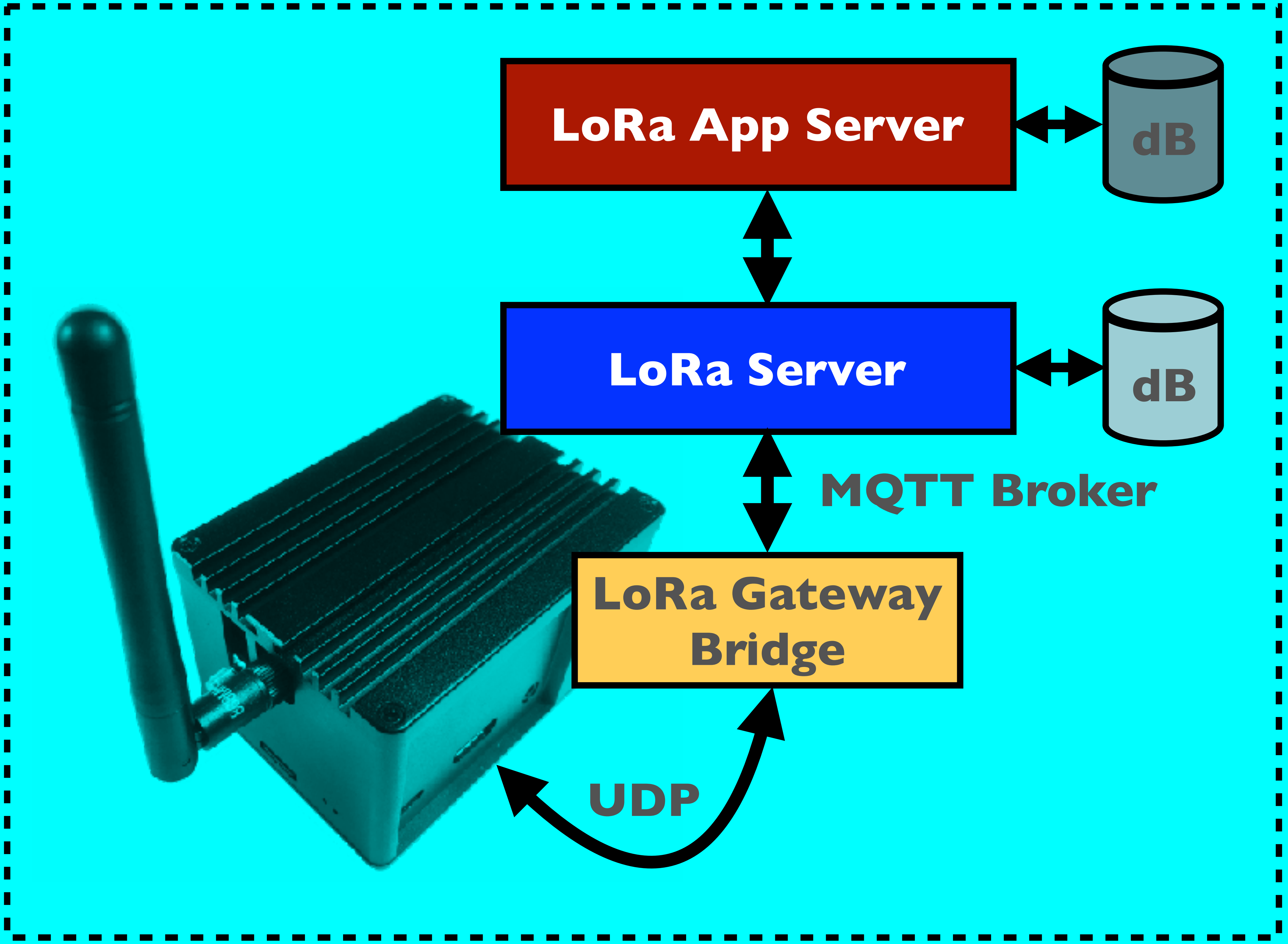
mobilefish.com

LORA / LORAWAN TUTORIAL 31

LoRa Server Project



v1.0.1

INTRO

- In this tutorial I will demonstrate how to install and configure the open source LoRaWAN network server called LoRaServer.

WHAT IS THE LORA SERVER PROJECT

- The LoRa Server project provides open source components for building LoRaWAN networks.

- The author of this project is Orne Brocaar, he started the project end 2015 as a side project and the project has grown significantly thanks to sponsoring.

- The LoRa Server project consists of four major components: LoRa Gateway Bridge, LoRa Server, LoRa App Server and LoRa Geo Server. The LoRa Geo Server will not be discussed in this tutorial.

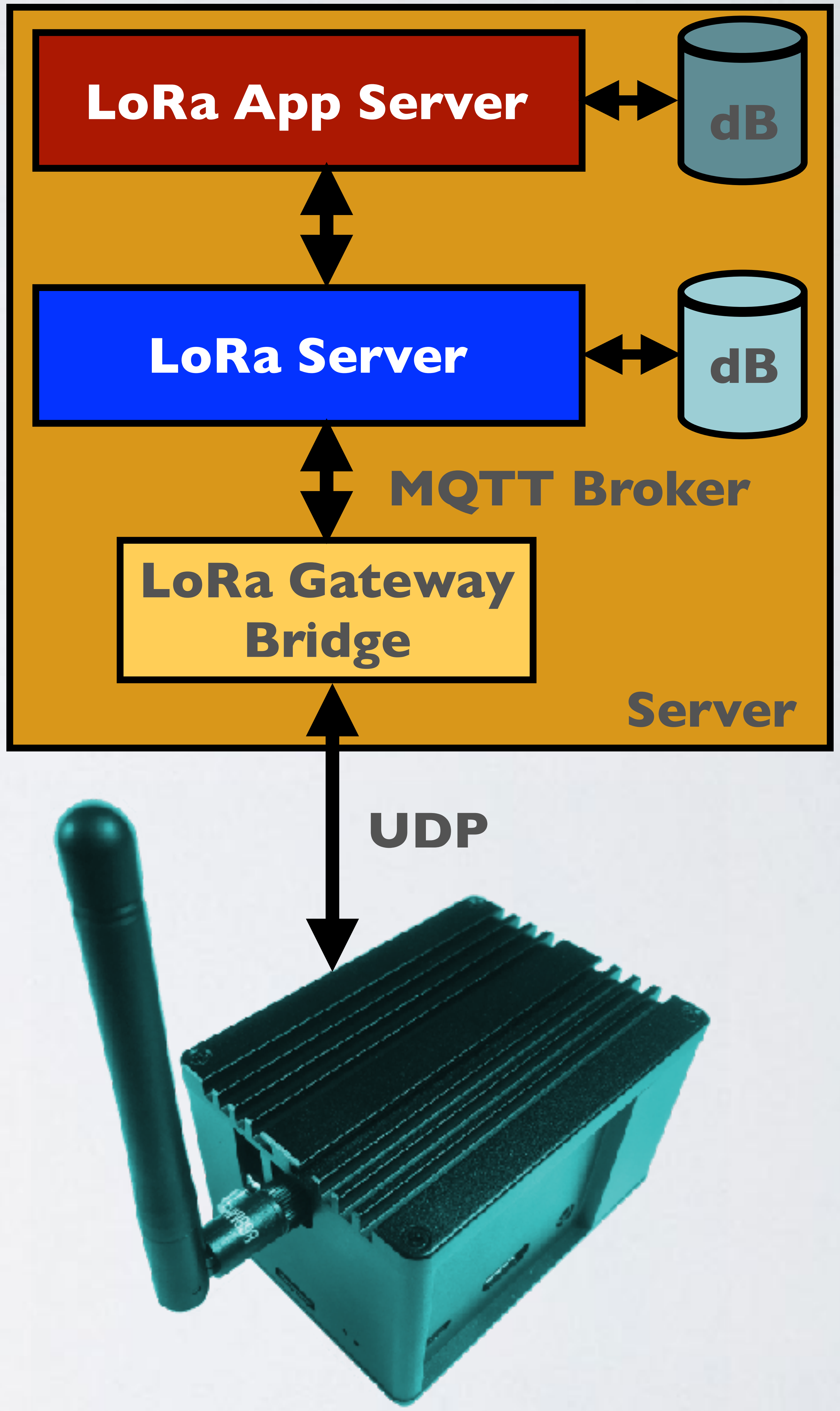
- All components are licensed under the MIT license and can be used for commercial purposes.

LORA SERVER PROJECT

- The **LoRa Gateway Bridge** abstracts the packet forwarder UDP protocol data into JSON and sends it to the LoRa Server over MQTT.

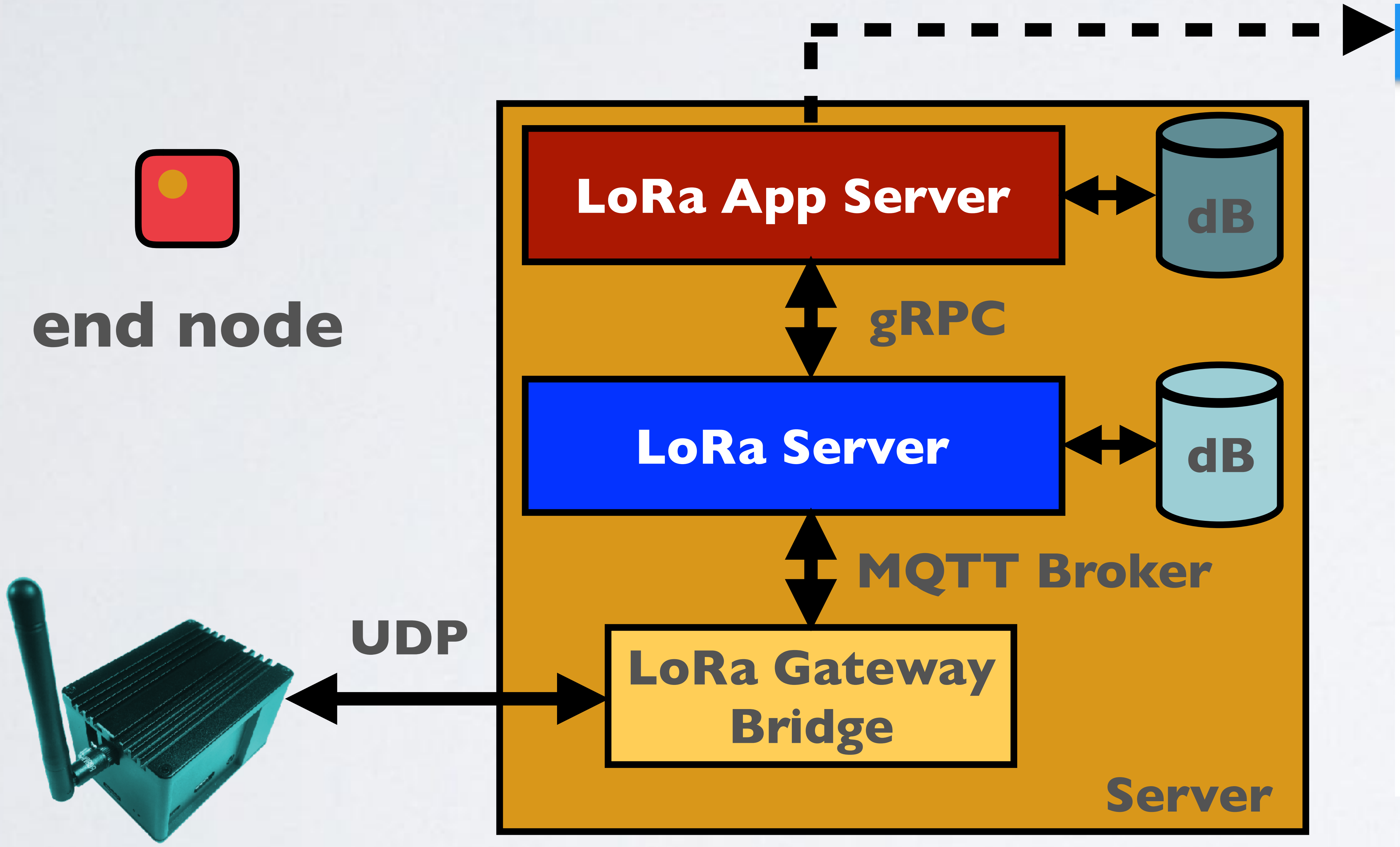
- The **LoRa Server** is the LoRaWAN network server. It de-duplicates and handles the received uplink frames from the gateway(s), handles the LoRaWAN mac layer and schedules the downlink data transmissions.

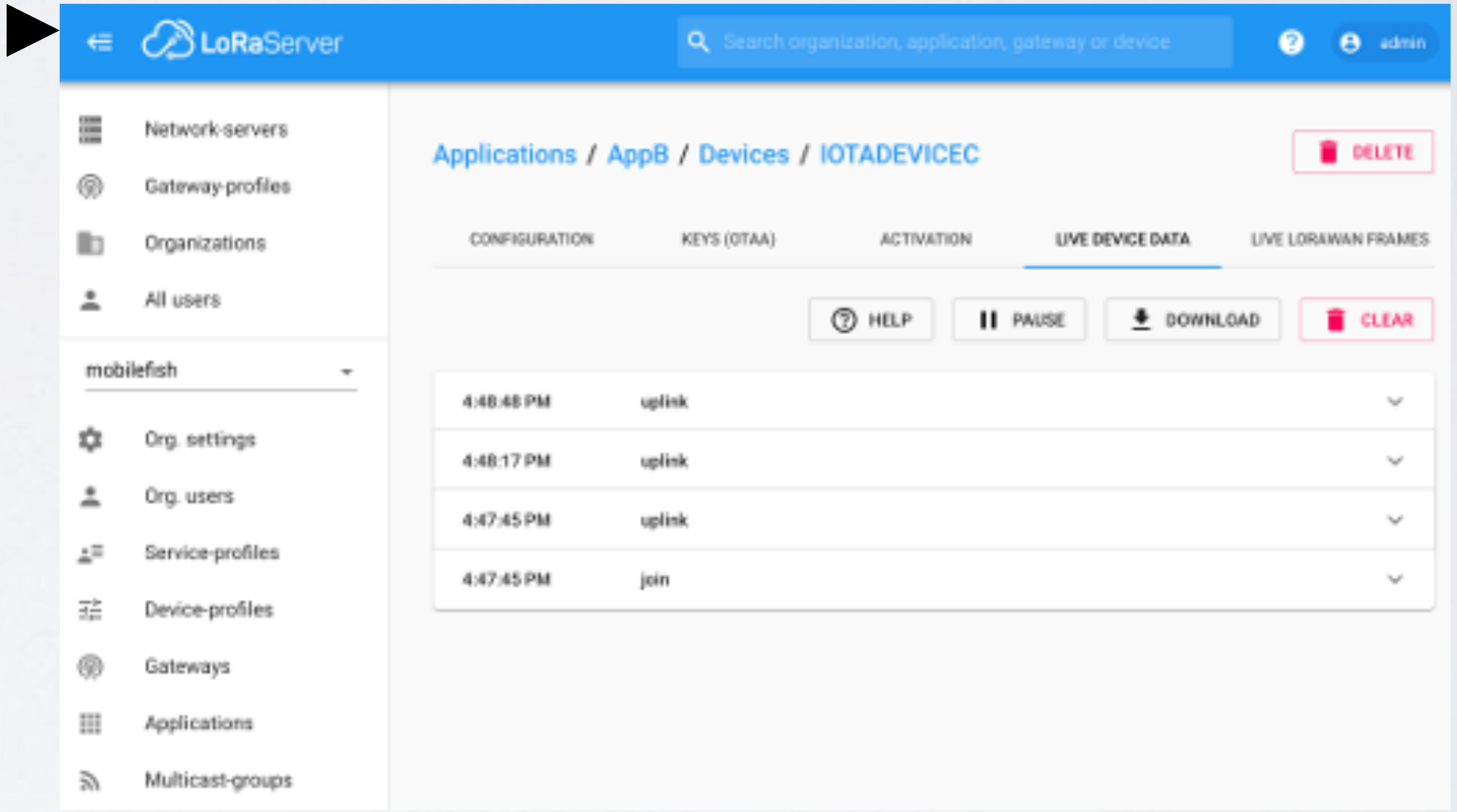
- The **LoRa App Server** is the LoRaWAN Application Server and handles the join-requests, encryption of application payloads and offers a RESTful JSON API, gRPC API or MQTT for external services.



LORA SERVER PROJECT

- The **LoRa App Server** has a web interface to manage users, organisations, applications and devices. It also allows you to see the received sensor data.





Applications / AppB / Devices / IOTADEVICEC

```
4:48:48 PM   uplink
4:48:17 PM   uplink
4:47:45 PM   uplink
4:47:45 PM   join
```

LORA SERVER PROJECT

- Many users are sending their sensor data to The Things Network.
- Now you can create your own private LoRaWAN network by using the LoRa Server components.

LORA SERVER PROJECT INFORMATION

- More information about the LoRa Server project:
  https://www.loraserver.io

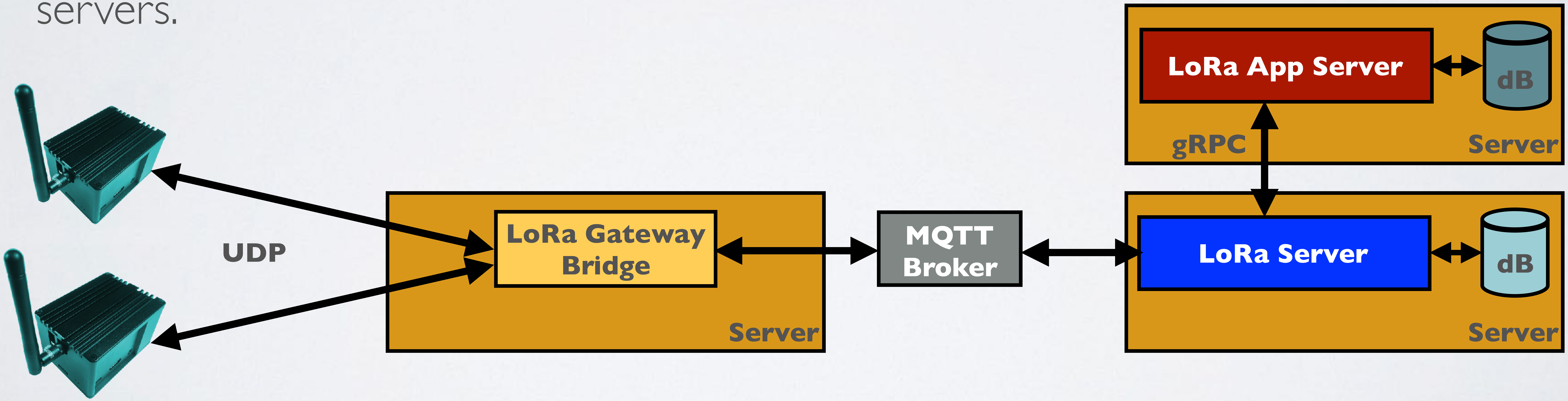
- All LoRa Server project code:
  https://github.com/brocaar

- LoRa Server forum:
  https://forum.loraserver.io

LORA SERVER COMPONENTS

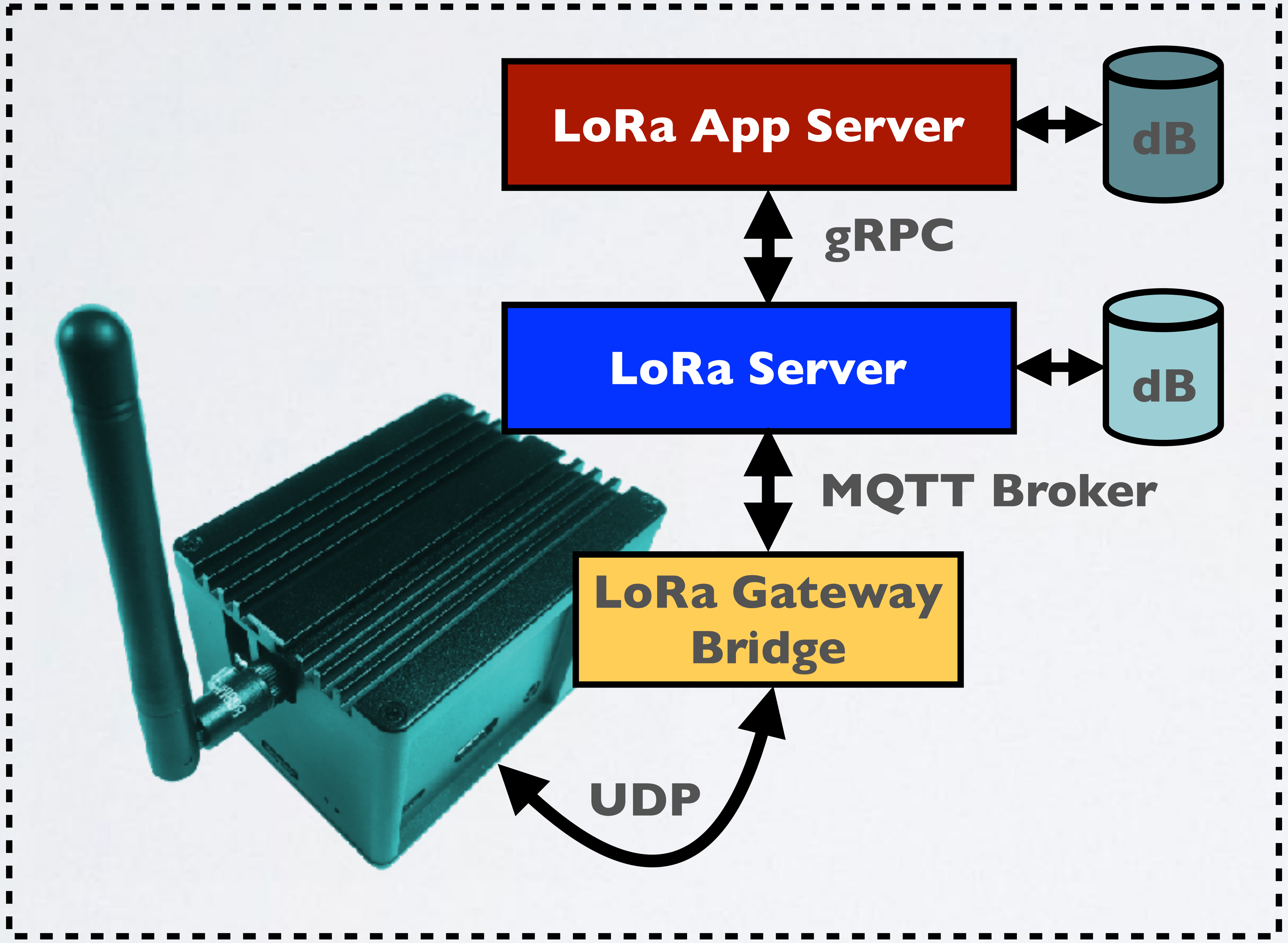
- The LoRa Server and the LoRa App Server each requires their own PostgreSQL database.

- There are many configurations of these components possible, for example:
  - All LoRa Server components and their dependencies can be installed on their own servers.



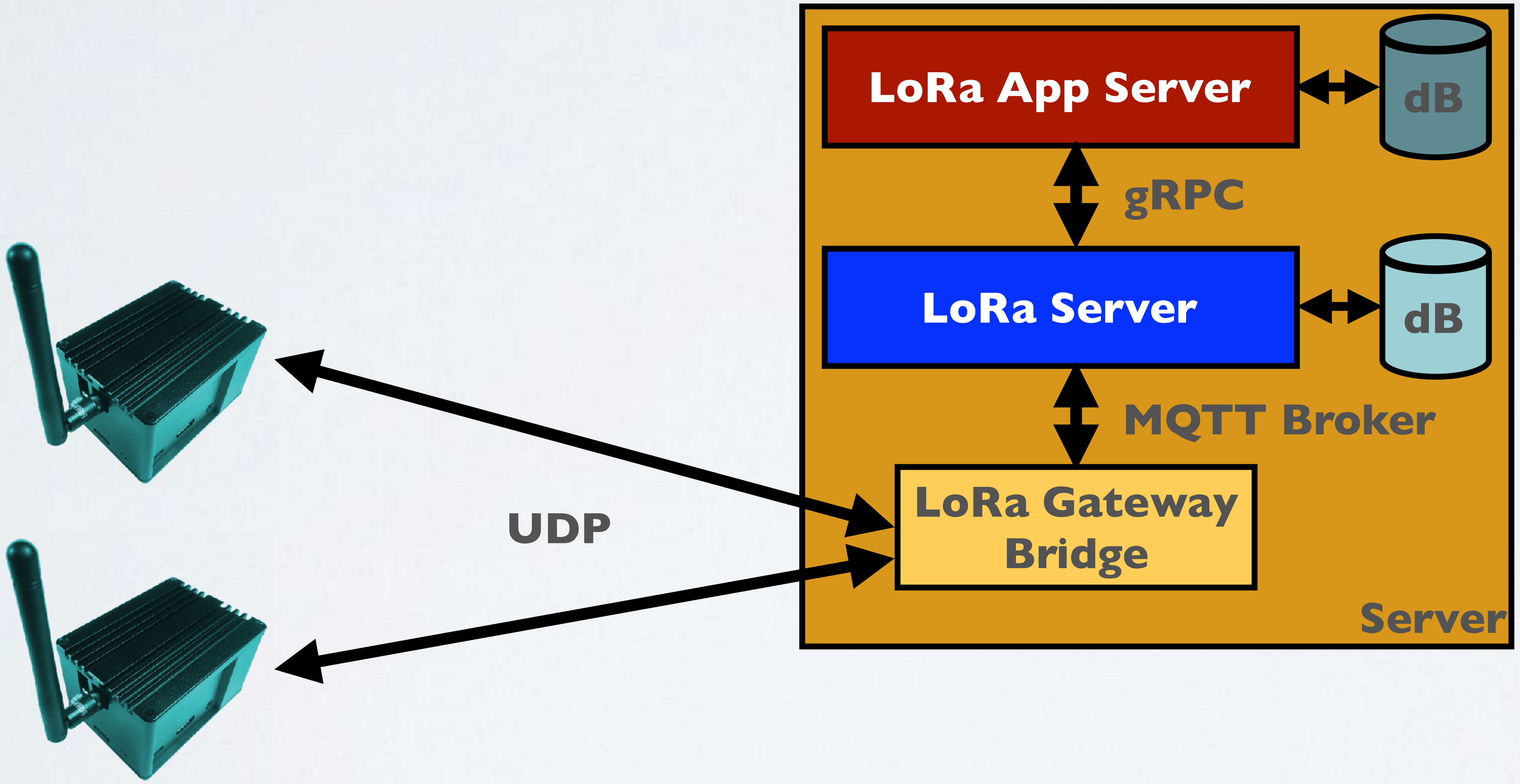
LORA SERVER COMPONENTS

- All LoRa Server components and their dependencies can be installed on the gateway itself, which I will demonstrate in this video.



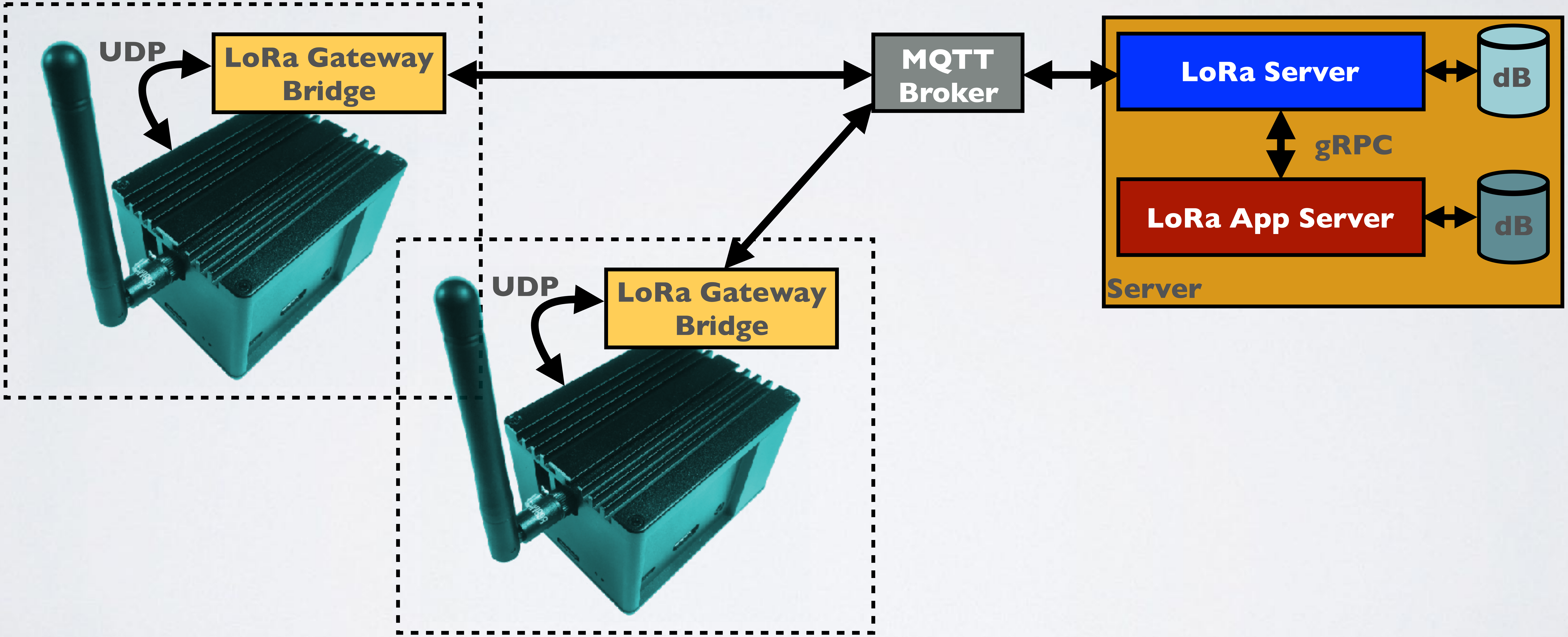
LORA SERVER COMPONENTS

- All LoRa Server components and their dependencies can be installed on a single server instance.

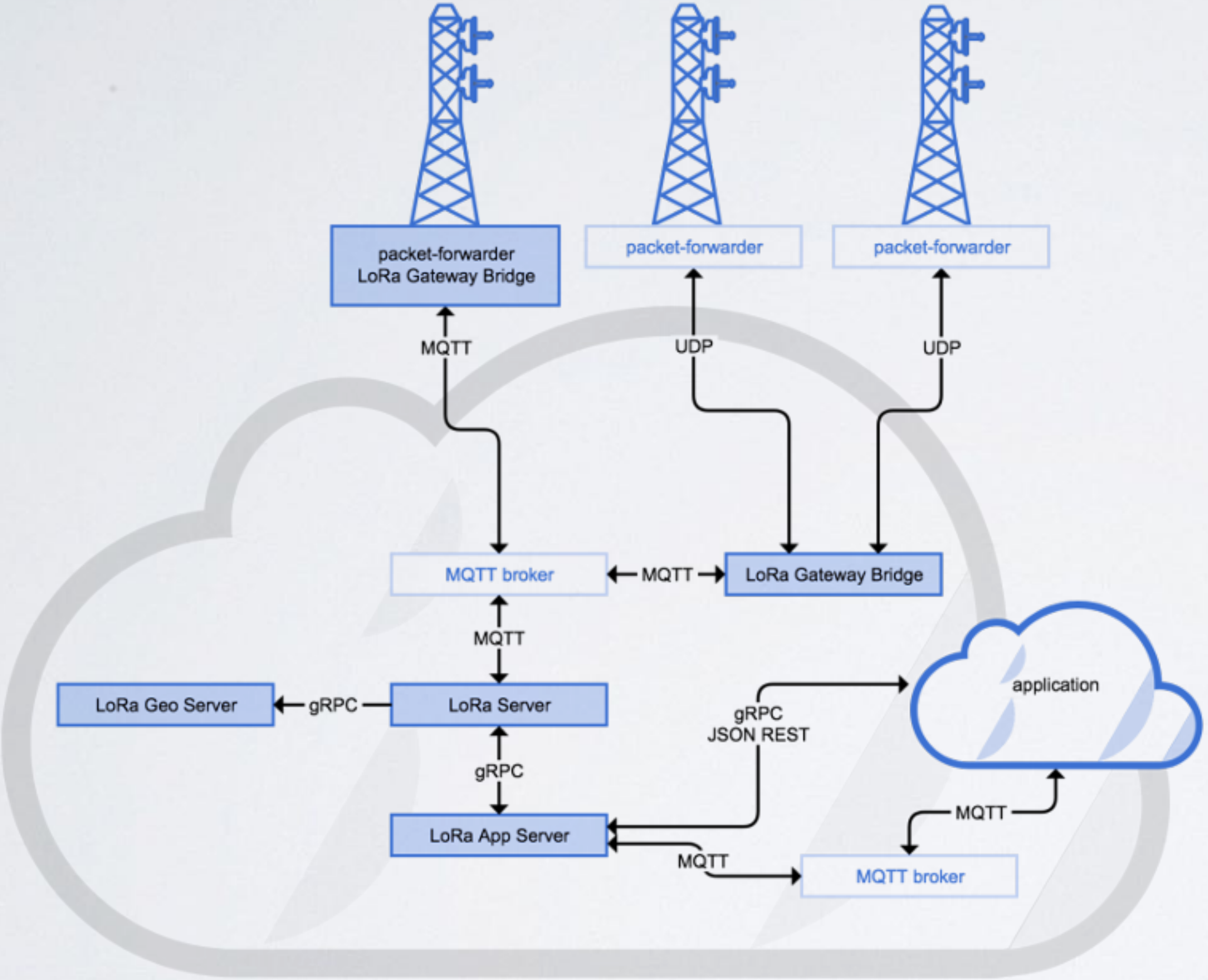


LORA SERVER COMPONENTS

- LoRa Gateway Bridge can be installed on the gateway itself, the other LoRa Server components and their dependencies can be installed on a single server instance.



LORA SERVER ARCHITECTURE

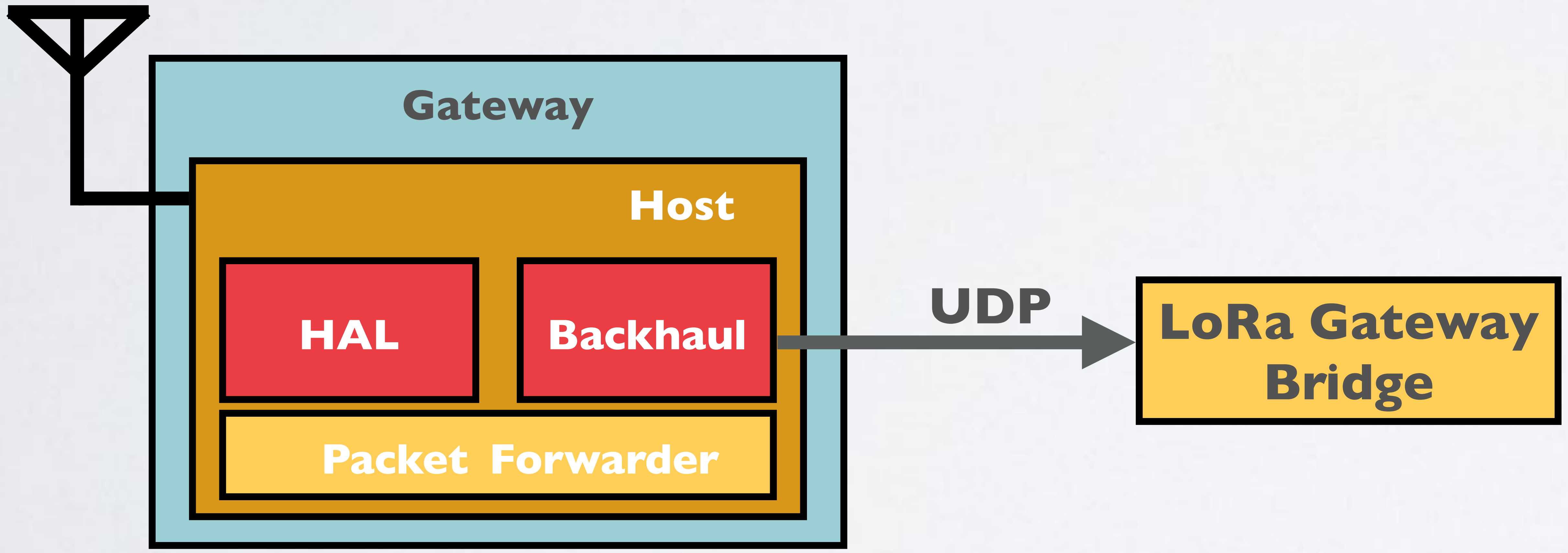


Source:
https://github.com/brocaar/loraserver

HOW IT WORKS

- The gateway has the Semtech packet forwarder installed. The global_conf.json and local_conf.json files are used to configure the packet forwarder.

- In the global_conf.json file the Lora Gateway Bridge server address and port (1700) are set. The packet forwarder has now all the information it needs to send data to the Lora Gateway Bridge.



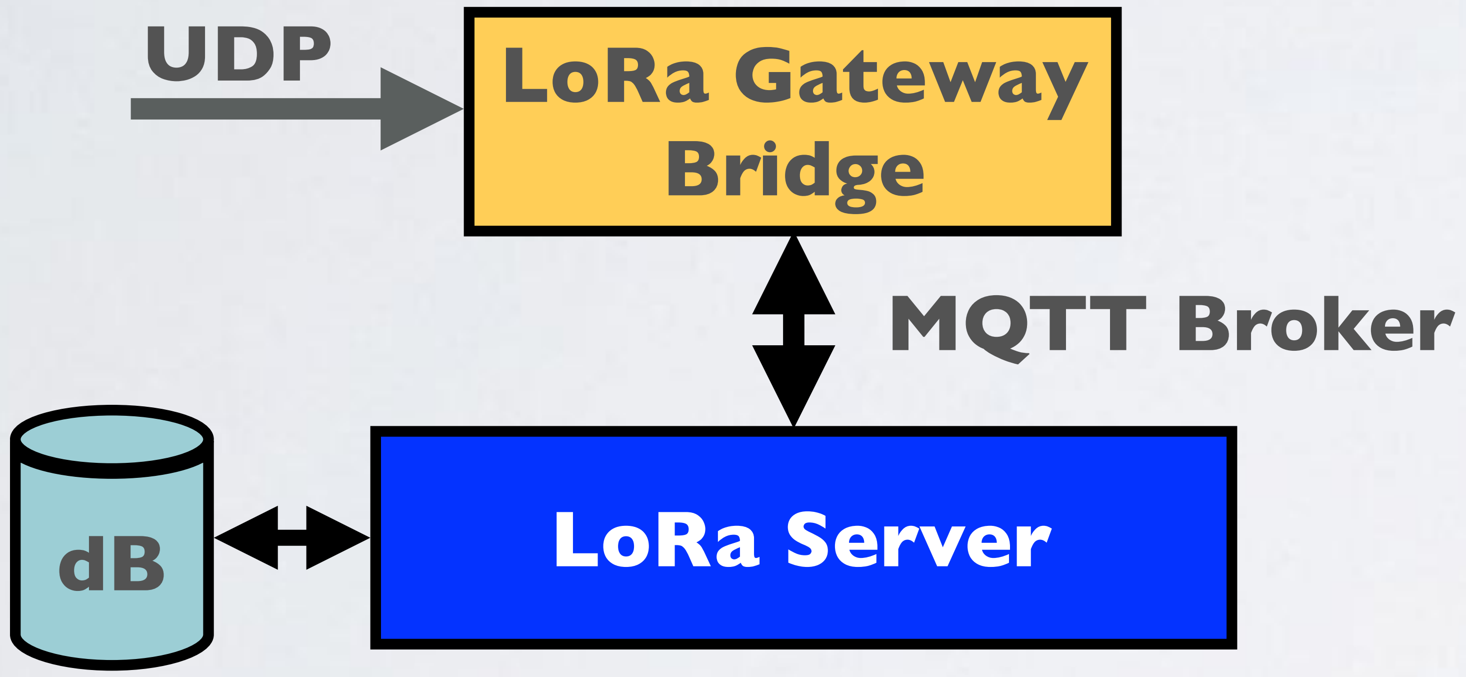
In this example the LoRa Gateway Bridge is installed on the Gateway.

global_conf.json

```
/* LoRa Gateway Bridge
   server address and port
*/

"gateway_conf": {
    "server_address": "localhost",
    "serv_port_up": 1700,
    "serv_port_down": 1700
}
```

HOW IT WORKS

- The Lora Gateway Bridge abstracts the received packet forwarder UDP protocol data. The data is converted into JSON and publishes it to a MQTT broker. The lora-gateway-bridge.toml file is used to configure the Lora Gateway Bridge.



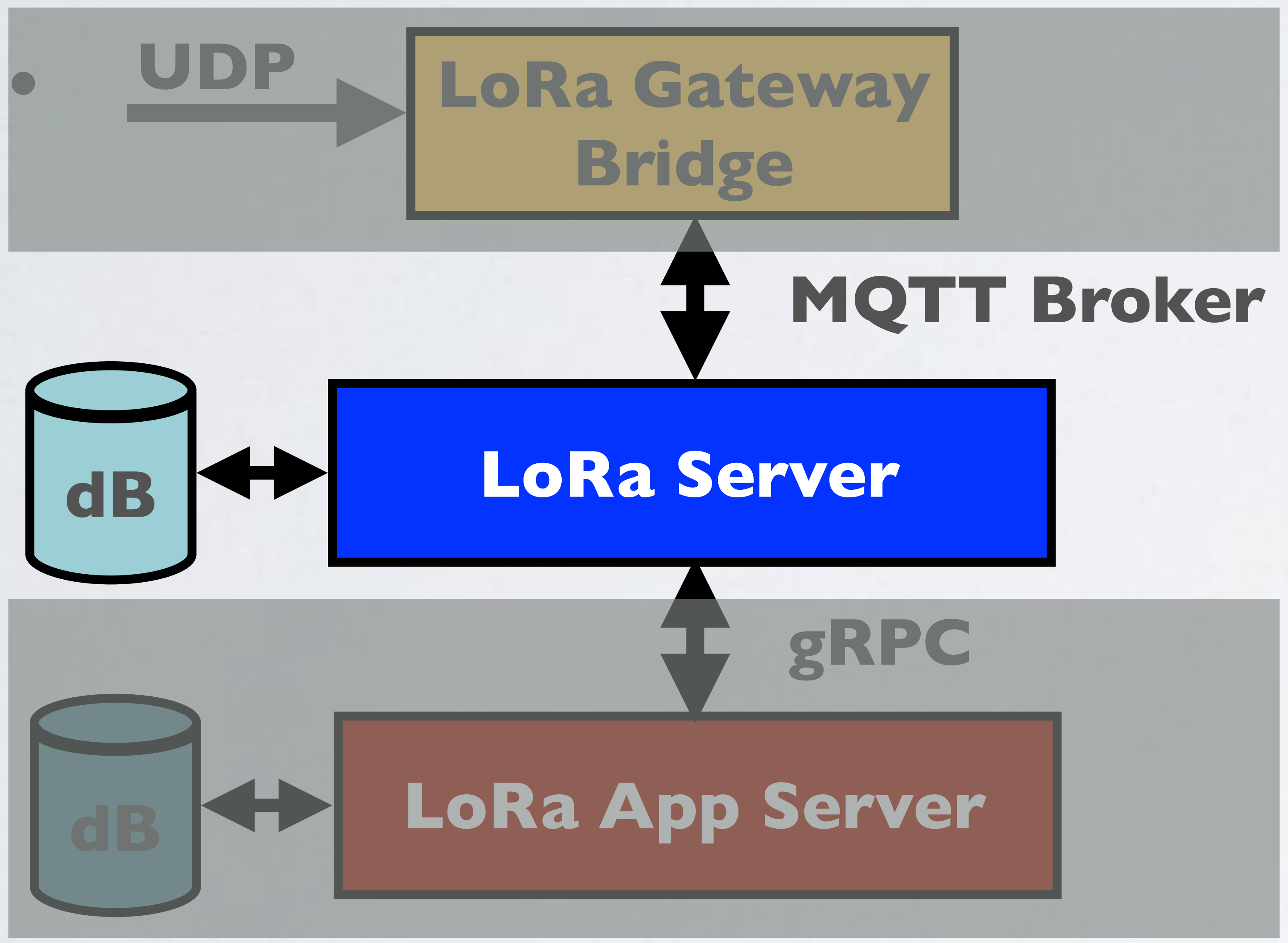
In this example the MQTT Broker and LoRa Gateway Bridge are installed on the same server.

lora-gateway-bridge.toml

```
# MQTT broker address and port
server=“tcp://127.0.0.1:1883”

# MQTT authentication
username=“”
password=“”
```

HOW IT WORKS

- The Lora Server is the LoRaWAN network server and it has a PostgreSQL database to store the gateway data. The loraserver.toml file is used to configure the Lora Server.



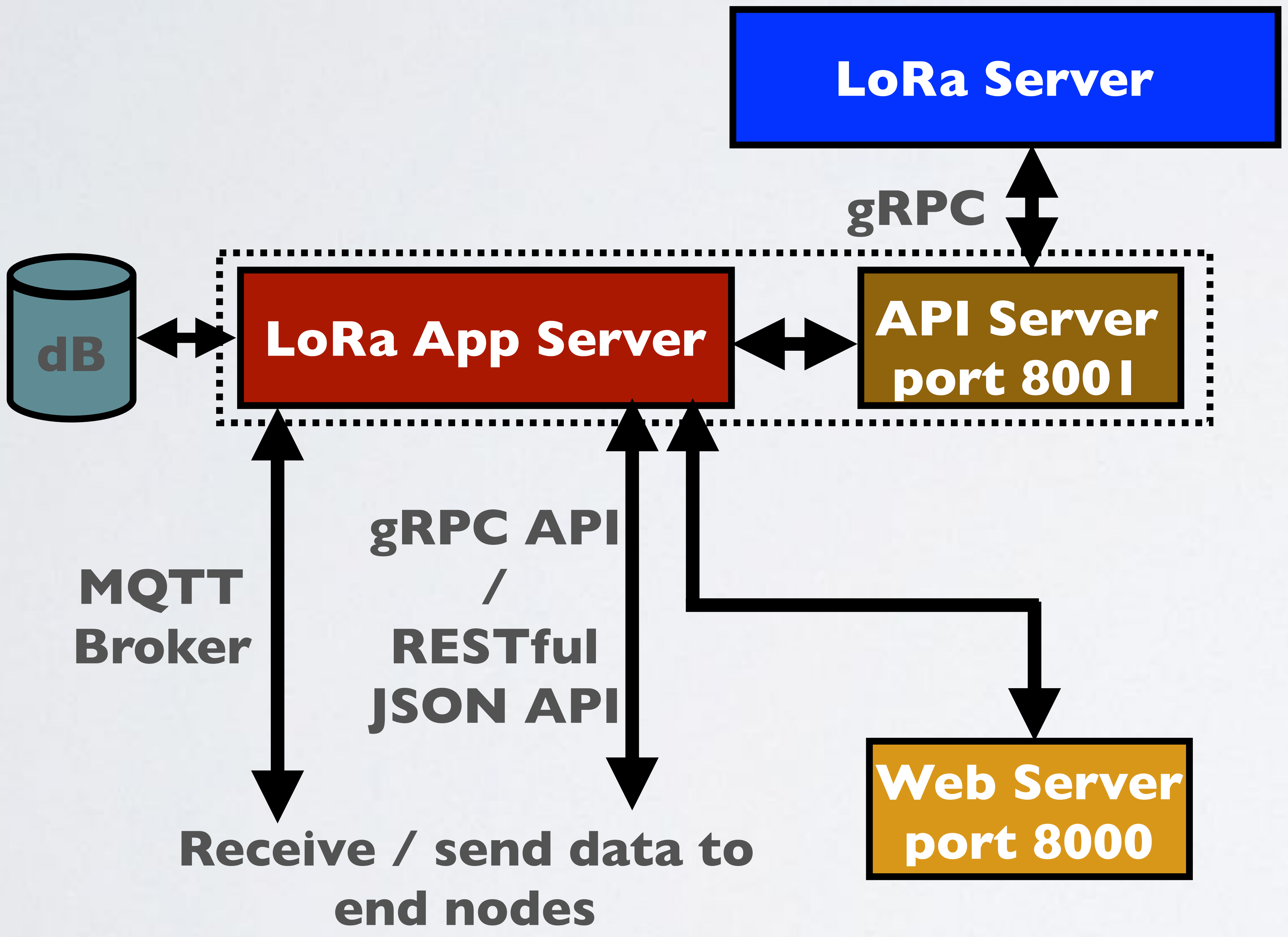
In this example the MQTT Broker, PostgreSQL database and LoRa Server are installed on the same server.

loraserver.toml

```
# PostgreSQL settings
# user=loraserver_ns, password=dbpassword
# hostname=localhost, database=loraserver_ns
dsn="postgres://loraserver_ns:dbpassword@localhost/
loraserver_ns?sslmode=disable"

#LoRaWAN regional band configuration.
name="EU_863_870"

# MQTT broker address and port
server="tcp://localhost:1883"

# MQTT authentication
username=""
password=""

# Use the system's local time zone
timezone="Local"
```

HOW IT WORKS

- The Lora App Server is the LoRaWAN application server and it also has a PostgreSQL database to store the gateway data. The lora-app-server.toml file is used to configure the Lora App Server.



lora-app-server.toml

```
# PostgreSQL settings
# user=loraserver_as, password=dbpassword
# hostname=localhost, database=loraserver_as
dsn="postgres://loraserver_as:dbpassword@localhost/
loraserver_as?sslmode=disable"

# MQTT broker address and port
server="tcp://localhost:1883"

# MQTT authentication
username=""
password=""

# The Internal API Server is used by LoRa Server
# to communicate with LoRa App Server
public_host=“localhost:8001"
```

HOW IT WORKS

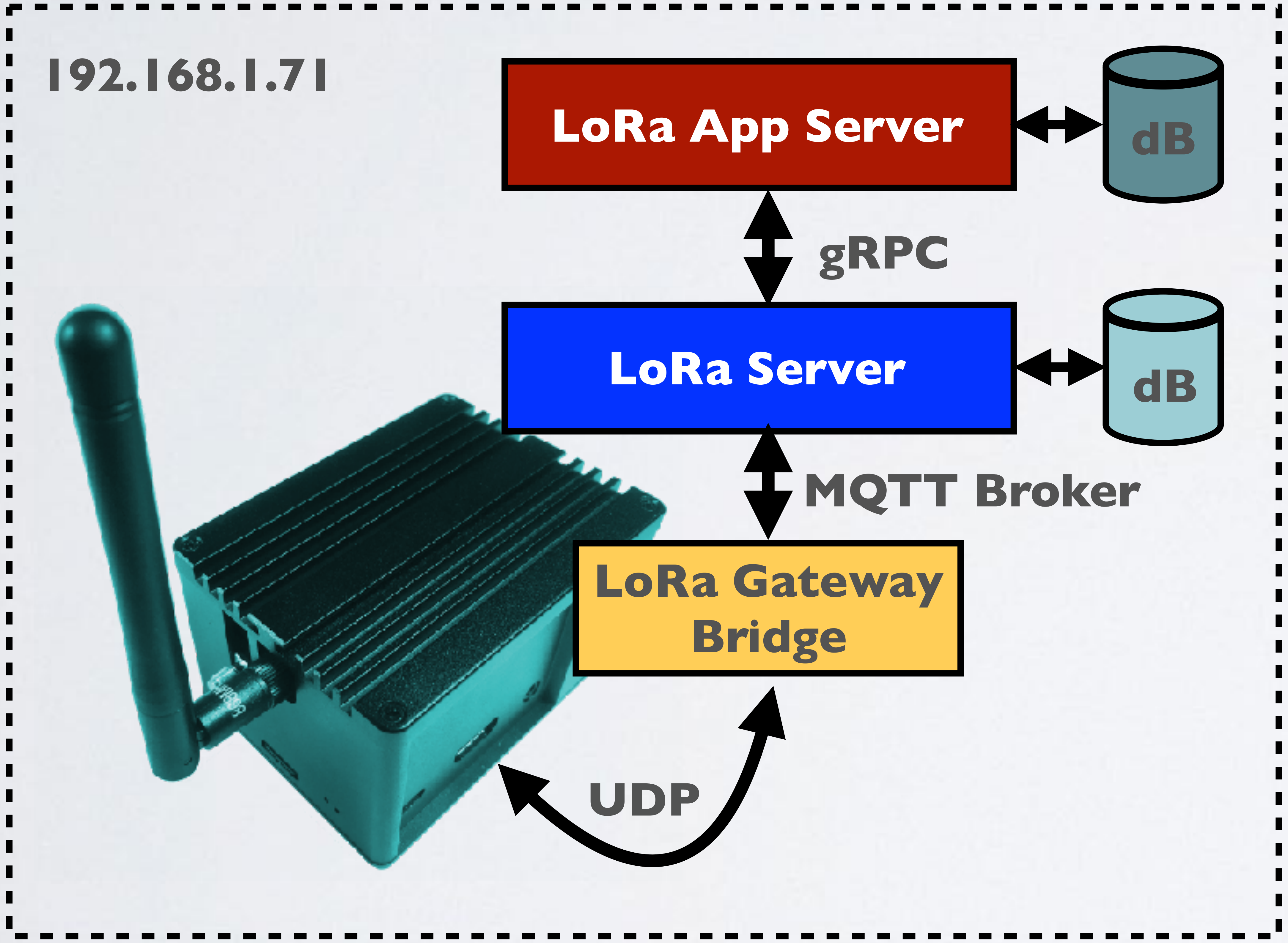
- This was a short and simplified explanation how the LoRa Server components works.

- The components have more configurations parameters which you can set.
  For more information, read the documentation, see: https://www.loraserver.io

USER MANUAL

- I have created a manual how to install and configure the LoRa Gateway Bridge, LoRa Server, LoRa App Server and their dependencies on the RAK 831 Pilot Gateway.

  https://www.mobilefish.com/developer/lorawan/lorawan_quickguide_loraserver_installation.html



ALL LORA SERVER COMPONENTS ON GATEWAY

Gateway

Configuration file: /opt/ttn-gateway/packet_forwarder/lora_pkt_fwd/global_conf.json
server_address=localhost (LoRa Gateway Bridge)
serv_port_up=1700 (the default port that LoRa Gateway Bridge is using)
serv_port_down=1700 (same)

LoRa Gateway Bridge

Configuration file: /etc/lora-gateway-bridge/lora-gateway-bridge.toml
MQTT broker address and port: server=“tcp://127.0.0.1:1883”

LoRa Server

Configuration file: /etc/loraserver/loraserver.toml
PostgreSQL settings: dsn=“postgres://loraserver_ns:dbpassword@localhost/loraserver_ns?sslmode=disable"
Automatically apply database migrations: automigrate=true
LoRaWAN regional band configuration: name=“EU_863_870"
System's local time zone: timezone=“Local”
MQTT broker address and port: server=“tcp://127.0.0.1:1883”

LoRa App Server

Configuration file: /etc/lora-app-server/lora-app-server.toml
PostgreSQL settings: dsn=“postgres://loraserver_as:dbpassword@localhost/loraserver_as?sslmode=disable"
JWT secret: jwt_secret=“zHj0RGotdJbgtKpPrv0/2axdER4bsnss6SHueANdEUg="
MQTT broker address and port: server=“tcp://127.0.0.1:1883”
Internal API server: public_host=“localhost:8001”

ALL LORA SERVER COMPONENTS ON GATEWAY

LoRa App Server Web Interface

Web UI: http[s]://<lora_app_server_hostname_or_ip_address>:8080
Username: admin
Password: admin

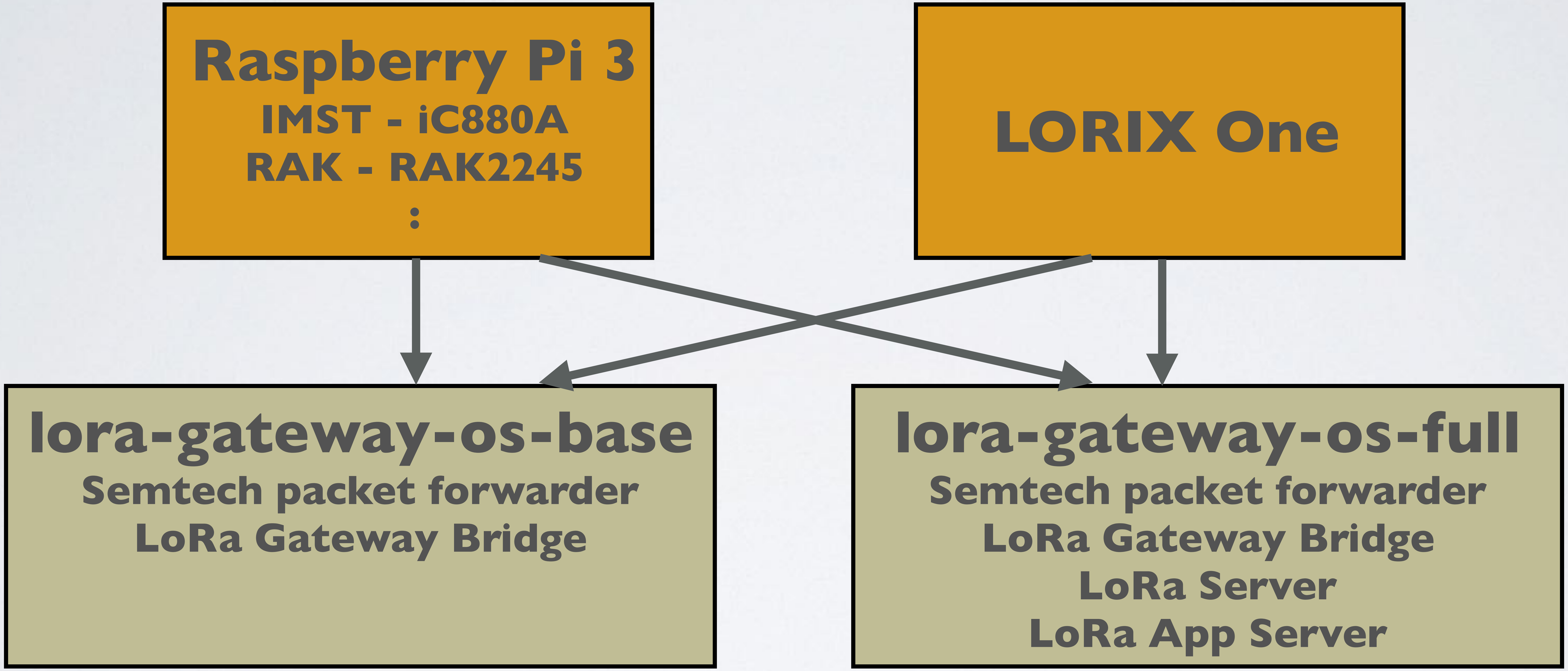
LoRa App Server API

API: http[s]://<lora_app_server_hostname_or_ip_address>:8080/api

LORA GATEWAY OS

- The LoRa Gateway OS is an easy way to get started with LoRaWAN and the LoRa Server project.
- LoRa Gateway OS are SD card images which contains out-of-the-box support for the Semtech packet forwarder, Lora Gateway Bridge, LoRa Server and LoRa App Server for certain concentrator shields.
- More information: https://www.loraserver.io/lora-gateway-os/overview/

LORA GATEWAY OS



Raspberry Pi 3
IMST - iC880A
RAK - RAK2245
:

LORIX One

lora-gateway-os-base
Semtech packet forwarder
LoRa Gateway Bridge

lora-gateway-os-full
Semtech packet forwarder
LoRa Gateway Bridge
LoRa Server
LoRa App Server

Status: May 2019

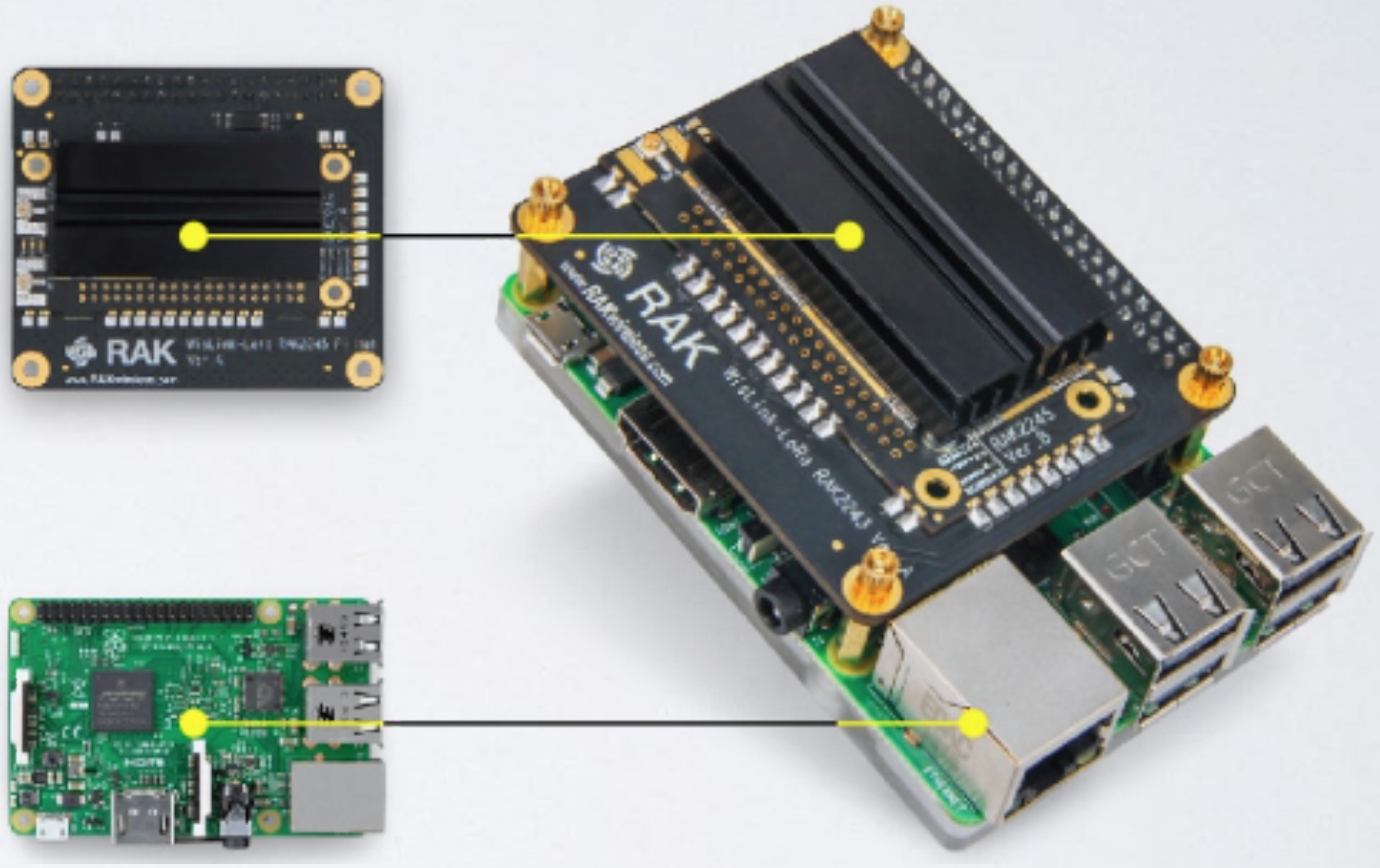
OTHER ALTERNATIVES

- There are also gateways on the market with the LoRa Server components already installed, such as the RAK WisKit.
  This kit contains among other things:
  - RAK2245 Pi HAT
    (LoRa concentrator module)
  - Raspberry Pi 3B+
  - WisNode
    (LoRa Node)

RAK WisKit

RAK2245 Pi Hat + Raspberry Pi

Prototype To Commercial



- A tutorial about this can be found at: https://www.hackster.io/fomi-T/simplest-lora-starter-kit-w-rak2245-rpi-ttn-loraserver-0ad993

SD CARD WEAR OUT

- An SD Card **MAY** wear out.The lifetime of SD cards is limited by the number of writes.

- It is recommended NOT to use cheap SD cards.

- LoRa Server tries to minimise the number of database writes.