

Android componentes (II)

Enrique López-Mañas

Contenido

Componentes de una aplicación:

Actividades (Activities)

Intents

Intent Filters

Servicios

Content Providers

Broadcast Receivers

Content Providers

Compartir datos entre aplicaciones

Contactos

Navegador

Registro de llamadas

Ajustes

Archivos multimedia

Funciona como una base de datos

Para hacer consultas se usa una URI

Content Providers URI

CONSTANTE	DESCRIPCIÓN	STRING CONSULTA
MediaStore.Images.Media.INTERNAL_CONTENT_URI	Devuelve una lista con todas las imágenes internas del dispositivo	content://media/internal/images
MediaStore.Images.Media.EXTERNAL_CONTENT_URI	Devuelve una lista con todas las imágenes guardadas en el dispositivo externo (SD Card)	content://media/external/images
CallLog.CONTENT_URI	Devuelve la lista de llamadas	content://call_log/calls
Browser.BOOKMARKS_URI	Devuelve una lista con marcadores guardados en el navegador	content://browser/bookmarks

Cómo hacer una consulta

- URI del provider
 - `android.provider.CallLog.CONTENT_URI`
- Nombres de los campos que queremos
 - `CallLog.Calls.NUMBER`
- Tipo de datos de los campos

_ID	_COUNT	NUMBER
44	3	212 55 1234
13	2	425 23 1231
53	5	543 56 1123

Ejemplo Content Providers

Aplicación que muestra el registro de llamadas

Uso de un ListView

Lectura de Content Providers

Servicios

Componente que funciona sin interactuar con el usuario

Reproductor de música

Log de coordenadas GPS

Realizar tareas costosas en otro hilo

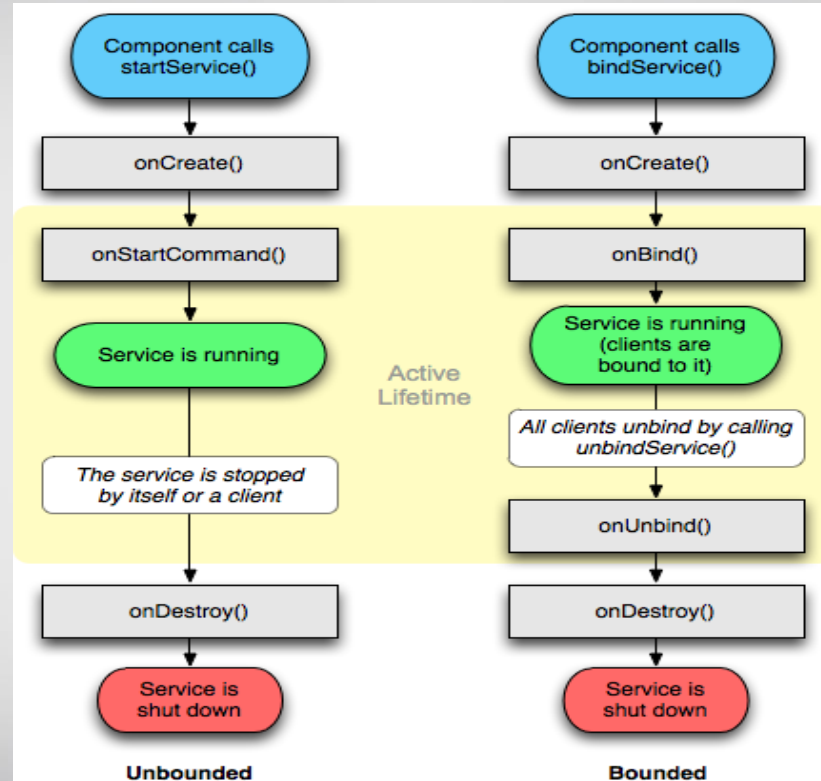
Realizar tareas repetitivas

Puede usarse de dos maneras:

Inicio por libre

Unido a una Activity

Servicio: ciclo de vida



BroadCast Receivers

Sirve para interceptar Intents

Nativos: `android.provider.Telephony.SMS_RECEIVE`

Propios: `com.tecnocom.intents.LOCATION_CHANGED`

Utiliza Intent Filter para especificar a qué tipo de
Intent escuchar

Se puede registrar en el código o en el Manifest

Ejemplo: descarga de archivo

Aplicación que descarga un archivo y
avisa cuando ha acabado

El servicio se debe comunicar con la
Activity (Broadcast Receiver)

Utilización de AsyncTask y