



Expert Systems and Applications Laboratory



VNiVERSiDAD  
DE SALAMANCA

CAMPUS DE EXCELENCIA INTERNACIONAL



Álvaro Lozano Murciego

PhD Student

- IoT
- Machine Learning
- Route Optimization
- Edge Computing

Android Developer

[loza@usal.es](mailto:loza@usal.es)

Twitter: \_lozanillo\_

# Contenido

1. Introducción: IoT, LPWANs y casos de uso
2. LoRa
3. LoRaWAN
4. Tipos de dispositivos
5. The Things Network (TTN)
6. Creación de un enlace P2P LoRa
7. Simulación de un nodo LoRaWAN con TTN
8. Recursos HW y bibliográficos

# 1. Introducción

IoT, LPWANs y casos de uso



VNiVERSiDAD  
DE SALAMANCA  
CAMPUS DE EXCELENCIA INTERNACIONAL

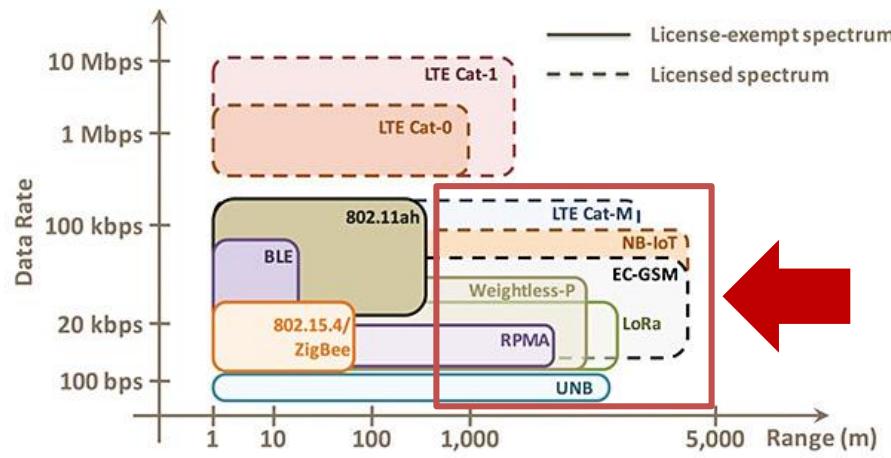
# Introducción

- ▶ Internet of Things: red de dispositivos que están conectados a Internet y son capaces de comunicarse entre sí.
- ▶ Existen múltiples tecnologías inalámbricas en el mercado para posibilitar la conexión de dispositivos a internet. De forma general:
  - Short-Range Wireless
  - Cellular
  - LPWANs



# Introducción

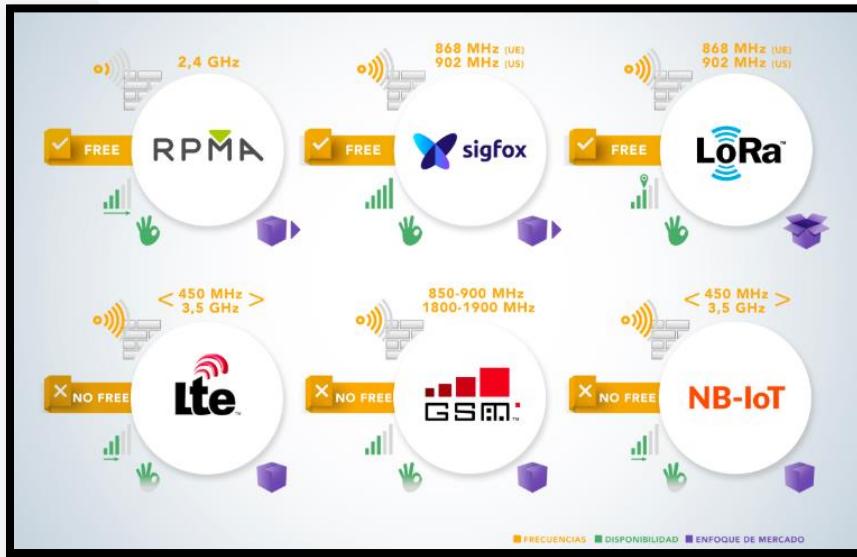
- ▶ LPWAN: (Low Power Wide Area Network) es un tipo de redes de comunicaciones inalámbricas que se han diseñado para transmitir paquetes de datos con un tamaño reducido, a grandes distancias y habitualmente en dispositivos dotados con una batería como fuente de energía.



Fuente: <http://iotbusinessnews.com/>

# Introducción

- Existen distintas tecnologías dentro de este grupo: NB-IoT, Sigfox, LoRa, etc.



|                     | SIGFOX                  | LoRa                      | clean slate                           | NB LTE-M Rel. 13                    | LTE-M Rel. 12/13                    | EC-GSM Rel. 13           | 5G (targets)             |
|---------------------|-------------------------|---------------------------|---------------------------------------|-------------------------------------|-------------------------------------|--------------------------|--------------------------|
| Range (outdoor) MCL | <13km 160 dB            | <11km 157 dB              | <15km 164 dB                          | <15km 164 dB                        | <11km 156 dB                        | <15km 164 dB             | <15km 164 dB             |
| Spectrum Bandwidth  | Unlicensed 900MHz 100Hz | Unlicensed 900MHz <500kHz | Licensed 7-900MHz 200kHz or dedicated | Licensed 7-900MHz 1.4 MHz or shared | Licensed 8-900MHz 2.4 MHz or shared | Licensed 8-900MHz shared | Licensed 7-900MHz shared |
| Data rate           | <100bps                 | <10 kbps                  | <50kbps                               | <150kbps                            | <1 Mbps                             | 10kbps                   | <1 Mbps                  |
| Battery life        | >10 years               | >10 years                 | >10 years                             | >10 years                           | >10 years                           | >10 years                | >10 years                |
| Availability        | Today                   | Today                     | 2016                                  | 2016                                | 2016                                | 2016                     | beyond 2020              |

Figure 3. LPWA IoT connectivity overview

Fuente: <https://www.thethingsnetwork.org/community/sevilla/post/por-que-lorawan>

# Introducción

| Wireless Technology | Wireless Communication | Range (m)   | Tx power (mW) |
|---------------------|------------------------|---|---------------|
| Bluetooth           | Short range            | ~10   | ~2.5          |
| WIFI                | Short range            | ~50   | ~80           |
| 3G / 4G             | Cellular               | ~5000   | ~500          |
| LoRa*               | LPWAN                  | 2000-5000 (urban area)<br>5000-15000 (rural area)<br>> 15000 (direct line of sight) | ~20           |

Fuente: [https://www.mobilefish.com/download/lora/lora\\_part1.pdf](https://www.mobilefish.com/download/lora/lora_part1.pdf)

# Introducción

- ▶ **¿Ventajas principales de las LPWANs?**
- Enlaces de comunicación a grandes distancias
- Consumo energético muy bajo en la comunicación del dispositivo
  
- ▶ **¿Restricciones?**
- Datos de un tamaño pequeño: sensores (~~Video, Imágenes~~)
- Limitación en el número de transmisiones de datos

# Introducción

## ► CASOS DE USO:

- Smart monitoring:
  - Nivel de agua en canales, ríos, etc.
  - Nivel de combustible
  - Monitorización del Ambiente: Temperatura, humedad, etc.
  - **Waste Management**
  - Smart Lighting
  - Noise monitoring
  - Radiactivity monitoring
  - ...
- Agricultura y ganadería
  - Humedad del suelo en cultivos
  - Detección de plagas
  - Control del ganado
  - ...
- Localización y seguridad
  - Control de flotas y assets
  - Control de flujo de personas
  - ...
- LPWAN + Edge computing

# 2. LoRa

Long Range



VNiVERSiDAD  
DE SALAMANCA  
CAMPUS DE EXCELENCIA INTERNACIONAL

# LoRa

- ▶ LoRa: tecnología inalámbrica desarrollada por la empresa francesa Cycleo y más tarde adquirida por la corporación Semtech.
- ▶ La tecnología de radio LoRa y su modulación están patentadas y su código es cerrado.
- ▶ Semtech otorga licencias a distintos fabricantes para que implementen su tecnología: HopeRF, Microchop, Dorji, etc.
- ▶ La palabra LoRa es una marca registrada por Semtech en 2015.



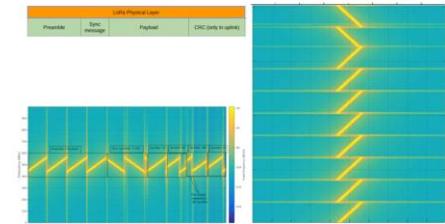
# LoRa

- ▶ Características principales de LoRa:
- Alta tolerancia a las interferencias
- Alta sensibilidad para recibir datos
- Basado en modulación CHIRP
- Bajo Consumo
- Largo alcance
- Baja transferencia de datos
- Conexión punto a punto
- Frecuencias de trabajo: 915Mhz América, 868 Europa, 433 Asia

Chirp  
Spread  
Spectrum

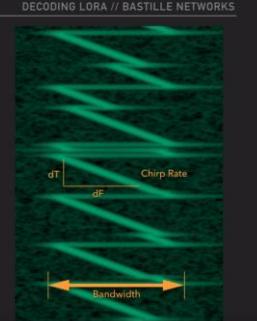
Part 1

Compressed  
High  
Intensity  
Radar  
Pulse



## SOME DEFINITIONS...

- ▶ **Bandwidth:** width of spectrum occupied by chirp
- ▶ **Spreading factor:** number of bits encoded per symbol (RF state, remember?)
- ▶ **Chirp rate:** first derivative of chirp frequency



# LoRa

► **Rango de comunicación:**

| Environment                  | Range (km) |
|------------------------------|------------|
| Urban areas (towns & cities) | 2-5        |
| Rural areas (countrysides)   | 5-15       |
| Direct Line Of Sight         | >15        |

Fuente: [https://www.mobilefish.com/download/lora/lora\\_part1.pdf](https://www.mobilefish.com/download/lora/lora_part1.pdf)

# LoRa

- ▶ Rango de comunicación:

**Theoretical maximum in free space**

2.4 GHz, with 95.5 dBm link budget:

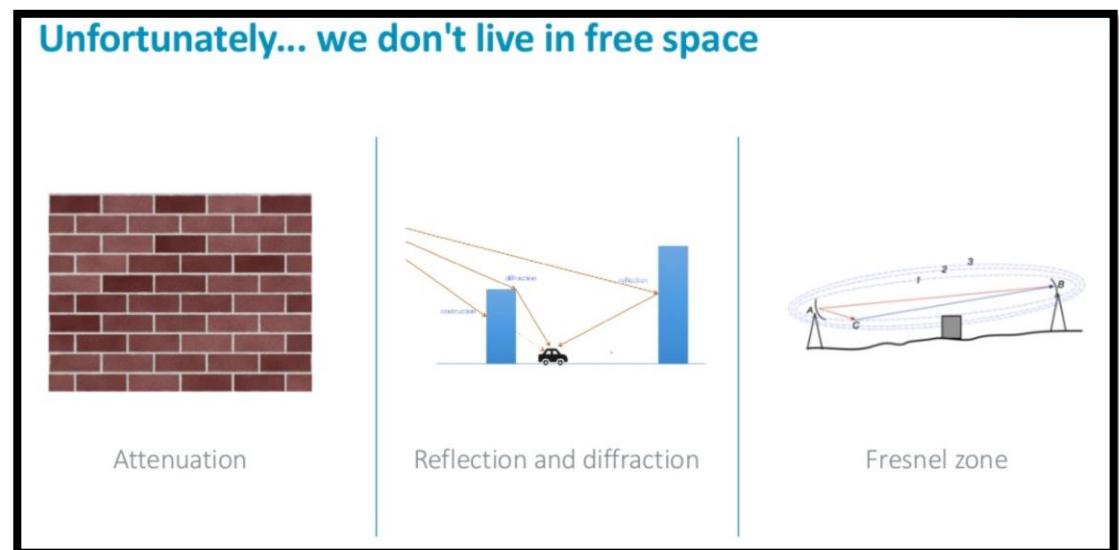
***550 meters***

915 MHz, with 151 dBm link budget:

***850,000 meters***

# LoRa

- ▶ Rango de comunicación:



Fuente: <https://es.slideshare.net/janjongboom/iot-deep-dive-lorawan>

# LoRa

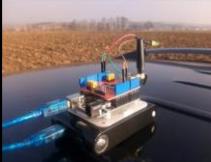
- ▶ Algunos récords:



- ▶ [Andreas Spiess](#)

212 KM

**World Record Attempt:**



**LoRa™**

**Longest Distance**

A graphic featuring a small LoRa transceiver module mounted on a car dashboard, with a signal icon above it. The text "World Record Attempt:" is at the top, followed by "LoRa™" with a signal icon, and "Longest Distance" at the bottom.

- ▶ [Koppelting event Holanda](#)

Globo de helio a una altitud  
de 38.772 km

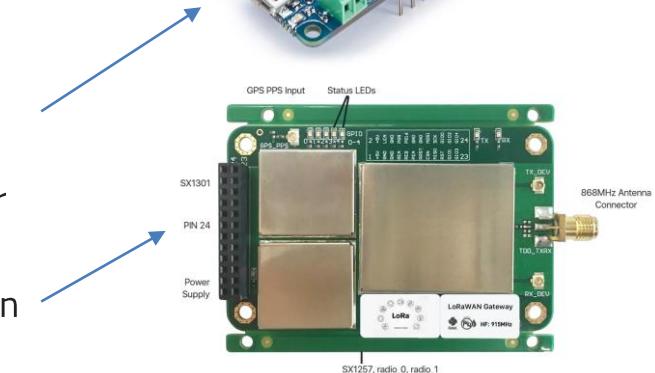
702 KM



# LoRa

- ▶ **LoRa:** acrónimo de **Long Range** y es una tecnología inalámbrica en la que un emisor envía pequeños paquetes de información (0.3kbps a 5.5 kbps) a un receptor ubicado a una gran distancia. Es posible encontrar los siguientes dispositivos:
- **End node:** formado por un microcontrolador y un módulo de radio LoRa. Pueden estar dotados de sensores o actuadores y suelen estar alimentados por una batería.
- **Gateway:** también conocido como concentrador, es un dispositivo que puede recibir transmisiones de múltiples end nodes al mismo tiempo. Los chip usados en gateways y end nodes **son distintos**.

Arduino MKR  
1300 LoRa



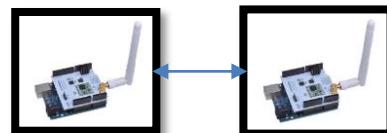
Gateway chip SX1301- RAK831

# LoRa: tipos de comunicación

## ► TIPOS DE COMUNICACIÓN CON LA CAPA FÍSICA LORA

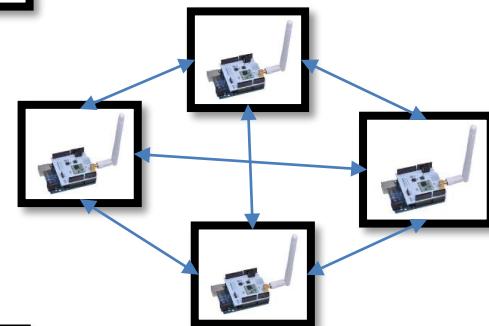
- End-node <> End-node (P2P)

(Veremos un ejemplo más tarde)



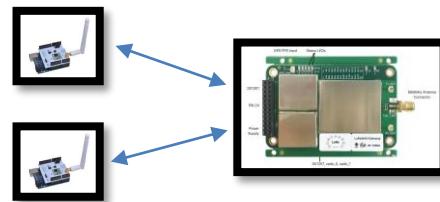
- End-node <> End-node <> End-node (Mesh)

Ejemplo de implementación con RadioHead



- End-node <> Gateway (Star)

Topología seguida por LoRaWAN



# LoRa: frecuencias utilizadas

- ▶ Frecuencias empleadas por región:

Habitualmente se engloban en 3 grupos:

- ▶ Europa 868Mhz and 433 Mhz,
- ▶ US: 915Mhz and
- ▶ AS: 430 Mhz.

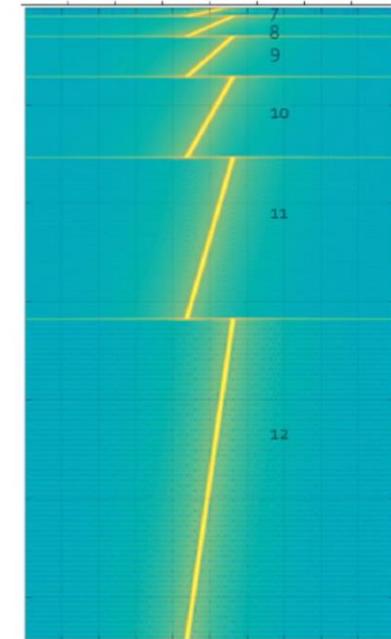
- AS\_923 : Asia 923 MHz
- AU\_915\_928 : Australia 915 - 928 MHz
- CN\_470\_510 : China 470 - 510 MHz
- CN\_779\_787 : China 779 - 787 MHz
- EU\_433 : Europe 433 MHz
- EU\_863\_870 : Europe 863 - 870 MHz
- IN\_865\_870 : India 865 - 867 MHz
- KR\_920\_923 : Korea 920 - 923 MHz
- RU\_864\_870 : Russia 864 - 870 MHz
- US\_902\_928 : United States 902 - 928 MHz

Dentro de las bandas anteriores, se definen canales de frecuencias concretas. Es complicado recordar las frecuencias de todos los canales, pero normalmente las bibliotecas que utilizaremos ya se encargan de facilitarnos esta tarea.

Como resumen para la banda de 868 MHz hay 10 canales y para la de 915 MHz hasta 64

# LoRa: Spread Factor

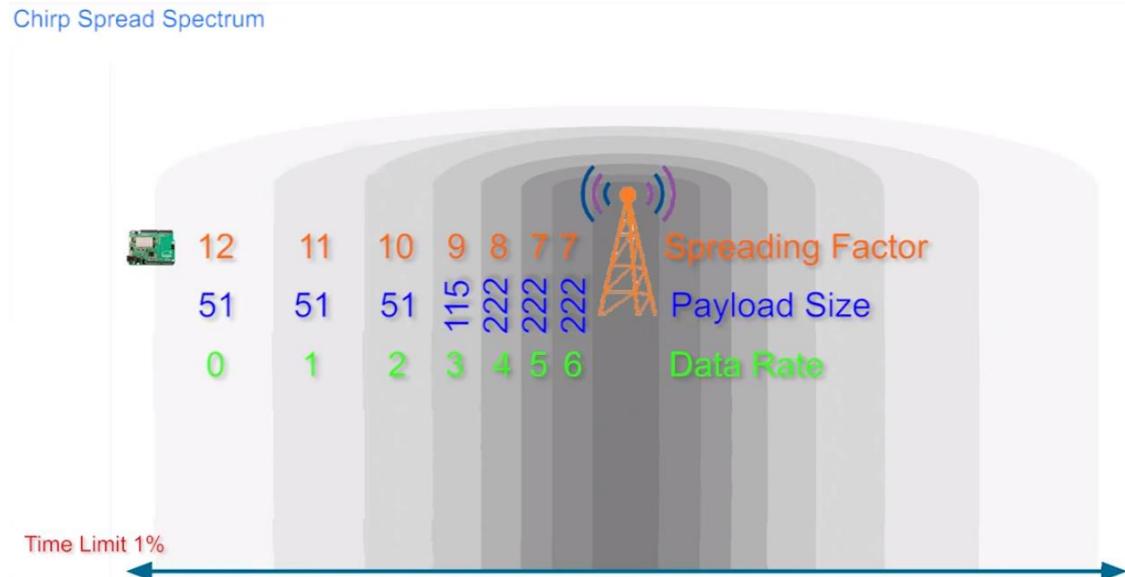
- ▶ **Spread Spectrum:** Esta técnica diseñada originalmente para aplicaciones militares donde la información (por ejemplo un bit) es esparcida a lo largo de una mayor. Realizando este proceso el ratio señal ruido es muy pequeño y la señal (el bit enviado) es mucho más resistente al ruido señales de interferencias.
- Parámetro: **Spread factor SF** → 7 al 12



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

# LoRa: SF vs payload

- ▶ Implicaciones del SF en la comunicación.

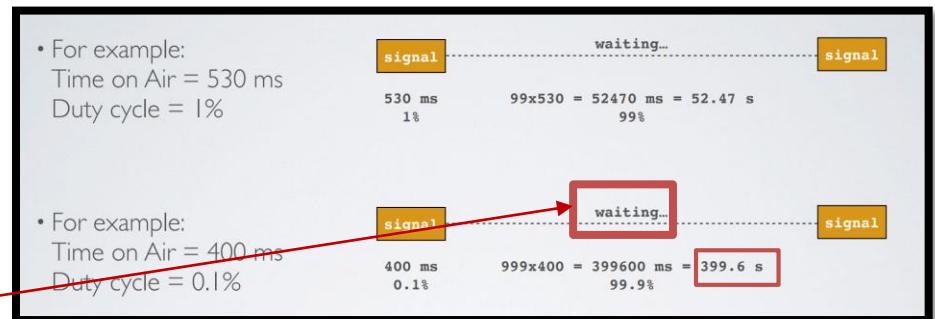


# LoRa: Time on air y Duty Cicle

- ▶ Cuando una señal es enviada desde un emisor toma un cierto tiempo en llegar al receptor. Este tiempo se denomina **Time on Air** o **ToA**.
- ▶ **Duty cycle** es la proporción de tiempo durante el cual un componente, dispositivo o sistema esta operando. El duty cycle puede ser expresado como un ratio o porcentaje.
- ▶ Determinadas regulaciones establecen políticas de un 0.1% o 1% de duty cycle por día, dependiendo del canal de emisión. Esto implica un tiempo de espera entre transmisiones.



Fuente: [https://www.mobilefish.com/download/lora/lora\\_part3.pdf](https://www.mobilefish.com/download/lora/lora_part3.pdf)



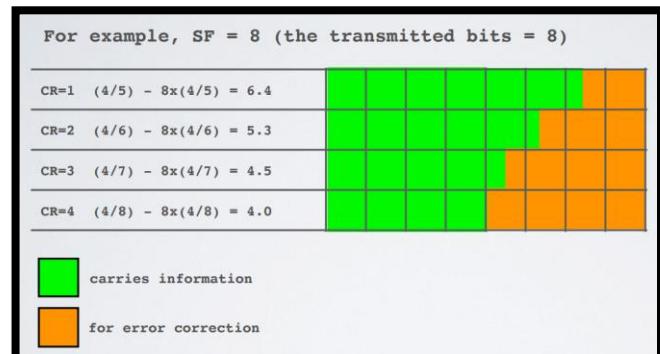
# LoRa: Forward Error Correction & Code Rate

- ▶ **Forward Error Correction (FEC):** es el proceso mediante el cual se añaden bits destinados a la corrección de errores en la transmisión de datos. Estos bits de redundancia ayudan a restaurar la información cuando esta se corrompe por una interferencia.
- ▶ A mas bit de corrección, más fácil la corrección
- ▶ A más bits de corrección, más datos son transmitidos lo que afecta a la autonomía.
- ▶ **Coding Rate (CR):** hace referencia a la proporción de bits transmitidos que portan la información.
- ❑ Parámetro: CR → 4/5, 4/6, 4/7 o 4/8.

CR → 1, 2, 3, 4.

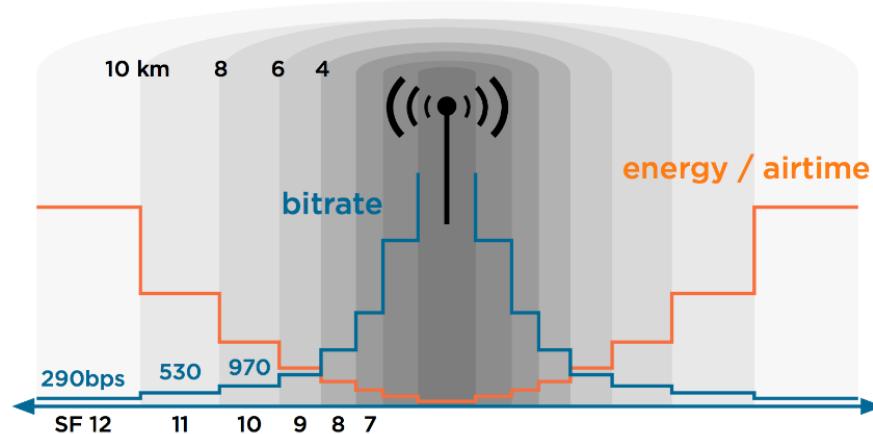
| Coding Rate (CR) | $CR = 4 / (4 + CR)$ |
|------------------|---------------------|
| 1                | 4/5                 |
| 2                | 4/6                 |
| 3                | 4/7                 |
| 4                | 4/8                 |

Fuente: [https://www.mobilefish.com/download/lora/lora\\_part14.pdf](https://www.mobilefish.com/download/lora/lora_part14.pdf)



## LoRa: Adaptative data rate

- ▶ **Adaptive Data Rate:** dinámicamente ajusta el SF teniendo en cuenta las condiciones ambientales del dispositivo
- Parámetro: ADR → On/off



# 3. LoRaWAN

Protocolo LoRaWAN



VNiVERSiDAD  
DE SALAMANCA  
CAMPUS DE EXCELENCIA INTERNACIONAL

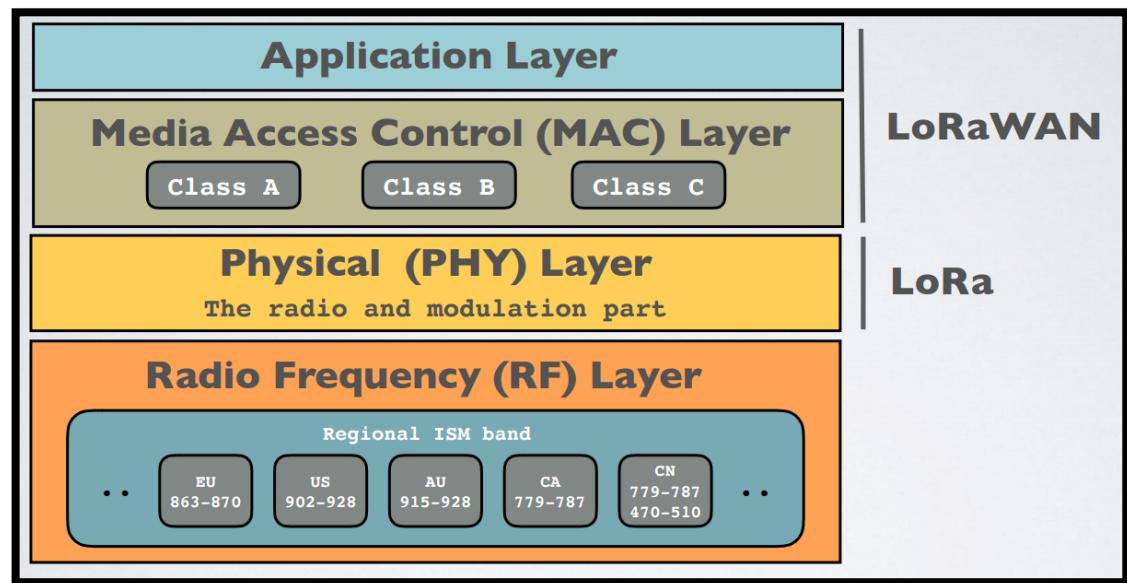
## LoRaWAN

- ▶ **LoRaWAN:** define tanto el protocolo de comunicación como la arquitectura del sistema para la red. Emplea LoRa como su capa física de comunicación que permite enlaces a grandes distancias. El protocolo y la arquitectura de red tienen mucha influencia en la autonomía final de los dispositivos, la capacidad de la red, la seguridad, la calidad del servicio y las aplicaciones finales que utilizan la red.



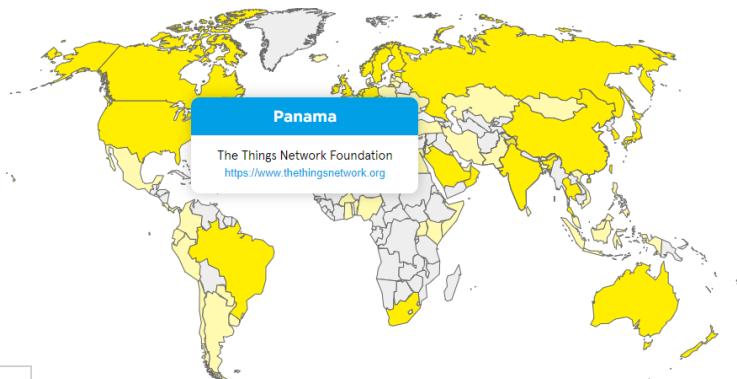
# LoRaWAN

## ► LoRaWAN STACK



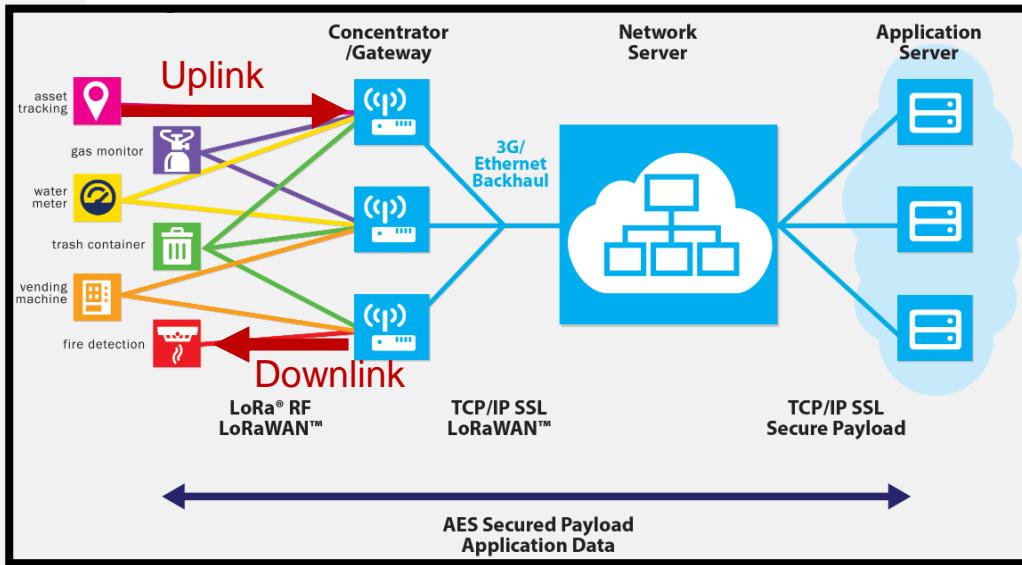
# LoRaWAN

- ▶ Los protocolos LoRaWAN son definidos por la LoRa Alliance. 
- ▶ Se trata de una organización sin animo de lucro formada por mas de 500 empresas del sector de la electrónica y telecomunicaciones, comprometidas con habilitar un despliegue a gran escala de redes LPWAN para IoT y con promocionar LoRaWAN como un estándar abierto.



# LoRaWAN

## ► Esquema de comunicación en LoRaWAN



### Funcionamiento:

- End-node emite su mensaje a los gateways cercanos
- Los gateways reenvían este mensaje al Network Server (NS)
- El NS recopila todos los mensajes de todos los gateways y filtra datos duplicados y determina que gateway tiene la mejor recepción
- El NS reenvía el paquete al servidor de aplicaciones donde el usuario puede procesar los datos.
- De manera opcional el servidor de aplicaciones puede enviar un mensaje al nodo como respuesta.
- Cuando una respuesta es enviada el NS la recibe y determina que GW emplear para retransmitirla al nodo

# LoRaWAN vs competidores

► LoRaWAN vs competidores

| Feature                      | LoRaWAN         | Narrow-Band                     | LTE Cat-1<br>2016 (Rel12) | LTE Cat-M<br>2018 (Rel13) | NB-LTE<br>2019(Rel13+)     |
|------------------------------|-----------------|---------------------------------|---------------------------|---------------------------|----------------------------|
| Modulation                   | SS Chirp        | UNB / GFSK/BPSK                 | OFDMA                     | OFDMA                     | OFDMA                      |
| Rx bandwidth                 | 500 - 125 KHz   | 100 Hz                          | 20 MHz                    | 20 - 1.4 MHz              | 200 KHz                    |
| Data Rate                    | 290bps - 50Kbps | 100 bit/sec<br>12 / 8 bytes Max | 10 Mbit/sec               | 200kbps –<br>1Mbps        | ~20K bit/sec               |
| Max. # Msgs/day              | Unlimited       | UL: 140 msgs/day                | Unlimited                 | Unlimited                 | Unlimited                  |
| Max Output Power             | 20 dBm          | 20 dBm                          | 23 - 46 dBm               | 23/30 dBm                 | 20 dBm                     |
| Link Budget                  | 154 dB          | 151 dB                          | 130 dB+                   | 146 dB                    | 150 dB                     |
| Batery lifetime -<br>2000mAh | 105 months      | 90 months                       |                           | 18 months                 |                            |
| Power Efficiency             | Very High       | Very High                       | Low                       | Medium                    | Med high                   |
| Interference immunity        | Very high       | Low                             | Medium                    | Medium                    | Low                        |
| Coexistence                  | Yes             | No                              | Yes                       | Yes                       | No                         |
| Security                     | Yes             | No                              | Yes                       | Yes                       | Yes                        |
| Mobility / localization      | Yes             | Limited mobility,<br>No loc     | Mobility                  | Mobility                  | Limited Mobility<br>No Loc |

# LoRaWAN Security

- ▶ Un nodo debe ser previamente activado antes de que pueda comunicarse con los NS.
- ▶ Existen dos tipos de activación de dispositivos:
- **Over-The-Air-Activation (OTAA)**: opción preferida ya que provee la forma más segura para conectar nodos al NS. Los nodos deben previamente conocer y almacenar su
  - DevEUI: identifica de forma única al dispositivo (similar a una MAC)
  - AppEUI : identifica de forma única al servidor de aplicaciones (similar a un numero de puerto)
  - AppKey: es una AES key de 128 bits que es empleada para generar los Message Integrity Codes (MIC) para asegurar la integridad de los mensajes. Tanto el nodo como el NS deben almacenar la misma Appkey.

EUI: Extended Unique Identifier (64-bits de longitud)

- **Activation-By-Personalization (ABP)**: En este modo, el dispositivo almacena directamente variables como DevAddr AppSKey y NwkSKey.
- Una explicación detallada de cada uno de los procedimientos puede encontrarse en la [especificación LoRaWAN](#).

# 4. Dispositivos

Tipos de dispositivos LoRaWAN



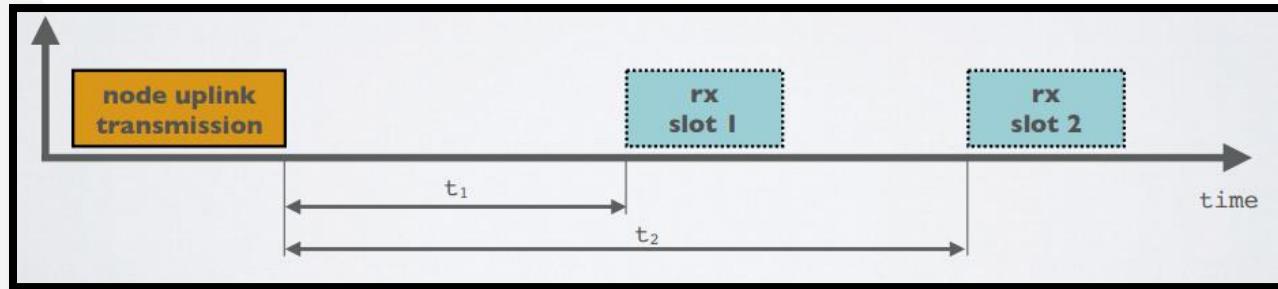
# Dispositivos LoRaWAN

## ► Tipos de dispositivos en el protocolo LoRaWAN

| Class        | Description  |
|--------------|--|
| A(II)        | Battery powered devices. Each device uplink to the gateway and is followed by two short downlink receive windows.        |
| B(eacon)     | Same as class A but these devices also opens extra receive windows at scheduled times.                                   |
| C(continuos) | Same as A but these devices are continuously listening. Hence these devices uses more power and are often mains powered. |

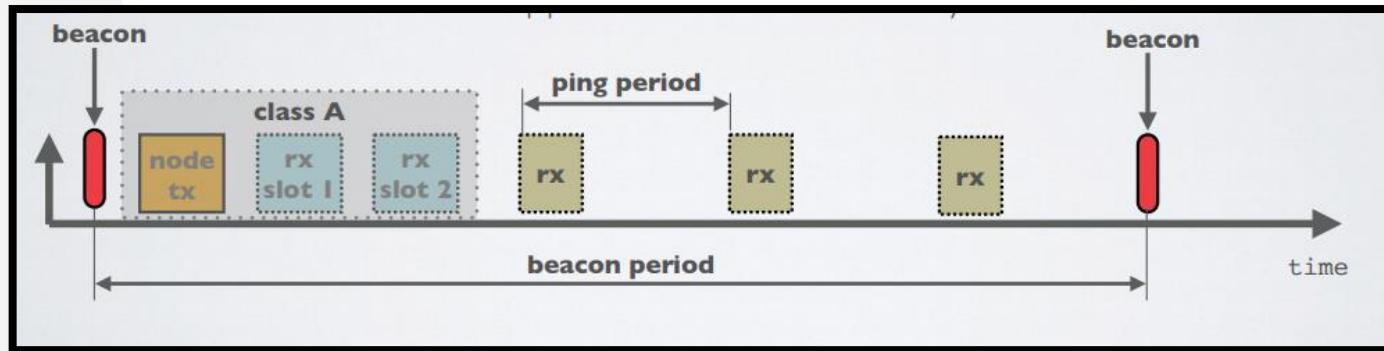
# Dispositivos LoRaWAN

- ▶ CLASE A: En cualquier momento puede retransmitir un mensaje. Despues del *uplink (TX)* el nodo abrirá una ventana para escuchar por una respuesta del GW. El nodo abre 2 ventanas de recepción  $t_1$  y  $t_2$  y el GW puede responder en cualquiera de las dos.
- ▶ Los dispositivos clase B y C deben soportar la funcionalidad de los tipos A.



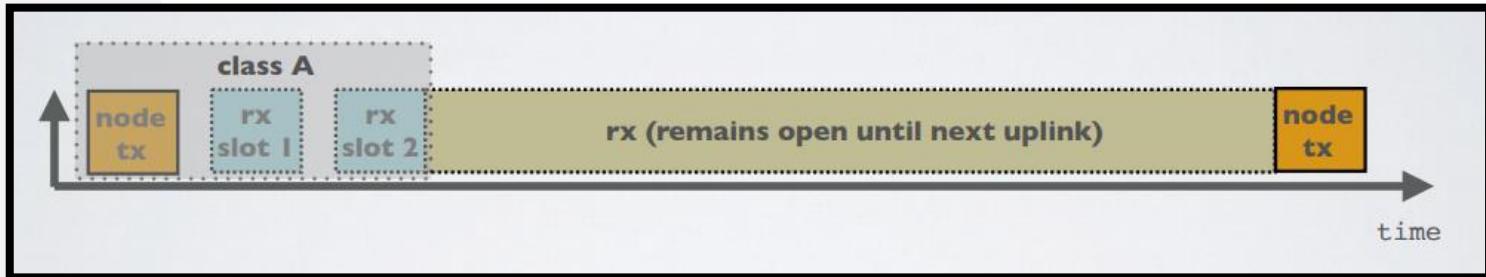
# Dispositivos LoRaWAN

- ▶ CLASE B: de manera adicional a las ventana de recepción de la clase A, este tipo de dispositivos abren otras ventanas de recepción en tiempo programados.
- ▶ El nodo recibe un beacon de sincronización del GW, permitiendo así a este conocer cuando el nodo estará escuchando.
- ▶ Los dispositivos de clase B no soportan la funcionalidad de los clase C.



# Dispositivos LoRaWAN

- ▶ CLASE C: de manera adicional a las ventana de recepción de la clase A, este tipo de dispositivos escuchan continuamente por respuestas del GW.
- ▶ Los dispositivos de clase C no soportan la funcionalidad de los clase B.



# 5. TTN

The Things Network



VNiVERSiDAD  
DE SALAMANCA  
CAMPUS DE EXCELENCIA INTERNACIONAL

# The Things Network

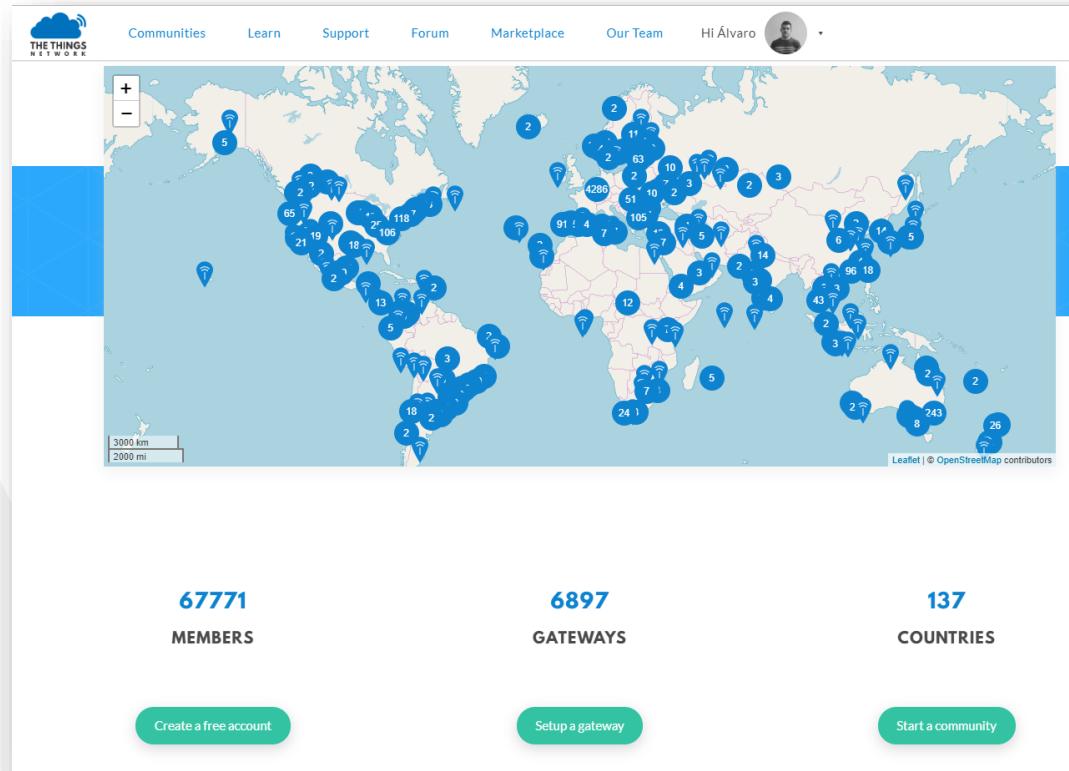
- ▶ Una red colaborativa que implementa LoRaWAN

## The Things Network Mission

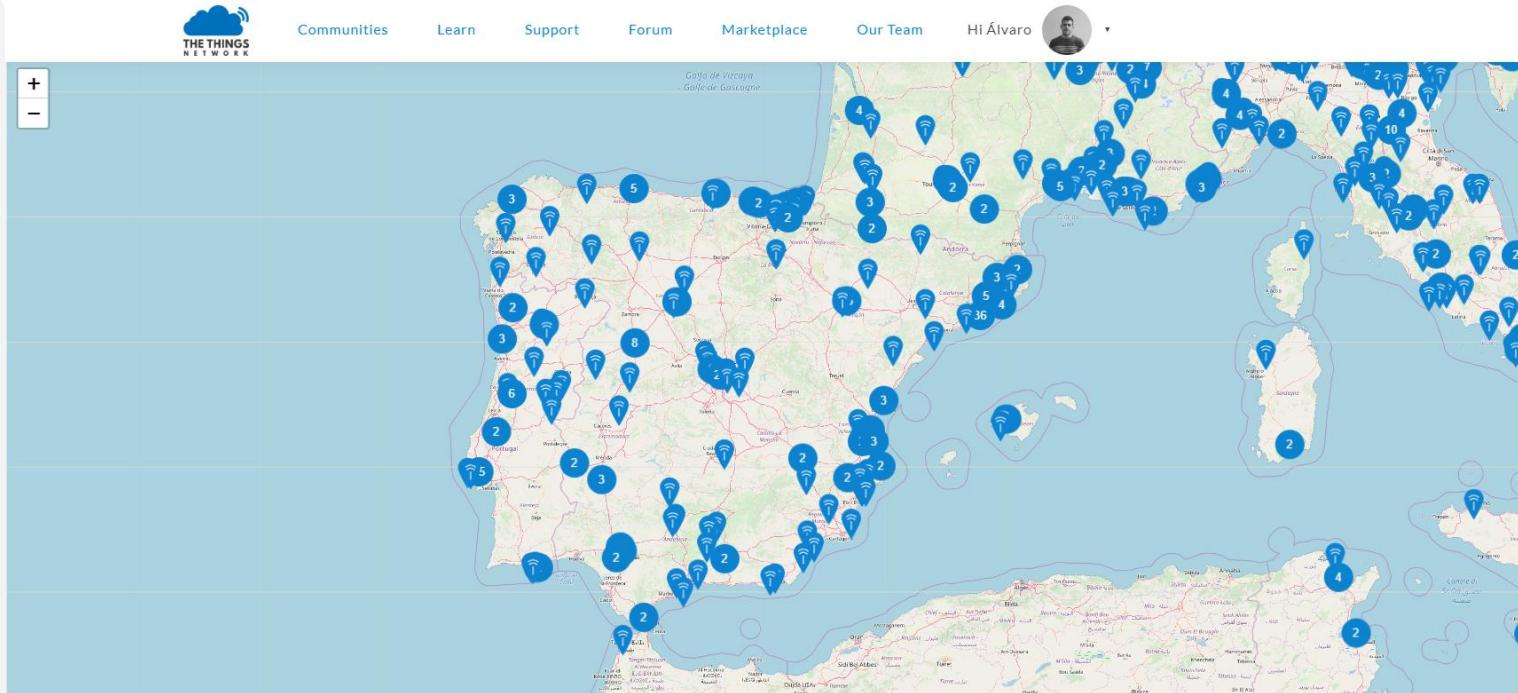
Our **mission**  
is to build  
**a decentralized,**  
**open** and  
**crowd sourced**  
IoT data network

**Owned** and **operated** by its users

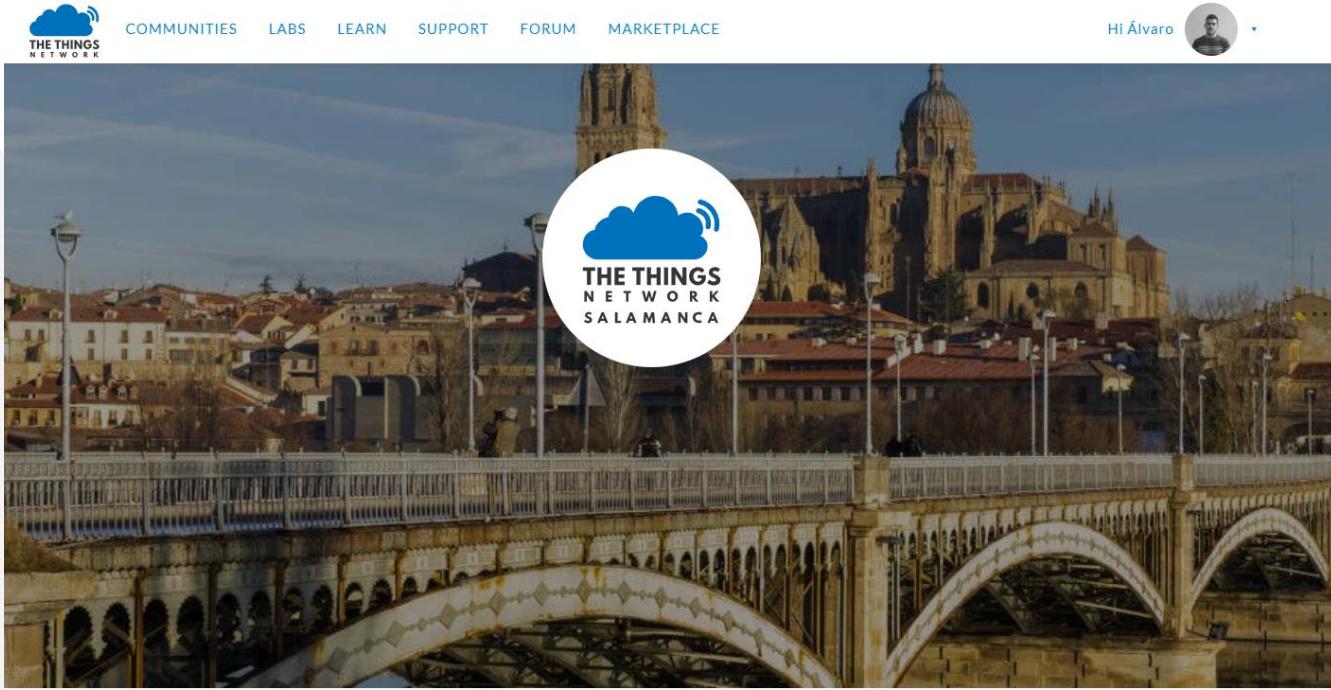
# The Things Network



# The Things Network - Panamá



# The Things Network - Panamá

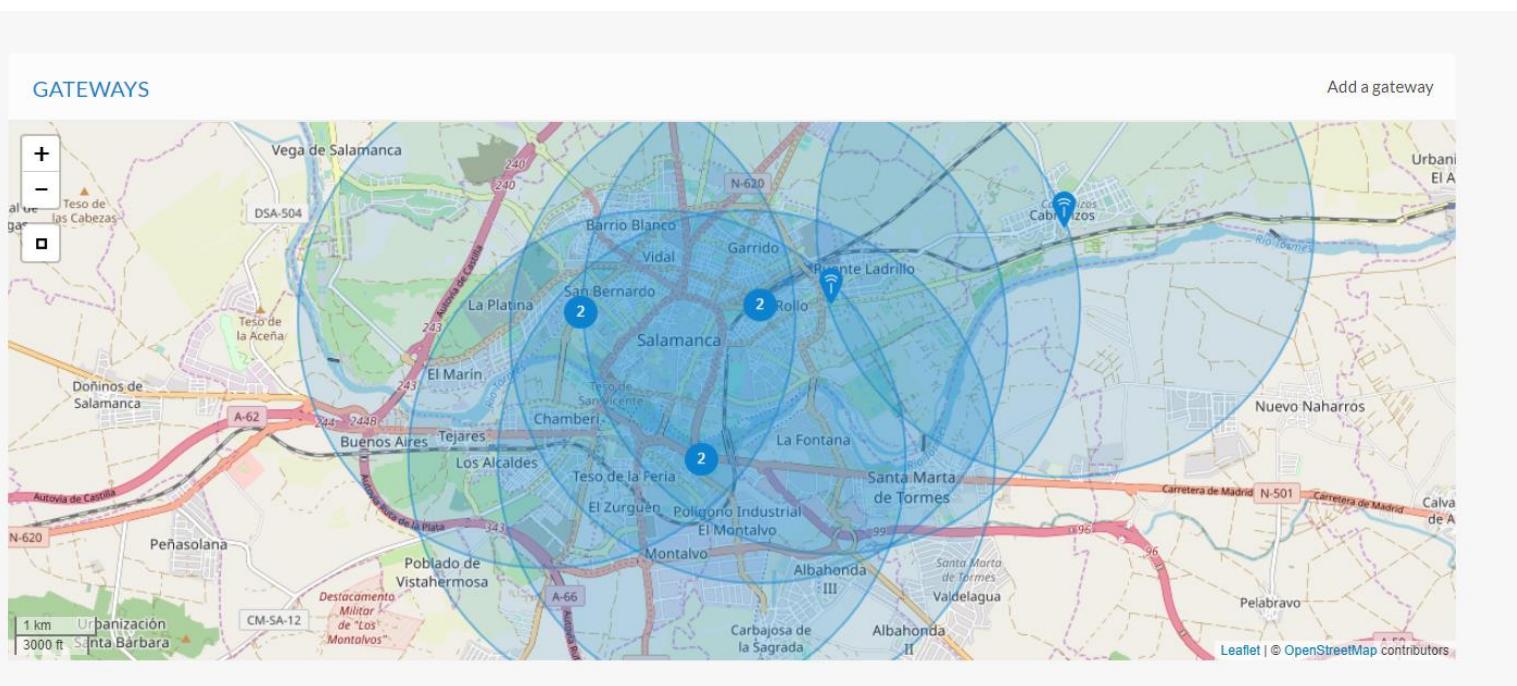


THE THINGS  
NETWORK  
SALAMANCA

COMMUNITIES LABS LEARN SUPPORT FORUM MARKETPLACE

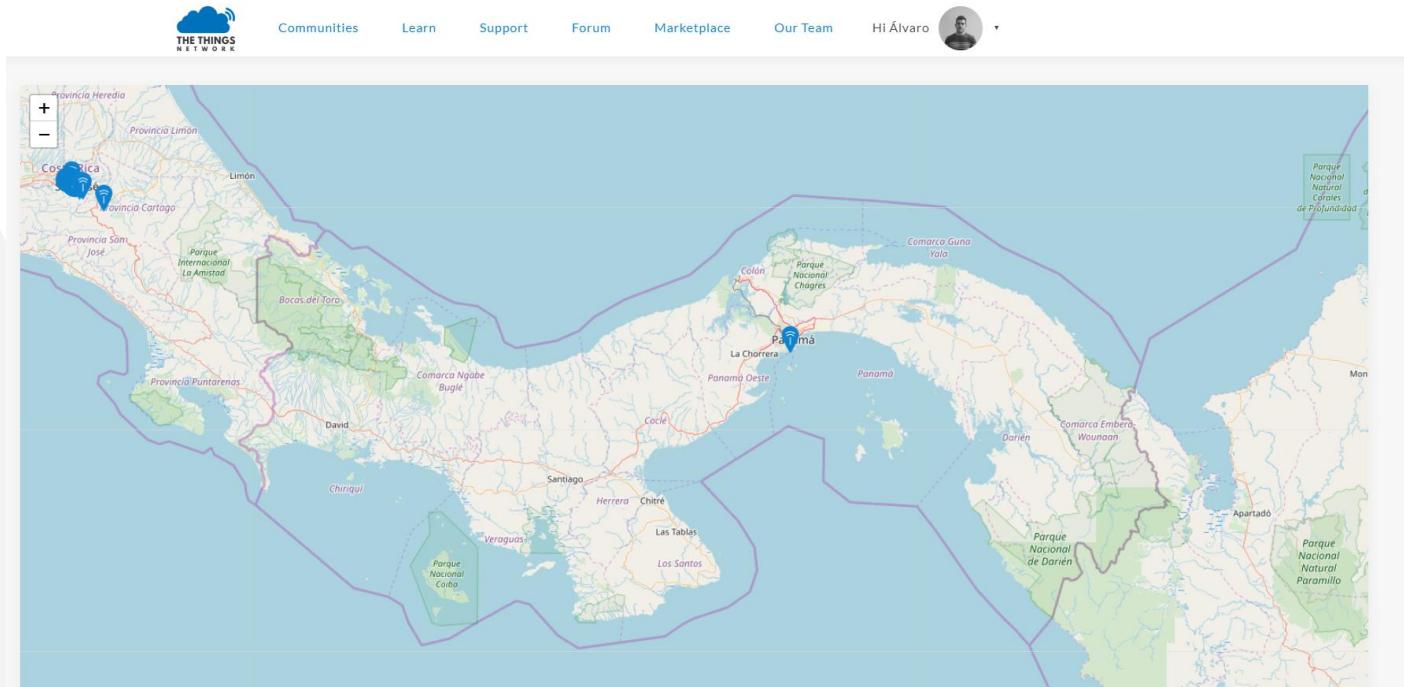
Hi Álvaro

# The Things Network - Panamá



# The Things Network - Panamá

## MAPA



# 6. Enlace P2P LoRa

Creación de un enlace P2P LoRa



## Creación de un enlace P2P LoRa

- ▶ Se realizará un enlace empleando la capa física de radio LoRa. **No el protocolo LoRaWAN!**



# Creación de un enlace P2P LoRa



## Material empleado:

- 2 Módulos TTGO v1 915Mhz version (Para Panamá)
- Código disponible en:

<https://github.com/elloza/LPWAN-IoT-LoRa>



El código presenta 2 dependencias:

- LoRa Arduino library
- Esp32 OLED 1306



Es necesario haber configurado previamente  
Arduino para utilizar placas basadas en ESP32:

[\[Tutorial\]](#)



## Creación de un enlace P2P LoRa

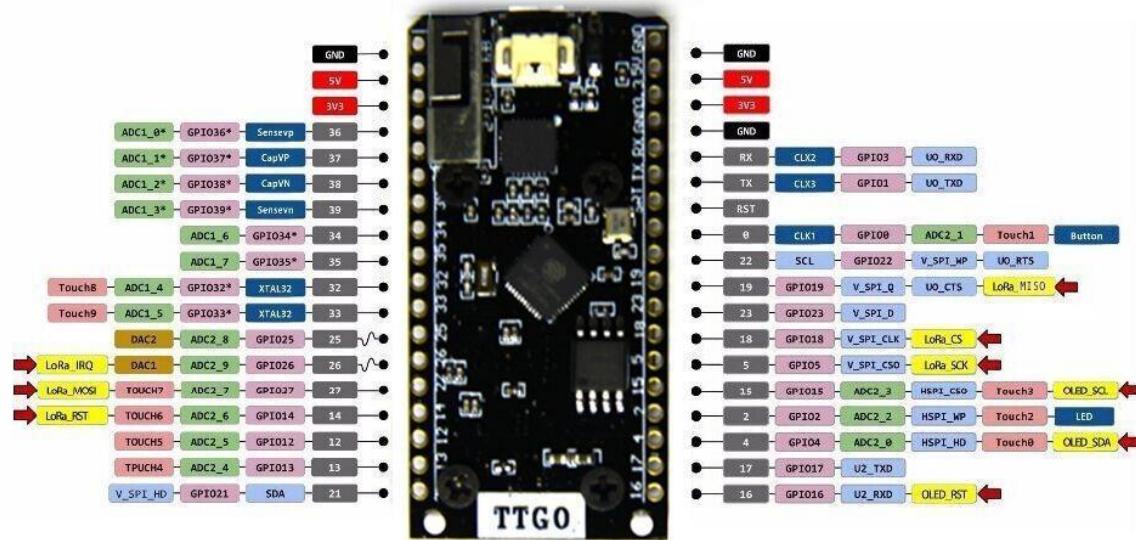
- ▶ Hardware:



[Link Compra](#)

# Creación de un enlace P2P LoRa

## ► Pinout:



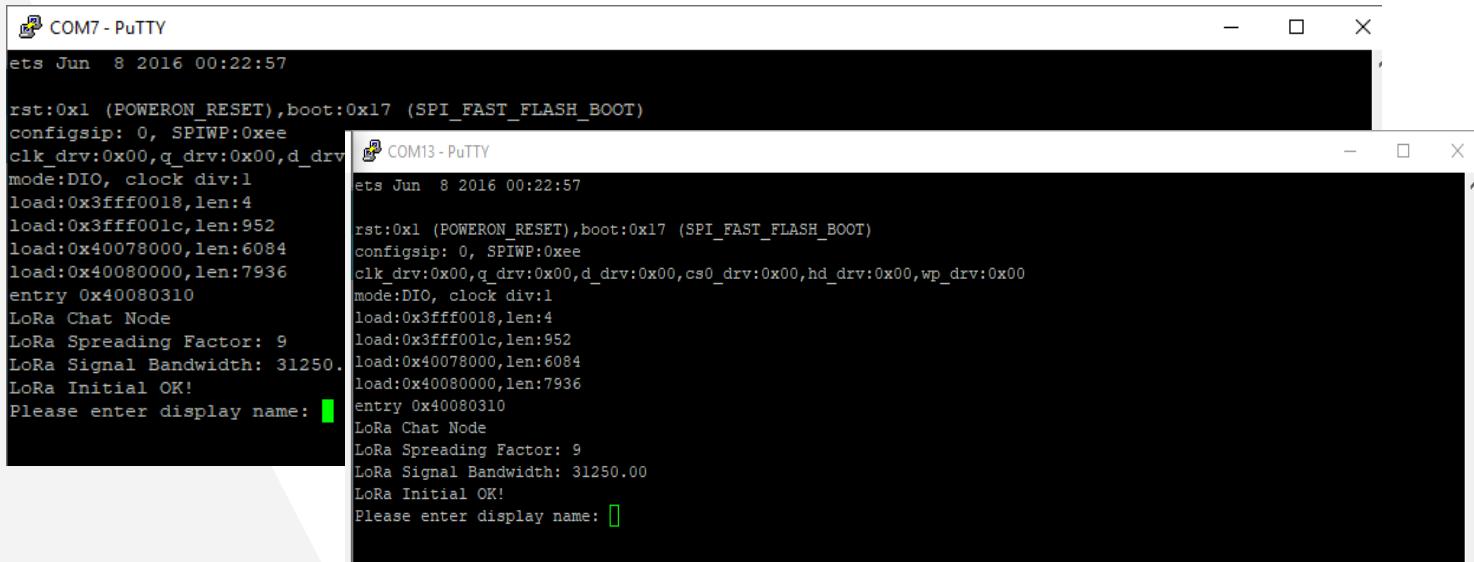
# Creación de un enlace P2P LoRa

- ▶ Resultado:



# Creación de un enlace P2P LoRa

## ► Resultado:



The image shows two terminal windows side-by-side, both titled "COM7 - PuTTY". The left window displays the initial boot log for a LoRa module, while the right window shows a similar log with some additional configuration details.

**Left Window Log (COM7 - PuTTY):**

```
ets Jun  8 2016 00:22:57

rst:0x1 (POWERON_RESET),boot:0x17 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv
mode:DIO, clock div:1
load:0x3fff0018,len:4
load:0x3fff001c,len:952
load:0x40078000,len:6084
load:0x40080000,len:7936
entry 0x40080310
LoRa Chat Node
LoRa Spreading Factor: 9
LoRa Signal Bandwidth: 31250.
LoRa Initial OK!
Please enter display name: [REDACTED]
```

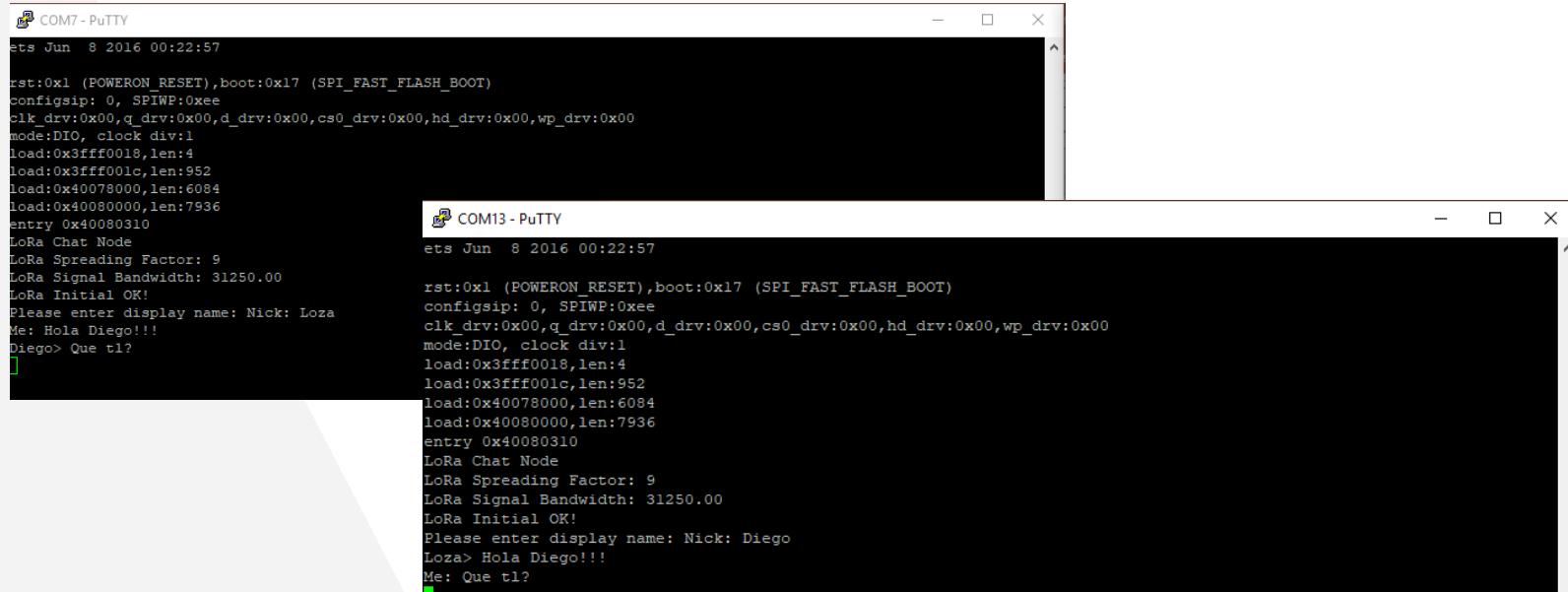
**Right Window Log (COM13 - PuTTY):**

```
ets Jun  8 2016 00:22:57

rst:0x1 (POWERON_RESET),boot:0x17 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:1
load:0x3fff0018,len:4
load:0x3fff001c,len:952
load:0x40078000,len:6084
load:0x40080000,len:7936
entry 0x40080310
LoRa Chat Node
LoRa Spreading Factor: 9
LoRa Signal Bandwidth: 31250.00
LoRa Initial OK!
Please enter display name: [REDACTED]
```

# Creación de un enlace P2P LoRa

## ► Resultado:



The image shows two terminal windows, both titled "COM - PuTTY". The left window is labeled "COM7 - PuTTY" and the right window is labeled "COM13 - PuTTY". Both windows display the same text output, which is a log of LoRa node initialization and communication.

```
ets Jun  8 2016 00:22:57

rst:0x1 (POWERON_RESET),boot:0x17 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:1
load:0x3fff0018,len:4
load:0x3fff001c,len:952
load:0x40078000,len:6084
load:0x40080000,len:7936
entry 0x40080310
LoRa Chat Node
LoRa Spreading Factor: 9
LoRa Signal Bandwidth: 31250.00
LoRa Initial OK!
Please enter display name: Nick: Loza
Me: Hola Diego!!!
Diego> Que tl?
```

The text in the logs includes boot parameters like power-on reset, SPI flash boot, and various driver configurations. It also shows the LoRa spreading factor (9) and signal bandwidth (31250.00). The communication part shows two nodes, "Loza" and "Diego", exchanging messages "Hola" and "Que tl?".

## Creación de un enlace P2P LoRa

- ▶ Resultado:



# 7. Nodo LoRaWAN

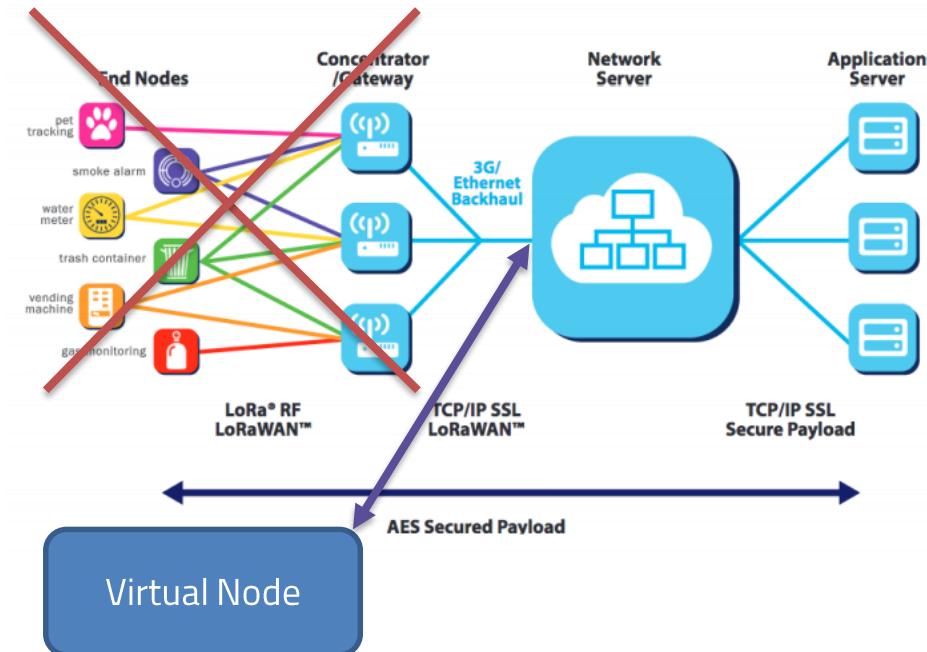
Simulación de un Nodo LoRaWAN TTN



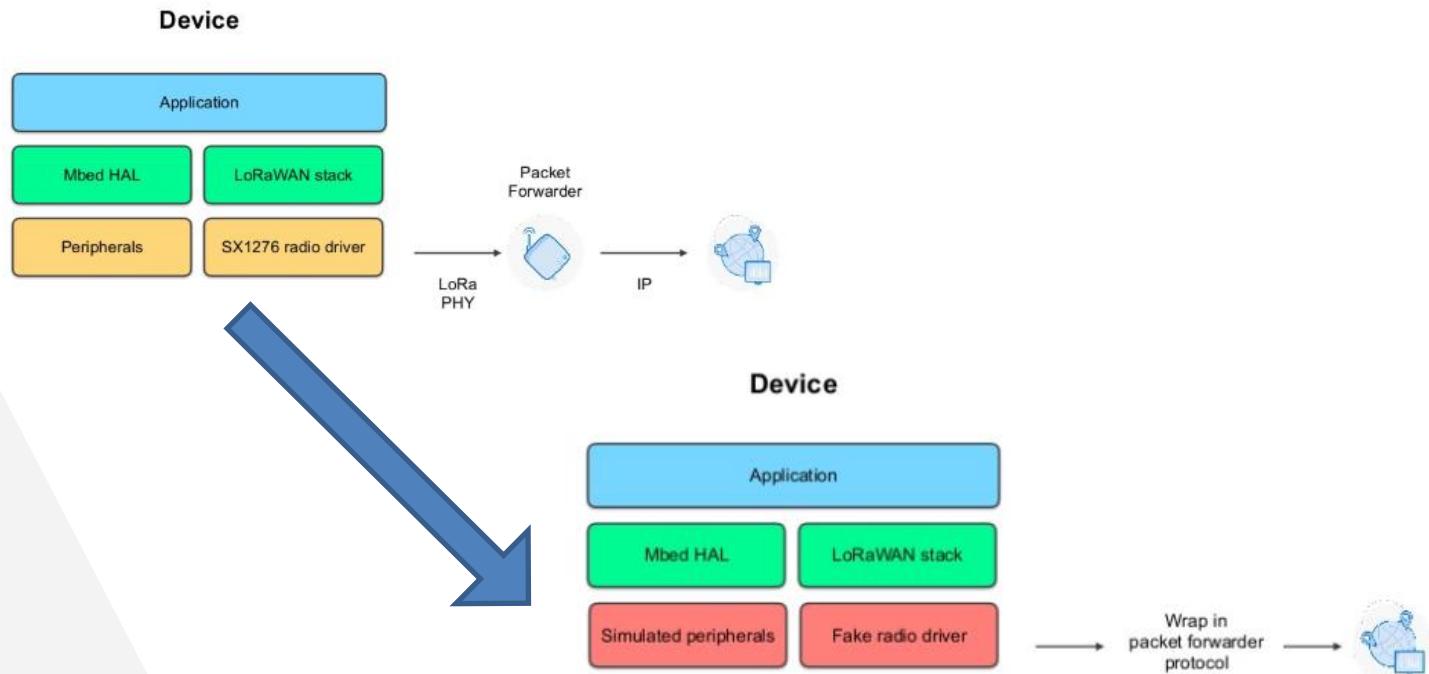
VNiVERSiDAD  
DE SALAMANCA  
CAMPUS DE EXCELENCIA INTERNACIONAL

# Simulación de un Nodo LoRaWAN TTN

- ▶ Requisitos previos:
  - ▶ Crear una cuenta en TTN
  - ▶ Navegador web



# Simulación de un Nodo LoRaWAN TTN



# Simulación de un Nodo LoRaWAN TTN

- ▶ Acceder al simulador ARM online
  - ▶ <https://github.com/janjongboom/mbed-simulator/tree/lora>
  - ▶ Hacer clic en la DEMO
- ▶ Seleccionar la demo de LoRaWAN

Arm Mbed OS simulator

LoRaWAN • Load demo

Download offline version | GitHub project

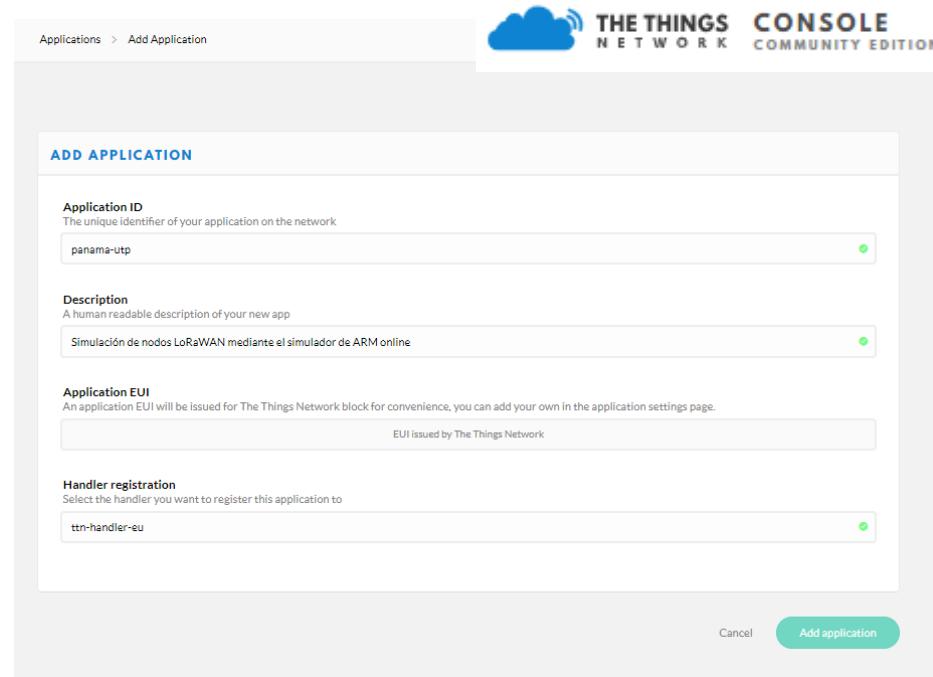
Serial output

Set your LoRaWAN credentials first!

```
int main() {
```

# Simulación de un Nodo LoRaWAN TTN

- ▶ En la cuenta de The Things Network:
  - ▶ Crear una aplicación



The screenshot shows the 'Add Application' page in the The Things Network Console. The header includes the logo and text 'THE THINGS NETWORK CONSOLE COMMUNITY EDITION'. The main form has a title 'ADD APPLICATION' and several fields:

- Application ID:** A text input field containing 'panama-utp' with a green checkmark icon.
- Description:** A text input field containing 'Simulación de nodos LoRaWAN mediante el simulador de ARM online' with a green checkmark icon.
- Application EUI:** A text input field with placeholder text 'An application EUI will be issued for The Things Network block for convenience, you can add your own in the application settings page.' and a note 'EUI issued by The Things Network' below it.
- Handler registration:** A text input field containing 'ttn-handler-eu' with a green checkmark icon.

At the bottom right are 'Cancel' and 'Add application' buttons.

# Simulación de un Nodo LoRaWAN TTN

## ► Registrar un dispositivo

The screenshot shows the TTN application interface. At the top, there's a navigation bar with tabs: Overview, Devices, Payload Formats, Integrations, Data, and Settings. The Overview tab is selected. Below it, the Application Overview section displays the Application ID (panama-utp), a description (Simulación de nodos LoRaWAN mediante el simulador de ARM online), and a creation timestamp (15 seconds ago). The Handler is set to ttn-handler-eu (current handler). The Application EUIS section shows a text input field with the value 70 83 D5 7E D0 01 A5 16. The Devices section shows a button labeled "register device" with a red box drawn around it. The bottom part of the interface shows a summary: 0 registered devices.

Que genere un EUI por nosotros

The screenshot shows the "REGISTER DEVICE" form. It has fields for "Device ID" (set to "virtual-device"), "Device EUI" (containing the value "2C" which is highlighted with a red box), "App Key" (a placeholder field with a note: "this field will be generated"), and "App EUI" (containing the value "70 83 D5 7E D0 01 A5 16"). A large blue arrow points from the "register device" button in the previous screenshot to this form. A red box highlights the "Device EUI" field. The bottom right corner of the form has a "Register" button, which is also highlighted with a red box.

# Simulación de un Nodo LoRaWAN TTN

- Dispositivo registrado: Deberemos copiar Device EUI, APP EUI y App Key

DEVICE OVERVIEW

Application ID panama-utp

Device ID virtual-device

Activation Method OTAA

Device EUI

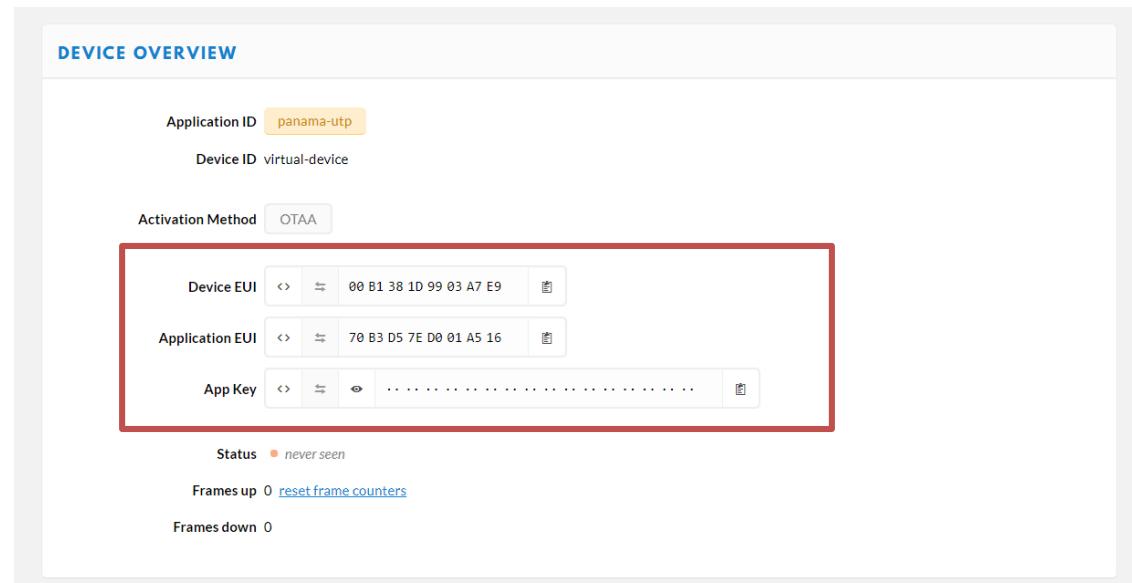
Application EUI

App Key

Status never seen

Frames up 0 [reset frame counters](#)

Frames down 0



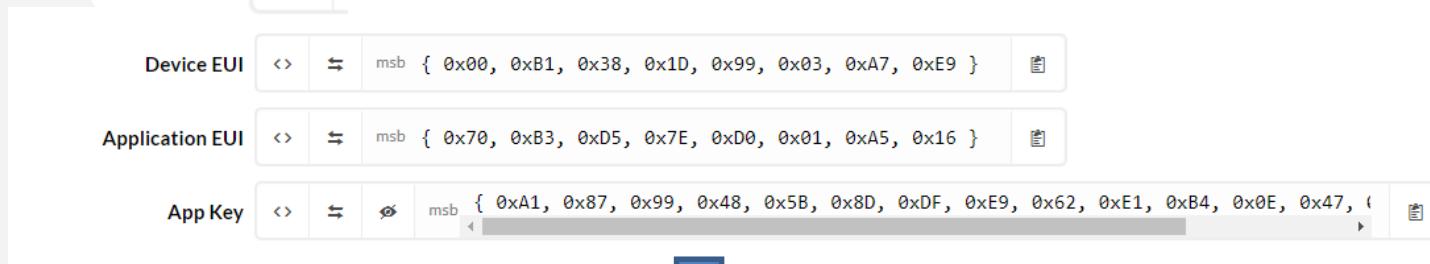
The screenshot shows the 'DEVICE OVERVIEW' section of the TTN web interface. It displays the following information:

- Application ID:** panama-utp
- Device ID:** virtual-device
- Activation Method:** OTAA
- Device EUI:** 00 B1 38 1D 99 03 A7 E9
- Application EUI:** 70 B3 D5 7E D0 01 A5 16
- App Key:** (redacted)
- Status:** never seen (indicated by an orange dot)
- Frames up:** 0 ([reset frame counters](#))
- Frames down:** 0

A red box highlights the three key identifier fields: Device EUI, Application EUI, and App Key.

# Simulación de un Nodo LoRaWAN TTN

- Deberemos copiar en MSB, para ello pulsaremos en el botón  de cada uno de los EUI y App key.



The screenshot shows three input fields for device credentials:

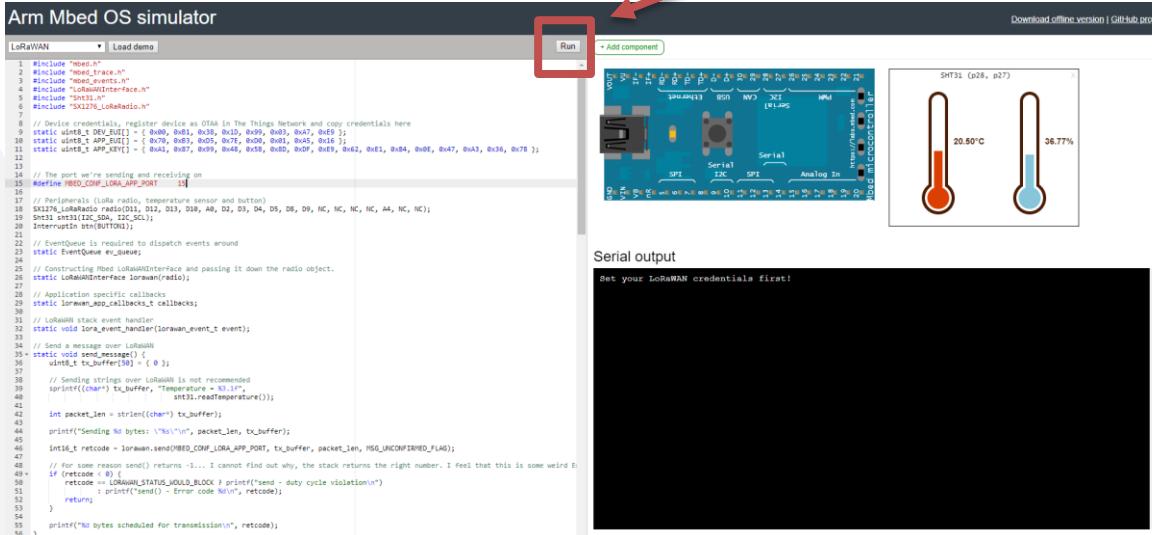
- Device EUI: msb { 0x00, 0xB1, 0x38, 0x1D, 0x99, 0x03, 0xA7, 0xE9 }
- Application EUI: msb { 0x70, 0xB3, 0xD5, 0x7E, 0xD0, 0x01, 0xA5, 0x16 }
- App Key: msb { 0xA1, 0x87, 0x99, 0x48, 0x5B, 0x8D, 0xDF, 0xE9, 0x62, 0xE1, 0xB4, 0x0E, 0x47, 0x1F }



```
8 // Device credentials, register device as OTAA in The Things Network and copy credentials here
9 static uint8_t DEV_EUI[] = { 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0 };
10 static uint8_t APP_EUI[] = { 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0 };
11 static uint8_t APP_KEY[] = { 0x0, 0x0 };
```

# Simulación de un Nodo LoRaWAN TTN

- ▶ Ejecutamos el código con el botón run:



The screenshot shows the Arm Mbed OS simulator interface. On the left is the code editor with the file name "LoRaWAN". The code is a C program for a LoRaWAN node. In the center is a schematic diagram of a LoRaWAN node with various components like a radio module, microcontroller, and sensors. On the right, there's a "Serial output" window and a "Sensor data" window. The "Sensor data" window displays two analog inputs from SHT31 sensors: one at 20.50°C and another at 36.77% relative humidity.

```

LoRaWAN
Load demo
Run + Add component
Download offline version | GitHub project

Schematic Diagram:
[Diagram showing a LoRaWAN node with a radio module, microcontroller, and various sensors and connectors]

Sensor data:
SHT31 (p28, p27)
20.50°C
36.77%

Serial output:
Set your LoRaWAN credentials first!

```

```

LoRaWAN
Load demo
Run + Add component
Download offline version | GitHub project

LoRaWAN
1 #include "mbed.h"
2 #include "radio_trace.h"
3 #include "mbed_events.h"
4 #include "LoRaWANInterface.h"
5 #include "SHT31.h"
6 #include "SX1276_LoRaRadio.h"
7
8 // Device credentials, replace device as OTAA in The Things Network and copy credentials here
9 static uint8_t DEV_EUI[] = { 0x00, 0x00, 0x39, 0x1D, 0x00, 0x07, 0x09 };
10 static uint8_t APP_EUI[] = { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 };
11 static uint8_t APP_KEY[] = { 0x01, 0x02, 0x03, 0x04, 0x05, 0x06, 0x07, 0x08, 0x09, 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F };
12
13 // The port we're sending and receiving on
14 static const uint8_t APP_PUBLISH_PORT = 1;
15
16 // Network details (LoRa radio, temperature sensor and button)
17 SX1276_LoRaRadio radio(D0, D1, D2, D3, D5, D9, A0, D2, D3, D4, D5, D8, D9, NC, NC, NC, NC, A4, NC, NC);
18 SHT31 sht31(I2C_SDA, I2C_SCL);
19 InterruptIn btn(BUTTON_PIN);
20
21 // EventQueue is required to dispatch events around
22 #include "EventQueue.h"
23 #include "LoRaWANInterface.h"
24
25 // Constructing New LoRaWANInterface and passing it down the radio object.
26 static LoRaWANInterface lorawan(radio);
27
28 // Application specific callbacks
29 static lorawan_app_callbacks_t callbacks;
30
31 // LoRaWAN stack event handler
32 static void lora_event_handler(lorawan_event_t event);
33
34 // Send a message over LoRaWAN
35 static void send_message() {
36     uint8_t tx_buffer[10] = { 0 };
37
38     // Sending strings over LoRa is not recommended
39     printf("Temp = %3.1f\n", sht31.readtemperature());
40
41     int packet_len = strlen((char*)tx_buffer);
42
43     printf("Sending %d bytes: \\\n\\n", packet_len, tx_buffer);
44
45     int8_t retcode = lorawan.send(0x00, COM_LORA_APP_PORT, tx_buffer, packet_len, MSG_UNCONFIRMED_FLAG);
46
47     // for some reason send() returns -1... I cannot find out why, the stack returns the right number. I feel that this is some weird E
48     if (retcode < 0) {
49         if (retcode == LORAWAN_STATUS_NO_DULG_BLOCK) {
50             printf("send - duty cycle violation\n");
51         } else {
52             printf("send - Error code %d\n", retcode);
53         }
54     }
55
56     printf("%d bytes scheduled for transmission\n", retcode);
57 }

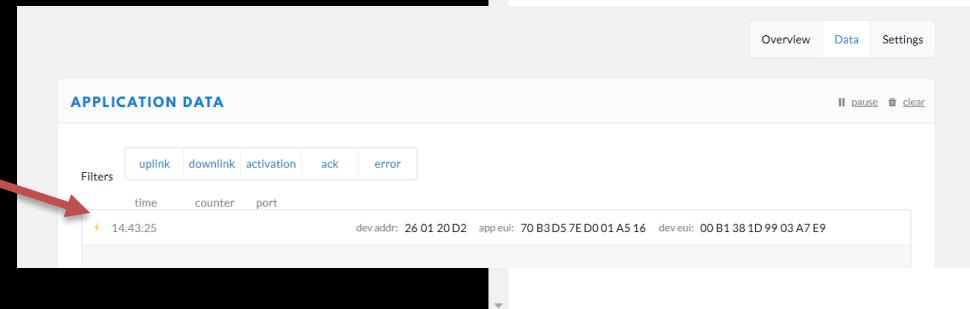
```

# Simulación de un Nodo LoRaWAN TTN

- ▶ Si todo ha ido bien..... El nodo se debería haber unido a la red LoRaWAN mediante OTAA y podremos verlo en TTN

## Serial output

```
Press BUTTON1 to send the current value of the temperature sensor!
[DBG ][LSTK]: Initializing MAC layer
[DBG ][LSTK]: Initiating OTAA
[DBG ][LSTK]: Sending Join Request ...
[DBG ][LMAC]: Frame prepared to send at port 0
[DBG ][LMAC]: TX: Channel=2, TX DR=5, RX1 DR=5
[DBG ][LRAD]: transmit channel=868500000 power=13 bandwidth=7 datarate=7
Connection - In Progress ...
[DBG ][LSTK]: Transmission completed
[DBG ][LRAD]: ] [LSTK]: Transmission completed
[DBG ][LMAC]: RX1 slot open, Freq = 868100000
[DBG ][LSTK]: OTAA Connection OK!
Connection - Successful
```

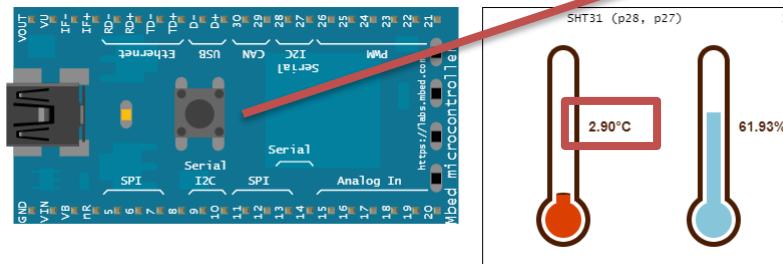


The screenshot shows the TTN Application Data interface. At the top, there are three tabs: Overview, Data (which is selected), and Settings. Below the tabs, the title "APPLICATION DATA" is centered. To the right of the title are three small icons: a play/pause button, a "clear" button, and a refresh/circular arrow icon. Underneath the title, there is a "Filters" section with five buttons: uplink, downlink, activation, ack, and error. Below the filters, there is a table with columns: time, counter, and port. A single row is visible in the table, with the timestamp "14:43:25" under the "time" column. At the bottom of the table, there are three status indicators: "dev addr: 26 01 20D2", "app eui: 70 B3 D5 7E D0 01 A5 16", and "dev eui: 00 B1 38 1D 99 03 A7 E9". A red arrow points from the "Successful" message in the serial output window to the "Connection - Successful" message in the TTN interface.

# Simulación de un Nodo LoRaWAN TTN

- ▶ Ahora si pulsamos el botón deberá enviarse un mensaje.

```
// Fire a message when the button is pressed
btn.fall(ev_queue.event(&send_message));
```



Serial output

```
Press BUTTON1 to send the current value of the temperature sensor!
[DBG ] [LSTK]: Initializing MAC layer
[DBG ] [LSTK]: Initiating OTAA
[DBG ] [LSTK]: Sending Join Request ...
[DBG ] [IMAC]: Frame prepared to send at port 0
[DBG ] [IMAC]: TX: Channel=2, TX DR=5, RX1 DR=5
[DBG ] [LRAD]: transmit channel=868500000 power=13 bandwidth=7 datarate=7
Connection - In Progress ...
[DBG ] [LSTK]: Transmission completed
[DBG ] [LRAD]: ] [LSTK]: Transmission completed
[DBG ] [IMAC]: RX1 slot open, Freq = 868100000
[DBG ] [LSTK]: OTAA Connection OK!
Connection - Successful
Sending 17 bytes: "temperature = 2.9"
[INFO] [IMAC]: RTS = 17 bytes, PEND = 0, Port: 15
[DBG ] [IMAC]: Frame prepared to send at port 15
[DBG ] [IMAC]: TX: Channel=1, TX DR=5, RX1 DR=5
[DBG ] [LRAD]: transmit channel=868300000 power=13 bandwidth=7 datarate=7
17 bytes scheduled for transmission
[DBG ] [LSTK]: Transmission completed
[DBG ] [IMAC]: RX1 slot open, Freq = 868500000
[DBG ] [IMAC]: RX2 slot open, Freq = 869525000
Message Sent to Network Server
```

# Simulación de un Nodo LoRaWAN TTN

- Y habrá llegado a TTN...

The screenshot shows a LoRaWAN simulation interface with two main panels. The top panel displays a log of messages:

| time       | counter | port |   |
|------------|---------|------|---|
| ▲ 14:48:31 | 0       | 15   | retry payload: 54 65 6D 70 65 72 61 74 75 72 65 20 3D 20 32 2E 39                       |
| ⚡ 14:43:25 |         |      | dev addr: 26 01 20 D2 app eui: 50 B3 D5 7E D0 01 A5 16 dev eui: 00 B1 38 1D 99 03 A7 E9 |

The bottom panel shows the details of the last message (14:48:31):

**Uplink**  
Payload  
54 65 6D 70 65 72 61 74 75 72 65 20 3D 20 32 2E 39

Fields  
no fields

Metadata

```
{ "time": "2019-04-05T12:48:31.978180582", "frequency": 868.3, "modulation": "LoRa", "data_rate": "SF7BW125", "coding_rate": "4/6", "gateways": [ { "gw_id": "eul-0242ae000084ee70", "timestamp": 305151000, "time": "2019-04-05T12:48:31.9712", "channel": 2, "rssi": -35, "snr": 5 } ] }
```

A red arrow points from the payload hex value in the log to the input field in the bottom panel. Another red arrow points from the input field to the output panel.

Enter 2 digits hex numbers with any prefix / postfix / button (e.g. 45 78 61 6d 70 6C 65 21):

54656D7065726174757265203D20322E39

Convert Reset Swap

Temperature = 2.9

# Simulación de un Nodo LoRaWAN TTN

- También es posible la comunicación en sentido inverso (*Downlink*) a través de TTN. El mensaje en este caso será recibido justo después de producirse un *Uplink*.

The screenshot shows the TTN Downlink simulator interface. On the left, there's a configuration panel with tabs for 'DOWNLINK' and 'UPLINK'. The 'DOWNLINK' tab is active, showing fields for 'Scheduling' (with options 'replace', 'first', and 'last') and 'Payload' (containing 'AA AA'). Above the payload, a code snippet defines a port:

```
// The port we're sending and receiving on  
#define MBED_CONF_LORA_APP_PORT 15
```

A red arrow points from the highlighted '15' in the code to the 'FPort' field in the scheduling panel, indicating they refer to the same port number. Below the scheduling panel is a 'Send' button.

To the right, a terminal window displays the transmission process and its successful reception by the Network Server:

```
Temperature = 6.1"  
bytes, PEND = 0, Port: 15  
Scheduled to send at port 15  
dl=1, TX DR=5, RX1 DR=5  
channel=868300000 power=13 bandwidth=7 datarate=7  
transmission  
on completed  
open, Freq = 868100000  
open, Freq = 869525000  
Server  
Temperature = 6.1"  
bytes, PEND = 0, Port: 15  
Scheduled to send at port 15  
dl=2, TX DR=5, RX1 DR=5  
channel=868300000 power=13 bandwidth=7 datarate=7  
transmission  
on completed  
[DBG ][LSTK]: Transmission completed  
[DBG ][LNWK]: TX/RX slot open, Freq = 868300000  
[DBG ][LSTK]: Pack received 2 bytes, Port=15  
Message Sent to Network Server  
Received message from Network Server  
Data received on port 15 (length 2): aa aa
```

A red arrow points from the 'aa aa' in the transmitted payload to the 'aa aa' in the received data, illustrating the bidirectional communication.

# 8. Recursos

Bibliografía y recursos útiles para empezar



VNiVERSiDAD  
DE SALAMANCA

CAMPUS DE EXCELENCIA INTERNACIONAL

# Recursos y bibliografía

## ❑ Repositorios:

- [Simulador MBED LoRaWAN](#)
- [TTN ESP32 Arduino](#)

## ❑ Guías y manuales:

- [LoRaWAN simple explained](#)
- [LoRa/LoRaWAN Youtube List](#)
- [ESP32 TTGO](#)
- [LoRa Crash Course from TTN](#)
- [Brico Labs LoRa y TTN](#)
- [Decoding LoRa PHY](#)
- [LoRa y ESP32](#)

## ❑ Hardware:

- [Andreas Spiess LoRa ESP32 HW](#)
- [2 ESP32 TTGO](#)

# ¡Muchas gracias!

¿Preguntas? 

