



Premiers pas

Communication pour l'IO

Rolf Ziegler, Microclub, Octobre 2018

Agenda

- Définition
- Modules entrée de gamme, comparaison
- Mode de fonctionnement, spreading factor
- Types de transmission
 - Point-Point
 - LoRa en Réseau, LoRaWan, TTN
- Matériel
- Démonstration
- Q&R

LoRa Définition

- Low-Power Wide Area Network (LPWA)
- Protocol permettant de connecter les objets avec le réseau IP
- Objets (things) sans fils alimentés par batterie (accus) connectés à Internet au réseau régional, national et global
- Communication bidirectionnelle
- Messages sécurisés de bout-en-bout
- **LORA est LA méthode pour connecter l'IoT (internet des objets)**

Marché LoRa

- Utilisation généralisée
 - Surveillance
 - Contrôle de processus industriels
 - Contrôle de machines
 - Dans des appareils chers et bon marchés
- Augmentation de l'efficacité
 - Parcomètres, Eclairage publique,....
 - Domotique, bâtiments publics,.....

LoRa

Modules HopeRF/XBee



RFM22b/RFM23
433/868/915MHz

- **Sensitivity -121db ~0.2uV**
- Low-Power 18.5mA
- Vitesse -> 256kbps
- 1-64 Bytes
- Dist. 100->500m
- Puissance 20dB
- Chip Si443x
- Prix ~chf 10.-



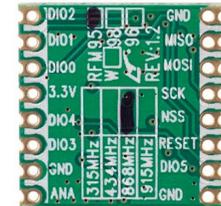
Zigbee
2.4GHz

- Sensivity -96dB
- Power 35mA
- Sommeil 10mA
- Up to 2MBps
- 1-32 Bytes
- Dist. 10->20m
- Puissance 20dB (70mA)
- Prix ~chf 30.- ->chf60.-



LoRa / RFM95
433/868/915MHz

- **Sensitivity -148db ~ 0.01uV**
- Low-Power 9.9mA/200nA
- Sommeil 2mA
- Vitesse ->300kbps
- Puissance ->20dB
- 1-256 Bytes/packet
- Dist. 100m-15km
- Chip: sx1276
- Prix ~chf 5.-

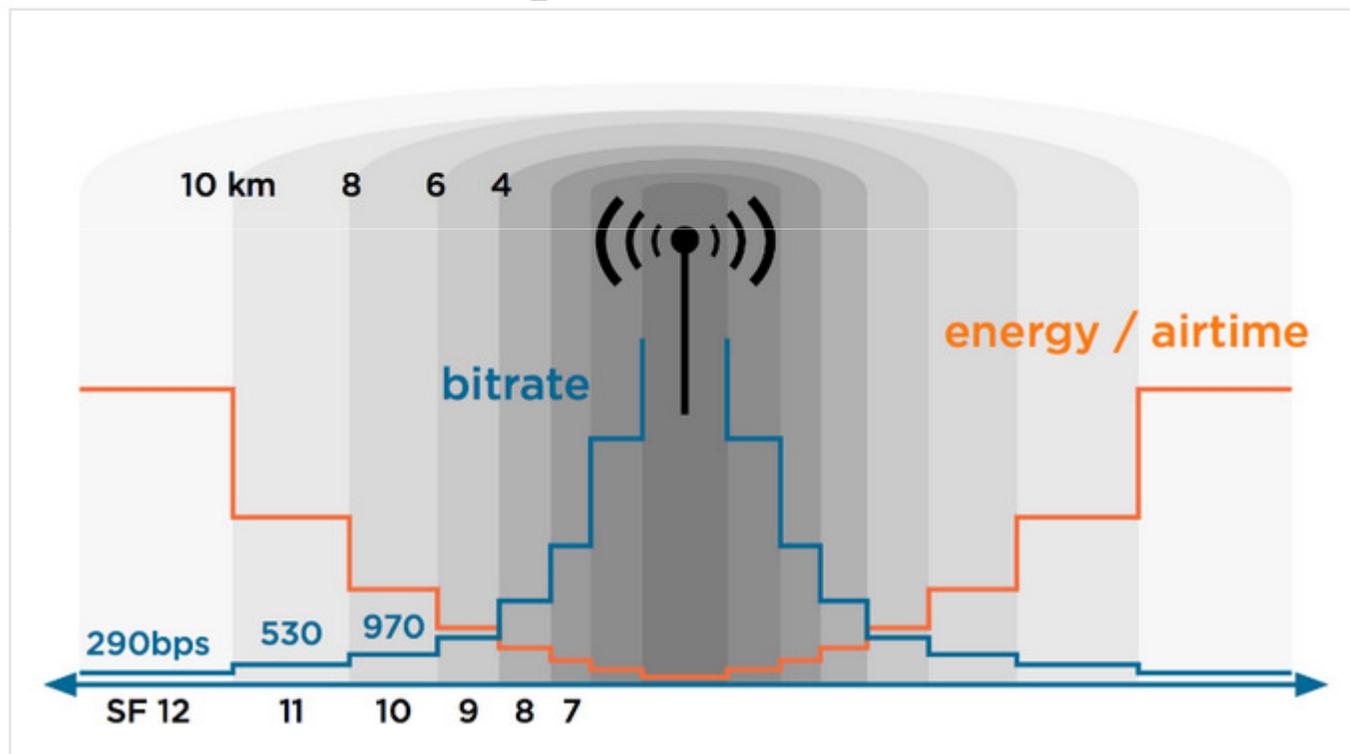


! Les circuits sont fixe au niveau fréquence !
+ 433MHz AP, 868 EU, 915 USA

LoRa

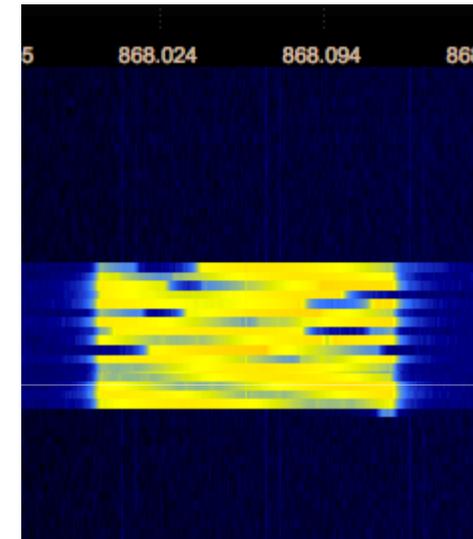
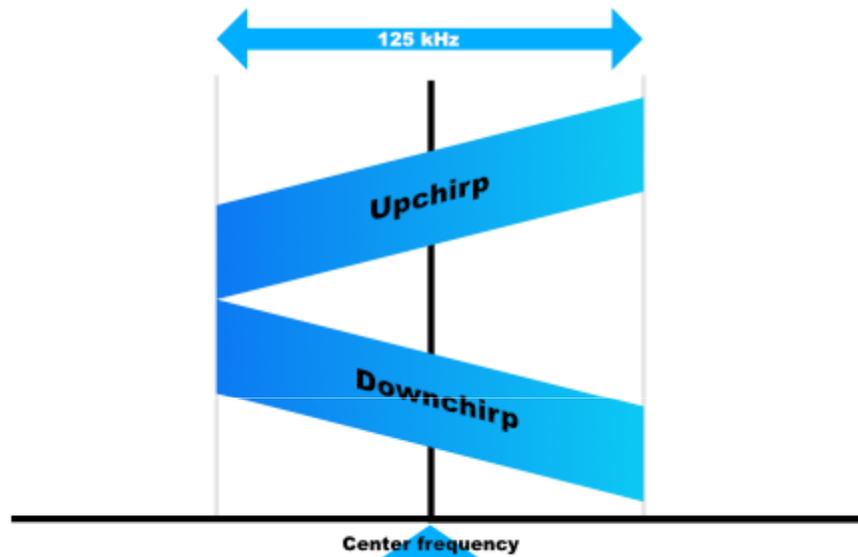
Mode de fonctionnement

- LoRa utilise le protocole « Chirp » utilisé par les sonars et radars durant la 2nd ère mondiale



- Avec une bande passante de 125,250 ou 500kHz, LoRa utilise la bande passante complète

LoRa Chirp Mode



Analyse SDR

- Chirp = variation de la fréquence dans la bande passante du canal, montante ou descendante.
- Le Spreading factor est la durée du « Chirp »
- Le Spreading factor + la bande passante déterminent la vitesse de transmission et la quantité de données transmises par paquet.

Chirp youtube

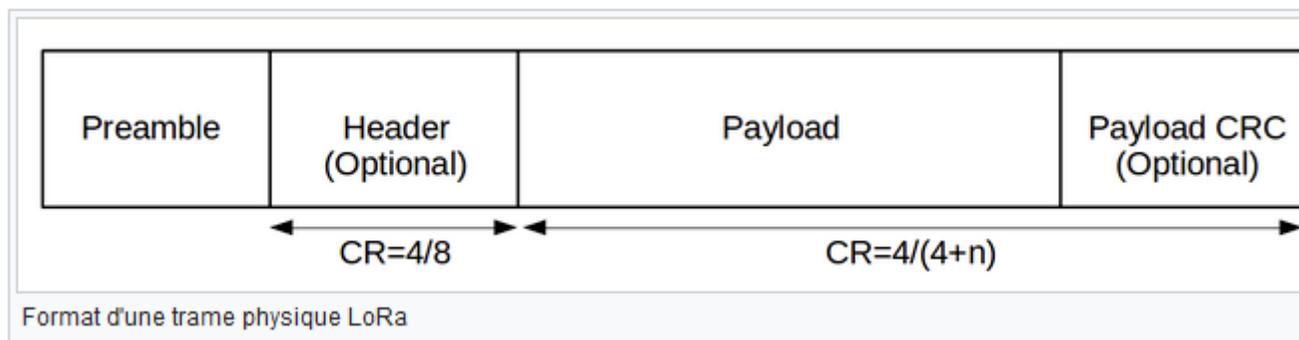
<https://www.youtube.com/watch?v=dxYY097QNs0>

LoRa

Vitesse ou distance

Data Rate (DR)	Modulation	Spreading Factor (SF)	Bande Passante	Débit Physique (bit/s)
0	LoRa	SF12	125 kHz	250
1	LoRa	SF11	125 kHz	440
2	LoRa	SF10	125 kHz	980
3	LoRa	SF9	125 kHz	1 760
4	LoRa	SF8	125 kHz	3 125
5	LoRa	SF7	125 kHz	5 470
6	LoRa	SF7	250 kHz	11 000
7	FSCK	50kbit/s		50 000
8	Réservé pour utilisation future			

↑ distance

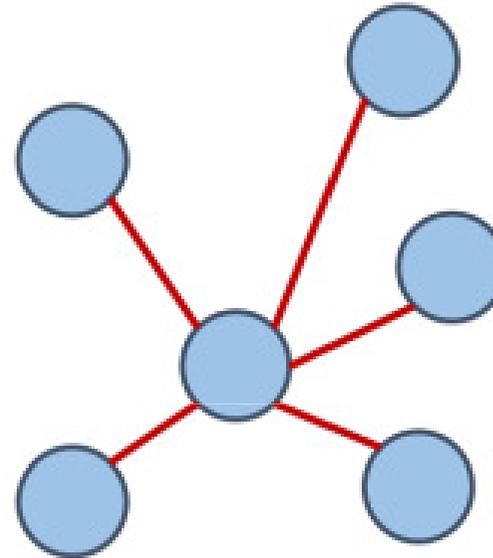


LoRa

Type de transmission



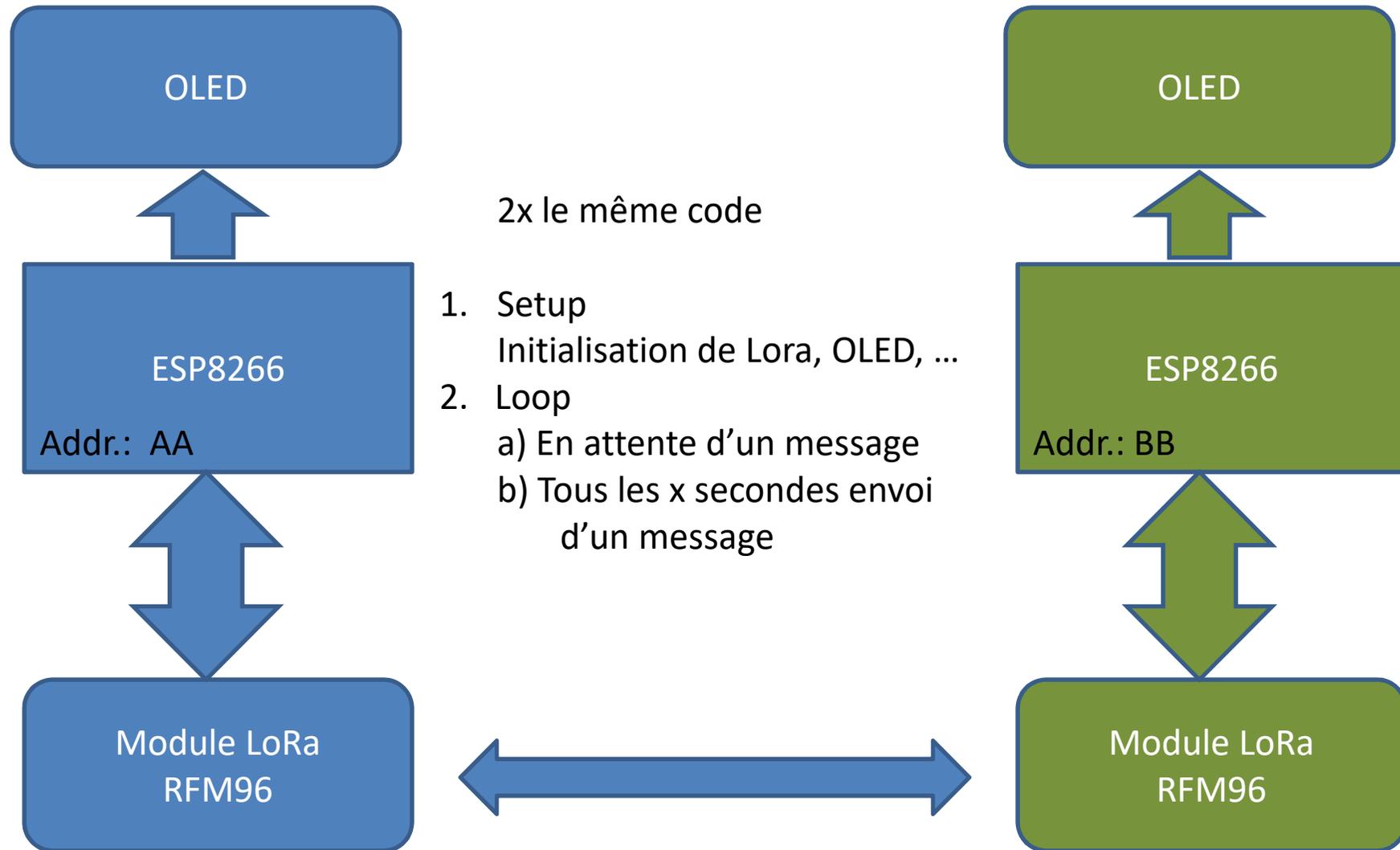
Point to point
communication



Network

LoRa Solution point-point

Test distance



Materiel LoRa Point-Point

- Wemos ESP8266, banggood.com,...
- RFM95/RFM96, ebay.com
- PCB <https://github.com/hallard/WeMos-Lora>
- Code:
https://github.com/fcgdam/TTGO_LoRa32
- Source + Librairies C, compatible Arduino

Firmware LoRa

Point-Point, émission

- Initialisation du module LoRa (SPI)
- Programmation de la fréquence (868MhZ)
- En boucle
 - Décodage et affichage des messages reçus
 - Envoi d'un message

```
void sendMessage(String outgoing) {  
  
    pixels.show();  
    LoRa.beginPacket(); // start packet  
    LoRa.write(destination); // add destination address  
    LoRa.write(localAddress); // add sender address  
    LoRa.write(msgCount); // add message ID  
    LoRa.write(outgoing.length()); // add payload length  
    LoRa.print(outgoing); // add payload  
    LoRa.endPacket(); // finish packet and send it  
    msgCount++; // increment message ID  
    // pixels.setPixelColor(1, pixels.Color(0,0,0)); // Moderately brig  
}
```

Firmware LoRa

Point-Point, réception

```
void onReceive(int packetSize) {  
if (packetSize == 0) return; // if there's no packet, return  
// read packet header bytes:  
int recipient = LoRa.read(); // recipient address  
byte sender = LoRa.read(); // sender address  
byte incomingMsgId = LoRa.read(); // incoming msg ID  
byte incomingLength = LoRa.read(); // incoming msg length  
  
String incoming = "";  
  
while (LoRa.available()) {  
incoming += (char)LoRa.read();  
}  
}
```

Démo point-point



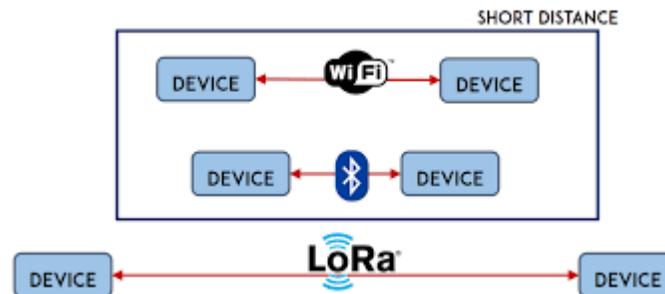
#AA Envoie par pression sur bouton



#BB renvoie automatiquement

LoRa RFM96

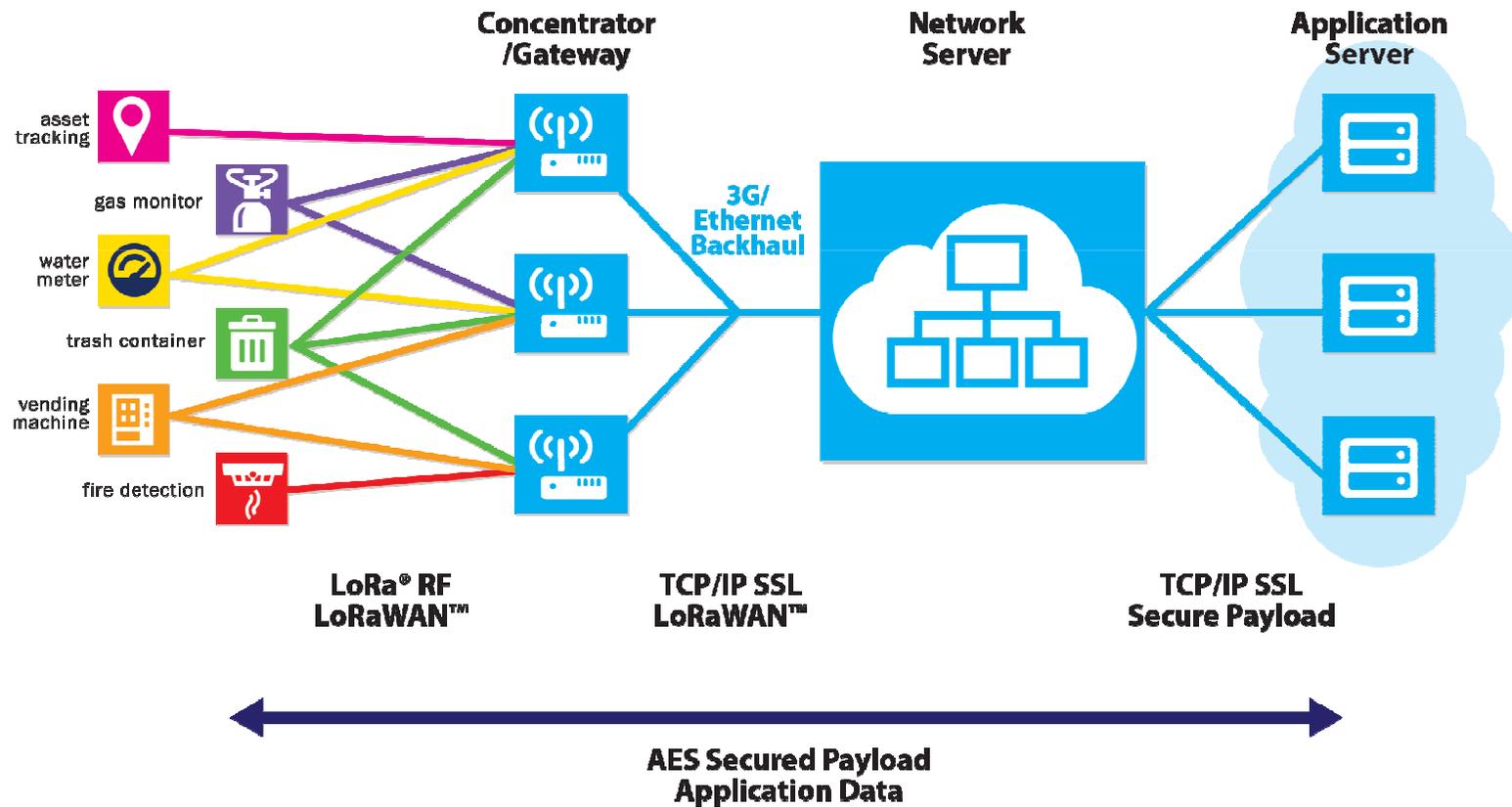
Résultats des tests point-point



- Tests effectués jusqu'à env. 500m avec 100% de transmission
- Point-Point seulement à vue !!
- Coupure quasi instantanée sans vue directe même à moins de 500m

Demo Lora Point-Point

LoRa en réseau



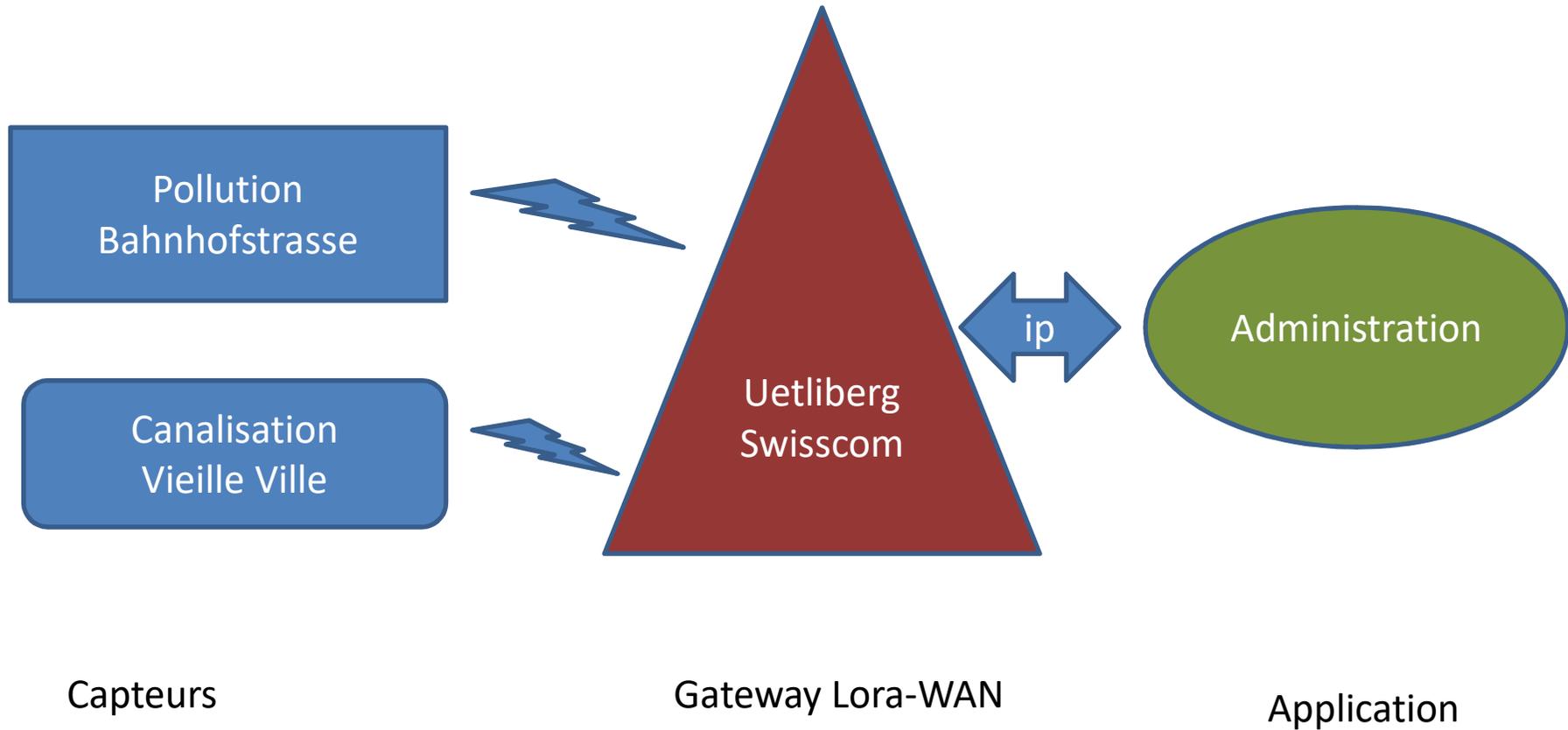
Application Lora

- Privé
 - Capteurs à distance (Cave, hangar, garage)
 - Mesure de température, humidité,...
- Ville de Zurich
 - Mesure de pollution
 - Mesure de niveau dans les canalisations
 - Identification de places de parc libres

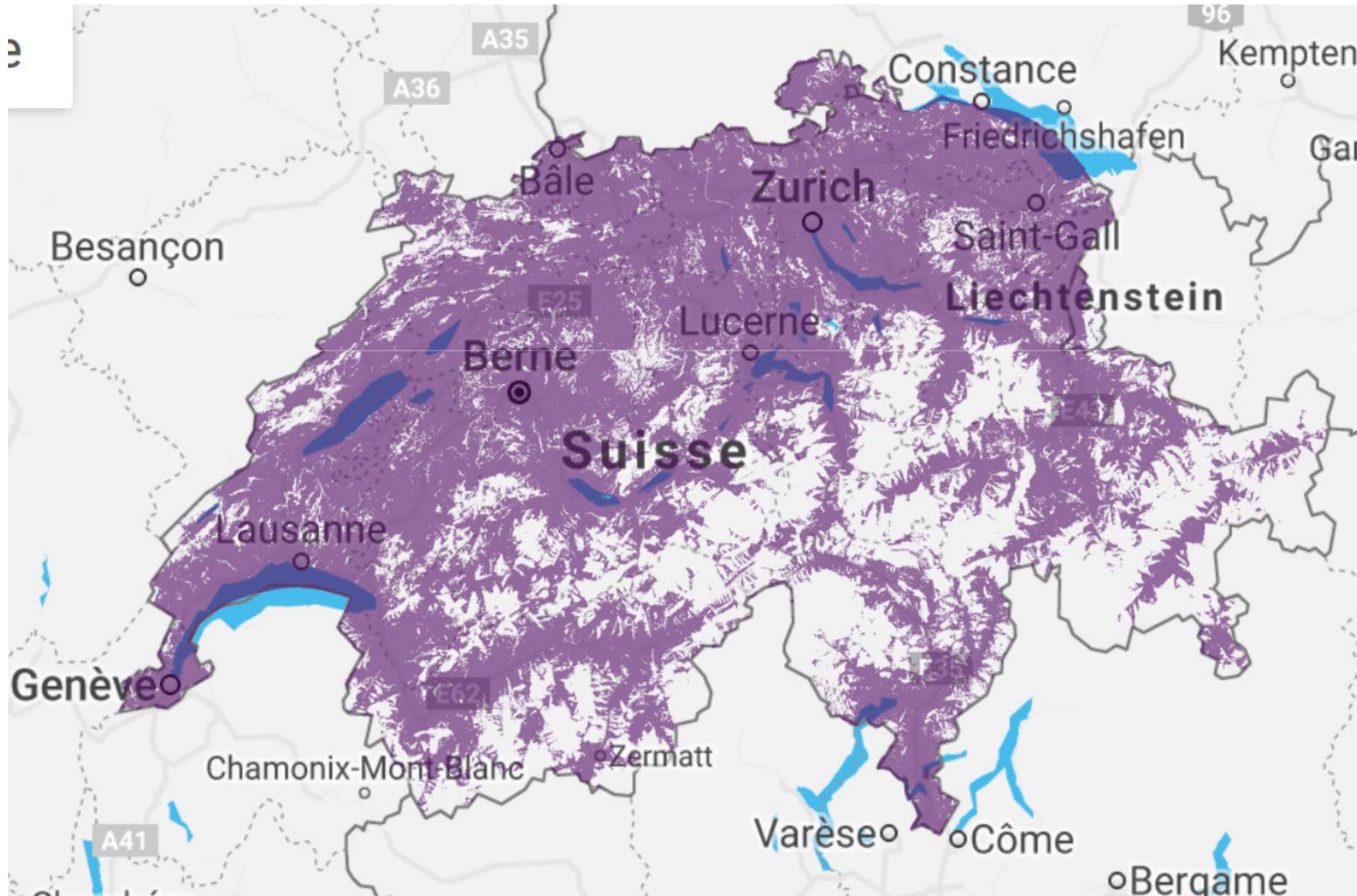
Swisscom, ville de Zurich

<https://www.srf.ch/sendungen/einstein/vom-internet-der-dinge-die-neue-macht-der-sensoren>

Ville de Zurich



Réseau LPN (LoRa) Swisscom (payant)



<http://lpn.swisscom.ch/f/notre-offre/>

LoraWan

TTN

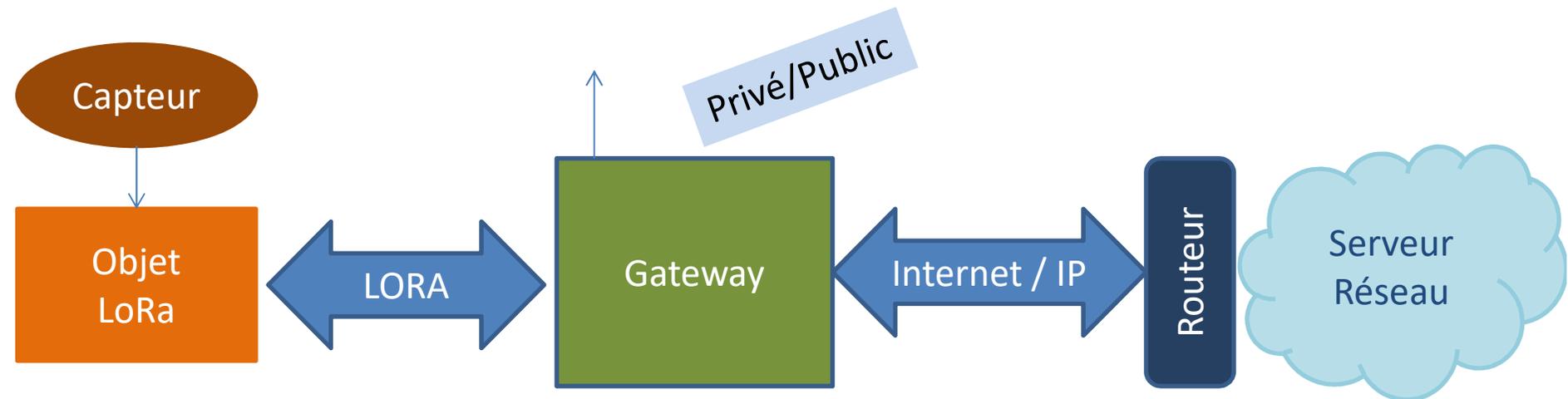


Réseau TTN

- Infrastructure mise à disposition par des privés (collaboratif)
- Server Cloud TTN
- Enregistrement et administration par l'utilisateur
- Stack/Firmware open-source
- Disponible pour ESPxx, RPI,...
- Gratuit

Principe de base de LoRaWan

Envoi de messages



- Emetteur LoRa
- Encryptage du message
- Module ESP + RFM95

- Passerelle entre les signaux LORA et Internet.
- Redirection des messages vers un serveur Cloud
- Récupération des messages par une application par réseau internet
- Module ESP +RFM95 (Limité 100m)
- Carte performante + RPI (->20km)

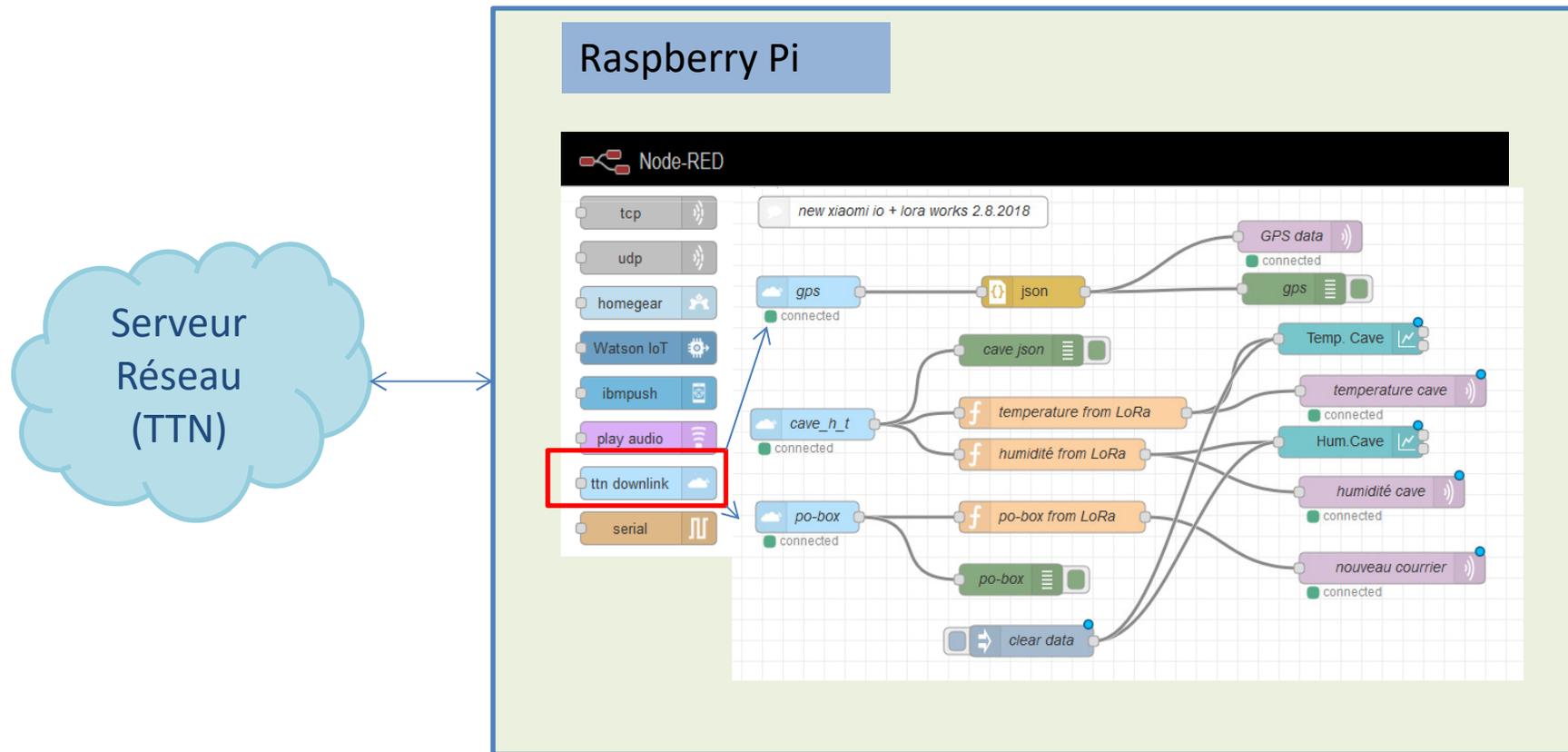
- Stockage ou redirection des messages

- Divers fournisseurs
- >Swisscom payant
 - >TTN gratuit

LoraWan

Récupération des données

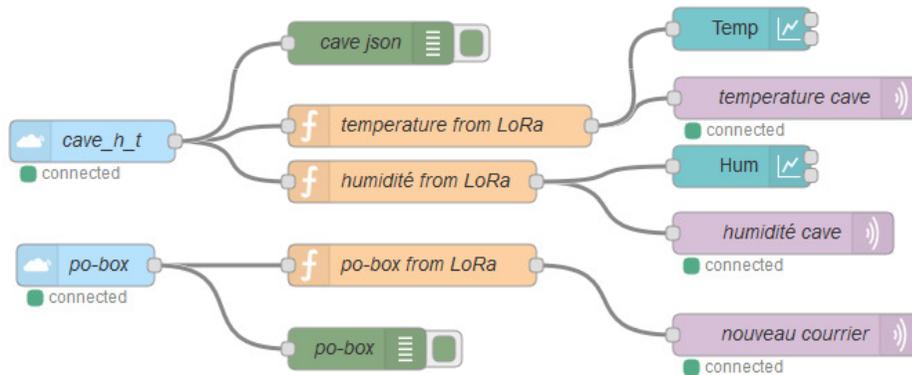
- Exemple TTN



Node TTN existant, décryptage du message

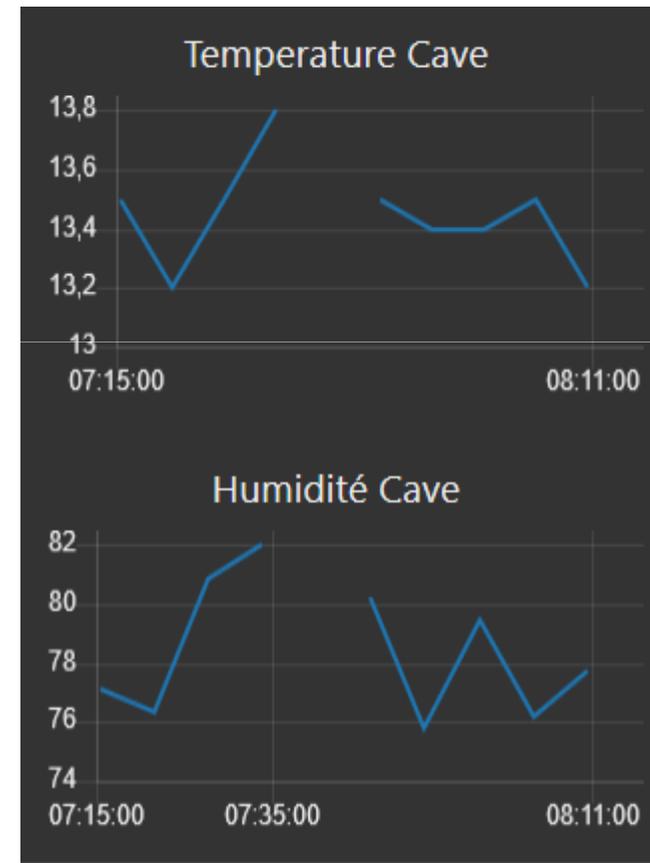
Lora

Node-Red avec graphique

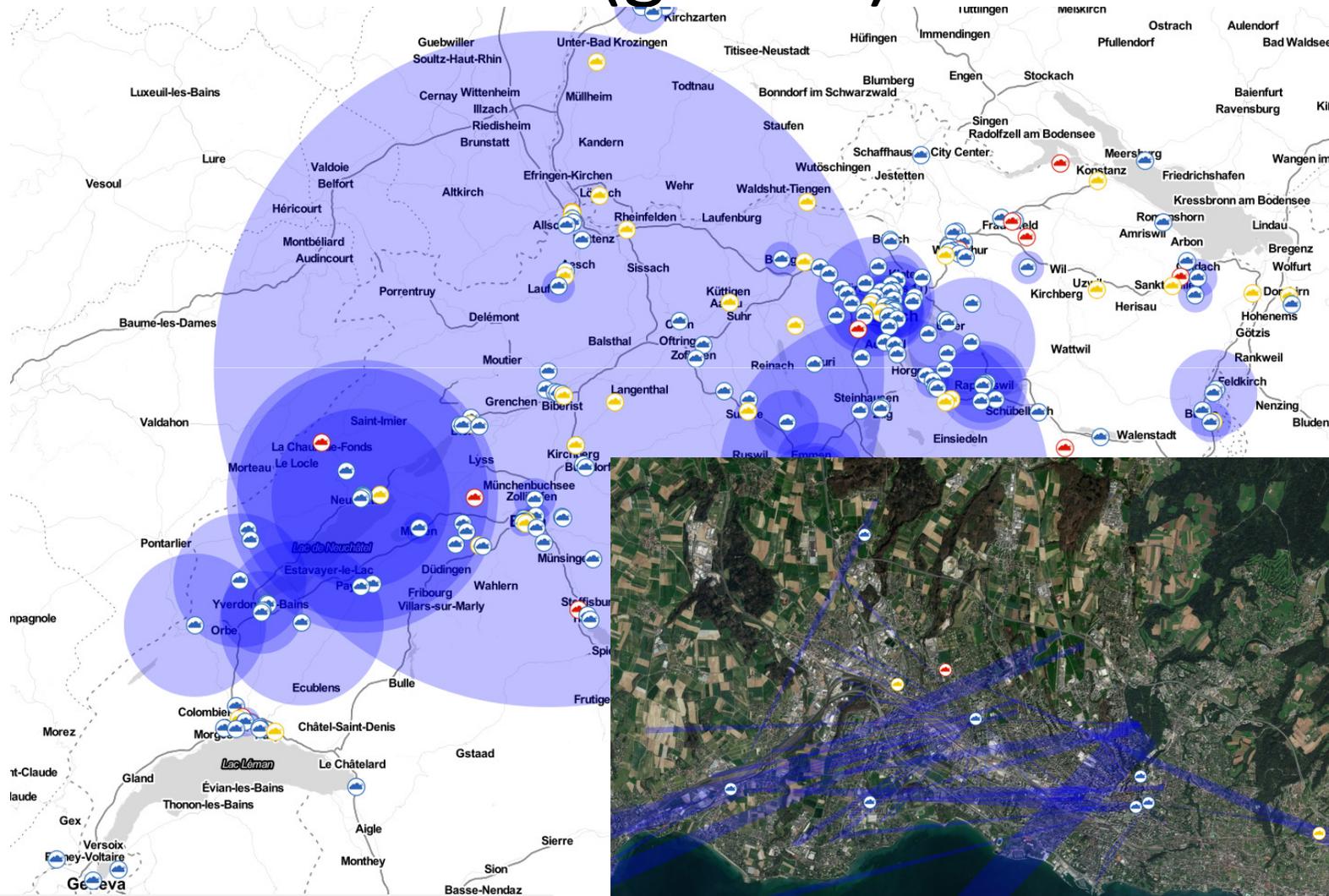


Exemple dans ma domotique:

- Récupération des données cave à vin (2dalles de béton à traverser dans PPE)
- Décodage et envoi vers le serveur MQTT et vers l'interface graphique de Node-Red



Réseau LoraWan TTN (gratuit)



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

LoRa TTN

Mon premier GW

- www.thethingsnetwork.org
- Créer un compte (gratuit)
- Dans la Console
 - Créer un Gateway
 - Créer une application (Chacune reçoit une adresse)
 - Choisir le mode de fonctionnement (ABP/OTAA)
 - Créer un ou des devices (chacune reçoit une adresse)
- En plus de l'adresse chaque élément reçoit un nom lisible

LoRa TTN

Marche à suivre

- Créer un compte TTN



CREATE AN ACCOUNT

Create an account for The Things Network and start exploring the world of Internet of Things with us.

USERNAME

This will be your username — pick a good one because you will not be able to change it.



EMAIL ADDRESS

You will receive a confirmation email, as well as occasional account related emails. If this email address is managed by a third party (such as for corporate email addresses), this third party might block emails coming from The Things Network. This email address is not public.



PASSWORD

Use at least 6 characters.



Create account

LoRa

Enregistrement du Gateway

REGISTER GATEWAY

Gateway EUI

The EUI of the gateway as read from the LoRa module

B8 27 EB FF FE 70 76 49

8 bytes

A lire sur la console de votre module LORA
(ligne Série, Teraterm,...)

I'm using the legacy packet forwarder

Select this if you are using the legacy [Semtech packet forwarder](#).

Description

A human-readable description of the gateway

Microclub Demo

```
MAC: d8:a0:1d:63:59:40, len=17
WlanConnect:: Init para 0
0:1. WiFi connect SSID=, pass=
WlanCStatus:: CONNECTED to BELLAVISTA2
Host esp32-635940 WiFi Connected to BELLAVISTA2 on IP=192.168.1.114
Local UDP port=1700
Connection successful
Gateway ID: D8A01DFFFF635940, Listening at SF8 on 868.10 Mhz.
Time: Thursday 11:24:18
Gateway configuration saved
WWW Server started on port 80
OLED_ADDR=0x3C
```

Frequency Plan

The [frequency plan](#) this gateway will use

Europe 868MHz

Router

The router this gateway will connect to. To reduce latency, pick a router that is in a region which is close to the location of the gateway.

ttn-router-eu

LoRa TTN

Marche à suivre

Adresse physique
De l'interface LAN/WLAN

- Créer un gateway

2

INFORMATION

Brand Single-channel DIY gateway

Model ESP8266-RFM95

Antenna ⓘ

LOCATION

3

Antenna Placement indoor

Altitude 590m



1

Gateway ID eui-b4e62dffff1426b9

Description Les-Brules

Owner z-control [Transfer ownership](#)

Status ● connected

Frequency Plan Europe 868MHz

Router ttn-router-eu



Gateway Key

Last Seen 11 minutes ago

Received Messages 14575

Transmitted Messages 892

LoRa TTN

Marche à suivre

- Enregistrer une application et un objet/device

APPLICATION OVERVIEW

Application ID **zcontrol-gps**

Description Lora GPS Test

Created 4 months ago

Handler **ttn-handler-eu** (current handler)

APPLICATION EUIS

<> ⇄ 70 B3 D5 7E D0 00 FE E1 📄

DEVICES

 2 registered devices

DEVICE OVERVIEW

Application ID **zcontrol-gps**

Device ID **lora_gps1**

Activation Method **ABP**

Device EUI <> ⇄ 00 05 D2 45 6E 5C 16 8E 📄

Application EUI <> ⇄ 70 B3 D5 7E D0 00 FE E1 📄

Device Address <> ⇄ 26 01 1D BD 📄

Network Session Key <> ⇄ 👁️ 📄

App Session Key <> ⇄ 👁️ 📄

Status ● 2 minutes ago

Frames up 42 [reset frame counters](#)

Frames down 1065

Rem.: une application peut avoir plusieurs appareils !

LoRa TTN

Décodage en ligne

Applications >  128854128854 > Payload Formats

PAYLOAD FORMATS

Payload Format

The payload format sent by your devices

Custom

decoder

converter

validator

encoder

```
1 function Decoder(bytes, port) {
2   // Decode plain text; for testing only
3
4   var mod= bytes[0];
5   var temp = bytes[1]<<8 | bytes[2];
6   var hum = bytes[3]<<8 | bytes[4];
7   var swi = bytes[5];
8   var bat = bytes[6];
9
10  return {
11
12    mode: mod, temperature: temp/10, humidity: hum/10, letter: swi, battery:bat
```

LoRa TTN

Test en temps réel

Applications >  128854128854 > Devices >  cave_h_t > Data

▲ 11:06:02	651	1		payload: 02 00 89 03 12 00 00	battery: 0	humidity: 78.6	letter: 0	mode: 2	temperature: 13.7
▲ 10:59:54	648	1		payload: 02 00 88 03 2D 00 00	battery: 0	humidity: 81.3	letter: 0	mode: 2	temperature: 13.6
▲ 10:53:46	645	1		payload: 02 00 85 03 0D 00 00	battery: 0	humidity: 78.1	letter: 0	mode: 2	temperature: 13.3
▲ 10:47:38	642	1		payload: 02 00 86 03 02 00 00	battery: 0	humidity: 77	letter: 0	mode: 2	temperature: 13.4
▲ 10:41:29	639	1		payload: 02 00 85 03 17 00 00	battery: 0	humidity: 79.1	letter: 0	mode: 2	temperature: 13.3
▲ 10:35:21	636	1		payload: 02 00 86 02 FC 00 00	battery: 0	humidity: 76.4	mode: 2	switch: 0	temperature: 13.4
▲ 10:23:04	630	1		payload: 02 00 86 02 F8 00 00	battery: 0	humidity: 76	mode: 2	switch: 0	temperature: 13.4
▲ 10:10:48	×	×	historical	payload: 02 00 85 02 F4 00 00	battery: 0	humidity: 75.6	mode: 2	switch: 0	temperature: 13.3
▲ 10:04:39	×	×	historical	payload: 02 00 89 03 14 00 00	battery: 0	humidity: 78.8	mode: 2	switch: 0	temperature: 13.7
▲ 09:58:31	×	×	historical	payload: 02 00 88 03 2B 00 00	battery: 0	humidity: 81.1	mode: 2	switch: 0	temperature: 13.6
▲ 09:52:23	×	×	historical	payload: 02 00 86 03 21 00 00	battery: 0	humidity: 80.1	mode: 2	switch: 0	temperature: 13.4

LoRa TTN

Décodage des messages

DEVICE OVERVIEW

Application ID **128854128854**

Device ID cave_h_t

Description Capteur Humidité Temperature Cave

Activation Method ABP

Device EUI <> ⇅ 00 33 2B 2F 1D 4C CC 7C 🗑

Application EUI <> ⇅ 70 B3 D5 7E D0 00 F6 76 🗑

Device Address <> ⇅ 26 01 1B 63 🗑

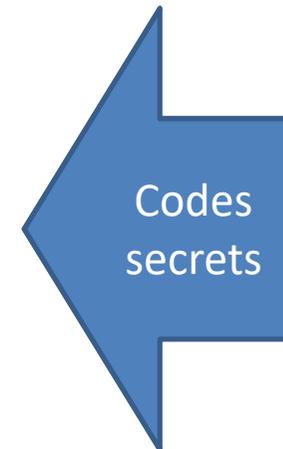
Network Session Key <> ⇅ 👁 🗑

App Session Key <> ⇅ 👁 🗑

Status ● 1 minute ago

Frames up 39313 [reset frame counters](#)

Frames down 0



LoRa TTN

Client/Capteur

- Plus facile,
- Un fichier « Arduino »
- Indiquer les code générés sur la page TTN

```
#ifdef SENSOR1
static u1_t NWKSKEY[16] = { 0x75, 0xE9, 0x6B, 0x72, 0x19, 0x66, 0x8C, 0xC0, 0x71, 0x54, 0xED, 0xE0, 0xDB, 0xBA, 0x3B, 0x42 };

static u1_t APPSKEY[16] = { 0x9F, 0xDA, 0xF3, 0x26, 0x15, 0x55, 0x46, 0xC8, 0x16, 0x01, 0x63, 0xD4, 0xA3, 0xB2, 0x3A, 0x33 };

static u4_t DEVADDR = 0x26011337; // Put here the device id in hexadecimal form.
#else
```

- Charger la variable dans le buffer
- Un timer envoie régulièrement le contenu vers le réseau TTN

```
// Payload to send (uplink)
byte buffer[7];

buffer[0]=1; // switch sensor
buffer[1]=0;
buffer[2]=0;
buffer[3]=0;
buffer[4]=0;

if(newPost) buffer[5]=1;
else buffer[5]=0;
buffer[6]=0;
```

LoRa TTN

Commentaires

- Réseau collaboratif
 - Les membres mettent à disposition leur Gateway
 - Le transit de message tiers est automatique (Crypté)
- Si un [Gateway TTN existe](#) à l'endroit que je veux connecter, je n'ai pas besoins de Gateway perso
- L'utilisation du réseau est gratuit
- Je peux enregistre un nombre infini d'objets(Things) ou de Gateway/passerelles.

LoRa

Performance

- La Performance est limitée à 200kbps
- La quantité de données est limitée (868MHz)
 - 25mW et une utilisation de 1% /heure
- Ceci correspond à 36s par heure !
- Seul des petites quantités de données sont donc possible et conseillées. Pas de signal Audio ou Vidéo !
- LoRa n'est donc pas une alternative à BT ou WiFi!

LoRa

Distance / Conformité

- La Distance de transmission dépend principalement de la sensibilité du Gateway et de son Chipset.
- Un Gateway simple est constitué du chipset SX1272/SX1276 et beaucoup moins cher qu'un gateway complet(Full)
- Le Gateway complet est souvent constitué du chipset SX1301/SX1257 et couvre simultanément 8 canaux
- Un Gateway simple n'est pas conforme à la norme LoRaWan et déconseillé pour des applications professionnelles LoRaWan!
- Le Gateway complet permet de communiquer à des « Spreading facteurs » différents et des fréquences différentes ce qui n'est pas le cas d'un GW simple !

LoRaWan

Résumé

- Facile à installer
- Très économique
- Flexible à adapter
- Évolutif (Scalabilité, adaptabilité)
- Bidirectionnel
- Sure, encrypté
- Optimisé en consommation (plusieurs années)

Demo

- Gateway: Single channel gateway
- Application: GPS, PO-Box
- Device: ESP8266+RFM95

LoRa

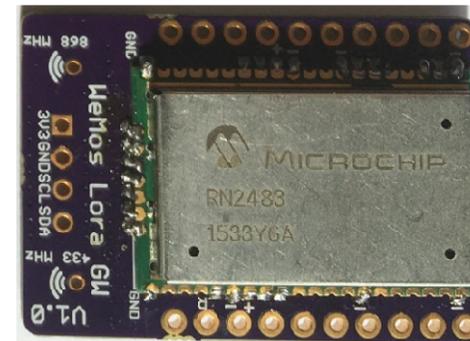
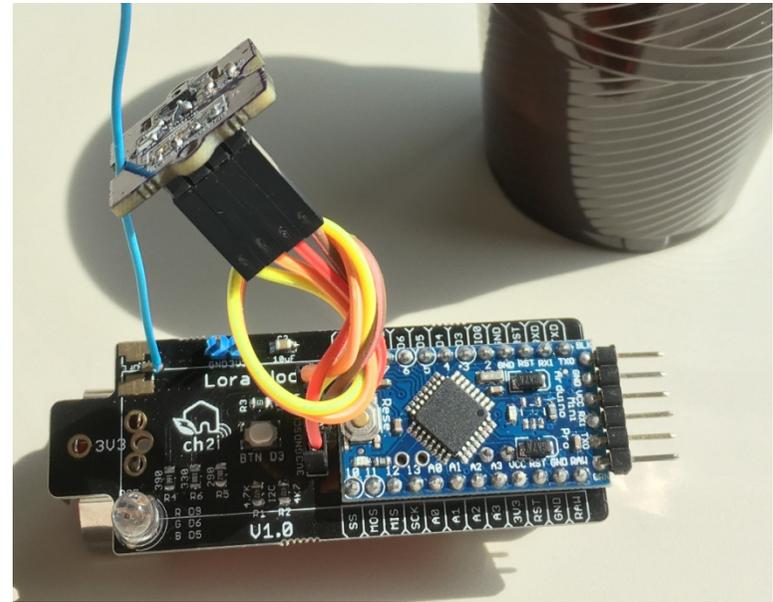
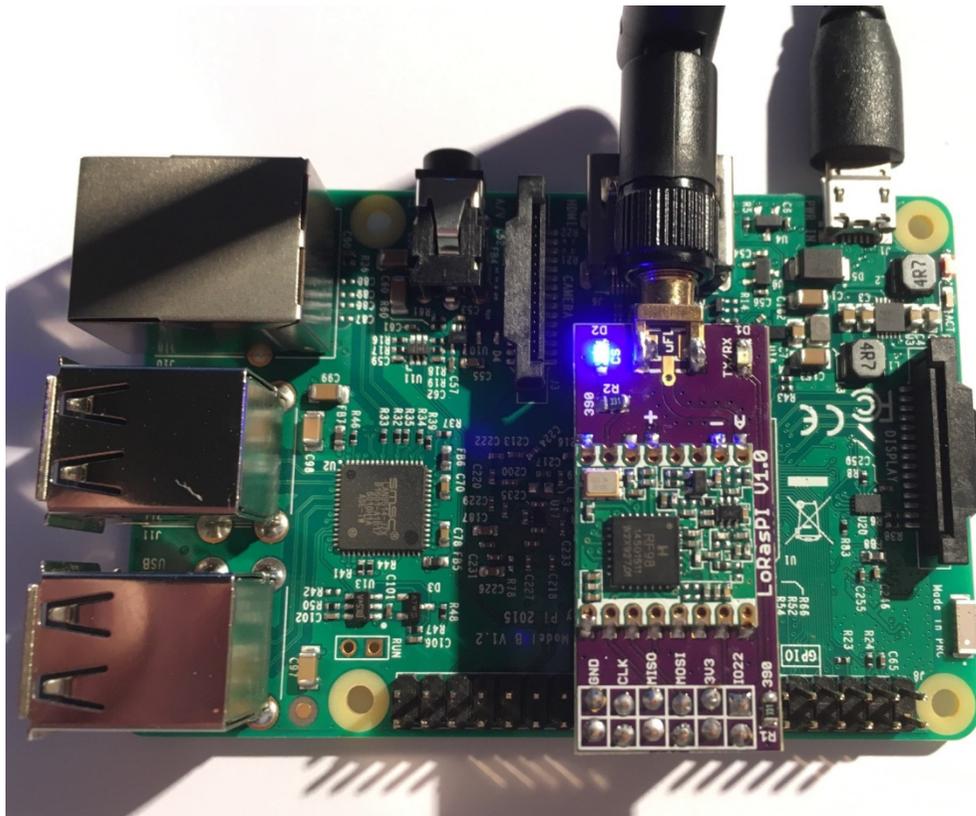
A retenir

- Petit circuits, faible consommation



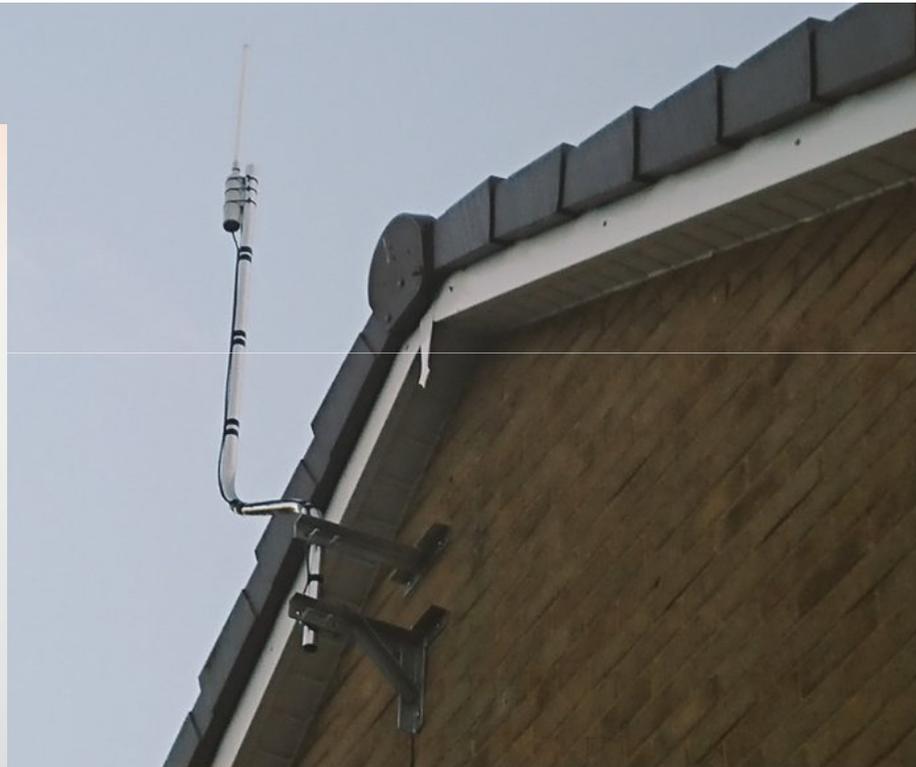
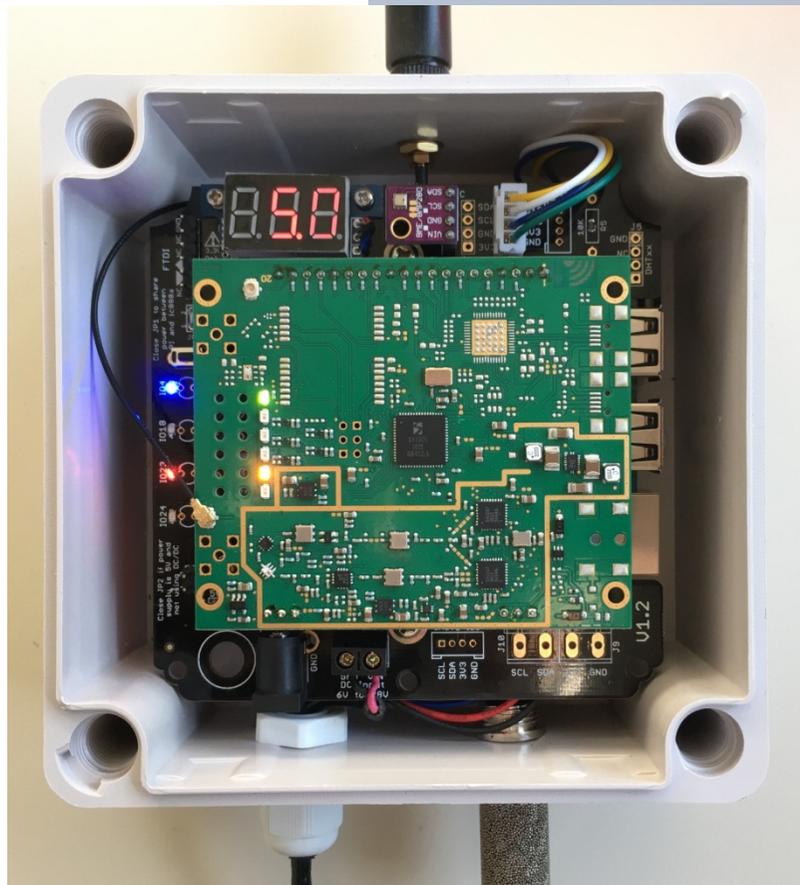
Source: lot-store.com

Multi-Plattformen



Source: [github/ch2i](https://github.com/ch2i)

Longue distance nécessite de l'équipement approprié

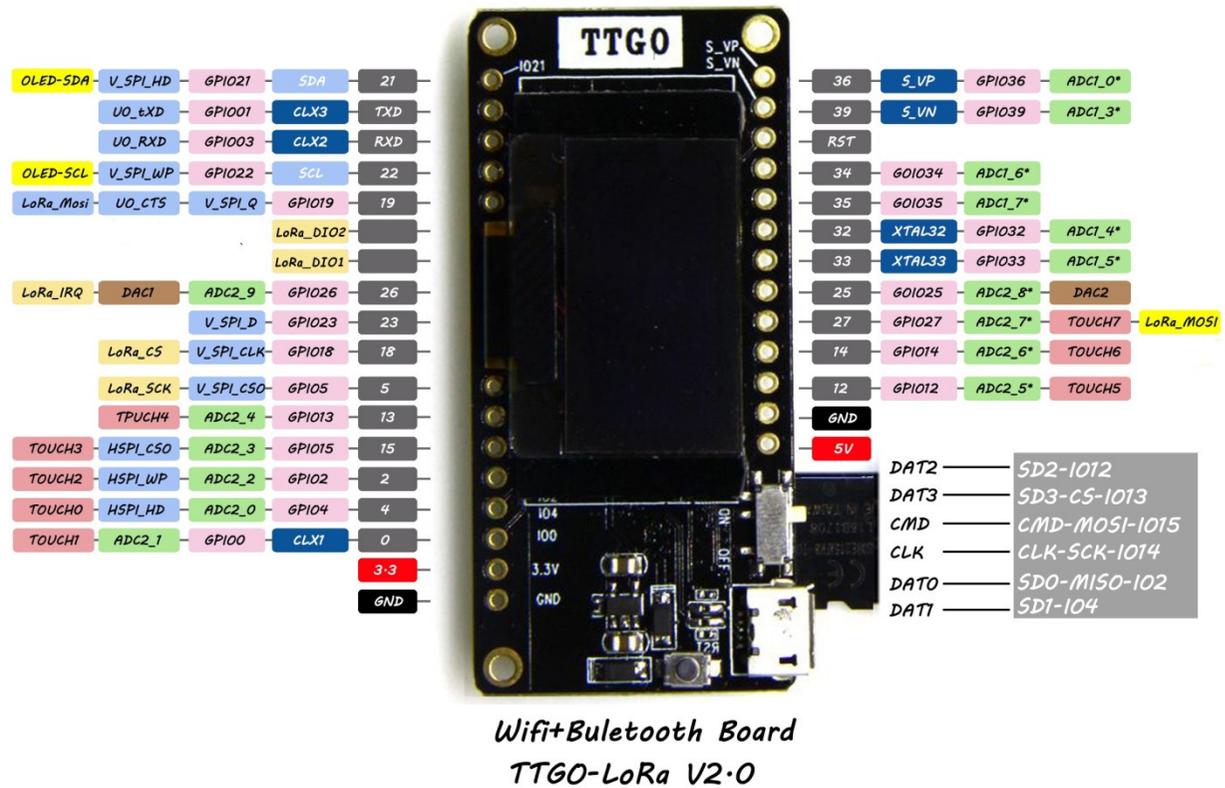


IC880a, basé Raspberry

<https://github.com/ch2i/iC880A-Raspberry-PI>

LoRaWan

- Questions ?

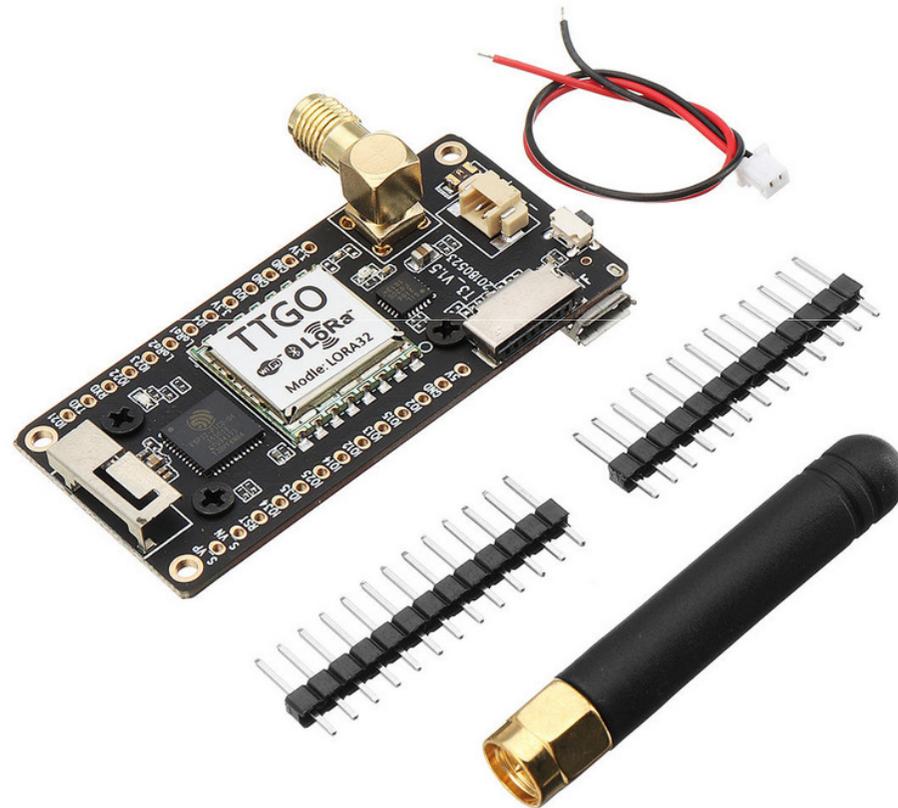


Banggood.com, pack de 2pc env. chf 30.-

!! Choisir la version 868MHz

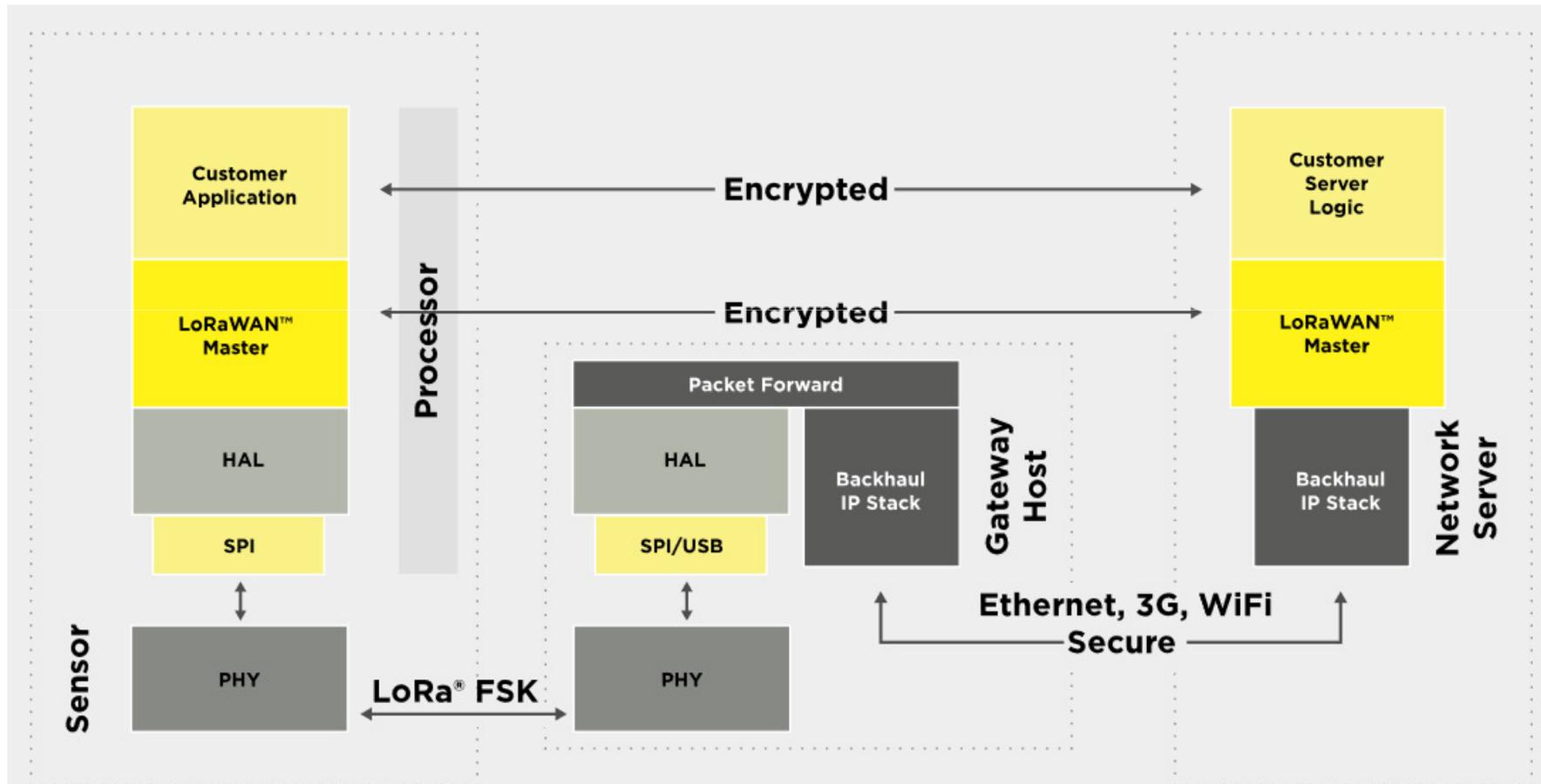
Actuellement version 2.0 du circuit (dernier chip ESP32)

Nouvelles versions touts les mois



Librairies LoRa

Structure des composants



PHY = module physique, HAL = Hardware abstraction layer (SW)