Whitepaper: Going the extra mile

A #digitalfirst approach to implementing travel range extensions for Battery-Electric Vehicles, utilizing COVESA's Vehicle Signal Specification (VSS) and digital.auto SDV

Hyundai Mobis: Pradeep Kumar K

Bosch / Ferdinand-Steinbeis-Institute: Dirk Slama

ANSYS: Gopinath Penamalli; Hemesh Patil; Tushar Sambharam

Ferdinand-Steinbeis-Institute: Chris Cheng

Content

Problem statement ... 2
Parameters affecting travel range ... 2
Optimization potential and general approach ... 3
Target architecture: SDV + vehicle APIs ... 4
#digitalfirst development approach ... 4
Step 1: Virtual prototyping ... 5
Approach ... 5
SDV app: digital.auto playground ... 6
Behind the APIs: Hardware virtualization and digital twin simulation ... 6
Step 2: From virtual hardware to real hardware ... 7
Hardware setup ... 7
Target architecture ... 8
Summary: Integrating digital.auto and Ansys via COVESA VSS ... 8
Conclusions & benefits ... 9

Problem statement

For drivers of Battery Electric Vehicles (BEVs), the fear of running out of power on a journey and not being able to find a charging point on time is a common concern, also referred to as range anxiety.

Of course, the effective navigation solution integrated with the BEV`s power management is a key element in addressing range anxiety. However, the world is not always perfect, and not every travel route, including breaks for charging, is working out as planned. Traffic jams or issues with charge point availability can cause unforeseen problems.

This is why the COVESA EV Power Optimization project is developing a safeguard solution that is constantly monitoring the BEV state of charge and taking appropriate measures in case of a range issue – for example, by automatically (or semi-automatically) powering down less important energy consumers in the car.

The idea of this project is that the BEVs will behave more like a smartphone, which offers users a “power save” mode if battery charge levels are dropping below a certain point. Similarly, our project will enable drivers of BEVs to “go the extra mile” to reach their next destination.

Parameters affecting travel range

There are a number of components on modern BEVs affecting driving range, including the power train, chassis, electronic systems, networking, safety, control, infotainment and comfort, and control. Range calculations are complex, and also depend on external factors such as driving style, vehicle loading weight, and weather conditions.

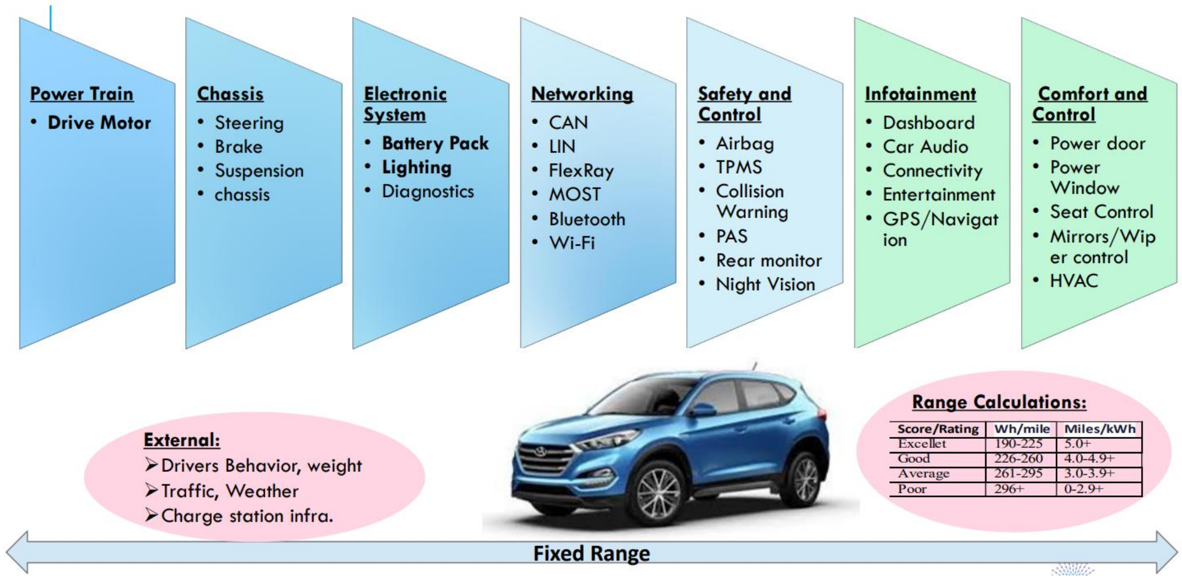

Figure 1: Parameters affecting travel range

Optimization potential and general approach

While not all the components in the BEV can be easily included in the power savings mode (e.g., power train), there are a number of natural candidates, as shown in the figure below. The potential for range extension differs. For example, HVAC at 10-15% power savings is an obvious candidate, but also windows, rooftop, lighting, and IVI are interesting possibilities. Together, the potential is close to 25% power savings for non-driving related energy consumers.

The general approach proposed here is to monitor the state of charge of the BEV, then implement a power savings policy which will power down non-critical energy consumers as the state of charge goes down. For example, a dip in the state of charge to 20% and below could be used to trigger a driver alert, followed by a powering down of non-critical components.

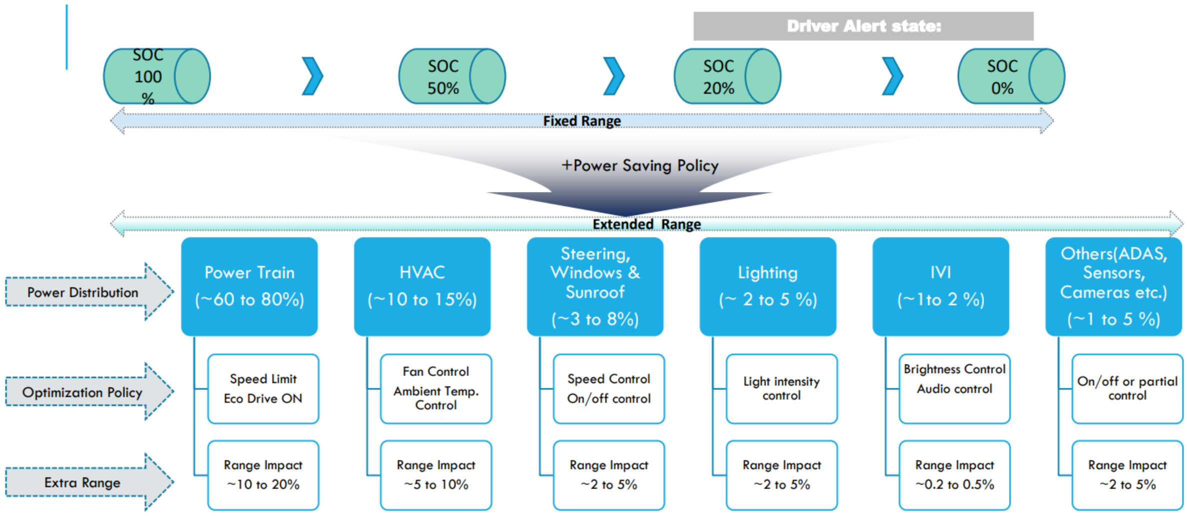

Figure 2: Optimization potential and general approach

Target architecture: SDV + vehicle APIs

For this power savings algorithm to be embedded in the overall vehicle architecture, software-defined vehicle (SDV) and the user of vehicle APIs are proposed. The SDV architecture shown in the figure below is a common target for many original equipment manufacturers (OEMs). It utilizes industry standards like the COVESA Vehicle Signal Specification (VSS) to provide an abstraction layer of the vehicle`s E/E architecture and make vehicle sensors and actuators available via well-defined APIs to higher-level software algorithms, such as our power savings algorithm.

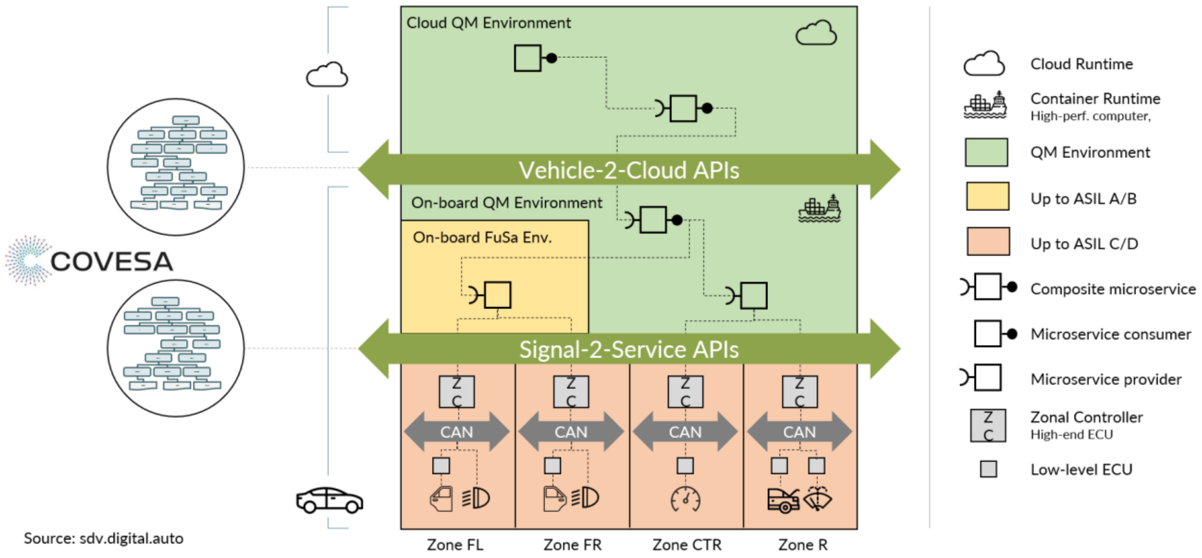

Figure 3: Target architecture: SDV + Vehicle APIs

#digitalfirst development approach

In this project, we are following the #digitalfirst approach developed by the digital.auto initiative. Taking a #digitalfirst approach means that we first focus on the higher-level algorithms implemented using SDV with simulated vehicle APIs in the backend.

Once the new SDV algorithms are stable and the API requirements have been finalized, we go into hardware integration. This way, any new API requirements are identified as early as possible, giving the OEM sufficient time to map these requirements to the hardware development side. For hardware tests, an open test suite like the digital.auto dreamPACK can be utilized, eventually leading to an automotive-grade, production-ready design.

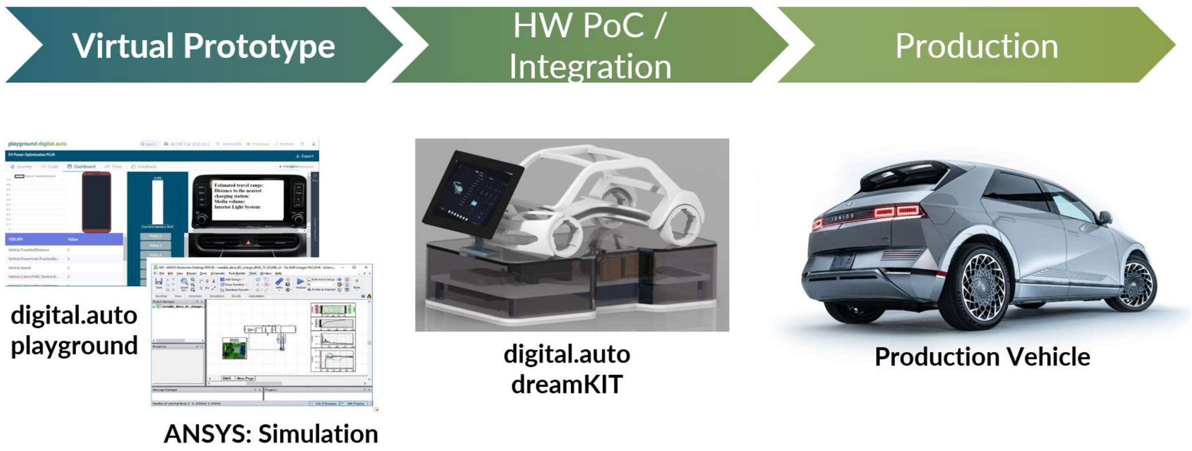

Figure 4: #digitalfirst development approach

Step 1: Virtual prototyping

The first step in this is virtual prototyping, which utilized rapid prototyping tools such as the open-source digital.auto playground for implementing new SDV algorithms against the COVESA Vehicle Signal Specification.

Approach

As shown in the architecture below, the new power savings algorithm runs in the digital.auto playground and is accessing the BEV Digital Twin via standardized COVESA Vehicle Signal Specification. In this case, the BEV power consumption system twin “south” of the APIs is provided by ANSYS (see below).

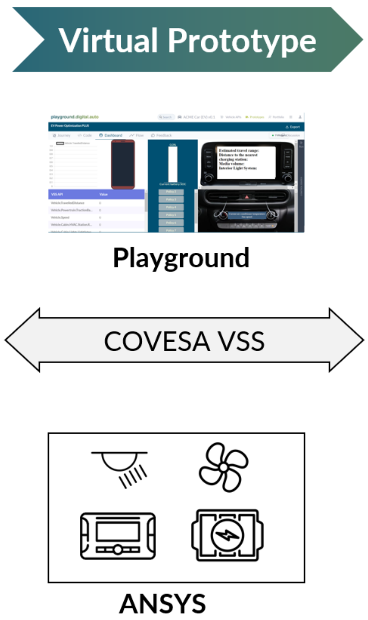

Figure 5: Virtual Prototyping

SDV app: digital.auto playground

The resulting power savings algorithm can easily be accessed via web browser, which shows the power savings application including vehicle state of charge, and a visualization of critical components such as HVAC and IVI. Taking data from the simulation backend, this test system shows how HVAC and IVI are gradually turned down, depending on the vehicle state of charge.

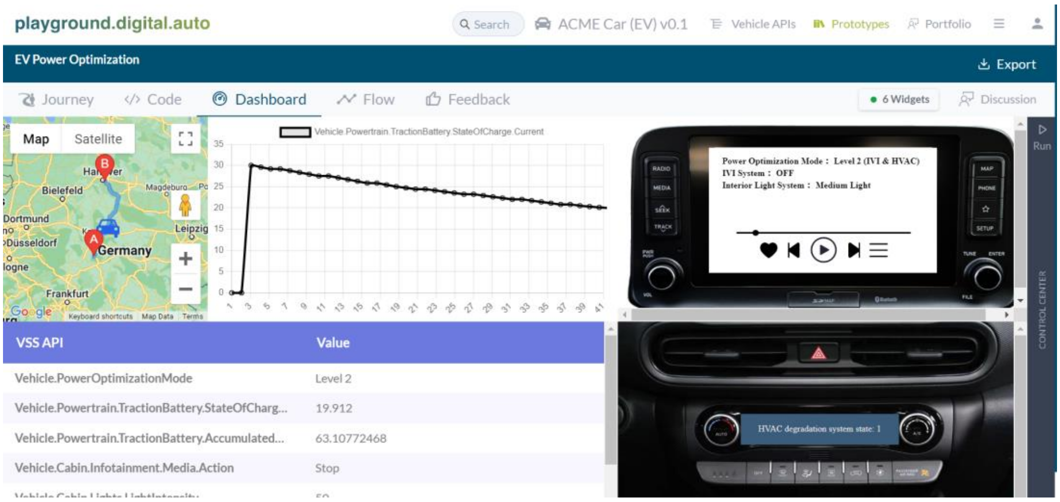

Figure 6: SDV Demo application in playground.digital.auto

Behind the APIs: Hardware virtualization and digital twin simulation

Hardware (HW) Virtualization is an essential part of SDV realization as it will enable the software to define hardware specifications suitable for SDV upfront.

In our case the BEV power consuming systems like motor, HVAC, IVI, and lighting all are created as individual models, and integrated together with the battery model to generate a system of systems model that replicates the BEV. Ansys Twin Builder enabled the quick creation and integration of these system models that could later be deployed as a digital twin.

Complex EV systems like motor, battery and HVAC are represented as reduced order models (ROMs), which are abstract models of a high-fidelity computational model. These ROMS can preserve essential behavior and dominant effects while significantly reducing the solution time and computational storage capacity required, without compromising the accuracy of the system's behavior.

Twin Builder was then coupled with Ansys' physics-based simulation technologies to bring the detail of 3D simulations, as ROMs, into the system's context to generate accurate and efficient system-level models. To this end, Twin Builder uses ROMs produced from Ansys structural, fluids, electromagnetics, and semiconductor products to model mechanical assemblies; electromagnetic actuators and machines; circuit and cable parasitic; thermal networks; and signal integrity.

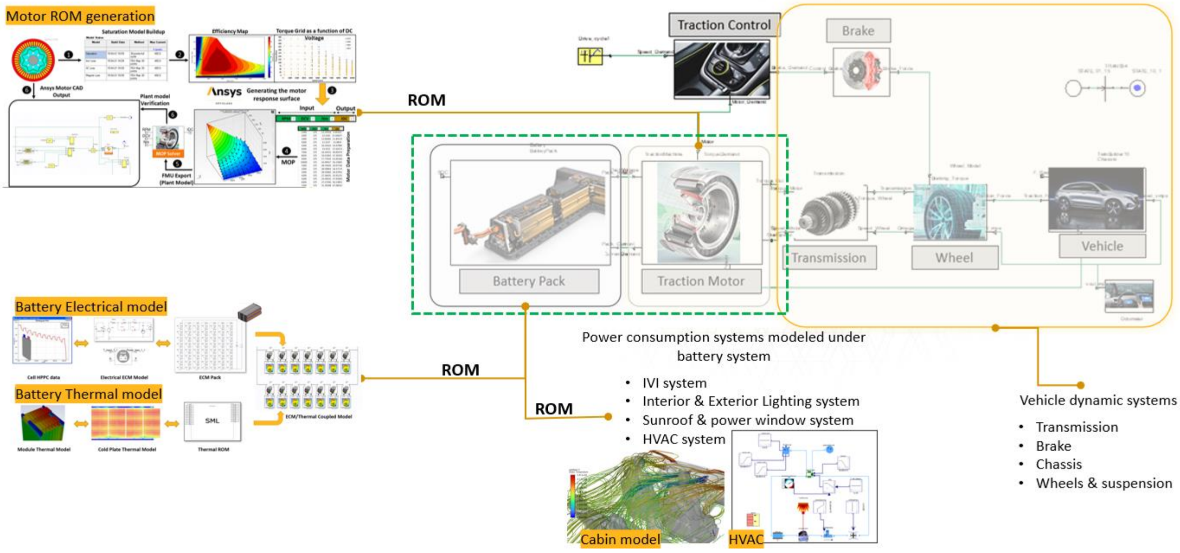

Figure 7: EV Virtualization with Ansys Twin Builder & ROM technology

Ansys Digital Twin can be deployed as a containerized app in any IIoT platform, or as a custom web application. These containers and apps can be linked to any asset through a standard API, utilizing the COVESA Vehicle Signal Specification. These Twins can be integrated to rapid prototyping platforms like digital.auto, and can provide real-time outputs and commands, prompting a much-faster-than-real-time response. This dynamic interaction ensures that Ansys Digital Twin remains aligned with the latest conditions and requirements, effectively bridging the gap between virtual simulations and real-world applications.

Step 2: From virtual hardware to real hardware

The next step is the integration with real hardware, eventually leading to an automotive-grade system design.

Hardware setup

In our case, the test hardware is based on the digital.auto dreamKIT, which is an open-source design for SDV test hardware. Prototypes from the open-source playground can be automatically deployed to the dreamKIT.

Figure 8: Hardware Proof-of-Concept

Target architecture

The target architecture is shown in the diagram below. In the integrated target architecture, all simulated components are replaced with real hardware, for example HVAC and IVI. Only the BEV battery is still simulated (because it is too large and hence not practical for such a demo).

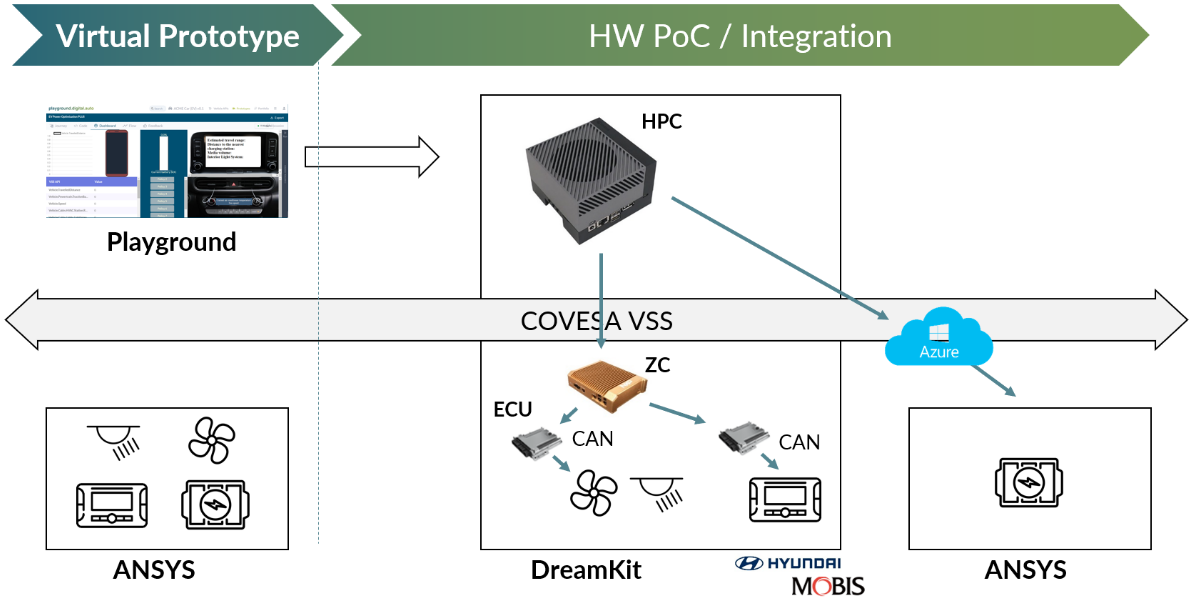

Figure 9: Target Architecture

Summary: Integrating digital.auto and Ansys via COVESA VSS

In our case Ansys' BEV digital twin model is deployed in the Digital.auto SDV platform. This digital twin communicates with the EV power optimization algorithms hosted within the platform via COVESA VSS. This results in a well-orchestrated flow of VSS signal information to and from which enables policy decision making.

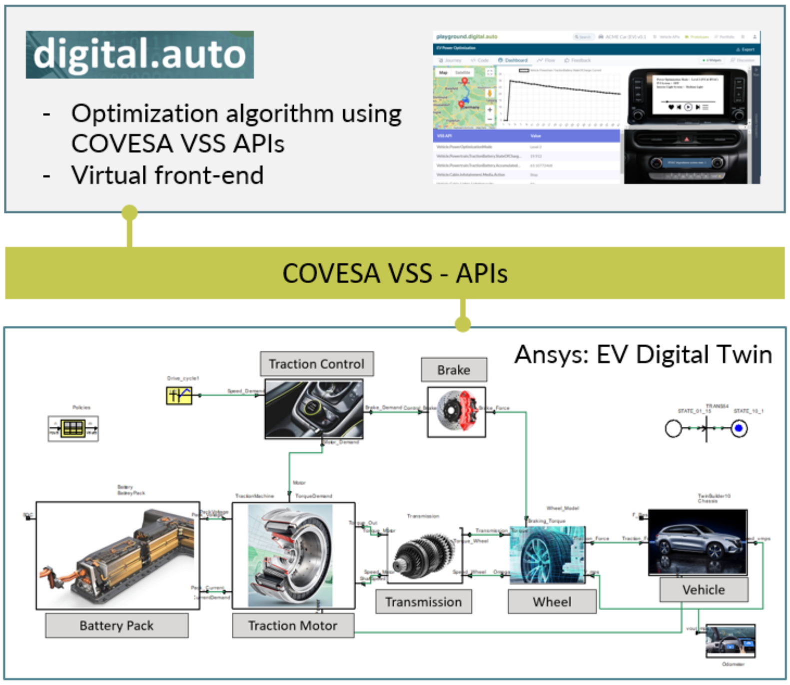

Figure 10: Integration with COVESA VSS & digital.auto

Our use case demonstrates the shift left philosophy with rapid prototyping (much faster than real time) of SDV SW applications, while adhering to industry standards.

Conclusions & benefits

We started with the goal to goal to realize and digitally validate new power optimization policies and algorithms to minimize the last mile anxiety of BEV drivers – a "power save" mode for BEVs like we are used from smart phones.

The #digitalfirst approach applied in this project should give us early validation of the interfaces required into the BEV hardware (using COVESA VSS) and feasibility of the algorithm.

Our solution is starting with the digital.auto playground to implement the power savings algorithm against standards COVESA vehicle APIs. In order to get a realistic virtual test environment without having to build up a test vehicle, we used the Ansys Twin Builder to create a physics and data based high fidelity EV power consumption system digital twin, helping us to validate power optimization the algorithms and policies.

The digital.auto playground for the actual power savings application was integrated with the Ansys Digital Twin via COVESA VSS or successful data exchange between digital twin and the development environment. We used on-the-fly synchronization with the digital.auto front end to share the feedback of vehicle parameters like state of charge (SoC) and range.

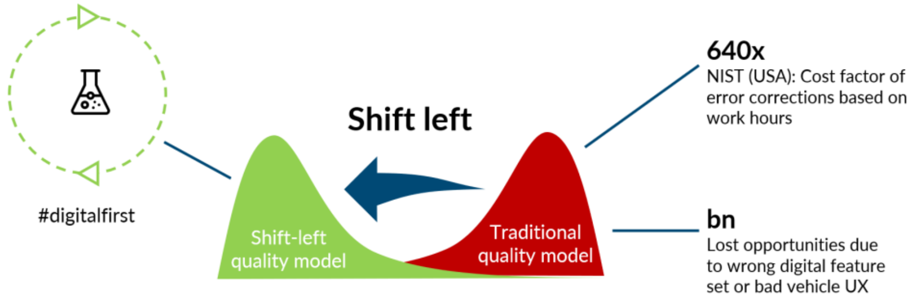

Figure 11: Benefits of digital.auto shift left approach

The benefits of this approach are manyfold. The digital.auto playground allowed use of real-world COVESA vehicle APIs and integrate everything in a cloud-native environment, supporting on-the-fly optimization. The vehicle virtualization using Digital Twin Simulation (based on Ansys Twin Builder for this unique application) is a key driver in making the SDV philosophy a reality. Adding the digital.auto dreamKIT was the next step in the evaluation of the feasibility, adding automotive-grade hardware test kit.

Overall, the #digitalfirst prototyping approach led to rapid verification and validation of the power optimization software algorithms – a manyfold improvement over the costly and time-consuming build-up of a physical test vehicle.