



ANDROID

An Open Handset Alliance Project

Intent implicites et explicites

Prendre des photos

G. Seront

Permissions caméra

- Dans le manifest.xml

```
<manifest ... >
    <uses-feature android:name="android.hardware.camera"
                  android:required="true" />
    ...
</manifest>
```

- Note: camera deprecated et remplacée par camera2 dans API 21 (Lollipop)




Intent implicite

- On va dire ce qu'on veut, pas comment on le fait

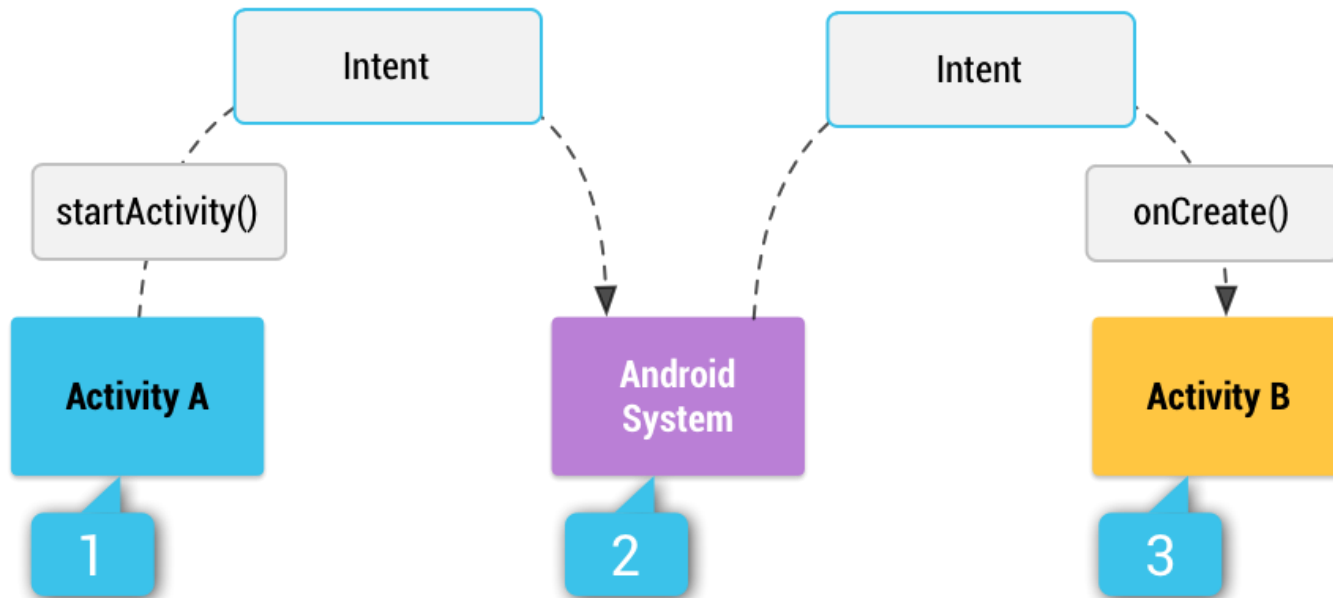
```
static final int REQUEST_IMAGE_CAPTURE = 1;

private void dispatchTakePictureIntent() {
    Intent takePictureIntent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
    if (takePictureIntent.resolveActivity(getPackageManager()) != null) {
        startActivityForResult(takePictureIntent, REQUEST_IMAGE_CAPTURE);
    }
}
```




Intent implicite

- Le système se charge de trouver la bonne activité installée sur le téléphone



Intent Filter

- Votre activité peut dire dans le manifest qu'elle traite certains intents implicites :

```
<activity android:name="ShareActivity">
    <!-- This activity handles "SEND" actions with text data -->
    <intent-filter>
        <action android:name="android.intent.action.SEND"/>
        <category android:name="android.intent.category.DEFAULT"/>
        <data android:mimeType="text/plain"/>
    </intent-filter>
```



Intent Filter

- Vous avez déjà utilisé les filtres sans le savoir:

```
<activity
    android:name=".views.JoueurDetails"
    android:label="JoueurTennis" >
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>
```



Intent standards

- Il existe une liste complète d'intents standards (voir [ref](#)).



Récupération du résultat

- Après la prise d'image, l'activité appelante est notifiée:

```
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    if (requestCode == REQUEST_IMAGE_CAPTURE && resultCode == RESULT_OK) {
        Bundle extras = data.getExtras();
        Bitmap imageBitmap = (Bitmap) extras.get("data");
        mImageView.setImageBitmap(imageBitmap);
    }
}
```



Sauvegarde de l'image

```
<manifest ...>
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    ...
</manifest>
```

```
File storageDir = Environment.getExternalStoragePublicDirectory(
    Environment.DIRECTORY_PICTURES);
File image = File.createTempFile(
    imageFileName, /* prefix */
    ".jpg",        /* suffix */
    storageDir     /* directory */
);
```

```
// Save a file: path for use with ACTION_VIEW intents
mCurrentPhotoPath = "file:" + image.getAbsolutePath();
```



Intent explicite

- On dit qui on veut
- Une activité précise

```
public void sendMessage(View view) {
    Intent intent = new Intent(this, DisplayMessageActivity.class);
}
```

- Mais aussi un autre composant applicatif

```
// Executed in an Activity, so 'this' is the Context
// The fileUrl is a string URL, such as "http://www.example.com/image.png"
Intent downloadIntent = new Intent(this, DownloadService.class);
downloadIntent.setData(Uri.parse(fileUrl));
startService(downloadIntent);
```



Intent explicite: passer de l'information

- Dans l'activité appelante

```
monIntent.putExtra("UneValeur", 36);
```

- Dans le onCreate de l'activité appelée

```
Intent intent = getIntent();
int valeur = intent.getIntExtra("UneValeur", 0);
```

- Best practice: définir une constante dans l'activité appelée

```
public static final String UNE_VALEUR = "UneValeur";
```

