

Android

Activity - Intent

Jean-Marc Lecarpentier
Université de Caen

Au programme

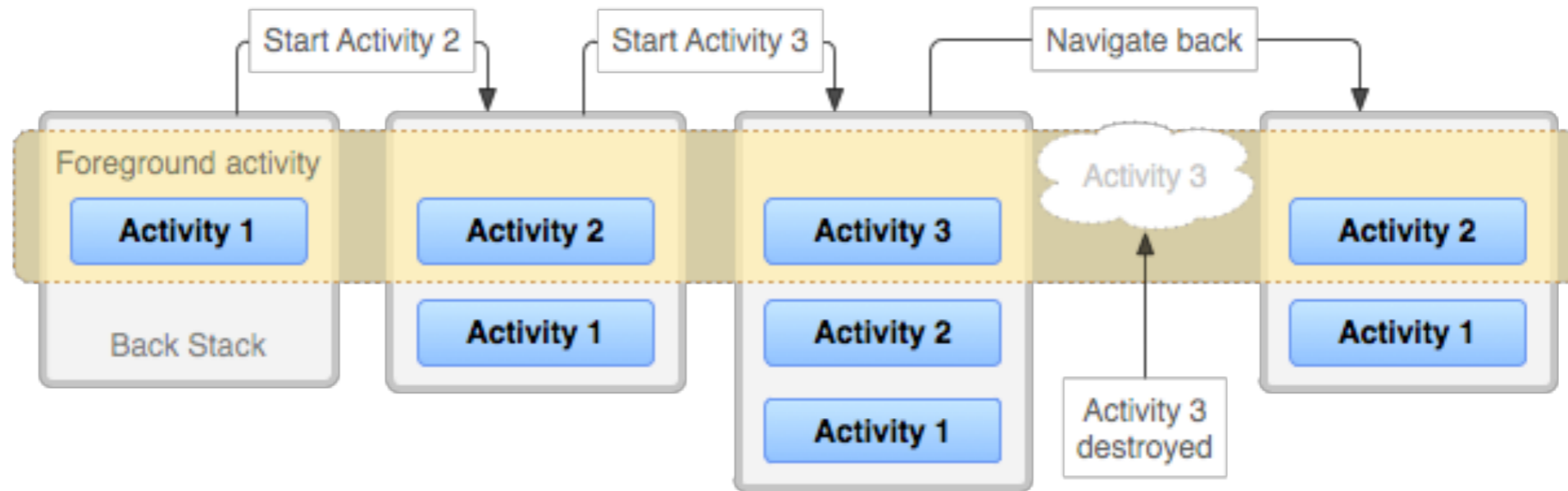
- Cycle de vie de Activity et leur gestion
- Intent : propriétés, Intent implicite ou explicite
- Gestion des intent-filters pour décider de quelle Activity déclencher



Activity

- Activity = écran visible à l'utilisateur
- Une seule Activity "active" à la fois
- Gestion de l'état d'une activité
- Gestion de la pile des Activity

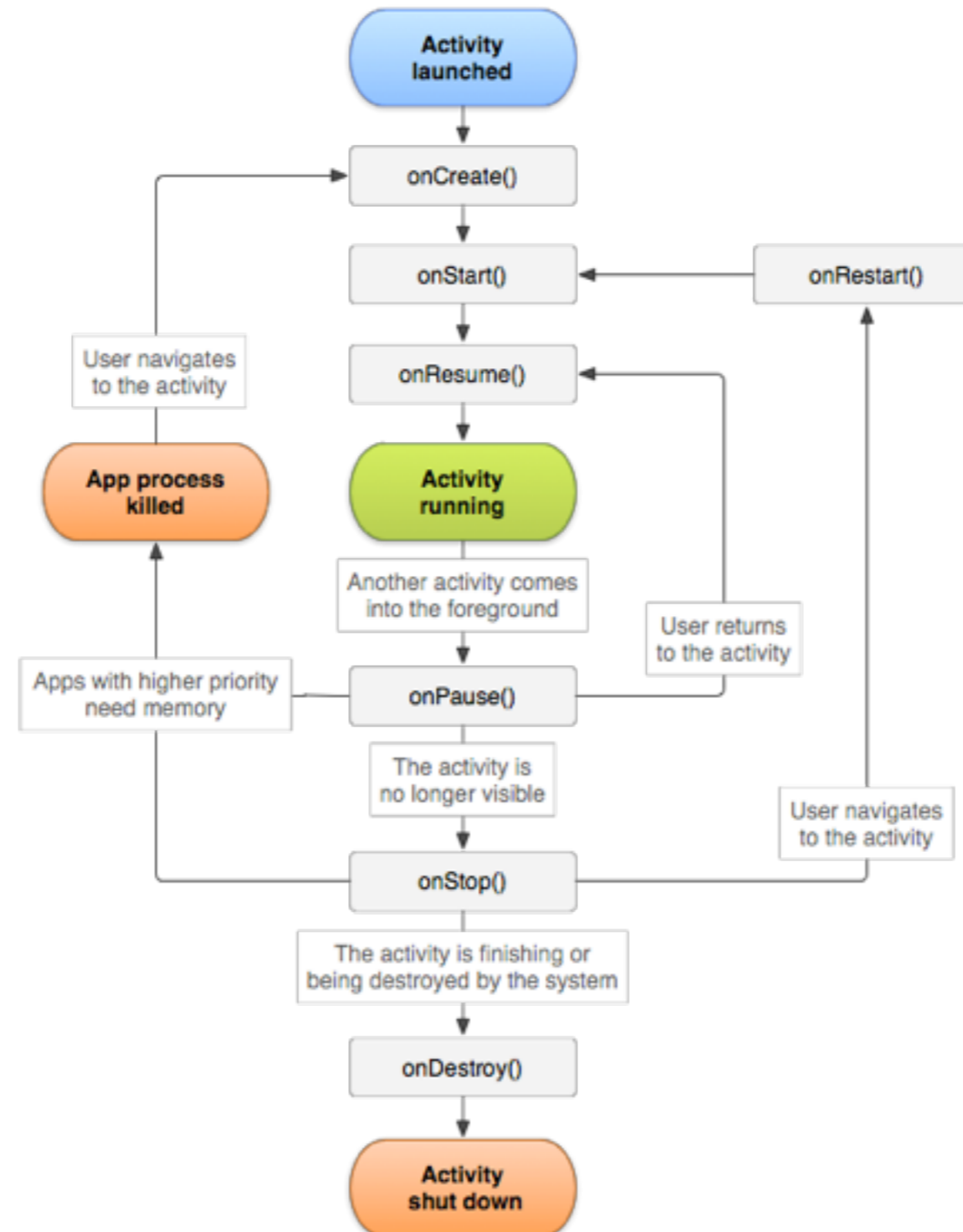
Pile



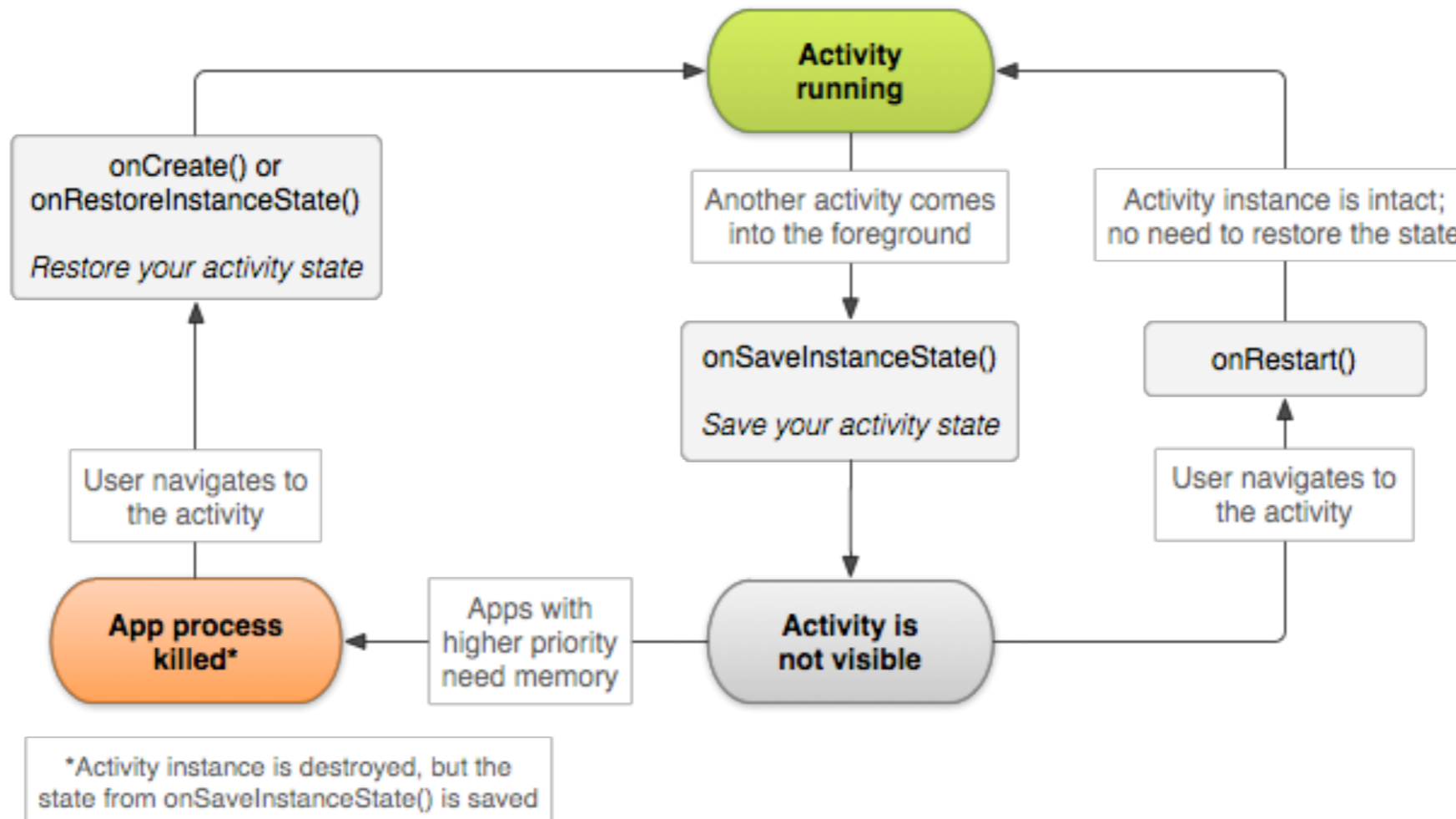
États

- Resumed (ou running) : focus utilisateur
- Paused : partiellement visible mais pas en haut de la pile car caché par une autre Activity
- Stopped : passée an arrière-plan, inactive mais toujours existante en mémoire

Cycle de vie et *callbacks*



Enregistrer un état



Activity A à B

1. Activity A : onPause()
2. Activity B : onCreate(), onStart(), onResume()
3. Activity A : onStop()

Attention à gérer les actions dans onPause() !

Activity et action

- Activity définie dans le Manifest
- Associée à un nom

```
<activity android:name=".BasicViewActivity" android:label="Basic View Tests">  
<intent-filter>  
    <action android:name="com.toto.intent.action.ShowBasicView"/>  
    <category android:name="android.intent.category.DEFAULT" />  
</intent-filter>  
</activity>
```

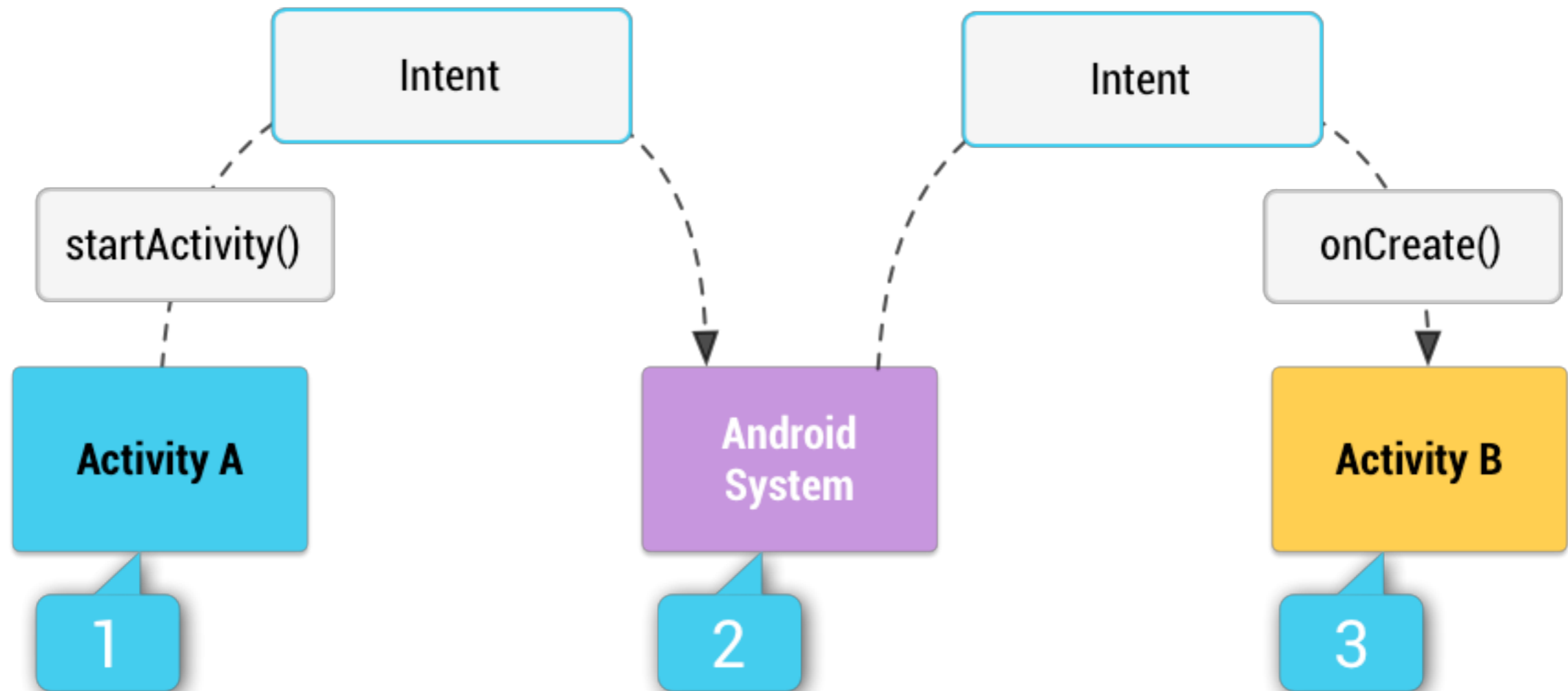
Intent

- Objet servant à démarrer Activity, Service, etc
- Associé à des données
- Intent prédéfinies et Intent de l'application

Activer des Intent prédéfinies

```
public class IntentsUtils {
    public static void invokeWebBrowser(Activity activity) {
        Intent intent = new Intent(Intent.ACTION_VIEW);
        intent.setData(Uri.parse("http://www.google.com"));
        activity.startActivity(intent);
    }
    public static void invokeWebSearch(Activity activity) {
        Intent intent = new Intent(Intent.ACTION_WEB_SEARCH);
        intent.setData(Uri.parse("http://www.google.com"));
        activity.startActivity(intent);
    }
    public static void dial(Activity activity) {
        Intent intent = new Intent(Intent.ACTION_DIAL);
        activity.startActivity(intent);
    }
    public static void call(Activity activity) {
        Intent intent = new Intent(Intent.ACTION_CALL);
        intent.setData(Uri.parse("tel:555-555-5555"));
        activity.startActivity(intent);
    }
    public static void showMapAtLatLng(Activity activity) {
        Intent intent = new Intent(Intent.ACTION_VIEW);
        //geo:lat,long?z=zoomlevel&q=question-string
        intent.setData(Uri.parse("geo:0,0?z=4&q=business+near+city"));
        activity.startActivity(intent);
    }
    public static void tryOneOfThese(Activity activity) {
        IntentsUtils.invokeWebBrowser(activity);
    }
}
```

Intent implicite

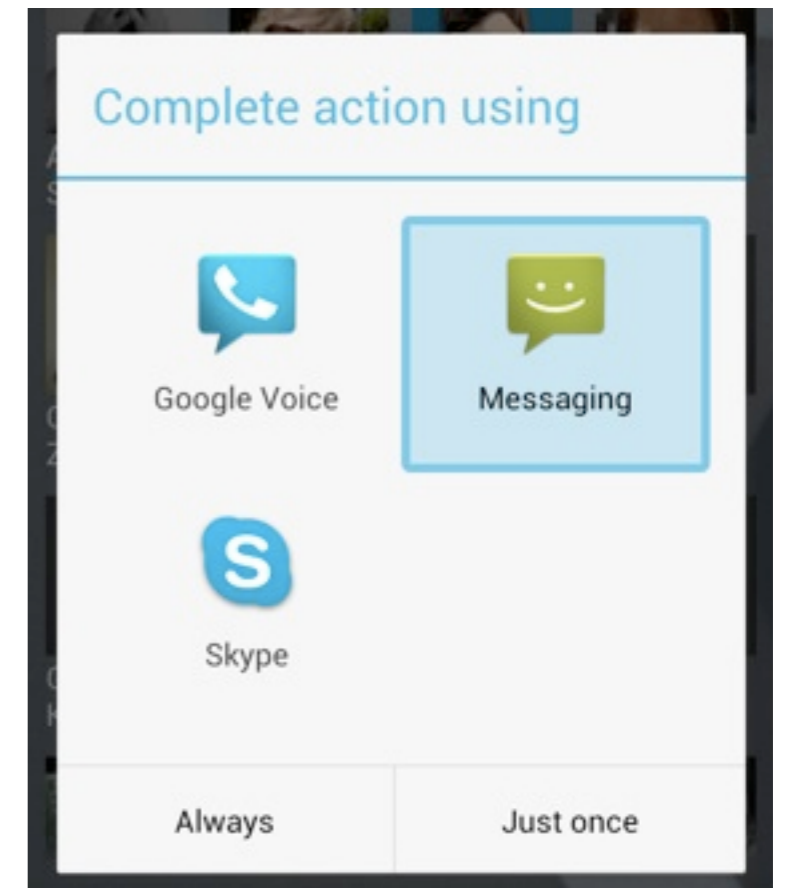


Propriétés de Intent

- action : action à exécuter (ACTION_VIEW, etc)
- data : données pour réaliser l'action, par ex. un n° de tél. pour ACTION_CALL. Sous la forme d'un *Data URI* : ex. tel:0231456543 ou pour voir un contact content://contacts/people/1
- category : informations complémentaires
- type : spécifie un type MIME (géré par l'objet Intent)
- component : spécifier le composant à invoquer (Intent explicite)
- extras : ensemble d'informations complémentaires

Intent implicite

- Android va regarder quelles activités peuvent répondre à l'action demandée
- Si plusieurs possibilités, demander à l'utilisateur
- Noms d'actions génériques : ACTION_SEND, etc
- Utiliser pour lancer des actions gérées en dehors de l'application elle-même



Vérifier Intent

- Créer une Intent implicite

```
Uri number = Uri.parse("tel:5551234");  
Intent callIntent = new Intent(Intent.ACTION_DIAL, number);
```

```
webpage = Uri.parse("http://www.android.com");  
Intent webIntent = new Intent(Intent.ACTION_VIEW, webpage);
```

- Question : existe-t-il une Activity pour répondre ?

```
PackageManager packageManager = getPackageManager();  
List<ResolveInfo> activities = packageManager.queryIntentActivities(intent, 0);  
boolean isIntentSafe = activities.size() > 0;
```

Intent explicite

- Intent demande une action spécifique d'un package défini
- Utilisé à l'extérieur d'une application

```
public static void invokeMyApplication(Activity act) {  
    String actionName= "com.toto.intent.action.ShowBasicView";  
    Intent intent = new Intent(actionName);  
    act.startActivity(intent);  
}
```

- Utilisé à l'intérieur d'une application

```
public void lancerActivite(View v) {  
    Intent intent = new Intent(this, AfficherMessage.class);  
    String message = txtSaisie.getText().toString().trim();  
    intent.putExtra(MON_MESSAGE, message);  
    startActivity(intent);  
}
```


Intent-filters

- Définis pour les objets Activity dans le Manifest

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

- action : définit les actions auxquelles l'Activity peut répondre
ACTION_MAIN précise que c'est le point d'entrée de l'application
- category : type de catégorie de l'Intent qui invoque
(CATEGORY_DEFAULT par défaut, CATEGORY_LAUNCHER pour un lanceur d'application, etc)
- data : type de données acceptées

Résolution des Intent

- Comment décider de l'Activity à démarrer pour une Intent implicite
- 3 critères : action, données, catégorie
- Détails <http://developer.android.com/guide/components/intents-filters.html>

En bref

- Gestion du cycle de vie d'une Activity
- Gestion de la pile des Activity
- Intent explicite vs. implicite
- Choix de l'Activity à démarrer selon l'action demandée
- Intent-filters