

# Android ListView and Adapter

James Douglass Lefruit  
Ingénieur de Recherche, INRIA Grenoble



# Android

- ListView and Adapter
  - Introduction au ListView
  - Manipuler une ListView
- Créer son Adapter



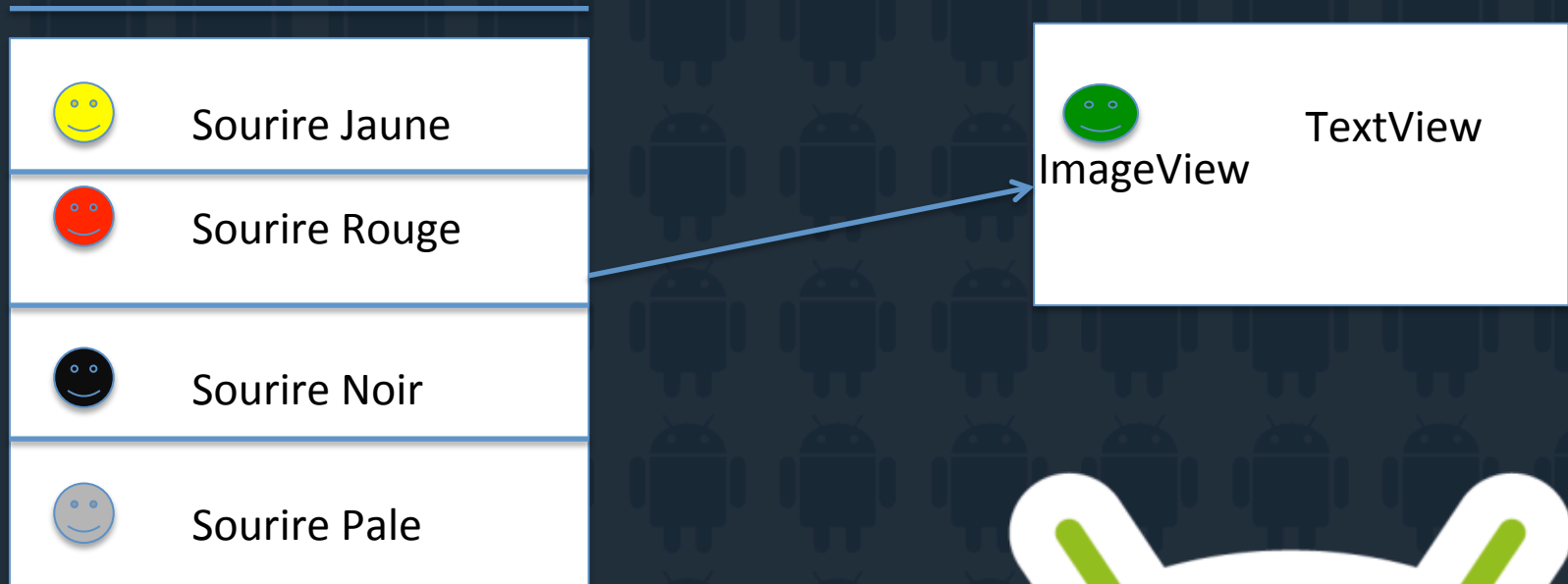
# ListView

- Composant graphique Android permettant répéter une liste d'élément horizontalement
  - Les éléments sont des objets graphiques
  - Une Répétition d'éléments graphiques instanciés
- Suit le modèle MVC pour les données
  - Les données sont contenues dans des objets qui héritent de la classe `BaseAdapter`
  - `BaseAdapter` encapsule les méthodes qui renseignent sur les données
    - Effectif d'éléments, l'objet graphique à dessiner dans la liste



# ListView

Exemple de ListView avec un fichier Layout Type



# Adapteur

- L'API Android fournit des Adapteurs pour des usages courants
  - Un `ArrayAdapter` permet de lister des textes en lui fournissant
    - Un tableau de chaînes de caractères
    - Un fichier Layout qui contient un objet graphique `TextView`



# ListView et Adapteur

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.test_theme);  
    String[] FRUITS = new String[] { "Apple", "Avocado", "Banana",  
                                     "Blueberry", "Coconut", "Durian", "Guava",  
                                     "Kiwifruit", "Jackfruit", "Mango", "Olive",  
                                     "Pear", "Sugar-apple" };  
  
    ListView listView=(ListView)this.findViewById(R.id.listView1);  
    ArrayAdapter adapter=new ArrayAdapter<String>(this.getApplicationContext(),  
                                                R.layout.listview_instance, FRUITS  
                                                );  
    listView.setAdapter(adapter);  
}
```



# Créer Adapteur

- Créer une classe qui hérite de `BaseAdapter`
  - Redéfinir les méthodes
    - `getCount()`, total d'éléments
    - `getItem(int position)`, récupérer l'élément via son indice dans la liste
    - `getView(int position, View convertView, ViewGroup parent)`, matérialiser une vue personnalisée

