Journal of Innovation

August 2024 | 25th Edition

Digital Engineering Enables Innovative Hardware Integration Opportunities in Aerospace

Authors:

Tram Chase
SimVentions, Inc.
tramchase@simventions.com

Traci McDonald
SimVentions, Inc.
tracimcdonald@simventions.com

Matt Wilson
SimVentions, Inc.
mattwilson@simventions.com

CONTENTS

FIGURES

1 INTRODUCTION

Aerospace and avionics systems have strict requirements for reliability and redundancy. However, like all systems development efforts, there are limited budgets and time constraints and a need to deliver increased capability on shorter timelines. One method for addressing this problem is establishing and using open technical architectures and solutions that leverage Digital Engineering (DE) innovations and opportunities.

Many open architecture efforts to enable reuse, extensibility, integration, and open interfaces in the avionics domain have been focused on the software aspect of the avionics system. The desire for open systems and reuse coupled with the difficulty of achieving and maintaining interoperability are just as prevalent in the computing hardware domain as in the software domain.

Simple design questions are very real and currently very difficult to address, such as: What computing chassis or backplanes are available? Which boards fit into which chassis? What form factors are available? Are the electrical and thermal loads appropriate and compatible with the operating environment? Can I replace a board from Vendor A with a board from Vendor B?

Even before addressing those kinds of questions, it is necessary to explore issues like: Can we find boards from multiple vendors that have comparable capabilities and operating requirements? Can smaller form factor boards be used without affecting the mission?

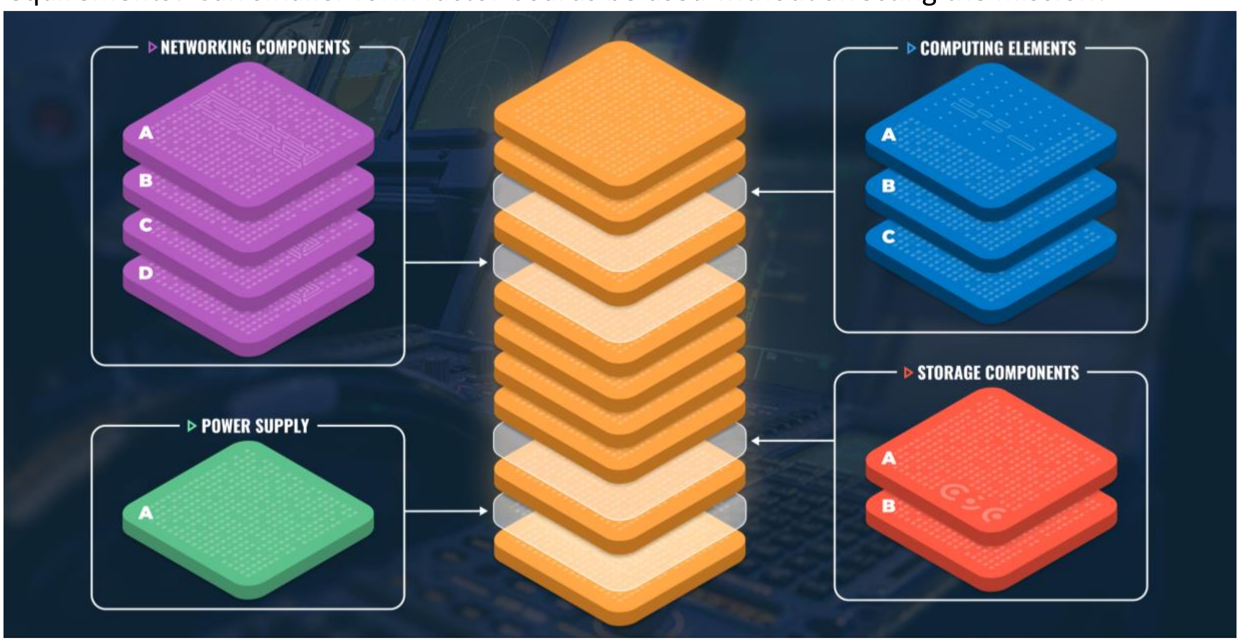

Figure 1-1: Introduction - HHITS tradeoffs for component integration.[1]

[1] To help address these issues, the US Navy helped to establish the Hardware Open Systems Technologies (HOST) standard. See *https://host-oa.com/*

The standard provides a Modular Open Systems Approach (MOSA) as an Open Systems Architecture (OSA) definition of virtual and physical interfaces to hardware components such that interoperability and reuse of hardware components can be realized. HOST intends to establish performance and interface requirements that are consistent, open, enforceable, and testable. But, even with the HOST standard, there are many opportunities to improve and achieve the desired objectives of interoperability and accelerated development.

Current methods for designing and integrating hardware systems often involve the use of homegrown and manually maintained knowledge sets captured in home-grown tools by individual engineers and teams. These methods (spreadsheets, docs, databases, etc.) may be suitable for specific and utilitarian needs; that is, they get the immediate job of adding up thermal and electrical loads, bus bandwidth requirements, et al. They are likely done correctly and efficiently but with limited value for long term maintenance or collateral application. Typically, these tools are limited by the awareness of the engineering teams and are manually populated with published vendor data by the constrained knowledge base of the engineers involved in the selection process. Given the complexity of options, engineers will often settle with the first solution that “works” or “fits” the need.

Often, these “homegrown” knowledge sets are duplicated and diverge amongst teams, thus creating a data reliability issue. After a project moves through the lifecycle of Research and Development (R&D) and testing, these knowledge sets are no longer needed or maintained.

By the time a deployed system is in sustainment mode, the knowledge set is no longer reliable and requires non-trivial effort to update. New boards will have come to market, which may or may not be useful as replacements. Certain hardware vendors may release newer versions of a board with similar model numbers and abilities but differing electrical or thermal requirements.

2 The Opportunity

To conquer these challenges and make both the knowledge sharing and the integration of avionics components easier to use and more cost-effective, the principles of Digital Engineering (DE) must be more thoroughly enabled, developed, and ultimately adopted. DE is an integrated approach that uses digital technologies to transform engineering practices and processes throughout the entire lifecycle of a system.

- DE on the front end helps with the identification, modeling, and design of systems.
- DE during development and production enables a living reference model and data set to adjudicate questions and access design decisions and constraints.
- DE during delivery and sustainment supports maintenance, spares, and logistics.

During the design and engineering phases, the robust use of DE leverages digital twins, simulation, data analytics, and model-based methodologies to enhance the efficiency, quality, and agility of engineering projects. Put another way, instead of relying on document-centric tools and processes like manuals, specification sheets, and application guides, the community needs to define a consistent structure for representing and capturing a component's physical attributes and performance characteristics to enable robust DE modeling and analysis as part of the engineering process.

Similar DE principles are used in many industries. One such example is the architectural industry. DE has significantly transformed that industry by streamlining and enhancing the entire design process through the usage of advanced digital tools and methodologies. With technologies such as Building Information Modeling (BIM) and digital twins, designers can create highly detailed and accurate models that improve visualization, collaboration, and decision-making.

These tools allow for real-time simulations and analyses, enabling early detection of design flaws and optimization of structural integrity and energy efficiency. Furthermore, digital engineering facilitates seamless integration and coordination among various stakeholders, reducing errors, accelerating project timelines, and lowering costs. The adoption of cloud-based platforms ensures that all design data is accessible and up to date to the broadest possible community, fostering a collaborative environment and enhancing productivity across the project lifecycle.

The reach of DE is ever-expanding. While there are valuable uses of DE techniques currently at work in the avionics space, our research has shown that these innovative approaches can be applied in the areas of designing and integrating computing systems in the avionics computer thus delivering similar benefits. A digital representation of the capabilities, compatibilities, and constraints of avionics computing components opens the door to a multitude of applications and benefits.

Pulling together a registry of avionics components based on a standardized digital format would allow vendors to expose products to potential customers, likewise, consumers could easily browse and compare offerings from across the market with homogenous data for each product. Vendors and integrators could publish and autogenerate documentation and specifications with the click of a button based on the data stored in the registry. Visualizations could be created for representing and communicating connectivity diagrams and pinout tables. Intelligent searches could be performed to find the "best fit" or "most affordable" options for replacing components that are no longer produced. Finally, tools that model the integration process could be created that would give real-time feedback to users while assembling a build without having to procure any products.

3 RESEARCH RESULTS

One example of innovative research into DE approaches to increase modeling and integration solutions for avionics computer design is the work done by SimVentions for the US Navy Naval Air Systems Command (NAVAIR). As a result of this effort, SimVentions created the HOST Hardware Integration Toolkit (HHITS). HHITS is a set of interoperable cloud-hosted components that show the value of and opportunities of pulling the DE thread to enable transformative automation, artificial intelligence, and integration for avionics systems.

To address challenges, the following steps were realized:

- Develop a robust but flexible data format to capture the characteristics of avionics systems and the components.
- Create a repository for managing component data.
- Implement visualization methods to represent the hierarchy of components.
- Develop automated decision aids and Artificial Intelligence tools to assist with composition and integration decisions.
- Develop reporting and documentation formats from the standardized data.
- Deploy a robust prototype for community use and exploration.

3.1 Robust and Flexible Data Format

Initial development involved defining a format to capture the characteristics of an avionics computing component. A JavaScript Object Notation (JSON) format was chosen because it is easily parsed by computers and is human-readable enough to facilitate understanding and experimentation during the early stages of development. The goal was to capture details down to the pin-out level so that tools could analyze these definitions and detect incompatibilities during integration. An example of the format for a module is shown in Figure 3-1: Component library for discovery and composition.


```
{
    "name": "SLT3-PAY-2F2U-14.2.3",
    "width": "96",
    "layout": "slotprofile",
    "children": [
        {
            "name": "key1",
            "label": "",
            "type": "key",
            "x": "0",
            "y": "0",
            "height": "1",
            "width": "12"
        },
        {
            "name": "j0",
            "label": "SE\nJ0",
            "type": "utilityplane",
            "x": "0",
            "y": "1",
            "height": "8",
            "width": "12",
            "columns": ["","Row I","Row H","Row G","Row F","Row E","Row D","Row C","Row B","Row A"],
            "signals": [
                "1",   "Vs1", "Vs1",      "Vs1",      "Vs1", "No_Pad*",   "Vs2",      "Vs2",       "Vs2",    "Vs2",
                "2",   "Vs1", "Vs1",      "Vs1",      "Vs1", "No_Pad*",   "Vs2",      "Vs2",       "Vs2",    "Vs2",
                "3",   "Vs3", "Vs3",      "Vs3",      "Vs3", "No_Pad*",   "Vs3",      "Vs3",       "Vs3",    "Vs3",
                "4",   "GND", "SM2",      "SM3",      "GND", "-12V_Aux",  "GND",      "SYSRESET*","NVMRO",   "GND",
                "5",   "GND", "GAP*",     "GA4*",     "GND", "3.3V_Aux",  "GND",      "SM0",       "SM1",    "GND",
                "6",   "GND", "GA3*",     "GA2*",     "GND", "+12V_Aux",  "GND",      "GA1*",      "GA0*",   "GND",
                "7",   "TCK", "GND",      "GND",      "TDO", "TDI",       "GND",      "GND",       "TMS",    "TRST*",
                "8",   "GND", "REF_CLK-", "REF_CLK+", "GND", "GND",       "AUX_CLK-", "AUX_CLK+",  "GND",    "GND"
```


Figure 3-1: Component library for discovery and composition.

Once the data format was established, a repository was built to allow for entering, browsing, searching, and updating component data. Various technologies were evaluated for the repository, and MongoDB was chosen. MongoDB, a NoSQL database, is known for its flexibility and scalability, storing data in JSON-like documents with dynamic schemas. It is designed to handle large volumes of unstructured data and supports a wide range of applications, from simple web apps to complex, data-intensive operations. See Figure 3-2.

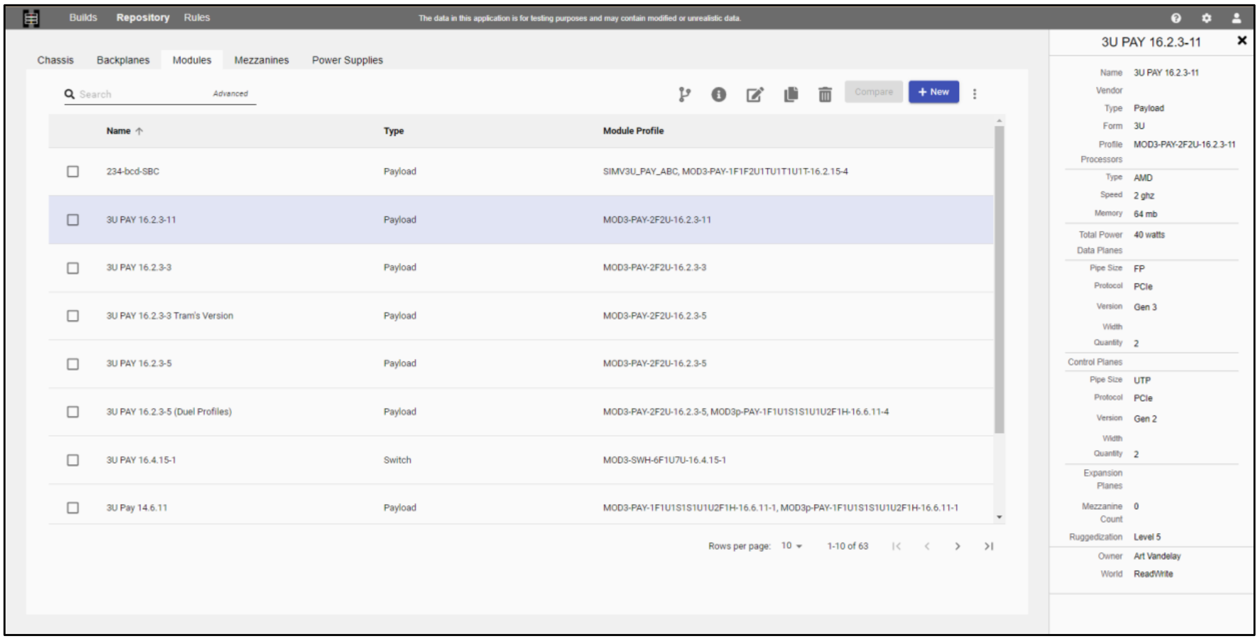

Figure 3-2: HHITS repository.[2]

3.2 Innovative Visualization

With a repository of components established, innovative visualization methods have been developed. These include a tree view to represent the hierarchy of components in a build and a virtual backplane with slots for drag-and-drop module placement. Additionally, dynamic and interactive rendering of module and slot profiles is now possible based on data captured in the newly created format (see Figure 3-3). Together, these advancements create a virtual workbench for integrating avionics computing components.

[2] *https://rockwellcollinsthoughtleadership.wordpress.com/wp-content/uploads/2018/06/avionics-open-system-architecture-standardization-shepherd-and-wills.pdf*

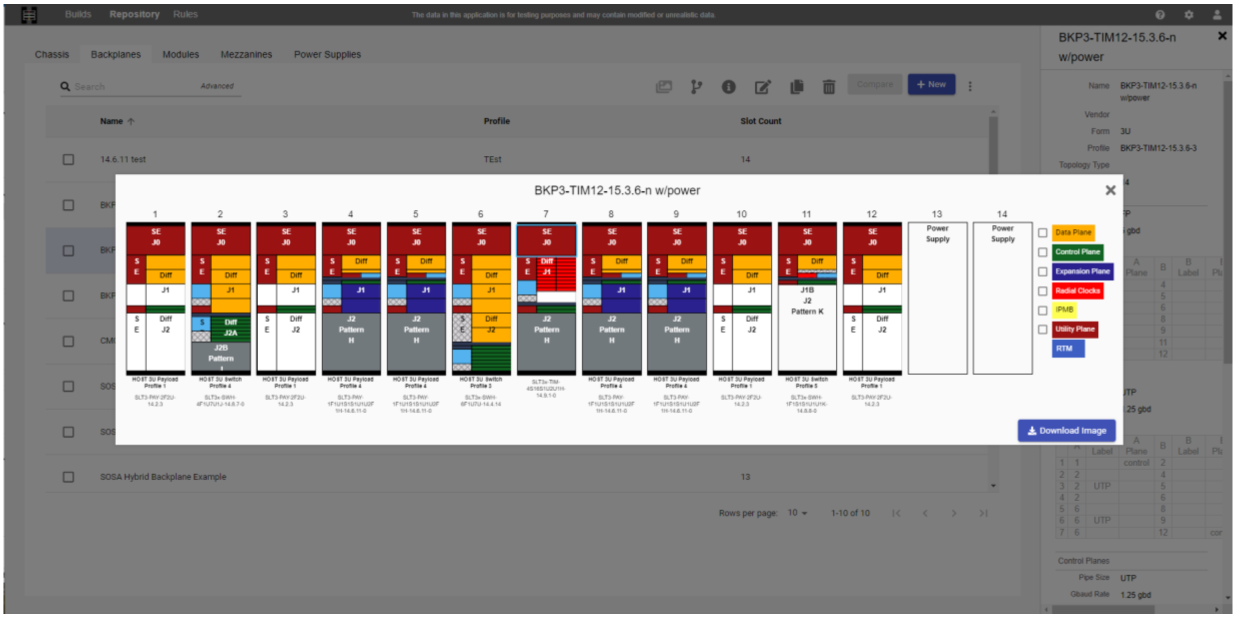

Figure 3-3: HHITS interactive diagram.[3]

3.3 Demonstrate Automated and AI Integration Tools

Finally, with the capability to virtually integrate components and their associated data, comprehensive evaluation of the entire build became possible. Key metrics such as total required power and overall weight can now be calculated. Additionally, build requirements were digitally captured as rules.

For example, requirements like having at least two USB ports or 256 gigabytes of memory are encoded as rules and evaluated against the build using a rule engine. These rules can be created and edited as necessary and are organized into rule sets that can be easily turned on or off during the build process. Figure 3-4 shows a screenshot of the HHITS workbench.

[3] *https://aws.amazon.com/blogs/industries/understanding-digital-engineering-and-how-it-drives-business-value/*

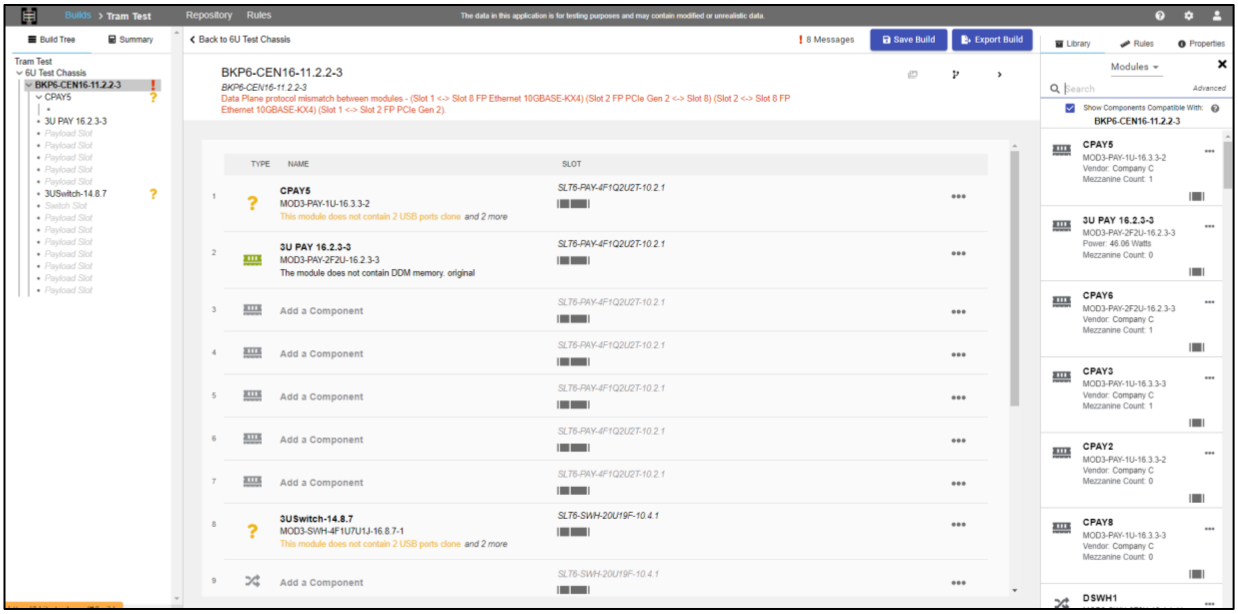

Figure 3-4: HHITS workbench.[4]

To summarize, this preliminary work has resulted in a prototype known as HOST - Hardware Integration Tool Set (HHITS), which was described in the previous section. HHITS is a web-based application that provides the following capabilities.

- **Repository of Hardware Components**: HHITS allows integrators to search and compare hardware components. This facilitates the selection of suitable components for specific systems, ensuring compatibility and reducing the likelihood of integration issues.
- **Interactive Diagrams**: HHITS renders the profiles for backplane slots and modules, allowing users to visualize the input/output connections, pin outs and backplane module connections.
- **Virtual Build and Simulation**: Users can virtually construct a chassis and integrate various hardware components (backplanes, modules, mezzanines, and power supplies) within the tool. By applying predefined rule sets, HHITS identifies and mitigates potential incompatibilities before any physical hardware is procured, saving time and resources.

[4] *https://www.researchgate.net/publication/315736224_An_Examination_of_Open_System_Architectures_for_Avionics_Systems_-_An_Update*

- **Standards Compliance**: The tool supports standards like HOST, SOSA™, and OpenVPX™, ensuring that the integration process adheres to industry best practices and promotes interoperability among different systems and components.
- **Enhanced Collaboration**: By providing a unified platform for information sharing and hardware integration, HHITS enhances collaboration among engineers and system integrators, leading to more efficient and effective integration processes.

4 Conclusion

Our research demonstrates the significant value and potential of advancing HOST data standards in conjunction with MOSA EE concepts, standards, and tooling. This advancement enables a multitude of analytic, economic, and engineering benefits across the avionics ecosystem. In particular, the business benefits of detailed data standards and tools like HHITS are extensive and impactful, promoting comprehensive Digital Engineering practices.

Established vendors can efficiently publish their hardware offerings in a consistent and accurate manner, ensuring clarity and reliability in product information. Buyers can easily find options that meet their specific performance requirements, fostering a competitive market that drives innovation and improves product quality.

Engineers benefit significantly by increasing their speed, accuracy, and confidence through the digital assembly of components before physical prototyping, reducing the risk of errors and accelerating the development process. Ultimately, the community and the public benefit from these advancements through better products delivered faster and at a lower cost, enhancing user satisfaction and contributing to technological progress.

Emerging innovations and standards further support these data-centric engineering capabilities. A key aspect of MOSA is an Enabling Environment (EE), which necessitates a combination of technical, business, and operational improvements throughout the enterprise.

Current efforts by OMG and other standards organizations are focused on defining the standards, patterns, and architectures necessary to realize a MOSA EE. This work, along with the development of Federated Digital Mesh architectures, is creating new opportunities for enhanced, faster, and more accessible engineering capabilities and services within avionics and other complex system domains.

5 References

[1] US Navy helped establish the Hardware Open Systems Technologies (HOST) standard. See *https://host-oa.com/*

[2] NAVAIR (2021). Hardware Open Systems Technologies (HOST) Tier 1 Standard v4.1 Draft. *https://host-oa.com/wp-content/uploads/2021/12/HOST_Standard_Tier_1.pdf*

[3] Shepherd, K. & Wills, J.D. (2018). Avionics Open Systems Architecture Standardization. *https://rockwellcollinsthoughtleadership.wordpress.com/wp-content/uploads/2018/06/avionics-open-system-architecture-standardization-shepherd-and-wills.pdf*

[4] Rasheed, A., Gozluklu, B., & Johnson, T. (2022). Understanding digital engineering and how it drives business value. *https://aws.amazon.com/blogs/industries/understanding-digital-engineering-and-how-it-drives-business-value/*

[5] Tokar, Joyce. (2017). An Examination of Open System Architectures for Avionics Systems – An Update. *https://www.researchgate.net/publication/315736224_An_Examination_of_Open_System_Architectures_for_Avionics_Systems_-_An_Update*

6 ACKNOWLEDGEMENTS

The views expressed in the *OMG Journal of Innovation* are the author's views and do not necessarily represent the views of their respective employers nor those of the Object Management Group® (OMG®).

➢ Return to *OMG Journal of Innovation landing page* for more articles and past editions.