This proposal has been approved and the **Eclipse Kuksa (/projects/iot.kuksa)** project has been created. ×

Eclipse Kuksa

BASICS

This proposal is in the Project Proposal Phase (as defined in the **Eclipse Development Process (http://www.eclipse.org/projects/dev_process/development_process.php)**) and is written to declare its intent and scope. We solicit additional participation and input from the community. Please login and add your feedback in the comments section.

Parent Project:

Eclipse IoT (/projects/iot)

Background:

Eclipse Kuksa provides technologies across the connected vehicle domain.

Automotive systems especially lack open technologies that are accessible to developers and enterprises focusing on IoT (Internet of Things) systems or cloud applications. Eclipse Kuksa closes these gaps and provides the comprehensive ecosystem to address related challenges with the help of platforms for (a) in-vehicle, (b) cloud, and (c) connected vehicle applications development. The publicly funded European ITEA research project **APPSTACLE** forms a basis for addressing such challenges. Eclipse Kuksa will further bring the ecosystem into the Eclipse community in order to widen its applicability and usage amongst and beyond its project partners such as Ericsson, Bosch, NXP and others.

The name Kuksa has its origins in a carved wooden mug from Finland. A Kuksa, with its individual character, has to be shared with and eventually bequeathed by close friends and cannot be bought for oneself. This sharing, as well as the connection that extends across generations, is meant also to reflect the spirit of the open source Eclipse Kuksa project.

Scope:

Eclipse Kuksa unifies technologies across the vehicle, IoT, cloud, and security domains in order to provide an open source ecosystem to developers addressing challenges of the electrified and connected vehicle era. In-vehicle and cloud applications as well as their development infrastructure, maintenance approaches, security, and over-the-air (OTA) update and upgrade management are, with a few exceptions, completely within the scope of Eclipse Kuksa.

Currently, vehicle connectivity applies to mobile networks (and related infrastructure such as clouds, servers, etc.), other vehicles, and local infrastructures (traffic lights, local entities cf. edge computing). Although proprietary approaches provide partial solutions, developers have no access to either open source solutions nor comprehensive ecosystems that allow common connected vehicle development initiatives across suppliers, original equipment manufacturer (OEM), and tooling enterprises. Eclipse Kuksa brings comprehensive connectivity solutions to the public while combining diverse technologies of vehicle, IoT, cloud, and security domains.

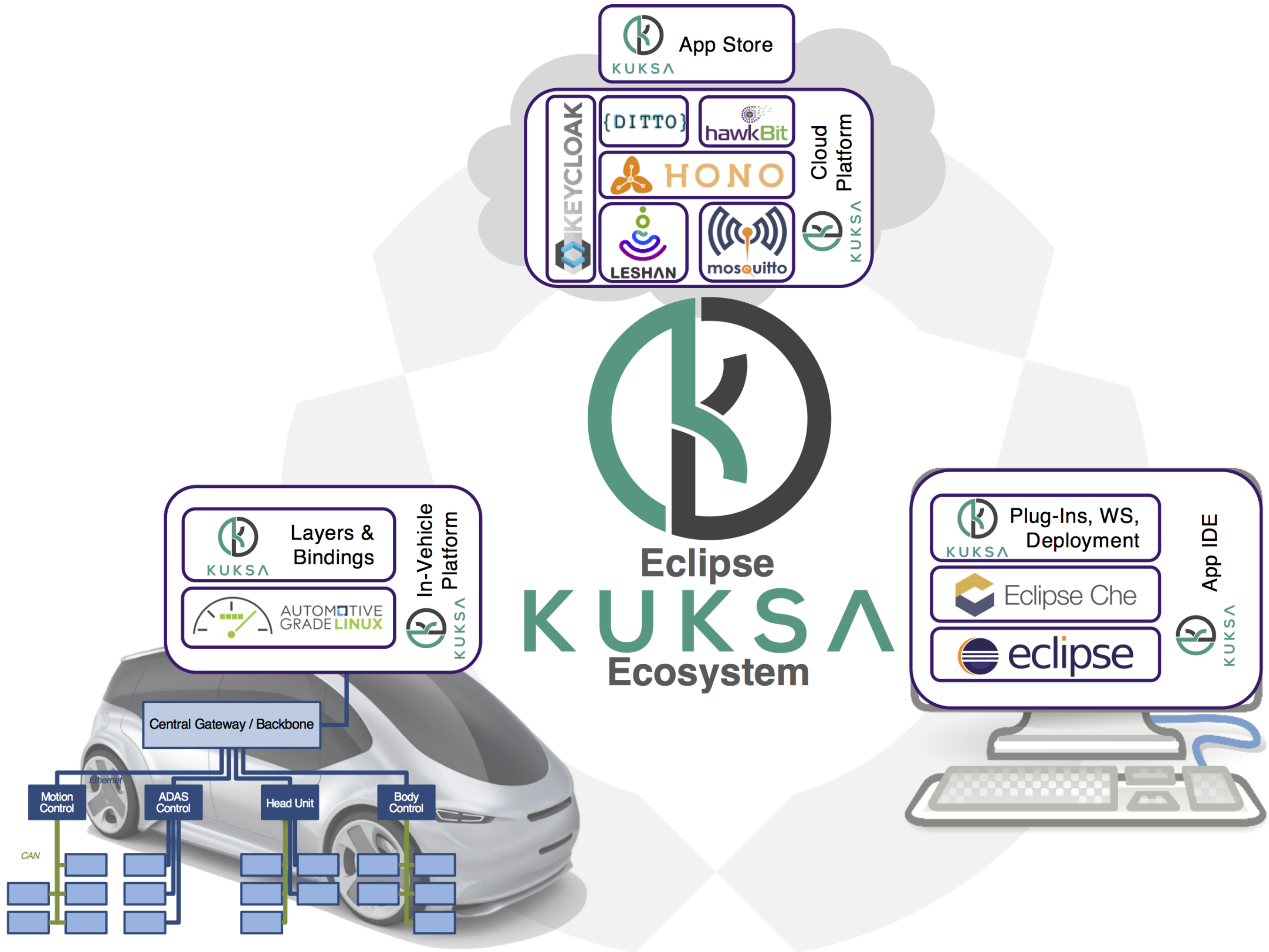
While safety is out of scope, Eclipse Kuksa addresses maintenance through providing, but not applying, OTA updates or upgrades, Cloud, and in-vehicle applications, as well as state-of-the-art platforms and technologies for in-vehicle communication and data transmission. Demonstration applications will be made available, but sophisticated cloud and in-vehicle applications are generally subject to community or closed-source development processes. Working with this premise, Eclipse Kuksa will support all the necessary development technologies required for such applications.

Description:

Because today's software-intensive automotive systems are still developed in silos by each car manufacturer or OEM in-house, long-term challenges in the industry are yet unresolved. Establishing a standard for car-to-cloud scenarios significantly improves comprehensive domain-related development activities and opens the market to external applications, service provider, and the use of open source software wherever possible without compromising security. Connectivity, OTA maintenance, automated driving, electric mobility, and related approaches increasingly demand technical innovations applicable across automotive players.

The open and secure Eclipse Kuksa project will contain a cloud platform that interconnects a wide range of vehicles to the cloud via in-car and internet connections. This platform will be supported by an integrated open source software development environment including technologies to cope especially with software challenges for vehicles designed in the IoT, Cloud, and digital era.

This ecosystem will provide a comprehensive environment across various frameworks and technologies for (a) the in-vehicle platform, (b) the cloud platform, and (c) an app development IDE - that is, the complete tooling stack for the connected vehicle domain (see Figure 1 below). Essential to this environment will be the capabilities for collecting, storing, and analysing vehicle data in the cloud as well as the transmission of diverse information such as cloud calculation results (e.g. improved routing), software maintenance updates or even complete new applications. While many IoT solutions exist in the Eclipse IoT ecosystem, Eclipse Kuksa combines the necessary existing technologies and fills the gaps for the specific requirements of the connected embedded real-time nature of the automotive domain.



Why Here?:

Eclipse is widely used in the automotive software industry as well as the IoT domain. Eclipse Kuksa will add value to the Eclipse community by establishing a connected vehicle ecosystem consisting of three major platforms. The ecosystem will be a basis for building customized, proprietary, open- and closed source systems. The Eclipse Kuksa project expects to benefit from wider visibility in the community and an extended open collaboration with interested parties.

Technical details and cooperation:

- The Eclipse Kuksa ecosystem is built on existing Eclipse Frameworks such as:
  - Eclipse Hono (Cloud backend and frontend, message routing)
  - Eclipse Hawkbit (device management, OTA)
  - Eclipse Ditto (digital twins: abstraction levels, synchronization, state,....)
  - Eclipse Leshan (IoT management, lightweight m2m)
  - Eclipse MosQuitto (MQTT)
  - Eclipse Keti (Security)
- More technologies likely to be used:
  - Eclipse Sumo (traffic simulation)
  - Keycloak (identity access management, security)
  - Automotive Grade Linux (AGL, in-vehicle basis platform)
  - Eclipse BIRT (Dev IDE visualizations)
  - InfluxDB, REST API, ...

Licenses:

Eclipse Public License 2.0 (/license/epl-2.0)

Legal Issues:

All committer agreed to publish under the EPL.

In-Vehicle developments will adapt to Automotive Grade Linux (AGL) that is partially LGPL and GPL. The Eclipse Kuksa developers will clarify if AGL code can be fully defined as a dependency i.e. that no AGL code must be included in Eclipse Kuksa. However, an adapted AGL build will be secondary licensed such that the combined work will be licensed under the GPL-2.0, and then distributed as combined work under the GPL-2.0.

AGL content that includes a secondary license clause will link directly with content that is licensed under the terms of that secondary license (see EPL2.0 FAQ). Secondary licensed build results are not intended to be hosted at Eclipse.

Project Scheduling:

- Initial contribution expected by Q1 2018 (Kuksa 0.1)
- Release cycle for the cloud platform is planned every 6 months (Kuksa 0.2 by Q3 2018, etc.)
- With the end of the publicly funded APPSTACLE project, the first major release Eclipse Kuksa 1.0 is expected
- In-vehicle and APP Dev IDE development activities are not planned yet

Future Work:

Eclipse Kuksa will combine the technologies mentioned in the 'Why Here?' section in order to establish the comprehensive ecosystem that securely allows using vehicle data and transmitting generated cloud results. Concurrent work packages thereby address connectivity challenges, demonstrations, as well as cloud and in-vehicle implementations. By Q4 2018, the infrastructure should be complete and by Q4 2019 the ecosystem should be fluently working with the demonstrators and libraries and APIs should allow tailoring cloud applications towards individual needs.

In order to grow the community around Eclipse Kuksa, its development will be spread across research publications at conferences, workshops, booths, or similar events as well as newsletters around the cloud, IoT, and automotive domains. Consequently, interested and related developers will clearly notice Eclipse Kuksa's benefits and hopefully participate at its open source basis. Already its close relationship with other projects will strengthen Eclipse Kuksa's dissemination. Additionally, the involved partners of the publicly funded APPSTACLE project will utilise Eclipse Kuksa around their developments and include it for customer related projects.

PEOPLE

Project Leads:

Robert Hoettger (/user/6264)
Johannes Kristan (/user/8023)

Committers:

Robert Hoettger (/user/6264)
Johannes Kristan (/user/8023)
Lukas Krawczyk (/user/5900)
Philipp Heisig (/user/8006)
Ahmad Bani Jamali (/user/9905)
Pedro Cuadra
Steffen Evers (/user/8039)
Philippe Krief (/user/7880)
Susan Iwai
Sebastian Schildt (/user/9906)
Marco Wagner (/user/9904)
David Schubert (/user/10397)
Peer Adelt
Jörg Tessmer (/user/8218)
Alexios Lekidis
Tobias Rawald

Mentors:

Jim Hughes (/user/4891)

Interested Parties:

- Robert Bosch GmbH
- Dortmund University of Applied Sciences and Arts
- NXP Semiconductors
- Technical University Eindhoven
- Ericsson
- Behr-Hella Thermocontrol GmbH
- Paderborn University
- SecurityMatters
- University Oulu
- taskit GmbH
- Technolution
- Halian Ltd
- Link Motion
- Tieto

- Assystem GmbH

SOURCE CODE

Initial Contribution:

Eclipse Kuksa will be divided into several repositories:

Github (basically for in-vehicle developments)

1. `Kuksa.AGL` (wrapper: scripts, separate builds for different licenses)
2. `Kuksa.meta` (recipes to utilize basic Kuksa technologies, e.g., regarding Eclipse Hono, Eclipse Mosquitto, ...)
3. `Kuksa.meta-dev` (recipes to utilize development tooling such as debugging, tracing, logging, ...)
4. `Kuksa.meta-rover` (available at https://github.com/mozcelikors/meta-rover)
5. `Kuksa.docs` (rst - sphinx - travis CI to build documentation webpage)
6. More Kuksa layers subject to community

Eclipse (IDE and bindings / libs)

1. `org.eclipse.kuksa` (IDE - plugins and build to support cloud and in-vehicle application development activities)
2. `org.eclipse.kuksa-apps` (e.g. for cloud related components. The initial contribution will contain a Spring Boot application with docker scripts to connect an Eclipse Hono instance with a consumer to allow parsing, storing, and visualizing any registered client's raw data)
3. `org.eclipse.kuksa-hono` (AGL bindings for Hono)
4. `org.eclipse.kuksa-mqtt` (AGL bindings for MQTT)
5. More binding repos very likely to be required

Source Repository Type:

GitHub (/source-repository-type/github)

SIGN IN (/USER/LOGIN?DESTINATION=PROPOSALS/ECLIPSE-KUKSA#COMMENT-FORM) to post comments.

(https://www.eclipse.org/projects/what-is-incubation.php)

KUKSA

RELATED PROJECTS

Project Hierarchy:

» Eclipse IoT (/projects/iot)
» Eclipse Kuksa (/projects/iot.kuksa)