

## KOMMUNIKATIONSMANAGEMENT

# Implementierung einer AUTOSAR DDS-Netzwerkanbindung

Die AUTOSAR DDS-Netzwerkanbindung ermöglicht es, dass AUTOSAR Adaptive-Anwendungen mit DDS-Systemen zusammenarbeiten. Doch was ist die DDS-Netzwerkanbindung genau und welche Herausforderungen gibt es hier beim Mapping, der Speicherverwaltung und der Compliance?

Für Unternehmen im AUTOSAR-Umfeld sind zwei verschiedene Plattformen entstanden: AUTOSAR Classic und AUTOSAR Adaptive sowie der Foundation-Standard, der die Interoperabilität zwischen den AUTOSAR-Plattformen ermöglicht. Die Beteiligung des Software-Framework-Anbieters RTI an AUTOSAR Adaptive geht mittlerweile weit über die Veröffentlichung von Standards hinaus und umfasst sowohl die Zusammenarbeit mit AUTOSAR Software-Anbietern als auch die interne Entwicklung einer funktionsfähigen Implementierung des Funktionsclusters für das Kommunikationsmanagement, genannt ara::com.

Die Spezifikation, Verwaltung und Weiterentwicklung der DDS-Netzwerkanbindung (Data Distribution Service) ermöglicht es, dass AUTOSAR Adaptive-

Anwendungen mit bestehenden und zukünftigen DDS-Systemen zusammenarbeiten können.

### Was genau ist die DDS-Netzwerkanbindung?

Innerhalb der AUTOSAR Adaptive-Plattform bietet das Funktionscluster für das Kommunikationsmanagement eine serviceorientierte Kommunikationsmodellierung und Infrastruktur. APIs auf Applikationsebene sind protokollunabhängig und basieren daher auf einer Middleware, die zwischen den APIs und den tatsächlich zugrunde liegenden Kommunikationstechnologien vermittelt (Bild 1). Solche Mappings werden im AUTOSAR-Sprachgebrauch als Netzwerkanbindungen bezeichnet und nur drei von ihnen sind standardisiert, nämlich DDS, SO-

ME/IP und die -signalbasierte Kommunikation.

### Herausforderungen

Das Entwerfen und Implementieren von Komponenten zur Netzwerkanbindung für AUTOSAR ist aufgrund diverser Design-Aspekte der AUTOSAR Adaptive Plattform generell eine komplexe Aufgabe: Es gibt keine standardisierte API-Schicht zwischen anwendungsbezogenen APIs und Komponenten zur Netzwerkanbindung, sodass bei der Implementierung spezieller Netzwerkanbindungen jede Integration etwas anders angegangen werden muss.

AUTOSAR verfügt über ein eigenes Typsystem, das orthogonale UML- und C++ – Programmiersprachen wiedergibt. Diese müssen von den Netzwerk-

anbindungen mit dem Typsystem und den Funktionen jeder zugrundeliegenden Kommunikationstechnologie, z. B. DDS-eigenen erweiterbaren Typen, abgeglichen werden.

Und AUTOSAR schreibt auch sehr spezifische Protokolle für den Objektlebenszyklus und die Speicherverwaltung vor, die wiederum mit den zugrunde liegenden Frameworks in der gekapselten Netzwerkanbindung abgeglichen werden müssen.

### Mapping von DDS und dem AUTOSAR-Typsystem

Auch wenn die AUTOSAR-Typen relativ weit verbreitet sind und naturgemäß auf DDS-XTypes (Booleans, Numerics, Strings, Strukturen, Arrays, Sequenzen usw.) abgebildet werden, erfordert die eigentliche Integration die Zusammenführung der plattformspezifischen Module (PSM) für beide Typsysteme in den Programmiersprachen C und C++.

Im Folgenden geht es um die einzige von AUTOSAR unterstützte Sprachanbindung C++ (bis hin zu C++14),

Und dann nach der Übersetzung von AUTOSARs ARXML nach C++:

```
struct Point3D{
float x;
float y;
float z;
};
```

Das Ergebnis fällt identisch aus. Ist die Typkompatibilität von AUTOSAR/DDS also bereits eine klare Sache? Entwickelt man jedoch den Typenkatalog noch etwas weiter und gruppiert 3D Points zu Point-Clouds, die der Einfachheit halber als reine Arrays mit fester Länge angeordnet sind, würde das gemäß DDS C PSM so aussehen:

```
typedef Point3D PointCloud[1024]
```

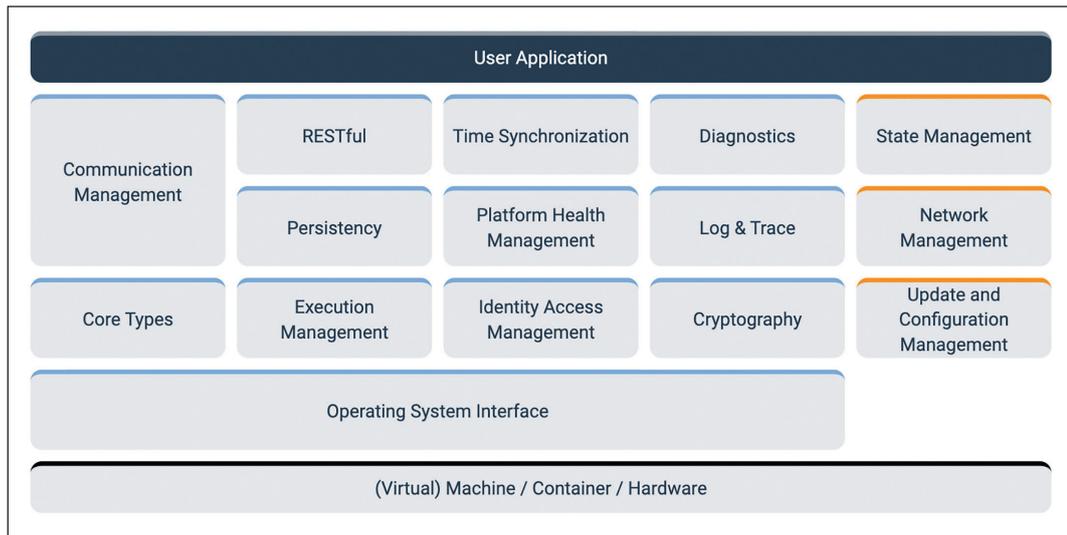
Der gleiche Typ (ein Point Array mit 1024 Elementen) würde in AUTOSAR Adaptive wie folgt erstellt werden:

```
typedef ara:: core::Array<Point3D, 1024> PointCloud;
```

mentierungstypen für strukturierte, benutzerdefinierte Typen über die AUTOSAR-Modellierung keine Wertefelder, sondern funktional kompatible Wrapper, die auf einen intern verwalteten „Backend-Speicher“ verweisen. Ein solcher Backend-Speicher könnte auf dem PSM der Adaptive Plattform (numerische C++-Standardtypen und ara::core-Containertypen) oder auf den automatisch generierten PSM-Typen von DDS aus DDS-IDL oder DDS-XML basieren. Dieser Ansatz vereinfacht nicht nur die Integration mit Kommunikationsframeworks verschiedenster Art, sondern hält auch weitere Möglichkeiten für eine effiziente Speicherverwaltung und die Reduzierung und Eliminierung von Sample-Kopien beim Austausch von Daten offen.

Mit der beschriebenen Flexibilität der Datentyp-Struktur ergeben sich zwei Speicherverwaltungs-Strategien für Data Samples, die sich gegenseitig nicht ausschließen:

- Instanzen des Implementierungstyps können direkt von der Anwendung gestapelt oder dem Heap



**Bild 1: Die AUTOSAR Adaptive-Plattform definiert eine Reihe von Funktionsclustern, die vom Ausführungsmanagement über die Zeitsynchronisation, Überwachung und Diagnose bis hin zum Kommunikationsmanagement reichen.**  
© AUTOSAR

sowie die C-Sprache PSM von DDS (letztere aus Gründen, die noch deutlich werden). Ein einfaches Beispiel zeigt, wie ein simpler Point3D-Typ nach der Übersetzung von DDS-IDL oder DDS-XML nach C aussehen könnte:

```
struct Point3D{
float x;
float y;
float z;
};
```

Hier wurden also ähnliche, aber nicht identische Typen erstellt. Tatsächlich erhält ara::core::Array<> einige Semantiken der einfachen Array-Typen mit fester Länge (z. B. den indizierten Zugriff), fügt aber auch STL-ähnliche Semantiken wie begin() und end() Methoden hinzu und konvertiert nicht vorbehaltlos in einen Point3D Pointer. Diese Diskrepanz vergrößert sich bei anderen Typen, wie z. B. Strings und Vektoren, noch mehr. In ara::com von RTI enthalten Imple-

zugewiesen werden. Dies veranlasst die interne Typ-Implementierung, die In-Memory-PSM-Typen von AUTOSAR zu verwenden. Bei der Übergabe an das zugrundeliegende Framework müssen diese Datentypen entsprechend dem gewünschten Zieldatenformat konvertiert und/oder serialisiert werden

- Bei der Suche nach Zero-Copy-Datentransfers können Applikationen auch die Standardmethode Alloca-

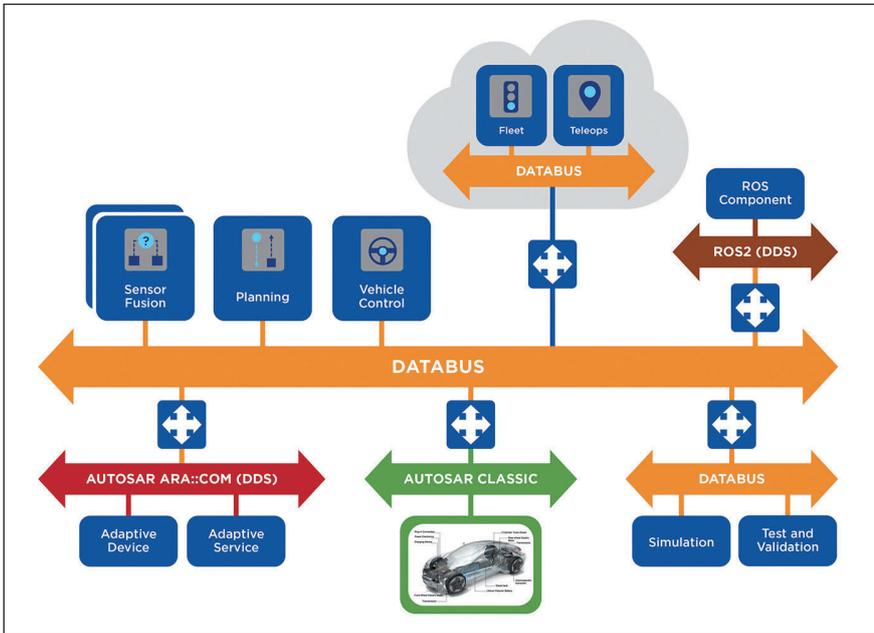


Bild 2: Der DDS-Datenbus teilt ein gemeinsames Datenmodell mit anderen Systemen und Standards einschließlich AUTOSAR Classic und AUTOSAR Adaptive. © RTI

te() nutzen, die von Events und Field Notifiers bereitgestellt wird. In diesem Fall fragt die Netzwerkanbindung das zugrundeliegende Framework ein intern zugewiesenes Datenmuster „auszuleihen“. Es wird „zurückgegeben“, wenn es von der Applikation ausgefüllt und zurückgesendet wurde.

Letztere Strategie stellt einen Mechanismus dar, der größenunabhängige und zeitlich konstante Datentransfers in verschiedenen Szenarien ermöglicht.

### Auf dem Weg zur Compliance

Die Bewertung einer DDS-Implementierung als SEoC (Safety Element out of Context) in einem beliebigen ISO-26262 ASIL-Level stellt eine eigene Herausforderung dar, die RTI mit dem Connex Drive Produkt bereits adressiert. Angesichts der Vorgehensweise und des finanziellen Aufwands bei den Bewertungen der funktionalen Sicherheit kann es aus strategischer Sicht notwendig sein, die Funktionalität gegen die Größe der

Codebasis und des Footprints einzutauschen. Solche Einschränkungen fließen in RTIs ara::com-Design-entscheidungen und in zukünftige AUTOSAR-Standards ein:

- C ist die einzige DDS-Sprache, die PSM bei der Implementierung von DDS-Netzwerkanbindungen erlaubt. Das ermöglicht eine einfache Anpassung für DDS-Implementierungen verschiedener Varianten.
- Die Unterstützung des Multi-Bindings ist erforderlich, da mehr als eine DDS-Netzwerkanbindung zur Verfügung stehen muss, um die erwähnte Vielfalt an DDS-Implementierungen zu unterstützen.
- Ein schlankere Ansatz für die Service-Erkennung und die Anbindungsfunktionen in der Spezifikation der AUTOSAR Adaptive DDS-Netzwerkanbindung besteht darin, allgemeine DDS-Funktionen wie Topics und Instanzen zu nutzen, die in den meisten DDS-Middleware-Implementierungen leichter zu finden sind.

■ (oe)

[www.rti.com](http://www.rti.com)

**Emilio Guijarro** ist Automotive Applications Engineer, **Peter Schmuckermaier** Senior Account Manager, beide bei Real-Time Innovations (RTI).

# SAVE THE DATE

**HANSER**  
Tagungen

**safetronic.** 2021

Ganzheitliche Sicherheit für Straßenfahrzeuge

16. - 17. November 2021

[www.hanser-tagungen.de/safetronic](http://www.hanser-tagungen.de/safetronic)

**SAFETY** meets **SECURITY**

Nächste Generation kollaborativer und hochautomatisierter Systeme

18. November 2021

[www.hanser-tagungen.de/security](http://www.hanser-tagungen.de/security)