WHITE PAPER

arm

SOAFEE SIG, One Year Later

Matt Spencer
Technical Director and Distinguished Engineer, Arm

SOAFEE SIG, One Year Later

arm

ABSTRACT

A year ago, the SOAFEE SIG released a whitepaper outlining the vision of the Technical Steering Committee (TSC) and how we wanted to bring the Automotive community together to solve complexities of deployment of the future Software Defined Vehicle (SDV).

This paper is a retrospective on what we have learnt to date and will outline the work we will undertake over the coming year.

Contents

Introduction

The SOAFEE SIG was launched in September 2021 with eight initial founding members from across the industry.

FIG. 1
SOAFEE Founding Members

The common goal of these founding members was to explore the use of cloud-native technologies for vehicles on the road. Explore the tools and methodologies widely used in the infrastructure world today to see how we can apply them to help mitigate the increase in software complexity that SDV's bring. Check out the blog post from Dennis Laudick, VP, Automotive Go-to-Market at Arm the Four pillars to achieve the software defined vehicle.

Since launch the SOAFEE SIG has been rapidly onboarding industry leaders to help with the journey. At the time of writing, the SOAFEE membership stands at 86 with key partners from across OEM's, Tier1's, OSV, ISV and Research Institutes to ensure we have the best minds in the industry collaborating to solve the fundamental, non-differentiating technical foundations for successful SDV's.

Achievements to Date

Since the posting of the last SOAFEE whitepaper, the community have been hard at work building the foundations to enable SOAFEE to succeed.

SOAFEE Architecture V1.0 Release

In order for the SOAFEE community to have a clear idea of our starting point, together we have released the SOAFEE Architecture Specification v1.0. This specification documents existing cloud native technologies and standards that are used today, without specific consideration of the full set of automotive software requirements but is important because it gives OSV's the opportunity to understand which aspects of today's cloud-native solutions are relevant to the SOAFEE architecture.

This specification has allowed third party OSV's to build out capabilities that enable portability of simple SDV workloads today and enable the community to showcase the value of cloud-native methodologies in the concept of SDV. The specification as it stands today is not capable of deploying safety or real-time services, but this is the primary reason for SOAFEE to exist—to understand the gaps between cloud-native as it is expressed today with the needs of modern safety relevant systems required in the automotive domain.

The architecture will be iterated over the coming period to add these missing capabilities.

SOAFEE Integration Lab

Under the governance of the Implementation Working Group, we have launched the SOAFEE Integration Lab with our partner Linaro. The purpose of the lab is to run continual integration of the key components of the SOAFEE architecture along with industry defined workloads.

The team are working on the SOAFEE Test Suite that is used by the integration lab to validate the reference implementation as a part of the CI workflow. This test suite can also be used by SOAFEE partners to validate their implementation of the SOAFEE architecture.

The lab has the capability of hosting multiple hardware platforms either locally, or as remote runners, enabling us to test multiple combinations of hardware with SOAFEE compliant middleware and automotive workloads.

The expectation is that over time, the SOAFEE Test Suite will become the foundation of a SOAFEE Compliance Test Suite that will enable independent validation of member implementations of the SOAFEE architecture to be SOAFEE Compliant.

Red Hat Remote Integration Lab

Once the SOAFEE Integration Lab was set up, we started looking to SOAFEE members to set up remote runners to showcase their solutions. Red Hat have set up a remote lab that works with the same test suites and blueprints as used by the upstream lab to prove that their implementation of the SOAFEE architecture is able to deploy the same SDV features as the SOAFEE reference implementation.

This remote integration enables partners to work on the differentiating and commercial offerings they bring to the market and ensure interoperability of SDV capabilities across the SOAFEE ecosystem.

Open AD Kit Blueprint Launched

Early in the year, we launched a successful collaboration with the Autoware Foundation with the launch of Open AD Kit. This project is a re-imagining of the Autoware Autonomous Drive (AD) solution as microservices architecture to enable the SOAFEE architecture to deploy the services to the correct part of the system based on the runtime requirements of the service.

This is a big step towards enabling a complex deployment such as an AD solution to be deployed to a complex system architecture without the need to re-engineer the AD functionality to target specific features of the target system.

This is the starting point for the SOAFEE architecture, as in its initial form it only makes use of homogeneous compute available in the target system. But it now gives us a platform to explore a move towards heterogeneous compute systems that more closely represent our target deployed vehicle architectures.

Heterogeneous Compute with Open AD Kit

Arm, in collaboration with SOAFEE members Kernkonzept, NXP and ZettaScale took the homogeneous compute Open AD Kit blueprint and made it more safety relevant by taking one of the microservices from the original Open AD Kit blueprint and migrating it to an NXP S32Z Real-Time processor. This processor makes use of the Arm Cortex-R52 split-lock capable core which is able to achieve ASIL D safety certification.

The team took the actuation module, the part that is responsible for controlling the steering, braking and acceleration of the vehicle and ported it to an RTOS for deployment on the Cortex-R52. The rest of the functionality of the AD solution was running on Cortex-A processors, making this a true heterogeneous deployment. Further, the Open AD Kit components were wrapped into virtual machines on both processors, showing both the isolation

capabilities of the underlying L4Re Hypervisor and L4Re Micro Hypervisor and the flexibility of the system architecture to host and consolidate diverse work loads.

The purpose of this demonstration is to show that by correct segmentation of functionality in your application through use of cloud-native technologies like microservices and virtualization, we are able to deploy the services to the compute entity that meets the functional and non-functional requirements of the system.

You can check out a video of the demonstration at Embedded World 23.

AWS Iceberg Blueprint

The Iceberg Blueprint for SOAFEE was designed with a clear mission: to eliminate the burden of repeated security work, standardization, certifications such as OCI and Parsec, and SystemReady compliance. By providing a comprehensive and validated architecture, developers and ISV partners can streamline their development process, ensuring security and compliance requirements are met without redundant efforts. This enables developers to focus on their core strength of innovation, allowing them to introduce their services in a marketplace-like environment for automakers to utilize or subscribe to.

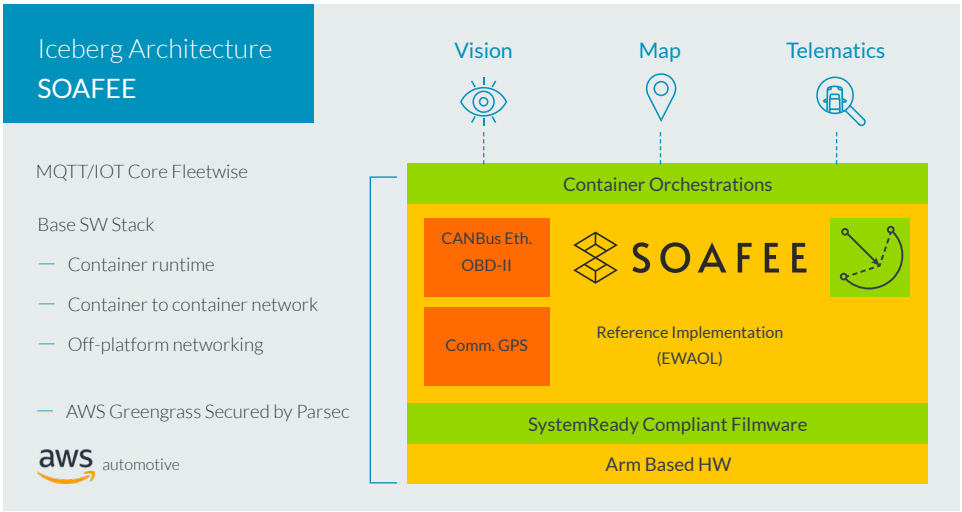

FIG. 2
Iceberg Architecture

At AWS, our customer-centric approach drives us to prioritize the needs of our customers. The SOAFEE blueprint aligns perfectly with this approach by enabling OEMs to seamlessly integrate certified containers from various solution providers, such as usage-based insurance or machine learning models for preventive maintenance, into their SDVs. This empowers OEMs to drive innovation and create customized solutions that cater to their specific requirements, while maintaining a high level of security and compliance.

For instance, the reference design showcased at Linaro Connect includes containers like vision (e.g., in-cabin solutions), map services, and TCU services for driver scoring or maintenance tracking. These options provide OEMs with the flexibility to choose and leverage the containers that best suit their needs, promoting innovation and enhancing the capabilities of their SDVs.

Autoware Foundation with eSync Alliance

At CES 2023, the Autoware Foundation showcased their AD solution alongside the secure Over the Air (OTA) framework from the eSync Alliance. Both of these components were deployed to the EWAOL, the reference implementation of the SOAFEE architecture. Details can be found in this blog post from eSync Alliance.

The combined solution of AD and OTA starts to prove the commercial value of SDV, as the SOAFEE architecture enabled two discrete vehicle functions coming from two separate organizations working in harmony due to the cloud-native approach to SDV that SOAFEE is building on.

With this clear service defined abstraction, it means that OEM's will be able to choose to implement in-house capabilities that are differentiating to their products, or easily integrate non-differentiating functionality from their preferred ISV to shorten the time to market. With the SOAFEE architecture taking ownership of many aspects of the complexity of integration and deployment.

Next Steps

The SOAFEE community is now turning its attention to how we iterate the foundational SOAFEE architecture in order to solve specific complexity problems for SDV's. We are achieving this through the expression of blueprints that define SDV features and gives us a framework for gap analysis with the current SOAFEE architecture to understand where we need to extend current standards and methodologies in use in modern cloud-native workflows to solve deployment complexities for SDV's.

Evolution of the Open AD Kit Blueprint

Our first step is to understand how we manage the lifecycle of heterogeneous workloads in the context of SDV. We are using the Open AD Kit proof-of-concept that was showcased at Embedded World 2023 as the definition of the blueprint. Early decomposition of this can be found in the Open AD Kit v2 definition.

By analyzing the deployment of the demonstration, we can understand the gaps in current capabilities for deploying a heterogeneous implementation of Open AD Kit in a production environment. The working groups of the SOAFEE SIG will be working to understand these gaps, and working with the relevant standards bodies and tools vendors to build a maintainable DevOps process that supports the blueprint.

The output of this work will be the ability to initially build, test and deploy SDV workloads to both Cortex-A and Cortex-R targets. Once this constrained problem is understood, we will start to investigate other processing entities such as Cortex-M, GPU, NPU etc.

This work will also form the foundations for understanding how we test, validate and deploy real-time and functional safety requirements in vehicle. It will require collaboration across all disciplines in the DevOps workflow but will result in a manageable delivery pipeline for SDV solutions.

Integration of Arm Virtual Hardware (AVH)

Part of the solution space for the Open AD Kit blueprint will include the ability to test real-time applications in the cloud where no physical real-time processors exist. This could make use of virtual prototyping technology from Arm known as Arm Virtual Hardware and through the SOAFEE analysis of the blueprint requirements, we can ensure that the tooling delivered into the ecosystem meets the demanding requirements of the SDV vision.

The aim for the SOAFEE community and Arm is to make these technologies easily available in cloud-based offerings to help with the shift left testing that the market requires in order to deliver high quality code to market on time and in budget.

Safety Compiler Integration

To achieve safety certification, software will need to be compiled using a compiler that is safety certified. Arm deliver the Arm Compiler for Embedded FuSa that enables the commercial software ecosystem to deliver fully qualified FuSa solutions.

The Arm compiler is LLVM based, so the first step for the SOAFEE community is to ensure that the open-source assets such as EWAOL, the able to be compiled with an LLVM based compiler such as Clang.

Digital Cluster and IVI

The SOAFEE blueprint team are working to build a definition of a digital cluster and IVI solution that makes use of the SOAFEE architecture. This blueprint has yet to be released but will be ready soon.

The aim of this blueprint is to understand the IO and accelerator needs for Cluster and IVI use-cases in order to define the underlying portability and security layers that will enable easy deployments of these business-critical workloads.

Move Towards Commercial Consumption

As the SOAFEE architecture matures, we need to understand what a go-to-market strategy looks like. The SOAFEE value proposition focuses around:

- maximize code reuse
- reduce the time to market
- constraining deployment complexity

These three areas, when combined, help to reduce the cost of delivering and maintaining software into an SDV context. The technologies, methodologies and best practices that SOAFEE is adopting and enhancing from the cloud-native world are aiming to deliver on these areas.

The SOAFEE community will be building more capabilities into the integration lab that will help to prove the vision of reduced complexity. This is achieved by showcasing commercially supported SOAFEE compliant middleware running on Automotive grade processors with industry relevant use-cases as defined by blueprints.

Conclusion

The SOAFEE community has made fantastic progress over the past 18 months putting the foundations in place for future SDV's showcasing practical implementation of SDV using modern software methodologies. We are now at the point where we move towards resolving some of the domain specific needs of SDV, such as real-time and safety requirements and what software portability means in the context of these requirements.

If you are interested in understanding more, or want to join the SOAFEE SIG, you can find more information at soafee.io or via SOAFEE gitlab.