Automotive Requirements and Definitions

Helge Zinner (Continental Automotive GmbH),

Oliver Kleineberg (Hirschmann Automation & Control GmbH),

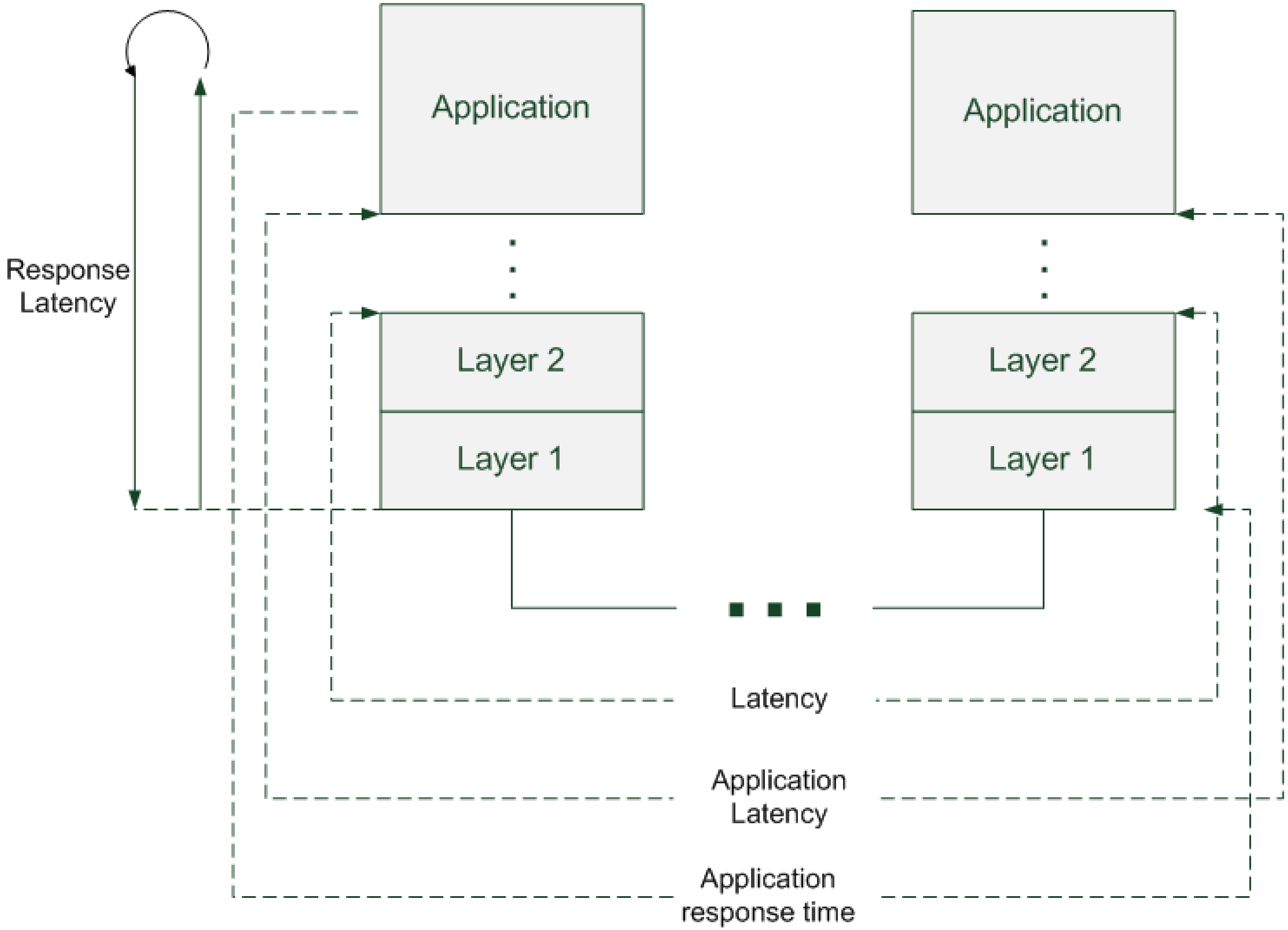
Christian Boiger, Andreas Grzemba (Hochschule Deggendorf)

Definitions
(with respect to automotive networks)

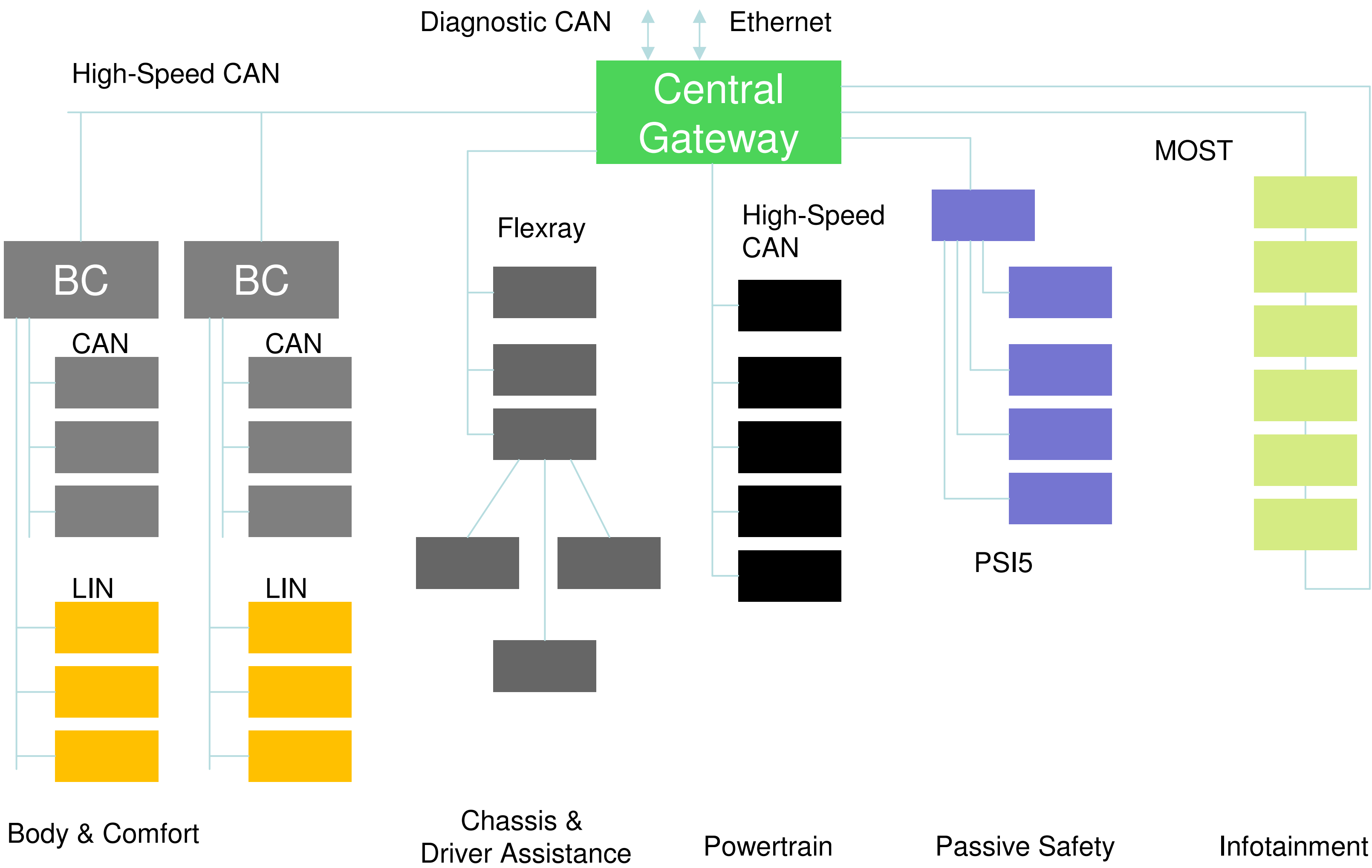
- Latency (Signaling Delay + Layer 2)
  - Time duration for the delivery of data (last bit in – last bit out)
- Application Latency
  - Time duration for the delivery of data from sender to receiver on the application layer
- Response latency
  - Time duration between the reception of data and the time to send a response
  - Important for in-cycle communication (FlexRay)
- Application response time
  - Time between event generation in the application and reception by the receiving station
  - Important for time-triggered network tunneling (how many delay is acceptable)

Definitions

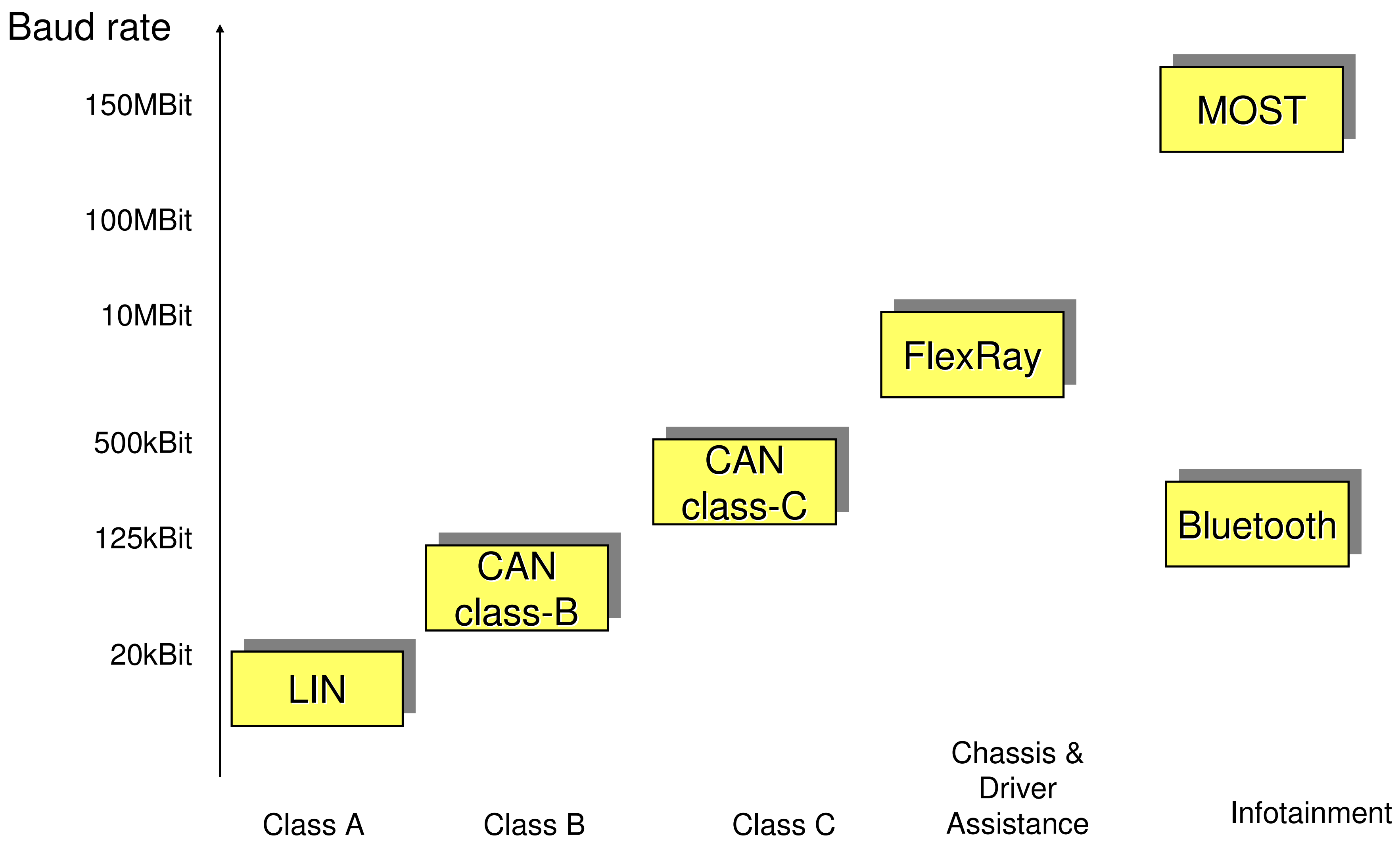
(with respect to automotive networks)



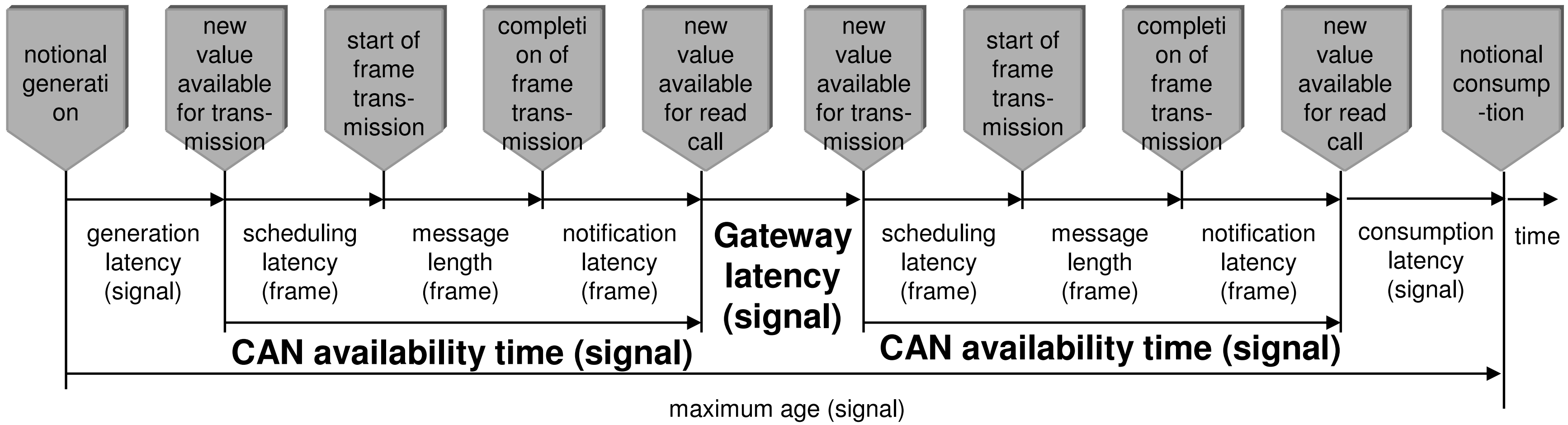
Automotive Network architecture (today)



Automotive Network Classification (SAE)



Determination of latency for multiple networks (CAN)

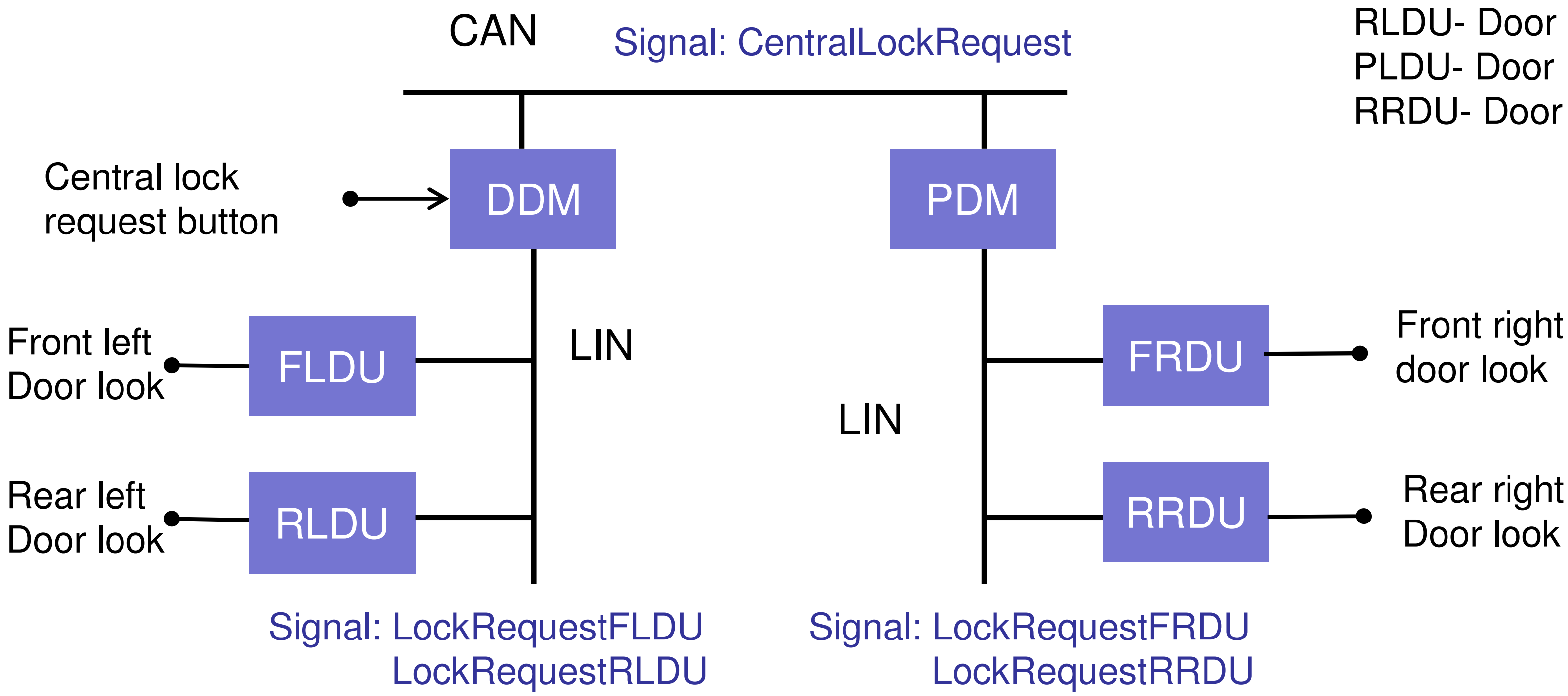


Use Case: Central Locking System

Timing Requirements

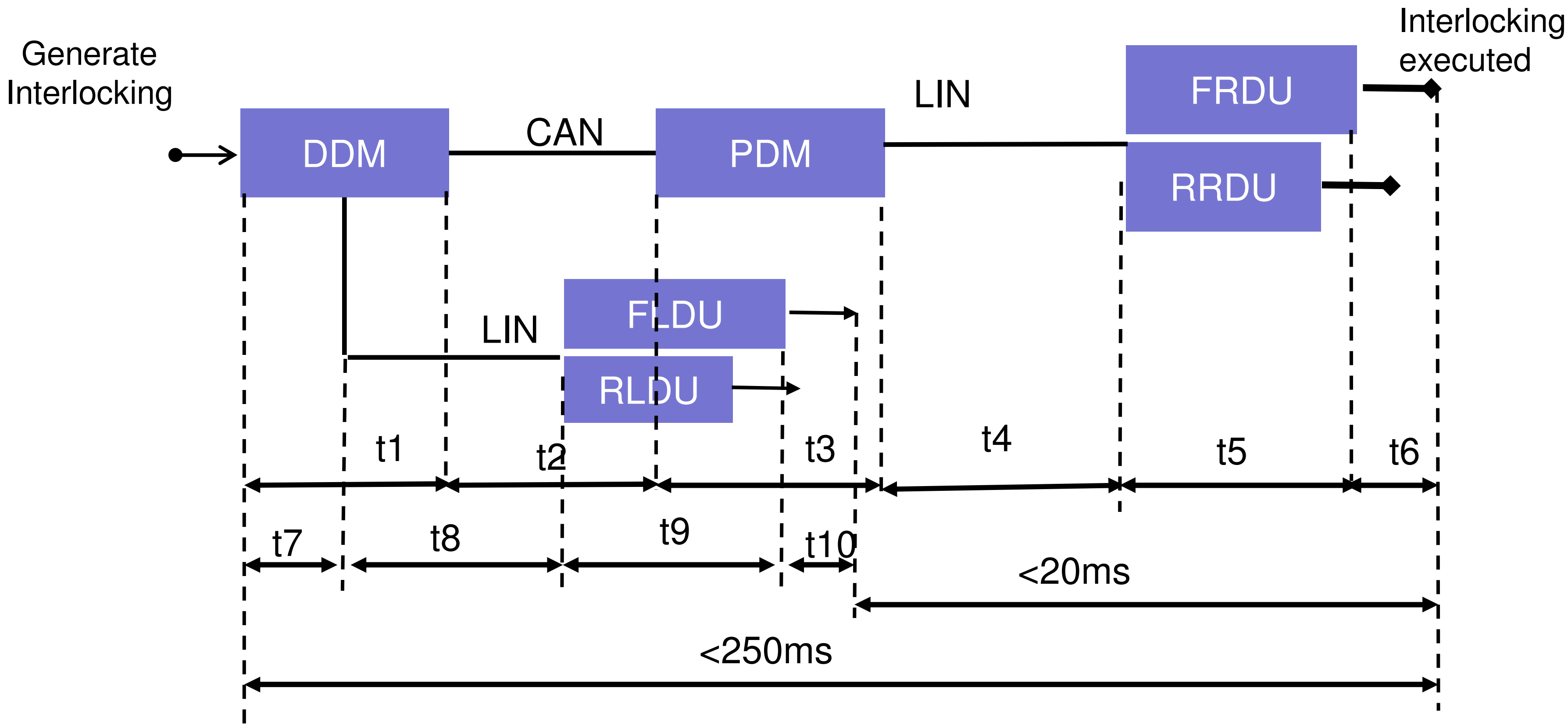
- Interlocking time < 250ms
- Time difference between interlocks < 20ms

DDM - Door module driver
PDM - Door module Co-driver
FLDU- Door module driver
RLDU- Door module rear left
PLDU- Door module Co-driver
RRDU- Door module rear right



Source: Lawrenz /Obermöller ;CAN ; 5.Auflage; VDE-Verlag

Use Case: Central Locking System

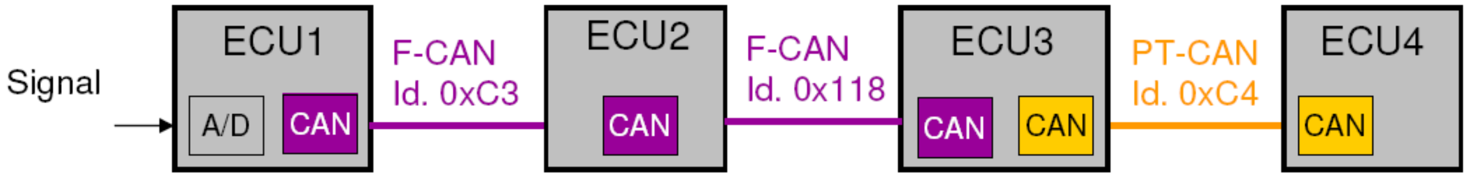


Requirement: timeliness $\sum_{i=1}^{6} ti \leq 250ms$ and $\sum_{i=10}^{7} ti \leq 250ms$

Requirement : simultaneously $\sum_{i=1}^{6} ti - \sum_{i=7}^{10} ti \leq 20\ ms$

Source: Lawrenz /Obermöller ;CAN ; 5.Auflage; VDE-Verlag

CAN Network Delay



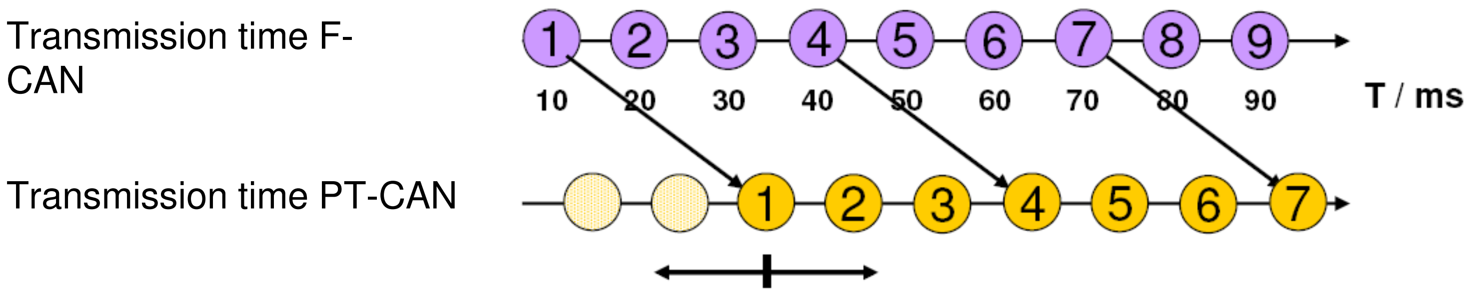
10 ms
Samling time

17.3 ….21.6 ms

0.5 … 10ms
asyncronous processing

Minimal dead time: 17.8 ms
Maximum dead time: 41.6 ms

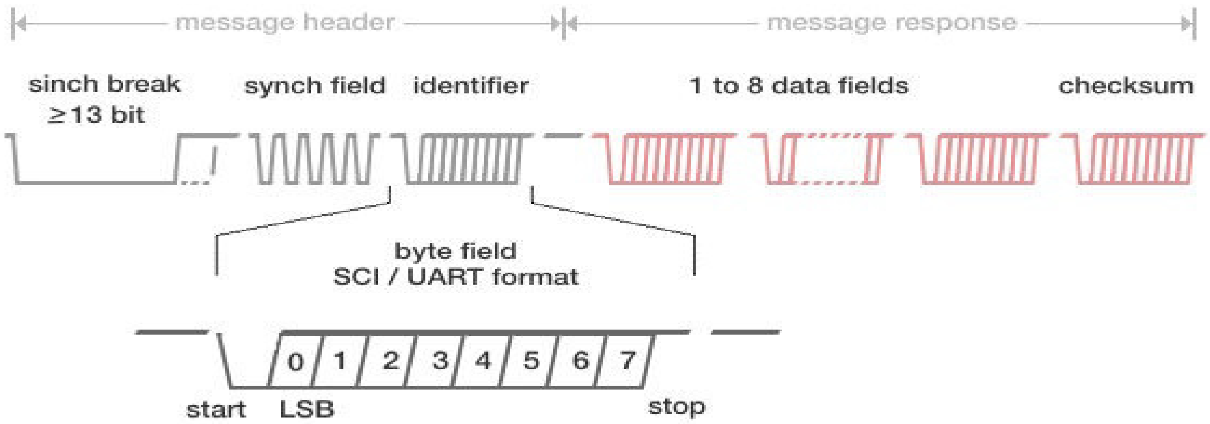
dead time > 4 times sampling time
Jitter >> sampling time



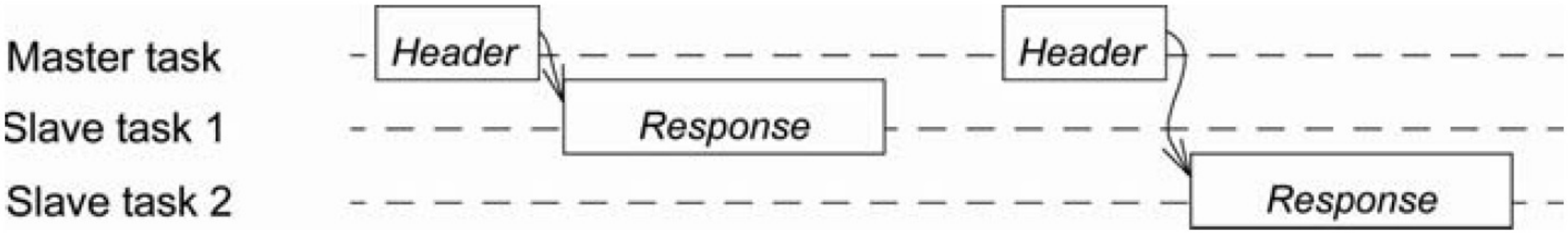
Processing at receiver: Jitter > 23ms

Source: Josef Berwanger, BMW, Ludwigsburg , 16.Mai.2006

LIN: Local Interconnect Network



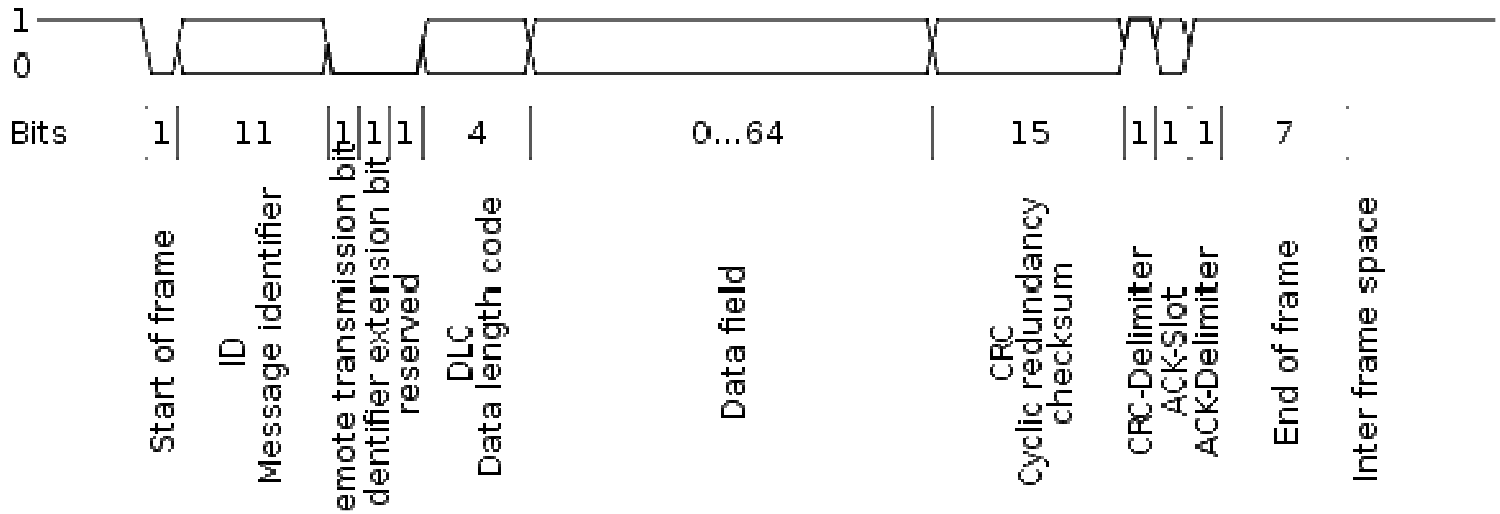
- 1. Master:
  - Frame start: 13 Bit dominant level (sync break)
  - Synchronization: alternating 1-0 bits (sync field)
  - 6 Bit message ID
- 2. Master or slave is addressed by message ID and send its payload:
  - 1-8 data bytes
  - Check sum
- Communication directions (always initiated by master)
  - From master to one or multiple slaves
  - From one slave to master and/or slaves



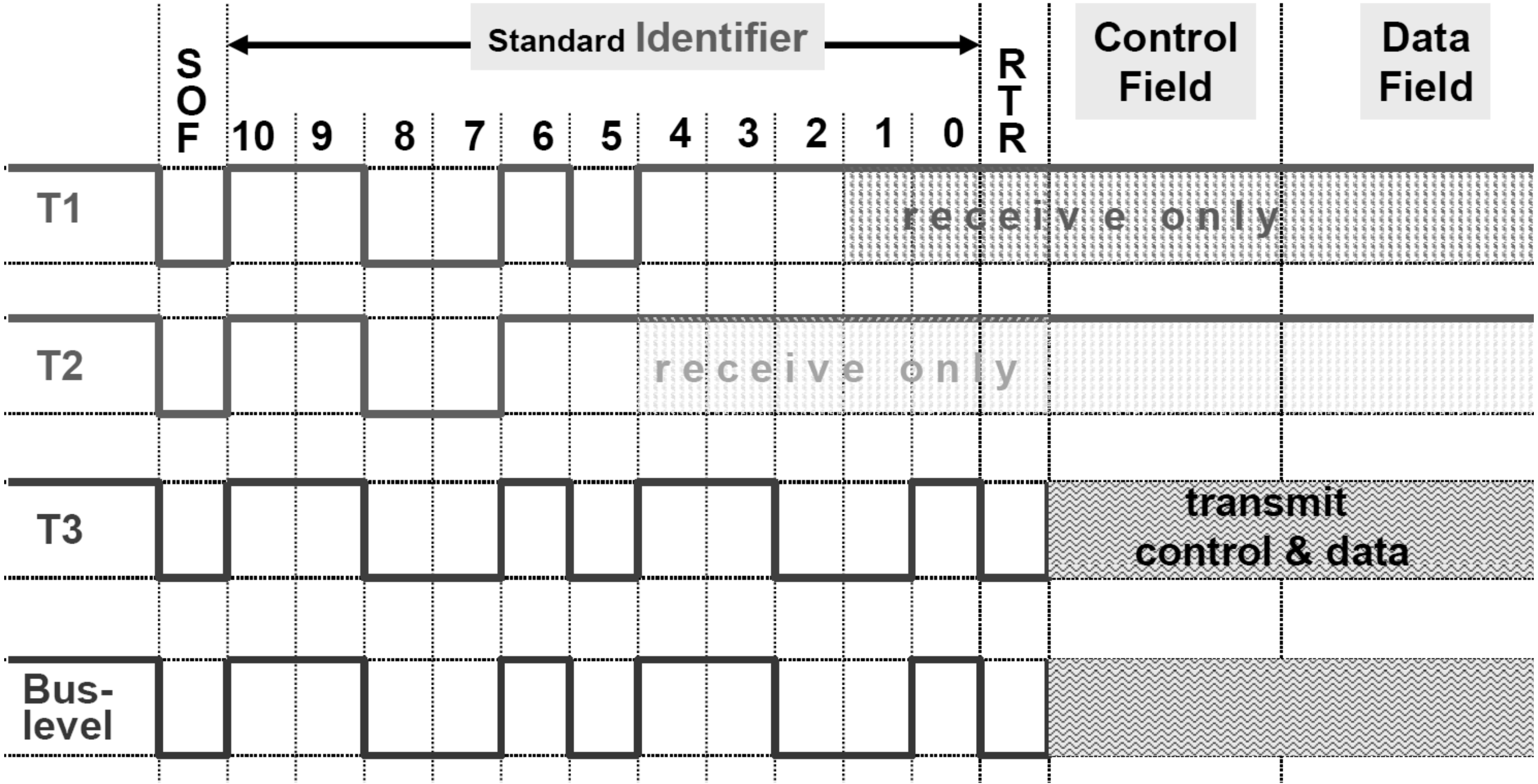
Source: http://images.tecchannel.de/images/tecchannel/bdb/361598/890.jpg

CAN-Frame

CAN-ID: Unique ID used for arbitration and identification of data field content
Remote Trans. Bit: By setting this bit another node is requested to send a message

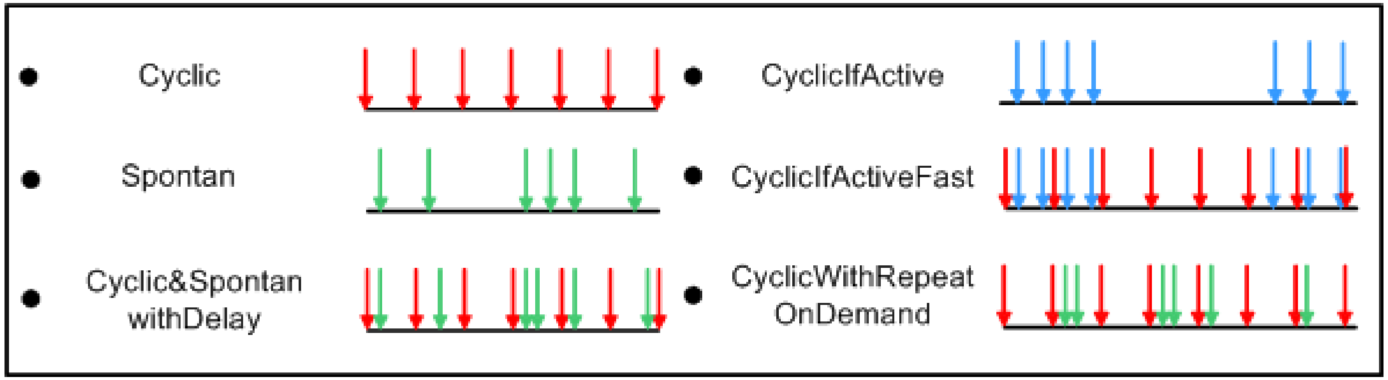


CAN-Arbitration



Typical Application Send Types

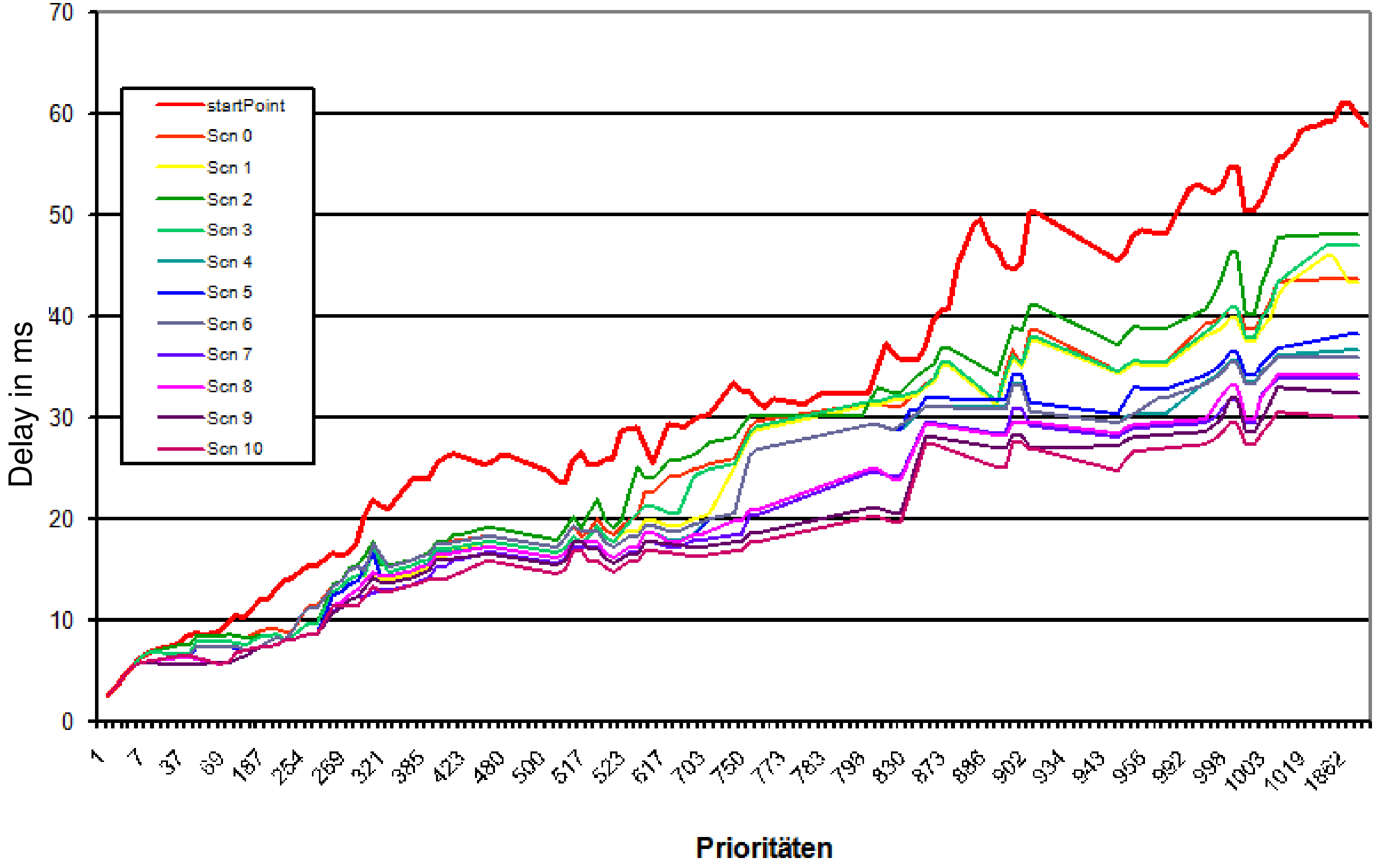
- CAN-DBC (pseudo cylic)



- AUTOSAR I-PDU Transmission Modes:
  - Direct/N-Times: PDU will be sent immediately N-times
  - Periodic: periodic transmission
  - Mixed: Mixture of Direct/N-Times and Periodic
  - None: No transmission via AUTOSAR COM

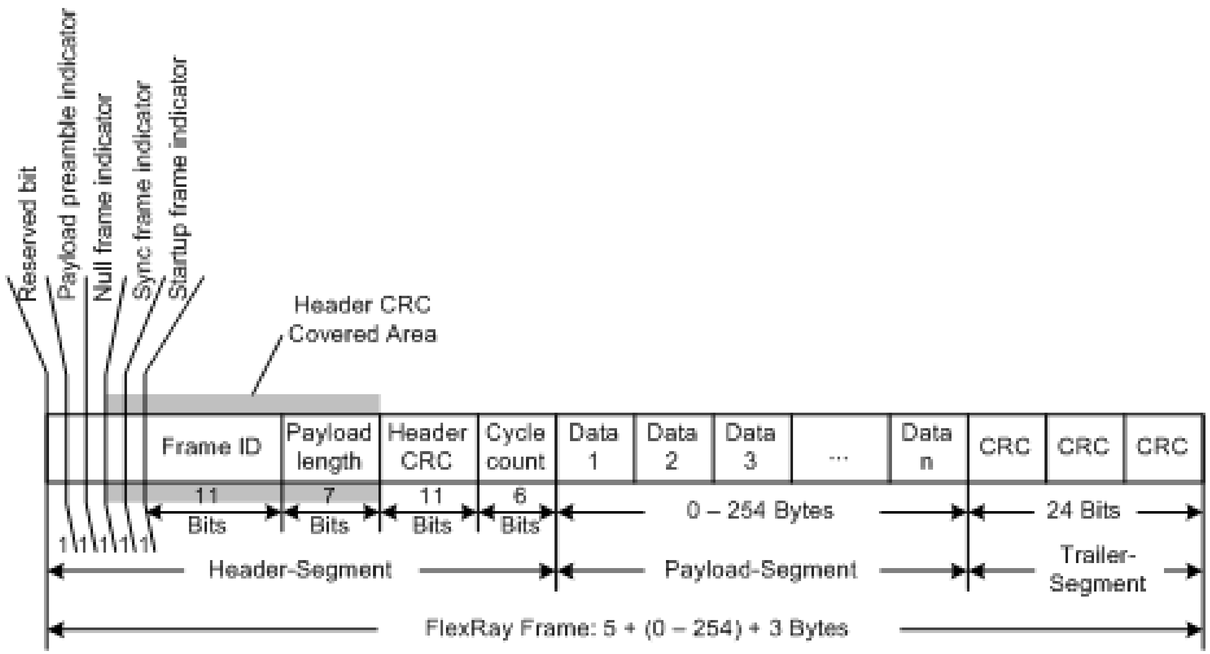
Example: Arbitration Effect on Response Time

- Worst-case response time can be analyzed in CAN-networks
- Example with typical automotive communication matrix:

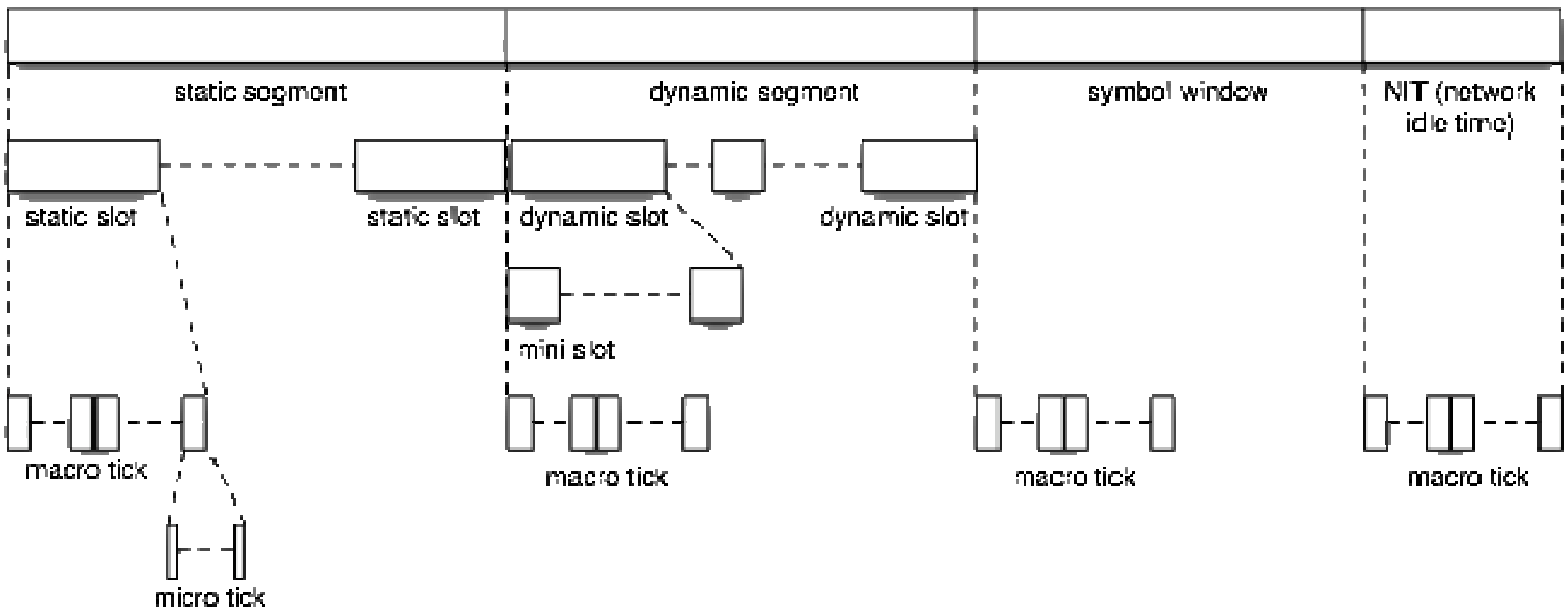


FlexRay-Frame

- Payload preamble indicator: denotes whether the payload contains control information
- Null frame indicator: if bit is set, the payload contains valid data
- Up to 254 byte payload data

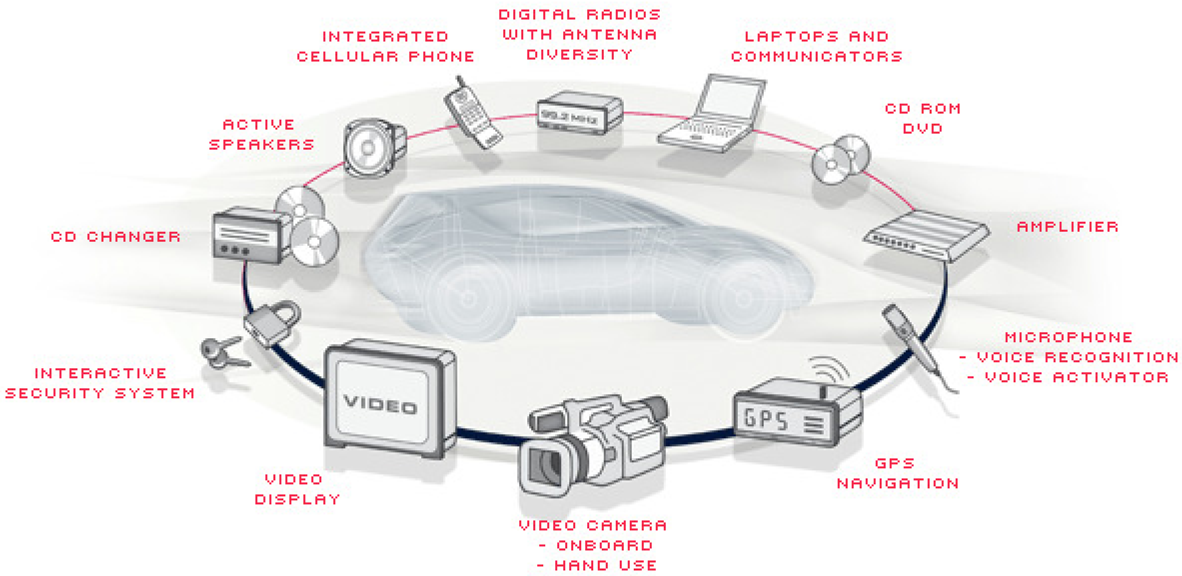


FlexRay Slots and Cycle



- Single or dual channel operation: second channel can be used for redundancy or additional bandwidth
- TDMA- with a kind of Round-Robin-Arbitration in the dynamic segment
- Data rate: 10 Mbit/s
- Typical sending cycle 5ms (possible: cycle times from 16µs to 16ms)
- Redundancy currently not used by OEMs

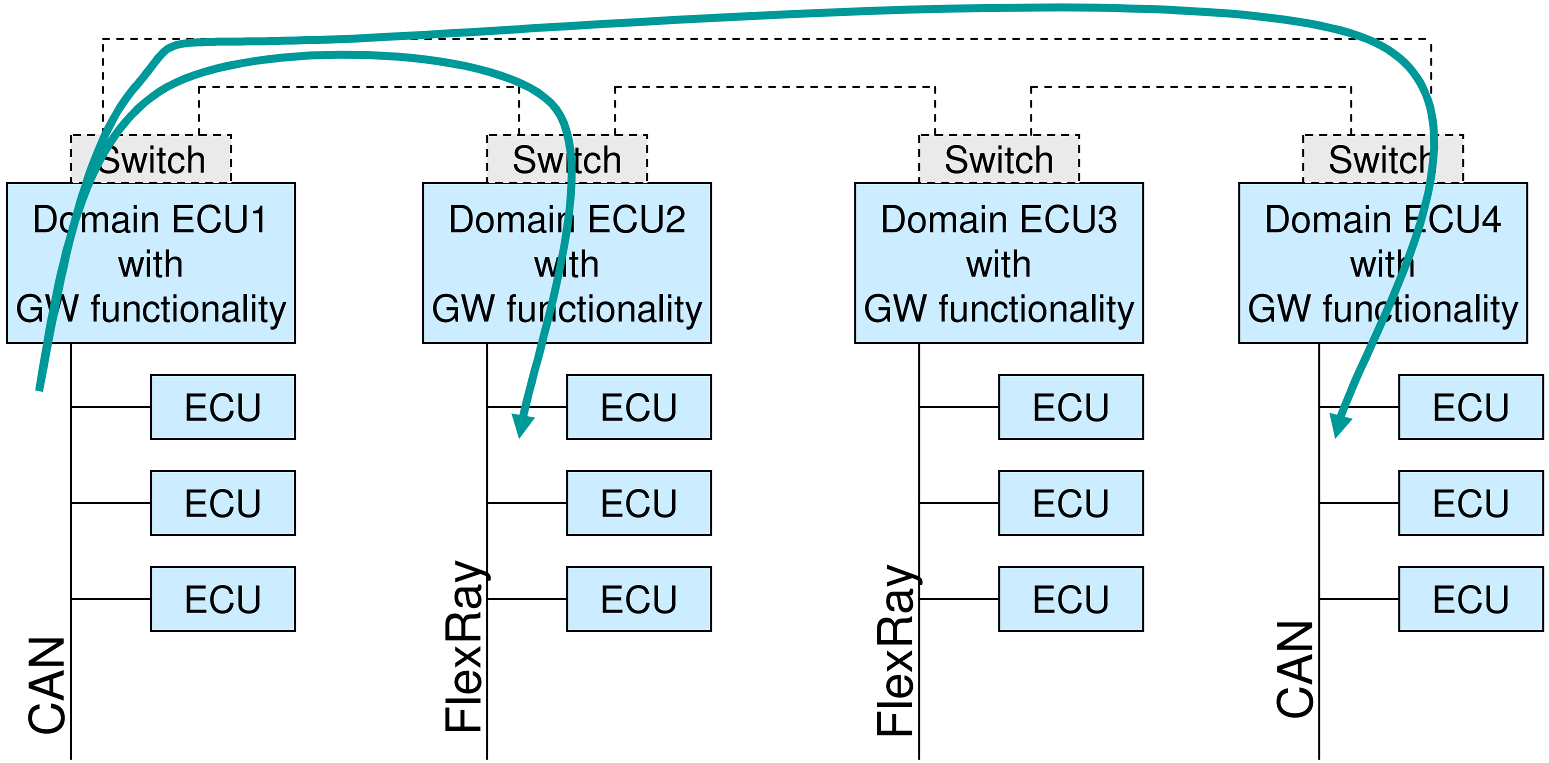
MOST



- Time triggered protocol
- Ringbus system
- Datenrate 25Mbit/s, 50Mbit/s, 150Mbit/s
- optical (25, 150) or electrical (50)
- Specifies physical layer, application framework and higher layer proocols

* Taken from http://www.mostcooperation.com

Use-Case1a: Ethernet as a Backbone „Bus“ (Message Extraction)



Transmission of data from a CAN bus over several gateways into arbitrary bus systems:

Messages on CAN-Bus:

CAN-Hdr1	Payload1	CAN-Hdr2	Payload2	CAN-Hdr3	Payload3

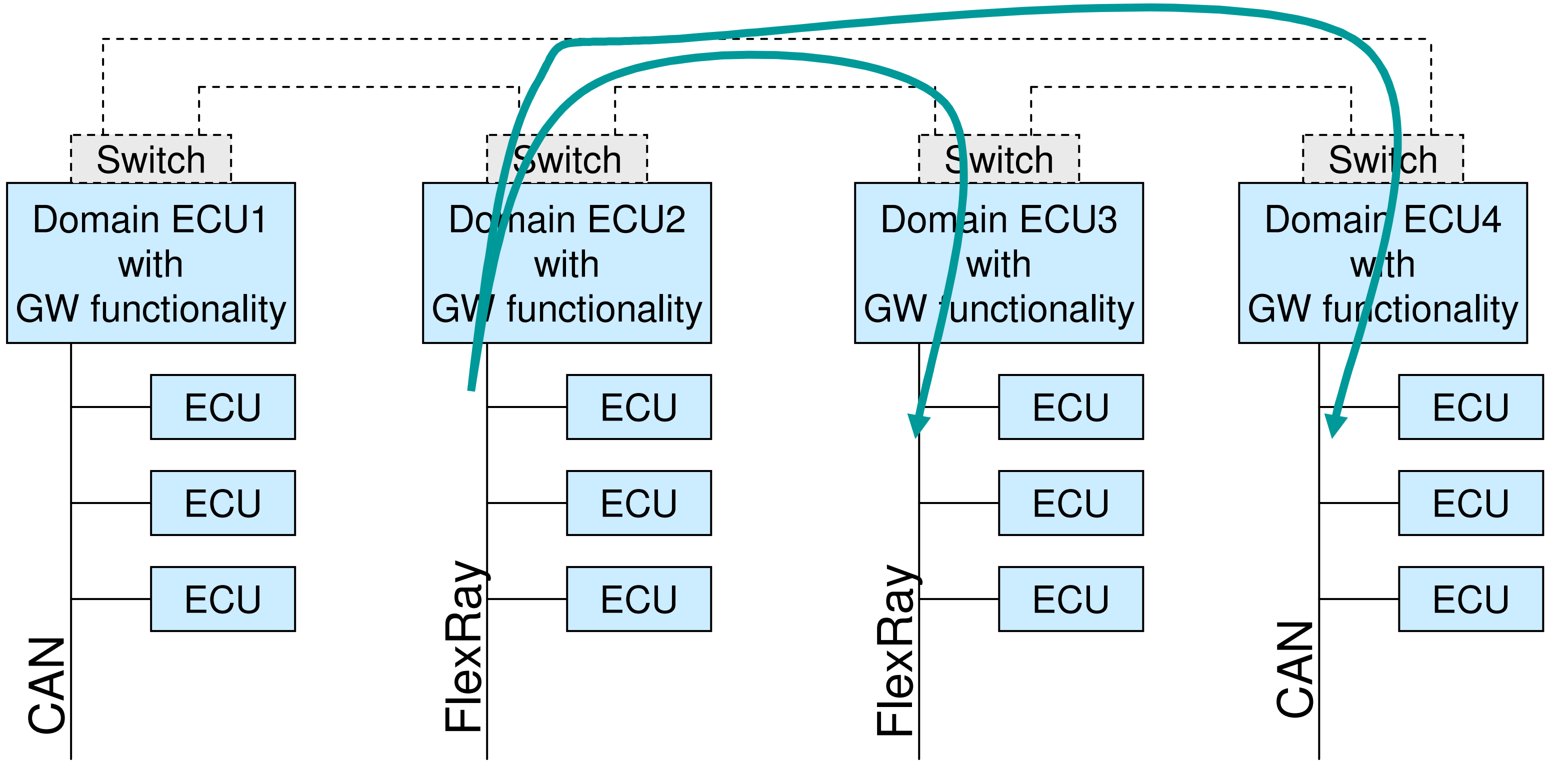
Messages in Ethernet-NW (v1):

ETH/1722-Hdr	CAN-ID1	Payload1	CAN-ID2	Payload2	CAN-ID3	Payload3

Messages in Ethernet-NW (v2):

ETH/1722-Hdr with unique ID	Payload with data from Payload 1, 2, and 3

Use-Case1b: Ethernet as a Backbone „Bus“ (Message Extraction)



→ Transmission of FlexRay payload without the FlexRay-header in a 1722 frame:

Messages on FR-Bus:

FR-Hdr1	Payload1	FR-Hdr2	Payload2	FR-Hdr3	Payload3

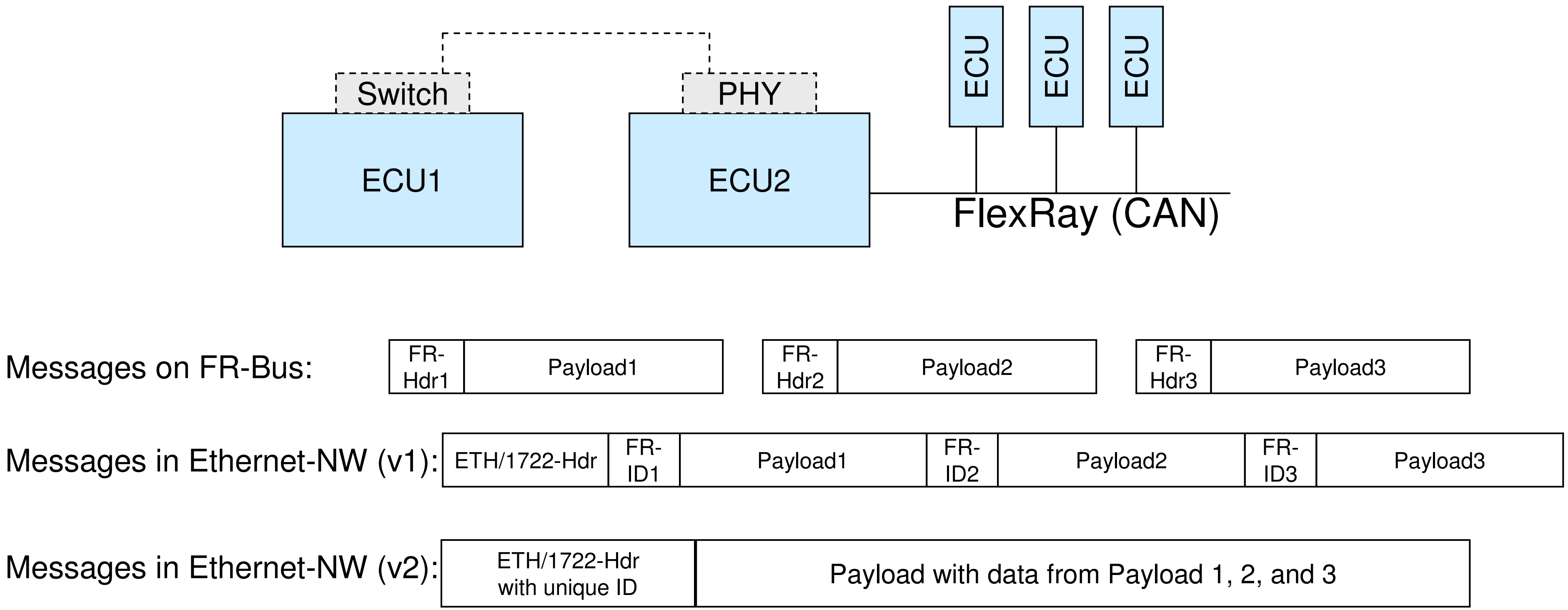
Messages in Ethernet-NW (v1):

ETH/1722-Hdr	FR-ID1	Payload1	FR-ID2	Payload2	FR-ID3	Payload3

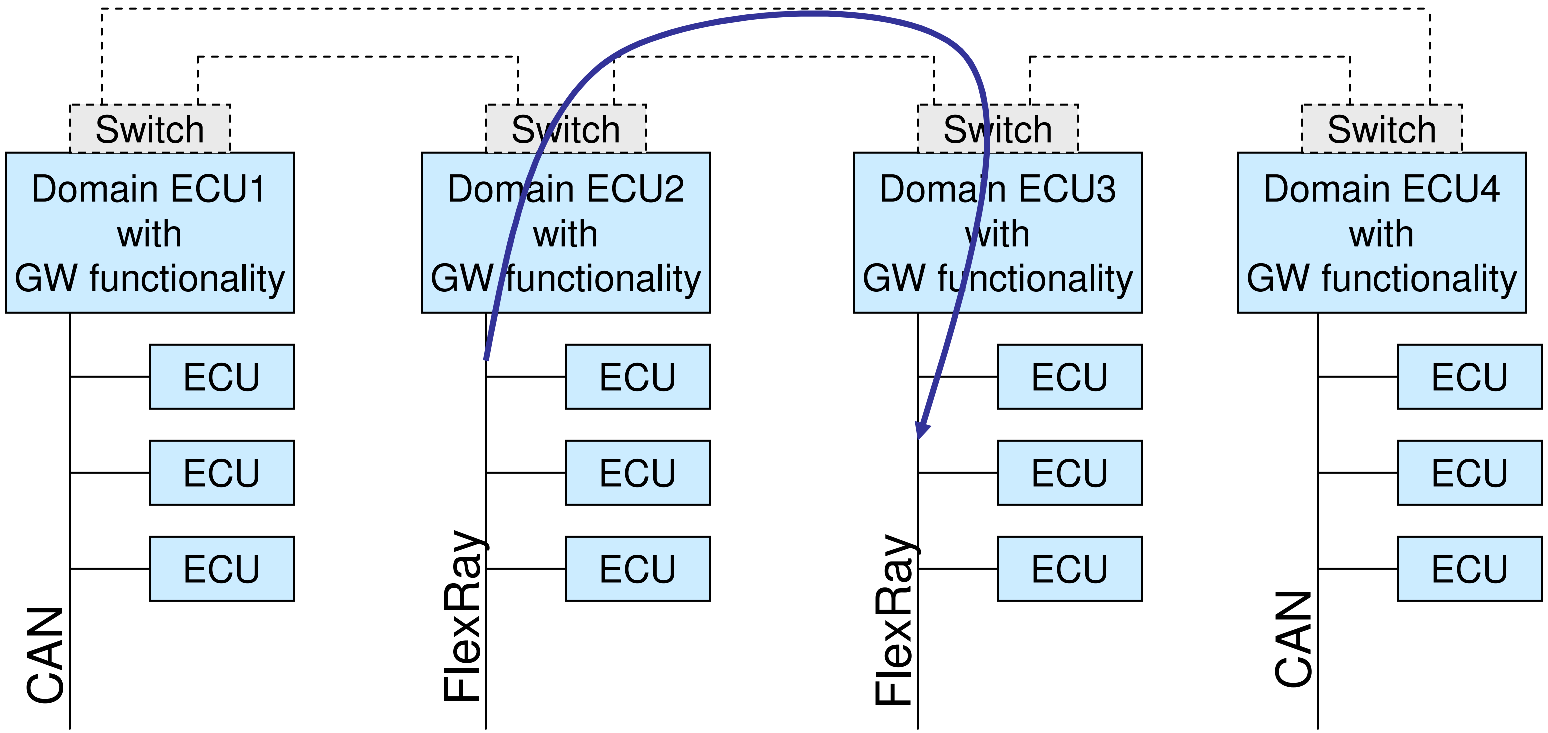
Messages in Ethernet-NW (v2):

ETH/1722-Hdr with unique ID	Payload with data from Payload 1, 2, and 3

Use Case1c – CAN/FlexRay-Ethernet (Message Extraction)



Use-Case2: Ethernet as a Backbone „Bus“ (Encapsulation)



Transmission of FlexRay-Frames between FlexRay-Clusters where the timing information (slot/cycle count) of a FR-message is required

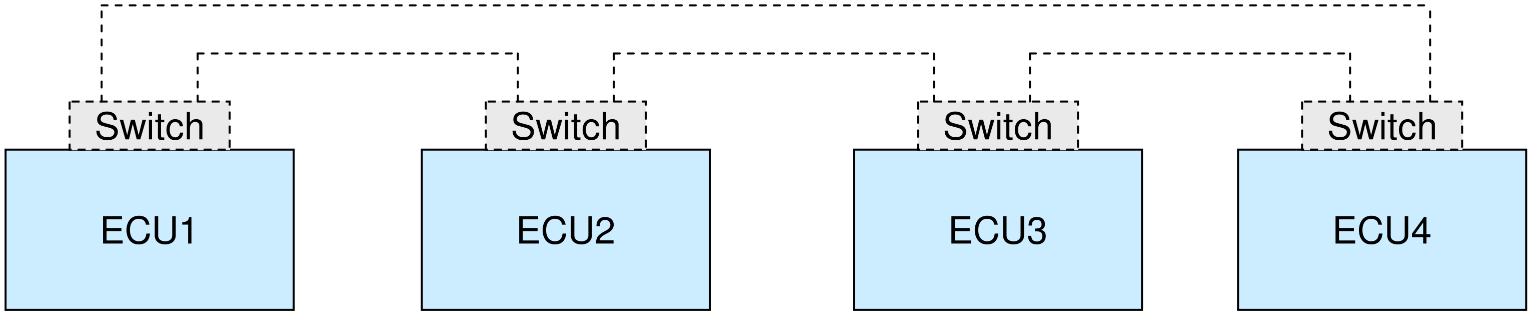
Messages on FR-Bus:

FR-Hdr1	Payload1	FR-Hdr2	Payload2	FR-Hdr3	Payload3

Messages in Ethernet-NW (v1):

ETH/1722-Hdr	FR-Hdr1	Payload1	FR-Hdr2	Payload2	FR-Hdr3	Payload3

Use-Case3: Pure Ethernet Network



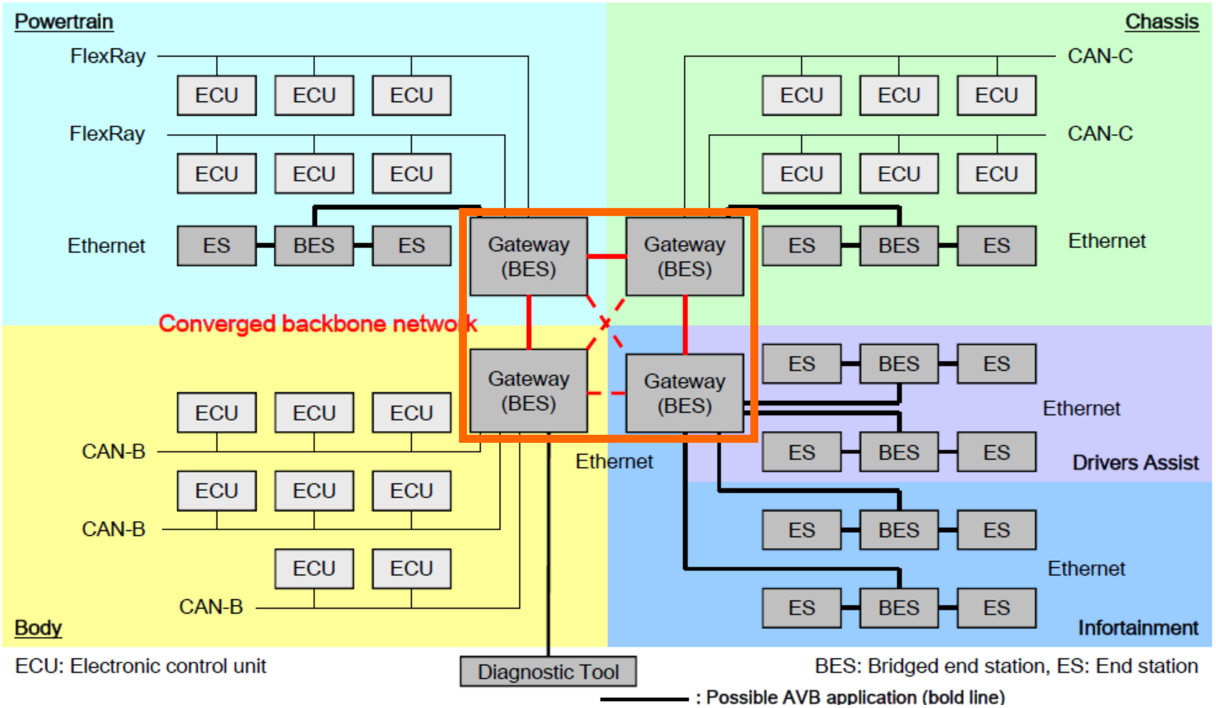
Transmission of PDUs within a 1722-Frame
- No frame encapsulation of CAN/FlexRay/... frames
- Time stamping is required depending on the application
- Unique identifier is required for identifying the content of the payload or other serialization

How do these use cases translate into actual possible network topologies in a vehicular network?

Automotive Network Topologies

Topology 1 – “Meshed” Star

“Toyota Topology*”



- Minimal redundant topology, could be expanded in the future (high flexibility)
- Small mesh topology
- Only parts of the topology are fault-tolerant

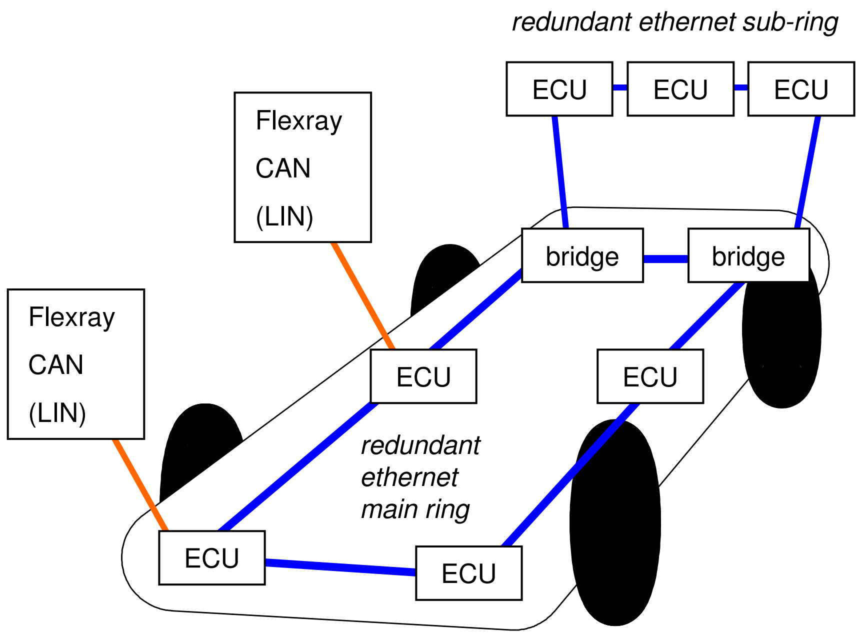
- Mesh keeps total hop count low, but introduces strong requirements on redundancy control protocol in fault-tolerant segments when combined with demanding timing requirements

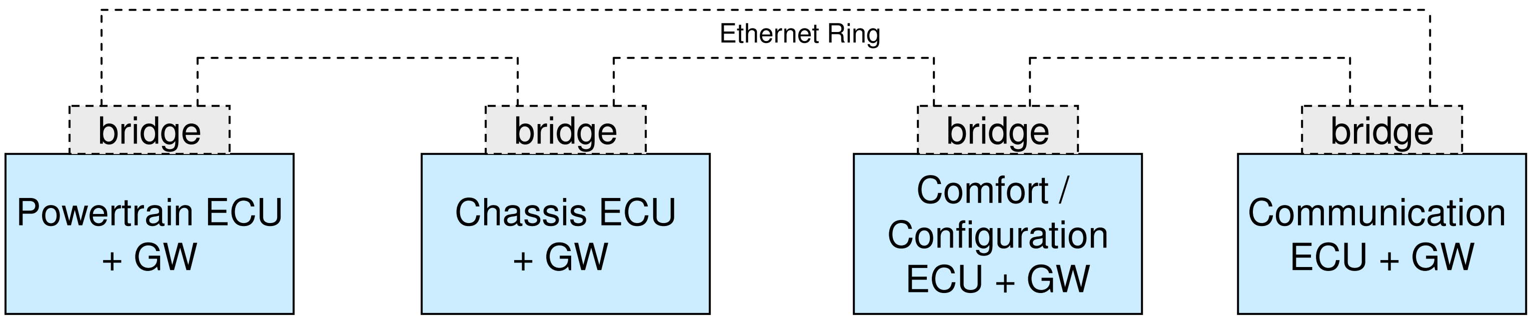
* Taken from http://www.ieee802.org/1/files/public/docs2011/new-avb-KimNakamura-automotive-network-requirements-0311.pdf

Automotive Network Topologies

Topology 2: Backbone ring with gateways and subrings

- ECUs with integrated bridges and gateways to other technologies
- Bridges with redundant sub-rings connecting ECUs
- Ethernet ring structure is fully fault-tolerant
- In a ring structure, demanding timing requirements can be met more easily by the redundancy protocol but…
- The ring introduces a higher hop count

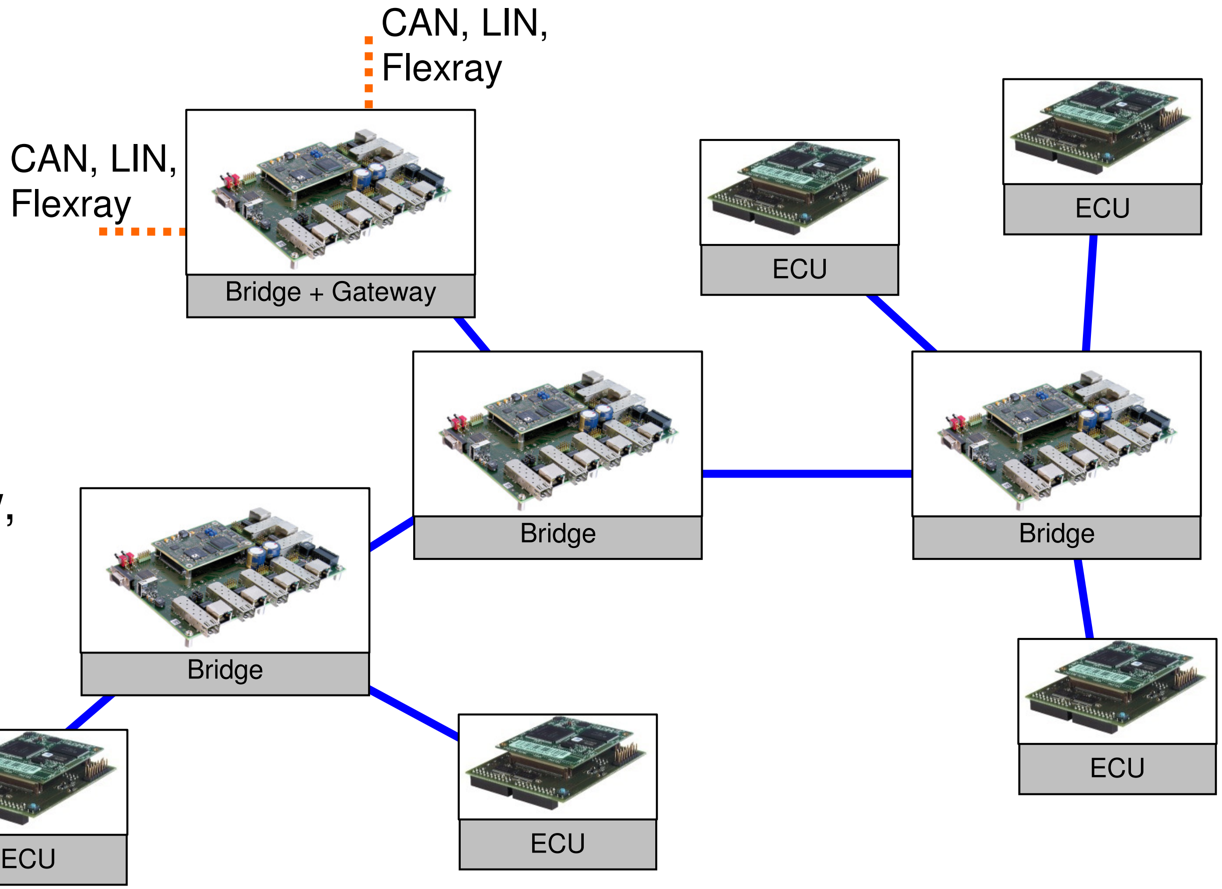




Automotive Network Topologies

Topology 3: Non-redundant extended star

- ECUs with integrated bridges and gateways to other technologies
- No redundant paths and therefore no fault-tolerance, but…
- No delay through redundancy control protocol
- Extended star topology allows to keep hop count low, even with a large number of bridges



Summary

- Automotive networks have also additional requirements regarding power consumption, startup times, EMI
- First Ethernet Network will be introduced in vehicles in 2013
- Scope of first implementations of Ethernet will be limited
- Complexity of implementations will grow over time
- The future could look like this

The future of Ethernet within Continental Automotive divisions

		2010	2015	2020
Powertrain	Combustion	CAN, Flex Ray	CAN, Flex Ray	CAN, Ether-net/IP
	Hybrid & EV's	CAN	CAN, Ether-net/IP	CAN, Ether-net/IP
Interior	Infotainment	CAN, MOST 25/50	CAN, MOST 150, Ether-net/IP	CAN, Ether-net/IP
	Body	CAN, Flex Ray, Ether-net/IP	CAN, Flex Ray, Ether-net/IP	CAN, Ether-net/IP
Chassis&Safety	Safety	CAN	CAN, Ether-net/IP	CAN, Ether-net/IP
	Motion	CAN, Flex Ray	CAN, Flex Ray	CAN, Flex Ray, Ether-net/IP