

LoRaWAN ChirpStack

## Contenido

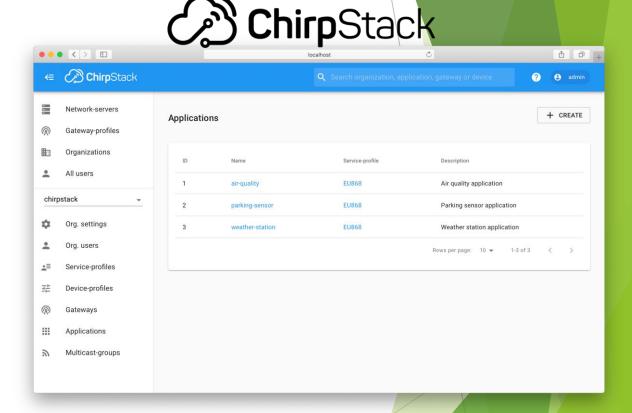
Componentes de ChirpStack Instalación

Instalación en docker

ChirpStack Gateway Installation Guide

Conectar un dispositivo

- https://www.chirpstack.io/
- Componentes open-source para redes LoRaWAN
- Interfaz Web para gestión de dispositivos
- APIs de integración



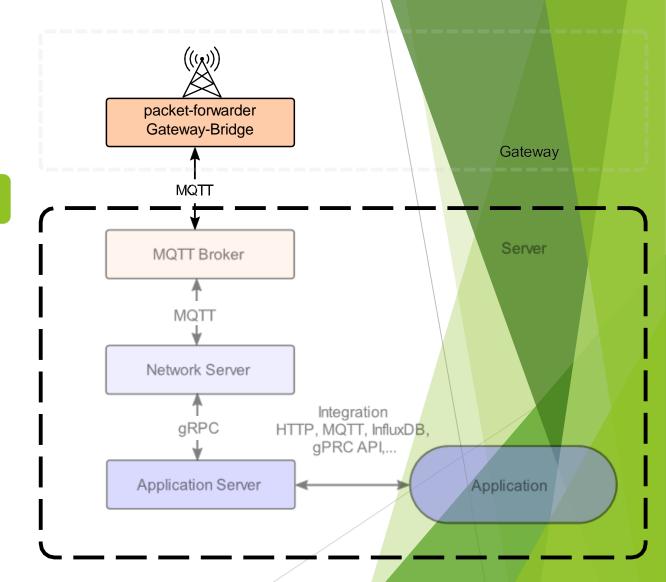
Interfaz web ChirpStack

## Packet-forwarder:

• Reenvía los paquetes LoRa al gateway-bridge

## **Gateway Bridge:**

- Convierte el protocolo del Packet Forwarder en JSON o formato de datos Protobuf
- Puede estar integrado en el gateway o en el servidor



## MQTT Broker:

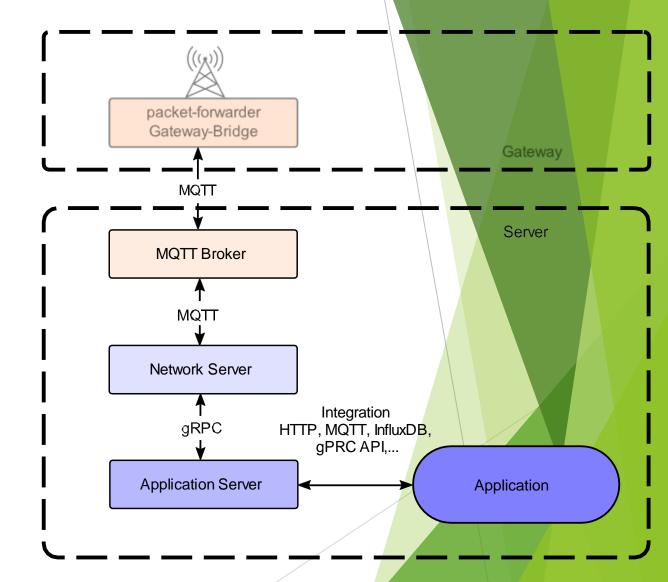
 Comunicación entre el gateway y en network server mediante MQTT (p.ej. Eclipse Mosquitto)

## **Network Server:**

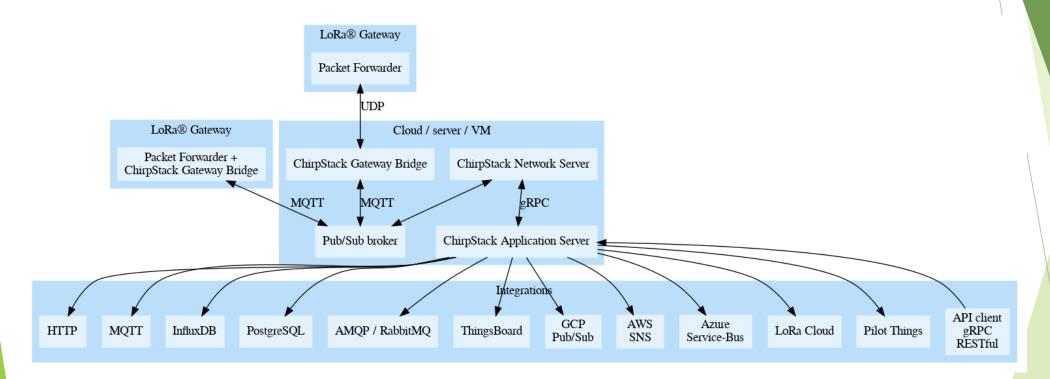
 Communication con el application server via gPRC

## **Application Server:**

 Varias implementaciones disponibles para conectar con aplicaciones



# Arquitectura de Chirpstack



# Instalación



#### ▶ <u>Dependencias auxiliares</u>:

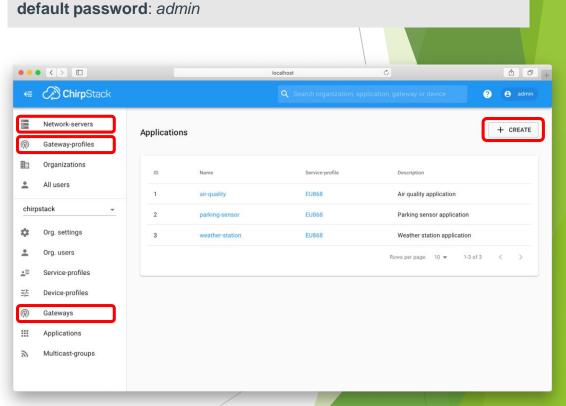
- MQTT broker: instalaremos mosquitto
- ► Redis: Una DB in-memory utilizada para almacenar datos temporales
- ► PostgreSQL: Base de datos para almacenamiento a largo plazo

## A INSTALAR

#### Conectar el gateway a ChirpStack:

• Para añadir el gateway a Chirpstack, debemos hacer login en el Application Server.

- Tres pasos necesarios:
  - Añadir el Network Server en la pestaña "Network-servers"
  - Añadir un Gateway-profile en la pestaña "Gateway-profiles"
  - Añadir el gateway en la pestaña "Gateways"



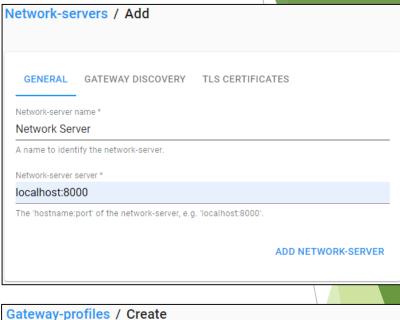
http://localhost:8080/

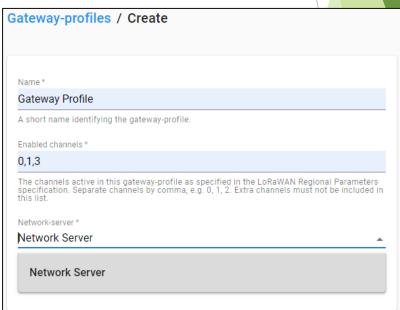
default username: admin

#### Conectar el gateway a ChirpStack:

- Añadir el Network Server:
  - Nombre del Network Server
  - Dirección:puerto del Network Server

- Añadir un Gateway Profile
  - Asignar un nombre al Gateway Profile
  - Habilitar canales de acuerdo a los LoRaWAN Regional Parameters
  - Asignar a un Network Server





#### Conectar el gateway a ChirpStack:

- Añadir el Gateway:
  - Gateway Name
  - Gateway Description
  - Gateway ID
  - Network Server
  - Gateway Profile



#### **Requisitos:**

Antes de conectar un dispositivo al sistema, debemos conocer la siguiente información:

- DevEUI: Identificador asignado por el fabricante
- Versión de LoRaWAN MAC implementada por el dispositivo
- Revisión de los Regional Parameters implementada por el dispositivo

También es necesario, según el método de activación, conocer los siguientes datos:

#### ABP:

- Device address
- Session Keys

#### OTAA:

Device root-keys

• Componentes de ChirpStack

• Conectar un dispositivo

• Podemos añadir nuevos dispositivos en su pestaña de la interfaz web del Application Server:

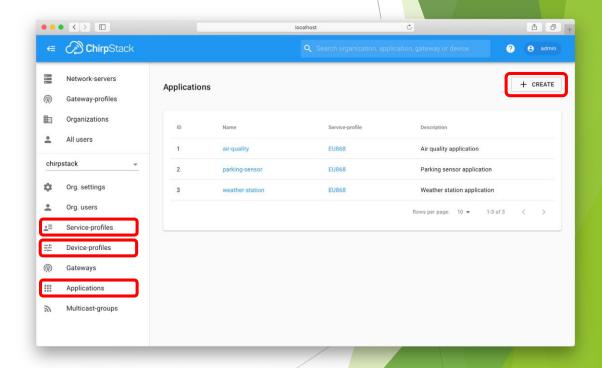
#### Pasos para añadir un dispositivo:

- Crear un perfil de servicio en la pestaña "Serviceprofiles"
- Crear un perfil de dispositivo en la pestaña "Device-profiles"
- Crear una aplicación en la pestaña "Applications"
- Añadir un dispositivo en la aplicación que hemos creado

http://localhost:8080/

default username: admin

default password: admin



15

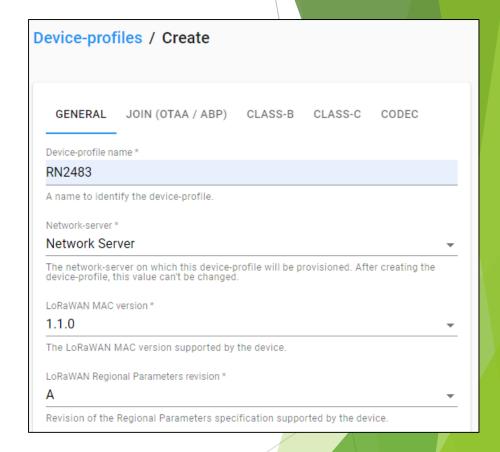
#### <u>Crear un Service Profile:</u>

• El Service Profile es el "contrato" entre un usuario y la red. Describe las características habilitadas para los usuarios del Service Profile y la cantidad de mensajes que pueden ser enviadas sobre la red.



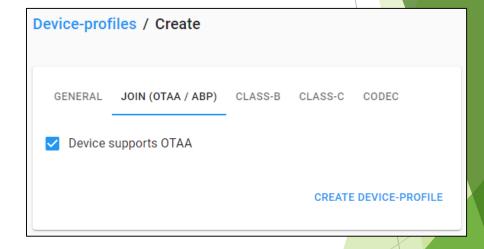
#### Crear un Device Profile:

 Un Device Profile define las capacidades y parámetros de inicio que necesita el Network Server para configurar el servicio de acceso radio LoRaWAN. Esta información debe ser proporcionada por el fabricante del dispositivo.



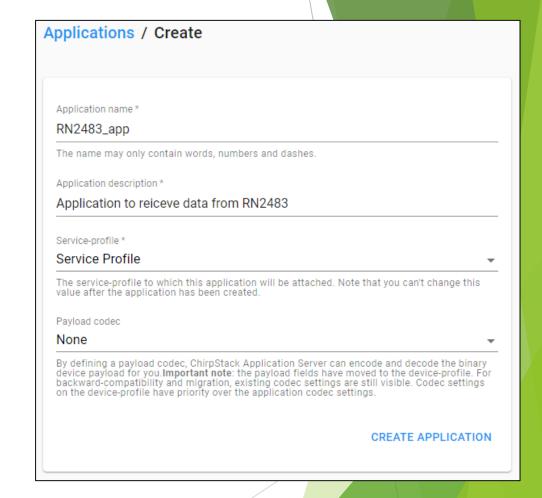
#### Crear un Device Profile:

- Definimos el método de activación en la pestaña "JOIN"
- Ajustes de las ventanas de downlink RX1 y RX2
- Lista de frecuencias de canales predefinidos de fábrica del dispositivo
- Ajustes de dispositivos Class-B (si lo soporta)
- Ajustes de dispositivos Class-C (si lo soporta)



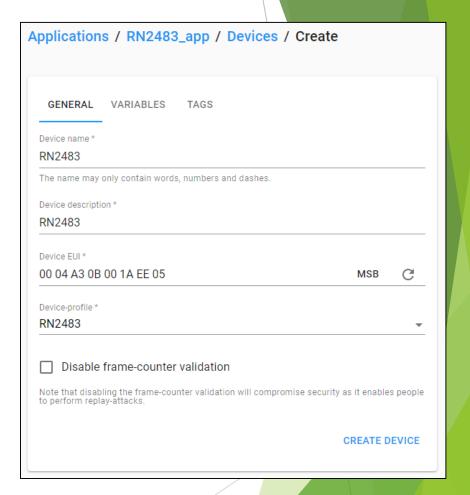
## Crear una Aplicación:

 Una aplicación es una colección de dispositivos con el mismo propósito o del mismo tipo



## Añadir un dispositivo:

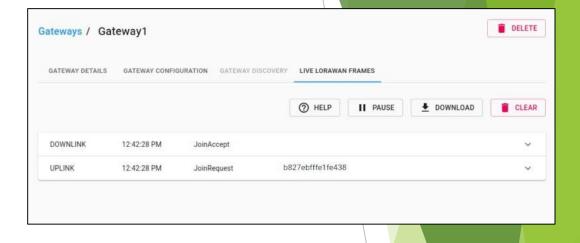
- Dentro de una aplicación podemos crear dispositivos
- Device EUI: Identificador del dispositivo por el fabricante
- Application Key: Proporcionada por el fabricante, aunque la podemos cambiar en el dispositivo

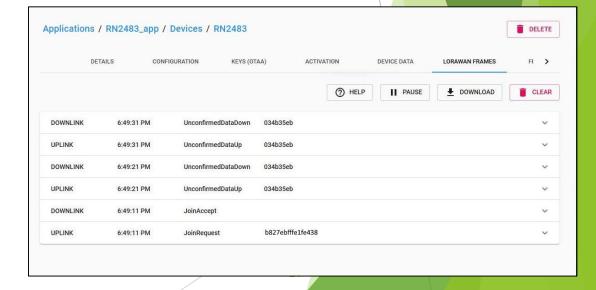


#### Flujo de mensajes:

• El flujo de los mensajes LoRa recibidos y transmitidos por el gateway puede consultarse en la sección "gateway" de la interfaz web

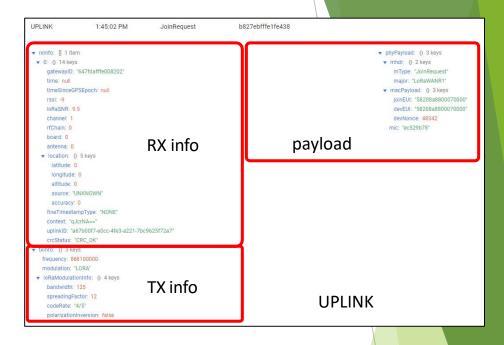
El flujo de los mensajes LoRaWAN puede consultarse en la sección de "application" de la interfaz web

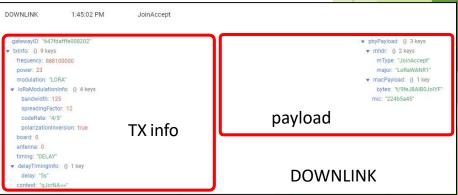




#### Flujo de mensajes:

- Un mensaje LoRaWAN se divide en 3 partes
- RX info: contiene información sobre la calidad de la señal recibida, como RSSI y SNR
- TX info: incluye información sobre los parámetros LoRa como el spreading factor, code rate, bandwidth y frecuencias
- Payload





22

#### Flujo de mensajes en un join por OTAA:

## Join Request

 Contiene el joinEUI, devEUI y devNonce para calcular las claves de sesión

## Join Accept

 Contiene datos cifrados como JoinNonce, NetID y DevAddr

DOWNLINK	6:49:31 PM	UnconfirmedDataDown	26011a87
UPLINK	6:49:31 PM	UnconfirmedDataUp	26011a87
DOWNLINK	6:49:21 PM	UnconfirmedDataDown	26011a87
UPLINK	6:49:21 PM	UnconfirmedDataUp	26011a87
DOWNLINK	6:49:11 PM	JoinAccept	
UPLINK	6:49:11 PM	JoinRequest	b827ebfffe1fe438

▼ phyPayload: {} 3 keys

▼ mhdr: {} 2 keys

mType: "JoinRequest" major: "LoRaWANR1"

▼ macPayload: {} 3 keys

devNonce: 48342

joinEUI: "58208a8800070000"

devEUI: "58208a8800070000"

mic: "ec529b78"

 ${\it Join Request:}$ 

JoinAccept:

▼ phyPayload: {} 3 keys

▼ mhdr: {} 2 keys

mType: "JoinAccept" major: "LoRaWANR1"

▼ macPayload: {} 1 key

bytes: "t/9feJ8AlB0JolYF"

mic: "224b5a45"

#### Flujo de mensajes después de un join:

En este ejemplo, se transmitieron varios uplinks sin confirmación tras el join

- A cada uplink le sigue un downlink con parámetros de red para el dispositivo
  - "RX ParamSetupReq": establece la frecuencia y el data rate de la segunda ventana de recepción
  - "RX TimingSetupReq": establece el delay entre el final de la recepción de un uplink y la apertura de la primera Ventana de recepción
- Parámetros de red adicionales como el duty cycle máximo o añadir nuevos canales
- Se puede consultar una lista de los comandos MAC en la LoRaWAN Alliance

DOWNLINK	6:49:31 PM	UnconfirmedDataDown	26011a87
UPLINK	6:49:31 PM	UnconfirmedDataUp	26011a87
DOWNLINK	6:49:21 PM	UnconfirmedDataDown	26011a87
UPLINK	6:49:21 PM	UnconfirmedDataUp	26011a87
DOWNLINK	6:49:11 PM	JoinAccept	
UPLINK	6:49:11 PM	JoinRequest	b827ebfffe1fe438

#### Uplink:

- ▼ phyPayload: {} 3 keys
- ▼ mhdr: {} 2 keys

mType: "UnconfirmedDataUp" maior: "LoRaWANR1"

- ▼ macPayload: {} 3 keys
- ▼ fhdr: {} 4 keys

devAddr: "26011a87"

▼ fCtrl: {} 5 keys

adr: true

adrAckReg: false

ack: false

fPending: false

classB: false

fCnt: 1

fOpts: null

fPort: 1

- ▼ frmPayload: [] 1 item
- ▼ 0: {} 1 key

bytes: "b2Y3IX9O/1/mhRlabQ=="

mic: "cee0d2e2"

#### Downlink:

- ▼ phyPayload: {} 3 keys
- ▼ mhdr: {} 2 keys

mType: "UnconfirmedDataDown" major: "LoRaWANR1"

- ▼ macPayload: {} 3 keys
- ▼ fhdr: {} 4 keys

devAddr: "26011a87"

▼ fCtrl: {} 5 keys

adr: true

adrAckReq: false

ack: false

fPending: false

classB: false fCnt: 1

- ▼ fOpts: [ 2 items
- ▼ 0: {} 2 keys

cid: "RX Param Setup Req"

- ▼ payload: () 2 keys
  - frequency: 868300000 dlSettings: "08"
- ▼ 1: {} 2 keys

cid: "RXTimingSetupReq"

▼ payload: () 1 key delay: 1

fPort: null frmPayload: null mic: "579fcb05"

#### Flujo de mensajes después de un join:

- Existen dos frame counters que llevan la cuenta de los frames de uplink y downlink:
  - FcntUp: Número de uplinks, transmitidos e incrementados por el dispositivo en el campo "fCnt"
  - FcntDown: Número de downlinks, transmitidos e incrementados por el network server en el campo "fCnt"
- Si el dispositivo o el network server reciben un mensaje con un frame counter inferior al último, el mensaje se ignora

DOWNLINK	6:49:31 PM	UnconfirmedDataDown	26011a87
UPLINK	6:49:31 PM	UnconfirmedDataUp	26011a87
DOWNLINK	6:49:21 PM	UnconfirmedDataDown	26011a87
UPLINK	6:49:21 PM	UnconfirmedDataUp	26011a87
DOWNLINK	6:49:11 PM	JoinAccept	
UPLINK	6:49:11 PM	JoinRequest	b827ebfffe1fe438

#### Uplink:

- ▼ phyPayload: {} 3 keys
- ▼ mhdr: {} 2 keys

mType: "UnconfirmedDataUp" major: "LoRaWANR1"

- ▼ macPayload: {} 3 keys
- v fhdr: {} 4 keys

devAddr: "26011a87"

v fCtrl: {} 5 keys

adr: true

adrAckReg: false

ack: false

fPending: false classB: false

fCnt: 1

fOpts: null

fPort: 1

- ▼ frmPayload: [] 1 item
- v 0: {} 1 key

bytes: "b2Y3IX9O/1/mhRlabQ=="

mic: "cee0d2e2"

#### Downlink:

- ▼ phyPayload: {} 3 keys
- ▼ mhdr: {} 2 keys

mType: "UnconfirmedDataDown" major: "LoRaWANR1"

- ▼ macPayload: {} 3 keys
- ▼ fhdr: {} 4 keys

devAddr: "26011a87"

▼ fCtrl: {} 5 keys

adr: true

adrAckReg: false

ack: false

fPending: false

classR: false

#### fCnt: 1

- ▼ fOpts: 2 items
- ▼ 0: {} 2 keys

cid: "RX ParamSetupReq"

▼ payload: {} 2 keys

frequency: 868300000 dlSettings; "08"

▼ 1: {} 2 keys

cid: "RX TimingSetupReg"

▼ payload: () 1 key delay: 1

fPort: null

frmPayload: null mic: "579fcb05"

UPLINK	7:01:25 PM	UnconfirmedDataUp	26011a87
DOWNLINK	7:01:13 PM	UnconfirmedDataDown	26011a87
UPLINK	7:01:13 PM	UnconfirmedDataUp	26011a87

#### Flujo de mensajes ADR:

Si el flag ADR está a 1, el network server puede enviar parámetros como el spreading factor, potencia y canales utilizables por el dispositivo.

#### **Uplink:**

- El bit ADR es uno -> ADR está activo
- Spreading factor 12

```
    ▼ txInfo: {} 3 keys
        frequency: 868300000
        modulation: "LORA"

            ▼ loRaModulationInfo: {} 4 keys
            ■ bandwidth: 125
            ■ spreadingFactor: 12
            ■ codeRate: "4/5"
            ■ polarizationInversion: false
```

```
▼ phyPayload: {} 3 keys
  ▼ mhdr: {} 2 keys
     mType: "UnconfirmedDataUp"
     major: "LoRaWANR1"
  ▼ macPayload: {} 3 keys

▼ fhdr: {} 4 keys
        devAddr: "26011a87"

▼ fCtrl: {} 5 kevs
         adr: true
         adrAckReg: false
         ack: false
         fPending: false
         classB: false
        fCnt: 3
       fOpts: null
      fPort: 6
   ▼ frmPayload: [] 1 item
     ▼ 0: {} 1 key
         bytes: "et4="
    mic: "b4198459"
```

#### Flujo de mensajes ADR:

Un downlink sigue con un comando "LinkADRReq" para activar nuevos parámetros en el dispositivo:

dataRate: 5

txPower: 5

 chMask: canales habilitados; 0, 1, y 2 son canales obligatorios

UPLINK	7:01:25 PM	UnconfirmedDataUp	26011a87	
DOWNLINK	7:01:13 PM	UnconfirmedDataDown	26011a87	
UPLINK	7:01:13 PM	UnconfirmedDataUp	26011a87	

```
▼ phyPayload: {} 3 keys
 ▼ mhdr: {} 2 keys
     mType: "UnconfirmedDataDown"
     major: "LoRaWANR1"
 ▼ macPayload: {} 3 keys
   ▼ fhdr: {} 4 keys
       devAddr: "26011a87"
     v fCtrl: {} 5 keys
        adr: true
        adrAckReq: false
        ack: true
        fPending: false
        classB: false
       fCnt: 3
     ▼ fOpts: [] 1 item
      ▼ 0: {} 2 keys
          cid: "LinkADRReg"
        ▼ payload: {} 4 keys
            dataRate: 5
            txPower: 5
          v chMask: ∏ 16 items
             0: true
             1: true
             2: true
             3: false
             4: false
             5: false
             7: false
             8: false
              10: false
             11: false
              12: false
             13: false
             14: false
15: false
          ▼ redundancy: {} 2 keys
             chMaskCntl: 0
```

#### Message Flow ADR:

- Next uplink message adapted the parameters from the ADR request
- "LinkADRAns" includes acknowledgments of the changed power, data rate, and channel mask.
- The network server changes transmission parameters until suitable parameters in terms of connectivity and battery efficiency is achieved

UPLINK	7:01:25 PM	UnconfirmedDataUp	26011a87
DOWNLINK	7:01:13 PM	UnconfirmedDataDown	26011a87
UPLINK	7:01:13 PM	UnconfirmedDataUp	26011a87

▼ txInfo: {} 3 keys
 frequency: 868300000
 modulation: "LORA"
 ▼ loRaModulationInfo: {} 4 keys
 bandwidth: 125
 spreadingFactor: 7
 codeRate: "4/5"
 polarizationInversion: false

```
▼ phyPayload: {} 3 keys
 ▼ mhdr: {} 2 keys
     mType: "UnconfirmedDataUp"
     major: "LoRaWANR1"
 ▼ macPayload: {} 3 keys

▼ fhdr: {} 4 keys
       devAddr: "26011a87"
     ▼ fCtrl: {} 5 keys
         adr: true
         adrAckReg: false
         ack: false
         fPending: false
         classB: false
       fCnt: 2
     ▼ fOpts: [] 1 item
       ▼ 0: {} 2 keys
           cid: "LinkADRAns"
         ▼ payload: {} 3 keys
             channelMaskAck: true
             dataRateAck: true
             powerAck: true
     fPort: 4
   ▼ frmPayload: [] 1 item
     ▼ 0: {} 1 key
         bytes: "p3+VVj+u"
```

mic: "c18eb467"

#### Message Flow:

• Si un dispositivo se estabiliza, tras varias respuestas a un uplink con parámetros y ajustes de transmisión ya no siguen downlinks.

UPLINK	6:15:16 PM
UPLINK	6:14:02 PM
UPLINK	6:12:48 PM
DOWNLINK	6:11:35 PM
UPLINK	6:11:35 PM
DOWNLINK	6:10:22 PM
UPLINK	6:10:22 PM
DOWNLINK	6:10:08 PM
UPLINK	6:10:08 PM
DOWNLINK	6:10:02 PM
UPLINK	6:10:02 PM