



# Getting Started with Android

# First things



- Download Android SDK and tools
  - ADT plugin for Eclipse
  - or C/C++ NDK
- Connect a device
  - Set developer options on phone
  - `adb -d install your.apk`
  - `adb -d uninstall your.package.name`

# More adb goodness



- `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`)

# Key concepts

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



# One app, many screens



- <http://developer.android.com/about/dashboards/index.html>

# 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

# 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



# Resources

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



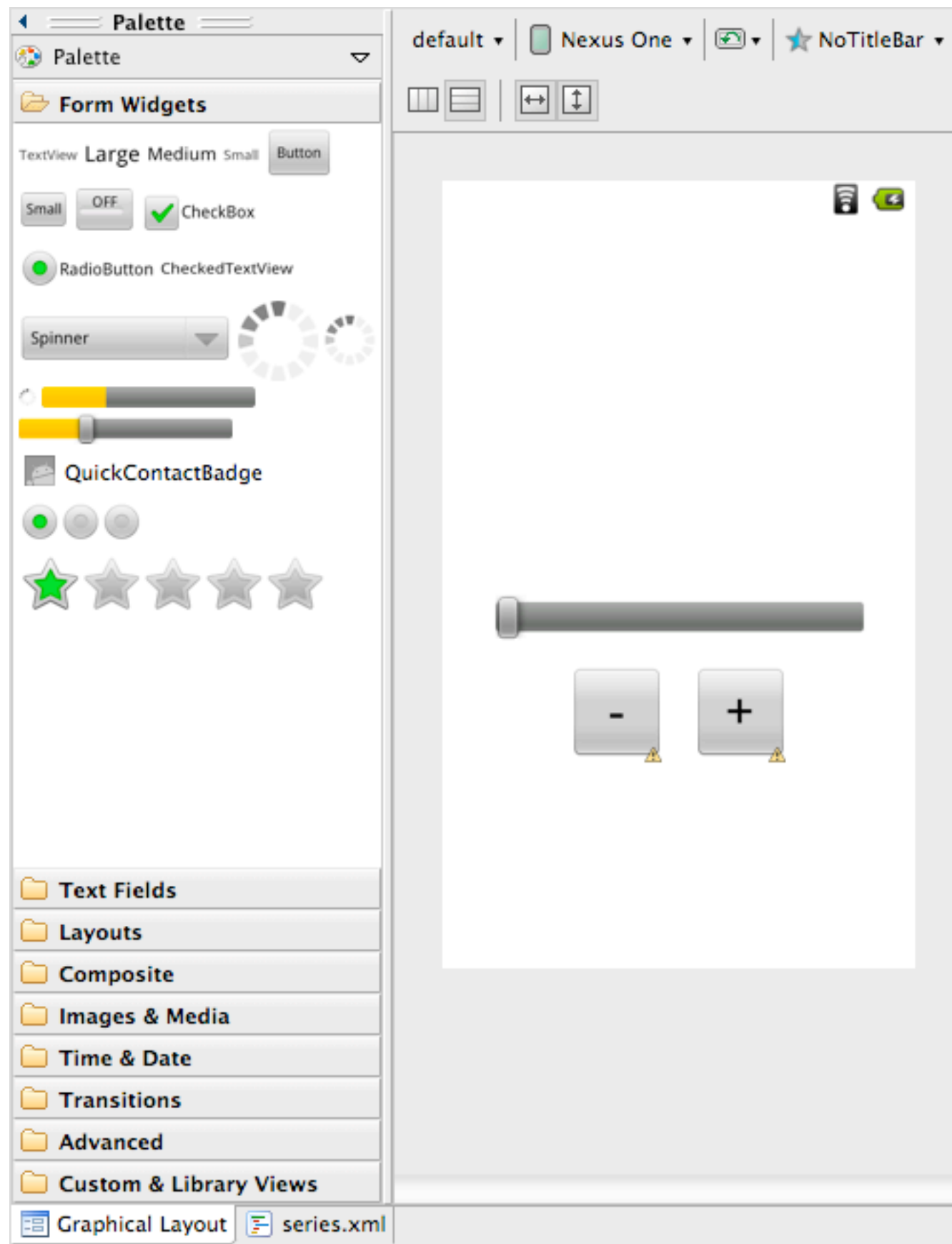
# Resource selectors



- -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

# Layouts

- XML



# Icons

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



# Stay off UI thread



- Use AsyncTask
  - easy, but...
  - cancelled on orientation change
- Use a Service
  - keeps running
  - can be used by other apps

# So many possibilities



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

# WebView pro tips



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

# Web RPC



- RequestFactory?
- 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



# Authentication



- Can use Google accounts on phone
- Old way: AccountManager
  - see Cloud Tasks IO 12
- New way: Google Play Services
  - also OAuth2 to Google APIs
  - see [Calendar Preview Sample](#)

# Making HTTP calls



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

# Resources

- [developer.android.com](http://developer.android.com)
- Common Tasks
- Google I/O sessions
- +Android Developers

