NASA/TP-20240003341

Decision Space Modeling: Trade Space Ontology

Kevin A. Reilley
Marshall Space Flight Center, Huntsville, Alabama

Isabeta Rountree
KODA Technologies, Huntsville, Alabama

Stephen J. Edwards
Marshall Space Flight Center, Huntsville, Alabama

April 2024

NASA STI Program Report Series

The NASA STI Program collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI program provides access to the NTRS Registered and its public interface, the NASA Technical Reports Server, thus providing one of the largest collections of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA Programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.

- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.

- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.

- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or co-sponsored by NASA.

- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.

- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services also include organizing and publishing research results, distributing specialized research announcements and feeds, providing information desk and personal search support, and enabling data exchange services.

For more information about the NASA STI program, see the following:

- Access the NASA STI program home page at http://www.sti.nasa.gov

- Help desk contact information:

https://www.sti.nasa.gov/sti-contact-form/ and select the "General" help request type.

NASA/TP-20240003341

Decision Space Modeling: Trade Space Ontology

Kevin A. Reilley
Marshall Space Flight Center, Huntsville, Alabama

Isabeta Rountree
KODA Technologies, Huntsville, Alabama

Stephen J. Edwards
Marshall Space Flight Center, Huntsville, Alabama

National Aeronautics and Space Administration
Marshall Space Flight Center Huntsville, Alabama 35812

April 2024

Available from:

NASA STI Information Desk
Mail Stop 148
NASA Langley Research Center
Hampton, VA 23681–2199, USA
757–864–9658

This report is also available in electronic form at
<http://www.sti.nasa.gov>

Disclaimer

EXECUTIVE SUMMARY

As the National Aeronautics and Space Administration (NASA) works to develop a crewed Moon to Mars Architecture, it is dealing with a large decision space consisting of the overlay of human exploration architectures for both the Moon and for Mars. Efforts are underway to enable reasoning, analysis, and deliberation on this decision space. A critical first step is to develop a model of the decision space, which will then allow for various methods and techniques to be applied in support of the larger architecture decision-making process.

The Trade Space Ontology consists of a set of terminologies and relations (an ontology) and a MagicDraw resource that enables documentation of decisions and alternatives. It also provides a means by which decisions and alternatives can be traced to other Systems Engineering artifacts. For documenting alternatives, the Trade Space Ontology adapts the Morphological Matrix methodology to The Systems Modeling Language (SysML) through a profile; custom diagrams are also implemented to simplify the profile's use. With the profile and custom diagrams, system architects can specify options for architecture attributes, as well as compatibility between them, in a compact visual format. While the approach shares similarities to a trade tree, the emphasis at this stage is less on enumerating specific combinations of options and instead on specifying the options and their compatibility. Enumeration of alternatives is performed by an external analysis that operates on an output file from a model constructed using the Trade Space Ontology.

For decisions, the Trade Space Ontology provides a way to model generic precedence relationships as well as documenting inputs and outputs. These may include what alternatives, criteria, and rationale are understood to be relevant for each decision. Importantly, the decision-making side of the Trade Space Ontology is defined at a more general level, such that it can be adapted to the specific terms in use by projects and programs at NASA. However, this adaptability also means that less capability is provided "out-of-the-box" from installation. Currently the resource includes plugin functionality to enumerate paths through generic precedence relationships between decisions and to export these paths to a spreadsheet. Custom dependency stereotypes are included in the profile to indicate the cross-cutting relationships between the trade space and the architecture decisions, providing a means to map which parts of the trade space enumerate alternatives for a decision, and to identify how the output of a decision may modify the trade space through pruning or down-selection. While the motivating use case for this resource is in human exploration architectures, the broad applicability of the Morphological Matrix methodology indicates that the Trade Space Ontology should also be useful for other activities and tasks at the agency.

Table of Contents

List of Tables

List of Figures

Acronyms

CPACS Common Parametric Aircraft Configuration Schema
DoE Design of Experiments
DRA 5.0 Design Reference Architecture 5.0
IRMA Interactive Reconfigurable Matrix of Alternatives
JSON JavaScript Object Notation
JSON-LD JSON for Linking Data
M&S Modeling and Simulation
MBSE Model-Based Systems Engineering
NASA National Aeronautics and Space Administration
OSLC Open Services for Lifecycle Collaboration
OWL Web Ontology Language
RIDM Risk Informed Decision Making
S&MA Safety and Mission Assurance
SE Systems Engineering
SysML The Systems Modeling Language
TDD Trade Definition Diagram
TRM Trade Relation Map
TSO Trade Space Ontology
UML Unified Modeling Language
XML Extensible Markup Language

1 Introduction

The National Aeronautics and Space Administration (NASA) Systems Engineering Handbook states that Systems Engineering (SE) "is a way of looking at the 'big picture' when making technical decisions." [1] Decision-making, therefore, is at the heart of the SE process, whether in the earliest stages of formulation or in the final stages of implementation. It is useful to look at a succinct and general description of decision-making:

> *"given a set of alternatives, choose a feasible alternative, which according to decision making circumstances is the most preferred."* [2]

From this description three key decision-making concepts are observed. First, the "set of alternatives" implies there are multiple possible decision outcomes; there is no decision to make in the trivial case of a set size of one. Second, the word "feasible" allows for the existence of constraints that may prevent the selection of some subset of alternatives. And thirdly, the concept of preference indicates the presence of multiple objectives, the weighting of which is driven by the "circumstances."

The methodology discussed here is aimed at addressing the first and second of these decision-making concepts with the rigor provided by formal modeling. This publication details the Trade Space Ontology (TSO) developed, which provides two interconnected components to assist in exploring alternatives and in decision-making: a means to represent trade spaces and a means to represent decision spaces. The approach focuses on structuring data related to these problems in The Systems Modeling Language (SysML), providing traceability between decisions and the trade space as well as other SE artifacts, and assisting in the enumeration of alternatives. Additionally, functionality is included to simplify the process of constructing SysML models to define trade spaces.

1.1 Exploration Architecture Decision Spaces

Motivation for this work is found in the challenge of formulating and implementing human exploration architectures. As architecting for human exploration is a system-of-systems problem, the decision space is very large and hierarchically-nested in structure. There is also significant coupling across the decision space, consisting of both logic-based constraints and physics-based sensitivities. Connolly et al demonstrate both the scale and the coupling for a human Mars mission. [3] Their estimate on the number of alternatives present in the decision space is on the order of 10^{37}. While this quantity would increase with further refinement by adding options not captured in the study, it is already large enough to show that a brute-force analysis of alternatives is intractable. An analysis which evaluates each alternative at 1 alternative per second would require $3\text{x}10^{29}$ years to evaluate all alternatives, more than a billion billion times the estimated age of the observable universe. [4] Similarly large decision spaces have been identified for lunar exploration architectures. [5]

With NASA's work to develop a Moon to Mars Architecture, the agency is dealing with the combined decision space consisting of the overlay of human exploration architectures for both the Moon and for Mars. [6] As part of this endeavour, efforts are underway to enable reasoning, analysis, and deliberation on this decision space. A critical first step is to develop a model of the decision space, which will then allow for various methods and techniques to be applied in support of decision-making. The TSO provides important capability for constructing this decision space model.

1.2 Relevant Standards

Several existing standards relate to the TSO effort.

1.2.1 Relationship to NASA-STD-7009

NASA Standard 7009, Standard for Models and Simulations [7] requires that Modeling and Simulation (M&S) activities "document the rationale for the setup and execution of the simulation and analysis", where setup may include an "explicit scenario definition" including a Design of Experiments (DoE). From Kerzhner [8] and Sharma [9], a design of experiments typically follows an enumeration of discrete architectural alternatives. Thus, the core functionality of the TSO would help with documenting setup of modeling and simulation, prior to a design of experiments, by giving a means to structure the enumeration of alternatives. However, it is possible that some analysis can be performed on the data authored using the TSO, especially regarding decision networks. Modeling and simulation that uses the data defined by a decision network as an input may be subject to 7009. SysML content authored using the TSO (i.e. SysML models which apply the TSO profile) with the functionality described by this report is not an analysis in itself, but serves as an aid for setting up analysis or possibly as an input to analysis.

1.2.2 Relationship to Existing NASA Handbooks

The NASA Systems Engineering Handbook [1] and Risk Informed Decision Making (RIDM) Handbook [10] both discuss enumerating feasible alternatives, and are addressed in detail in later sections. SysML models authored using the TSO may constitute Engineering Model-Based x (MBx) Models and Associated Data under NASA Handbook 1004, NASA Digital Engineering Acquisition Framework Handbook [11]. SysML diagrams authored using the TSO may constitute portions of the technical solution view product "Soln-2 Analysis of Alternatives (AoA)" as defined by NASA Handbook 1005, NASA Space Mission Architecture Framework (SMAF) Handbook for Uncrewed Space Missions [12]. The Soln-2 view product "defines the trade space for the operational, functional, and physical architectures" [12]. Application of the TSO may also help provide additional detail and rationale to refine relationships between functions and requirements or allocate relationships between system elements and functions discussed in NASA Handbook 1009, NASA Systems Modeling Handbook for Systems Engineering [13].

1.2.3 Proposed Standards

Ontologies have been proposed in recent years to assist with space systems design and Model-Based Systems Engineering (MBSE). Hennig et al [14] illustrate how ontologies can play the role of a system model in standardizing and facilitating data exchange between other engineering models in a commercial context. Some governments are also investigating the applications of ontologies to space systems [15]. Partnerships between NASA and academia are also investigating vocabularies to cover space systems [16].

1.3 Ontology at NASA

NASA currently uses ontologies or is investigating deploying them for several different purposes. Rovetto et al [17] list several potential functions of ontologies at NASA. Specific ontologies and applications listed here are examples to illustrate key points regarding functionality, but are not an exhaustive list.

1.3.1 Knowledge Management

One application of ontologies is knowledge management, where the ontology is a model of domain knowledge. This application often sees ontology modeling languages playing a similar role to MBSE languages such as SysML in defining systems, behavior, or requirements. Johnson et al [16] investigated an application of this kind, as previously mentioned, to attempt to describe all space systems. Berrios et al [18] created an ontology describing the domain of radiation biology. O'Neil and Rovetto describe a process to create applications based on data models described by ontologies, applied to both web applications in JavaScript and desktop applications using the language R [19] – their approach has the ontology serving as both a model of the domain knowledge and, as discussed in the next section, an interface for software tools. The translation of the ontology model into a format such as JavaScript Object Notation (JSON), but more specifically JSON for Linking Data (JSON-LD), allows it to serve as a basis for a common data interface as well as a model of the domain of interest.

1.3.2 Data Interface

For the application of ontologies to data and software interfaces, there are several different approaches. One application to desktop software was described by Feyock [20] to specify application interfaces. In contrast, Tissot and Menzel [21] describe infrastructure and benefits of using ontologies for data integration across networks for managing data consistency. The networked solution has been realized in several different examples. The conceptual model for NASA Air Traffic Management by Keller was intended for use with ontology technologies such as a triple-store database, with which web applications could communicate and share data via the common data model [22]. The previously described application by O'Neil and Rovetto also can support communication of interoperable networked applications, as shown by the web application example [19]. An important set of vocabularies in use via commercial software tools are Open Services for Lifecycle Collaboration (OSLC), which have proven useful for programs such as the Mars Ascent Vehicle [23] and which specify data interfaces that are intended to be largely hidden from users [24].

1.3.3 Ontology applied to SysML Modeling

An important application of ontology at NASA has been in the definition of terms for the Europa mission [25] and the State Analysis methodology [26, 27]. Many of these terms were based on a mapping between the Web Ontology Language (OWL) and Unified Modeling Language (UML)/SysML [28–30], with the result for SysML users being SysML profiles, libraries, and modeling patterns to assist performing Systems Engineering activities as seen in Bayer et al [25] and Castet et al [27]. Much of the publicly available work on this topic has shifted emphasis, with the creation of the Ontological Modeling Language, the open-CAESAR project, and the development of SysML Version 2 [31]. At the present moment, many ontology tools are looking forwards towards drafts of SysML Version 2, while much commercial application of MBSE continues to rely on SysML 1.x versions.

2 Trade Space

2.1 Background

The primary tool for enumerating alternatives at the NASA is the trade tree. The NASA Systems Engineering Handbook defines a trade tree as "A representation of trade study alternatives in which each layer represents some system aspect that needs to be treated in a trade study to determine the best alternative" [1]. The intended use of the trade tree is to down-select alternatives prior to running a potentially costly analysis. The Design Reference Architecture 5.0 (DRA 5.0) [32] provides a clear example of a heritage architecture with a trade tree for human exploration of Mars – noting that DRA 5.0 has not been selected for crewed Mars missions.

The DRA 5.0 trade tree provides an illustration of both the benefits and downsides of using this kind of visualization to represent alternatives. In its favor, alternatives are presented hierarchically starting from a goal or objective, and cascade down to more specific decisions. An individual alternative solution is defined by a path from the root or objective, to a leaf, so that the number of alternatives under consideration is equal to the number of leaves. This improves readability of the decisions being made for different functionalities. For example, the issue of "combinatorial explosion", or rapid growth in the size of a problem to the point of computational intractability [33, 34], of the trade space is clearly illustrated with the repetition of functional alternatives within a layer of the tree. However, within these benefits are several important downsides. Firstly, the vertical spread of the trade tree limits the number of functionalities that can be considered in a single visualiztion. Second, the number of options per functionality also has a similar readability limit, as seen in the horizonal vs. vertical orientation transition for the DRA 5.0 Interplanetary Propulsion functionality. Further, the hierarchical nature of a trade tree can give a potentially incorrect perception of the sequence of decisions for choices which might be made in any order from the perspective of conceptual design – noting that a potential benefit of the ordering in a

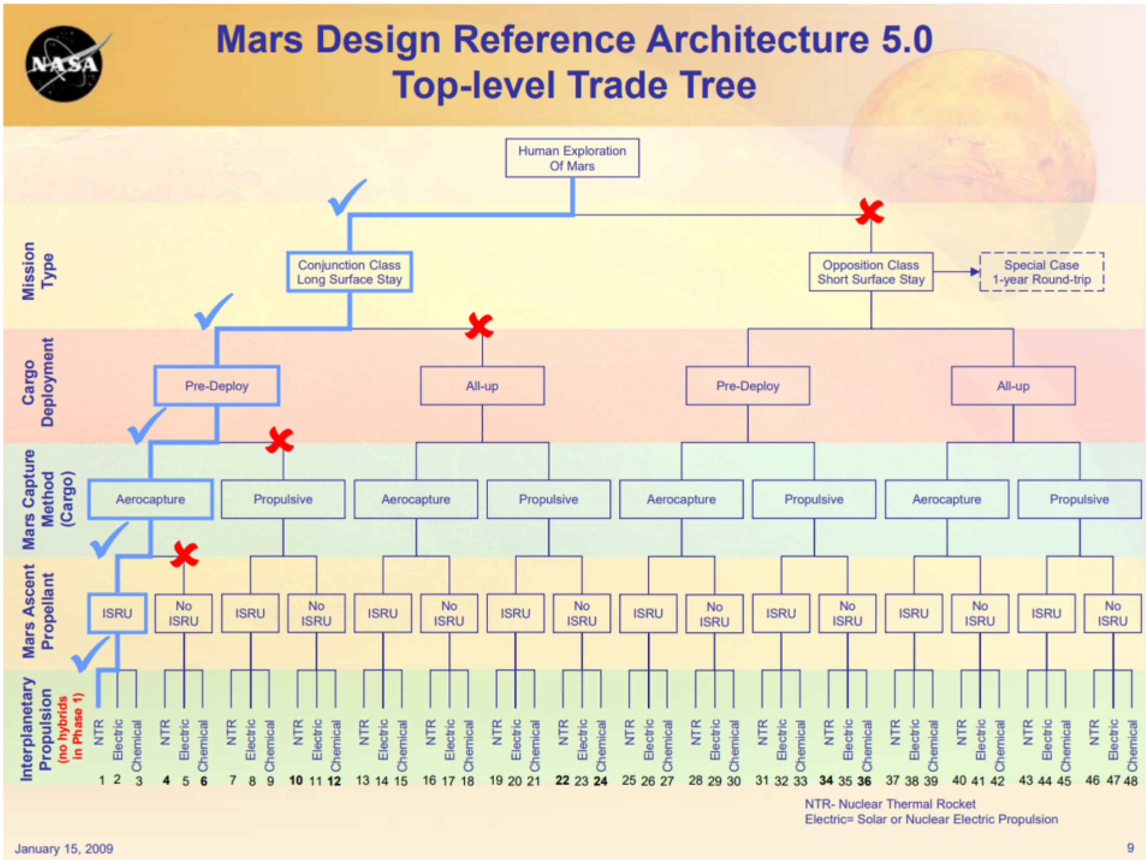

Figure 2.1: Example trade tree and traversal showing selections from the DRA 5.0

trade tree would be to document the sequence in which decision makers will or have reviewed specific selections. Finally, the trade tree does not provide any direct representation of compatibility or incompatibility between individual solution alternatives. Instead, users of trade trees are meant to "prune" the tree according to known compatibilities and incompatibilities, removing alternatives by effectively cutting branches off the tree. [10] To solve these issues, outside of the NASA community a commonly used tool for the same purpose is the morphological matrix (or matrix of alternatives), supplemented by a compatibility matrix [35, 36]. The morphological matrix is a table where the first column presents a list of desired functionality or problems, and the subsequent columns in each row are the potential physical implementations or solutions. Selections can begin on any row in the morphological matrix. An example of a morphological matrix for DRA 5.0 is shown in Figure 2.2 from Edwards et al. [37].

Choice	Option 1	Option 2	Option 3
Mission Type	Long Stay	Short Stay	
Cargo Deployment	Pre-Deploy	All-Up	
Mars Capture Method	Aerocapture	Propulsive	
Mars Ascent Propellant	ISRU	No ISRU	
Interplanetary Propulsion	NTR	Electric	Chemical

Figure 2.2: Revision of the DRA 5.0 trade tree into a morphological matrix

However, when augmented with a compatibility matrix, each selection may result in some options being removed from consideration as incompatible with the selected alternatives. As a tool, the morphological matrix provides a compact means to enumerate feasible alternatives. The correspondence to a trade tree would be as follows: starting with the first row, list all options. Then for each option in the first row, proceed to the second row and list all options for that functionality, repeating under each option from row one, and repeating for each row of the matrix. After enumerating all branches, prune the tree for compatibilities. The number of alternatives in the morphological matrix is, before accounting for compatibility and for matrices where only one selection is permitted per row:

$$\mathrm{A} = \prod_{i=0}^{i=n} \mathrm{len}\left(\mathrm{o}_i\right)$$

or the product of the length of the option sets o from each row i. In the case of Figure 2.2 this is $2*2*2*2*3 = 48$, the same as indicated by Figure 2.1. The number of alternatives is reduced when compatibility information, for example via a compatibility matrix, is included. However, the number of alternatives increases if multiple selections per row are permitted, when it would follow the rule of:

$$\mathrm{A} = 2^{\sum_{i=0}^{i=n} \mathrm{len}(\mathrm{o}_i)}$$

or 2 raised to the power of the sum of the lengths of the options available on each row where multiple options might be selected, where each option has become a binary variable (is selected or not) [38].

Additionally, the morphological matrix also allows for hierarchical selection problems. Hierarchical selection occurs where selection in a morphological matrix results in a new problem space, then a subsequent lower-level morphological matrix can represent the alternatives in the new area [35]. Additionally, while not specifically mentioned by the NASA Systems Engineering Handbook, morphological matrices are in use at NASA at least within the field of aeronautics [39]. Importantly, note that the morphological matrix is distinct from the "decision matrix" in the Systems Engineering handbook [1] which is a matrix that compares alternatives according to criteria, may apply weights or preferences to the criteria, and then be used to give a ranking of the alternatives with respect to preferences and criteria. A decision matrix can be used alongside a morphological matrix or matrix of alternatives, but they serve different purposes, as the output of enumerating alternatives – the activity performed with a morphological matrix – is one input to creating a decision matrix.

The typical implementation of a morphological matrix and compatibility matrix is via spreadsheet software as seen in Quinlan et al [39] figures 3-5, although some web-based implementations may exist. However, with the transition towards MBSE, the question arises how to represent this information in a model which can both maintain utility for generating alternatives, as well as provide traceability to other systems engineering artifacts as described by the model. Some progress in this field has been published already. Franzen et al [40] showed a function-means tree, or a mapping between a functional decomposition and potentially allocated solutions, as a hierarchical matrix of alternatives encoded using OWL to replicate an Interactive Reconfigurable Matrix of Alternatives (IRMA) for aircraft Search and Rescue System of Systems Systems Engineering – this same kind of modified morphological matrix is used by NASA aeronautics [39]. The benefits to their approach lie in the ability to standardize data interchange between tools referring to a common design space by leverage OWL and related technologies - the authors note the similar to such an effort with early efforts regarding the Common Parametric Aircraft Configuration Schema (CPACS) Extensible Markup Language (XML) schema.

However, while ontology technologies provide benefits for data exchange and reasoning on the knowledge base, other capabilities may lag due to increasing complexity of technology requirements. For example, visualization such as creating the matrix of alternatives from the knowledge base was listed as future work by the authors [40]. Schmit et al [41] detail a toolchain necessary to replace the spreadsheet approach with ontologies, though it should be noted that according to the figures in their paper, their ontology appears to formalize terms related to the system domain instead of a generic representation of morphological matrices. While some ontologies like OSLC focus on data interfaces hidden from users, and some ontology technologies have a sophisticated "tech stack" requiring extensive maintenance, the approach for the TSO is simpler. The emphasis for the TSO is on documentation and traceability of the trade space as well as analysis or decisions made based on the trade space, with any improvements to data exchange as desirable but not strictly necessary. To meet these particular needs, the proposed solution detailed in this report defines a partial ontology in concepts, with concrete implementation using an MBSE language (SysML). MBSE languages provide enhanced descriptive and diagrammatic capability, query formulation, and for some use cases simulation capability beyond what is available in spreadsheet tools. SysML is already deployed at NASA across the agency on several programs and projects, and compatibility with this ecosystem should provide the most benefit to the agency.

This report establishes progress towards a model-based alternative generation capability with additional contextualization for alternatives and the trade space in decision-making problems. The utility of this capability, like the morphological matrix approach, is that it helps the engineer emphasize definition of solution options for identifying alternatives, while the basis of the implementation being SysML serves to improve traceability and computer interpretability of the resulting user-defined trade space.

2.2 Terminology

Application of the word ontology in this work is at a general level. In this section, the terminology box or TBox for the TSO will be defined. Table 2.1 provides the terms that will be defined for specifying a trade space.

Trade Space	A trade space defines a problem with a set of attributes for which solutions must be selected
TSO	Trade Space Ontology
Continuous Dimension	A continuous dimension is a real-valued degree of freedom that defines an attribute of the trade space
Cardinal Dimension	A cardinal dimension is a degree of freedom valued in the natural numbers that defines an attribute of the trade space regarding a quantity
Discrete Dimension	A discrete dimension represents a specific option which may be selected
Compound Choice	A compound choice groups discrete options, allows for selection of only one or of multiple options, and may act as a trade space
Trade Relationship	A trade relationship indicates that a dimension is a variable in a trade space/compound choice or that a discrete dimension is an option in a compound choice
Mutually Required Choice	Mutually Required Choice is a relationship between two trade relationships such that if one side is selected, the other must also be selected
Directed Required Choice	Directed Required Choice is a relationship between two trade relationships such that if the source selected, the target must also be selected
Mutually Disallowed Choice	Mutually Disallowed Choice is a relationship between two trade relationships such that if one side is selected, the other must not be selected
Directed Disallowed Choice	Directed Disallowed Choice is a relationship between two trade relationships such that if the source selected, the target must not be selected

Table 2.1: Trade Space Terminology

2.3 SysML Implementation

Three components are necessary for the SysML implementation: a profile, custom diagram definitions, and a plugin. In SysML, the language provides a means for customizing the general systems syntax to specific domains using what are called profiles, in which the new more-specific syntax is represented by user-defined stereotypes and indicated by guillemets, a chevron-style quotation mark used in many languages. The tool used in this paper for editing SysML, MagicDraw , provides additional functionality through « Customization » applied to stereotypes which alter the behavior of the tool when those customized stereotypes are applied to elements, for example through the definition of derived properties and element numbering. The profile for the TSO is focused on defining SysML stereotypes, MagicDraw customizations, and SysML library elements (e.g. blocks) which capture the terminology from Section 2.2. The conceptual relationships that the TSO defines are illustrated in Figure 2.3. The trade space ontology is in the process of being refactored in the direction of this "concept architecture" as the concept architecture better reflects the actual relationships between elements in common usage of the tool. Specifically, in Figure 2.3 the "trade space

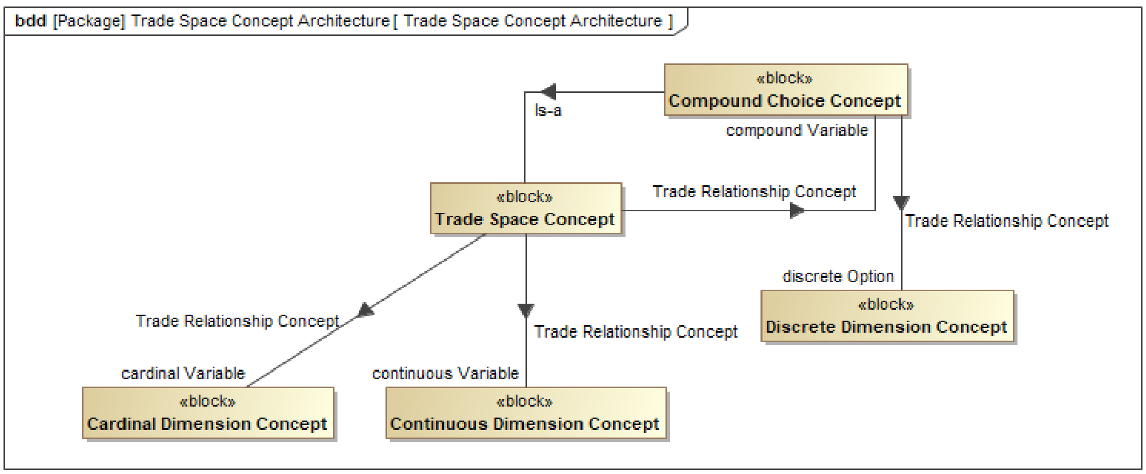

Figure 2.3: Trade Space Ontology Conceptual Architecture Diagram

concept" is the central item. Attributes of the trade space are defined by the existence of trade relationships to either cardinal dimensions, continuous dimensions, or compound choices. Compound choices collect a set of discrete dimensions, but may act as trade spaces with appropriate attributes. Here, the word "attribute" is being used in the sense of the trade space or morphological matrix, where options are defined for each attribute, and not in the sense of the UML or SysML definition. The implementation for trade relationship is via dependency, and no SysML attributes are required. Additionally, the blocks in Figure 2.3 are merely representative of the concepts. The user model will contain instances of classifiers provided in the library, defined in Figure 2.4.

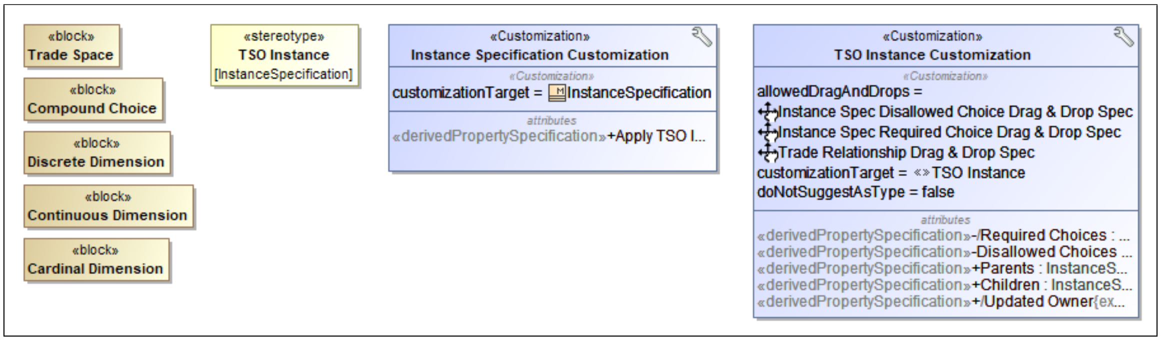

Figure 2.4: Definition of the « TSO Instance »

Each instance using a classifier from the library should carry the « TSO Instance » stereotype. The customization of Instance Specification should enforce this constraint in most models, though intended application would be to create elements via the custom diagrams (defined in the next section), and the buttons on the custom diagrams enforce the correct classifiers and stereotypes. Additionally, customization of the stereotype itself provides additional information about the trade space, include rollups of required or disallowed choices, parents and children in the hierarchy defined by trade relationships, etc.

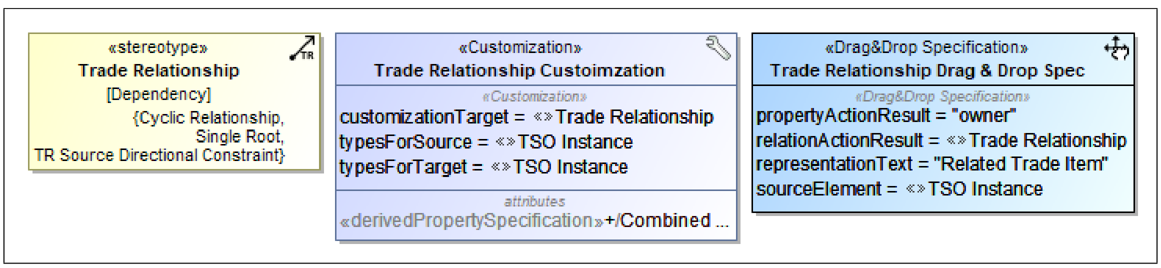

Figure 2.5: Definition of the « Trade Relationship » stereotype

Figure 2.5 provides the definition for the « Trade Relationship » dependency. Trade relationships are the mechanism for specifying trade space attributes and options. The dependency provides a specific usage of an option relative to an attribute.

In order to constrain choices in the trade space, Figure 2.6 illustrates the definition of four dependency stereotypes. These include directed required choice, directed disallowed choice, mutually required choice, and mutually disallowed choice. These dependencies provide the equivalent data for a compatibility matrix, insofar as two choices are compatible if there are no dis/allowed relationships between their « Trade Relationship » dependencies, while the presences of any of the four dependencies from Figure 2.6 provides the means to annotate a limitation between two « Trade Relationship » dependencies, either exclusive or inclusive, depending the on the kind.

2.3.1 Informing Use of Stereotypes with Validation Rules

Since stereotypes act as a kind of vocabulary, a missing ingredient for the TSO to act as a domain-specific language is a grammar, or set of rules through which using the stereotypes in the vocabulary describe valid trade spaces. In SysML using MagicDraw , these rules are often

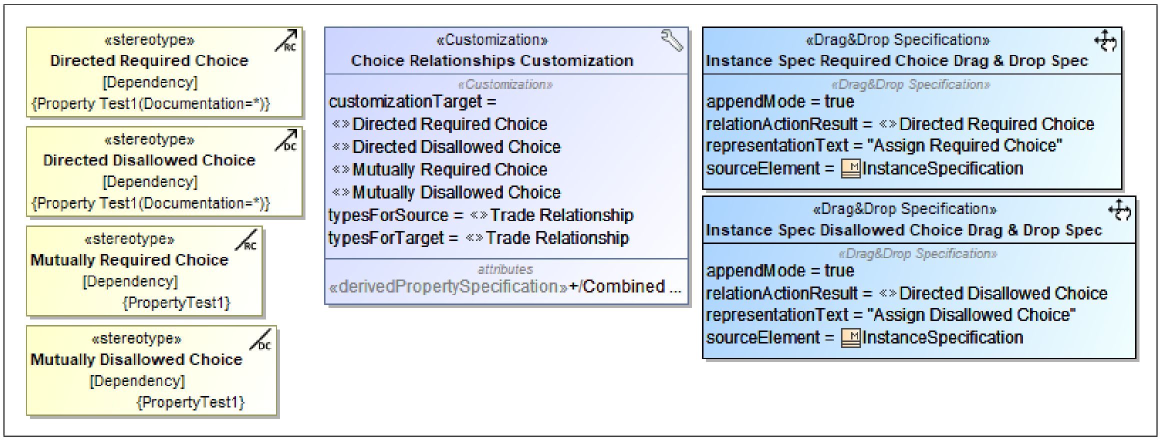

Figure 2.6: Definition of dependencies for constraints between options.

defined using validation rules. Validation rules in MagicDraw check the correctness of usage of specific elements in a model, either on demand or at recurring intervals. Validation rules in a SysML profile provide a means to restrict the usage of a profile and provide guidance to users on the intended application of the customized syntax. Already in the preceding figures, several validation rules are named through the display of constraints associated with stereotypes in the profile. Table 2.2 provides an overview of all validation rules currently applied to the profile.

Rule Name	Constrained Element	Function
Cyclic Relationship	« Trade Relationship »	See Section 2.5.2
Directed Disallowed Choice Doc Constraint	« Directed Disallowed Choice »	Require documentation on dependency
Directed Required Choice Doc Constraint	« Directed Required Choice »	Require documentation on dependency
Multiple Root Tradespace	« TSO Instance »	See Section 2.5.3
Mutually Disallowed Choice Doc Constraint	« Mutually Disallowed Choice »	Require documentation on dependency
Mutually Required Choice Doc Constraint	« Mutually Required Choice »	Require documentation on dependency
TR Source Directional Constraint	« Trade Relationship »	Enforce « Trade Relationship » end types

Table 2.2: TSO validation rules at the time of writing

As noted in Table 2.2, subsequent sections discuss specific validation rules in greater detail. However, most other validation rules utilize the structured expression syntax to enforce rules based on query criteria, such as the existence of documentation for dis/allowed choices. In the case of the "TR Source Directional Constraint" validation rule, the structured expression checks whether the source of a « Trade Relationship » is an instance of Trade Space or Compound Choice. This rule enforces a key aspect of the grammar of the TSO such that can be notified of an error if they make a « Trade Relationship » in the wrong direction, e.g. from Discrete Dimension to Compound Choice.

2.4 Custom Diagrams for MagicDraw

Included with the MagicDraw resource and plugin are five custom diagrams tailored to use of the TSO profile. MagicDraw provides a capability to modify built-in diagram types for specific purposes as well as to establish new diagrams for domain-specific notation. The diagrams included with the TSO are intended to lower the barrier of entry on building a trade space model in MagicDraw , and are supplementary to the standard SysML diagrams. They can be used to create, manage, and visualize TSO elements. Two diagrams are for building and visualizing the trade space: Trade Definition Diagram and Trade Relation Map. Two diagrams for viewing trade space ontology instance and dependency information respectively: Trade Space Instance Summary Table and Trade Space Dependency Summary Table. One final diagram is included for viewing and managing constraint relationships (required and disallowed choices) – the Trade Space Choice Relation Matrix. These diagrams are listed under a new diagram category "Trade Definition."

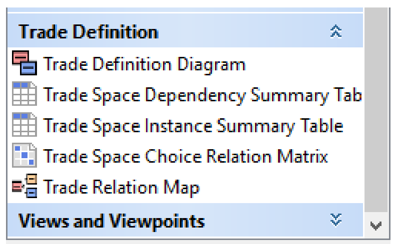

Figure 2.7: Custom diagrams available in the Trade Definition category of diagrams

2.4.1 Trade Definition Diagram

The Trade Definition Diagram (TDD) (quick abbreviation being "tdd" in MagicDraw menus) includes custom buttons to allow easy creation of « TSO Instance » elements with appropriate classifiers assigned, as well as the appropriately stereotyped dependencies between these instances. Additionally, the smart manipulator menu can be used from « TSO Instance » elements on the TDD to create a new « Trade Relationship » and end element of selected type via right click. Elements created on a TDD are housed by default in the same package as the diagram.

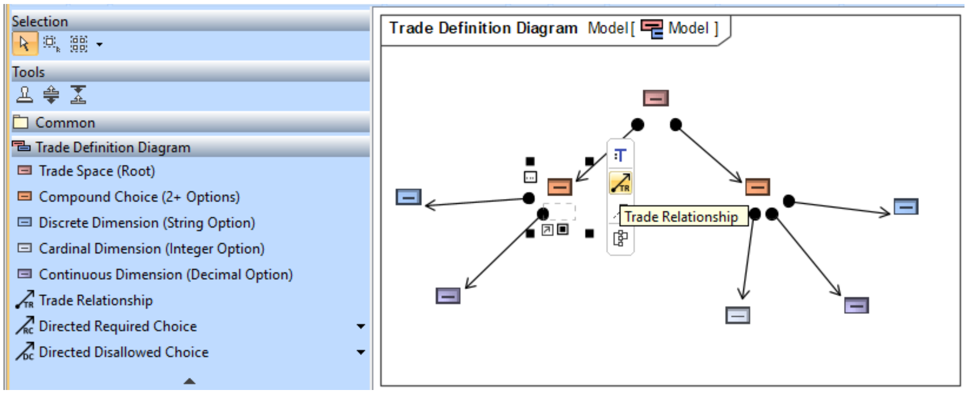

Figure 2.8: TDD smart manipulator menu from instance

Figure 2.8 illustrates how the trade relationship smart manipulator is accessed from a « TSO Instance » on a TDD. Figure 2.9 shows the element selection menu available when creating a « Trade Relationship » from the smart manipulator menu.

2.4.2 Trade Relation Map

The Trade Relation Map (TRM) is included as a secondary diagram for visualization of the trade tree. A « TSO Instance » must be dragged onto the diagram or place in the context box. Once the context is set, the tree generates automatically to reflect the context trade tree. This diagram is useful for a compact tree view where branches can be collapsed and expanded by level or by individual preference. An example TRM is shown in Figure 2.10.

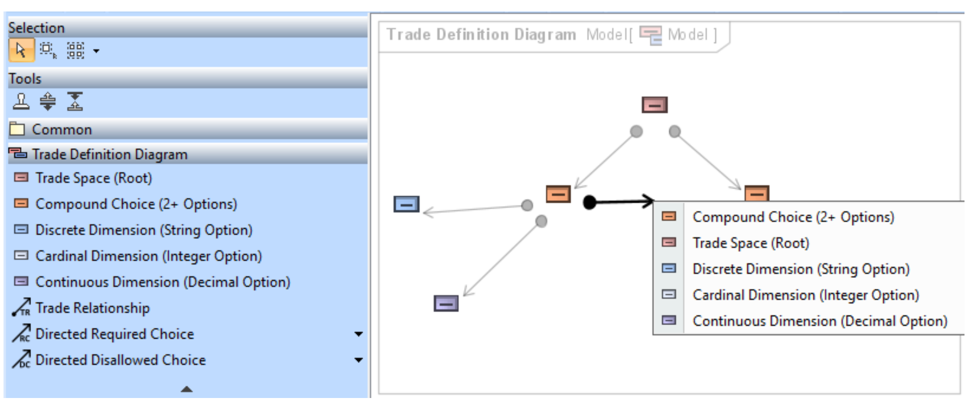

Figure 2.9: Smart Manipulator « Trade Relationship » end element menu

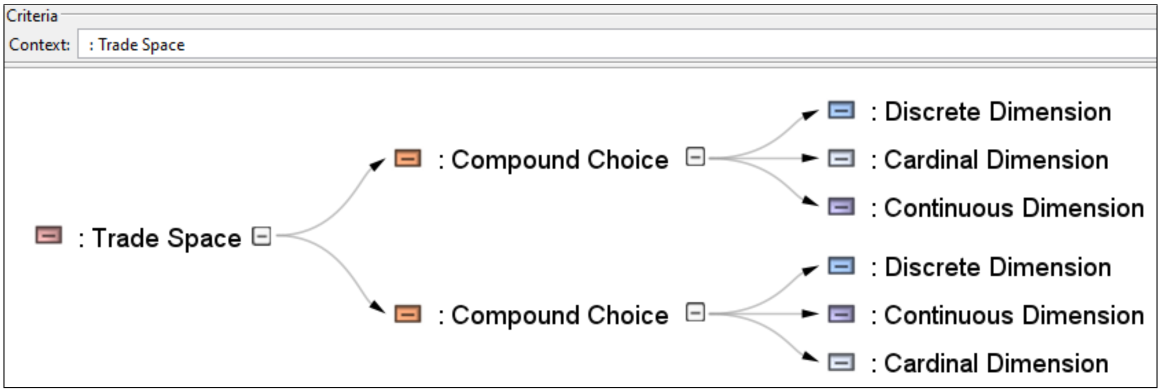

Figure 2.10: Example TRM

2.4.3 Trade Space Choice Relation Matrix

The Trade Space Choice Relation Matrix collects all « Trade Relationship » dependencies in a model and displays them in a matrix. This diagram can be used to create and manage required and disallowed choice relationships. Figure 2.11 shows an example Trade Space Choice Relation Matrix.

2.4.4 Trade Space Instance Summary Table

The Trade Space Instance Summary Table diagram builds a table showing all « TSO Instance » elements in the project. This table can used to view, manage, and export instance data. The table automatically shows columns for Name, Classifier, Parents, Children, and Documentation. An example Trade Space Instance Summary Table is shown in Figure 2.12.

2.4.5 Trade Space Dependency Summary Table

The Trade Space Dependency Summary Table diagram builds a table showing all trade space ontology dependencies in the model. This includes « Trade Relationship », required

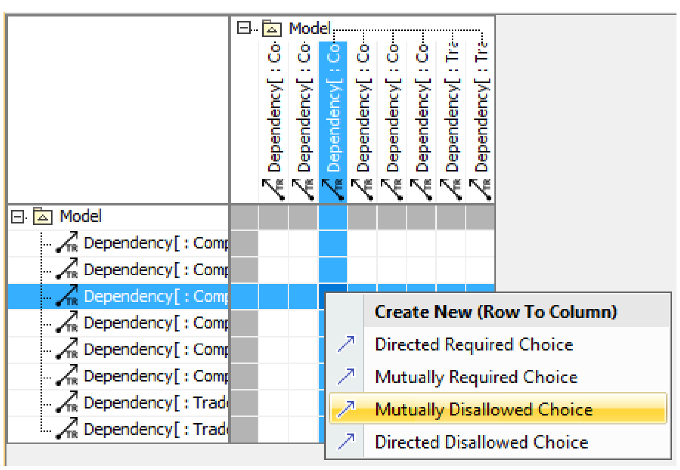

Figure 2.11: Example Trade Space Choice Relation Matrix

Criteria

Element Type: TSO Instance ... Filter:

#	Name	Element Type	Parents	Children	Documentation
1		Cardinal Dimension	: Compound Choice		
2		Cardinal Dimension	: Compound Choice		
3		Compound Choice	: Trade Space	: Discrete Dimension : Continuous Dimension : Cardinal Dimension	
4		Compound Choice	: Trade Space	: Discrete Dimension : Continuous Dimension : Cardinal Dimension	
5		Continuous Dimension	: Compound Choice		Continuous dimension - a decimal value representing an option that may be selected.
6		Continuous Dimension	: Compound Choice		Continuous dimension - a decimal value representing an option that may be selected.
7		Discrete Dimension	: Compound Choice		
8		Discrete Dimension	: Compound Choice		
9		Trade Space		: Compound Choice : Compound Choice	

Figure 2.12: Example Trade Space Instance Summary Table

choices, and disallowed choices. An example based on the DRA 5.0 information is shown in Figure 2.13. This table can be used to view, manage, and export dependency information. The table automatically shows columns for Name, Derived Name (an automatically generated property that combines the names of the instance on either end of the relationship), source, target, and documentation.

Element Type	Name	Derived Name (Auto)	Source	Target	Documentation
CustomImageHolder [Element] Trade Relationship [Dependency]		[Cargo Deployment] -> [All-Up Cargo]	Cargo Deployment : Compounc	All-Up Cargo : Discrete Dimension	Tree building relationship. A trade relationship indicates options in trade space and compound choices.
CustomImageHolder [Element] Trade Relationship [Dependency]		[Cargo Deployment] -> [Pre-Deploy Cargo]	Cargo Deployment : Compounc	Pre-Deploy Cargo : Discrete Dimer	Tree building relationship. A trade relationship indicates options in trade space and compound choices.
CustomImageHolder [Element] Trade Relationship [Dependency]		[Crewed Mars Exploration] -> [Cargo Deployment]	Crewed Mars Exploration : Trac	Cargo Deployment : Compound C	Tree building relationship. A trade relationship indicates options in trade space and compound choices.
CustomImageHolder [Element] Trade Relationship [Dependency]		[Crewed Mars Exploration] -> [Interplanetary Propulsion Type]	Crewed Mars Exploration : Trac	Interplanetary Propulsion Type : C	Tree building relationship. A trade relationship indicates options in trade space and compound choices.
CustomImageHolder [Element] Trade Relationship [Dependency]		[Crewed Mars Exploration] -> [Mars Ascent Propellant Source]	Crewed Mars Exploration : Trac	Mars Ascent Propellant Source : C	Tree building relationship. A trade relationship indicates options in trade space and compound choices.
CustomImageHolder [Element] Trade Relationship [Dependency]		[Crewed Mars Exploration] -> [Mars Cargo Capture Method]	Crewed Mars Exploration : Trac	Mars Cargo Capture Method : Cor	Tree building relationship. A trade relationship indicates options in trade space and compound choices.
CustomImageHolder [Element] Trade Relationship [Dependency]		[Crewed Mars Exploration] -> [Mission]	Crewed Mars Exploration : Trac	Mission : Compound Choice	Tree building relationship. A trade relationship indicates options in trade space and compound choices.
CustomImageHolder [Element] Trade Relationship [Dependency]		[Interplanetary Propulsion Type] -> [Chemical]	Interplanetary Propulsion Type	Chemical : Discrete Dimension	Tree building relationship. A trade relationship indicates options in trade space and compound choices.
CustomImageHolder [Element] Trade Relationship [Dependency]		[Interplanetary Propulsion Type] -> [Electric]	Interplanetary Propulsion Type	Electric : Discrete Dimension	Tree building relationship. A trade relationship indicates options in trade space and compound choices.
CustomImageHolder [Element] Trade Relationship [Dependency]		[Interplanetary Propulsion Type] -> [NTR]	Interplanetary Propulsion Type	NTR : Discrete Dimension	Tree building relationship. A trade relationship indicates options in trade space and compound choices.
CustomImageHolder [Element] Trade Relationship [Dependency]		[Mars Ascent Propellant Source] -> [ISRU]	Mars Ascent Propellant Source	ISRU : Discrete Dimension	Tree building relationship. A trade relationship indicates options in trade space and compound choices.
CustomImageHolder [Element] Trade Relationship [Dependency]		[Mars Ascent Propellant Source] -> [Pre-Deployed Propellant]	Mars Ascent Propellant Source	Pre-Deployed Propellant : Discrete	Tree building relationship. A trade relationship indicates options in trade space and compound choices.
CustomImageHolder [Element] Trade Relationship [Dependency]		[Mars Cargo Capture Method] -> [Aerocapture]	Mars Cargo Capture Method : (	Aerocapture : Discrete Dimension	Tree building relationship. A trade relationship indicates options in trade space and compound choices.
CustomImageHolder [Element] Trade Relationship [Dependency]		[Mars Cargo Capture Method] -> [Propulsive]	Mars Cargo Capture Method : (	Propulsive : Discrete Dimension	Tree building relationship. A trade relationship indicates options in trade space and compound choices.
CustomImageHolder [Element] Trade Relationship [Dependency]		[Mission] -> [Conjunction Class Long Surface Stay]	Mission : Compound Choice	Conjunction Class Long Surface St	Tree building relationship. A trade relationship indicates options in trade space and compound choices.
CustomImageHolder [Element] Trade Relationship [Dependency]		[Mission] -> [Opposition Class Short Surface Stay]	Mission : Compound Choice	Opposition Class Short Surface St	Tree building relationship. A trade relationship indicates options in trade space and compound choices.

Figure 2.13: Example Trade Space Dependency Summary Table

2.5 Plugin Software for MagicDraw

The initial motivation for the plugin software was to replace dependency on the Cameo Simulation Toolkit license for evaluation of process automation components like sending data about a trade space to external alternative generation software. Over time, the plugin expanded to include application to validation rules. Two of the three features will be discussed here, with the third discussed later.

2.5.1 Transform Tradespace to JSON

The "Transform Tradespace to JSON" feature is a button available when right-clicking a valid Trade Space instance in the containment tree. The button directly replaces simulation for configuring the trade space to be used by the external alternative generation software. When the button is clicked, the plugin software performs several mapping and transformation steps to convert the trade space, as defined by the « Trade Relationship » dependencies connecting any « TSO Instance » to the selected Trade Space instance. Currently, due to limitations in the external alternative generation software discussed later, this transformation ignores Continuous Dimensions and Cardinal Dimensions. Additionally, based on the nested level of Compound Choices, the Compound Choice may be inferred as a Trade Space and annotated as such in the JSON. At the end of the transformation, the user is prompted to save the file to their machine. This JSON file can then be used directly with the external alternative generation software if the user has access to it, or passed to the software by some other means.

2.5.2 Cyclic Relationship Validation

While many validation rules can be written without the support of additional code in the form of plugin software by using MagicDraw's structured expression feature, some capabilities are easier to implement directly via source code. Cases where applying graph algorithms are notable, as the queries available normally in structured expressions may not provide the necessary level of detail. One example is the "Cyclic Relationship Validation" feature, which is a method to detect whether a « Trade Relationship » induces a cycle in the network of « TSO Instance » that make up a trade space. A cycle in this network is a loop in the relationships that can cause unbounded behavior, for example in the process to extract JSON from Section 2.5.1. By blocking the button activation for the JSON activation and also providing an annotation in the model that notifies the user of the undesirable model structure, the logic behind the validation rule helps inform the user on best practices. In the profile as seen on Figure 2.5, this validation rule is present to annotate cycles as an error in the user model. At the time of writing, the rule is not implemented as a "Binary Validation Rule" within MagicDraw , instead relying on the "Script Validation Rule" style to dynamically test whether the plugin is loaded and run the method for validation from an available static property of the plugin class. This differs from the "Binary Validation Rule" implementation which would require the plugin to implement a class loader logic to load the validation rule class at runtime, with active validation presumably calling the redefined methods of the specified validation rule class. In the future, the implementation may be revised towards the “Binary Validation Rule” format; however, the current format seemed simpler to implement, provided a control for users who have the profile but not the plugin, and made the cyclic relationship validation method readily available for use as a condition in “Transform Tradespace to JSON”. Additionally, the same script rule pattern can be applied in other potential validation rules to quickly reuse plugin source code to enforce new rules. The result of the cyclic relationship rule is shown in Figure 2.14.

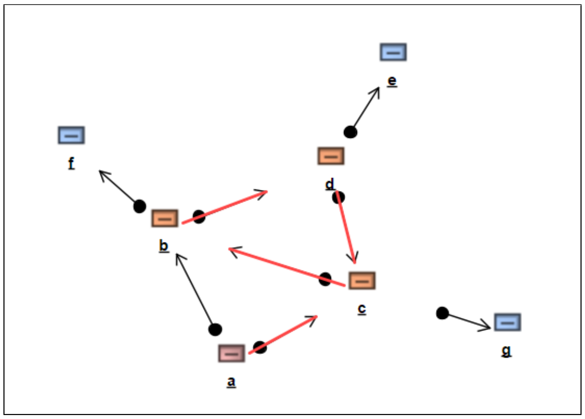

Figure 2.14: Example of a trade space representation that would violate the cyclic relationship rule.

2.5.3 Multiple Root Validation

Another issue checked by validation rule is the existence of multiple roots in the trade space. The current implementation of transforming the trade space to JSON requires that the trade space being transformed has one root element, or a single element with no incoming trade relationships. This validation rule provides both an annotation in the model according to the same logic as the code that checks whether the transformation button should be active. Figure 2.15 illustrates the effect of the validation rule on trade spaces with multiple root elements. Notably, this limits the ability to use Compound Choices or Discrete Dimensions across multiple disjoint trade spaces, though reuse within a trade space is permitted as long as the cyclic rule is not violated. As with Section 2.5.2, the validation rule is implemented as a script validation rule with checks as to whether the plugin is installed. If the plugin classes can be loaded, then the appropriate methods are called in order to perform the check for multiple roots.

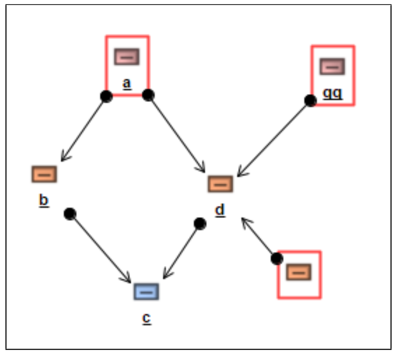

Figure 2.15: Example of a trade space that violates the multiple roots rule.

2.5.4 Planned Features

Several features are under consideration but have not been fully added to the plugin at this time. First would be to define custom diagrams in the plugin vs. manually using the GUI for customizing diagrams in MagicDraw . However, this feature may also be required to easily register custom diagrams without additional manual configuration. Full implementations of custom diagrams appear to leverage both plugin code as well as diagram descriptors configured to leverage the plugin code, vs. the options normally available within the custom diagram GUI. A second feature under consideration, once the profile and library have settled and are receiving fewer changes, would be to use the MagicDraw developer tools to export profile classes and refer to profile elements by the generated profile classes. The third main feature under consideration would be to roll the alternative generation into the plugin as an option, though this is not strictly required due to the external alternative generation software. Several other smaller features are also under consideration as quality-of-life improvements, such as GraphML import and export.

2.6 Example

The first example, in Figure 2.16, illustrates a replica of the DRA 5.0 heritage architecture trade space from Figure 2.1 and Figure 2.2 using the SysML implementation of the Trade Space Ontology. As can be seen, the user model has a hierarchical nature in the form of a tree, but this tree is not the same as the trade tree.

In a trade tree, the root is the objective and each step $n + 1$ away from the root is an alternative, and all combinations of alternatives are enumerated by following the paths from root to “leaves”. However, a path from root to leaf in Figure 2.16 does not enumerate a specific alternative, as it locates one “cell” in the equivalent matrix of alternatives or morphological matrix.

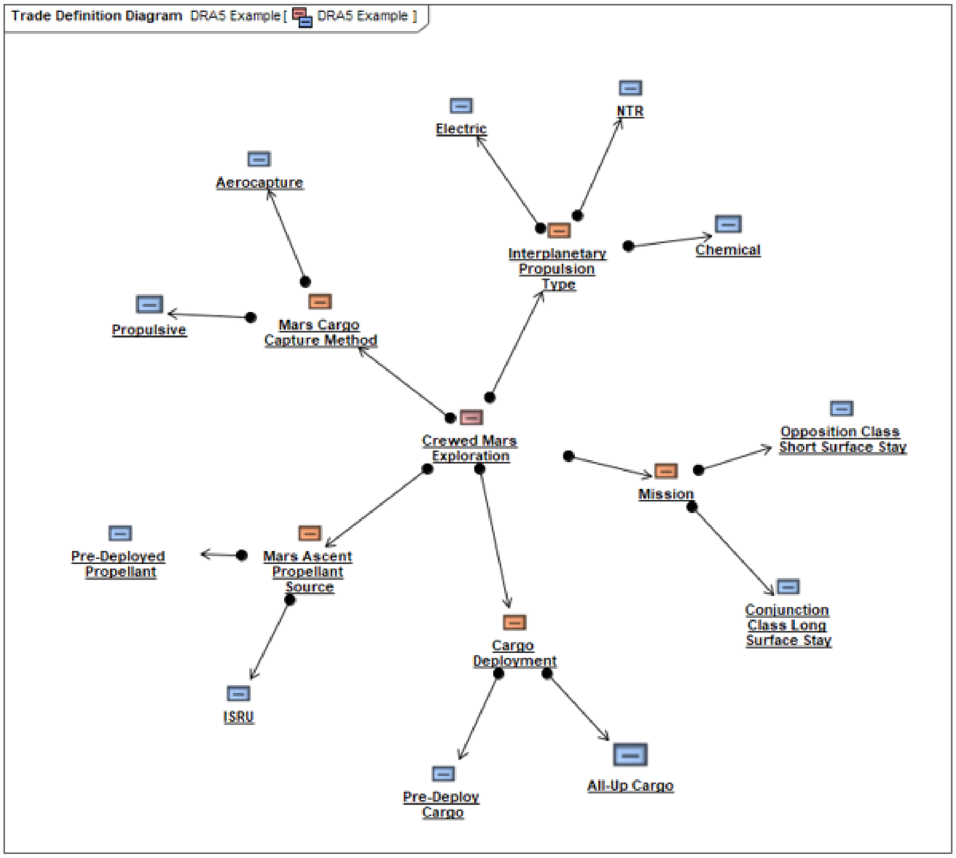

Figure 2.16: Simple Trade Space representation of the DRA 5.0

However, this view of the heritage DRA 5.0 trade space is simpler, as stated previously. It does not have many attributes, nor many options per attributes, nor any stated incompatibilities between options. On the other hand, Figure 2.17 represents a *notional* in-space propulsion stage trade space with more detailed choices as well as incompatibility information. While there are more individual options listed, due to the compatibility data, there are fewer feasible enumerated options than the DRA 5.0 trade tree.

Importantly, in the user model compatibility data is expressed between « Trade Relationship » dependencies. This includes the directional sense (« Directed Disallowed Choice » and « Directed Required Choice ») and undirected or bi-directional sense (« Mutually Disallowed Choice » and « Mutually Required Choice »). Each dependency expressing compatibility is required to have documentation text populated in validation rules, ensuring that the user model has documented the reasoning behind constraints in the trade space.

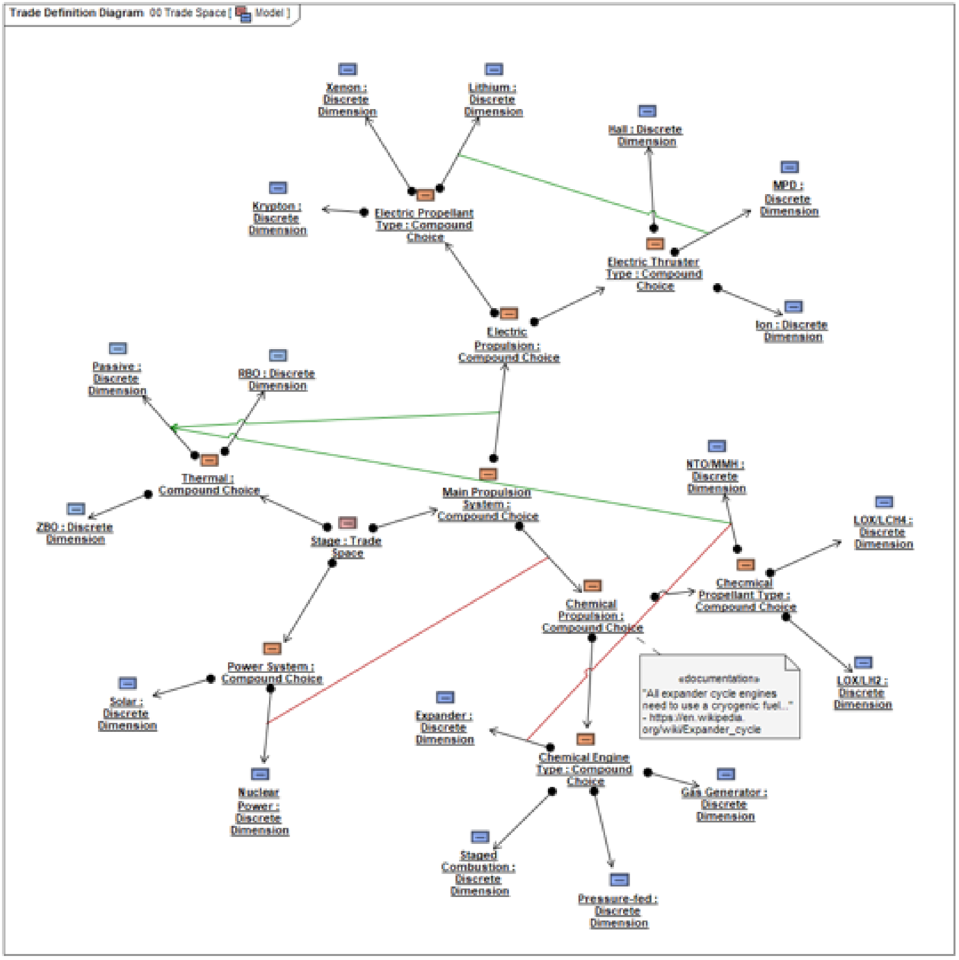

Figure 2.17: Notional trade space example with constraints between options.

3 Decision Space

3.1 Background

At first glance, a matrix of alternatives may appear to represent individual decisions. For example, if the attributes are decisions, then the options are the potential outcomes. However, often the decision-making process requires more information, or possibly formalized techniques for decision-making, or reports. Modeling the decision space requires enabling the user to model this information if desired. Additionally, a concrete linkage exists between decisions and trade spaces or trade-off analysis, and the model must also express this relationship even if it is more complicated than applying the terms regarding decisions to the trade space material.

The philosophy regarding decisions is that there may be a variety of potential decision input artifacts, but that generally there should be some definition of alternatives, criteria for evaluating alternatives, and preferences for criteria, similar in many ways to the approach described by Hazelrigg [42]. However, the method of decision making is not specified nor constrained by the profile here to the approach by Hazelrigg. The terms defined below help to organize material related to decision inputs and outputs, while as the Risk-Informed Decision Making Handbook states, "Sequential downselection, like all decision making, must be done in the context of stakeholder values and decision-maker responsibility/accountability" [10]. Nor is the intent here to replace processes used by Safety and Mission Assurance (S&MA); rather, the intent is to bring many of the concepts already used by S&MA to design and performance analysis, and to provide a means of traceability between decision-making artifacts and other Systems Engineering material, including the trade space.

3.2 Terminology

While developing the trade space capability, the team determined that representing decisions at a general level would also be of use. Table 3.1 provides the terms that will be defined for specifying a decision, as well as how decisions might relate to the trade space.

TSO Decision	TSO Decision represents all aspects of a decision
TSO Decision Input	TSO Decision Input represents the input provided to and/or considered by a decision maker
TSO Preference Set	TSO Preference Set is a kind of decision input which may be used to represent decision maker preferences for criteria
TSO Criteria Set	TSO Criteria Set is a kind of decision input which expresses aspects of the alternatives evaluated for the decision
TSO Alternatives Set	TSO Alternatives Set is the list of alternatives considered in the decision – this list could be infinite
TSO Alternative	TSO Alternative represents a particular alternative considered in the decision
TSO Decision Output	TSO Decision Output represents all output products after a decision is made by a decision maker
TSO Decision Refines	TSO Decision Refines is a decision-to-decision relationship that indicates which decisions follow from others
TSO Generates	TSO Generates relates a trade space or compound choice to a TSO Alternatives Set, indicates that the alternatives set is generated by the trade space or compound choice
TSO Modification	TSO Modification is a relationship between a decision output and any trade relationships, required choices, or disallowed choices created because of the decision
TSO Selection	TSO Selection is a relationship between a decision output and a discrete dimension which must be included in all alternatives considered by future decisions

Table 3.1: Decision Terminology

3.3 SysML Implementation

The terminology for decisions is encoded for SysML as stereotypes, as seen in Figure 3.1. These stereotypes provide a basis for further customization in user models.

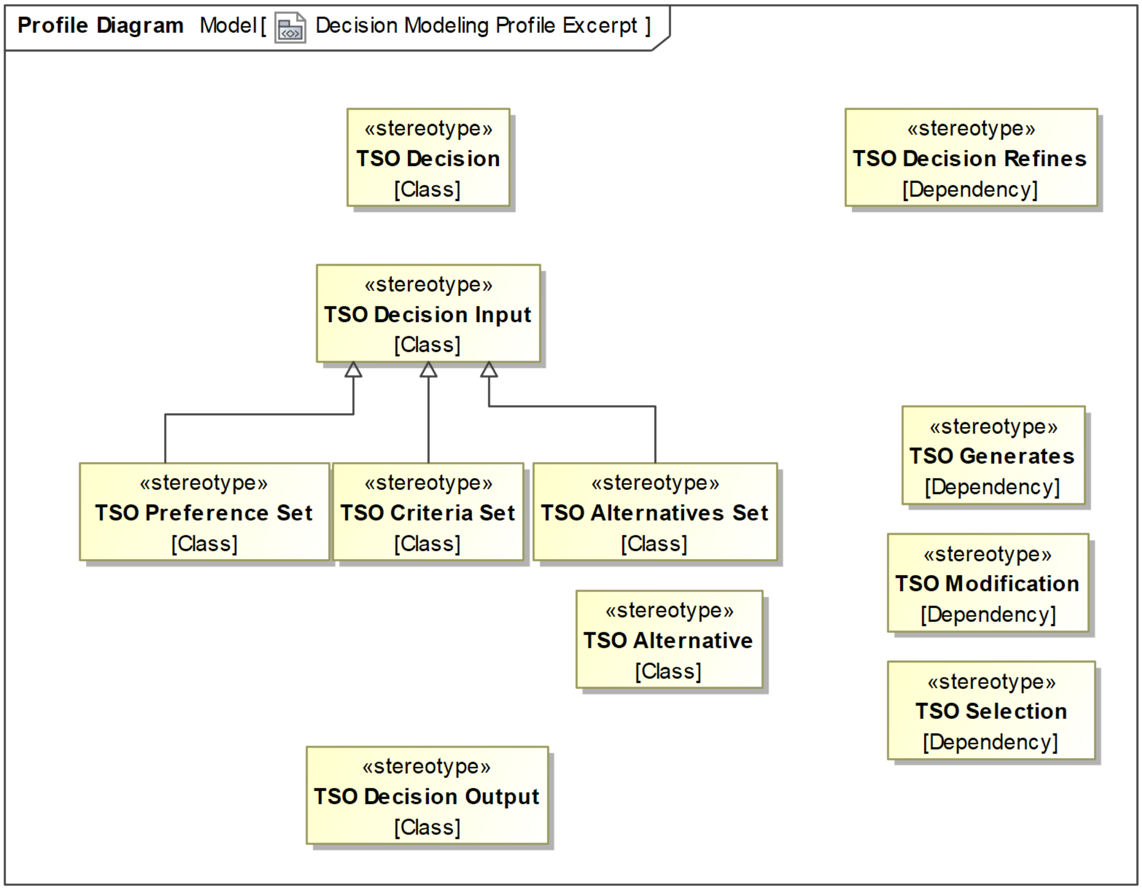

Figure 3.1: Stereotypes for defining decision information and linkages.

Specifically, while these stereotypes might be used directly by application to Blocks, Dependencies, etc in a user model, some projects and programs may have specific terms for decision-making. Therefore, at the profile level, only generic terms are provided. Additionally, since decision-making is likely to be commonly referenced in other profiles, the terms provided here are made specific with the acronym "TSO" added to each term. If a program or project needs more specific terms, these can be defined with inheritance to the TSO terms.

3.4 Limitations

The degree of customizability afforded by the approach for decisions enables program-specific terms for user models, but comes with some drawbacks. Primarily, this is that any MagicDraw customization of stereotypes to make application and usage easier must occur in the user model profile. The end result is that these stereotypes are more difficult to use than the trade space terms, as shown by Figure 3.2.

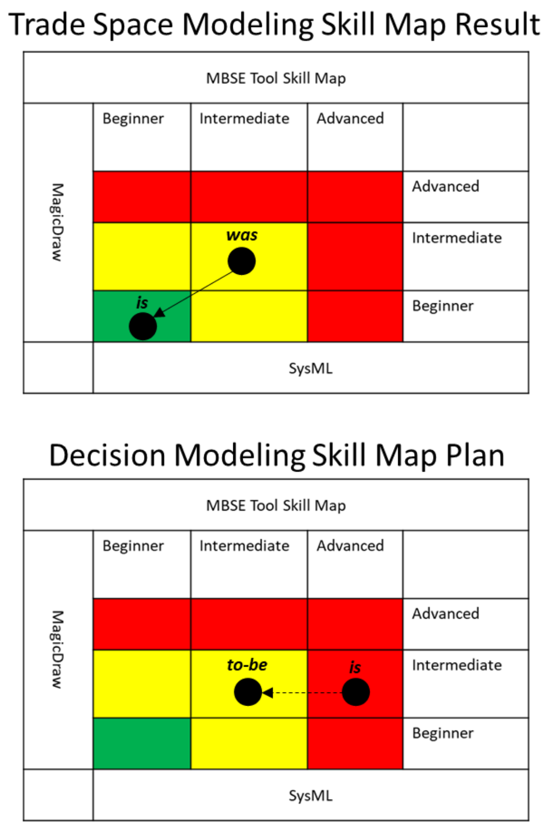

Figure 3.2: Estimated reduction in tool or language expertise for trade space vs. decision modeling

While significant effort was applied to lower the barrier to entry on the trade space side of the work, the decision aspect must preserve flexibility in user models and therefore comes with a higher burden on users to be aware of MagicDraw user interface features for applying stereotypes, and user knowledge of SysML as a language and how stereotypes work.

3.5 Plugin Software for MagicDraw

Currently, only one feature in the plugin supports the decision modeling capability.

3.5.1 Make Decision Paths

The plugin supports a feature to "Make Decision Paths" when right-clicking blocks with stereotypes inheriting from « TSO Decision » in the containment tree. The method called by the button first finds all « TSO Decision » blocks connected to the selected block by dependencies with stereotypes inheriting from « TSO Decision Refines », which is the implied network of decisions. Then, it enumerates paths from "source" decisions to "target" decisions as if the implied network were a directed acyclic graph. If the network does contain cycles, these are identified and discarded from the list of "terminated paths" which are provided in the enumerated results. When the enumeration is complete, the user is prompted to save a

Microsoft Excel file. The file has two sheets: the first sheet provides a list of all the paths through the decision network, while the second sheet provides statistics about the the number of decisions in each path plus the list of decisions not included in a given path. Current application of this data includes external visualization, but future application could include introduction of the critical path method [43], which may require a similar enumeration step in calculating the critical path in a schedule of tasks as in Baker's procedure [44].

3.6 Example

One example of creating a decision network that is readily available is the apparent sequence of decisions from the trade tree in Figure 2.1 [32]. This network of decisions is shown using the terms of the profile in Figure 3.3.

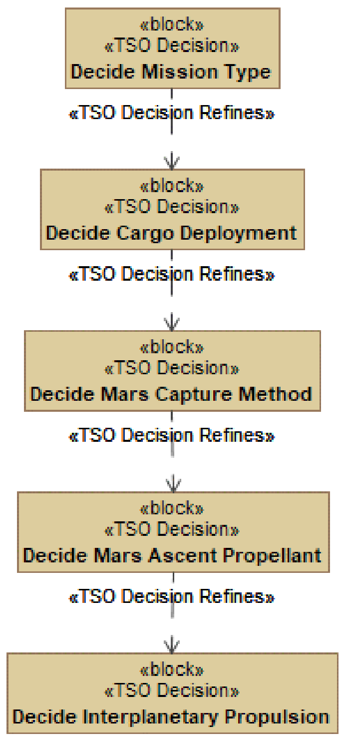

Figure 3.3: Hypothetical network of DRA 5.0 decisions

Besides a sense of temporal ordering or precedence, it is also possible to indicate other data associated with each decision. The traceability from trade space to decision input is shown in Figure 3.4. In effect, a subset of the alternatives generated by the trade space may be the alternatives of interest for a particular decision.

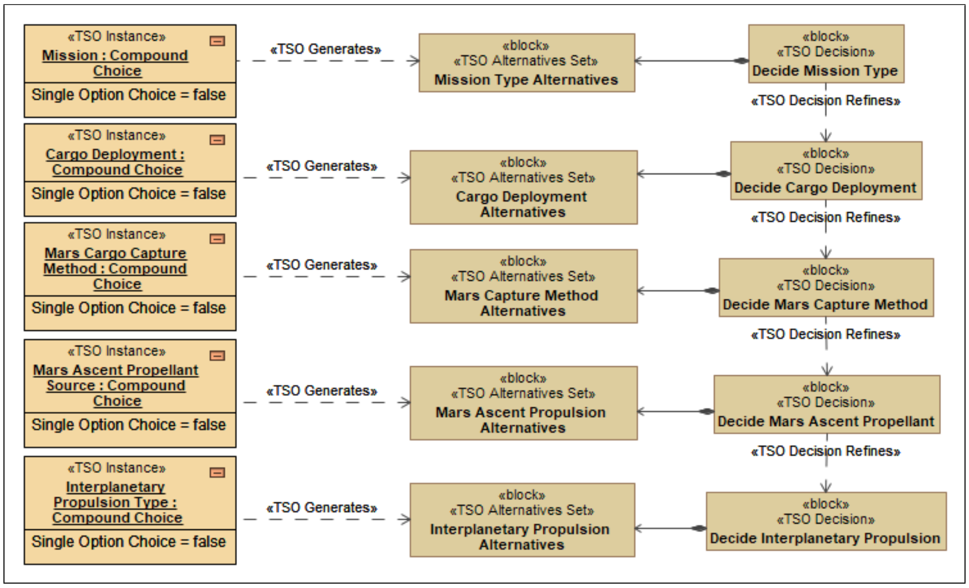

Figure 3.4: Mapping from trade space elements to decision input types following DRA 5.0

To illustrate a possible outcome of decisions, Figure 3.5 shows how decision output blocks can be tied to specific discrete dimensions or other dimension of the trade space to indicate selection of an alternative. This example aligns with the DRA 5.0 reference shown in Figure 2.1 [32]. Other forms of traceability are possible as well, including documenting the addition of options or addition of constraints between « Trade Relationship » dependencies.

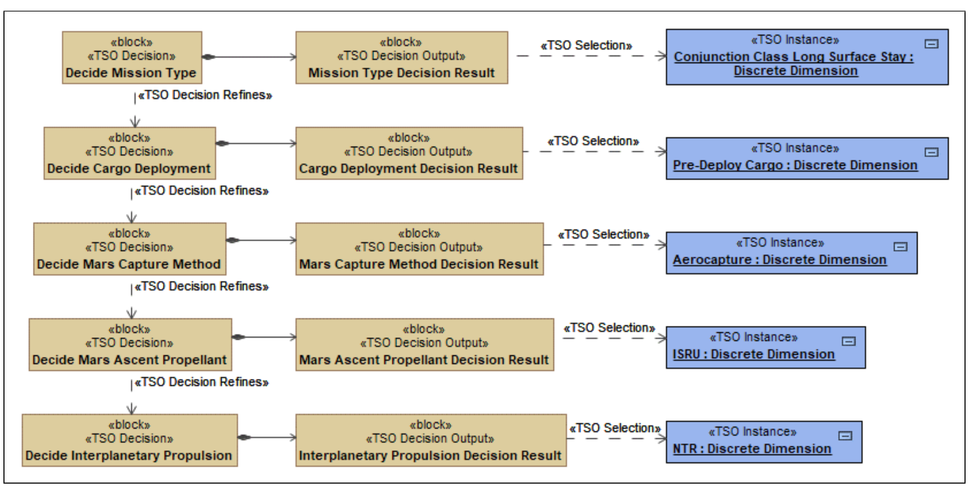

Figure 3.5: Example showing selected options from the trade space as a result of decisions

While the other dependencies for linking decisions and trade space are purely for traceability, functionality is planned to tie « TSO Selection » dependencies to the interface of the alternative generation software for what that software refers to as "desired subsets", which enforces a filtering of the feasible alternatives according to the selection.

Finally, it is possible to run the plugin feature to "Make Decision Paths" on this example. However, as might be expected from the previous figures, the result in Figure 3.6 is trivial as there is only one path in this example, which traverses all the connected decisions.

Key	Decision 1	Decision 2	Decision 3	Decision 4	Decision 5
Path 1	Decide Mission Type	Decide Cargo Deployment	Decide Mars Capture Method	Decide Mars Ascent Propellant	Decide Interplanetary Propulsion

Figure 3.6: Resulting decision path

3.7 External Visualizations

A capability to render Sankey diagrams, a visualization first introduced by Kennedy and Sankey [45], based on the spreadsheet output of the plugin was created in an external Python code. While this capability is not part of the tool, the spreadsheet itself is not always the best visualization of the individual paths. Rather than pulling the paths out onto separate Block Definition Diagrams, the Sankey diagram was configured to illustrate visually the paths described in the spreadsheet. The example from Figure 3.6 is shown in Figure 3.7.

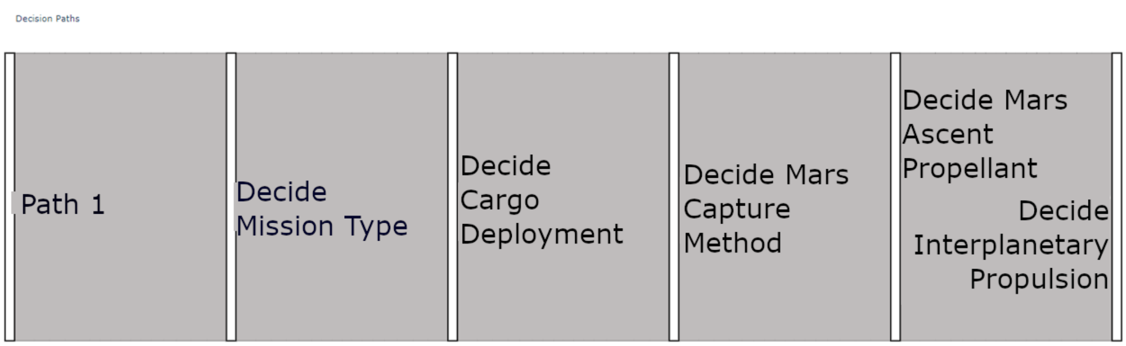

Figure 3.7: Sankey diagram for single path

While the DRA 5.0 Sankey diagram is again rather trivial, consider a notional decision network based on the trade space from Figure 2.17. One possible configuration for such a network is presented in Figure 3.8.

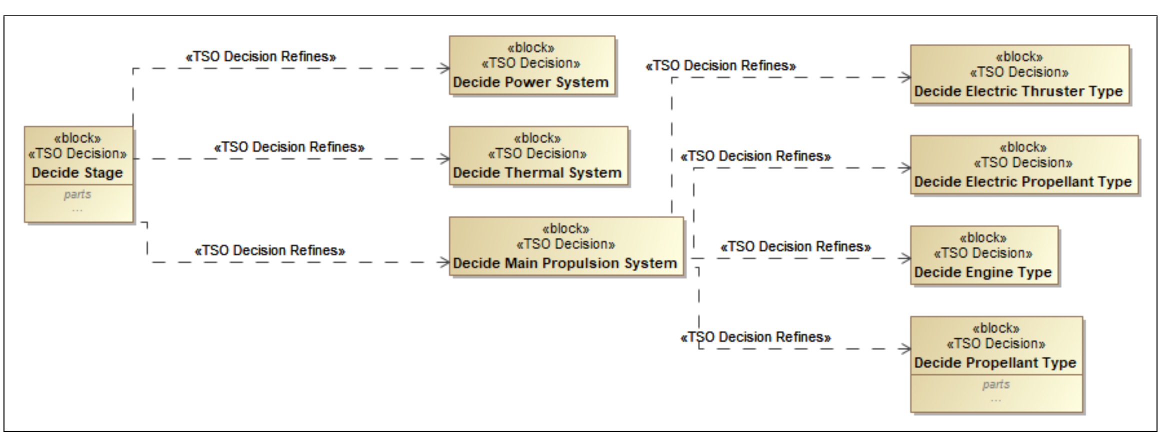

Figure 3.8: Notional decision network for the more detailed trade space example

By running "Make Decision Paths" on Figure 3.8, Figure 3.9 can be produced from the resulting spreadsheet, showing a more detailed Sankey diagram illustration.

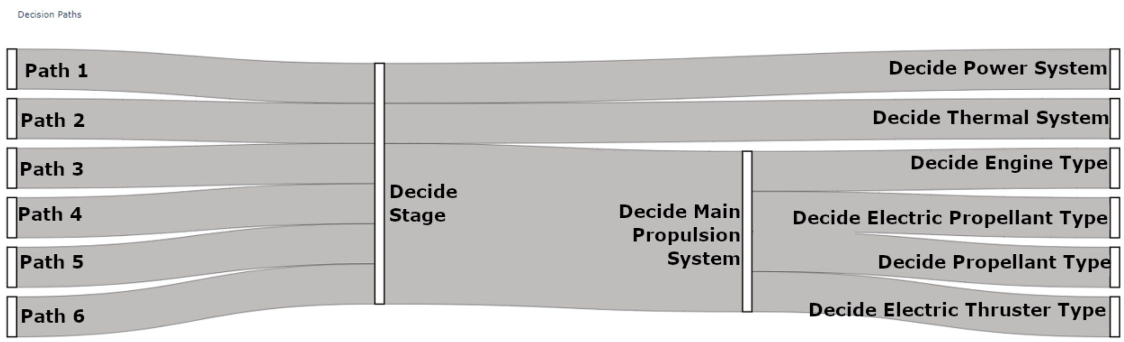

Figure 3.9: Sankey diagram for paths in the more detailed example decision network

Overall, this kind of visualization is just one way to present the data in the output spreadsheet. The spreadsheet remains the output of the tool. Other visualizations may be preferred. Additionally, the enumerated path information could be used for further analysis as part of future functionality.

4 Enumerating Alternatives

As with the decision network Sankey visualization, enumeration of alternatives is not included in the tool, but can be performed with external tools and software. Currently, the output of "Transform Tradespace to JSON" – a JSON file – is the interface between the tool described by this report and an external capability to generate alternatives. The current capability is a standalone Python script with a dictionary-based interface that corresponds closely with the JSON formatted by the plugin. However, this in-development tool is not suitable for all possible configurations of trade spaces, and is only useful for initial testing. The general capability to enumerate alternatives must filter out unnecessary attributes (currently: continuous and cardinal dimensions, which provide infinite alternatives), generate combinations, filter out combinations based on compatibility data, and provide a filter for "desired subsets". The current interface does not cover "desired subsets", but the presence of that functionality in the external software provides initial exploratory capability regarding selections. The combined set of capabilities enables both enumeration of alternatives, as well as investigating scenarios to see how specific selections might reduce the number of feasible alternatives. An illustration of this kind of exploration is shown in Figure 4.1, based on the trade space from Figure 2.17.

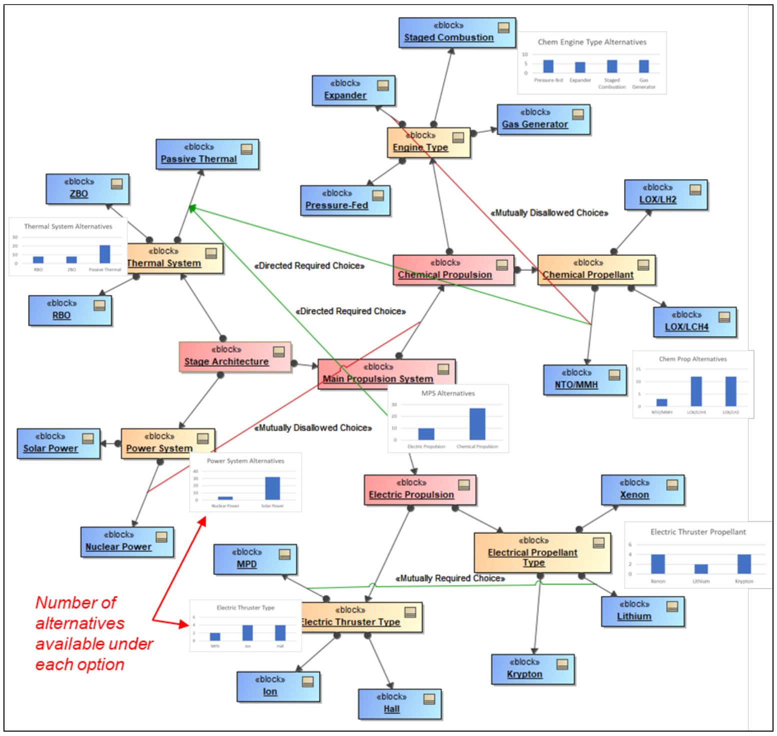

Figure 4.1: Example showing number of alternatives available by option in a trade space

In Figure 4.1, the numbers from the alternative enumeration are plotted as bar charts representing the number of feasible alternatives that had each option in the alternative set. Thus, it can be shown the degree to which a selection might reduce the space of feasible alternative, if a selected option present in fewer of the initial set of feasible alternatives were to be selected as the result of a decision. In effect, the result of this kind of analysis replicates one kind of scenario which might be played out by an IRMA as mentioned by Franzen et al [40], to investigate how the feasible alternatives available change as decisions are made (reducing the available space of alternatives).

5 Conclusion

The TSO improves the ability to author trade space information in SysML while lowering the barrier to entry. Additionally, it provides a mechanism to document decisions, relate them to the trade space, and relate both to the larger Systems Engineering universe. Simplifications to user interface components will help support adoption of MBSE while the core functionalities included improve the workflow of engineers engaged in conceptual design activities, especially when enumerating feasible alternatives.

5.1 Forward Work

Known aspects of forward work including features for the plugin software, the SysML profile, the custom diagrams, and continuous improvement as the tool is used.

Several items are currently under investigation for the plugin. First are features to support the current capability or improve workflow, such as better user feedback, improvements to validation rules, etc. Second, as the SysML profile settles down in terms of definition, the team will investigate generating profile classes and potential modifications to custom diagram classes to support the diagram descriptors. A third class of features involves interaction with external capability, such as improvements to alternative enumeration or implementing analytical techniques such as critical path method for the decision network. Additionally, it is likely that as the software is tested, that issues will be identified with the current and future functionality. Occasional maintenance, patches, and bug fixes are expected going forward.

The profile is currently undergoing initial user testing. As feedback is received, the profile may be modified, including specific names and relationships. The block library used for the trade space classifiers will also be revised to streamline relationships and properties to align with Figure 2.3, in order to clean up the instance slots that are carried throughout user models.

Some continued improvements may arise during user testing for custom diagrams. These changes may be different from potential interaction with the plugin software. Additionally, the specific colors used for model elements in the TSO may be revised according to user feedback.

Over all, the largest current weakness in the TSO is the lack of broad user testing and feedback. An important element to focus on next will be distribution of the resource bundle and training in use of the tool, to support conceptual design activities.

References

[1] National Aeronautics and Space Administration, *NASA Systems Engineering Handbook*, 2nd. National Aeronautics and Space Administration, 2016.

[2] Kaliszewski, I., *Soft Computing For Complex Multiple Criteria Decision Making*. 2006.

[3] Connoly, J. F., Joosten, B. K., Drake, B., Hoffman, S., Polsgrove, T., Rucker, M., Andrews, A., and Williams, N., "Human Mars Mission Design: The Ultimate Systems Challenge," in *IAC 2017 Congress Proceedings, 68th International Astronautical Congress (IAC)*, Sep. 25, 2017.

[4] Wollack, E. J., "How Old is the Universe?" Feb. 22, 2024, [Online]: `https://map.gsfc.nasa.gov/universe/uni_age.html`.

[5] Mazanek, D. D., Lepsch, R. A., and Saucillo, R. J., "Lunar Architecture Broad Trade Study Final Report," ESMD-RQ-0006, National Aeronautics and Space Administration.

[6] NASA, "Moon to Mars Architecture Definition Document (ESDMD-001) – Revision A," NASA/TP-20230017458, National Aeronautics and Space Administration.

[7] "Standard for Models and Simulations," NASA-STD-7009A, National Aeronautics and Space Administration.

[8] Kerzhner, A. A., "Using Logic-Based Approaches to Explore System Architectures for Systems Engineering," Ph.D. Dissertation, Georgia Institute of Technology, 2012.

[9] Sharma, J. L., "STASE: Set Theory-Influenced Architecture Space Eexploration," Ph.D. Dissertation, Georgia Institute of Technology, 2014.

[10] Dezfuli, H., Stamatelatos, M., Maggio, G., Everett, C., Youngblood, R., Rutledge, P., Benjamin, A., Williams, R., Smith, C., and Guarro, S., *NASA Risk-Informed Decision Making Handbook*. Apr. 1, 2010.

[11] "NASA Digital Engineering Acquisition Framework Handbook," NASA-HDBK-1004, National Aeronautics and Space Administration, 2020.

[12] "NASA Space Mission Architecture Framework (SMAF) Handbook for Uncrewed Space Missions," NASA-HDBK-1005, National Aeronautics and Space Administration, 2021.

[13] "NASA Systems Modeling Handbook for Systems Engineering," NASA-HDBK-1009, National Aeronautics and Space Administration, 2022.

[14] Hennig, C., Viehl, A., Kämpgen, B., and Eisenmann, H., "Ontology-Based Design of Space Systems," in *International Semantic Web Conference*, 2016, pp. 308–324.

[15] "Space System Ontology," 2021, European Space Agency, [Online]: `https://mb4se.esa.int/OSMOSE_SpaceSystemOntology.html` (visited on 11/21/2023).

[16] Johnson, H. E., Thomas, D. L., and Diaz, M. J., "Developing and Testing a Common Space Systems Ontology using the Ontological Modeling Language," in *2023 IEEE Aerospace Conference*.

[17] Rovetto, R. J., Kelso, T., and O'Neil, D. A., "Orbital Debris Ontology, Terminology, and Knowledge Modeling," in *International Orbital Debris Conference*.

[18] Berrios, D., Slater, L. T., Miller, J., Peach, K., Costes, S. V., and Schofield, P. N., "The Radiation Biology Ontology: A New Tool Supporting FAIR Principles Across Radiation Biology Facilitating Data Discovery and Integration," in *European Radiation Protection Week (ERPW) 2022*.

[19] O'Neil, D. A. and Rovetto, R. J., "An Ontology-Based Virtual Orrery," National Aeronautics and Space Administration.

[20] Feyock, S., "A Uniform Ontology for Software Interfaces," National Aeronautics and Space Administration.

[21] Tissot, F. and Menzel, C., "Ontology-driven information integration," in *NASA Tech Briefs, July 2005*.

[22] Keller, R. M., "The NASA Air Traffic Management Ontology: Technical Documentation," National Aeronautics and Space Administration.

[23] Rountree, I., "The Mars Ascent Vehicle & Model-Based Systems Engineering," in *MBSE Presentation to Auburn University Lifecycle Systems Engineering Class and JBS Solutions*.

[24] "Why OSLC?" OASIS, [Online]: `https://open-services.net/why/` (visited on 11/21/2023).

[25] Bayer, T. J., Chung, S., Cole, B., Cooke, B., Dekens, F., Delp, C., Gontijo, I., Lewis, K., Moshir, M., Rasmussen, R., and Wagner, D., "Model Based Systems Engineering on the Europa Mission Concept Study," in *2012 IEEE Aerospace Conference*.

[26] Wagner, D. A., Bennett, M. B., Karban, R., Rouquette, N., Jenkins, S., and Ingham, M., "An Ontology for State Analysis: Formalizing the Mapping to SysML," in *2012 IEEE Aerospace Conference*.

[27] Castet, J.-F., Rozek, M., Ingham, M. D., Rouquette, N. F., Chung, S. H., Kerzhner, A. A., Donahue, K. M., Jenkins, J. S., Wagner, D. A., Dvorak, D. L., and Karban, R., "Ontology and modeling patterns for state-based behavior representation," in *AIAA Scitech 2015*.

[28] Rouquette, N., "IMCE MOF2/OWL2 Integration," in *OMG Systems Engineering Domain-Specific Interest Group Meeting*.

[29] Jenkins, J. S. and Rouquette, N. F., "Semantically-Rigorous Systems Engineering Modeling Using SysML and OWL," in *International Workshop on Systems & Concurrent Engineering for Space Applications*.

[30] Jenkins, S. and Rouquette, N., "Progress on Integrating OWL and SysML," in *International Council on Systems Engineering (INCOSE)*.

[31] Wagner, D., Chodas, M., Elaasar, M., Jenkins, J., and Rouquette, N., "Ontological metamodeling and analysis using openCAESAR," in *Handbook of Model-Based Systems Engineering*.

[32] Drake, B. G., "Human Exploration of Mars Design Reference Architecture 5.0," [Online]: `https://ntrs.nasa.gov/citations/20090012109`.

[33] Donoho, D. L., "High-dimensional data analysis: The curses and blessings of dimensionality," *AMS math challenges lecture*, Vol. 1, No. 2000, p. 32, 2000.

[34] Richard, B., "Adaptive control processes: a guided tour," *Princeton, New Jersey, USA*, 1961.

[35] Weber, R. and Condoor, S., "Conceptual design using a synergistically compatible morphological matrix," in *FIE '98 28th Annual Frontiers in Education Conference. Moving from 'Teacher-Centered' to 'Learner-Centered' Education. Conference Proceedings (Cat. No. 98CH36214)*.

[36] Jimenez, H. and Mavris, D., "An Evolution of Morphological Analysis Applications in Systems Engineering," in *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*.

[37] Edwards, S., Hetterich, R., Trent, D., and Thomas, H., "Towards a Common Basis of Comparison Across a Broad Trade Space of Mars Architecture Options," in *2021 IEEE Aerospace Conference (50100)*.

[38] Engler, W., Bilgen, P., and Mavris, D., "Concept Selection Using an Interactive Reconfigurable Matrix of Alternatives (IRMA)," in *45th AIAA Aerospace Sciences Meeting and Exhibit*, Jan. 11, 2007.

[39] Quinlan, J. R., Marien, T. V., Blaesser, N. J., Kirk, J. T., Perkins, H. D., and Fisher, K. L., "Far-Term Exploration of Advanced Single-Aisle Subsonic Transport Aircraft Concepts," in *AIAA Aviation 2019*.

[40] Franzén, L. K., Staack, I., Krus, P., Jouannet, C., and Amadori, K., "Ontology-Represented Design Space Processing," in *AIAA AVIATION 2021 FORUM*, American Institute of Aeronautics and Astronautics, Jul. 2021.

[41] Schmit, M., Briceno, S., Collins, K., Mavris, D., Lynch, K., and Ball, G., "Semantic design space refinement for model-based systems engineering," in *2016 Annual IEEE Systems Conference*.

[42] Hazelrigg, G. A., *Fundamentals of Decision Making for Engineering Design and Systems Engineering*. 2012.

[43] Santiago, J. and Magallon, D., "Critical Path Method," in *CEE 320 - VDC Seminar*. *https://web.archive.org/web/20181024032348/http://web.stanford.edu/class/cee320/CEE320B/CPM.pdf*, 2009.

[44] Baker, S. L., "Critical Path Method (CPM)," in *HSPM J716*, University of South Carolina, Arnold School of Public Health, Dept. of Health Services Policy, Management Courses, and Curricula. https://web.archive.org/web/20100612142236/http://hspm.sph.sc.edu/COURSES/J716/CPM/CPM.html, 2004.

[45] Kennedy, A. and Sankey, H., "The Thermal Efficiency of Steam-Engines," in *Minutes of the Proceedings of the Institution of Civil Engineers*, Vol. 134, 1898, pp. 278–312.