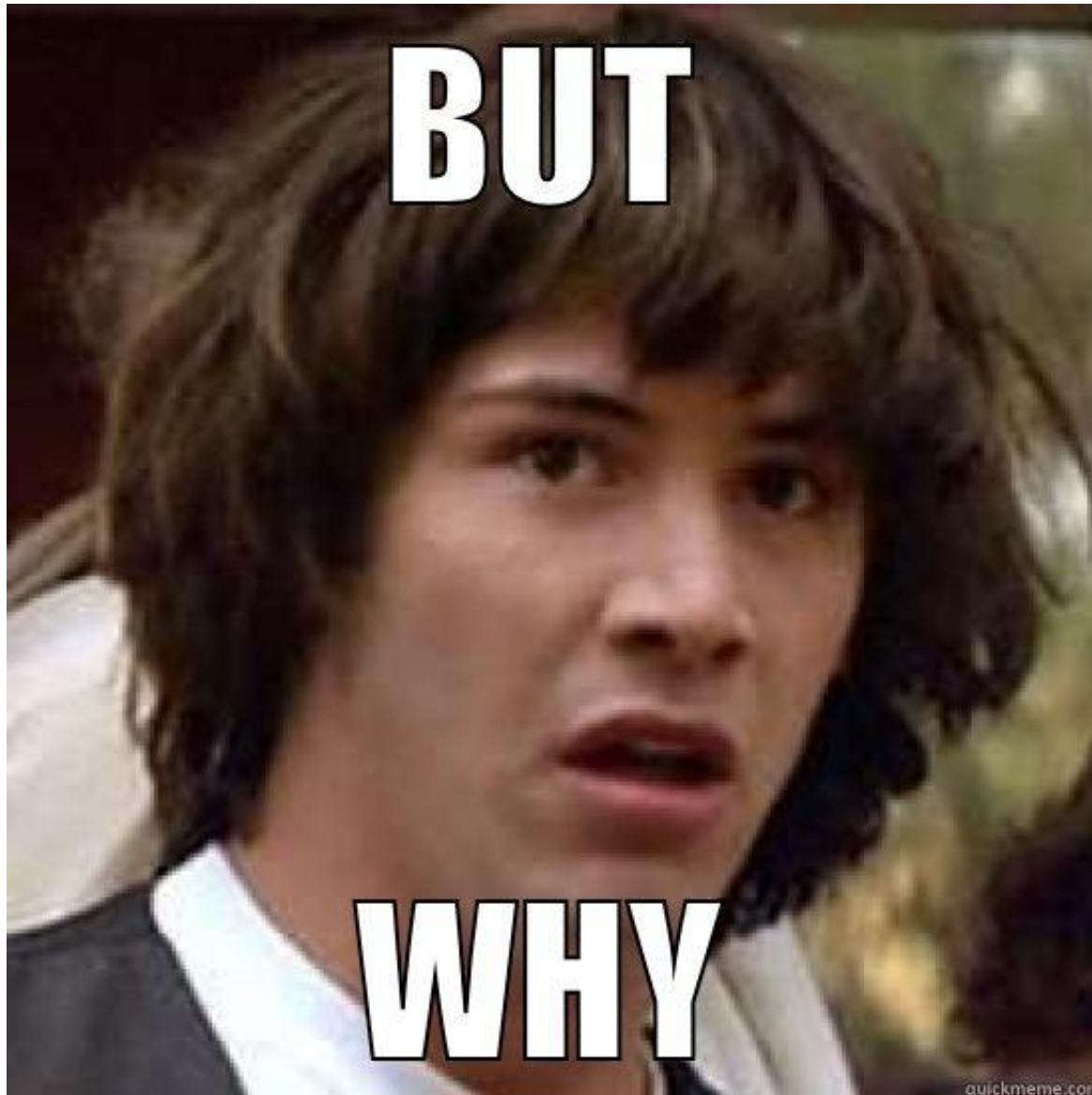


Entwicklung eines echtzeitfähigen Funknetzwerks im Zeitmultiplex mit Bluetooth LE



Wir haben Warum?
unsere eigene Wie?
Mesh
erfunden



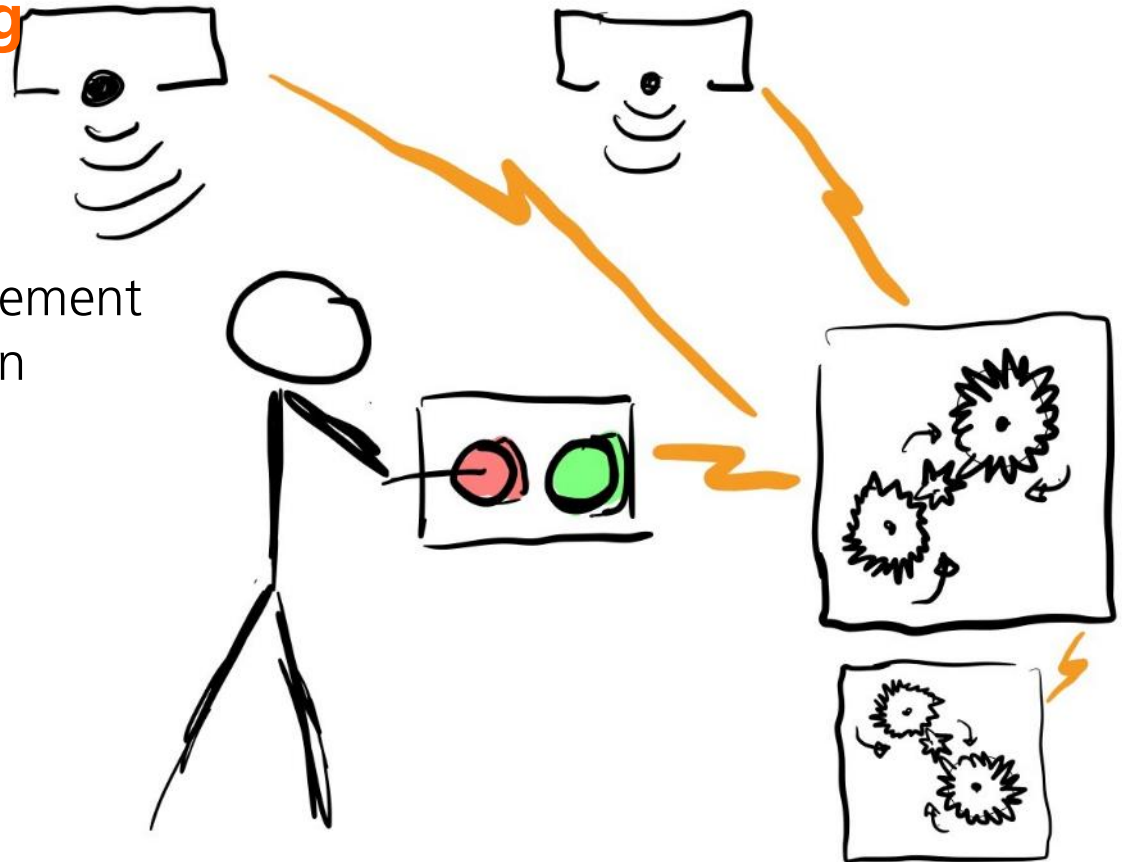
- Warum?
- Wie?

Produktpalette

- Kundenprojekt
- Verschiedene Komponenten in Kommunikationsverbund
- Verkabelung unpraktisch
- Hardwareplattform basiert auf Vorgängermodell

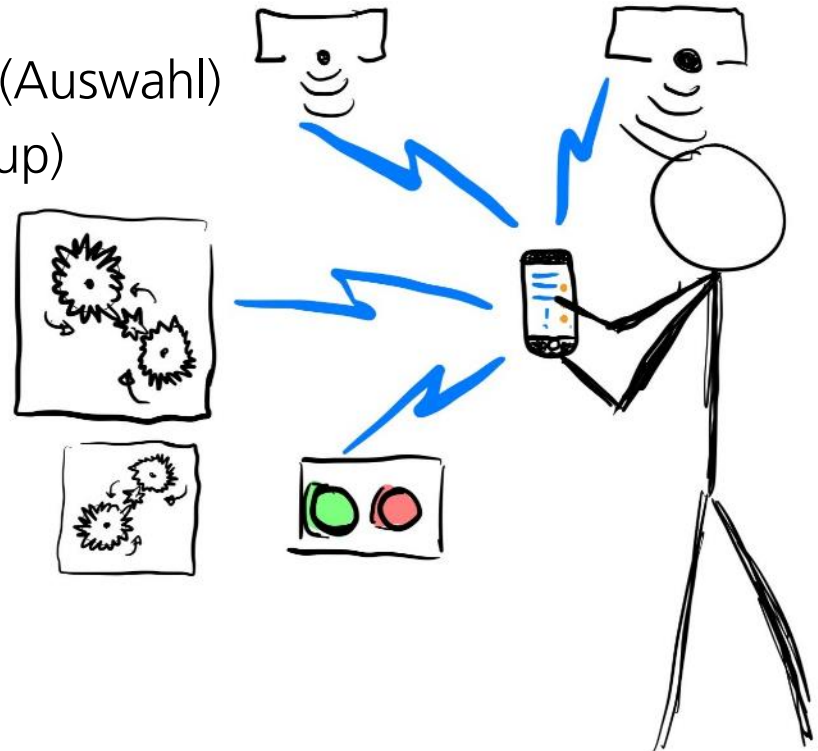
Use Case: Anwendung

- Interaktion mit Bedienelement
→ Abfrage von Sensoren
→ koordinierte Aktion
- Latenz: 100ms



Use Case: Setup

- Verschiedene Sensor/Aktor-Kombinationen möglich
- Konfiguration:
 - Interaktion mit Komponenten (Auswahl)
 - Smart Phone UI (geführtes Setup)



Allgegenwärtiges Bluetooth LE (BLE)



Bluetooth[®]

SMART

- Jedes Smart Phone kann Bluetooth Low Energy
- Aus Nutzer*innen-Sicht einfach
- Getaktete Kommunikation erlaubt Zeitmultiplex mit anderen Funktechnologien
- (schon vorhandene Hardware-Plattform mit Nordic nRF52832 BLE-SoC)

...und die anderen Technologien?

- Muss parallel zu BLE lauffähig sein

- Kandidaten

- Bluetooth Mesh

- Flooding
 - Latenzen >>
 - Stackgrösse >



- Zigbee

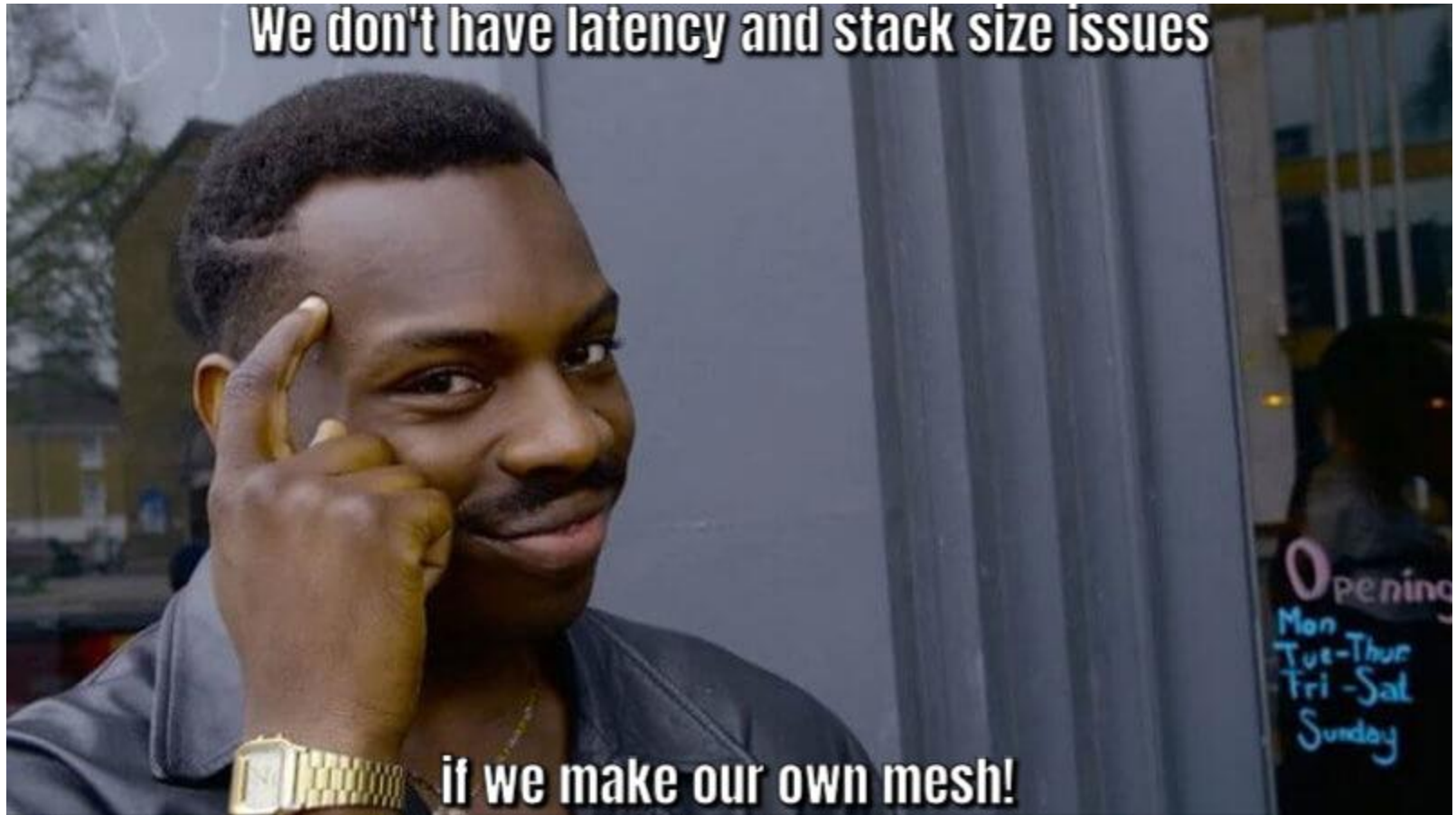
- Routed
 - Latenzen >
 - Stackgrösse >>

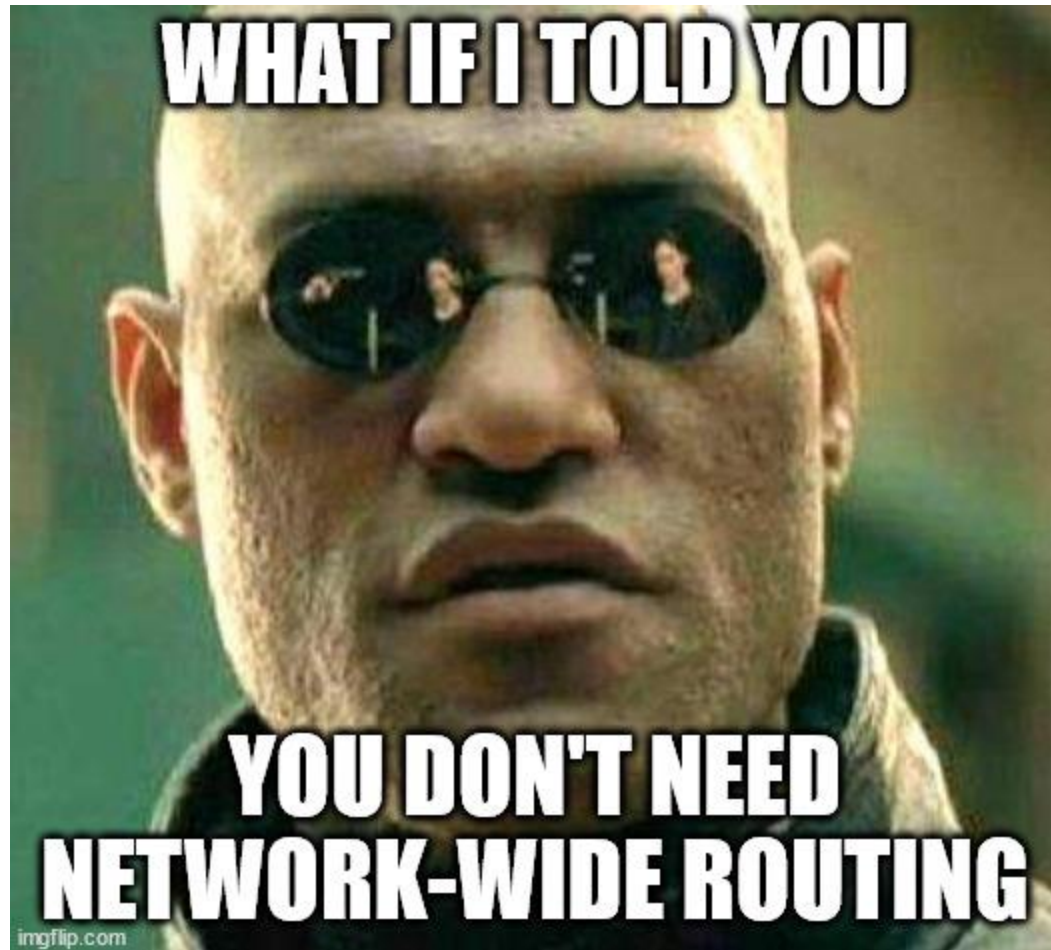


matter



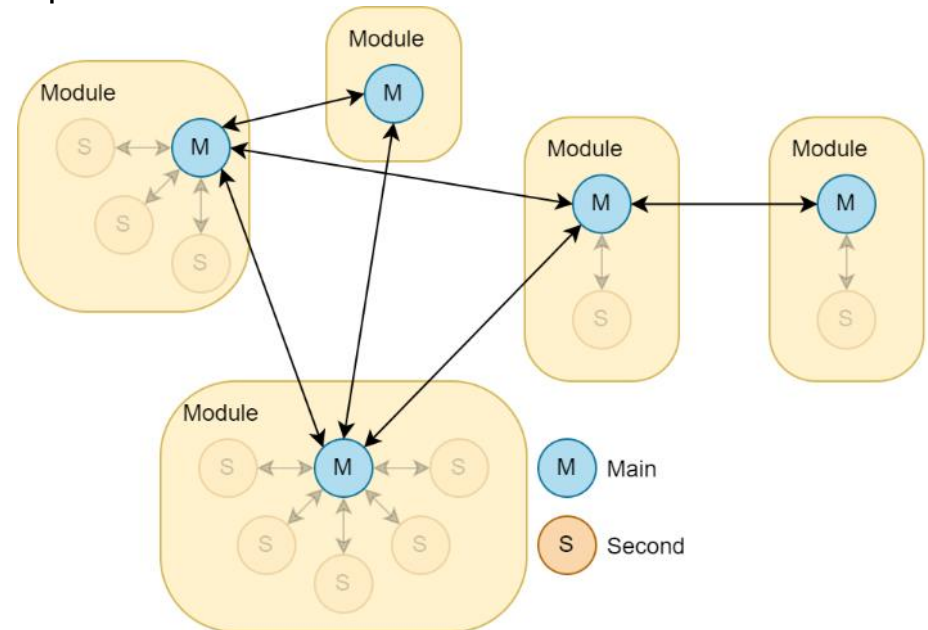
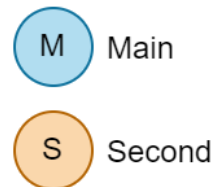
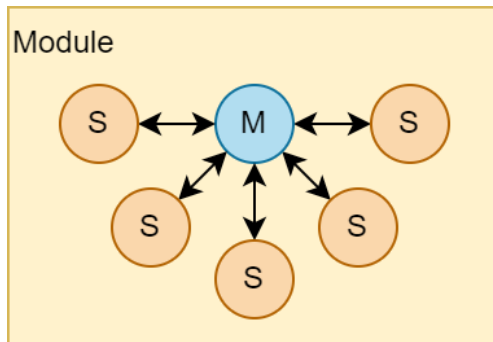
Zigbee
A

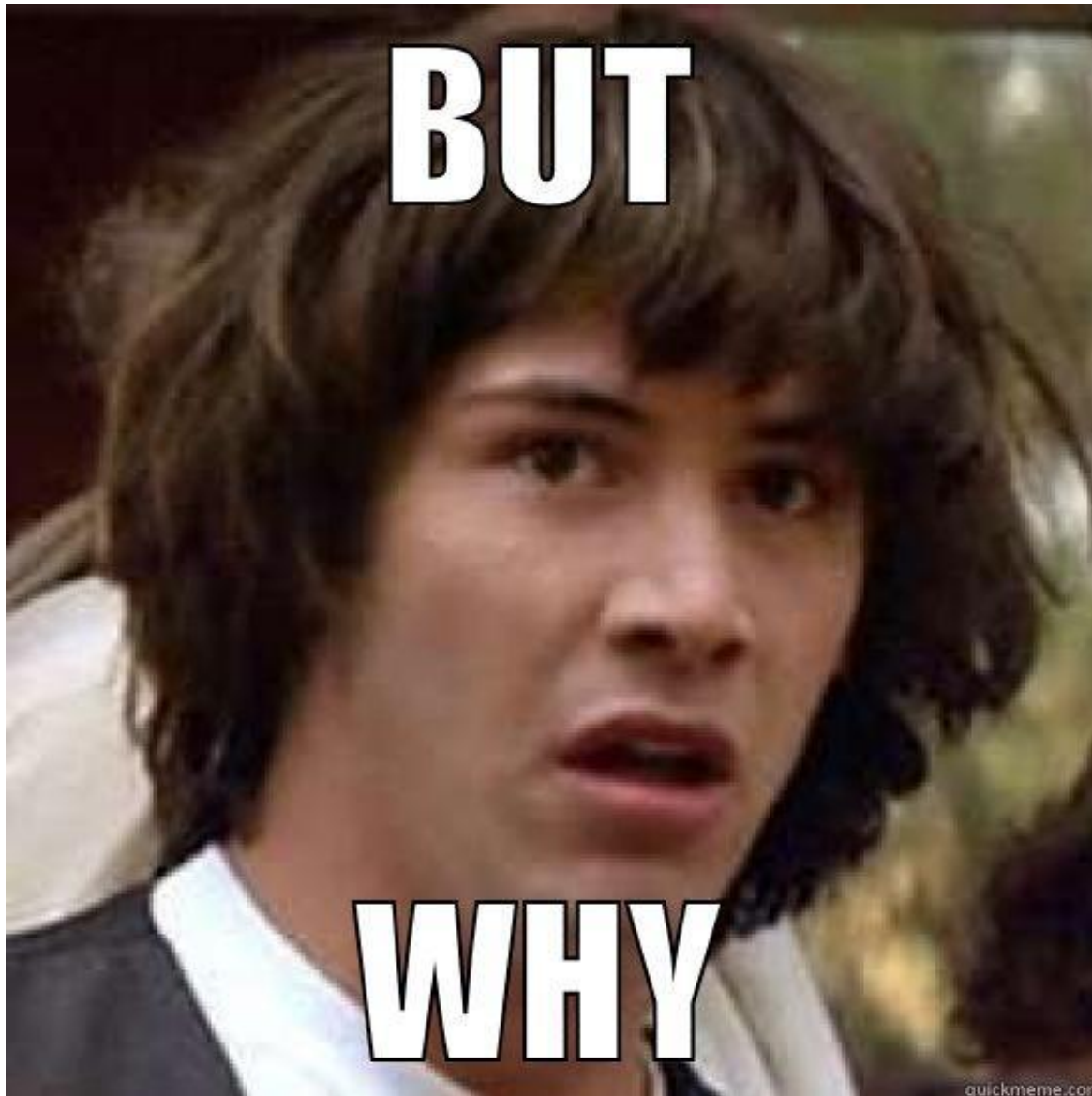




Netzwerkweites Routing unnötig

- Räumlich begrenztes Netzwerk
- Hierarchische Topologie → keine Kommunikation zwischen allen TN nötig
- Routing nur über maximal einen Hop





- **Latenzen**
- Räumliche Begrenzung
- Stack-Grösse
- (Hardware)

How does this thing

work???

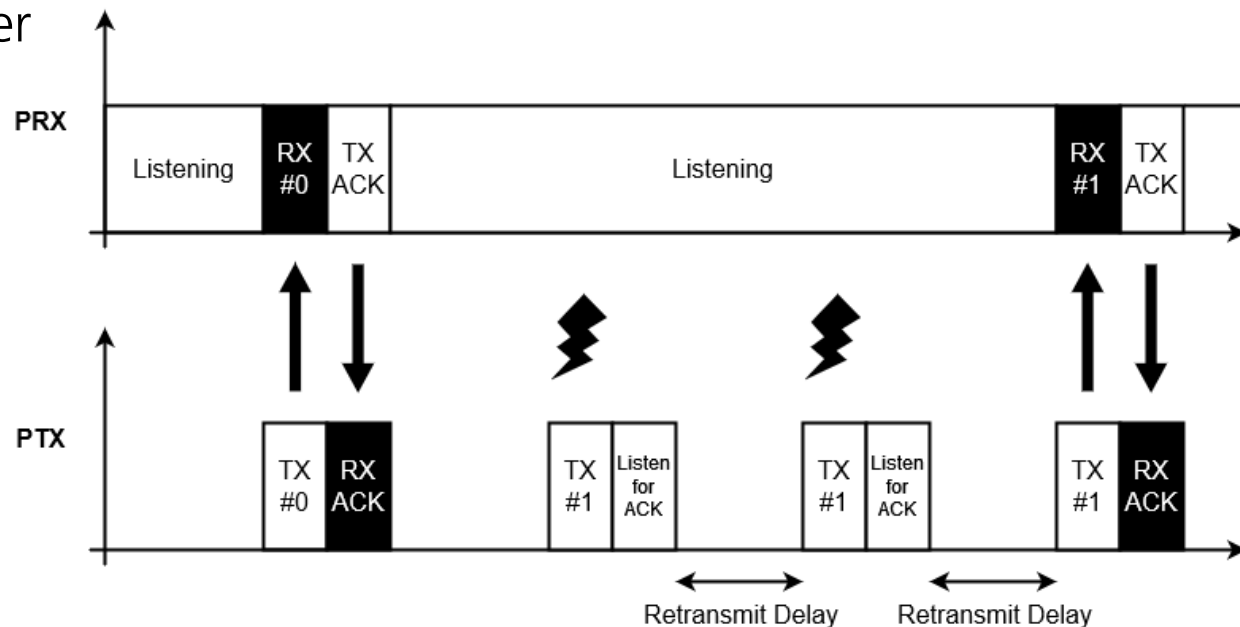
arendi

- Warum?
- **Wie?**

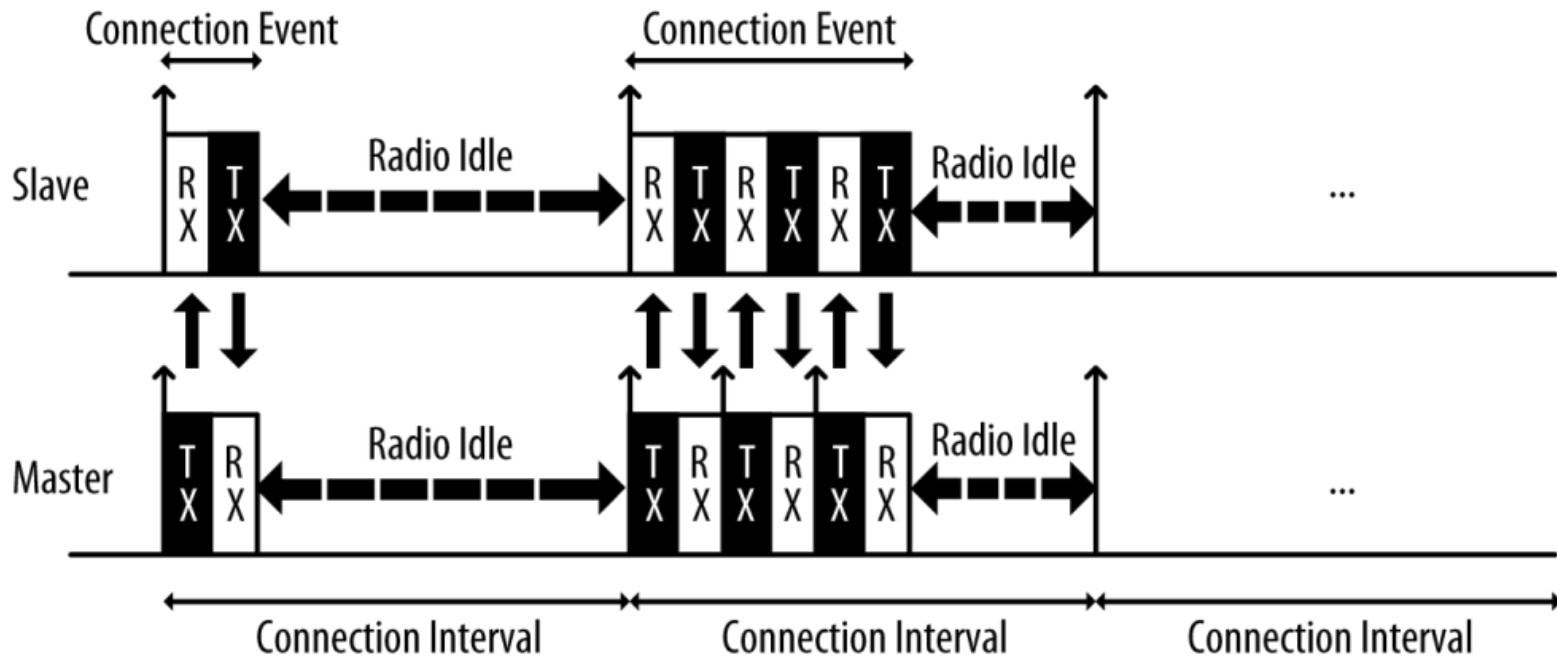
r&d

Enhanced ShockBurst (ESB) enters the stage...

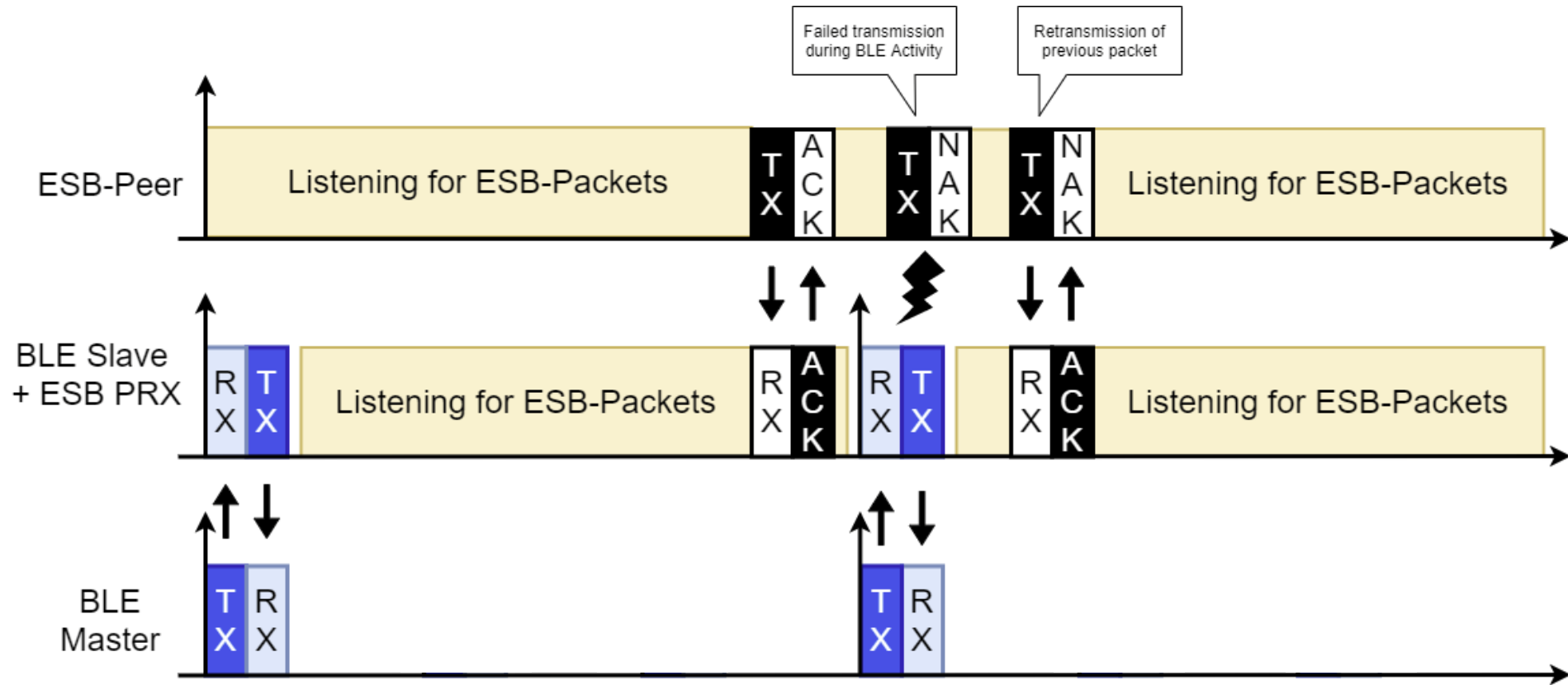
- Proprietäre Funktechnologie von Nordic Semiconductors (Einsatz z.B. in Funktastaturen und -mäusen)
- Zwei Rollen
 - Primary Transmitter
 - Primary Receiver
- Adressiert
- OSI Layer 1-2



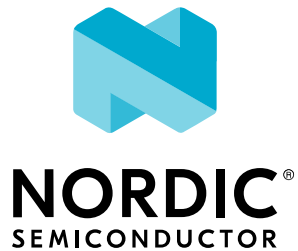
Bluetooth LE: Connections



Bluetooth LE x Enhanced ShockBurst



Nordic Bluetooth Stack

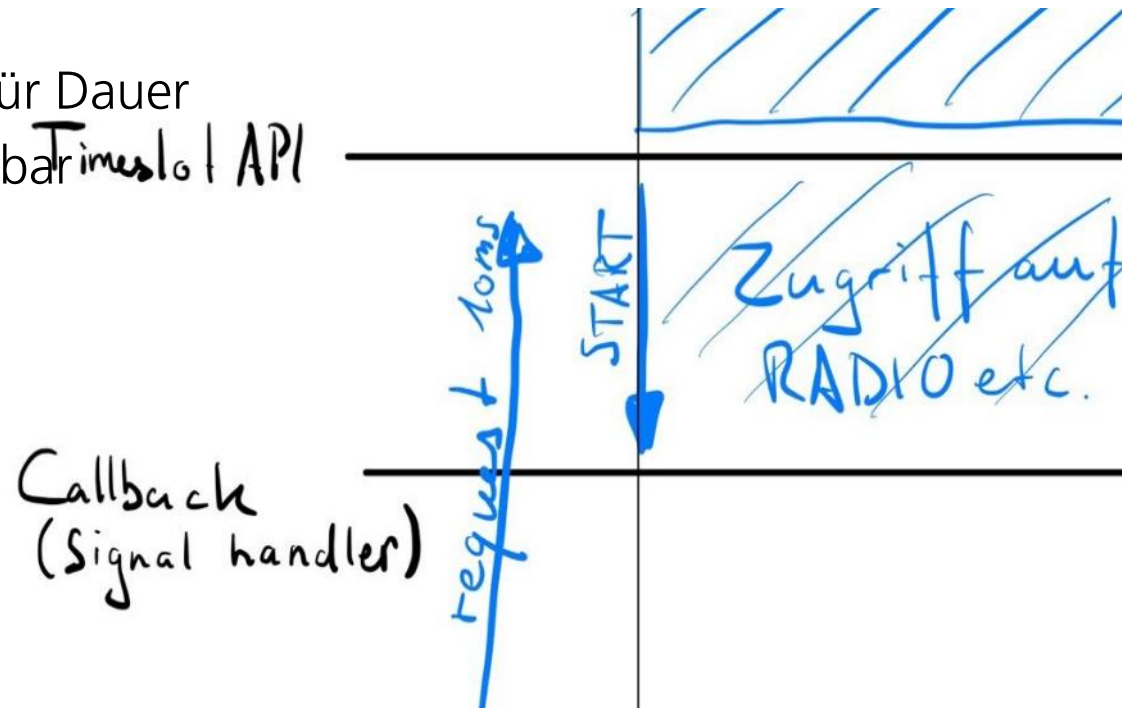


- Proprietärer Bluetooth Stack (Blob → «**Softdevice**»)
- Muss Timing garantieren
- Verwaltet relevante HW-Ressourcen
 - RADIO
 - TIMERO
 - etc.
- Multiprotokoll über «Timeslot API»
 - Verwaltet temporären Zugriff auf HW
- nRF5 SDK → alt
- (In neuem «nRF Connect SDK» ähnlich)
→ MPLS (Multi Protocol Service Layer) bietet ebenfalls Timeslot API

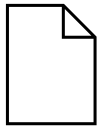


Timeslot API

- Callback-basierte Schnittstelle
- Anforderung von *Timeslots* bestimmter Dauer
- Gewährter *Timeslot*:
 - Zugriff auf RADIO etc. für Dauer
 - Verlängerung verhandelbar
 - Muss selbst beendet werden
- Looks easy...



Examples Zusammenleimen?



nrf_esb.c → ESB-Implementation in nRF5 SDK

+

<https://devzone.nordicsemi.com/nordic/nordic-blog/b/blog/posts/running-micro-esb-concurrently-with-ble>

=

SUCCESS?

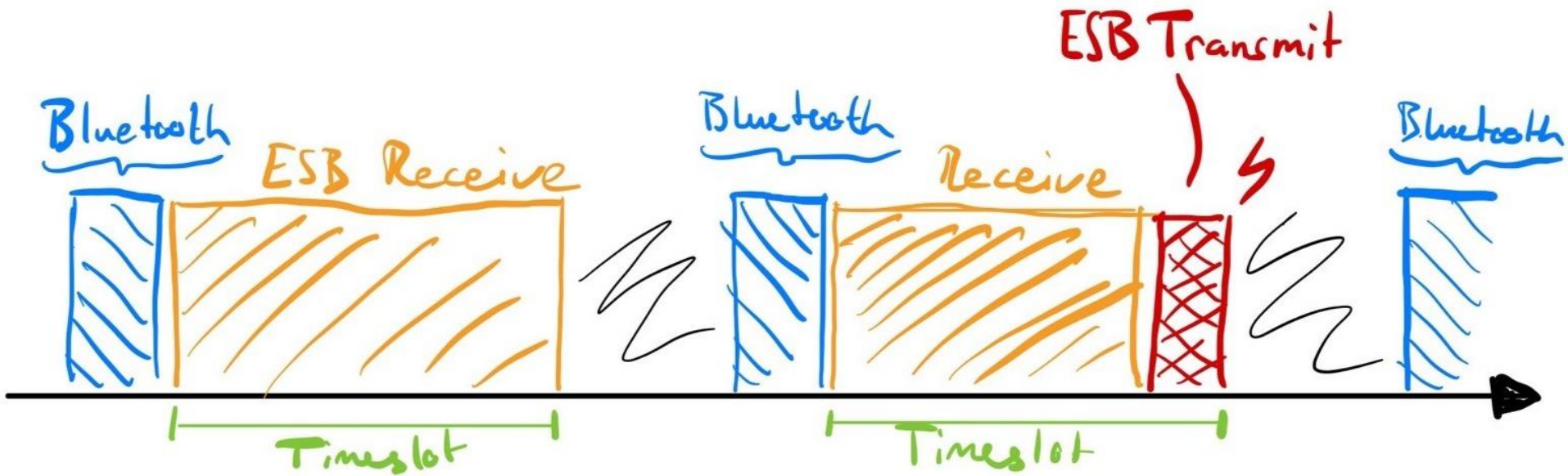
(Ähnliches, Example für nRF Conenct SDK:

https://github.com/inductivekickback/ncs_ble_esb_demo/)

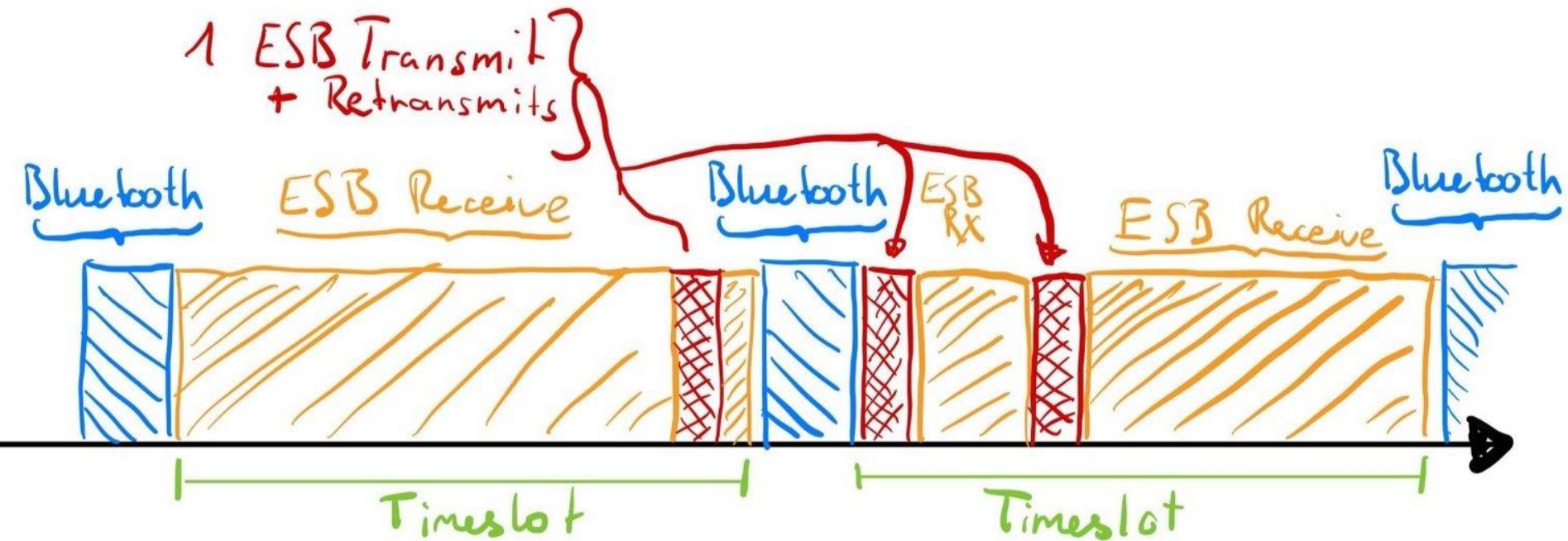
Shortcomings der Examples

- ESB-Implementation:
 - Fehlerhaft
 - Fixe Buffergrößen
 - Senden unterbricht Empfang
- Einbindung in Timeslots nicht optimal
 - Senden verzögert
 - Probleme mit Transaktionen an Timeslot-Grenzen:
 - illegale HW-Zugriffe
 - Senden abgebrochen
- Timeslot-Verfügbarkeit nicht ausgereizt

Shortcomings der Examples



Was wir eigentlich wollen



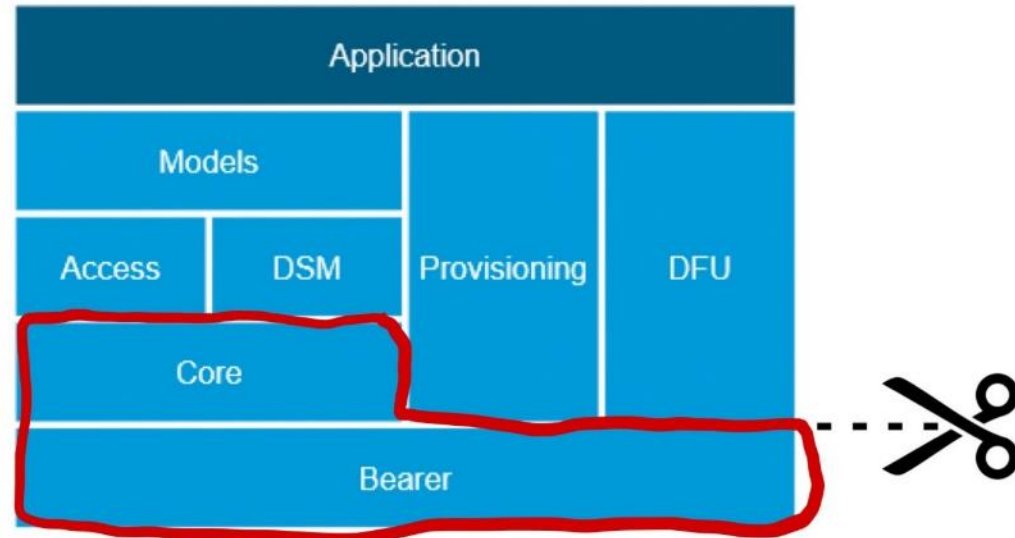
Lösung: nRF5 SDK for Mesh

- nutzt Timeslot API
- Timeslot-Scheduling optimiert
- Empfang wenn «Idle»
- Support

- Verwendete Module:

timeslot: Anforderung der Timeslots

bearer_handler: Scheduling aller Aktivitäten auf Timeslot



Lösung: Architektur

- Netzwerk-Layer und darüber:
Eigenes Protokoll mit Verschlüsselung + primitivem Routing
- Eigene ESB-Implementation auf *bearer_handler*
- Angepasste Mesh-Komponenten

rf_protocol / router / device database

Application data, Netzwerkmanagement, Routing

esb_receiver

Aktiv, solange timeslot "idle"

esb_transmitter

Transmissions: *Aktionen* auf *bearer_handler*

bearer_handler

Scheduling von in ihrer Zeitdauer definierten *Aktionen* innerhalb des Timeslots

timeslot

Optimierte Verhandlung der Timeslot-Dauer

(verschiedene helper)

Probleme?

- Timing Diskrepanzen
- Edge Cases Timeslot Scheduling
- Two Generals' Problem: Paket angekommen, ACK nicht
- Synchronisierung der *device database*
- Testing der Applikation: Aufwändig
- Auch 1-Hop-Routing ist komplex...

Wir sind Ihre Lösung.

Arendi AG
Eichtalstrasse 55
8634 Hombrechtikon
Schweiz

Telefon +41 55 254 30 30
Fax +41 55 254 30 31
www.arendi.ch