



Développement mobile avancé

Laboratoire 4 - Présentation

Fabien Dutoit

Laboratoire 4

- Manipulations avec un périphérique *BLE*
 - Nécessite d'avoir un smartphone physique avec *Bluetooth*, et que celui-ci soit activé
 - Pour les versions d'*Android* plus anciennes, il faudra également activer la géolocalisation
- Matériel fourni :
 - Squelette d'une app *Android*
 - 1 écran connecté par personne
 - Prêt de smartphones *Android* sur demande
- Quelques questions théoriques

L'écran connecté

- 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 !

dma-labo-4



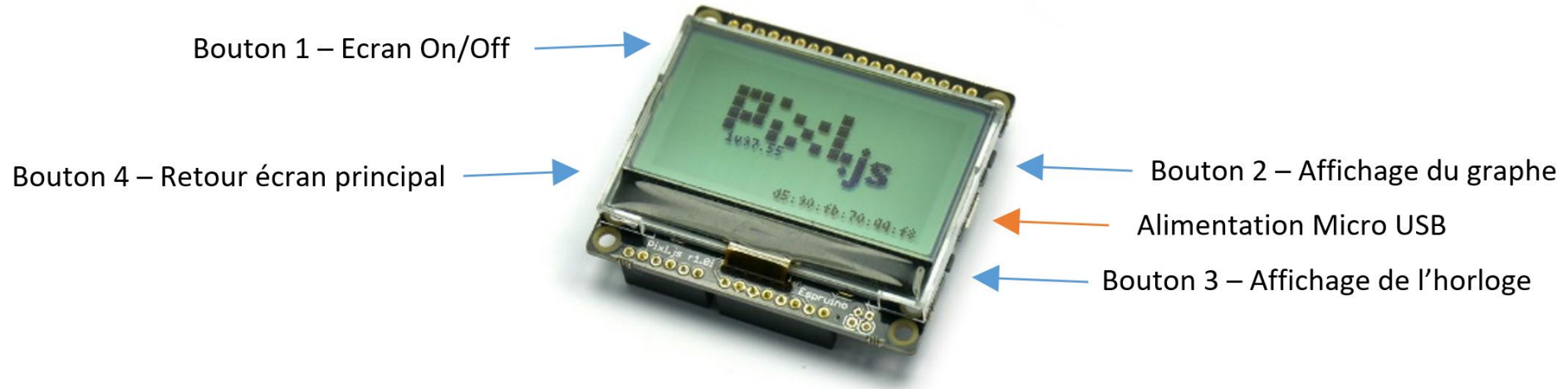
D4:EF:76:04:B4:F2

SYM Pixl f2


- L'adresse de l'écran est affichée sur celui-ci
- Le dernier byte de l'adresse est repris dans le nom du périphérique



L'écran connecté



- 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
→ risque de réaliser un *factory reset* du device

- Les boutons sont relativement fragiles...

BLE – Current Time Service

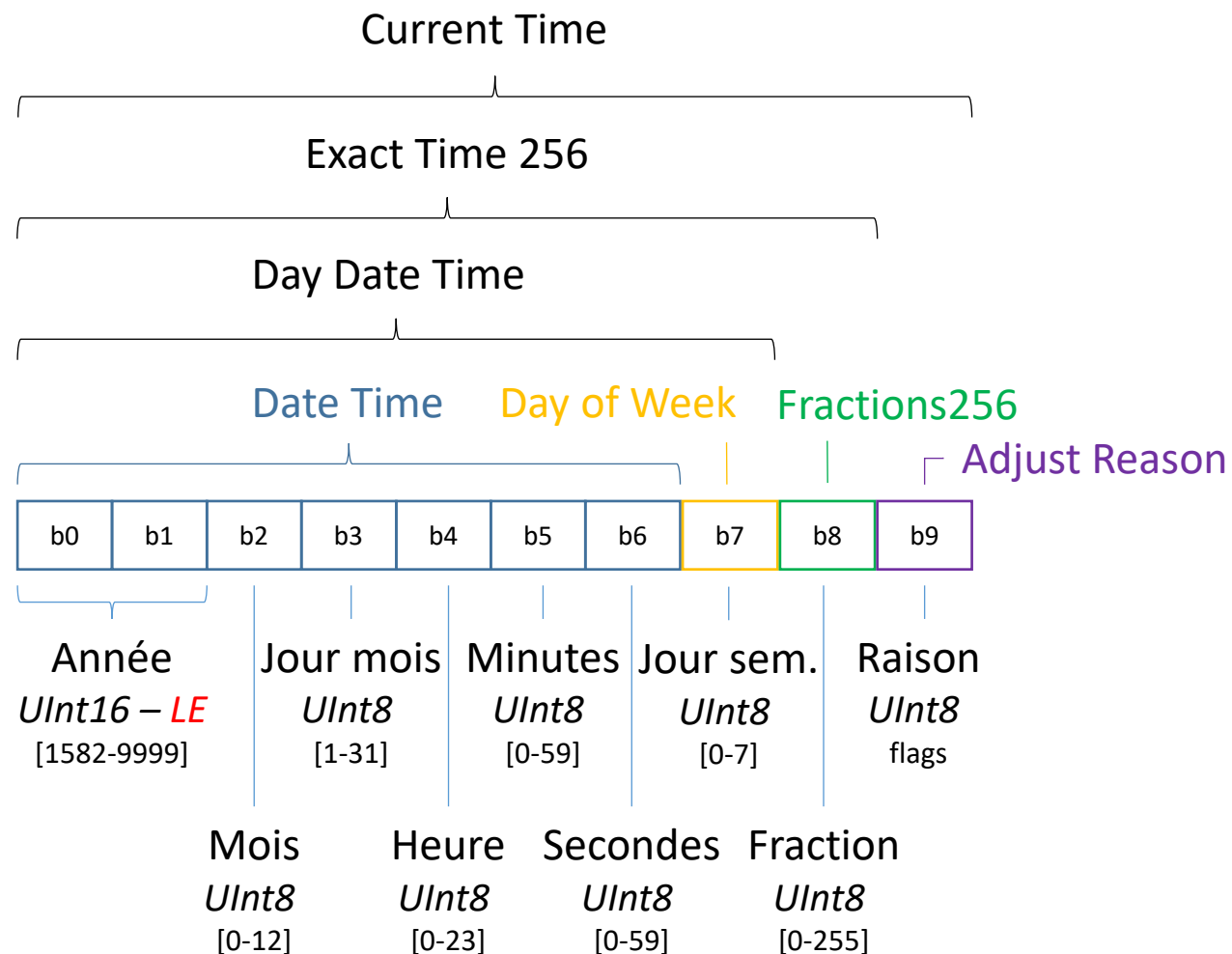
0x1805 -> 0000**1805**-0000-1000-8000-00805f9b34fb

- Les *Pixl.js* n'étant pas alimentés en permanence, leur horloge interne se réinitialise lorsque 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)



BLE – Current Time Service

0x1805 -> 00001805-0000-1000-8000-00805f9b34fb



BLE – Custom Service

3c0a1000-281d-4b48-b2a7-f15579a1c38f

- **Graphe**

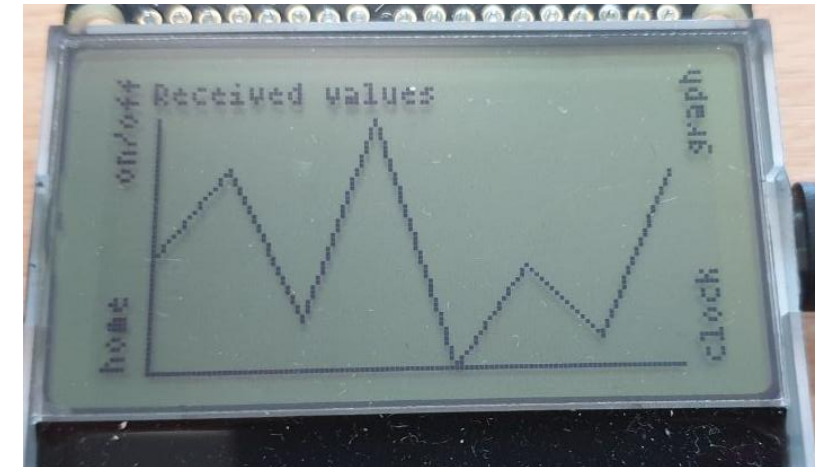
Permet au smartphone d'envoyer un entier.
Le périphérique stocke les 10 derniers entiers reçus et les affiche sur un graphe à l'écran (écriture)

- **Température**

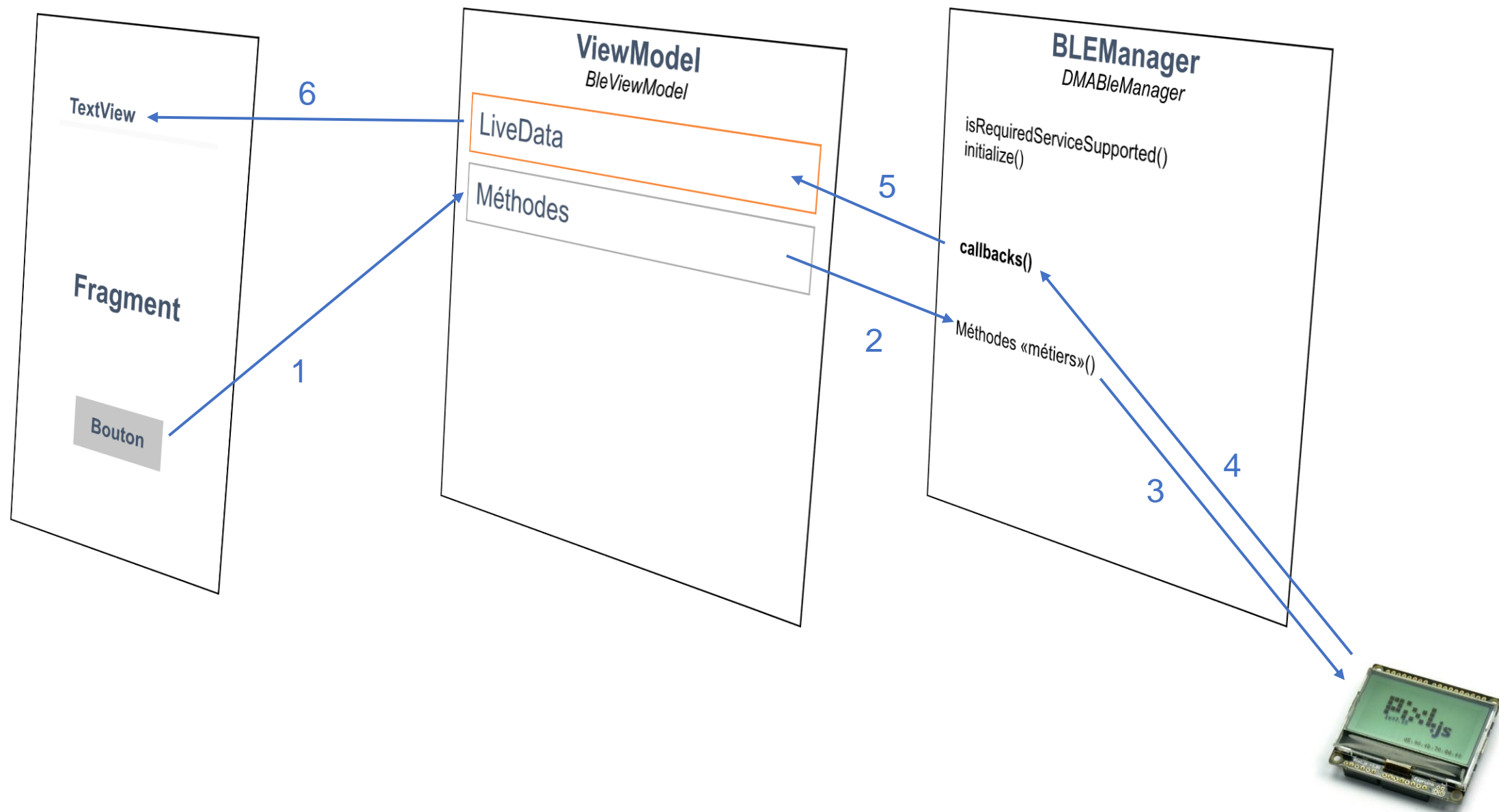
Permet au smartphone de lire la température mesurée par le thermomètre du périphérique (lecture)

- **Boutons**

Un compteur est mis à jour à chaque fois qu'un des quatre boutons de l'écran est cliqué, cette valeur est envoyée au smartphone par notification



Architecture



DMABLeManager – Méthodes à surcharger

- *isRequiredServiceSupported()*
 - Permet de vérifier que le périphérique auquel on vient de se connecter possède bien tous les services et toutes les caractéristiques attendues
 - Vérifie également que les caractéristiques présentent bien les opérations attendues (lecture, écriture ou/et notification)
- *initialize()*
 - Permet de s'enregistrer pour recevoir les notifications proposées par certaines caractéristiques
 - Mise en place des callbacks correspondants

DMABleManager – Méthodes métiers

- *readTemperature()*
 - Lecture de la caractéristique température
 - La valeur récupérée sera envoyée au *ViewModel* qui implémente *DMAServiceListener*

- *API* pour la lecture

`readCharacteristic().with{}.enqueue()`

La caractéristique

Le callback

L'appel à *enqueue()* permet d'enregistrer l'opération dans la file

- Dans le *callback* vous recevrez les données au format *no.nordicsemi.android.ble.data.Data*, ce type fourni par la librairie met à disposition des méthodes très utiles pour la gestion des données binaires

HEIG
VD
HAUTE ÉCOLE
D'INGÉNIEURIE
ET DE GESTION
DU CANTON
DE VAUD

HE^{VD}
IG

HAUTE ÉCOLE
D'INGÉNIEURIE
ET DE GESTION
DU CANTON
DE VAUD