



Master SYSTEMES ET SERVICES POUR L'INTERNET DES OBJETS

UE RESEAUX

Technologie d'accès sans fil – LoRa®

TD Réseau LoRa®



Contenu

1. Introduction.....	3
2. Liste des outils et documents	4
3. Prise en main du device LoRa® Adeunis FTD	5
4. Configuration du device LoRa®	7
4.1 Configuration du device via commandes AT :	7
4.2 Analyse de la procédure d'activation :	8
5. Provisionning du device sur le portail Live Objects.....	10
6. Représentation des messages échangés	12
7. Analyse du mode acquitté.....	13
8. Analyse du mode ADR (Adaptive Data Rate).....	14
9. Décodage de la payload data UL.....	15
10. Dashboard sur Live Objects	16
11. Message DL applicatif depuis Live Objects	17
12. Récupération des données sur la plateforme de data management via le mode API :	19
12.1 Architecture Live Objects :	19
12.2 Création de la clé API :	20
12.3 Utilisation de l'outil Swagger :	21
13. Routage des données vers une plateforme tierce.....	22
13.1 Récupération des données via la méthode HTTP PUSH:	22
13.2 Récupération des données via le protocole MQTT :	23

1. Introduction

- **Objectifs du TD Réseau LoRa® :**

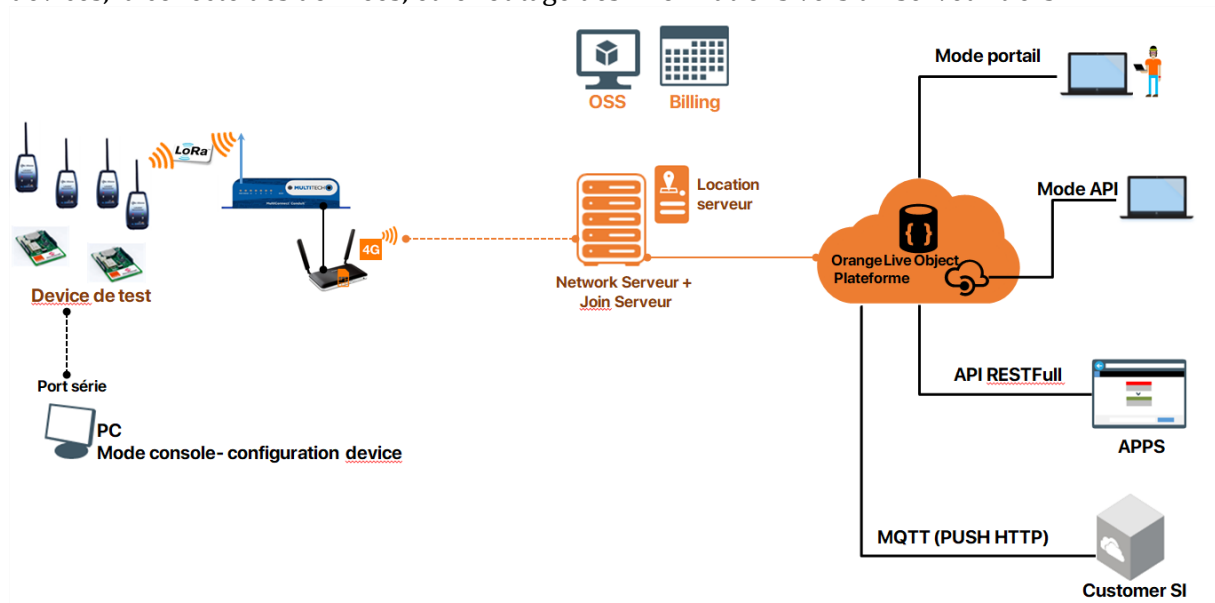
L'objectif du TD est de mettre en application les notions apprises lors du cours théorique, de comprendre l'architecture du réseau LoRa® d'un point de vue radio, protocolaire, et informatique.

Le TD est constitué de 3 parties :

1. Configuration du device LoRa®
2. Provisionning et analyse du fonctionnement du device sur Live Objects.
3. Collecte des données via le mode portail, via le mode API et routage des données sur un serveur via les protocoles http Push et MQTT.

- **Schéma d'architecture :**

Les devices LoRa® (LoRaWAN demonstrators) sont connectés via le réseau LoRa® à une nano gateway. Cette nano gateway permet de couvrir le site de l'Université, celle-ci est connectée via un câble Ethernet à un dongle 4G. Cela permet d'envoyer les trames issues des devices au network serveur via le backhaul cellulaire 4 G d'Orange. Un tunnel IPSEC est créé entre le network serveur et la nano gateway à travers le réseau cellulaire puis Internet. La plateforme de data management appelée **Live Object** est utilisée pour le provisionning des devices, la collecte des données, et le routage des informations vers un serveur tiers.



2. Liste des outils et documents

OUTILS A DISPOSITION :

- **Device Adeunis FTD avec câble USB**
- Logiciel utilisé pour se connecter au terminal LoRa® - **HyperTerminal**
Sous windows: <https://hercules-setup.soft32.com>
Autres outils alternatifs : Putty, Tera Term...
- Accès au portail **Live Objects** :
En mode portail: <https://liveobjects.orange-business.com/#/login>
En mode API: Utilisation du **swagger** (<https://liveobjects.orange-business.com/swagger-ui/index.html>)
- **Serveur HTTP** : <https://rbaskets.in/web>
Routage des données émises par le device sur le **serveur HTTPS** rbaskets via la méthode **http Push**.

DOCUMENTS A DISPOSITION :

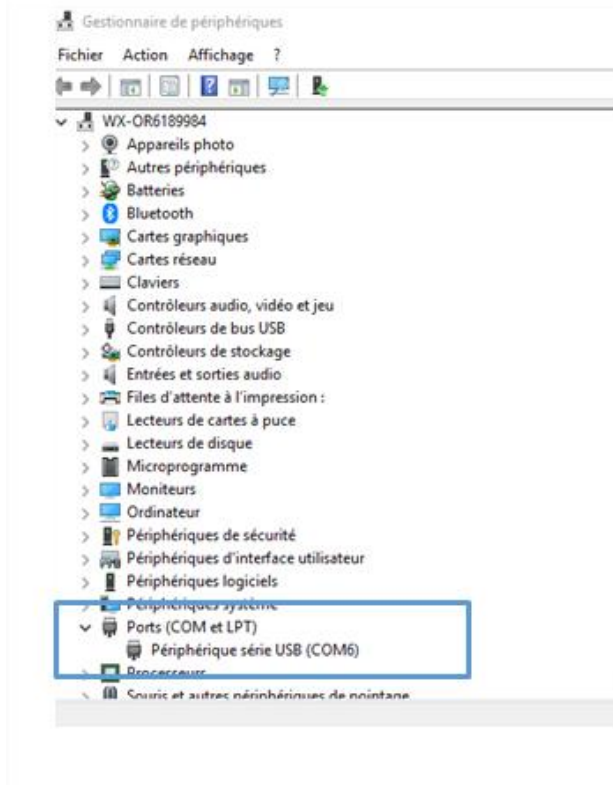
- Documentation du device Adeunis Demonstrator FTD
- Support de cours « Réseau LoRa® »
- Guide utilisateur du portail Live Objects disponible via le portail.
- Internet

3. Prise en main du device LoRa® Adeunis FTD

Cf Documentation du device

Récupérer le port COM utilisé par le device sur le PC.

Explorateur de fichier/ clic droit sur « Ce PC » / Gestionnaire des périphériques/ Ports (COM)

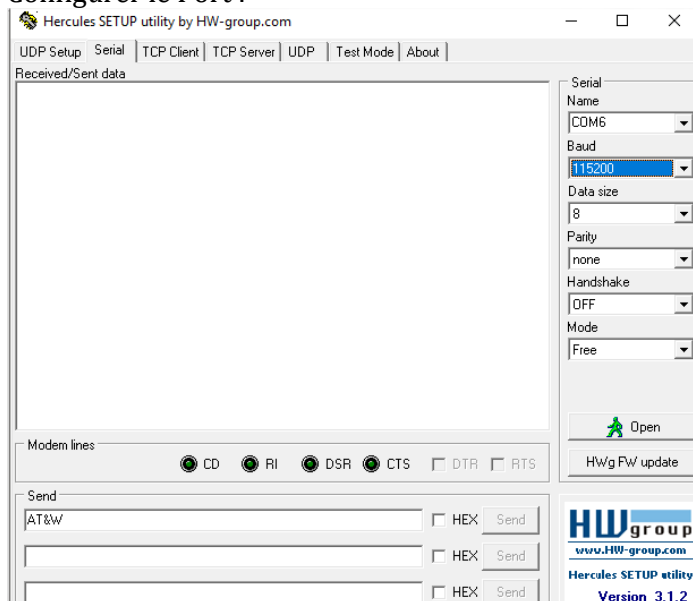


➔ Se connecter via l'Hyper terminal Hercules :

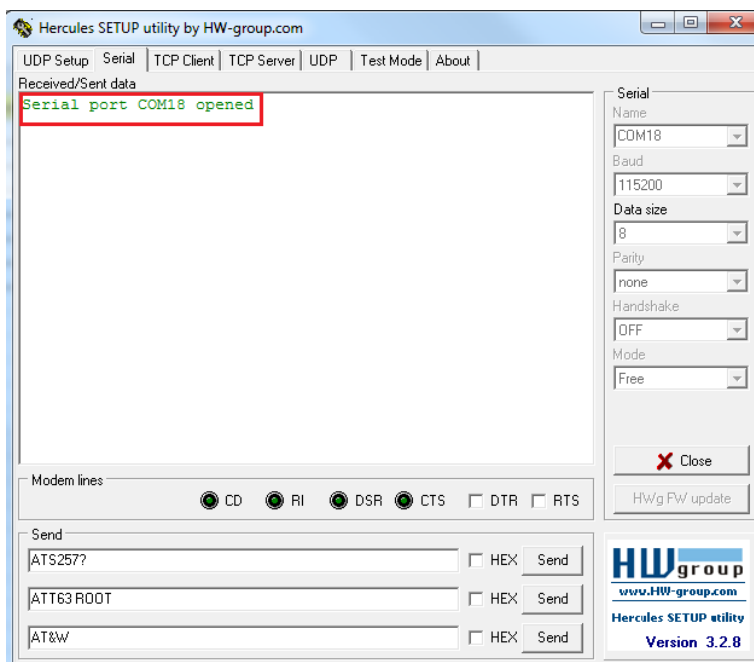
Ouvrir le logiciel Hercules.

Sélectionner le menu « Serial »

Configurer le Port :



➔ Cliquer sur le bouton Open.

**Entrer en mode Configuration:**

A chaque commande envoyée, le terminal retourne l'information suivante:

« O » commande effectuée avec succès

« E » erreur

« CM » passage en mode commande

➔ **Entrer dans le mode configuration:**

Lancer le mode commande **+++**

4. Configuration du device LoRa®

A partir de la documentation du device, renseigner les commandes à utiliser sur l'Hyperterminal pour répondre aux questions et indiquer les résultats obtenus.

Attention : pensez à débloquent l'accès aux registres et à enregistrer toute modification !

Commande d'accès aux registres : *ATT63 PROVIDER*

Commande de sauvegarde de configuration : *AT&W*

➔ Remplir le tableau ci-dessous :

	Commandes utilisée ?	Résultat
Device ID ? Numéro collé au dos		
DevEUI ? 16 caractères au dos		
APPEUI 16 caractères (n° registre ?)		
APPKEY 32 caractères(n° registre ?)		

4.1 Configuration du device via commandes AT :

Dans cette partie, il faut retrouver les registres / commandes à utiliser pour modifier la configuration du device sur différents aspects.

➔ Vérification du mode de provisionning : ABP ou OTAA ?

	Commande utilisée	Résultat
Lecture		
Ecriture		

➔ Activation de l'ADR : activé ou non ?

	Commande utilisée	Résultat
Lecture		
Ecriture		

➔ Valeur du Spreading Factor de départ :

	Commande utilisée	résultat (valeur du registre ET du SF)
Lecture		
Ecriture		

➔ Cycle du device :

	Commande utilisée	Résultat
Lecture		
Ecriture (cycle à 30s)		

➔ Configuration du MType UL :

	Commande utilisée	Résultat (valeur du registre ET du MType)
Lecture		
Ecriture (autre MType)		

4.2 Analyse de la procédure d'activation :

Si ce n'est pas/plus le cas, **configurer le device en ABP**, **enregistrer** et sortir du mode Commande avec la commande **ATO**.

Démarrer le device et analyser les données fournies sur l'écran du device lors des premiers cycles d'émission (3 premières trames UL). Pour cela, naviguer à travers les différents écrans d'affichage du device.

Questions :

Analyse du mode ABP	Réponses
Comment est faite l'activation du device (cf. cours) ?	En ABP, il n'y a pas de procédure JOIN. L'activation est faite de manière intrinsèque (les clés de sessions étant déjà connues) lors du provisionning du device dans Live Objects).
Quel MType est utilisé (cf. cours) ? Pourquoi ?	
Quels canaux sont utilisés ? Pourquoi ?	Les 3 canaux obligatoires (cf. cours)
Quel(s) Spreading Factor(s) est(sont) utilisé(s) ?	Les SF configurés (les mêmes pour Tx et RX1), et celui configuré spécialement pour RX2
Quelle est la bande passante utilisée (cf. cours) ?	125 kHz
Quelle est la réponse du réseau ?	En principe, le réseau ne doit pas forcément répondre. Il peut seulement envoyer une commande MAC ou répondre à une commande MAC si nécessaire.

Configurer désormais le device en mode **OTAA**. Regarder de nouveau sur le device après l'avoir redémarré.

Analyse du message JOIN_REQUEST (en pratique)	Réponses
Quel est la première trame envoyée (MType) ?	JOIN Request
Quelle est la fréquence utilisée ?	Cf. cours
Quel est le SF utilisé ?	Celui configuré pour TX
Quelle est la réponse du réseau ?	JOIN Accept

Analyse de la fenêtre de réception RX1 (théorie)	Réponses
Quel est le canal logique utilisé ?	Le même qu'en Tx
Quelle est la fréquence utilisée ?	La même qu'en Tx
Quelle est le SF utilisé ?	Le même qu'en Tx
Quelle est la bande passante utilisée ?	125 kHz

Analyse de la fenêtre de réception RX2 (théorie)	Réponses
Quel est le canal logique utilisé ?	LC-RX2 (canal propre à la fenêtre RX2)
Quelle est la fréquence utilisée ?	La seule fréquence autorisée en Europe pour Rx2 : 869.525 MHz
Quelle est le SF utilisé ?	Celui configuré précédemment
Combien de canaux de fréquences sont configurés ?	1 seul correspondant à la fréquence 869.525 MHz

Avant de passer à la suite, configurer le device en ABP et en mode ACK (MType ConfirmedDataUL)

5. Provisionning du device sur le portail Live Objects

➔ Se connecter au **portail Live Object**:

<https://liveobjects.orange-business.com/#/login>

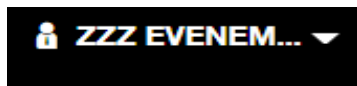
Utiliser le login/Password correspondant à votre équipe :

ID : SSIO_<n> (<n> : numéro d'équipe annoncé au dos du device)

Pwd : M2SSIO_pwd<n>

LES DEUX REGLES D'OR SUR LIVE OBJECTS :

- 1- NE JAMAIS MANIPULER LE DEVICE D'UN OU PLUSIEURS AUTRES GROUPES
- 2- NE JAMAIS CLIQUER SUR LE LOGO SUPPRESSION – PAS DE SUPPRESSION DE DEVICE
(VOIR PRLA CAPTURE D'ECRAN NT CI-DESSOUS)



Toute infraction à au moins une de ces deux règles vous vaudra un zéro pointé 😈

➔ Provisionner le device LoRa® sur le network serveur d'Orange via Live Objects (Parc > Ajouter un nouvel équipement) :

- Renseigner le **DevEUI** (ID inscrit au dos du device)
- Sélectionner le profil : **Generic_ClassA_RX2_SF12**
- Décodeur : laisser vide
- **Ne pas cocher l'option Ack Uplink** (laisser tel quel)
- Conserver le **Connectivity Plan** sélectionné par défaut : **CP_Basic**
- Renseigner les identifiants **AppEUI** et **AppKey**
- Renseigner un nom : **Demo_Device_<ID au dos du device>**
- Renseigner un tag : **M2SSIO**
- Cliquer sur Créer

➔ (Re)démarrer le device

➔ Analyser les logs sur le device

➔ Est-ce que le device démarre correctement? Pourquoi ?

Le device étant en ABP, il ne pourra pas être vu par le réseau car il a été provisionné (déclaré/créé) en mode OTAA (déclaration des clés racines AppEUI et AppKEY). Il faut donc reconfigurer le device en mode OTAA pour qu'il puisse être activé sur le réseau, et ne plus rester à l'état « enregistré ».

➔ Effectuer la modification adéquate et redémarrer le device : que voit-on sur le portail Live Object ? Quel est le nouveau statut du device sur le portail ?

➔ Quel autre problème se pose ? Corriger via modification du bon registre et redémarrer le device une nouvelle fois. Commenter

L'autre problème vient du fait que le device est configuré pour envoyer des trames UL en mode Confirmed (c'est-à-dire qu'il demande implicitement un acquittement de chacune de ses trames UL au réseau). Or, comme nous n'avons pas (encore) activé l'option ACK sur Live Objects, il ne recevra jamais de DL (applicatif et/ou MAC) de Live Objects et/ou du réseau.


6. Représentation des messages échangés

A partir des données acquises dans le portail (menu « Journal de Log / Message UL / Données»), représenter le call flow de la procédure JOIN ainsi que le call flow de la procédure d'envoi d'un message data UL :

**Device****gateway****Network Serveur****Data Management**

7. Analyse du mode acquitté

Activer le mode Acquittement UL sur le device (modification de registre).

Retourner sur Live Objects et activer l'option ACK (Parc > Cliquer sur le device > Identité > Informations sur l'équipement > ). Que voit-on de nouveau ?

Démarrer le device et analyser le cycle JOIN sur la session HyperTerminal : quels changements pouvons-nous voir au niveau de la procédure de JOIN ?

On ne voit aucun changement dans la procédure JOIN. En effet, JOIN Request et JOIN Accept étant eux-mêmes des MTypes différents, la procédure JOIN est indépendante de la configuration « MType UL » du device (mode Confirmed ou Unconfirmed).

Poursuivre l'analyse des trames UL du device : qu'observe-t-on à chaque cycle de transmission d'un message UL ?

Cette fois-ci, le device reçoit TOUJOURS (sauf perte paquet au niveau radio) une trame DL dans l'une de ces deux fenêtres de réception. Ces trames DL ont toutes un flag « ACK » à 1 (ou TRUE), qui montre que le réseau dit au device qu'il a bien reçu sa dernière trame UL.

→ Représenter les messages échangés lors d'un cycle UL en mode acquitté :



Device



gateway



Network Serveur



Data Management

8. Analyse du mode ADR (Adaptive Data Rate)

- Activer le mode ADR sur le device via la session HyperTerminal
Rappeler la procédure complète pour cette opération

- Démarrer le device et analyser les logs sur l'écran du device et sur Live Objects.

En particulier, analyser les 6 premiers cycles UL : quel(s) changement(s) observons-nous ?

En activant l'ADR, on doit s'apercevoir que le réseau peut potentiellement envoyer des commandes *LinkADRRequest* au device pour adapter son débit (donc son SF), sa puissance d'émission et/ou sa redondance (nombre de répétitions d'une trame UL) selon la QoS radio mesurée. Ainsi, le device peut par exemple voir son SF (pour Tx et donc Rx1), en répondant bien évidemment au réseau à la commande *LinkADRReq* par une commande *LinkADRAns*.

- Représenter les messages échangés lors de la procédure de changement de Spreading Factor faisant apparaître les messages *LinkADRReq/LinkADRAns*



Device



gateway



Network Serveur



Data Management

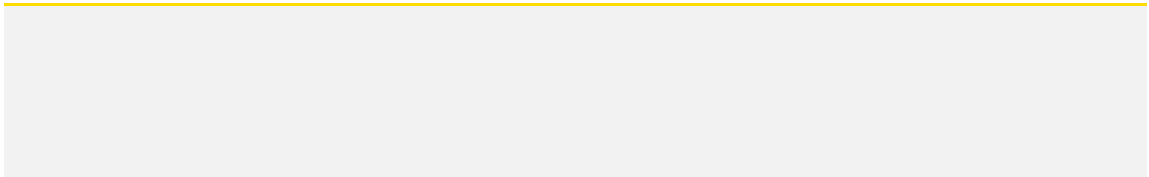
9. Décodage de la payload data UL

Noter la valeur de la payload data UL envoyée par le device: (*ex : 8E20D3090CDB*)
Trouver la signification de la payload et le moyen de décoder celle-ci.

Sur Live Objects, sélectionner le décodeur approprié pour le device de test :
Adeunis-LoRaWanDemonstrator

Une fois le décodeur appliqué au device, attendre plusieurs cycles d'émission.

En analysant les trames UL du device sur Live Objects, détailler le contenu de la payload applicative :



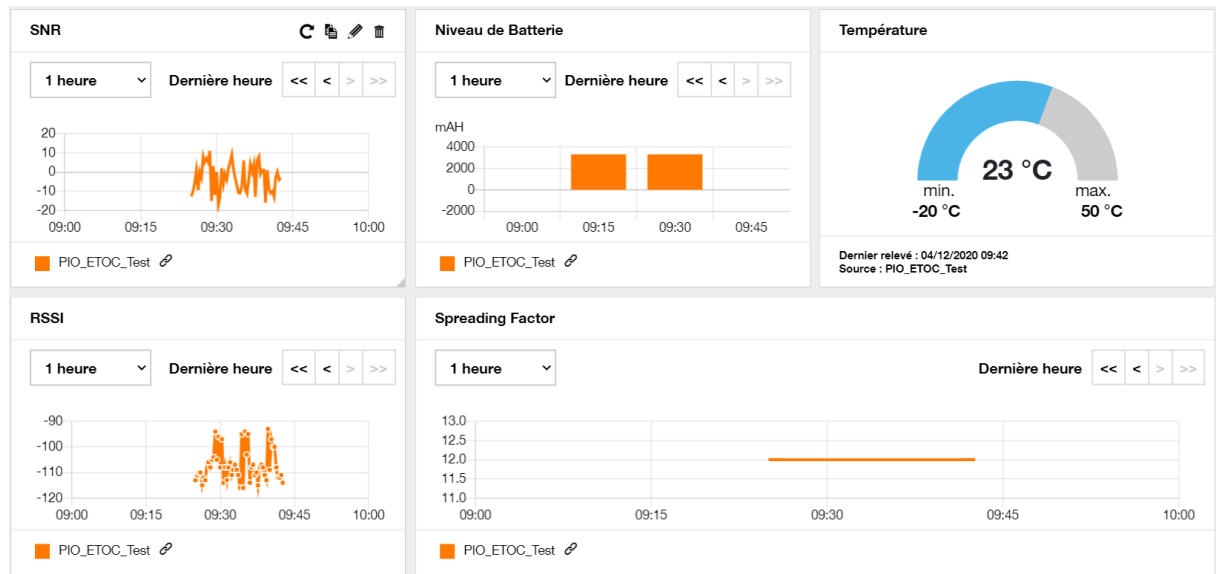
Note complémentaire : il est possible de créer son propre décodeur dans Live Objects. Des informations sur le codage de la donnée applicative du device sont détaillées pour vous permettre de développer un décodeur sur la plateforme d'accès, ou sur un potentiel serveur applicatif.

Mais nous manquons de temps et donc nous allons directement passer à la suite pour aujourd'hui 😊

10. Dashboard sur Live Objects

Créer un dashboard personnalisé sur le portail Live Objects pour le device de test avec les informations suivantes :

- Température (sous forme de Jauge)
- Etat de la batterie du device (remonté par le device)
- Spreading Factor (remonté par le réseau)
- SNR(remonté par le réseau)
- RSSI(remonté par le réseau)



Quel est l'intérêt d'un Dashboard pour un client IOT ?

Un Dashboard est très utile, en LoRa et plus globalement dans le monde de l'Internet des Objets.

En effet, un client (ou même un opérateur) peut se baser sur des dashboards pour analyser les données remontées par le device (aussi bien pour des métriques radios qu'applicatives) et ainsi exploiter de façon plus efficace le parc de devices déployés.

Il est d'ailleurs possible, en complément d'un ou plusieurs Dashboards, de définir des seuils d'alertes pour détecter des anomalies, comme par exemple une température trop basse, un device géolocalisé hors de sa position (détection de vol), ou encore des terres pas assez arrosées (taux d'humidité trop bas)...

11. Message DL applicatif depuis Live Objects

Désactiver le mode ACK sur le device pour la suite du TP.

Depuis la plateforme Live Object envoyer une commande downlink au device en suivant les instructions suivantes :

- ➔ Depuis le portail : Parc > sélection du device > Commande (Downlink) / Ajouter une commande :
- ➔ Définir le port : exemple : 5.
Pourquoi le port doit être différent de 0 ?

Le port 0 est réservé à la couche MAC. Il est donc utilisé exclusivement pour les commandes DL « purement » MAC. On ne peut l'utiliser pour les commandes DATA (applicatives) et a fortiori MAC+DATA (payload applicative + commande MAC).

- ➔ Entrée une donnée Hexadécimale (ex : CAFE)

- ➔ Durée max PENDING = 5 minutes
- ➔ Pas d'acquiescement réseau
- ➔ Vérifier sur Live Objects l'envoi de la commande :

Renvoyer une autre commande applicative **en demandant cette fois-ci l'acquiescement réseau.**

Quelle(s) différence(s) y a-t-il dans les traces (Journal) ?

Le DL envoyé vers le device est un MType ConfirmedDataDL. Ainsi, lors de la trame UL suivante, quelque soit le MType utilisé par le device (ConfirmedDataUL ou UnconfirmedDataUL), la trame UL contiendra un flag « ACK » = 1.

- ➔ Représenter les messages échangés lors de l'envoi d'un message DL (de la plateforme vers le device) dans les deux cas de figure précédents :



Device



gateway



Network Serveur



Data Management

12. Récupération des données sur la plateforme de data management via le mode API :

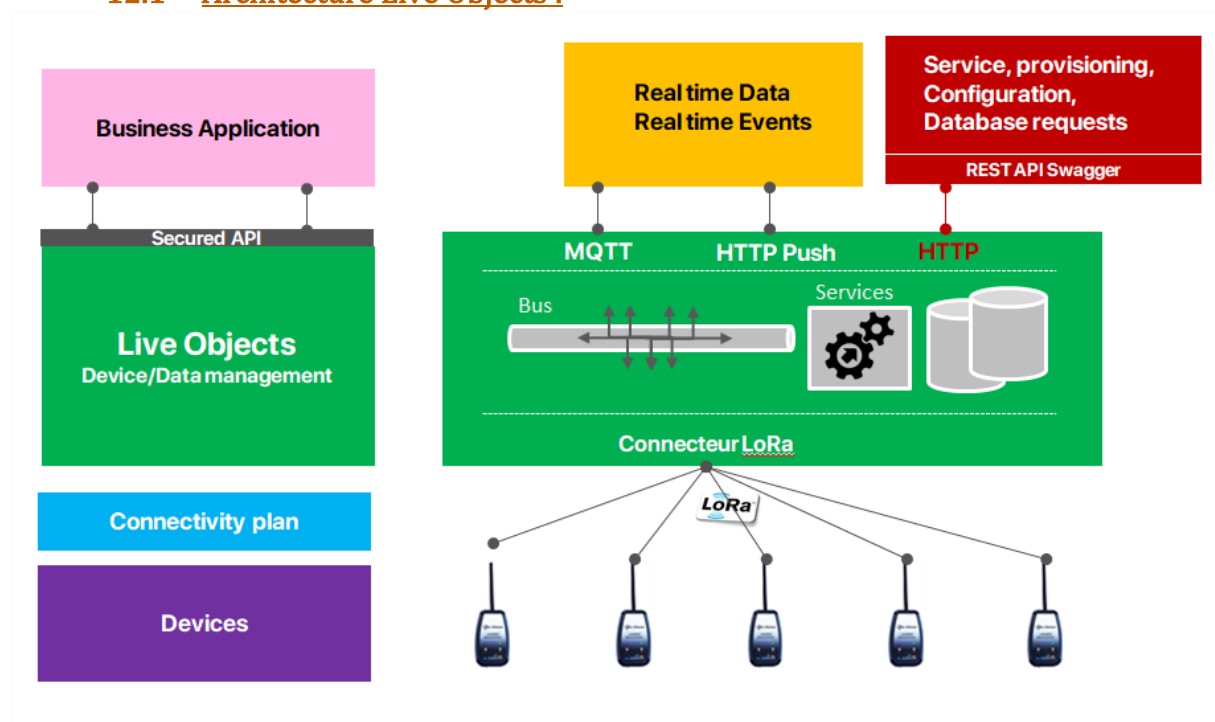
Tout ce qui est disponible via le mode portail est disponible via le mode API.

API : Application Programming Interface.

API est un ensemble normalisé de classes, de méthodes, de fonctions et de constantes qui sert de façade par laquelle un logiciel offre des services à d'autres logiciels. Elle est offerte par une bibliothèque logicielle ou un service web, le plus souvent accompagnée d'une description qui spécifie comment des programmes consommateurs peuvent se servir des fonctionnalités du programme fournisseur.

Une API permet d'exposer les données et fonctions à un autre logiciel.

12.1 Architecture Live Objects :



Le mode API au sein de la plateforme Live Object peut être utilisé pour :

- Du device management.
- Du data management.
- Configurer le routage des messages.
- Configurer les alarmes.
- Accéder aux données.

12.2 Création de la clé API :

L'ensemble des informations liées aux API Live Object est disponible via le swagger suivant : <https://liveobjects.orange-business.com/swagger-ui/index.html>

Swagger est une infrastructure **logicielle open-source** reposant sur un vaste écosystème d'outils qui aide les développeurs à concevoir, créer, documenter et utiliser des services Web RESTful.

On peut aussi utiliser l'outil **Postman**, qui est un outil open source permettant de tester le mode API. Postman existe sous la forme d'une App (Windows/MacOS/Linux) et d'une Chrome App.

Avant d'utiliser le mode API il faut créer une **clé API**.

La clé API identifie l'utilisateur, et associe les droits nécessaires pour se servir de l'API.

➔ **Créer une clé API sur le portail Live Objects :**

Administration > Clés d'API > Ajouter une clé d'API.

➔ **Sélectionner le profil « Personnalisé », cocher toutes les cases en « Lecture » comme ci-dessous (sans restriction au Files ni authentification forte) puis cliquer sur Créer**

☐ Equipement MQTT
 ☐ Application
 ☐ Connecteur externe
 ☒ Personnalisé

Nom	Description	Lecture	Ecriture
Clé d'API	Gestion des clés d'API Attribue les rôles API_KEY_R et API_KEY_W	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Utilisateur	Gestion des utilisateurs, Consultation des logs d'accès Attribue les rôles USER_R et USER_W	<input type="checkbox"/>	<input type="checkbox"/>
Compte	Paramétrage du compte Attribue les rôles SETTINGS_R et SETTINGS_W	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Equipement	Gestion des équipements connectés Attribue les rôles DEVICE_R et DEVICE_W	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Traitement des données	Configuration du décodage et traitement des données collectées Attribue les rôles DATA_PROCESSING_R et DATA_PROCESSING_W	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Campagne	Gestion des opérations en masse sur un ensemble d'équipements Attribue les rôles CAMPAIGN_R et CAMPAIGN_W	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Configuration du bus	Gestion du routage et files de messages Attribue les rôles BUS_CONFIG_R et BUS_CONFIG_W	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Données	Accès aux données collectées Attribue les rôles DATA_R et DATA_W	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Kibana	Accès à Kibana Attribue le rôle KIBANA_R	<input type="checkbox"/>	
Logs	Accès au Journal de logs Attribue le rôle LOGS_R	<input checked="" type="checkbox"/>	
Accès Equipement	Accès en mode Device d'un équipement MQTT Attribue le rôle DEVICE_ACCESS	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Accès Connecteur	Accès MQTT pour un connecteur externe Attribue le rôle CONNECTOR_ACCESS	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Accès au Bus	Accès aux bus de messages via MQTT ou HTTP Attribue les rôles BUS_R et BUS_W	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Nom de l'API : SSIO<n>_APIKey

Mémoriser la clé API pour la réutiliser par la suite.

Clé API	Note : la clé d'API est un moyen sécurisé pour permettre l'accès/l'envoi d'une donnée depuis un équipement (en l'occurrence Live Objects) vers un serveur applicatif.
---------	---

12.3 Utilisation de l'outil Swagger :

Ouvrir le lien suivant :

<https://liveobjects.orange-business.com/swagger-ui/index.html#/>

➔ **Obtenir les informations « Device » via le mode API :**

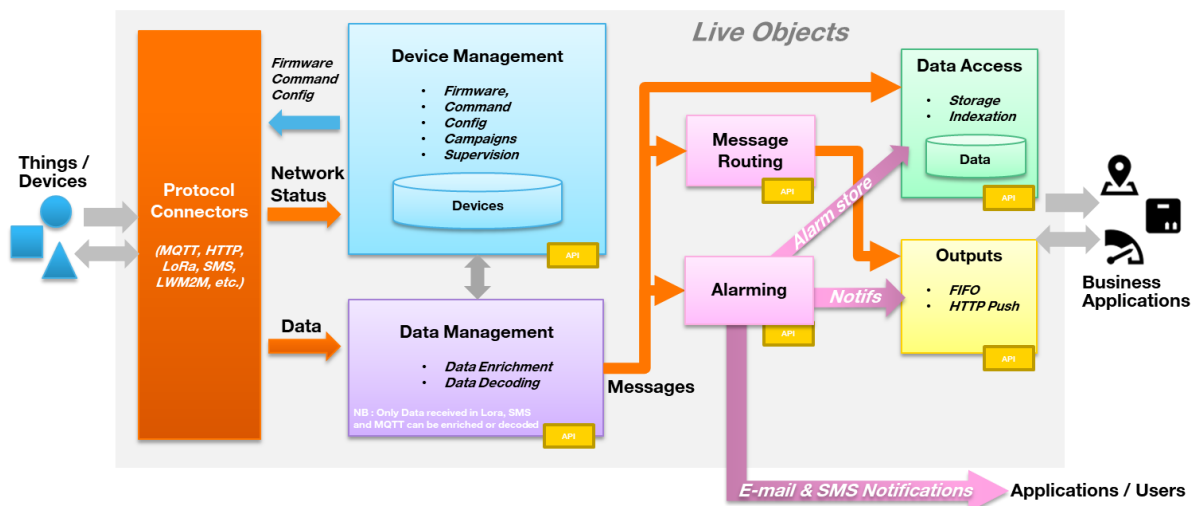
- Sélectionner la bonne rubrique (à vous de la trouver 😊)
- Dérouler la requête et écrire les entrées nécessaires (clés d'API et DeviceID)
- Exécuter ("Try it out") et observer l'URL de requête et la réponse associée

Copier la fin de l'URL permettant d'obtenir la configuration de tous les devices du projet (tag « M2SSIO »)	
Copier l'URL permettant d'obtenir la configuration du device	

Bonus : Vous pouvez continuer à effectuer d'autres requêtes API pour mieux appréhender cette notion d'API.

13. Routage des données vers une plateforme tierce

Architecture Live Objects : routage des données vers une plateforme tierce :

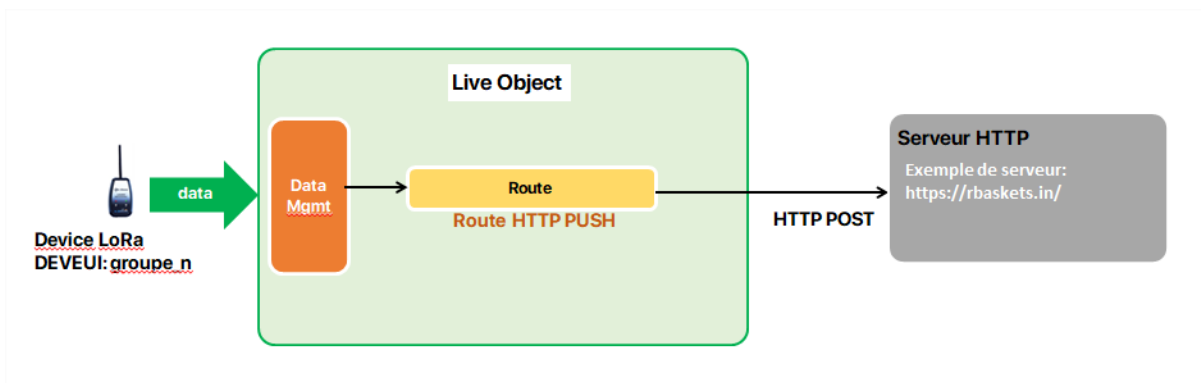


Il existe deux principales méthodes de routage des données :

- Via le protocole HTTP Push.
- Via le protocole MQTT en utilisant une file d'attente FIFO.

13.1 Récupération des données via la méthode HTTP PUSH:

L'exercice suivant consiste à router tous les messages issus de son device vers un serveur HTTP (<https://rbaskets.in/web>) via la méthode de HTTP Push :



➔ Configurer le serveur HTTP :

Créer un emplacement sur le serveur rbaskets pour récupérer les requêtes HTTP POST émises par Live Object à chaque transmission de message UL.

Adresse HTTPS de l'emplacement de réception des messages

➔ Configurer le routage HTTP Push sur la plateforme Live Objects :

Menu Données > Routage > Ajouter une règle de routage

Renseigner les informations permettant de router tous les messages de son device LoRa® vers l'adresse du serveur de destination HTTPS rbaskets en mode HTTP Push.

- ➔ **Redémarrer le device et vérifier la réception des messages LoRa® UL sur l'emplacement du serveur rbaskets.**

Pourquoi ne voit-on pas les messages JOIN Request et JOIN Accept ?

Le routage des données ne se fait que pour la payload applicative. Or, durant la procédure JOIN, il n'y a aucune payload applicative transmise par le device.

Dans Live Objects, supprimer la règle de routage créée ci-dessus, et créer une autre règle en sélectionnant cette fois-ci la bonne option permettant de voir la trace de la procédure JOIN.

13.2 Récupération des données via le protocole MQTT :

MQTT (Message Queuing Telemetry Transport):

MQTT est un protocole léger de type publish-subscribe basé sur le protocole TCP/IP.

Utilisation du port 1883 TCP pour le protocole MQTT.

Utilisation du port 8883 TCP pour le protocole MQTTS.

Plutôt que l'architecture Client / Serveur classique qui fonctionne avec des Requêtes / Réponses, MQTT est basé sur un modèle Publisher / Subscriber. La différence est importante, car cela évite d'avoir à demander (Requête) des données dont on n'a aucune idée du moment où elles vont arriver. Une donnée sera donc directement transmise au Subscriber dès lors que celle-ci a été reçue dans le Broker (serveur central).

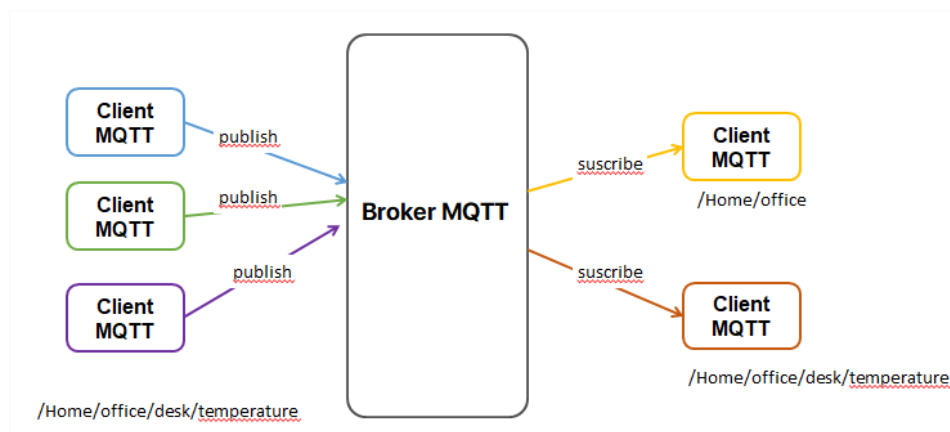
Avantages du protocole MQTT :

- Protocole de transport agnostique quant aux informations qu'il transporte, la taille maximale de payload est de 256Mo.
- Sa légèreté : n'augmente que légèrement la consommation de bande passante
- Il permet de contrôler facilement la fiabilité de transmission des informations
- Il constitue une abstraction pour la gestion du réseau : pour des connexions instables, la gestion des déconnexions/reconnexions est simplifiée
- Il permet à de nombreux clients de recevoir ou de diffuser une information
- Le chiffrement via TLS/SSL.
- La possibilité de gérer les clients pour le droit d'accès à une information ou de sa publication.

Principe de fonctionnement :

MQTT utilise le principe de Publish/Subscribe, un client publie une donnée, un client s'abonne à des informations appelées Topic.

L'ensemble des clients communiquent via un broker qui est un programme en charge de réceptionner les informations publiées et de les retransmettre aux clients abonnés.

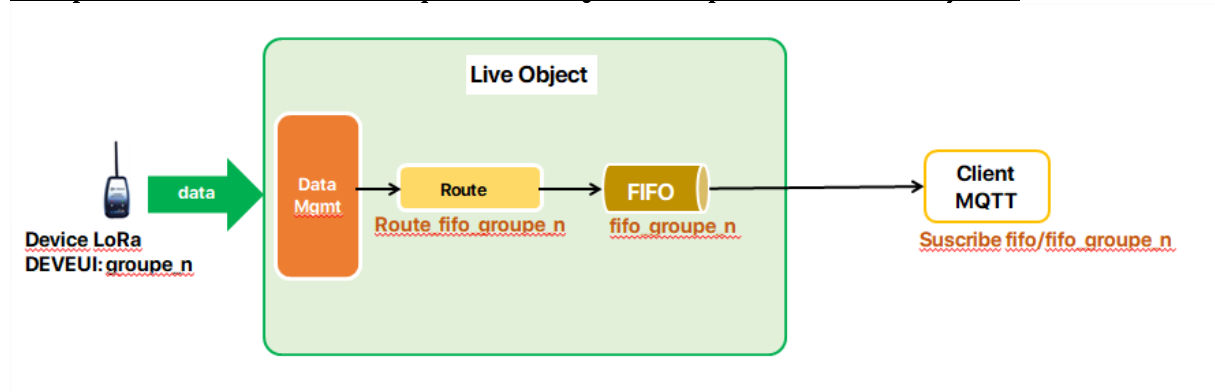


Les clients qui s'abonnent à la rubrique /Home/office recevront la température de tous les bureaux localisés dans le office.

MQTT utilise trois niveaux de qualité de service :

- Qos0 : le message est transmis une seule fois.
- Qos1 : le message est transmis au moins une seule fois.
- Qos2 : le message est transmis exactement une seule fois.

Récupération des données via le protocole MQTT sur la plateforme Live Objects :



- 1) Création d'une file d'attente FIFO.
- 2) Création du routage des messages data de son device vers la file d'attente FIFO.
- 3) Récupération des données de son device en souscrivant depuis un client MQTT à sa FIFO.

FIFO: First Input First Output.

Avantage: il garantit que les messages sont remis à l'application. Les messages sont stockés dans une file d'attente sur le disque jusqu'à ce qu'ils soient consommés et acquittés (7 jours maximum).

BONUS :

Dans Live Objects, modifier le routage vers une FIFO que vous devez créer.

Dans l'outil Swagger, retrouver l'URL permettant de voir le détail de cette pile FIFO

Copier l'URL permettant d'obtenir les infos de la pile FIFO