



Présentation et mise en œuvre LoRa et LoRaWan

Ce diaporama a été réalisé en partie à partir d'images provenant des sites :

<https://www.st.com/>

<https://www.multitech.com/>

<https://mydevices.com/>

<https://www.thethingsnetwork.org/>

<https://lora-alliance.org/>

<https://fr.wikipedia.org/wiki/LoRaWAN>

<https://www.semtech.com/>

https://www.mobilefish.com/developer/lorawan/lorawan_quickguide_tutorial.html

Internet of Things - (IoT)

On prévoit en 2022 plus de 50 milliards d'objets sans fil connectés à Internet.

- Liaisons courte distance, WIFI, Bluetooth, zigbee.
- Liaisons cellulaire , 4G et surtout 5G
- Liaisons longue distance LPWAN (low-power wide-area network)

Sigfox (compagnie française)

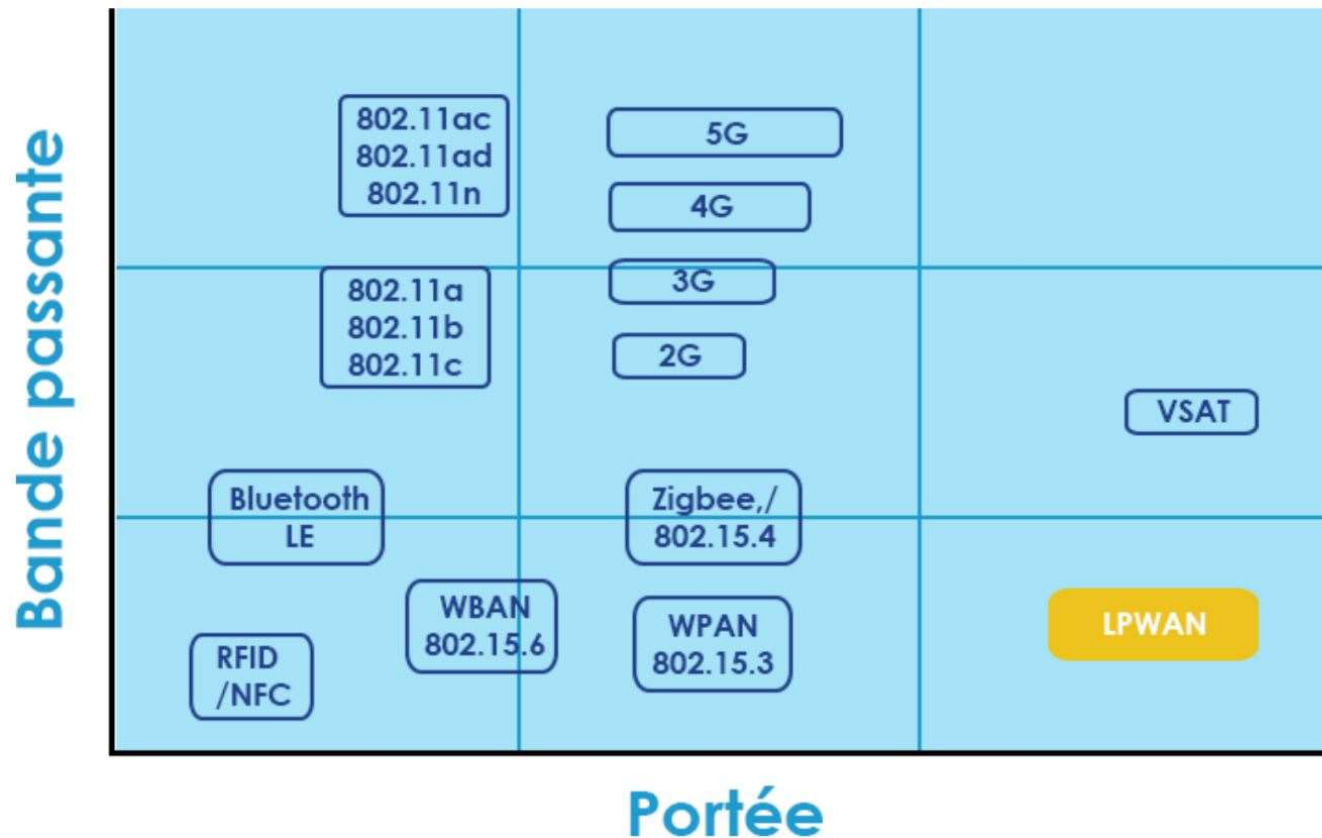
Nware (MIT).

Telensa (Cambridge University)

LoRa (Breveté à l'origine par la compagne française Cycleo à Grenoble, rachetée par **Semtech** USA)

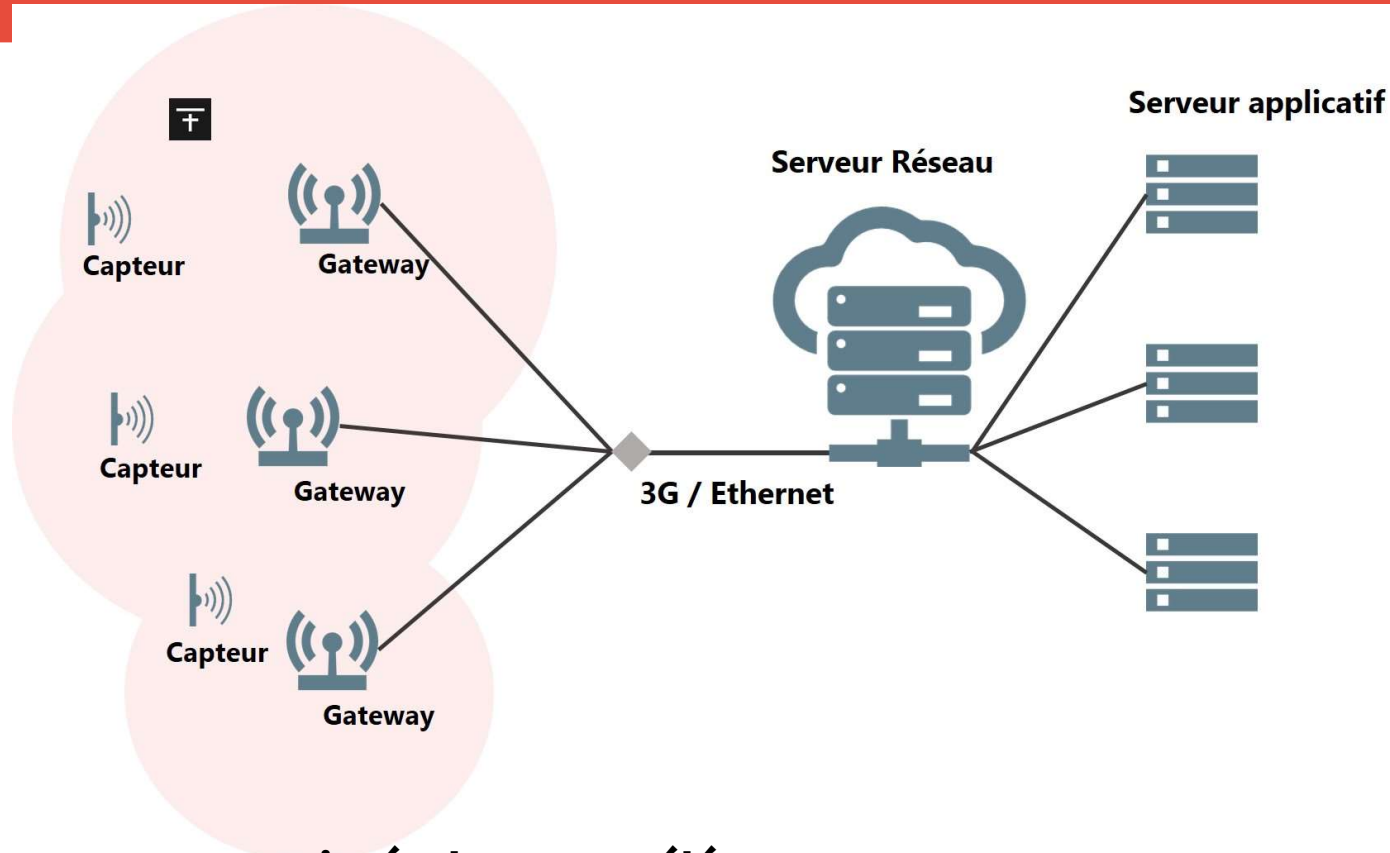


Technologies sans fil. Comparatif LPWAN vs les autres



LPWAN : moins de débit mais plus de portée

Structure d'un réseau LPWAN



Les réseaux LPWAN sont constitués de quatre éléments :

- Les capteurs/actionneurs généralement appelés « nodes » qui communiquent par radio
- Les passerelles (gateways) qui établissent le lien entre les communications radios et Internet
- Le serveur réseau qui sécurise et stocke les données
- le serveur d'application finale qui présente les données à l'utilisateur

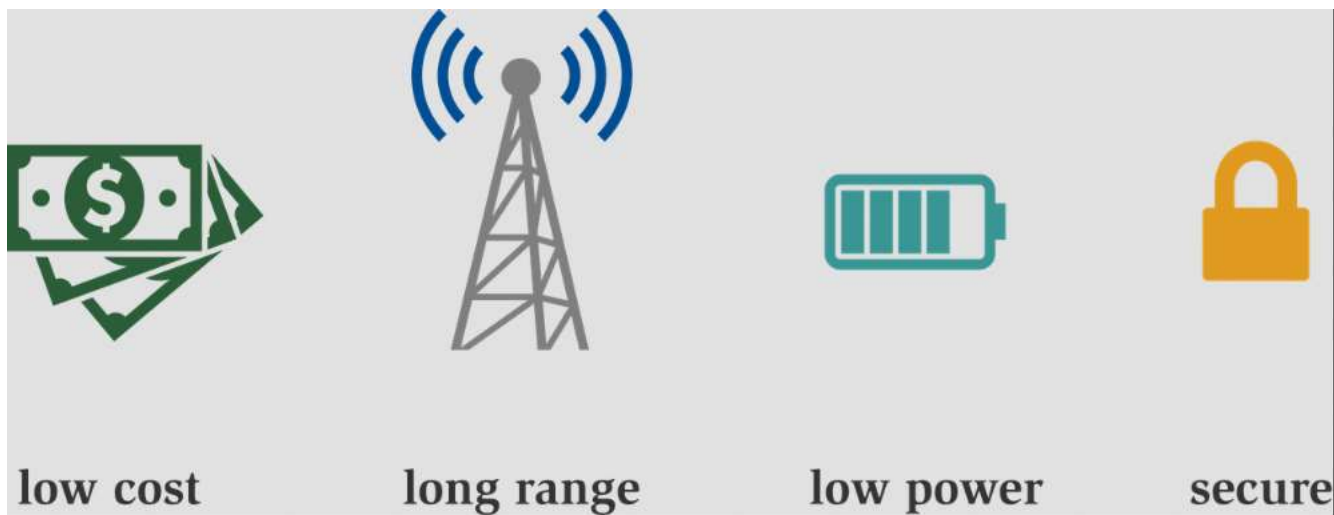
Réseaux LPWAN : LoRa et Sigfox

Les réseaux LPWAN :

Portées > 10Km

Puissance d'émission très faible, ce qui entraîne une grande autonomie des nœuds (node) > 1an

Grande portée et faible puissance limite le débit binaire



LoRa vs sigfox



Sigfox est prévu pour envoyer 140 messages de 12 octets à 300 bauds par jour et recevoir 4 messages par jour.

LoRa envoie des messages plus longs (5KO) et sans limitation.

LoRa optimise dynamiquement le lien entre l'objet (node) et la passerelle (gateway) ce qui permet des portées plus grandes (plus de 15Km en plaine)

Le protocole **LoRa** est sous licence Semtech mais s'appuie sur un réseau « libre »

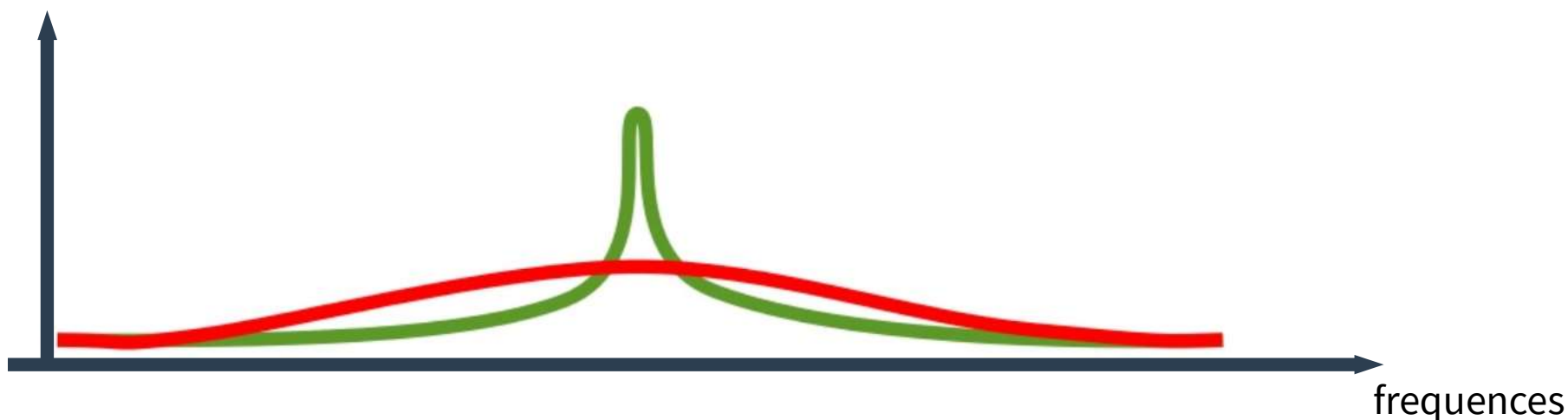
Sigfox utilise les réseaux privés des opérateur téléphoniques

LoRa et **Sigfox** utilise en Europe la bande des 868MHz

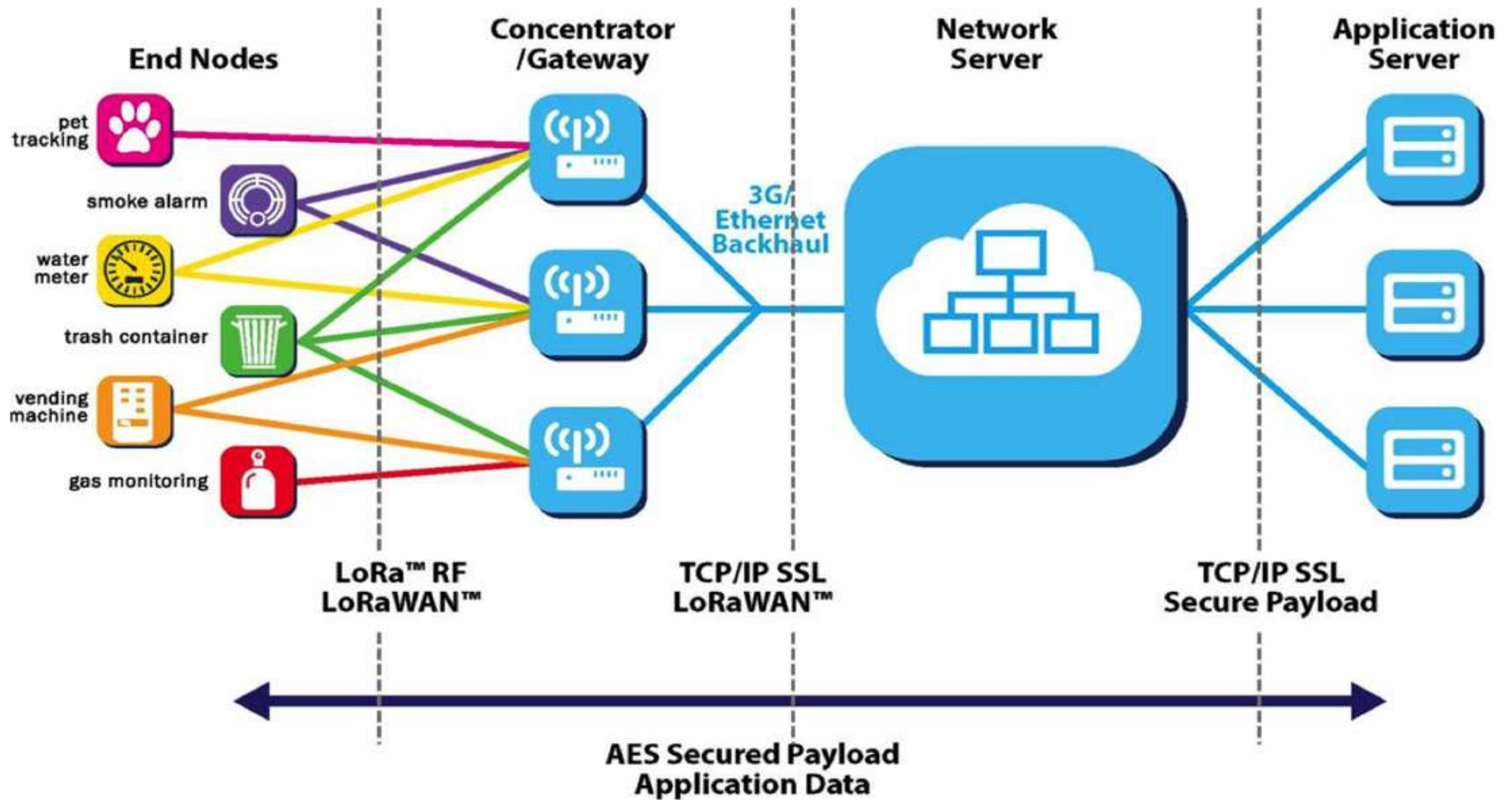
Technologies LoRa / Sigfox

La radio LoRa repose sur une transmission à étalement de spectre. **Faible amplitude mais spectre large.**

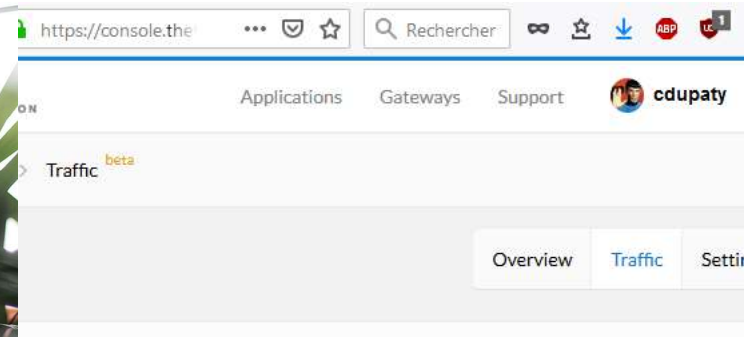
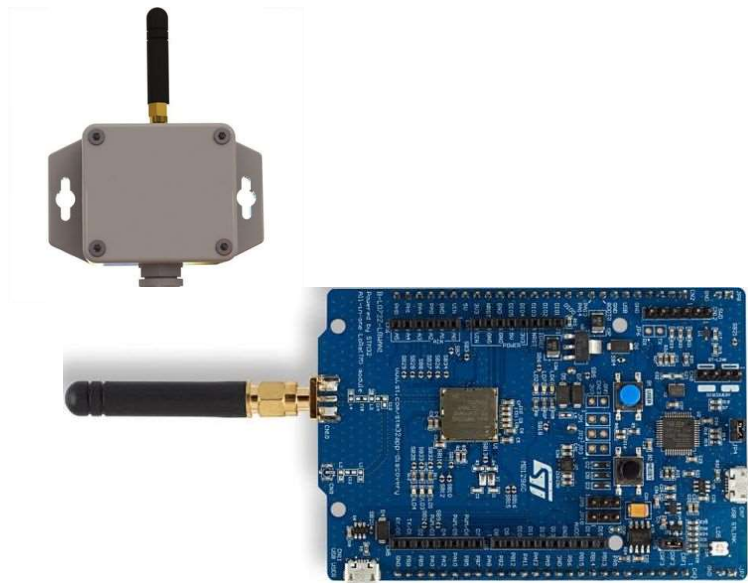
La radio Sigfox repose sur une transmission sur un **spectre de fréquences très étroit (Ultra Narrow Band)** mais avec plus de puissance.



Le réseau LoRaWan

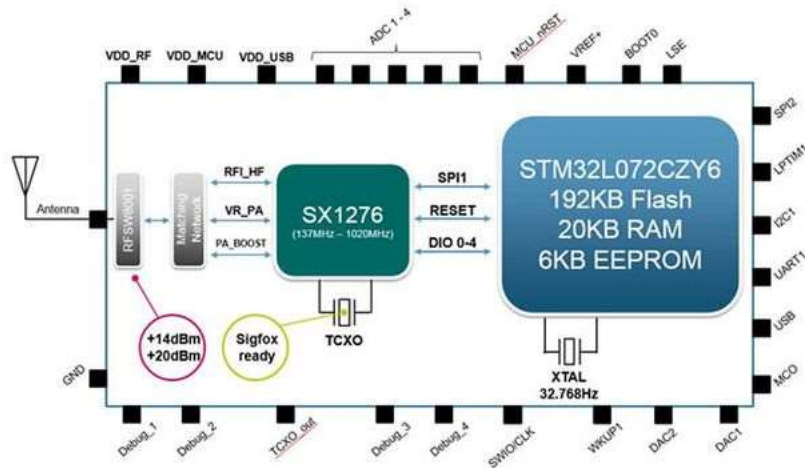


Node – Gateway – Network Server



GATEWAY TRAFFIC

uplink	downlink	join	0 bytes					pause	clear
time	frequency	mod.	CR	data rate	airtime (ms)	cnt			
17:36:28	869.525	lor	4/5	SF 9 BW 125	144.4	32	dev addr: 26 01 2F A8	payl	
17:36:27	868.1	lor	4/5	SF 12 BW 125	2138.1	25	dev addr: 26 01 2F A8	payl	
17:33:36	869.525	lor	4/5	SF 9 BW 125	144.4	31	dev addr: 26 01 2F A8	payl	
							dev addr: 26 01 2F A8	payl	
							dev addr: 26 01 2F A8	payl	
							dev addr: 26 01 2F A8	payl	



Reglementation



Le rapport cyclique (Duty Cycle) correspond au temps d'émission par rapport au temps total entre deux émissions.

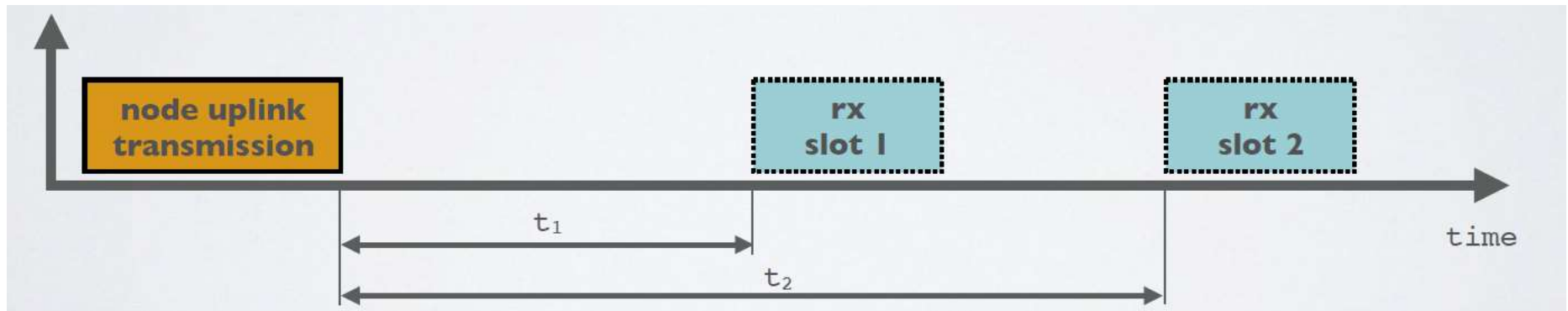
En Europe il est compris en fonction des canaux radio entre 0,1 % et 1 %

LoRa utilise la bande publique des 868MHz

LoRa classe A

Un node peut transmettre un message **n'importe quand**

Une réponse peut être transmise pas la passerelle après les temps t_1 ou t_2 , généralement 1s et 2s.

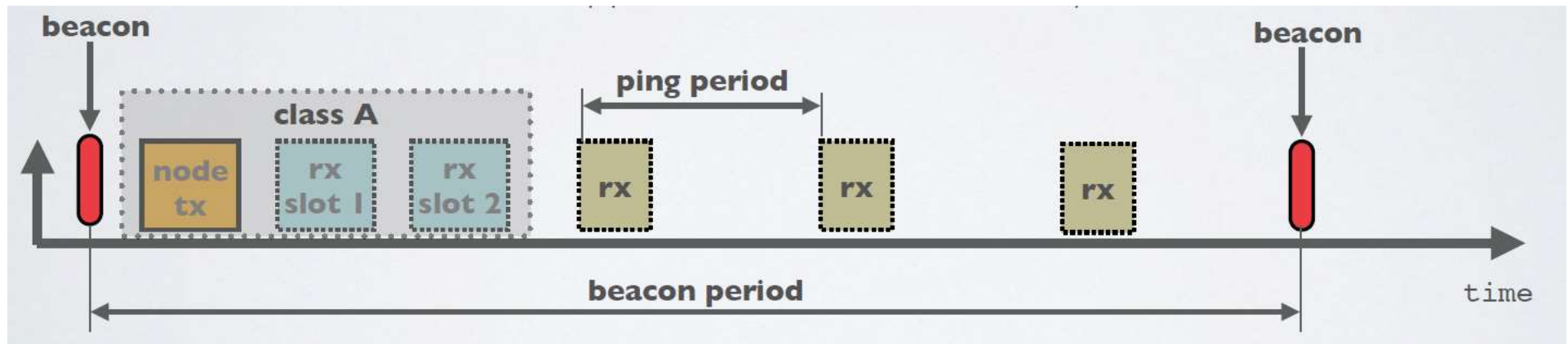


LoRa classe B

Même fonctionnalités que la classe A

Le node peut recevoir plusieurs réponses.

Le échanges commencent lorsque la passerelle transmet une balise (beacon)



LoRa Classe C

Même fonctionnalités que la classe A

Le node écoute durant toute la période entre deux transmissions. **Cela induit une plus grande consommation**



Concepts radio

La portée et la qualité d'une liaison radio dépendent de très nombreux facteurs.

En particulier

Le RSSI

Le SNR

Le SF

La BW

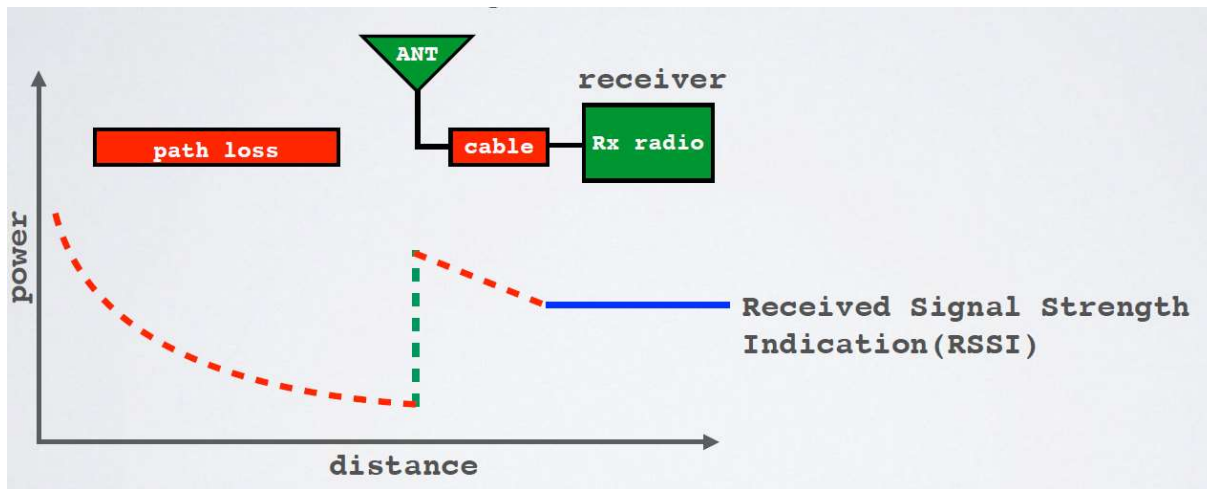


Les diapositives suivantes rappellent ces notions

RSSI

Received Signal Strength Indication (RSSI)

Indication de l'intensité du signal reçu



Le RSSI est mesuré en dBm et est toujours négatif
Un RSSI de 0 correspond à une réception maximale

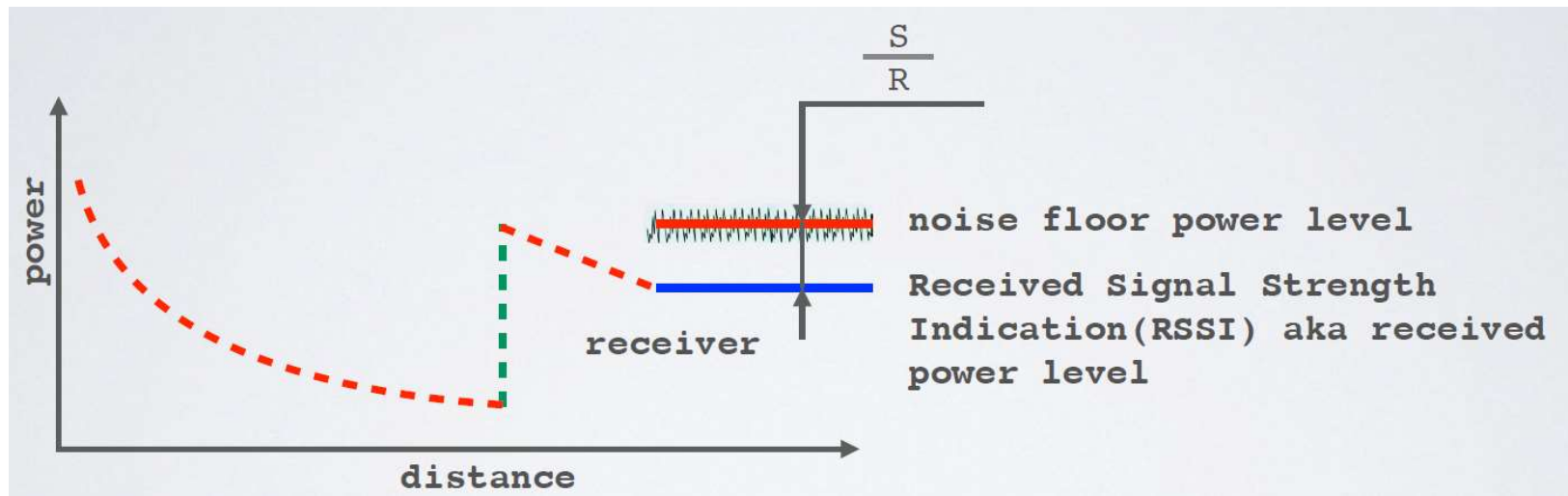
Sur LoRa le RSSI est compris entre -30dBm et -120dBm

SNR

Signal to Noise Ratio (SNR) ou Rapport signal/bruit

C'est le rapport entre la puissance du signal reçu et le niveau du bruit. Plus ce rapport est grand et plus il sera facile d'extraire le signal du bruit.

La technologie LoRa permet de démoduler un signal sous le niveau de bruit. (-7.5dBm à -20dBm sous le bruit)



Spreading Factor et BW

Band With (BW) : largeur de bande. Le BW représente la largeur de la modulation de fréquence autour de la porteuse

Spreading factor (SF): L'étalement de spectre représente l'excursion en fréquence utilisée pour transmettre un motif(généralement un octet)

L'augmentation du Spreading Factor permet de couvrir une distance plus grande entre l'équipement et la passerelle au détriment de la bande passante disponible.

Plus le motif est étalé, plus le débit baisse mais plus la portée augmente.

Data Rate vs SF vs BP

Data Rate (DR)	Spreading Factor (SF)	Bande Passante	Débit Physique (bit/s)
1	SF11	125KHz	440
2	SF10	125KHz	980
3	SF9	125KHz	1760
4	SF8	125KHz	3125
5	SF7	125KHz	5470
6	SF7	250KHz	11000

Bandes passantes européennes

LoRaWAN frequencies used in Europe
TTN freq. plan: EU863-870

Bandwidth

- 125 kHz @ SF 1-7, for downlink slot 1
- 250 kHz @ SF7, for downlink slot 1
- 125 kHz @ SF9 for downlink slot 2

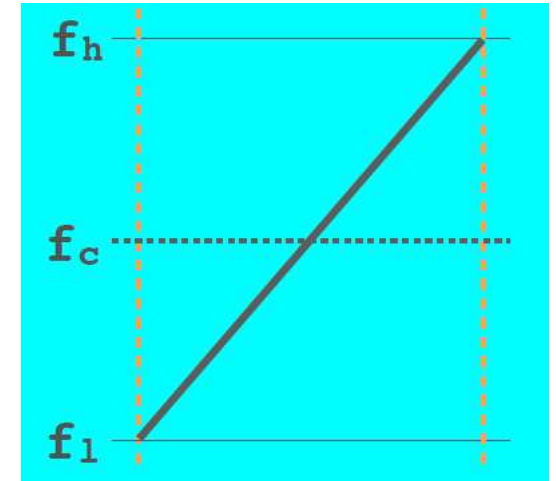
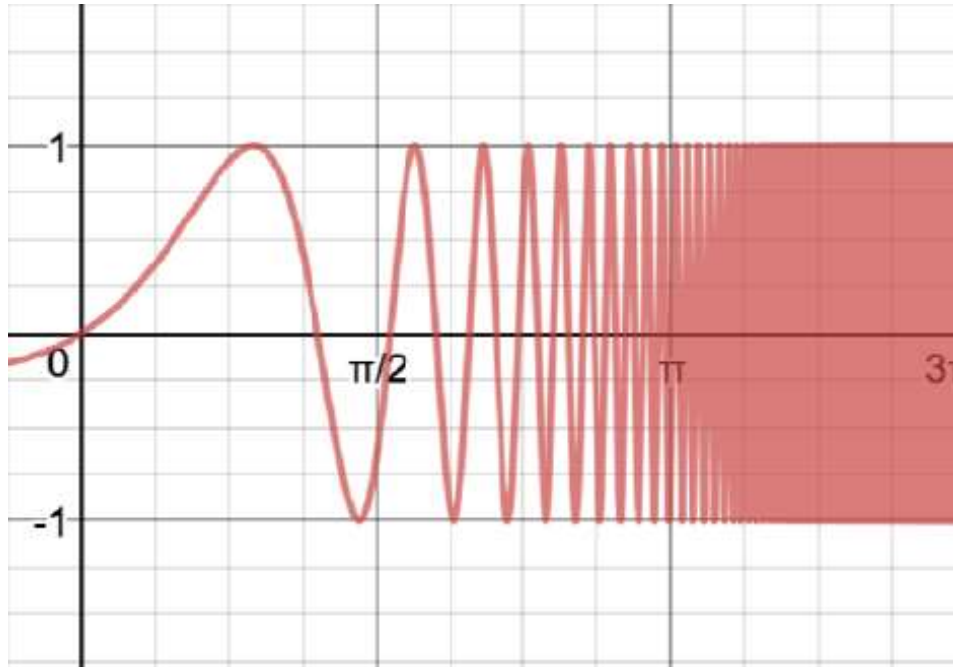
Uplink frequencies



Downlink frequencies



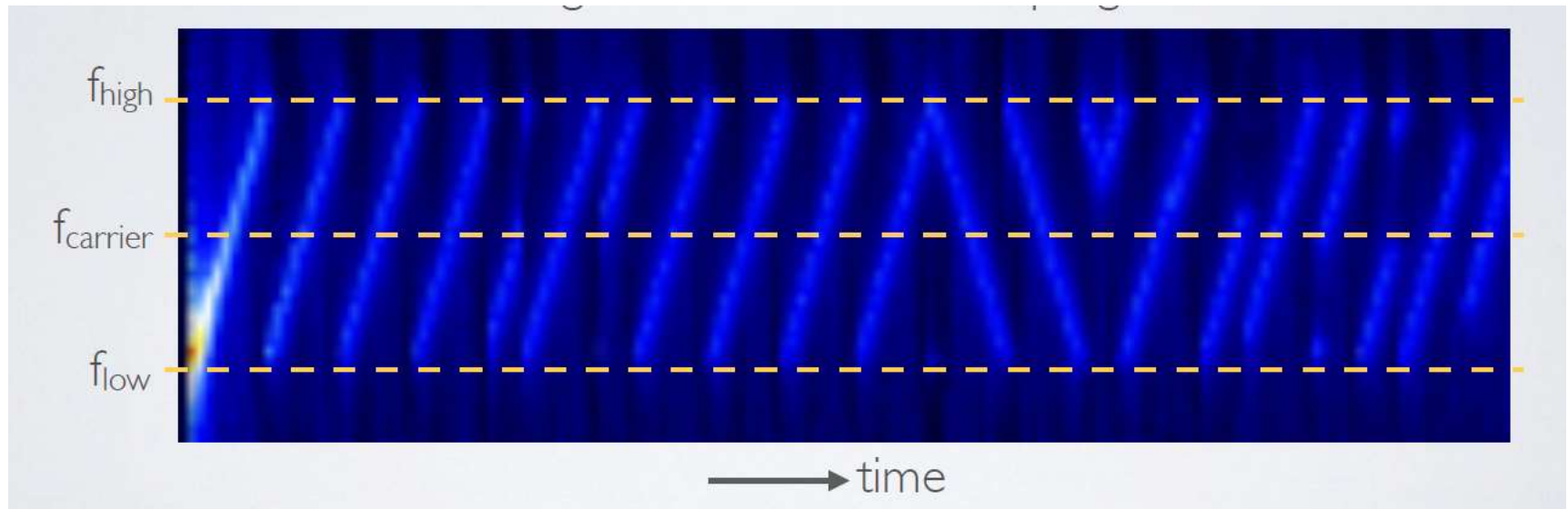
Modulation de fréquence et modulation LoRa



La modulation LoRa est une modulation de fréquence appliquée aux transmissions numériques.

Contrairement à la FSK qui possède deux fréquences, une pour le 0 logique et une pour le 1. LoRa repose sur des sauts de fréquences qui représentent des motifs (des groupes de bits). La modulation est adaptative

Modulation LoRa

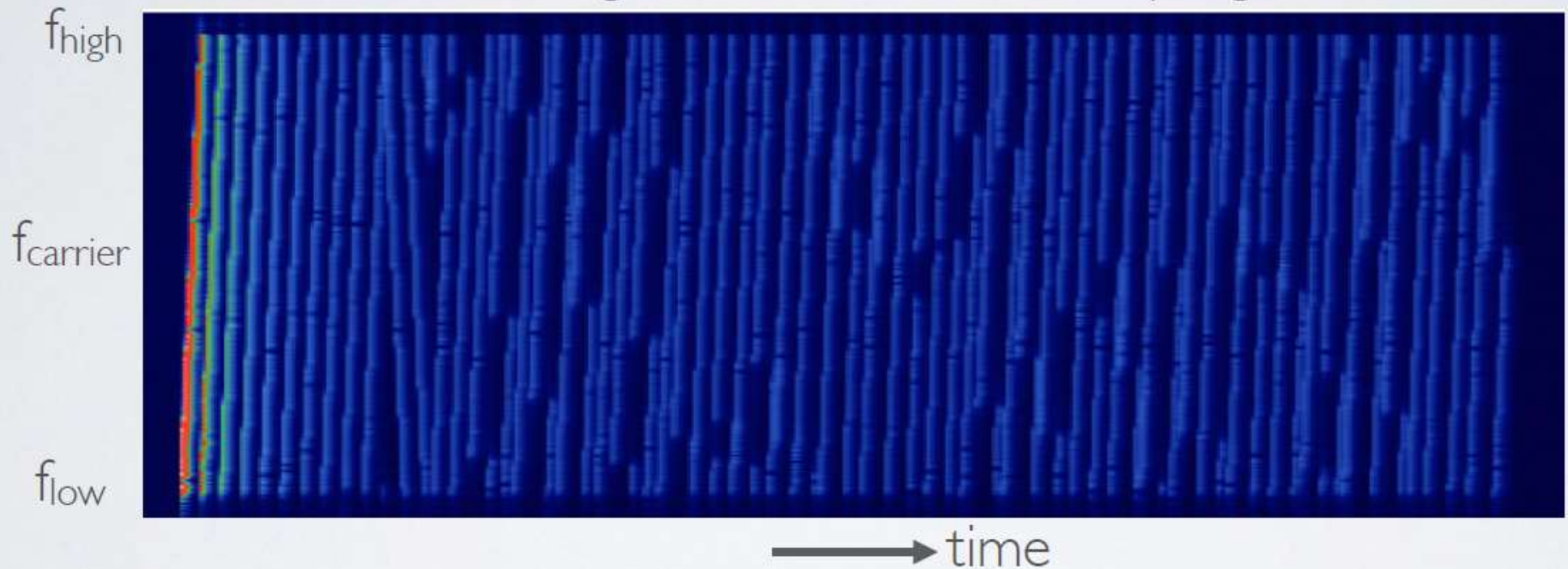


On voit ici les sauts de modulations montante ou descendantes représentant les motifs

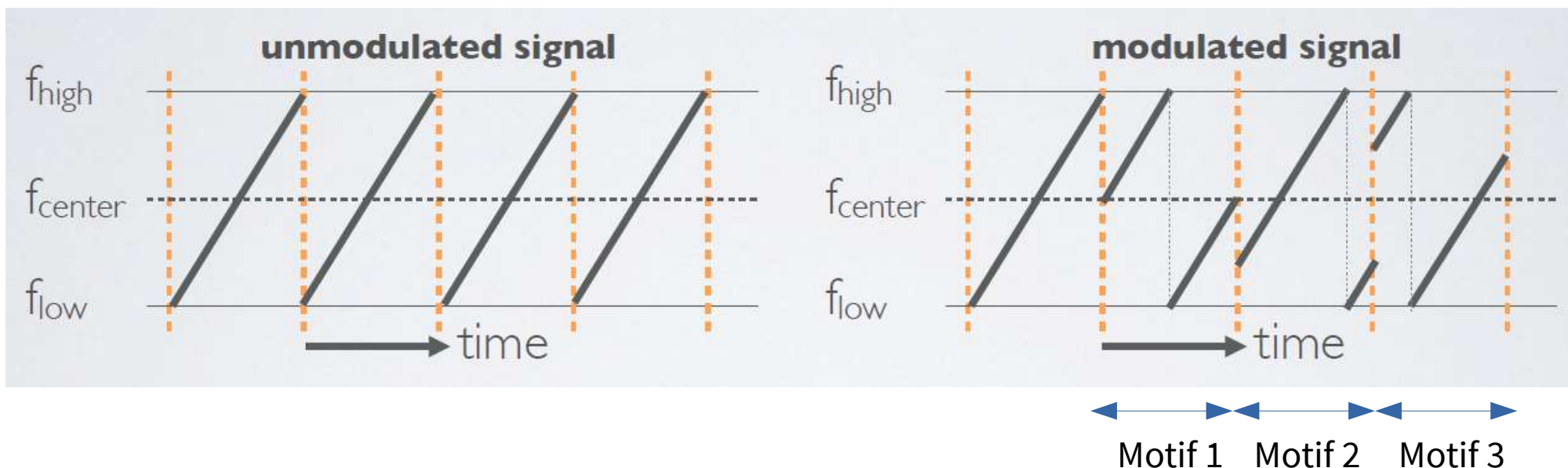
Trame LoRa complète

Preamble	Header (Optional)	Payload	Payload CRC (Optional)
----------	----------------------	---------	---------------------------

Message encoded on the chirp signals



Modulation LoRA



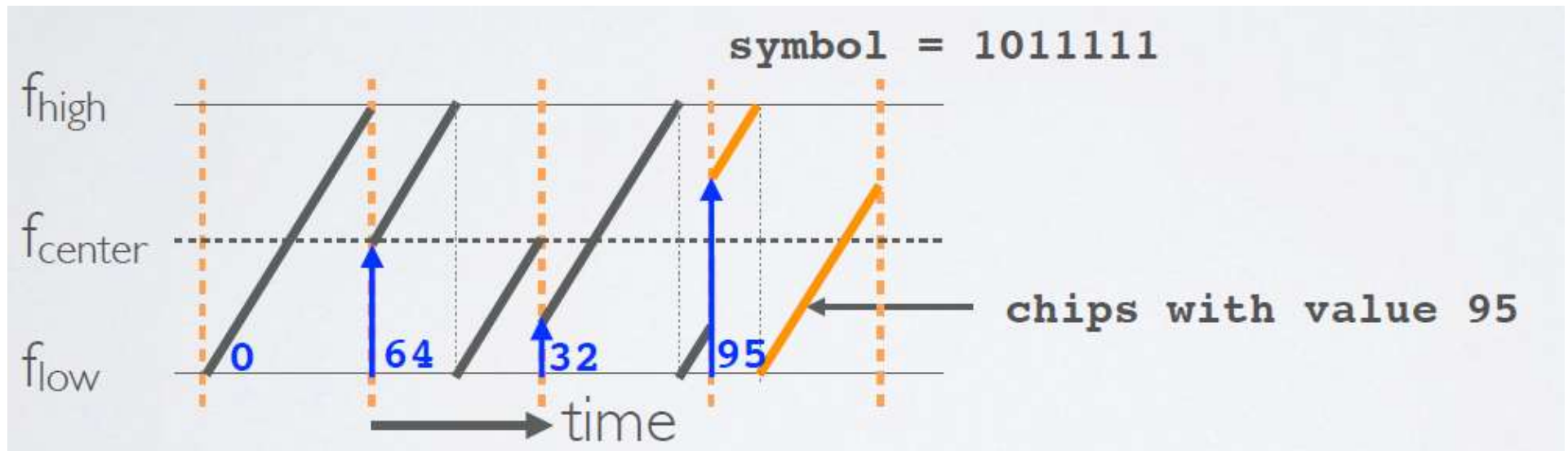
Chaque motif est représenté par une fréquence de début d'étalement.
Arrivée à f_{high} , la fréquence bascule sur f_{low} pour rejoindre la fréquence de départ du motif

Exemple de modulation LoRa

Par exemple si le motif est : 1011111 (95 en décimal)

Le motif peut être codé avec 7 valeurs (7 bits) (SF=7)

La variation de fréquence est divisée en 2^{SF} pas = $2^7 = 128$ pas ou chips.



Coding Rate : CR

Un motif est accompagné de bits permettant une détection d'erreur

Le CR définit le rapport entre le nombre de bits portant l'information et le nombre de bits total . **Plus le CR est bas et plus il sera facile de détecter les erreurs**

CR = 4 / (4 + CR) ou CR = 1.2.3.4

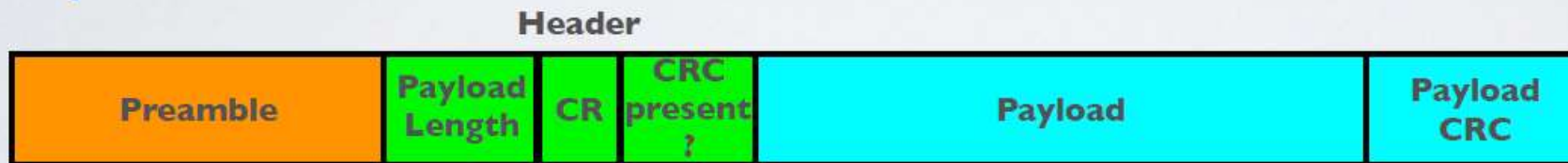
Coding Rate (CR)	CR = 4 / (4 + CR)
1	4/5
2	4/6
3	4/7
4	4/8

Format d'un packet LoRa

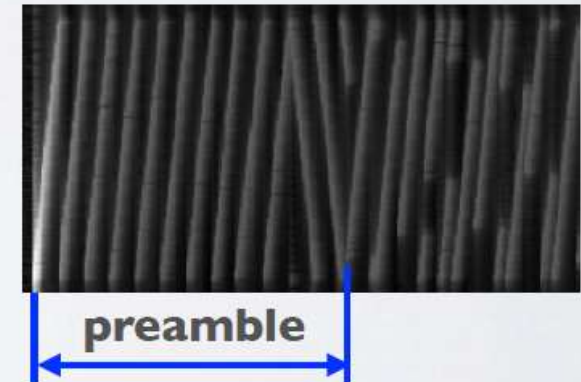
La détection d'erreur est optionnelle

Le Payload Length également mais il faudra que la longueur du message soit fixe et connu du récepteur

Explicit header mode



Implicit header mode



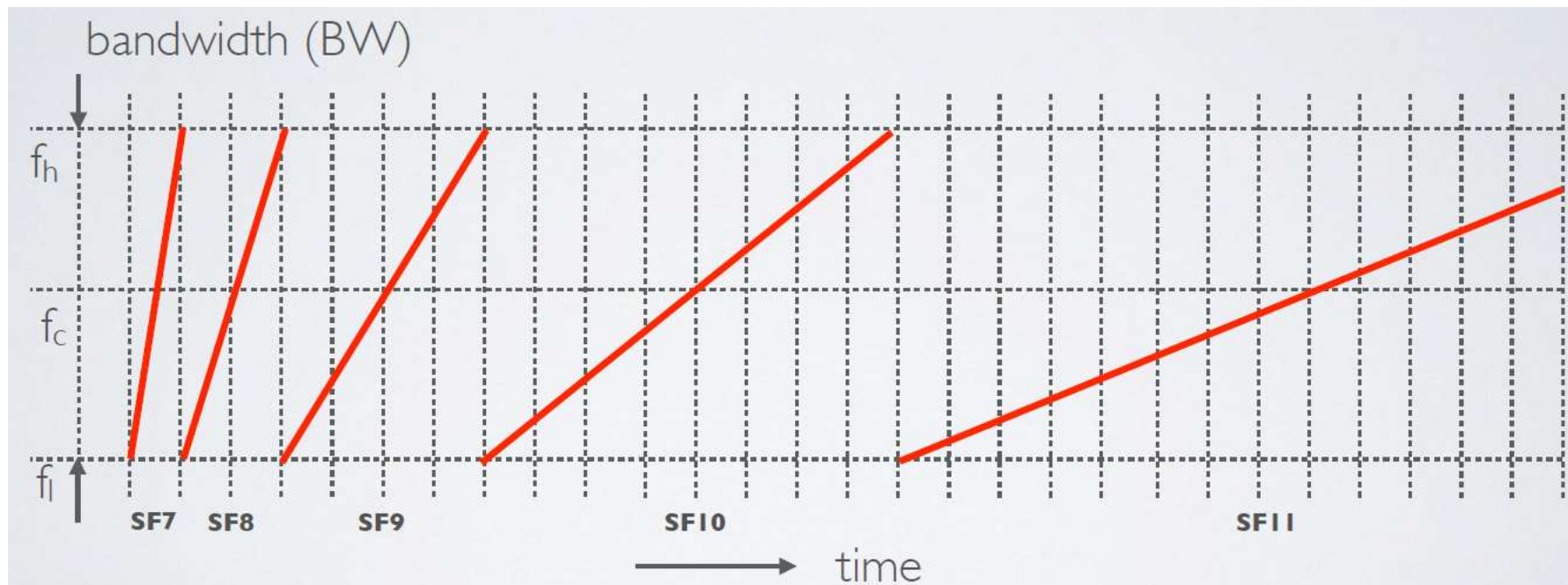
Time on Air (TOA)

La portée est inversement proportionnelle à la vitesse de modulation.

La modulation est auto-adaptative.

Il est nécessaire de réduire la vitesse de modulation si la distance augmente et donc d'augmenter le TOA.

Cela a pour conséquence de diminuer le débit binaire (bits/sec)



SF et TOA

Exemple :

Préambule = 8bits

Explicite header activé

CRC activé

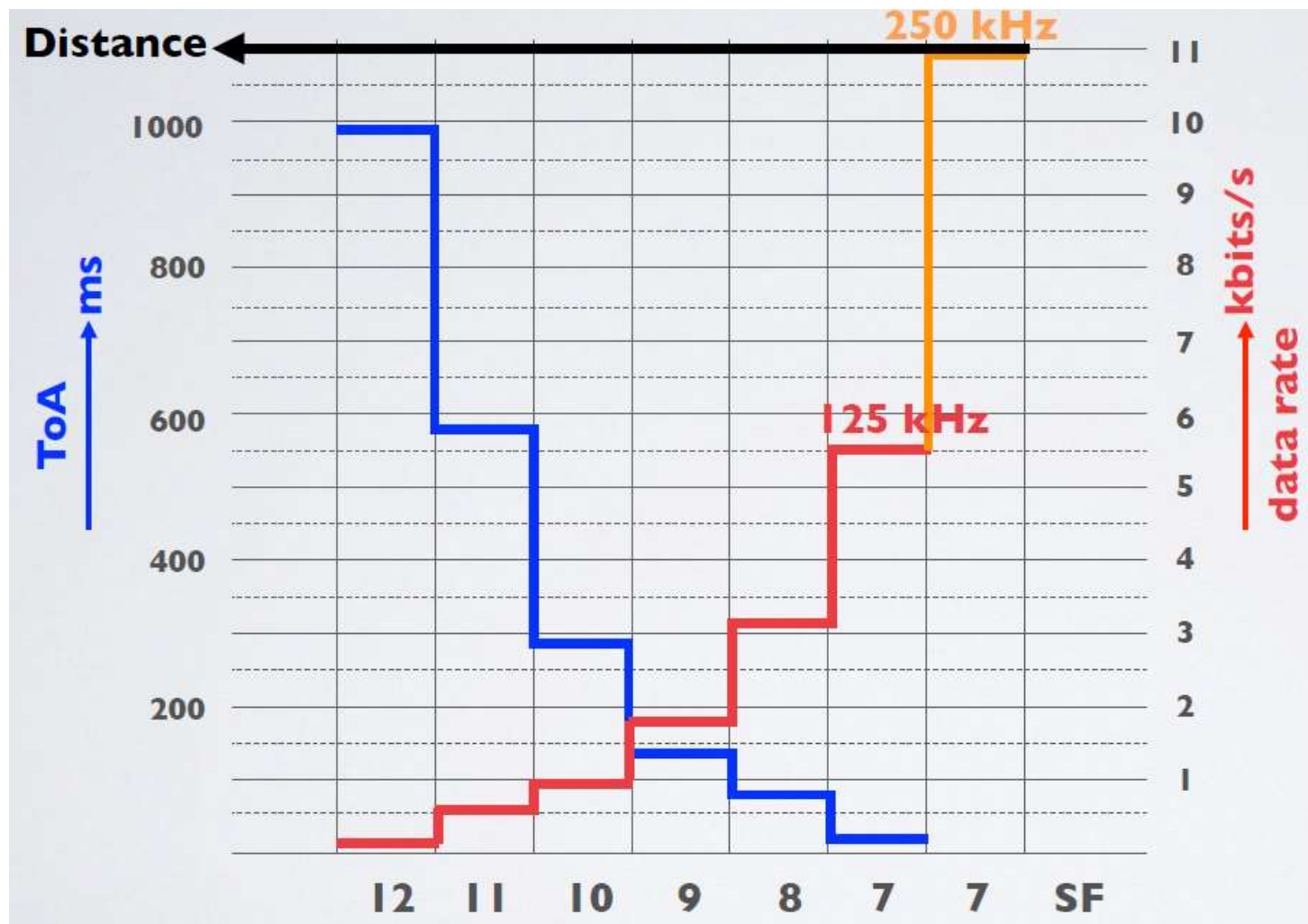
Charge (payload)=10 octets

CR=1

BW=125KHz

Spreading Factor	Time on Air (ms)
7	41
8	72
9	144
10	289
11	578
12	991

Time On Air vs Data Rate / Spreading Factor



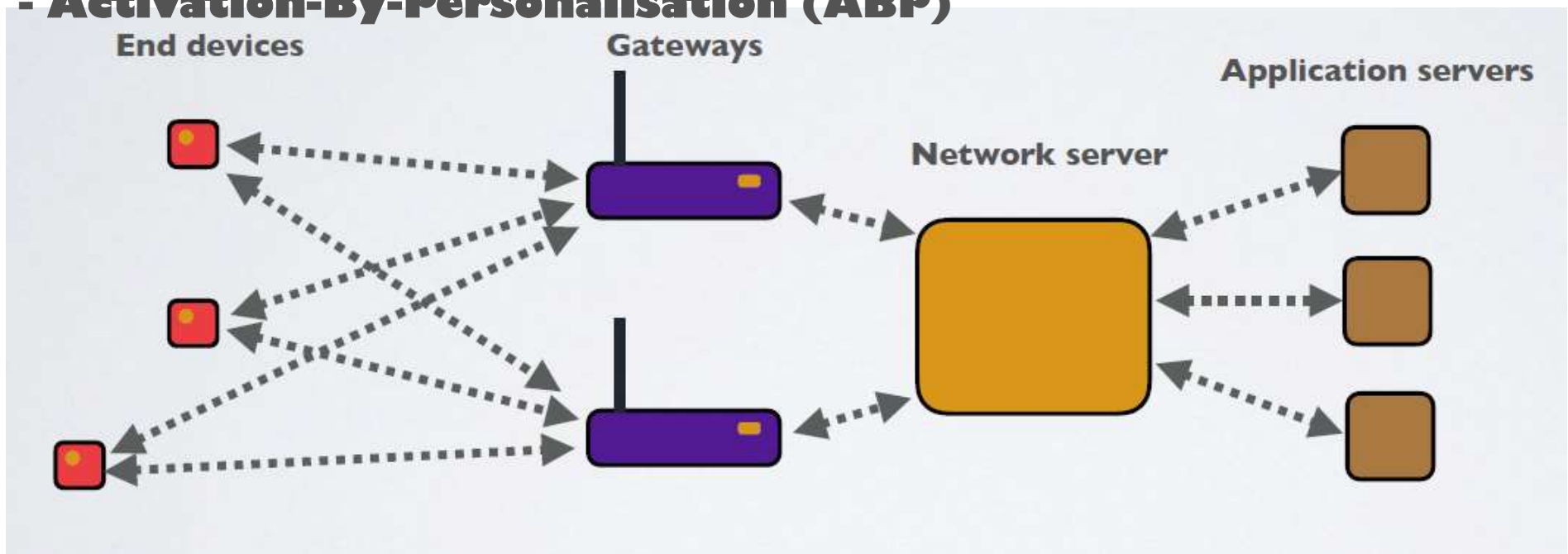
Sécurité LoRa → LoRaWan

Le node (nœud final) doit être activé et reconnu par le serveur

Deux méthodes sont possibles

- **Over-The-Air-Activation (OTAA)** cette méthode apporte plus de sécurité

- **Activation-By-Personalisation (ABP)**



Identifiants

EUI (Extended Unique Identifier) est un nombre codé sur 64 bits (8 octets)

DevEUI est un nombre unique qui identifie le **node** (comme une adresse MAC)

il est donné par le fabriquant du node mais peut être modifié

DevAddr est l'adresse du node (**comme une adresse IP**) **OTAA**

AppEUI identifie le serveur d'application (**comme un port IP**)

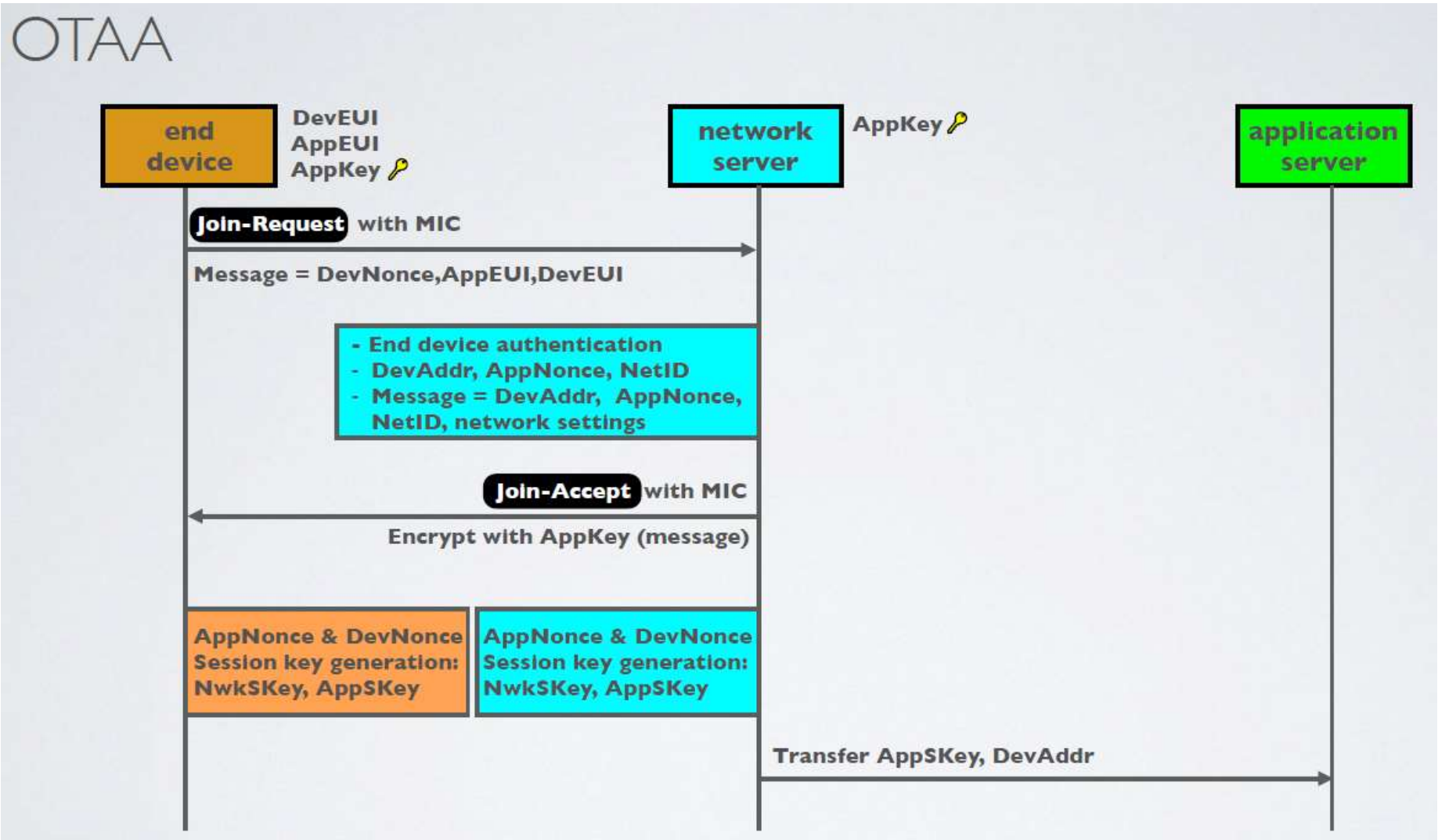
NwkSKey est une clef utilisée par le **serveur et le node** pour calculer et vérifier le champ MIC qui permet le calcul d'intégrité du paquet.

OTAA

AppSKey est une clef utilisée par le **serveur et le node** pour chiffrer et déchiffrer les données des paquets **OTAA**

AppKey AES (Advanced Encryption Standard) est une clé symétrique codée sur 128 bit destinée à crypter les messages **entre le node et le serveur d'application sur Internet.** **OTAA**

OTAA – Other The Air Activation



LoRaWan sécurité

Deux couches de sécurité :

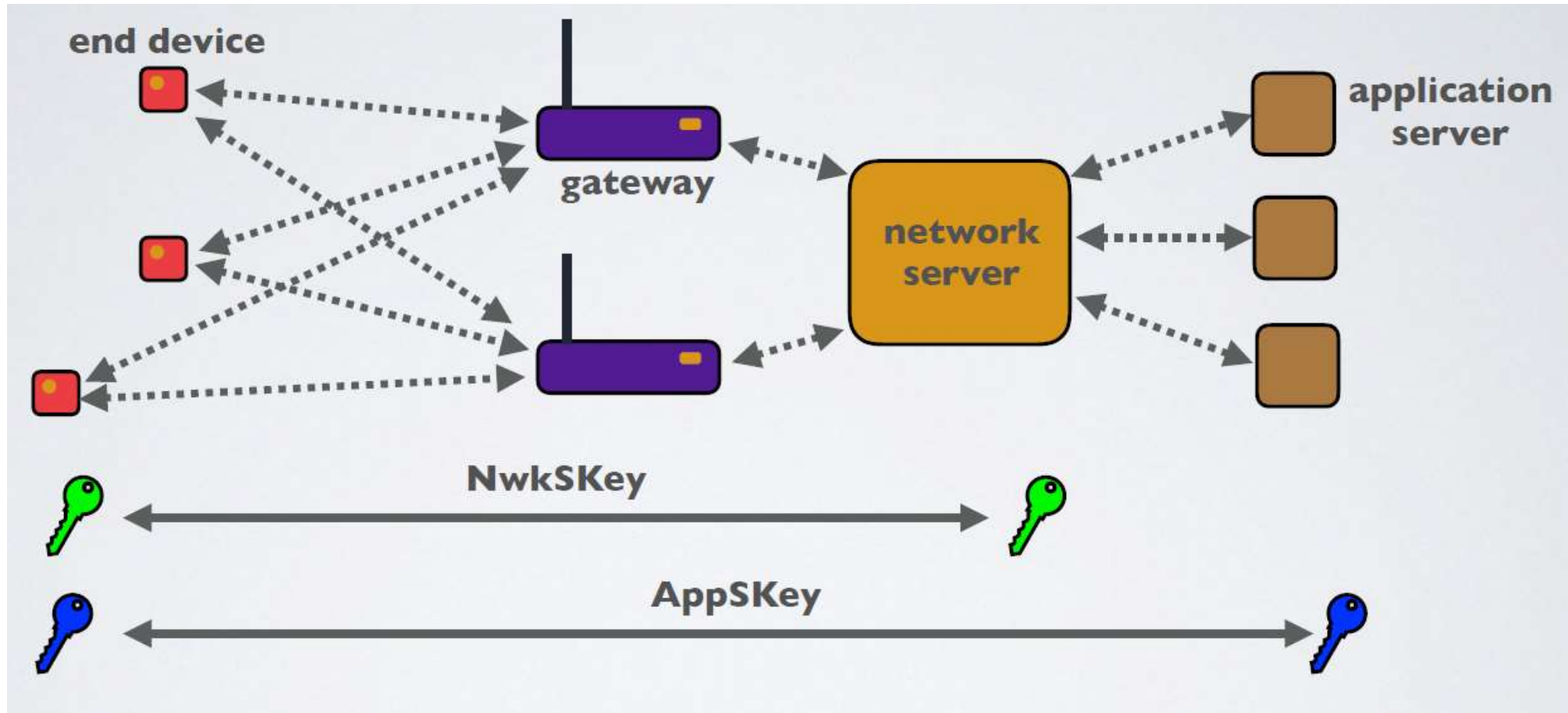
Sur la couche réseau le message est crypté avec le Message Integrity Code (MIC) et la clé NwkSKey.

La passerelle reçoit et transmet un code crypté au serveur

Les données utiles (Payload) sont sur-cryptées depuis le node jusqu'au serveur d'application avec la clé AppSKey.

Le serveur transmet le payload crypté à l'application

Clés OTAA



Les configurations :

Exemple pour un seveur The Things Network (TTN)

ADD APPLICATION

Application ID
The unique identifier of your application on the network

humidite_du_champs_de_fraises

Description
A human readable description of your new app

Les fraises doivent toujours etre bien aroseees

Application EUI
An application EUI will be issued for The Things Network block for convenience, you can add your own in the application settings page.

EUI issued by The Things Network

Handler registration
Select the handler you want to register this application to

ttn-handler-eu

Cancel Add application

TTN fournit Application EUIS et ACCES KEYS

Application EUIS est le code de l'application pouvant regrouper plusieurs nodes

The screenshot displays the TTN Application EUIS interface. It features four main sections: **APPLICATION EUIS** with a hex address `70 B3 D5 7E D0 01 55 BA` and a `manage euis` link; **DEVICES** showing `0 registered devices` with `register device` and `manage devices` links; **COLLABORATORS** listing `cdupaty` with `collaborators`, `delete`, `devices`, and `settings` buttons; and **ACCESS KEYS** showing a `default key` with `devices` and `messages` buttons, and a key value `ttn-account-v2.ngkrPhBZM0qRedLdIupgJ6F9rN6MXeuoupLtMq65YTI` in base64 format with a `manage keys` link.

Acces Keys est le code utilisé par l'application finale pour accéder à la base de données

Ajout de devices (nodes)

REGISTER D

C'est l'utilisateur qui choisit le device ID

[bulk import devices](#)

Device ID

This is the unique identifier for the device in this app. The device ID will be immutable.

1234567890

Device EUI

The device EUI is the unique identifier for this device on the network. You can change the EUI later.



this field will be generated

App Key

The App Key will be used to secure the communication between you device and the network.



this field will be generated

App EUI

70 B3 D5 7E D0 01 55 BA

L'App EUI a laquelle appartient le node

Cancel

Register

Fin de configuration du TTN

DEVICE OVERVIEW

Application ID `humidite_du_champs_de_fraises`

Device ID 1234567890

Activation Method `OTAA`

Device EUI `<> ⇅ 00 2F C3 1F 39 4A EB 31`

Le numero du node

Application EUI `<> ⇅ 70 B3 D5 7E D0 01 55 BA`

L'App EUI a laquelle appartient le node

App Key `<> ⇅ 62 81 2E BC 30 50 D8 40 67 82 99 A5 A5 29 FE C2`

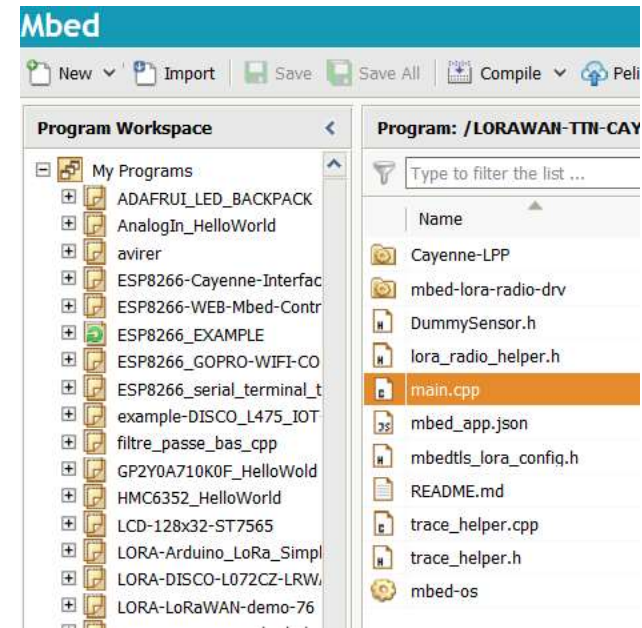
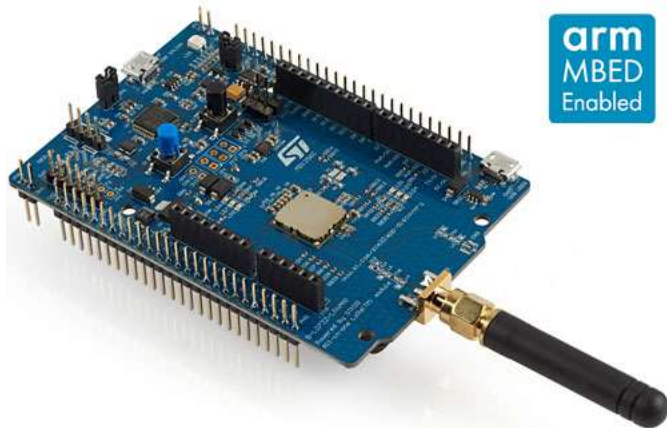
La clé de cryptage du node au Serveur TTN

Status ● *never seen*

Frames up 0 [reset frame counters](#)

Frames down 0

Configuration du Node (ex ST B-L072Z-LRWAN1)



Créer un compte sur mbed.com pour importer l'exemple ci dessous dans le « compiler »
Récupérer l'exemple <https://os.mbed.com/teams/mbed-os-examples/code/mbed-os-example-lorawan/>

 [mbed-os-examples /](https://os.mbed.com/teams/mbed-os-examples/)  [mbed-os-example-lorawan](https://os.mbed.com/teams/mbed-os-examples/code/mbed-os-example-lorawan/)

Configuration LoRaWan, fichier mbed_app.json

Après avoir sélectionné la cible dans le compiler Mbed, modifier mbed_app.json comme suit

```
"target_overrides": {  
  "*": {  
    "platform.stdio-convert-newlines": true,  
    "platform.stdio-baud-rate": 115200,  
    "platform.default-serial-baud-rate": 115200,  
    "lora.over-the-air-activation": true,  
    "lora.duty-cycle-on": true,  
    "lora.phy": "EU868",  
    "lora.device-eui": "{ 0x00, 0x2F, 0xC3, 0x1F, 0x39, 0x4A, 0xEB, 0x13 }",  
    "lora.application-eui": "{ 0x70, 0xB3, 0xD5, 0x7E, 0xD0, 0x01, 0x55, 0xAB }",  
    "lora.application-key": "{ 0x62, 0x81, 0x2E, 0xBC, 0x30, 0x50, 0xD8, 0x40, 0x67, 0x82, 0x99,  
0xA5, 0xA5, 0x29, 0xFE, 0x2C }"  
  },  
}
```

Adaptation du programme de l'application

```
static void send_message()  
{  
    uint16_t packet_len;  
    int16_t retcode;  
    float sensor_value;  
    if (ds1820.begin()) {  
        ds1820.startConversion();  
        sensor_value = ds1820.read();  
        printf("\r\n Dummy Sensor Value = %3.1f \r\n", sensor_value);  
        ds1820.startConversion();  
    } else {  
        printf("\r\n No sensor found \r\n");  
        return;  
    }  
    packet_len = sprintf((char*) tx_buffer,  
"Dummy Sensor Value is %3.1f",sensor_value);  
    retcode = lorawan.send(MBED_CONF_LORA_APP_PORT, tx_buffer,  
packet_len,MSG_CONFIRMED_FLAG);
```

Reception des données sur TTN

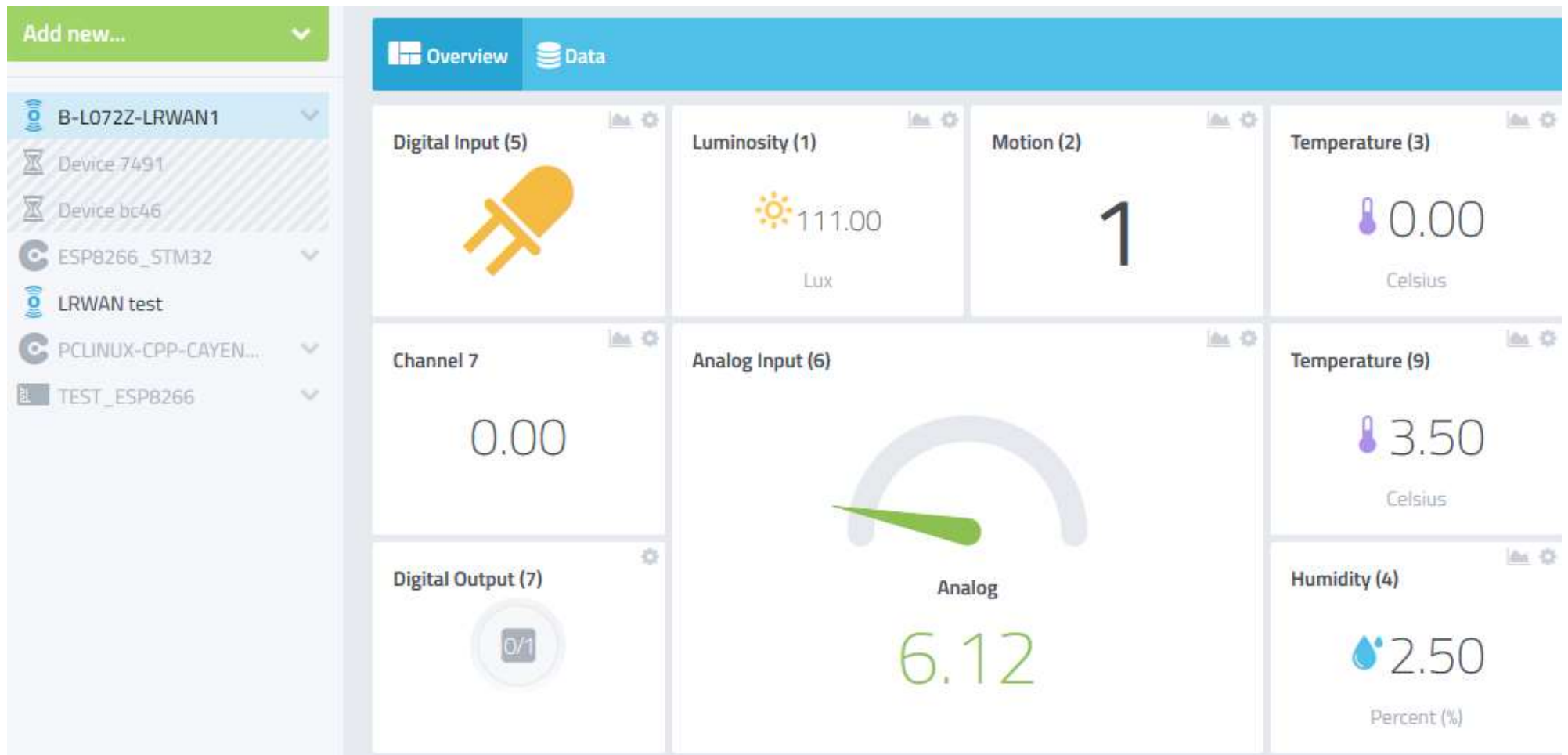
APPLICATION DATA

|| pause clear

Filters							
					uplink	downlink	activation
					ack	error	
					time	counter	port
▼	19:10:44		0		dev id: test		
▲	19:10:43	3	15	confirmed	dev id: test	payload: 01 65 00 6F 02 66 01 03 67 00 00 04 68 05 05 00 01 06 02 02 58 07 01 00 08 67 00 C7	
<					>		
▼	19:10:43		0		dev id: test		
▲	19:10:42	2	15	confirmed	dev id: test	payload: 01 65 00 6F 02 66 01 03 67 00 00 04 68 05 05 00 01 06 02 02 58 07 01 00 08 67 00 C7	
<					>		
▼	19:10:35		0		dev id: test		
▲	19:10:33	1	15	confirmed	dev id: test	payload: 01 65 00 6F 02 66 01 03 67 00 00 04 68 05 05 00 01 06 02 02 5B 07 01 00 08 67 00 C7	
<					>		
▼	19:10:34		0		dev id: test		
▲	19:10:32	0	15	confirmed	dev id: test	payload: 01 65 00 6F 02 66 01 03 67 00 00 04 68 05 05 00 01 06 02 02 58 07 01 00 08 67 00 C7	
<					>		

Serveur d'application, exemple : mydevice.com

données formatées « cayenne »



IoT applications

INTERNET OF THINGS

Enter your sub headline here



Vehicle, Asset, Person & Pet
Monitoring & Controlling



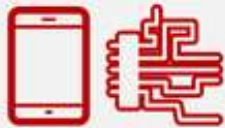
Agriculture Automation



Energy consumption



Security



Embedded Mobile



Building management



M2M & wireless
Sensor network



Everyday things



Smart homes & cities



Everyday things get connected
For smarter tomorrow



Telemedicine & Healthcare