EUROPEAN UNION

Interreg
EUROPEAN UNION
Austria-Czech Republic
European Regional Development Fund

LoRa(WAN) Webinar

LoRaWAN

Author: Harald Eigner (TU Wien)

These slides give an overview on the LoRaWAN network layer (on top of LoRa communication). The fully standardized layer's fundamental properties are explained. Readers of this document shall gain insight into how LoRaWAN devices can be enrolled (OTAA vs. ABP), what device classes A/B/C can be used, how the key exchange works, etc.

TU WIEN
TECHNISCHE UNIVERSITÄT WIEN
Vienna | Austria

EMCE

Contents

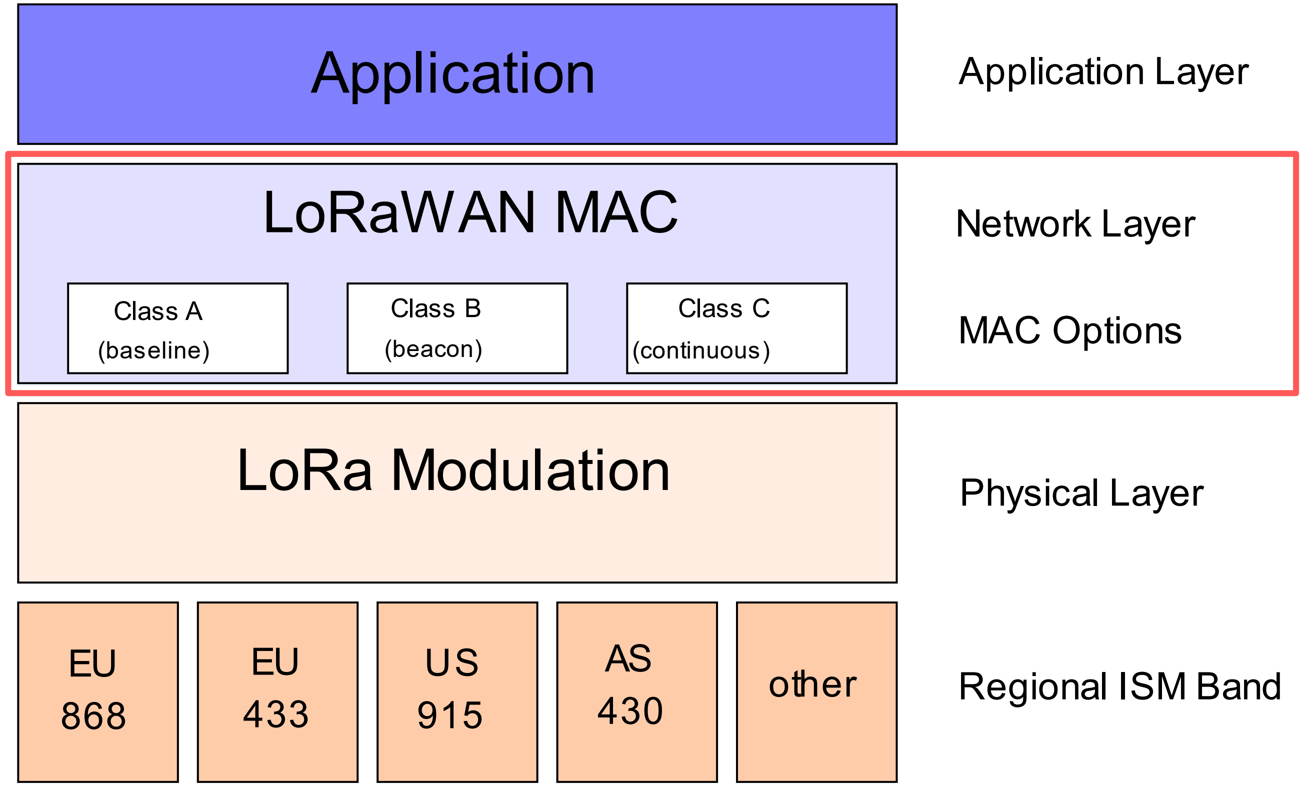
- LoRaWAN specifications
- Activation of an End-Device
- Adaptive Data Rate
- Classes

Contents

- LoRaWAN specifications
- Activation of an End-Device
- Adaptive Data Rate
- Classes

LoRaWAN

- Network layer protocol designed for the use on top of LoRa
- ALOHA-type random access
- Energy efficient
- Low complexity



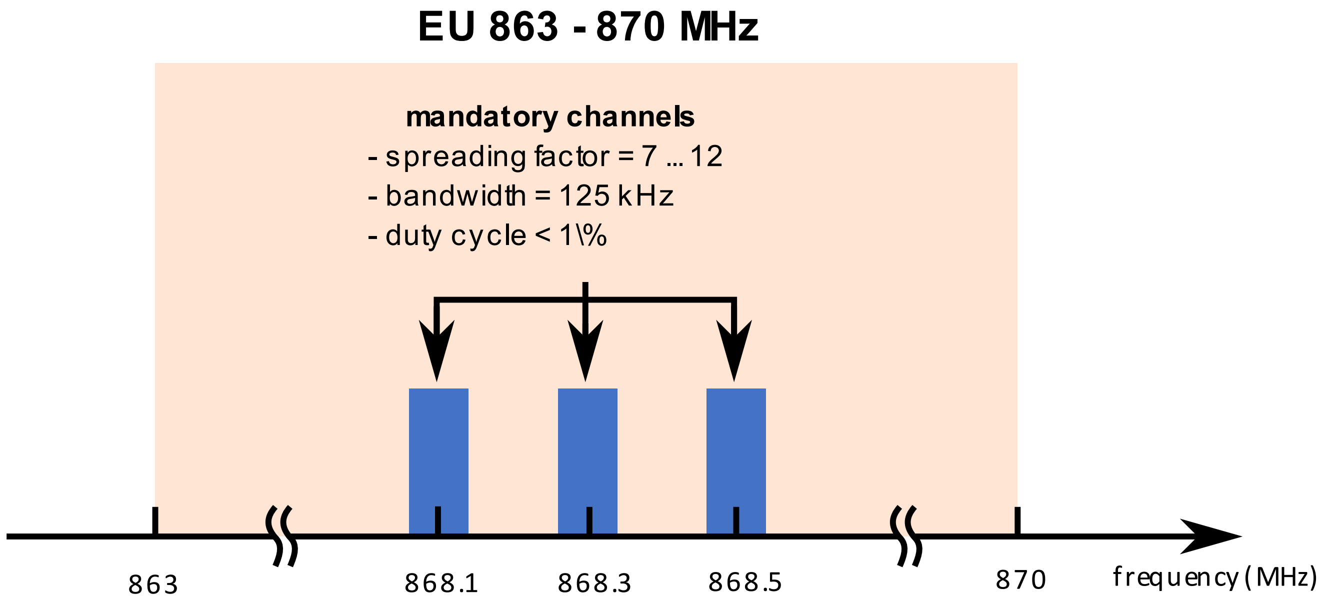
LoRaWAN

- Region specific spectrum allocations and regulatory requirements

	Europe	North America
Frequency band	867 – 869 MHz	902 – 928 MHz
Channels	3 mandatory	80
Channel BW up	125/250 kHz	125/500 kHz
Channel BW down	125 kHz	500 kHz
TX Power up	+14 dBm	+20 dBm
TX Power down	+14 dBm	+27 dBm
SF up	7 - 12	7 - 10
Data Rate	250 bps - 50 kbps	980 bps - 21.9 kbps
Link Budget up	155 dB	154 dB
Link Budget down	155 dB	157 dB

LoRaWAN

- 3 mandatory frequency channels must be supported by all devices
- ETSI regulations:
  - Duty cycle or
  - LBT AFA (Listen Before Talk Adaptive Frequency Agility): device senses a channel to determine if there is activity by measuring the RSSI before transmitting.



Mandatory channels for EU 863-870 frequency band

LoRaWAN

Duty Cycle Regulations in the 863–870 MHz frequency band in Europe:

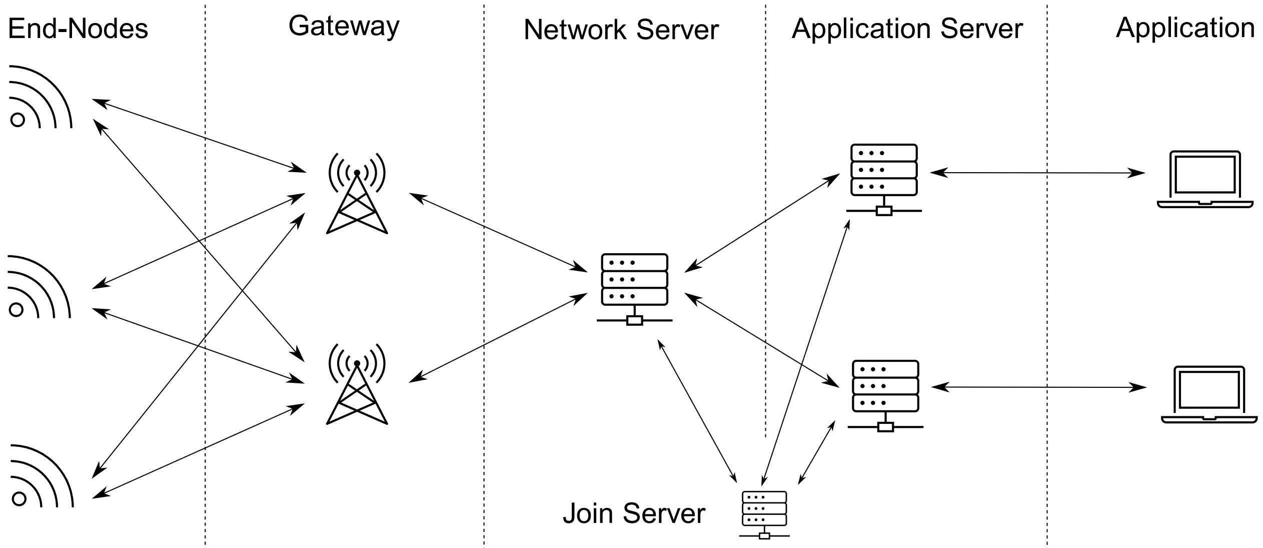
Frequency	Max. ERP	Duty Cycle
863.0 – 865.0 MHz	25 mW	< 0.1 %
865.0 – 868.0 MHz	25 mW	< 1 %
868.0 – 868.6 MHz	25 mW	< 1 %
868.7 – 869.2 MHz	25 mW	< 0.1 %
869.4 – 869.65 MHz	500 mW	< 10 %
869.7 – 870.0 MHz	5 mW	< 0 – 100 %

Frequency	Max. ERP	Duty Cycle
868.6 – 868.7 MHz	10 mW	< 0.1 %
869.2 – 869.3 MHz	10 mW	< 0.1 %
869.3 – 869.4 MHz	10 mW	< 1 %
869.65 – 869.7 MHz	25 mW	< 0 – 100 %

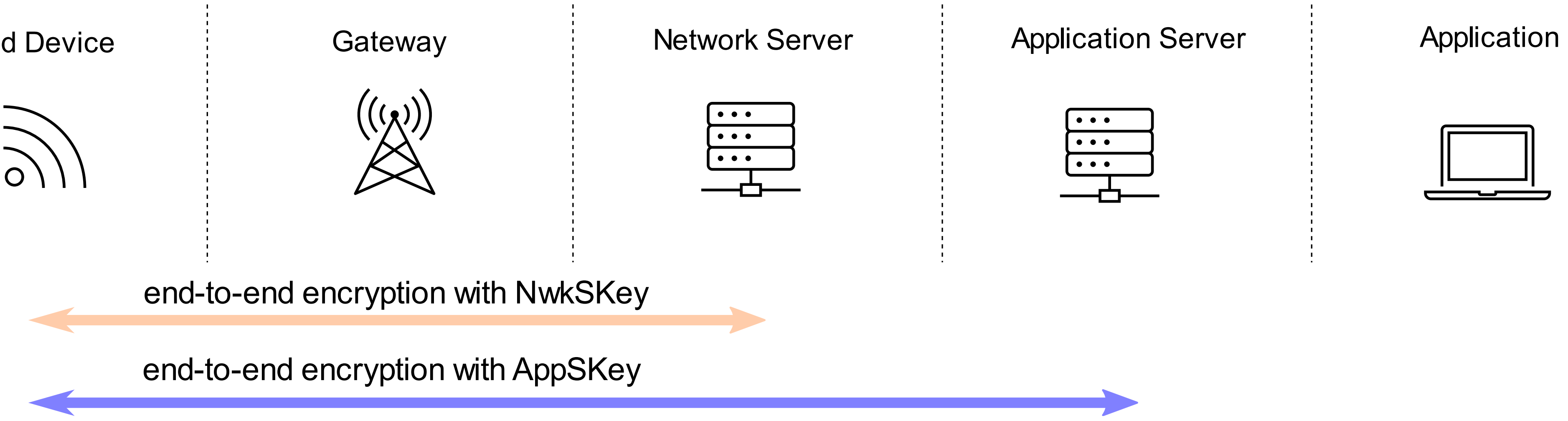
- Mandatory LoRa channels are in the 868.0 – 868.6 MHz range
- The second receive window for downlink communication uses a fixed frequency and data rate. The default parameters are 869.525 MHz / DR0 (SF12, 125 kHz)

Network Architecture

- Gateway:
  - Receives/transmits LoRa packets from/to end-device
  - Forwards them to network server
- Network Server:
  - Manages gateways
  - Responsible for routing, security and power management
  - Deduplicates packets received from several end-devices
- Application Server:
  - Manages the infrastructure of the end-devices
  - Responsible for processing data and downlink payloads
- Join Server:
  - Manages the over the air activation (OTAA) process



Security


End Device
Gateway
Network Server
Application Server
Application
end-to-end encryption with NwkSKey
end-to-end encryption with AppSKey

- Two-layer encryption to secure data being transmitted
- Network Session Key (NwkSKey) is used to validate the integrity of each message in the network layer
- Application Session Key (AppSKey) is used to encrypt the payload data in the application layer

Security

Frame Counter

- Two counters for uplink and downlink messages are implemented:
    - FcntUp: incremented by the end device and transmitted to the network server
    - FcntDown: incremented by the network server and transmitted to the end device
- Every message with a counter value lower than the previous one will be neglected
- Retransmissions do not increment the frame counter

Contents

- LoRaWAN specifications
- Activation of an End-Device
- Adaptive Data Rate
- Classes

Activation of an End-device

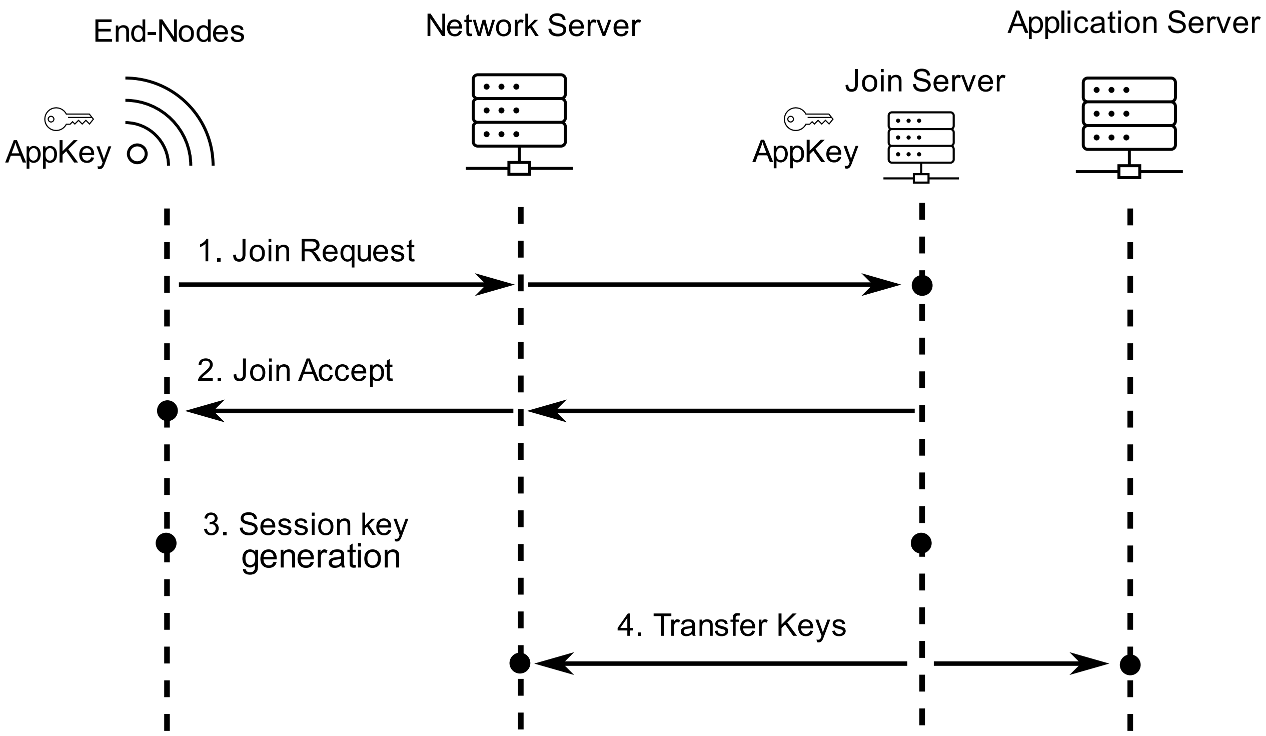
Two different procedures are available:

- Over the Air Activation (OTAA)
  - Join Procedure, security keys are negotiated during the process
  - Dynamic device address
- Activation by Personalization
  - No join procedure, security keys are set beforehand
  - Static device address

Over the Air Activation (OTAA)

4 step process:

- Join Request
- Join Accept
- Session key generation
- Transfer of Session Keys



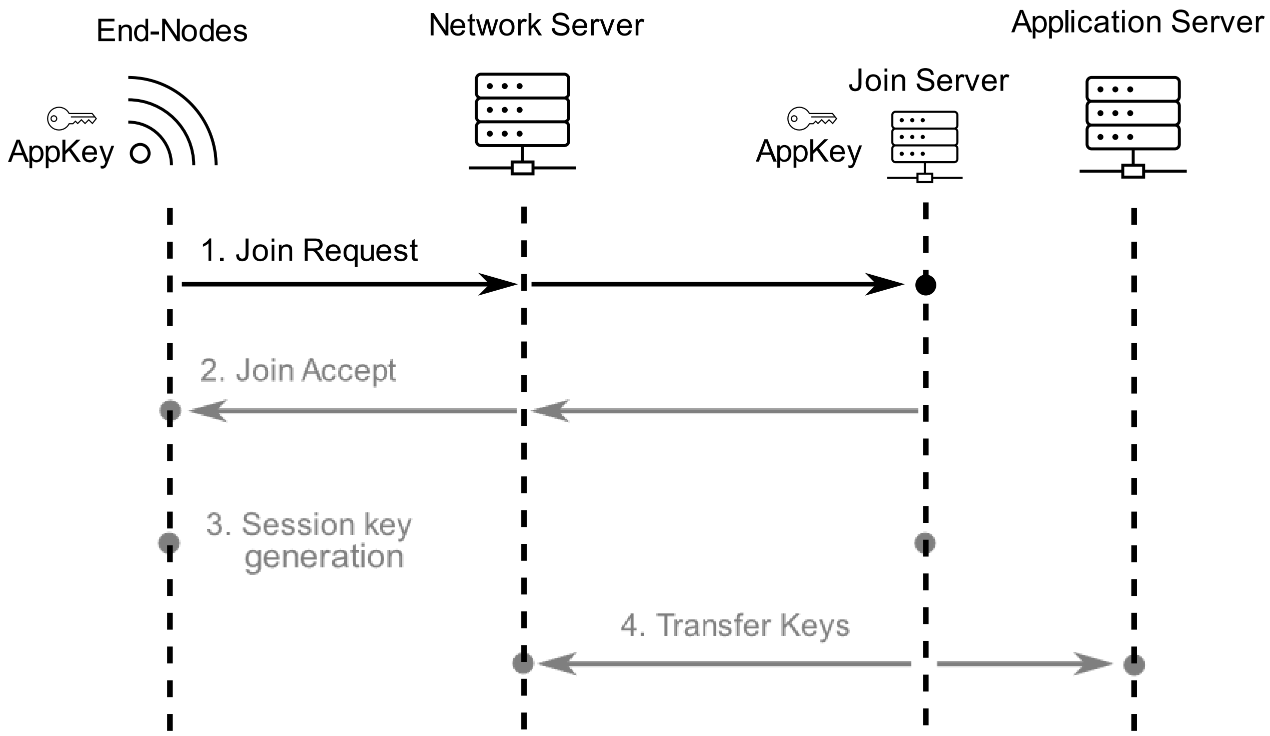
Over the Air Activation (OTAA)

Join Request:

- Request message from end-device to corresponding join server via network server, including identifier for device and application (DevEUI, JoinEUI) and a DevNonce for validation
- No encryption used
- Transmission possible with any spreading factor and the mandatory channels at 868.1, 868.3, and 868.5 MHz

Join Request message

Size(Bytes)	8	8	2
Join Request	JoinEUI	DevEUI	DevNonce



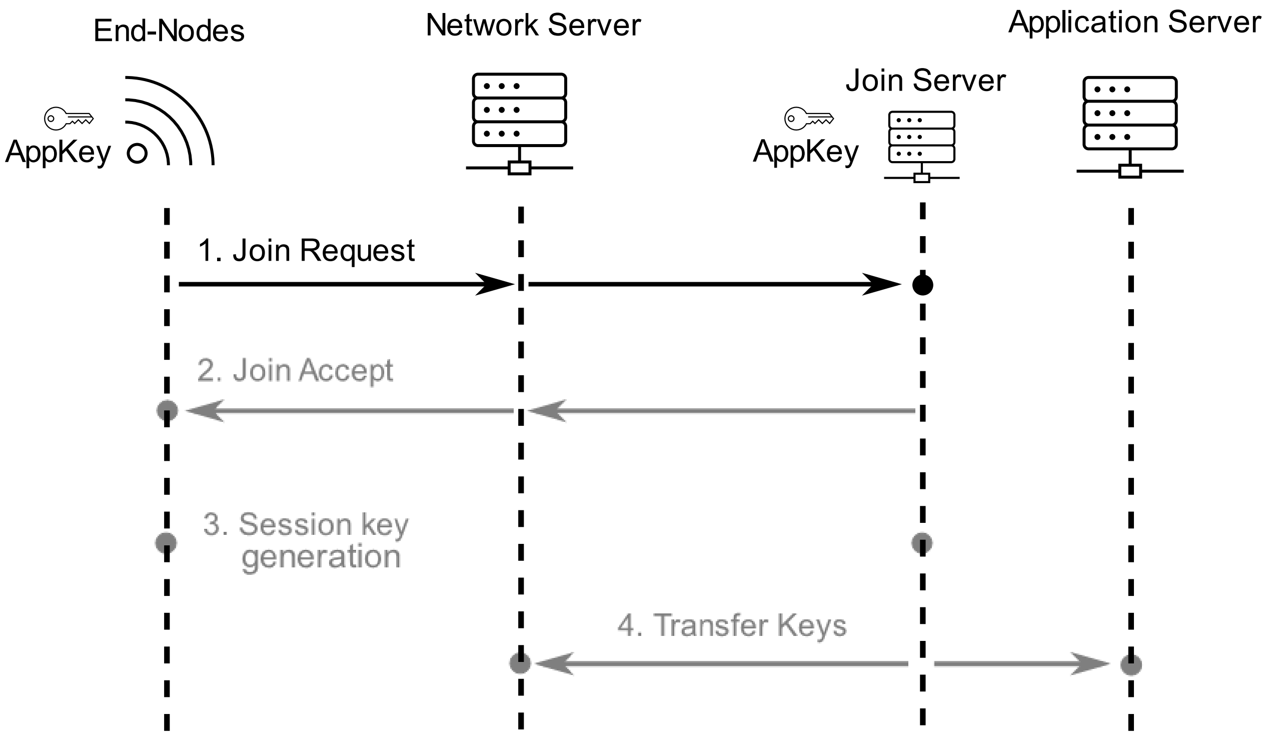
Over the Air Activation (OTAA)

Join Request:

- JoinEUI: global application ID in IEEE EUI64 address space that uniquely identifies the Join Server
- DevEUI: global end-device ID in IEEE EUI64 address space that uniquely identifies the end-device
- DevNonce: unique identifier exchanged by end-device and join server
  - *LoRaWAN 1.0.3 Specification: Random value*
  - *LoRaWAN 1.1 Specification: counter starting at zero and incrementing every Join Request*

Join Request message

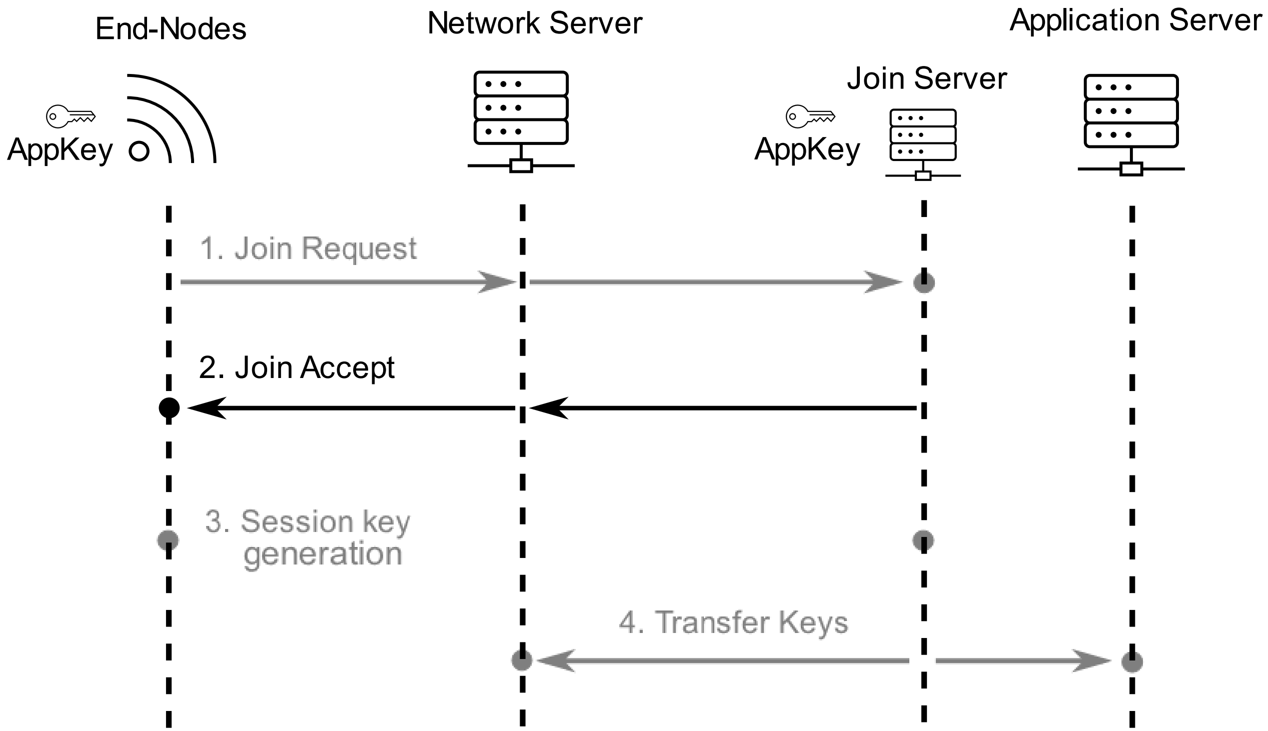
Size(Bytes)	8	8	2
Join Request	JoinEUI	DevEUI	DevNonce



Over the Air Activation (OTAA)

Join Accept:

- Join accept message sent back to the end-device, containing information for the derivation of session keys
- JoinNonce: unique identifier exchanged by end-device and join server
- NetID: Network identifier
- DevAddr: Network address of the end-device
- DLSettings: Downlink configuration settings
- RxDelay: Delay between up- and downlink
- CFList: contains optional network parameters and frequency channels



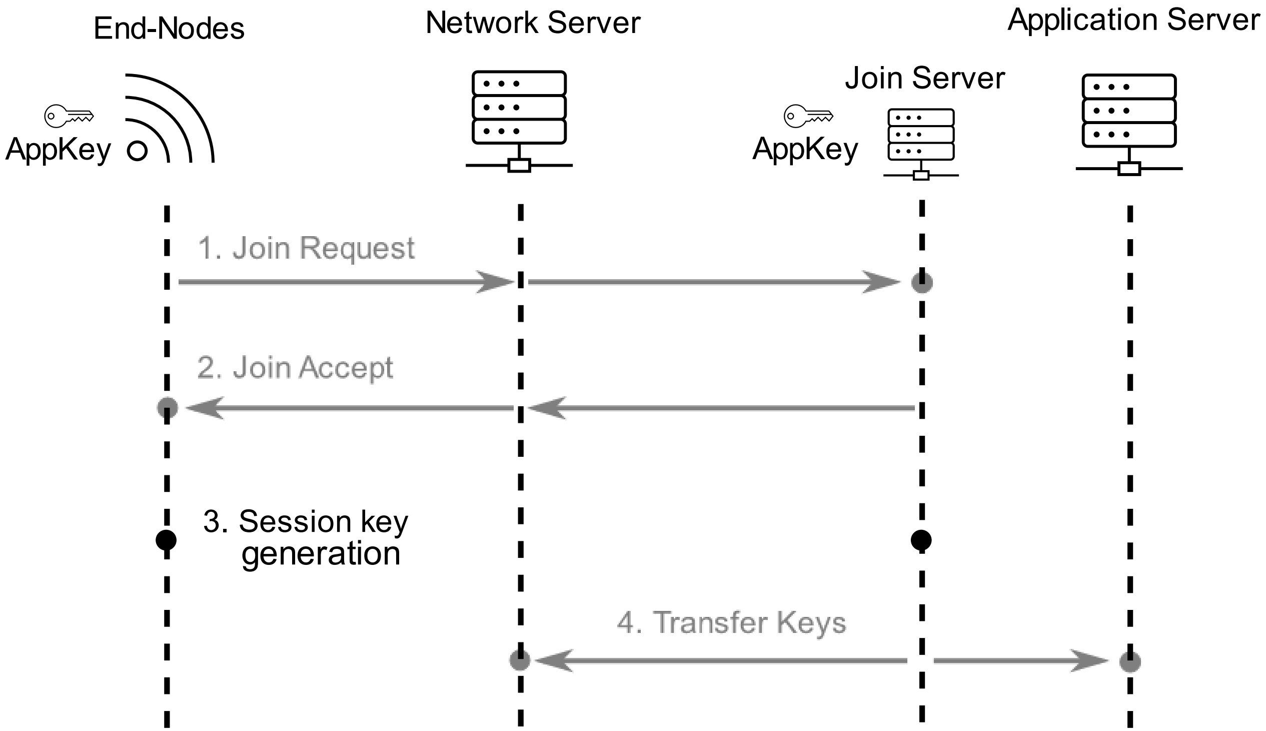
Join Accept message

Size(Bytes)	3	3	4	1	1	16 (optional)
Join Accept	JoinNonce	NetID	DevAddr	DLSettings	RxDelay	CFList

Over the Air Activation (OTAA)

Session Key Generation:

- Keys are generated at the join server and the end-device based on exchanged data from Join Request and Join Accept message



Over the Air Activation (OTAA)

Calculation of session keys:

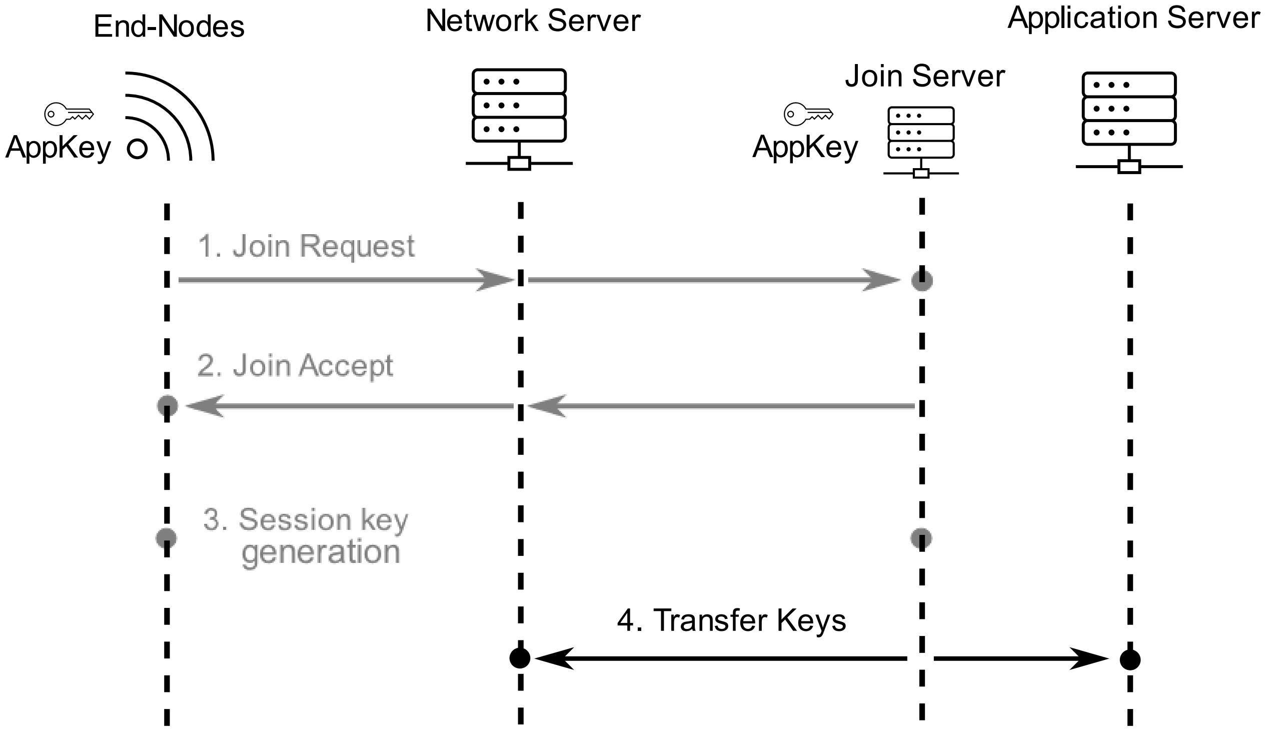
- **NwkSKey = aes128_encrypt(AppKey, 0x01 | JoinNonce | NetID | DevNonce | pad16)**
- **AppSKey = aes128_encrypt(AppKey, 0x02 | JoinNonce | NetID | DevNonce | pad16)**

  - 128-bit advanced encryption standard
  - AppKey: Root key of the end device, stored at end-device and Join Server
  - NetID: Unique 24-bit network identifier of the device's home network
  - Pad16: appends zero octets so that the length of data is a multiple of 16

Over the Air Activation (OTAA)

Transfer of session keys:

- Join server transfers session keys to respective servers:
    - Application session key (AppSKey) -> Application Server
    - Network session key (NwkSKey) -> Network Server



Activation by Personalization

- Application session key, network session key and device adress are exchanged beforehand
- No join procedure needed
- End-device is activated after first uplink message
- Frame counters must be updated after a restart of the end device
- Parameters exchanged during join procedure in OTAA like
    - CFlist: Frequency channel list
    - DLSetting: Downlink configuration settings or
    - RxDelay: Delay between up- and downlink

are exchanged with the first uplink messages

Activation of an End-Device

OTAA

Advantages

- Session keys are only generated when required
- Frame counters will be renegotiated after restart of the device
- Rejoin after switching network
- Network settings like RxDelay or frequency channel list CFList can be specified at join procedure

Disadvantages:

- A scheme is required to pre-program each device with a unique JoinEUI , DevEUI and AppKey
- The device must support the join function and be able to store dynamically generated keys

ABP

Advantages

- Capabilities and resources for a join procedure is not needed
- No scheme is necessary to specify a unique JoinEUI, DevEUI and AppKey

Disadvantages:

- Session keys must be ensured to be unique
- Network setting cannot be specified at join time, only with an exchange of payload
- Events that warrant a change of keys (moving to a new network, keys being expired) require a re-programming of the device
- Frame counter must be reset after a restart of the device

Contents

- LoRaWAN specifications
- Activation of an End-Device
- Adaptive Data Rate
- Classes

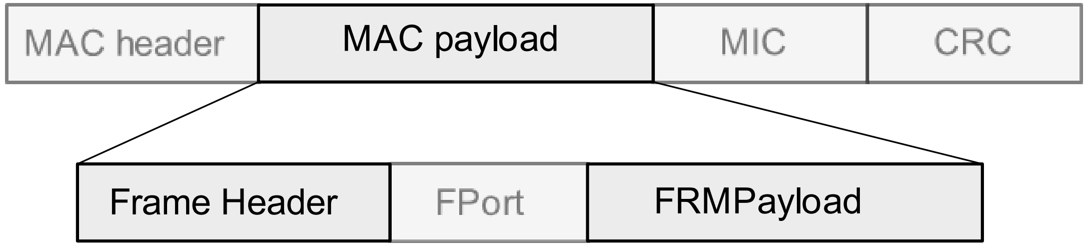
Adaptive Data Rate

- LoRaWAN protocol defines the Adaptive Data Rate (ADR) scheme to control the uplink transmission parameters of LoRa devices.
  - Transmission parameters
    - Spreading Factor
    - Bandwidth
    - Transmission power

- Whether ADR functionality will be used is requested by the end device by setting the ADR flag in the uplink message. If ADR flag is set, the network server can control the end device's transmission parameters.
- ADR should only be used in stable RF situations where end devices do not move.

Adaptive Data Rate

- ADR setting are transmitted as MAC commands
  - Piggybacked in Frame Header or
  - As separate message in FRMPayload
- LinkADRReq: Requests the end-device to change data rate, transmit power, repetition rate or channel
  - DataRate_TXPower: requested data rate and TX power
  - ChMask: channels useble for uplink access
- LinkADRAns: Acknowledges the LinkADRReq
  - Bit representing the acknowledged parameters are set to one

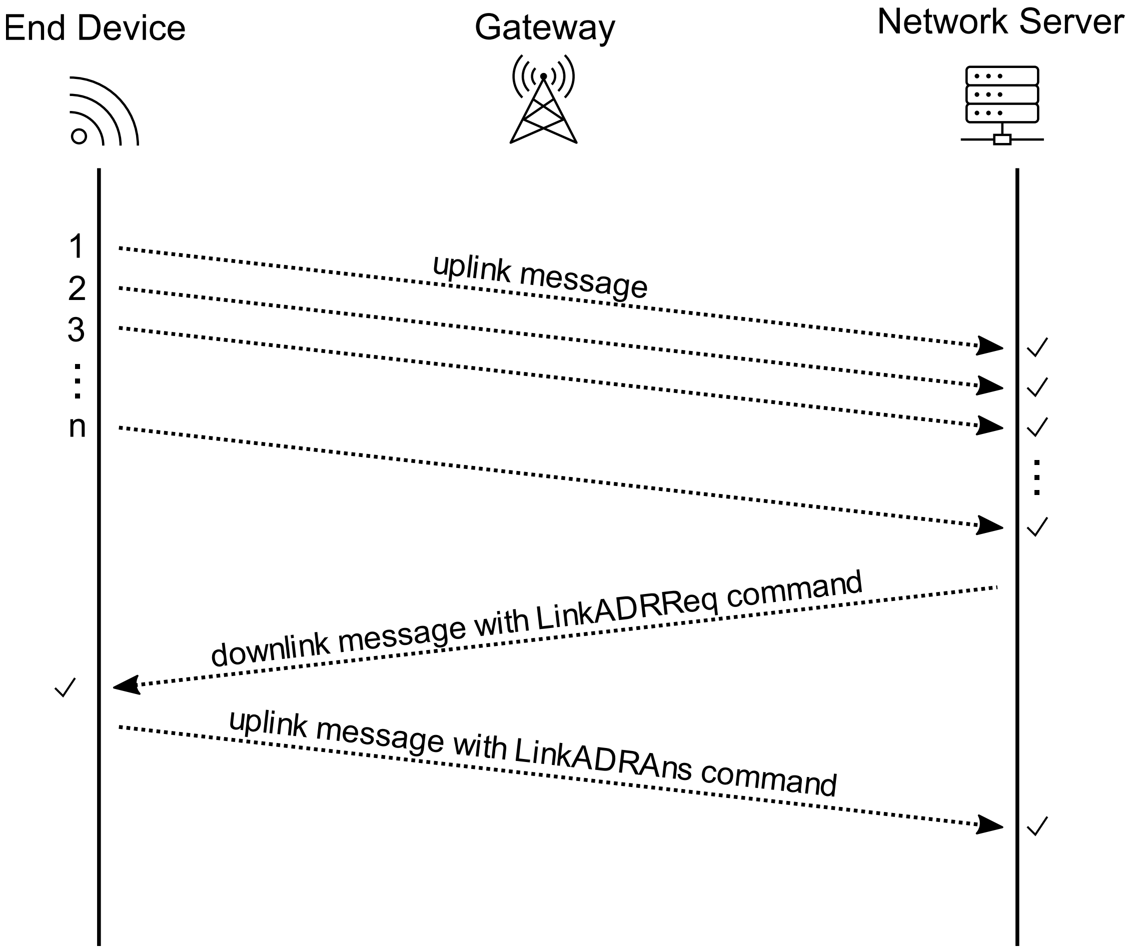


Size (Bytes)	1	1	2
LinkADRReq	DataRate_TXPower	ChMask	Redundancy

Bit #	7 ... 3	2	1	0
LinkADRAns	RFU	Power ACK	Data Rate ACK	Channel mask ACK

Adaptive Data Rate

- The network server collects the n most recent uplink transmission data from an end device (such as spreading factor, RSSI and SNR)
- Based on the signal strength, the network server determines the minimum data rate and link budget that can be supported by the end device
- The network server sends an ADR request command LinkADRReq to the end device
- The end device sends back an acknowledgement


End Device
Gateway
Network Server
1
2
3
⋮
n
uplink message
✓
✓
✓
⋮
✓
downlink message with LinkADRReq command
✓
uplink message with LinkADRAns command
✓

Adaptive Data Rate - Lost Link

What happens if the link is lost?

- The end device increments the counter ADR_ACK_CNT every time the uplink frame counter is incremented

- When ADR_ACK_CNT reaches the limit ADR_ACK_LIMIT, the end device sets ADR_ACL_REQ to one

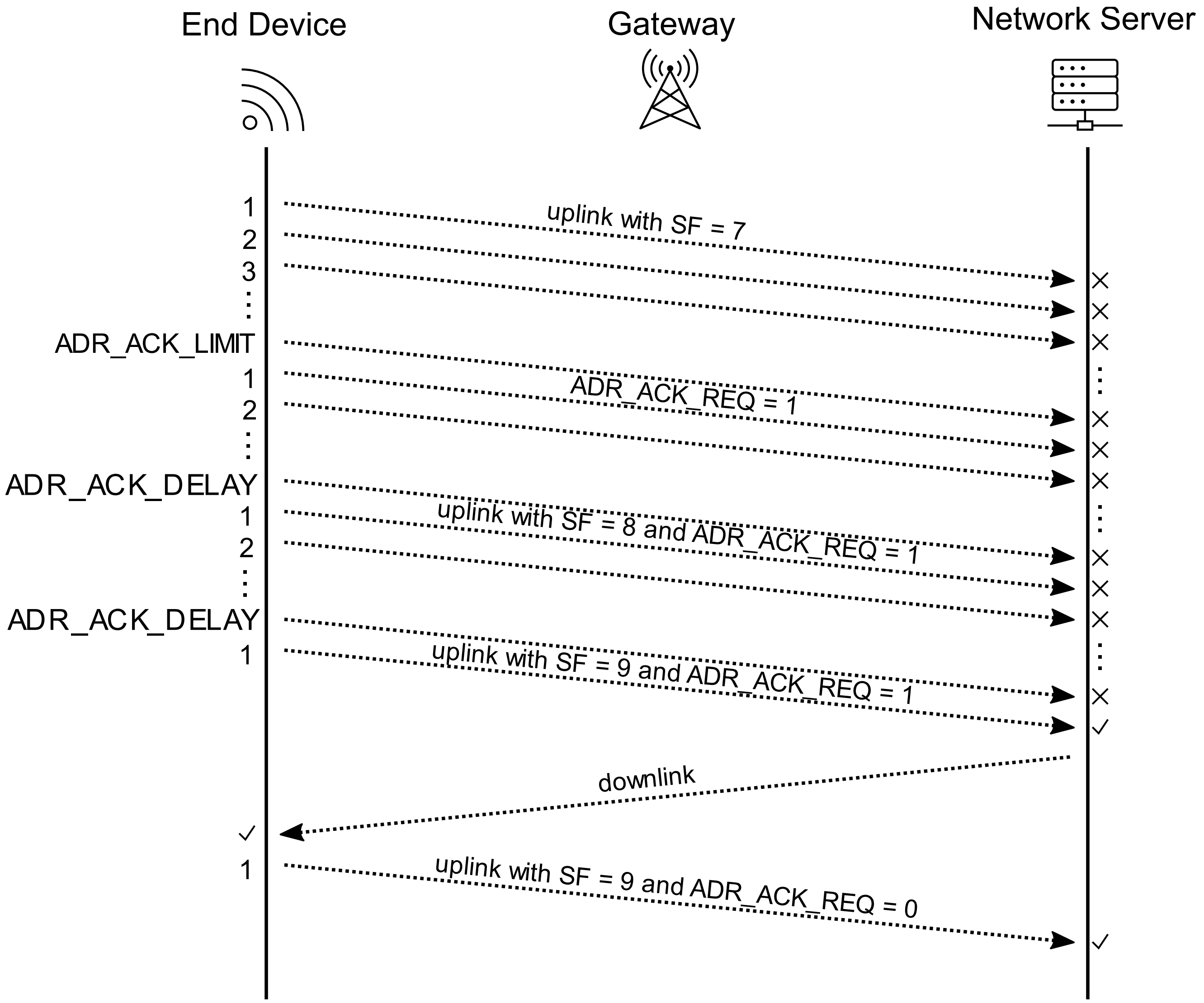
Size (Bytes)	4	**1**	2	1
Frame Header	DevAddr	**FCtrl**	FCnt	FOpts

downlink frame

Bit #	7	6	5	4	3 ... 0
FCtrl bits	ADR	RFU	ACK	FPending	FOptsLen

uplink frame

Bit #	7	6	5	4	3 ... 0
FCtrl bits	ADR	ADRACKReq	ACK	Class B	FOptsLen



Adaptive Data Rate - Lost Link

- If no downlink frame was received after the next ADR_ACK_DELAY uplinks, the end-node changes to a higher spreading factor
- The end device must further increase its spreading factor step by step every ADR_ACK_DELAY uplinks
- If any downlink is received, ADR_ACL_REQ is set back to zero

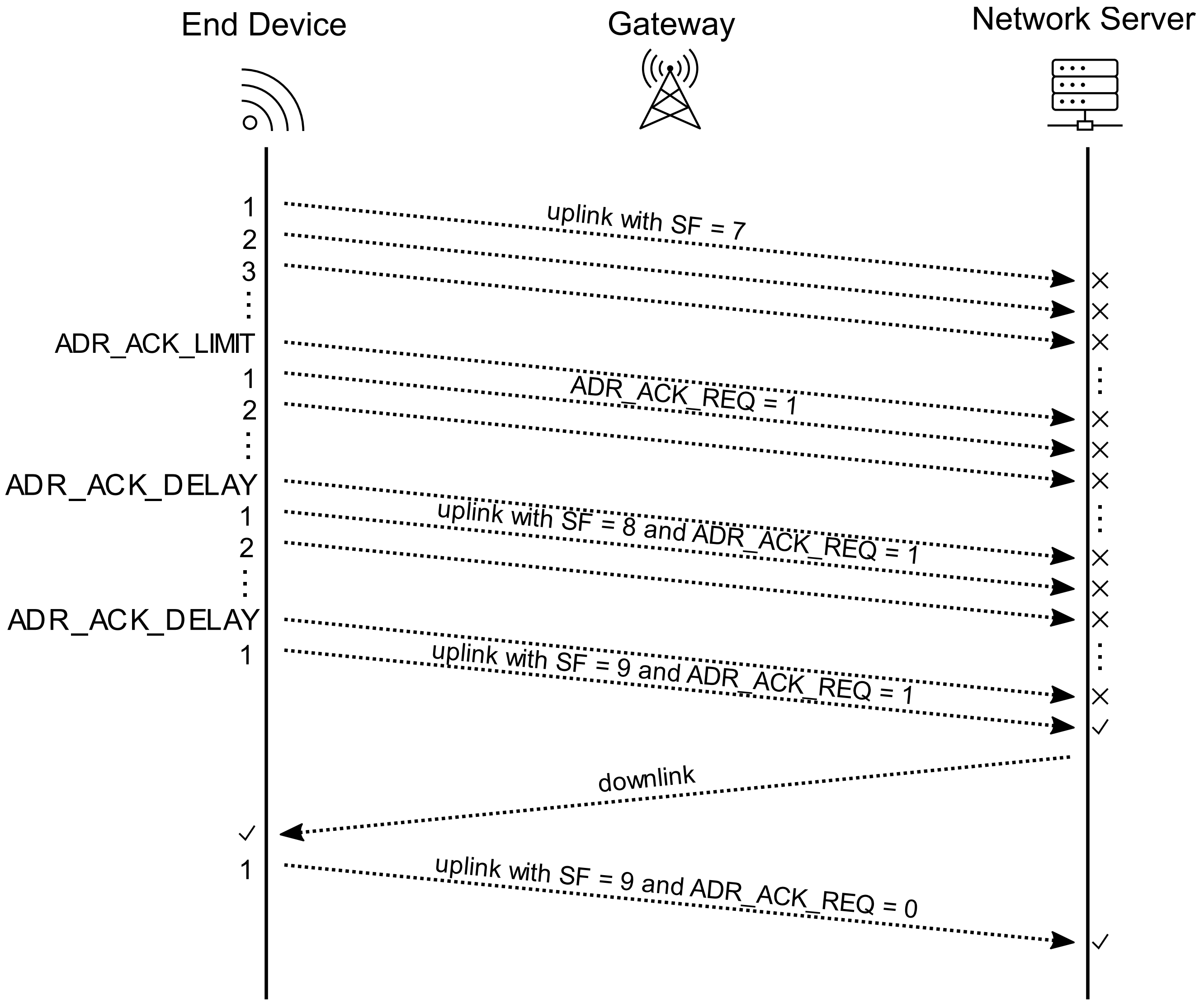
Size (Bytes)	4	**1**	2	1
Frame Header	DevAddr	**FCtrl**	FCnt	FOpts

downlink frame

Bit #	7	6	5	4	3 ... 0
FCtrl bits	ADR	RFU	ACK	FPending	FOptsLen

uplink frame

Bit #	7	6	5	4	3 ... 0
FCtrl bits	ADR	ADRACKReq	ACK	Class B	FOptsLen



Message Types

- There are two uplink message types in LoRaWAN communication:
  - Confirmed uplink
    - No acknowledgement by the receiver is needed
  - Unconfirmed uplink
    - The uplink message must be confirmed by the receiver

Contents

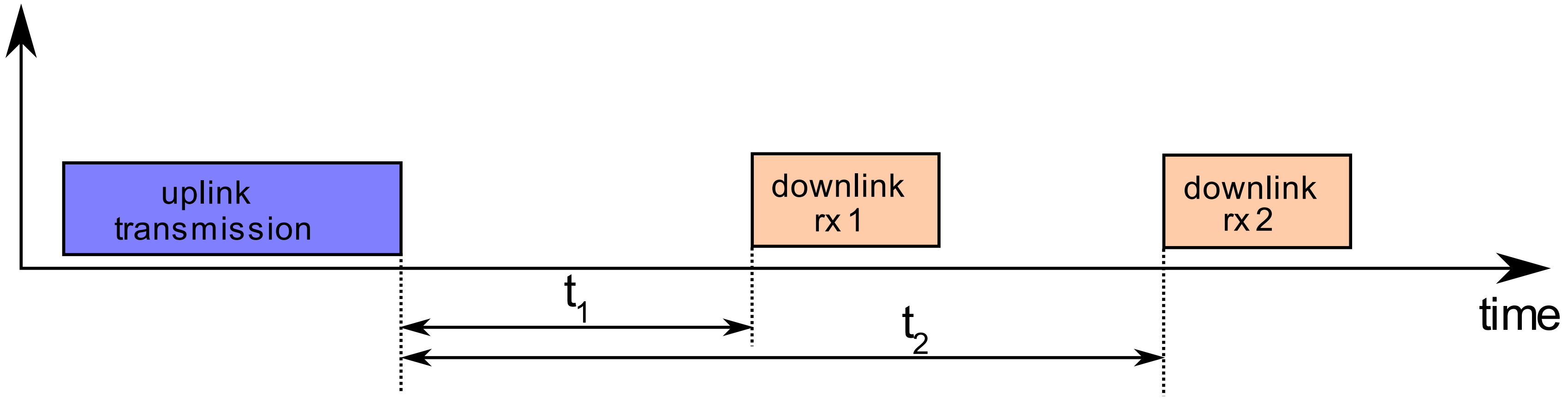
- LoRaWAN specifications
- Activation of an End-Device
- Classes

Classes

- LoRa-based end devices operate in one of three modes, depending on their device class
- Three different classes are available:
  - Class A: must be supported by all devices
  - Class B: must support class B and class A mode
  - Class C: must support all classes

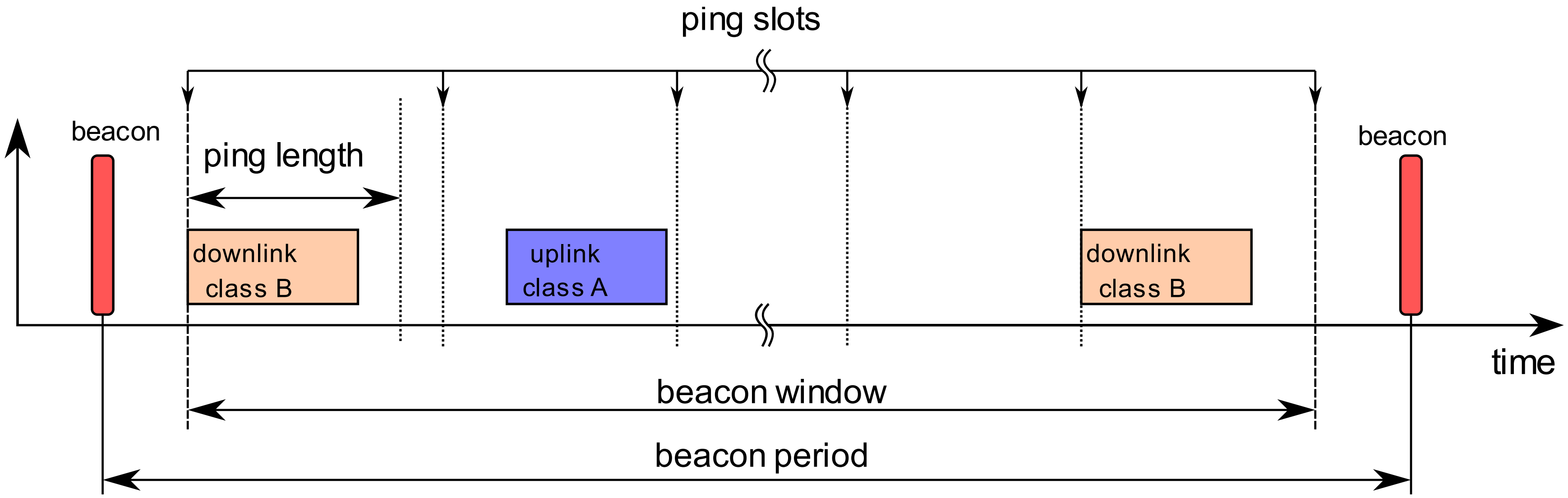
Class A

- Must be implemented by every LoRaWAN device
- Downlink only possible at two slots after uplink
- First receive window rx1 uses same channel as the preceding uplink.
- Second receive window rx2 uses fixed parameters (869.25 MHz / SF12 / 125 kHz; Duty cycle <10%, max ERP 500 mW)
- Low power consumption
- Application: battery powered sensors



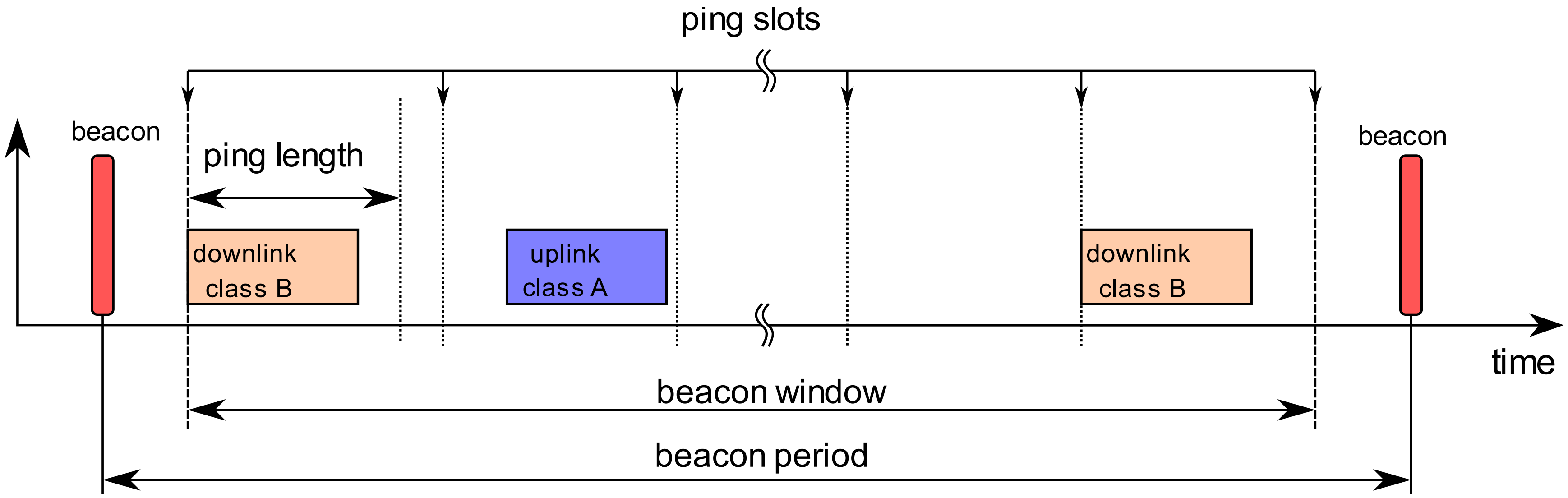
Class B

- Upgraded class A with additional synchronized reception slots
- Broadcast message (beacon) is used as a timing reference
- Regional specific beacon channel configuration
  - Europe: $\mathrm{SF} = 9$, code rate $4/5$, Channel $869.525\ \mathrm{MHz}$, $\mathrm{BW} = 125\ \mathrm{kHz}$
- Higher power consumption
- Applications: battery powered actuators or smart meters



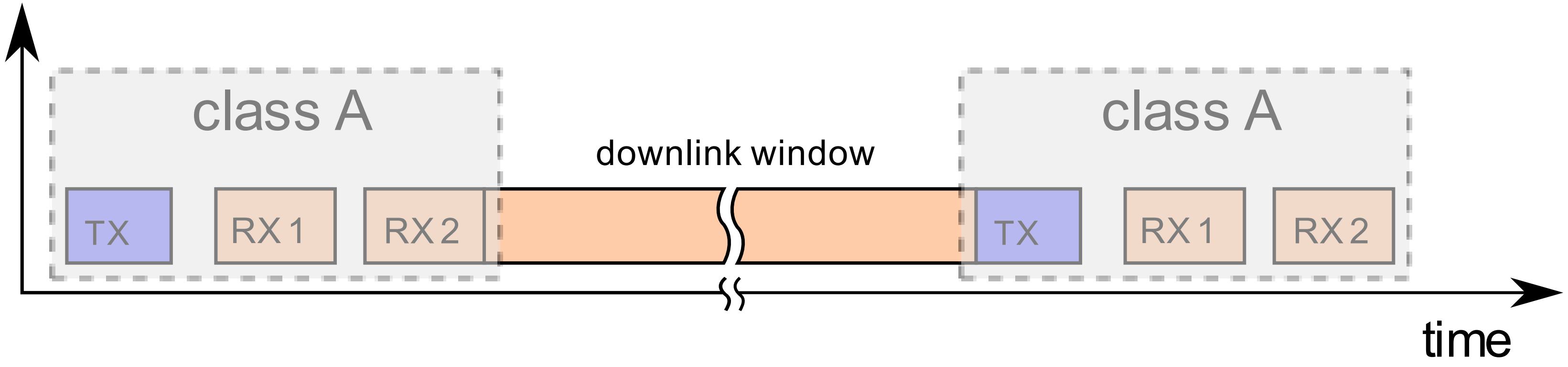
Class B

- Beacon period $128\ s$ by default
- 4096 pings are available per beacon, $\text{ping length} = \ 30\text{ ms}$
- Number of ping slots active for a device must be a power of 2 integer: $\text{pingNb} \ = \ 2^{\text{k}}$ where $0 \ <= \ \text{k} \ <= 7$

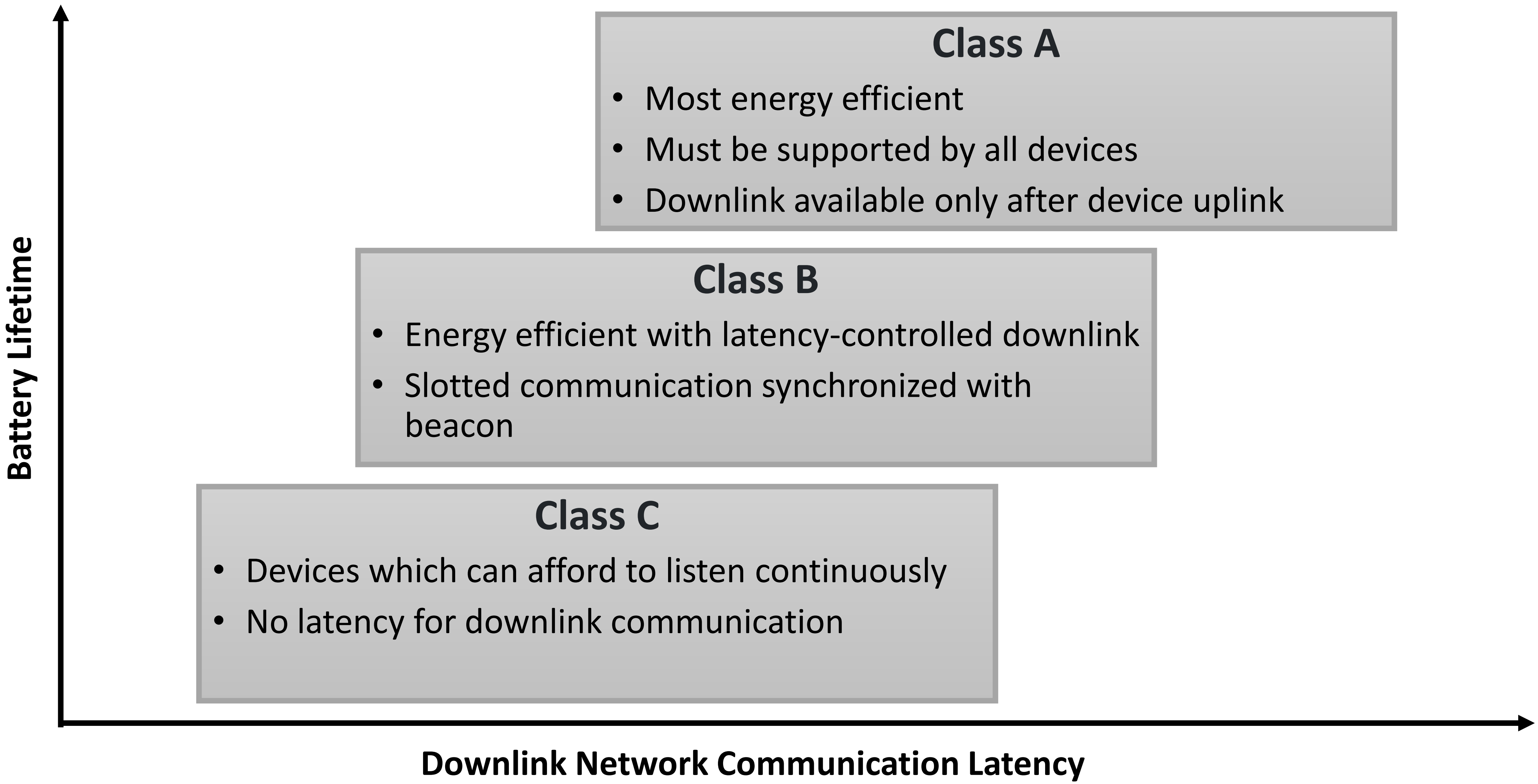


Class C

- Nearly continuously open reception window
- Reception only during uplink not possible
- High power consumption
- Applications: Smart meters with power source available, Main powered actuators



Classes



Class A

- Most energy efficient
- Must be supported by all devices
- Downlink available only after device uplink

Class B

- Energy efficient with latency-controlled downlink
- Slotted communication synchronized with beacon

Class C

- Devices which can afford to listen continuously
- No latency for downlink communication