Date: April 2025

Standards
Development
Organization®

System Profile for Effective Cyber Threat-based Risk Assessments (SPECTRA), version 1.0

Volume 1: Introduction

OMG Document Number: **ptc/25-04-13**

Standard Document URL: **https://www.omg.org/spec/SPECTRA/**

This OMG document replaces the submission document (Sysa/24-11-02). It is an OMG Adopted Beta Specification and is currently in the finalization phase. Comments on the content of this document are welcome and should be directed to issues@omg.org by June 2025.

You may view the pending issues for this specification from the OMG revision issues web page https://issues.omg.org/issues/lists.

The FTF Recommendation and Report for this specification will be published in April 2026. If you are reading this after that date, please download the available specification from the OMG Specifications Catalog.

USE OF SPECIFICATION – TERMS, CONDITIONS & NOTICES

LICENSES

PATENTS

GENERAL USE RESTRICTIONS

DISCLAIMER OF WARRANTY

WHILE THIS PUBLICATION IS BELIEVED TO BE ACCURATE, IT IS PROVIDED "AS IS" AND MAY CONTAIN ERRORS OR MISPRINTS. THE OBJECT MANAGEMENT GROUP AND THE COMPANIES LISTED ABOVE MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS PUBLICATION, INCLUDING BUT NOT LIMITED TO ANY WARRANTY OF TITLE OR OWNERSHIP, IMPLIED WARRANTY OF MERCHANTABILITY OR WARRANTY OF FITNESS FOR A PARTICULAR PURPOSE OR USE. IN NO EVENT SHALL THE OBJECT MANAGEMENT GROUP OR ANY OF THE COMPANIES LISTED ABOVE BE LIABLE FOR ERRORS CONTAINED HEREIN OR FOR DIRECT, INDIRECT, INCIDENTAL, SPECIAL, CONSEQUENTIAL, RELIANCE OR COVER DAMAGES, INCLUDING LOSS OF PROFITS, REVENUE, DATA OR USE, INCURRED BY ANY USER OR ANY THIRD PARTY IN CONNECTION WITH THE FURNISHING, PERFORMANCE, OR USE OF THIS MATERIAL, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

The entire risk as to the quality and performance of software developed using this specification is borne by you. This disclaimer of warranty constitutes an essential part of the license granted to you to use this specification.

RESTRICTED RIGHTS LEGEND

Use, duplication or disclosure by the U.S. Government is subject to the restrictions set forth in subparagraph (c) (1) (ii) of The Rights in Technical Data and Computer Software Clause at DFARS 252.227-7013 or in subparagraph (c)(1) and (2) of the Commercial Computer Software - Restricted Rights clauses at 48 C.F.R. 52.227-19 or as specified in 48 C.F.R. 227-7202-2 of the DoD F.A.R. Supplement and its successors, or as specified in 48 C.F.R. 12.212 of the Federal Acquisition Regulations and its successors, as applicable. The specification copyright owners are as indicated above and may be contacted through the Object Management Group, 9C Medway Rd, PMB 274, Milford, MA 01757, U.S.A.

TRADEMARKS

CORBA®, CORBA logos®, FIBO®, Financial Industry Business Ontology®, FINANCIAL INSTRUMENT GLOBAL IDENTIFIER®, IIOP®, IMM®, Model Driven Architecture®, MDA®, Object Management Group®, OMG®, OMG Logo®, SoaML®, SOAML®, SysML®, UAF®, Unified Modeling Language®, UML®, UML Cube Logo®, VSIPL®, and XMI® are registered trademarks of the Object Management Group, Inc.

For a complete list of trademarks, see: https://www.omg.org/legal/tm_list.htm. All other products or company names mentioned are used for identification purposes only and may be trademarks of their respective owners.

COMPLIANCE

The copyright holders listed above acknowledge that the Object Management Group (acting itself or through its designees) is and shall at all times be the sole entity that may authorize developers, suppliers and sellers of computer software to use certification marks, trademarks or other special designations to indicate compliance with these materials.

Software developed under the terms of this license may claim compliance or conformance with this specification if and only if the software compliance is of a nature fully matching the applicable compliance points as stated in the specification. Software developed only partially matching the applicable compliance points may claim only that the software was based on this specification, but may not claim compliance or conformance with this specification. In the event that testing suites are implemented or approved by Object Management Group, Inc., software developed using this specification may claim compliance or conformance with the specification only if the software satisfactorily completes the testing suites.

OMG's Issue Reporting Procedure

All OMG specifications are subject to continuous review and improvement. As part of this process we encourage readers to report any ambiguities, inconsistencies, or inaccuracies they may find by completing the Issue Reporting Form listed on the main web page https://www.omg.org, under Specifications, Report a Bug/Issue.

Table of Contents

Preface

About the Object Management Group

Founded in 1989, the Object Management Group, Inc. (OMG) is an open membership, not-for-profit computer industry standards consortium that produces and maintains computer industry specifications for interoperable, portable and reusable enterprise applications in distributed, heterogeneous environments. Membership includes Information Technology vendors, end users, government agencies and academia.

OMG member companies write, adopt, and maintain its specifications following a mature, open process. OMG's specifications implement the Model Driven Architecture® (MDA®), maximizing ROI through a full-lifecycle approach to enterprise integration that covers multiple operating systems, programming languages, middleware and networking infrastructures, and software development environments. OMG's specifications include: UML® (Unified Modeling Language™); CORBA® (Common Object Request Broker Architecture); CWM™ (Common Warehouse Meta-model); and industry-specific standards for dozens of vertical markets.

More information on the OMG is available at *https://www.omg.org/*.

OMG Specifications

As noted, OMG specifications address middleware, modeling and vertical domain frameworks. All OMG Formal Specifications are available from this URL: *https://www.omg.org/spec*

All of OMG‟s formal specifications may be downloaded without charge from our website. (Products implementing OMG specifications are available from individual suppliers.) Copies of specifications, available in PostScript and PDF format, may be obtained from the Specifications Catalog cited above or by contacting the Object Management Group, Inc. at:

OMG Headquarters
9C Medway Road, PMB 274
Milford, MA 01757
USA

Tel: +1-781-444-0404
Fax: +1-781-444-0320

Email: *pubs@omg.org*

Certain OMG specifications are also available as ISO/IEC standards. Please consult: http://www.iso.org

Issues

The reader is encouraged to report and technical or editing issues/problems with this specification to: https://www.omg.org/report_issue.htm

1 Preface

SPECTRA is a language for describing cyber and cyber-physical systems for the purposes of risk assessments, cybersecurity assessments and vulnerability assessments. System descriptions - including models, consist of many artifacts which are of importance for one or more lifecycle phases. For the purposes of a cybersecurity assessment, certain artifacts are of essence - for example, what are the parts of the system, how these parts are connected to convey information, what information is being conveyed, and what is the nature of the parts. Cybersecurity implies a filter for the level of technical detail, compared to other disciplines involved in the system lifecycle. Effectively extracting only the relevant cybersecurity assertions for a system description is a challenging task. The SPECTRA language - a set of conceptual entities and relations, collectively referred to as Core Assertions for cybersecurity - extends Systems Engineering languages with the means to identify the core entities and their relationships to support the task of interpreting and postprocessing a system description by automated tools and enabling cybersecurity analytics.

SPECTRA facilitates ingesting normalized machine-consumable system descriptions into compliant tools for big data analytics in cybersecurity.

SPECTRA's objective is to provide a standard compliance reference for acquisition contracts soliciting models for various assessments, as well as tools and services for performing such assessments automatically.

This specification defines a SysML v1 profile to unambiguously identify Core Assertion in a SysML model.

SPECTRA is a family of standards

SPECTRA is a family of standards, consisting of a common foundation - the SPECTRA Core Assertions Metamodel, and several concrete representations of the Core Assertions as specific SPECTRA languages.

The Core Assertions Metamodel establishes the concepts and relationships to be used in representing cyber and cyber-physical systems for the purposes of subsequent risk assessment, cybersecurity assessment or vulnerability assessment.

SPECTRA for SysML v1 is a Profile for SysML v1 (this specification). It represents core SPECTRA concepts as stereotypes for SysML elements, and maps SPECTRA relationships to SysML v1. SPECTRA v1 provides guidance on the best practices for cyber and cyber-systems, and a path to compliance, so that compliant analytics tools can correctly extract the core assertions about the system of interest from the SysML v1 model.

SPECTRA for SysML v2 defines metadata for SysML v2 that aims to accomplish the same as SPECTRA for SysML v1.

SPECTRA CSV is a representation of SPECTRA concepts and relationships as a set of CSV tables. SPECTRA CSV is to be used as an alternative format for ingesting core assertions into analytics tools, jump-starting SysML v1, v2 or UAF models, or summarizing a model of the system of interest for validation.

SPECTRA for UAF is the representation of SPECTRA concepts and relationships in UAF models that involve cyber and cyber-physical systems for the purpose of extracting the Core Assertions and ingesting them by compliant analytics tools.

The SPECTRA Ontology is aligned with MITRE D3FEND Digital Artifacts and represents SPECTRA concepts and relationships in OWL.

The Core Assertions Metamodel defines its own format in JSON.

Each part of the SPECTRA family is developed as a stand-alone specification and defines its own compliance points. The reference to the common foundation allows multiple possibilities of interoperability between compliance points.

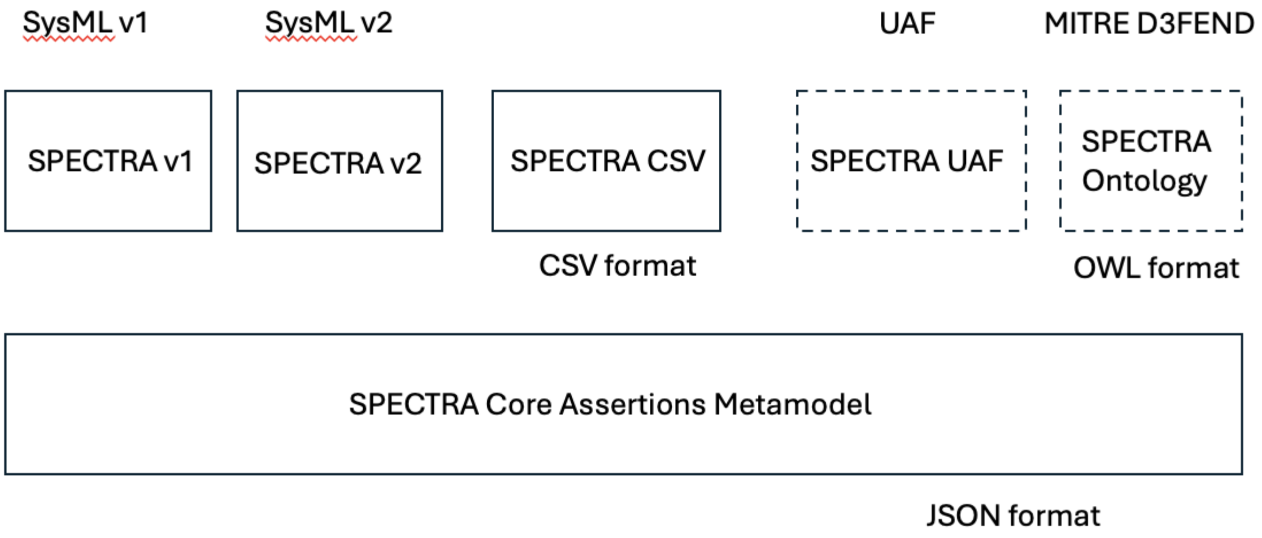

Figure 1 SPECTRA family of specifications

SPECTRA provides a language-neutral foundation to develop shared content (such as metrics and rules) for cyber and cyber-physical systems.

Core Assertions about Cyber and Cyber-Physical Systems

SPECTRA provides a language for describing cyber and cyber-physical systems that can be used stand-alone, or as extensions of SysML v1,v2 or UAF, to clarify the following questions:

- What are the replaceable parts/assemblies in the system-of-interest?
- How are the parts connected?
- How do parts collaborate?
- What is conveyed between the parts (data in transit)?
  - application data
  - security data
  - maintenance data
- What is stored in the parts (data at rest)?
- How do parts convey? What are the carrier interfaces? and what are the carrier data?
- What do the parts do?
  - at a minimum: send, receive, store, retrieve (as defined by items above)
  - but we can optionally have a custom vocabulary of named "functions", saying how parts process/transform data, how they store data, and how they are connected
- What is the nature of the replaceable parts/assemblies?
- What are the capabilities, implemented by the system of interest?
- What are the missions supported by the system-of-interest?
- Who can access what?
- Who supplies parts? (who assembles parts, who does maintenance, who replaces parts, who upgrades parts)
- What are the mitigations?
- What are the names of the controls?
- What is the context in which to interpret control names?

1.1 SPECTRA and Systems Engineering

SysML has dramatically improved the way system engineers (SE) communicate their designs. As described in [4], “previously, systems engineers would keep track of the system design using disparate tools such as Word documents, Excel spreadsheets, computer models, program schedules, and PowerPoint, in fact, lots and lots of PowerPoint. Diagrams were created in PowerPoint to describe various aspects of the system design, often incorporating information from other documents, using symbology chosen by the author to best communicate their thoughts.” SysML as a standard modeling language advances this situation “by introducing a “vocabulary” for describing systems (ie. – a standard symbology to describe different aspects of the system). Now engineers need only learn the modeling language and develop diagrams in the descriptive modeling tool to communicate the system design. Stakeholders that also understand the modeling language immediately know how to interpret the diagrams." [4].

The benefits of SysML do not stop with improving communication within the engineering team and communicating designs to the stakeholders. As SysML models are fully digital, consumable by machines, SysML allows building a database of system design facts: "Because the descriptive modeling tool is a database, the symbols on the diagrams also hold relevant details about the system object they represent and, most importantly, enter into relationships with other system objects (stored in the database). With all the information describing the system design stored in the database, the modeling tool is referred to as "the single source of truth" for the system design" [4].

SysML is a general-purpose language for Systems Engineering, and as such it provides standard meanings for representing any systems that involve any kinds of elements, any kinds of connectors, collaborating by exchanging any kind of items. SysML is a flexible language that supports a multitude of Systems Engineering methodologies and allows multiple patterns and language constructs to represent its meanings.

As SysML models of cyber and cyber-physical systems are more and more utilized in subsequent automated analysis such as a threat-based risk analysis, engineers need the ability to clarify their intent and communicate their designs above and beyond the standard SysML meanings. Cyber systems involve many additional meanings that are common among such systems. For example, elements of a cyber system usually involve digital information processing using some computing hardware, operating system, and software. Connectors usually involve computer networks, links and buses. The exchanges between the system elements usually involve digital information items.

Also, the supply chain aspect of the cyber system is very important: the elements of a cyber system are assembled from predefined components, as described by a Bill-Of-Material (BOM), and a Software Bill-Of-Material (SBOM). Understanding the supply chain is important for understanding attack vectors of a cyber system of interest.

Cyber-physical systems (cyber systems that include additional elements to interact with the physical environment) add further meanings, as some elements and connectors can be mechanical, electrical, or even human, exchange elements involve some physical things.

Cyber and cyber-physical systems involve so-called emergent elements, such as system functions, capabilities, missions, that result from collaboration and exchanges of digital information between the individual system parts.

SysML guidance offered in the literature suggests several stereotypes that clarify the generic meanings and patterns of SysML for more precise representation of cyber and cyber-physical systems [1]. However, such stereotypes are not standardized and cannot be used by a multitude of analytics tools that can ingest SysML models to "understand" the System of Interest (SOI).

The main objective of SPECTRA is to define a comprehensive profile for representing cyber and cyber-physical systems in SysML as a collection of more detailed meanings on top of the base SysML language.

Identifying Core Assertions in SysML models

A SysML model uses the syntax and semantics of the SysML language to describe a system of interest (or a system-of-systems, an enterprise, a mission, or a subsystem, or one of the components, depending on the scope of the system engineering project). A SysML model makes various assertions according to the selected System Engineering methodology. Only a subset of these assertions is related to the Core Assertions outlined above. Other assertions are related to the Requirements, Context, various viewpoints, design alternatives, technical considerations, etc. Since the same set of language elements is used in a SysML model to represent various assertions, identifying and extracting only the Core Assertions about the System of Interest (SOI), can be a challenging task, that amounts to reverse engineering of the entire SysML model.

As SysML is applicable to a variety of domains, and when it comes to modeling cyber and cyber-physical systems for the purposes of risk assessments and cybersecurity activities, many concepts that are quite common among such systems are represented in different ways, using proprietary profiles and modeling conventions [1,2,3].

Use Cases for SPECTRA

There are three main use cases for SPECTRA:

- Take an existing SysML model, add SPECTRA stereotypes making the model interpretable by compliant tools
- Start from scratch, use SPECTRA guidance and stereotypes to represent the system in SysML resulting in a compliant model interpretable by compliant tools
- Develop common content related to cyber and cyber-physical systems using SPECTRA Core Assertions as a language-neutral foundation. The content directly references SPECTRA conceptual entities and relationships rather than language-specific modeling elements (such as SysML v1 or v2, or UAF).

Background

The US government (USAF Authorization Office for Aircraft) has been leading the digitization of system architecture and other inputs for the ATO process and has created a structured MS Word template for Architecture Analysis Report (AAR), referred to as the AAR template.

The authorization process, resulting in issuing an Authorization to Operate (ATO) is performed in the context of NIST RMF (NIST SP 800-37, NIST SP 800-30, NIST SP 800-53, FIPS-199, CNSSI-1253).

The AAR template involves about 15 tables and defines structured, machine-consumable content embedded directly in MS Word documents. Other parts of the AAR document include informal prose and illustrations, but the structured tables can be interpreted on their own as the structured assertions defining the SOI. The AAR template involves only the core assertions relevant to the subsequent cyber risk assessment, in accordance with the guidance in NIST SP 800-30 and NIST SP 800-37. The AAR template can model very large systems. The AAR template has been successfully used to model systems at various levels:

- Mission
- System
- Subsystem
- Component

The Program Offices that use the AAR template want to use a Model-Based Systems Engineering approach and SysML as the “single source of truth” about the system and use the SysML models directly as input to digital engineering and analytics tools. However, the SysML standard lacks guidance on how to represent a cyber or a cyber-physical system for the postprocessing analytics (preferably automated) in the area of threat-based risk assessment and allows a multitude of approaches to model organization and structure, patterns, as well as custom profiles.

Systems Assurance Task Force of the OMG involves collaboration with USAF and many other organizations to develop standards in the area of Systems Assurance that work together in the form of a digital ecosystem for the next generation smart digital cybersecurity. This roadmap and some of the key standards have been discussed in [5].

SPECTRA is based on key tables of system information that have been identified an AAR template as essential for cybersecurity risk assessments but are hard to identify in SysML modeled systems, such as “an indented list of equipment”, “nodes inside the risk-assessment boundary”, “nodes outside the risk-assessment boundary”, and “channels crossing the risk-assessment boundary”.

SPECTRA uses Architecture Analysis Report (AAR) info as a basis for clarifying system architecture as described by the base SysML model. SPECTRA brings the systems engineering guidance and expertise for representing cyber systems from the AAR for the SysML community in the form of the SPECTRA profile and the best practices guide for modeling cyber systems with SPECTRA.

SPECTRA clarifies the Digital Technical Surface of a System of Interest

Based on the typical needs of the cyber risk assessment community, the AAR template focuses on the so-called “system architecture” viewpoint which enables investigation of attack vectors and attack paths for the system-of-interest (SOI).

“Digital technical surface” of a cyber system is the set of replaceable parts and the corresponding communications fabric that describes all digital information pathways in the SOI.

This concept is related to the “as-build architecture” of the SOI and is strangely aligned to the BOM/SBOM integration view of the SOI. In the ISO/IEC/IEEE/INCOSE usage of terminology, the system elements can be atomic (i.e. not further decomposed), or they can be systems on their own merit (i.e. decomposed into further subordinate system elements). The integration of the system elements must establish the relationship between the effects that organizing the elements has on their interactions and how these effects enable the system to achieve its purpose [7].

One of the challenges of system definition is to understand what level of detail is necessary to define each system element and the interrelations between elements [7]. One approach to defining the elements of a system and their interrelations is to identify a complete set of distinct system elements with regard only to their relation to the whole (system) by suppressing details of their interactions and interrelations. This is referred to as partitioning of the system [7]. Each element can be either atomic or it can be a much higher level that could be viewed as a system itself. At any given level, the elements are grouped into distinct subsets of elements subordinated to a higher-level system. Thus, a hierarchy within a system is an organizational representation of system structure using a partitioning relation [7].

According to [7], the art of defining a hierarchy within a system relies on the ability of the system engineer to strike a balance between clearly and simply defining span of control and resolving the structure of the SOI into a complete set of system elements that can be implemented with confidence.

While from the perspective of defining a new system this may seem like a rather arbitrary exercise, cybersecurity risk assessment works from the perspective of an “as-build system”, where these decisions have been made (at least tentatively, as in an iterative DevSecOps-like process). Also, the nature of digital cyber (and cyber physical) systems dictates some of the decisions to “strike a balance” and define good system elements at the appropriate level, especially when it comes to considering the details of computing hardware organization and the interactions between software and the corresponding computing devices.

Critical in the system hierarchy is the integration phase where multiple components, sub-assemblies, and assemblies come together to produce high-quality output on-time and within budget. In an integration chain for a digital cyber system many components, sub-assemblies, and assemblies are developed concurrently and integrated at different levels – this can be referred to as a “solution integration value chain” or “supply chain”. For the system at later phases of the lifecycle the integration process may be firmly committed to and described in the BOM/SBOM.

The key concept of a “digital technical surface” viewpoint is to show the SOI as a union of unique “replaceable units”, arranged into subsystems and/or other kinds of groups. In avionics an “replaceable unit” is often referred to as a Line Replaceable Unit (LRU), since it is an individually supplied, maintained, configured, and thus provides a unique context for cyber attacks. Often an LRU is recognizable by a ruggedized chassis with standard-based sockets and mounts, although in the world of IT one can think of a container image.

A replaceable unit may have subcomponents at deeper levels in the indented list of equipment. For example, a radio box may involve an internal transducer, a digital signal processing board, several printed circuit boards, a power supply, etc. It may also involve a unique assortment of firmware and software, including an operating system and custom application software. On the other hand, a subsystem is not a tangible thing of its own but a logical/functional configuration of replaceable units.

From the architecture perspective it is important to systematically identify the “replaceable unit” level in the indented list of equipment.

The “replaceable units” enumerate the architectural “places” in the SOI, with the understanding that there is nothing outside of the union of all replaceable units. Replaceable units perform functions, are units of configuration, they collaborate by exchanging information elements through links and buses to achieve the so-called emergent behaviors of the SOI, such as system functions, capabilities, and missions. Some groupings may be useful as the targets for defensive mechanisms.

Networks, links and buses are the connectors in the SOI, with the understanding that they are the only ways information can be exchanged between “replaceable units”. Connectors describe the “information pathways” in the SOI and are critical for enumerating attack paths for the SOI.

The following figure illustrates the “digital technical surface” of an aircraft system [6]. It illustrates replaceable units, connected by buses, and arranged into subsystems, and also shows a functional thread that follows the information pathways allowed by the system, with a starting point at one of the entry points, and “pivot points” in Comms and Mission Computing, ending in Weapon Computing. This representation is “aggregated” since it hides the internal structure of the “replaceable units”.

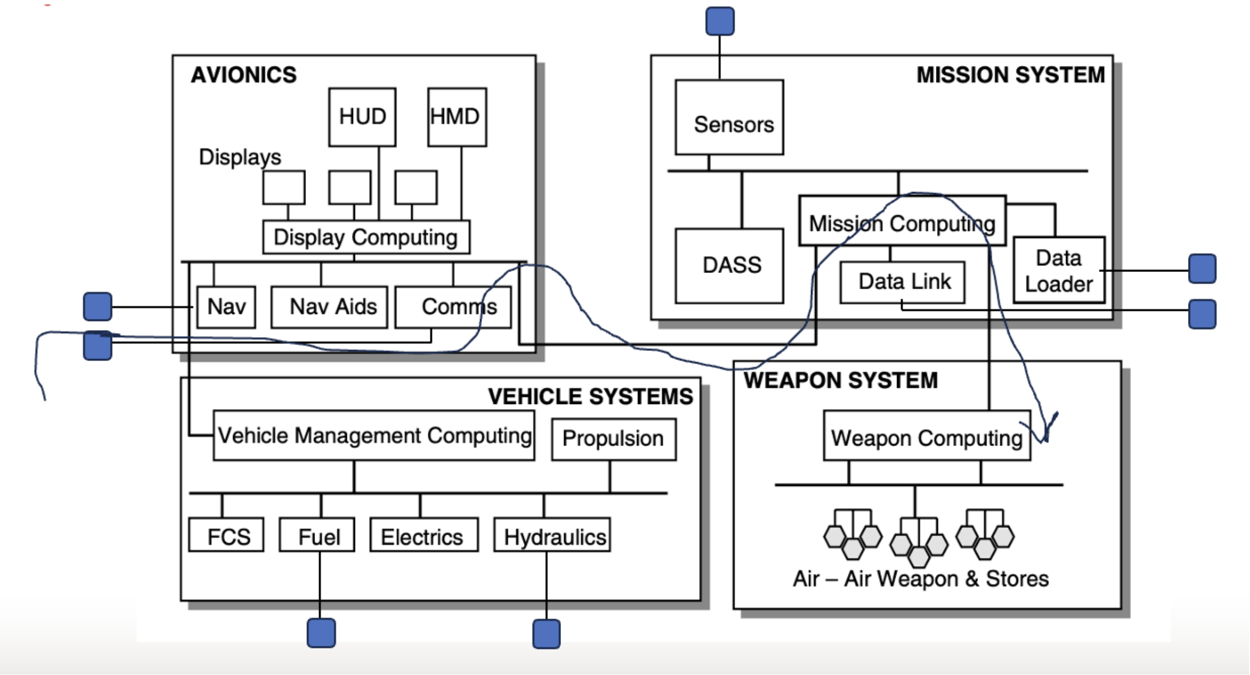

Figure 2 Digital Technical Surface of a system

From the risk assessment perspective [5], the digital technical surface of the system is important since it is used for describing attack paths for the SOI by performing a systematic “penetration walkthru”. Reliable identification of all elements of the technical surface is the logical basis for building confidence of the risk assessment assurance case [5].

In addition, the AAR template emphasizes the “protocol” viewpoint for the information pathways in the SOI and the utmost importance of separating the application domain data from the carrier domain data involved in the protocols. Risk assessment of cyber systems deals with the application domain data [5]. On the other hand, systematic enumeration cyberattacks on a specific SOI involves understanding of the technical communication protocols. Understanding of the protocols implemented by links and buses is important to understand how information flows through the different levels of the protocol stack (e.g. logical data encoded in a low-level packet), and how attacks on the connectors can involve different protocol levels.

The AAR template emphasizes the “supply chain” viewpoint, where each attackable unit has a uniform enumeration of its Bill-Of-Material (BOM) and Software Bill-Of-Material (SBOM) in a commonly interpretable way. The following figure illustrates the “artifacts” and the supply chain relationships for the Data Loader unit of the aircraft system [6].

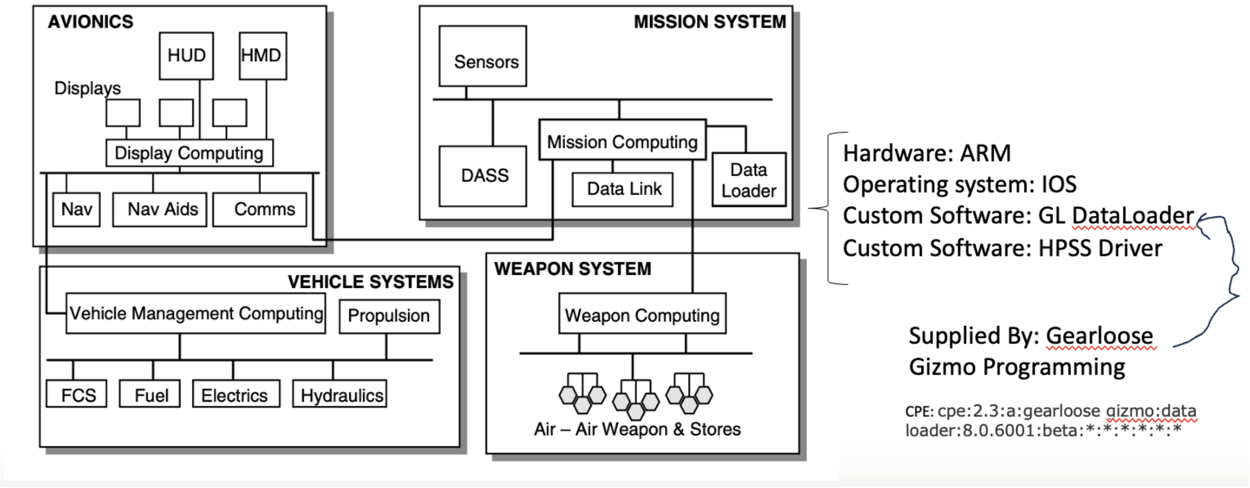

Figure 3 Artifacts with supply chain detail

Following the AAR template, SPECTRA includes a common vocabulary of the key "artifacts", such as hardware, firmware, operating system, custom application software, etc. which provide unique attack opportunities, and provides relationships to associate this information to the "attackable units". These "artifacts" can also be associated with the suppliers, as well as with the Common Platform Enumeration (CPE) codes. This viewpoint is aligned with the concepts of BOM/SBOM.

The approach behind the AAR templates is tailored to cyber systems. The Core Assertions of the AAR and SPECTRA work successfully for cybersecurity risk assessments (information assurance) of cyber (and cyber-physical systems), but for many other purposes such as for example safety analysis of materials used to build an aircraft wing, aerodynamic properties of a missile, physical security of a building, or resilience of a production plan to a nuclear blast.

As a profile for SysML, SPECTRA enables unique identification of the "digital technical surface" in the model, if it exists, or adding it to the model as part of SPECTRA annotations. Therefore, SPECTRA helps clearly communicate the nature of the digital "technical surface" of the SOI.

SPECTRA and Smart Digital Cybersecurity

SPECTRA is a vital element of a larger framework for digital cybersecurity analytics that is centered on automated, continuous assessments of systems, and optimized planning of the response and mitigation measures.

Our world has been transformed by digital innovation. The degree to which digital technology is now integrated into our daily lives would have been unimaginable just a few years ago. From social media, smartphone applications, online shopping, networked devices, the cloud, and beyond, we rely on digital technologies for more than personal enjoyment — they are integral to the systems that underpin our economy and our way of life. These interdependent systems include the communications networks that connect across the country and around the world, energy to heat our homes and power our industry, and air, train, and road travel we use every day.

To be precise, a Cyber system (CS) involves computing devices, networks, and programs to process digital information.

A Cyber-Physical System (CPS) is a cyber system that integrates physical processes with computational algorithms and networked sensors to monitor and control physical processes in real time. A cyber-physical system is a collection of computing devices communicating with one another and interacting with the physical world via sensors and actuators in a feedback loop.

Cyber Security is protection of digital information, as well as the integrity of the infrastructure housing and transmitting digital information. More specifically, cyber security includes the body of technologies, processes, practices and response and mitigation measures designed to protect networks, computers, programs and data from attack, damage or unauthorized access so as to ensure confidentiality, integrity and availability.

The smart digital cybersecurity is about the synergy between digital engineering and cybersecurity on one hand, and between cybersecurity and big data analytics on the other hand. Cybersecurity is done by the combined efforts of the systems integrators, the engineering team and their entire supply chain, and the local operations team.

Usually, an engineering team builds COTS components, the system gets integrated and deployed once or all over the world by one or more system integration teams. Then the cybersecurity team at each local installation builds the cybersecurity defenses for their own deployed system and assumes responsibility for secure operations. Some organizations are mandated to get an authorization to operate (ATO) for their system. ATO is obtained through the process of Authorization and Certification, for example (NIST-800-37).

Cybersecurity is an ongoing, cycling activity. Engineering teams issue updates (this is one cycle); while the local cybersecurity team assesses their defenses and improves them (this is a different cycle). At the same time cyber attackers figure out the gaps, improve their weapons, search for the targets, and attack them. This is yet another cycle.

Cybersecurity can benefit from a strong, cybersecurity minded engineering team, and a strong cybersecurity team.

As engineering teams become more cybersecurity aware, more cybersecurity protection technologies and processes become integrated into the next generation components making defenses stronger and responses more automated, adaptive, and rapid reducing the gaps that can be exploited by the cyber threats. This makes the job of the cybersecurity teams easier.

A smart cybersecurity strategy at the local level will focus on proactively closing the gaps (no recurring gaps, and gaps being closed faster than attackers can get to them). This strategy would optimize the cybersecurity budget(s) by focusing on the responses, mitigations, processes and technologies with the biggest yield. The key to the prioritization of cybersecurity effort is understanding the risk situations.

However, before such analytics can be implemented, the obvious bottleneck must be addressed: how to effectively transfer the knowledge of the system from the engineering team to a local cyber team. In fact, this is the largest barrier of all in the way of smart digital cybersecurity.

For a cybersecurity team to add a strong protection layer, they must first understand the deployed system. A smart team would not attempt building the big data analytics related to system-of-interest manually. Therefore, a smart cybersecurity strategy involves an automated tool that:

- ingests a machine-consumable description of the system
- generates the relevant data
- produces the analytics,

leaving the humans to interpret the analytics and make planning decisions. Ideally, the machine-consumable description of the system is the single source of truth that the engineering team is developing and using.

The main objective of SPECTRA is to define a mechanism by which systems engineering models can be used as input into automated digital engineering tools to perform big data analytics for cybersecurity risk assessments.

SPECTRA and Digital Engineering Tools

SPECTRA enables Digital Engineering (DE) tools so that they can ingest SysML models of cyber and cyber-physical systems and interpret such models to reconstruct the System of Interest (SOI). There are many ways in which a cyber system represented by its SysML model may be assessed, including (but not limited to) cyber risk assessment as defined by NIST Risk Management Framework (RMF) (NIST-800-30), and ISO/IEC 27005, cybersecurity assessment, safety and reliability assessment. SPECTRA's key use case is the so-called threat-based risk assessment. A threat-based risk assessment identifies the threat profile of the SOI, that involves capable and motivated attackers, reconstructs the attack vectors and possible attack paths for the SOI, identifies common failures and evaluates how effective existing mitigation controls are against the predicted attacks. A threat-based risk assessment often involves an "attack tree", which correlates attacks and attackers to common failures and then to common situations, referred to as "risks".

A DE tool performing cyber analytics involves multitude of inputs, as illustrated below. Often, inputs involve the system architecture of SOI (at the level relevant for cyber assessment), categorizations (e.g. how sensitive information elements and other assets are to impacts on confidentiality, integrity and availability), current mitigations, current known vulnerabilities, and current threats. Typical outputs of a cyber assessment involve identified vulnerabilities, identified risks and recommended controls. Typical tool may include some generic cyber security knowledge (shown as "context" of the assessment below). The context may include attack techniques catalog (e.g. MITRE ATT&CK), defense techniques catalog (e.g. MITRE D3FEND), digital artifacts catalog (e.g. part of MITRE D3FEND), control catalog (e.g. NIST 800-53a), threat catalog (e.g. part of MITRE ATT&CK), vulnerabilities catalog (e.g. CWE, CVE, NVD, etc.).

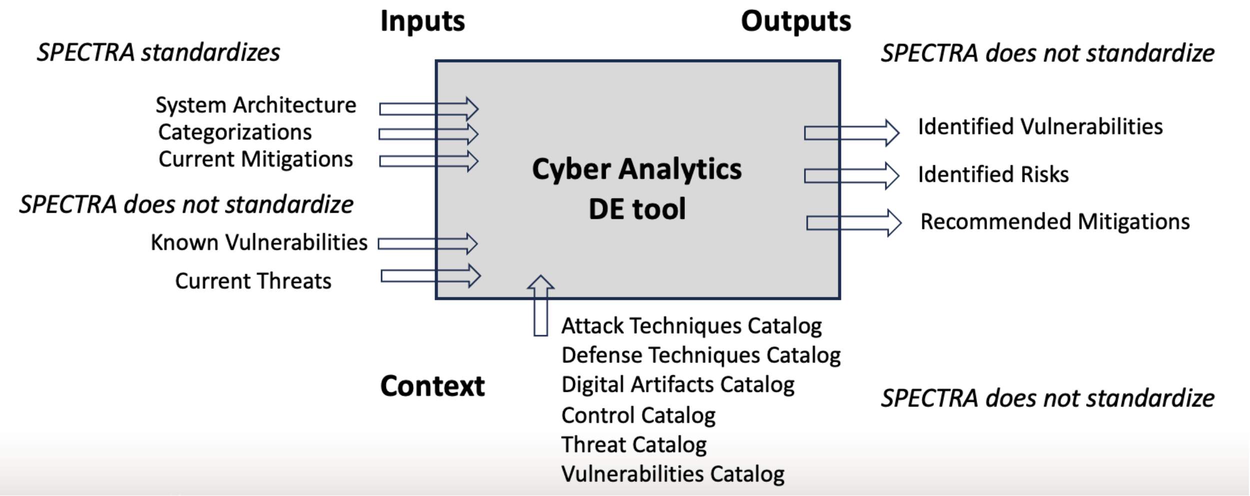

Figure 4 SPECTRA standardizes systems engineering inputs for cybersecurity analytics

SPECTRA Objectives

The overall objective of SPECTRA is to standardize the representations of the system architecture and (optionally) categorizations of its elements and existing mitigations. This overall objective breaks down into several sub-objectives.

- SPECTRA defines a more precise vocabulary for representing cyber and cyber-physical systems, especially for (but not limited to) the purposes of interpreting such systems when post-processed in the context of a threat-based cyber risks assessment
- SPECTRA specification can be referenced by acquisition contract for both models and tools
- SPECTRA includes guidance on how to model a cyber-physical system in SysML for subsequent threat-based cyber risk assessment
- SPECTRA has built-in validation of the model:
    - consistency,
    - correctness,
    - completeness
- SPECTRA enables automated cyber analytics tools to ingest and interpret SysML models of cyber and cyber-physical systems
- SPECTRA defines several key viewpoints that represent some important concerns for threat-based risk assessment:
    - information pathways allowed by the SOI
    - artifacts as both unique attack opportunities and supply chain elements
    - information flows and the corresponding protocols
    - impactful elements of the SOI
    - functional threads through the SOI
- SPECTRA addresses some of the fundamental questions about the SOI:
    - Where? (places on the information pathways)
    - When? (places/moments on the system functional threads)
    - To what? (artifacts, representing unique attack opportunities)
    - Impacting what? (impactful elements)
    - By what? (existing mitigation by controls)
- SPECTRA connects detailed, technology-specific elements to the impactful elements

- SPECTRA can be aligned with ontologies that define risks, offensive and defensive techniques, digital artifacts, mitigations, etc.
- SPECTRA provides a language-neutral foundation for developing shared content related for cyber and cyber-physical systems. Examples of such shared content are metrics and rules.

What SPECTRA Is and Isn't

SPECTRA does not describe any outputs of any Cyber Analytics tool. SPECTRA does not describe any of the ontologies/knowledge bases that are typically used as the context of the assessment by a Cyber Analytics tool. Further, SPECTRA does not describe all inputs, focusing only on the architecture of the SOI, and (optionally) categorization of system's elements (aligned with the FIPS-199) and existing mitigations.

Table 1 What SPECTRA Is and Isn't

What SPECTRA Is	What SPECTRA Isn't
SysML profile for modeling cyber systems ● intended for modeling tools (e.g. Cameo, Rhapsody, Sparx Architect) to model Cyber/Cyber-physical systems ● Brings clarity to disparate modeling patterns when applied to existing models or when defining new models	Profile for Risk Assessment artifacts Profile for Cybersecurity artifacts Cybersecurity schema Threat profile Does not capture any analysis artifacts
Architectural viewpoint for cyber systems System as a union of replaceable units and channels between them	Technology-specific viewpoint Catalog/ontology of digital artifacts
Information pathways viewpoint	Catalog/ontology of attacks/offensive techniques
Supply chain viewpoint for cyber systems	Catalog/ontology of digital artifacts
Protocol viewpoint	Detailed and complete specification of system's functions and their collaborations
Mitigation viewpoint	Catalog/ontology of defense techniques
Standard interface	Proprietary interface/capability/tool

1.2 Copyright Waiver

1.3 Submitter Representative

Dr. Nikolai Mansourov
KDM Analytics
nick@kdmanalytics.com

1.4 Submitter Team

Dr. Nikolai Mansourov
KDM Analytics

Harrell Van Norman
USAF Office for Aircraft Authorization and Certification

Joe Jarzombek
Acquired Data Solutions

Jim Logan
Ontogenesis Solutions

Cory Casanave
Model Driven Solutions

Manfred Koethe
88 Solutions

Dr Lawrence Dobranski
Catalone IT Security Inc

Supporters

MITRE
University of Queensland
CAST
Intercax
Consortium for IT Software Quality

Reviewers

Darren Harper, DND Canada

Roderick Fernandes, DND Canada

Roghieh Rousine–Webb, DND Canada

Stuart Fowler, University of New South Wales, Australia

Christopher John, Defense Australia

Ravi Toor, University of Queensland, Australia

Timothy Alvord, Boeing

Peter Richardson, Boeing

Sharon Fitzsimmons, Boeing

James Ciarcia, Navy

Keith Jordan, Navy

Peter E Kaloroumakis, MITRE

Edward Bowen, L3Harris

Matthew Hause, INCOSE

1.5 Alignment with Related Specifications

The following illustration is an informal Wenn diagram representing various standards and their alignment with SPECTRA. The vertical axes on the diagram can be interpreted as "the level of abstraction", as some standards may focus on addressing the same level concerns as SPECTRA, or higher level concerns, or lower level concerns. There is no specific meaning of the horizontal axes, and it is used mostly to assert that some standards address different concerns. There is no specific meaning of the size of the circles, size is simply used to provide some visual separation between the circles. The size of overlapping areas (or lack thereof) very informally asserts the magnitude of the shared concerns. Brief descriptions of each related standard and comments related to the alignments with SPECTRA are provided in individual subsections below.

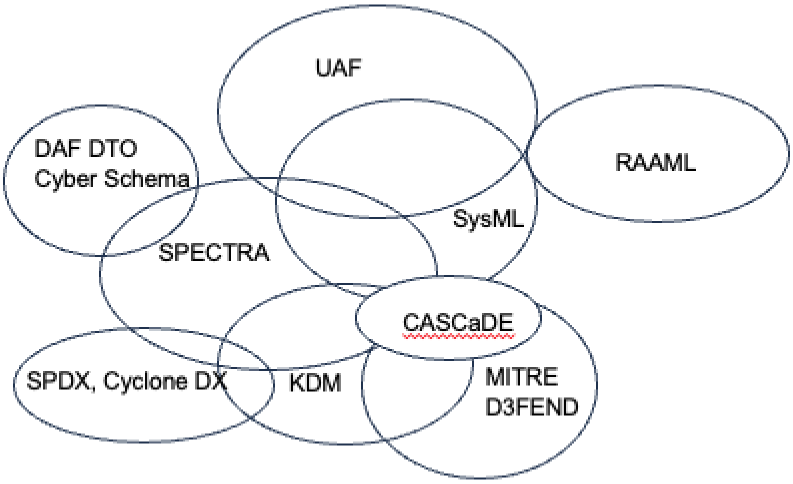

Figure 5 SPECTRA alignment with related specifications

1.5.1 SPECTRA and SysML

SysML is a general-purpose modeling language for modeling systems that is intended to facilitate a model-based systems engineering (MBSE) approach to engineer systems. It provides the capability to create and visualize models that represent many different aspects of a system. This includes representing the requirements, structure, and behavior of the system, and the specification of analysis cases and verification cases used to analyze and verify the system. The language is intended to support multiple systems engineering methods and practices. The specific methods and practices may impose additional constraints on how the language is used.

It is anticipated that SysML will be customized using this language extension mechanism to model more specialized domain-specific applications, such as automotive, aerospace, healthcare, and information systems, as well as discipline specific extensions such as safety and reliability.

As a language from the entire field of Systems Engineering, SysML has a broad scope, not specific to cyber and cyber-physical systems.

SysML does not separate several domains that are critical to cyber and cyber-physical systems. Each such system involves at least the following separate domain

- application domain - elements of the system that are responsible for its primary domain
- security domain - elements of the system-of-interest that are responsible for cybersecurity
- maintenance domain - elements that support the evolution of the system (e.g. configuration, upgrades, hot-swap, patching)
- supply domain - elements of the system that are related to the engineering, integration and supply chain

While SysML is applicable to various application domains, the other three are critical to the cybersecurity analysis of cyber systems. A profile for SysML addressing cyber system in various application domains must provide means to identify elements of each domain.

SysML lacks technology-specific detail for cyber systems. For example, if the system-of-interest uses a “network switch”, there is no standard element in SysML that can be used, so the model of the system-of-interest will use the string “network switch”, and some informal, proprietary guidance to convey the meaning. A profile for SysML addressing cyber system in various application domains must provide means to identify some key technology-specific meanings in such a way that it does not become a union “ontology for avionics”, am “ontology for electrical vehicles”, and “ontology for digital electrical substations”, etc. but a useful upper ontology level of common meanings for all these domain that facilitate the interpretation of models, plus a more in-depth ontologies for the other three domains (security, maintenance and supply).

SysML as a standard does not specify what is a good model of a cyber system, fit for the purpose of cybersecurity analytics.

The main focus of SysML as a standard is to build an extensible kernel and facilitate the MBSE tools. As the result, each organization customizes SysML with proprietary profiles and preferred modeling patterns that help encode certain meanings - to the reader who is suitably indoctrinated to this encoding. However, from the perspective of building a cybersecurity analytics tool, this is a barrier. Models with different proprietary encodings of certain meanings related to cybersecurity (this specification refers to these meanings as the “core assertions about the cyber system-of-interest”) require tailoring the cyber analytics tool.

Also there is a gap between the SysML modeling constructs (model assertions, covered by the SysML semantics), and the “core assertions about the system-of-interest”, and addressing this gap requires reverse engineering the model. In other words, different SysML modeling patterns can be used to represent the same “core assertion”; also, the same SysML element may represent very different “core assertions”. For example, not every SysML block represents a replaceable part in the system-of-interest, therefore, to build the list of all replaceable parts in the system-of-interest, one cannot simply collect certain syntactic elements or SysML, a deep interpretation of the whole model is required to extract a “core assertion”, based on the patterns in which the element is involved.

SPECTRA for SysML v1, and for SysML v2 address the above issues of developing good SysML models for cyber and cyber-physical systems that are fit for the purpose of subsequent digital cybersecurity analytics by compliant tools. The entire family of SPECTRA languages facilitates development of compliant tools that can ingest compliant models without constant tailoring, thus allowing development of COTS digital cybersecurity tools.

SPECTRA Core Assertions Metamodel formalizes the meanings that are essential for describing digital systems.

SPECTRA provides guidance to the best practices for building good models of cyber and cyber-physical systems in SysML

1.5.2 SPECTRA and UAF

UAF is a language for describing an enterprise and major entities within the enterprise.
The intent of UAF is to provide a standard representation for describing enterprise architectures using a Model Based Systems Engineering (MBSE) approach.
The core concepts in the UAF are based upon the DoDAF 2.0.2 Domain Metamodel (DM2) and the MODAF ontological data exchange mechanism (MODEM), Security Views from Canada's Department of National Defense Architecture Framework (DNDAF) and the North Atlantic Treaty Organization (NATO) Architecture Framework (NAF) v 4.
UAF models describe a system1 from a set of stakeholders’ concerns such as security or information through a set of predefined viewpoints. Developed models can also reflect custom viewpoints or users can develop more formal extensions for new viewpoints.

UAF language includes a metamodel core defining the concepts, relationships, and UAF Grid view specifications, and one representation of these concepts, relationships and views using UML/SysML v1. This language architecture allows multiple representations of the core concepts, for example in SysML v2.

UAF is defined as a family of standards, that at the moment of writing this specification, consists of the following:

1. The UAF Domain Metamodel (DMM) establishes the underlying foundational modeling constructs to be used in modeling an enterprise and major entities within the enterprise. It provides the definition of concepts, relationships, and UAF Grid view specifications. The UAF DMM is the basis for any implementation of UAF including non-UML/SysML implementations.

2. The UAF Modeling Language (UAFML) provides the modeling language specification for implementing the UAF DMM using UML/SysML.

UAF supports the capability to:
• model architectures for a broad range of complex systems, which may include hardware, software, data, personnel, and facility elements;
• model consistent architectures for system-of-systems (SoS) down to lower levels of design and implementation.
• support the analysis, specification, design, and verification of complex systems; and
• improve the ability to exchange architecture information among related tools that are SysML based.

- UAF is easier to reverse engineer than SysML because it already provides a richer set of modeling elements.
- UAF includes security views, which to some extent addresses another of the three key domain for cyber systems. However, UAF does not address the maintenance and supply chain domains.
- As a profile for SysML, UAF shares most issues with SysML, outlined in the previous section

1.5.3 SPECTRA and SPDX

Software Package Data Exchange® (SPDX®) specification defines a standard data format for communicating the component and metadata information associated with software packages, such as bills of material (BOM) and software bill of materials (SBOM). An SPDX document can be associated with a set of software packages, files or snippets and contains information about the software in the SPDX format described in this specification.
SPDX and related standardization efforts for BOM/SBOM, such as Cyclone DX, address the Supply Domain, as described in the previous sections. In particular, BOM/SBOM models describe assemblies and subassemblies of components as they are integrated into the deployed system. These specifications provide the ontology of the components used in BOM/SBOM.
SPECTRA is by design aligned with SPDX and CycloneDX

1.5.4 SPECTRA and KDM

Knowledge Discovery Meta-model (KDM) defines a meta-model for representing existing software, its elements, associations, and runtime operating environments. KDM is not a Systems Engineering language. It is a language for describing systems with the focus on the details of code. KDM also describes the runtime environment used by the system, as well as the build process that takes some artifacts (for example, source files) and produces the executable images that are deployed into the runtime environment. KDM formalizes the “core assertions related to the code and its runtime environment”. KDM can be used to reverse engineer software artifact, or to develop standard, vendor-neutral, language-neutral content related to code, such as metrics or rules. In this later capacity, KDM is widely used the foundation for various OMG standards, such as SFPM, Automated Code Quality Measures, etc.

Platform package defines a set of meta-model elements whose purpose is to represent the runtime operating environments of existing software systems. Application code is not self-contained as it is determined not only by the selected programming languages, but also by the selected Runtime platform. Platform elements determine the execution context for the application. Platform package defines meta-model elements that represent common Runtime platform concerns:

- Runtime platform consists of many diverse elements (platform resources).
- Platform provides resources to deployment components.
- Platform provides services that are related to resources.
- Application code invokes services to manage the life-cycle of a resource.

- Control flow between application components is often determined by the platform. Platform provides error handling across application components.
- Platform provides integration of application components.

Structure package defines meta-model elements that represent architectural components of existing software systems, such as subsystems, layers, packages, etc. and define traceability of these elements to other KDM facts for the same system.

The Build package defines meta-model elements that represent the facts involved in the build process of the given software system (including but not limited to the engineering transformations of the “source code” to “executables”). The Build package also includes the meta-model elements to represent the artifacts that are generated by the build process.

KDM Build Package is in alignment with BOM/SBOM standards.

SPECTRA is in alignment with KDM. SPECTRA uses some of the common technology-specific meanings for the KDM Platform Package. The Core Assertions Metamodel of SPECTRA can be used to develop vendor-neutral content related to cyber and cyber physical systems, for example fit-for-purpose metrics, validation rules, as well as cybersecurity metrics and risk metrics.

Further, the alignment with KDM is a bridge between system assessments and the code assessments.

1.5.5 SPECTRA and MITRE D3FEND

MITRE D3FEND is known for its ontology of defensive tactics and techniques. However, D3FEND also provides a large collection of digital artifact definitions—the specific technical elements these cyber products secure or analyze—to model cyber systems and related countermeasures. This creates a foundation for digital engineering and automated analysis about the complex interplay between computer network architectures, threats, and cyber countermeasures, helping security architects understand how a new product will interact with or complement others as part of an integrated network defense.

- SPECTRA can be viewed as contributing to the upper ontology for Digital Artifacts
- SPECTRA is essential for the application of the Digital Artifacts ontology to build a tailored model of a system-of-interest
- SPECTRA adds several missing viewpoints (Architecture viewpoint, supply chain viewpoint);
- SPECTRA adds domains essential for cyber analytics (application, security, maintenance and carrier)

Discussion of the alignment between MITRE D3FEND and SPECTRA can be done in the context of SPECTRA Core Assertions Metamodel, or SPECTRA Ontology.

1.5.6 SPECTRA and RAAML

The Risk Analysis and Assessment Modeling Language (RAAML) Library and Profile (this document) defines concepts and relationships for capturing safety and reliability aspects of a system in the library and profile form.

Model-Based Systems Engineering (MBSE) is gaining popularity in organizations creating complex systems where it is crucial to collaborate in a multi-disciplinary environment. SysML, being one of the key MBSE components, has a good foundation for capturing requirements, architecture, constraints, views, and viewpoints. However, SysML does not provide the constructs to capture safety and reliability information in the system model.

This RAAML 1.0 specification defines extensions to SysML needed to support safety and reliability analysis. It describes:

• the core concepts and shows how the simple concepts are powerful enough to unite all safety and reliability information across a variety of analysis methods

• the approach to automating several safety and reliability analyses, which is built on leveraging existing SysML functionalities to ensure that the profile and library is usable with existing tooling

• specific safety and reliability analysis methods and application domains that are supported

o Failure Mode and Effect Analysis (FMEA)
o Fault Tree Analysis (FTA)
o Systems Theoretic Process Analysis (STPA)
o Goal Structuring Notation (GSN)
o ISO 26262 Road Vehicles - Functional Safety

• extension mechanisms that are typically needed by the industry to apply the specification in practice

The RAAML specification provides the foundation for conducting various safety and quality engineering activities including safety and reliability analysis methods. Besides the method support, linkages to the SysML model-of-interest are provided, enabling integration with and traceability to the analyses. The specification can be used for modeling safety and reliability aspects directly in the model or as a standard language to import and export from external safety and reliability tools.

The organization of RAAML facilitates tailoring the methodologies to specific engineering domains and industries to support the various assessment and certification agencies.

- Apart from the fact that both specifications align with SysML, SPECTRA and RAAML occupy two entirely different spaces and address unrelated needs.
- In relation to digital analytics tools, RAAML describes outputs rather than inputs, while SPECTRA addresses the (some of the) inputs, namely a normalized architectural description of a cyber system, fit for the purpose of subsequent analysis by fully automated tools.
- RAAML is aligned with safety and reliability, while SPECTRA is aligned with cybersecurity.
- RAAML tools may benefit from normalized input models

1.5.7 SPECTRA and CASCaDE RFP

SPECTRA is complementary to the CASCaDE RFP and may in the future provide a gateway to be plugged into CASCaDE as one of the domain-specific ontologies.

CASCaDE stands for Collaborative Artifact, Specification, Context and Data Exchange. CASCaDE is aligned with the manufacturing processes and STeP. With its focus on mechatronic or software products, CASCaDE RFP acknowledges that such products “consists of very many immaterial and material artifacts which are of importance in one or more lifecycle phases. The artifacts are created, detailed using specific authoring systems and reused by various disciplines and organizations within and outside of the enterprise.”. These artifacts are produced in different development phases and by different teams using different tools, from elicitation of user needs to physical design for production, assembly, operation, repair and disposal. There is a need to put the artifacts of all participants and phases with their relations in a common context.

It happens by nature that the same artifact appears in different data sets, making it difficult to propagate changes or even detect dependencies. Therefore, a translation of individual data sets and integration to an overarching Knowledge Graph is required. CASCaDE RFP calls for a meta-model with schema and constraints for the CASCaDE Data Package (CDP) containing product information, i.e. information from multiple sources pertaining to a particular product. The CDP shall lend itself for use by machines and humans.

SPECTRA for SysML v1 considers a single SysML model of a SOI as such model describes some facts of importance to the risk and cybersecurity assessment of the SOI. Because of the nature of SysML, it is difficult to automatically elicit just these facts from the model without a full reverse engineering of the model. Therefore, a markup language is required to unambiguously and effectively identify the core assertions for cyber in SysML models of cyber and cyber-physical systems.

SPECTRA as a family of specifications sharing the core meta-model addresses the need to enumerate the set of core assertions for cyber to have a compliance point for both developing models and the analytics tools that automatically ingest these models. Core Assertions is similar to the Knowledge Graph in CASCaDE.

Core Assertions can be represented in a variety of formats.

SPECTRA also addresses the need for a language-neutral foundation for representing common content for cyber, such as metrics and rules. In this capacity is aligned with the Knowledge Discovery Metamodel.

1.5.8 SPECTRA and DAF Digital Transformation Office Cyber Data Schema

The DAF Digital Transformation Office and AFIT’s Digital Innovation & Integration Center of Excellence, have published the initial open-source release of the Cyber Data Schema (CDS) in September 2024. The CDS was developed to facilitate the digital transformation of cybersecurity assessment and authorization products and processes.

The CDS is a relationally joined database schema (a json format is also available) designed to represent cybersecurity content in a machine-readable and queryable format. It serves as a comprehensive collection of data elements and attributes, specifically tailored to support cybersecurity assessments and authorization content across various platforms and tools.

Interoperability: The CDS enables seamless exchange of cyber content between different organizations, systems, and tools. This feature is crucial for fostering collaboration and maintaining consistency in cybersecurity efforts across various entities.

Data Standardization/Normalization: By using minimally viable elements, the CDS standardizes and normalizes cybersecurity assessment and authorization content. This standardization is essential for reducing redundancy and ensuring that data remains consistent and comparable across different platforms.

Assessment Automation: The CDS paves the way for automation in cybersecurity assessment processes. By providing a common data model, it allows organizations to integrate their assessment toolchains, thereby streamlining the overall assessment and authorization workflow.

DAF DTO Cyber Schema focuses on the outputs of a cybersecurity risk assessment. A small part of the schema describes system information at a high level. This part is aligned with SPECTRA, however SPECTRA covers more technical detail for cyber and cyber-physical systems. Information from SPECTRA can be passed to Cyber Schema.

2 Terms and Definitions

For the purposes of this specification, the following terms and definitions apply.

Term	Definition
MBSE	Model-Based Systems Engineering
MBEE	Model-Based Engineering Environment
SysML	The Systems Modeling Language is a general-purpose modeling language for systems engineering applications. It supports the specification, analysis, design, verification and validation of a broad range of systems and systems-of-systems.
UAF	Unified Architecture Framework
SysML profile	An extension mechanism for SysML v1, based on the underlying UML profile mechanism. Allows extending standard UML metamodel to define custom stereotypes, tagged values, and constraints. It enables users to tailor SysML to their specific domain or application, making it a versatile and adaptable modeling tool. Profiles are particularly useful when you need to create models that capture domain-specific concepts, as they allow you to define new elements and relationships that are not available in standard SysML. In SysML v2, this mechanism is defined as SysML v2 metadata.
SysML metadata	An extension mechanism for SysML v2, similar to a SysML v1 profile.
SOI	System Of Interest

Core Assertions	facts about the structure, behavior and objectives of the system of interest (SOI). Also known as “system facts”
Cybersecurity	Cyber Security is protection of digital information, as well as the integrity of the infrastructure housing and transmitting digital information. More specifically, cyber security includes the body of technologies, processes, practices and response and mitigation measures designed to protect networks, computers, programs and data from attack, damage or unauthorized access so as to ensure confidentiality, integrity and availability.
CRA	Cybersecurity Risk Assessment
Threat-based CRA	A systematic method of conducting a cybersecurity risk assessment driven by identifying and evaluating attack paths for the SOI in a given threat environment.
Analytics tool	tool that ingests a SysML model, possibly other inputs, and provides some useful outputs
Risk	Situation with a specific impact from many similar cyber attacks. A typical system under assessment has many risks.
RMF	Risk Management Framework
Context (of risk assessment)	1) scope, objectives, assumptions, circumstances, environment, background or settings that determine, specify, or clarify the meaning of the core assertions about the system under assessment. 2) the boundary of an assessment of an ingested model, within which elements are considered internal or external.
Digital Information Pathway	a collection of elements that together describe the "geography"[1] of the SOI in terms of "places" (nodes), connectors between them (channels or edges) and some descriptions of the flows of digital information (and material items) between the places
Digital Technical Surface of a system	“Digital technical surface” of a cyber system is the set of replaceable parts and the corresponding communications fabric that describes all digital information pathways in the SOI.
Node	a certain "place" in the digital information pathways of the SOI: a bearer of information, a producer and/or consumer of information flows, a performer of a function, a unit of integration, supply and maintenance, a unit of configuration, a unit of defense by way of

[1] In the meaning “An ordered arrangement of constituent elements”, but also implying the study of some physical characteristics, especially the surface features of the SOI, and the distribution of other features on this surface.

	allocated controls. Some "places" are more tangible, while others are intangible, virtual arrangements of other units.
Tangible	Possible to understand or realize. Discernable. Possible to be treated as a fact; real or concrete. A commonplace understanding of "tangibility" renders it as an attribute allowing something to be perceptible to the senses.
Configuration	An arrangement of parts or elements
Bill of Material (BOM)	A bill of materials or product structure (sometimes bill of material, BOM or indented list of equipment) is a list of the raw materials, sub-assemblies, intermediate assemblies, sub-components, parts, and the quantities of each needed to manufacture an end product. BOMs are of hierarchical nature, with the top level representing the finished product which may be a sub-assembly or a completed item. BOMs that describe the sub-assemblies are referred to as modular BOMs. A multi-level bill of materials (BOM), referred to as an indented BOM, is a bill of materials that lists the assemblies, components, and parts required to make a product in a parent-child, top-down method. BOM records the exact quantities and types of components required for each level of the assembly, from raw materials such as fastenings to sub-assemblies and packing materials. A bill of materials can cover the smallest of assemblies to large machinery and equipment builds. Also known as “Indented list of equipment”.
Assembly	A component assembly is a collection of parts and sub-assemblies that are integrated together to serve a common purpose. An assembly can be a sub-assembly in a larger system or system-of-systems.
Flow	A transfer of items between connected nodes. A continuous or discrete movement that involves shifting some items. The sequence of steps taken to propagate an item between nodes, e.g. produce an item by one node and consume the item by the neighboring node. All unique flows in and out of a node constitute an “interface” of the node. For cyber systems, flows are information flows. Information is usually copied from sender to the receiver. For cyber-physical systems, one is also interested in certain flows of material items.
Thread	Sequence of steps taken in a network of interconnected, collaborating nodes to achieve some result, usually involving a flow of some information items (or a sequence of flows between neighboring nodes).

Integrative level	set of phenomena emerging from pre-existing phenomena of a lower level. The level concept is an intellectual framework for structuring systems. It arranges all entities, structures, and processes in the system into a hierarchy, typically based on how complex their organization is. When arranged this way, each entity is three things at the same time: 1) It is made up of parts from the previous level below. 2) It is a whole in its own right. And 2) it is a part of the whole that is on the next level above. Also related to a hierarchy of assemblies in BOM
Emergence	properties or behaviors that a complex entity's parts do not have on their own and emerge only when those parts interact in a wider whole. Emergence plays a central role in theories of integrative levels and of complex systems.
Organization	A specific real-world assemblage of people and other resources organized for an on-going purpose.
Enterprise	sociotechnical system including people, information, processes, and technologies considered as an organizational unit, organization, or collection of organizations that share a set of common goals and collaborate to provide specific products or services to customers. The term enterprise covers various types of organizations, regardless of their size, ownership model, operational model, or geographical distribution.