



android

An Open Handset Alliance Project

Connexion réseau UI Thread JSON

G.Seront

Connexion réseau

- Permissions requises

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
```



Statut de la connexion réseau

```
public void myClickHandler(View view) {
    ...
    ConnectivityManager connMgr = (ConnectivityManager)
        getSystemService(Context.CONNECTIVITY_SERVICE);
    NetworkInfo networkInfo = connMgr.getActiveNetworkInfo();
    if (networkInfo != null && networkInfo.isConnected()) {
        // fetch data
    } else {
        // display error
    }
    ...
}
```



Client Http

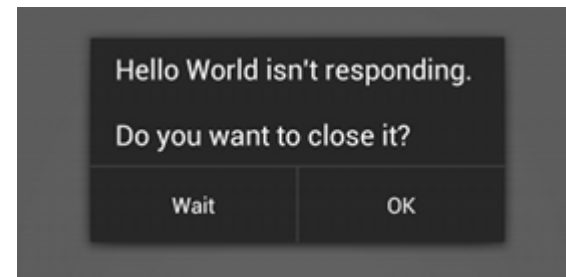
- HttpURLConnection

```
URL url = new URL(myurl);
HttpURLConnection conn = (HttpURLConnection) url.openConnection();
conn.setReadTimeout(2000 /* milliseconds */);
conn.setConnectTimeout(2500 /* milliseconds */);
conn.setRequestMethod("GET");
conn.setDoInput(true);
// Starts the query
conn.connect();
int response = conn.getResponseCode();
Log.d("debug", "The response is: " + response);
is = conn.getInputStream();
```



UI Thread

- Le thread UI est responsable
 - De la mise à jour graphique
 - De la gestion des évènements
 - onCreate, onClick, ...
- Faire des choses qui prennent du temps sur le thread UI:
 - Application non réactive
 - Freeze
 - Mauvaise idée!



UI Thread & requête réseau



Opérations sur un thread séparé

- Classe AsyncTask
- doInBackground()
 - Exécuté sur un **thread séparé du thread UI**
 - C'est là qu'on fait le job
- onPostExecute()
 - Reçoit le résultat de doInBackground
 - Tourne sur **le thread UI**
 - Indispensable car on ne peut pas modifier l'UI à partir d'un thread autre que le thread UI.



Opérations sur un thread séparé

```
private class DownloadWebpageTask extends AsyncTask<String, Void, String> {  
    @Override  
    protected String doInBackground(String... urls) {  
  
        // params comes from the execute() call: params[0] is the url.  
        try {  
            return downloadUrl(urls[0]);  
        } catch (IOException e) {  
            return "Unable to retrieve web page. URL may be invalid.";  
        }  
    }  
    // onPostExecute displays the results of the AsyncTask.  
    @Override  
    protected void onPostExecute(String result) {  
        textView.setText(result);  
    }  
}
```



Autre solution: Thread standard

- Ok mais
- Pas le droit de modifier la GUI sur un thread autre que le thread UI
- On doit utiliser `runOnUiThread`



Opérations sur un thread séparé

- On est en Android
- Dangers possibles?



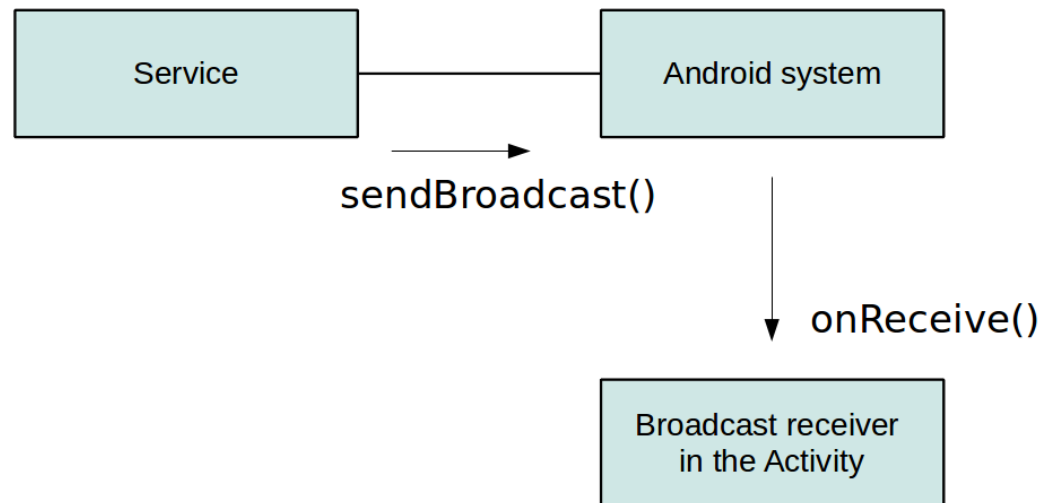
Opérations sur un thread séparé

- On est en Android
- Dangers possibles?
- Activité disparaît avant la fin de la requête



Requête dans un IntentService

- On va faire la requête dans un Service ou IntentService
- Quand il a fini il nous prévient via en broadcastant un event que l'activité écoute



JSON Parsing

- `new JSONObject(String)`
 - Parse le JSON dans la string
- On peut chercher un tag particulier
 - Un objet (`JSONObject`)
 - Un tableau (`JSONArray`)



JSON Parsing

```
JSONObject jsonObj = new JSONObject(jsonStr);

// Getting JSON Array node
contacts = jsonObj.getJSONArray(TAG_CONTACTS);

// looping through All Contacts
for (int i = 0; i < contacts.length(); i++) {
    JSONObject c = contacts.getJSONObject(i);

    String id = c.getString(TAG_ID);
    String name = c.getString(TAG_NAME);
    ... ..
}
```

