

© Bertrand

## Performancebetrachtung innerhalb einer serviceorientierten E/E-Architektur

Die serviceorientierte Kommunikation über Ethernet beschäftigt derzeit die Branche. Ansätze, mit denen die Performance dieser Kommunikationsstrukturen analysiert werden kann, zeigt Bertrand mit seinem Projekt „Park and Charge“. Ziel ist, die Perspektiven und Grenzen einer serviceorientierten Kommunikation zu erkennen, um eine sinnvolle und effiziente Nutzung im Verbund mit herkömmlichen automotiven Kommunikationstechnologien sicherzustellen.

### ANFORDERUNG AN ZUKÜNFTIGE E/E-ARCHITEKTUREN

Der drastische technologische Wandel in der Mobilität wirkt sich in einer äquivalenten Umstellung auf die E/E-Architektur der Fahrzeuge aus. Der Einzug

von Automotive Ethernet und somit von Netzwerkarchitekturen eröffnet weitere Möglichkeiten. Ein aktueller Trend zeigt sich in der Einführung von E/E-Architekturen mit hochperformanten Domänencontrollern, die einerseits untereinander in einem leistungsstarken Back-

### AUTOREN



**Dipl.-Ing. Cornelius Butzkamm**  
ist Teamleiter Architektur und Vernetzung bei Bertrand in Ingolstadt.



**Konstantin Brand, M. Sc.**  
ist Spezialist für Automotive Ethernet und SOA bei Bertrand in Ingolstadt.

bone kommunizieren und andererseits ihre Domäne mit Subbussen vereinen. Durch die technologischen Neuerungen werden vermehrt serviceorientierte Kommunikationskonzepte auf ethernet-basierten Netzwerkstrukturen eingesetzt. Ein Protokoll, das serviceorientierte Kommunikation gewährt, ist das SOME/IP-Protokoll (Scalable Oriented MiddlewarE over IP). Dabei handelt es sich um ein Remote Procedure Call (RPC) Protocol, das nach den Bedürfnissen der Automobilindustrie entwickelt wurde [1].

Aufgrund der neuen Kommunikationsstrukturen mit deutlichen Unterschieden zu den herkömmlichen botschaftsorientierten Kommunikationskonzepten (CAN, LIN, FlexRay) rücken Fragestellungen wie Machbarkeit und Leistungsfähigkeit bei Fahrzeug- und Funktionsarchitekten in den Vordergrund, **BILD 1**.

### PROJEKT PARK AND CHARGE

Bertrandt entwickelt im Zuge des internen Projekts Park and Charge Lösungen für künftige technische Fragestellungen der automobilen Welt. Neben technischen Lösungen sollen auch offene Fragestellungen wie die beschriebene Leistungsfähigkeit von serviceorientierten

Architekturen, aber auch niederlassungsübergreifendes Arbeiten, Agilität und Identifikation von Kernkompetenzzellen beantwortet werden.

Die technischen Lösungen werden auf mehrere Träger übertragen. Um die technischen Schwerpunkte rund um die Themen autonomes Fahren, Connectivity und Elektromobilität realisieren zu können, wurde auf den Technikträgern eine neue E/E-Architektur mit fünf Domänencontrollern implementiert, bei der auf Interdomänenebene mittels SOME/IP kommuniziert wird.

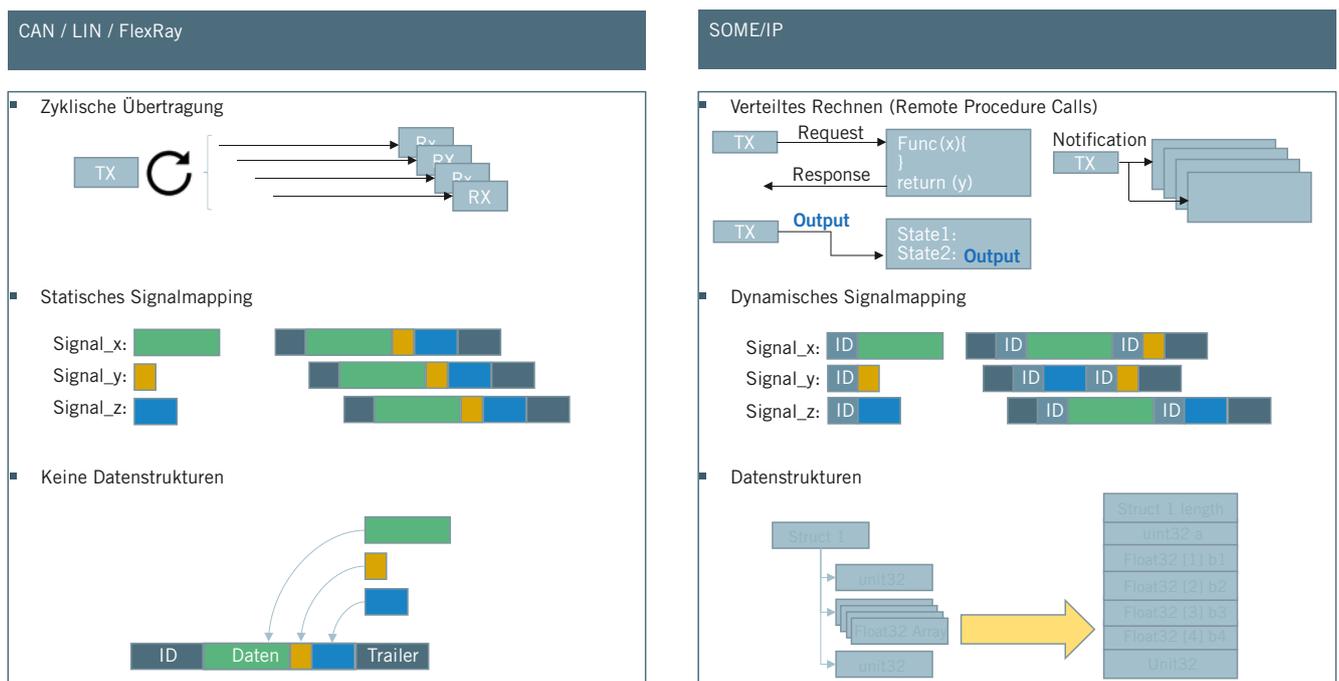
Der Autonomous Drive Domain Controller (ADC) vereint in seiner Subdomäne vor allem Sensorik zur Umfelderkennung (etwa Lidar und Kamera). In der Subdomäne des Electric Vehicle Domain Controllers (EDC) befinden sich Aktuatorik/Regelkreise zum Bewegen eines Elektrofahrzeugs (X-by-Wire-Anwendungen). Unterhalb des HMI Domain Controllers (HDC) wurde ein HMI-Konzept für autonom fahrende Fahrzeuge umgesetzt. Der Connectivity Domain Controller (CDC) ist für die Kommunikation zwischen Fahrzeug und Backend mit weiteren Frontend-Anwendungen verantwortlich. In der Body-Domain-Controller-Subdomäne (BDC) wurden Sitzplatzbelegungen und Lichtfunktionen umgesetzt, **BILD 2**.

### SERVICEORIENTIERTE KOMMUNIKATION

Um eine neuartige Systemarchitektur aufsetzen und Fragen bezüglich Timing-Verhalten, Latenzzeiten und Routingeigenschaften von einem Protokoll zum anderen beantworten zu können, ist es notwendig, die Kommunikationseigenschaften jedes einzelnen Protokolls genauestens zu verstehen. Dabei wurde schrittweise im Zuge des Projekts die Komplexität der oben dargestellten Architektur erhöht, Anforderungen erweitert sowie Parameterstellungen nachjustiert.

Für das initiale Aufsetzen einer serviceorientierten Systemarchitektur wurde zunächst eine Restbussimulation mit SOME/IP erstellt. Mit dieser Simulation konnte festgelegt werden, welche Signale und Funktionen definiert werden müssen, um damit die Anforderungen der Funktionen des Fahrzeugs zu bedienen. Insgesamt wurden neun Services und 46 Fields definiert, die alle grundlegenden Funktionen im Fahrzeug, zum Beispiel Lenken, Bremsen und Beschleunigen, realisieren.

Nach erfolgreichem Test mithilfe der Simulation wurde ein Technikträger, zunächst mit drei Domänenrechnern



**BILD 1** Unterschiede zwischen den herkömmlichen Kommunikationsprotokollen und SOME/IP (© Bertrandt)

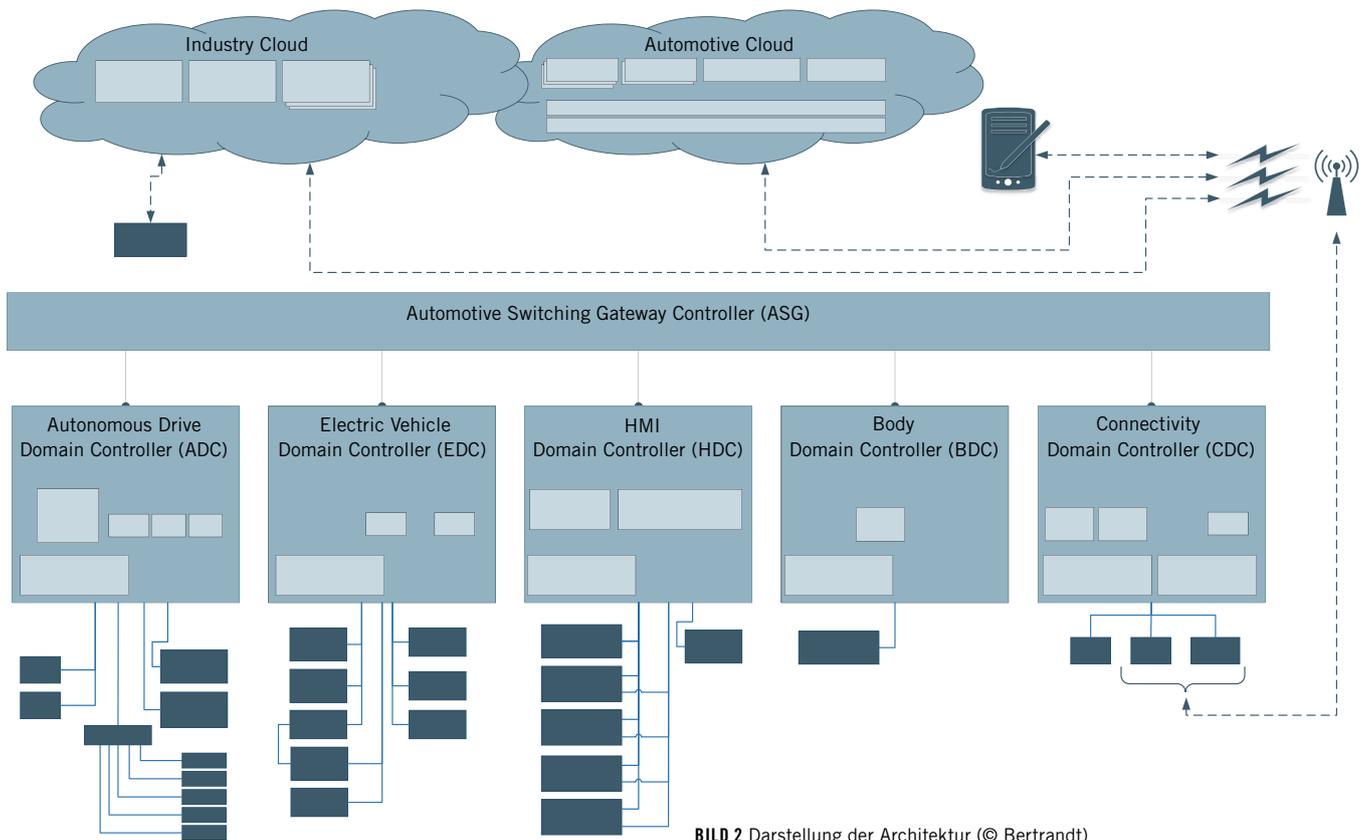


BILD 2 Darstellung der Architektur (© Bertrandt)

(ADC, EDC und HDC), aufgesetzt. Die Domänenrechner wurden anfangs mit einer herkömmlichen Ethernet-Verbindung (100 Base-TX) und einem handelsüblichen Switch verbunden. Die Konfiguration der Domänenrechner und die Umsetzung der SOME/IP-Kommunikation fanden mittels der erstellten SOME/IP-Restbussimulation statt. Der Technikträger als Grundlage für künftige Analysen ist voll funktionsfähig und wird ständig erweitert.

Mit einer funktionierenden SOME/IP-Kommunikation in dem Technikträger stellt sich die Frage, wie zuverlässig eine Ethernet-Verbindung über SOME/IP unter realen Bedingungen ist. Des Weiteren standen Fragestellungen bezüglich der Realisierbarkeit von komplexen Anwendungen wie Fahrerassistenzfunktionen oder X-by-Wire-Funktionen zur Diskussion.

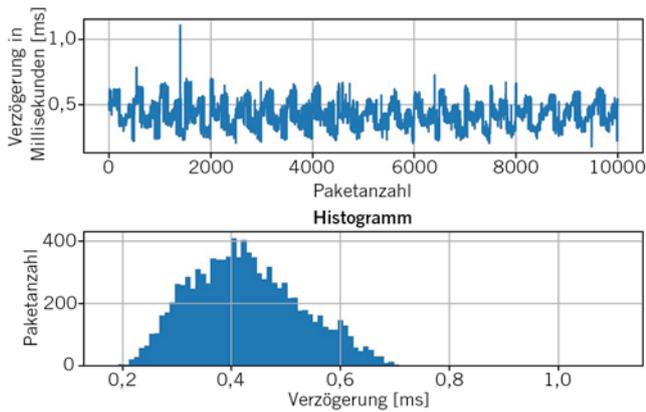
Um diese Fragen zu beantworten, wurde eine Reihe von Messungen durchgeführt. Für die Ermittlung der Latenzzeit wurde die Zeit gemessen, die zwei Domänenrechner brauchen, um eine Nutzinformation (Payload) mit dem SOME/IP-Protokoll zu versenden und zu

empfangen. Dafür wurde untersucht, welche Zeit benötigt wird, um ein SOME/IP-Paket zu erstellen (Encapsulation), es zu übertragen und wieder zu empfangen, bis die Nutzinformation wieder zur Verfügung steht (Decapsulation) [2]. Anschließend galt es, die Grenzen der Kommunikation zu identifizieren und deren Auswirkung zu analysieren. Für eine genaue Latenzzeit-Messung waren synchrone Zeitstempel notwendig, die zu einem bestimmten Zeitpunkt im System erstellt werden. Mit ihnen lassen sich einzelne Abschnitte im Übertragungsweg untersuchen. Um ein aufwendiges Zeitsynchronisationsverfahren zu vermeiden, wurden auf einem Domänenrechner zwei Netzwerkkarten integriert, auf denen jeweils eine SOME/IP-Instanz arbeitet. Somit ist der Domänenrechner Sender und Empfänger zugleich und hat die Besonderheit, dass alle generierten Zeitstempel vergleichbar sind. Der Informationsaustausch erfolgt über das reale Ethernet-Kabel und den Switch.

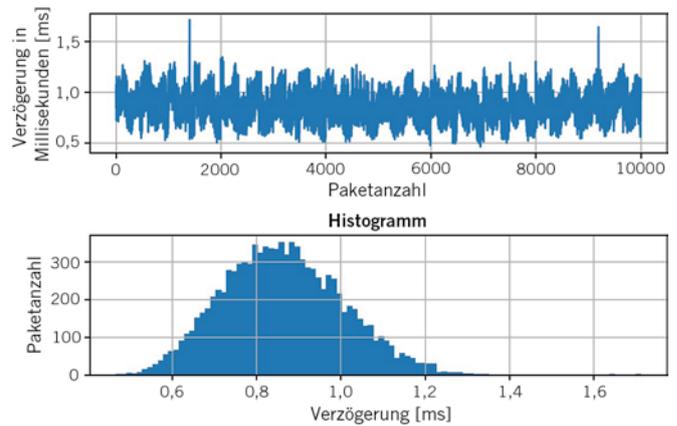
Um Schwankungen (Jitter) zu erfassen, wurde eine zyklische SOME/IP-Kommunikation erstellt, in der eine

Field-Notification einmal pro 1 ms gesendet wird – in einer weiteren Messung einmal pro 10 ms. Die Nutzinformation enthält den aktuellen Zeitstempel vor dem Erstellen des SOME/IP-Pakets. Nach dem Empfangen der Nachricht wird nochmals ein Zeitstempel generiert, sobald die Nutzinformation weiterverarbeitet werden kann. Zudem wurden die Zeitstempel der Netzwerkkarte aufgenommen und den Paketen zugeordnet. Somit konnte ermittelt werden, wie lange die Erstellung eines SOME/IP-Pakets, dessen Übertragungszeit und das Entpacken des empfangenen Pakets dauern. Zusätzlich konnte die Gesamtübertragungszeit berechnet werden. Die Messungen ergaben, dass die Erstellung eines SOME/IP-Pakets im Schnitt 0,44 ms (mindestens 0,18 ms, maximal 1,13 ms) benötigt, BILD 3.

Die Realisierung der Domänenrechner fand anwendungsabhängig mit verschiedenen Systemen statt. Für einen kompletten Übertragungsablauf, der das Erstellen des Pakets, das Übertragen und das Auslesen beinhaltet, wurden durchschnittliche Zeiten von 0,9 ms



**BILD 3** Das Ergebnis der Messungen im SOME/IP-Paket (© Bertrandt)



**BILD 4** Messungen mit dem Domänenrechner (© Bertrandt)

(mindestens 0,47 ms, maximal 1,7 ms) erfasst, **BILD 4**. Bei den hier dargestellten Ergebnissen ist allerdings zu beachten, dass Sender und Empfänger unter idealen Bedingungen kommunizieren.

Beim Hinzufügen eines weiteren Domänenrechners, der ebenfalls mit der Empfängerseite kommuniziert, wei-

chen die oben genannten Werte des kompletten Übertragungsablaufs ab, da die Anzahl der ankommenden SOME/IP-Pakete von zwei Faktoren abhängt – einerseits von der Intensität der Kommunikation von einem Domänenrechner zum anderen und andererseits von der Anzahl der Domänenrechner, die

mit ihm kommunizieren wollen. Senden viele verschiedene Domänenrechner gleichzeitig an einen Empfänger, kann es zu einer Überlastung kommen (sogenanntes Paket Bunching) [3]. Im schlimmsten Fall können Empfangspuffer überlaufen und Informationen verloren gehen.

## IDENTIFIZIERTE SYSTEMGRENZEN

Um weiter einzugrenzen, inwiefern Probleme in der Kommunikation auftreten können, wurde bei weiteren Analysen in zwei Arten unterschieden. Zum einen wurden viele kleine Pakete mit einer hohen Wiederholungszahl (60 Byte/Frame mit 30.000 Ethernet-Frames pro s) betrachtet, die die Grenze der Übertragungslast ausreizen. Zum anderen wurden Pakete mit einer maximalen Größe (1500 Byte/Frame mit 7000 Ethernet-Frames pro s) berücksichtigt, die die Grenze der Übertragungskapazität ausreizen.

Die Auswertungen bestätigen, dass die Kommunikation mit großen Paketen einen geringeren Einfluss auf die Latenzzeit hat als eine Kommunikation mit vielen kleinen Paketen. Das liegt daran, dass ein SOME/IP-Paket mit beispielsweise 1-Byte-Nutzdaten zur Weiterverarbeitung die kompletten 7 OSI-Schichten des Kommunikationsstacks durchlaufen muss. Kommt dieser Vorgang häufig vor, ist das Verhältnis von Nutzlast zu Overhead zu groß. Da Ethernet-Pakete normalerweise sequenziell abgearbeitet werden, kann es dazu kommen, dass bei Verzögerungen Empfangspuffer überlaufen und Signale verloren gehen. Bei großen Paketen müssen deutlich weniger Pakete pro s versendet und verarbeitet werden, damit die 100-Mit/s-Grenze nicht überschritten wird.

## ABLEITUNGEN UND AUSWERTUNGEN

Somit erschließt sich, dass sich eine Kommunikation über SOME/IP lohnt, wenn auch die mögliche Nutzlast des Pakets ausreichend genutzt wird. Aus fachlicher Sicht erscheint eine Payload von über 70 Byte sinnvoll, da der SOME/IP Overhead 59 Byte beträgt. Es empfiehlt sich, viele kleine 1-Byte-Informationen in Form von Datenstrukturen zu gruppieren.

Abschließend lässt sich festhalten, dass sich eine Anforderung von 1 ms Latenzzeit mit SOME/IP im gegebenen System nicht realisieren lässt, da die Menge an zu verarbeitenden Daten nicht deterministisch ist. Allerdings haben weitere Messungen im gegebenen System gezeigt, dass eine zyklische Kommunikation von 10 ms erfolgreich umgesetzt werden kann.

## AUSBLICK

Anhand des Projekts Park and Charge hat Bertrandt Lösungen für aktuelle Themenstellungen der Automobilindustrie erarbeitet. So wurde ein autonom fahrender Technikträger mit einer serviceorientierten Kommunikationsarchitektur realisiert, die mit einer Cloud kommuniziert und mit neuartigen Bedienkonzepten innerhalb der HMI ausgestattet ist. Neben Lösungsansätzen können auch aktuellste Fragestellungen zu verschiedenen Fachthemen, zum Beispiel die Performance-Analyse von serviceorientierten E/E-Architekturen, beantwortet werden.

Zum heutigen Stand sind in dem Technikträger fünf Domänenrechner verbaut. Die Verbindung der Interdomänenebene wird über BroadR-Reach (100 Base-T1) und einen Automotive-Switch realisiert. Zudem sind weitere Funktionen umgesetzt, die mit 25 Services und insgesamt 140 Fields, Events und Methoden bedient werden können.

Die nächsten anstehenden Analysen betreffen den Verbund der Subdomänen mit dem Domänenrechner. Das Ziel ist, End-to-End-Latenzzeiten messen zu können, um Routingzeiten von Ethernet zu beispielsweise CAN zu erfassen. Der Fokus wird dabei sein, CAN-Zykluszeiten mit einer serviceorientierten Architektur zu kombinieren. Ein Thema, das dabei angegangen wird, ist der Vergleich der Protokoll-Latenzzeiten, der bei der zyklischen Kommunikation vom CAN-Protokoll zu einer Asynchronisation mit SOME/IP führt.

## LITERATURHINWEISE

- [1] Autosar SOME/IP protocol specification. Online: <http://www.some-ip.com/papers.shtml>, aufgerufen am 04.02.2018
- [2] Matheus, K.; Königseder, T.: Automotive Ethernet. Cambridge University Press, 2017
- [3] Metcalfe, B. et al.: Automotive Ethernet – the Definitive Guide. Intrepid Control Systems, 2014



DIESER BEITRAG IST IM E-MAGAZIN  
VERFÜGBAR UNTER:  
[www.emag.springerprofessional.de/atz](http://www.emag.springerprofessional.de/atz)