Proceedings of the Annual General Donald R. Keith Memorial Conference
West Point, New York, USA
April 23, 2026
A Regional Conference of the Society for Industrial and Systems Engineering

Model-Based Systems Engineering for the General Air Defense System (GADS) Interceptor

John Choung, Soo-Ahn Kim, Mark Rybaltowski, Erin Son, Ryan Stader, and Andrew Brigman

Department of Systems Engineering, United States Military Academy, West Point, New York 10996

Corresponding author's Email: rstader22@gmail.com

Author Note: We are extremely thankful for the support, mentorship, and funding provided by the Department of Systems Engineering at the United States Military Academy and our project sponsor, who provided us with the necessary materials and information to complete this project. The views expressed herein are those of the authors and do not reflect the position of the United States Military Academy, the Department of the Army, or the Department of War.

Abstract: This work seeks to provide an integrated analysis of a General Air Defense System (GADS) Interceptor by integrating technical research on the system within a Model-Based Systems Engineering (MBSE) framework. Technical research was broken down into five primary areas: MBSE, the system Concept of Operations (CONOPs), the interceptor's physical architecture, the communication system, and the power distribution system. The technical data gained from this research allowed for the creation of an interactive system model sustained by the Systems Modeling Language (SysML) in conjunction with a systems modeling tool, Magic System of Systems Architect (MSOSA). This model conveys an architecture that will enable stakeholders to easily visualize subsystem interactions, maintain configuration traceability, and support future system architecture development.

Keywords: Interceptor, Model-Based Systems Engineering (MBSE), Concept of Operations (CONOPs), Systems Modeling Language (SysML), Magic System of Systems Architect (MSOSA).

1. Introduction

Relying on legacy document-based systems engineering (DBSE) for a complex system significantly limits growth capacity, hampers efficiency, and makes it difficult to achieve a shared understanding across an organization. The General Air Defense System (GADS) Interceptor is an example of such a system, where extensive organizational effort is put into maintaining and understanding DBSE artifacts. The GADS interceptor is a critical air-defense asset that has seen service across the world. Its decades of service, paired with constant production and system upgrades, make the DBSE process a work-intensive and time-consuming endeavor that must be completed before making any changes to the system. By adopting a Model-Based Systems Engineering (MBSE) architecture, this manpower can be redirected towards pursuing efforts that directly add value to the system, stakeholder, or organization. This project develops an integrated, end-to-end system model that demonstrates the value of MBSE, evaluating how MBSE offers engineers a superior interface when working with a complex system as compared to DBSE. The MBSE products created through this project enable engineers to visualize GADS subsystem interactions, maintain configuration traceability, and support future system architecture development.

2. MBSE Overview

The International Council on Systems Engineering recognizes MBSE as a model-centric application of systems engineering to support system design, traceability, and validation throughout the entirety of a system's life cycle. The MBSE initiative was established in 2007, as it was deemed necessary to increase the scope of systems engineering and replace document-based systems engineering (INCOSE, 2025). DBSE is the manual creation of program artifacts such as the concept of operations (CONOPs), system design specifications, and test case specifications (Delligatti, 2013). These artifacts are produced in the form of text documents, spreadsheets, diagrams, and presentations in a decentralized manner. DBSE is not obsolete; however, time and effort must be dedicated to maintaining document-based artifacts. Failure to do so incurs cost and scheduling risks, as resources, time, and manpower continue to be redirected to keep design artifacts up to date.

With the MBSE approach, systems engineers will execute the same system development activities and create similar deliverables as document-based systems engineering. The key difference with MBSE is the building of a modular and traceable system through a software-based modeling tool (Delligatti, 2013). This digitized system will serve as a centralized archive for design decisions, rather than having to maintain multiple artifacts in different forms. Furthermore, with tools such as Magic System of Systems Architecture (MSOSA), any changes that are made will be automatically propagated throughout the entire system; however, despite this automatic propagation, traceability must be carefully curated and maintained by the modeler.

3. The Systems Modeling Language

SysML is a broad and expansive graphical modeling language that allows a user to visualize and communicate the core aspects of a system's structure, behavior, requirements, and mathematical models. The main purpose of SysML is the visualization and communication of a system's design among stakeholders (Delligatti, 2013). This is paramount to conveying the design of the Interceptor to our primary stakeholders. Within SysML, three out of the nine key diagrams were used in this project: the Block Definition Diagram (BDD), the Internal Block Diagram (IBD), and the Package Diagram. These defined diagrams can be used to convey system design information, and each diagram serves its own purpose, detailing specific information about each aspect of the system.

3.1 Block Definition Diagrams

Block Definition Diagrams are used to show functional and hierarchical relationships between the components of a system. BDDs can be used to display multiple kinds of model elements and highlight various relationships to express information about the physical architecture of the system. The specific model elements that are displayed on BDDs, such as blocks, value types, and constraints, serve as archetypes for other model elements that will appear on other SysML diagrams, such as IBDs (Delligatti, 2013). BDDs can be created in combination with other SysML diagrams to help provide context and understanding of whatever aspect of the system is of primary interest at that moment within the system development stage (No Magic, Inc.). Overall, the BDD is the primary diagram used to communicate structural information about a system, and it allows system engineers to convey the types of structures that exist inside the system and outside the system.

3.2 Internal Block Diagrams

The Internal Block Diagram (IBD) is used to show the connections within the subsystems of the overarching system. It can be used to display various elements to express parts of a system's structure and build upon the design aspects iterated in the BDD (Delligatti, 2013). IBDs are created to specify the internal structure of a single block (No Magic, Inc.). Compared to a BDD, an IBD does not display a hierarchy, but instead defines the interactions between system components and external systems. Said interactions come in three primary forms: physical, power, and data/signal. A fully developed IBD can be leveraged to establish the production specifications of system components, subcomponents, and parts across varying functional disciplines.

3.3 Package Diagrams

Package diagrams are primarily used to display the organization and hierarchy of a system model (Delligatti, 2013). Rather than illustrate the technical functions and connections of certain components, subcomponents, and parts, package diagrams serve to help the user and stakeholders visualize all aspects of a digitized system model. This is especially critical for developing highly technical system models and helping coordinate with the various functional engineers. Overall, package diagrams allow for a greater visualization and understanding of the components, subcomponents, and parts of the system.

4. Modeling & Results

Stakeholder analysis identified significant drawbacks to relying on document-based systems engineering for the GADS Interceptor. The engineering process for the Interceptor was one of intense effort, given the complexity of the Interceptor, which had been translated into multiple iterations of documentation and diagrams. Significant work was required to understand and share knowledge from legacy documents prior to any substantial work having been done on the system. It was determined that the team would create a modern, improved alternative to DBSE: a comprehensive digital model using MBSE, based on existing Interceptor documentation. Extensive missile defense interceptor research was conducted to create a large knowledge pool from which information was pulled to build the model. To create SysML products, we used the MSOSA modeling tool. MSOSA is specifically designed for MBSE and is used to build, analyze, and manage complex systems. While MSOSA can be used with other modeling languages, it is the industry standard for SysML modeling. MSOSA allows for less ambiguity and easier traceability in comparison to document-based systems engineering. The model created provides engineers with a structured, intuitive, and user-friendly interface to understand the GADS Interceptor that demonstrates the value of MBSE for a complex system.

4.1 BDD Creation

The hierarchical structure of the Interceptor, and the related component structures, have been captured in a BDD (Figure 1). This BDD allows for an intuitive understanding of the system, enabling engineers to visualize the decomposition of the system's elements. Each subsystem is comprised of multiple components, which have unique value types that aid in understanding the greater role of each component as part of the Interceptor. These value metrics also allow engineers to understand how to assess individual components, where further subsystem-level understanding can be reached through IBDs. The Interceptor is comprised of three primary subsystems: the Intercept Vehicle (IV), the Propulsion Assembly, and the Interstage, displayed in the block connections of our BDD. (Figure 1) The BDD also demonstrates an example of a lateral system, the Cannister, which identifies that the Interceptor is contained within the Cannister.

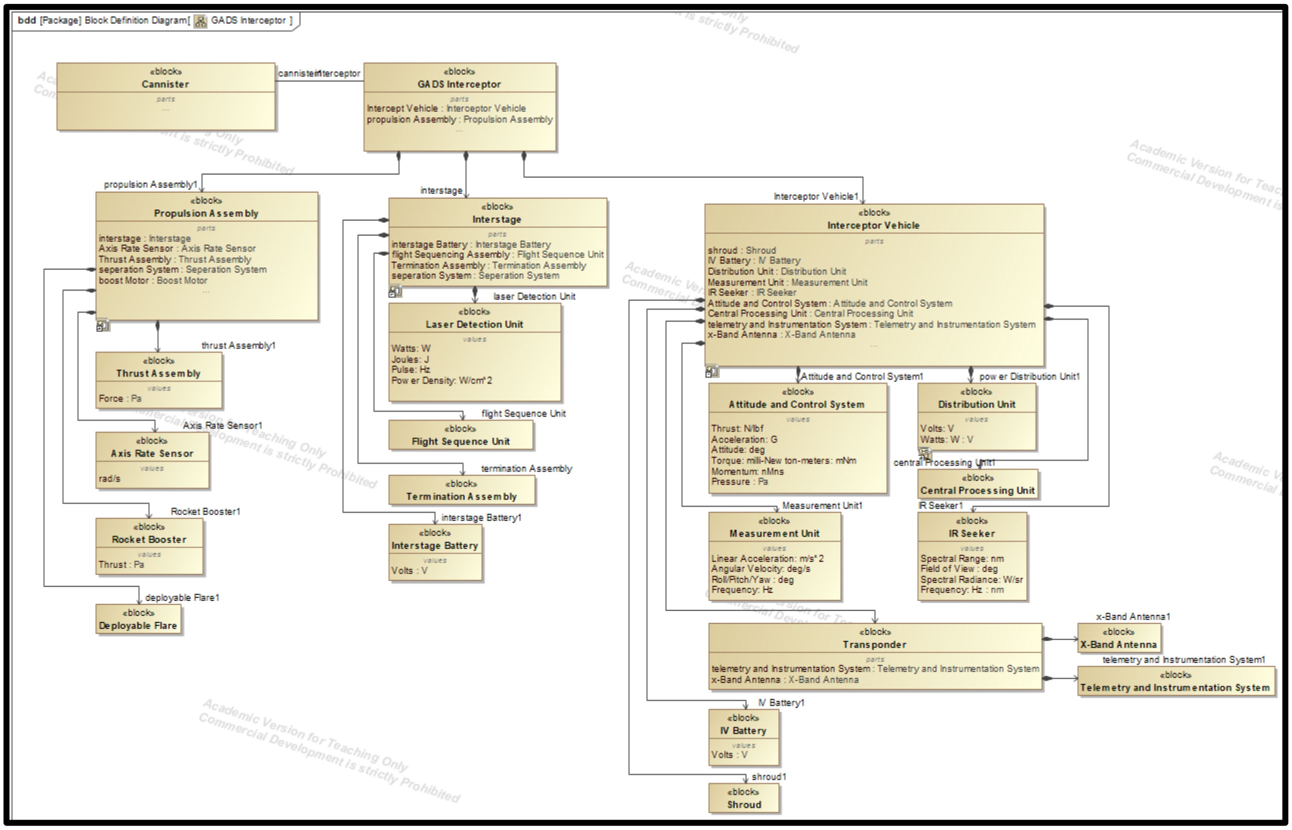


Figure 1: GADS Interceptor BDD

The process of creating this BDD required extensive research across all packages and multiple levels of interface control documents (ICDs). DBSE legacy ICDs made it exponentially time-consuming to find each subsystem and visualize them within the hierarchy of our BDD. There were few primary or foundational ICDs for identifying the subsystems of the interceptor, and multiple documents were either outdated or scattered with information. For example, vague information regarding the Transponder would be in one ICD, yet more in-depth information would be scattered across multiple ICDs, some with outdated information. The complexity of finding all up-to-date information regarding the relationships between each package made our research difficult in determining which components fell under which subsystem of our BDD. A client would have the same difficulty when reviewing current documentation whenever an update or a revision is needed for the system.

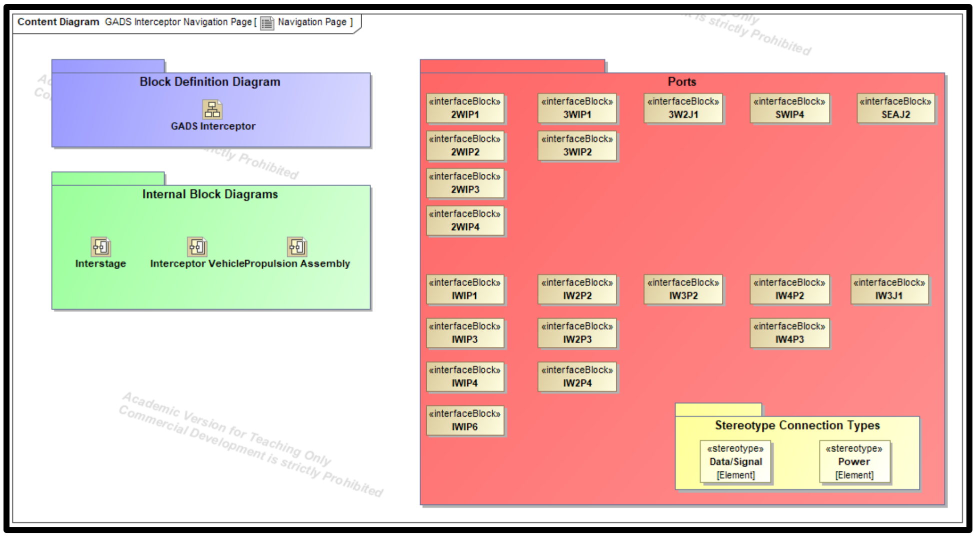


Figure 2: GADS Model Navigation Page

MSOSA also allowed manual package diagram creation, which further enables centralization of information within the GADS. Instead of having to look for different subsections of information from multiple documentation sources, the localization of information simplifies information logging and is accessible with the creation of a package diagram navigational page. DBSE requires individuals to find information in multiple sources of diagrams and documents, but with the use of a package diagram, as demonstrated in Figure 2, all information relevant to the system can be compiled into one easy-to-access and tailor-made source. The display of all the elements within our model, ranging from the BDD, IBD, Port connections, and Stereotypes, allows for a cleaner UI in comparison to DBSE.

4.2 KV IBD Creation

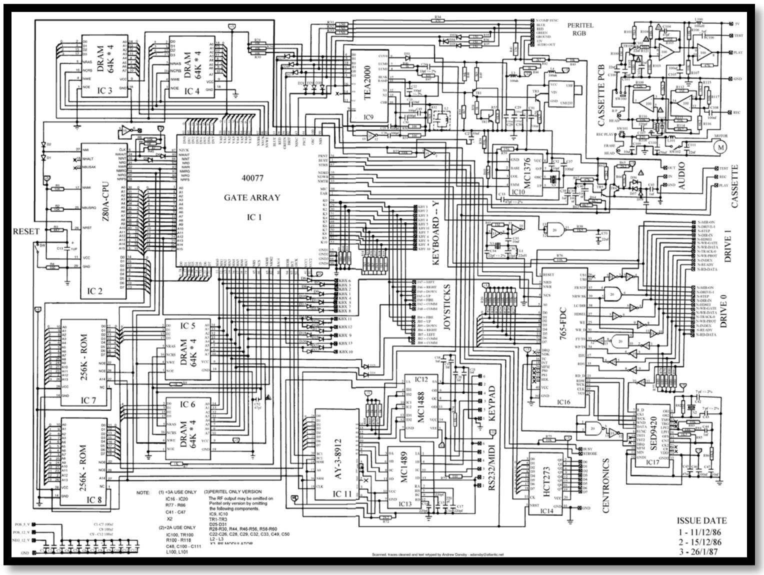


Figure 3: Example Interconnect Diagram of Circuit Board

Figure 3 above is a visual representation of how the interconnect diagrams for various GADS components are displayed in ICDs, which were the primary source used to create our IBDs. Due to the confidentiality of our research, we could not provide an interconnect diagram that corresponds to our modeling, but this figure demonstrates the level of complexity inherent between the system components. The physical tracing of the lines from each bus took an extensive amount of time to complete and introduced the additional risk factor of human error. The complexity of each of these interconnect diagrams meant that it took weeks to find and distribute the port identities to each block within our IBDs. Furthermore, the datatypes that would be transmitted through these connections were found in separate ICDs, adding to the complexity of the task of correctly understanding the system.

Our Intercept Vehicle IBD (Figure 4) provides a window into the Intercept Vehicle subsystem, allowing for a better understanding of how individual components interact with each other, connect outside of the subsystem, and contribute to the Interceptor's operation. The IV subsystem is responsible for communication and Interceptor guidance. The components contained in the IV include the Transponder, Attitude Control System (ACS), Measurement Unit (MU), Seeker, Distribution Unit (DU), Central Processing Unit (CPU), IV Battery, and Shroud. Each individual component is connected through ports, labeled with a unique nomenclature to identify each port. The first name in each port identifies the female connection that is leaving the component, with following names preceded by a colon identifying the male connection that is leaving from the origin port to the connecting components port. For example, in the IV IBD (Figure 4), the CPU is connected to the Seeker. The male port would be considered the MCUJ1 port, while the subsequent female port to the Seeker would be the SEAJ2. This allows engineers to understand how blocks interact through ports within the model and identify these connections by name, consistent with the naming conventions used in legacy DBSE efforts.

These connections involve two datatypes: power and data. Within the IBD, Power is identified by red connectors, labeled as "Power," and data is identified by green connectors, labeled as "Data/Signal," as constructed in stereotypes within the IBD. If additional stereotypes are needed, it is possible to create more data types to further describe the system. The arrows indicate the direction of power and data flow, enabling traceability throughout the IBD. Ports also exist on the IBD frame to represent connections to external components, illustrating how subsystems interact through individual components and their value types.

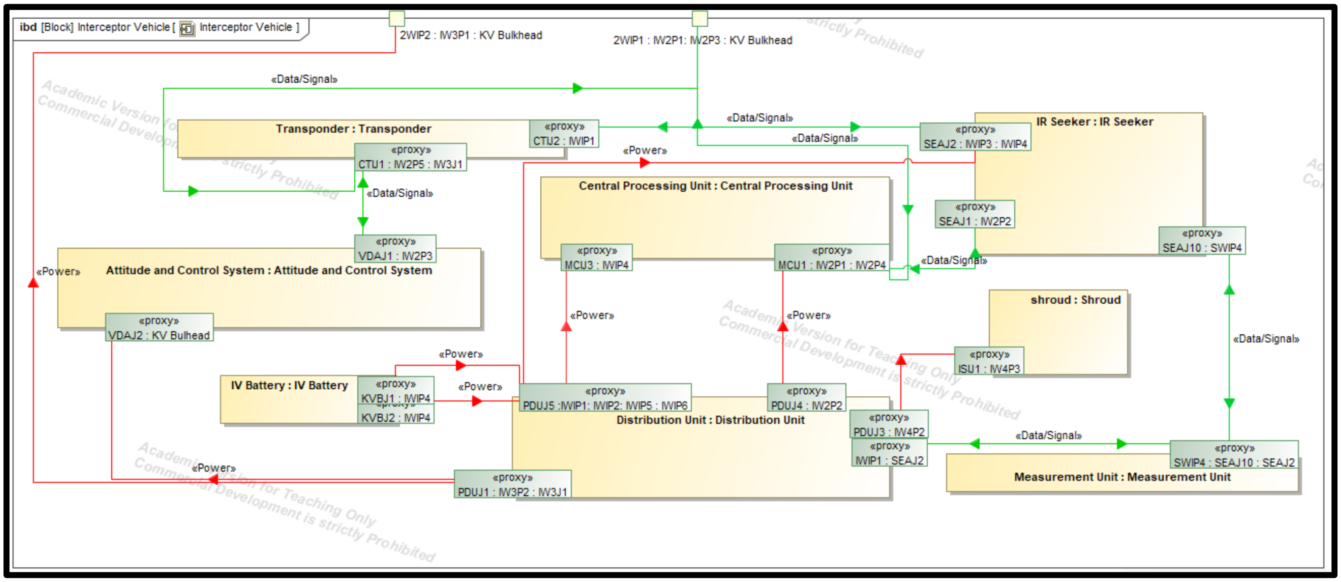
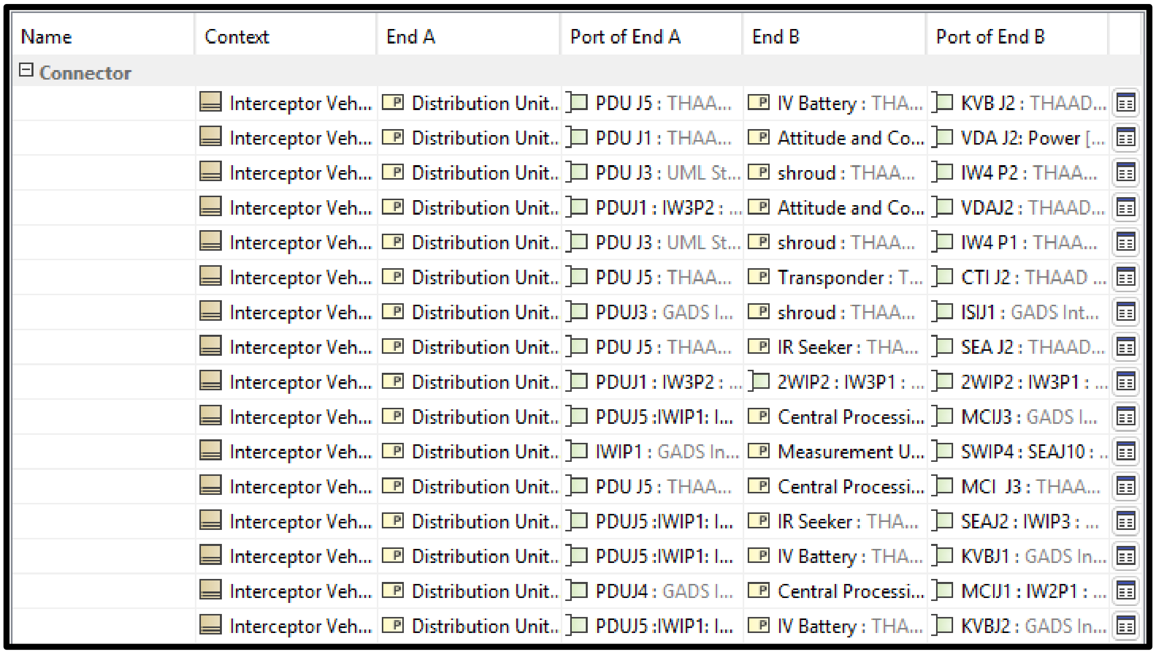


Figure 4: GADS Intercept Vehicle IBD

Our modeling also includes digital traceability through the MSOSA system, contrasting the physical tracing of port connections from the interconnect diagrams prevalent in DBSE. MSOSA offers an optimized alternative by allowing users to find associated port connections through specification filtering, which can be applied to the block of interest itself (Figure 5). Every connection is described with a start point and endpoint, designating which port they exit and enter from. Rather than having to put extensive effort into searching through multiple documents to find this information using DBSE methods, SysML, on the contrary, allows for identifiable traceability measures in relationships and IBD connections.



Name	Context	End A	Port of End A	End B	Port of End B
⊟ Connector					
	Interceptor Veh...	Distribution Unit..	PDU J5 : THAA...	IV Battery : THA...	KVB J2 : THAAD...
	Interceptor Veh...	Distribution Unit..	PDU J1 : THAA...	Attitude and Co...	VDA J2: Power [...
	Interceptor Veh...	Distribution Unit..	PDU J3 : UML St...	shroud : THAA...	IW4 P2 : THAA...
	Interceptor Veh...	Distribution Unit..	PDUJ1 : IW3P2 : ...	Attitude and Co...	VDAJ2 : THAAD...
	Interceptor Veh...	Distribution Unit..	PDU J3 : UML St...	shroud : THAA...	IW4 P1 : THAA...
	Interceptor Veh...	Distribution Unit..	PDU J5 : THAA...	Transponder : T...	CTI J2 : THAAD ...
	Interceptor Veh...	Distribution Unit..	PDUJ3 : GADS I...	shroud : THAA...	ISIJ1 : GADS Int...
	Interceptor Veh...	Distribution Unit..	PDU J5 : THAA...	IR Seeker : THA...	SEA J2 : THAAD...
	Interceptor Veh...	Distribution Unit..	PDUJ1 : IW3P2 : ...	2WIP2 : IW3P1 : ...	2WIP2 : IW3P1 : ...
	Interceptor Veh...	Distribution Unit..	PDUJ5 :IWIP1: I...	Central Processi...	MCIJ3 : GADS I...
	Interceptor Veh...	Distribution Unit..	IWIP1 : GADS In...	Measurement U...	SWIP4 : SEAJ10 : ..
	Interceptor Veh...	Distribution Unit..	PDU J5 : THAA...	Central Processi...	MCI J3 : THAA...
	Interceptor Veh...	Distribution Unit..	PDUJ5 :IWIP1: I...	IR Seeker : THA...	SEAJ2 : IWIP3 : ...
	Interceptor Veh...	Distribution Unit..	PDUJ5 :IWIP1: I...	IV Battery : THA...	KVBJ1 : GADS In...
	Interceptor Veh...	Distribution Unit..	PDUJ4 : GADS I...	Central Processi...	MCIJ1 : IW2P1 : ...
	Interceptor Veh...	Distribution Unit..	PDUJ5 :IWIP1: I...	IV Battery : THA...	KVBJ2 : GADS In...

Figure 5: Distribution Unit Ports and Relationships

5. Conclusion

Our research provides substantial evidence that advocates for the usage of MBSE over DBSE practices. The ability for MBSE, specifically SysML, to trace system requirements to structural elements, minimize time spent reviewing technical and research documents, and reduce human error, proves that MBSE is substantially more efficient than DBSE. MBSE provides the necessary framework for research across multiple documents, functional areas, and domains to be consolidated in a single platform that enables engineers to track changes in the system and continue to develop the system from the meta-system down to the part level. Moving forward, further research should go into traceability between requirements and components. This work displays the baseline capabilities of MBSE. The ultimate functionality of MBSE comes to fruition once an organization dedicates an initial effort towards transitioning to MBSE, culminating in an optimal interface, especially once tailored to specific operating procedures and system needs.

7. References

Delligatti, L. (2013). SysML Distilled: A Brief Guide to the Systems Modeling Language. Addison-Wesley. Friedenthal, S., Moore, A., & Steiner, R. (2011). *A practical guide to SysML: The systems modeling language* (2nd ed.). MK/OMG Press.

No Magic, Inc. (n.d.). *SysML Block Definition Diagram*. Retrieved November 4, 2025, from https://docs.nomagic.com/spaces/SYSMLP190SP1/pages/36330100/SysML+Block+Definition+Diagram

No Magic, Inc. (n.d.). *Defining blocks in the Block Definition Diagram*. Retrieved November 3, 2025, from https://docs.nomagic.com/spaces/SYSMLP190/pages/30375448/Defining+Blocks+in+Block+Definition+Diagram

Vaccaro, A., Adarkwah, D., Tyquiengco, B., McDermott, R., Katsigiannis, V., & Enos, J. (2025, April 24). *Modernizing air defense: Applying model-based systems engineering to the Notional Air Defense System*. Department of Systems Engineering, West Point, NY, United States.