OpenADx Manifesto

1. Autonomous driving & software involved

Software is more and more dominating our daily life, also in the automotive world. Almost every function in the vehicle is defined by software. Solutions for Autonomous Driving solutions introduces a new complexity into the development of embedded systems in the vehicle. This complexity rises with each level of control and autonomy of the automated driving systems and you need deeper expertise and more software. The focus on software is higher than on "driving computing and data centers", which you can call "embedded vehicle clouds".

Software is more and more becoming an important differentiator for autonomous vehicles.

New players appear on the market who develops functions from scratch without any legacy and new and/or adapted development methods. You have also to use them to cope with the requested development speed which is needed to stay in touch with expectations.

New technologies like AI has to be added to reach the goals and intended functionalities of Autonomous Driving, but you have also to deal with legacy systems which has to be also integrated into the vehicle system. There is a high demand on different and new technologies as well.

All these new things have to be adopted, knowledge has to be built-up and integrated to stay competitive.

Who should read this document?

This document is created to inform:

- Anyone who is searching for solutions for autonomous driving
- Anyone who wants to collaborate in an open environment
- Persons interested in developing tools and functions for autonomous driving and more
- Persons interested in Open Source in an automotive world
- Companies who see the benefit of using Open Source
- Tool vendors who would like to bring their commercial solution into operation together with open solutions without changing their business model too much
- Anyone interested in our work

2.Challenges in AD software engineering

Autonomous Driving (AD) is a complex challenge

Increasing embedded software complexity with an increase in level of automation is putting new demands on the existing tools and frameworks. Instead of solving these challenges alone, partnering in non-differentiating areas and collaborating on a toolchain through the OpenADx ecosystem is the path forward.

Few reasons for this recent rise in complexity are:

1. Deploying machine learning and deep learning algorithms in embedded software
2. Multi-domain simulation of virtual vehicles to capture errors early and to verify and validate
3. Virtual test driving solutions should interface seamlessly with the embedded software
4. Scaling up simulated kilometers

The toolchain for such challenges is complex and the integration of all the tools coming from different domains costs a lot of effort for each market participant without a real competitive advantage towards the solution. Therefore, we began to enhance the OpenADx ecosystem including OEMs, tier 1 suppliers, tool vendors, research organizations but also companies from related industries like the IT industry.

The goal of this endeavor is to reduce spend on the introduction of a proprietary toolchain in each of the companies and to share the development costs for the toolchain. Our goal is not to define a single toolchain with fixed elements, but rather integrate existing tools, even if they are open source or closed source. Every tool used or that should be used will find its place in the OpenADx toolchain, possibly requiring adapted interfaces or data models for integration.

Besides the benefit of sharing the costs, the expected result is a better integration of the toolchain within the organizations, but especially also at the interface between cooperating organizations. For tool vendors and research organizations, the advantage of the approach is in the existence of an integration backbone which allows the provider to easily integrate new technologies or tools into a working environment that runs in a multitude of customer organizations, instead of providing proprietary solutions for single customers.

Need for Data

The use of Artificial Intelligence (AI) for AD relies heavily on massive amounts of (annotated) data for training and validation. This is a major barrier for many small to medium enterprises (SME) and even for large players. The dependency on data is even more important when looking at “living” AI systems, i.e., neural networks that can be adapted for training purposes, learning from the driver or even adapting to driving styles they are exposed to. Current approaches for AD are based on closed data loops that effectively hinder cooperation and progress. The relevant edge cases that truly advance the development of a self driving system occur so rarely that millions of kilometers must be driven on real roads in order to validate the safety of such AI-based systems.

This leads to the following problems:

- Development is closed, data and algorithms are proprietary
- Systems and algorithms are limited to well-defined areas (e.g. parking, highways)
- Much of the effort is duplicated, results are not easily transferable or reusable due to incompatibilities
- Collaboration is needed to accelerate innovation and progress

The ability to share and quality-assure the relevant real-world drive data using an open data platform for the free exchange of AI data that will accelerate the creation of AI-based systems for AD. In particular, an open AI data platform will remove significant barriers for AI-based startups as it will provide an abundance of annotated high-quality data, thereby removing market entry barriers for innovation in AI-based AD.

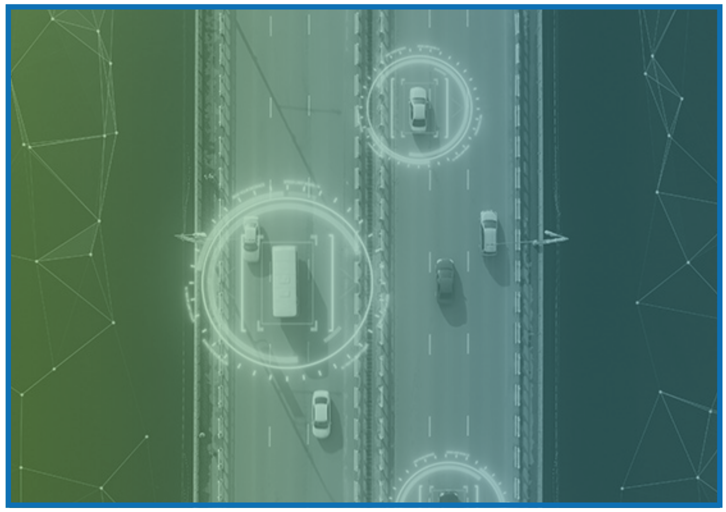
3. OpenADx - xcelerate your Autonomous Driving development

What is OpenADx?

OpenADx ("open" -> open for everyone, "AD" -> autonomous driving, "x" -> accelerate) is an industry approach to collaborate on neutral ground on toolchains for developing Autonomous Driving solutions. The goal is to set up toolchains with less effort to accelerate the development of Autonomous Driving solutions.

All interested parties are invited to join and support all of us in getting the right toolchains for Autonomous Driving development.

Collaboration of companies requires a legal body (e.g. imprint of website, hosting of open source activities and open source management processes) which OpenADx found in the Eclipse Foundation.



The Eclipse Foundation

"The Eclipse Foundation provides our global community of individuals and organizations with a mature, scalable, and business-friendly environment for open source software collaboration and innovation. The Foundation is home to the Eclipse IDE, Jakarta EE, and over 375 open source projects, including runtimes, tools, and frameworks for cloud and edge applications, IoT, AI, automotive, systems engineering, digital ledger technologies, open processor designs, and many others.

The Eclipse Foundation is a not-for-profit organization supported by over 300 members, including industry leaders who value open source as a key enabler for their business strategies. [5]

The Eclipse Foundation provides exactly the governance, legal, and community-building support we need for our working group:

- Industrial Working Groups
- Open source contribution management/legal support
- Eclipse Public License
- Eclipse Foundation Bylaws (Eclipse Bylaws, Industry Working Group Process, Eclipse Intellectual Property Policy, Eclipse Anti-Trust Policy)
- Experience in ecosystem setup and management
- Good member representation in Europe and the US to support cross continental collaboration

The OpenADx Working Group

Since 25th June 2019 OpenADx is an official Eclipse Working Group [1] [2]. Currently we have the following members:

ADLINK, AVL, DENSO Crafting the Core, eteration source of open, EclipseSource, IBM, itemis, JCIM JC Information Management, Microsoft, open robotics, Red Hat, BOSCH, science + computing an atos company, SIEMENS, Fachhochschule Dortmund University of Applied Sciences and Arts

(Please find the current members[6] on our website[3] and more information on our wiki[4])

OpenADx collaboration model

OpenADx is about working together to solve a common problem in an open environment where ideas and concepts can be brought together. By pooling our resources and competencies for the toolchain, we can free up resources in our respective companies to focus on other organizational needs.

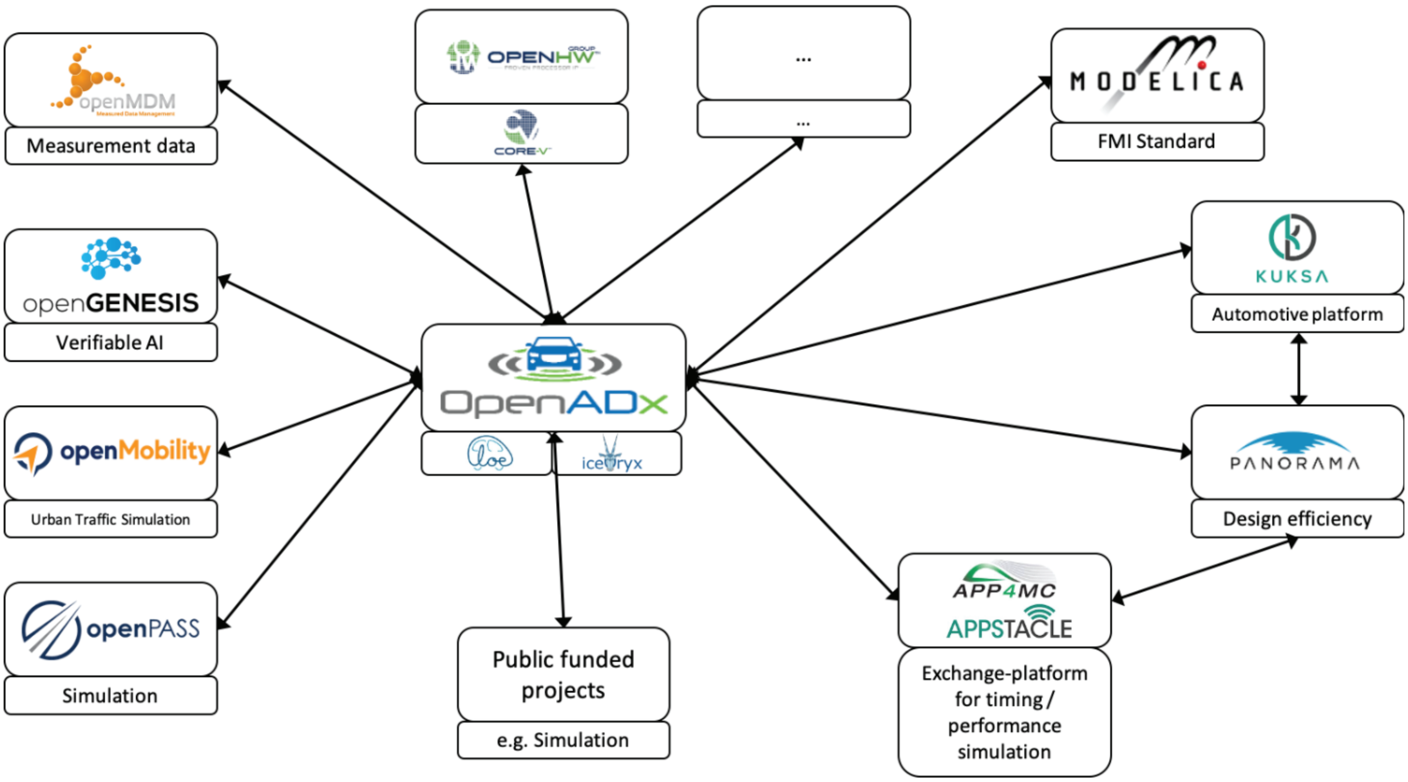
OpenADx embraces open solutions and does not compete against others.

The target of OpenADx is to embrace open solutions, primarily Eclipse based open projects.

OpenADx integrates exciting solutions and can be commercial or open, independent of the license will be considered, described in toolchain proposals and books for anyone to use as is or adapted as needed.

Commercial solution providers who integrate with OpenADx projects may choose to make their solutions proprietary, even as they build on top of the provided tooling platforms.

Collaboration with other Eclipse Working Groups



As the development is really complex and requires a variety of tools, it is best to not reinvent the wheel. Ideally we recommend using available solutions and working together to build a great ecosystem. In this spirit, OpenADx collaborates with other working groups who focus on other important topics for Autonomous Driving development.

Open Source Foundations joint effort to advance Autonomous Driving

Eclipse Foundation, Autoware Foundation and Open Robotics are working together to advance autonomous driving for passengers and cargo delivery.

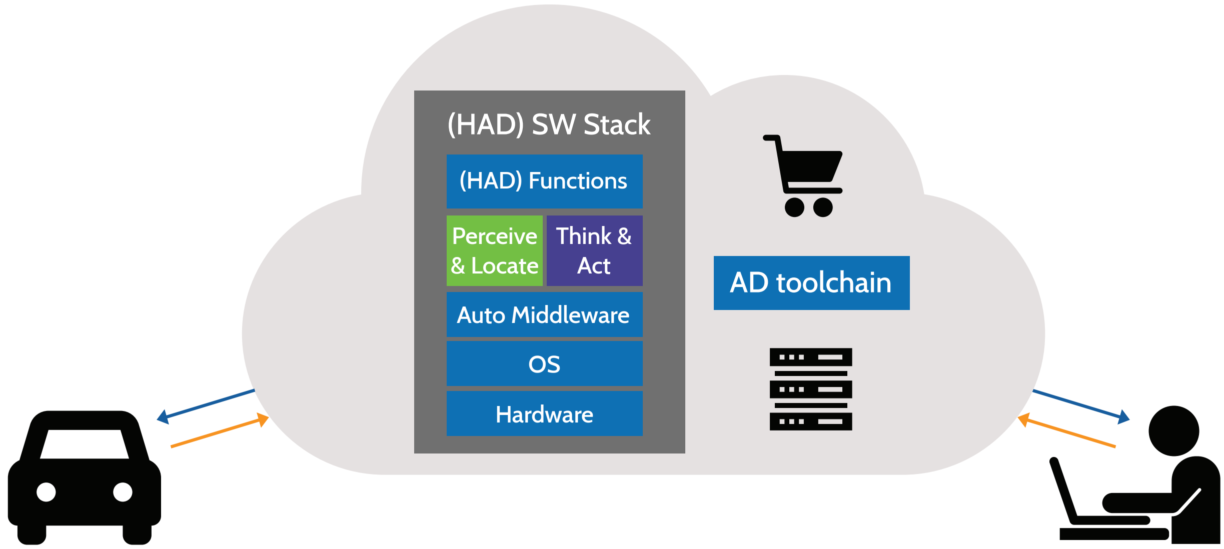
Cooperation with partners outside Eclipse

OpenADx focuses also on standardization of interfaces and data models between tools to ensure an interoperability within the OpenADx toolchain. ASAM can help us to shape and define standards.

ASAM (Association for Standardization of Automation and Measuring Systems - https://www.asam.net)[13] is a non-profit organization that promotes standardization for tool chains in automotive development and testing.

As we operate as an open working group, we collaborate with all and are always seeking collaboration partners.

Project vision



OpenADx focuses mainly on the development toolchain. To have easy and fast access to the toolchain, it should be hosted in the cloud, to ensure an up-to-date, maintained and hosted toolchain, -accessible by everyone, everywhere, without requiring installation.

The purpose of a toolchain is to create cool innovative software products! It is essential that software creation and development is easy as well. Therefore we would like to have a marketplace which provides you software parts, you can combine to new products or develop your own functionalities and share them in the marketplace. The same will happen with tools! You can use the provided toolchain or you can build your own, develop some tools and share them with the community.

Hosting in the cloud provides the advantage of high performance computing for resource-consuming applications such as simulations. You will also be able to store results, scenarios, data and a lot more in cloud storage, so that we shorten the way between the developer and the test fleet.

Therefore we focus on three main building blocks:

- The AD toolchain (ecosystem) and its reference architecture for interoperability
- The software stack (ecosystem) and its reference architecture for interoperability
- Infrastructure like integration backend (ecosystem)

Beneficiaries and why you should participate!

Our goal is to integrate a range of existing products into the development suite. We believe an initiative like this should be inclusive, not exclusive. This will ensure transparency and make a complex tool landscape more accessible to enterprise users.

OEMs and Tier 1s will benefit from the integration with increased efficiency.

Being integrated in a standardized tool chain framework makes the tooling providers more attractive to the companies using their products as well.

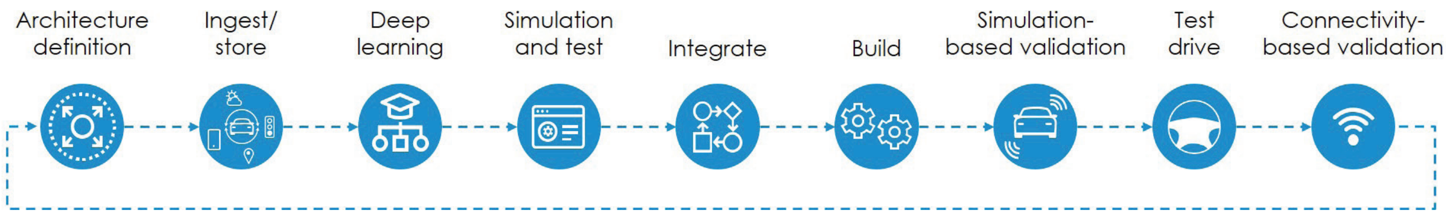
Combining the domain knowledge of all participating companies, independently in which domain they are working, gives the working group unbeatable chance to build great solutions.

In other words ... everyone wins.

4. AD toolchain

The full toolchain for autonomous driving consists of a high number of tools that must exchange data and interoperate to support a seamless development process. The integration of heterogeneous tools is often done on a per-company basis. OpenADx can serve as a forum to share experiences and efforts.

A full AD toolchain consists of several components for:



- Data ingestion, transformation and management
- Data analytics
- Machine learning training
- Open-Loop Model/Software in the Loop (MiL/SiL) testing
- Closed-Loop MiL/SiL testing using simulation environments
- Hardware in the Loop testing, both Open- and Closed-Loop
- Engineering workplaces with software development tools, data management tools, scenario editing tools and many more
- Continuous Integration and Continuous Delivery (CI/CD) platform - for a single tool as well as for a toolchain
- Project or organization specific services like specialized database

4.1. Simulation and test / Simulation-based validation

Simulation is one of the core functionalities for AD development. It is not possible to satisfy the requirements on reproducibility and quantitative coverage of driving scenarios with real physical vehicles. Simulation tools are used in all steps of the development, e.g. for creating synthetic training data (automated labeling), training and validation of ML algorithms and system validation (coverage of driving scenarios). OpenADx will cover approaches on integrating simulation tools (commercial and open source) into the tool chain.

Current offering: Eclipse Cloe[7] as a simulation middleware to connect the simulation engine with the software-under-test, further solutions will come.

4.2. Architecture (SUT) , adapter and middleware can be excluded

The term architecture in the context of OpenADx is at least two-fold. To the extent that a simulation of the developed system is executed, a formal architecture is required to describe the structure and input/output of the system-under-test (SUT). This could be specified e.g. by means of UML/SysML, AUTOSAR or other standard modelling method.

In addition, OpenADx also requires means to describe the "infrastructure architecture", i.e. everything that is required to perform the simulation, e.g. required HW (virtual machines, real machines), required SW (simulation, test, etc.).

OpenADx provides a forum to discuss / develop a common approach to specify these architectures.

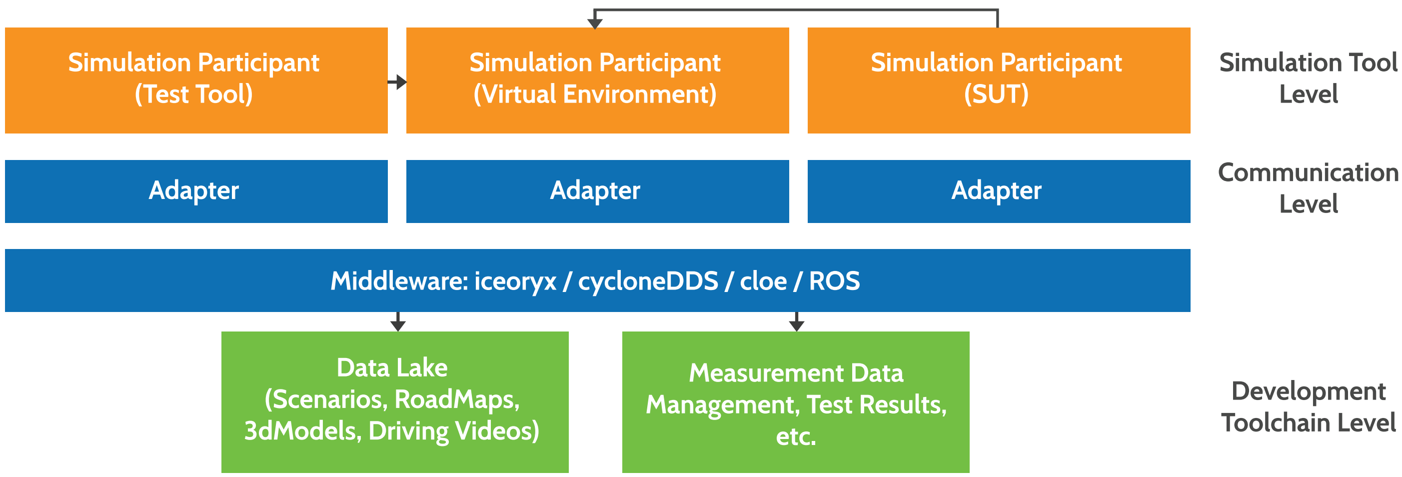
4.3 Integration and build

OpenADx deals with tools on three levels / layers:

1. The **simulation level** covers the semantic interoperability of the participants in a simulation. The involved tools must speak a “common language” so that a meaningful data exchange is possible. E.g., the test tool must know which stimulations the SUT needs and what is to be expected as output.

2. On the **communication level**, the tools in the simulation must use the same communication mechanism. A test tool that has an interface based on ROS could not speak to a SUT based on AUTOSAR. The Eclipse ecosystem already provides projects such as Iceoryx, CycloneDDS and Cloe which will be investigated for the use for this situation.

3. On the **development tool** chain level, the integration of other tools in the development chain is discussed. This focuses on data exchange (import, export, common data). As compared to the communication level, this is not at runtime of a simulation.



5. AD Reference Software Architecture

A full AD toolchain consists of several phases each with different requirements that can be possibly addressed via different infrastructures out of the box.

In this chapter, first we discuss alternatives for hosting OpenADx toolchain and the requirements they can manage to serve (chapter 5.1).

Afterwards, we elaborate on the requirements for various components of the toolchain (chapter 5.2) and suggest a matching infrastructure (chapter 5.3).

5.1 Infrastructure: Requirement Mapping

There are several infrastructure options for running the OpenADx toolchain environment:

1. On-Premise Data Centers
2. Hyperscale Cloud Infrastructures (IaaS = Infrastructure as a Service)
3. An Intelligent Cloud (PaaS = Platform as a Service) with platform services, like fast data ingest, AI and DevOps.
4. Hybrid Cloud which means to serve requests with a combination of on-premise and public cloud (e.g., system integrator via a service provider). This implies :
    1. having the possibility to extend on-premise data centers with cloud resources as soon as the demand is beyond the capability of the on-premise infrastructure.
    2. hosting part of the service in cloud and partially on-premise, i.e. Hardware in the Loop-Testing requires specific HW, that can not be integrated to a highly standardized public cloud infrastructure.

5.1 Infrastructure: Requirement Mapping (Continued)

In the following we discuss decision criteria to choose the right infrastructure based on requirements.

- **High Storage Demand:** OEMs claim that they need multiple hundreds of PB of data to develop algorithms for autonomous cars. This data is generated and has to be available worldwide, while the algorithms need direct access to the data.

- **Compatibility with Toolchain and Data Gravity:** Since BigData cannot physically be moved around to the infrastructure of the algorithms, the algorithms of the toolchain have to run on the infrastructure in which the data resides, or is replicated economically. The infrastructure has to be compatible with the requirements of the applications.

- **Hardware Scalability and Availability:** Algorithms of the OpenADx Toolchain have specific Hardware requirements. Some toolchain components require multiple of 10.000 cores of computing resources.

- **Flexible computing demand:** On-Premise data centers require a huge upfront investment, while the demand for training, simulation, and DevOps remains flexible. The sizing for on-premise datacenters is therefore critical to balance return-of-invest, while maximizing utilization and speed of execution. Cloud infrastructure enables the fast provisioning of modern Hardware resources, without any upfront investments. This would be ideal for smaller companies, like innovative startups or even smaller departments of engineering service providers.

- **Hardware Evergreen:** Hardware capabilities are changing fast from year to year. This raises fast replacement cycles for on-premise data centers, e.g. latest GPUs, FPGAs Hardware. Cloud datacenters enable a fast Hardware refresh, since older Hardware can be redeployed for less critical and demanding workloads.

- **Fast Data Ingest:** The huge amount of data is generated by test fleets, around the world. Some cloud infrastructures offer services for efficient data ingestion, e.g. high bandwidth connections or offline solutions, like storage boxes for cars or fleet locations.

- **CI/CD - Continuous Integration, Continuous Deployment:** The last step of the toolchain is the deployment of new algorithms to millions of cars, where algorithms have to work properly. Some cloud infrastructures offer intelligent edge mechanisms to continuously deploy new algorithms to specific cars on-time and over the air.

- **Artificial Intelligence Services:** Intelligent Cloud service offers a full AI stack, ranging from specific HW, data lake, services for data scientists, support popular deep learning frameworks and data science tools, up to pre-trained domain specific models, including ML-Ops Services.

- **Integrated Security:** The infrastructure has to support high-level security to protect engineering results, while open data up the data in a fine granular way for a consortium of toolchain providers. It also has to support a fine grained rights management for all parties.

- **Compliance:** The infrastructure has to be compliant to security and data protection standards like TISAX.

5.1 Infrastructure: Requirement Mapping (Continued)

We apply the decision criteria to the different Infrastructure options. The OpenADx Toolchain Workgroup decided for an intelligent cloud approach, to speed up the time to provide a ready-to-go infrastructure for all Workgroup members. An example for an intelligent cloud implementation is the Microsoft Azure Cloud.

	On-Premise	Hyperscale Cloud (IaaS)	Intelligent Cloud (PaaS)	Hybrid Cloud
High Storage Demand	Limited	Yes	Yes	Yes
Compatibility with Toolchain	Yes with exceptions	Yes with exceptions	Yes with exceptions	Yes
Data Gravity	Limited	Yes	Yes	Depends on architecture
Hardware Scalability & Availability	Limited	Yes	Yes	Yes
Flexible computing demand	No	Yes	Yes	Yes
Hardware Evergreen	No	Yes	Yes	Yes
Fast Data Ingest	No	No	Yes	Yes
CI/CD - Continuous Integration, Continuous Deployment	No	No	Yes	Depends on the Architecture
Artificial Intelligence Services	No	No	Yes	Yes
Integrated Security	Yes	Yes	Yes	Need to be implemented
Compliance	Need to be implemented	Yes	Yes	Need to be implemented

Finally, we discuss various alternatives for hosting a reference software architecture for an AD toolchain **phase/req/infra** (chapter 5.4).

5.2 Infrastructure requirements for the OpenADx Toolchain components

Toolchain components have different **requirements** for the underlying infrastructure to run optimally, e.g. storage/file system characteristics, interconnect or use of accelerators and different formats to communicate. In the following we focus on detailed storage and acceleration requirements of specific toolchain components and we provide a high-level aggregated view in the next section.

Req1: Storage/file systems (which storage solution is the best for which component of AD toolchain)

Ingest and store components of openADx toolchain requires a storage solution to deal with hundreds of PB of data scalably, cost effectively and globally. The priorities for this central data lake are on scalability, data security/safety and low cost per TB. Since only a fraction of the overall data is actively used this data lake can be considered warm/cold storage and thus does not have high requirements on the performance (latency, bandwidth). This means Cloud/Object Storage technologies are a good match for this use case.

In contrast to this HPC clusters for Machine Learning (ML) require high performance storage with a HPC parallel file system. For ML training, especially when doing scale out training over multiple nodes to speed up individual experiments, a parallel HPC file system backed by high performance SSD based storage is needed to deliver the required throughput and metadata performance at low latency to keep the ML training accelerators busy. Since these annotated datasets for ML are only a fraction of the overall AD data, the focus for such solutions will rather be on cost per performance (GB/s bandwidth, IOPS) than on cost per capacity.

Similar to the ML use case also for open-loop MiL/SiL testing at scale a high bandwidth parallel HPC file system is needed to provide data streams of several GB/s per compute node in such clusters. Compared to the ML use case the focus is still on the performance, but more on the bandwidth than on IOPS because the applications usually read large files with several GB of size sequentially. Also the capacity of the file system for open-loop MiL/SiL testing is an important factor. Assuming an average throughput of 3 GB/s per compute node a single node will run through almost 260 TB per day. A high density compute rack with 40 of such compute nodes will consume more than 10 PB per day. This means while pure SSD storage will fulfil the performance requirements it will currently still be too expensive compared to a hybrid SSD/HDD storage backend.

For data analytics but also data transformation often applications based on Big Data frameworks like Hadoop and Spark are used. These frameworks are optimized to work with filesystems like HDFS or compatible implementations provided by different vendors. Again the focus will be on bandwidth and capacity, but depending on the framework used a converged solution with storage inside the compute nodes will be advantageous because the frameworks are aware about the data location and will try to schedule the compute load to the node with the data.

Finally for use cases like engineering workplaces that may run in a mixed Windows and Linux environment and require low latency for interactive use, modern Filers providing combined CIFS and NFS access may be the best solution.

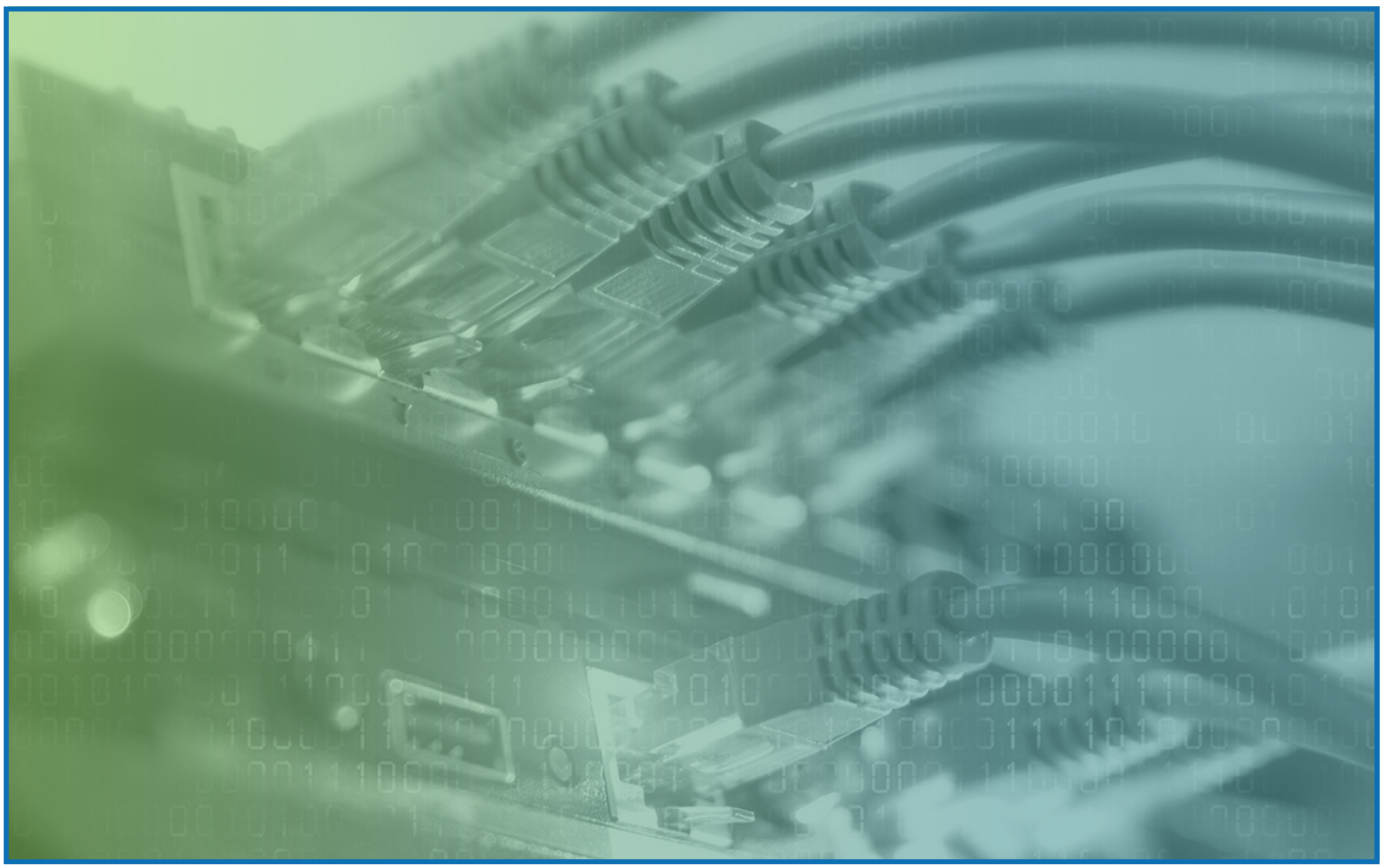
5.2 Infrastructure requirements for the OpenADx Toolchain components (Continued)

Req2: Interconnect and accelerators (network/computation focus)

The different use cases also have different requirements on the interconnect between compute nodes and the availability of accelerators. To be able to scale out Machine Learning with a high bandwidth, low latency interconnect between the nodes is required and the compute nodes use accelerators like GPUs for the computation. To achieve the best bandwidth and latency performance the interconnect should provide hardware accelerated Remote Direct Memory Access (RDMA) capabilities directly from the compute accelerators. An example for this is the use of Mellanox Infiniband with Nvidia GPUs as accelerators in the compute nodes. Using Ethernet network equipment with RDMA over Converged Ethernet (RoCE) capabilities would be an alternative to Infiniband.

For the other use cases, Ethernet with the appropriate bandwidth (10GE - 100GE) is the most common choice. For HPC cluster like setups for open-loop Mil/SiL testing and data analytics, care must be taken with the cluster network design to avoid congestion e.g. by using a super-spine/spine/leaf architecture. Advanced security features or hardware accelerated overlay networking may also be important if this is a multi-tenancy environment.

Besides the ML training use case the open-loop Mil/SiL use case will use accelerators in the compute nodes. While for the ML training usually the highest performing accelerator is the best choice for the MiL/SiL testing the focus will be more on cost and power consumption because the compute requirements per node are less than for the ML training. Based on the models and software modules to be tested, service and cloud providers can provide guidance for the choice of the configuration (CPU, accelerator, networking) of such nodes.



5.3 Hosting a reference software architecture for an AD toolchain

A toolchain consists of various components and therefore has a set of mixed requirements. There is a need for an independent platform that merges all these individual components, which suggests a hybrid cloud scenario.

Moreover Toolchain components must comply with a standard data exchange which sets the baseline for hybrid cloud communication.

Toolchain component Requirement	Architecture definition tools	Data Ingest/ Store	ML training	Simulation based validation	Build	Integrate	Analytics
High Storage demand	No	Yes	Yes	Yes	Maybe	Maybe	Yes
Compatibility with toolchain	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Data Gravity	No	Yes	Yes	Yes	No	Maybe	Maybe
Hardware scalability and availability	No	No	Yes	Yes	No	Yes	Yes
Flexible computing demand	No	No	Yes	Yes	No	Yes	Yes
Hardware evergreen	No	No	Yes	Yes	No	Yes	No
Fast data ingest	No	Yes	Yes	Yes	No	No	Yes
CI/CD	Yes	No	Yes	Yes	Yes	Yes	Yes
AI services (e.g. Machine Learning, AI-Frameworks - TensorFlow)	Yes (parameter tuning for architectural blocks)	No	Yes (parameter tuning for models)	No	No	No	No
Security	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Compliance (e.g. TISAX)	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Suggested Infrastructure	Private cloud	Hybrid cloud	Public/hybrid cloud	Public/hybrid cloud	Public cloud	Hybrid cloud	Public cloud

5.4 Open Data Platform for AD (DRAIVE)

One of the main challenges in AD software engineering is the "need for data" challenge (see section 2 - Challenges). This challenge will create in the near future new approaches and methodologies that will help to acquire, collect, annotate, convert, and provide massive amounts of data in a uniform and open manner. Currently, OpenADx does not cover data collection, unification, and vendor-neutral provisioning. However, as OpenADx seeks to integrate available tools and approaches, such tools will radically improve the data availability for new AD applications.

Open AI data framework such as the one proposed by the DRAIVE[1] project, will make data-sets and catalogs openly available to encourage companies to share and exchange data. The automotive industry has to participate in R&D efforts in which future-proof technologies for Autonomous Driving (AD) are looked after and developed and where corresponding standards are about to emerge with the need to consider the specific European and national regulatory and legal constraints, i.e., in terms of data protection. This justifies an international collaborative approach. An open common data platform for the cooperative exchange of AI training data would accelerate the process of training AI-based systems for AD applications and thereby the entire self-driving industry. In particular, an open common AI data platform would remove significant barriers for AI-based startups as it could provide an abundance of annotated high-quality data.

The need for data makes the DRAIVE open data platform and architecture necessary, and extends the objectives that are put forward by OpenADx and the Eclipse ecosystem for open automotive solutions. The proposed architecture of the DRAIVE data platform is described in the following figure. The aim is to specify an AI data framework, including the implementation of data models, interfaces, services, and software components, enabling data sharing and knowledge transfer between participants for AI applications in the autonomous driving domain. DRAIVE proposes a decentralized architecture with ad-hoc data processing and exchange that focuses on standardized interoperability and enabling of AI supported applications. The specified data model needs to be flexible, as it covers data from different sources and domains, and needs to be expandable according to further needs of the participants.

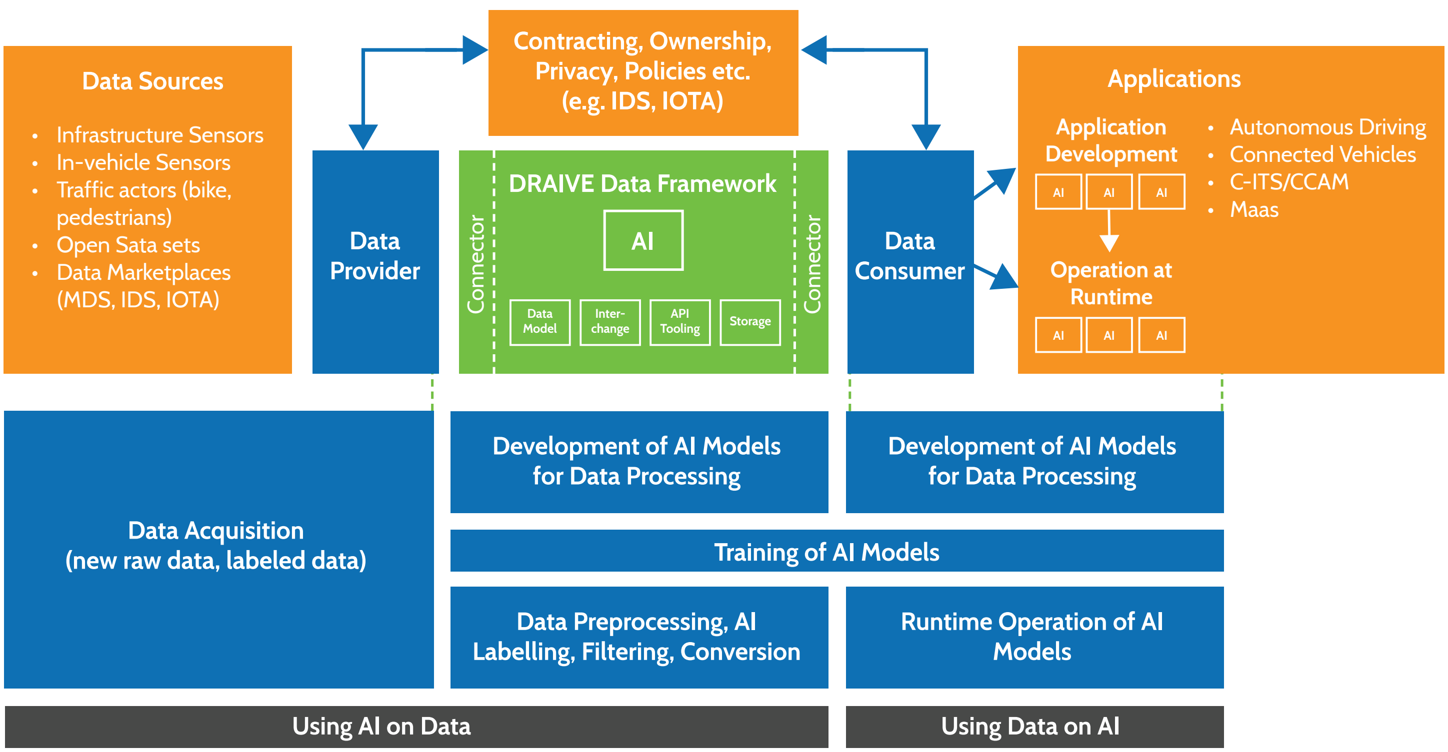


Figure 1: Technological value chain (TVC) for Open AD Data

[1] DRAIVE is a project for AD proposed to https://eureka-clusters-ai.eu/,which is a common call for the EUREKA clusters such as ITEA 3, for a common cross domain interest in developing, adapting and utilising emerging Artificial Intelligence within and across their focus areas.

6. How to get involved in OpenADx? (Call for participation)

Who we are and How to join?

Join our public website[3] to gain an understanding of who is involved and how to interact with us.

How to participate in OpenADx?

OpenADx is an early stage (June 25, 2019) official open Eclipse Working Group, which consists of more than 10 member companies, who are shaping the Working Group, which is done in the Steering Committee.

Feel free to use our mailing list to start a discussion or to be part of one, to get answers to your questions around OpenADx and of course to stay tuned! You can use the solutions we provide in compliance with the open source license of the specific product.

From time to time, there will also be some workshops, which will be announced via our website, wiki and mailing list. So feel free to participate!

If you would like to engage more, e.g. to shape the working group together with us, you have to fulfil some prerequisites:

- You / your company has to be at least an Eclipse Solution Member
- You / your company has to sign the OpenADx Working Group Participation Agreement and to decide your current membership level (explained in the Working Group Charter)

Our recommendation is "Driver Member" to have the biggest possible influence in the working group.

For further questions, you can use our mailing list.

You are invited to shape with us the future of OpenADx!

7. Outlook / Roadmap

Currently we have the following projects:

Eclipse Cloe

Eclipse Cloe[7] is an enhanced middleware solution for closed-loop simulations, with a focus on functional software tests on system level. Cloe is used as a development tool for interactive and scripted workstation simulations, for code debugging, and for automated tests on servers. Cloe covers a central part of the verification and validation strategy, needed by all vendors in the field of Automated Driving.
Cloe acts as closed-loop simulation master and middleware for all involved components, such as simulator engine, vehicle controller, and component models.

- Cloe provides a tool for orchestrating such closed-loop simulations in a variety of environments, e.g. on localhost or in a Kubernetes cluster.
- Cloe provides a lightweight web-based user-interface for visualizing the simulation state.
- Simulations are described through Cloe stack-files for reproducibility.

These core characteristics of Cloe allow it to (partially) fulfil the closed-loop simulation in the V&V strategy. Vehicle controllers and models can be easily integrated with Cloe and immediately have access to multiple simulator engines. Given M simulator engines and N vehicle controllers, Cloe allows the number of integrations to be reduced to M+N, rather than M×N that would otherwise be necessary. Multiple simulator engines are supported, proprietary as well as open source. This allows a user to mix-and-match engines to their system-under-test based on test requirements. Cloe provides generalized interfaces allowing ground truth and key-performance indicator extraction, as well as fault injection.

Eclipse iceoryx

In domains like automotive, robotics or gaming, a huge amount of data must be transferred between different parts of the system. If these parts are actually different processes on a POSIX based operating system like Linux, this huge amount of data has to be transferred via an inter-process-communication (IPC) mechanism. The mechanisms provided by the operating system, like message queues, normally require to copy the data and are subject to context switches between user and kernel space. In fact, it is often not only one but many copies of the transferred data between the sender and the receiver. If it comes to GBytes/s data exchange rates, avoiding these copies becomes a crucial factor. Eclipse iceoryx[8] is a middleware with a zero-copy shared memory approach which is optimized for the huge data inter-process-communication.

Eclipse Cyclone DDS

Eclipse Cyclone DDS[9] is an implementation of the OMG Data Distribution Service (DDS) specification and the related specifications for interoperability.

With 'DATA' being 'the currency of the IOT', having a proper data-sharing technology will be a key-asset in any IOT-platform. The OMG DDS standard is recognized as a highly applicable standard for reliable and robust data-sharing in business- and mission-critical environments and with that a great fit with the Eclipse-IOT ecosystem.

Eclipse Cyclone DDS offers unique data-sharing capabilities compared to the already existing Eclipse solutions (i.e. for messaging).

Its data-centric architecture (where data has RDBMS-like structure and relations rather than 'blobs' as typically used in lower level messaging products) combined with the capability to 'annotate' the data(-models) with fine-grained QoS properties (for reliability, urgency, persistency etc.) provide unrivalled functional and non-functional properties for time- and mission/business-critical IOT-systems.

With a history of proven applicability in the military C4I domain (that are not so much unlike upcoming large-scale IOT-systems where you can also distinguish between planning, awareness and execution-grids that all impose specific requirements with regards to interoperability, high-volume data-sharing and deterministic/real-time data-access) we are confident that this offering can significantly reduce the complexity of many IOT-systems and with that reduce their integration efforts and risks.

Furthermore we are confident that the Eclipse-IOT ecosystem will benefit even more by active participation in the evolution of this technology and the (OMG-)standard its based upon.

Watch out for further projects on the OpenADx website.[12]

7. Outlook / Roadmap (Continued)

Cooperations

We are working to improve the automotive area by seeking collaboration between all the Eclipse automotive working groups. We began with a joint workshop to locate common use cases for openMDM and OpenADx. Further collaborations will follow.

What is ahead of us?

OpenADx left the incubation phase on July 1, 2020 and entered normal operation.

Next steps will be to build a program plan for OpenADx for having a planning basement and to derive and steer work packages for the next steps.

We will also intensify the collaboration with other working groups, e.g. find synergies in workshops with other working groups, not only related to automotive working groups. Other packages will be building demonstrators to show the benefit of OpenADx.

8. Disclaimer

OpenADx working group will not complete safety qualification or verification for the provided projects. For this we invite commercial vendors to incorporate the software provided by OpenADx and complete the required certification process for their customers. If you have some safety requirements, we can support you with these needs.

9. References & Further Reading

[1] Press Release Globe Newswire about founding OpenADx Working Group: https://www.globenewswire.com/news-release/2019/08/27/1906959/0/en/OpenADx-Becomes-Eclipse-Working-Group-Focused-on-Interoperability-in-the-Autonomous-Driving-Software-Stack.html

[2] Press Release Bloomberg about founding OpenADx Working Group:
https://www.bloomberg.com/press-releases/2019-08-27/openadx-becomes-eclipse-working-group-focused-on-interoperability-in-the-autonomous-driving-software-stack

[3] OpenADx Web Page:
https://openadx.eclipse.org/

[4] OpenADx Wiki:
https://wiki.eclipse.org/OpenADx

[5] About the Eclipse Foundation:
https://www.eclipse.org/org/

[6] Members on OpenADx Web Page:
https://openadx.eclipse.org/members/

[7] Eclipse Cloe Project page
https://projects.eclipse.org/projects/technology.cloe

[8] Eclipse iceoryx Project page
https://projects.eclipse.org/projects/technology.iceoryx

[9] Eclipse Cyclone DDS Project page
https://projects.eclipse.org/projects/iot.cyclonedds

[10] OMG Data Distribution Service (DDS) specification
http://www.omg.org/spec/DDS/

[11] OMG Data Distribution Service (DDS) specifications for interoperability
http://www.omg.org/spec/DDSI-RTPS/

[12] Projects on OpenADx website
https://openadx.eclipse.org/projects/

[13] ASAM (Association for Standardization of Automation and Measuring Systems)
https://www.asam.net

Learn more:

openadx.eclipse.org/resources/OpenADx-Manifesto-v07-2020.pdf

openadx.eclipse.org

eclipse.org/org