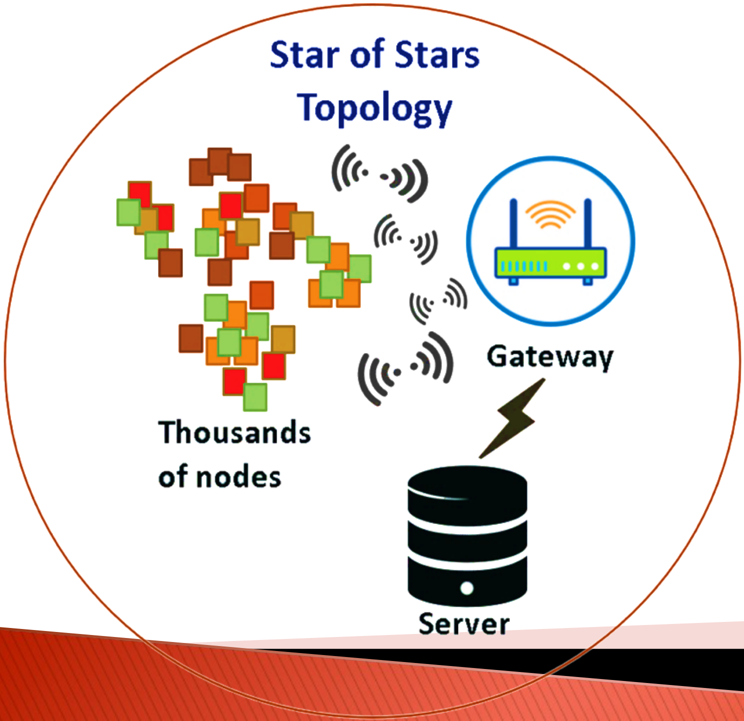
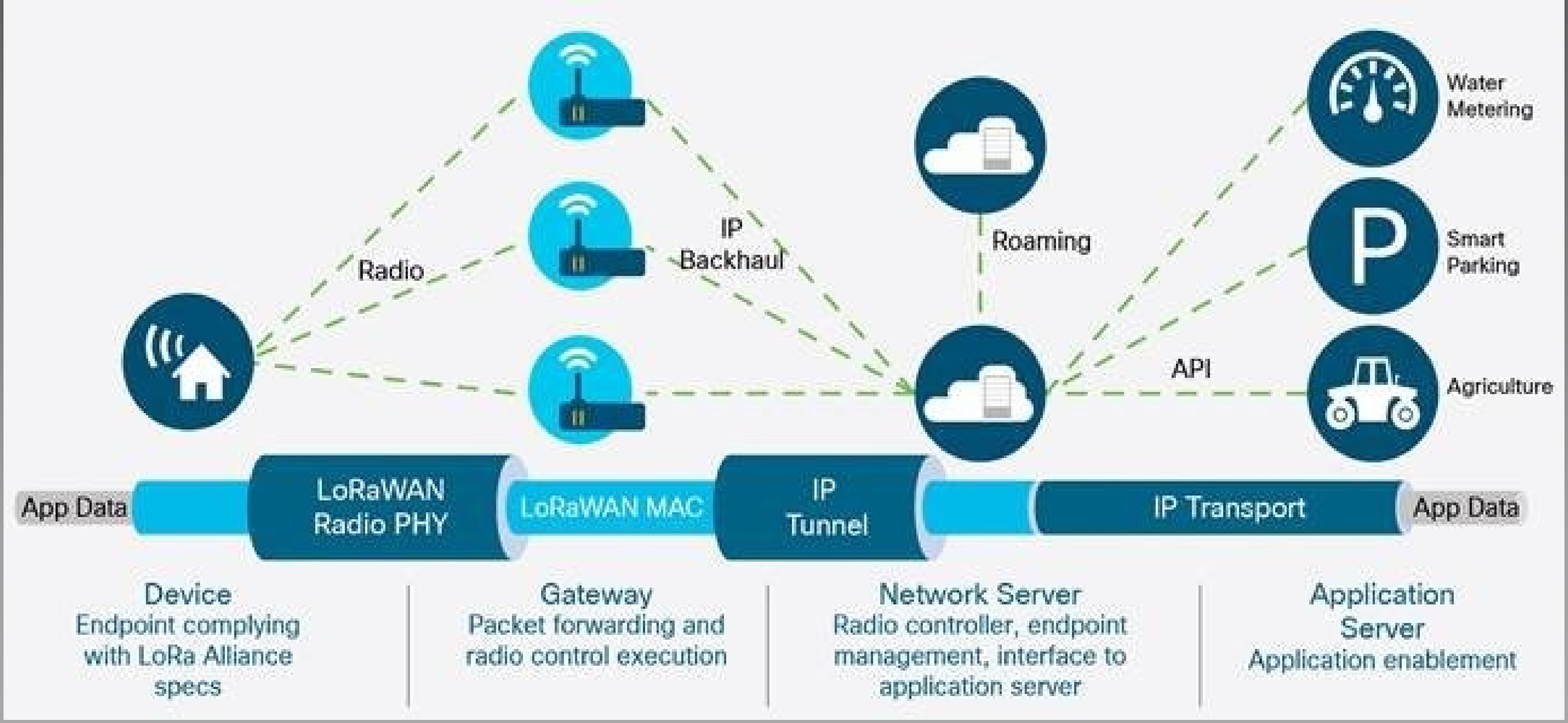
Building a Simple LoRaWAN Network



https://github.com/LoRaWAN-workshop/tutorial

LoRaWAN 1.0.3



Network Encryption

Application Encryption

Data

From: https://www.digitalmatter.com/Solutions/Whitepapers/LoRaWAN-Introduction

ABP (Activation-By-Personalization)

End device

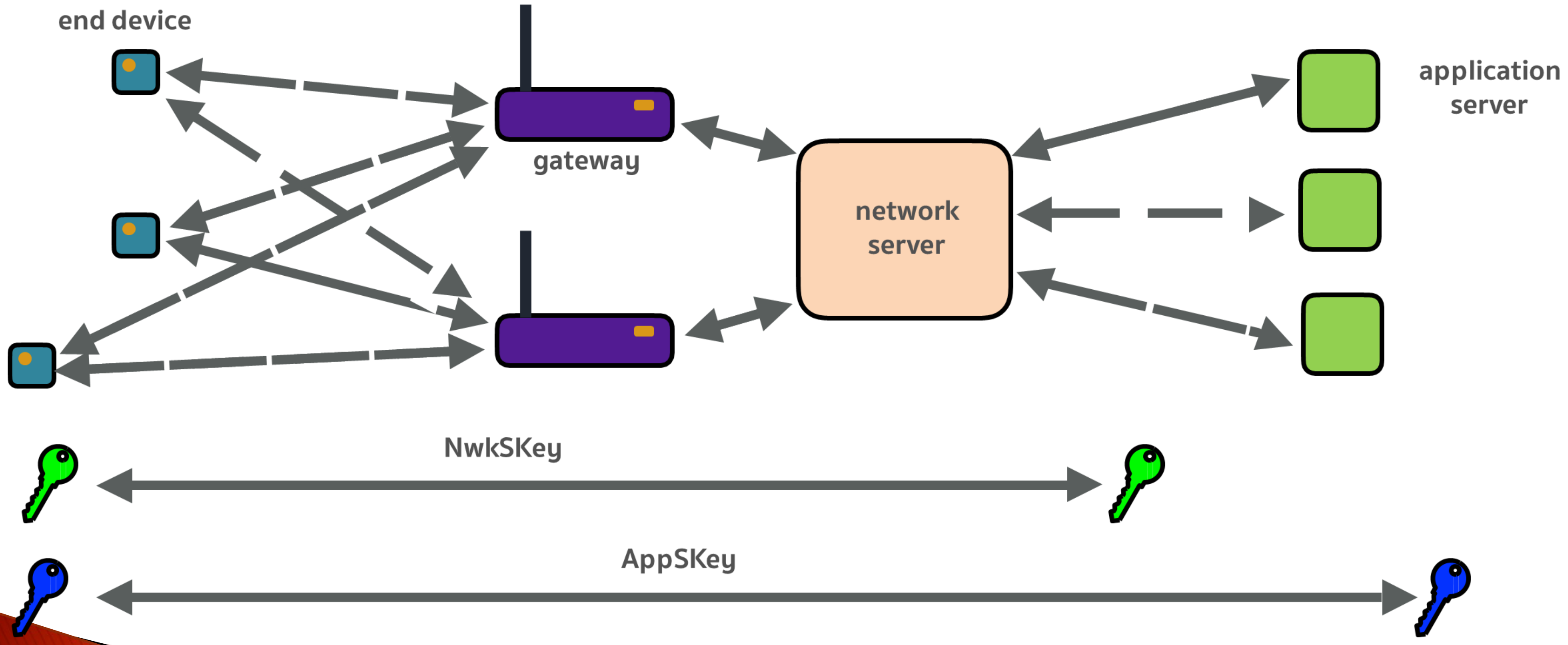
DevAddr
AppSKey
NwkSKey

Network Server

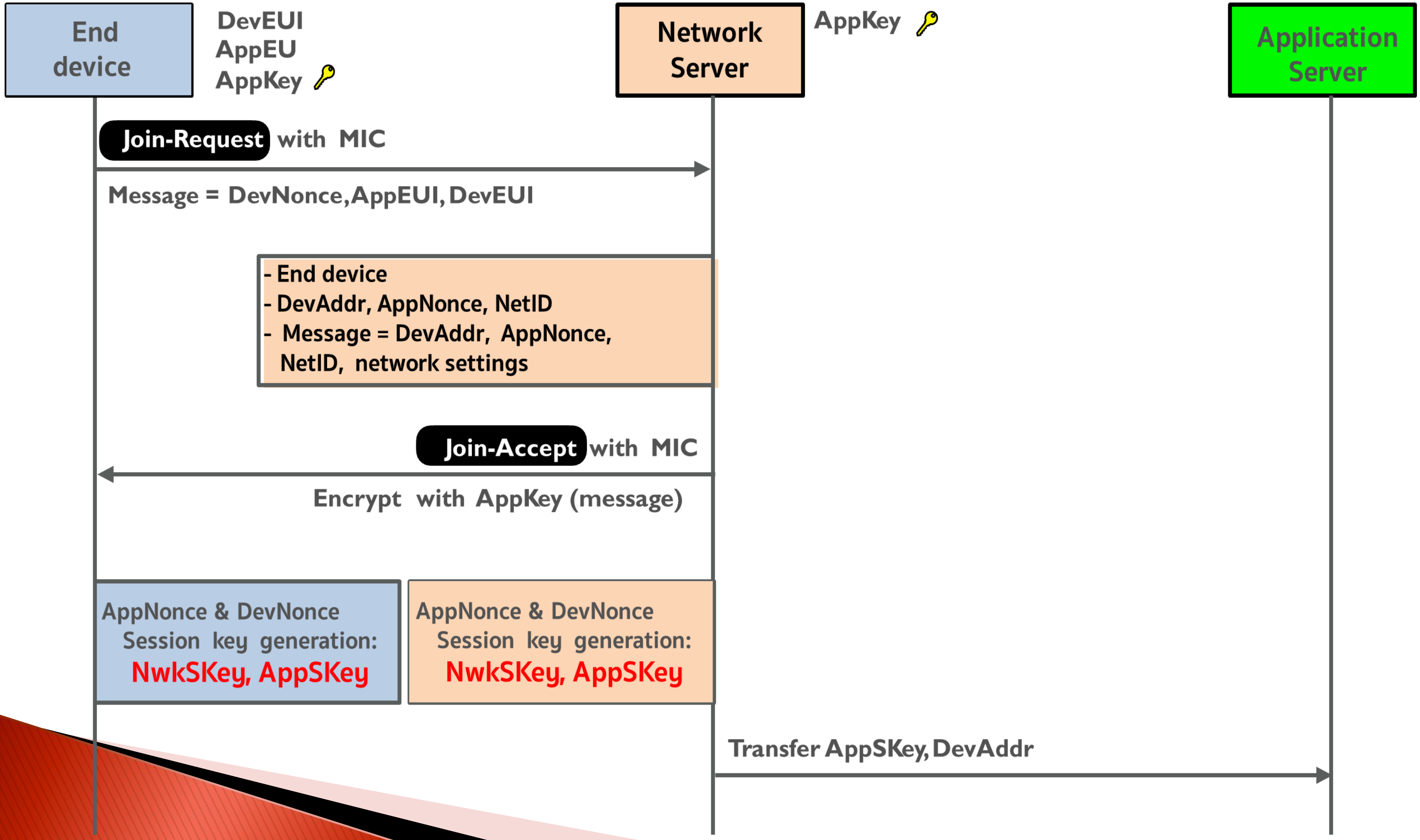
DevAddr
NwkSKey

Application Server

DevAddr
AppSKey



OTAA(Over-The-Air Activation)



End device

DevEUI
AppEU
AppKey

Network Server

AppKey

Application Server

Join-Request with MIC

Message = DevNonce,AppEUI, DevEUI

- End device
- DevAddr, AppNonce, NetID
- Message = DevAddr, AppNonce, NetID, network settings

Join-Accept with MIC

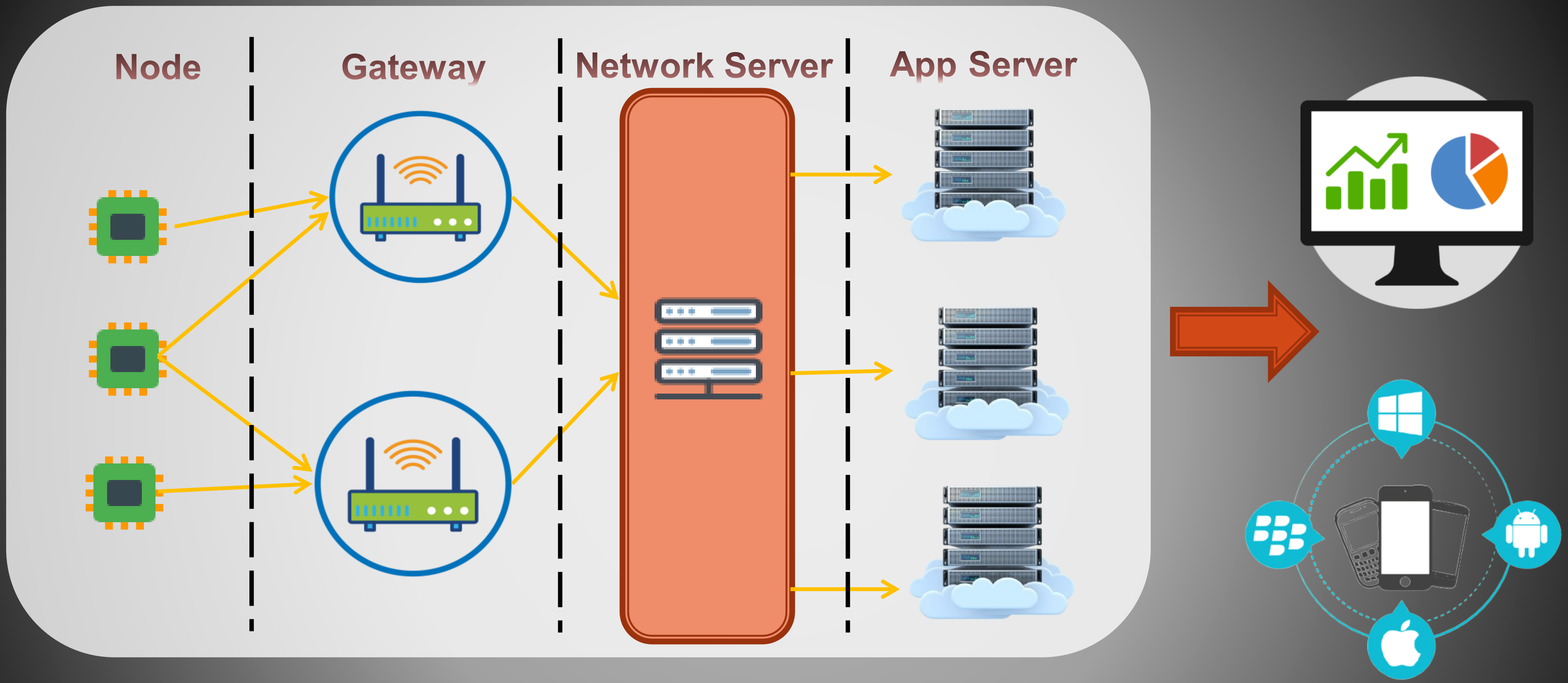
Encrypt with AppKey (message)

AppNonce & DevNonce
Session key generation:
NwkSKey, AppSKey

AppNonce & DevNonce
Session key generation:
NwkSKey, AppSKey

Transfer AppSKey, DevAddr

LoRaWAN 1.0.3



Node

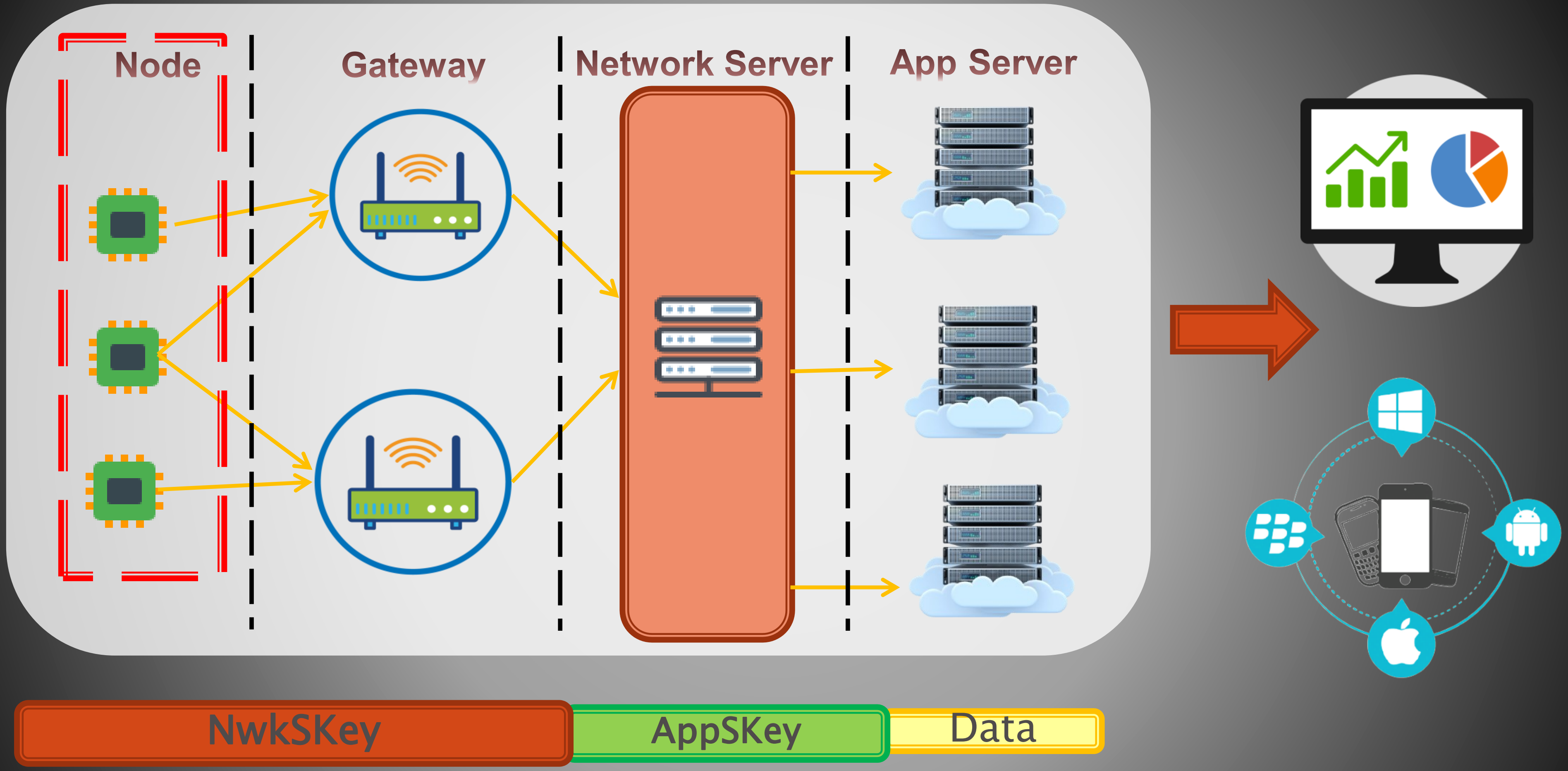
Gateway

Network Server

App Server

NwkSKey AppSKey Data

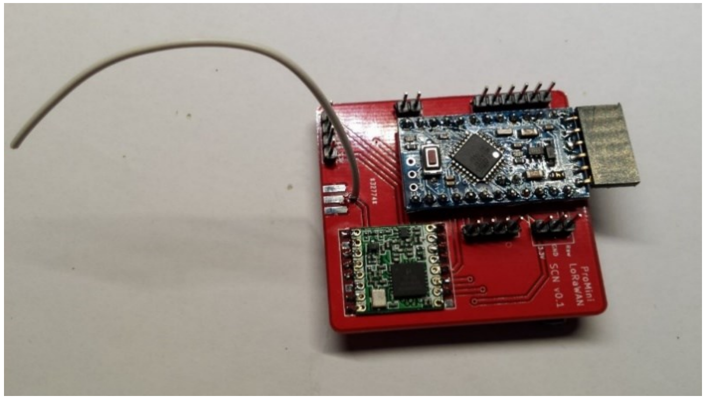
LoRaWAN 1.0.3


Node
Gateway
Network Server
App Server
NwkSKey
AppSKey
Data

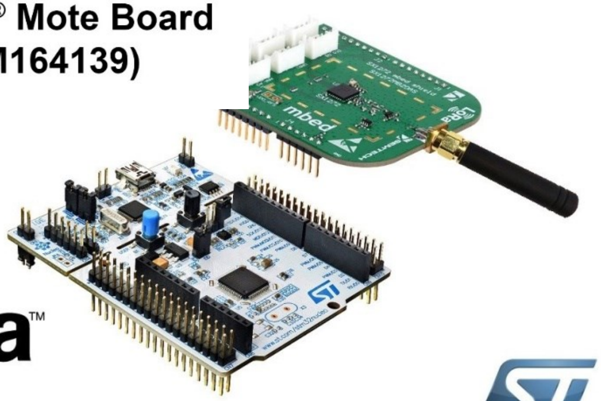
Examples of LoRaWAN Node

LoRa Shield + Arduino

RN2903 LoRa® Mote Board
(Part # DM164139)



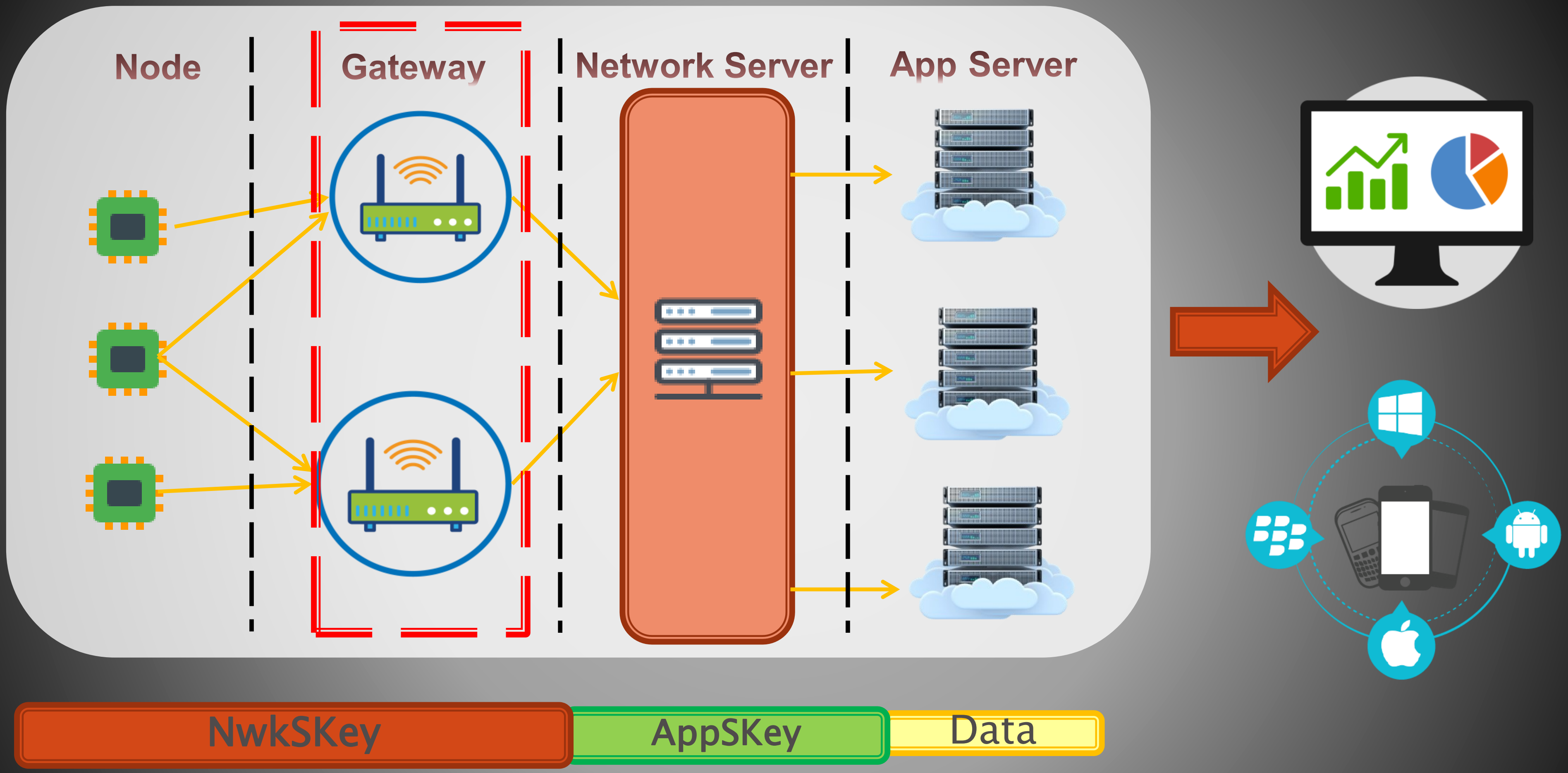
DIY LoRa Shield + Arduino



STM32 Nucleo-64 board+LoRa

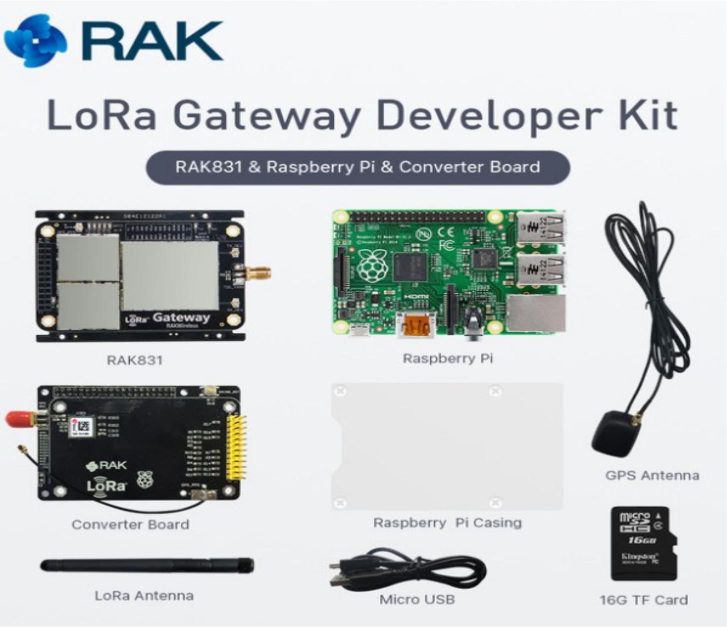
LMIC : https://github.com/matthijskooijman/arduino-lmic-v1.5

LoRaWAN 1.0.3


Node
Gateway
Network Server
App Server
NwkSKey
AppSKey
Data

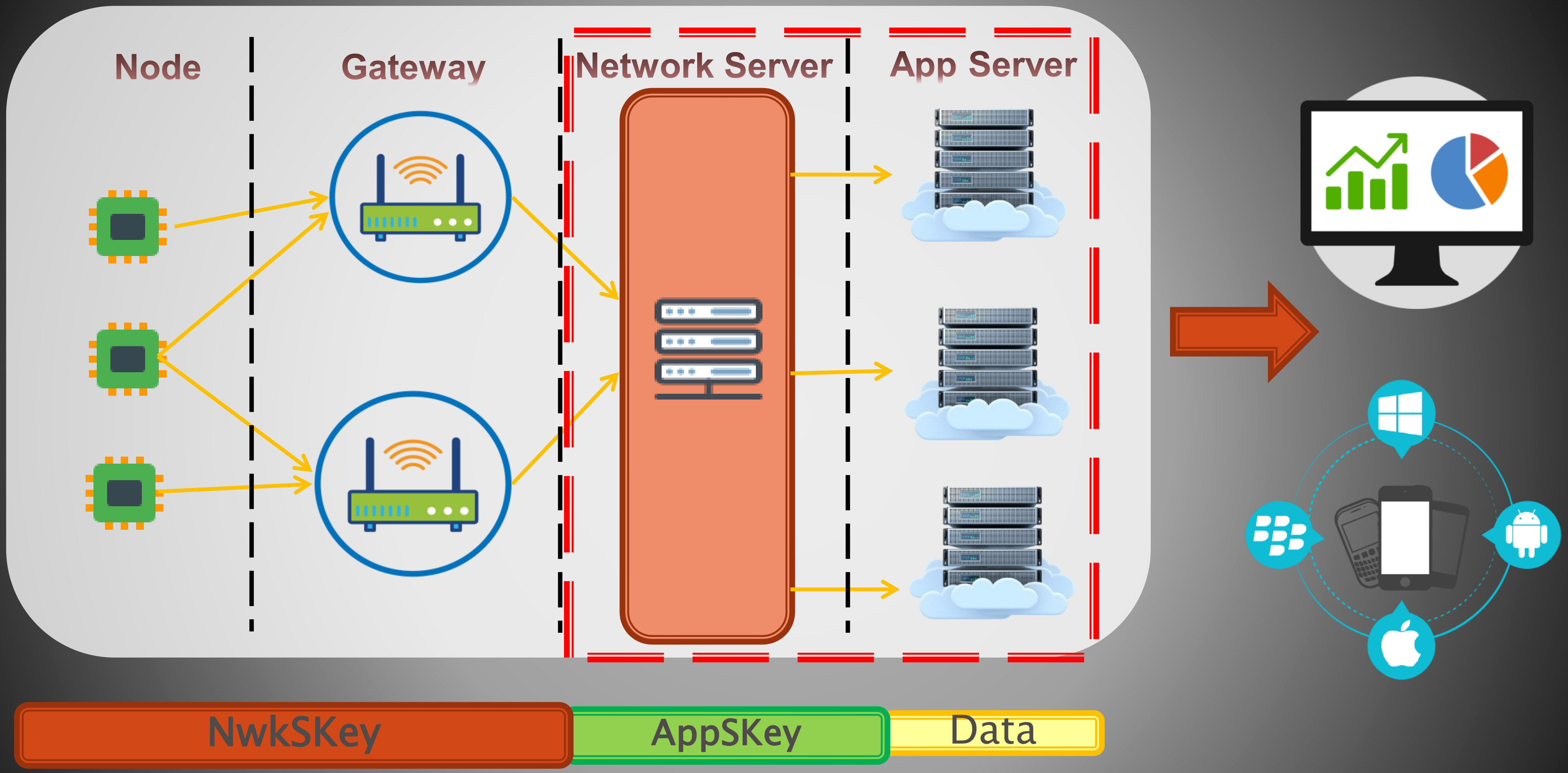
Examples of LoRaWAN Gateway

Gateway with Packet forwarder module



https://github.com/ttn-zh/ic880a-gateway/wiki

LoRaWAN 1.0.3



Node

Gateway

Network Server

App Server

NwkSKey

AppSKey

Data

Examples of LoRaWAN Server

Private:

kerlink
communication is everything

ChirpStack

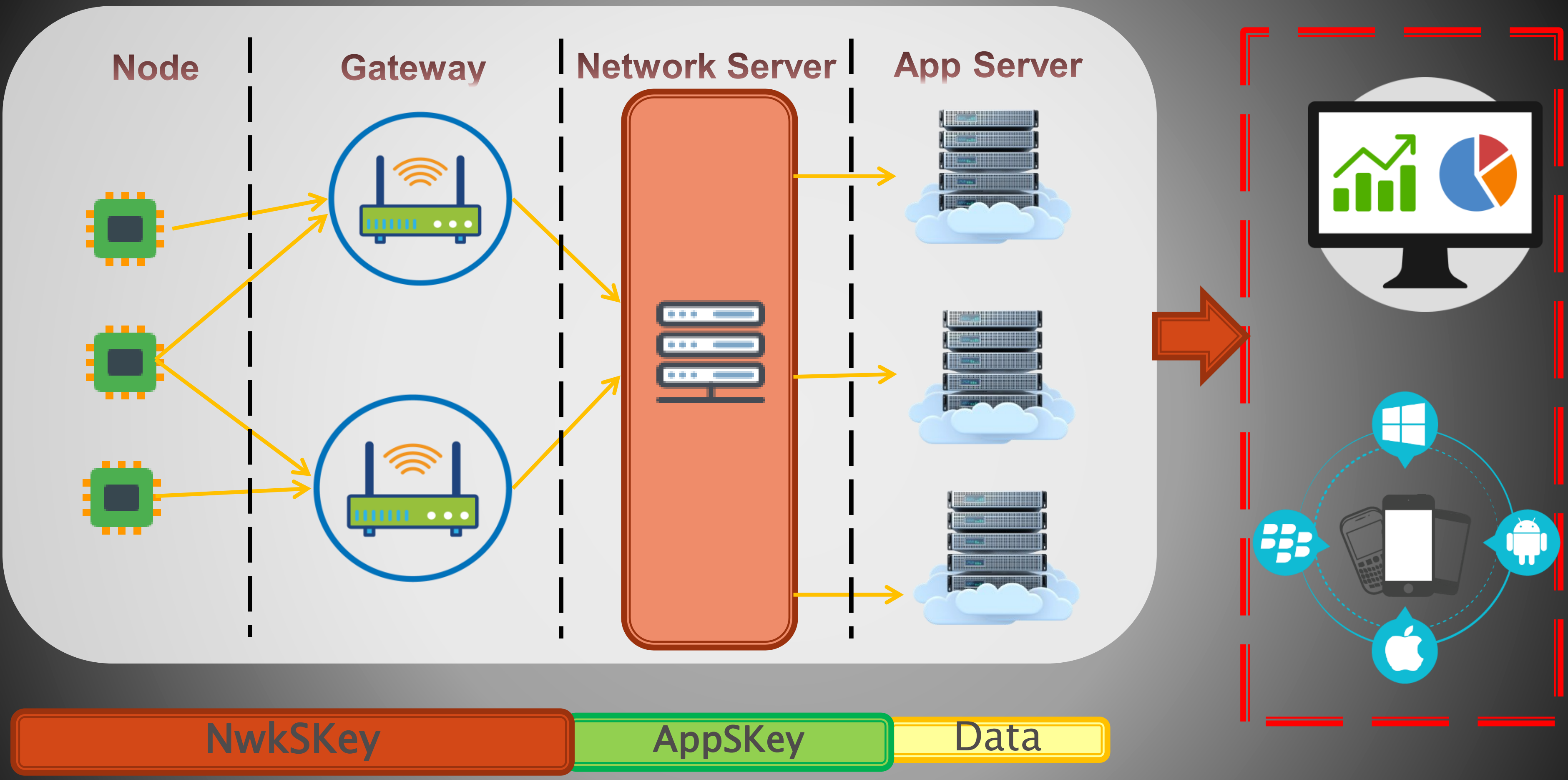
Public

Actility

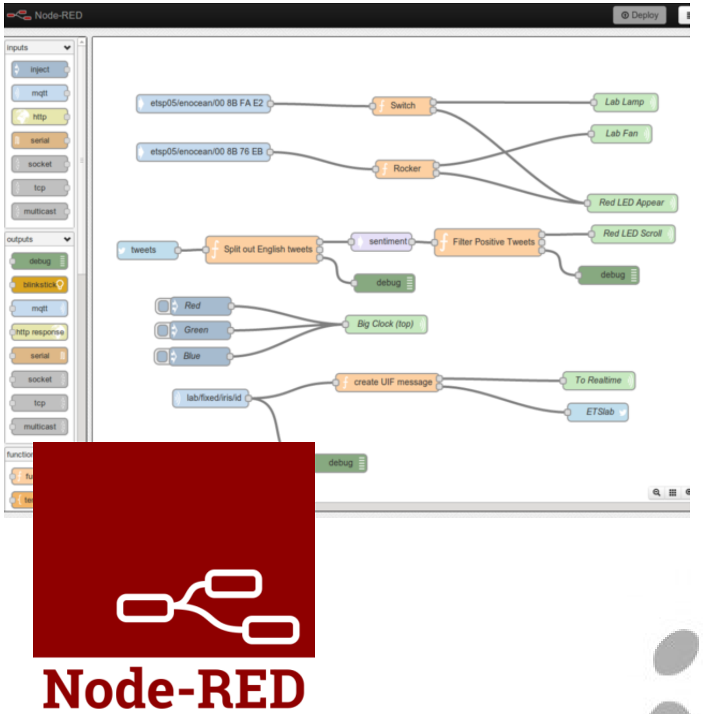
THE THINGS NETWORK

LORIOT

LoRaWAN 1.0.3


Node
Gateway
Network Server
App Server
NwkSKey
AppSKey
Data

Example of Dashboards



Node-RED

Cayenne myDevices

TagoIO

ubidots

Procedures

Our LoRaWAN Server

THE THINGS NETWORK

Website:

https://www.thethingsnetwork.org/

Username: LoRaWAN-workshop

Password: 654321

Connect Gateway with THE THINGS NETWORK

Website:
https://www.thethingsnetwork.org/

Username: LoRaWAN-workshop
Password: 654321

Gateway EUI (MAC Address):
7276FF004501022D

Develop Node

- Hardware
  - Arduino Uno with RFM95
- Software
  - LMIC Library
- Complier
  - Arduino IDE

Develop Node

- Hardware
  - Arduino Uno with RFM95
- Software
  - LMIC Library

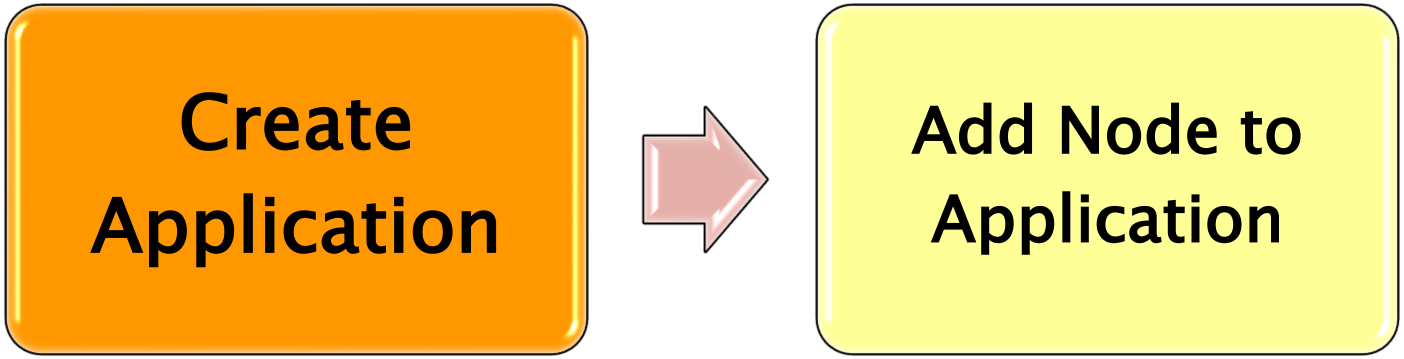
  https://github.com/LoRaWAN-workshop/tutorial
- Complier
  - Arduino IDE

Register Node with THE THINGS NETWORK

Website:
https://www.thethingsnetwork.org/

Username: LoRaWAN-workshop
Password: 654321

Register Node with THE THINGS NETWORK



Let’s Change some Parameters

- SF

  LMIC_setDrTxpow(DR_SFx,14);

  - DR_SF7 – DR_SF12

- Unconfirmed/Confirmed

  LMIC_setTxData2(1, mydata, sizeof(mydata)-1, x);

  - Unconfirmed = 0, Confirmed =1

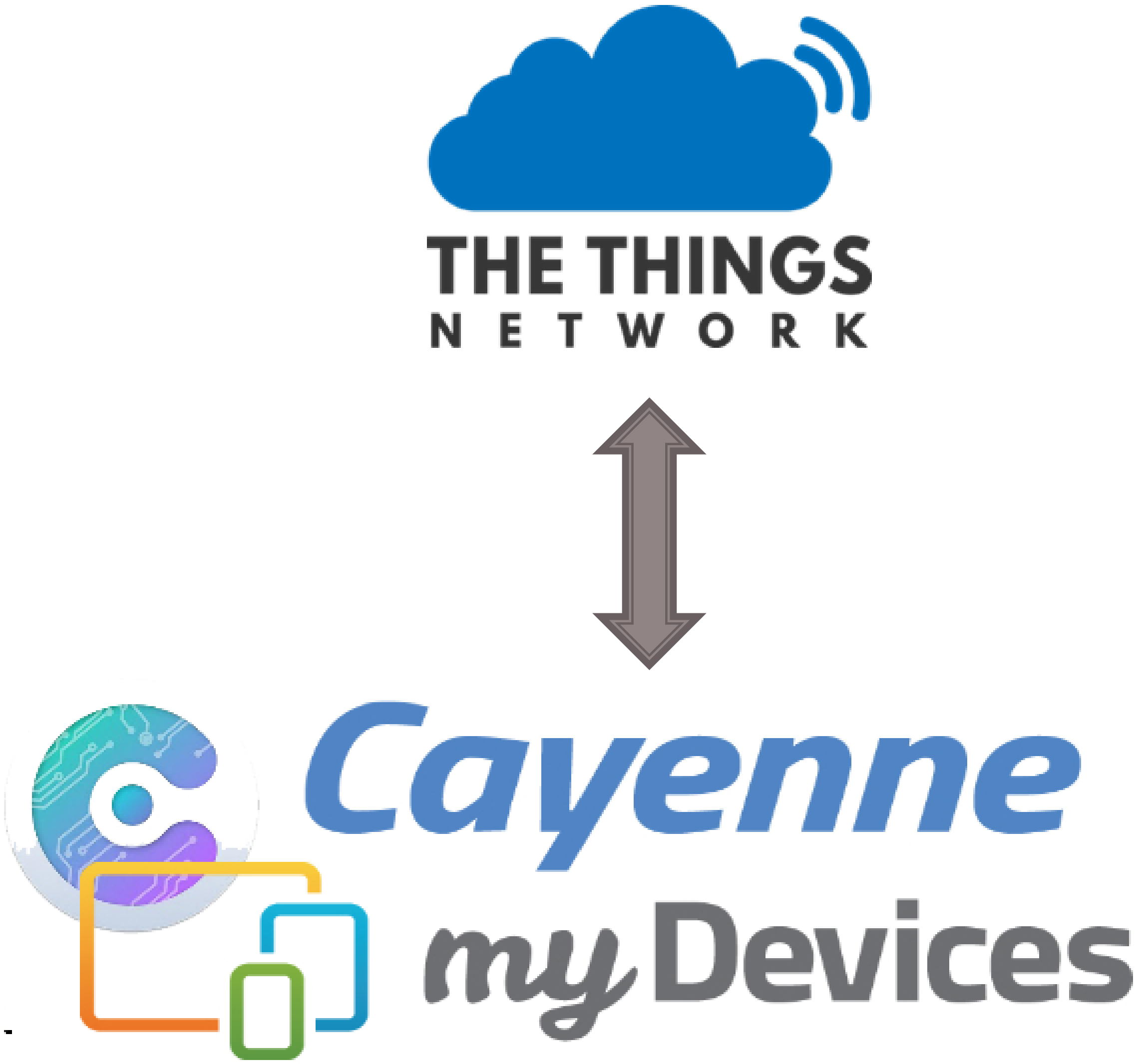
AS923

Data Rate	Settings	Bit Rate (bit/s)	Max MAC Payload Length (byte)
0	SF12/BW125kHz	250	59
1	SF11/BW125 kHz	440	59
2	SF10/BW125 kHz	980	59
3	SF9/BW125 kHz	1760	123
4	SF8/BW125 kHz	3125	230
5	SF7/BW125 kHz	5470	250
6	SF7/BW250 kHz	11000	250

Duration

- Maximum period = Airtime/dutycycle

Integration with Dashboard


THE THINGS NETWORK
Cayenne
myDevices

Cayenne myDevices

Cayenne is a solution for building IoT applications based on known platforms

- Arduino
- Raspberry pi
- ESP8266
- Serial devices
- Wifi
- Lora devices →
- Mqtt client API

CayenneLPP

http://mydevices.com/cayenne/docs/lora/#lora-cayenne-low-power-payload

Cayenne

myDevices

Requirement:

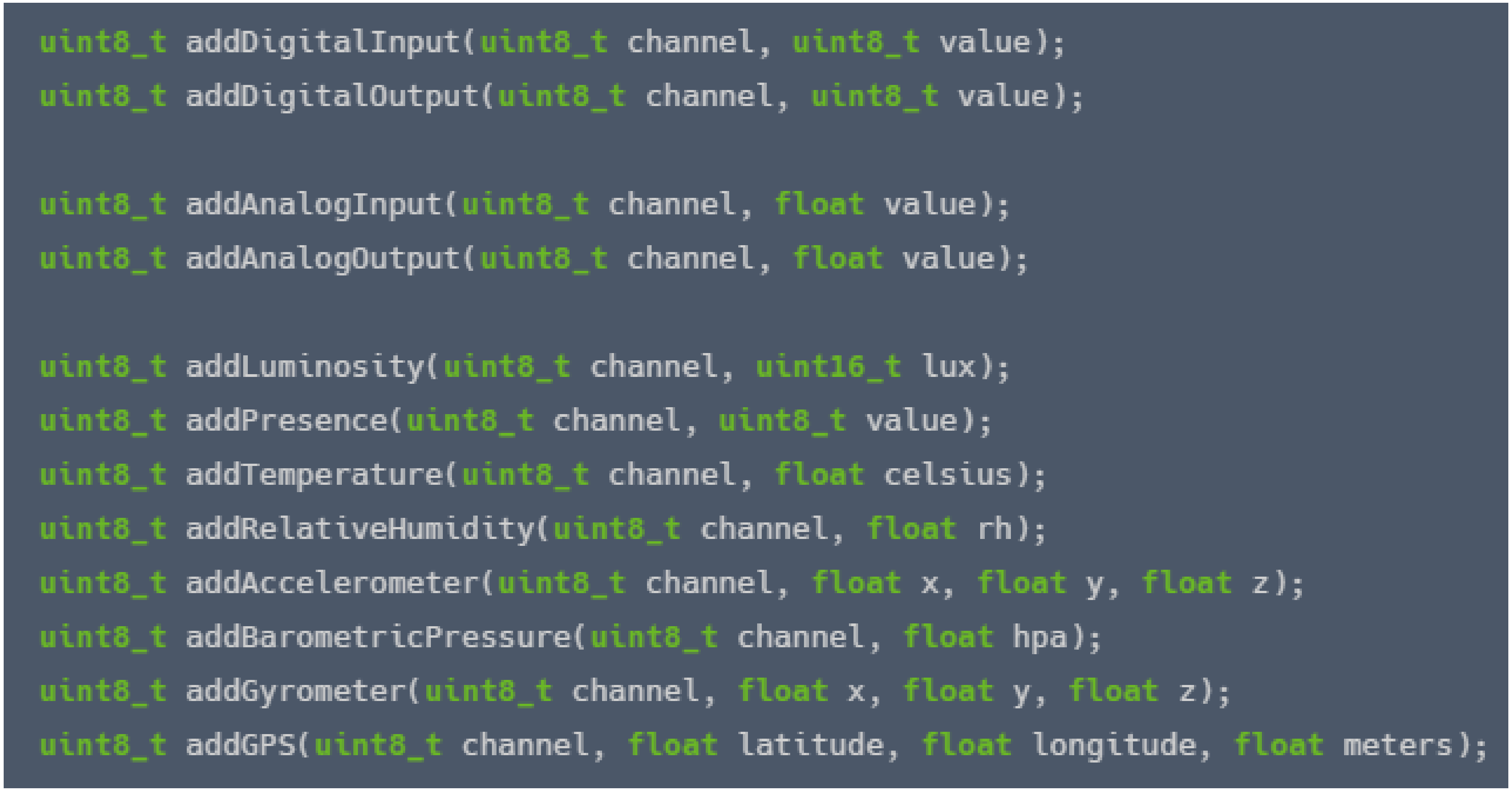
1. Node with CaynenneLPP Class
https://www.thethingsnetwork.org/docs/devices/arduino/api/cayennelpp.html

2. Integrate Application on TTN with Cayenne
(Integration Page)

3. Sign up and Create Dashboard at Cayenne
LoRa -> The Things Network -> Cayenne LPP

Cayenne

myDevices



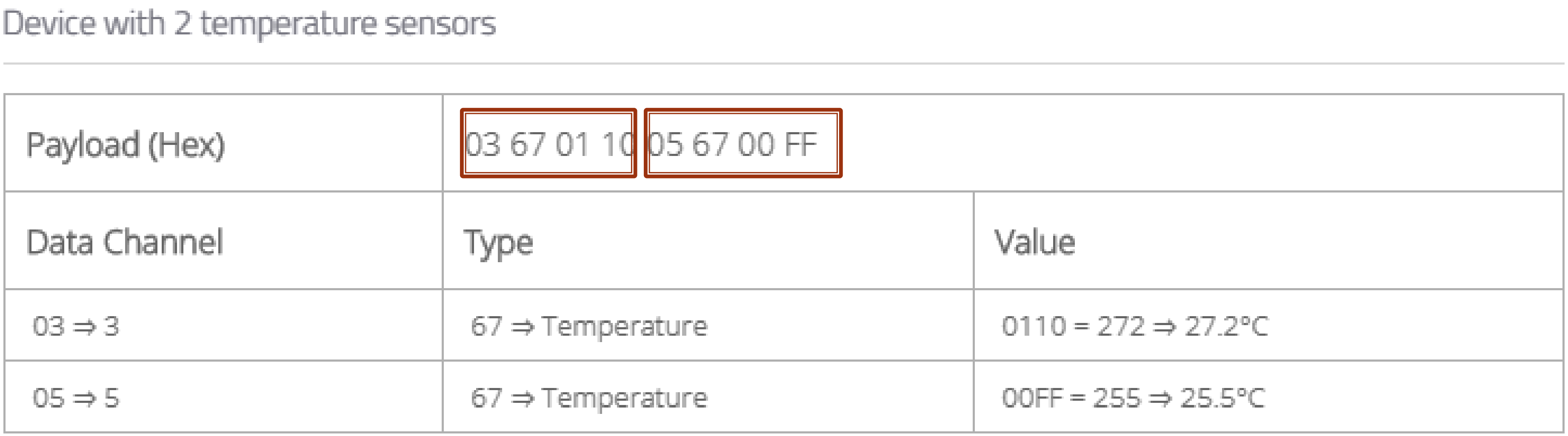
```
uint8_t addDigitalInput(uint8_t channel, uint8_t value);
uint8_t addDigitalOutput(uint8_t channel, uint8_t value);

uint8_t addAnalogInput(uint8_t channel, float value);
uint8_t addAnalogOutput(uint8_t channel, float value);

uint8_t addLuminosity(uint8_t channel, uint16_t lux);
uint8_t addPresence(uint8_t channel, uint8_t value);
uint8_t addTemperature(uint8_t channel, float celsius);
uint8_t addRelativeHumidity(uint8_t channel, float rh);
uint8_t addAccelerometer(uint8_t channel, float x, float y, float z);
uint8_t addBarometricPressure(uint8_t channel, float hpa);
uint8_t addGyrometer(uint8_t channel, float x, float y, float z);
uint8_t addGPS(uint8_t channel, float latitude, float longitude, float meters);
```

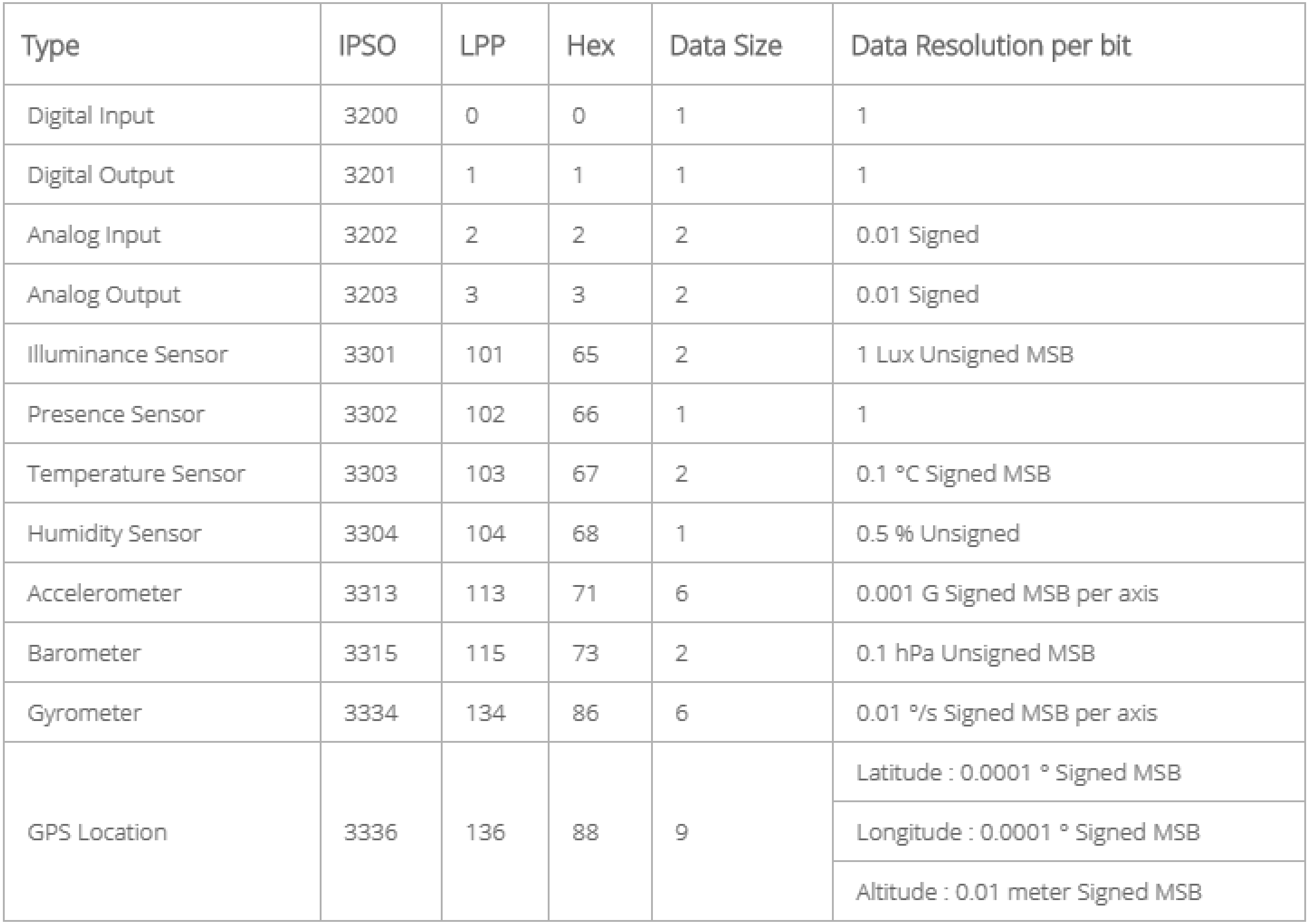
Cayenne

myDevices



Device with 2 temperature sensors

Payload (Hex)	03 67 01 10 05 67 00 FF	
Data Channel	Type	Value
03 ⇒ 3	67 ⇒ Temperature	0110 = 272 ⇒ 27.2°C
05 ⇒ 5	67 ⇒ Temperature	00FF = 255 ⇒ 25.5°C



Type	IPSO	LPP	Hex	Data Size	Data Resolution per bit
Digital Input	3200	0	0	1	1
Digital Output	3201	1	1	1	1
Analog Input	3202	2	2	2	0.01 Signed
Analog Output	3203	3	3	2	0.01 Signed
Illuminance Sensor	3301	101	65	2	1 Lux Unsigned MSB
Presence Sensor	3302	102	66	1	1
Temperature Sensor	3303	103	67	2	0.1 °C Signed MSB
Humidity Sensor	3304	104	68	1	0.5 % Unsigned
Accelerometer	3313	113	71	6	0.001 G Signed MSB per axis
Barometer	3315	115	73	2	0.1 hPa Unsigned MSB
Gyrometer	3334	134	86	6	0.01 °/s Signed MSB per axis
GPS Location	3336	136	88	9	Latitude : 0.0001 ° Signed MSB
					Longitude : 0.0001 ° Signed MSB
					Altitude : 0.01 meter Signed MSB

Q&A

Workshop Evaluation

https://qrgo.page.link/UFwtJ

THANK YOU

IEEE CONSUMER ELECTRONICS SOCIETY

IoT TENTRACK

KMUTT

Internet of Things eXpedition