Layout

Les Layout sont des composants de l'interface graphique qui n'ont pas de rendu à l'écran, mais vont décider comme les composants qu'ils contiennent vont être disposés à l'écran. Il en existe de très nombreuses variantes, mais nous allons étudier les trois principales qui sont les plus régulièrement utilisées.

Pour positionner un élément dans un Layout, deux propriétés XML sont obligatoire : `layout_width` et `layout_height`. Elle peuvent prendre des valeur numériques de taille ou alors :

- `fill_parent` : Remplir l'élément parent
- `wrap_content` : S'adapter à la taille du contenu

Linear Layout

Le Linear Layout positionne les éléments les uns à la suites des autres en ligne ou en colonne selon l'orientation. On peut donner un poids au éléments pour le permettre de s'étendre plus ou moins par rapport aux autres.

Faisons par exmple un Linear Layout pour notre élément de liste de Todo :

```
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="92dp">


</LinearLayout>
```

On commence par définir un LinearLayout avec un largeur qui remplit toute la ligne et une hauteur qui correspond à la hauteur qu'on choisit pour une ligne.

On va ensuite rajouter le contenu :

```
    <TextView
        android:id="@+id/todo_list_item_title"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
```

```
        android:layout_weight="2"
        android:text="Todo Title"
        android:textSize="28dp"
        android:textFontWeight="700"
        >
    </TextView>

    <TextView
        android:id="@+id/todo_list_item_description"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="Todo Description"></TextView>
```

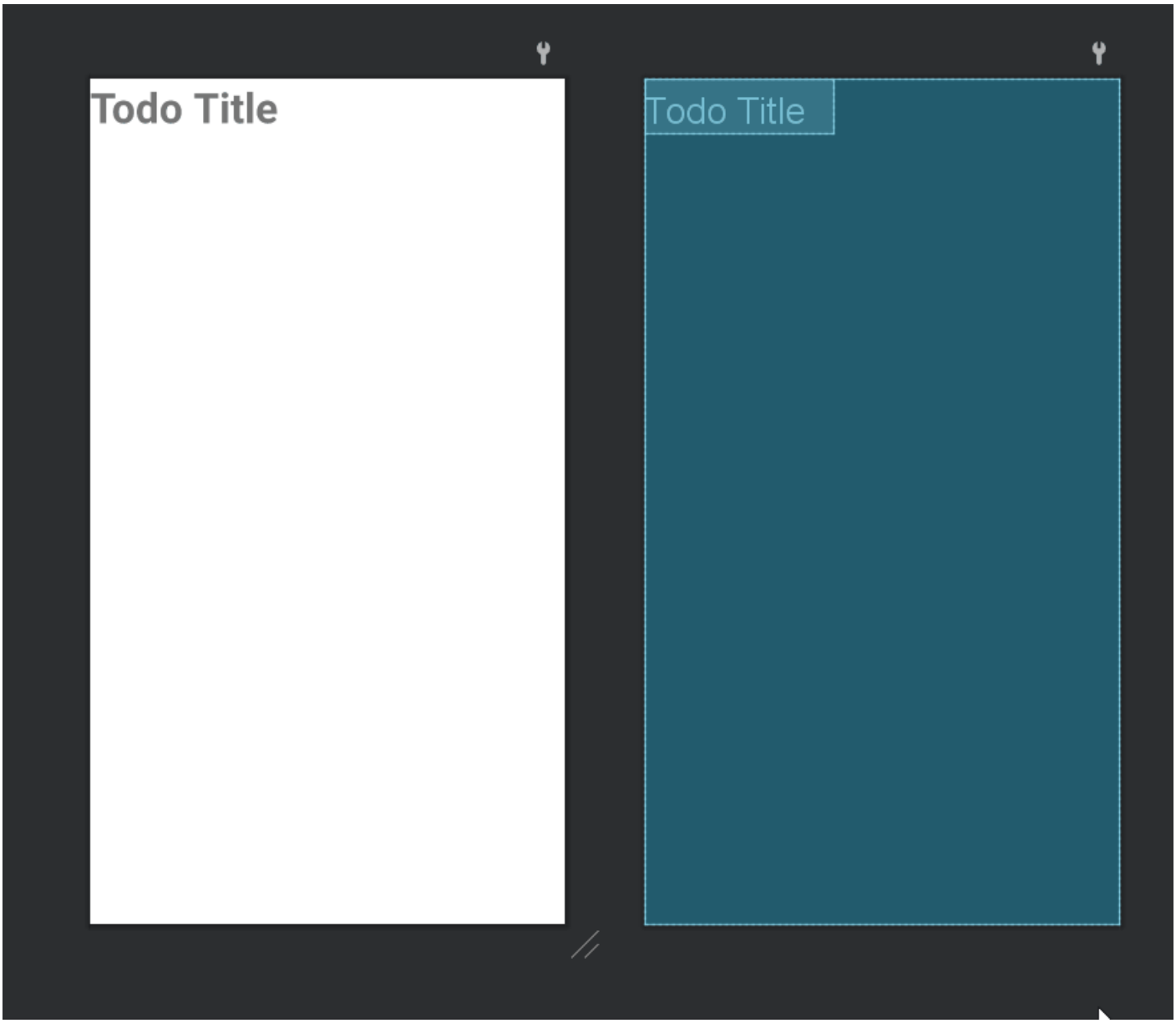
On donne au titre un poid double pour qu'il puisse prendre plus de place. Cela donne le résultat suivant :



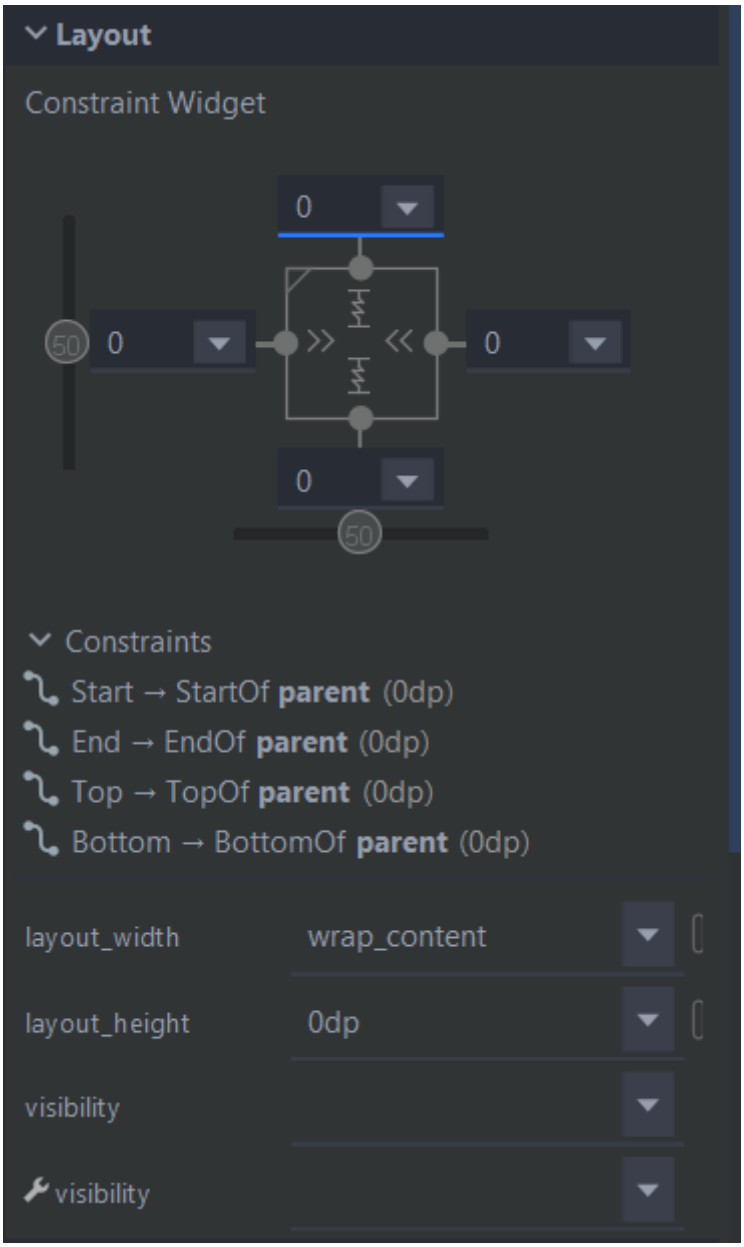
Contraint Layout

Le Constraint Layout permet de disposer les éléments en définissant des contraintes. On peut établir des contraintes de positionnement pour chaque controles par rapport aux autres éléments, au parent, par rapport au bord, etc ... Ce controle est très puissant notamment pour définir des Layouts responsive qui vont s'adapter à des téléphones de différentes dimensions.

Le Constraint Layout est fait pour être manipuler avec l'éditeur d'interface d'Android Studio. Nous allons l'utiliser pour concevoir l'interface de notre Todo Detail Fragment :

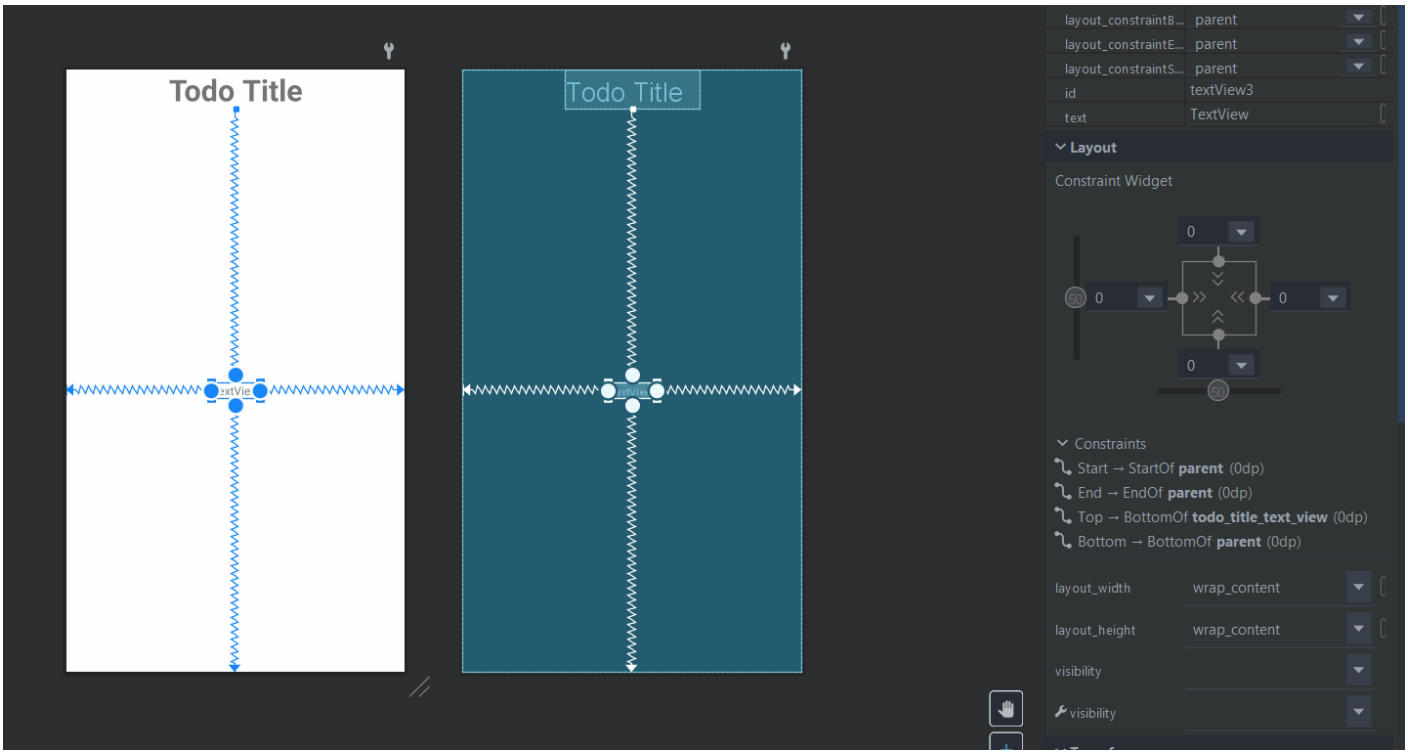


On utilise les connecteurs de contraintes sur l'éléments pour définir une contrainte par rapport à un autre élément (ici les bords). Il faut ensuite utiliser l'interface de Layout pour éditer les contraintes en changeant leur type ou leur longueur :

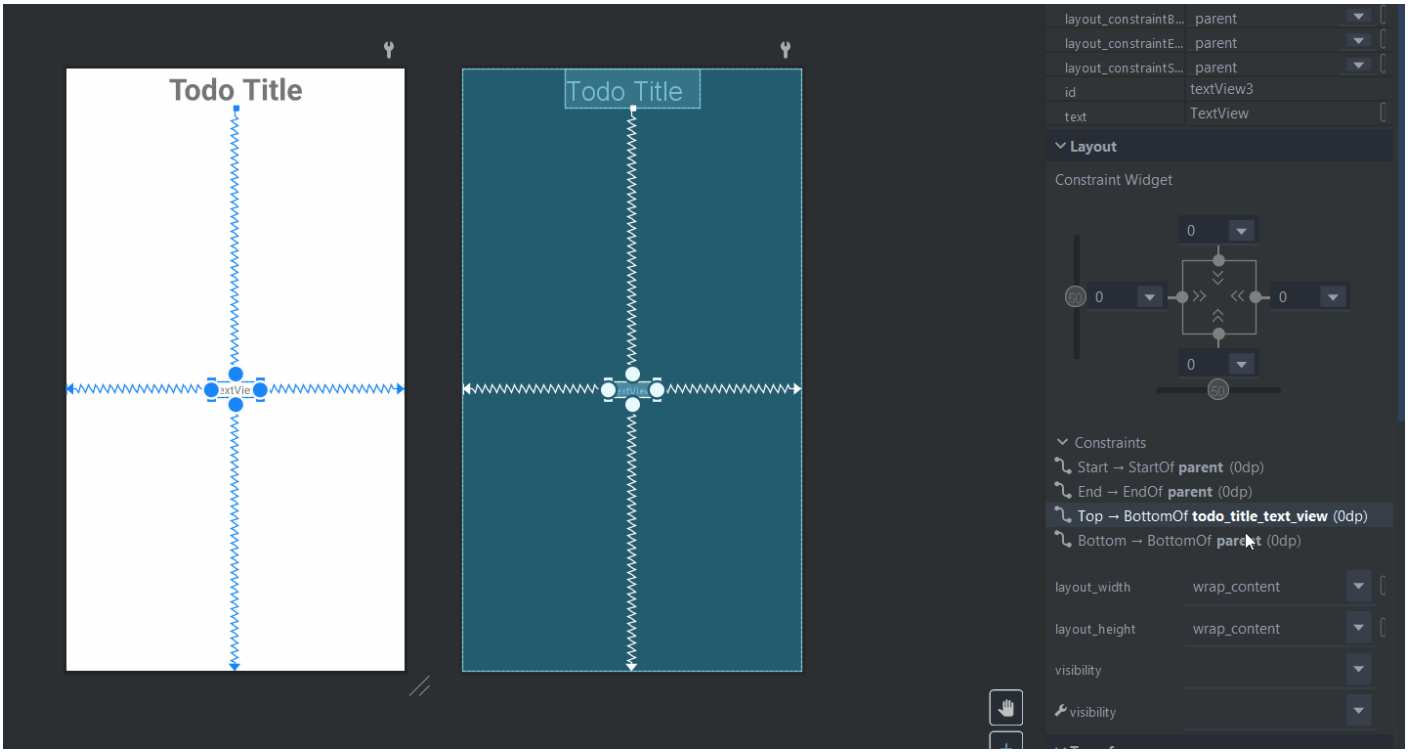


Sur les schéma des contraintes vous pouvez :

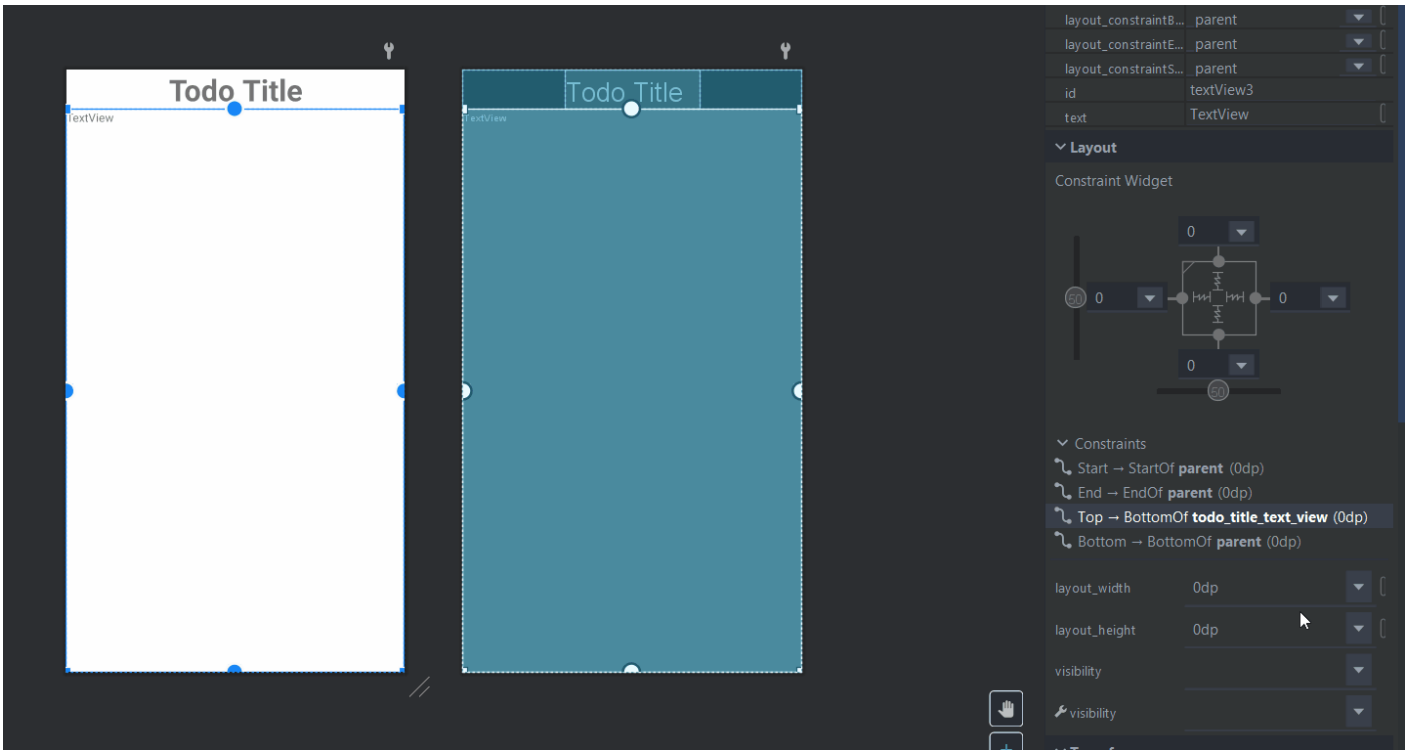
- Supprimer les contraintes en cliquant sur le rond :



- Changes le type de contrainte en cliquant sur le trait :



- Ajouter du margin sur la contrainte grace à la combo box :



Revision #6
Created 2021-01-31 14:43:06 UTC by Arsène Lapostolet
Updated 2021-02-07 17:15:57 UTC by Arsène Lapostolet