ThingPark Partner

ThingPark Wireless ™
Advanced Developer Guide

V4 - June 2017

Under NDA

NOTICE

This document contains proprietary and confidential material of ACTILITY SA. This document is provided under and governed by either a license or confidentiality agreement. Any unauthorized reproduction, use, or disclosure of this material, or any part thereof, is strictly prohibited.

The material provided in this document is believed to be accurate and reliable. However, no responsibility is assumed by Actility SA for the use of this material. Actility SA reserves the right to make changes to the material at any time and without notice. This document is intended for information and operational purposes only. No part of this document shall constitute any contractual commitment by Actility SA.

Headquarters

Actility Lannion,

Actility S.A 4 rue Ampère BP 30225

22300 Lannion France

www.actility.com

Table of Contents

Definitions and Acronyms

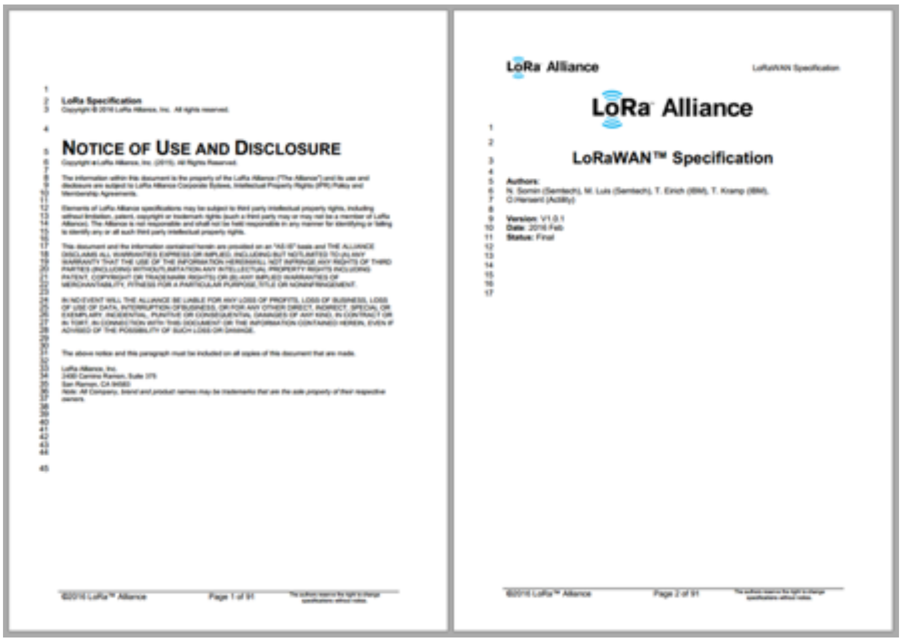
ABP	Activation By Personalization
ADR	Adaptive Data Rate
AES	Advanced Encryption Standard
AS	Application Server
ESP	Estimated Signal Power
LC	Logical Channel
LRC	Long-Range Controller: Network Server
LRR	Long-Range Relay: software inside the gateway
OTAA	Over The Air Activation
RF	Radio Frequency
RSSI	Received Signal Strength Indicator
SF	Spreading Factor
SNR	Signal to Noise Ratio

1 Scope

The Scope of this Device Developer Guide is to provide the guidelines to a developer during the Thingpark Wireless connectivity integration.

2 LoRaWAN specification

The LoRaWAN specification is publicly available from the LoRa™ alliance web site: lora-alliance.org



LoRa Specification

Notice of Use and Disclosure

[illegible]

©2016 LoRa™ Alliance Page 1 of 91

LoRa Alliance

LoRaWAN™ Specification

Authors:
N. Sornin (Semtech), M. Luis (Semtech), T. Eirich (IBM), T. Kramp (IBM), O.Hersent (Actility)

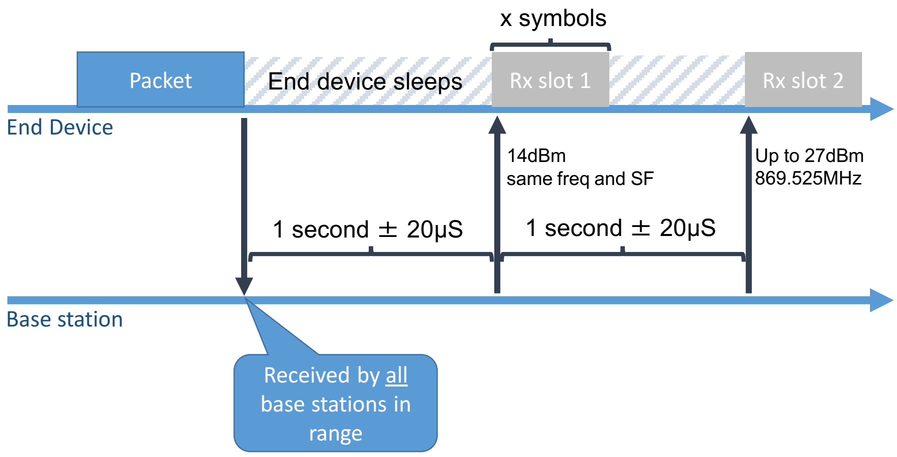
Version: V1.0.1
Date: 2016 Feb
Status: Final

©2016 LoRa™ Alliance Page 2 of 91

The LoRaWAN specification enables bidirectional communications, and currently offers three variants of the MAC layer: class A, optimized for battery usage and sensors; class B, optimized for battery powered actuators with low command latency requirements, and class C for mains powered devices including actuators.

Class name	Intended usage
A (« **all** »)	**Battery powered sensors**, or actuators with no latency constraint Most energy efficient communication class. Must be supported by all devices
B (« **beacon** »)	**Battery powered actuators** Energy efficient communication class for latency controlled downlink. Based on slotted communication synchronized with a network beacon.
C (« **continuous** »)	**Mains powered actuators** Devices which can afford to listen continuously. No latency for downlink communication.

LoRaWAN class A employs the well-known receiver initiated transmit strategy to enable communication bidirectionality.



The End-device, e.g. a temperature sensor, wakes-up to transmit its measurement. The LoRaWAN radio frame will be received by all nearby base stations of the RF network. The device then immediately goes to sleep for a specified amount of time, by default 1 second, in order to preserve the battery.

After the exact sleep time, the End-device must wake up to receive potential downlink communication from the network. The downlink communication may be an ACK from the network if the End-device had sent a confirmed LoRaWAN frame, or it may be a command from the MAC layer network controller or from an application server.

The Core network will use, at its choice, this first receive window (RX1), or the second receive window (RX2) do send downlink frames. So if no frame has been received during the RX1 slot, the device must go to sleep again and wake up another time for the RX2 slot.

2.1 Globally unique EUI-64: DevEUI

Each LoRaWAN end device has a **globally unique** IEEE EUI-64 address, the DevEUI. These addresses are allocated by manufacturers within address blocks that must be purchased from IEEE, three blocks are available:

- Organizational Unique Identifier (OUI) / MAC Address Block Large (MA-L)
- MAC Address Block Medium (MA-M)
- MAC Address Block Small (MA-S)

These 3 different blocks of addresses can be purchased from IEEE here:

- http://standards.ieee.org/develop/regauth/oui/
- http://standards.ieee.org/develop/regauth/oui28/index.html
- http://standards.ieee.org/develop/regauth/oui36/index.html

2.2 *Over-The-Air Activation (OTAA)*

OTAA device derive its NwkSkey and AppSkey using the Join key negotiation procedure as they first attach to a network. This procedure uses a master AppKey secret that must be personalized at production in the device.

A single application server, identified by its AppEUI (JoinEUI), is supported per device.

OTAA	Who?	What is it?
DevEUI	**IEEE**	The DevEUI identifies the device on the LoRaWAN network during the JOIN request
AppEUI (joinEUI)	**Operator**	The AppEUI identifies the join server during the JOIN request
AppKey	**Device manufacturer**	The AppKey encrypts the data during the JOIN request

2.2.1 128 bit application key: AppKey

The 128 bit AppKey must be personalized in each device during production. It may be distinct per device or unique per application depending on the use-case.

Typically the AppKey must be part of the Excel file delivered with each production batch to the customer, so that the customer may be able to provision the AppKey to the application server associated with the Device.

2.2.2 64 bit application server identifier: AppEUI (joinEUI)

The AppEUI is a globally unique identifier of the target application server that will process all exchanges with the device.

The network forwards the join message to the application server identified by the AppEUI. This application server is supposed to have been provisioned with the Device AppKey. Based on the AppKey and the content of the Join message sent by the device, the Application server:

1. Generates a NwkSkey and AppSkey and sends the NwkSkey information to the Core Network
2. Forms a cryptographic Join response payload that will allow the device to compute a NwkSkey and AppSkey

As part of the Join procedure, the network also allocates a DevAddr address to the LoRaWAN device.

An available joinEUI is available from Actility as a join server: F0:3D:29:AC:71:00:00:00.

2.3 *Activation By Personalization (ABP)*

ABP	Who?	What is it?
DevEUI	**IEEE**	DevEUI is not used in LoRa communication in ABP but is used to identify the device at the Network Server side
DevAddr	**Operator**	The DevAddr is the Device Address on the LoRa Network
NwkSKey	**Device manufacturer**	The NwkSKey encrypts the data during the transmission. Gateways from other networks cannot see the content of messages. The NwkSKey authenticates the device on a LoRa network
AppSKeys	**Device manufacturer**	The AppSKey encrypts the payload data

2.3.1 Device address: DevAddr

The DevAddr identifies the device on the network, together with the Network secret for the sensor. **The group (DevAddr, NwkSkey) must be globally unique**.

If the end device is not using the JOIN LoRaWAN procedure, it must also be personalized with the DevAddr.

On the Core network, the DevAddr must be formed as in

NwkID [31..25]	NwkAddr (24..00]
These identifiers are preallocated awaiting NwkID management by LoRa alliance. *The NwkID identifies the operator.*	Ask your operator for a block of NwkAddr for the devices that you will produce for the target network. *Device manufacturers must ensure that each DevAddr=NwkID+NwkAddr corresponds to very few, ideally a single DevEUI.* Collisions may be resolved as each device has a unique NwkSkey, however such collisions should be avoided.

The DevAddr will be provided by your LoRaWAN network operator.

2.3.2 128 bit network secret: NwkSkey

The 128 bit NwkSkey is used by the Core network to verify the authenticity and integrity of each message. Use a **random NwkSkey for each device**.

⚠ Allocating a **random network secret per device** is very important for security, but also to ensure that the short address collision resolution algorithm will work appropriately. The pair (DevAddr, NwkSkey) must be globally unique.

2.3.2.1 128 bit application secret: AppSkey

The 128 bit AppSkey is used to encrypt the payload of messages. You may decide to use a unique AppSkey for all LoRaWAN ports used by your device, or to allocate one AppSkey for each port.

AppSkeys must be known to the Application Server. Typically AppSkeys are part of a production Excel file providing the associations between DevEUI, DevAddr, NwkSkey, AppSkey(s) of the devices part of the production batch.

When adding the device to your account using the Device Manager application:

- You do not need to provision the AppSkey. ThingPark Wireless will then forward the payload in encrypted form to the application servers and has no access to the payload clear-form content. Network tools like Wireless Logger will be unable to decode the payload and will present it in encrypted form.
- If you provision the AppSkey(s), then the Core Network will decode the payload before forwarding it to the application server(s), and the Wireless Logger tool will also present the packet payload in decoded form (and parse this payload if the payload syntax is supported).

2.4 Channel plans

2.4.1 ETSI EU 868

The implementation in Europe is the following default and mandatory channel plan:

- LC1: 868.1 MHz
- LC2: 868.3 MHz
- LC3: 868.5 MHz
- RX2: 869.525 MHz / SF12

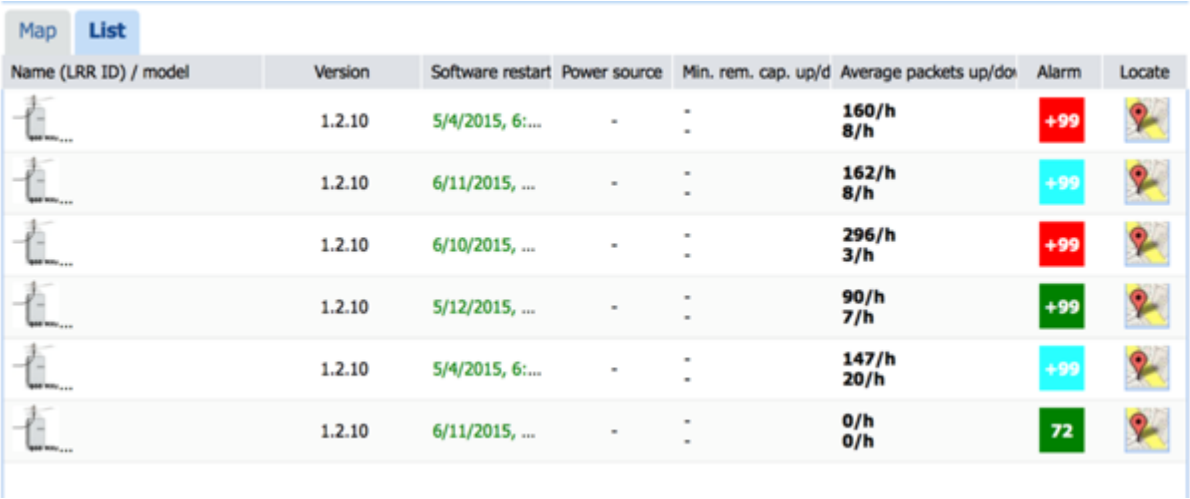
Then the network will configure the device with the operator settings (add new channels, change RX2 configuration).

3 NetworkManager

- Launch the NetworkManager from your Dashboard menu.

3.1 *Base stations list*

The Base stations list displays all the filtered base stations into a list.



Name (LRR ID) / model	Version	Software restart	Power source	Min. rem. cap. up/d	Average packets up/do	Alarm	Locate
...	1.2.10	5/4/2015, 6:...	-	- -	160/h 8/h	+99	
...	1.2.10	6/11/2015, ...	-	- -	162/h 8/h	+99	
...	1.2.10	6/10/2015, ...	-	- -	296/h 3/h	+99	
...	1.2.10	5/12/2015, ...	-	- -	90/h 7/h	+99	
...	1.2.10	5/4/2015, 6:...	-	- -	147/h 20/h	+99	
...	1.2.10	6/11/2015, ...	-	- -	0/h 0/h	72	

In the Search section, a new check-box “Restrict search to visible map area” is available, in order to filter only base stations on the previous map view.

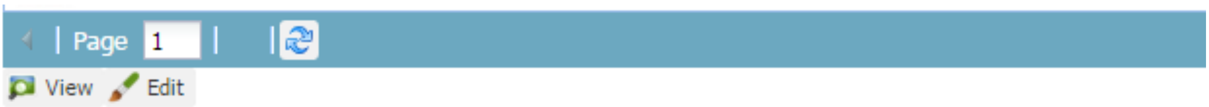
The displayed field are:

- Name (LRR ID) / model: name with ID and model of the LRR
- Version: version of the LRR software installed on the base station
- Software restart: last date of software restart
- Power source: type of power source when available
- Min. rem. Cap. up/down: min. remaining capacity up/down depending on potential Duty Cycle limitations
- Average packets up/down: average number of packets per hour
- Alarm: number of alarms not acknowledged
- Locate: open a pop-up and display the base station on a map

The filtering/sorting can be done on all the above mentioned properties. This way, one can easily display all base stations for which e.g. the average packet level is high, that have raised critical alarms, etc.

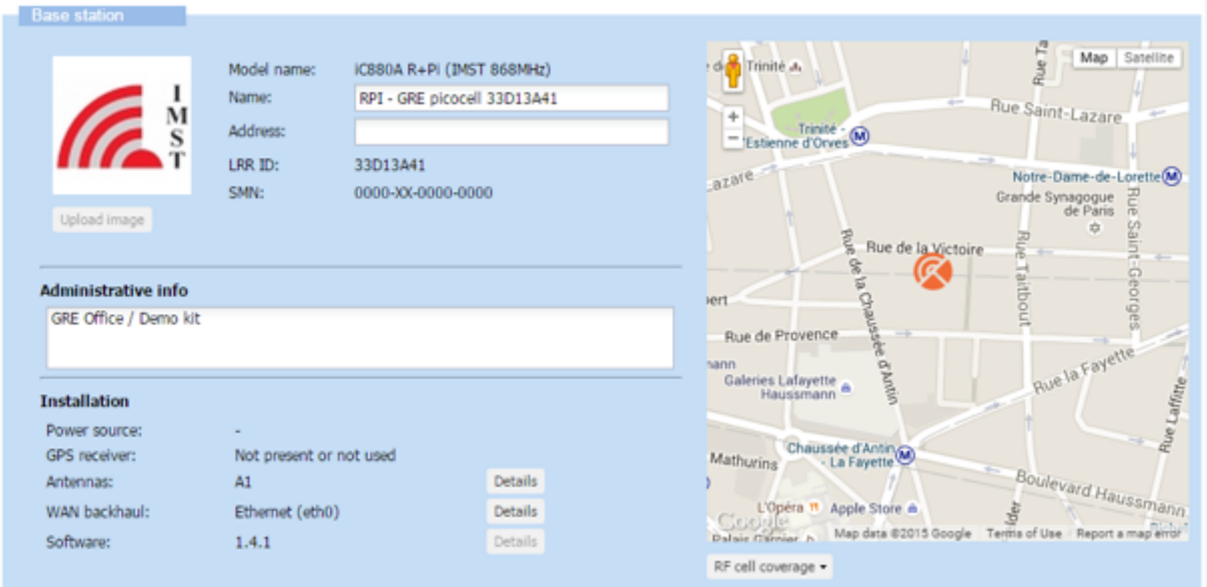
In order to View or Edit the device:

- View
  - double click on a base station
  - select a base station and click on **View** at the bottom of the list
- Edit
  - Select a base station and click on **Edit** at the bottom of the list



3.2 *Base station details*

- Base station frame



Base station

Model name: iC880A R+Pi (IMST 868MHz)

Name: RPI - GRE picocell 33D13A41

Address:

LRR ID: 33D13A41

SMN: 0000-XX-0000-0000

Administrative info

GRE Office / Demo kit

Installation

Power source:	-	
GPS receiver:	Not present or not used	
Antennas:	A1	Details
WAN backhaul:	Ethernet (eth0)	Details
Software:	1.4.1	Details

This frame displays the basic information of the base station such as the model, name, LRR ID, SMN (Serial Number).

Based on the LRR Base station Model and the available information at system level, the ThingPark Wireless OSS will also present the Installation information such as Power source, GPS receiver, Antennas, WAN backhaul, Software version.

The location of the base station on the map could be provided in 2 different ways:

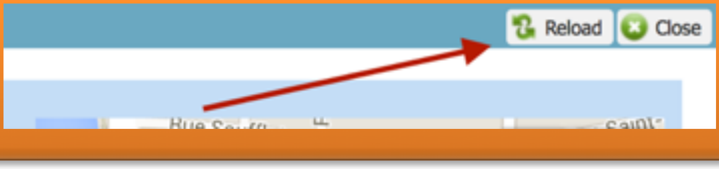
- ○ Location provided by the GPS receiver inside the base station
- ○ Manual location set by the administrator in the Network Manager GUI (Edit the LRR, Click on “Update Location” button, Select Manual, enter GPS location then Save).

Location provided by the GPS receiver are used by Actility tools such as Wlogger or Network without further action.

Please contact Actility to use the tools when location is set manually.

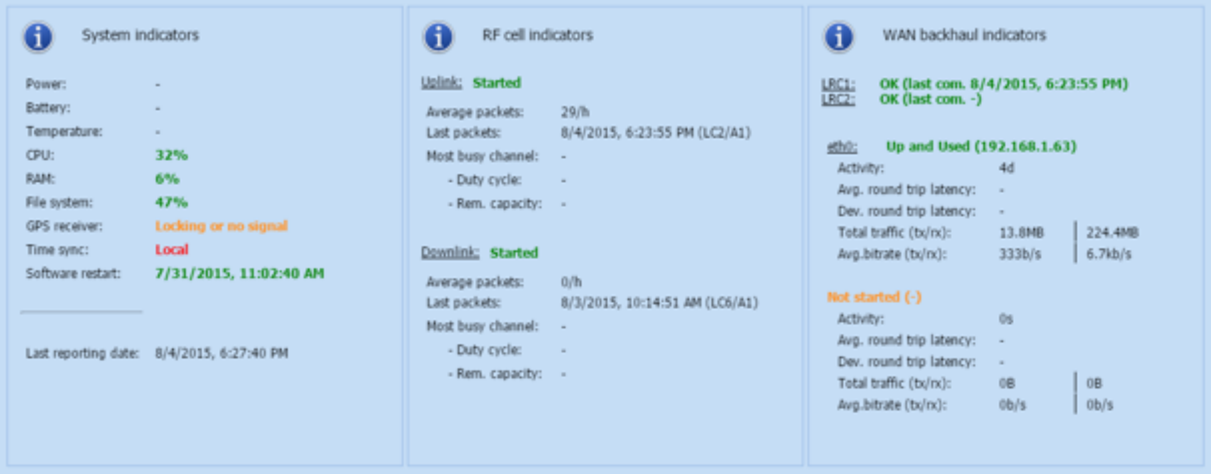
NOTE

You can reload all indicators by clicking on the reload button in the right corner



- Indicators frame

A summary view of indicators for System, RF Cell indicators and WAN backhaul indicators is available for the LRR as shown below:



System indicators

Power:	-
Battery:	-
Temperature:	-
CPU:	32%
RAM:	6%
File system:	47%
GPS receiver:	Locking or no signal
Time sync:	Local
Software restart:	7/31/2015, 11:02:40 AM
Last reporting date:	8/4/2015, 6:27:40 PM

RF cell indicators

Uplink: Started

Average packets:	29/h
Last packets:	8/4/2015, 6:23:55 PM (LC2/A1)
Most busy channel:	-
- Duty cycle:	-
- Rem. capacity:	-

Downlink: Started

Average packets:	0/h
Last packets:	8/3/2015, 10:14:51 AM (LC6/A1)
Most busy channel:	-
- Duty cycle:	-
- Rem. capacity:	-

WAN backhaul indicators

LRC1: OK (last com. 8/4/2015, 6:23:55 PM)
LRC2: OK (last com. -)

eth0: Up and Used (192.168.1.63)

Activity:	4d	
Avg. round trip latency:	-	
Dev. round trip latency:	-	
Total traffic (tx/rx):	13.8MB	224.4MB
Avg.bitrate (tx/rx):	333b/s	6.7kb/s

Not started (-)

Activity:	0s	
Avg. round trip latency:	-	
Dev. round trip latency:	-	
Total traffic (tx/rx):	0B	0B
Avg.bitrate (tx/rx):	0b/s	0b/s

- System indicators frame

The availability of System indicators depends on the model name of LRR (Example: not all LRR gateways support embedded GPS). Available system indicators include the following:

- Power type
- Battery option
- Temperature
- CPU
- RAM
- Mass Storage Usage
- GPS receiver
- Time sync
- Software restart date

- RF Cell Indicators frame

RF Cell indicators include Uplink and Downlink traffic indicators on the RF interface. Available indicators include the following:

- Uplink: message from the device to the network
- Downlink: message from the network to the device
- Average packets: number of packets in the last 1h window
- Last packets: date time of the last packet
- Most busy channel
- Duty cycle: utilization of the duty cycle on the most busy channel
- Rem. capacity: remaining capacity of the duty cycle

A provisioned LRR comes with pre-provisioned logical channels (LCs). The summary section provides Network Managers with information on the most busy uplink and downlink logical channels based in Duty Cycle data.

- WAN indicators frame

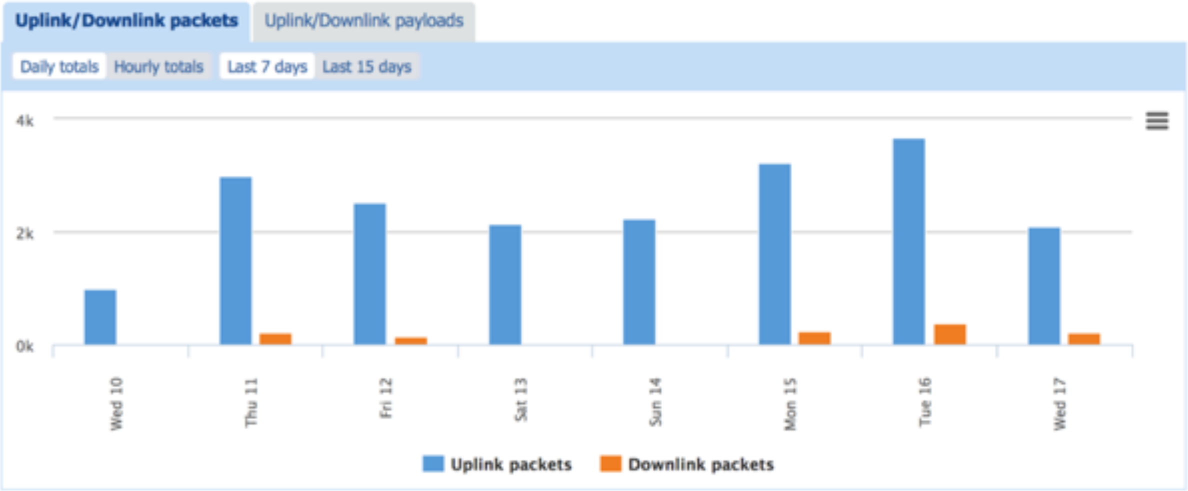
The supported LRRs may have multiple backhauling options, it could be Ethernet or Cellular or others. Each interfaces can be monitored and managed from the interface.

Also the connection status to the LRC including the date of the last messages transferred to the LRC.

- LRC1 / LRC2:
- Network interface (eth, ppp, wlan, …)
  - Activity
  - Avg. round trip latency
  - Dev. round trip latency
  - Total traffic
  - Avg. bitrate

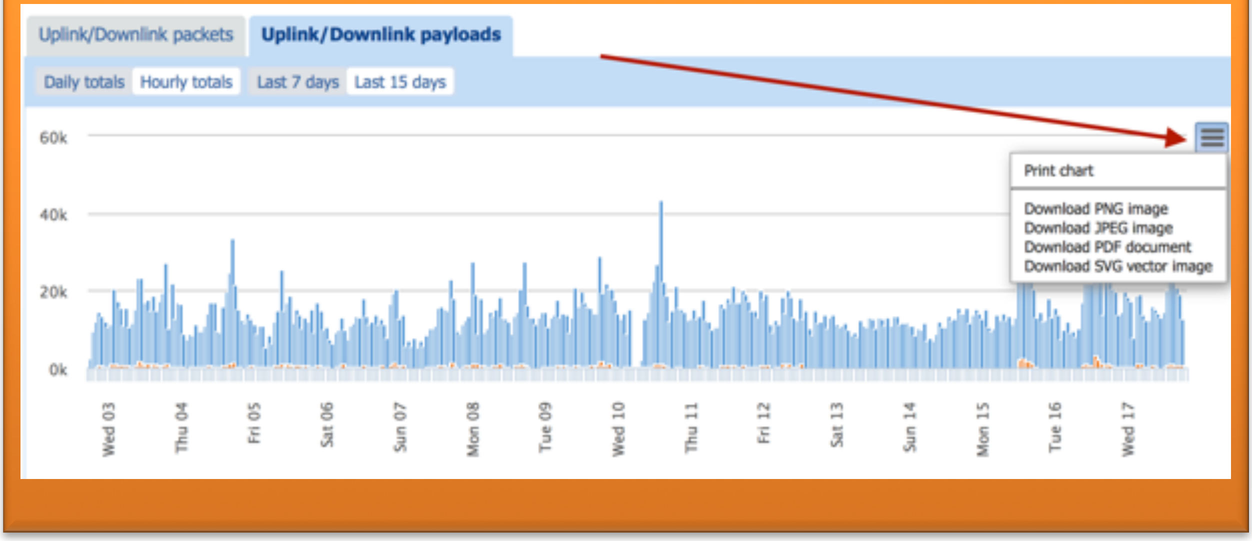
- Uplink/Downlink packets frame

The uplink/downlink packets and payload frame presents the number of packets or payload size transmitted uplink and downlink by an LRR with multiple sub-total options (Daily, Hourly, Last 7 days, Last 15 days)



NOTE

All charts can be exported or printed by clicking on the top right menu icon in the chart as shown below:



3.3 Base station provisioning

The base station provisioning allows the Network Partner to add a new base station to its network.

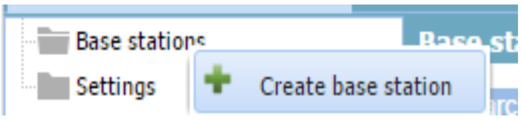
Information required to create a new base station:

- ID
- Serial Number (SMN)
- Model

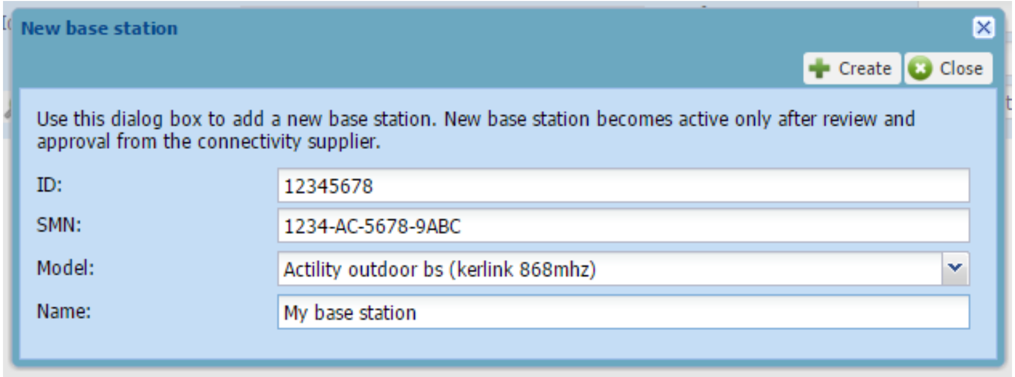
3.3.1 Manual Base station creation

To create a new base station:

- Right click on **Base stations**



- Click on **Create base station**
- A new pop-up appears with 4 fields: ID, SMN, Model, Name



Use this dialog box to add a new base station. New base station becomes active only after review and approval from the connectivity supplier.

ID: 12345678

SMN: 1234-AC-5678-9ABC

Model: Actility outdoor bs (kerlink 868mhz)

Name: My base station

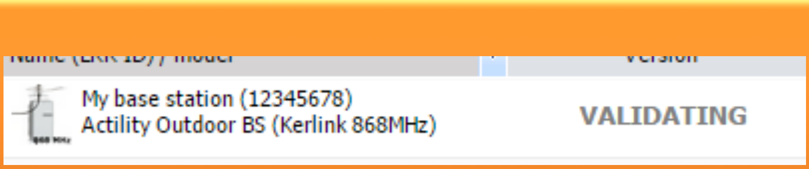
- Fill out the fields. Please note that all fields are mandatory
  - ID: Identification of the base station(8 hexadecimal digits)
  - SMN: Serial Number (format: 0000-XX-0000-0000)

The fields ID and SMN are written on the label sticked on the base station, if you do not have it, enter 0000-XX-0000-0000

  - Select the right model
  - Enter a name

- Click on Create to add the base station.
- To view the base station in the list, click on Search in the search section.

NOTE



Name (LRR ID) / model	Version
My base station (12345678) Actility Outdoor BS (Kerlink 868MHz)	VALIDATING

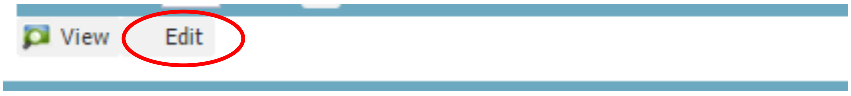
Once a new base station has been created, it stays in status **VALIDATING** until your **Operator Manager** has validated the base station on its network.

3.3.2 Modify a Base station

Modifying a base station allows you to update base station-related data such as the name, the address, the manually entered location or the administrative info.

Start by opening a base station in Edit view:

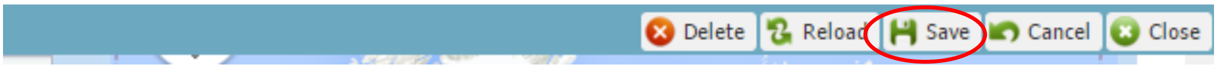
- Select the base station to edit in the list
- Click on **Edit** to enter in the Edit view



Modification allowed:

- Base station name
- Address
- Administrative info
- Update location / altitude
- Alarm settings

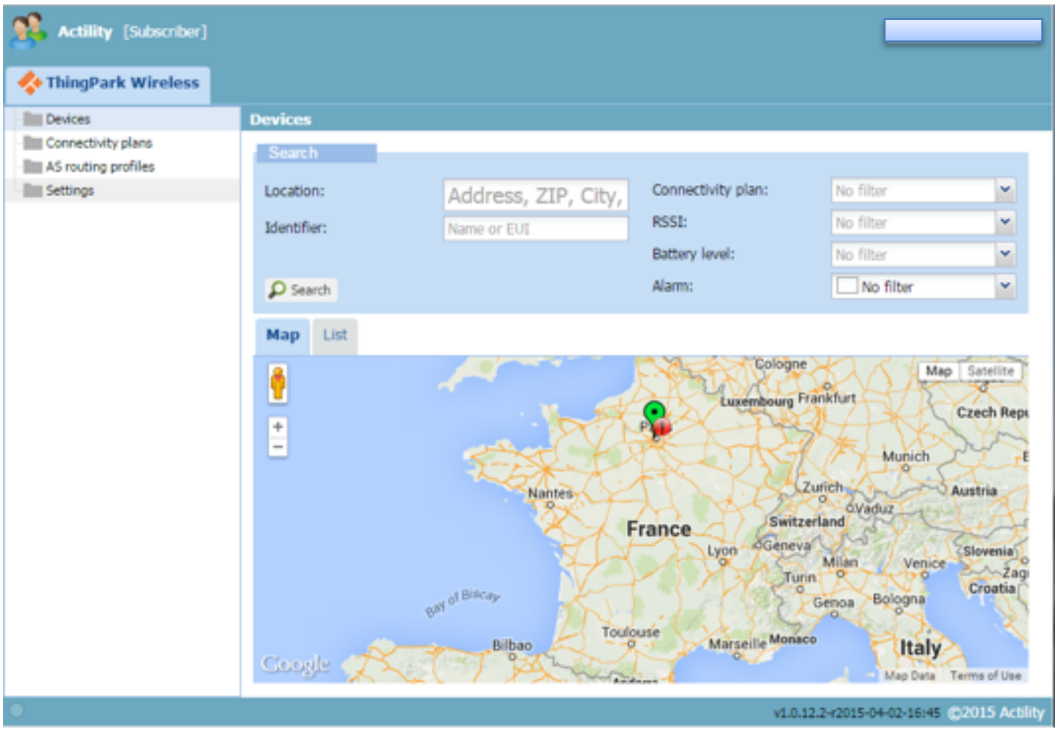
Finally, to confirm the changes, go back to the device details by selecting the device in the column in the left sidebar, then click **Save** in the top-right corner of the screen.



4 DeviceManager

- Launch the DeviceManager from your Dashboard menu.

When you login to the Device Management portal, you get an overview of all the devices of the account is listed. You can easily shift between the **Map** and **List** view of devices, by clicking the corresponding tabs.



Location: Address, ZIP, City,

Identifier: Name or EUI

Connectivity plan: No filter

RSSI: No filter

Battery level: No filter

Alarm: □ No filter

The interface is based on 2 frames, a left sidebar menu and a main application frame showing device data.

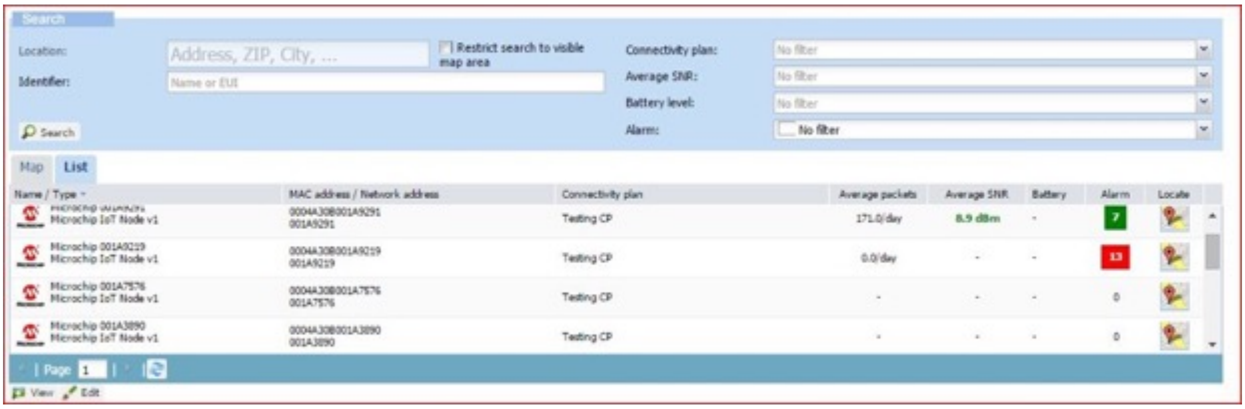
The left sidebar menu gives access to Devices, Connectivity plans, AS routing profiles and Settings.

The first main frame contains a Search bar, allowing users to search devices by Location, device Identifier or another filtering criteria.

4.1 Devices Creation and Management

4.1.1 Device list

Device List displays all the filtered devices in a list.



Search

Location: Address, ZIP, City, ...

□ Restrict search to visible map area

Identifier: Name or EUI

Connectivity plan: No filter

Average SNR: No filter

Battery level: No filter

Alarm: No filter

Name / Type	MAC address / Network address	Connectivity plan	Average packets	Average SNR	Battery	Alarm	Locate
Microchip 001A9291 Microchip IoT Node v1	0004A30B001A9291 001A9291	Testing CP	171.0/day	8.9 dBm	-	7	
Microchip 001A9219 Microchip IoT Node v1	0004A30B001A9219 001A9219	Testing CP	0.0/day	-	-	13	
Microchip 001A7576 Microchip IoT Node v1	0004A30B001A7576 001A7576	Testing CP	-	-	-	0	
Microchip 001A3890 Microchip IoT Node v1	0004A30B001A3890 001A3890	Testing CP	-	-	-	0	

The displayed field are:

- Name / Type: name and device profile
- MAC address / Network address: DevEUI and DevAddr of the device
- Connectivity plan
- Average packets
- Average SNR: based on the last 5 packets received
- Battery
- Alarm: number of alarms not acknowledged
- Locate: open a pop-up and display the device on a map

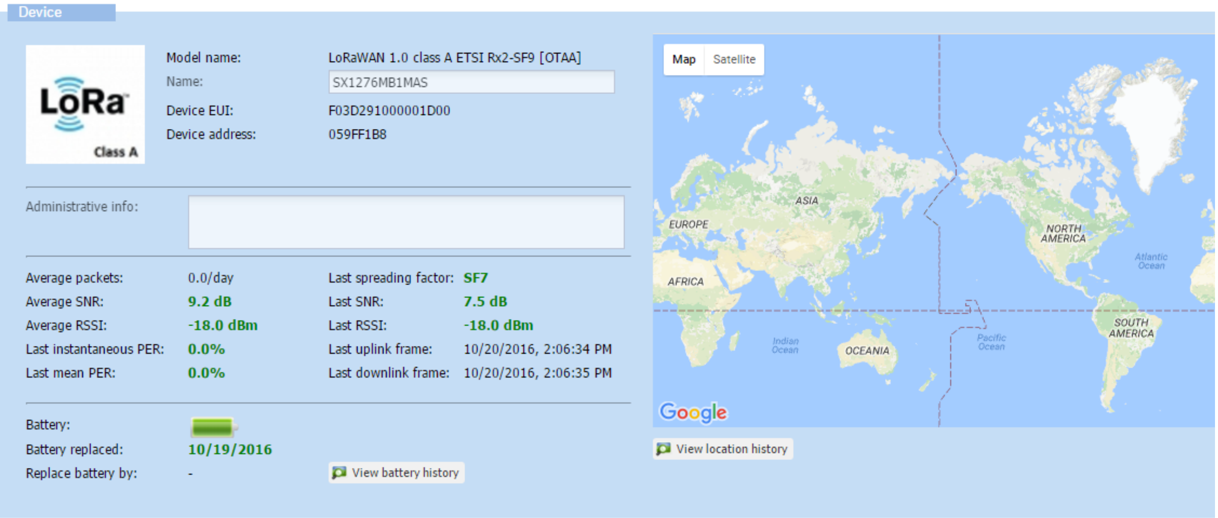
The filtering/sorting can be done on all the above mentioned properties. Users can therefore easily display all devices for which e.g. the battery level is low, that have raised critical alarms, etc.

In order to View or Edit the device:

- View
  - double click on a device
  - select a device and click on **View** at the bottom of the list
- Edit
  - Select a device and click on **Edit** at the bottom of the list

4.1.2 Device details

- Device frame



Device

Model name:	LoRaWAN 1.0 class A ETSI Rx2-SF9 [OTAA]
Name:	SX1276MB1MAS
Device EUI:	F03D291000001D00
Device address:	059FF1B8

Administrative info:

Average packets:	0.0/day	Last spreading factor:	SF7
Average SNR:	9.2 dB	Last SNR:	7.5 dB
Average RSSI:	-18.0 dBm	Last RSSI:	-18.0 dBm
Last instantaneous PER:	0.0%	Last uplink frame:	10/20/2016, 2:06:34 PM
Last mean PER:	0.0%	Last downlink frame:	10/20/2016, 2:06:35 PM

Battery:	
Battery replaced:	10/19/2016
Replace battery by:	-

This frame displays the basic information of the device such as the model (device profile), name, DevEUI, devAddr.

The interface also provides information on the LoRaWAN traffic such as the average number of packets, average RSSI and SNR, last instantaneous/mean PERs, last RSSI/SNR/SF, date/time of the last message received/sent.

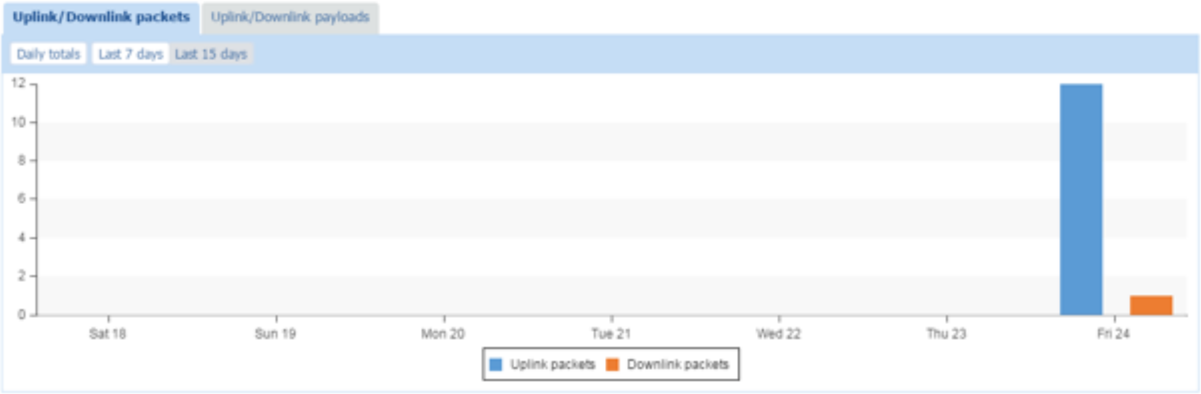
For devices supporting the feature, the battery status information can also be displayed.

The location of the device on the map could be provided in 3 different ways:

- Last GPS position reported by the device
- Manual location
- gateway position which has received the last message

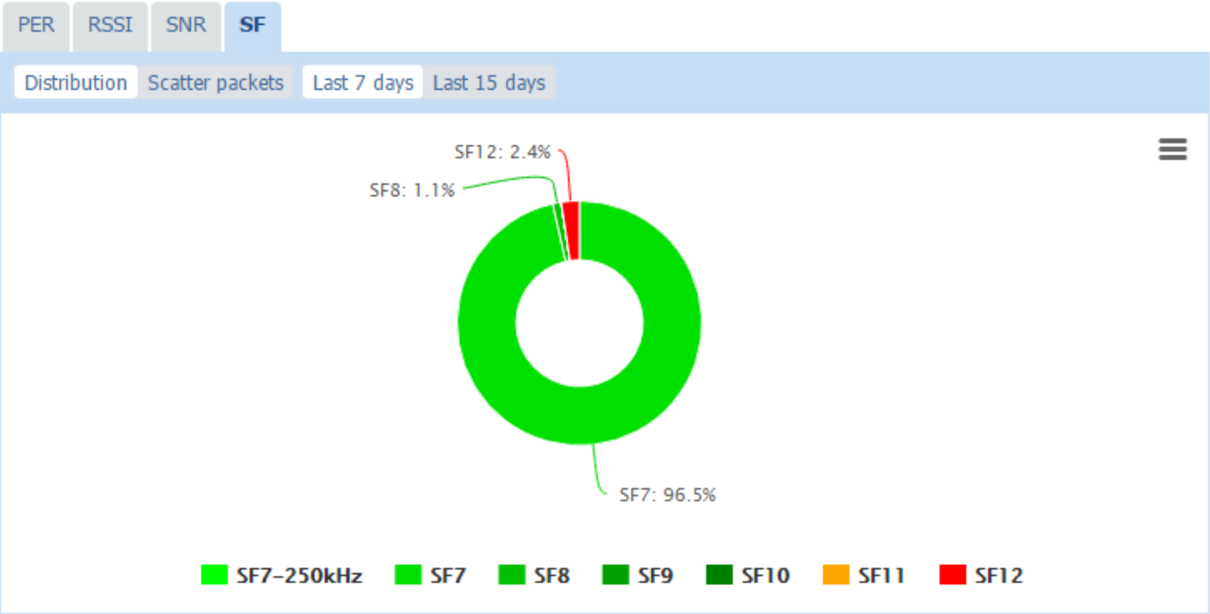
The **View location history** button displays a map with markers showing where the devices are located.

- Uplink/Downlink frame



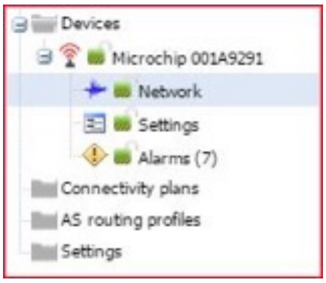
The graphic above displays the number of uplink/downlink packets and payloads (bytes) over the selected period (Daily totals, last 7 days, last 15 days).

- PER/RSSI/SNR/SF graph



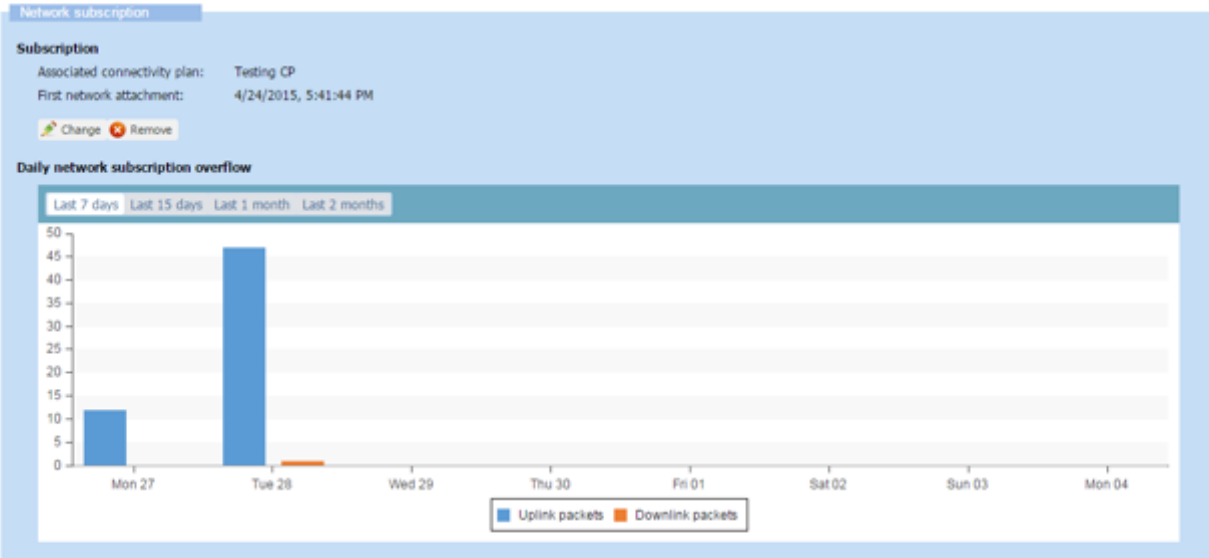
The graphic above displays the PER, RSSI, SNR or SF distribution over the selected period.

4.1.2.1 Device Network



The Devices/Network section provides information on the Network subscription, associated connectivity plan, routing plans associated to the device and the Network Coverage.

- Network subscription



Network subscription

Subscription

Associated connectivity plan: Testing CP

First network attachment: 4/24/2015, 5:41:44 PM

Daily network subscription overflow

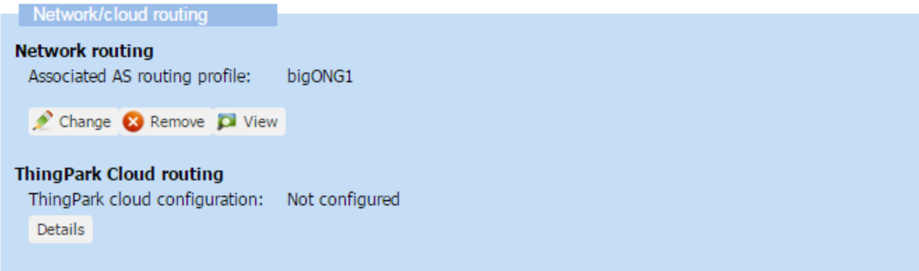
The Subscription section displays the current associated connectivity plan and the first date on which the device has communicated.

The graphic displays the usage of the device in the connectivity plan, showing the number of packets on the selected period.

HOW-TO change the Connectivity plan

- Go to the device Edit view
- Click on **Change** in the subscription section
- Select the desired Connectivity Plan from the drop-down menu
- Click **Save**

- Network/cloud routing



Network/cloud routing

Network routing

Associated AS routing profile: bigONG1

ThingPark Cloud routing

ThingPark cloud configuration: Not configured

The Network routing section displays the current associated AS routing profile. It is possible to view the details of this routing profile, or also change and remove it.

HOW-TO change or assign an AS Routing Profile

- Go to the device Edit view
- Click on **Change** in the Network routing section
- Select the preferred AS routing profile from the drop-down menu in the pop-up
- Click **Save**

4.1.3 Device provisioning

The device provisioning allows users to create devices and register them on the network through **Activation By Personalization**.

There are two ways to create a device

- Manual creation: create devices one by one
- Batch creation: mass import from a csv file to create several devices in one go

Information required to create a new device:

- DevEUI
- DevAddr
- NwkSKey

Optional information may be required:

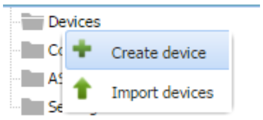
- AppSKey

4.1.3.1 ABP Manual Device creation

The manual creation is mainly for support or development purpose. Once the device is in production, refer to the Batch provisioning mode.

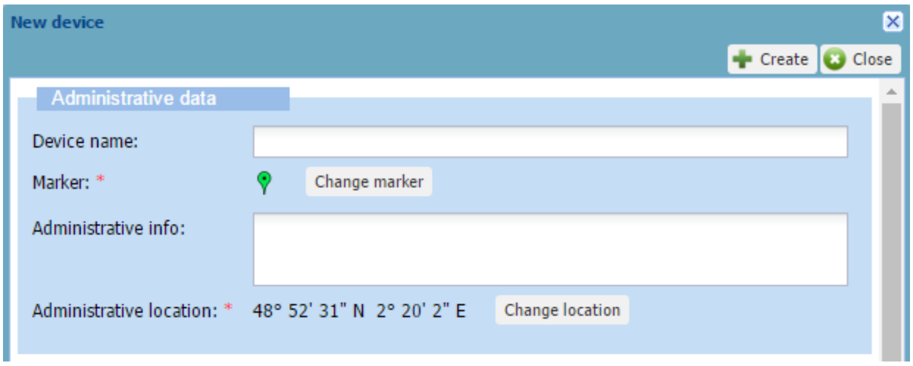
To create a new device manually:

- Right click on **Devices**



- Click on **Create device**
- A new pop-up appears with 4 sections: Administrative data, Device identification, Network parameters, Application layer handling

- Fill out the Administrative data section

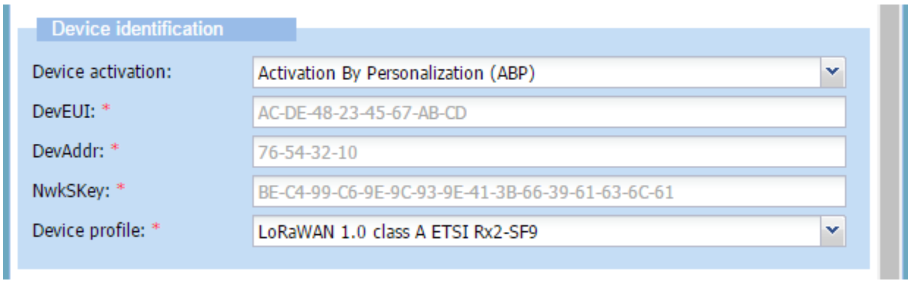


Administrative data

Device name:	
Marker: *	
Administrative info:	
Administrative location: *	48° 52' 31" N 2° 20' 2" E

  - Enter a name for the device
  - Change the marker if you wish to customize the device marker
  - Enter any relevant administrative information (Such information will be displayed in the e-mail generated by an alarm)
  - Set a location
    - Network location
    - Manual location

- Fill out the Device identification section. Please note that all fields are mandatory

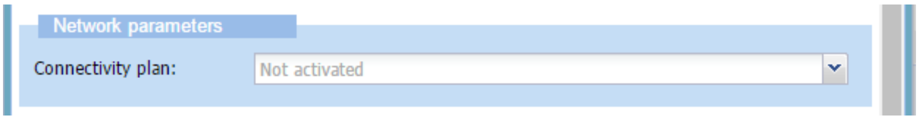


Device identification

Device activation:	Activation By Personalization (ABP)
DevEUI: *	AC-DE-48-23-45-67-AB-CD
DevAddr: *	76-54-32-10
NwkSKey: *	BE-C4-99-C6-9E-9C-93-9E-41-3B-66-39-61-63-6C-61
Device profile: *	LoRaWAN 1.0 class A ETSI Rx2-SF9

  - Device EUI (hexa): **devEUI**, device EUI, globally unique IEEE EUI-64 address
  - Network address (hexa): **devAddr**, device Address
  - Device profile: please refer to Device Profile
  - Network key (hexa): **NwkSKey**, Network Session Key, 128-bit key

- Fill the Network section

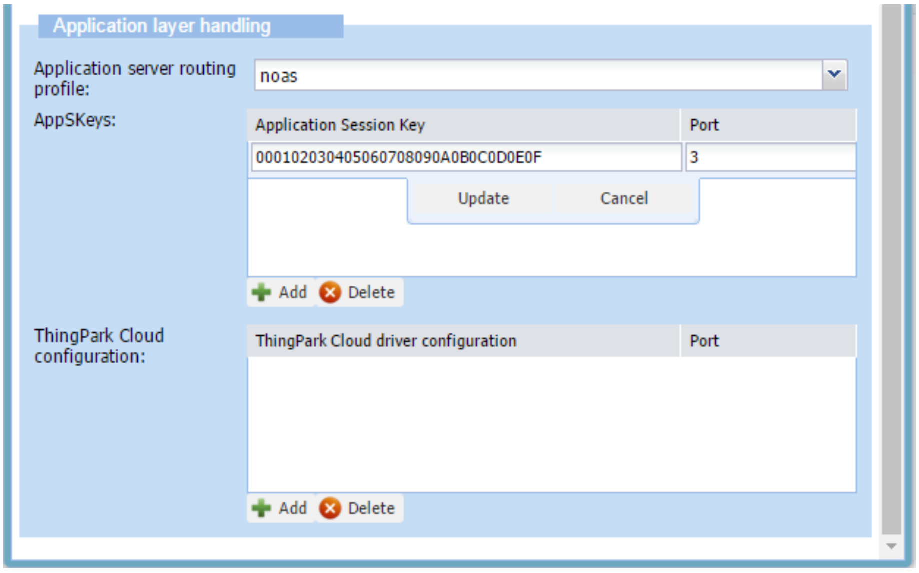


Network parameters

Connectivity plan:	Not activated

  - Choose a **Connectivity plan** in the drop-down menu, the displayed count indicates the remaining connectivity plans available
  - Choose an **AS routing profile** in the drop-down menu

- Fill out the Application layer section



Application layer handling

Application server routing profile: noas

AppSKeys:

Application Session Key	Port
000102030405060708090A0B0C0D0E0F	3

Update
Cancel
Add
Delete
ThingPark Cloud configuration:
ThingPark Cloud driver configuration
Port
Add
Delete

- To add a new Application Key
  - Click on **Add**
  - Fill the **Key**, 128-bit key
  - Enter a **port** number
  - Click on **Update**

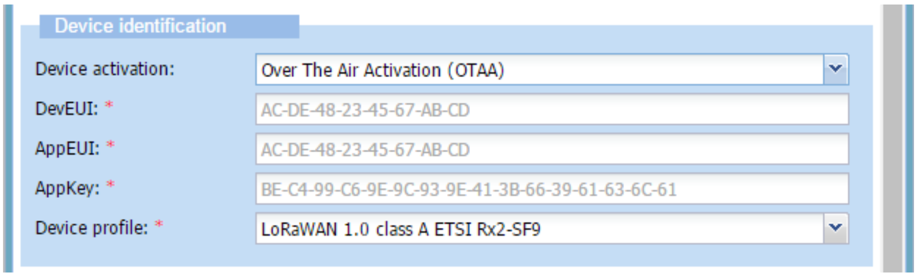
NOTE

The port 0 is encrypted by the NwkSKey

The optional 128-bit AppSkey is used to encrypt the payload of the messages, and has to be shared with the application server. You may decide to use a unique AppSkey for all LoRaWAN ports used by your device (* keyword), or to allocate one AppSkey for each port.

- If you do not provision the AppSkey, ThingPark Wireless will forward the payload in encrypted form to the application servers and has no access to the payload clear-form content. Network tools like Wireless Logger will be unable to decode the payload and will present it in encrypted form.
- If you provision the AppSkey(s), the LRC will decode the payload before forwarding it to the application server(s) and the Wireless Logger tool will also present the packet payload in decoded form (and parse this payload if the payload syntax is supported).

4.1.3.2 OTAA Manual Device creation



Device identification

Device activation:	Over The Air Activation (OTAA)
DevEUI: *	AC-DE-48-23-45-67-AB-CD
AppEUI: *	AC-DE-48-23-45-67-AB-CD
AppKey: *	BE-C4-99-C6-9E-9C-93-9E-41-3B-66-39-61-63-6C-61
Device profile: *	LoRaWAN 1.0 class A ETSI Rx2-SF9

The Join device activation procedure is similar as the ABP provisioning, except about the keys to inquire.

Fill out the Device identification section. Please note that all fields are mandatory

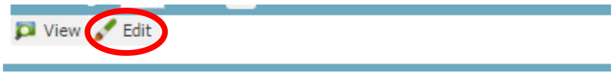
- Device EUI (hexa): **devEUI**, globally unique IEEE EUI-64 address
- Device profile: please refer to Device Profile
- Application EUI (hexa): AppEUI is a global application ID in IEEE EUI64 address space that uniquely identifies application provider of the end-device
- Application key (hexa): is an AES-128 application key specific for the end-device that is assigned by the application owner to the end-device and is responsible to encrypt JOIN communication.

4.1.3.3 Modify a device

Modifying a device allows you to update device-related data such as the name, the manually entered location, define another connectivity plan or define a routing plan.

Start by opening a device in Edit view:

- Select the device to delete in the list
- Click on **Edit** to enter the Edit view



Modification allowed:

- Device name
- Administrative info
- Device location
- Device marker
- Change/remove a connectivity plan
- Change/remove a AS routing plan

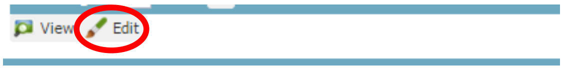
Finally, to confirm the changes, go back to the device details by selecting the device in the column in the left sidebar, then click **Save** in the top-right corner of the screen.

4.1.3.4 Delete a device

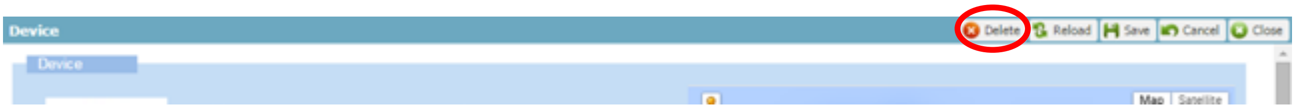
Deleting a device is an action which cannot be undone and should be handled with care. All device details and device status information will be lost.

Below are the steps to delete a device:

- Select the device to delete in the list
- Click on **Edit** to enter the Edit view



- Click on **Delete**

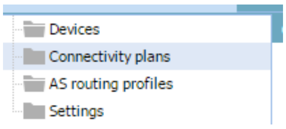


- Confirm in the pop-up to delete the device
- Go back to the device list, click on **Refresh** to refresh the list

4.2 Connectivity plan

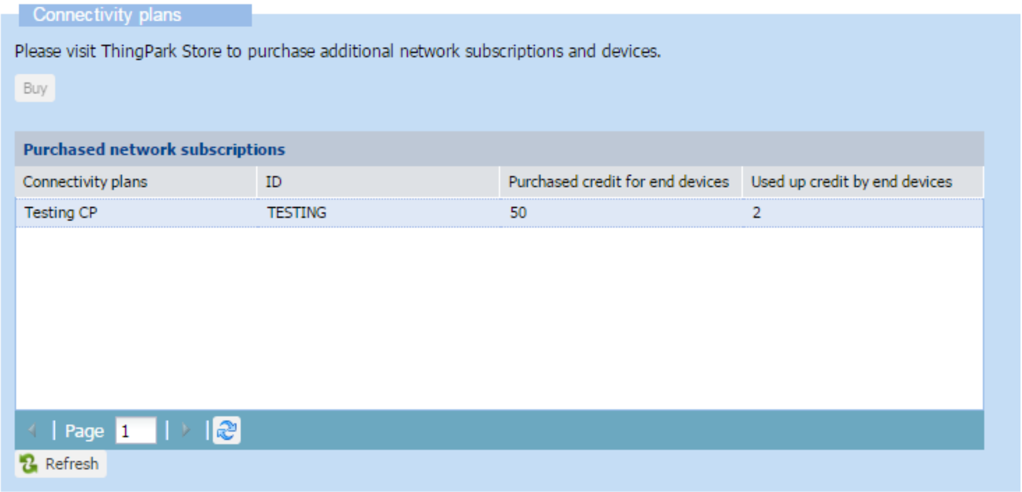
The Connectivity plan defines the network connectivity features (e.g. confirmed messages, downlink traffic), and traffic policing parameters (token bucket regulators for uplink and downlink traffic) and is associated to a given activation and recurring fee.

To access Connectivity plans, click on Connectivity plans in the left sidebar menu:



- Connectivity plan section

First Connectivity plan section displays the available plans in your account:



Please visit ThingPark Store to purchase additional network subscriptions and devices.

Purchased network subscriptions

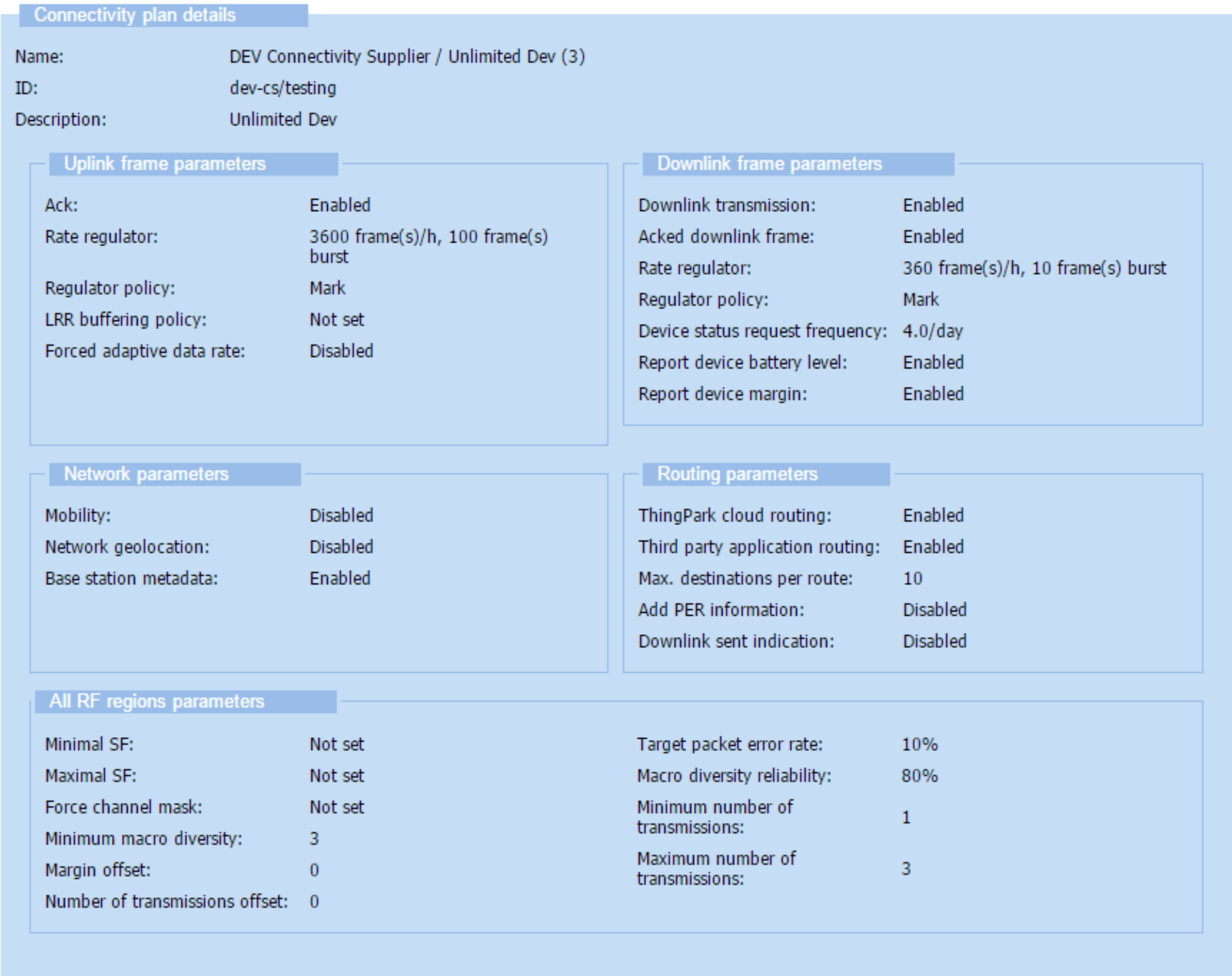
Connectivity plans	ID	Purchased credit for end devices	Used up credit by end devices
Testing CP	TESTING	50	2

- Connectivity plans: name of the connectivity plan
- ID of the connectivity plan (required for batch provisioning)
- Purchased credit for end devices: number of maximum devices allowed in the plan
- Used up credit by end devices: number of devices registered on the selected plan

So the remaining devices that could be provisioned are “Purchased credit” - “Used up credit”.

Ask Actility to purchase additional Connectivity Plans.

4.2.1 Connectivity Plan details



Connectivity plan details

Name:	DEV Connectivity Supplier / Unlimited Dev (3)
ID:	dev-cs/testing
Description:	Unlimited Dev

Uplink frame parameters

Ack:	Enabled
Rate regulator:	3600 frame(s)/h, 100 frame(s) burst
Regulator policy:	Mark
LRR buffering policy:	Not set
Forced adaptive data rate:	Disabled

Downlink frame parameters

Downlink transmission:	Enabled
Acked downlink frame:	Enabled
Rate regulator:	360 frame(s)/h, 10 frame(s) burst
Regulator policy:	Mark
Device status request frequency:	4.0/day
Report device battery level:	Enabled
Report device margin:	Enabled

Network parameters

Mobility:	Disabled
Network geolocation:	Disabled
Base station metadata:	Enabled

Routing parameters

ThingPark cloud routing:	Enabled
Third party application routing:	Enabled
Max. destinations per route:	10
Add PER information:	Disabled
Downlink sent indication:	Disabled

All RF regions parameters

Minimal SF:	Not set	Target packet error rate:	10%
Maximal SF:	Not set	Macro diversity reliability:	80%
Force channel mask:	Not set	Minimum number of transmissions:	1
Minimum macro diversity:	3	Maximum number of transmissions:	3
Margin offset:	0		
Number of transmissions offset:	0		

In the **Connectivity Plan details** tab, the name, ID and description of the plan are displayed.

Furthermore, this tab provides a view on the following characteristics of the plan:

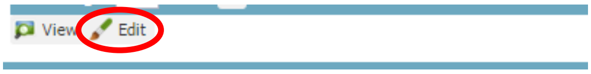
- Uplink frame parameters
  - Ack: Feature flag to allow the network to send an ACK when requested by a device operating in confirmed frame up mode.
  - Rate regulator: nb of frames allowed per hour, nb of frames allowed in burst
  - Regulator policy: uplink and downlink regulator policy
    - Mark: the system will keep track of devices exceeding the limits set
    - Drop: the system will drop packets that are beyond the limits set
  - LRR buffering policy: RFU
  - Forced adaptive data rate: Force a specific DataRate
- Downlink frame parameters
  - Downlink transmission: activation state of the downlink transmission
  - Acked downlink frame: Allow to send downlink confirmed
  - Rate regulator: number of frames allowed per hour, number of frames allowed in burst
  - Regulator policy: uplink and downlink regulator policy (see Uplink frame Regulatory policy above)
  - Device status request frequency: Number of DevStatusReq sent by the server in 24h
  - Report device battery level: Report Device battery usage level to the Device Manager application and to Third Party Application Servers
  - Report device margin: Report Device signal margin to the Device Manager application and to Third Party Application Servers
- Network parameters

  - Mobility: RFU
  - Network geolocation: RFU
  - Base station metadata: feature flag to provide LRR meta information (RSSI, SR, SNR, LRR, …) to third party application
- Routing parameters
  - ThingPark cloud routing: Feature flag to allow routing to the ThingPark cloud service
  - Third party application routing: Feature flag to allow routing to the ThingPark cloud service
  - Max. destinations per route: maximum number of destinations per route
  - Add PER information: Forward PER to the Application Server
  - Downlink sent indication: Forward sent downlink indication to the Application Server
- All RF regions parameters
  - Minimal/Maximal SF: Lowest/highest spreading factor allowed for a device
  - Force channel mask: Force a specific channel mask
  - Minimum macro diversity: Minimum number of gateways which received the uplink
  - Margin offset: The offset to apply on top of the global SNR margin set at RFregion level to tune the global margin differently for different class of services to control uplink PER
  - Number of transmissions offset: Tune the number of transmissions for each uplink packet (aka redundancy) differently for different class of service to optimize uplink performance
  - Target packet error rate: "Packet Error Rate value targeted by the ADR algorithm
  - Macro diversity reliability: Minimum probability target of having N Base Stations receiving Device uplink packets
  - Minimum/Maximum number of transmissions: Minimum/Maximum number of Device uplink transmissions to ensure quality of service will not be degraded

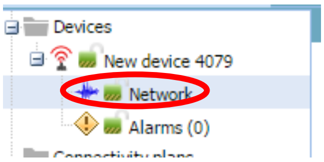
4.2.2 Assign/Change a Connectivity Plan

First of all, go to the Device Edit view in order to modify a device:

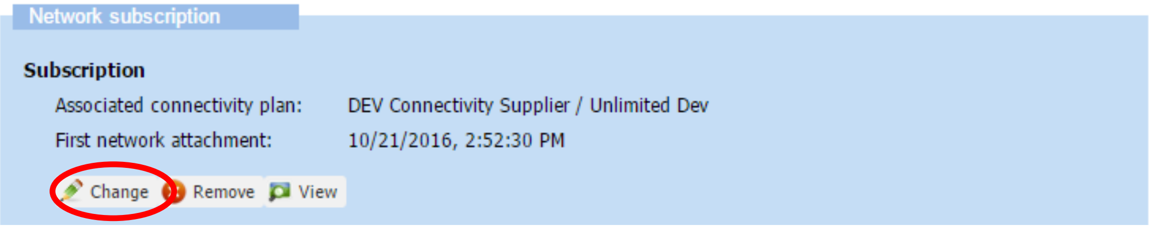
- Select the device to delete in the list
- Click on **Edit** to enter the Edit view



- Go to the **Network** section



- In the **Network subscription > Subscription** section
- Click on **Change**

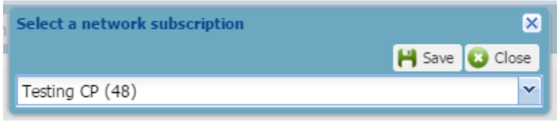

Network subscription

Subscription

Associated connectivity plan: DEV Connectivity Supplier / Unlimited Dev

First network attachment: 10/21/2016, 2:52:30 PM

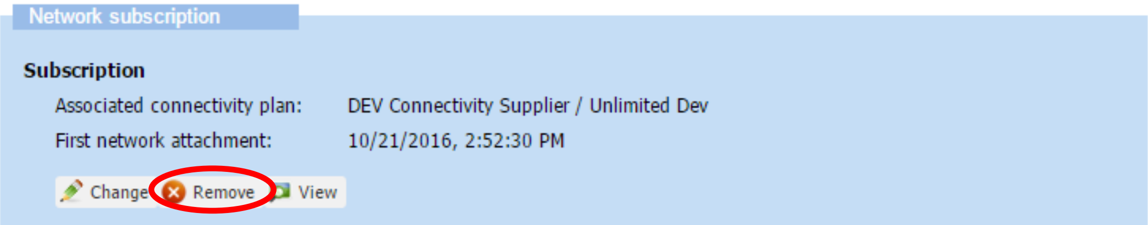
- Select your new Connectivity plan in the drop-down menu



Testing CP (48)

- Click on **Save**

4.2.3 Remove a Connectivity Plan



Subscription

Associated connectivity plan: DEV Connectivity Supplier / Unlimited Dev

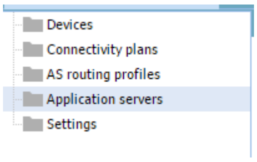
First network attachment: 10/21/2016, 2:52:30 PM

To remove a Connectivity Plan, use the same process as above to assign/change a Connectivity Plan, then click **Remove**.

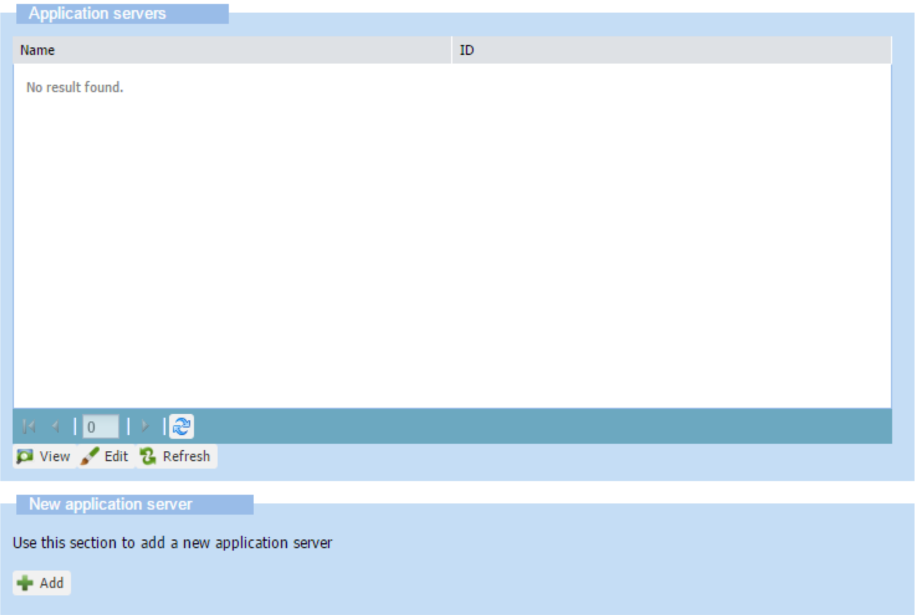
4.3 Application servers

The Application Server (AS) needs to be defined prior to adding a routing profile to it.

TO access the Application Server settings, click on Application servers on the left sidebar menu:



The right-hand side will then be populated with the following dialog:

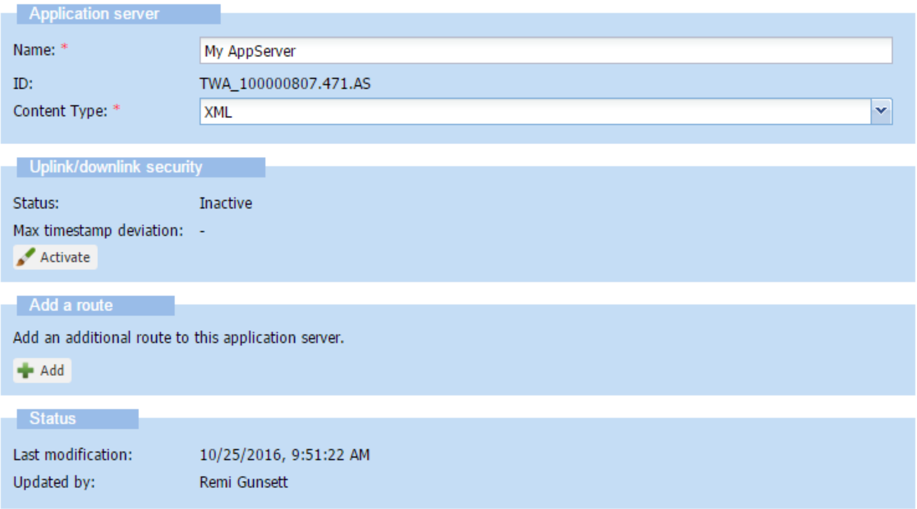


Name	ID
No result found.	

Use this section to add a new application server

4.3.1 Create a new application server

- In the 'New application server' section, click on ➕ Add
- In the name section of the new dialog, enter the name of your new Application server:



Application server

Name: *	My AppServer
ID:	TWA_100000807.471.AS
Content Type: *	XML

Uplink/downlink security

Status:	Inactive
Max timestamp deviation:	-

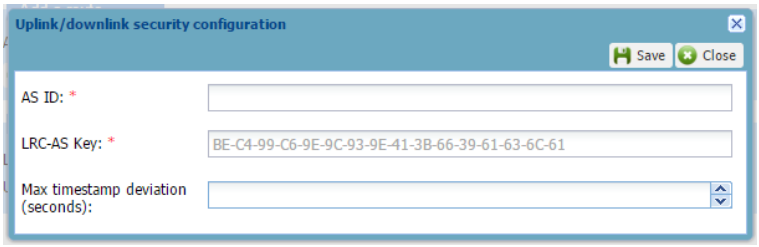
Add a route

Add an additional route to this application server.

Status

Last modification:	10/25/2016, 9:51:22 AM
Updated by:	Remi Gunsett

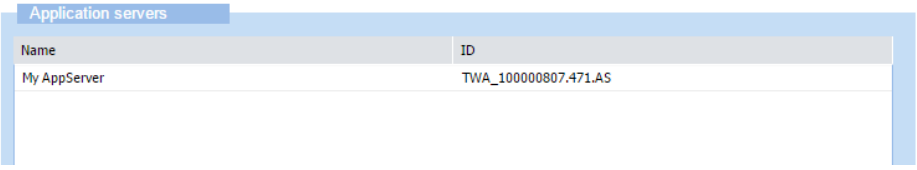
- A new dialog appears.
- Select the type of content your Application Server can handle (xml or json) for the metadata, including a field for the payload
- If you want the security feature to be enabled, click on 'Activate' in the 'Uplink/downlink security' section



Uplink/downlink security configuration

AS ID: *	
LRC-AS Key: *	BE-C4-99-C6-9E-9C-93-9E-41-3B-66-39-61-63-6C-61
Max timestamp deviation (seconds):	

  - Enter the Application Server unique ID, signature key and allowed timestamp deviation between the LRC and your Application Server. This can seriously affect communication if both sides are not properly synchronized.
  - Click on 'Save' to validate the information
- Click on 'Save' to save the new Application server, then close.

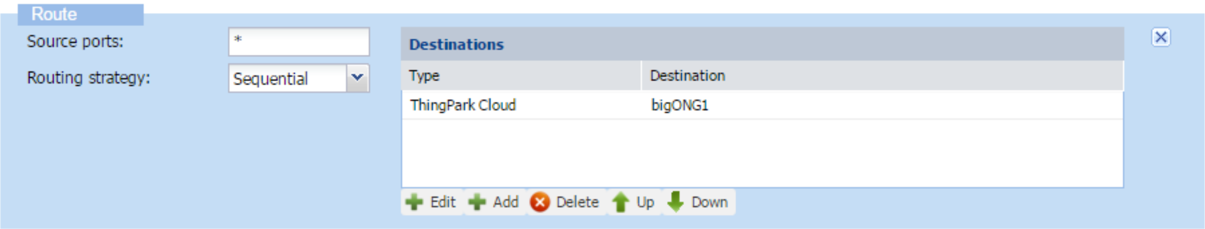


Application servers

Name	ID
My AppServer	TWA_100000807.471.AS

4.3.2 Edit an Application server

- In the list of application server, select the server you want to delete and click on 'Edit', you might have to confirm that you want to edit if not already done.
- In the 'Add a route' section, click on 'Add'
  - A new layer "Route" appears



Source ports: *

Routing strategy: Sequential

Destinations

Type	Destination
ThingPark Cloud	bigONG1

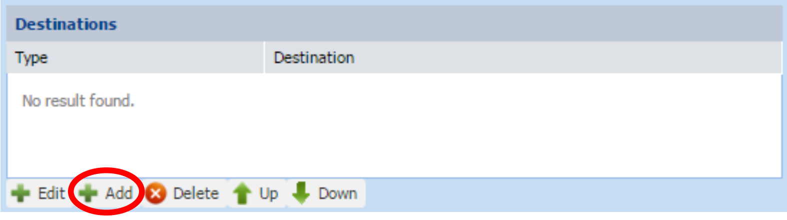
- Enter the **Source ports** to route

The Source ports are the LoRaWAN ports, it could be only one port (1), a range of ports (1-4), or all ports ('*').

- Choose the **Routing strategy**

In case multiple destinations are given, the routing strategy defines how the data will be sent to these destinations.
If "Sequential" is selected, the data will be sent to the first destination and only be sent to the subsequent destination if the previous one is not available. If "Blast" is selected, the data will be sent to all destinations at the same time.

- Click on **Add** to create a new one

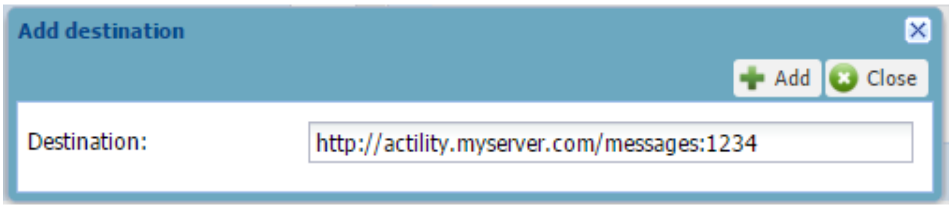


Destinations

Type	Destination

No result found.

- Select the Destination of the new route

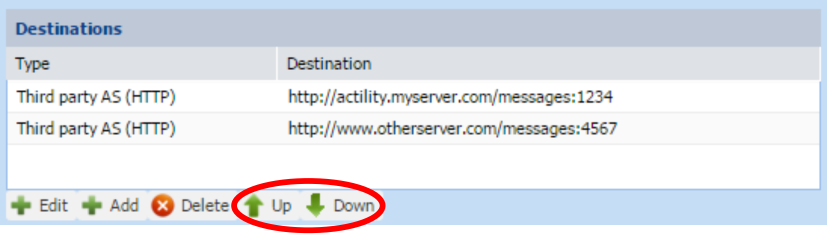


Add destination

Destination: http://actility.myserver.com/messages:1234

- Click on **Add** to add this new destination

- Sort between different destinations, for order selection if using Sequential strategy
  - You can change the order in which messages will be sent using the arrow buttons



Destinations

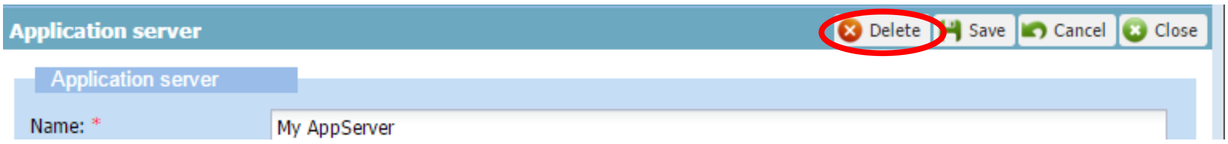
Type	Destination
Third party AS (HTTP)	http://actility.myserver.com/messages:1234
Third party AS (HTTP)	http://www.otherserver.com/messages:4567

- To edit a destination, select it and click on **Edit**
- To delete a destination, select it and click on **Delete**

4.3.3 Delete an Application server

- In the list of application server, select the server you want to delete and click on 'Edit', you might have to confirm that you want to edit if not already done.

- In the new dialog, click on ‘Delete’ on the top part of the dialog and confirm

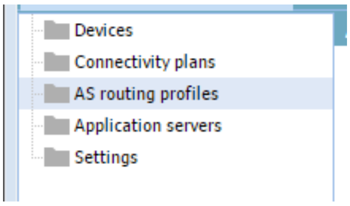


Name: * My AppServer

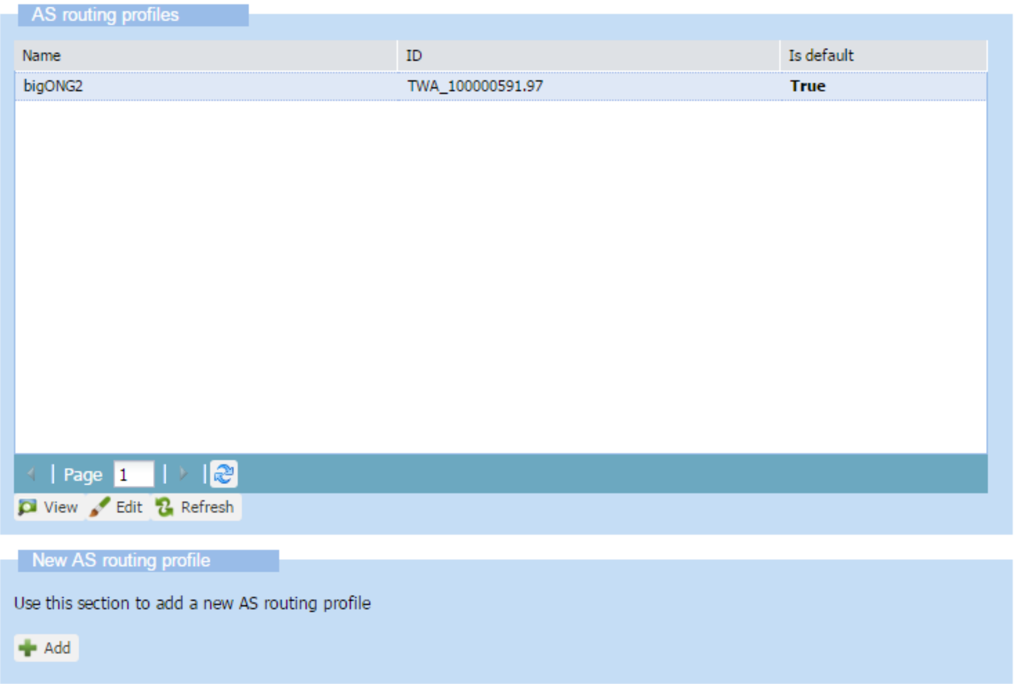
4.4 *AS routing profiles*

The AS (Application Server) routing profile defines how the sensor data is routed to a back-end application.

To access an AS routing profile, click on AS routing profiles on the left sidebar menu:



You will see the existing AS routing profiles in the list where you can View details or Edit an AS routing profile.



Name	ID	Is default
bigONG2	TWA_100000591.97	**True**

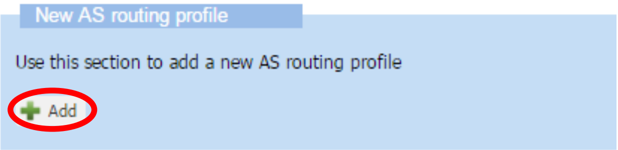
Use this section to add a new AS routing profile

The second section **New AS routing profile** gives the capability to add a new AS routing profile.

4.4.1 Create an AS Routing Profile

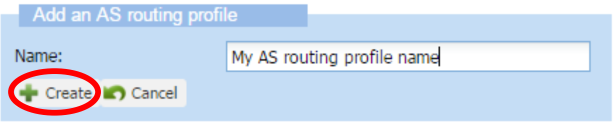
In order to create a new AS Routing Profile, go through the following steps:

- Click on **Add** in the **New AS routing profile** section



Use this section to add a new AS routing profile

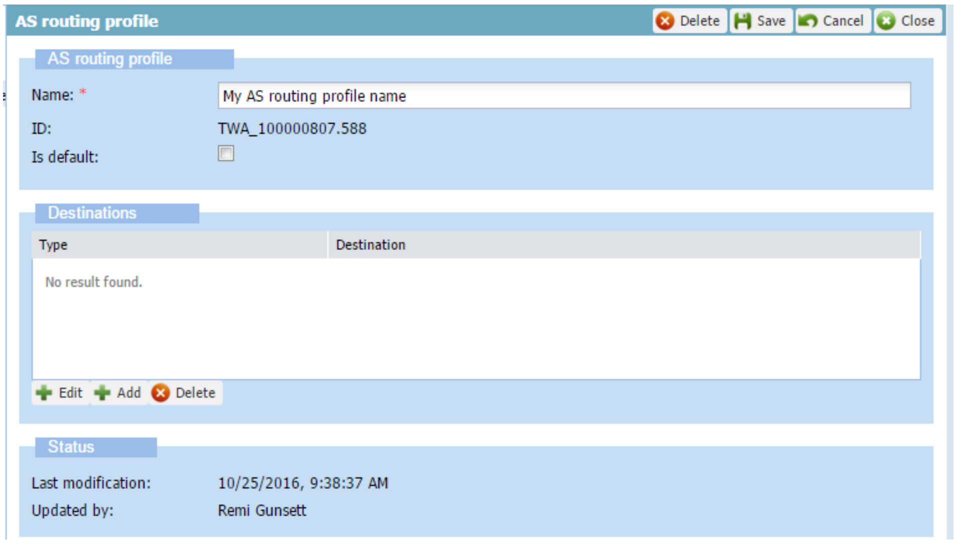
- Enter a name to the desired new AS routing profile



Add an AS routing profile

Name: My AS routing profile name

- Click on **Create**
- This new AS routing profile is opened and you can now edit it



AS routing profile

AS routing profile

Name: * My AS routing profile name

ID: TWA_100000807.588

Is default: □

Destinations

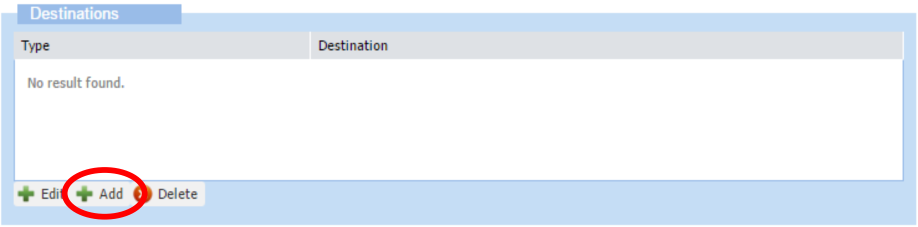
Type	Destination
No result found.	

Status

Last modification: 10/25/2016, 9:38:37 AM

Updated by: Remi Gunsett

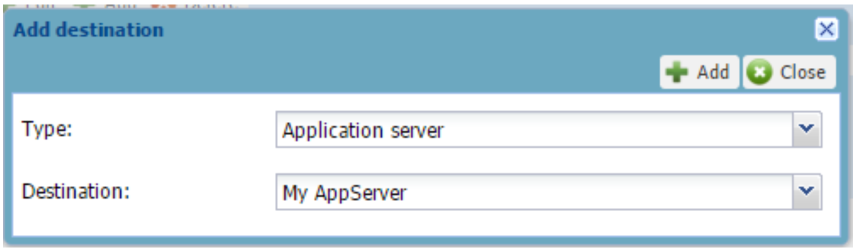
- Set or unset the profile as default: check **Is default** checkbox
- Add a route
    - Click on **Add** in the “Destinations” section



Destinations

Type	Destination
No result found.	

    - A new pop up ‘Add destination’ appears



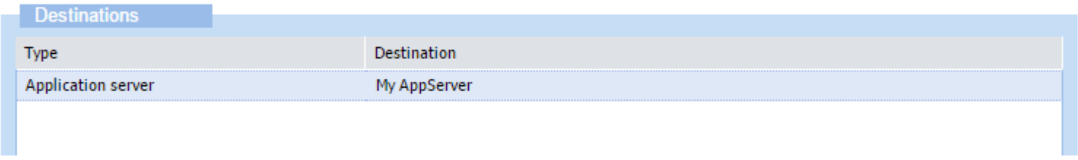
Add destination

Type: Application server

Destination: My AppServer

    - Select the type of destination (ThinkPark Cloud or Application server) and the destination
    - Click on ‘Add’

- The Application server is now in the list



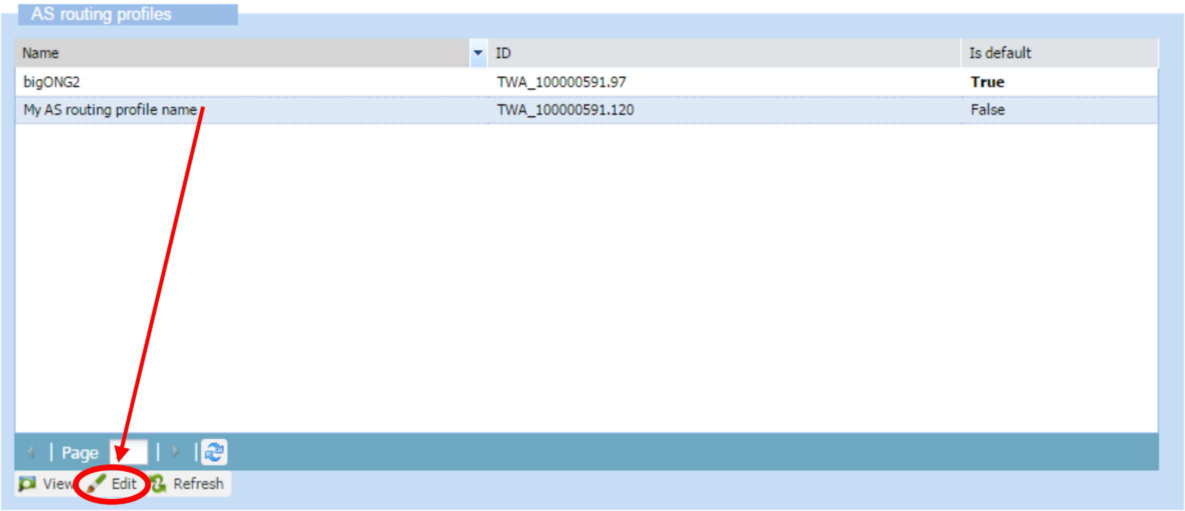
Destinations

Type	Destination
Application server	My AppServer

4.4.2 Modify or Delete an AS Routing Profile

In order to modify or remove an AS Routing Profile

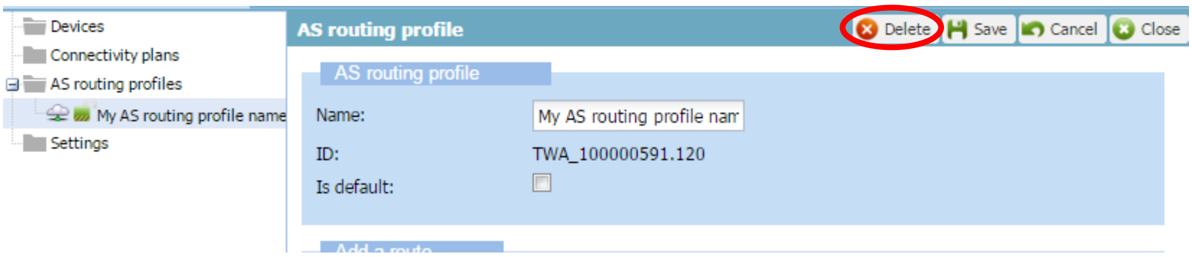
- Select the applicable Profile in the list
- Click on **Edit**



AS routing profiles

Name	ID	Is default
bigONG2	TWA_100000591.97	**True**
My AS routing profile name	TWA_100000591.120	False

- On the AS routing profile details in Edit view if edition is desired
- Click on **Delete** if deletion is desired



AS routing profile

Name: My AS routing profile nam

ID: TWA_100000591.120

Is default: □

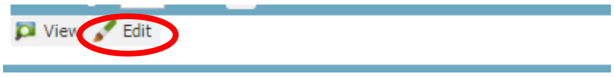
NOTE

Remember to click on Save after any modification made in case of edition.

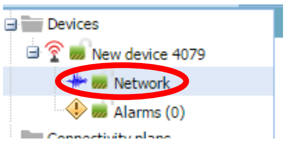
4.4.3 Define or Remove an AS Routing Profile

First, go to the Device Edit view in order to modify a device:

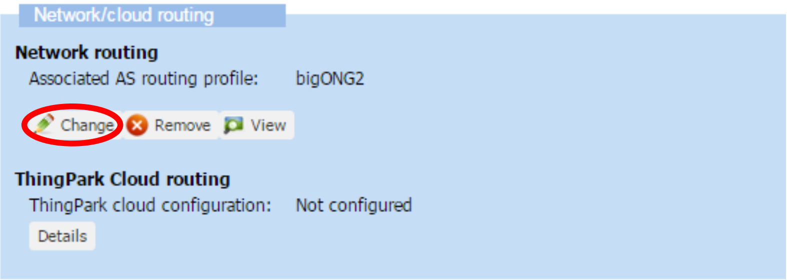
- Select the device to delete in the list
- Click on **Edit** to enter the Edit view



- Go to the **Network** section



- In the **Network/cloud routing** section
- Click on **Change**



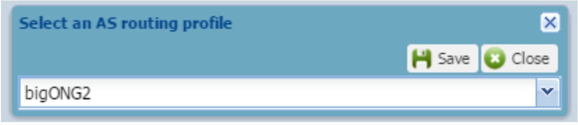
Network routing

Associated AS routing profile: bigONG2

ThingPark Cloud routing

ThingPark cloud configuration: Not configured

- Select your new AS routing profile in the drop-down menu



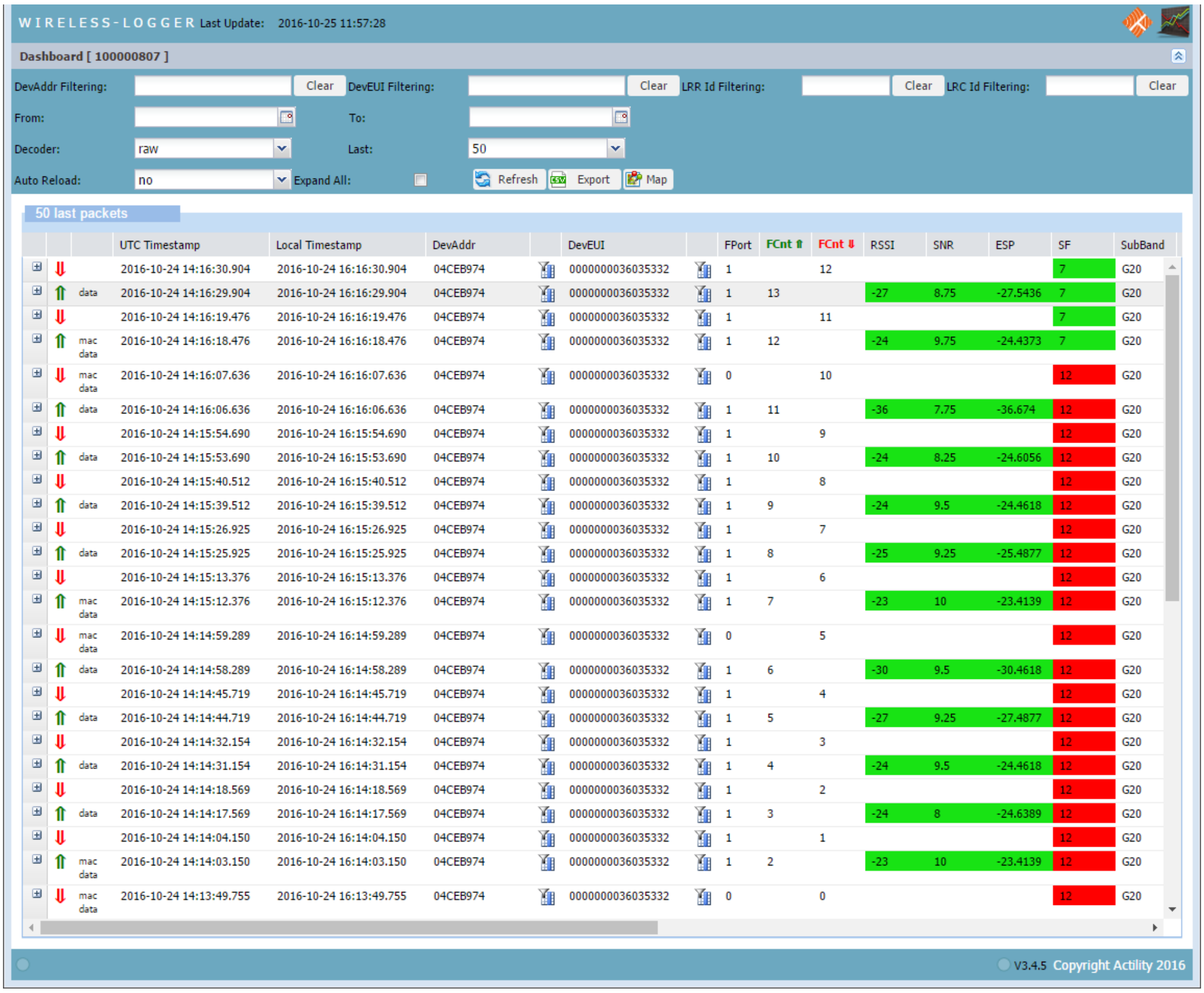
bigONG2

- To remove the current associated AS routing profile, click on **Remove**

5 WirelessLogger

- ❖ Launch the WirelessLogger from your Dashboard menu.

When you login to the Wireless Logger interface, the interface automatically displays the 50 last messages received from the devices provisioned into your Device Manager.



WIRELESS-LOGGER Last Update: 2016-10-25 11:57:28

Dashboard [100000807]

DevAddr Filtering: Clear DevEUI Filtering: Clear LRR Id Filtering: Clear LRC Id Filtering: Clear

From: To:

Decoder: raw Last: 50

Auto Reload: no Expand All: □

Refresh Export Map

50 last packets

			UTC Timestamp	Local Timestamp	DevAddr	DevEUI	FPort	FCnt ⇑	FCnt ⇓	RSSI	SNR	ESP	SF	SubBand
⊞	⇓		2016-10-24 14:16:30.904	2016-10-24 16:16:30.904	04CEB974	0000000036035332	1		12				7	G20
⊞	⇑	data	2016-10-24 14:16:29.904	2016-10-24 16:16:29.904	04CEB974	0000000036035332	1	13		-27	8.75	-27.5436	7	G20
⊞	⇓		2016-10-24 14:16:19.476	2016-10-24 16:16:19.476	04CEB974	0000000036035332	1		11				7	G20
⊞	⇑	mac data	2016-10-24 14:16:18.476	2016-10-24 16:16:18.476	04CEB974	0000000036035332	1	12		-24	9.75	-24.4373	7	G20
⊞	⇓	mac data	2016-10-24 14:16:07.636	2016-10-24 16:16:07.636	04CEB974	0000000036035332	0		10				12	G20
⊞	⇑	data	2016-10-24 14:16:06.636	2016-10-24 16:16:06.636	04CEB974	0000000036035332	1	11		-36	7.75	-36.674	12	G20
⊞	⇓		2016-10-24 14:15:54.690	2016-10-24 16:15:54.690	04CEB974	0000000036035332	1		9				12	G20
⊞	⇑	data	2016-10-24 14:15:53.690	2016-10-24 16:15:53.690	04CEB974	0000000036035332	1	10		-24	8.25	-24.6056	12	G20
⊞	⇓		2016-10-24 14:15:40.512	2016-10-24 16:15:40.512	04CEB974	0000000036035332	1		8				12	G20
⊞	⇑	data	2016-10-24 14:15:39.512	2016-10-24 16:15:39.512	04CEB974	0000000036035332	1	9		-24	9.5	-24.4618	12	G20
⊞	⇓		2016-10-24 14:15:26.925	2016-10-24 16:15:26.925	04CEB974	0000000036035332	1		7				12	G20
⊞	⇑	data	2016-10-24 14:15:25.925	2016-10-24 16:15:25.925	04CEB974	0000000036035332	1	8		-25	9.25	-25.4877	12	G20
⊞	⇓		2016-10-24 14:15:13.376	2016-10-24 16:15:13.376	04CEB974	0000000036035332	1		6				12	G20
⊞	⇑	mac data	2016-10-24 14:15:12.376	2016-10-24 16:15:12.376	04CEB974	0000000036035332	1	7		-23	10	-23.4139	12	G20
⊞	⇓	mac data	2016-10-24 14:14:59.289	2016-10-24 16:14:59.289	04CEB974	0000000036035332	0		5				12	G20
⊞	⇑	data	2016-10-24 14:14:58.289	2016-10-24 16:14:58.289	04CEB974	0000000036035332	1	6		-30	9.5	-30.4618	12	G20
⊞	⇓		2016-10-24 14:14:45.719	2016-10-24 16:14:45.719	04CEB974	0000000036035332	1		4				12	G20
⊞	⇑	data	2016-10-24 14:14:44.719	2016-10-24 16:14:44.719	04CEB974	0000000036035332	1	5		-27	9.25	-27.4877	12	G20
⊞	⇓		2016-10-24 14:14:32.154	2016-10-24 16:14:32.154	04CEB974	0000000036035332	1		3				12	G20
⊞	⇑	data	2016-10-24 14:14:31.154	2016-10-24 16:14:31.154	04CEB974	0000000036035332	1	4		-24	9.5	-24.4618	12	G20
⊞	⇓		2016-10-24 14:14:18.569	2016-10-24 16:14:18.569	04CEB974	0000000036035332	1		2				12	G20
⊞	⇑	data	2016-10-24 14:14:17.569	2016-10-24 16:14:17.569	04CEB974	0000000036035332	1	3		-24	8	-24.6389	12	G20
⊞	⇓		2016-10-24 14:14:04.150	2016-10-24 16:14:04.150	04CEB974	0000000036035332	1		1				12	G20
⊞	⇑	mac data	2016-10-24 14:14:03.150	2016-10-24 16:14:03.150	04CEB974	0000000036035332	1	2		-23	10	-23.4139	12	G20
⊞	⇓	mac data	2016-10-24 14:13:49.755	2016-10-24 16:13:49.755	04CEB974	0000000036035332	0		0				12	G20

V3.4.5 Copyright Actility 2016

The interface contains a search bar and a result window displaying the messages.

5.1 Metadata

The metadata available for each message are:

- ❖ Direction of the data: up or down represented by an arrow
- ❖ Type of transmission: data, mac or simply an acknowledge
- ❖ UTC Timestamp
- ❖ Local Timestamp: UTC Timestamp translated to browser timezone
- ❖ Device Address
- ❖ Device EUI
- ❖ Port: Application port of the message
- ❖ Counter UP and Counter DOWN
- ❖ LRR RSSI: RSSI of the received message on LRR side
- ❖ LRR SNR: SNR of the received message on LRR side
- ❖ LRR ESP: ESP of the received message on LRR side
- ❖ SF: Spreading Factor

- Sub Band: LoRa sub band used for the message
- Channel: LoRa logical channel used for the message
- LRC Id: Id of the LRC server
- LRR Id: LRR with better SNR
- LRR Lat: LRR latitude
- LRR Long: LRR longitude
- LRR Count: number of LRR receiving this message. The system performs a 250ms buffering upon receiving a message to check if the same message arrives though other LRR, in which case LRR Count is incremented. If latency is uneven in the network, a message (with the same Counter Up and payload) may appear more than once in the Wlogger.
- Device Lat: Device latitude
- Device Lon: Device longitude
- LoS Distance (m): distance between the device and the LRR
- Map: displays the device and LRR on a map
- Trip: displays the location path of the device (if device location available)
- MIC: Checksum

NOTE

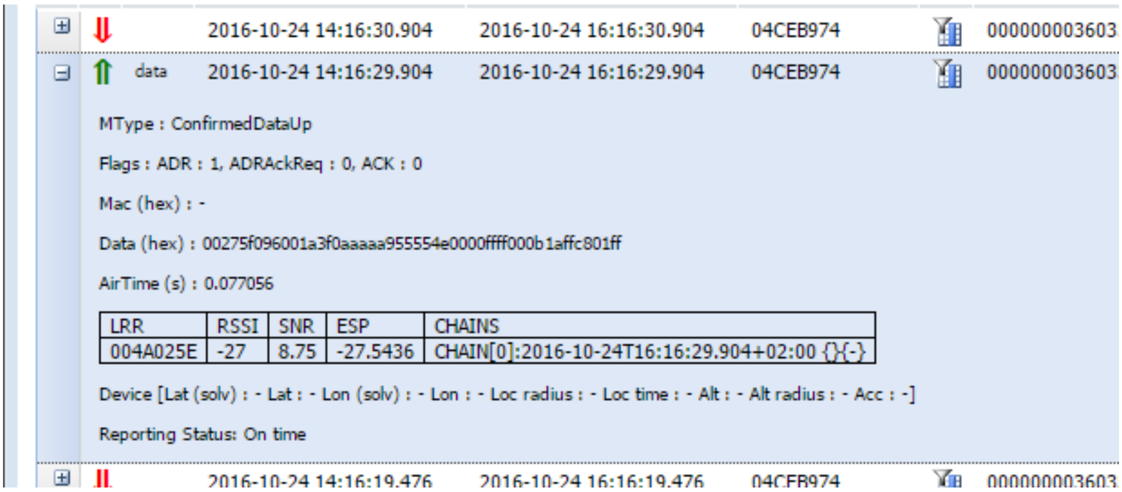
The GPS data (Device lat/lon, LoS, map and trip) are filled only if the device gives its location:

- Manual location
- Decoder selected if the location is in the payload

5.2 Payload

Click on the ⊞ icon on the left of a message in order to expand and display the message payload.

By default the message is displayed in raw data:



```
⊞ ⇓        2016-10-24 14:16:30.904    2016-10-24 16:16:30.904    04CEB974        000000003603
⊟ ⇑  data  2016-10-24 14:16:29.904    2016-10-24 16:16:29.904    04CEB974        000000003603

MType : ConfirmedDataUp

Flags : ADR : 1, ADRAckReq : 0, ACK : 0

Mac (hex) : -

Data (hex) : 00275f096001a3f0aaaaa955554e0000ffff000b1affc801ff

AirTime (s) : 0.077056
```

LRR	RSSI	SNR	ESP	CHAINS
004A025E	-27	8.75	-27.5436	CHAIN[0]:2016-10-24T16:16:29.904+02:00 {}{-}

```
Device [Lat (solv) : - Lat : - Lon (solv) : - Lon : - Loc radius : - Loc time : - Alt : - Alt radius : - Acc : -]

Reporting Status: On time

⊞ ⇓        2016-10-24 14:16:19.476    2016-10-24 16:16:19.476    04CEB974        000000003603
```

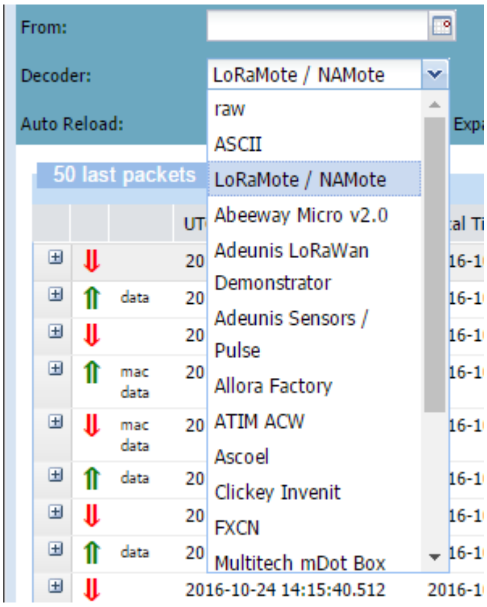
NOTE

If the AppSKey has been entered during the Device provisioning, then the payload has been decrypted by the LRC, if not then the payload is displayed encrypted.

5.3 Payload decoder

The payload can be displayed with the application data format if the decoded is known by the Wireless Logger.

- Select your decoder in the Decoder list:

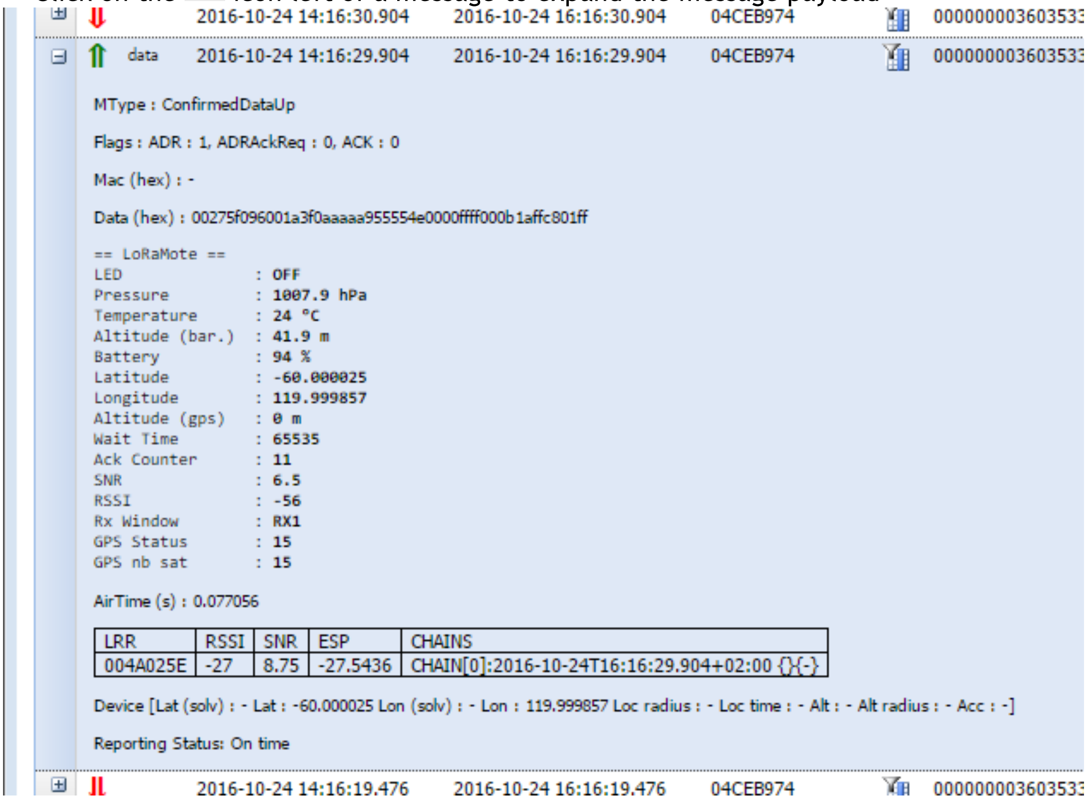


From:

Decoder: LoRaMote / NAMote

Auto Reload:

- Click on Refresh to refresh the messages
- Click on the ⊞ icon left of a message to expand the message payload



```
⊞ ⇓        2016-10-24 14:16:30.904   2016-10-24 16:16:30.904   04CEB974        00000000360353
⊟ ⇑  data  2016-10-24 14:16:29.904   2016-10-24 16:16:29.904   04CEB974        00000000360353
```

MType : ConfirmedDataUp

Flags : ADR : 1, ADRAckReq : 0, ACK : 0

Mac (hex) : -

Data (hex) : 00275f096001a3f0aaaaa955554e0000ffff000b1affc801ff

```
== LoRaMote ==
LED                : OFF
Pressure           : 1007.9 hPa
Temperature        : 24 °C
Altitude (bar.)    : 41.9 m
Battery            : 94 %
Latitude           : -60.000025
Longitude          : 119.999857
Altitude (gps)     : 0 m
Wait Time          : 65535
Ack Counter        : 11
SNR                : 6.5
RSSI               : -56
Rx Window          : RX1
GPS Status         : 15
GPS nb sat         : 15
```

AirTime (s) : 0.077056

LRR	RSSI	SNR	ESP	CHAINS
004A025E	-27	8.75	-27.5436	CHAIN[0]:2016-10-24T16:16:29.904+02:00 {}{-}

Device [Lat (solv) : - Lat : -60.000025 Lon (solv) : - Lon : 119.999857 Loc radius : - Loc time : - Alt : - Alt radius : - Acc : -]

Reporting Status: On time

```
⊞ ⇓        2016-10-24 14:16:19.476   2016-10-24 16:16:19.476   04CEB974        00000000360353
```

NOTE

Payload can only be decoded if the AppSKey is known by the ThingPark Wireless

6 Application Server

6.1 Uplink interface:

Uplink destination URIs are defined for each device in service profiles (basically one per device). The main criteria is the LoRa port numbers expressed as intervals, lists or single values. A default LoRa port destination can be declared.

There are two modes of distribution which are not exclusive:

- Blast/blind mode: packets are delivered to a set of destinations without acknowledgment control.
- Order mode: packets are delivered to a list of destinations until one of them confirms receipt (200OK).

Duplicate packets (same counter up and same payload) are not sent to application servers, as long as the multiple copies are received by the network infrastructure within a maximum delay of 250ms (configurable by the LPWA operator). The XML payload sent to the application servers still include the RF metadata corresponding to all receiving base stations.

The uplink message is transmitted in a HTTP/POST request with query parameters and an XML payload.

6.1.1 Query parameters:

```
LrnDevEui: 00000000007E074F          // the device DevEUI
LrnFPort: 2                          // the LoRaWAN port number
LrnInfos: UPHTTP_LAB_LORA            // the service profile name used to route the packet
```

6.1.2 XML payload

```
<?xml version="1.0" encoding="UTF-8"?>
<DevEUI_uplink xmlns="http://uri.actility.com/lora">
  <Time>2015-07-09T16:06:38.49+02:00</Time>         // timestamp for the packet
  <DevEUI>00000000007E074F</DevEUI>
  <FPort>2</FPort>                                  //LoRaWAN port number
  <FCntUp>11</FCntUp>                               // the uplink counter for this packet
  <FCntDn>10</FCntDn>                         // the downlink counter for the previous packet
  <ADRbit>1</ADRbit>
  <FCntDn>0</FCntDn>                                // the last downlink counter to the device
  <payload_hex>0027…bd00</payload_hex>              //LoRaWAN payload in hexa ascii format
  <mic_hex>38e7a3b9</mic_hex>                       // MIC in hexa ascii format
  <Lrcid>00000065</Lrcid>
  <LrrRSSI>-60.000000</LrrRSSI>
  <LrrSNR>9.750000</LrrSNR>
  <SpFact>7</SpFact>
  <SubBand>G1</SubBand>
  <Channel>LC2</Channel>
  <DevLrrCnt>3</DevLrrCnt>                    // number of LRRs which received this packet
  <Lrrid>08040059</Lrrid>
  <LrrLAT>48.874931</LrrLAT>
  <LrrLON>2.333673</LrrLON>
  <Lrrs>
    <Lrr>
      <Lrrid>08040059</Lrrid>
      <Chain>0</Chain>
      <LrrRSSI>-60.000000</LrrRSSI>
      <LrrSNR>9.750000</LrrSNR>
      <LrrESP>-51.568496</LrrESP>
    </Lrr>
    <Lrr>
      <Lrrid>33d13a41</Lrrid>
      <Chain>0</Chain>
      <LrrRSSI>-73.000000</LrrRSSI>
      <LrrSNR>9.750000</LrrSNR>
      <LrrESP>-41.754934</LrrESP>
    </Lrr>
    <Lrr>
      <Lrrid>a74e48b4</Lrrid>
      <Chain>0</Chain>
      <LrrRSSI>-38.000000</LrrRSSI>
      <LrrSNR>9.250000</LrrSNR>
      <LrrESP>-46.497314</LrrESP>
    </Lrr>
  </Lrrs>
  <CustomerID>100000507</CustomerID>
  <CustomerData>…</CustomerData>                    // ascii customer data set by provisioning
  <ModelCfg>0</ModelCfg>
  <DevAddr>007E074F</DevAddr>
</DevEUI_uplink>
```

6.1.3 JSON payload

```
{
	"DevEUI_uplink": {
		"Time": "2016-10-26T11:08:00.609+02:00",
		"DevEUI": "A81758FFFE0301EA",
		"FPort": "5",
		"FCntUp": "43749",
		"ADRbit": "1",
		"MType": "2",
		"FCntDn": "784",
		"payload_hex": "0100fd02290404b505020603c2070e0c",
		"mic_hex": "975457b1",
		"Lrcid": "00000127",
		"LrrRSSI": "-45.000000",
		"LrrSNR": "7.750000",
		"SpFact": "7",
		"SubBand": "G0",
		"Channel": "LC6",
		"DevLrrCnt": "3",
		"Lrrid": "29000119",
		"Late": "0",
		"LrrLAT": "48.875259",
		"LrrLON": "2.332962",
		"Lrrs": {
			"Lrr": [{
				"Lrrid": "29000119",
				"Chain": "0",
				"LrrRSSI": "-45.000000",
				"LrrSNR": "7.750000",
				"LrrESP": "-45.673985"
			},
			{
				"Lrrid": "080E0E94",
				"Chain": "0",
                        "LrrRSSI": "-51.000000",
                        "LrrSNR": "6.750000",
                        "LrrESP": "-51.832691"
                    },
                    {
                        "Lrrid": "08060311",
                        "Chain": "0",
                        "LrrRSSI": "-59.000000",
                        "LrrSNR": "10.250000",
                        "LrrESP": "-59.391785"
                    }]
                },
                "CustomerID": "100000128",
                "CustomerData": {
                    "alr": {
                        "pro": "LORA/Generic",
                        "ver": "1"
                    }
                },
                "ModelCfg": "0",
                "DevAddr": "040874A4"
            }
```

6.2 Downlink interface

Depending on the device provisioning (application secret keys) encryption/decryption can be performed by the Network Server.

The following HTTP/POST message format is used to tunnel the radio frame payload and associated metadata from the target application server to the Network Server. The downlink destination URI is the primary LRC cluster (lrc-dev.thingpark.com). The application server acts as a HTTP client and the Network Server acts as a HTTP server.

Such POST command may be generated easily by tools such as curl or POSTman.

```
curl -H "Content-type:application/x-www-form-urlencoded" -X POST
"https://api-dev1.thingpark.com/thingpark/lrc/rest/downlink?DevEUI=000000000F1D8693&FPort=1&Payload=01020304050607080900A0B0C0D0E0F&FCntDn=1234"
```

Query parameters:

- DevEUI (Mandatory): target device IEEE EUI64 in hexadecimal form (representing 8 octets)
- FPort (Mandatory): target port (in decimal format)
- Payload (Mandatory): hexadecimal payload. The hexadecimal payload will be encrypted by the LRC cluster if FCntDn parameter is absent, and if the LRC has been configured with an AppSKey for the specified LoRaWAN port, otherwise the Payload must be encrypted by the Application Server according to the LoRaWAN specification. The Application Server encryption uses the downlink counter, which is why the FCntDn query parameter is required in this case.
- FCntDn (Optional); LoRaWAN Downlink Counter value used to encrypt the payload. This query parameter is needed only if the Application server (not the Network Server) encrypts the payload. If present, FCntDn will be copied in the LoRaWAN header field FCnt, and the encrypted payload will be copied as-is to the LoRaWAN downlink frame by the Network Server.
- Confirmed (Optional). A value of Confirmed=0 requests transmission of an UNCONFIRMED downlink frame. A value of Confirmed=1 requests transmission of a CONFIRMED downlink frame. Default value Confirmed=0 (UNCONFIRMED).
  Support of Confirmed frame transmission is subject to Connectivity plan feature flag

"ackedDownlinkFrame": if the Confirmed flag is set on the HTTP POST and the device is associated with a Connectivity plan where the "ackedDownlinkFrame" feature flag is set, the downlink packet is processed, otherwise the processing is aborted and a specific error code is returned to the AS in the HTTP response.

- FlushDownlinkQueue : (Optional) Empties the device AS DL queue of the device (Boolean). When this parameter is set to FlushDownlinkQueue=1, the AS requests the LRC to purge the AS DL queue of the device prior to adding the DL payload transported by this HTTP POST.
- ValidityTime : (Optional) Associates the AS DL payload with an expiration date (ISO date, or Duration in second) in the device AS DL queue. If the AS DL payload has not yet been sent to the device, the AS DL payload will be discarded by the LRC when the expiration date is reached.

Example (sample)

- curl -H "Content-type:application/x-www-form-urlencoded" -X POST "https://api.thingpark.com/thingpark/lrc/rest/downlink?DevEUI=xxx&FPort=1&Payload=0102030405060708090A&FCntDn=1234&FlushDownlinkQueue=1&ValidityTime=2016-01-11T14:28:00+02:00"
- When this HTTP POST is received by the LRC, the LRC purges the DL queue associated with the 000000000F1D8693 device, and queues the 0102030405060708090A DL payload, waiting for an UL from the device. The queued DL payload is associated with the 2016-01-11T14:28:00+02:00 expiration date. If the DL payload cannot be sent to the device before the expiration date, the DL payload will be discarded by the LRC.

LRC HTTP response codes:

- 200 “Request queued by LRC”: request accepted and queued until the class A device opens Rx slots by sending an uplink. In the case of a class C device, the downlink command will be sent as soon as the LRR base station radio is available and the maximum regulatory Tx duty cycle allows transmission.
- 350 “Invalid DevEUI”
- 350 “Downlink counter value already used. Expected=1238”: the downlink counter value was already used, for instance due to a race condition with another Application server.
- 350 “Downlink counter value increment too large. Expected=1001”: the AS supplied downlink counter value is much larger than the expected downlink counter value and was rejected by the LRC.
- 350 "Confirmed downlink is not authorized for this device": the request for transmission of a confirmed downlink packet was rejected by the LRC due to absence of “ackedDownlinkFrame” feature flag in the Connectivity plan associated to the device.