

Lab 12

Simple RESTful App

NetDB

CS, NTHU,
Fall, 2013

Hello Smail!

Outline

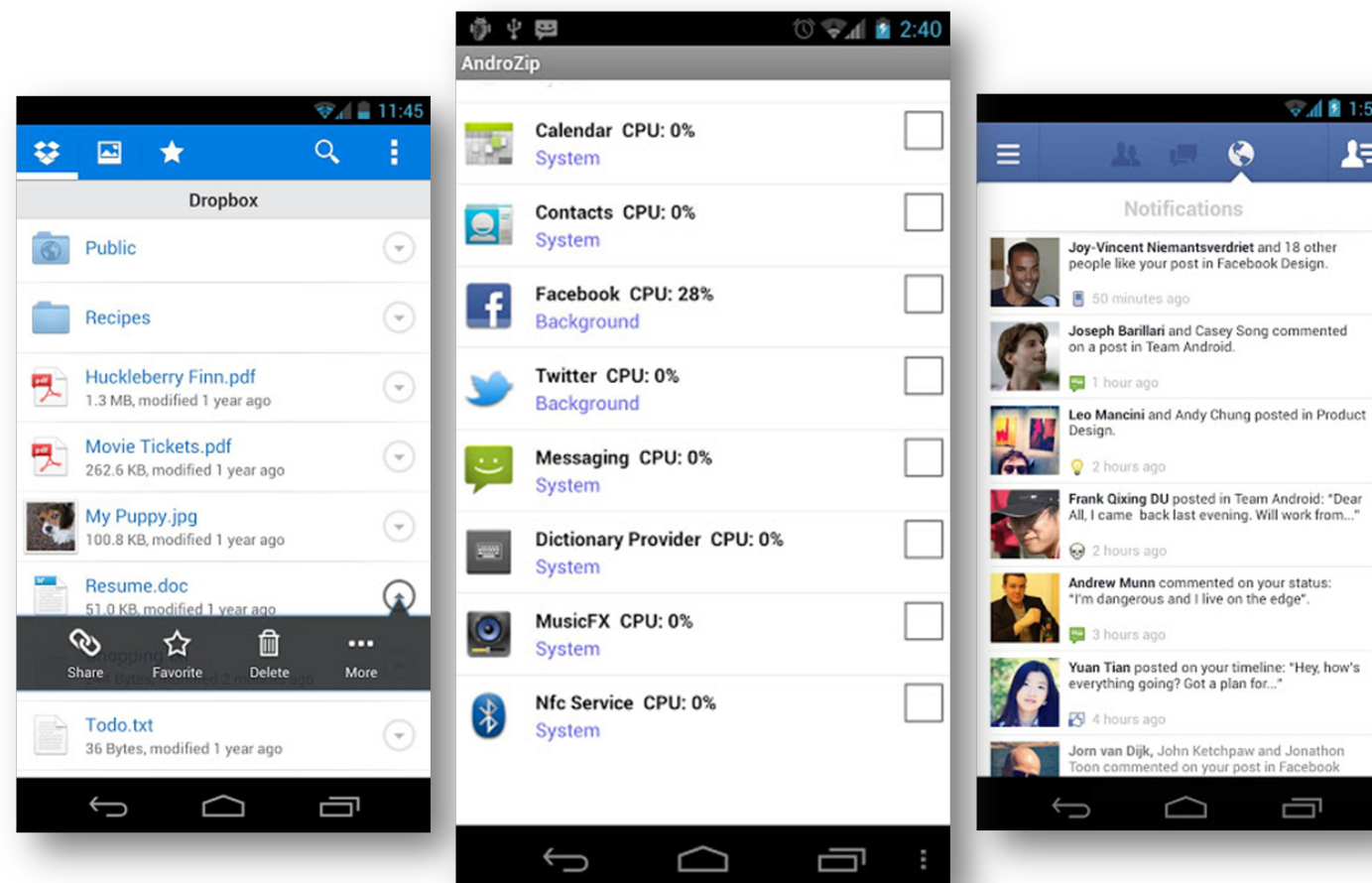
- ListView and Adapter
- Get some REST
- Today's Mission

Outline

- ListView and Adapter
- Get some REST
- Today's Mission

ListView

- Display the content in a scrollable list



Default ListView

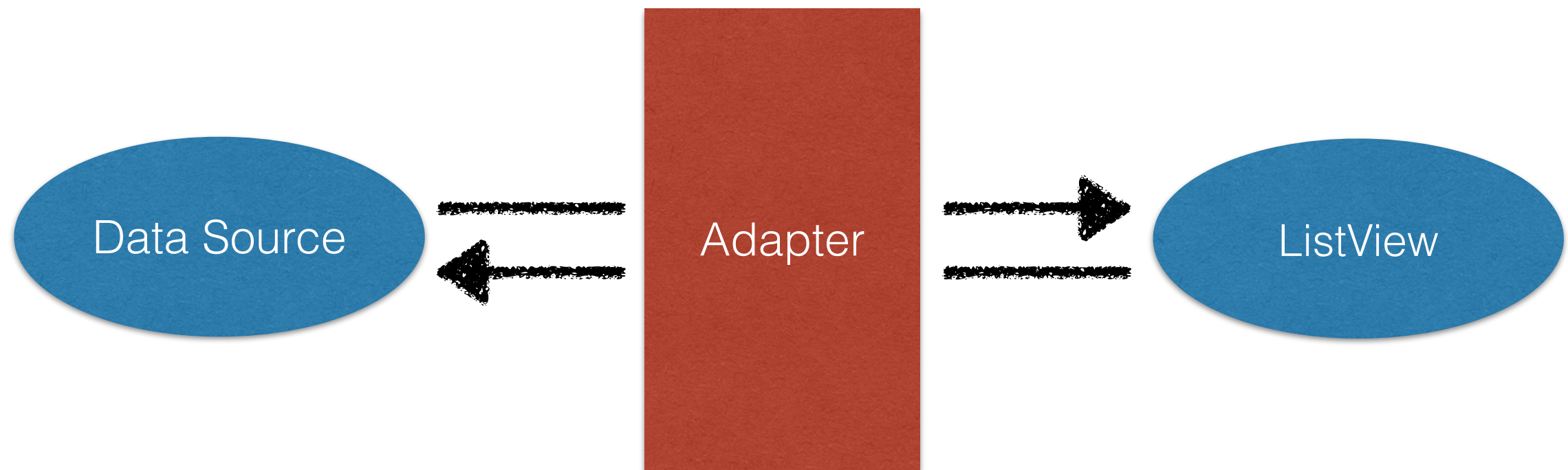
- `android.R.layout.simple_list_item_1`
 - Single text line
- `android.R.layout.simple_list_item_2`
 - One title and one subtitle
- `android.R.layout.simple_list_item_single_choice`
- `android.R.layout.simple_list_item_multiple_choice`
- `android.R.layout.simple_list_item_checked`

Custom ListView

- Define a layout under res/layout for the items of your list
- Get your layout by `R.layout.${file name}`

Fill the Content of ListView

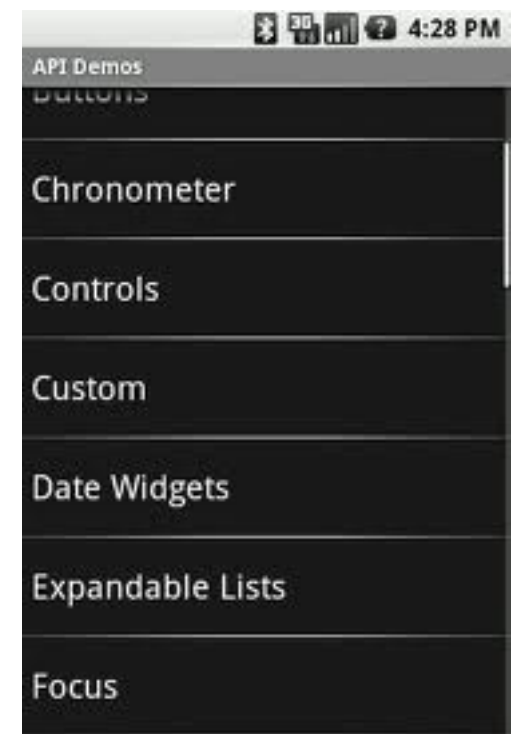
- In android, we use an [Adapter](#) as the bridge between the data source and the list view



Adapter

- ArrayAdapter
 - Map an array of data to views defined in an XML file
- Updatable

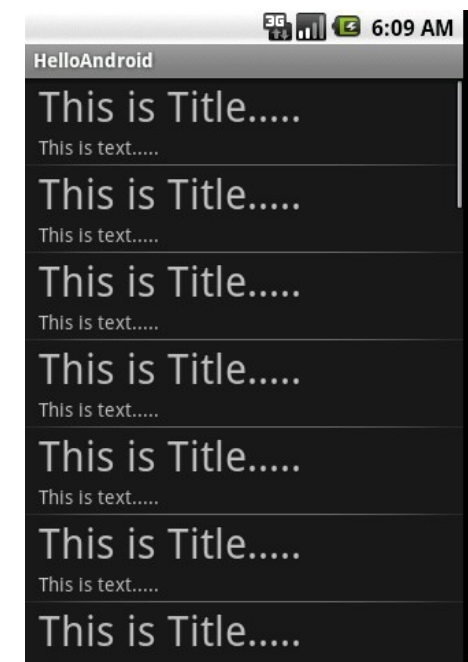
```
String[] data = {"row1", "row2", "row3"};  
  
ArrayAdapter adapter = new ArrayAdapter(this,  
    android.R.layout.simple_list_item_1, data);  
  
listView.setAdapter(adapter);
```



Adapter

- SimpleAdapter
 - Map **static** data to views defined in an XML file.
 - You can specify the data backing the list as an ArrayList of Maps. Each entry in the ArrayList corresponds to one row in the list. The Maps contain the data for each row

```
ArrayList<HashMap<String,String>> list;  
...  
listView.setAdapter(new SimpleAdapter(this, list  
    android.R.layout.simple_list_item_2,  
    new String[] { "name", "phone" },  
    new int[] { android.R.id.text1,  
                android.R.id.text2 }  
));
```



Adapter

- BaseAdapter
 - Abstract class. When you want to get more control of the adapter, create your own adapter which extends BaseAdapter
- SimpleCursorAdapter
 - Use content provider

Update List View

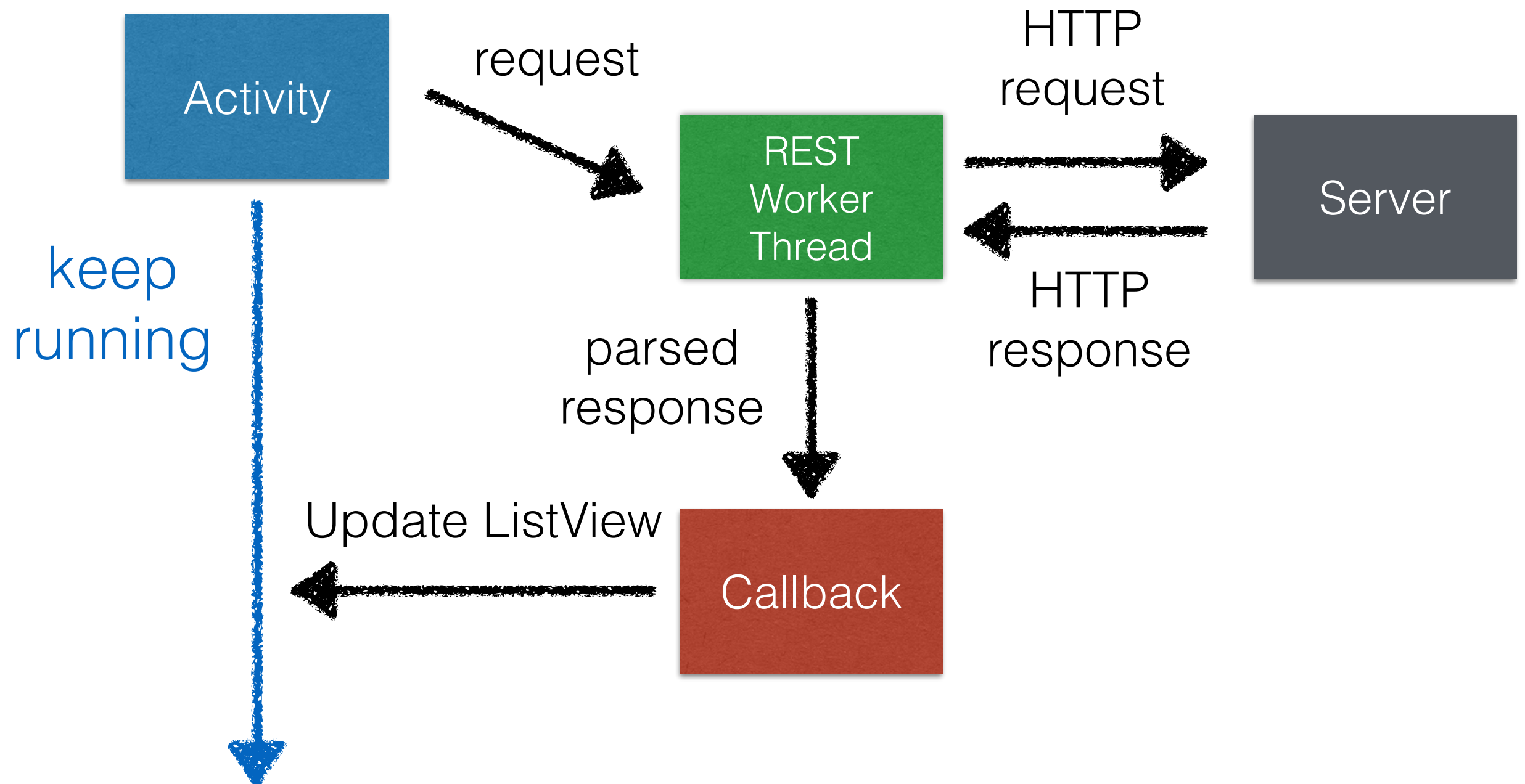
- How can we update the list view when the data is changed ?
 - You can use `adapter.notifyDataSetChanged()` to notify the Views reflecting the data set to refresh itself
 - Note that you can only update the UI in the UI Thread
 - Use Activity's Handler or `runOnUiThread` as a workaround

```
YourActivity.this.runOnUiThread(new Runnable() {  
    @Override  
    public void run() {  
        adapter.notifyDataSetChanged();  
    }  
}).start();
```

Outline

- ListView and Adapter
- Get some REST
- Today's Mission

Get some REST



Restful API

	URL	Parameter	Description
GET	http://netdb-smail2.appspot.com/emails	sort="stamp" dir="desc"	Get mails from server
POST	http://netdb-smail2.appspot.com/emails	Json object represent the Mail	Create new mail

Better Architecture

- The example in today's lab is a simple approach to demonstrate the REST implementation in Android
- For better architecture, please watch the 2010 Google I/O session
- <http://www.google.com/events/io/2010/sessions/developing-RESTful-android-apps.html>

Outline

- ListView and Adapter
- Get some REST
- Today's Mission

Today's Mission

- Trace the simple Smail app
 - The web version is over [here](#)
- Raise your hand after you understand the code
 - TAs will ask you some questions