

ANDROID

SÉANCE 3

Raphaël Bischof

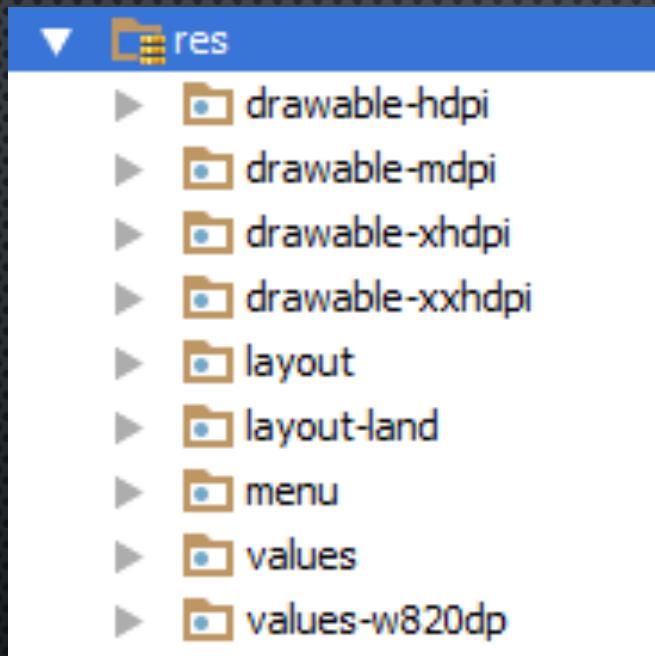
LES RÉPERTOIRES DE RESSOURCES

Répertoires de ressources utilisés à l'heure actuelle :

- Values
- Layout
- Drawable
- Menu

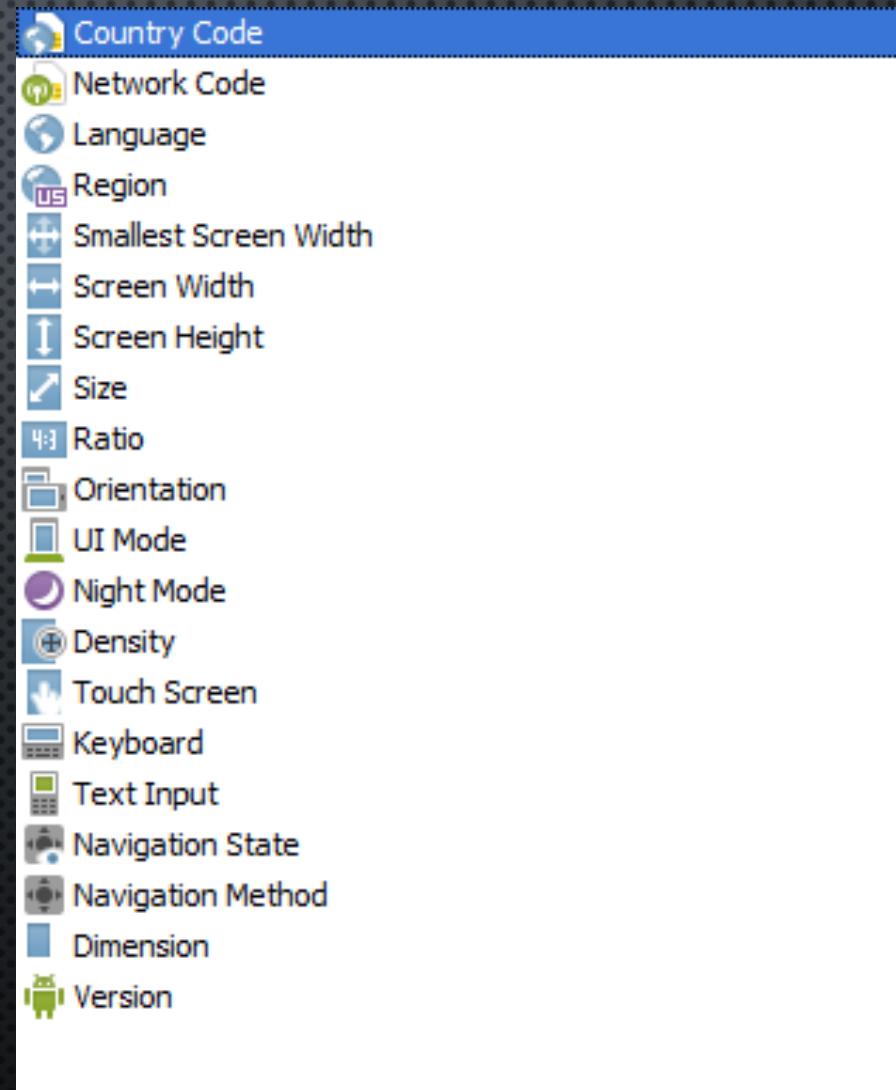
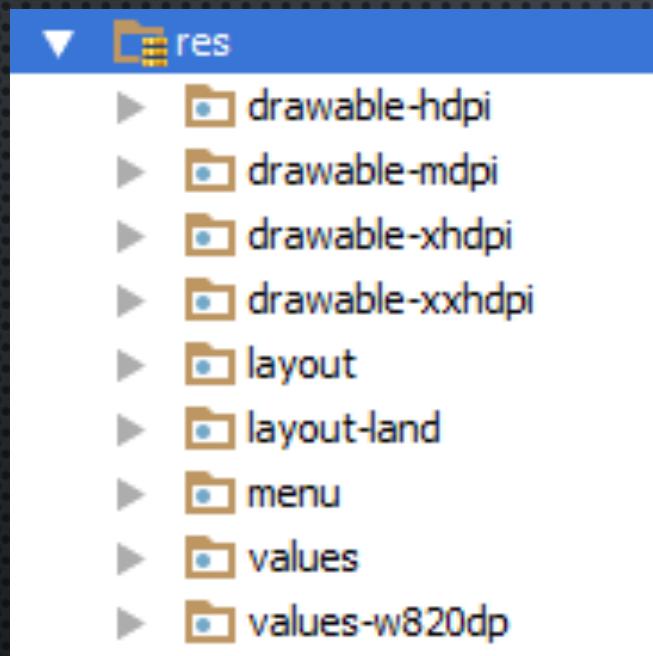
LES RÉPERTOIRES DE RESSOURCES

Possibilité de déclinaison de ces répertoires :



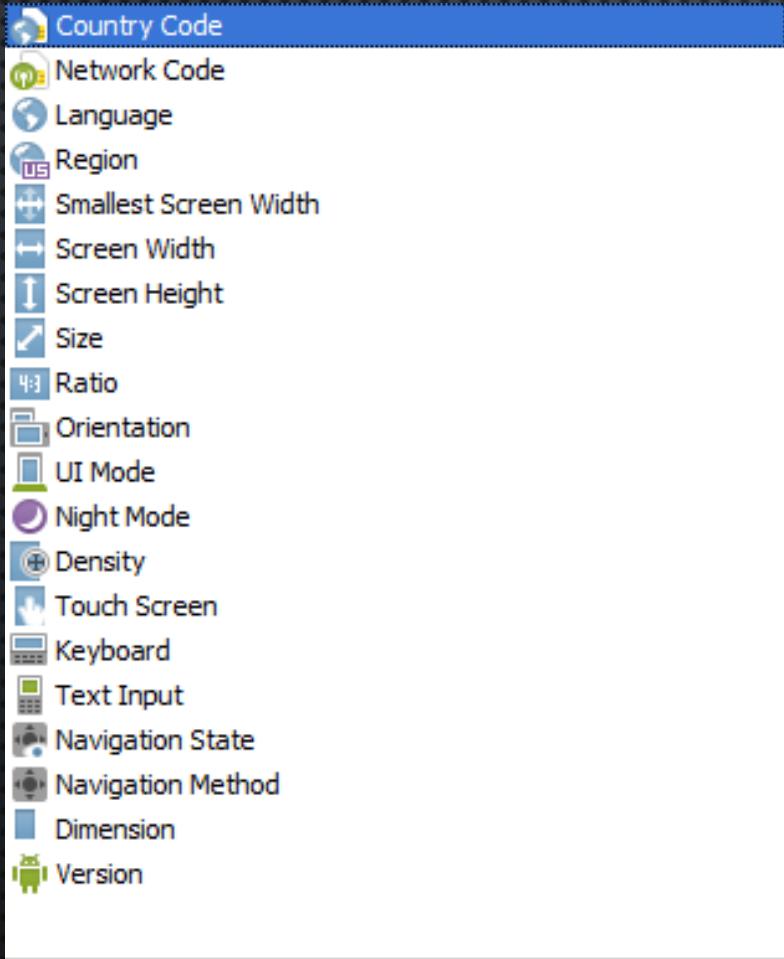
LES RÉPERTOIRES DE RESSOURCES

Possibilité de déclinaison de ces répertoires :



LES RÉPERTOIRES DE RESSOURCES

Possibilité de déclinaison de ces répertoires :



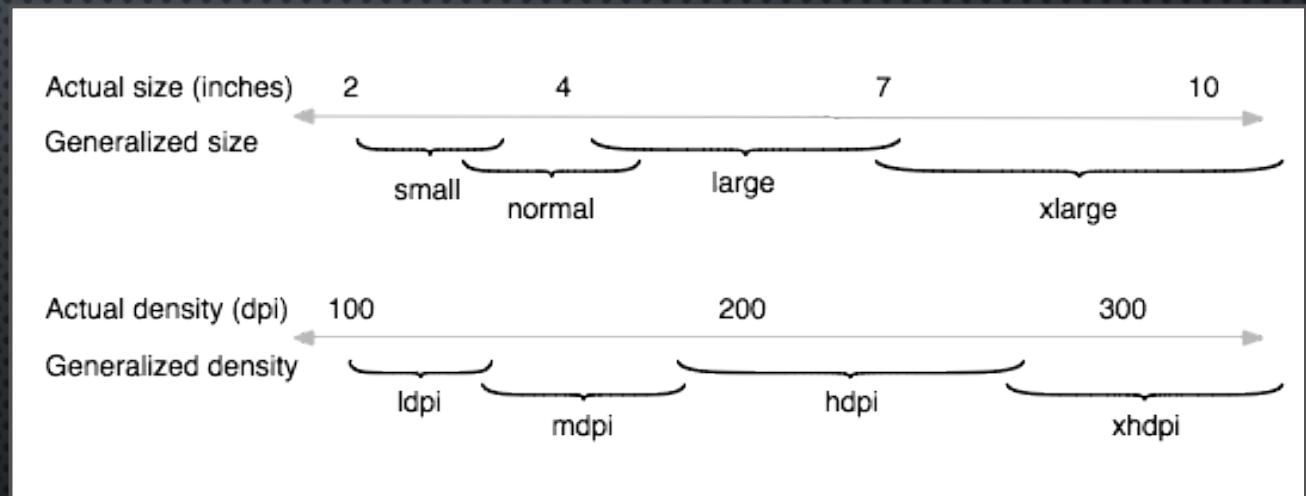
Les déclinaisons les plus utilisés sont :

- Language (notamment dans values pour définir les différents strings.xml qui contiendront les textes traduits)
Exemples : values-en, values-fr, values-de
- Orientation pour définir vos layout en fonction de l'orientation Exemples : layout-port, layout-land
- Size pour définir la taille d'écran pour lesquels seront vos layout ([Voir +...](#)) Exemples : layout-normal, layout-large
- Density pour définir la définition que devront avoir vos drawable Exemple : drawable-hdpi, drawable-xhdpi

LES RÉPERTOIRES DE RESSOURCES

Déclinaison qui nous intéresse pour gérer les tailles d'écrans multiples :

- Orientation pour définir vos layout en fonction de l'orientation
 - Exemples : layout-port, layout-land
- Size pour définir la taille d'écran pour lesquels seront vos layout ([Voir +...](#))
 - Exemples : layout-normal, layout-large
- Density pour définir la définition que devront avoir vos drawable
 - Exemples : drawable-hdpi, drawable-xhdpi



LES FRAGMENTS

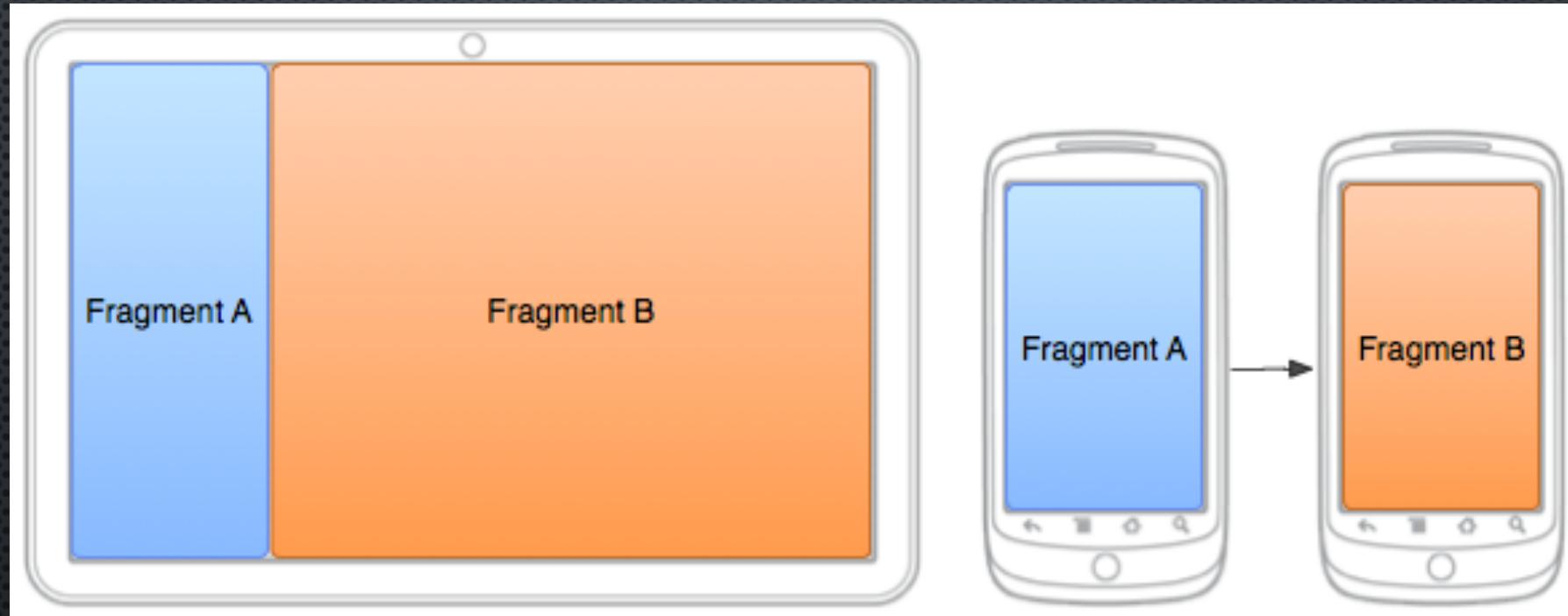
La définition :

- Un fragment est un morceau d'Activity, il a le même comportement qu'une Activity. Une Activity peut comporter plusieurs fragments.

Le but :

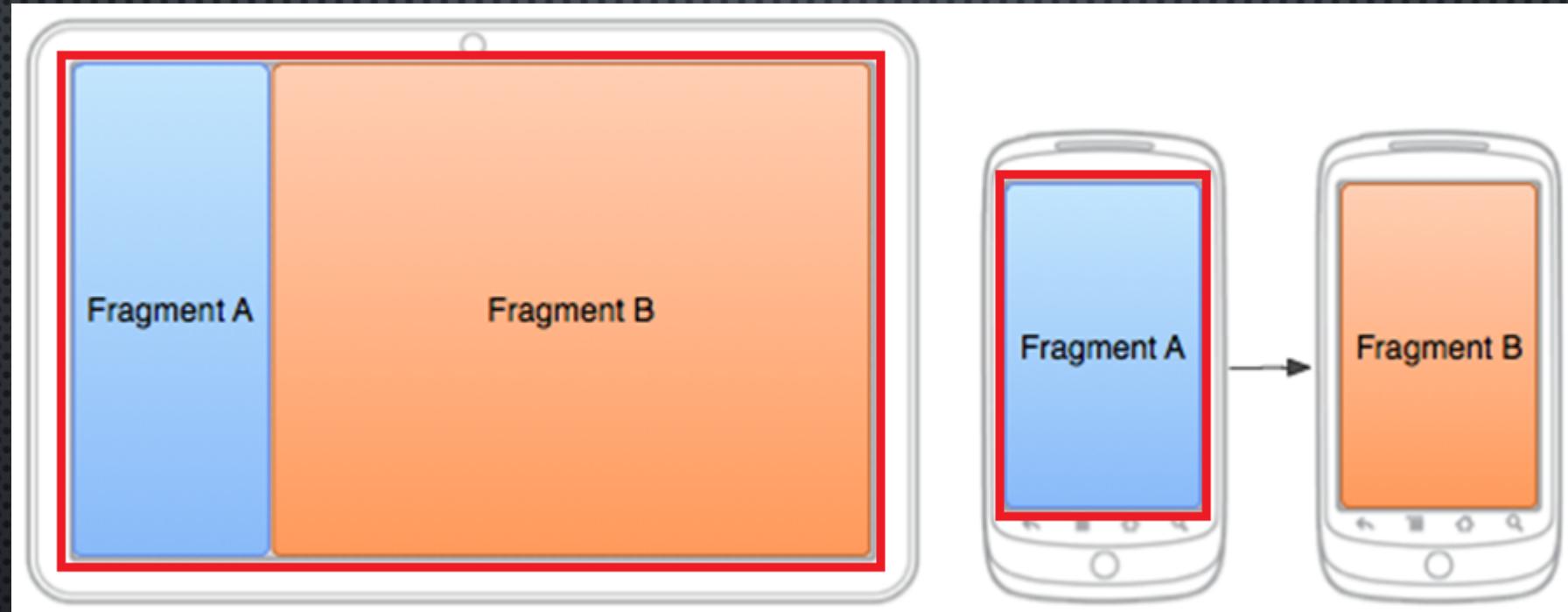
- Gérer l'affichage des informations selon l'appareil utilisé

LES FRAGMENTS



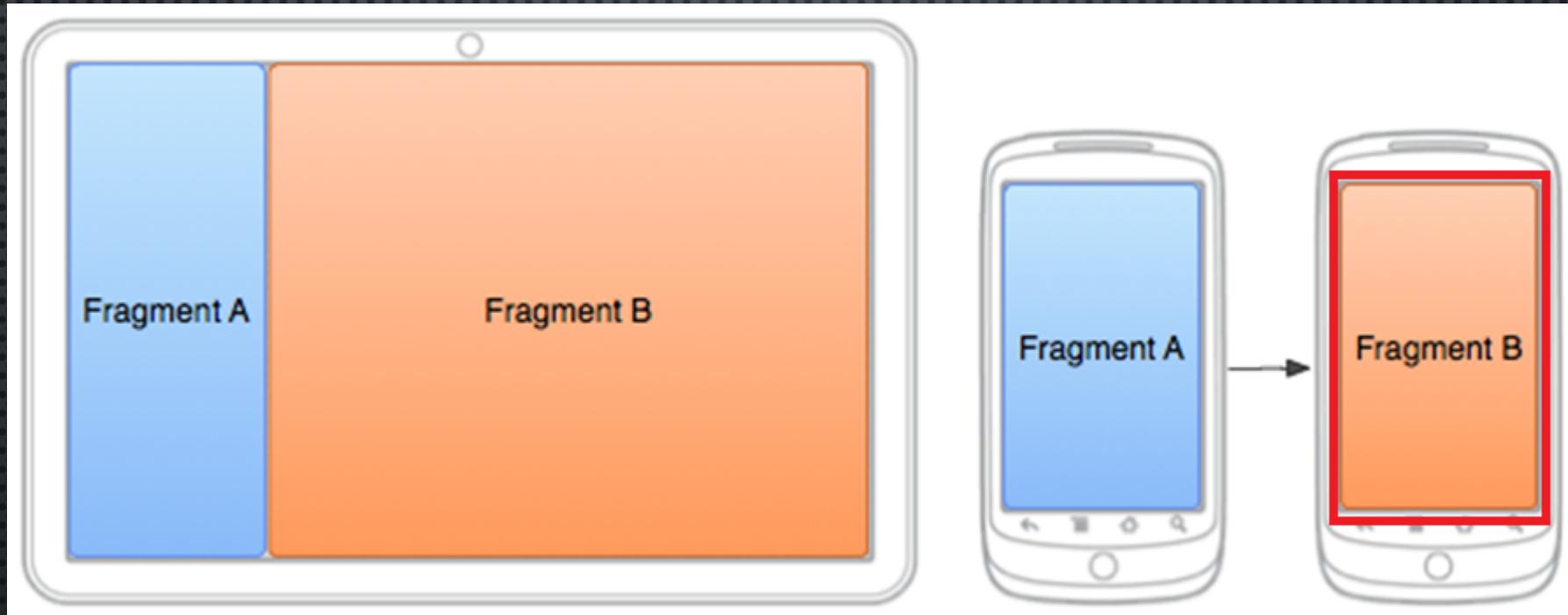
Une seule définition d'un Fragment réutilisable

LES FRAGMENTS



Entouré en rouge, la MainActivity
Un layout par Activity et par type d'appareil

LES FRAGMENTS

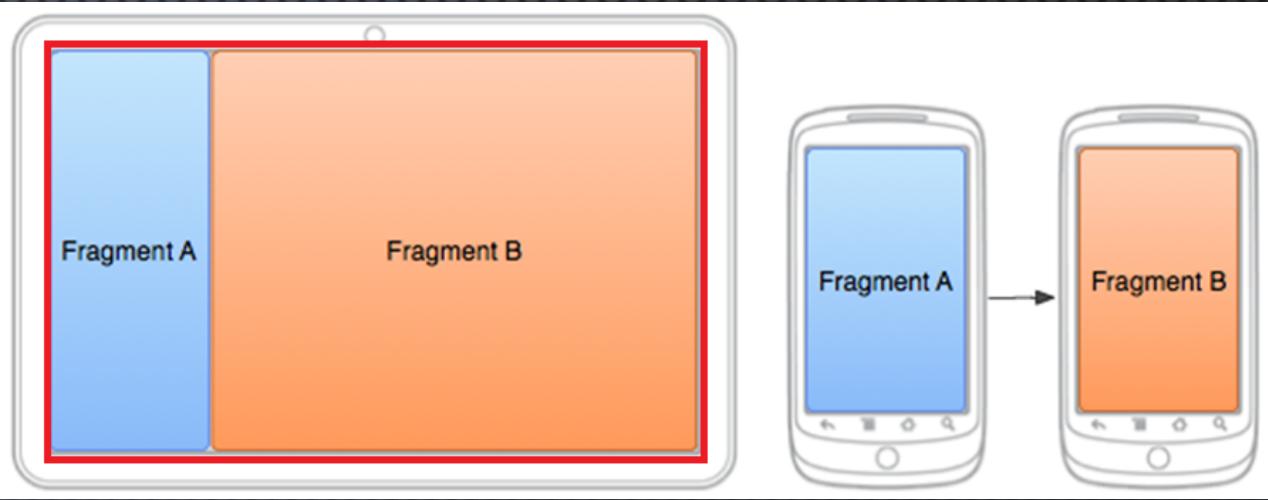


Entouré en rouge une DetailActivity démarrée uniquement si on a un smartphone

LES FRAGMENTS

Définition du layout-large de la
MainActivity

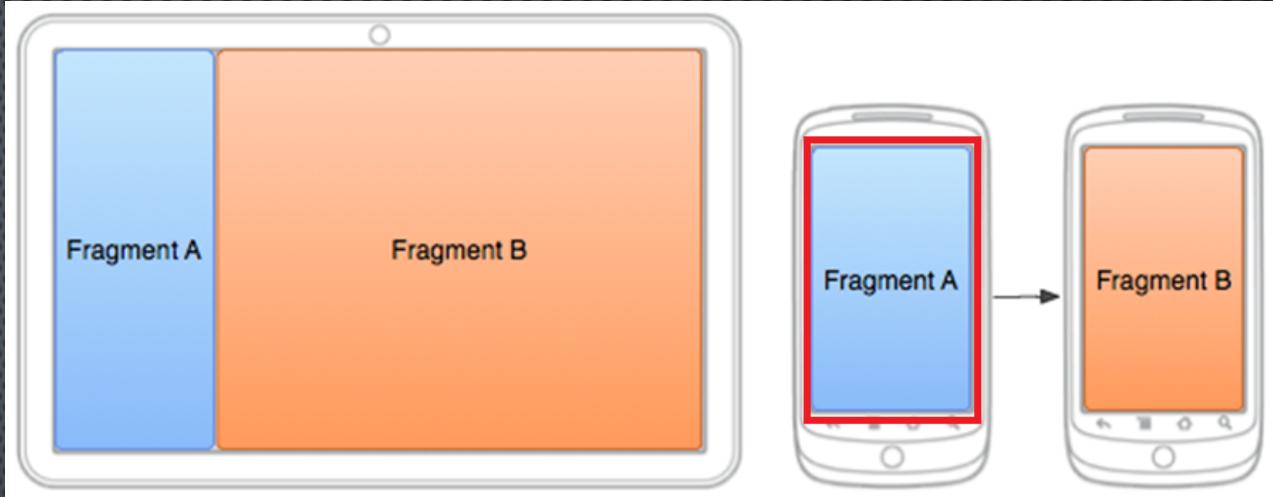
```
<LinearLayout  
    xmlns:android="http://schemas.android.com/apk/res/android"  
    android:id="@+id/container"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:orientation="horizontal">  
    <fragment  
        android:layout_width="0dp"  
        android:layout_height="match_parent"  
        android:layout_weight="1"  
        android:name="com.example.fragments.FragmentA"  
        android:id="@+id/fragmentA_ID" />  
  
    <fragment  
        android:layout_width="0dp"  
        android:layout_height="match_parent"  
        android:layout_weight="3"  
        android:name="com.example.fragments.FragmentB"  
        android:id="@+id/fragmentB_ID"  
        android:layout_gravity="center_vertical" />  
</LinearLayout>
```



- Layout contient les 2 fragments à afficher
- Android name correspond au chemin de la classe du fragment
- Pour ce layout : définition de 0dp en largeur des fragments + définition du poids -> répartition de la largeur de l'écran en fonction du poids

LES FRAGMENTS

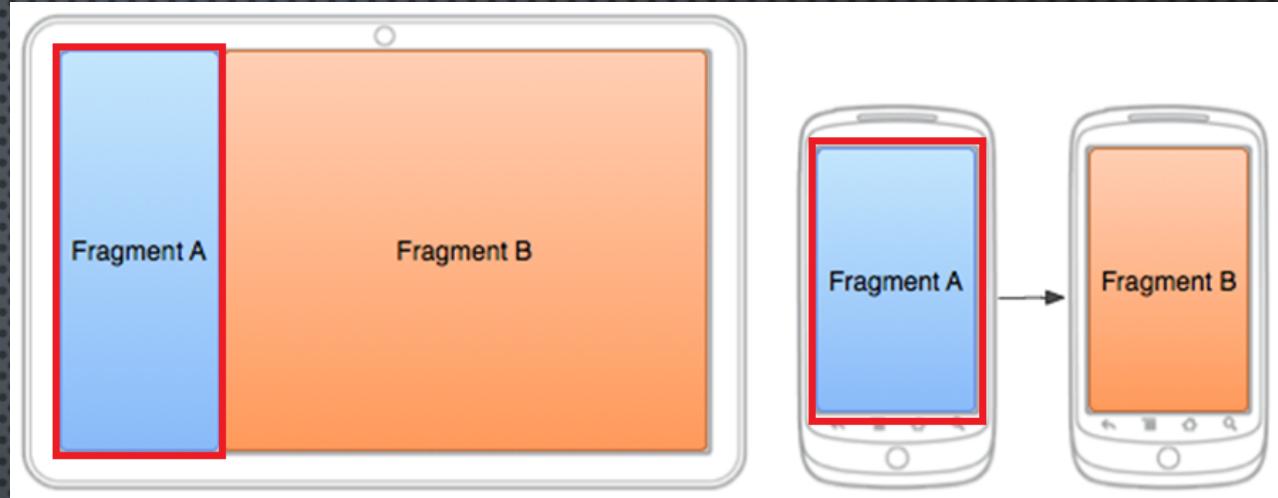
Définition du layout par défaut de la MainActivity



```
<LinearLayout  
    xmlns:android="http://schemas.android.com/apk/res/android"  
        android:id="@+id/container"  
        android:layout_width="match_parent"  
        android:layout_height="match_parent"  
        android:orientation="horizontal">  
    <fragment  
        android:layout_width="0dp"  
        android:layout_height="match_parent"  
        android:layout_weight="1"  
        android:name="com.example.fragments.FragmentA"  
        android:id="@+id/fragmentA_ID" />  
</LinearLayout>
```

LES FRAGMENTS

Définition du layout du FragmentA



```
<LinearLayout  
    xmlns:android="http://schemas.android.com/apk/res/android"  
        android:orientation="vertical"  
        android:layout_width="match_parent"  
        android:layout_height="match_parent">  
    <ListView  
        android:layout_width="fill_parent"  
        android:layout_height="fill_parent"  
        android:id="@+id/lvFragment" />  
</LinearLayout>
```

LES FRAGMENTS

Définition du layout du FragmentB



```
<LinearLayout  
    xmlns:android="http://schemas.android.com/apk/res/android"  
        android:orientation="vertical"  
        android:layout_width="match_parent"  
        android:layout_height="match_parent">  
    <TextView  
        android:layout_width="wrap_content"  
        android:layout_height="wrap_content"  
        android:id="@+id/textView"  
        android:layout_gravity="center_horizontal" />  
</LinearLayout>
```

LES FRAGMENTS

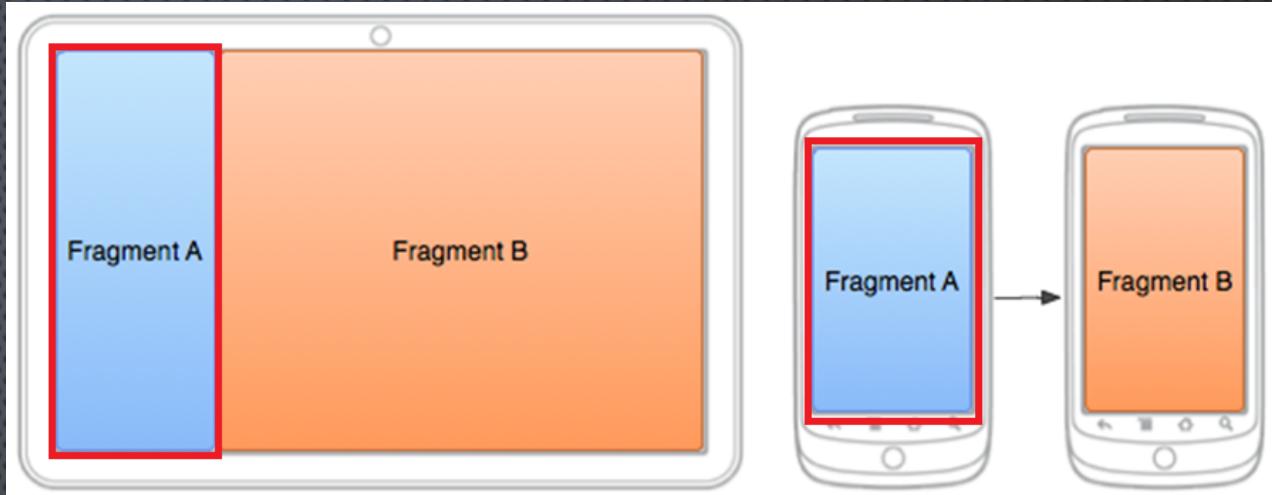
Définition du layout de la
DetailActivity



```
<LinearLayout  
    xmlns:android="http://schemas.android.com/apk/res/android"  
        android:orientation="vertical"  
        android:layout_width="match_parent"  
        android:layout_height="match_parent">  
  
    <fragment  
        android:layout_width="match_parent"  
        android:layout_height="wrap_content"  
        android:name="com.example.fragments.FragmentB"  
        android:id="@+id/fragmentB_ID"  
        android:layout_gravity="right" />  
    </LinearLayout>
```

LES FRAGMENTS

Définition du code du FragmentA



```
public class FragmentA extends Fragment {  
    private ListView lvFragment;  
    public String[] listItems = {"item 1", "item 2 ", "list", "android", "item 3", "foobar", "bar", };  
  
    @Override  
    public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {  
        View v = inflater.inflate(R.layout.fragment_a, container);  
        lvFragment = (ListView)v.findViewById(R.id.lvFragment);  
        return v;  
    }  
  
    @Override  
    public void onActivityCreated(Bundle savedInstanceState) {  
        super.onActivityCreated(savedInstanceState);  
        lvFragment.setAdapter(new ArrayAdapter(getActivity(),  
                android.R.layout.simple_list_item_1, listItems));  
        lvFragment.setOnItemClickListener((MainActivity) getActivity());  
    }  
}
```

LES FRAGMENTS

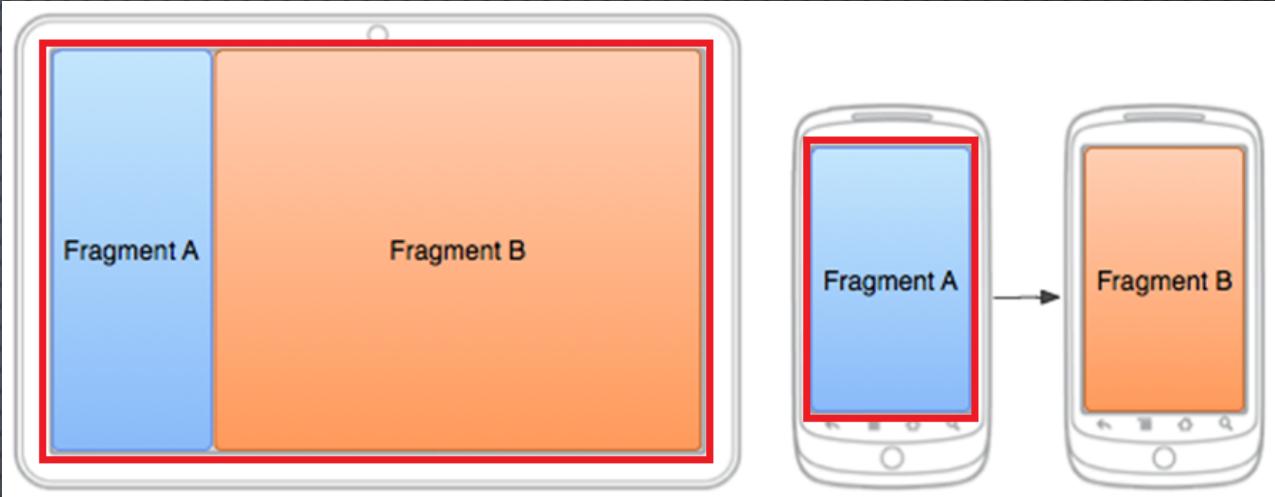
Définition du code du FragmentB



```
public class FragmentB extends Fragment {  
    private TextView tvExample;  
    @Override  
    public View onCreateView(LayoutInflater inflater, ViewGroup  
    container, Bundle savedInstanceState) {  
        View v = inflater.inflate(R.layout.fragment_b,container);  
        tvExample = (TextView)v.findViewById(R.id.textView);  
        return v;  
    }  
  
    public void fillTextView(String listItem) {  
        tvExample.setText(listItem);  
    }  
}
```

LES FRAGMENTS

Définition du code de la MainActivity



```
public class MainActivity extends ActionBarActivity implements AdapterView.OnItemClickListener {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main_activity);
    }

    @Override
    public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
        FragmentA fragA = (FragmentA) getSupportFragmentManager().findFragmentById(R.id.fragmentA_ID);
        FragmentB fragB = (FragmentB) getSupportFragmentManager().findFragmentById(R.id.fragmentB_ID);
        if(fragB == null || !fragB.isInLayout()){
            Intent i = new Intent(getApplicationContext(),DetailActivity.class);
            i.putExtra("monTextAAfficher",fragA.listItems[position]);
            startActivity(i);
        } else {
            fragB.fillTextView(fragA.listItems[position]);
        }
    }
}
```

LES FRAGMENTS

Définition du code de la DetailActivity



```
public class DetailActivity extends ActionBarActivity {  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.detail_activity);  
        String texteAAfficher =  
getIntent().getStringExtra("monTextAAfficher");  
        FragmentB fragB =  
(FragmentB) getSupportFragmentManager().findFragmentById(R.id.fragmentB_ID);  
        fragB.fillTextView(texteAAfficher);  
    }  
}
```