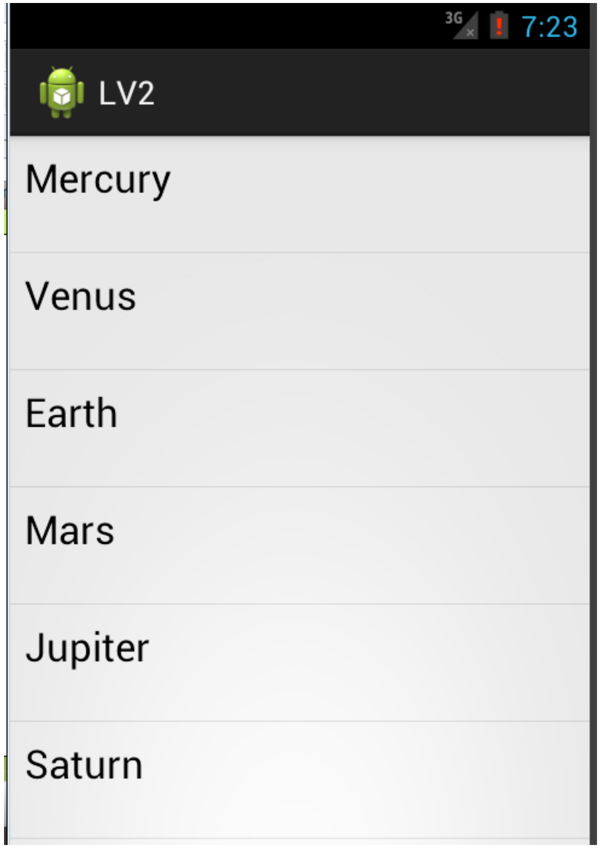
ListView
and
ListActivity

Topics

- **ListView and ListActivity**
- **Adapters**
  - Built in
  - Custom
- **Sorting**
- **Listeners**
- **Optimization**
  - ConvertView
  - Viewholder pattern

ListView and ListActivity

- Common data pattern
- Scrolling list of data
- MVC design pattern
  - Model – the data
  - View – UI
  - Controller – Logic



Mercury

Venus

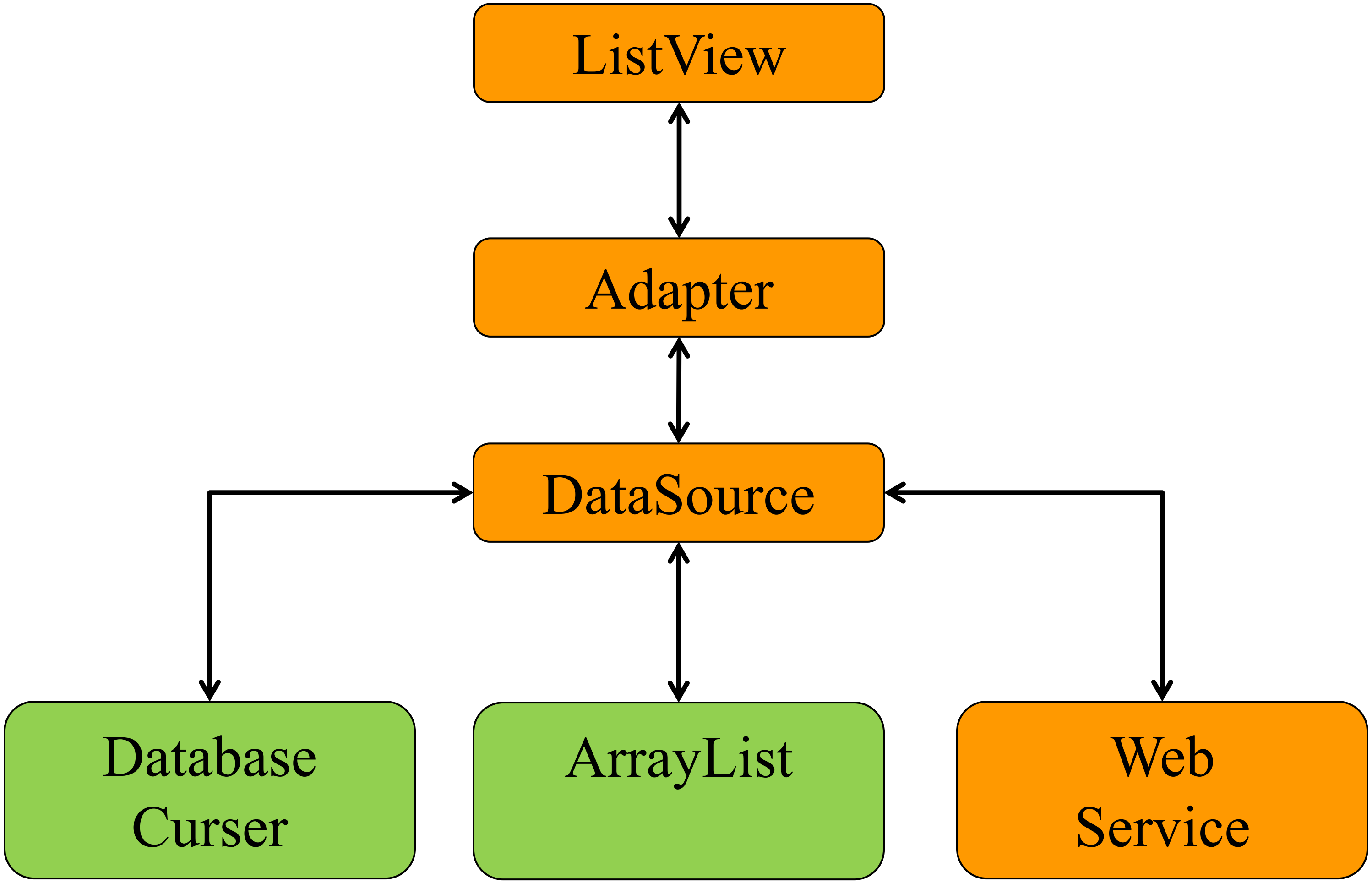
Earth

Mars

Jupiter

Saturn

ListView and ListActivity Overview



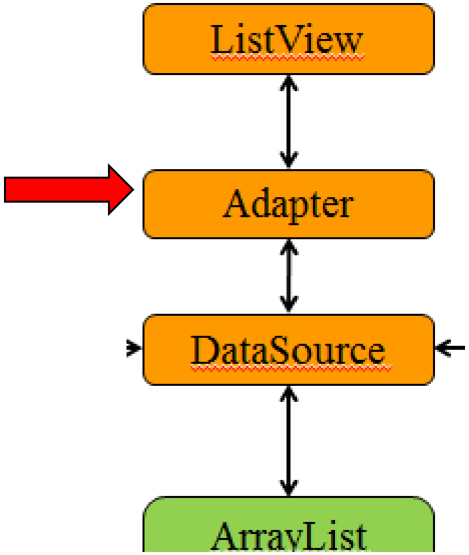
ListView and ListActivity

- **To create;**
  - Either add a listview to your layout
    - If you use defaults its very easy
    - See LV1
  - Or derive your activity from ListActivity
    - Simplest way, the activity is the listview
    - See LV2

- **To access data use adapters**

Adapters

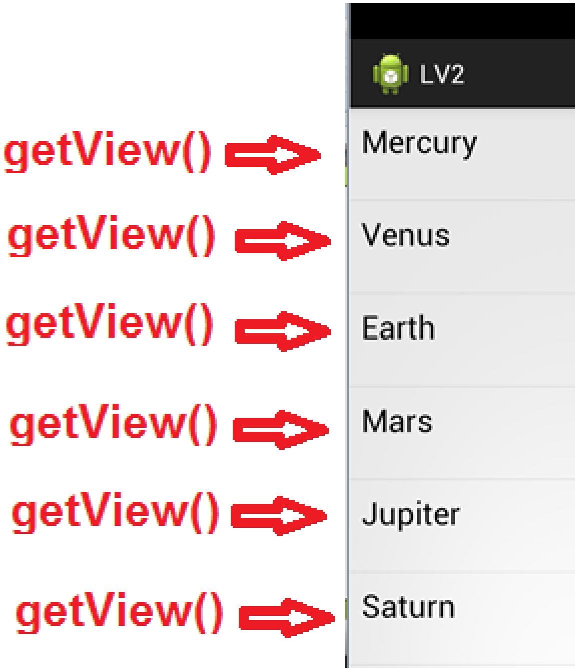
- **Manages datasource for a view**
- **Consistent access protocol**
- **Easy datasource swapping**
- **Used by a lots of things**
  - ListView
  - Spinner
  - Contacts list



- **Standard and Custom**

Standard Adapters

- Handles standard data sources
- getView() – simple text string
- Easy to use
- Aren’t very flexible



```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    //notice no setcontent view no R.Layout.main needed

    String[] values = new String[] { "Mercury", "Venus", "Earth",
              "Mars", "Jupiter", "Saturn", "Uranus", "Neptune",
              "Ceres", "Pluto","Haumea","Makemake", "Eris" };

    // Second parameter - Layout for the row
    // Third parameter - ID of the TextView to which the data is written
    ArrayAdapter<String> adapter = new ArrayAdapter<String>(this,
              android.R.layout.simple_list_item_activated_2, android.R.id.text1, values);

    setListAdapter(adapter);
}
```

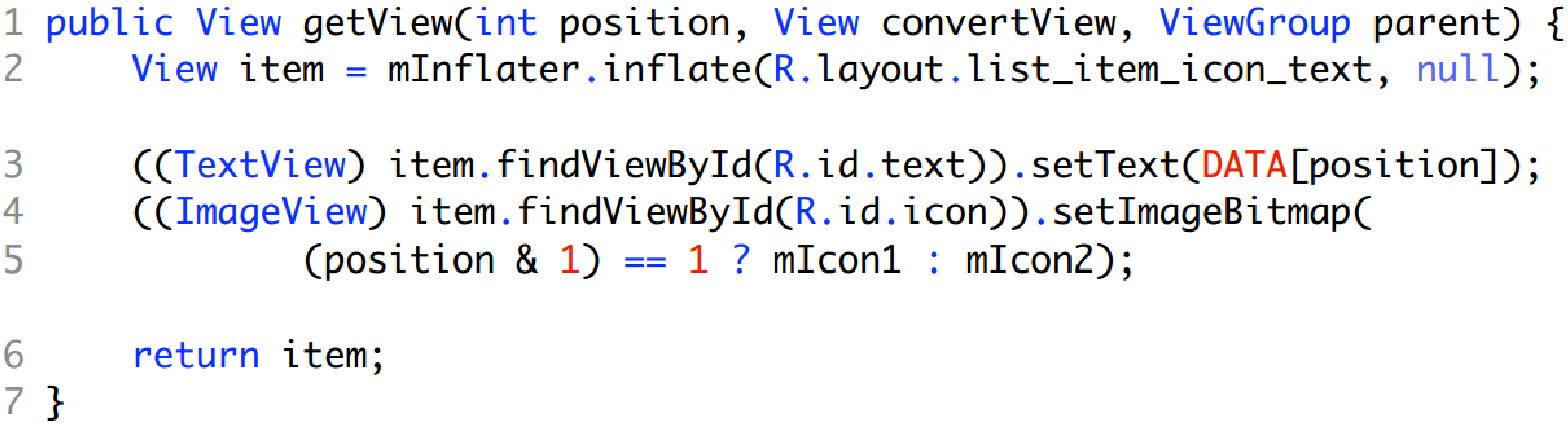
Custom Adapters

- getView() can generate any kind of row
- Can handle any kind of datasource
- The catch is you have to do all the work yourself
- See LV3

Custom Adapter - Recipe

- **Define datasource**
- **Define what each row in the list should look like**
  - Add layout in Res\layout (listview_row_layout.xml)
- **CustomAdapter - Extend BaseAdapter and fill in overridden methods**
  - getView() most important
  - Called for each row in ListView
  - Use logic and data from list to populate above layout for each row
- **Instantiate CustomAdapter in Activities onCreate and then set it as ListViews adapter (setAdapter(myCustomAdapter))**

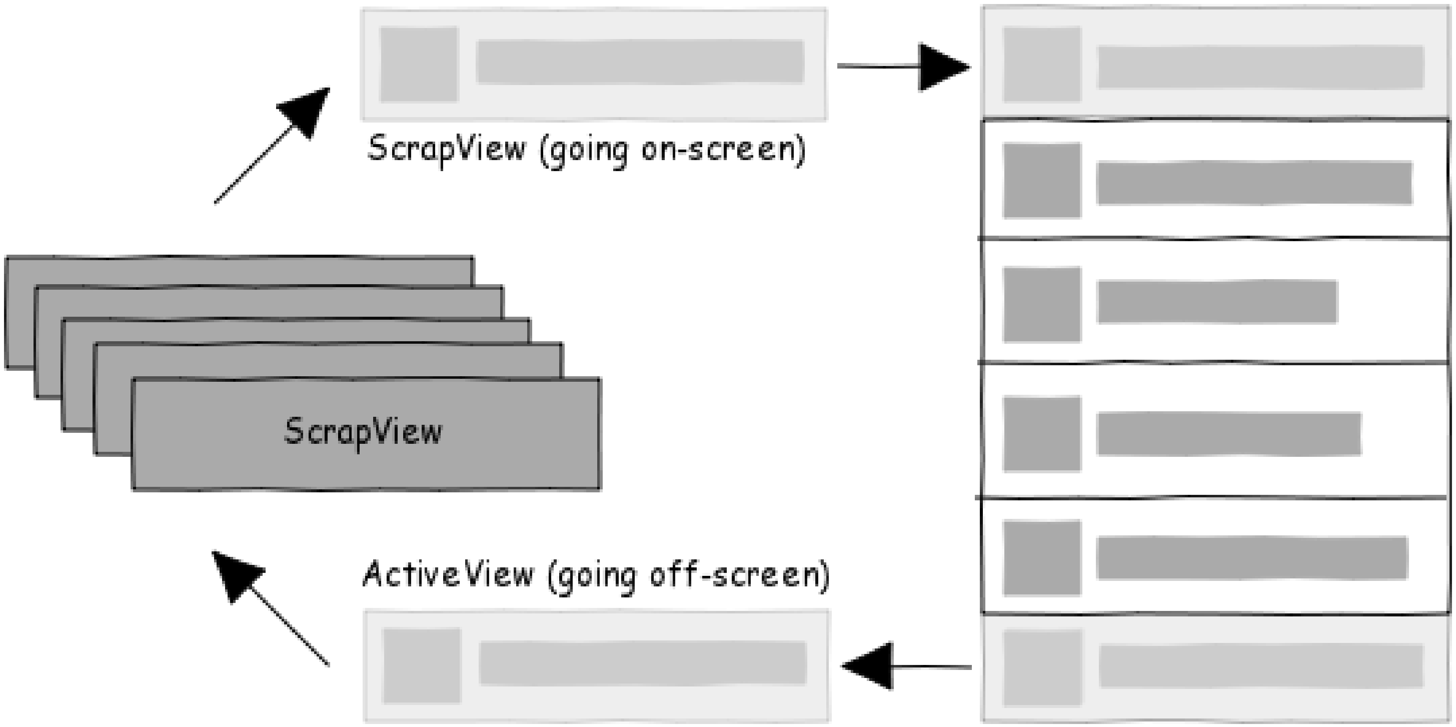
Adapter.getView()
The slow way



```
public View getView(int position, View convertView, ViewGroup parent) {
    View item = mInflater.inflate(R.layout.list_item_icon_text, null);

    ((TextView) item.findViewById(R.id.text)).setText(DATA[position]);
    ((ImageView) item.findViewById(R.id.icon)).setImageBitmap(
            (position & 1) == 1 ? mIcon1 : mIcon2);

    return item;
}
```

ListView

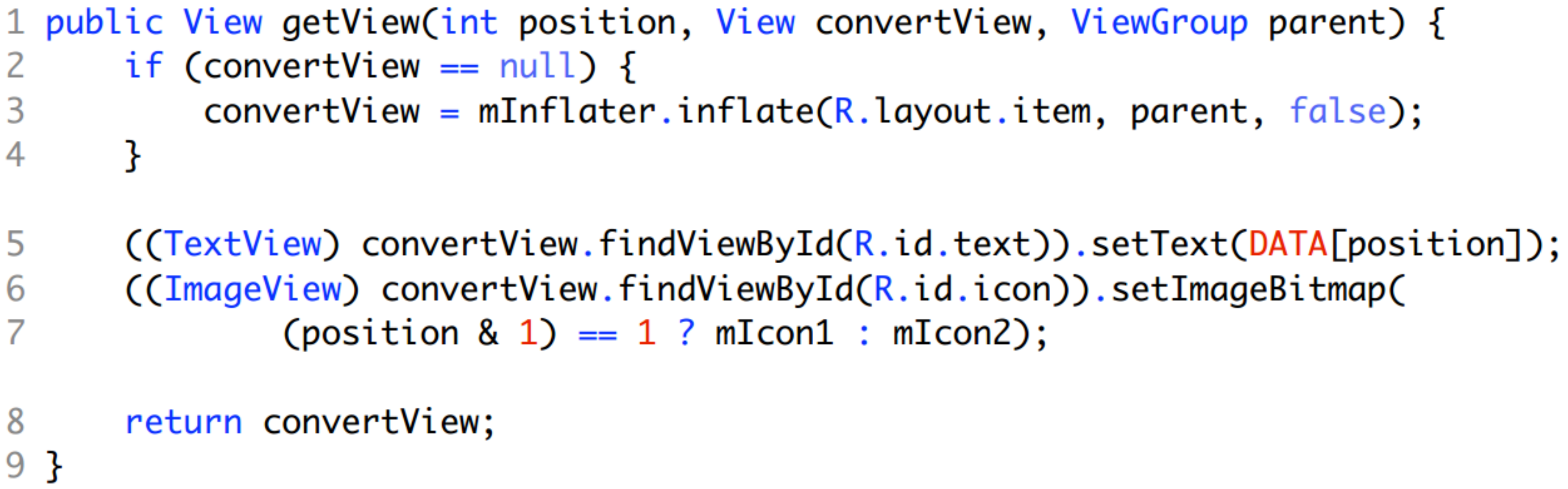
- *It tries to do as few view inflations as possible.*
- *It only paints and lays out children that are (or are about to become) visible on screen*$^{\underline{c}}$



Optimizations (ConvertView)

- Object instantiation expensive
- List could have 1000s of items
- Only a few visible at a time
- Ones that “roll off” screen useless
- Android recycles these via “convertview”
- part of Adapters getView() method
- Complete object with stale data
- Load new data and insert it in list
- Saves lots of time

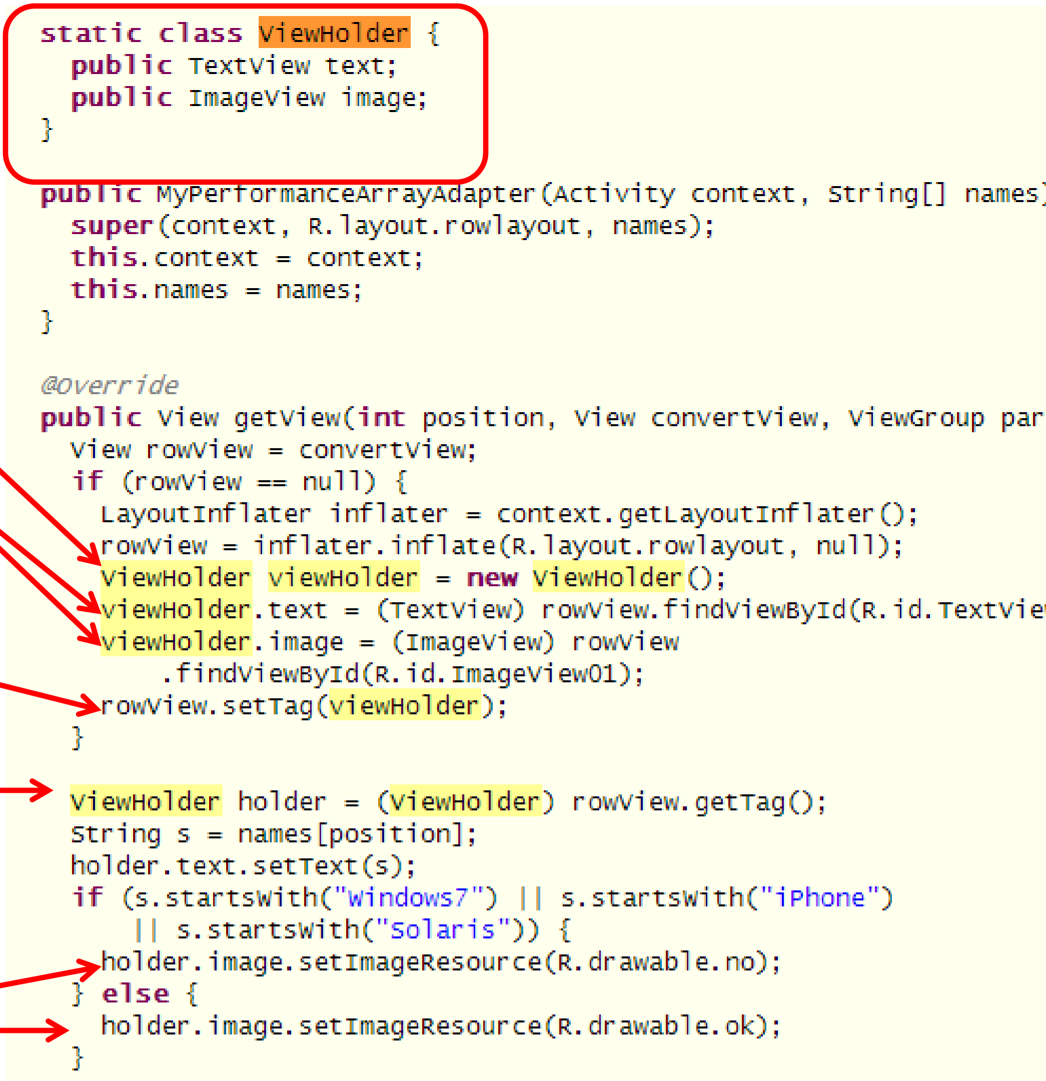
Adapter.getView()
A faster way



```java
public View getView(int position, View convertView, ViewGroup parent) {
    if (convertView == null) {
        convertView = mInflater.inflate(R.layout.item, parent, false);
    }

    ((TextView) convertView.findViewById(R.id.text)).setText(DATA[position]);
    ((ImageView) convertView.findViewById(R.id.icon)).setImageBitmap(
            (position & 1) == 1 ? mIcon1 : mIcon2);

    return convertView;
}
```

Adapter.getView()
The fastest (Viewholder pattern)

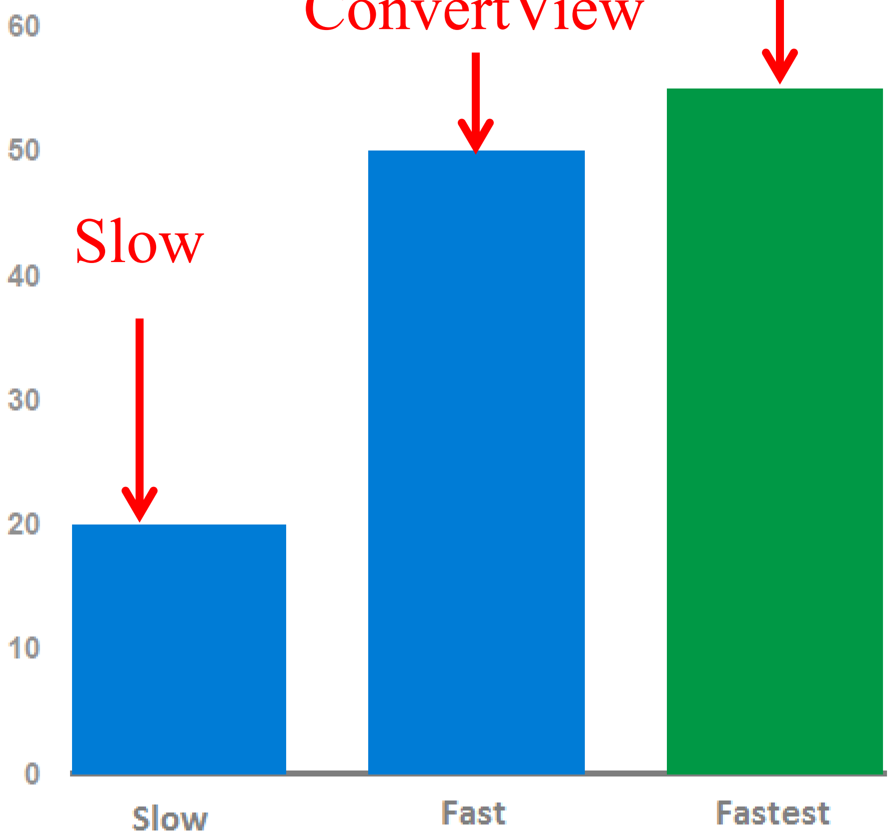
- **Avoids repeated calls to findViewById()**
- **Create a holder class that holds pointers to views of interest**
- **Set views to holder class members**
- **Save viewHolder to convertview tag**
- **From then on get the viewholder**
- **And access the views without calling findViewByID()**



```
static class ViewHolder {
  public TextView text;
  public ImageView image;
}

public MyPerformanceArrayAdapter(Activity context, String[] names)
  super(context, R.layout.rowlayout, names);
  this.context = context;
  this.names = names;
}

@Override
public View getView(int position, View convertView, ViewGroup par
  View rowView = convertView;
  if (rowView == null) {
    LayoutInflater inflater = context.getLayoutInflater();
    rowView = inflater.inflate(R.layout.rowlayout, null);
    ViewHolder viewHolder = new ViewHolder();
    viewHolder.text = (TextView) rowView.findViewById(R.id.TextVie
    viewHolder.image = (ImageView) rowView
        .findViewById(R.id.ImageView01);
    rowView.setTag(viewHolder);
  }

  ViewHolder holder = (ViewHolder) rowView.getTag();
  String s = names[position];
  holder.text.setText(s);
  if (s.startsWith("Windows7") || s.startsWith("iPhone")
      || s.startsWith("Solaris")) {
    holder.image.setImageResource(R.drawable.no);
  } else {
    holder.image.setImageResource(R.drawable.ok);
  }
```

See http://www.vogella.com/articles/AndroidListView/article.html#adapterown_updates

getView() speed test



ConvertView and ViewHolder

ConvertView

Slow

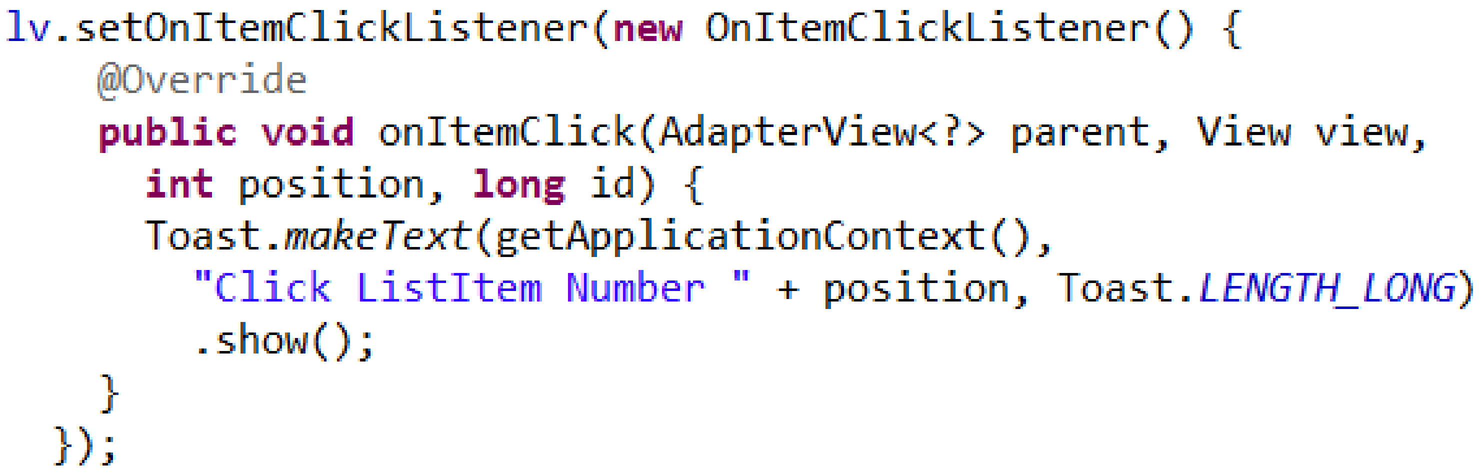
List of 10,000 items, Nexus One device

Sorting List

- **Sort underlying datastructure**
  - How? Collection.sort(myList) Collection.reverse(myList)
- **What about noncomparable or complex objects?**
- **Use comparator interface on data**
  - Define class that implements comparator
- **Sort it when necessary**
- **Call notifyDataSetChanged() to refresh adapter after sort**
- **See LV3**

Listeners

- Responding to List touch events
- onItemClickListener set on Listview



```
lv.setOnItemClickListener(new OnItemClickListener() {
    @Override
    public void onItemClick(AdapterView<?> parent, View view,
      int position, long id) {
      Toast.makeText(getApplicationContext(),
        "Click ListItem Number " + position, Toast.LENGTH_LONG)
        .show();
    }
  });
```

Please read or watch

- http://lucasr.org/2012/04/05/performance-tips-for-androids-listview/
- http://www.youtube.com/watch?v=wDBM6wVEO70
- http://www.vogella.com/articles/AndroidListView/article.html#adapterperformance_problems
- http://stackoverflow.com/questions/1737009/how-to-make-a-nice-looking-listview-filter-on-android
- http://stackoverflow.com/questions/11619874/how-to-implement-getfilter-on-a-baseadapter