Formation Android

Épisode V

Xavier MARIN

@XavMarin

https://github.com/Giwi

CTO chez @qaobee

Objectifs de la formation

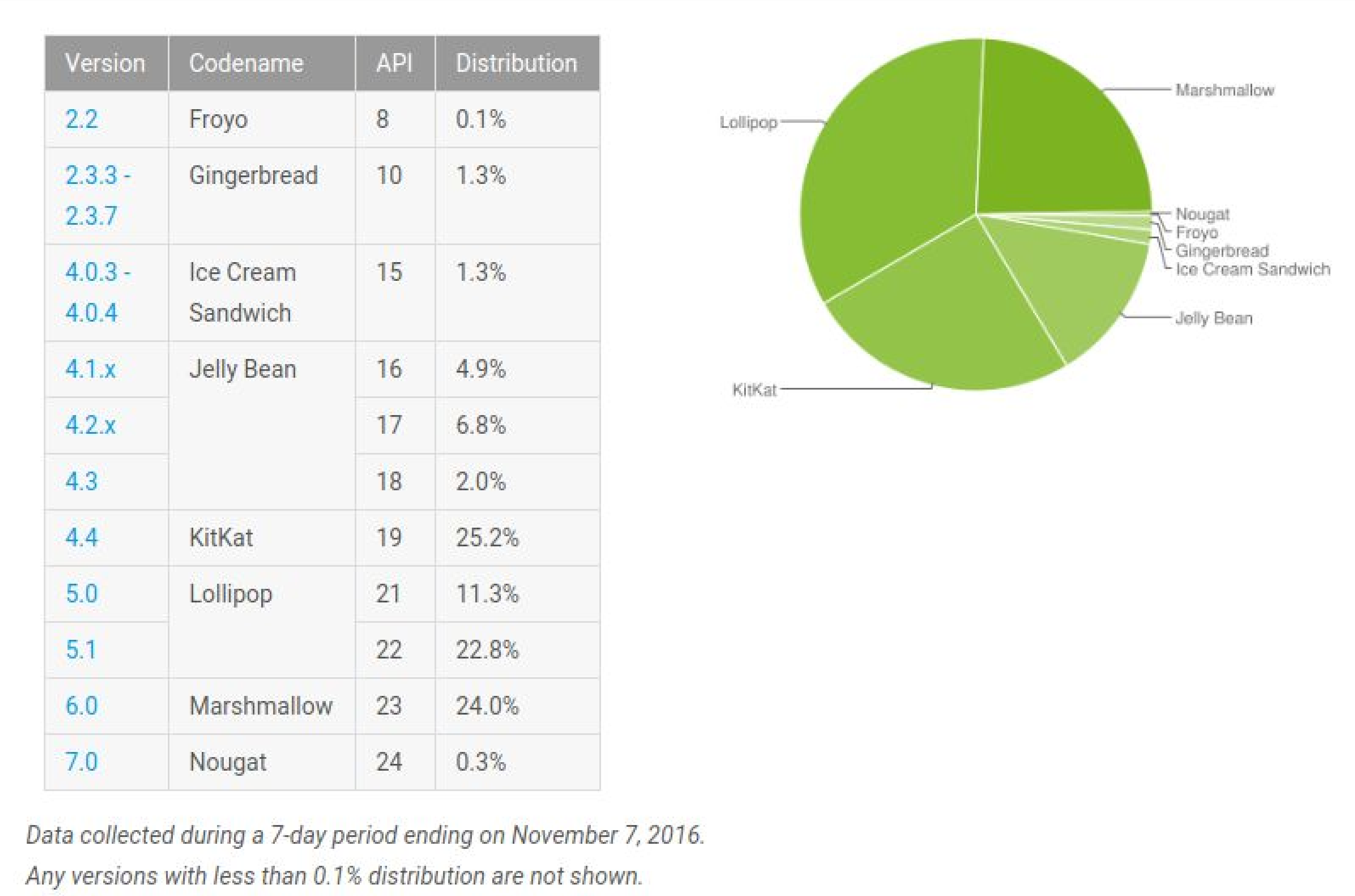
Maîtriser et être autonome sur les éléments suivants:

- Comprendre la fragmentation
- Savoir utiliser les fragments

Android Fragmentation

- Taille écran
- Densité de l’écran
- Version de l’OS

Version d’OS



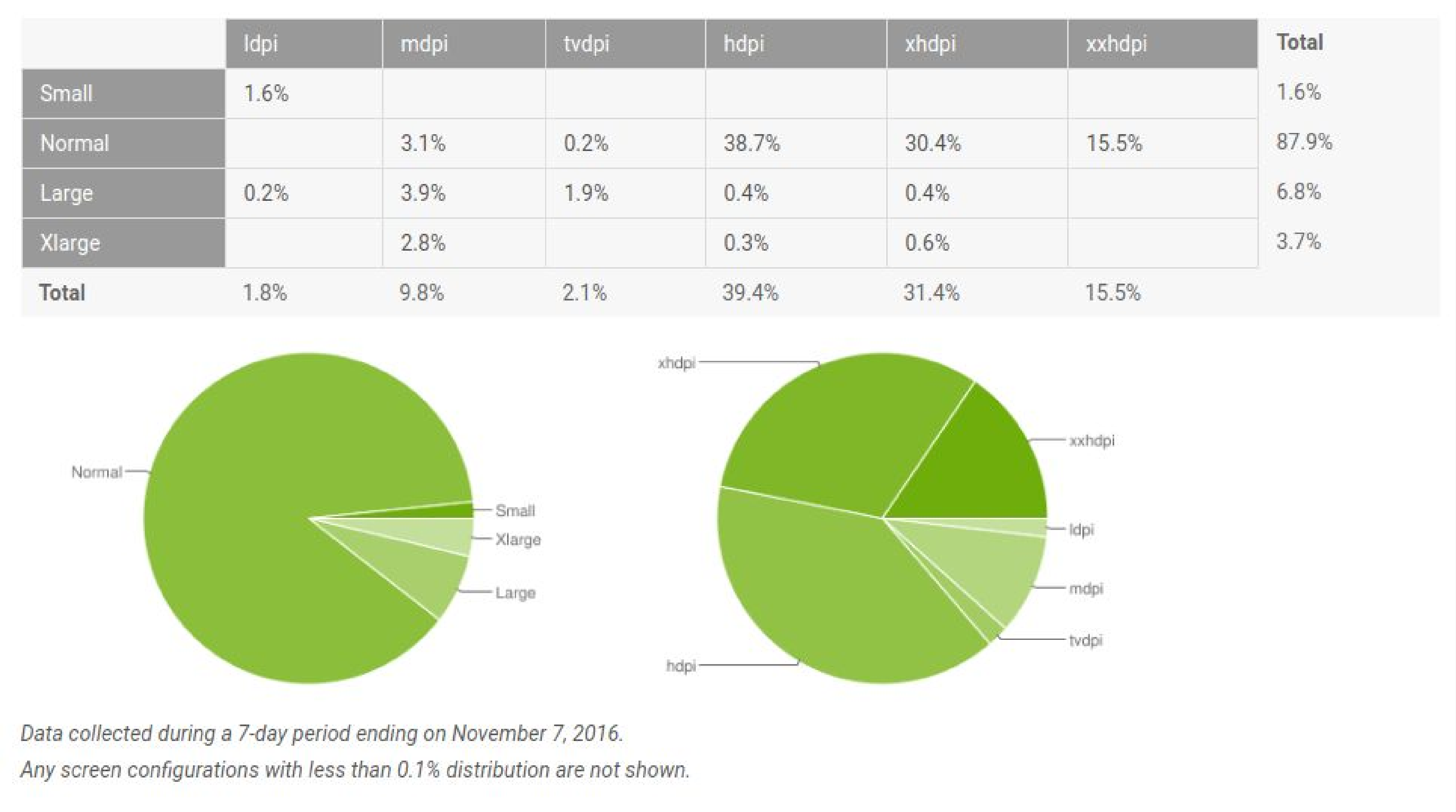
Version	Codename	API	Distribution
2.2	Froyo	8	0.1%
2.3.3 - 2.3.7	Gingerbread	10	1.3%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	1.3%
4.1.x	Jelly Bean	16	4.9%
4.2.x		17	6.8%
4.3		18	2.0%
4.4	KitKat	19	25.2%
5.0	Lollipop	21	11.3%
5.1		22	22.8%
6.0	Marshmallow	23	24.0%
7.0	Nougat	24	0.3%

Data collected during a 7-day period ending on November 7, 2016.

Any versions with less than 0.1% distribution are not shown.

https://developer.android.com/about/dashboards/index.html

Screen sizes



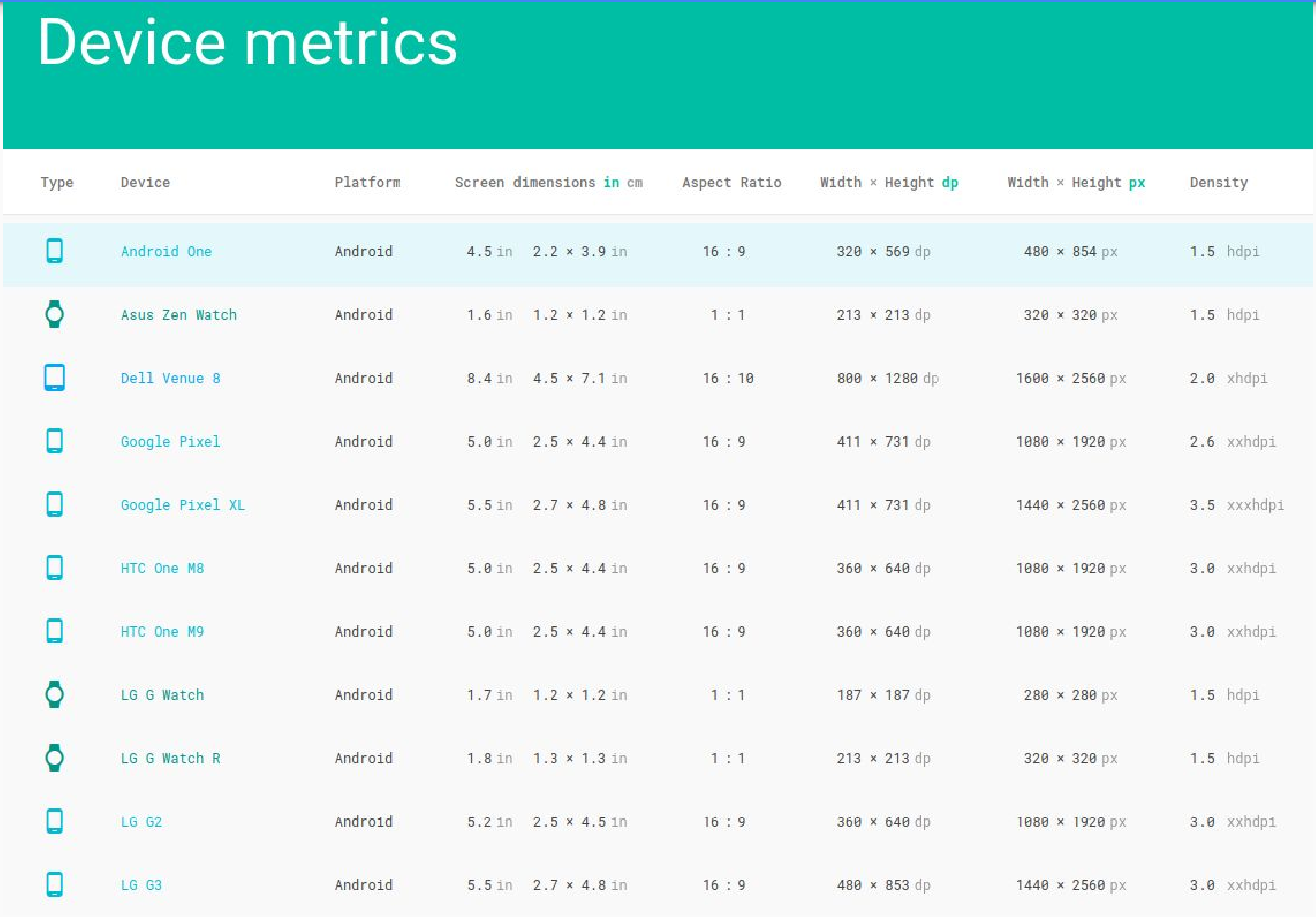
	ldpi	mdpi	tvdpi	hdpi	xhdpi	xxhdpi	Total
Small	1.6%						1.6%
Normal		3.1%	0.2%	38.7%	30.4%	15.5%	87.9%
Large	0.2%	3.9%	1.9%	0.4%	0.4%		6.8%
Xlarge		2.8%		0.3%	0.6%		3.7%
Total	1.8%	9.8%	2.1%	39.4%	31.4%	15.5%	

Data collected during a 7-day period ending on November 7, 2016.

Any screen configurations with less than 0.1% distribution are not shown.

https://developer.android.com/about/dashboards/index.html

Vue intéressante



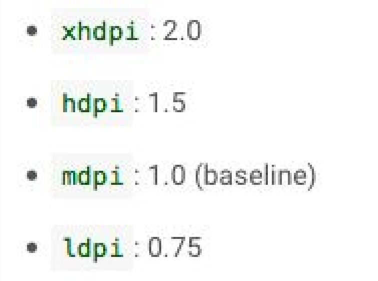
Device metrics

Type	Device	Platform	Screen dimensions in cm	Aspect Ratio	Width × Height dp	Width × Height px	Density
📱	Android One	Android	4.5 in 2.2 × 3.9 in	16 : 9	320 × 569 dp	480 × 854 px	1.5 hdpi
⌚	Asus Zen Watch	Android	1.6 in 1.2 × 1.2 in	1 : 1	213 × 213 dp	320 × 320 px	1.5 hdpi
📱	Dell Venue 8	Android	8.4 in 4.5 × 7.1 in	16 : 10	800 × 1280 dp	1600 × 2560 px	2.0 xhdpi
📱	Google Pixel	Android	5.0 in 2.5 × 4.4 in	16 : 9	411 × 731 dp	1080 × 1920 px	2.6 xxhdpi
📱	Google Pixel XL	Android	5.5 in 2.7 × 4.8 in	16 : 9	411 × 731 dp	1440 × 2560 px	3.5 xxxhdpi
📱	HTC One M8	Android	5.0 in 2.5 × 4.4 in	16 : 9	360 × 640 dp	1080 × 1920 px	3.0 xxhdpi
📱	HTC One M9	Android	5.0 in 2.5 × 4.4 in	16 : 9	360 × 640 dp	1080 × 1920 px	3.0 xxhdpi
⌚	LG G Watch	Android	1.7 in 1.2 × 1.2 in	1 : 1	187 × 187 dp	280 × 280 px	1.5 hdpi
⌚	LG G Watch R	Android	1.8 in 1.3 × 1.3 in	1 : 1	213 × 213 dp	320 × 320 px	1.5 hdpi
📱	LG G2	Android	5.2 in 2.5 × 4.5 in	16 : 9	360 × 640 dp	1080 × 1920 px	3.0 xxhdpi
📱	LG G3	Android	5.5 in 2.7 × 4.8 in	16 : 9	480 × 853 dp	1440 × 2560 px	3.0 xxhdpi

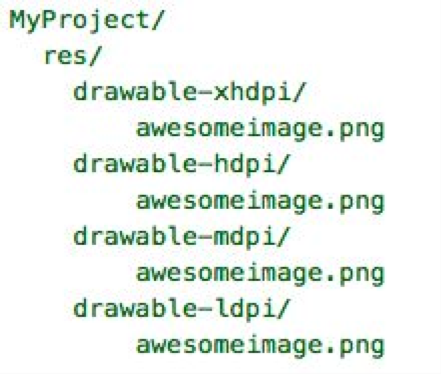
http://www.google.com/design/tool/devices/

Gérer la densité

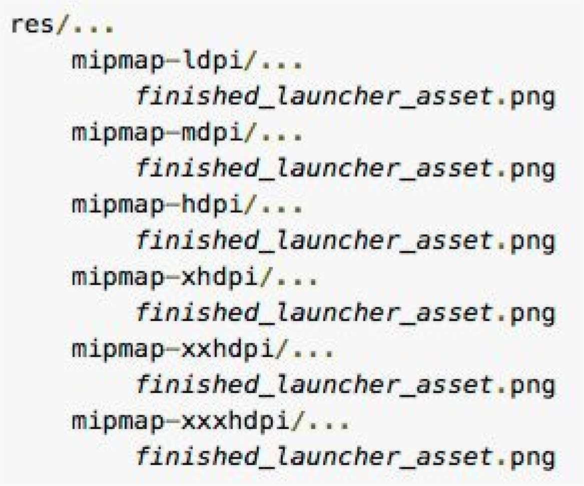
- Ne “jamais” utiliser de pixels
  - Utilisation des dip
- Différents images pour différentes densités



- `xhdpi` : 2.0
- `hdpi` : 1.5
- `mdpi` : 1.0 (baseline)
- `ldpi` : 0.75



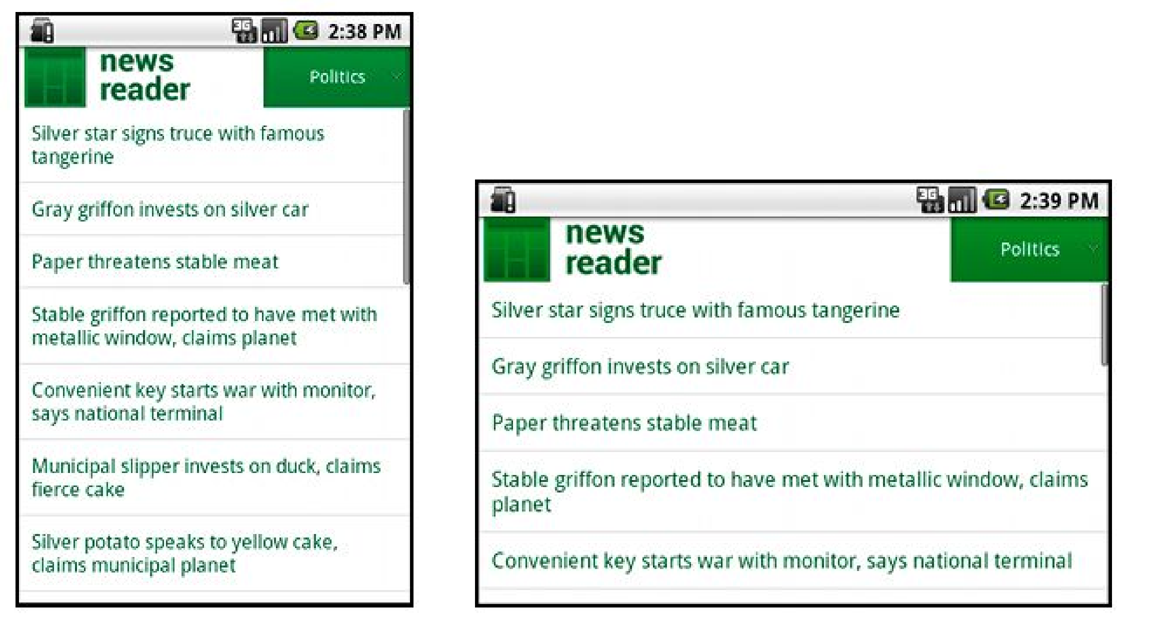
```
MyProject/
  res/
    drawable-xhdpi/
        awesomeimage.png
    drawable-hdpi/
        awesomeimage.png
    drawable-mdpi/
        awesomeimage.png
    drawable-ldpi/
        awesomeimage.png
```



```
res/...
    mipmap-ldpi/...
        finished_launcher_asset.png
    mipmap-mdpi/...
        finished_launcher_asset.png
    mipmap-hdpi/...
        finished_launcher_asset.png
    mipmap-xhdpi/...
        finished_launcher_asset.png
    mipmap-xxhdpi/...
        finished_launcher_asset.png
    mipmap-xxxhdpi/...
        finished_launcher_asset.png
```

Gérer les tailles - Layouts

- Taille des layouts
  - match_parent / wrap_content
- RelativeLayout
- 9patch
- Alias & les qualifier



2:38 PM

news reader

Silver star signs truce with famous tangerine

Gray griffon invests on silver car

Paper threatens stable meat

Stable griffon reported to have met with metallic window, claims planet

Convenient key starts war with monitor, says national terminal

Municipal slipper invests on duck, claims fierce cake

Silver potato speaks to yellow cake, claims municipal planet

2:39 PM

news reader

Silver star signs truce with famous tangerine

Gray griffon invests on silver car

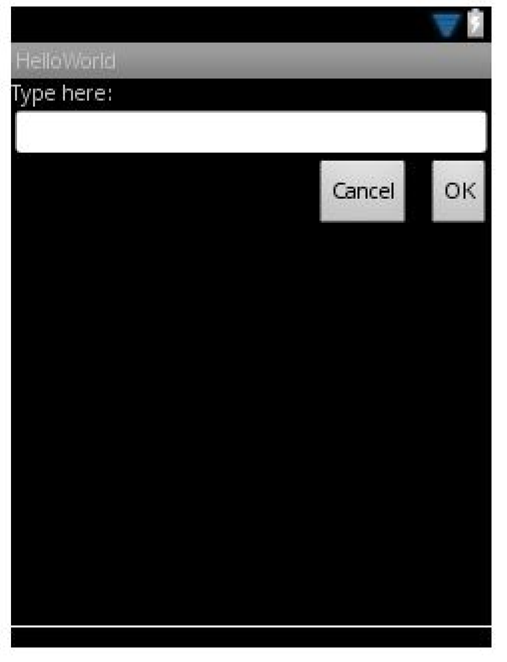
Paper threatens stable meat

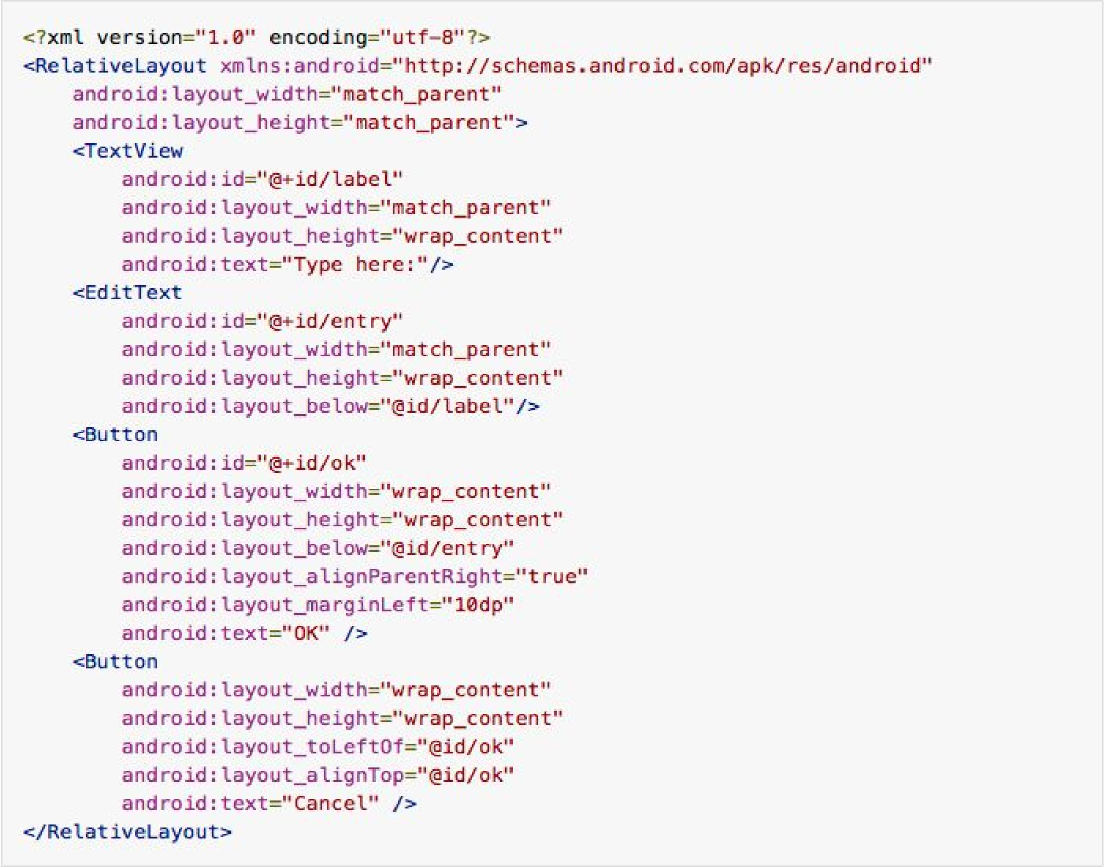
Stable griffon reported to have met with metallic window, claims planet

Convenient key starts war with monitor, says national terminal

Gérer les tailles - Layouts

RelativeLayout



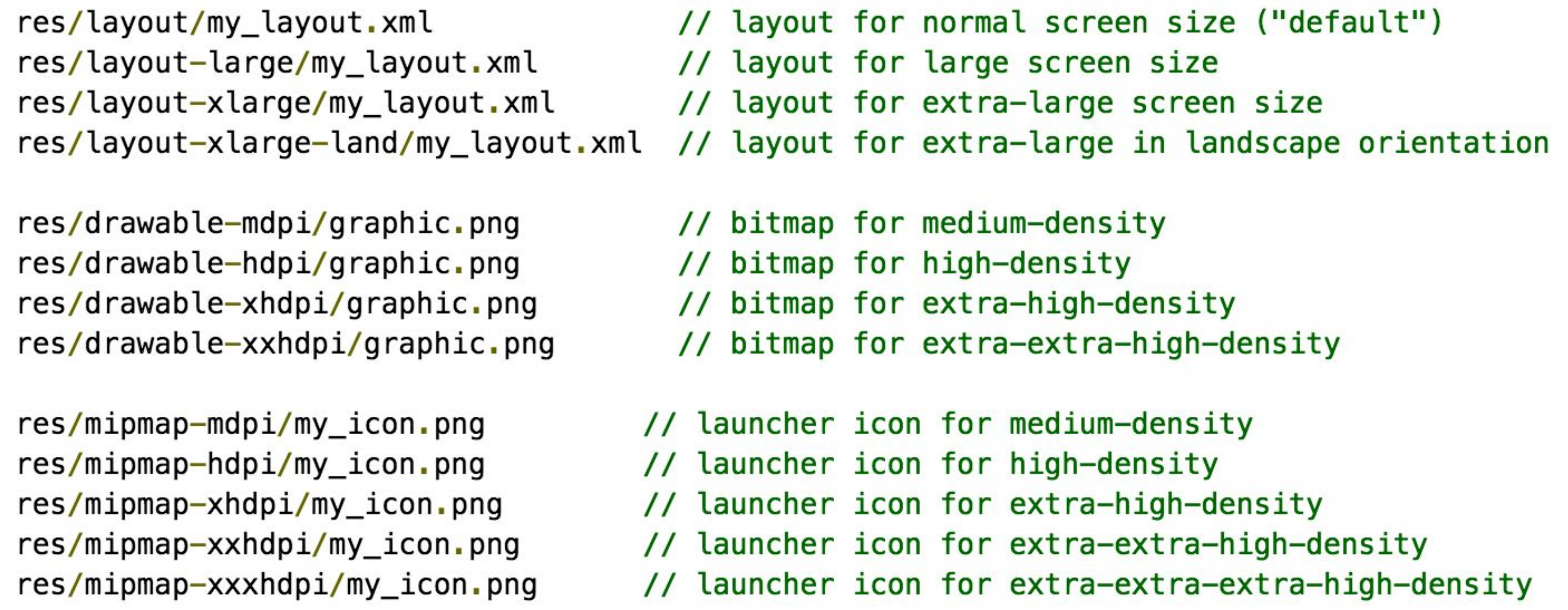


```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <TextView
        android:id="@+id/label"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Type here:"/>
    <EditText
        android:id="@+id/entry"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/label"/>
    <Button
        android:id="@+id/ok"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/entry"
        android:layout_alignParentRight="true"
        android:layout_marginLeft="10dp"
        android:text="OK" />
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_toLeftOf="@id/ok"
        android:layout_alignTop="@id/ok"
        android:text="Cancel" />
</RelativeLayout>
```

Gérer les tailles - Layouts

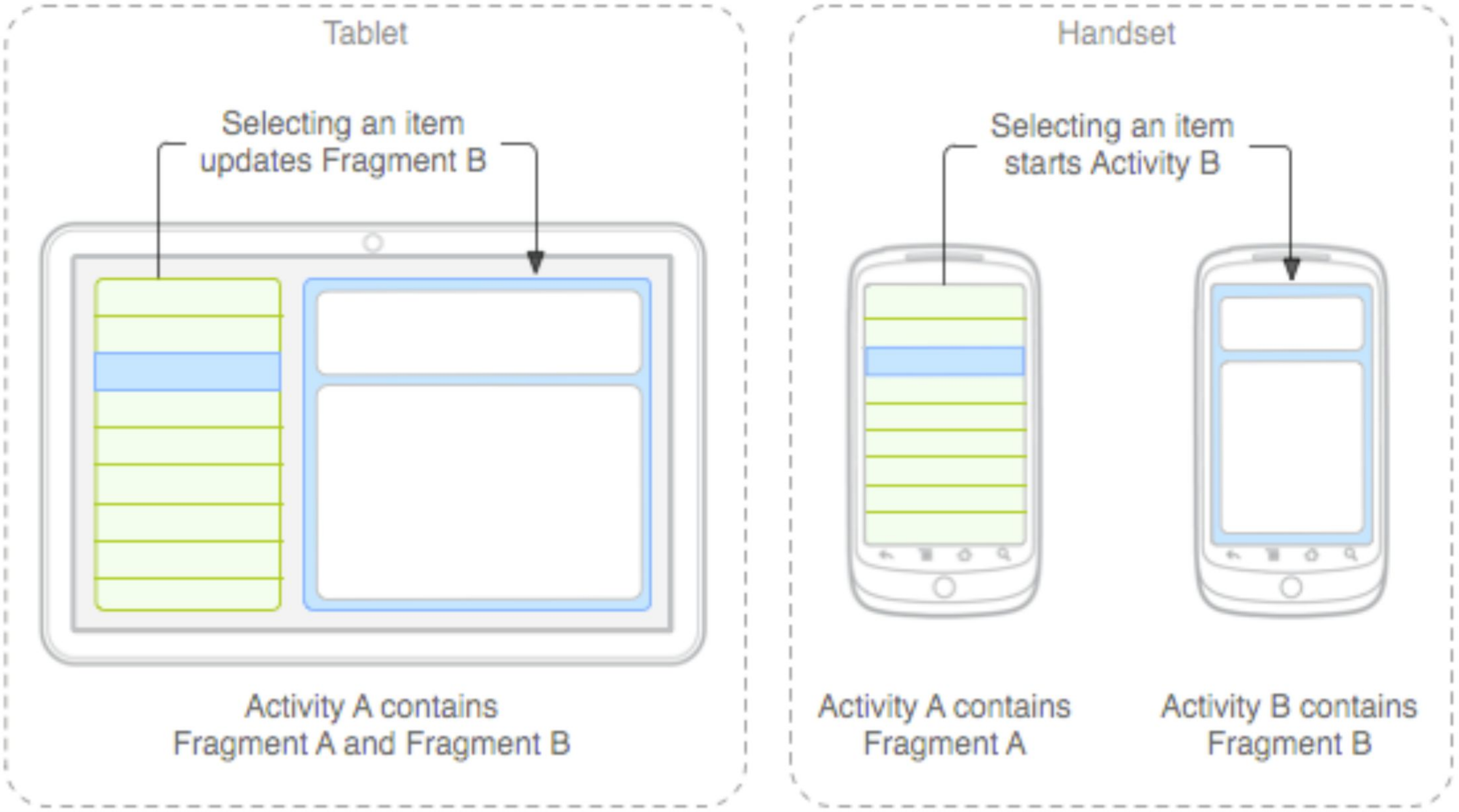
Les qualifiers

Objectifs: utiliser différents layout/*.xml en fonction du média



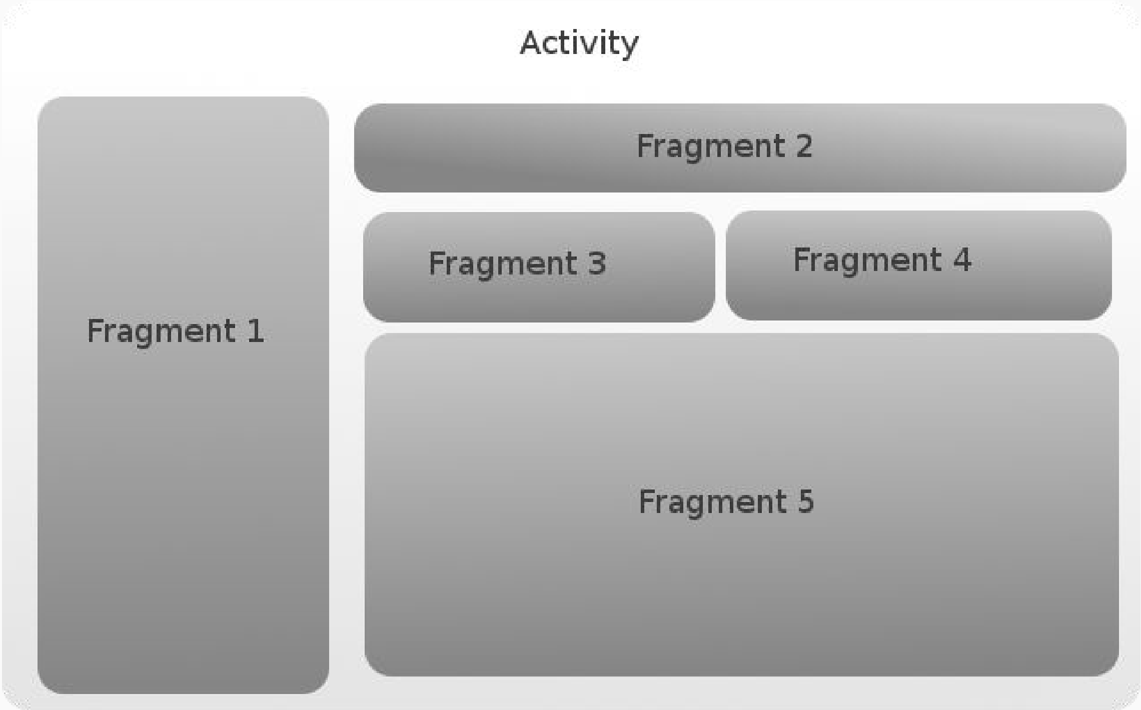
```
res/layout/my_layout.xml              // layout for normal screen size ("default")
res/layout-large/my_layout.xml        // layout for large screen size
res/layout-xlarge/my_layout.xml       // layout for extra-large screen size
res/layout-xlarge-land/my_layout.xml  // layout for extra-large in landscape orientation

res/drawable-mdpi/graphic.png         // bitmap for medium-density
res/drawable-hdpi/graphic.png         // bitmap for high-density
res/drawable-xhdpi/graphic.png        // bitmap for extra-high-density
res/drawable-xxhdpi/graphic.png       // bitmap for extra-extra-high-density

res/mipmap-mdpi/my_icon.png         // launcher icon for medium-density
res/mipmap-hdpi/my_icon.png         // launcher icon for high-density
res/mipmap-xhdpi/my_icon.png        // launcher icon for extra-high-density
res/mipmap-xxhdpi/my_icon.png       // launcher icon for extra-extra-high-density
res/mipmap-xxxhdpi/my_icon.png      // launcher icon for extra-extra-extra-high-density
```

Exemple


Tablet
Selecting an item updates Fragment B
Activity A contains Fragment A and Fragment B
Handset
Selecting an item starts Activity B
Activity A contains Fragment A
Activity B contains Fragment B

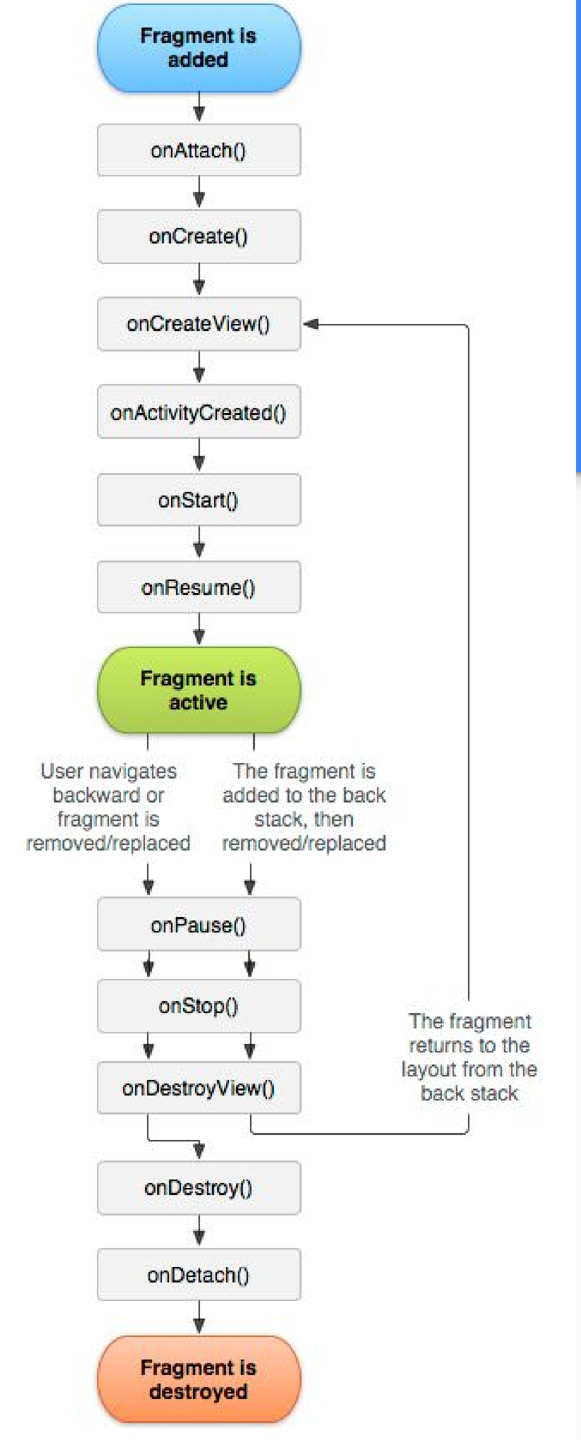
Les fragments

Découpage en composants fonctionnels, indépendants et réutilisables

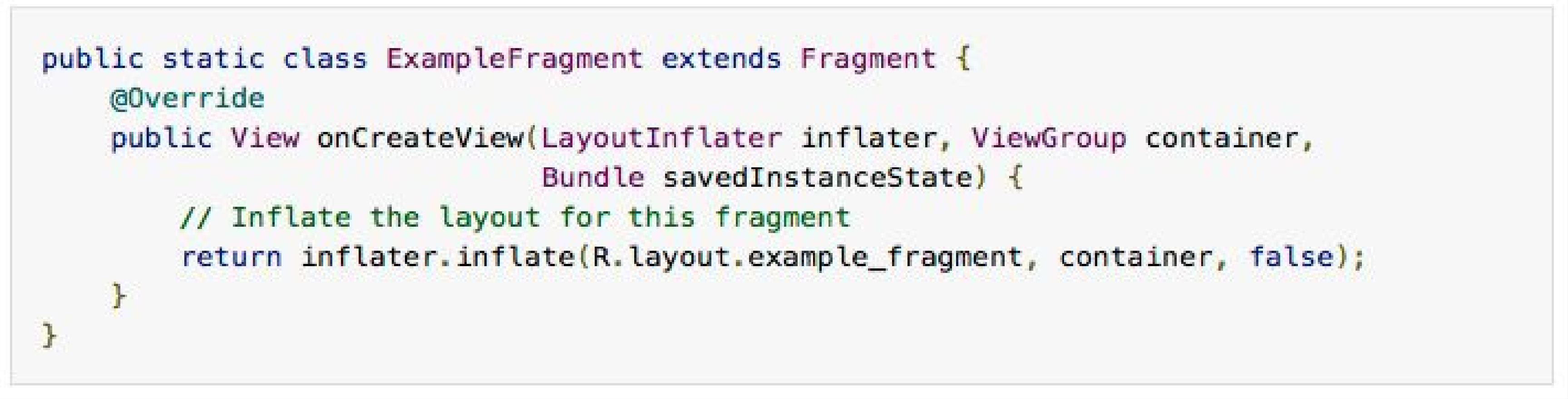


Les fragments

- Widget
  - Sous partie d’une activité
- Cycle de vie dédié
- API Level 11
  - Android lib support v4
- Un fichier layout dédié
- Extends Fragment
- Override onCreateView
- Dans le xml, utiliser <Fragment class=« »/>

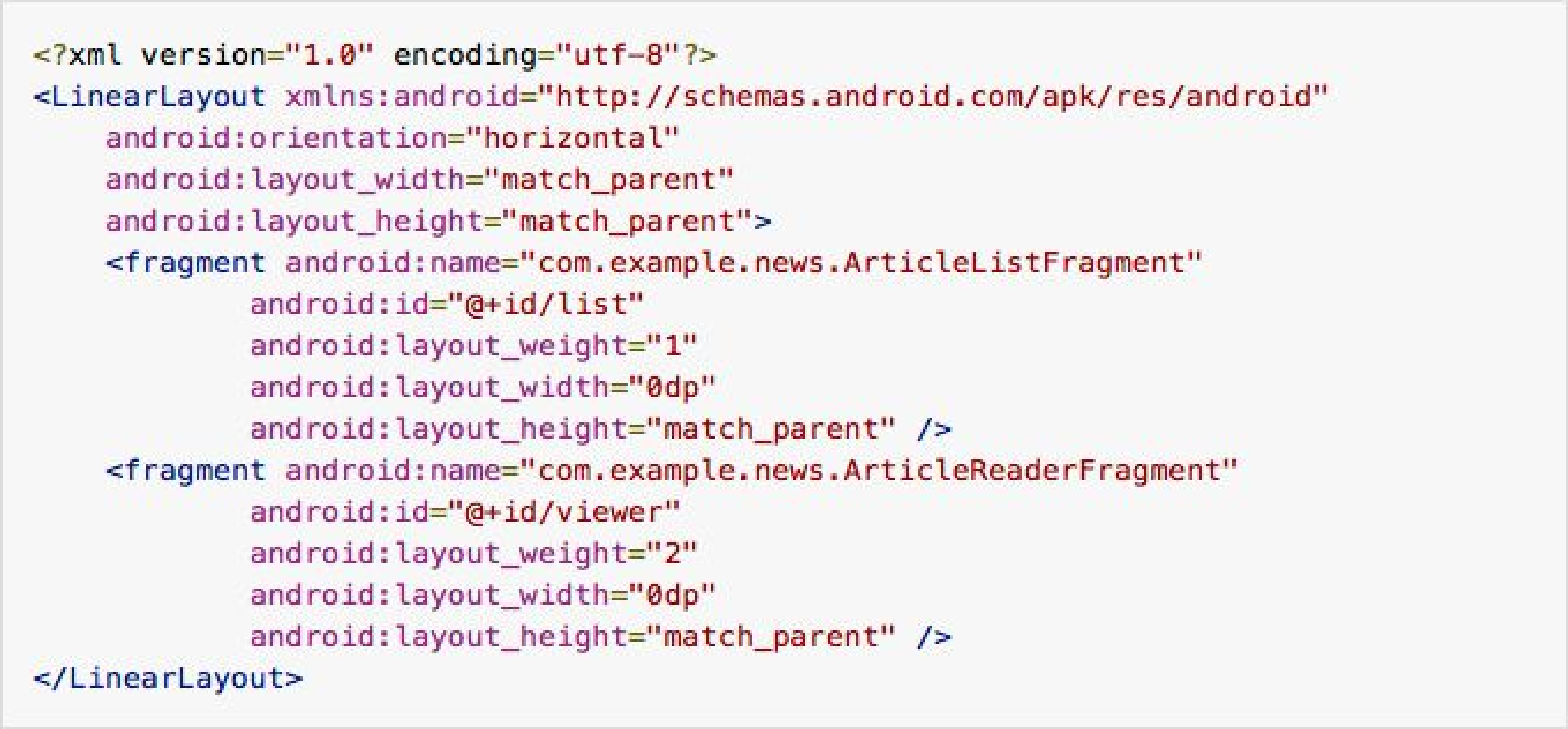


Exemple



```
public static class ExampleFragment extends Fragment {
    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        return inflater.inflate(R.layout.example_fragment, container, false);
    }
}
```

Exemple



```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="horizontal"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <fragment android:name="com.example.news.ArticleListFragment"
            android:id="@+id/list"
            android:layout_weight="1"
            android:layout_width="0dp"
            android:layout_height="match_parent" />
    <fragment android:name="com.example.news.ArticleReaderFragment"
            android:id="@+id/viewer"
            android:layout_weight="2"
            android:layout_width="0dp"
            android:layout_height="match_parent" />
</LinearLayout>
```

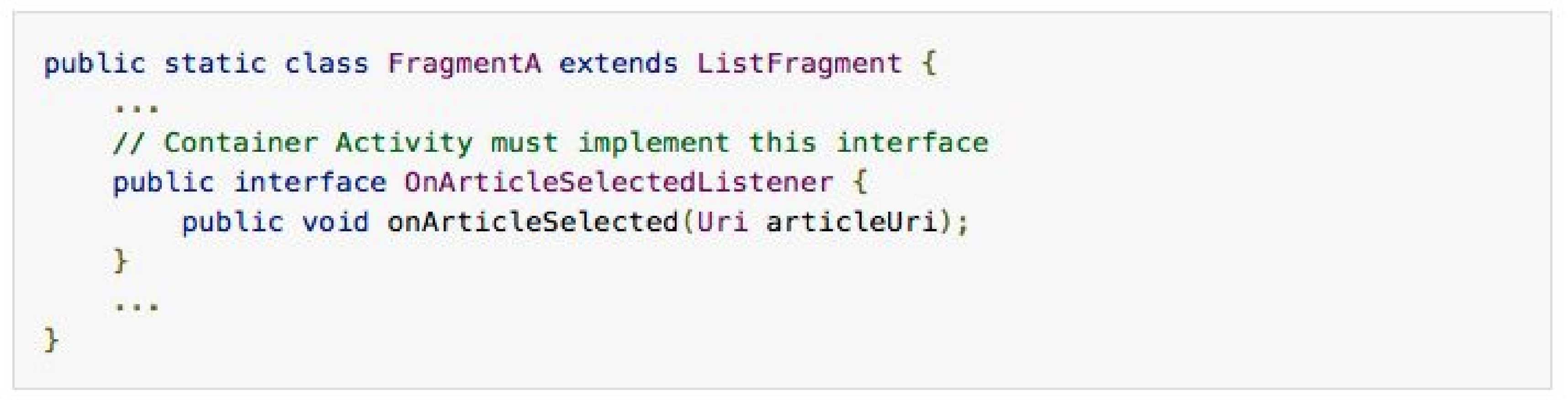
Communication inter-fragments

- Activity sert de proxy
  - FragmentA -> activity -> FragmentB
  - Version officielle
- Event bus
  - Version officieuse
  - Plus simple mais utilisation de lib tierce (Otto)
  - Perte en lisibilité

Communication inter-fragments

FragmentA -> activity -> FragmentB

1 : le fragment défini une callback
2 : L’activity implémente la callback

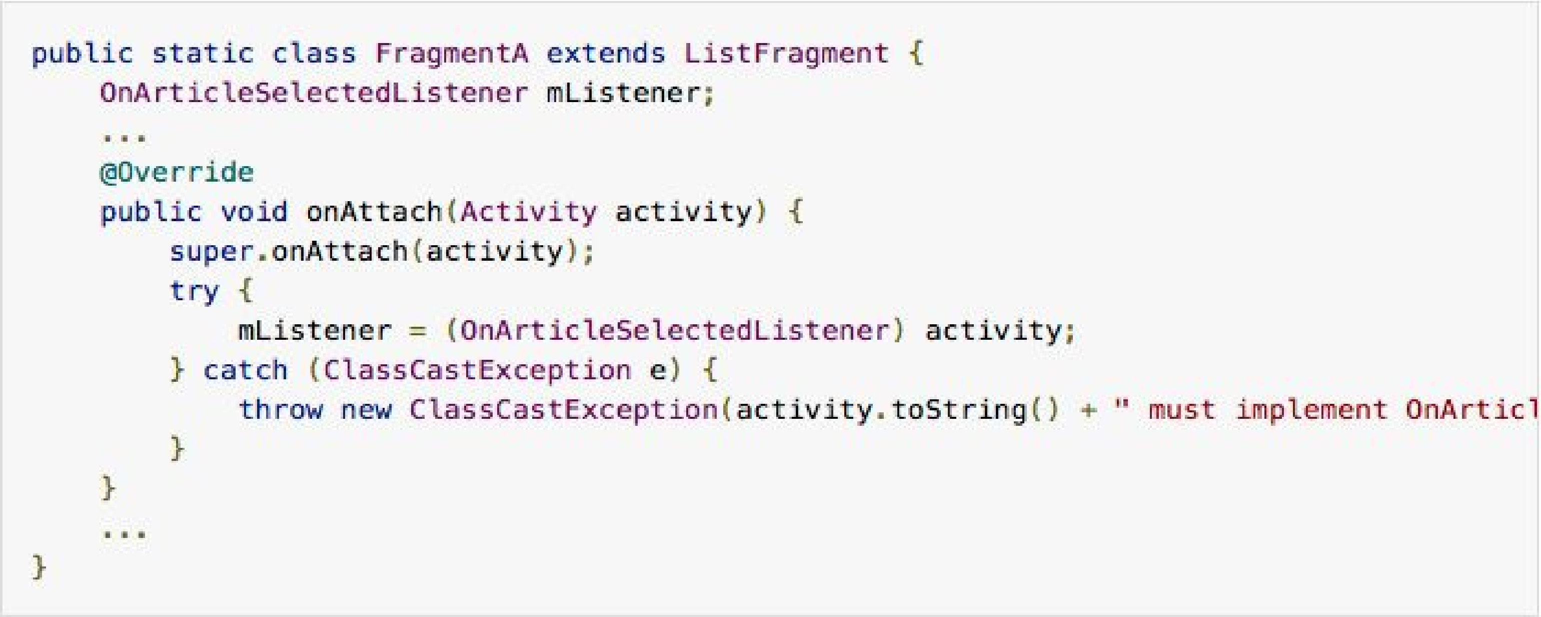


```
public static class FragmentA extends ListFragment {
    ...
    // Container Activity must implement this interface
    public interface OnArticleSelectedListener {
        public void onArticleSelected(Uri articleUri);
    }
    ...
}
```

Communication inter-fragments

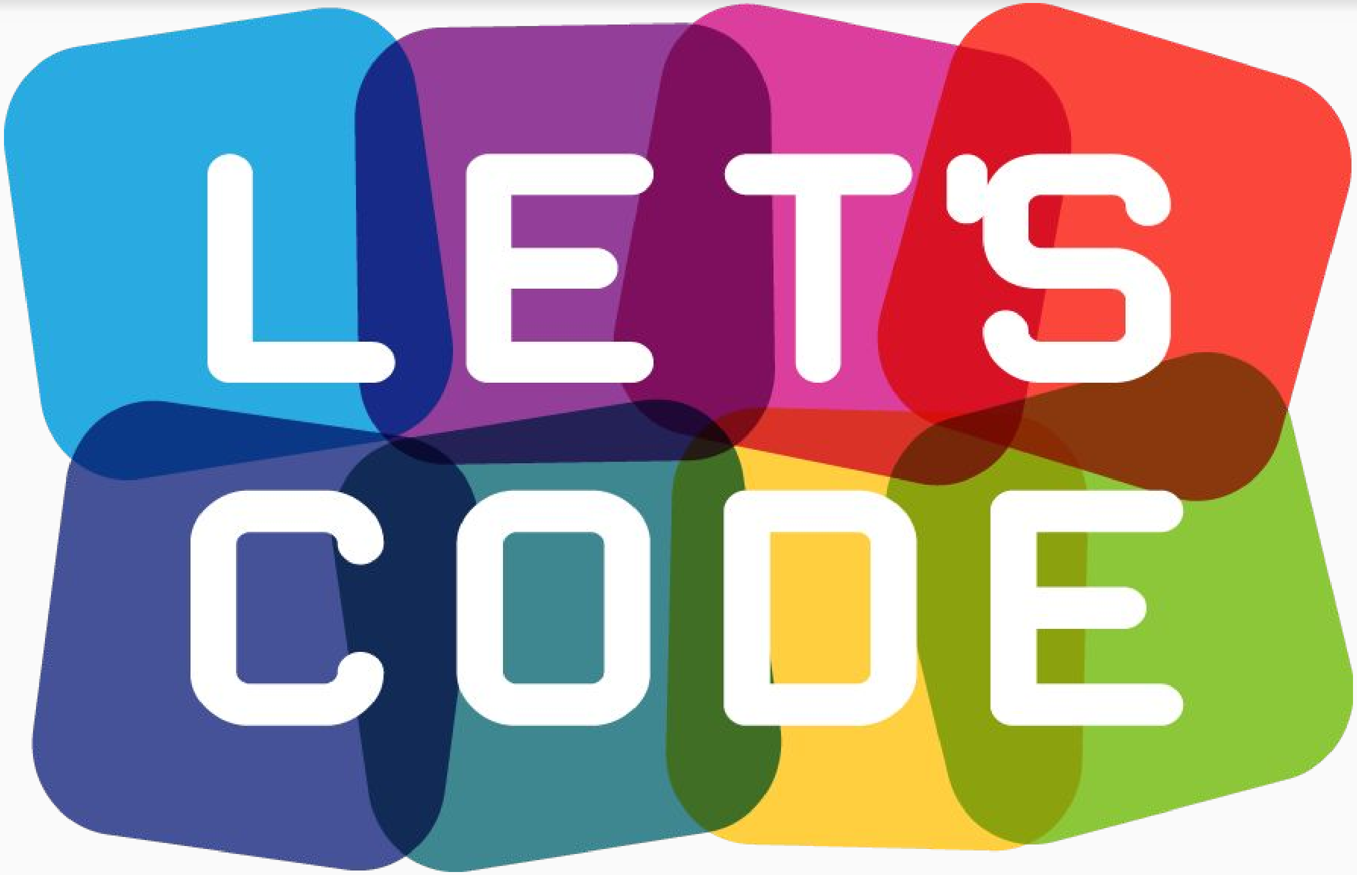
FragmentA -> activity -> FragmentB

3 : Le fragment attache la callback à l'activity

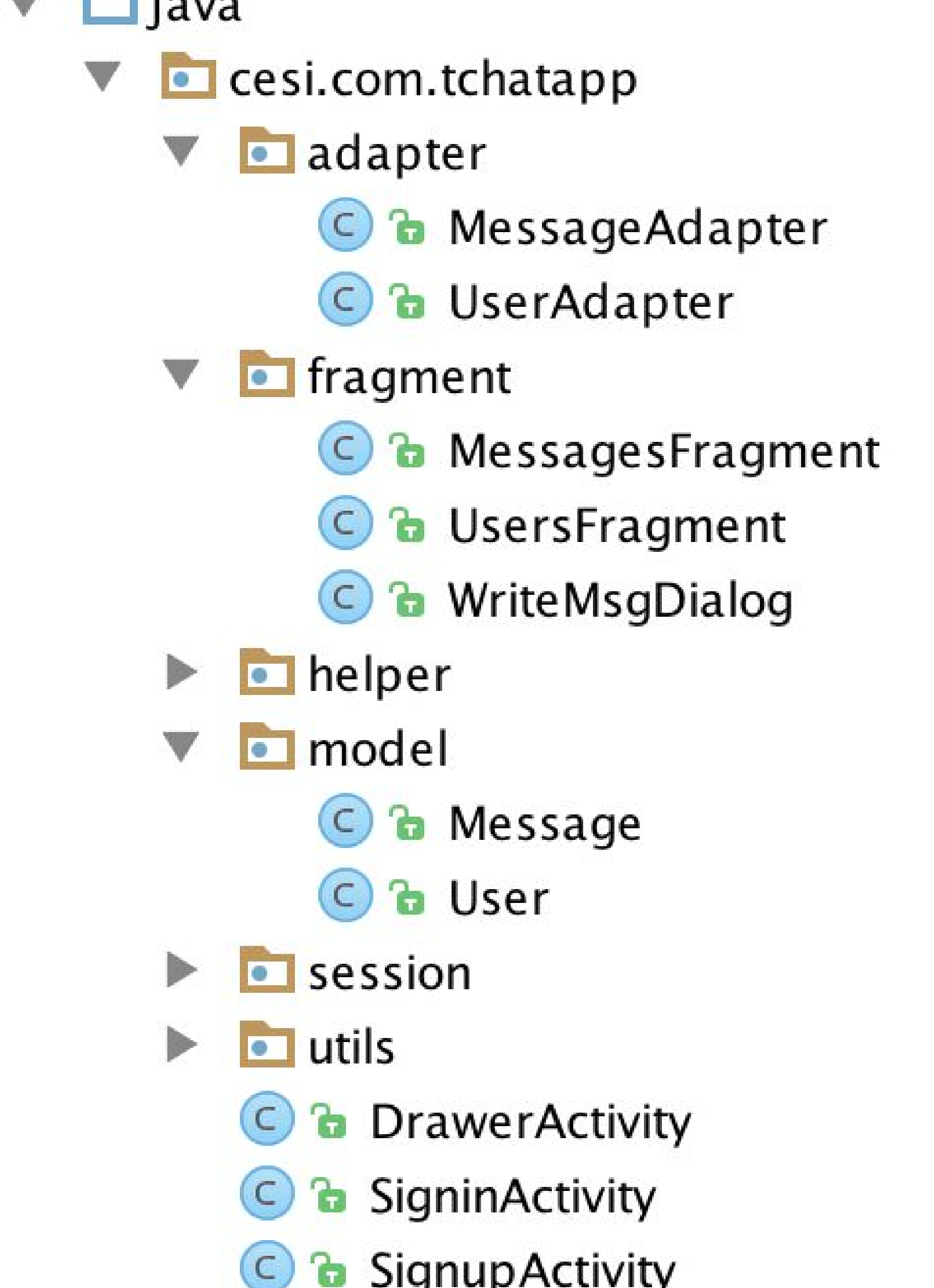


```
public static class FragmentA extends ListFragment {
    OnArticleSelectedListener mListener;
    ...
    @Override
    public void onAttach(Activity activity) {
        super.onAttach(activity);
        try {
            mListener = (OnArticleSelectedListener) activity;
        } catch (ClassCastException e) {
            throw new ClassCastException(activity.toString() + " must implement OnArticl
        }
    }
    ...
}
```

Bon, on code?

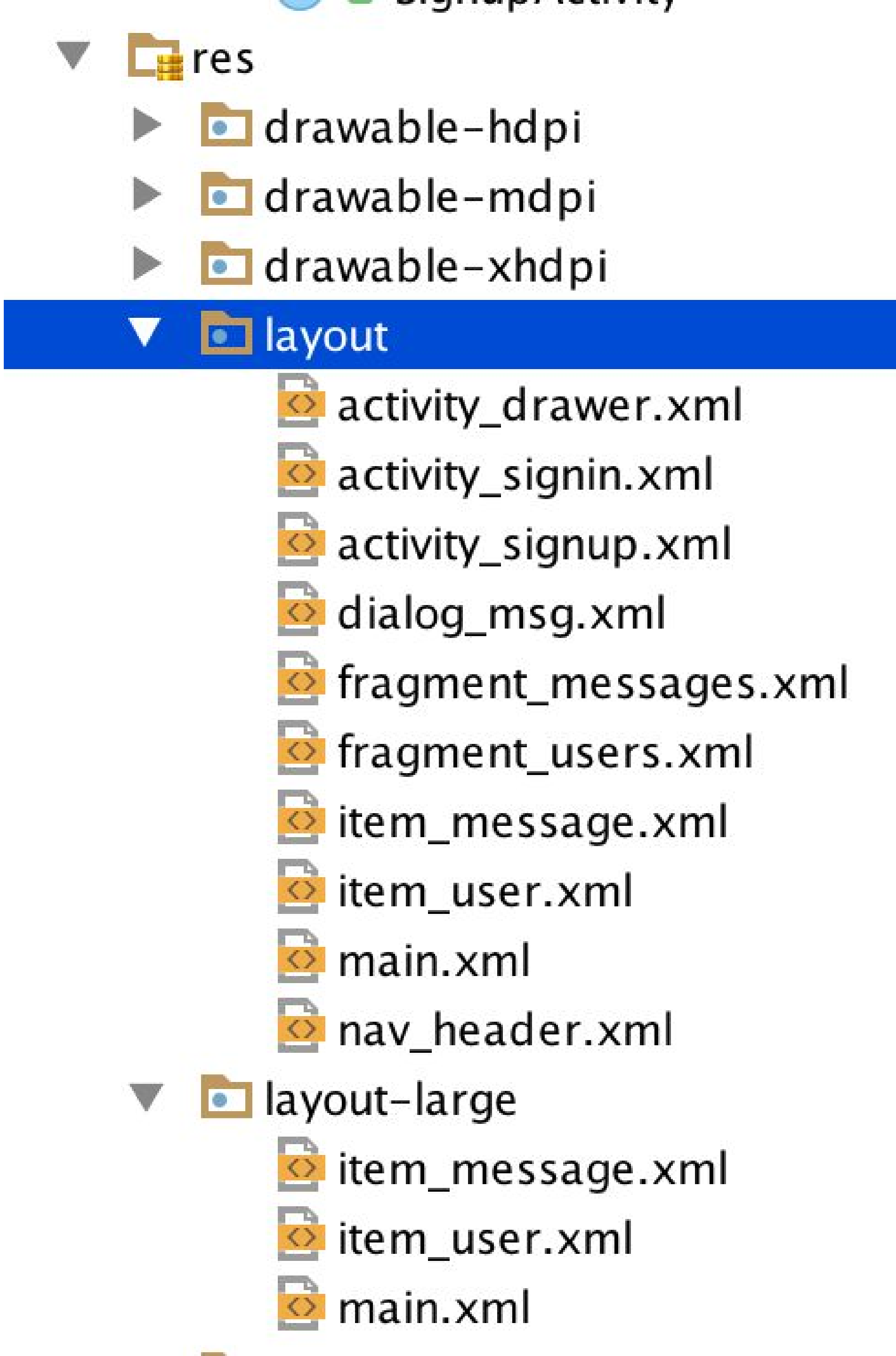


Etape 1:
passer notre app avec des fragments



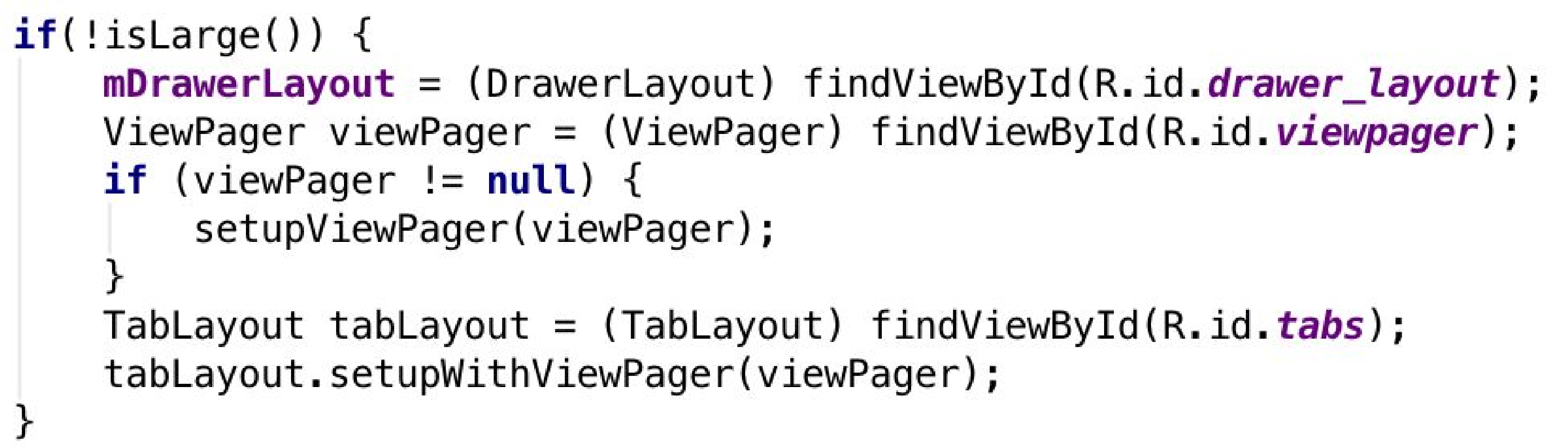
```
java
  cesi.com.tchatapp
    adapter
      MessageAdapter
      UserAdapter
    fragment
      MessagesFragment
      UsersFragment
      WriteMsgDialog
    helper
    model
      Message
      User
    session
    utils
    DrawerActivity
    SigninActivity
    SignupActivity
```

Etape 2:
Créer des layouts pour les tablettes



```
    SignupActivity
▼ res
  ▶ drawable-hdpi
  ▶ drawable-mdpi
  ▶ drawable-xhdpi
  ▼ layout
      activity_drawer.xml
      activity_signin.xml
      activity_signup.xml
      dialog_msg.xml
      fragment_messages.xml
      fragment_users.xml
      item_message.xml
      item_user.xml
      main.xml
      nav_header.xml
  ▼ layout-large
      item_message.xml
      item_user.xml
      main.xml
```

Etape 3:
Utiliser les composants adéquates



```java
if(!isLarge()) {
    mDrawerLayout = (DrawerLayout) findViewById(R.id.drawer_layout);
    ViewPager viewPager = (ViewPager) findViewById(R.id.viewpager);
    if (viewPager != null) {
        setupViewPager(viewPager);
    }
    TabLayout tabLayout = (TabLayout) findViewById(R.id.tabs);
    tabLayout.setupWithViewPager(viewPager);
}
```