

Laboratoire 4

Utilisation de données environnementales II

Fabien Dutoit

SYM – Systèmes mobiles

Labo 4

- *Une grande donnée – à lire*
- *Manipulations – 2 parties*
 - *Utilisation des capteurs*
 - *Bluetooth Low Energy*
- *Questions associées à chaque manipulation*
- *Rendu: **dimanche 30.01.2022 à 23h55***
- *Veuillez nous indiquer les éventuels changements de groupes*

Labo 4



- *Les 2 manipulations nécessitent l'utilisation de smartphones physiques !*
- *N'oubliez pas d'activer le Bluetooth sur votre smartphone !*

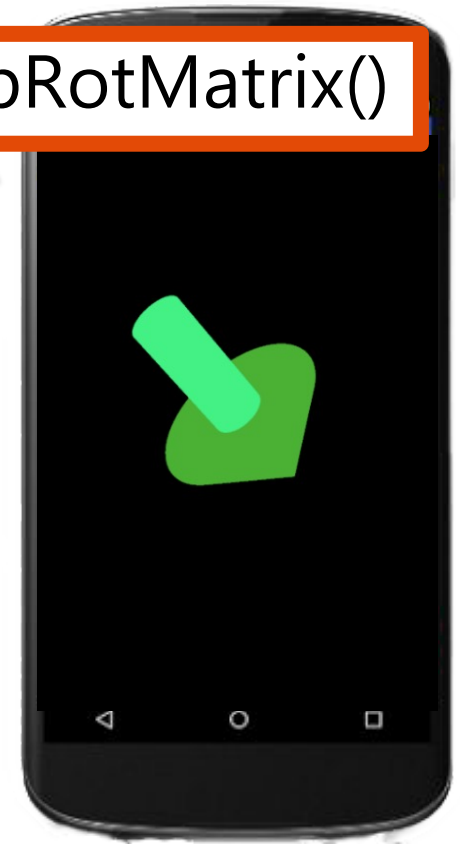
Labo 4 – Manipulation avec les capteurs – Boussole 3D

Magnétomètre

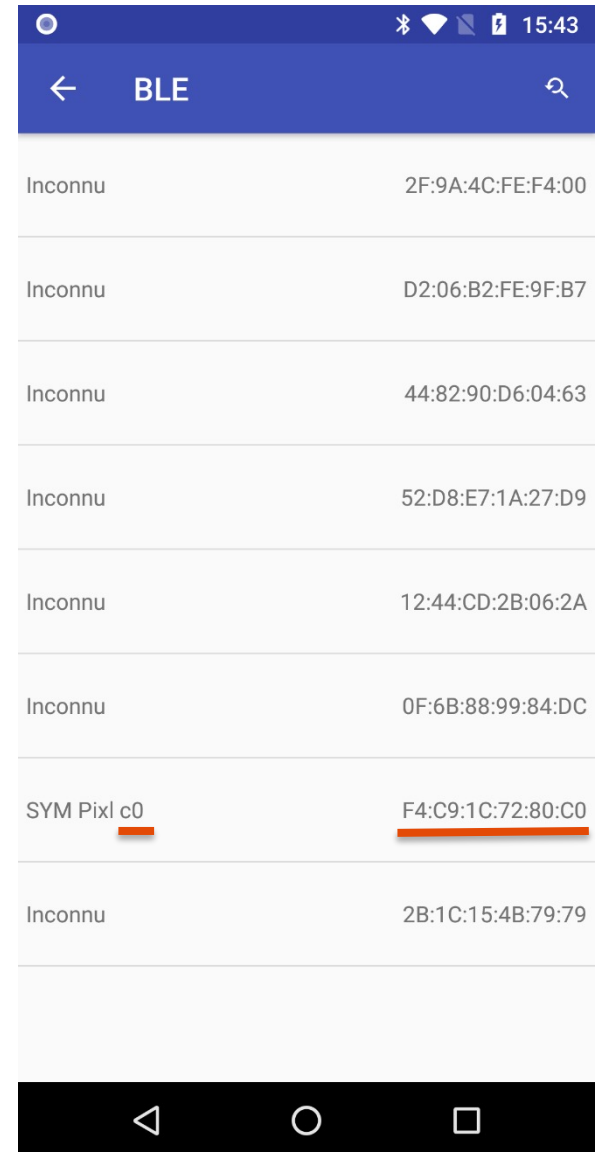
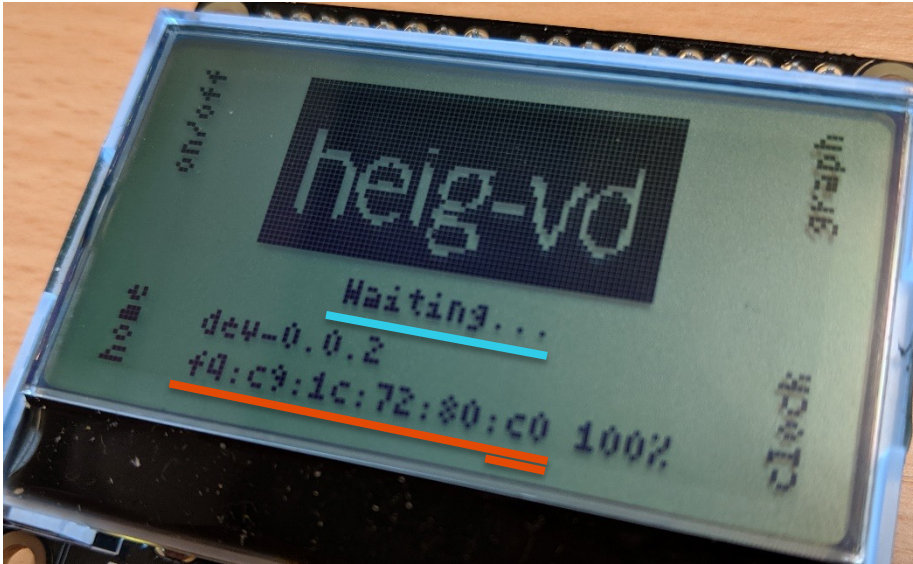
Accéléromètre

```
SensorManager.getRotationMatrix(  
    float[] R,  
    null,  
    float[] gravity,  
    float[] geomagnetic)
```

swapRotMatrix()

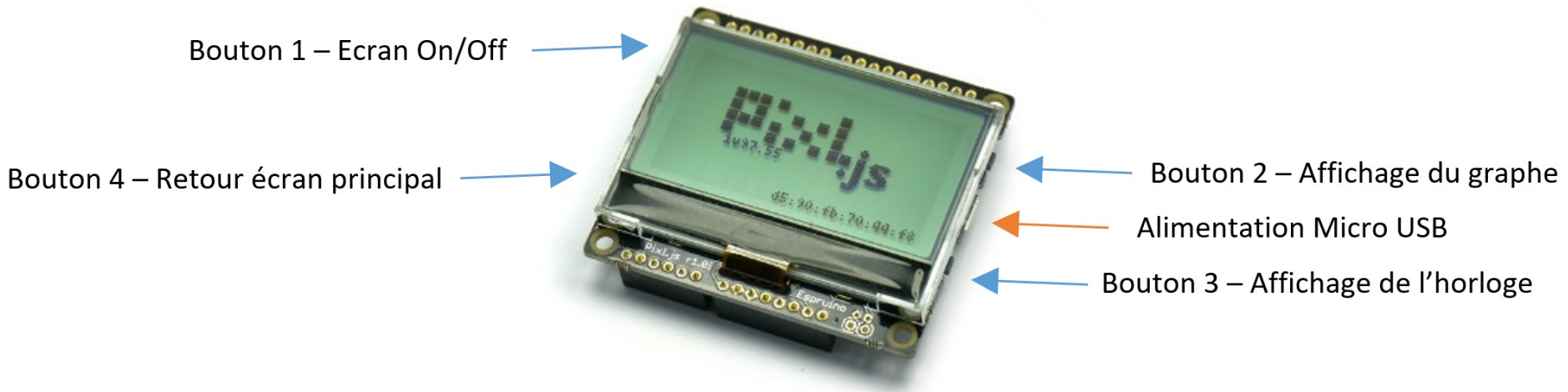


Labo 4 – Périphérique *BLE*



- Si *Waiting...* est affiché, le périphérique est en attente d'une connexion, sinon il affichera l'adresse du téléphone connecté
- Connectez-vous à **votre** périphérique !

Labo 4 – Périphérique *BLE*



- 2 possibilités pour l'alimenter:
 - Une pile bouton CR2032
 - Via la prise Micro USB



Ne pas toucher les boutons lors de la mise sous tension

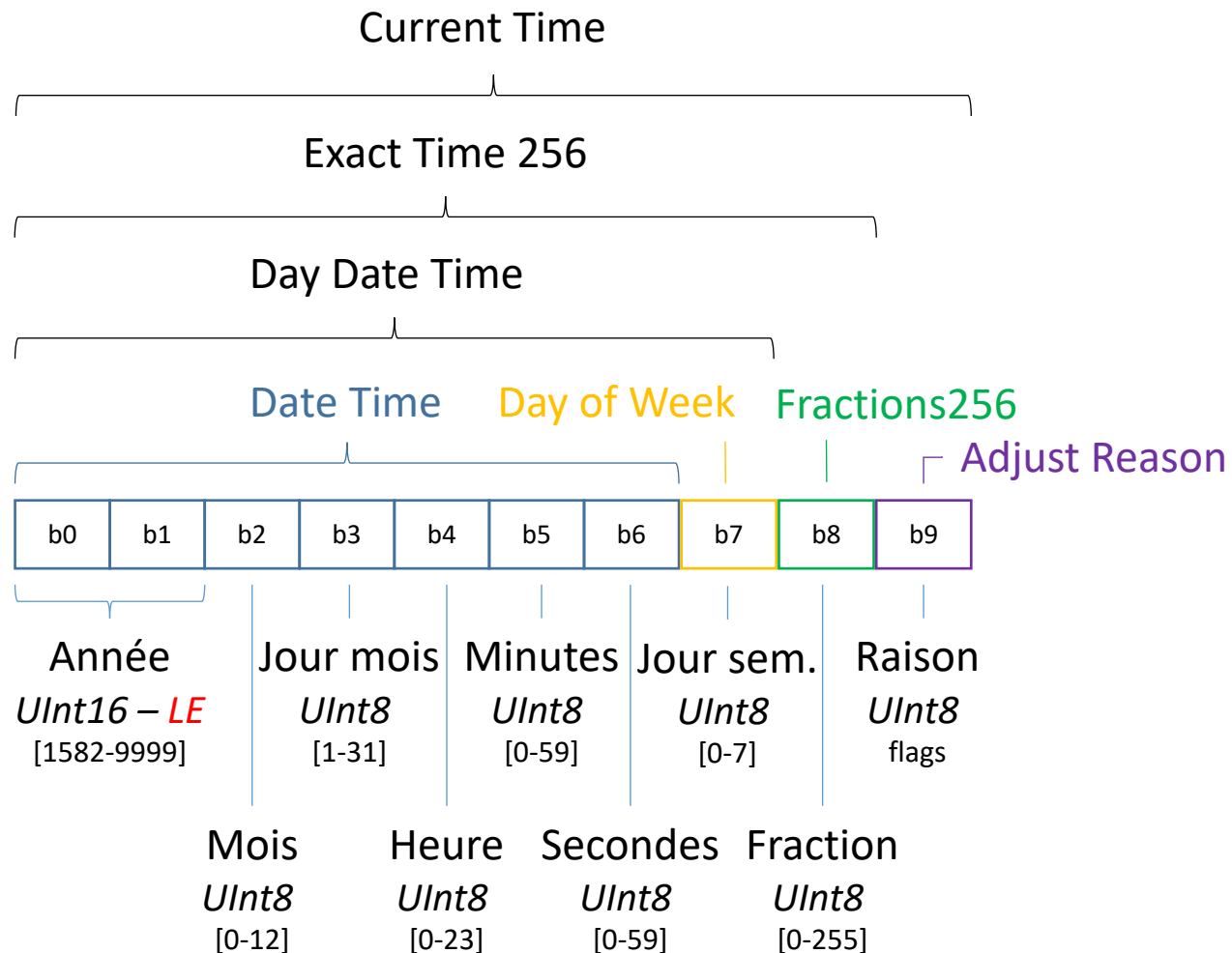
Labo 4 – BLE – Current Time Service

0x1085 -> 0000**1805**-0000-1000-8000-00805f9b34fb

- Les Pixl.js n'étant pas alimentés en permanence, leur horloge interne se réinitialise lorsque que l'alimentation est coupée
- Ce service permet de remettre l'horloge interne du périphérique à l'heure courante (écriture)
- Permet également au smartphone de s'enregistrer au périphérique pour recevoir régulièrement son heure actuelle (notification)



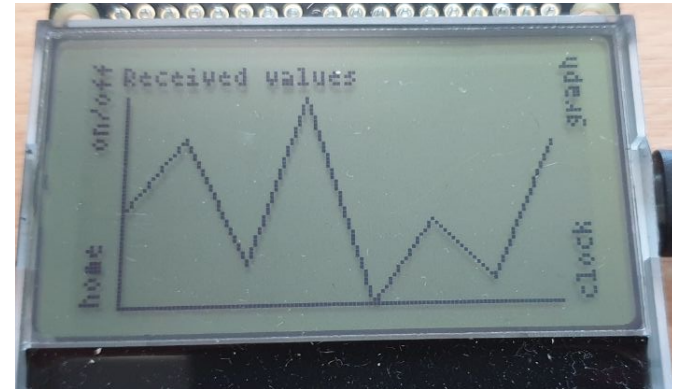
Labo 4 – BLE – Caractéristique Current Time



Labo 4 – BLE – Service Custom SYM

- **Graphe**

Permet au smartphone d'envoyer un entier. Le périphérique stocke les 10 derniers entiers reçus et les affiche sur un graphe à l'écran (écriture).



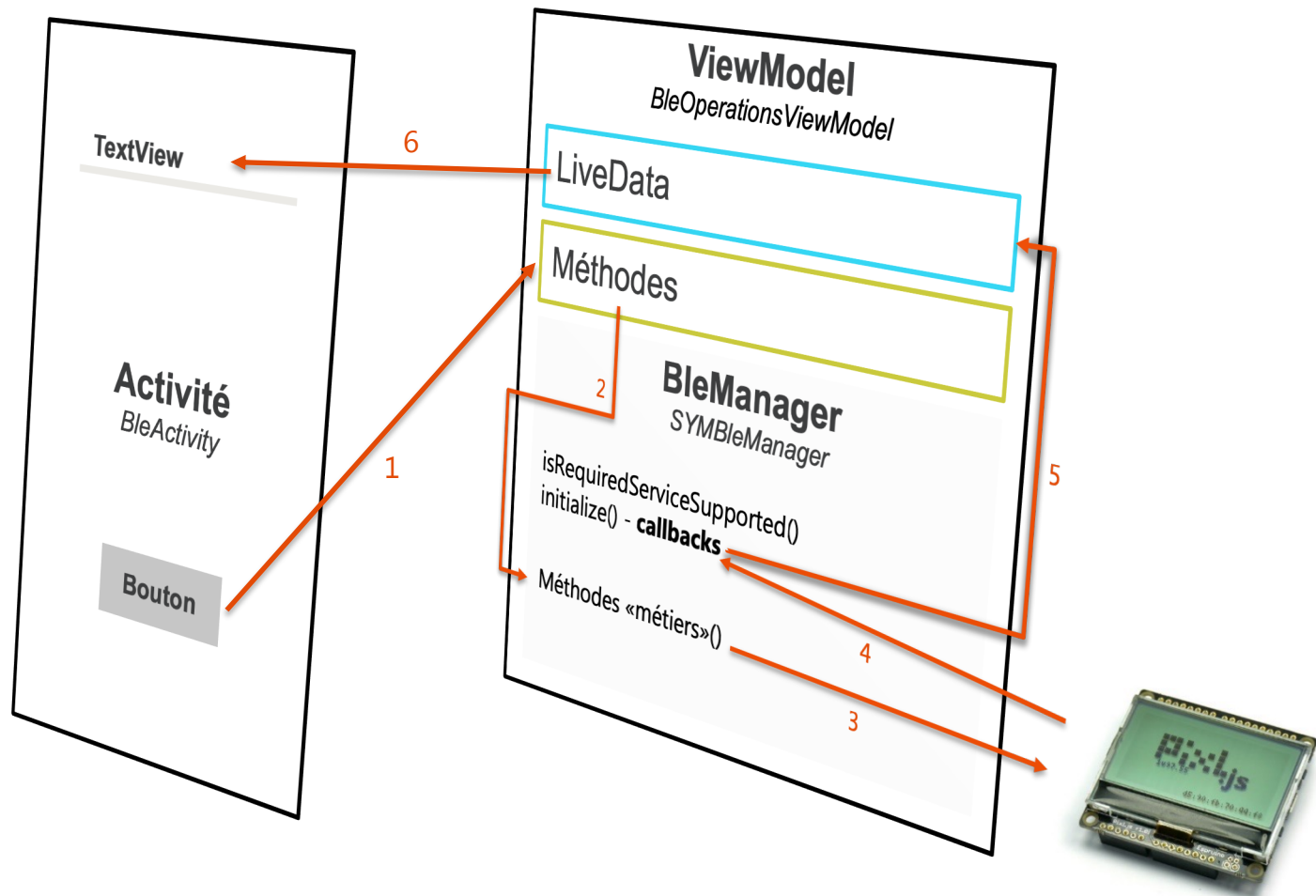
- **Température**

Permet au smartphone de lire la température du thermomètre interne du périphérique (lecture)

- **Boutons**

Un compteur qui est mis à jour à chaque fois qu'un des quatre boutons de l'écran est appuyé, celle valeur est envoyée au smartphone s'il s'est inscrit (notification).

Labo 4 – BLE – Architecture



Labo 4 – BLE – SYMBleManager

- **isRequiredServiceSupported()**

Permet de vérifier que le périphérique auquel on vient de se connecter possède bien tous les services et les caractéristiques attendues.

Vérifie également que les caractéristiques présentent bien les opérations attendues (lecture, écriture ou/et notification).

- **initialize()**

Permet de s'enregistrer afin de recevoir les notifications proposées par certaines caractéristiques.

Mise en place des callbacks correspondants.

Labo 4 – BLE – SYMBleManager

- **readTemperature()**

Lecture de la caractéristique température

La valeur récupérée sera envoyée à l'activité en utilisant le mécanisme des MutableLiveData

Exemple Kotlin:

```
readCharacteristic(temperatureChar).with { _: BluetoothDevice, data: Data ->
    // mise à jour de la température (MutableLiveData)
}.enqueue()
```

- data

Valeur lue pour la caractéristique temperatureChar

Type Data de la librairie offrant des méthodes helper

- enqueue()

Met en file d'attente la demande (exécution asynchrone)