34th Annual INCOSE international symposium
hybrid event
Dublin, Ireland
July 2 - 6, 2024

Traceability – A vision for now and tomorrow

Adriana D'Souza
Airbus
Bristol, BS34 7PA, UK
+447793905547
adriana.dsouza@airbus.com

Louis S. Wheatcraft
Wheatland Consulting, LLC
Houston, TX 77062, USA
+1 (832)971-3550
wheatland.consulting@gmail.com

Tami Katz
BAE Systems, Inc.
Boulder, Colorado, USA
+1 303 939-4000
tami.katz@incose.net

A. Larry Gurule
i-Infusion/CMPIC/SAE G33
Fleming Island, Florida, USA
+1 904 215-2044
lgurule@i-infusion.com

Michael J. Ryan
Capability Associates Pty Ltd
Canberra ACT, Australia
+61 481 474479
michael.ryan@incose.net

Aleksander Przybylo
Boeing
Seattle, Washington State, USA
+1 (425) 234-4831
aleksander.przybylo2@boeing.com

Abstract. Traceability has been addressed in the past from the perspective of relationships between the digital artifacts within the data and the information model of the system of interest (SoI) being developed. This paper enhances this view from a project management (PM), systems engineering (SE), and a configuration management (CM) perspective. The paper discusses what traceability is today and how it can help PM and SE practitioners meet future needs to realize the INCOSE Vision 2035 and satisfy all needs and requirements baselined for a SoI, including compliance with standards and regulations. Provenance and pedigree are two aspects discussed of how traceability enhances the management of digital artifacts from a CM point of view. This paper provides a vision of how traceability can aid PM and SE practitioners to develop quality products that deliver what is needed, within cost and schedule, with the needed quality.

Keywords. Traceability, Systems Engineering, Configuration Management, INCOSE Vision 2035

Introduction

Systemic failures in the food (BBC 2013, BBC 2008, The Guardian 2013, Asset Guardian 2018) and cybersecurity (CSO Online 2021) industries have brought the traceability topic to the limelight. Traceability allows products/systems/artifacts and their configuration information, according to current SE guidance, to be made available and referred to precisely. *Note:* Throughout this paper, the term "artifact" refers to "work products that are produced and used during a project to capture and convey information". (ISO/IEC/IEEE 15288:2023)

According to ISO 9000:2015, traceability for products and systems is defined as “the ability to trace the history, application or location of an object/entity/item”. Item is defined as “a nonspecific term used to denote any product, including systems, materiel, parts, subassemblies, sets, accessories, etc.” (MIL HDBK 61B)

Traceability is also defined as a “domain of consideration encompassing the process for determining the provenance of an item. (also referred to as tracking)” (NIST 2022). Provenance is defined by (NIST 2022) as “The chronology of the origin, development, ownership, location and changes to a system or system component and associated data. It may also include personnel and process used to interact with or make modifications to the system, component or associated data.” This, in combination with the SE definition, is what constitutes modern CM as defined in the global consensus standard for CM, SAE EIA 649C. That is, a technical and management process applying appropriate processes, resources, and controls, to establish and maintain consistency between product configuration information, and the product.

Standards and regulations for various industries require traceability to be established across the lifecycle of the product/system; examples include ARP 4754, “Guidelines for Development of Civil Aircraft and Systems”; ISO 13485, “Medical devices - Quality management systems - Requirements for regulatory purposes”; ISO 26262, “Road vehicles — functional safety”; and USC Title 21 Part 820, “Quality System Regulation” for medical devices, traceability references in these are outlined below.

- ARP 4754A Section 5.3.1.1 requires requirements dealing with safety to be “uniquely identified and traceable” to “ensure visibility of the safety requirements at the software and electronic hardware design level.”
- ISO 13485 Section 7.3.2 requires organizations to document “methods to ensure traceability of design and development outputs to design and development inputs.”
- ISO26262 Section 6.4.3.2 requires “Safety requirements shall be traceable with a reference being made to:
    - a) each source of a safety requirement at the next upper hierarchical level;
    - b) each derived safety requirement at the next lower hierarchical level, or to its realization in the design; and
    - c) the verification specification in accordance with 9.4.2.”
- USC Title 21 Part 820 requires developing organizations of medical devices to develop and maintain a Device History File (DHF) that “shall contain or reference the records necessary to demonstrate that the design was developed in accordance with the approved design plan and the requirements of this part.” Traceability is a critical part of the DHF.

Failure to show required traceability can lead to a product not being approved for its intended use in the marketplace. For example, when developing medical devices, the US Food and Drug Administration (FDA) requires developers to identify and manage hazards to users of the device. These hazards represent a risk that must be managed for the FDA to approve the device for its intended use. For those hazards that will be mitigated via design, the FDA requires traceability from risk identification, to needs, to design input requirements, to design, to design output specifications, and finally to the design and system verification artifacts that provide evidence that the realized system meets the requirements. Of critical importance is traceability to system validation artifacts that provide objective evidence that the realized system meets the needs associated with the device when used for its intended use in its intended operating environment as specified by its instructions for use. Without this traceability clearly communicated, the FDA will not approve the device for use.

Overview of Traceability in Systems Engineering

Traceability in SE provides the ability to establish an association or relationship between two or more objects/entities/items and to track objects/entities/items from their origin to the activities and deliverables that satisfy them, as well as assess the effects on artifacts across the lifecycle when change occurs. As addressed in INCOSE Vision 2035 (INCOSE 2021), "Model -Based Systems Engineering (MBSE) descriptive models created using semantically rich modeling standards provide systems abstraction, data *traceability*, separations of view, and leverage Artificial Intelligence/Machine Learning (AI/ML)-based reference model reuse at both systems and product realization levels."

Traceability can be bidirectional or unidirectional and horizontal or vertical.

Bidirectional traceability is the ability to establish a two-way link between entities such that each has knowledge of the other. A trace between entity A to entity B allows entity B to have knowledge of entity A. This capability enables practitioners to move forward, backward, or up and down digital threads that result from establishing bidirectional traceability. A bi-directional relationship facilitates the ability to analyze (not only create) the trace relationships from all traced entities. (Example: If a trace is created from a Need to a System Requirement, when one analyzes the System Requirement, they will be able to see that a trace to the need is present.)

Unidirectional traceability is the ability to establish a one-way trace from one entity to another, where the receiving entity has no knowledge of the source entity. Entity A establishes a trace to entity B, but entity B has no knowledge that entity A has a trace to it. Examples include: A GPS does not know the receivers, but the receivers know about the GPS. Broadcasting is also unidirectional as the receivers know the broadcaster, but the broadcaster does not know its receivers. In engineering applications, a typical example of a unidirectional link is a reference to a Uniform Resource Locator (URL). By simply referencing a published resource by its URL, we do not notify the resource that it is being linked to.

Bidirectional traceability is an inherent characteristic of Requirement Management Tools (RMTs) and some modeling tools, in that once a link is made from one object/entity/item to another, a reverse link is automatically formed. Within language-based modeling tools, relationships between model elements are formally defined as part of the language (e.g., containment, trace, derive, refine, satisfy, verify).

There are SE tools whose sole purpose is to establish traceability between:

- data, information, and artifacts developed and managed in an SE tool (e.g., needs and requirements in an RMT),
- data, information, and artifacts developed and managed in other dedicated tools for specific activities such as risk and test (verification and validation) management,
- models using a language-based modeling tool and

- design artifacts to the actual physical elements.

Interrelationships of these various SE tools are shown in Figure 1. For example, a requirement managed in a RMT is the same requirement that exists in a modeling tool; the two representations are linked to each other via traceability between the tools.

A major challenge for the future as discussed in INCOSE Vision 2025 (INCOSE 2021) is: "Federation across different domain specific tools, and integration of data, are becoming a focus for enabling collaboration and analysis." For the interactions shown in Figure 1, this challenge must be addressed by tool vendors.

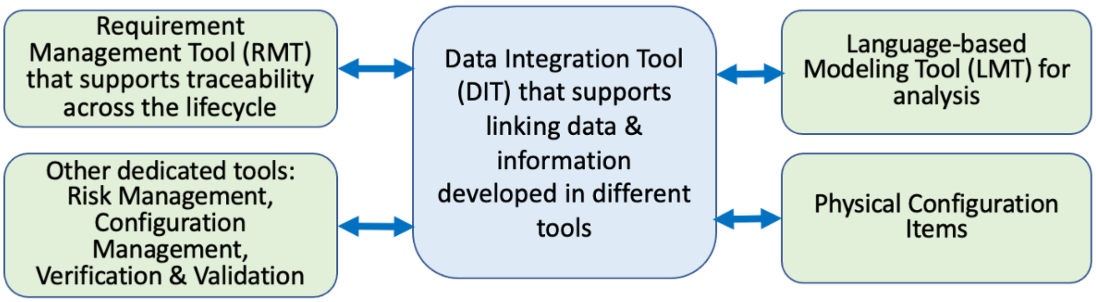


Figure 1: Example Establishing Traceability between data and information from different tools within a projects SE toolset.

Given the importance of traceability to project success, many SE tools enable practitioners to define a traceability relationship meta-model within the tool and then enforce adherence to that model as data and information is entered into the tool, flagging discrepancies when the model is not followed. An example model is shown in Figure 2. Note: For a comprehensive listing of available SE tools, refer to the SE Tools Database (SETDB).

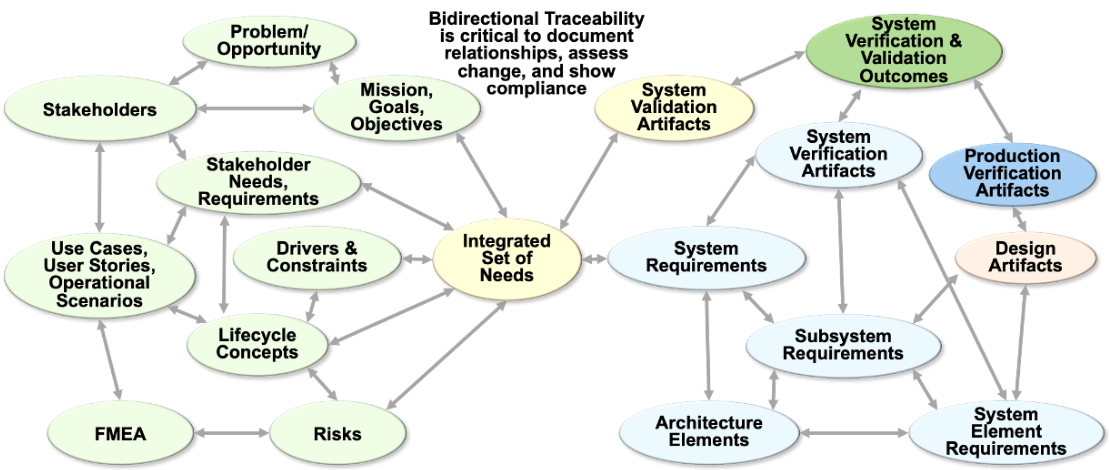


Figure 2: Example Traceability Relationship Model

As shown in Figure 2, traceability can be established between several types of data over the SoI lifecycle.

Horizontal traceability involves the forward and backward traceability between entities across the SoI lifecycle (from concept to retirement). *Horizontal traceability* links data, information, and artifacts generated in one lifecycle process activity to data, information, and artifacts generated in other lifecycle process activities, resulting in a "digital thread" connecting these data, information, and artifacts across the life cycle.

Vertical traceability is most often referred to in the context of levels of organization and architectural levels of the system or product under development. Each level has lifecycle concepts, needs, and requirements defined at various levels of abstraction (INCOSE 2022). Organizational-level business requirements drive and constrain the development of the business operations level lifecycle concepts, needs and requirements. As the business operations level needs and requirements are defined, bidirectional traceability is established with the higher-level business requirements.

Various levels exist from a hierarchical architecture view of a SoI. Higher-level needs and requirements are allocated to the SoI, and bidirectional traceability is established as these requirements are defined. The SoI system level requirements are allocated to the lower-level subsystems. Again, bidirectional horizontal and vertical traceability is established, and the process repeats for requirements across the hierarchy.

Regardless of the level, as requirements are defined, traceability is established with their system verification artifacts. Likewise, during the Architectural Definition and Design Definition Processes, as architectural entities and design artifacts are defined, bidirectional traceability is established with the realized requirements as well as between architectural entities and design artifacts.

Due to the variety and type of traceability, unidirectional and bidirectional as well as horizontal and vertical links can be established. This can be represented differently in various tools, hence, for simplicity the link arrow representation shown in Figure 2 is used to reflect an example traceability relationship model.

Overview of Traceability in Configuration Management

CM as a discipline came about in the 1950s when industry failed to reproduce a successful prototype (Gonzalez, 2002). Hence, CM is rooted in the need to manage changes to an item so that reproducibility is enabled. CM, through its evolution until today, is focused on providing enduring truth, trust and traceability to enterprises and their supply base.

In terms of traceability, in many industries, CM has focused on change tracking, as well as the tracking of part numbers and serial numbers across the production lifecycle. More recently CM expands this view to providing truth, trust, and traceability to the full scope of systems engineering.

Establishing an authoritative source of truth (ASoT) is an important concept in CM. PM, and SE. In a digital environment, an ASoT makes it clear where to find data and information. In the digital world of the future, practitioners have the ability to plan for and manage digital artifacts created, changed and managed in line with agreed processes and rules that are put in place as a "framework for execution". By participating and interacting within this framework, an ASOT can be established (i.e., provenance, traceability, pedigree, and non-repudiation leading to incontrovertibility). Truth allows a path to be chosen. Trust allows practitioners to go down a path with others. SE takes advantage of truth to build trust in the data and information that we have created and are managing.

With evolved CM an integral part of this framework, practitioners will no longer have to worry about one finding the latest information pertinent to their task, role, and associated responsibilities. This is not a concept that has to be waited for. Today, given current CM processes and available enabling technology, just-in-time (JIT) products and their information can be provided with the integrity desired, while assuring the incontrovertibility of work products realized and/or sustained.

Traceability, representing both relations and provenance, is facilitated through the appropriate application of Configuration Management process. More details on CM ‘s vision can be found in D’Souza et al., 2016 paper.

Traceability - Enabling Data-Centric Practice of SE

Historically, organizations have defined, and recorded data and information associated with the various artifacts in the form of “documents”. As systems become more complex and regulated, the sheer volume of documentation has become overwhelming; especially in terms of configuration management, change control, completeness, correctness, and consistency. Because of this complexity, there are more people involved in the development of these systems spread over different geographical locations. This results in many of the documents being developed and managed within silos with limited collaboration.

Because of these issues, it can be arduous to keep all the data and information contained within the various documents in sync, current, correct, and consistent causing challenges in locating the ASoT. For highly regulated systems, the amount of documentation that must be developed, maintained, and supplied to regulators to show compliance has become a major burden. Inconsistencies in these documents can result in a SoI that fails system validation and is not approved for use. In some cases, failure to show compliance can result in significant penalties to the developing organizations.

The old, 20^{th} century, document-centric approach is no longer effective for many of today’s systems of the 21^{st} century. Because of this, organizations are moving to a data-centric practice of SE. In a data-centric approach, much of the data and information is captured electronically in an integrated data and information model that represents both the SoI under development as well as the SE artifacts generated across the system lifecycle.

In the past, from a requirements perspective, traceability was limited to establishing vertical traceability (linkage) between a lower-level child requirement back to its higher-level parent requirement (child-to-parent), a requirement to its source from which it was derived, or a set of related requirements to each other (peer-peer). Using a data-centric approach, the focus is on vertical and horizontal traceability between artifacts generated across the lifecycle.

Establishing Traceability

When establishing horizontal traceability, one challenge for today’s world is how to ensure traceability when configuration management structures are siloed and managed in different locations or tools. In an environment with siloed tools and structures with no data exchanges (links defined), it is difficult to ensure the pedigree and provenance of data. More often than not, practitioners ensure continuity of data flow by the manual transfer of information. The pace of development for various components, systems, and services are not often aligned, i.e., there is no clearly defined ASoT, highlighting the need for CM to align these artifacts for the integration of the SoI.

A key starting point for establishing and managing traceability is determining the meta-model i.e., the logic between the data. This can be defining what data is to be managed within the project toolset and the links between these types of data. On a mature project however, this activity may be performed to better understand the existing meta-model (objects type and how they link to each other).

The individual sets of lifecycle concepts, needs, design input requirements, design output specifications, system validation artifacts, system verification artifacts do not exist in isolation, rather they represent a multi-level, multi-dimensional web of relationships. These relationships are documented via links that allow the relationships to be traced between the entities that are linked both vertically across levels and horizontally across the lifecycle as shown in Figure 2.

There are various types of traceability links, including:

- *Abstraction* - a link connecting representations of a concept at different levels of detail. Subtypes of abstraction include derivation, refinement, decomposition, elaboration.
- *Allocation* - a link typically connecting elements across two domains (a requirement allocated to a function, a function allocated to a logical element, or a logical element allocated to a physical part). Satisfaction, Performance and Implementation are inverse relationships of specific allocation types (e.g., a function satisfies a requirement, a logical item performs a function, a part implements a logical item, etc.).
- *Association* - a generic purpose relationship establishing a link between two elements. Associations are used when there is no lifecycle dependency or hierarchy to be inferred.
- *Composition*- a link connecting an assembly to its parts, a system to its subsystems, a function to its sub-functions etc.
- *Dependency* - a link indicating an impact to the dependent element.
- *Substantiation* - a link connecting rationale (e.g., an analysis artifact) to a statement (e.g., a requirement being satisfied). Verification and validation are processes that establish substantiation.

Establishing traceability is critical to project success enabling more effective management of the SoI and SE artifacts across the lifecycle. A large part of this management is CM. Establishing traceability within a project requires several key steps:

1. *Planning*. The project must define at the beginning of the project the approach to managing data, information, and artifacts within a centralized database as well as the list of tools the project is going to use. The tools used need to support the concepts of data sharing, centralized management, and the ability to do and manage traceability as discussed in this paper. A traceability and data meta-model showing how the project plans to manage traceability between tools as shown in Figure 1 must be clearly defined.

2. *Implementation*. Within the ASoT enabling capability (people, process, and tools), the traceability and dependence meta-model must be implemented. Each artifact within the model must be defined and a set of rules defined for establishing and managing traceability among the artifacts. A key part of this is the use of the rules to manage traceability.

   As a result, the process and tools inform the person inputting data of the traceability rules so that proper traceability can be established. This capability also enables the management of traceability by informing those accessing the data when a traceability rule has not been followed. (e.g., a required link between artifacts has not been established as defined by the rules).

3. *Management*. It is extremely important that traceability is established and maintained properly per the established traceability rules. If not, the ability to use traceability to better define and manage the SoI is compromised. Assuming the capabilities to address rule violations, verification activities should be defined to monitor the proper traceability and ensure that the person responsible for an artifact is notified of the defect and the defect is corrected.

Digital Thread

Managing system development from a data-centric perspective expands the reach of traceability, enabling data and information, including requirements, to be linked across all lifecycle stages as shown in Figure 2. This results in digital threads linking related information across the lifecycle. Multiple types of artifacts can be connected to each other where the trace relationship has specific meaning (e.g., requirements are derived from user needs; models are derived from requirements but also, in turn, models elicit further requirements as well).

As stated in the INCOSE SE HB v5: "*The digital thread establishes communication paths between the individually configured domains. It is also responsible for correctly tying together the appropriate configurations in each domain and to form a consistent configuration for a specific system/product and their elements.*

A digital thread is a set of interconnected, cross-discipline model data that seamlessly expedite the controlled interplay of digital artifacts to inform decision makers throughout a system's life cycle. Digital threads can be used to produce digital artifacts that are a combination of authoritative data, information, knowledge, and wisdom addressing stakeholders' unique perspective."

Digital threads represent the associated information like materials, but also drawings and any lifecycle artifact information. The notion of configuration item (CI) as a large collector of data representing an end-item comes from a time when the overhead of CM processes was too overwhelming and the automation capabilities of CM platforms too limited. Today, this is no longer the case, which is why the idea that a physical item is reduced to a single CI is no longer true. Rather, the configuration of an end-item is defined as a set of individually configuration managed fine-grained elements. What needs to be focused on is the digital artifacts associated with a system, system element, artifact, service, or process that configuration management is applied to.

Once the meta-model has been established and the logical links between model entities have been defined, an end-to-end path, often referred to as a traceability path or digital thread, is established. The traceability path can be represented in the Model Based Engineering (MBE) Diamond, as illustrated in the following example of a change to a specific domain and its effects.

1. Changes to needs and requirements and interfaces drive changes to other needs and requirements in the "as-specified" domain as well as to the architecture models in the "product model" domain. This illustrates a vertical inter-quadrant configuration management scenario.
2. Changes to Computer Aided Design (CAD) models in the "physical models" domain are identified through impact analysis on the architecture model changes. This illustrates an intra-quadrant configuration management scenario.
3. Validation of accuracy against design changes on product model is enabled within the "virtual certification" domain. This illustrates a horizontal inter-quadrant configuration management scenario.

Throughout this scenario, traceability is maintained from requirement to certification data set as illustrated on Figure 3

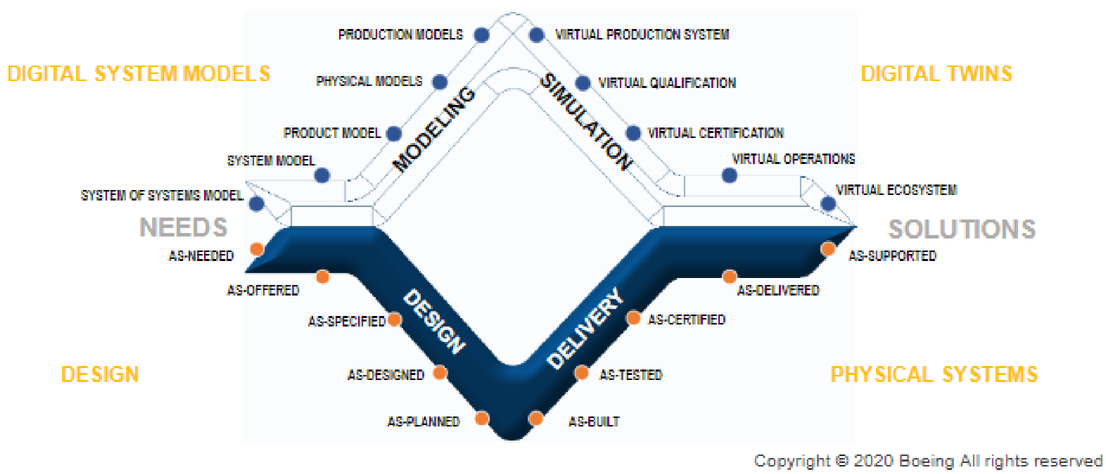


Figure 3: The MBE Diamond.

Traceability is a crucial component of the digital thread, enabling the linking of the data under CM. In turn, the digital thread is also a critical part of CM. A customer requirement can be traced through the functional and logical representations all the way to the physical SoI, thus enabling the identification of specific physical assets (and their specific configurations) that are impacted by a change to a given requirement. Vice versa, traceability identifies the requirements that need to be verified when a given physical asset is modified.

As discussed previously, in the medical device industry, digital threads are critical to acceptance of a device in terms of the intended use, risks, and compliance. During acceptance, the FDA expects to see a clear path from the identification of a risk that can cause harm, to the design inputs related to mitigating a risk, to the design outputs that result in the risk being mitigated, to the verification and validation artifacts that provide objective evidence that the risk has been successfully mitigated per the approved mitigation plan.

Similarly, in the aerospace industry, the Federal Aviation Administration (FAA) expects the Original Equipment Manufacturers (OEMs) to provide full traceability of data across the entire Requirements-Functions-Logical-Physical (RFLP) spectrum, including direct relationships to all substantiating analyses that provide rationale for the design and modeling of the product.

Cybersecurity has moved to the forefront of SE. Newly released National Institute of Standards and Technology (NIST) SP 800-160v1r1, Engineering Trustworthy Systems, addresses cybersecurity across all lifecycle process activities emphasizing traceability and the establishment of a digital thread across the lifecycle from a problem statement, to a mission statement, goals, objectives, measures, mis-use cases, loss scenarios, risks, needs, lifecycle concepts, needs, requirements, architecture, design, realization, verification, validation, operations, maintenance, and disposal. The term “traceability” can be found on 25 of the 207 pages of the standard.

Digitally enabled traceability methods are a major capability enabled by a data-centric practice of SE that enables SE practitioners to ensure the customer and regulatory agency needs are met. Digitally enabled traceability can even trigger further needs enabled by the transparency of information. Continuous assurance and command platforms are an enabler for techno-mechanical products where software, AI, and autonomous operation require broad range traceability for non-deterministic operation.

Capitalizing on Traceability

Traceability provides CM, PM, and SE practitioners the ability to more effectively manage products and their associated SE artifacts across their lifecycle. Proper and maintained traceability enables several key functions that are key to successful product development. The work done by (Roedler et al, 2010) is great for inspiration on indicators, however, a further step can be taken to connect the dots in an end-to-end perspective.

1. *Helping to ensure completeness and correctness of the data model*. The use of the rules discussed above helps to avoid missing information in the dataset. For example, a parent requirement has been allocated, yet there are no child requirements defined for the subsystem or system element to which the requirement was allocated; or child requirements that do not trace to a parent or source.
2. *Aiding in effective change control and change impact analysi*s. A key part of change control and change impact analysis is the ability to know whether or to what extent a proposed change impacts other artifacts within the traceability and dependency model. How does a change to a high-level requirement affect lower-level requirements; how does a change to a requirement affect other requirements for other system element requirements at the same level? How does a change to a need ripple across the lifecycle to implementing requirements, design, design output specifications, the realized system, verification, and validation? How does a change to design impact the design input requirements implemented by the design?
3. *Supporting Risk Management*. A key activity in SE at the beginning of a project is identifying risk, assessing the various risks, deciding which risks are going to be mitigated,

and developing mitigation strategies. For those risks that are going to be mitigated by the design of the SoI, traceability across the lifecycle is critical, especially for highly regulated products like medical devices, to be able to clearly show objective evidence (via verification and validation activities) that those risks have been successfully mitigated via the needs, requirements, design, design output specifications, and by the realized product.

4. *Ensuring compliance with higher level needs and requirements*. Higher-level requirements are allocated or assigned to the system, subsystems, or system elements. There is an expectation, as discussed above, that the project supplies objective evidence that these higher-level needs and requirements have been adequately addressed.
5. *Ensuring compliance with standards and regulations*. Likewise, there is an expectation that the project supplies objective evidence that applicable requirements within standards and regulations have been adequately addressed. Failure to show this objective evidence can result in regulatory agencies failing to approve the product for its intended use.
6. *Managing verification and validation across the lifecycle*. Verification and validation are continuous activities that occur across the lifecycle of key artifacts including needs, design input requirements, design, design output specifications, the realized system, as well as the system in operations. Verification and validation involve artifacts being generated associated with the planning, execution of the verification and validation activities, and the resulting data that is used to provide objective evidence that the activities were successful in showing the artifact met the needs and requirements it was verified and validated against. To do this, traceability must be established and managed between all these artifacts across the lifecycle.
7. *Supporting interface management activities*. For successful interface management, traceability is used to link interface requirements to other interface requirements involved in a given interaction across an interface boundary as well as to link those requirements to a common definition of the specific interaction. Establishing traceability rules for interface requirements within the data set is critical to successful interface management. Based on the rules it can be determined if there are missing links, thus allowing these defects to be addressed. Failing to do so can result in a defective design and failed system integration, system verification, and system validation.

In support of these activities and to have an accurate overview and status of how the development of an SoI is progressing, key performance indicators need to be established that communicate to management the progress of the project as well as any issues that may exist. These indicators enable management to take actions addressing the issues earlier in the lifecycle in order to enable delivery of a SoI that will be accepted by both the customers and regulatory agencies.

Traceability enables this information to be defined, maintained, and reported. Any gaps in traceability, i.e., failure of a project to define, implement, and manage the traceability and dependency model, prevents this information from being accurately reported to management. As a result, key issues may not be identified, which can result in significant schedule slips and cost overruns as well as the project not being able to show compliance and thus fail to be accepted for its intended use. Because of this, the traceability and relationship meta-model itself should be a key part of CM activities.

Conclusion

This paper provides practitioners insights concerning traceability from both a systems engineering and a configuration management perspective. The paper discusses what traceability is today and how it can help CM, PM, and SE practitioners meet future needs to realize the INCOSE Vision 2035 and satisfy the needs and requirements for a SoI, including compliance with standards and regulations. It provides a vision about how traceability can aid PM and SE practitioners to develop quality products that deliver what is needed, within cost and schedule and with the needed quality.

The role that traceability and the digital thread have in project management and systems engineering is discussed in practical terms; illustrating, with examples, the need to establish traceability between digital artifacts. Secondly, the authors outlined how to capitalize on the traceability to better manage system development across the SoI lifecycle.

References

ARP4754A, Aerospace Recommended Practice (ARP) ARP4754A (Guidelines for Development of Civil Aircraft and Systems), SAE International

Asset Guardian 2018, "Cyber Security and Disaster Recovery in the Food and Beverage Industry" https://www.assetguardian.com/cyber-security-and-disaster-recovery-in-the-food-and-beverage-industry/

BBC 2013, "Findus beef lasagne contained up to 100% horsemeat, FSA says" https://www.bbc.co.uk/news/uk-21375594

BBC 2008, "Toxic milk toll rockets in China" http://news.bbc.co.uk/1/hi/world/asia-pacific/7616346.stm

CSO Online 2021, "The biggest data breaches of the 21st Century" https://www.csoonline.com/article/2130877/the-biggest-data-breaches-of-the-21st-century.html

D'Souza, Adriana, Kossmann, Mario, Watts, Stephen 2016, "The enabling role of Configuration Management for the Systems Engineering of tomorrow's complex Systems and Systems of Systems", July 2016, INCOSE International Symposium 26(1):2180-2194, DOI:10.1002/j.2334-5837.2016.00288.x

Food and Drug Administration, USC Title 21, Part 820, Quality System Regulation.

Gonzalez, P. J. 2002. *A Guide to Configuration Management for Intelligent Transportation Systems.* By Mitretek Systems, Inc. US Department of Transportation. http://ntl.bts.gov/lib/jpodocs/repts_te/13622.htmlhttps://rosap.ntl.bts.gov/view/dot/37371

ISO/IEC/IEEE 15288:2023, "Systems and software engineering — System life cycle processes" 16 May 2023.

INCOSE 2023, "Systems Engineering Handbook", 5th Edition, INCOSE–TP–2003–002–05 2023

INCOSE 2021, "Systems Engineering Vision 2035"

INCOSE 2022, "Needs and Requirements Manual", V1.1, INCOSE-TP-2021.002-01, May 2022.

ISO13485: 2016, Quality Management for Medical Devices

ISO 26262-1:2018, Road vehicles - Functional safety

MIL HDBK 61B, Configuration Management Guidance

NIST 2022, "Computer Security Resource Center", https://csrc.nist.gov/glossary/term/Provenance

NIST 2022, "Engineering Trustworthy Systems", November 2022, NIST.SP.800-160v1r1 Available at https://doi.org/10.6028/NIST.SP.800-160v1r1

Roedler, G., Rhodes, D.H. , Schimmoler, and Jones, C. (eds). 2010. System Engineering Leading Indicators Guide, version 2.0. Published jointly by LAI, SEARI, INCOSE, and PSM. INCOSE-TP-2005-001-03. Accessed on September 14, 2011. Available at http://seari.mit.edu/documents/SELI-Guide-Rev2.pdf

The Guardian 2013, "China arrests fake meat scandal" https://www.theguardian.com/world/2013/may/03/china-arrests-fake-meat-scandal

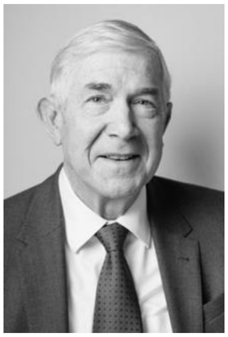
Biography

Adriana D'Souza. (CSEP) is a Configuration Management Process Architect for Systems for Airbus, having previously worked as a Systems Architect and as a Design and Development Engineer on challenging and complex projects like large border security projects and air traffic management projects both at the subsystem, system and system of systems level in the Airbus Group. She was awarded an Honours M.Sc. in Computational Science and Engineering from the Technical University of Munich (Germany) and a B.Sc. in Mathematics and Computer Science from the Ovidius University of Constanta (Romania). Adriana is also a Certified Systems Engineering Professional with INCOSE.

Lou Wheatcraft. has a BS degree in Electrical Engineering from Oklahoma State University; an MA degree in Computer Information Systems; an MS degree in Environmental Management; and has completed the course work for an MS degree in Studies of the Future from the University of Houston – Clear Lake, Texas. Lou has over 50 years' experience in systems engineering, including 22 years in the United States Air Force specializing in the development of space systems and is a senior consultant and managing member of Wheatland Consulting, LLC. Lou is active in the INCOSE technical operations organization as the Chair of the Requirements Working Group.

Tami Katz. (ESEP) received a Ph.D. in Systems Engineering from Colorado State University, and is an INCOSE Expert Systems Engineering Professional (ESEP). She has been working in the aerospace and defense industry since 1990, and is currently a Staff Consultant at BAS Systems. Dr. Katz is active in the INCOSE technical operations organization as the Deputy Technical Director and the Co-Chair of the Requirements Working Group.

A. Larry Gurule. Mr. Gurule is the President of CMPIC LLC, lead instructor, and author of CMPIC's course material. As President of i-infusion, Inc. he has decades of experience in configuration management, including consulting, assessing, and teaching on the subject matter to a wide variety of both commercial and government organizations across the globe. Mr. Gurule is an active member in the CM community and is the current Chairperson of the global SAE G-33 Configuration Management committee. Larry is also advancing CM harmonization through his active participation with SAE International, IEEE, ISO/TC 176/SC2, INCOSE, and ASME Committees.



Michael J. Ryan. holds BE, MEngSc and PhD degrees in electrical engineering. He is a Fellow of Engineers Australia, a Senior Member of IEEE, a Fellow of the International Council on Systems Engineering, a Fellow of the Institute of Managers and Leaders, and a Fellow of the Royal Society of NSW. Since 1981, he has held a number of positions in communications and systems engineering and in management and project management. From 1998 to 2020, he was with the University of New South Wales, where he was most recently the Director of the Capability Systems Centre. He is currently the Director of Capability Associates Pty Ltd. He is the Co-Chair of the Requirements Working Group INCOSE. He is the author or co-author of thirteen books, four book chapters, and over 400 technical papers and reports.

Aleksander Przybylo. System integrator and architect at Boeing. Specializes in Model Based Systems Engineering, Configuration Management and Data Analytics. Successfully led multiple enterprise wide deployments of software solutions for data integration, configuration management, graph-based analytics and integration platforms and data distribution systems. Holds master's degrees in Mechanical Engineering and Computer Science.