

Compute Platform - Project Proposal

(Wind River's modified text → draft for official charter)

Compute Platform Project (CPP) Background

For a few years, GENIVI→COVESA has had a project named Hypervisor Project. The primary activity was promoting and writing a common Automotive Virtual Platform Specification, which defines a hypervisor-neutral, automotive-tailored, virtualized computer platform. The goal of that project was to lift the automotive industry to a higher level of understanding of virtualization solutions.

The name *Hypervisor Project* was not ideal as it led to some believing that a new and unique hypervisor implementation was being developed as opposed to the reality of unifying standards for multiple existing hypervisor implementations. The project ran with this name anyway under the assumption that it is the content that matters more than the name.

With the new mandate of COVESA, it is now time to take a broader approach to automotive compute platform architecture that considers diverse integrated platform scenarios including multi-OS, multi-computing, mixed criticality, multi-partition systems, and complex heterogeneous multicore SoCs - in addition to virtualization. We propose the creation of the new COVESA project as below.

Compute Platform Project Overview

The diversity of design ideas for compute-platforms relevant to automotive systems has increased. We see some promoting the use of a single Linux kernel that uses containers for separation of concerns, whereas others mandate virtualization as a critical technology for safe and secure mixed-criticality computing. In recent times, architects propose independent payloads each running their own kernel without necessarily being considered virtual machines. The term compute-island crops up, and appropriate integration of Trusted Execution Environment and specialized computing cores remains another important part. Such designs communicate between independently booted kernels running on various parts of the SoC, and some propose to forsake a hypervisor completely. Comparing these approaches with different names is important for understanding, but in many cases a combination of them is likely to be used.

The purpose of the COVESA Compute Platform Project is to share knowledge, align on terminology, analyze choices, and create recommendations and standards for when to use different design methods and patterns. As we define it, the Compute Platform is the hardware/firmware/software combination that lies near the boundary between hardware and software. Higher-level software components in a software stack (e.g., ROS2) tend to be built on abstractions that already exist and be less concerned with the exact design of the platform below, so those can largely be kept out of the scope of this project.

Optimizing the Compute Platform will require understanding of features built into modern SoC hardware, the use of hypervisors and virtualization where it is appropriate, and the low-level communication methods between computing units. Cloud servers are based on general-purpose Intel/AMD/Arm server CPUs and are for the purposes of this discussion not a challenge to be solved. In contrast, embedded SoCs are more diverse and continue to build in increasingly unique hardware blocks, and new features explicitly for separation of mixed-criticality computing tasks. This means the project shall focus primarily on in-vehicle compute platforms. At the same time, in future outlooks we tend to reduce the distinction between vehicle-computers and cloud-computers, and when special-purpose hardware (e.g., AI/neural-net optimized companion cores) is involved in the cloud-computing part of the full system architecture, then it cannot be avoided to also have those parts in scope for the Compute Platform project.

Why is COVESA the right place for this discussion?

Just like before, individual software-platform organizations such as AUTOSAR do a significant job in defining core computing technology. Part of that work includes tying their proposed technologies to an abstraction of the underlying hardware. AUTOSAR is nowadays complemented by several recent open software development initiatives and company-launched "platform" projects. It is popular for these to be in the direction of creating fundamental software technology for the future "Software Defined Vehicle". Some of them tend to be more focused on the higher-level software stack rather than the near hardware challenges and will thus require some other project to perform analysis, optimization, and definition of common abstractions for the compute platform.

As described above, there is a significant diversity in technologies to build a complete vehicle system. The trend of consolidation into fewer, more powerful computing units also continues. Therefore, this diversity is not between different computers but may appear in various parts of a single System-on-Chip. Just as a few years ago when GENIVI refocused on being a "multi-OS organization", this reflects the current state of the automotive industry. For better or worse, the industry has a diversity of proposed operating systems and middleware, and some projects focus primarily on the higher-level parts of the software stack. A fundamental challenge remains to optimize and improve the understanding of platform-neutral compute platforms and the system assembly process of creating dependable multi-element, integrated systems from multiple sources and providers.

COVESA continues to be the organization of choice for open and independent alignment of core communication and integration technologies and standards that are required to produce the end-to-end connected automotive systems independent of the choice of middleware-technology, operating system, and hardware. The COVESA Compute Platform Project will continue the work of the previous Hypervisor Project and its Automotive Virtual Platform Specification definition to widen the scope of discussion to all necessary compute platform aspects.

In addition to new position papers, white papers, recommendations and proposed standards for compute-platforms, this project expects further stewardship of the [Automotive Virtual Platform Specification \(AVPS\)](#) to be developed together with collaborating partners.

Original text / draft for blog.

For a few years, GENIVI→COVESA has had a project named Hypervisor Project. The primary activity was promoting and writing a common Automotive Virtual Platform Specification, which essentially defines a hypervisor vendor neutral, automotive-tailored, virtualization compute platform, and to lift the industry to a higher level of understanding of virtualization solutions in automotive. The name Hypervisor Project was not ideal, and have even led to some believing that a new and unique hypervisor implementation was being developed (as opposed to the reality of unifying standards for multiple existing hypervisor implementations). The project ran with this name anyway under the assumption that it is the content that matters more than the name. It is now time to take the opportunity to define a new project that better describes the activities and to add on recent trends and a broader view of the computing environment. We propose the creation of the:

Compute Platform Project (CPP)

The diversity of design ideas for compute-platform has increased. We see some promoting the use of a single Linux kernel that uses containers for separation of concerns, whereas others mandate virtualization as a critical technology for safe and secure mixed-criticality computing. In recent times, some design proposals propose independent parts each running their own kernel without necessarily being considered virtual machines. The term compute-island crops up, and appropriate integration of Trusted Execution Environment and specialized computing cores remains another important part. Such designs communicate between independently booted kernels running on different parts of the SoC, and some propose to forsake a hypervisor completely. Comparing these approaches with different names is important for understanding, but in many cases a combination of them is likely to be used.

The purpose of the COVESA Compute Platform Project is to share knowledge, analyze choices, and create recommendations and standards for when to use different design methods. As we define it, the Compute Platform is the hardware/firmware/software combination that lies near the boundary between hardware and software. Higher-level software components in a software stack tend to be built on abstractions that already exist and be less concerned with the exact design of the platform below, so those can largely be kept out of the scope of this project.

Optimizing the Compute Platform will require understanding of features built into modern SoC hardware, the use of hypervisors and virtualization where it is appropriate, and the appropriate low-level communication 'methods between computing units. Cloud servers are based on' general-purpose Intel/AMD /Arm server CPUs and are for the purposes of this discussion not a challenge to be solved. In contrast, embedded SoCs are more diverse and continue to build in more and more unique hardware blocks, and new features explicitly for separation of mixed-criticality computing tasks. This means the project shall focus primarily on the embedded compute platform. At the same time, in future outlooks we tend to reduce the distinction between vehicle-computers and cloud-computers, and when special-purpose hardware (e.g. AI/neural-net optimized companion cores) is involved in the cloud-computing part of the full system architecture, then it cannot be avoided to have also those parts in scope for the Compute Platform project.

Why is COVESA the right place for this discussion?

Just like before, individual software-platform organizations such as AUTOSAR do a significant job in defining the core computing technology. Part of that work includes tying their proposed technologies to an abstraction of the underlying hardware (e.g. MCAL for the Classic Platform). The industry is also seeing several recent open software development initiatives and company-launched "platform" projects. It is popular for these to be in the direction of creating fundamental software technology for the future "Software Defined Vehicle". Some of them tend to be more focused on the higher level software stack rather than the near-hardware challenges, and will thus require some other project to perform analysis, optimization, and possibly definition of common abstractions for the compute platform.

As described above, there is a significant diversity in technologies to build a complete vehicle system. The trend of consolidation into fewer, more powerful computing units also continues. Therefore this diversity is not between different computers but may appear in different parts of a single System-on-Chip. Just as a few years ago when GENIVI refocused on being a "multi-OS organization", this is a reflection of the current state of the automotive industry. For better or worse, the industry has a diversity of proposed operating systems and middleware, and some projects focus primarily on the higher level parts of the software stack. A fundamental challenge remains to optimize and improve the understanding of platform-neutral compute platforms.

COVESA continues to be the organization of choice for open and independent alignment of core communication/integration technologies and standards that are required to produce the end-to-end connected automotive systems independent of the choice of middleware-technology, operating-system and hardware. The COVESA Compute Platform Project will continue the work of the previous Hypervisor Project and its Automotive Virtual Platform Specification definition to widen the scope of discussion to all necessary compute platform aspects.

In addition to new position papers, white papers, recommendations and proposed standards for compute-platforms, this project expects further stewardship of the Automotive Virtual Platform Specification (AVPS) to be developed together with collaborating partners.