

### YOUR AUTOMOTIVE PROTOCOL STACK.

A journey from specification to SOP.

Dr. Lars Völker 2020-12-03/04

### TECHNICA ENGINEERING YOUR AUTOMOTIVE PROTOCOL STACK. TABLE OF CONTENTS

- Protocol stack design.
- Requirements engineering.
- Testing and integration.
- Logging and recording.



## CHAPTER PROTOCOL STACK DESIGN.



- Designing a protocol stack is not difficult, if you follow some rules:
  - Reuse standards as much as possible because this leads to better understanding and quality.
  - Limit the number of protocols to lower complexity and raise quality, since your resources are limited.
  - Learn from other OEMs and don't reinvent the wheel. What has been implemented and tested for another OEMs has already a better quality.



	Diagnostics/ Flash Update	Control Communication	Network Management	Audio Video	Time Sync
Layer 5-7	DoIP ISO 13400	SOME/IP & FDN e.g. AUTOSAR	UDP-NM AUTOSAR	AVTP	gPTP
Layer 4	(e.g. TCF	IEEE	IEEE		
Layer 3		1722	802.1AS		
Layer 2	ayer 2 MAC Layer and VLANs IEEE 802.1Q		Credit-based S IEEE 802.1C	Time stamping	
	100BASE-TX	100BASE-T1	1000BASE-T	1 N	/ulti-Gig

**IEEE 802.3** 

Figure: Technica engineering basic protocol stack recommendation.

**IEEE 802.3** 

2020-12-03/04

Layer 1

**IEEE 802.3** 

IEEE 802.3ch



- Layer 1: 10BASE-T1S:
  - New 10 Mbit/s Ethernet standard.
  - Half duplex bus with special "weighted round robin"-like arbitration.
  - May allow to close gap on low end.
- IEEE Time Sensitive Networking (TSN) features, e.g.:
  - 802.1CB Frame Replication and Elimination for Reliability
  - 802.1Qbu Frame preemption
  - 802.1Qbv Enhancements for Scheduled Traffic
  - 802.1Qch Cyclic Queuing and Forwarding
  - 802.1Qci Per-Stream Filtering and Policing
- Optimal TSN feature set still under industry-wide discussion.

## **Stechnica** PROTOCOL STACK EXTENSIONS (2).

- Network Security
  - IEEE MACsec (802.1AE) can protect all unicast, multicast, and broadcast messages at line-speed.
  - SSL/TLS, IPsec, and VPN protocols for backend connectivity.
  - SecOC allows application layer protection for selected use cases.
- Access Control
  - Ethernet Access Control (802.1X).
  - SOME/IP policy enforcement.
- Filtering
  - Packet filters (mainly on layer 2, 3, 4).
- Intrusion Detection (IDS)
  - Scalable, distributed solution based on protocols build in.



- Why is SOME/IP the most used middleware for automotive?
  - SOME/IP creates the abstraction needed for automotive applications.
  - SOME/IP scales from very small embedded devices to high performance ECUs.
  - SOME/IP Service Discovery allows "controlled flexibility" with user control.
  - SOME/IP supported by AUTOSAR (Classic and Adaptive).
  - SOME/IP is license free.
  - SOME/IP serialization is very efficient and fast.
  - SOME/IP is more than just serialization and service discovery.
  - SOME/IP was purpose-designed for automotive use cases.

## technica INSIGHT: SOME/IP SERIALIZATION.

- The faster your communication gets, the more resources serialization needs. On embedded systems your resources are very limited.
- You want a middleware with high efficiency and high performance.
- Text-based formats are typically very slow in serialization and deserialization due to string operations.
- Self-descriptive formats are slow due to copy operations.
- SOME/IP allows you to build the most efficient highperformance system!
  - Format optimized for low resources and high speed.



Fastest due to

zero-copy!





Figure: SOME/IP-SD timing example.

- SOME/IP-SD defines three phases:
  - Initial wait phase: allows to system to get ready.
  - Repetition phase: used for fast synchronization.
  - Main phase: stabilizes the system.
- The choice of the timings is very critical:
  - Determines startup performance.
  - Defines the behavior, when errors occur.
  - A thorough analysis is required, and experience helps!
- Starting point:
- RepetitionBaseDelay=30 RepetitionsMax=3 MainCycle=1s TTL=3s



- Transporting legacy CAN messages.
  - How to transport and how to gateway?
- Insights of our "Flexible Digital Network" work:
  - Avoid complex gatewaying to keep performance up and latency down.
  - Use standard transport for CAN/FlexRay.
  - CAN-FD to Ethernet and back to another CAN-FD: below 2ms latency.



# CHAPTER REQUIREMENTS ENGINEERING.



- Typical issues with specifications:
  - Specification is hard to understand. Often the goal is to specify and not to explain. But Tier-1 needs to understand.
  - Specifications are too large and too many.
     Implementers cannot remember 1000+ spec pages.
     Reference standards instead of creating your own.
  - AUTOSAR configuration changes behavior a lot. Hard to understand the behavior OEMs want.

#### • Better:

- Include explanations in specification.
- Reference as much as possible and stick to standards.
- Presentations for the specification and system design.

Туре	Text
Info	This specification determines how SOME/IP is used as the in-vehicle Middleware.
Req.	The ECU shall support SOME/IP based on [1].
	Type Info Req.



# CHAPTER TESTING AND INTEGRATION.



- Many OEMs are not testing the stacks enough.
- Since this is "new" technology and quite complex:
  - Stack testing is hard.
  - Stack testing requires experts.
- Common mistake: leave Tier-1 alone with stack testing:
  - If Tier-1 misunderstands requirements, Tier-1 cannot find the issue.
  - Recommendation: use 3<sup>rd</sup> party or 3<sup>rd</sup> party tools for testing.
- Common problem: OEM specifics are not tested by standard tools.





- Many aspects of the communication stack are distributed.
- Problem: You cannot find all problems in just testing a single ECU. So you find them, when you put together the vehicle for testing.
- Solution: Front-loading of integration of Ethernet network.
  - For better quality you want to integrate all Ethernet ECUs as early as possible.
  - You need to record the data.
  - You need to test the integration.
  - You need to analyze results.
- Benefit: Ethernet cluster is stable faster. Overall quality much better!





# CHAPTER LOGGING AND RECORDING.

## **Stechnica** COMMON PROBLEMS IN LOGGING.

- Ethernet logging is more difficult than CAN logging.
  - Data is only present on the links needed  $\rightarrow$  you need to log all links!
  - The amount of data can be quite high.
- The logging setup must no interfere with timing sensitive protocols (e.g. IEEE 802.1AS / PTP).
- Even CAN-FD does not like changes to the topology!
  - In many vehicles it is impossible to record all CAN-FDs with a logger only.
- Solution:
  - Look for equipment designed for Ethernet.
  - Split data acquisition probes off the data logger.

## **Stechnica** DLT, XCP, ETC. IN VEHICLES.

- Problem:
  - Packet injection for DLT or XCP needed for logger to talk directly to ECU.
  - Packet injection can interfere with timings on link (changing the system you are measuring) and this is not acceptable for IEEE 802.1AS and others.
- Solution:
  - Ethernet packet injection without changing the link timings when injecting is needed.

## **Stechnica** LINKS AND TIMESTAMPS MATTER.

- Problem: you need to analyze lost or misrouted data, but the recorded logs do not have the required information to find the problem.
- Solution:
  - Data needs to be recorded on every link and marked with link and direction as well.
  - Data needs to be timestamped right at the link.



- Problem: In vehicles with a logging setup<sup>(1)</sup>, packets seemed to be lost and the communication was not stable.
- Analysis: It turned out that links were not stable and short link-down/link-up cycles happened from time to time. Analysis was very hard and the root cause was even harder to find.
- Solution:
  - Record link quality.
  - Record link-up / link-down.
  - Do automated analysis on this data.
  - Quality control you test car fleet.

(1) Equipment of another vendor, so we cannot share details.

2020-12-03/04



## **Stechnica** LOGGING – STATE OF THE ART (2).

- Technically Enhanced Capture Module Protocol (TECMP)
  - Free and open state-of-the-art protocol.
  - Support data and meta data exchange between capture modules and logger.
  - Support Ethernet, CAN, CAN-FD, FlexRay, LIN, RS-232, Serial, Analog, and others.
  - Fully supported by Wireshark 3.4. Out now!

🦲 🔳 🔝	💌 📙 🛅 🗙 🖾 🔍 🗢 =	* 😫 🕆 🕹 📃 🗏			
Apply a	display filter <ctrl-></ctrl->				
rime	Source	Destination	Timestamp Info	Counter Payload	TX ( Frame 55510: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface \Device\N
35.768	05 Technica_00	IPv4mcast_00	Jun 2, 2020 13:53:56.405902776 UTC TECMP Payload:	CAN Data 43660 98170198170122ff	Ethernet II, Src: Fechnica 00 (00:500:22:44:30:00), Dst: IPv4mcast_00 (01:00:50:00:00) transferration in the second and the
35.769	59 Technica_00	IPv4mcast_00	Jun 2, 2020 13:53:56.406068048 UTC TECMP Payload:	CAN Data 43661 fa00faaa7800d0fe	<ul> <li>Technically Enhanced Capture House Proceeds</li> </ul>
35.769	60 Technica_00	IPv4mcast_00	Jun 2, 2020 13:53:56.406160992 UTC TECMP Payload:	CAN Data 43662 507800507800ddfe	Capture Hodule 10: 0x0040 (CH-CAN-Combo 0 (Default))
35.769	60 Technica_00	IPv4mcast_00	Jun 2, 2020 13:53:56.406302256 UTC TECMP Payload:	CAN Data 43663 aa7800aa7800ddfe	Counter: 45067
35.769	60 Technica_00	IPv4mcast_00	Jun 2, 2020 13:53:56.406399192 UTC TECMP Payload:	CAN Data 43664 fa00faaa7800d0fe	Version: 2
35.769	97 Technica_00	IPv4mcast_00	Jun 2, 2020 13:53:56.406635400 UTC TECMP Payload:	CAN Data 43665 aa7800aa7800ddfe	message Type: Logging Stream (0x05)
35.771	15 Technica_00	IPv4mcast_00	Jun 2, 2020 13:53:56.408203048 UTC TECMP Payload:	LIN 10376 6392df00008900fe	Data Type: CAN Data (0x0002)
35.771	15 160.48.199.16	239.192.255.251	Jun 2, 2020 13:53:56.408273088 UTC 30490 → 30490 L	_en=904 1669	Reserved: 0X0000
35.771	17 160.48.199.16	239.192.255.251	Jun 2, 2020 13:53:56.408275248 UTC 30490 → 30490 L	Len=904 1670	CM Flags: 0x000+, End of Segment, Start of Segment, Spy, Multi Frame
35.771	18 160.48.199.16	239.192.255.251	Jun 2, 2020 13:53:56.408273088 UTC 30490 → 30490 L	.en=904 1671	I = tha of Segment: True
35.771	18 160.48.199.16	239.192.255.251	Jun 2, 2020 13:53:56.408275168 UTC 30490 → 30490 L	en=904 1672	
35.773	43 Technica_00	IPv4mcast_00	Jun 2, 2020 13:53:56.410195248 UTC TECMP Payload:	Flexray Data 43666 000000000000000000000000000000000	0
35.776	10 Technica 00	IPv4mcast 00	Jun 2, 2020 13:53:56.413145128 UTC TECMP Payload:	LIN 10377 ffff0f040000000	1 = Multi Frame: True
35.776	11 Technica 00	IPv4mcast 00	Jun 2, 2020 13:53:56.413198688 UTC TECMP Pavload:	LIN 10378 ff0100040000000	0 = Capture Module Overflow: False
35.777	60 Technica 00	IPv4mcast 00	Jun 2, 2020 13:53:56.414313648 UTC TECMP Pavload:	LIN 10379	Technically Enhanced Capture Module Protocol Payload (CAN Data)
35,778	45 Technica 00	IPv4mcast 00	Jun 2, 2020 13:53:56,415182648 UTC TECMP Pavload:	CAN Data 43667 fffffffffffffff	Channel ID: 0x00000164
35,778	46 Technica 00	IPv4mcast 00	Jun 2, 2020 13:53:56,415432848 UTC TECMP Pavload:	CAN Data 43668 fffffffffffffff	<ul> <li>Timestamp: Jun 2, 2020 13:53:56.415182648 UTC (synchronized or master)</li> </ul>
35.780	86 Visteon 4c:6b:2f	LLDP Multicast	Jun 2, 2020 13:53:56,416925728 UTC Sync Message	1673	0 = Timestamp Synchronisation Status: Synchronized or Master
35,781	73 Visteon 4c:6b:2f	LLDP Multicast	Jun 2, 2020 13:53:56.417678768 UTC Follow Up Messa	age 1674	Length: 13
35,783	46 Technica 00	IPv4mcast 00	Jun 2, 2020 13:53:56,420195448 UTC TECMP Pavload:	Flexray Data 43669 000000000000000000000000	0
35,786	78 160.48.199.16	239, 192, 255, 251	Jun 2, 2020 13:53:56,423016768 UTC 30490 → 30490 L	en=72 1675	
35,786	78 160.48.199.16	239.192.255.251	Jun 2, 2020 13:53:56.423016768 UTC 30490 → 30490 L	en=72 1676	
35,786	78 160.48.199.16	239,192,255,251	Jun 2, 2020 13:53:56.423018928 UTC 30490 → 30490 L	en=72 1677	
35.786	79 160.48.199.16	239, 192, 255, 251	Jun 2, 2020 13:53:56.423018928 UTC 30490 → 30490 L	en=72 1678	0 = Frame Error: False
35.786	79 Technica 00	TPv4mcast 00	Jun 2, 2020 13:53:56.423145768 UTC TECMP Pavload:	LTN 10380 ££££0£040000000	0 = CRC Error: False
35.786	79 Technica 00	TPv4mcast 00	Jun 2, 2020 13:53:56.423200088 UTC TECMP Payload:	10381 ff0100040000000	.0 = TX (sent by Capture Module): False
35.787	21 Technica 00	IPv4mcast 00	Jun 2, 2020 13:53:56.424310408 UTC TECMP Pavload:	LTN 10382	0 = Overflow (lost data): False
35.787	22 Technica 00	IPv4mcast 00	Jun 2, 2020 13:53:56.424379488 UTC TECMP Payload:	LIN 10383	✓ ID: 0×00000153
35.787	84 Technica 00	TPv4mcast 00	Jun 2, 2020 13:53:56.424980928 UTC TECMP Pavload:	CAN Data 43670 db8c600fffffffff	0 = ID Type: 11bit CAN Identifier
35,787	91 Technica 00	TPv4mcast 00	Jun 2, 2020 13:53:56.425094688 UTC TECMP Payload:	LTN 10384 e0ff	
35.787	92 Technica 00	IPv4mcast 00	Jun 2, 2020 13:53:56.425131256 UTC TECMP Payload:	CAN Data 43671 d09700d0970022ff	Payload Length: 8
35.787	92 Technica 00	TPv4mcast 00	Jun 2, 2020 13:53:56.425217128 UTC TECMP Payload:	CAN Data 43672 590045b901ffffff	Payload: fffffffffffff
35.787	92 Technica 00	IPv4mcast 00	Jun 2, 2020 13:53:56.425195568 UTC TECMP Payload	Elexrav Data 43673 00000000000000000000000000000000000	8
35.787	92 Technica 00	TPv4mcast 00	Jun 2, 2020 13:53:56.425371472 UTC TECMP Payload:	CAN Data 43674 db8c600fffffffff	
35 787	93 Technica 00	TPv4mcast_00	Jun 2 2020 13:53:56 425435992 UTC TECMP Payload:	CAN Data 43675 d09700d0970022ff	
35 788	37 Technica 00	TPv4mcast_00	Jun 2 2020 13:53:56 425607672 UTC TECHP Payload:	CAN Data 43676 9817019817012255	
35 700	37 Technica 00	IPv4mcast_00	Jun 2 2020 13:53:56 425687096 UTC TECHP Payload.	CAN Data 43677 db8c600ffffffffff	

#### NikkeiBP Automotive Ethernet Tech Days 2020 | Technica Engineering GmbH



- Protocol stack design:
  - Reuse standards and be focused.
- Requirements engineering:
  - Make Tier-1s/2s understand.
- Testing and integration:
  - Let a 3<sup>rd</sup> party test and integrate early.
- Logging and recording:
  - Go for state-of-the-art solutions with all key features.
- Overall these recommendation raise the quality of your product.



Dr. Lars Völker Technical Fellow Lars.Voelker@technica-engineering.de

Technica Engineering GmbH Leopoldstraße 236 80807 Munich, Germany

#### **Product distribution** in Japan:

Tel: +81-422-26-8211 Email: sales@gailogic.co.jp please visit the website: https://www.gailogic.co.jp/ae/te

#### Engineering partner in Japan: TOYO Corporation

Quest for Precision Tel: +81-3-3245-1250 Email: automotive-ict@toyo.co.jp