TrAF-Cloud

Truck Architecture for Functionality in the Cloud

Public report
Diarienr 2018-05010

Författare: Niklas Mellegård
Datum: 2022-08-31
Projekt inom Elektronik, mjukvara och kommunikation - FFI

FFI Fordonsstrategisk Forskning och Innovation
VINNOVA Energimyndigheten TRAFIKVERKET
FKG VOLVO SCANIA VOLVO

Innehållsförteckning

1 Sammanfattning på svenska

Vårt transportsystem är mitt i en förändring att bli mer hållbar, effektiv och tillgänglig. Framsteg inom elektifiering och automation är möjliggörare för en fundamental transformation. För att dra full nytta av sådana framsteg krävs uppkoppling där fordon, infrastruktur och mobilitetsanvändare (person- såväl som godstransport) kan kommunicera och dynamiskt anpassa sig efter rådande omständigheter—såväl planera för mer effektiva transporter som att kontinuerligt bevaka och anpassa planerna vid oförutsedda händelser.

För kommersiella fordon, uppkoppling är en möjliggörare för tätare integration i de olika logistikkedjor de är del av, men möjliggör också avancerad funktionalitet för säkrare och effektivare transporter genom maskinellt samarbete—t.ex, C-ITS-funktionalitet.

Denna typ av funktionalitet kräver oftast att fordonen stöder den nya teknologin, och därför kan uppdateras i takt med att innovationer distribueras på marknaden. Även om innovationer rullas ut redan idag är dessa ofta begränsade till nya fordonsmodeller vilket resulterar i långsamt anammande av innovationer. Fordon förväntas idag alltmer stödja möjligheten att uppdatera sin ombordmjukvara för att fordonet kontinuerligt skall kunna förbättras, men är fortfrande begränsat. Utöver detta kommer fordon behöva kunna utnyttja molnfunktionlitet (inklusive Fog och Edge). Drivet av sakernas internet (internet-of-things, IoT) kan en molnplatform avlasta den begränsade datorplatformen ombord på fordonen, samt ge tillgång till stora datamängder som kan användas till att ytterligare förbättra funktionalitet för säkerhet, hållbarhet och effektivitet.

För att åstadkomma detta krävs en tätare integration mellan systemarkitekturerna i ombordsystemen och molnet. En tätare integration förväntas ge:

i) Snabbare utrullning av ny funktionlitet;
ii) Möjlighet att samla och dela stora datamängder;
iii) Tillgång till beräkningskraft som inte finns tillgängligt ombord;
iv) Möjliggöra sofistikerad funktionalitet som spänner over flottor av fordon;
v) Tillgång till mycket skalbara resurser relaterade till datalagring, -bearbetning och –analys genom nyttjande av molnteknologier.

För att realisera en tätare integration krävs dock fundamentala förändringar i hur dagens el- och mjukvaruarkitektur är utformad ombord på fordonen. Därför var målen med projektet TrAF-Cloud att undersöka arkitekturella principer för att kommersiella fordon ska kunna integreras med molntjänster på ett effektivt sätt. Projektets huvudsakliga forskningsfråga var:

Hur kan man systematiskt nyttja beräknings- och lagringskapacitet tillgängligt i molnet som ett komplement till ombordsystemen?

Projektet adresserade huvudfrågan genom följande mer detaljerade forskningsfrågor:

1. Hur kan man designa en gemensam fordonsarkitektur som spänner över både fordonet och en molnplatform?
2. Hur kan man sömlöst allokera funktioner att exekvera i molnet eller ombord på fordonet?
3. Hur kan man säkerställa transparens i exekveringen av funktionen så att denna inte behöver designas specifkt för exekveringsplatformen?
4. Vilka är de arktiekturella beroendena mella ombord- och molnplatformen?
5. Vilka arkitekturella principer kan användas för att nyttja en naturligt molnintegrerad fordonsplaform som en öppen innovationsplatform?

Genom att adressera dessa frågor hade projektet följande mål:

G1. Projektet skall skapa och demonstrera en referensplatform som integrerar ombord- och molnfunktionalitet;

G2. Kraven på beräkningskraft ombord kommer att minska för vissa typer av funktioner med bibehållen kvalitet;

G3. Ledtider för integration kommer att minska avsevärt;

G4. Projektet skall demonstrera och evaluera genomförbarhet av ett affärsorienterat ekosystem för transportrelaterade tjänster.

TrAF-Cloud-projektet har tagit en bred ansats genom att:

- Undersöka ekosystem som möjliggörs av en uppkopplad fordonsplatform (leverablerna R4.1 och R4.2 som bidrar till mål G4);
- Identifierat centrala teknologier som krävs för att göra en sådan platform möjlig (leverabel R4.1 som bidrar till mål G4);
- Identifierat viktiga användarfall och icke-funktionella krav för en sådan platformsarkitektur, samt specifik analyserat krav kring datasäkerhet (leverablerna R1.2 och R1.3, dokumenterade som del av leverabel R2.1, och som bidrar till målen G1-G3);
- Utforskat mikrotjänstdesign i ett automotive-kontext, samt strategier för migrering från dagens monolitiska design (leverablerna R3.1a, R3.1b, R3.1c and R3.1d, även summerade i R2.1, som bidrar till målen G1-G3);
- Utvecklat och demonstrerat tekniska principer som krävs för sömlöst uppkopplade fordon som inbegriper state-of-the-art från både inbyggda fordonssystem och molnmiljöer (leverablerna D5.1-D5.3, dokumenterade i R2.1, och som bidrar till mål G1)

Projektet resulterade i (för detaljer, se kap. 7 nedan):

- Tre publika rapporter
- Två projektinterna rapporter
- Åtta akademiska publikationer (varav fyra accepterade vid denna rapports författande)
- Två akademiska uppsatser, en licentiatavhandling och en Masteruppsats
- Tre demonstratorer
- 16 öppna och halvöppna (öppen inbjudan bland projektets parter) seminarier och presentationer

Huvudfokus i projektet TrAF-Cloud var att undersöka en gemensam ombord- och molnarkitektur för att möjliggöra mer sofistikerad funktionalitet som inte begränsas av den tillgängliga beräknings- och lagringskapaciteten ombord, samt som möjliggör ett bredare och mer systematiskt samarbete mellan olka fordon. Projektets resultat innefattar beskrivna och demonstrerade tekniska lösningar för att åstadkomma en tätar integration mellan fordon och molnmiljöer. Bland resultaten finns även en inventering av liknande (i allmänhet nystartade) branchinitiativ som kan utgöra en grund för standardiserade lösningar. Som fortsatt arbete rekommenderas att undersöka hur sådana initiativ kan användas för att att nå TrAF-Clouds uppsatta mål.

Vidare, projekt som MAUS[1] har demonstrerat hur system-av-system (SoS) som en lösningsansats kan förbättra effektiviteten på systemnivå, men också att det kräver att deltagande system (smo individuella fordon) behöver kunna anpassa sina egenskaper efter rådande omständigheter—något som varit med bland de tekniska delmålen i TrAF-Cloud. Det rekommenderas som fortsatt arbete att undersöka möjligheter och krav på en systemnivå för att etablera transport-SoS—dvs, att närmare undersöka hur deltagande system i ett transport-SoS behöver kommunicera för att koordinera sig, och därmed också kartlägga kraven på de deltagande systemen.

---

[1] *https://www.vinnova.se/p/monster-for-arkitektur-och-affarsmodeller-for-urban-mobilitet-system-av-system-maus/*

2 Summary in English

While current on-board and off-board architectures for commercial vehicles are largely disconnected with different design principles, as well as development and deployment cadence, future vehicle platforms will be expected to constitute open innovation platforms where vehicle manufacturers as well as third party actors can develop innovative services to make transports more sustainable, efficient and safe.

To accomplish this, there is a need to achieve better integration between on-board and off-board services. This includes seamlessly utilizing cloud infrastructure as a complement to the on-board computational platform, as well as accessing on-board data and functionality from a cloud environment. The challenges that needs investigation are related to both technology and business aspects. Thus, the objective of the TrAF-Cloud project was to investigate business and technological enablers to establish a platform eco-system around natively connected commercial vehicles.

The main results from the project include:

- Investigation of business related aspects for establishing a platform eco-system around commercial vehicles that includes third-party developers
- Investigation and description of architectural principles for seamless deployment of services to on- and off-board platforms
- Demonstrators of key technologies to achieve seamless integration, as well as demonstrating their viability by integrating with existing vehicle systems.
- Investigation (reported in several academic publications) of micro-service architectures as well as migration strategies from monolithic designs (commonly used in the automotive domain).

In addition, one licentiate thesis and one master thesis were written as part of the project, and 16 open and semi-open seminars were organized for dissemination purposes.

3 Background

Our transport system is changing to become more sustainable, efficient and accessible. New technological advances in fields such as electrification and automation provide opportunities for fundamental transformation. Such advances depend on a connected transport system where vehicles, infrastructure, and mobility users communicate to adapt their behaviour seamlessly. Specifically, machine-to-machine communication enables to optimize on strategic, tactical, and operational levels—i.e., to plan for more efficient transports as well as to monitor and quickly respond to unforeseen changes in those plans.

From the perspective of commercial vehicles, connectedness allows not only for better integration with the range of different logistic chains in which they operate, but also for safer and more efficient operations—for instance through C-ITS[2] applications enabled by V2X[3]-communication.

While such technological advances are already being deployed, these are often only with new vehicle models—thus, the uptake is slow. In addition, vehicles will be expected to systematically utilize cloud, fog and edge capabilities as these—driven by trends such as internet-of-things (IoT)—become increasingly ubiquitous and capable. To leverage such innovations as they become available, vehicle need the ability to integrate tighter with off-board capabilities. This in turn requires new architectural principles for the on-board systems.

To this end, the TrAF-Cloud project set out to investigate aspects of a natively connected vehicle architecture. Such an architecture can be seen as a natural step along an already established trajectory, as outlined in what follows.

3.1 The TrAF-Cloud Project in Context

The on-board vehicle architecture has a natural emphasis on mechanics. The vehicle was from the outset a purely mechanical machine where the main concerns were about ensuring loadbearing, vehicle dynamics, and efficient transmission of force from the engine to the wheels. The evolution of the architecture perfected such aspects, but also came to focus on new and increasingly important concerns of the stakeholders, such as minimizing component wear and easy maintenance (increased up-time), and driver comfort.

Electronic controllers were initially introduced to replace or enhance mechanical parts where those could deliver better performance; for example, to more accurately controlling the mixture of fuel and air in the carburettor to increase engine efficiency. Initially, this had little effect on the vehicle architecture as the electronic units only enhanced or replaced existing mechanical parts. However, the ability to quickly transmit information (in the form of electrical signals) enabled various parts of the vehicle to be integrated without physical proximity—a fundamental difference from mechanical parts. This fact impacted the previously strictly mechanical architecture not only logically—as there now was interaction between parts of the vehicle that were not necessarily physically connected—but also physically, as wires needed to be run through the chassis affecting its loadbearing properties.

Eventually, the complexity of the vehicle electronic system warranted its own architectural view, although still with strong interdependency with the mechanical architecture. There was therefore a need to rethink how the complete vehicle architecture was developed, described and documented. The main concerns, however, were still centred on optimizing the performance of the on-board systems.

The introduction of software followed a similar path as electronics did. Initially, software components replaced logic that was previously implemented with electronics as they could realize more sophisticated behaviour, and could additionally be updated without requiring

---

[2] *Cooperative Intelligent Transport Systems and Services (e.g. https://www.car-2-car.org/about-c-its/)*
[3] *Shorthand for 'vehicle-to-vehicle' and 'vehicle-to-infrastructure' communication*

physical changes to the vehicle. Furthermore, the integration of functionality distributed around the vehicle could be realized using serial communication busses, considerably reducing the wiring required—affecting space, weight, and cost. Like the evolution of electronics, as the amount of functionality realized by software grew, eventually it too warranted its own architectural view; still tightly integrated with the electronic and mechanical dimensions.

Today, software enables functionality that would have been considered science fiction not that long ago—already in 2007 it was estimated that over 80% of innovation in the automotive industry can be linked to software[4]. Still, the current on-board architecture is centred on optimizing the performance of the vehicle itself.

Going forward, it is clear that vehicles will need to better integrate—and optimize their performance together—with the transportation system (e.g. logistic chains) in which they operate. This in turn will require the vehicle to better integrate with off-board systems using ICT[5]. While such integration has been emerging for some time, the approach has typically been to shield the on-board software systems by using a proxy component (a telematics unit) that presents available vehicle capabilities to the off-board systems—an approach with similarities to how both electronics and software was previously introduced into the vehicle architecture.

In that respect, the telematics unit acts as an abstraction layer that hides details of the on-board systems from the off-board systems that accesses them—i.e. encapsulation as a means to localize change and the impact of such change. This kind of abstraction layer is required as there are fundamental differences in how the on- and off-board systems are designed, but also has the drawback of concentrating the complexity of managing the variability of vehicle configurations to the telematics unit. This is a limiting factor as to what capabilities the vehicle can expose to an off-board platform, but especially to the speed at which new capabilities can be made available.

Today, vehicle functionality is made available to an off-board platform in a relatively flexible way—both in terms of accessing vehicle data and to trigger certain kinds of vehicle functionality. An off-board ecosystem of fleet management applications has emerged over many years and is today sophisticated in terms of its support for the fleet owner to manage their vehicles.

Still, there are considerable bottlenecks that hamper the vehicles' ability to become a fully integrated component in the transportation system. Two examples that have been used as guiding scenarios in the project are here given for illustration.

3.2 Dealing with Future Needs

Predictive or preventive maintenance is a highly valuable service for a fleet owner as it has the potential to improve vehicle up-time. Such a service allows not only for the fleet owner to schedule service before component malfunction, but also to plan replacement while the component may still have a second-hand value and thereby reducing maintenance costs and at the same time providing an opportunity for circular business models. Such a service requires the vehicle to collect and report data to an off-board service for analysis.

Currently, this is already possible but to a limited extent. The telematics unit is equipped with the capability to deliver a pre-defined set of data in a pre-defined set of formats and/or resolutions triggered by a pre-defined set of events. Similarly, the telematics unit also has the capability to activate certain on-board functionalities on request. The telematics unit thus acts as an abstraction layer, presenting the off-board platform with a consistent interface and encapsulates variability in the on-board platform.

However, if the data need is outside of the pre-defined scope—either when new types of data are required, with a different resolution, or triggered by new kinds of events—it is not straight-

---

[4] *Broy M., Kruger I. H., Pretschner A., and Salzmann C., "Engineering Automotive Software," Proceedings of the IEEE, vol. 95, no. 2, pp. 356–373, 2007, doi: 10.1109/JPROC.2006.888386.*

[5] *Information and Communication Technology*

forward to realize and often not within the short timeframe required to meet market demand; especially if that demand has a limited customer base. This becomes a limiting factor for the predictive maintenance service, as it is often not until components begin to show wear that it becomes known precisely what data (including format and resolution) that needs monitoring—this situation also applies to troubleshooting unexpected behaviour or malfunctions in the field.

This example illustrates that there is a need to examine how the integration between the connectivity sub-system (i.e. the telematics unit) and the rest of the on-board vehicle software systems can be made more flexible to allow quick adaptation to new requirements from the off-board systems. Furthermore, it illustrates the need to share—on- and off-board—a common view of available vehicle capabilities as the variability between individual vehicles is large.

Similar needs can be seen for other services. For instance:

- For a fleet manager to plan routes that includes charging for electrical vehicles may require new access to on-board capabilities, but also to collect and report data to plan optimal locations for charging infrastructure;
- A more flexible integration between the vehicle and add-ons or connected trailers would allow a fleet owner to utilize such capabilities in a more uniform way with the vehicle as the service provider;
- New C-ITS solutions are emerging with the potential to improve efficiency, safety and sustainability, but may also pose new requirements on the vehicle to be compliant.

In such cases, it is not known at design time what vehicle capabilities will need integration with an off-board platform. Rather the vehicle must be able to evolve with the new external requirements. While this is to some extent already possible today, the effort required by the vehicle manufacturer adds friction that risk making uptake of new transportation system innovations slow.

3.3 Utilizing Cloud Capabilities

Unit cost is a strong driver in the automotive industry with a long and sensible tradition. Dimensioning components based on their required performance enables the OEM to choose the component that matches the need while keeping the total material cost at a minimum. This makes perfect sense provided that the vehicle, once manufactured, remains static. However, the flexibility of software allows to continuously evolve vehicle capabilities at a high pace, as it does not require physical changes to—or even contact with—the vehicle. This, however, is in direct conflict with the focus on unit cost, as it would require on-board computer systems to be specified with capacity above what the software known at design time required. Furthermore, it is challenging to reliably predict what kind of functionality will be deployed in the future and therefore also to predict what will be the required specification of the computational units.

Advancements in edge, fog and cloud capabilities—primarily availability and reliability (e.g. with 5G)—opens up new possibilities with deployment of functionality off-board circumventing the need to oversize on-board computers to cater for unknown future needs. Furthermore, an on-board software architecture that incorporates scalable high-performance off-board components has the possibility to allow for functionality that would not be feasible to deployed on-board at all—for instance, control algorithms to optimize driveline parameters taking large-scale off-board data into consideration (such as current traffic situation) or collaborative training and utilization of deep-learning algorithms.

There are already today examples where on-board systems realize more sophisticated behaviour than what would be possible without utilizing off-board capabilities. One such example is where the driveline software requests information about road topology (extracted from map data acquired from an off-board service) to calculate an optimal speed profile for more optimize fuel consumption. However, such services are still rare and typically designed on a case-by-case basis without systematic architectural support.

4 Objectives, Research Questions, and Methodology

As described, the next generation vehicle will need a tighter integration of the on-board and off-board platforms. The objectives include both to improve the efficiency of the vehicle systems themselves, and to be able to optimize the traffic and transport systems through cooperative functionality (e.g., C-ITS and collaborative last-mile transports).

For this, a common architectural approach to vehicle systems that includes both the on- and off-board platforms is required. The benefits include[6]:

i) Faster deployment and integration of new functionality;

ii) Possibility of collecting and sharing data on a large scale;

iii) Access to computational resources not available on-board vehicles;

iv) Enabling more sophisticated functionality that spans multiple vehicles;

v) Access highly scalable data storage, processing, and analytics capabilities by cloud computing.

To realize such closer integration, however, fundamental changes to the current vehicle electronic and software architecture are required. The TrAF-Cloud[7,8] project therefore aimed to provide architectural basis to enable commercial vehicles to integrate cloud services efficiently and effectively, including future third-party services.

The TrAF-Cloud project is based on the main question:

> *“How to systematically utilize the computational and storage capacity available in the cloud as complement to on-board functionality?”*

The sub-questions are:

1. How can we **design a secure and safe common truck on-board and off-board architecture** that enables seamless allocation and execution of functionality in the cloud?
2. How can we **securely and safely allocate functions** to execute either in the cloud or in the truck?
3. How can we ensure **transparency during the execution** of functions?
4. What are the **architectural dependencies between on-board and off-board** architecture?
5. What **principles** can be used to **leverage cloud-enabled** truck architectures into **open service platforms**?

---

[6] *From the TrAF-Cloud FFI project application*

[7] *FFI project no. 2018-05010 (https://www.vinnova.se/p/lastbilsarkitektur-for-funktionalitet-i-molnet-traf-cloud/)*

[8] *https://trafcloud-project.se*

The approach taken in the TrAF-Cloud project was formalized in six work-packages (WP), as illustrated in the figure below.

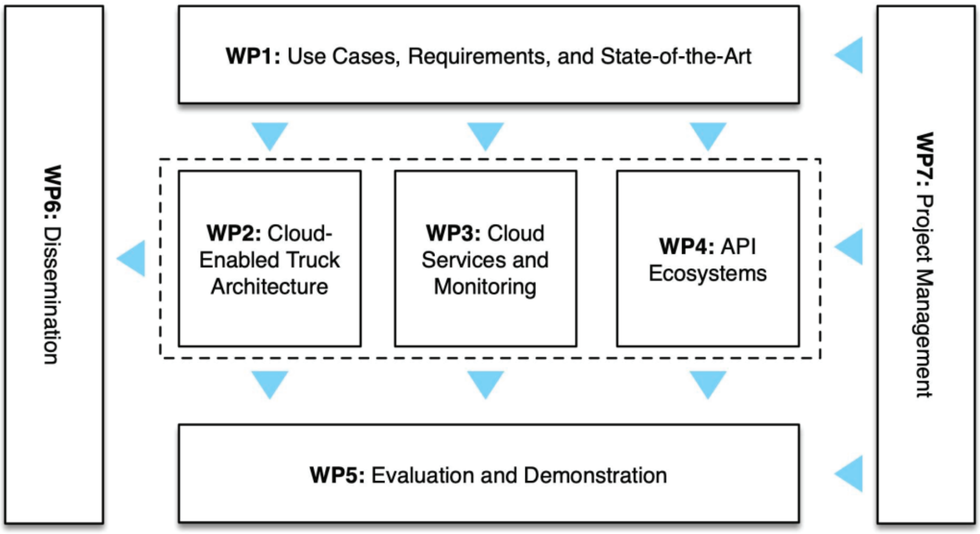


Figure 1. TrAF-Cloud work-package logic

The **objective of WP1** was to establish scientific context and current state-of-the-art as well as to formally define requirements and guiding use-cases for the other WPs.

The **deliverables planned for WP1** were:

- **R1.1**, State of the Art (report);
- **R1.2**, Use Case definition (report);
- **R1.3**, Architecture requirements (report).

The **objective of WP2** was to design and describe a novel cloud-native vehicle reference architecture.

The **deliverables planned for WP2** were:

- **R2.1**, Architecture description & architecture design principles, (report);

The **objective of WP3** was to explore and describe techniques, approaches, and algorithms for transparently integrate cloud services into truck architectures.

The planned **deliverables from WP3** were:

- **R3.1**, Monitoring and decision procedures, report;
- **R3.2**, Cloud Service Description Language specification, report

The **objective of WP4** was to research API ecosystems, devising methods to develop cloud-enabled trucks into platforms suitable for third-party providers. The objectives also included to develop understanding of governance mechanisms and generativity that the required platform boundary resources can create for the actors in such a digital eco-system.

The planned **deliverables from WP4** were:

- **R4.1**, Ecosystem Mapping (Current Partners and relationships) and Boundary Resources (APIs, SDK or any other types of resources), report;
- **R4.2**, Design principles for future platforms and ecosystems - governance and strategy, report.

The **objective of WP5** was to develop proof-of-concepts to demonstrate key concepts from the other work-packages.

The planned **deliverables from WP5** were:

- **R5.1**, Established virtual platform for evaluation, report;
- **D5.2**, Demonstration platform for PoC , demonstrator 1;
- **D5.3**, Demonstration platform for PoC , demonstrator 2;

- **D5.4**, Innovation challenge based on the developed platform;
- **D5.5**, Demonstration platform for PoC, final demonstrator.

The **objective of WP6** was concerned with collaboration and dissemination activities.

The planned **deliverables from WP6** were:

- **D6.1**, Project Website;
- **R6.2**, Combined plan and tracking report (planning of dissemination and exploitation and tracking of realized dissemination and exploitation activities), report;
- **R6.3**, Evaluation result, report.

Finally, **WP7** was dedicated to project management and reporting according to contract.

5 Goals

The goals as stated in the TrAF-Cloud project proposal were:

G1. The project will create and demonstrate a reference platform that integrates on-board and off-board functionality.

G2. The utilisation of in-vehicle computation power will decrease for a certain class of applications (to be identified in the project) while maintaining adequate responsiveness with unchanged or increased quality of service.

G3. The integration lead times will decrease with several orders of magnitude for certain functionality.

G4. The project will demonstrate and evaluate the viability of a business-oriented ecosystem for transport solution services.

To reach those goals, the following were the **main expected results** as stated in the proposal:

Result 1: Report on a common **reference architecture**, including associated requirements and design principles, that enables fast and flexible integration of both on-board and off-board (cloud) functionality with the truck.

Result 2: Demonstration and **proof-of-concept implementations**/simulations of the benefits based on selected use cases.

Additional expected results were stated as:

Result 3: Report on **State-of-the-Art** regarding cloud functionality for a truck for which parts of its functionality are allocated to the cloud

Result 4: Use **case definition** detailing the example system and scenarios that the project will focus on

Result 5: **Report** on current and future actors' requirements for providing services in the **transportation industry ecosystem** such as platform boundary resources (API's, SDKs, guidelines, etc.)

Result 6: Specification of a **cloud service definition language** that can be applied to all services regardless of the allocation to an in-vehicle ECU or cloud

Result 7: **Licentiate thesis** on the subject of integration of cloud services with an on-board embedded vehicle platform

Result 8: At least **5 scientific publications**

Result 9: **Innovation challenge** using the demonstration platform

The goals remained unchanged through the project although the approach, direction, and specific results (deliverables) were adapted. These deviations from the original plan are described in the next section.

6 Results

Project results, deviations from plan, as well as contributions to expected results and goals are reported below.

6.1 Deliverables

The following deliverables were produced in the project:

WP1—Use-cases, Requirements and State-of-the-art

- **R1.1**, *State of the Art*: A public report[9] summarizing relevant technologies for cloud and vehicle systems.
- **R1.2**, *Use Case definition (iteration 1)*: A public report[10] describing use-cases to guide the design of the TrAF-Cloud reference architecture, as well as relevant technologies and challenges. The published report is the first draft of the report, the final report was included as part of R2.1 (see 'Deviations from plan', below);

WP2—Cloud-enabled Truck Architecture

- **R2.1**, *Architecture description & architecture design principles*: A confidential technical report of describing a reference architecture for a cloud-enabled truck architecture. The 100+ page report includes background information justifying the design of the reference architecture (fulfilling the objectives of R1.2) and the high-level requirements that need to be taking into considerations (fulfilling the objective with R1.3). The report describes relevant design patterns and existing technologies, expanding the state-of-the-art report (R1.1). A detailed description of the reference architecture illustrated with UML-models are included, as well as documentation of the proof-of-concepts developed as part of WP5 to show applicability of the key concepts described in the reference architecture.

WP3—Cloud Services and Monitoring

For this work-package, there was a need to re-define its contents (for details, see 'Deviations from plan', below). For this work-package several publications were produced in three main topics:

- **R3.1** *Micro-service architectures and migrating monolithic architectures*. Five academic publications (three of which were published at the time of writing this report) and a licentiate thesis was produced as part of this deliverable. The publications and thesis are listed in section 7.2, below.
- **R3.2**, *Automatic architecture compliance evaluation*. Two academic publications (one submitted and one in progress), as well as one thesis on Master level were produced.
- **R3.3**, *Performance evaluation of serverless applications*. One academic publication was produced.

WP4—API Ecosystems

The deliverables produced by WP 4 were (also, see 'Deviations from plan', below):

- **R4.1**, *Ecosystem Mapping (Current Partners and relationships) and Boundary Resources (APIs, SDK or any other types of resources)*. The confidential report describes a design thinking approach for designing platform resources to enable an eco-system of contributors. The report outlines the methodology (a Platform Innovation Kit, a tailored set of business model canvases) as exemplifies its application through a series of workshops conducted with key stakeholders of the TrAF-Cloud partners.
- **R4.2**, Design principles for future platforms and ecosystems - governance and strategy. This public report[11] outlines the concept of a (digital) platform and platform eco-system from a business perspective, as well as methods for evaluating maturity and viability of such eco-systems. It further outlines key technical assets that must be provided by such a digital platform, and important qualities they must possess.

---

[9] https://trafcloud-project.se/wp-content/uploads/2019/10/TrAF-Cloud-State-of-the-art.pdf

[10] https://trafcloud-project.se/wp-content/uploads/2021/04/R1.2-TrAF-Cloud-WP1-Use-Case_Iteration1.pdf

[11] https://trafcloud-project.se/wp-content/uploads/2021/04/R4.2_Technical-Report-on-Platform-Ecosystem_2021-02-07.pdf

WP5—Evaluation and Demonstration

The deliverables produced in WP5 were all concerned with implementing and demonstrating key concepts described in R2.1. All documentation is included as a chapter in R2.1. Deviations from planned deliverables are described below (see 'Deviations from plan'). The deliverables were:

- **D5.2**, *Demonstration platform for PoC, demonstrator 1*. The first demonstrator comprised configuring cloud and edge technologies (AWS IoT Core and GreenGrass) for integration with the on-board systems. Concepts were also developed and described for using design artefacts for the onboard systems (arXML, part of the the AUTOSAR platform specification) to automatically generate software APIs and communication pipelines. These concepts were partly implemented and described in R2.1.
- **D5.3**, *Demonstration platform for PoC, demonstrator 2*. The second demonstrator implemented key part of an API Gateway that has the purpose of providing an abstraction layer to make deployment seamless. The gateway maps in runtime an invocation request from a service consumer to an available service producer. This component allows seamless deployment of service such that a consumer does not need to know available providers before-hand, and thus allows deployment of services to be determined in run-time by contextual factors (such as the quality of the connection).
- **D5.5**, *Demonstration platform for PoC, final demonstrator*. The final iteration of the demonstrator integrated the API Gateway with a hardware-in-the-loop rig used in development at Volvo Trucks. The objective with the demonstrator was to show that the concepts developed in TrAF-Cloud are compatible with existing truck technologies.

WP6—Dissemination

The deliverables produced in WP6 concerned disseminating project results.

- **D6.1**, A website was used throughout the project to distribute information and results: https://trafcloud-project.se
- **R6.2**, A dissemination plan and tracking report was continuously kept updated throughout the project, and is included in section 7 of this report;
- **R6.3**, An evaluation report is part of R2.1, and the current report constitutes a summary.

6.2 Deviations from plan

The consequences of the Corona pandemic had an impact on the project. While remote working did have some influence, it quickly became the new normal and was not considered to have any lasting or significant impact on the project results. The short term lay-offs during in beginning of the pandemic, and especially the uncertainties of how that was going to develop, did have considerable effects that lead to re-scoping. These are reported below.

- **WP1**: Initial version of the deliverables R1.2 and R1.3 (Use-case definition and Architecture requirements) were developed prior to the pandemic measures. Once activities resumed at the partner organizations, work with other WPs (WP2) was considered to already cover the objectives of the affected deliverables. Therefore, R1.2 and R1.3 are included as chapters in R2.1 (the reference architecture description).
- **WP3**: during the lay-offs, WP3 (mainly conducted as a PhD-project) were re-scoped to comprise more theoretical work. The planned deliverable **R3.1** was re-scoped to a topic about micro-service design and specifically organizational aspects of migrating a legacy monolithic architecture. This topic is highly relevance to OEMs as the traditional on-board software systems are of monolithic design. In the TrAF-Cloud vision, such monoliths need to be broken into smaller and independent units to allow defining and exposing suitable APIs, as well as to allow each to follow their own best suited development and deployment cadence. However, as access to the TrAF-Cloud automotive partners were highly limited during the pandemic layoffs, studies included several other sectors—including finance, gaming, aviation, and telecommunication. Once activities resumed, studies were done to contextualize the results to the automotive sector. The planned deliverable **R3.2** was re-scoped to a topic about automatic architecture conformance checking, where methods were

researched and developed to extract information from architecture models and source code for compliance checking. The objective is to enable automatic verification the developers do not break design in their implementation. The topic is relevant for the TrAF-Cloud architecture as it aims for a seamless integration between on-board (traditionally embedded systems) and cloud systems that have different design traditions and concerns. The ability to automatically evaluate the compliance of source code to formal design documentation has the potential to lower the threshold for a service developed to implement services that span the platforms while ensuring compliance to design. The vision is to be able to include such automatic compliance methods in the continuous integration pipeline typically used when developing for large software-intensive systems. Finally, the project contributed to a study developing methodology for benchmarking serverless applications and specifically to pin-point sources of latency (**R3.3**). For TrAF-Cloud, such methods are relevant as part of the vision is seamless deployment, where functionality that today is part of the on-board (monolithic) systems could instead be executing on a cloud (serverless) environment. Therefore, the ability to ensure low latency and high reliability of applications executing on such a platform is crucial. The WP3 deliverables were thus re-scoped to:

  - **R3.1** Academic publications relating to micro-service architectures and migrating monolithic architectures (adaptation of Result 6)
  - **R3.2** Automatic architecture compliance evaluation (adaptation of Result 6)
  - **R3.3** Performance evaluation of serverless applications (TrAF-Cloud contributed with one study)

- **WP4**: The initial objective was to investigate specifically the concrete mechanisms for a digital platform eco-system around commercial vehicles, partly to specify high-level requirements on the vehicle architecture. Due to the lay-offs, however, access to key individuals and departments were limited. The focus of the work-package was re-scoped to describe and evaluate mainly methodology for conducting such investigations.
- **WP5**: The initial plan included organizing an innovation challenge where invited third-party developers and organizations were to develop services on top of the proof-of-concept platform (Result 9). However, due to the pandemic, and specifically to the uncertainties of whether restrictions would be lifted in time, it was decided to cancel this activity. Instead, the demonstrator was integrated with a hardware-in-the-loop rig available at VGTT to investigate and describe how the TrAF-Cloud architecture can integrate with the existing on-board platform (as well as with legacy technologies). The documentation of the WP5-deliverables is included as part of R2.1.
- **WP6**: There was several dissemination activities planned for the first half of the project which were cancelled or postponed due to the pandemic. As mitigation, focus was shifted to organizing online seminars around TrAF-Cloud results and other relevant topics. The events were mostly open invitation to project partners.

6.3 Deliverables and Contribution to Project Goals

With the exception of Result 9, all expected results set out in the project proposal have been achieved, as follows:

Result 1: The **R2.1 report** produced in work-package 2 described a common reference architecture, including associated requirements and design principles as set out as one of the main expected results.

Result 2: Key parts of the reference architecture (Result 1) were **developed and demonstrated** in proof-of-concept implementations as set out by this main expected result.

Result 3: A **State-of-the-Art report (R1.1)** describing relevant cloud concepts and truck technologies were produces as set out as expected Result 3.

Result 4: An initial report draft describing **Use cases (R1.2)** to detail example system and scenarios that the project will focus on was produced (with a final version included as part of R2.1) as set out as expected result 4.

Result 5: Reports related to current and future actors' requirements for providing services in the transportation industry ecosystem such as platform boundary resources (API's, SDKs, guidelines, etc.) were detailed in **two reports (R4.1 and R4.2) as well as in a number of workshops and seminars** (see 'Other Dissemination Activities' below). These contribute to the expected Result 5.

Result 6: Result 6 was set out as a specification of a cloud service definition language that can be applied to all services regardless of the allocation to an in-vehicle ECU or cloud. This expected result was partly re-defined to focus on micro-service design and migration, automated architecture compliance checking, and performance benchmarking as described in 'Deviations from plan', above. The set of deliverables produces within the **topics R3.1, R3.2, and R3.3** contribute to the re-scoped expected Result 6. Deliverable D5.2 contributed to the original Result 6 by examining how the interface definition language specified as part of the AUTOSAR standard can be used to generate artefacts also for the off-board platform. The concepts are described in R2.1.

Result 7: The deliverables produced in R3.1 also **resulted in a licentiate thesis** with the title: '*Migrations to Microservice-Based Architectures - a Tale of Technical and Organizational Change*' (to be presented in September 2022), fulfilling expected result 7.

Result 8: The results from work-package 3 include **8 academic publications** (of which 4 are published, 3 are submitted and one in progress at the time of writing), thus fulfilling the expected result 8 (to produce least 5 scientific publications).

Result 9: The project was expected to organize an innovation challenge using the demonstration platform (WP5) to demonstrate viability of the architecture as a platform for a digital eco-system (as outlined in WP4). Due to the circumstances, this result **was not achieved** (see 'Deviations from plan' above).

Finally, the deliverables map to the project goals as flows:

- Deliverables **R4.1** and **R4.2** contribute to **G4** by examining **business service eco-systems** enabled by a connected vehicle platform;
- Deliverable **R4.1** contribute to **G4** by identifying **key vehicle technologies** that should be made available to make such a platform viable;
- Deliverables **R1.2** and **R1.3** (part of **R2.1**) contribute to goals **G1-G3** by identifying important **use-cases** and **non-functional requirements** of such architecture, and specifically investigating aspects related to security;
- Deliverables R3.1a, R3.1b, R3.1c and R3.1d (summarized in R2.1) contribute to ***G1-G3*** by exploring **micro-service design** in an automotive context, as well as identifying strategies for migrating current monolithic services to a micro-service design;
- Deliverables D5.1-D5.3 and D5.5 (documented in R2.1) contribute to ***G1*** by **developing and demonstrating technical principles** required for seamlessly connected vehicles incorporating state-of-the-art technologies from both the embedded vehicle software and cloud service domains.

7 Publications and Dissemination

This section summarizes publications, reports and various dissemination activities that resulted from the project (described in the previous section).

7.1 Kunskaps- och resultatspridning

Hur har/planeras projektresultatet att användas och spridas?	Markera med X	Kommentar
Öka kunskapen inom området	X	The project has examined novel cloud-native vehicle architecture by taking existing vehicle architecture as a point of departure as well as by aligning with industry state-of-the-art and emerging technologies. The project has thereby increased knowledge of what existing vehicle systems require to be viable in a future environment, how to realize those requirements, and what the industry trends are. The project has, furthermore, organized several seminars and workshops both within the partner organizations and external actors to disseminate and to elicit comments and opinions. As the objective was to bridge several platforms (embedded vehicle systems, on-board connectivity systems, and off-board systems) that today operate largely isolated, the project has identified common concerns and technologies that spans these platforms. It is expected that collaboration across these platforms will be necessary, and the project has contributed with relevant knowledge and recommendations.
Föras vidare till andra avancerade tekniska utvecklingsprojekt	X	The project results have been documented and has continually been evaluated how to be incorporated into development projects. In addition, new projects proposals are being developed to continue were TrAF-Cloud left off.
Föras vidare till produktutvecklingsprojekt		Most concepts developed in the TrAF-Cloud project are considered experimental and require further investigations to be incorporated in product development.
Introduceras på marknaden		See above
Användas i utredningar/regelverk/ tillståndsärenden/ politiska beslut		Not in the scope of the TrAF-Cloud project

7.2 Publications

Academic publications

Report id	Reference	Status
R3.1a	Ayas H. M., Leitner P., and Hebig R., *Facing the Giant: A Grounded Theory Study of Decision-Making in Microservices Migrations*. In proceedings of the 15th ACM / IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM'21) Pp. 1–11. , 16. New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/3475716.3475792	Published
R3.1b	Ayas H. M. Leitner P., and Hebig R,. *The Migration Journey Towards Microservices*. In Product-Focused Software Process Improvement (PROFES'21) Pp. 20–35. Lecture Notes in Computer Science. Cham: Springer International Publishing.	Published

R3.1c	Ayas H. M., Leitner P., and Hebig R., *An Empirical Study of the Systemic and Technical Migration Towards Microservices.* Journal of Empirical Software Engineering (EMSE)	Submitted
R3.1d	Ayas H. M., Fischer H., Leitner P., and Gomes F., *An Empirical Analysis of Microservices Systems Using Consumer-Driven Contract Testing* Euromicro/SEAA 2022	Accepted
R3.1e	Mortada M., Ayas H. M., and Hebig R,. *Why Do Software Teams Deviate from Scrum? Reasons and Implications*. In Proceedings of the International Conference on Software and System Processes Pp. 71–80. ICSSP '20. New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/3379177.3388899	Published
R3.2a	Hujainah F., Sharafi S., Palma F., Chaudron M.R.V., Hebig R., Khomh F., Guéhéneuc Y.G., Jolak R., Truong H., Wagner S., Wyrich M., *A Systematic Literature Review on Program-Comprehension*	Draft
R3.2b	**Automatically measuring conformance between implementation and architecture model**	In progress?
R3.3	Scheuner J., Eismann S., Talluri S., Van Eyk E., Abad C., Leitner P., Iosup A., *Let's Trace It: Fine-Grained Serverless Benchmarking Using Synchronous and Asynchronous Orchestrated Applications* (2022) http://arxiv.org/abs/2205.07696	Published

Theses

Type	Title	Author	Status
Licentiate	Migrations to Microservice-Based Architectures - a Tale of Technical and Organizational Change	Ayas M. H. (Chalmers)	To be presented Sept. 14, 2022
MSc	Automatically measuring conformance between implementation and architecture model	Satyam V. (Chalmers)	In progress, planned for XXX

Public Reports

Report id	Title
R1.1	*TrAF-Cloud State of the Art*
R1.2a	*TrAF-Cloud Use-cases* (iteration 1)
R4.2	*Design principles for future platforms and eco-systems* Technical Report on Platform Ecosystems

Confidential Reports

Report id	Title
R2.1	*TrAF-Cloud Reference Architecture – Architecture Description and Design Principles.* This report includes also R1.2, R1.3 and R5.1 (and documentation for the demonstrators)
R4.1	*Eco-system mapping and Boundary resources*

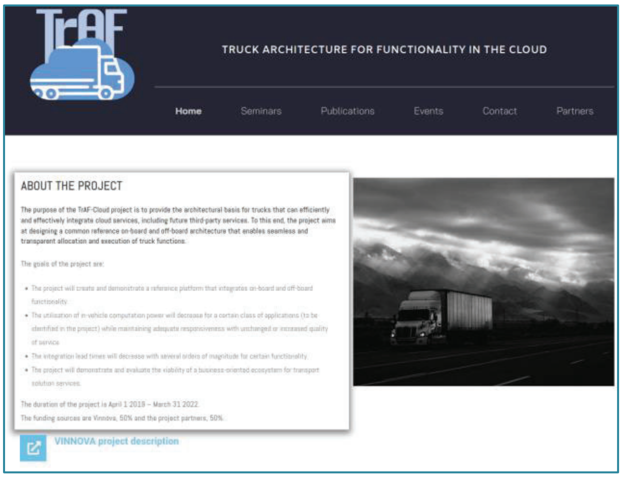
Demonstrators

In addition to academic publications and reports, the project developed demonstrators (proof-of-concept, PoC) as part of work package 5. The PoCs implemented key architectural components (identified and described in R2.1) to show how a seamless integration of on- and off-board functionality could be achieved. The PoC demonstrated a flexible approach to bridging the on-board and off-board platforms as well as how to utilize a cloud-native edge node deployed on-board a vehicle. In addition, the PoC was integrated with a hardware-in-the-loop rig internally at Volvo demonstrating the ability to use with existing technology, and possible approaches to including sub-sets of the TrAF-Cloud results into products.

Several demonstrations were organized during the project.

7.3 Other Dissemination Activities

Public material and other information was published on a website throughout the project:



ABOUT THE PROJECT

The purpose of the TrAF-Cloud project is to provide the architectural basis for trucks that can efficiently and effectively integrate cloud services, including future third-party services. To this end, the project aims at designing a common reference on-board and off-board architecture that enables seamless and transparent allocation and execution of truck functions.

The goals of the project are:

- The project will create and demonstrate a reference platform that integrates on-board and off-board functionality.
- The utilisation of in-vehicle computation power will decrease for a certain class of applications (to be identified in the project) while maintaining adequate responsiveness with unchanged or increased quality of service.
- The integration lead times will decrease with several orders of magnitude for certain functionality.
- The project will demonstrate and evaluate the viability of a business-oriented ecosystem for transport solution services.

The duration of the project is April 1 2019 – March 31 2022.

The funding sources are Vinnova, 50% and the project partners, 50%.

VINNOVA project description

https://trafcloud-project.se

In addition to presentations at academic conferences, the following open and semi-open dissemination activities have been conducted.

Date	Reference	Organizer
2021-03-08	Presented first iteration of proof-of-concept with open invitation within the Volvo Group (D5.2)	RISE and AB Volvo
2021-04-06	Seminar on the topic ‘Boundary resources and micro-service migration’ (R4.2 and R3.1a). Open invitation to all project partners	Combitech and Chalmers
2021-05-05	Seminar on cybersecurity. Open invitation to project partners	Combitech
2021-06-07	Seminar on the topic ‘Trends in on-board and backend’	Amazon Web services
2021-09-13	Seminar on the topic ‘From Business Strategy to Design Tactic: Roadmap to Designing Platform Ecosystems’ (R4.1)	Combitech
2021-10-11	Seminar and workshop on Blackberry IVY	Amazon Web Services and Blackberry
2021-12-08	Presentation of the TrAF-Cloud project to the Volvo GTT internal initiative ‘Volvo Data Logging Group’ responsible for tracking functionality that needs to collect and report diagnostic data	RISE and AB Volvo
2021-12-13	Workshop with the Connectivity department	RISE and AB Volvo
2022-01-24	Presentation on ‘Platform Boundary objects workshop’ (R4.1)	Combitech
2022-01-25	Demo of the second proof-of-concept with invitees from the Connectivity department at Volvo GTT (D5.3)	RISE and AB Volvo
2022-01-26	Seminar and workshop on cybersecurity	RISE, Combitech, and AB Volvo
2022-02-28	Seminar on the topic ‘Application-level Benchmarking of Serverless Application’	Chalmers
2022-04-01	Workshop with CEVT, presented the TrAF-Cloud project and discussed future collaboration	RISE, Chalmers, and AB Volvo
2022-04-29	Presented final proof-of-concept with hardware integration. Open invitation to project partners (D5.5)	AB Volvo
2022-05-18	Presented the TrAF-Cloud project at the Vehicle Electronics and Connected Services conference (VECS)	AB Volvo and RISE
2022-06-30	Presented the project at Volvo Group ‘Innovationsfika’ (similar to a ‘TED-talk’) with more than 700 invitees	AB Volvo and RISE

8 Conclusions and Future Work

The objective of the TrAF-Cloud project was to investigate aspects of a future natively connected architecture for commercial vehicles. These included investigating business related aspects such as business models with an digital eco-system approach to investigate the broader context in which such connected vehicles are to operate. They also included investigating guiding scenarios and high-level requirement as well as technical aspects which were realized as a reference architecture with a number of proof-of-concepts implemented in the project.

The results of the project show that to establish commercial vehicles a constituent component of a digital eco-system, there is need for a flexible on-board vehicle architecture that is designed to evolve with new cloud innovations. It is specifically within areas that are evolving at a fast pace that stand to benefit most from a more flexible architecture that is able to expose on-board capabilities as well as to integrate cloud-provided capabilities—examples include capabilities related to electro-mobility, autonomous and remote driving. The results also show that there are viable approaches to designing a connected vehicle architecture that can fulfil the goals set out while still complying with the hard requirements of the vehicle systems that has resulted in the ridged (monolithic) design of current vehicles. The project has, furthermore, contributed with research on how an organization can migrate toward a micro-service design of their systems; a migration that is not only technical in nature.

During the course of the project, several related initiatives have been announced that have a strong bearing and resemblance to the objectives of the TrAF-Cloud project—examples include SOAFEE[12], Eclipse Working Group on Software Defined Vehicle[13], Blackberry IVY Connected Vehicle Platform[14], and the automotive specific Linux distribution by Red Hat[15] that aims to support OEMs with safety assurance (related to ISO 26262). These and other initiatives have been launched recently, the timing and similarities with the objectives (and to some extent also the approach of) the TrAF-Cloud project indicate that the project is well aligned with current industry trends.

While the TrAF-Cloud project has taken a broad approach to investigate aspects on several abstraction levels, from the business level down to architectural principles demonstrated in concrete proof-of-concept implementations, there are a range of challenges that requires attention.

On a technical level, the vision of the TrAF-Cloud architecture presumes a service oriented architecture with loose binding between service producer and service consumer. On-board vehicle systems, on the other hand, have a legacy in monolithic designs with static service binding. While the AUTOSAR Adaptive standard include a service-oriented on-board communication protocol (SOME/IP), it is not clear how to arbitrate between multiple asymmetric providers of a service (e.g., that differ in provided level of quality) or how to bridge on- and off-board platforms in a generic manner (including managing availability of service providers, and routing of invocation requests). Such aspects become important when aiming for a digital eco-system as new providers of a service can become available throughout the life of a vehicle, and where that vehicle is expected to seamlessly be able to utilize such services.

Furthermore, a monolithic system design with static service binding allows analyses to assure correctness of the implementation (e.g., for safety assurance according to ISO 26262). With deeper and more flexible integration between on- and off-board systems the underlying assumptions of current engineering best practices may not hold. Therefore, there is a need to examine how those will be affected.

---

[12] https://soafee.io/
[13] https://sdv.eclipse.org/
[14] https://www.blackberry.com/us/en/products/automotive/blackberry-ivy
[15] https://www.redhat.com/en/solutions/automotive

Furthermore, to establish a digital eco-system around the vehicle there is a need to define a standard and stable set of vehicle APIs (platform boundary objects). These would serve as the abstract software-centric abstraction of the physical vehicle. Similar objectives underlie the concept of a Software Defined Vehicle (see Eclipse working group, and SOAFEE). For a vehicle to be represented as a fully capable digital twin and participate in a digital representation of a transport system—cf. traffic control towers—such a representation is required, and must also be standardized to be viable. Similar standards exist, but with highly limited scope such as rFMS[16] and ITxPT[17].

A viable digital representation of the complete vehicle would not only enable to establish it as a platform for an eco-system of to provide third-party services for the vehicles, but also provide a foundation for vehicle to become efficient constituent systems in transportation system-of-system (SoS), such as investigated in the VINNOVA project MAUS[18]. A SoS-approach to innovations has the potential to contribute on a societal level, where examples include: geofencing applications[19], automatic reporting vehicle data to ensure compliance with rule exemptions[20], and collaborative approaches to urban goods transports[21]. Common to such initiatives is that they require several independent actors to collaborate to realize the full system solution. The collaboration is not only on an organization level, but also on a technical level where the participating constituent systems (e.g., vehicles, infrastructure etc) can communication and adapt their behaviour based on the current system properties. Furthermore, as the communication needs may be dependent on the specific SoS, vehicle must be able to seamlessly adapt to the current needs. Such aspects are of interest to further investigate as it spans societal benefit, organizational collaboration as well as technical enablers. Although the TrAF-Cloud project has contributed to these aspects, the main focus was on the vehicle itself. Going forward, it is recommended to investigate how natively connected commercial vehicles can collaborate across brands and models to make our shared transportation system more efficient, environmentally neutral, and safer.

---

[16] *http://www.fms-standard.com/Truck/index.htm*

[17] *https://itxpt.org/*

[18] *https://www.vinnova.se/p/monster-for-arkitektur-och-affarsmodeller-for-urban-mobilitet-system-av-system-maus/*

[19] *https://closer.lindholmen.se/projekt/geofencing*

[20] *For instance, as investigated as part of the HCT City project, https://www.vinnova.se/p/hct-city-fallstudie-massgods-i-stader.-piloter-och-systemanalys/*

[21] *For instance, Smoovit https://www.vinnova.se/p/smooth-system-av-system-for-hallbara-urbana-godstransporter2/*

9 Project Partners and Contact Information

Volvo Group Trucks Technology
Volvo Group Connected Solutions

Research Institutes of Sweden

Chalmers University of Technology

Contacts

Name	Email	partner
Niklas Mellegård (Coordinator)	niklas.mellegard@ri.se	RISE
Daniel Karlsson	daniel.b.karlsson@volvo.com	Volvo Technology
Tobias Sternvik	tobias.sternvik@volvo.com	Volvo Group Connected Solutions
Philipp Leitner	philipp.leitner@chalmers.se	Chalmers
Christer Berntsson	christer.berntsson@combitech.com	Combitech
Per Ankarås	ankaras@amazon.com	Amazon Web Services
Bengt Samuelsson	bengt.samuelsson@vector.com	Vector