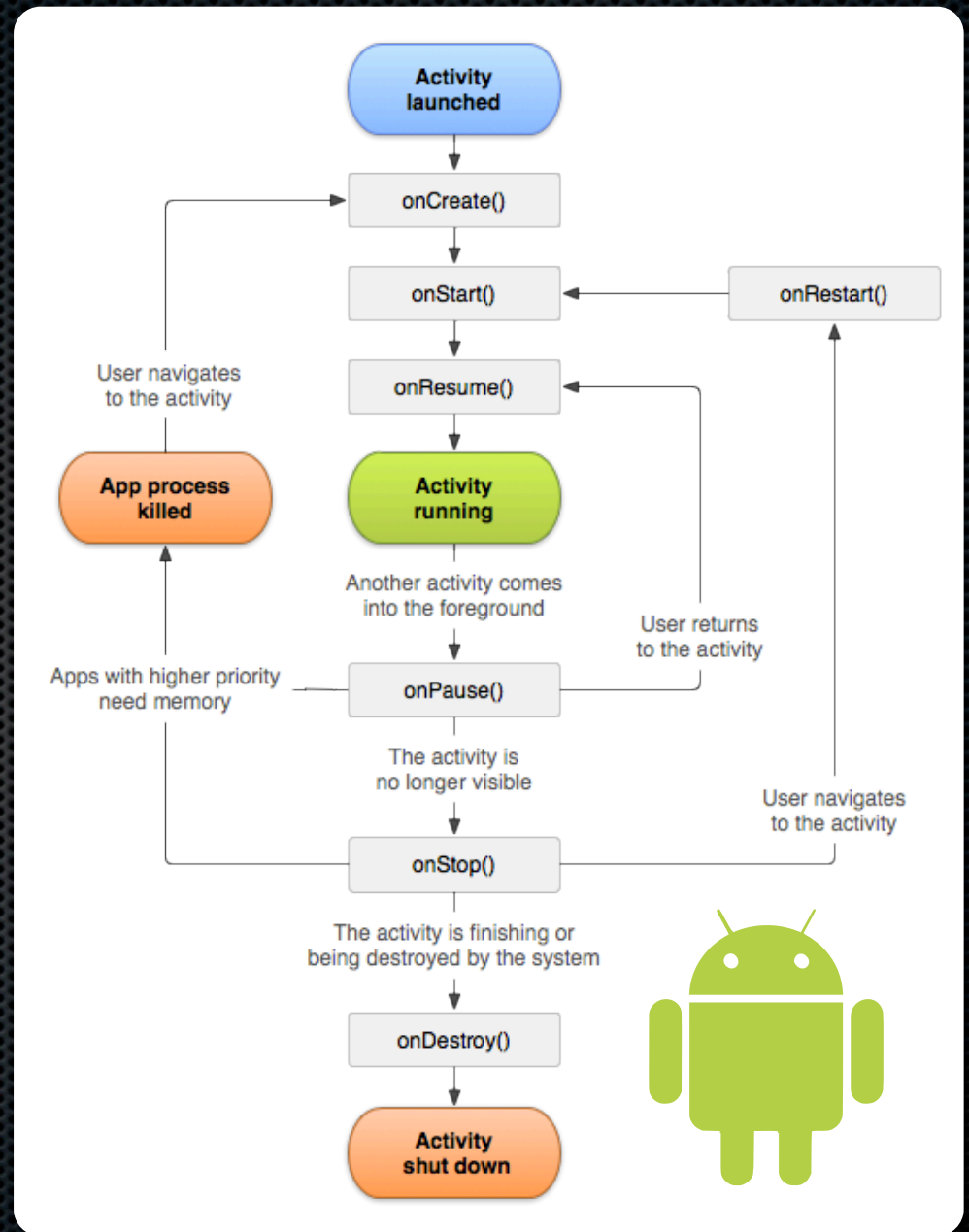


Mobile Application Programming: Android Activities

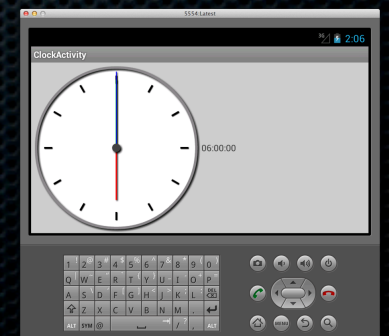
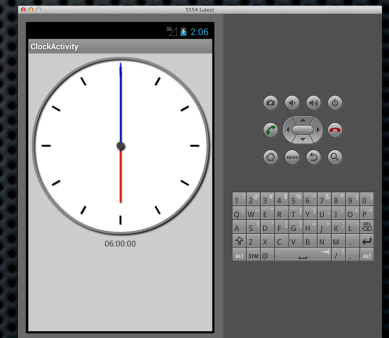
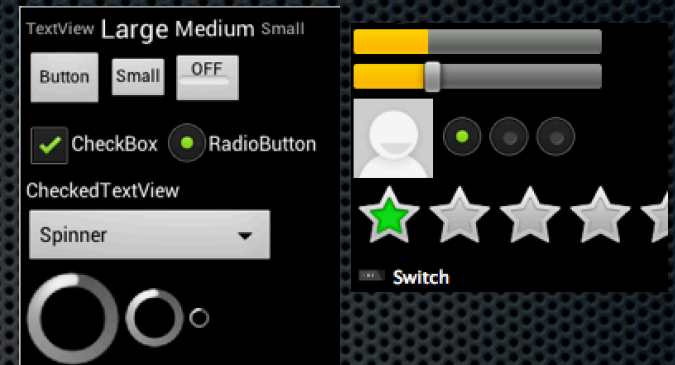
Activities

- ❖ Apps are **composed of activities**
- ❖ Activities are self-contained tasks made up of **one screen-full** of information
- ❖ Activities **start one another** and are **destroyed commonly**
- ❖ Apps can **use activities belonging to another app**



Custom Controls

- ✦ Create subclass of View class
- ✦ Override:
 - ✦ `onDraw(Canvas c)`
 - ✦ `onMeasure(int wMeasure, int hMeasure)`
- ✦ Add listener interface and listener property for the interesting events the control generates and call `on...` methods when events occur
- ✦ Implement view-level transient data saving



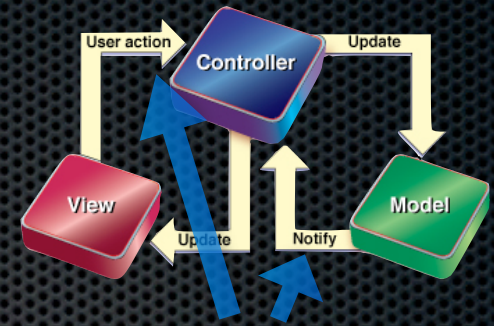
Application



Application Controller (MVC)

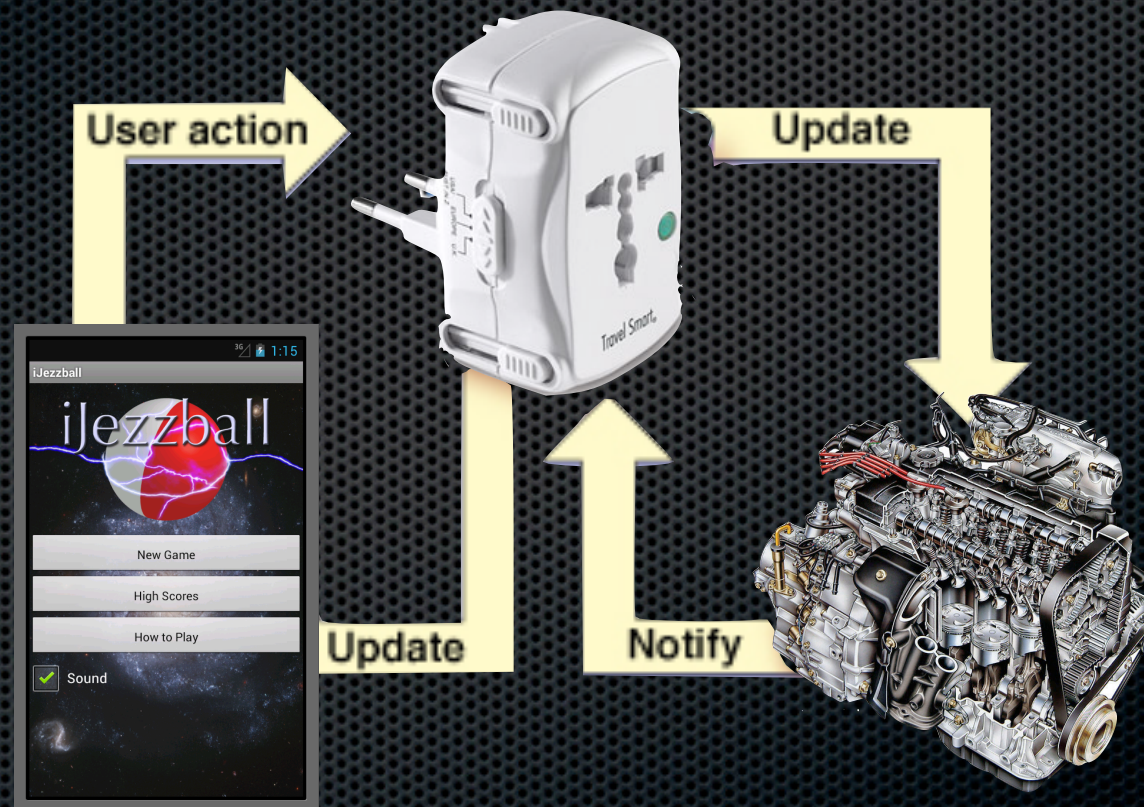


Notification Options

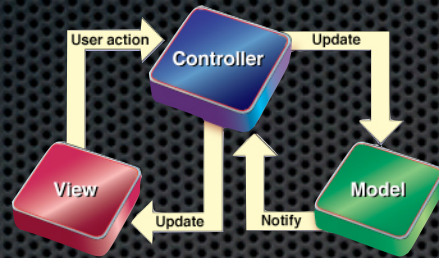


- ✦ **Listeners**
 - ✦ Declare listener **interface**
 - ✦ Add **getter/setter** for listener
 - ✦ **Call methods** on listener when appropriate
- ✦ **Interfaces** like listeners but more complex (eg. **ListAdapter**)
- ✦ **LocalBroadcastManager** (Android Support Package)
 - ✦ Call **registerReceiver**(BroadcastReceiver r, IntentFilter i)
 - ✦ Send using **sendBroadcast**(Intent i)

Model View Adapter (MVA)



The Controller



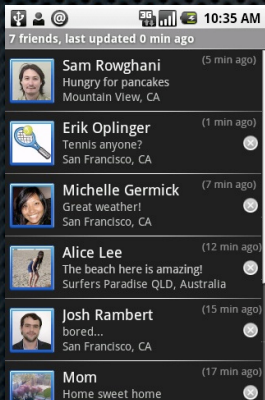
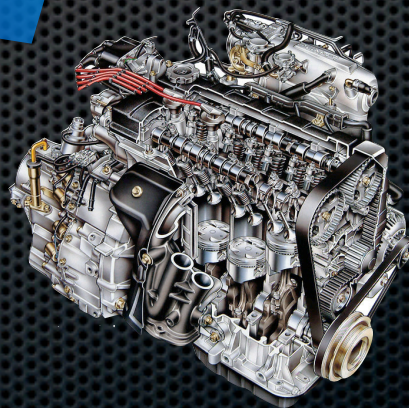
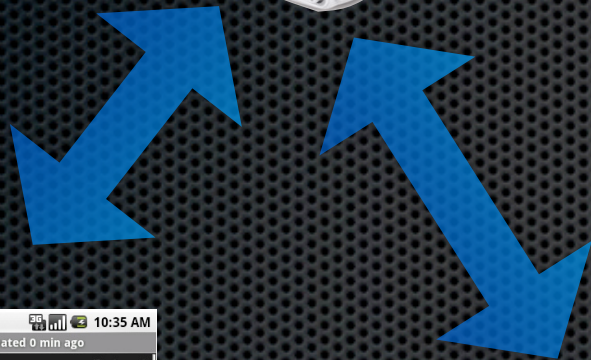
- ✦ Acts as an **adapter** between model and view
- ✦ **Keeps the model in sync** with changes in the view (user interaction), and **the view in sync** with changes to the model (network access, etc.)
- ✦ Is the **least portable** part of the program
- ✦ Typically implemented using the **Activity** class

Typical Activity Tasks

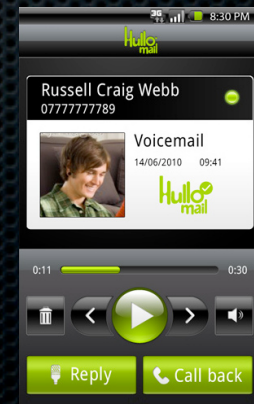
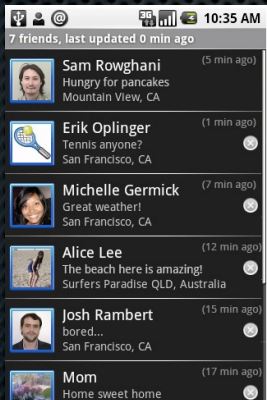
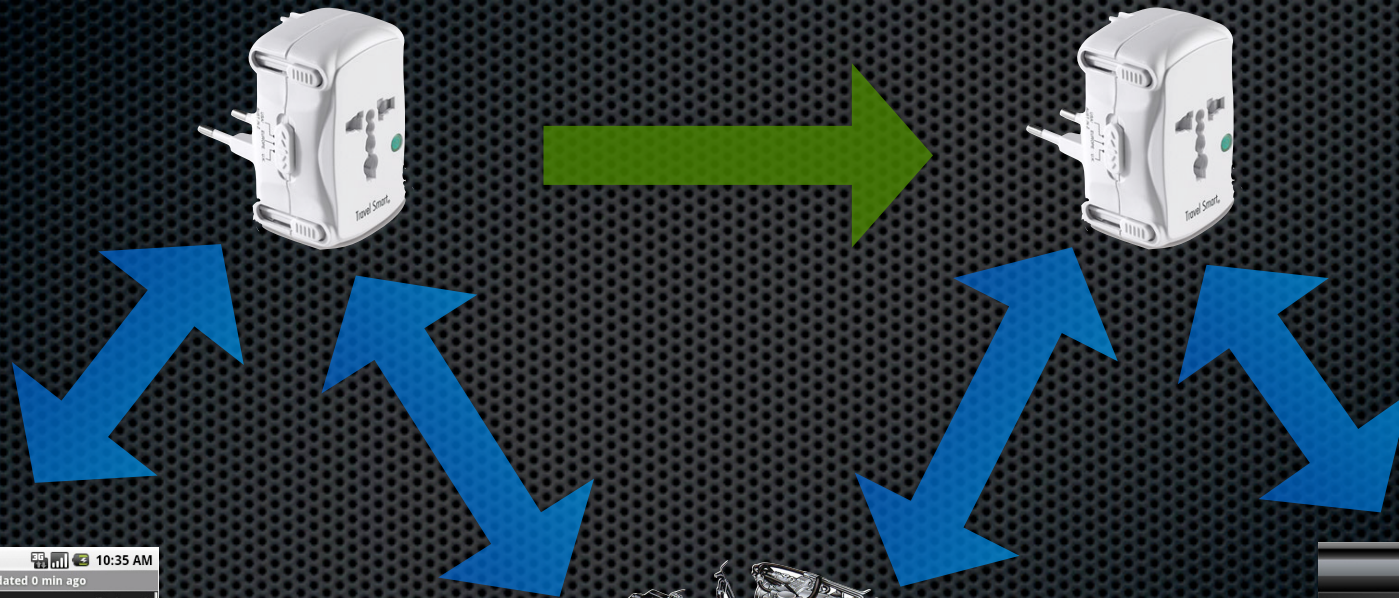


- ✦ **onCreate**
 - ✦ Create data model or **load from file**
 - ✦ Create UI using data provided by model
- ✦ **onPause**
 - ✦ Save data model **to file**
- ✦ **Listeners** / Interfaces / Broadcast Responses
 - ✦ Update data model in response to user interactions
 - ✦ Update UI in response to model changes
 - ✦ **Launch other activities** when required

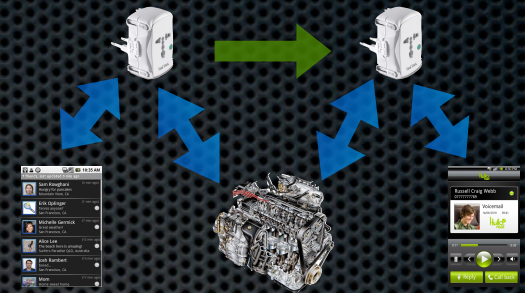
Launching Activities



Launching Activities



Launching Activities



- ✦ Create **Intent** instance
- ✦ Set intent **class** property
- ✦ Fill in intent **extras** using `putExtra(key, value)`
 - ✦ Get keys from activity-to-be-launched class
- ✦ Call `Activity.startActivity(Intent)`
- ✦ Can open activities **from other apps** similarly as well