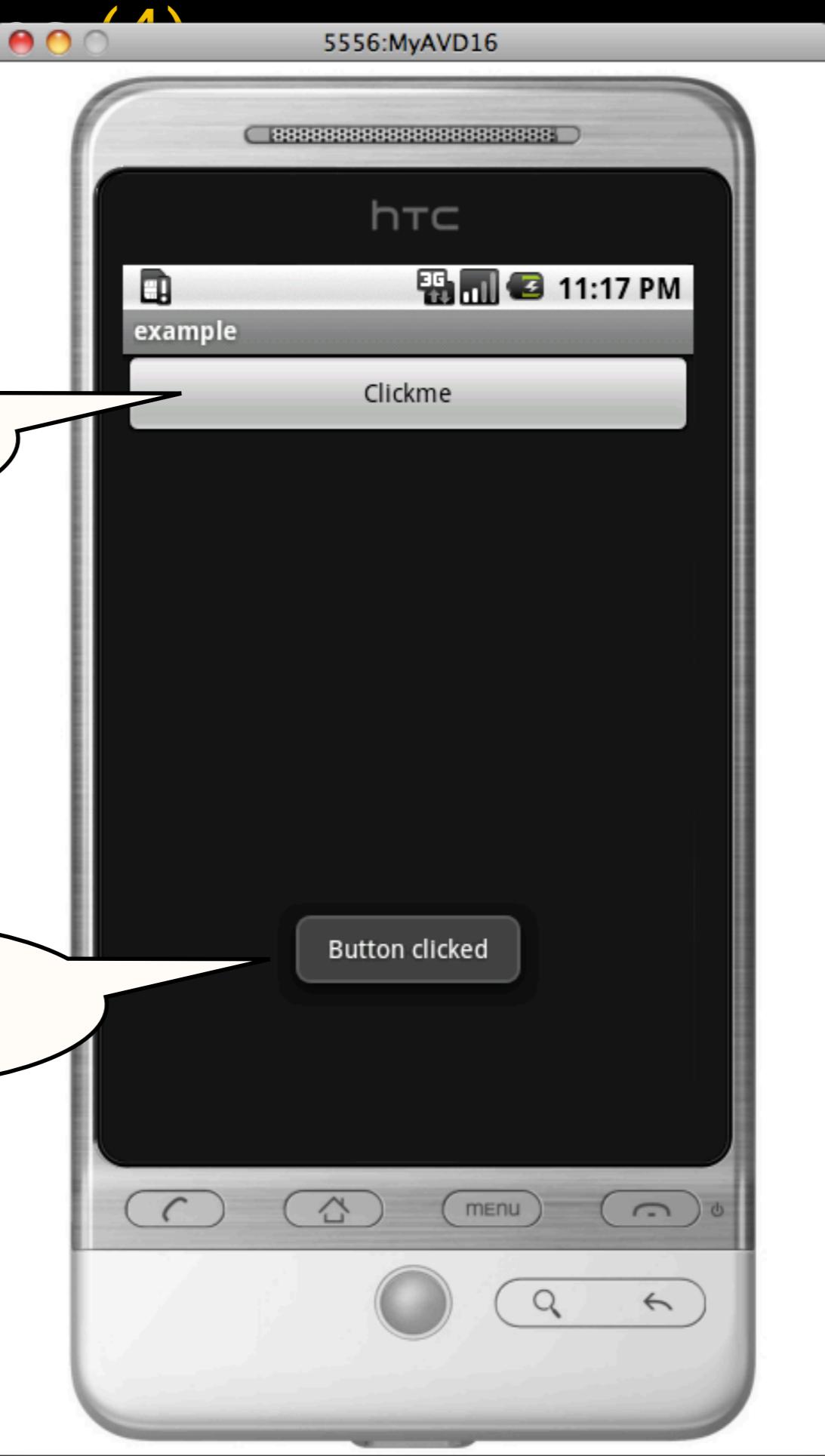


Code example

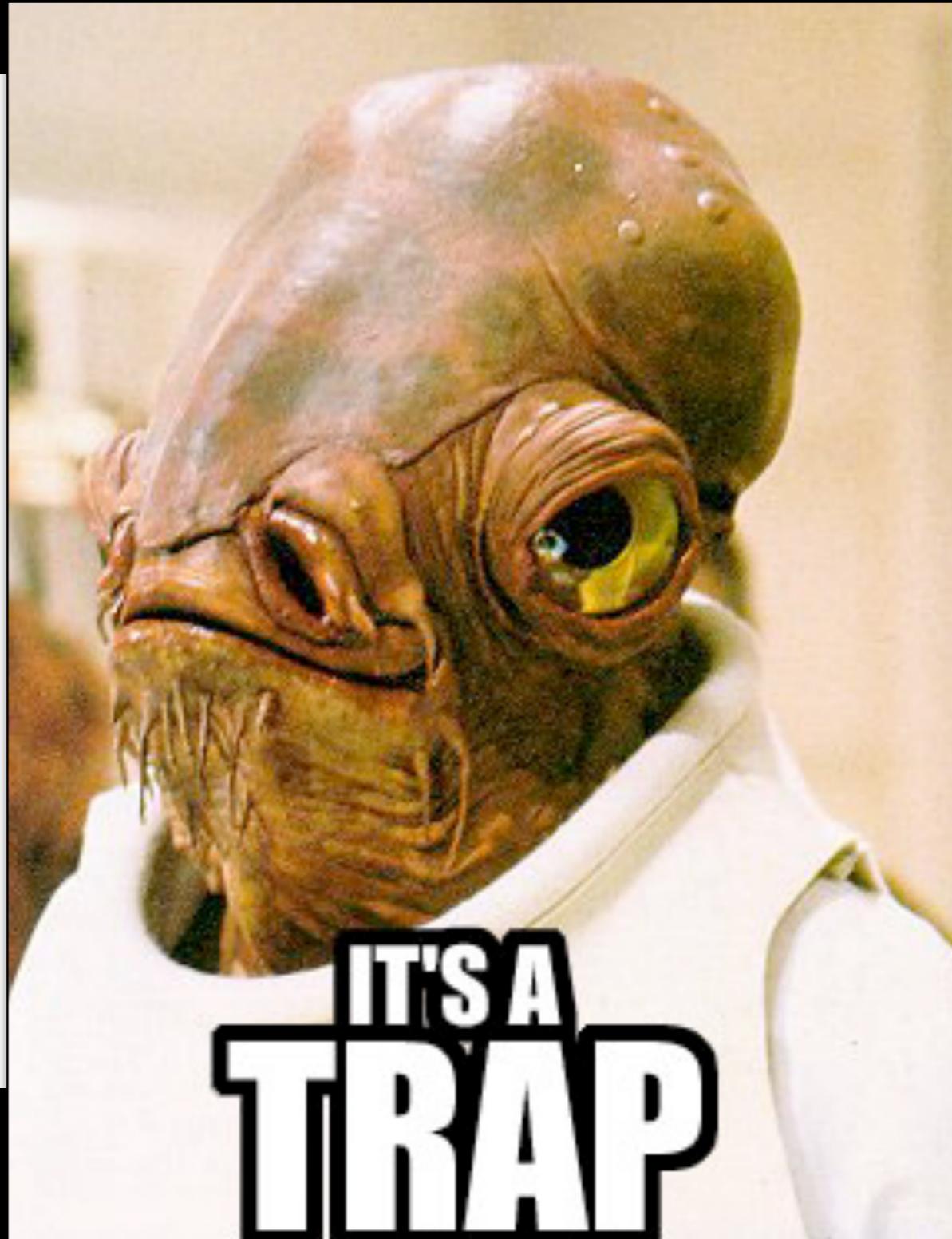




Code example



About the catches



Tips for writing Android applications (1)

Tips for writing Android applications (1)

- Don't assume anything (Phones, Tablets, TV?, desktop?)

Tips for writing Android applications (1)

- Don't assume anything (Phones, Tablets, TV?, desktop?)
- Don't assume connectivity.

Tips for writing Android applications (1)

- Don't assume anything (Phones, Tablets, TV?, desktop?)
- Don't assume connectivity.
- You're not alone. Don't hog resources.

Writing (native) Android applications (2)

Writing (native) Android applications (2)

- Do your business in separate threads.

Writing (native) Android applications (2)

- Do your business in separate threads.
- Cache EVERYTHING!

Writing (native) Android applications (2)

- Do your business in separate threads.
- Cache EVERYTHING!
- show what you can, fetch more up-to-date info in the background.



Writing (native) Android applications (2)

- Do your business in separate threads.
- Cache EVERYTHING!
- show what you can, fetch more up-to-date info in the background.
- Don't quit when something fails, display what you know.

Writing (native) Android applications (2)

Writing (native) Android applications (2)

- BC breaks are a no-go.

Writing (native) Android applications (2)

- BC breaks are a no-go.
- Make sure your portrait and landscape are working.

Writing (native) Android applications (2)

- BC breaks are a no-go.
- Make sure your portrait and landscape are working.
- Usability is an art (and I'm not an artist)

7 Sizes and usability (1)

Pixels
instead
of
Dip's





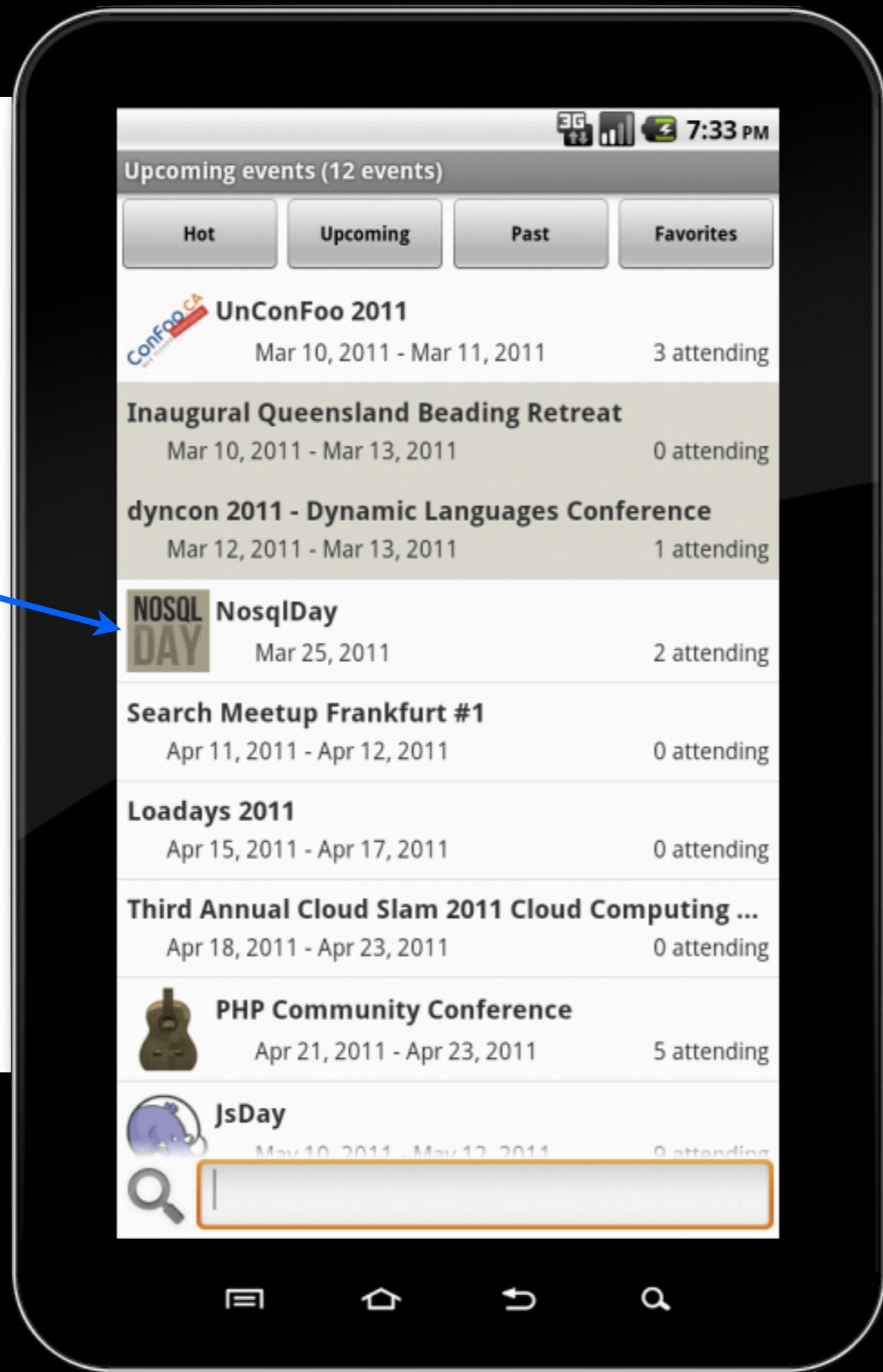
Sizes and usability (2)



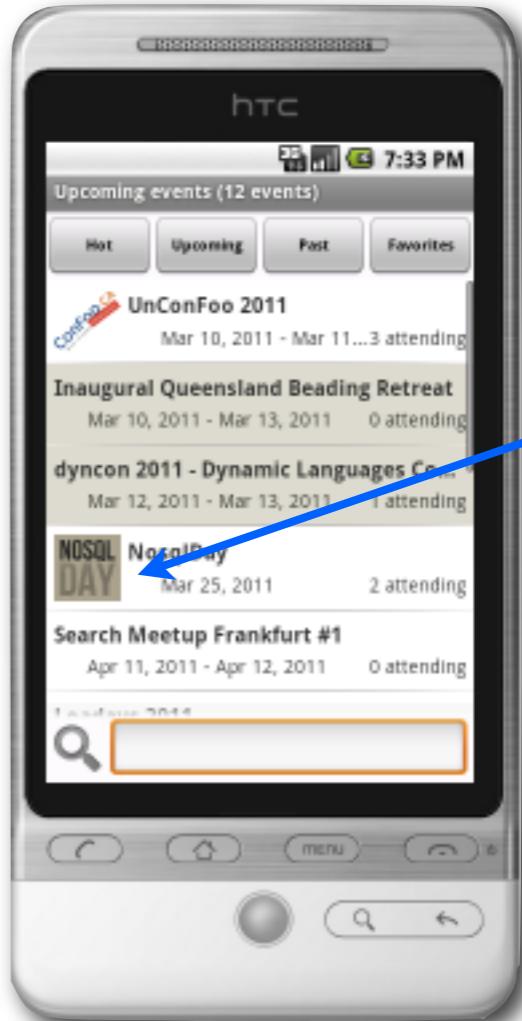


Sizes and usability (3)

Scaled
properly

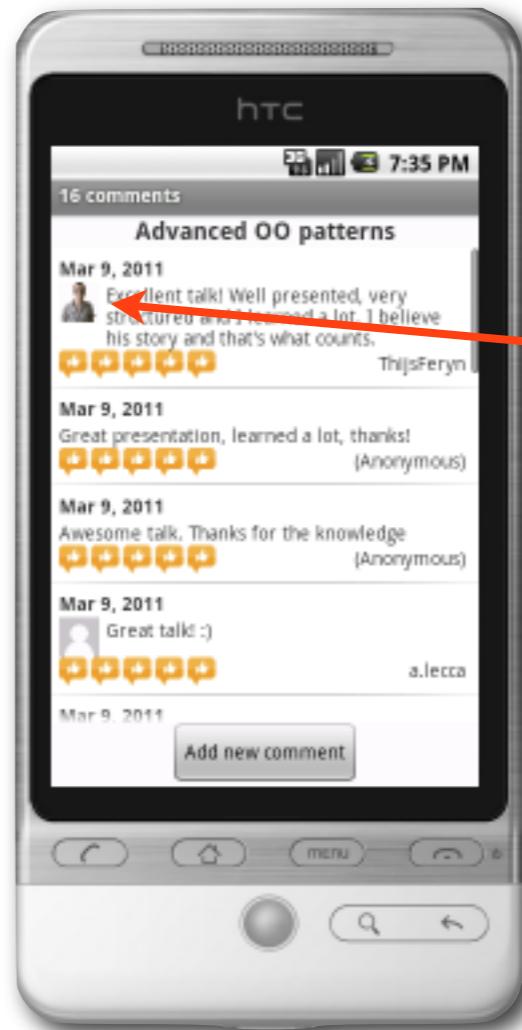
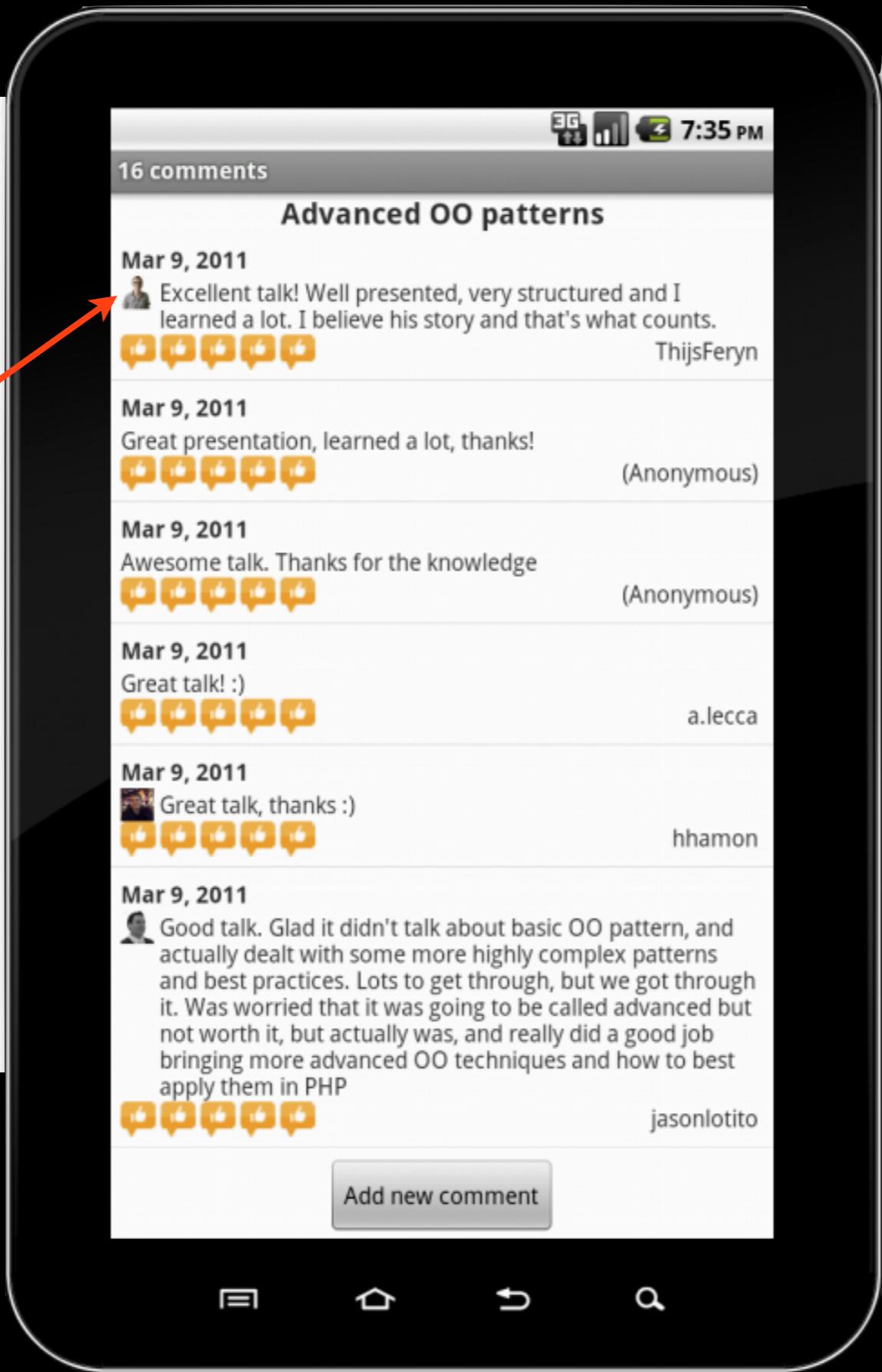


Scaled
properly



7 Sizes and usability (4)

Gravatar
image
not
scaled
properly



Here
it's
ok

7 The most useful link

developer.android.com

 And of course...

<http://joind.in/talk/view/3180>

7 Any questions?



► QUESTIONS?

http://farm1.static.flickr.com/73/163450213_18478d3aa6_d.jpg



Code Cuisine™

- ▶ THANK YOU FOR YOUR ATTENTION