2024 NDIA MICHIGAN CHAPTER
GROUND VEHICLE SYSTEMS ENGINEERING
AND TECHNOLOGY SYMPOSIUM
DIGITAL ENGINEERING / SYSTEMS ENGINEERING TECHNICAL SESSION
AUGUST 13-15, 2024 - NOVI, MICHIGAN

APPLYING DIGITAL ENGINEERING DIGITAL TWIN TO SUPPORT GROUND VEHICLE VIRTUAL EXPERIMENTATION

Robert J. Kanon[1], Kevin W. Griffin[1], Raveen Fernando[2], Amir Shah[2], Russ Kouba[3], Mark Feury[3]

SAIC[1], Sterling Heights, MI
Applied Intuition[2], Detroit, MI
DEVCOM GVSC[3], Warren, MI

ABSTRACT

A digital twin is a virtual model that accurately imitates a physical asset. This can be as complex as an entire vehicle, a subsystem, and down to a small functioning component. The digital twin has a level of fidelity that aligns to the goals of the project team. The usage of a digital twin inside a digital engineering (DE) ecosystem permits architecture and design decisions for optimized product behavior, performance, and interactions. This paper demonstrates a methodology to incorporate the digital twin concept from requirement analysis, low fidelity feature level simulation, rapid prototypes running inside a System Integration Lab, and high fidelity virtual prototypes executing in an entirely virtual environment.

Citation: R. J. Kanon, K. W. Griffin, R. Fernando, A. Shah, R. Kouba, M. Feury, "Applying Digital Engineering Digital Twin To Support Vehicle Virtual Experimentation," *In Proceedings of the Ground Vehicle Systems Engineering and Technology Symposium* (GVSETS), NDIA, Novi, MI, Aug. 13-15, 2024.

1. INTRODUCTION

A digital twin is a virtual model that represents an intended or actual physical system, subsystem, or component. Many benefits evolve from digital twins, including improved program management, quicker and more accurate risk assessment and mitigation, and the ability to make better decisions earlier in the development lifecycle. Another valuable deliverable from a digital twin approach is the capability to perform virtual experimentation which enables rapid iterations through concept designs, testing various scenarios, and the ability to identify and tackle issues earlier, thus saving time and costs that would normally occur well after the first physical

---

prototype hardware/software are produced much further in the product lifecycle.

We have developed the Ground Vehicle Rapid Prototype (GVRP) project in collaboration with DEVCOM Ground Vehicle Systems Center (GVSC) with the intention of demonstrating virtual experimentation through a digital twin approach and ultimately informing various ongoing DE activities and pilot projects within GVSC and the Virtual Prototyping (VP) S&T project. The planned processes, lessons learned, and observed results are described in the next sections of this whitepaper.

2. REQUIREMENTS ANALYSIS

We started the GVRP project through development of a set of requirements for a family of ground vehicles ranging in size from a class 3 ultralight through a class 8 medium tactical transport vehicle. Requirements were captured as stakeholder needs and baselined in Doors Next Generation (DNG). These requirements were synchronized to the digital engineering ecosystem, and imported into our Cameo system modeling tool. Requirements analysis started with use case analysis, tracing of requirements to use cases, derivation of system requirements, and identification of functionality via features. System model artifacts, attributes, and relationships were all captured in the system model environment and synchronized to the digital engineering ecosystem. Data and information captured from the stakeholder requirements, derived requirements, and system modeling tasks are now all available and accessible to downstream engineering disciplines to utilize in the development of specialized modeling, analysis, simulation, and virtual testing and experimentation.

3. DIGITAL TWIN AND TESTING FRAMEWORK

A digital twin is a dynamic virtual representation that goes beyond traditional 3D models. It encompasses a comprehensive data model that includes the physical asset it mirrors—such as a ground vehicle, its software, hardware, computer, and physical properties, but also the operating environment within which the asset functions. This concept, therefore, extends to creating a digital twin of the world or test environment, alongside the digital representation of the vehicle. This approach enhances the predictive capabilities and the scope of simulations that can be performed, providing a more holistic view of how the production ground vehicle would perform under various real-world conditions.

Establishing a ground vehicle program from its inception through deployment of the initial physical prototype necessitates incremental validation and testing at each phase. Real-world testing incurs significant expenses, both in terms of time and resources, limiting the number of test cycles a program can execute to achieve incremental improvements on the design of the vehicle based on mission-oriented performance data. A more effective strategy involves employing extensive, low-cost synthetic testing early on, progressively focusing on more detailed, high-cost evaluations.

Figure 1 depicts the steps needed to develop a virtual simulation to support vehicle experiments. We apply these steps to support the development of four Hybrid Electric Transport Vehicle (HETV) design configurations to evaluate key performance metrics and vehicle characteristics.

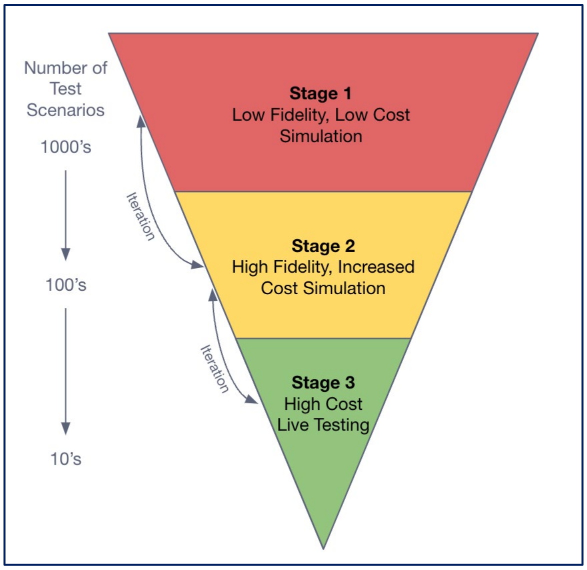


Figure 1 – Funnel depicting low-cost large-scale testing leading to high-cost scale testing

Prioritizing the extensive testing of systems and subsystems within cost-effective virtual environments enables the early identification of bugs, architectural flaws, and gives a program insight into system performance in advance of hardware procurement or buildup. This approach reduces the time and cost associated with modifications once prototypes are constructed. Virtual environments facilitate the validation of system and subsystem functionality. For example, they allow for the assessment of vehicle dynamics and performance across a broad range of criteria. For vehicles incorporating automated systems, virtual simulations provide a preliminary platform to evaluate the system-level performance of the software stack before real-world implementation.

Subsystems benefit from execution within these virtual settings as well, where preliminary feature simulations and interface testing can be conducted using model-based system engineering tools. Additionally, the development of autonomy and crew reduction subsystems, including sensor selection, behavior verification, model refinement, and training for human-machine collaboration, can be efficiently managed in simulated environments.

A program can strategically progress from low-level, cost-effective testing to advanced sensor in the loop or hardware-in-the-loop (HIL) environments, enabling the program to understand more nuanced tradeoffs between architectural design decisions like required engine torque or optimal sensor layout. Initially focusing on simple component tests allows for quick, inexpensive assessment of solutions and resolution of issues at the developer level. Progressing to higher fidelity testing facilitates interaction checks between software and sensors in a simulated environment, both catching discrepancies early and allowing the trade space between integrated systems to be fully explored. The final move to HIL testing, where real hardware is tested under controlled conditions, ensures comprehensive evaluation of integrated systems. This tiered approach prioritizes early problem discovery, minimizing later-stage costly corrections and focuses higher-cost testing on refining and validating system integration, thereby enhancing development efficiency and product reliability.

The process of HIL testing also incorporates physical components into the test environment. A HIL setup can be equipped with a user station interface and both simulated and physical modules, facilitating advanced validation of human-machine interactions in a synthetic environment, allowing for rapid iteration and validation.

Ultimately, this methodical approach significantly reduces the volume of tests required on actual prototype vehicles, thanks to the exhaustive evaluations conducted during the preceding stages. This not only accelerates the development timeline but also enhances the reliability and performance of the final product, thereby exemplifying a strategic, cost-effective methodology in ground vehicle program development.

4. LOW FIDELITY FEATURE SIMULATION

As features are identified via the requirements analysis process, they are examined further to understand safety concerns, failure mode analysis, and a deeper understanding of the complexity. These features move into a feature model phase, where the functionality is developed further in a low fidelity simulation within the system modeling environment. This approach mitigates risk, and ensures requirements are complete and well-understood.

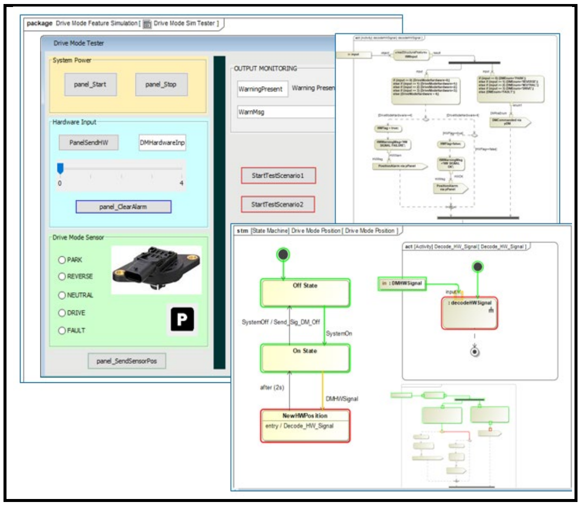


Figure 2 – Feature Simulation

Through simulation at the feature level, issues and defects within the requirements are identified, and this leads to the creation of additional system level, performance and timing requirements that often get overlooked early in the development process. An executable simulation drives a common understanding to all stakeholders and serves as an initial digital twin for key features.

5. CONCEPT DEVELOPMENT

With the digital engineering ecosystem, we have the ability to create data along the development process that becomes available for our partners to use, from requirements and system model data to simulation and analysis results. This information is created and maintained in a single authoritative source of truth, and then synchronized to the digital environment so that our team members and project stakeholders can access the latest information in a format they can utilize. This data sharing provides traceability between the upstream requirements analysis and system modeling with concept design, trade studies, simulations, and other artifacts important to the overall project. Digital threads are created and maintained throughout the process. This results in partners working in their own specialty tools and having access to the latest information to provide seamless engineering collaboration.

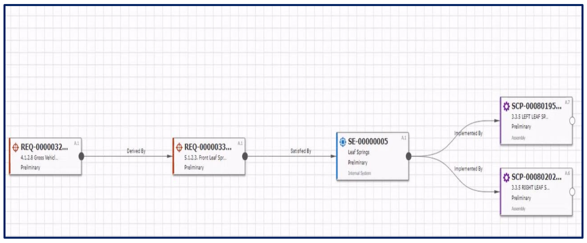


Figure 3 – Digital Thread view from requirements to derived requirements to system elements to CAD Parts

6. RAPID PROTOTYPING

As the concept designs evolve and decision-making is performed, we take the digital twin simulations through a rapid prototyping process which results in auto-generated code from models executing in a target environment, such as a System Integration Laboratory. Additional valuable insight is developed through this software factory process which permits visualization of hardware and software running within the laboratory environment. Benefits of code generation to a PC environment, then to a real-time embedded operating system, provide opportunities for early testing and verification.

Key artifacts are captured, traced to other project artifacts, and synchronized to the digital engineering ecosystem, ensuring configuration management, and accessibility to all team members.

7. VIRTUAL EXPERIMENTATION

7.1. Role of simulation in design and development phases

Figure 4 – Vehicle Driving in Applied Intuition Virtual Environment

The role of virtual experimentation is to be the backbone of the digital ecosystem and maintain a digital thread between concept, requirements, and validation. As expressed in the section 3, digital twin and testing framework, simulation allows for quick and frequent testing of a large breadth of tests in a low-cost environment before committing to high-cost real world tests. This approach allows for design optimization, and finding bugs and integration issues sooner in development before design decisions are finalized at which point changes to the platform become much more challenging.

Simulation allows for vehicle architects and developers to quickly assess different vehicle models, physical configurations, and software configurations in their performance on any given requirement or deployed environment. An example is that of swapping complete vehicle architectures to understand the difference of a tracked vehicle and a wheeled vehicle in any given Operation Design Domain (ODD), be it city streets, highways, forested off-road, desert off-road, or more. A simulated environment allows engineers to conduct virtual trade studies on hardware items like sensors or suspension based on prescribed ODD. During the development phases, this allows for rapid capability improvement using regression testing and automation. Simulation can provide automated bug discovery, which accelerates developer teams in reaching a stable and mature system.

While the simulation framework accelerates development, it remains grounded in open standards that are needed for program validation and verification. Simulated test cases are rooted in the base program requirements. A digital thread is maintained from requirements and specifications to the test cases via a verification matrix. Simulated test plans, concrete test cases, and results are linked back to the program requirements for traceability.

7.2. Virtual Environment Simulation Layers

Virtual experimentation in simulation can be grouped into a few themes: simulation of automated systems, simulation of vehicle architecture, and the fusion of the two. For automated systems, simulation is generally broken into two primary layers; object level simulation and sensor level simulation. For vehicle architecture, there is vehicle dynamics simulation, and subcomponent software in the loop (SIL) simulation. A culmination fusion of the types of simulation occurs in a HIL station that consists of subcomponent simulation, autonomy simulation, and human interfaces all teaming together.

7.3. Object Level Simulation

Figure 5 – Object Level Simulation within Applied Intuition's Object Sim

Object level simulation is primarily used in autonomy applications. This level of autonomy is a low-fidelity environment in which ground truth object level classifications and position of the world state is sent to the vehicle's autonomy stack. This type of simulation bypasses sensing and perception and feeds ground truth information of the world state directly to the autonomy stack motion planner and execution controllers. This allows those controllers to be the system under test, isolated from potential noise and false world interpretations from sensor inputs. Validation of general behaviors around various classes of actors, obstacles, and scenarios can be tested in this environment. An example of this simulation is with Applied Intuitions Object Sim simulation tool.

7.4. Sensor Level Simulation

Sensor level simulation is a high-fidelity physics based simulation environment for testing a vehicle's perception for autonomous applications. A digital twin environment, vehicle, and sensors are created for accurate virtual experimentation. The generated world along with the assets and dynamic actors used in the simulation are physics-based to provide accurate electromagnetic responses as seen by both active and passive sensors such as LIDAR's, radars, or cameras. An example of a digital twin environment is seen at Holly Oaks, a local testing site near SAIC's Michigan location.

Figure 6 – Top: Real Holly Oaks. Bottom: Digital Twin Holly Oaks in Applied Intuition tools

Digital twins of real-world sensors are modeled in simulation to provide the same input to the autonomy stack from the digital twin environment as it would ingest from the real world. As with other aspects of the simulated approach, sensor types, models, and placements on the vehicle can be altered and tested quickly. An example of this type of simulation is with Applied Intuition's combining sensor level simulation with object level simulation, the testing can cover full system evaluation of the autonomy system. This method can then be expanded to full-vehicle system evaluation if combined with vehicle dynamics simulation.

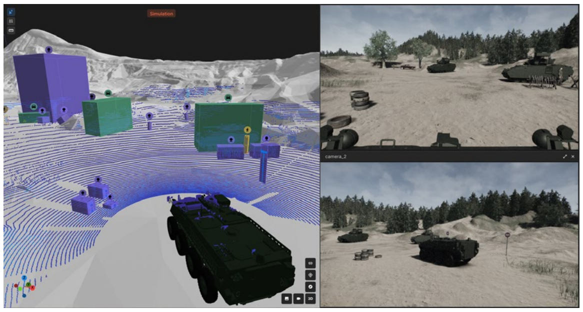


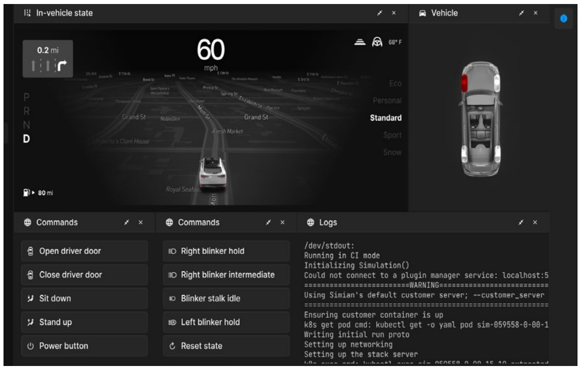
Figure 7 – Sensor Simulation in Applied Intuition's Sensor Sim tool

7.5. Vehicle Dynamics Simulation

Figure 8 – Vehicle Dynamics Simulation

Vehicle dynamics simulation is the area of modeling and testing a digital twin based on vehicle specifications of a real-world vehicle architecture. In the case of developing a new ground vehicle program, this digital twin would be based on the specifications required of the new vehicle and evaluations on performance can be done prior to any real-world vehicle being created. Information such as powertrain configuration, suspension assets, wheeled or track-based, and various vehicle dimensions go into development of the digital twin model. When paired with a digital twin environment that mimics the terrain and material types of the intended ODD or testing environment, virtual experimentation can be performed to assess the vehicle architecture system level dynamics. Examples of vehicle dynamics evaluations in simulation include roll over mitigation, powertrain performance, and performance on various terrains and slopes. Vehicle specifications can be quickly iterated and run in simulation to gain confidence in the vehicle design prior to building any prototype. Additionally, critical test cases can be identified in simulation that should be repeated in real world environments, such as the boundaries between any safe and unsafe behaviors seen in simulation.

7.6. Module & ECU Simulation



```
/dev/stdout:
Running in CI mode
Initializing Simulation()
Could not connect to a plugin manager service: localhost:5
=======================WARNING=========================
Using Simian's default customer server; --customer_server
=========================================================
Ensuring customer container is up
k8s get pod cmd: kubectl get -o yaml pod sim-059558-0-00-1
Writing initial run proto
Setting up networking
Setting up the stack server
```

Figure 9 – Vehicle Dynamics Simulation

Subcomponent SIL testing is a cloud based virtual development and testing environment for onboard vehicle software and ECUs. Virtualized ECUs are able to be used for both unit tests and system integration tests involving the interaction and communication between ECUs, so software functionality can be tested across modules and domains. Engineers and developers can configure, build, deploy virtual vehicles and test end to end functional behavior across an entire vehicle (multiple ECUs) and visualize output of the vehicle. The nature of the interconnectability of the subcomponents, leads to a range of virtualization levels and provides a flexibility of fidelity & scope of testing. This allows for rapid testing between developers and suppliers. The full scale integration of ECUs means continuous integration and continuous development methodologies are retained for faster software delivery and faster validation cycles. With full traceability back to the base vehicle and software requirements, the subcomponent SIL simulation testing leads to rapid vehicle development and establishes a robust, and rapid method of verification as programs progress through developmental phases.

7.7. Crew Station HIL Simulation

Figure 10 – Applied Intuition prototype Crew Station simulator

A hardware-in-the-loop simulation fitted with a crew station is a hybrid format in which a physical driver station is connected to a real-time system with both autonomy and ECU modules in the loop. There is flexibility weighting of virtualized components and real components in a HIL rig. ECUs can either be in the form of physical modules, or in a simulated state. In contrast to earlier simulation phases, the autonomy is not simulated in the same environment as seen in the object level and sensor level simulation. In a HIL implementation the vehicles autonomy and controllers are based in a real-time PC while an external simulator simulates the vehicle dynamics and the digital twin world state. World state sensor information is fed back to the real-time PC in a format that is compatible with the desired sensors on the vehicle platform.

The end product is a human machine interface teaming with a validation set up. Requirements and specifications surrounding how a crew is intended to interact and operate a vehicle equipped with autonomy can be validated in this environment. As with earlier stages, quick iteration when identifying bugs or regressions is possible.

Beyond validation of requirements, a crew station HIL setup becomes a crew trainer, in which crews are trained quickly and safely in how to operate a vehicle equipped with autonomy. The interactions of engaging and disengaging autonomy is something that should be done in an environment that is safe from real world consequences. Once confidence is established within a crew trainer, graduation to a physical vehicle will be a short jump.

8. GVRP OBJECTIVES AND SUMMARY

The DE activities and processes described are being developed and demonstrated in the ongoing GVRP project. The high-level objectives include:

1. Implement Department of Defense DE strategy and address culture and organizational change.
2. Demonstrate a GVRP ecosystem.
3. Develop software production factory ecosystem to support SW application development.
4. Apply DE process and methods to demonstrate a GVRP ecosystem with focus on fuel efficiency improvements in commercial & military environments.

The GVRP project kicked off in 1QFY24 with the period of performance planned over the next 18 months. The project will develop a GVRP ecosystem leveraging commercial tools and data, and culminate in a series of Technical Data Packages (TDPs), model libraries, and demonstrations with GVSC. The activities described and the demonstrations planned directly inform ongoing pilot projects and future DE tools, processes, and planning.

Using a digital twin to support virtual experimentation encourages and promotes verification performed by simulation at each step of the product development. This clearly results in a better quality deliverable along each step of the product development process. This approach focuses on usage of simulation to dynamically execute models within the virtual arena. Low-fidelity and high-fidelity simulations are created representing vehicle electronics, controls, sensors, and physics. An additional benefit is that early assessments of vehicle concepts can be performed which greatly help to develop and test requirements. The early engagement of stakeholders in the product development cycle drives project success, and being able to showcase digital twin-based simulations enable better decision making and risk mitigation strategies. Using a digital twin to represent the physical hardware during vehicle testing results in reduced costs, risk, and physical testing.