Getting Started with Android

David Chandler
+Zuriel Corp. S.A.C.
http://turbomanage.com

A bit of history

- first Android device
  - Oct 2008
- Aug 2010: 200k activations per day
- Sep 2012: 1.3m activations per day
- Jun 2014: 1.5m activations per day
- over 900m devices activated
- 700,000 apps in Google Play Store
- over 25b app downloads

OS Versions

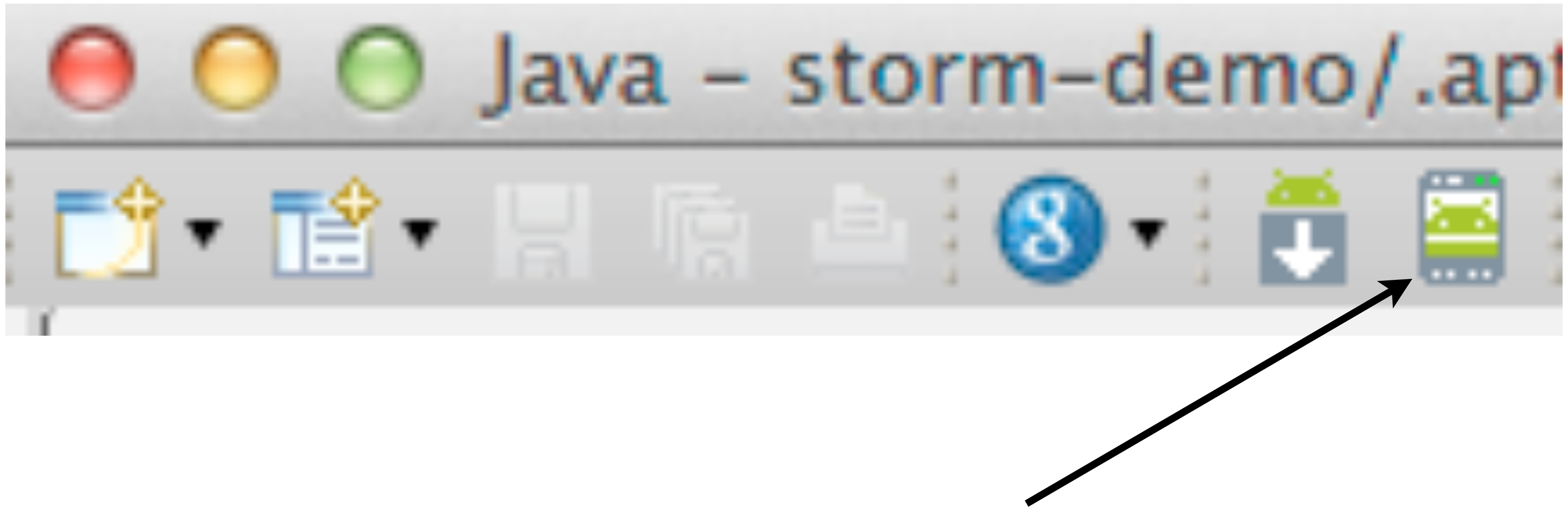
http://developer.android.com/about/dashboards/index.html

Setup

- Download Android SDK and tools
  - ADT bundle or SDK + Eclipse + plugin
  - Android Studio beta (IntelliJ)
  - or C/C++ NDK
- Connect a device
  - Set developer options on phone (4.2: tap 7x)
  - adb -d install your.apk
  - adb -d uninstall your.package.name

Setup

- Configure an emulator



- For better performance
  - Install Intel HAX (hardware acceleration) in SDK Manager, Extras
  - See Using the Emulator on d.a.c.

Key concepts

- Manifest
- Activity
- Intent
- Resources
- AsyncTask
- Service

What's in an app?

- AndroidManifest.xml
- Declares
  - targetSdk, minSdk
  - permissions
  - activities
  - intents
  - services
  - broadcast receivers
  - and more

Activities

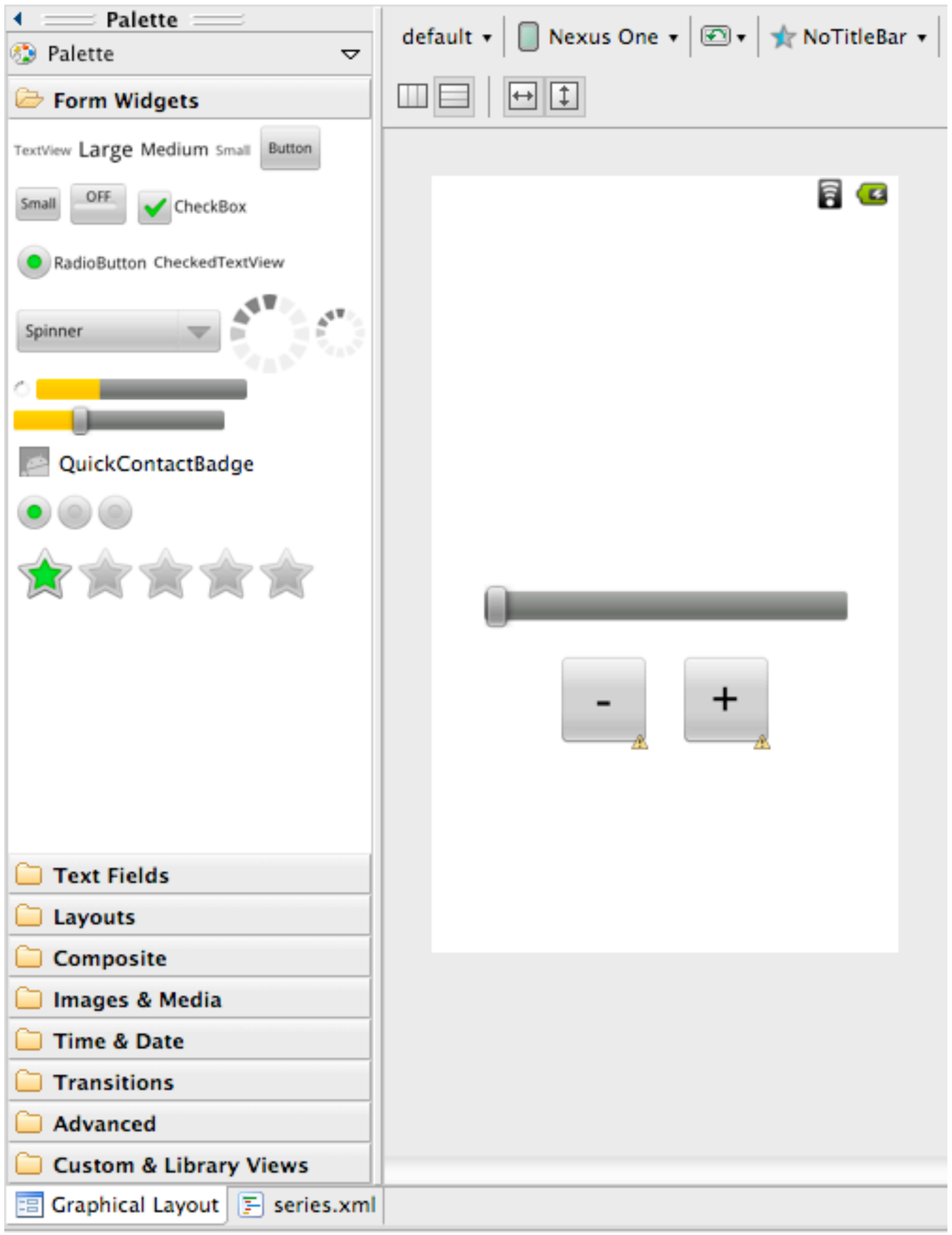
- Provides a screen (View, Fragment, ...)
- Launched via Intent filter
- Methods
  - onCreate()
  - onPause()
  - onResume()
  - onCreateOptionsMenu()
  - to name a few

Resources

- res/
- drawable
- layout
- values
  - strings.xml
  - styles.xml
- Get compiled to R class

Layouts

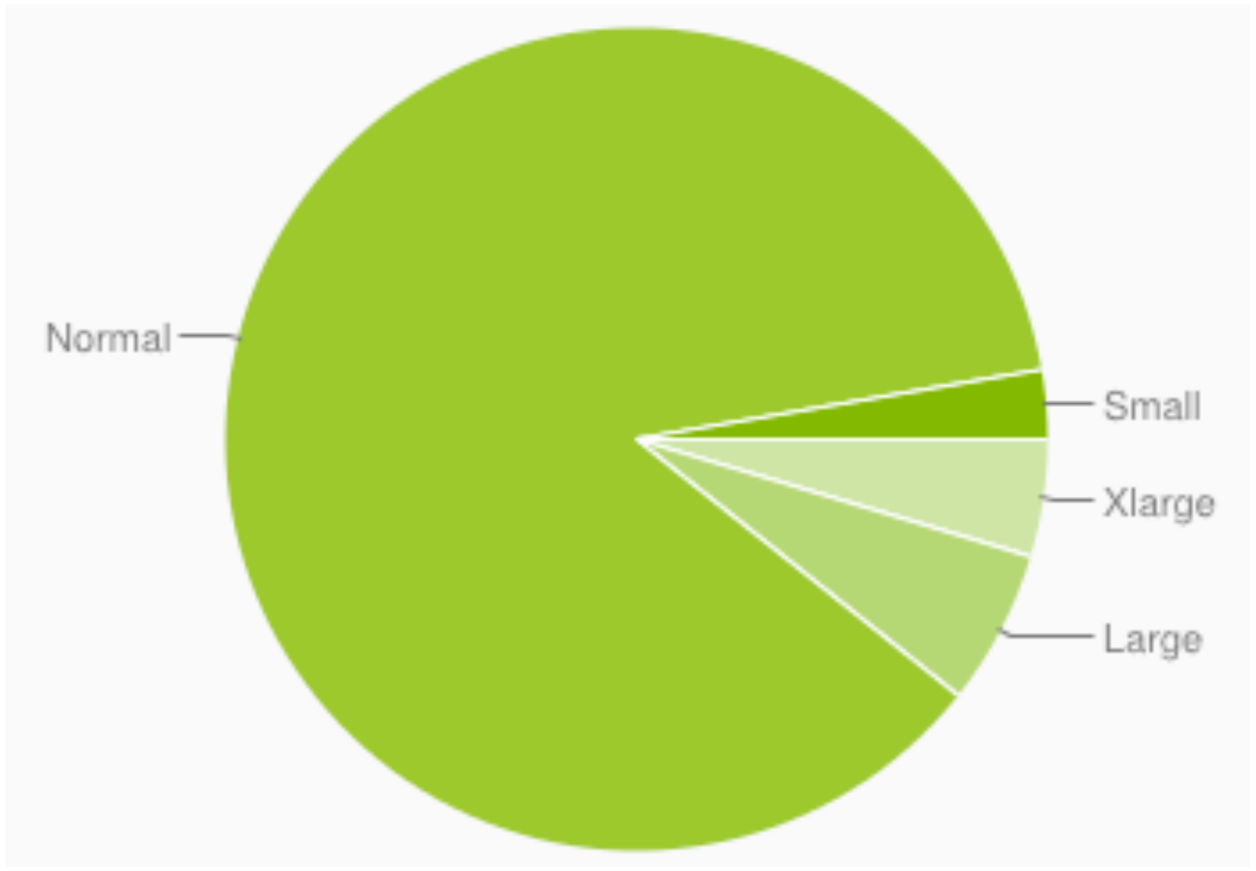
- XML

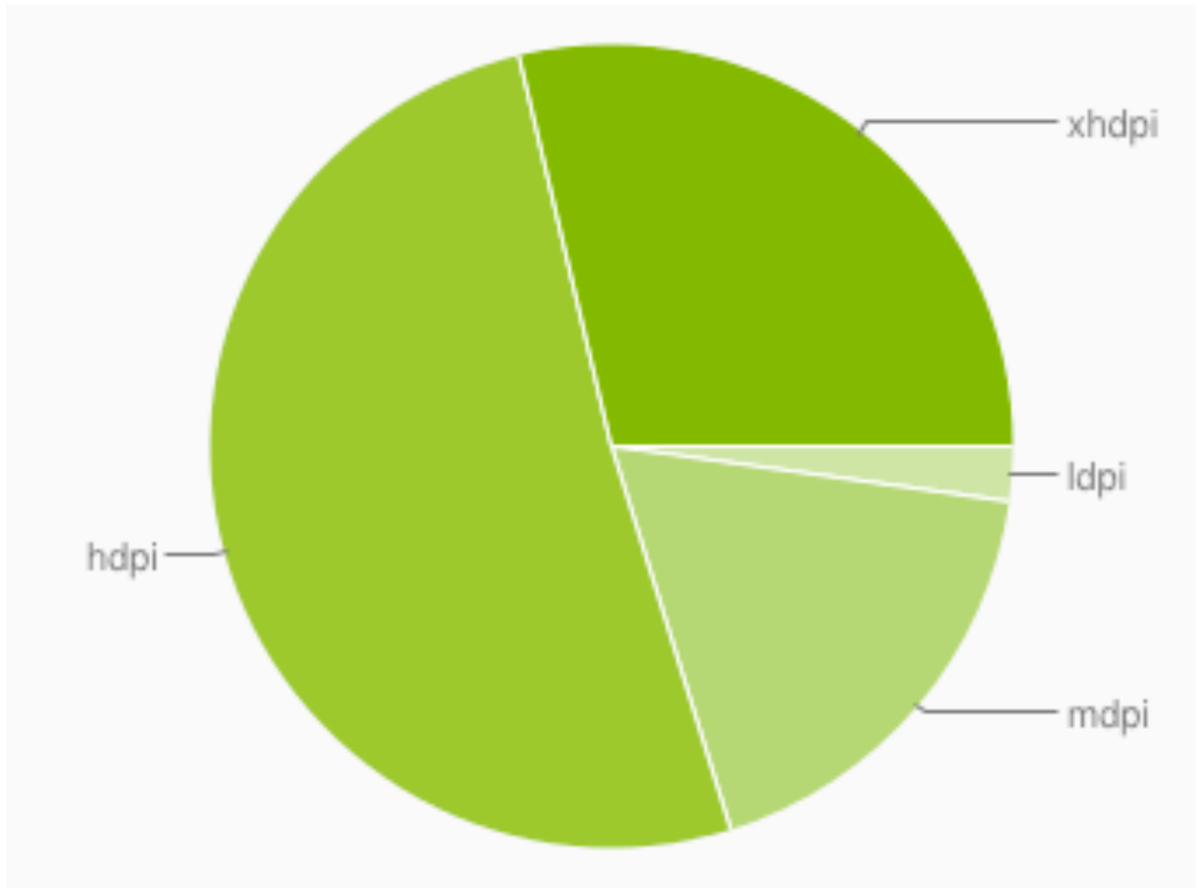

Palette
Palette
Form Widgets
TextView Large Medium Small Button
Small OFF CheckBox
RadioButton CheckedTextView
Spinner
QuickContactBadge
Text Fields
Layouts
Composite
Images & Media
Time & Date
Transitions
Advanced
Custom & Library Views
Graphical Layout
series.xml
default
Nexus One
NoTitleBar
-
+

Lab time

- Demo
  - Create application
  - About the support library
  - Graphical tools & property editor
- Go to https://github.com/turbomanage/language-helper-arequipa/wiki/Lab
  - Do labs 1, 2, 3
  - 20 min

One app, many screens





http://developer.android.com/about/dashboards/index.html

Resource selectors

- -ldpi, -mdpi, -hdpi, -xhdpi, -tvdpi
- -large, -sw600, -port / -land
- -v11
- combined: layout-large-land
- to accommodate light / dark themes
  - drawable-(xh/h/m)dpi
  - drawable-(xh/h/m)dpi-v11

Resources in code

- Every XML element needs an id
- @+id/my_id
- Reference in code with
  - findViewById(R.id.my_id)
  - or setContentView(R.layout.some_id)
- Easy to set ID in properties panel
- Beware the two R classes!
  - yourapp.R..., android.R...

What's a Fragment?

- Like Activity, but only controls part of the screen
- Way to reuse code between single-pane or dual-pane layouts (phone and tablet, for example)
- Similar lifecycle methods
  - you should save state just like Activity
  - or setRetainInstance(true);

Working with ListView

- Very common view class
- Shows any type of data
- An Adapter binds data to a row layout in getView()
- ListView + ListAdapter = ListActivity (Fragment)
- ExpandableListView
  + ExpandableListAdapter
  = ExpandableListActivity (Fragment)

Lab time

- Demo
  - Import existing Android code
  - Ctrl+click to follow ID (MainActivity)
  - Demo findViewById (code right to left)
  - ExpandableListAdapter.getChildView()
  - properties editor, auto-complete
- Lab 4 Run Language Helper (5 min)
- Lab 5 Work with a ListView (15 min)

Styles

- res/values/styles.xml
- Use in layouts
- inheritance
- standard styles
  - @color, @android:color
  - @style, @android:style
  - extract style
  - extract string

Icons

- drawables/
- easy way
  - New | Android Icon Set
- see also Android Asset Studio
  - with device frame generator!

Lab time

- Demo
  - Ctrl+N New...
  - Ctrl+/ Toggle comment
  - Ctrl+Shift+R Open Resource
  - Ctrl+Shift+T Open Type
- Labs 6, 7, 8 (15 min)

Intents

- Message for activating other components or apps
- Standard Intents let you launch maps, browser, etc.
- Lets the user choose an app
- Your app can handle Intents
  - set Intent filters in AndroidManifest.xml
  - for example, browser Intent for your site

WebView

- browser in a box
- build your app in HTML5
- many popular apps use
  - Gmail message view
  - wikipedia, news sites (WSJ)
- generally less performant for games
- watch out on older versions
- <u>recently unbundled</u>

WebView pro tips

- If you have lots of images, watch out for this memory leak pre-JB
- Workaround: instantiate WebView programatically instead of layout XML
- Follow Android design guidelines

Lab time

- Demo
  - Ctrl+O outline view
  - bug in case statement
- Lab 9 (15 min)

stORM

- Extend DatabaseHelper, annotate with @Database
- Annotate POJOs with @Entity
- Generates
  - DbFactory
  - EntityTable
  - EntityDao
- new EntityDao().insert/get/query...

src

storm-gen.googlecode.com

Lab time

- Demo
  - create new @Entity in stORM
  - show generated code
  - new Dao() pattern
  - discuss APT config (bug in ADT 23)
- Lab 10 (15 min)

Making HTTP calls

- Apache HttpClient
- HttpUrlConnection
- google-api-java-client
- google-http-java-client
- basic-http-client

Synchronous API

```java
// Example code to login to App Engine dev server
public void loginDev(String userEmail) {
    BasicHttpClient httpClient = new BasicHttpClient("http://localhost:8888");
    ParameterMap params = httpClient.newParams()
            .add("continue", "/")
            .add("email", userEmail)
            .add("action", "Log In");
    httpClient.addHeader("name", "value");
    httpClient.setConnectionTimeout(2000);
    HttpResponse httpResponse = httpClient.post("/_ah/login", params);
}
```

Two truths of Android

- Activities die
  - on rotate
  - whenever the OS feels like it
- If you tie up the UI thread, users will hate you
  - use a ThreadExecutor or AsyncTask
  - for longer running stuff, use a Service

Asynchronous API

```
// Example code to login to App Engine dev server off UI thread
AndroidHttpClient httpClient =
         new AndroidHttpClient("http://192.168.1.1:8888");
httpClient.setMaxRetries(5);
ParameterMap params = httpClient.newParams()
        .add("continue", "/")
        .add("email", "test@example.com")
        .add("action", "Log In");

httpClient.post("/_ah/login", params, new AsyncCallback() {
    @Override
    public void onSuccess(HttpResponse httpResponse) {
        System.out.println(httpResponse.getBodyAsString());
    }
    @Override
    public void onError(Exception e) {
        e.printStackTrace();
    }
});
```

Stay off the UI thread

- Use AsyncTask
  - easy, but...
  - beware orientation change
- Use a Service
  - keeps running
  - can be used by other apps
  - IntentService is easy

Getting data with HTTP

- REST + JSON
- Frameworks that can help
  - Spring Android RestTemplate
  - Jersey -- works on App Engine
- Google Cloud Endpoints
  - RESTful service, handles auth
  - GPE tooling generates client/server code

Web authentication

- Can use Google accounts on phone
- Automatically authenticate to Google APIs
- Old way: AccountManager
  - see Cloud Tasks IO 12
- New way: Google Play Services
  - also OAuth2 to Google APIs
  - see Calendar Preview Sample

src

basic-http-client.googlecode.com

Architecture

- Model – View – Presenter (MVP)
  - decouples business logic from view
  - facilitates testing with JUnit
  - http://fernandocejas.com/2014/09/03/architecting-android-the-clean-way/
- Event bus (Otto)
- Dependency injection (Dagger)
- Fragments or not?
  - http://corner.squareup.com

So many possibilities

- Sensors
- Widgets
- Services
- Notifications
- Content providers
- Broadcast receivers
- Quick Search Box integration
- Live folders / wallpaper / daydreams

Resources

- developer.android.com
  - Training
  - Blog
- Common Tasks
- Google I/O sessions
- +Android Developers
  - Pro tips
  - DevBytes

Freebie: adb back up

```
adb backup -apk -all -nosystem -f ~/mybackupfile.ab

adb restore ~/mybackupfile.ab
```

Android Debug Bridge

- adb logcat
- adb shell
- adb shell dumpsys meminfo <pkg>
- adb kill-server :-(
- other command lines
  - android (launches SDK manager)
  - hierarchyviewer
  - emulator @avd_name (see ~/.android/avd)