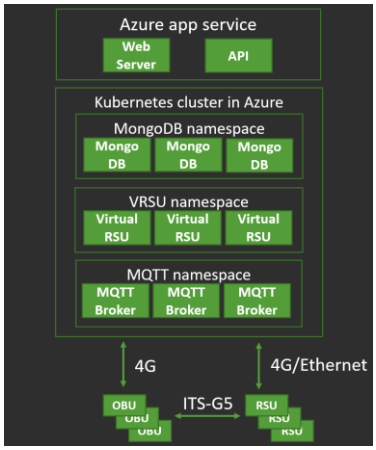
Aventi Technology

C-ITS

Complete platform for C-ITS, Cooperative Intelligent Transport Systems

Together with the Norwegian Public Roads Administration (NPRA) and SINTEF Digital, and with funding from the Norwegian Research Council's Pilot-T program, Aventi is developing a highly scalable backend system for C-ITS capable of serving millions of Vehicle ITS Stations (OBU, *Onboard Units*) and Roadside ITS Stations (RSU, *Roadside Units*). The backend system is called **Central ITS Server.**
Read more here: https://www.sintef.no/en/digital/software-and-service-innovation/secure-iot-software/asam

"If one of the Docker containers running a microservice crashes, Kubernetes will spin up a new one automatically"



1. Scalability
2. Productivity
3. Fault Tolerance

The C-ITS-S architecture is based on two current C-ITS pilot installations by the NPRA and Aventi: E8-Borealis in Northern Norway, and E6 Patterød south of Oslo. Both installations are part of the EU funded Nordic Way project.
Read more about it here: https://www.nordicway.net/

Microservices for C-ITS

Most C-ITS pilots in Europe consist for a few dozen up to a couple of thousand vehicles. This does not put any strain on the backend systems, and these solutions will have to be completely redesigned before they can be put in production for a future pan-European digital single market for road traffic.

Aventi, on the other hand, is designing its Central ITS Server from the ground up to handle all 2.5 million vehicles in Norway. It is based on microservices running in Kubernetes with scalable clusters increasing the number of pods and containers during rush hours, while reducing the number on slow weekends and holidays. This way no more resources than necessary will be running in the Microsoft Azure cloud, saving money.

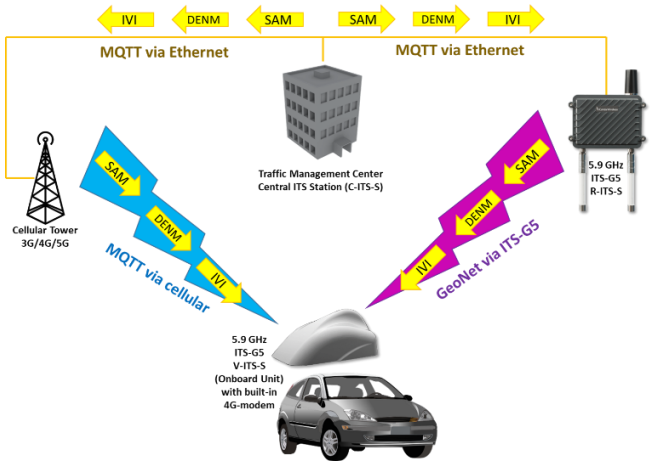
Specialized services are put in separate containers, avoiding one big monolith program, making system development and operations more productive. And if one container crashes, the load is transferred to the other containers while a new container is started automatically.

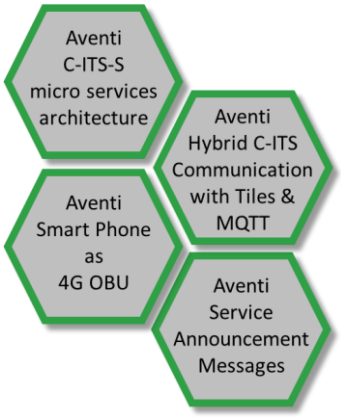
MQTT + Kafka

Currently we are using MQTT to communicate with the OBUs and RSUs via Telenor's cellular 4G network and the NPRAs own fiber Ethernet along the roads as illustrated in the figure to the left. MQTT is perfect for this type of IoT communication where the connection sometimes may drop out. We are however extending this, where MQTT hands over internal communication in the backend system to Kafka.
Read more about it here: https://www.emqx.io/blog/44

We have achieved some ambitious, but straight forward goals with our C-ITS solution:

1. Must cover all vehicles
2. Must work everywhere
3. Must do everything





Something new

Numerous C-ITS pilot projects have focused on use cases for SPaT, MAP, IVI, DENM, and CAM messages. So Aventi saw no great need to redo all that good work. Instead we focused on missing pieces in the C-ITS ecosystem, as described in this document.

Contact information:

Aventi Technology AS
Østre Aker vei 19
NO-0581 Oslo
Norway
+47 22 10 67 70

Aventi Smart Phone as 4G OBU

As described on the first page, we are developing a scalable backend system that can communicate with all 2.5 million vehicles in Norway. In addition, we are developing an Android app in Android Studio that can connect to the MQTT Brokers in the same way as the RSUs and OBUs do. In this way, no special equipment is needed in the vehicles, the smartphones in many cases nowadays being more powerful than the RSUs. The Android app is also capable of connecting to low cost OBD-II/Bluetooth adapters found in many auto parts stores, enabling reading of ECU PIDs. Alternatively, the app can collect some of the same data from the vehicle OEMs via the API from smartcar.com.

Aventi Hybrid C-ITS communication

We are using OBUs from Kapsch, Cohda Wireless, and Q-Free in our projects, where we add USB/Bluetooth communication to an Android app, and more importantly 4G cellular communication for areas with no RSUs present. This is illustrated in the figure above to the right, and standard message types are exchanged in both ASN1 and JSON format. For Kapsch and Cohda Wireless we develop our applications using C-programming, while for Q-Free we develop applications using Java and OSGI. All three vendors provide SDKs for interfacing to their stack.

Aventi Service Announcement Messages

ISO and ETSI have specified a message type called SAM (Service Announcement Message) which can be used to bring other traffic related services into the C-ITS platform like Road Use Charges and geo fencing use cases. Aventi is in the forefront in Europe with respect to SAM based use cases. SAM can be used to do everything.

Aventi Web Server and API

The Aventi Web Server with map and API are .Net applications developed in C# in Microsoft Visual Studio.

Demo videos:

Here is a short demo video: https://youtu.be/aVkTRgRFQ9o
Here is a long demo video: https://youtu.be/iy0QqnO6mDQ