

Dati e View

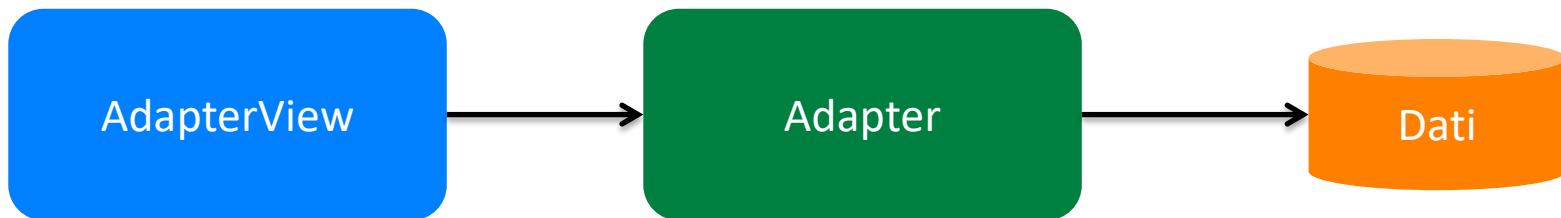
Layout dinamici

Problema

spesso i dati non disponibili se non in esecuzione e quindi devo gestire le view dinamicamente

Pattern Proposto:

Il layout (**AdapterView**) affida ad un oggetto **Adapter** la creazione delle view per visualizzare i dati



AdapterView

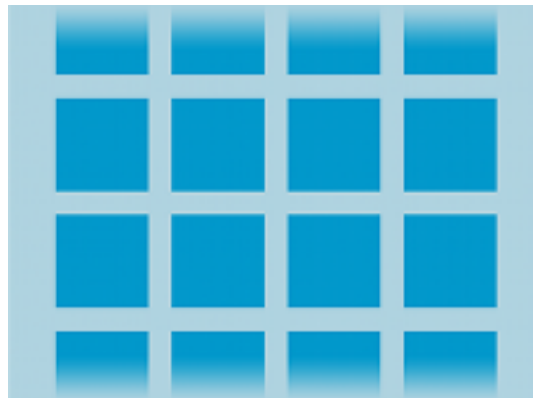
java.lang.Object

↳ android.view.View

↳ android.view.ViewGroup

↳ android.widget.AdapterView

- **ListView**, **GridView**, **RecyclerView**, etc.



Adapter

- Adapter/ListAdapter
 - `ArrayAdapter<T>`, `CursorAdapter`, `SimpleAdapter`, `RecyclerView.Adapter`, etc.
- Adattano dinamicamente i dati alla "vista" corrente
 - gestisce i dati (organizzati come lista)
 - se cambia la visuale vanno cambiate le view
- Gestisce le view che mostrano il singolo dato
 - se cambiano il dato va aggiornata le view

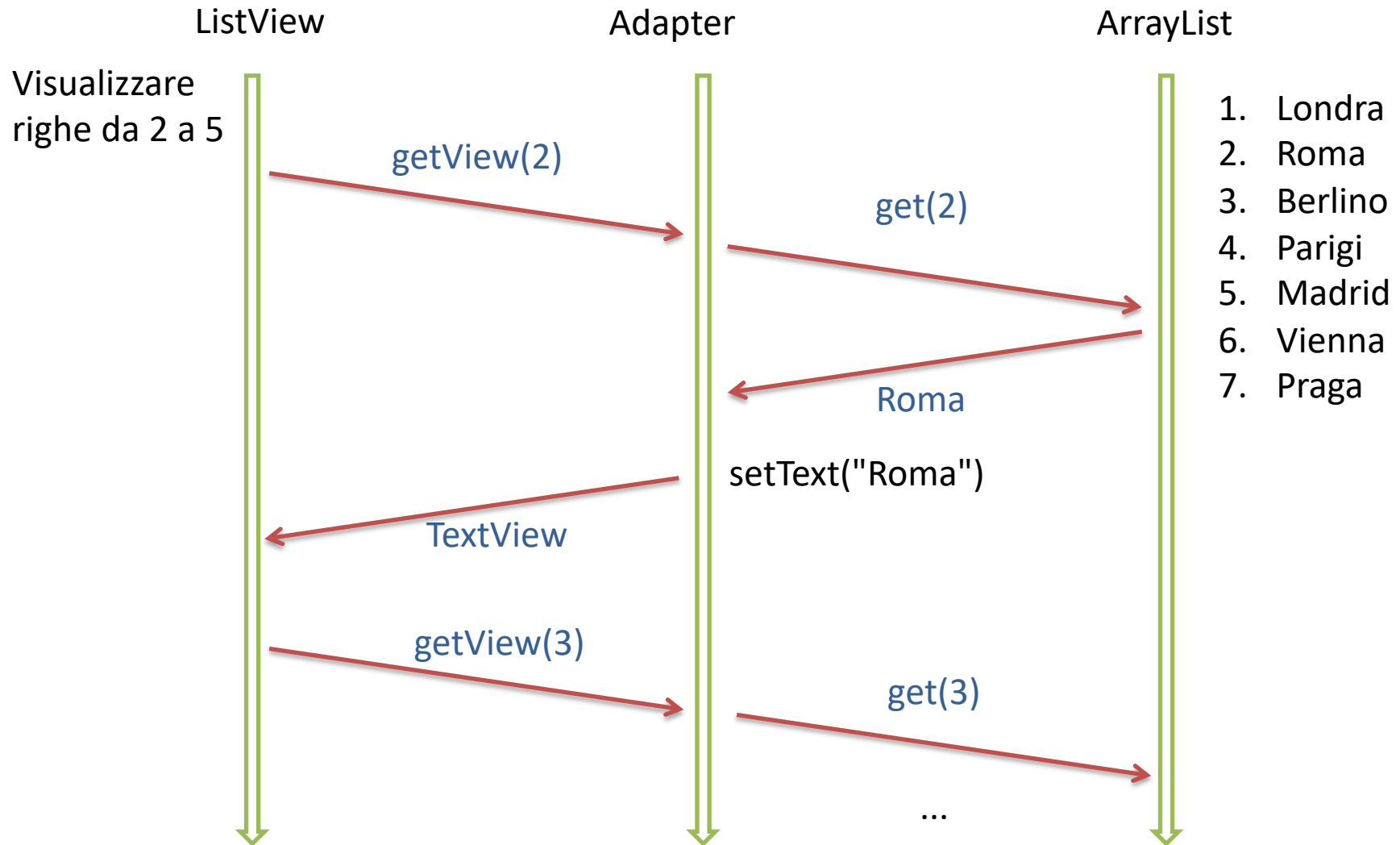
Interfaccia Adapter

Chiedo all'adapter una view

View getView (int position, View convertView, ViewGroup parent)

- Gestione dei dati
 - Object **getItem** (int position)
 - long **getItemId** (int position)
 - int **getCount** ()
 - boolean **isEmpty** ()
- Cambio dei dati
 - void **registerDataSetObserver** (DataSetObserver observer)
 - void **unregisterDataSetObserver** (DataSetObserver observer)
- Base Adapter
 - **notifyDataSetChanged();**

Interazioni



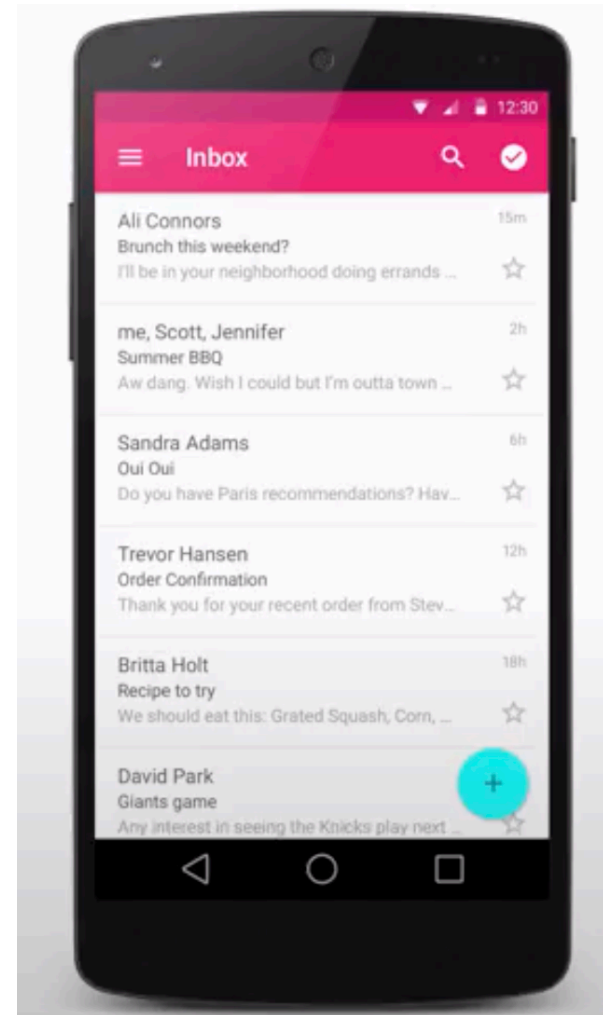
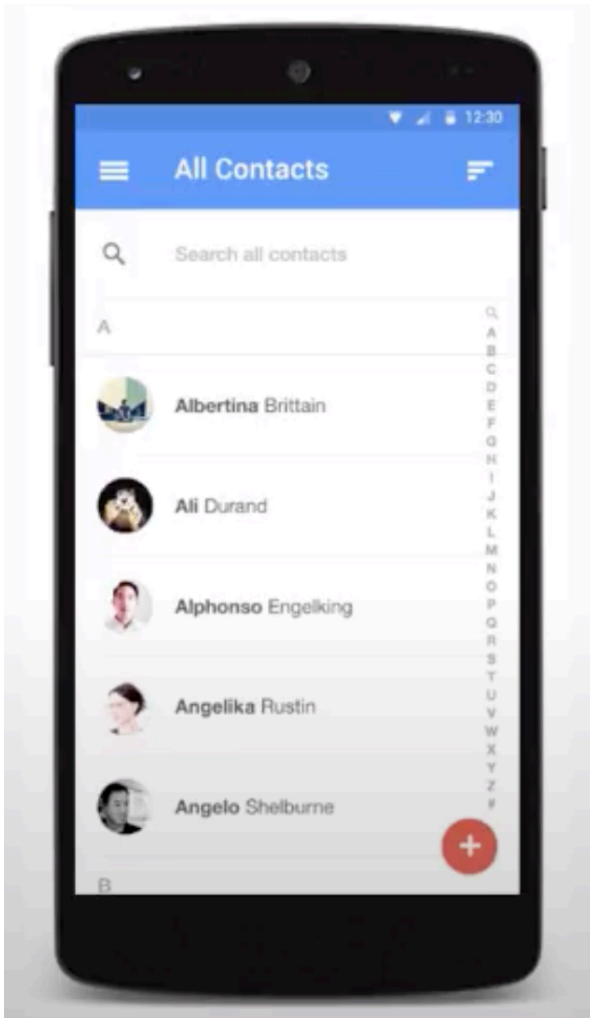
ListView

- uso di un ArrayAdapter con una ListView
 - progetto ArrayAdapterTest

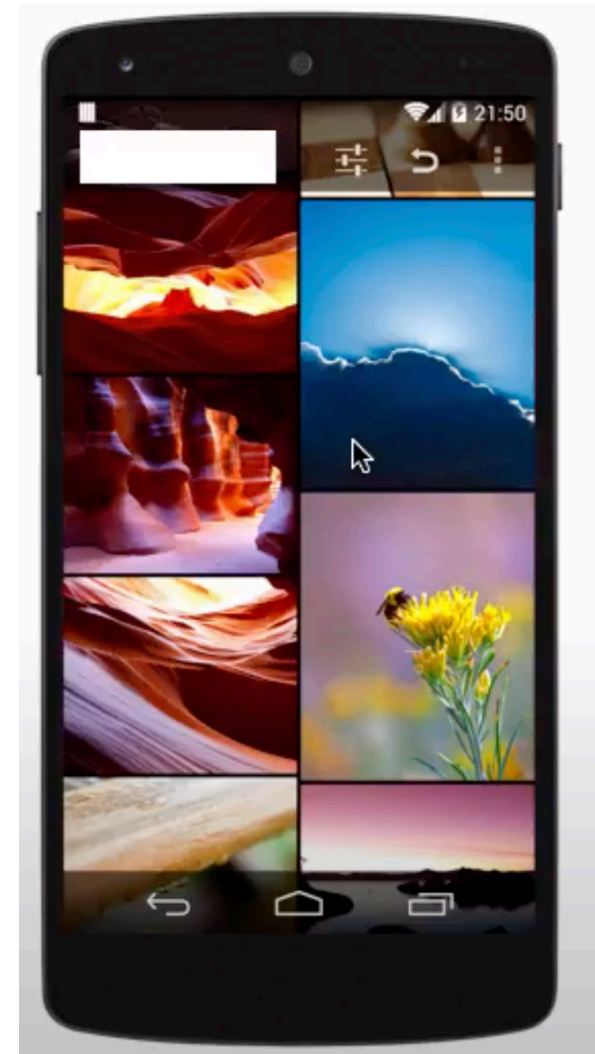
```
ListView listView = (ListView) findViewById(R.id.arrayList);  
String[] arrayData = getResources().getStringArray(R.array.array_data);  
ArrayAdapter<String> arrayAdapter =  
    new ArrayAdapter<String>  
    (this,R.layout.row,R.id.rowText,arrayData);  
listView.setAdapter(arrayAdapter);
```

RECYCLERVIEW E CARD VIEW

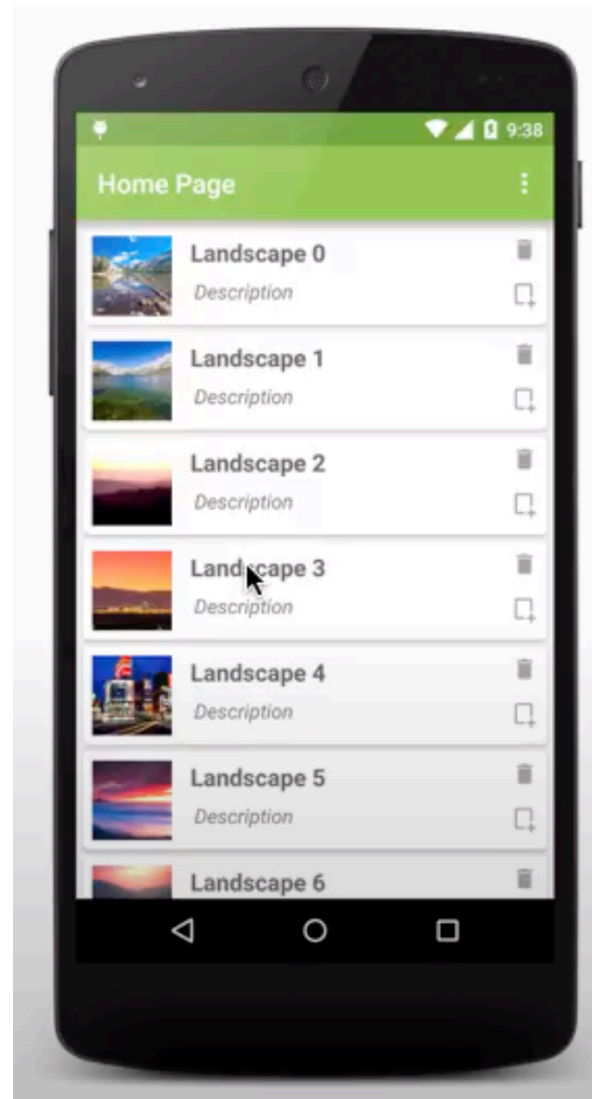
RecyclerView



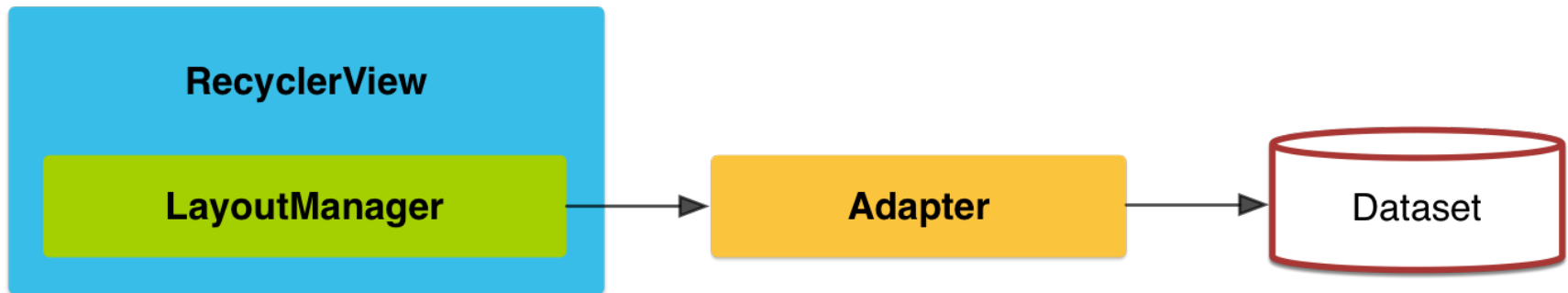
RecyclerView



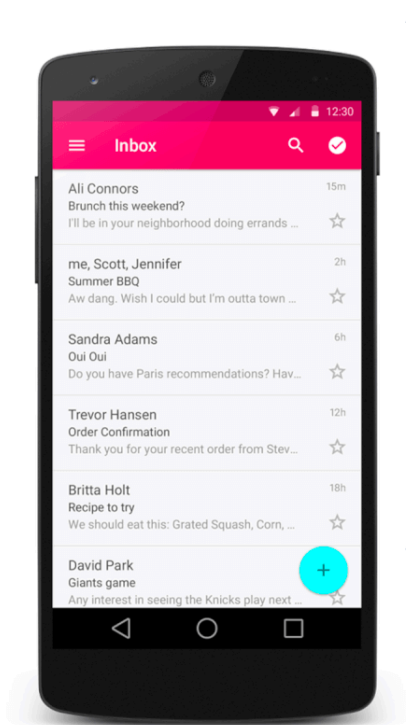
CardView



- RecyclerView
- LayoutManager
 - LinearLayoutManager
 - GridLayoutManager
 - StaggeredGridLayoutManager



RecyclerView

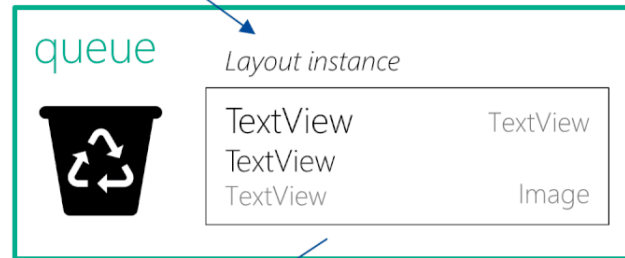


1. Layout instance scrolls out of view

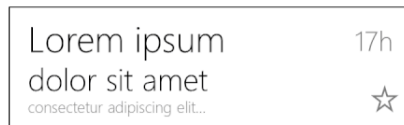
Layout instance



2. Placed in queue



Layout instance



3. Filled with new content & scrolls in again

Adapter

