Space Systems, Policy, and Architecture Research Consortium (SSPARC)
Draft 11/4/03

"SSPARC BOOK" MATERIAL for Lecture 3

Prepared by:

Hugh McManus
Metis Design

Draft for evaluation only. Do not distruibute.

DEDICATION AND NOTE ON SOURCES

This document is an excerpt of a future book or hyper-book on the MATE-CON method. It is provided for class use as a draft. Suggestions for improvement are welcome, as are warnings of errors or omissions. The notes below apply to the entire work in progress; the work or excepts of it should not be reproduced in any form without these notes.

This document is dedicated to the memory of Joyce Warmkessel, a colleague, mentor, and friend to many in the SSPARC and LAI communities. Many of the core ideas behind this work were originally expressed and developed by her, and she was a key mentor and facilitator to the development of all of this work.

The content of this document was developed by the SSPARC consortium. The primary compilers and codifiers of the MATE-CON method were Lt. Nathan Dillard and Adam Ross, in Master's thesis entitled, respectively, "Utilizing Multiple Attribute Tradespace Exploration with Concurrent Design for Creating Aerospace Systems Requirement,"[1] and "Multi-Attribute Tradespace Exploration with Concurrent Design as a Value-Centric Framework for Space System Architecture and Design."[2] Major contributors of the original concepts within the method, and/or complimentary methods and tools, include our SSPARC faculty and staff colleagues Elisabeth Paté-Cornell of Stanford University, Joel Sercel and Fred Cullick of Cal Tech, and Amar Gupta of MIT, post-doctoral researcher Bill Kaliardos, and graduate students Jimmy Benjamin, Jason Derleth, Bobak Ferdowsi, Dave Ferris, Russ Garber, Andre Girerd, Seth Guikema, Cyrus Jilla, Chris Roberts, Satwik Seshasai, Nirav Shah, Todd Shuman, Tim Spaulding, Dave Stagney, Dan Thunnissen, Myles Walton, Annalisa Wiegel, and Brandon Wood, along with their advisors and committees. Many other students, staff, and undergraduate researchers also contributed. Bill Borer, Kevin Ray, and John Ballenthin of the Air Force Research Laboratory, Steve Wall of NASA JPL, and Pete Hendrickson of the Department of Defense aided with the development of the method and the development of the case studies. SSPARC research work has been supported by an active group of industry practitioners, through both an Industrial Advisory Board (IAB) and on-site implementation activities.

The text of this manual is built on SSPARC research and member documents. Much of its contents are excerpts, modifications, or paraphrases of published or unpublished work done under SSPARC sponsorship. Every effort has been made to correctly attribute all contributions. Word-for-word excerpts are identified with quotes or indented, with citations. Many other excerpts have been edited to varying degrees and are integrated into the text for clarity. Their sources are cited in the text or in endnotes. Any omissions or errors of attribution should be brought to the authors' immediate attention for correction.

TABLE OF CONTENTS

1. NEED FOR A NEW FRONT-END METHOD

1.1. Critical role of front-end work in program success

Good up-front work in the eventual success of a program. It has been stated that 80% of the eventual costs of a system are determined before the first 20% of the funds have actually been spent.[3] Figure 1 illustrates this graphically. It is therefore not surprising that programs that under-fund front-end work (from mission feasibility through preliminary design) will have higher costs later in the program. This trend is dramatically illustrated in Figure 2, taken from the NASA Systems Engineering Handbook.[4] Note that this figure does not consider *failed* programs, many of which fail because of poor up-front work.

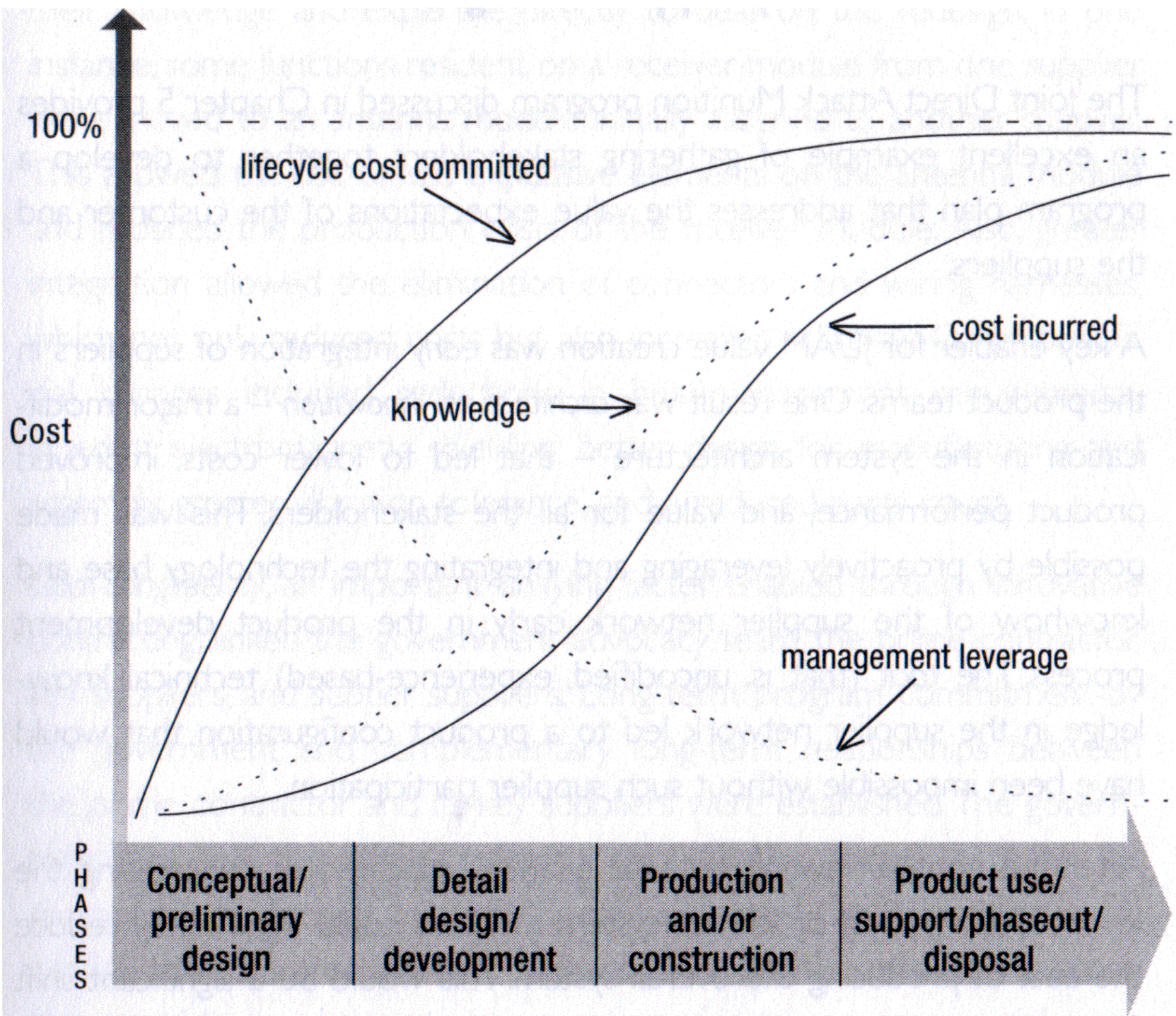


Figure 1. Notional view of costs committed vs. costs incured over time (from Ref 3)

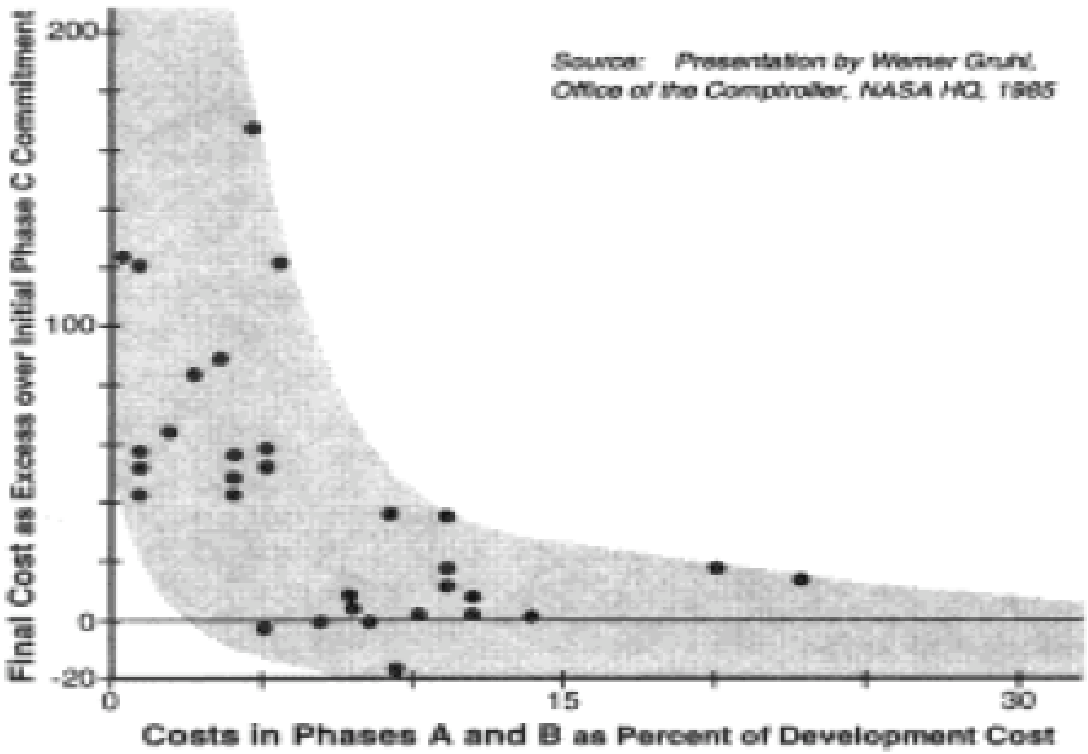


Figure 2. Overruns correlate with inadequate front end spending (from Ref 4)

There are good technical and historical reasons for current practices. The overwhelming technical reason is that, if done competently and with sufficient resources (see Figure 2) they work. Systems engineering practices growing out of the aerospace and defense industries of the 1950's and 60's have allowed the creation of systems of unprecedented complexity and technical sophistication. Historically, they were developed in an environment of relatively abundant resources and the attention of a large and highly competent workforce. Most systems were doing either unprecedented new missions, pushing the limits of performance, or incorporating new technologies – often all three at once. Performance and mission success, for national defense and prestige, were the driving motivations.

The historical basis for current practices in the aerospace industry, and an analysis of the structural changes that the industry has undergone, are covered in detail in chapters 2 and 3 of the Lean Aerospace Initiative book Lean Enterprise Value.[5]

1.2. Problems with classical architecting methods

The importance of good front-end work is clear. However, the methods for doing it are often ill-suited to the current environment and do not exploit the power of modern tools and computational capabilities. From Ross *et al.*:[6]

> Space system engineers have been developing effective systems for about fifty years and their accomplishments are a testament to human ingenuity. In addition to tackling the complex technical challenges in building these systems, engineers must also cope with the changing political and economic context for space system design and development. The history, scope, and scale of space systems results in a close tie with government and large budgets. The post-Cold War era has resulted in much smaller budgets and a space industry that needs to do more with less. Time and budget pressures can result in corner cutting (such as the Mars Program), and careless accounting (such as Space Station Program).
>
> Space system design often starts with needs and a concept. Engineers perform trade studies by setting baselines and making minor changes to seek improvement in performance, cost, schedule, and risk. The culture of an industry that grew through an Apollo race to the moon and large defense contracts in the 1970s and 1980s is slow to adapt a better way to design systems to ensure competitiveness in a rapidly changing world.
>
> Current approaches to creating aerospace systems requirements do not adequately consider the full range of possible designs and their associated costs and utilities throughout the development and lifecycle.[7] These approaches can lead to long design times and designs that are locally optimized but may not be globally optimized. This paper develops a systematic approach for space system design by addressing the following problems: 1) A priori design selections without analysis or consideration of other options; 2) Inadequate technical feasibility studies in the early stages of design; 3) Insufficient regard for the preferences of key decision makers; 4) Disconnects between perceived and actual decision maker preferences; 5) Pursuit of a detailed design without understanding the effects on the larger system; and, 6) Limited incorporation of interdisciplinary expert opinion and diverse stakeholder interest.

Ross *et al.* concentrate on the fact that current processes may not result in an optimal solution. Current processes are also badly disrupted by changes in environments and/or user needs. If the technology used on a subsystem changes (due to lack of readiness, for example), the effects on the other systems, and the ability to meet requirements, "ripples out". If a top level requirement changes, changes flow down to all subsystems, and then the effects of the changes on interfaces and system integration must be considered. Such disruptions take time, and may result in a "patched" solution which is not optimal (even locally).

Examining Figure 1, we would like a process that would put off the commitment of program costs as long as possible, maintain management leverage as long as possible, and increase knowledge as quickly as possible, while not increasing costs incurred. In light of the above comments, we would also like it to avoid early a priori design selections, include the preferences of key stakeholders, and increase knowledge specifically of technical feasibility and system interactions, while remaining flexible to changes in environments and/or user needs. MATE –CON is an attempt to create such a process.

2. OVERVIEW OF MATE-CON PROCESS

Here we will walk through the process used in MATE-CON. The intent is to give the reader a conceptual understanding of the method and its aims so that the examples and lessons covered in the next sections can be understood, and the advanced methods covered later in the book put in an overall context.

2.1. Purpose

MATE-CON is a process for understanding both the possibilities and the difficulties when looking for solutions to complex problems. Its intent is to allow informed upfront decisions and planning, so that the detailed design process which follows is aimed at the right solution, and is forewarned of potential problems and forearmed to seize potential opportunities.

The MATE-CON process is intended to be used in the early stages of product development. Figure 3 shows the overall product development process from concept to production ramp-up; the MATE-CON process is aimed at the initial steps. This does not preclude its use for other purposes – our SpaceTug example is an example of the exploration of a capability, which proceeds only as far as concept development, while others have pushed the Concurrent Engineering idea (the CON of MATE-CON) to the production of hardware.

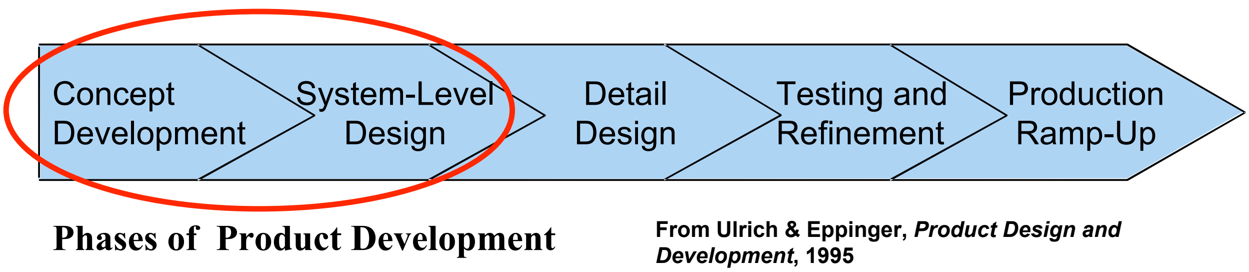


Figure 3 MATE-CON addresses early phases of product development.

The Multi-Attribute Tradespace Exploration (MATE) is a model-based high-level assessment of many possible solutions to the problem to be considered. Ideally, the full sweep of possible solutions to the problem are considered here. The key purpose of this step is to avoid premature concentration on a point solution. A prematurely selected point solution may be simply a poor choice. It may be an acceptable solution which nevertheless misses the possibility of a better value solution. It may even be a solution that is optimal given the information available at the conceptual development phase, but which is not robust to changes in environments or user needs. MATE gives the early decision makers a basis to explore a large number of solutions and their adaptability to changes. It allows this through the quantitative consideration of many aspects of uncertainty, including environmental or user needs changes, technical developments, policy changes, and market instability. It also provides a quantitative way of assessing potential capabilities (of, for example, proposed or hoped-for new technologies) through the use of what-if scenarios.

In this section we will walk through a quick introduction the steps in the process. The MATE process, exploration of the resulting tradespace, an the Concurrent Engineering process will be described with detailed step-by-step instruction in how to carry them out in the following sections. All will be illustrated with two examples: the X-TOS ionospheric explorer vehicle, and the SpaceTug orbital transfer and servicing vehicle.

2.2. *Background and Origins*

For a brief description of the intellectual origins of MATE-CON as a method, see Reference 8. The referenced paper provides an overview of both the MATE-CON method and a series of other papers covering detailed topics within the method. The techniques used in each of the following sections will be referenced as they are introduced.

2.3. *MATE*[8]

Figure 4 shows the conceptual flow of the MATE process.

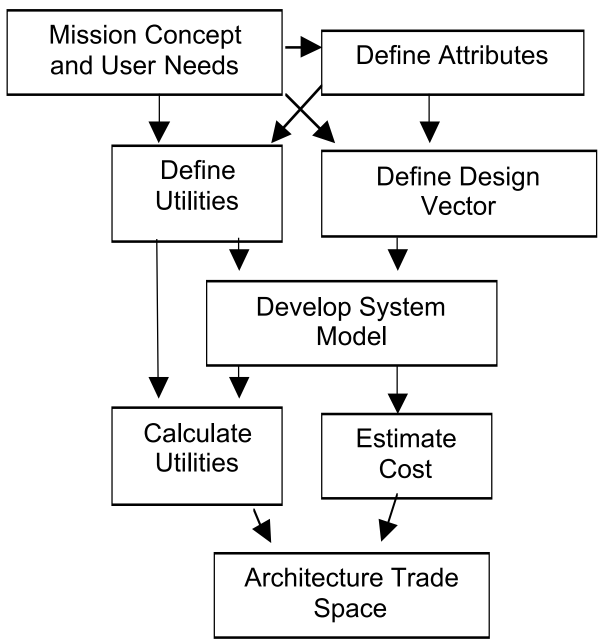


Figure 4 High level description of MATE process

The first step is selection and bounding of a "mission concept." Here, the basic issue to be addressed (i.e. the user needs to be satisfied), the broad scope of the solution space (i.e. what kinds of systems will be considered) and the scope of the analysis to be performed must be decided.

The next step is a critical one for reducing qualitative user needs to quantitative metrics. A limited number of *attributes* of the system need to be specified. Attributes have been described as "what the decision makers need to consider" and/or "what the user truly cares about"; they must also be quantifiable, and capable of being predicted with reasonable fidelity by fairly high-level models. It is usually the result or effects of the mission that concern the user, not the characteristics of the system designed to carry it out. This lack of concern for the physical system is typical, and illustrates a key feature of the entire method—that it is driven by a set of quantified user needs, rather than requirements pertaining to a specific system. The attributes ideally need to be complete (capture all important user needs) and independent; this is sometimes hard to accomplish at the beginning of a study, and the attribute list will sometimes evolve during the process.

Once a list of attributes is settled on, a formal MAU process is used to determine the *utility* to the user of values of each attribute. These individual utilities are then integrated into an overall utility. In both cases, "utility" is a dimensionless metric of "goodness" that is customarily normalized to be between 0 (no user needs satisfied) and 1 (delighted user). In some cases, this metric can be given units (e.g. cost per billable minute of a broadband telecom system), in others (e.g. usefulness, to scientists, of scientific data) it can only be used as a relative metric. In the latter case, interpretation of these metrics is somewhat dangerous—a higher metric is better than a lower one, but a utility of 0.5 may not be "half as good" as a 1.0, nor a 0.99 "only 1% off" from a 1.0. Such interpretations usually require returning to the individual metrics, or the decision makers. The single utility chart on the left side of Figure 5 reflects (and quantifies) the fact that lower altitude data is more useful to the users, with an premium on very low altitudes.

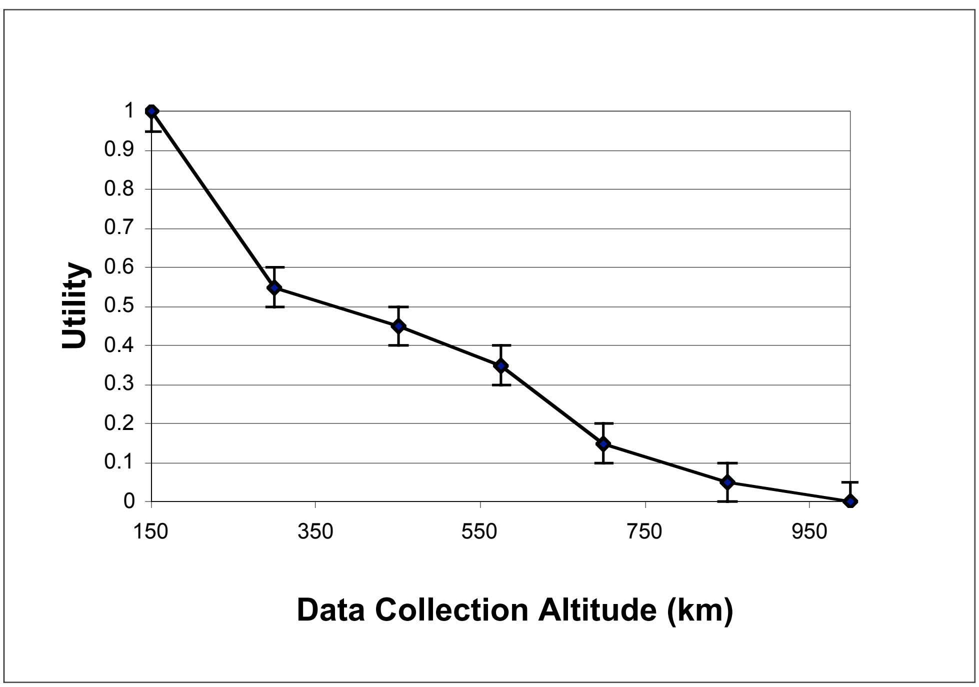


Figure 5 A single attribute utility curve, showing utility of data colleted declining with altitude of the vehicle doing the collecting.

The *design vector* is a list of variables that define the system architecture. To keep the analyses tractable, this vector must be limited to those variables that will have the largest effect on the attributes. The design vector may need to be revisited as the models mature.[9] Often, the exercise of picking the design vector is one of exclusion, as variables of undoubted importance in the final design are excluded from the initial studies.

The system *model* has a well-defined and straightforward goal—calculate the attributes given a set of specific values of the design vector. There is no one best way to do this modeling, but experience has indicated that a few commercial tools (e.g. Analytical Graphics' Satellite Tool Kit® (STK), and simple analysis techniques (e.g the methods in *Space Mission Analysis and Design* (SMAD)[10]) are of appropriate fidelity. Other models may have to be custom-developed for specific applications. These models can be linked to automate, or at least partially automate, the analyses, allowing large design spaces to be analyzed efficiently with commonly available computer resources.

The results of the modeling are then reduced to utilities and costs. Utilities are calculated using the formalisms of MAU theory. Cost is estimated based on the best (and most appropriate) available cost model. The cost models are known to have low fidelity and also to disagree by large factors.[11,12] Interpreted correctly, the calculated cost should be viewed, like the utility metric, as a ranking rather than an absolute and correct value. If the cost models are used in this sense, the danger to watched for is incorrect *sensitivities* in the cost models, which would cause the relative costs of design options to be incorrectly ranked. To date, it has been found that this has not been a major problem *at the level of fidelity of the analyses used*. In general, for example, more complex designs have resulted in bigger and heavier vehicles requiring larger launch vehicles, and hence more expense to build and launch; the current models capture this trend.

The result of the analysis is a database of the *trade space*, with thousands of potential architectures mapped to the resulting attributes, utilities and costs. This database is the basis of the exploration phase—the learning of the lessons that the process has uncovered. At a minimum, the desire is to reduce the trade space to designs worth considering, uncover the controlling physics or other constraints, and uncover the key design trades.
Data visualization and manipulation techniques are usually needed, along with patience and curiosity, to understand the complex lessons of the design space. MDO methods may be very useful, and indeed necessary, for exploring very large design spaces.[47,48]

The region of the trade space where attention should be focused are the designs that, for a given cost, produce the most utility (or, conversely, that produce a given level of utility for minimal cost). This region is referred to as the Pareto front. Designs that are not on the Pareto front are said to be "dominated"—better designs are available at the same or lower cost. Choosing between designs on the Pareto front means making real trades—better utility for greater costs, or trading one desired utility against another.

Figure 6 shows one possible slice of the trade space—the combined utility plotted against the total mission cost. In these plots, each point represents a potential architecture. The Pareto front is clearly visible to the upper left—the few dozen architectures that give the maximum utility for a given cost. This plot does NOT uncover the controlling physics—that takes considerably more delving into the database. By comparing the designs along the Pareto front, and perhaps carrying out some additional sensitivity studies on designs on the front, the key design drivers that move the design to the front, the key design trades that move a design along the front, and the key physics that prevents designs from getting any better than those on the front can be determined.

Figure 6 shows a deterministic tradespace. The values of the utility and cost are assumed to be known accurately. In many cases, uncertainties from many sources can make the exact positions of each of the points in the trade space uncertain. Early in the design process it is not unusual for user needs, the performances of various technologies, or their actual costs to be unclear. The simple models used in the tradespace analysis may introduce additional uncertainties or inaccuracies. Finally, there may be uncertainties or risks inherent in the mission to be performed. All of these can be included in the tradespace analysis using tools to be explored later in this book. As an introduction, consider the first twelve pages of an unpublished paper by Hastings, Weigel and Walton.[13]

The process in Figure 4 is shown as being sequential, with each step following the previous one. In practice, as the process proceeds, circumstances can change, or knowledge can be gained that changes perceptions, causing earlier decisions to be called into question. For example, user needs can to shift late in the process, or the choice of attributes or design vector can change based on knowledge gained during analytical model development. The process is quite robust to iterations, however. The major time commitments are to getting "up to speed" on the proposed system and its related technologies, and building the analytical models. If the user needs, utilities, attributes, or design vector change, the process can be repeated relatively quickly by modifying the analyses as necessary and rerunning them with new inputs.

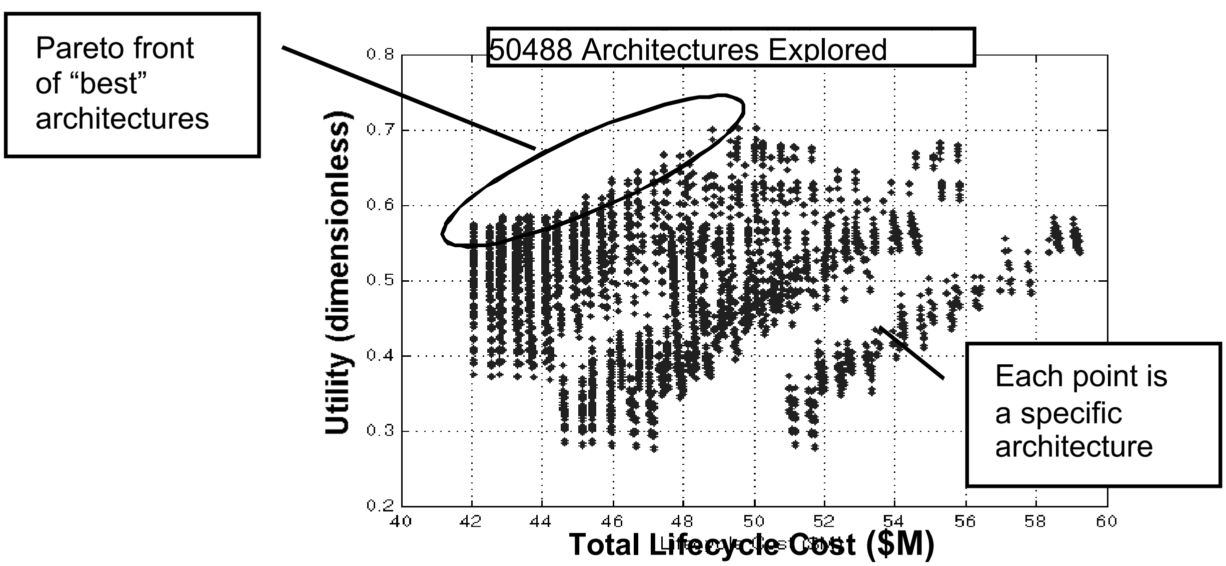


Figure 6 Combined utilities and costs of fifty thousand evaluated systems

Once the trade space is explored, an architecture or architectures can be selected. This may be the optimum architecture as determined by the analysis, i.e. the one delivering the most utility for the minimum cost. More likely, it will be selected from a reasonable subset of architectures (usually on the Pareto front) by the designers and users based on a deeper exploration of the attributes of the architectures and the characteristics of the surrounding trade space. For example, architectures whose attributes are relatively insensitive to changes in assumptions or poorly controlled variables may be selected as being robust, or architectures that can be rapidly improved with additional resources or technology (even if they are not immediately available) may be selected as being versatile or upgradeable.

2.4. MATE-CON[14]

Once an architecture has been selected, rapid development of a design or set of vehicle designs is done using ICE. An interdisciplinary team with tools that communicate seamlessly through a common database does design *sessions* in physical or at least virtual co-location. Figure 7 shows the computer tools, referred to as *sheets*, linked to a server. Each tool is tended by a human operator who updates the tool as necessary (e.g. updates a CAD model), makes major design decisions that are input to the tool (e.g. changes the propulsion type), and provides common sense and wisdom unavailable to automated methods (e.g. breaks non-convergent behavior in the iterations). The combination of the human and the tended tool is referred to as a *chair*. The tools perform rote calculation (e.g. rough sizing of solar panels), pass information, and sum up system characteristics (e.g. mass and power budgets) automatically with each design change. A *session* consists of inputting design changes and iterating the calculations (by having each *chair* execute its *sheet* in turn, tended by the human engineer as required) until stable values are reached for all major system characteristics. Design changes are tried until a design is found that satisfies all major requirements.

ICE design sessions typically last several hours and usually address one major trade per design session. A senior team member, or "facilitator," leads the design sessions and helps to resolve disconnects between the clients. The design sessions are iterative, with each subsystem sending and receiving many times in order for the point design to converge. Although it has recently become possible to automate this iterative process, human operation of the client stations is almost always preferred. The human element is actually key to the method. The human expert can guide the iterations, catching bugs, nonsensical answers, divergence, and other pathologies that complex computational systems are prone to. More importantly, the experts make major discontinuous design decisions, or go "outside the box" by stretching parameter ranges or even adding new computational capabilities, making the ICE method a true design tool, not just a non-linear equation solver.[15]

The steering by the session leader is based on a combination of traditional system requirements and user inputs. The latter are ideally provided by direct user/customer involvement in the ICE session. ICE becomes MATE-CON with the inclusion of a MATE chair that has the results, and often the models, of the preceding MATE effort at his or her fingertips. The MATE chair can quantitatively assess the progress of the design not just towards meeting requirements, but towards maximizing the overall utility of the system containing the design. He or she can also help the user/customer translate needs into design changes, and thus steer the design changes towards "sweet spots" in the trade space. Finally, in the absence of a customer present throughout the session (or the absence of one of several decision-making stakeholders, which is likely) the MATE chair can provide a surrogate presence, assuming the stakeholders will in the end desire the maximum utility.

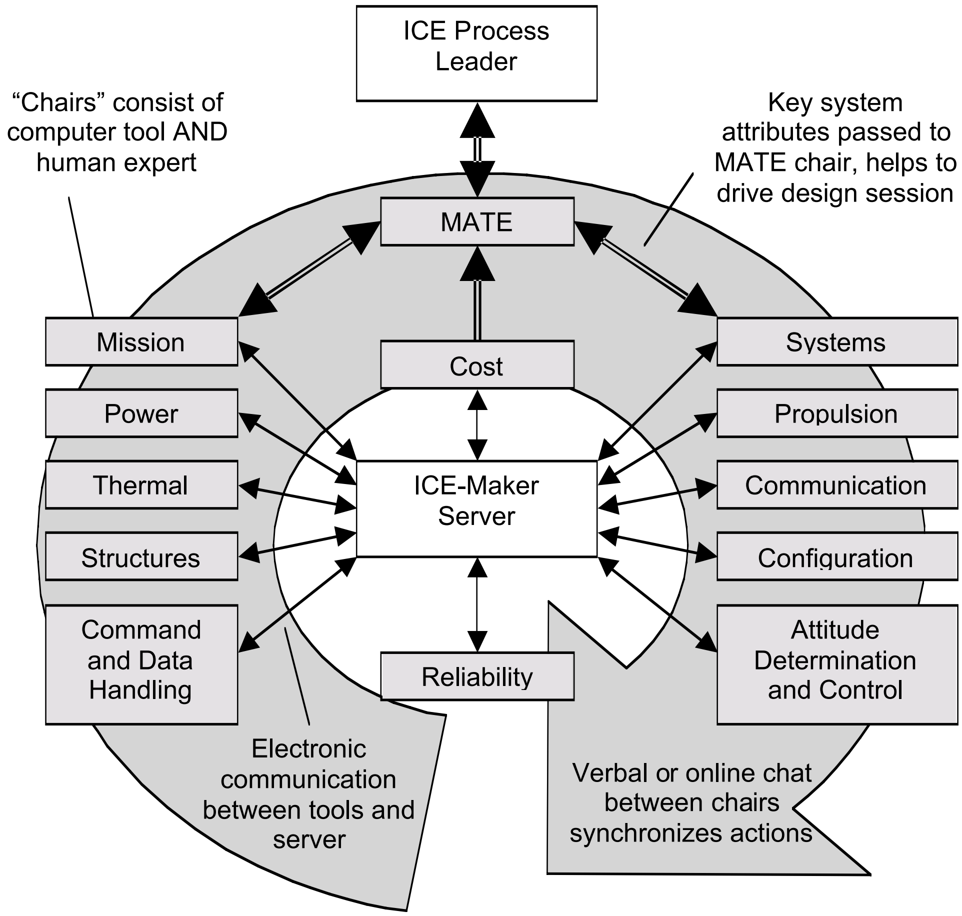


Figure 7 Overview of ICE process

The typical results of an ICE session is a design or designs at a level of detail somewhere between a conceptual design and a preliminary design. In the examples considered here, spacecraft are designed to a conceptual design level, with some additional detail in key systems. Figure 8 shows a typical spacecraft layout, with mass and power budgets, which are the typical outputs reported from an ICE session. More detail often exists with the ICE "sheets" which can be extracted as desired (see Figure 9), although at this stage of design the accuracy and relevance of more detailed information should be carefully considered.

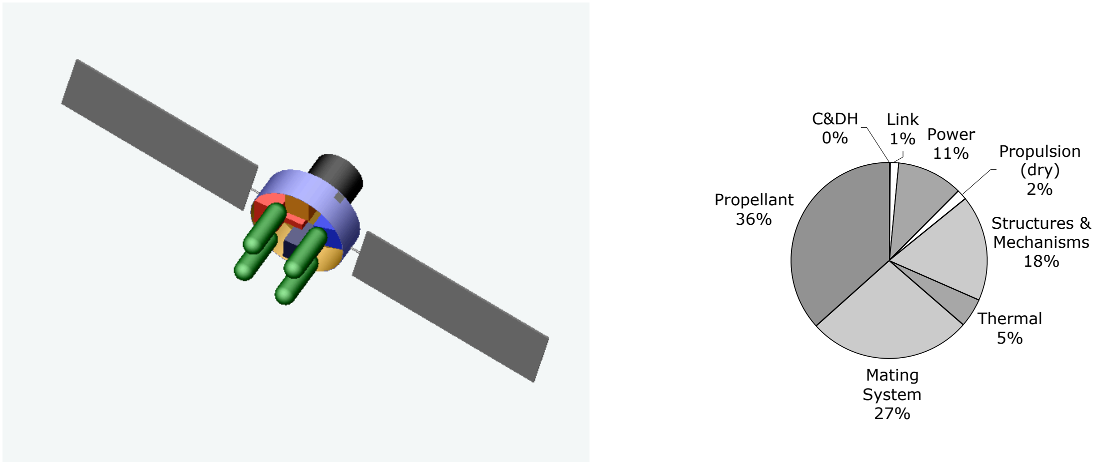
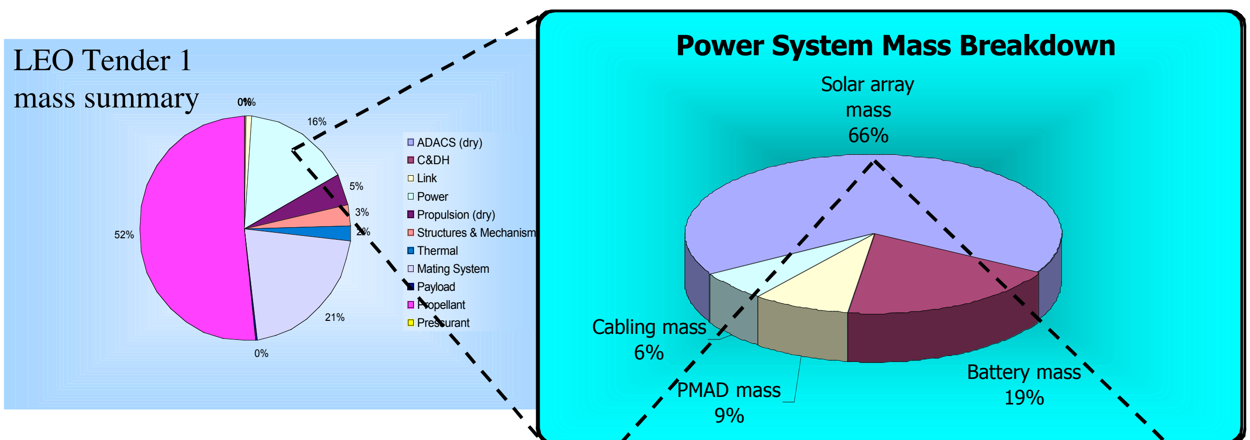


Figure 8 Typical ICE output: vehicle configuration and mass budget for an electric propulsion orbital transfer vehicle.[16]

ICE methods can be used for more detailed design studies, up to and including creating hardware drawings and/or CAD tapes. They can also be used for higher-level, "systems-of-systems" studies. At least with current technology, there is a practical tradeoff between the fidelity of the study and its scope; simpler systems (e.g. instruments and other sub-components) can be designed in detail, while complex systems are typically designed only to the preliminary or conceptual level. [17]



Select solar array material:	Triple Junction (InGaP/GaAs/Ge)	
Minimum efficiency	24.5	%
Maximum efficiency	28.0	%
Nominal temperature	28.0	C
Temperature loss	0.5	%/deg C
Performance degredation	2.6	% / year
Minimum temperature	0.5	C
Maximum temperature	85.0	C
Energy density	25.0	W / kg
Solar array mass	150.6685167	kg
Total solar array area	9.965098159	m^2
# of solar arrays	2	#
Individual solar array area	4.98254908	m^2

Figure 9 Example of details available within sheets after ICE study[18]

2.5. *Notes on terminology, requirements, and limits*

Process terminology[14]

The terminology used for the methods described here is far from stable. In this work, the architectural-level trade space exploration is referred to as MATE, the rapid conceptual design process as ICE, and the integrated process as MATE-CON. The MATE method is an expansion of the Generalized Information Network Analysis (GINA) method, and many of the publications that preceded this work refer to the GINA method. GINA includes the system modeling and trade space exploration aspects of MATE without the front-end of a generalized multi-attribute utility method, and is specialized for systems that are primarily focused on information transfer, but has been used generally in a similar fashion to MATE. Other researchers working on similar methods have used terms such as Collaborative Engineering, Collaborative Optimization and, to describe the laying out of a tradespace for the user to select from, "Design-by-Shopping."[192021]

The techniques for Concurrent Engineering referred to here as ICE go by a number of names. Concurrent Engineering, or the "Design Room" method, are commonly used. The best known examples (from which this work directly descends) are The Jet Propulsion Laboratory's Advanced Projects Design Team (Team X),[22] the related Next Generation Payload Development Team (NDPT, or Team I) [23,24] and the Aerospace Corporation's Concept Design Center (CDC).[25]

One section of this work refers to MMDOSA, which is a complement to MATE: it is a rigorous process for exploring extremely large trade spaces with multi-disciplinary optimization techniques. Others may refer Stanford's separate SAM framework, which includes a quantitative risk analysis model of the not only the physical system, but also management decisions made during the design effort.

On requirements[14]

The methods described here take place at the beginning of the system design process. Traditional product development process descriptions often identify "establish requirements" as this first step, so it is natural to ask how the current method interacts with the determination of system requirements.

The present method can be thought of as a powerful tool for coming up with the right requirements at the right time. It has been noted that current processes are not efficient at coming up with requirements, the resulting requirements do not necessarily provide a good statement of user needs, and the potential value of the system (and even its physical feasibility) are not well reflected by the requirements.[26] To this, we add the observation that most requirements are written with a solution to the design problem in mind, and hence reinforce the premature narrowing of the design space that we attempt to avoid. For these reasons, requirements determination is replaced by the much more general collection of user utilities in the MATE process.

Requirements for the space vehicle to be designed in the ICE process can be generated at the conclusion of the MATE process. However, by including MATE and risk chairs in the ICE process, the richness of the knowledge the user utilities, vehicle robustness, and the interaction between the vehicle and the rest of the system can be preserved into the conceptual design process, providing more flexible guidance than a set of fixed requirements.

At the conclusion of the MATE-CON process, on the other hand, sufficient information is available to write very good requirements for the detailed design of the vehicle. This capability is key to avoiding classic requirements traps. The utilities capture the needs of the key stakeholders, without which instability is likely. Trade space knowledge allows avoidance of both physically unrealistic requirements, and requirements that artificially preclude the best solutions. System interactions and program and technical risks can be estimated; they are very difficult to determine requirements for *a priori*. Finally, although flexibility and upgradeability are clearly key to modern acquisition models (e.g. spiral development) there is little experience in writing requirements for them. Most historic examples of flexible systems are serendipitous.[27] The present method can aid in understanding flexibility issues through understanding of the trade space. Designs can be specified which can be improved to provide enhanced utility with reasonable expense, risk, and/or need for technology advancement.

Limits and Caveats[14]

The MATE-CON method is a useful tool for architecture selection and conceptual design, but it must be used with a full understanding of the limits of the method and its component parts. The method requires careful selection of the attributes and design vector – these define the problem that is being addressed. Changes in these selections late in the process may require substantial "rework." The definition of the trade space requires models with the right fidelity. They must capture the factors that differentiate the architectures under consideration without being computationally intractable or excessively difficult to prepare and integrate. They must also have the correct precision given the uncertainties involved. Highly precise calculations based on sweeping assumptions will give misleading answers. If the problem is dominated by uncertainties, these uncertainties will have to be considered as part of the trade space analysis. Particular care must be given to the use and interpretation of cost models, which are unlikely to give very accurate absolute results. The key is to assure that the cost models used provide the right relative answers, discriminating more expensive options from less expensive ones. The utility models must also be used with care. Ideally, real users, acquirers, and other stakeholders should be brought into the process as often as possible, to prevent the creation of utility functions based on poorly captured or shifting user needs.

2.6. Running Example one: Terrestrial Observer Satellite X (X-TOS)[14]

The X-TOS project, originally a graduate space systems design exercise at MIT, designed a mission for collecting information about the Earth's ionosphere necessary for the updating of the AFRL atmospheric drag model. The project was motivated by the poor quality of current atmospheric drag models when used for predictions of re-entry time and location for uncontrolled bodies such as spent satellites.

Figure 10 shows the MATE process as carried out for the X-TOS project. The X-TOS project was scoped fairly narrowly—the customer needed a system that could deliver and support a set of three pre-existing instruments designed to take *in-situ* measurements of ionospheric conditions. The solution space was restricted to conventional-technology space vehicles, and the scope to the design and operation of these vehicles. A single AFRL scientist, representing the users of the data, provided the user utility; other stakeholders were not considered.

The attributes of interest to the user were all characteristics of the data collected: its time span (time between the very first data point collected and the very last), altitude, maximum latitude, latency (from collection to useful presentation to user), and the percentage of the data collected at or near the equator.

The solution space (design vector) was reduced to a set of choices of mission design, e.g. how many vehicles and when they are flown, orbit elements, and some simple vehicle characteristics.

For the simulations, both STK and student-written orbital calculations were carried out; spacecraft characteristics were calculated based on SMAD, and a launch module (selecting the best launcher for a given orbit and vehicle) was written based on an existing database of launch vehicles. These modules were used to build a database of the attributes of single vehicles in given orbits; for multi-vehicle mission designs these attributes were integrated over the lifetimes of the multiple vehicles. A design room with multiple personal computers considered powerful by the standards of the year 2001 was used to do the calculations. They took only hours, and in fact were entirely repeated on short notice late in the project due to a shift in user preferences.

A MAU model was used to calculate the utilities of each architecture. Costs were calculated using a hybrid of the cost estimation model in SMAD and NASA's Space Operations Cost Model (http://www.jsc.nasa.gov/bu2/ SOCM/SOCM.html.

In the case studied, drag at the low altitude where the most valuable data could be collected limited mission life, becoming the key physical constraint, and also setting up the key trades—increased lifetime for either increased altitude (and hence reduced data utility) or increased vehicle weight (and hence cost) for added maneuver fuel. These trades are visible on the Pareto front—short lifetime missions are somewhat cheaper at a penalty in utility.

The reduction of the design considerations to the key trades allowed the user a greater perspective into what was possible and desirable for this mission. This additional perspective in turn altered the users preferences, resulting in an updated utility model. In a demonstration of the adaptability of the process, this change in user preferences at the *conclusion* of the process

The MATE trade space was used to drive an ICE session to design vehicles for X-TOS. The ICE vehicle design trades reflected the MATE trades of orbit and re-boost fuel capacity versus cost, lifetime and the usefulness of the data collected. The designs, one of which is shown in Figure 11, illustrated the consequences to the vehicle of the trades that were discovered as abstractions in the MATE part of the process. Note the large fuel tanks required by the need for sustained low-altitude flight.

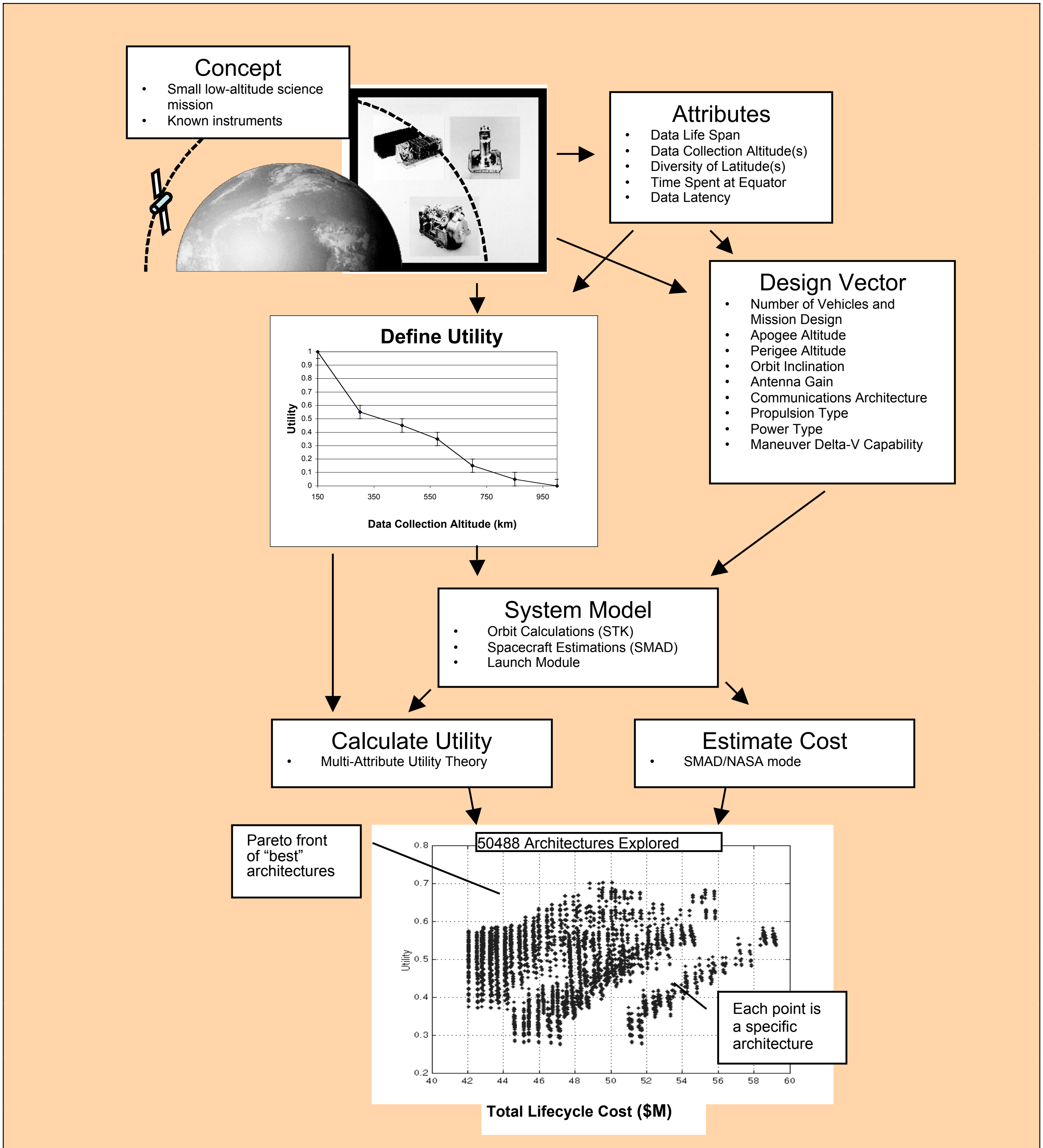


Figure 10 MATE process for X-TOS

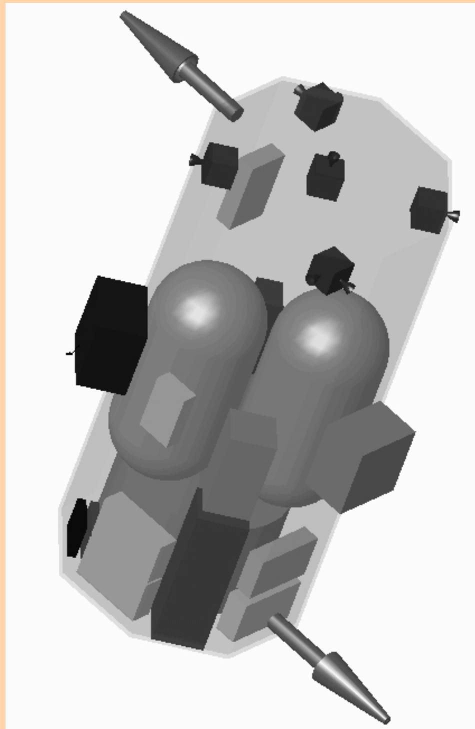


Figure 11 ICE result: X-TOS vehicle CAD model

2.7. Running Example two: general purpose orbit transfer and servicing vehicle (SpaceTug)[28]

The SpaceTug project was carried out by a team of undergraduate and graduate students, postdoctoral and staff researchers, and faculty in a single summer. It was the first use of the MATE-CON method under contract with a government sponsor. The aim was to explore the tradespace of possible orbit transfer and service vehicles, looking for potential cost-effective capabilities that might be of national interest.

The space tug concept is for a vehicle or vehicles to loiter in earth orbit and carry out multiple missions involving visiting existing assets in orbit and observing, servicing, or moving them. The project was motivated by a general interest in such systems as a national capability, and poor results when proposing such systems for specific missions, without looking at the wider tradespace of possible uses and designs.

Figure 12 shows the MATE process as carried out for the SpaceTug project. The project was scoped widely, as the possible uses for such a system are not currently know. A somewhat simplified version of the MATE method was used. The method was adapted in response to difficulties including the lack of an immediate customer and a very open design space. The customer utilities were handled parametrically to understand the sensitivities of the tradespace to ranges of, and changes in, user needs. The analysis was done at a high level, using low-fidelity models, but covering a large range of possible designs.

The capabilities of a SpaceTug vehicle determined to be useful to a potential user include: (1) total delta-V capability, which determines where the SpaceTug can go and how far it can change the orbits of target vehicles; (2) mass of observation and manipulation equipment carried, which determines at a high level what it can do to interact with targets, referred to here as its *capability*; and (3) response time, or how fast it can get to a potential target and interact with it in the desired way.

These attributes are translated into a single utility function. In the absence of real users from which to collect more sophisticated functions, it was decided that a simple function that could be explored parametrically was most appropriate. The utility was a weighted sum of utilities from the three attributes above, with the weights being considered parametically. The figure shows a single-attribute utility for Delta-V. In this case, utility is assumed to increase linearly with delta-V, with diminishing returns above the levels necessary to do Low Earth Orbit (LEO) to Geosynchronous Earth Orbit (GEO) transfers.

A set of design variables (in MATE parlance, a design vector) was selected to represent possible tug vehicles. The following variables were selected: (1) observation and manipulator system mass; (2) propulsion type, and (3) mass of fuel carried.

For the simulations, simple parametric relationships and design rules were used to compute the

The results revealed key constraints, trades, and promising types of designs. Chemical fueled tugs were severely limited, especially for higher-energy missions such as GEO transfer rescues, by the specific impulse of the fuel. Alternate propulsion concepts had other limits: electric propulsion (which is slow) was highly sensitive to the assumed utility of timely response, and nuclear propulsion results in high base costs. Independent of propulsion system, low weight grappling, observation, and control equipment was always desirable.

The tradespace analysis reveals three classes of potentially useful space tug vehicles. The Electric Cruiser occupies the “knee in the curve” for our nominal utilities, providing good value for cost. The “Nuclear Monster” is only design that can meet a desire for a high delta-V, high capability, rapid response system; electric monsters (not shown) might be interesting to users not interested in rapid response time. A final range of vehicles occupies the lower left region of the Pareto front. These are cost effective vehicles build using existing technology (e.g. storable bi-propellant systems) that can do a variety of jobs requiring lower delta-V. They could, for example, tend set of vehicles in similar orbits, doing a variety of maintenance tasks. For this reason (and to extend the naval support vessel metaphor) they have been dubbed “Tenders.”

The MATE trade space was used to drive an ICE session to design a variety of tug vehicles. Several “cruiser” vehicles were designed. From on or near the Pareto front, electric cruisers such as the one shown in Figure 8 were designed. High delta-V chemical propulsion vehicles are not optimal according the MATE analysis; the ICE results (which had difficulty closing because of extreme fuel loads) helped to illustrate why. Finally, a variety of Tender vehicles were designed; some for specific missions and some for generic service; these designs showed that a modular approach to tender vehicle design might be the best approach.[29]

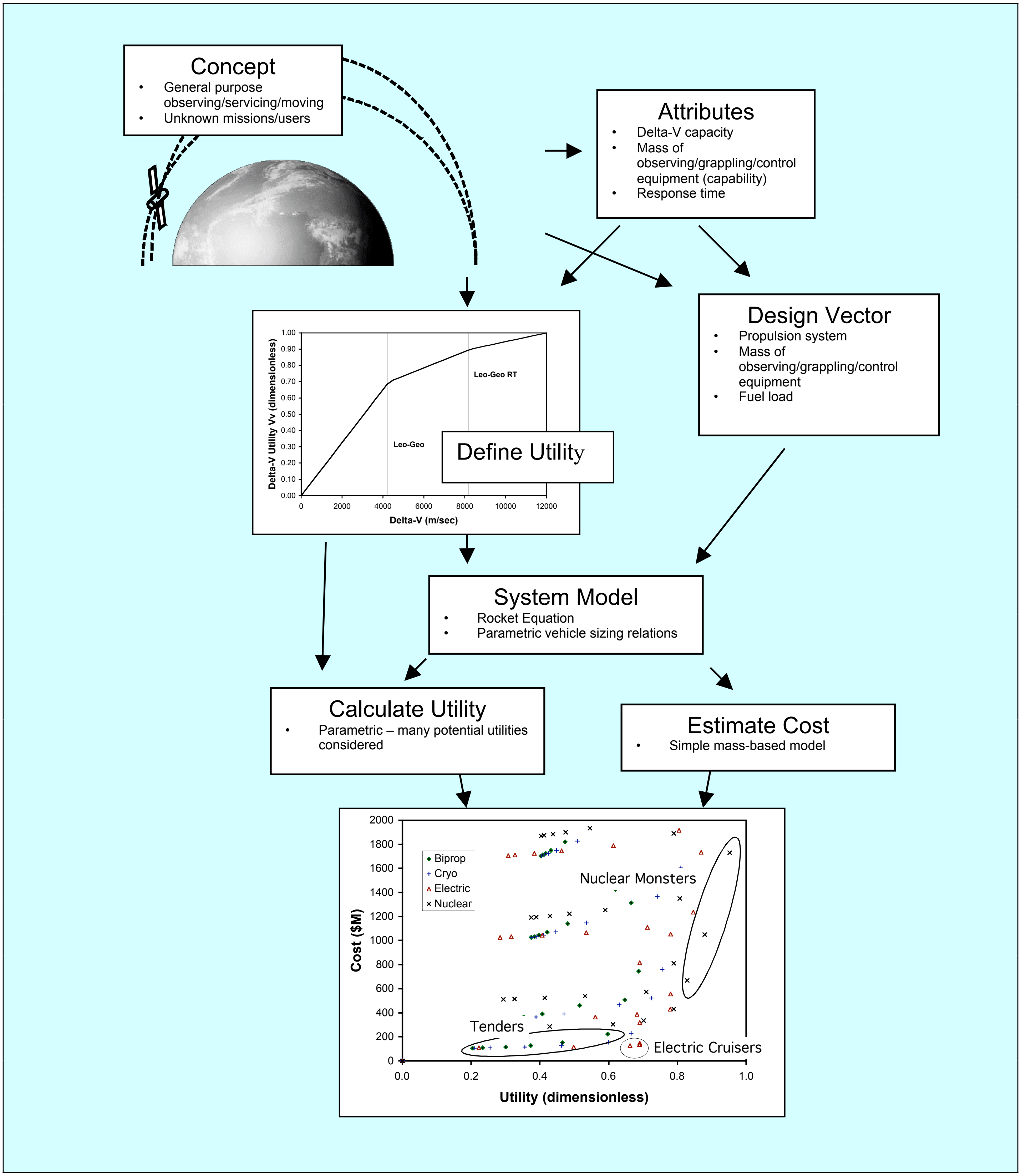


Figure 12 MATE process for SPACE TUG

3. DETAILED DESCRIPTION OF MATE-CON PROCESS

The following sections will describe in considerably more detail the steps in the MATE-CON process. The sections will cover the major steps identified in the previous section. The major MATE steps are shown in Figure 4, and are the primary emphasis of the next sections. The reduction of the tradespace and the building and running of an ICE model will also be covered.

Each step will be broken down into several tasks. They will be presented in an order designed for teaching; this order is also reasonable for implementation. A slightly different order is included in the task checklist in the MATE Short Book, by Ross and Diller.[30] The order of the tasks may be adjusted, and some tasks omitted or modified, as circumstances and the needs of your project require. Task ordering and its effects on process efficiency are discussed in a paper by Ross and Hastings.[31] Some of the tasks will also prove to be inter-related – one cannot be worked without the other – and will have to be worked in parallel.

Under some circumstances there may be some interdependencies between tasks in different sections. For this reason, it is important to have a reasonable understanding of the overall process, as covered in the previous section, before proceeding. With this understanding, it should be possible to work the following sections more-or-less sequentially.

The two running examples will continue in detail on a section-by-section basis. The two examples illustrate two quite different approaches to the process, in terms of the type of mission studied, the goals of the project, and the level of design detail and maturity. They are intended to provide the reader with ideas for implementation, rather than a rigid template.

4. IDENTIFYING STAKEHOLDERS, NEEDS, MISSION CONCEPT, AND PROJECT SCOPE

The first step is selection and bounding of a "mission concept." Here, the basic issue to be addressed (i.e. the user needs to be satisfied), the broad scope of the solution space (i.e. what kinds of systems will be considered) and the scope of the analysis (i.e. the boundaries of the system(s) to be considered) must be decided.

4.1. *Identify Need*[32]

First, a need which a new system might satisfy must be identified. The need identification activity involves identifying the initial impetus for the creation of a system. Additional needs may be identified throughout the process, but the initial need is the initial driver for the system. Without a clearly identified need, the designers may find it very difficult to resolve system ambiguity and create a useful product.

Need usually involves the addressing some problem with the status quo. Need can arise from almost any stakeholder with a problem. The key issue is to communicate the need, origin and context of the need to the decision makers in the system architecting and design process.

Typically, need identification is interdependent with the identification of the key stakeholders in the proposed system, discussed below. If the system is to be built in response to marked demand, or the desire by a government or agency for a specific capability, the needs of the market or agency must be well understood. In marketing terms, this is sometimes referred to as "responding to customer pull."[33] The X-TOS project is an example of a "pull" project. If, on the other hand, the desire is to create a *new* capability, the interested stakeholders may not be the final users. The motivation may be described as technology or concept "push", with a set of stakeholders interested in creating a capability, with the hope that if it exists it will create market demand. In this case, the needs of the interested stakeholders needs to be addressed, as well as the *potential* needs of future users of the system. The Space Tug project illustrates this case.

4.2. *Define System Concept and Scope*[32]

Along with the need, the basic, highest-possible-level system concept for addressing the need needs to be made explicit. Some a-priori choices about what types of systems are to be considered may be made here, depending on the needs of the project. Typically (this is a space systems book, after all) this might involve a choice that the need be addressed by a space system, using current or a select set of near-future technologies. Care must be taken, however, to not over-constrain the scope, especially in ways that might bias the solutions considered. The method is most powerful when considering high-level, open solution spaces. Therefore, consideration of space, air, or ground components, or advanced technologies, should not be dismissed out of hand.

Typically, space systems are part of more complex systems-of-systems. Their interactions with the larger systems in which they operate will have a strong impact on their architectures. These interactions must be made explicit, and if necessary made part of the system architecture study.

In order to make the problem tractable, however, it is necessary to define the boundaries of the system. Scoping the problem restricts the possible problem and solution space to something that can be specifically addressed by the designers. Scoping defines what is within and without the areas that are to be considered. Scoping should also involve the collection of explicit assumptions of the system, and explicit assumptions about its interfaces with the larger world.

The examples again provide contrasting approaches. X-TOS is scoped fairly narrowly in order to quickly arrive at a system architecture and vehicle design that will respond to the user's needs. Space Tug, on the other hand, includes a large space of possible solutions, but is scoped in terms of the aspects of the problem to be considered and the level of detail of the solutions to be developed.

4.3. Identify Stakeholders and Decision Makers[32]

In order to understand the true or potential needs for a system, the people, groups and organizations that have interests in the system and its end products must be identified. A stakeholder is a person or organization that has "a need or expectation with respect to system products or outcomes of their development and use." Examples of stakeholders include "acquirer, user, customer, manufacturer, installer, tester, maintainer, executive manager, and project manager… corporation… and the general public."[34] Given definition and examples of stakeholders, decision makers are a subset of the set of stakeholders, with the key distinguishing feature being the ability to influence the allocation of resources. Having direct control over the allocation of resources makes a stakeholder an obvious decision maker, however those stakeholders with indirect influence are not as obvious.

As an aid in understanding various upstream stakeholders typically not considered by the design engineer, a nominal framework of stakeholder and their relationships are shown in Figure 13. This framework is typical for a commercial or government procured vehicle built by a for-profit firm; the relationships will be different (but no less complex) for other types of systems.

Level 0 decision makers are classified as External Stakeholders. These stakeholders have little direct stake in the system, although they may be the ultimate beneficiaries of its use. They typically have control over policies or budgets that affect many systems. An example of an External Stakeholder for a space system architecture is Congress or the American people.

Level 1 decision makers include the Firm and the Customer. The Firm role includes those who have organizational stakes in the project and manage the Designers. This decision maker may have stakes in multiple projects, but has specific preferences for the system in question. An example of a Firm is an aerospace company. The Customer role includes those who control the money for financing the project. According to (Martin 1997), the Customer "is an individual or organization that (1) commission the engineering of a system, or (2) is a prospective purchaser of an end product." The Customer typically has preferences that balance product performance meeting User needs, cost of the system, and political considerations. This decision maker typically contracts to the Firm in order to build the system and provides requirements to the Designer.

Level 2 decision makers include the Designer and the User. The User role has direct preferences for the system and typically is the originator of need for the system. Need can originate within an organization, such as the Firm, as well. See Ulrich and Eppinger[33] for discussions on firm strategies and enterprise opportunities. An example of a User is a scientist or war fighter. The Designer role has direct interaction with the creation of the system and tries to create a product that meets the preferences of the Firm, Customer, and User roles. An example of a Designer is the system engineering group within the aerospace company building the system. The arrows in the figure depict the predominate direction of information flow, though some reverse flow does occur (requirements push-back, for instance).

The explicit task for this step is to identify the stakeholders and decision makers and their relation to the system under consideration. The template in Figure 13 may be appropriate, as it was in the X-TOS example, or a different set of stakeholders might need to be imaged, as was the case in Space Tug. Ultimately, the goal is to identify the end users whose needs are to be satisfied, the decision makers who control resources on behalf of the users, and the other stakeholders who may need to be satisfied in some way, who may create constraints on the system, or who may control resources useful or necessary to the systems success.

A much deeper exploration of the concept of stakeholders and their interactions can be found in Chapters 7 and 8 of the Lean Enterprise Value book.[5]

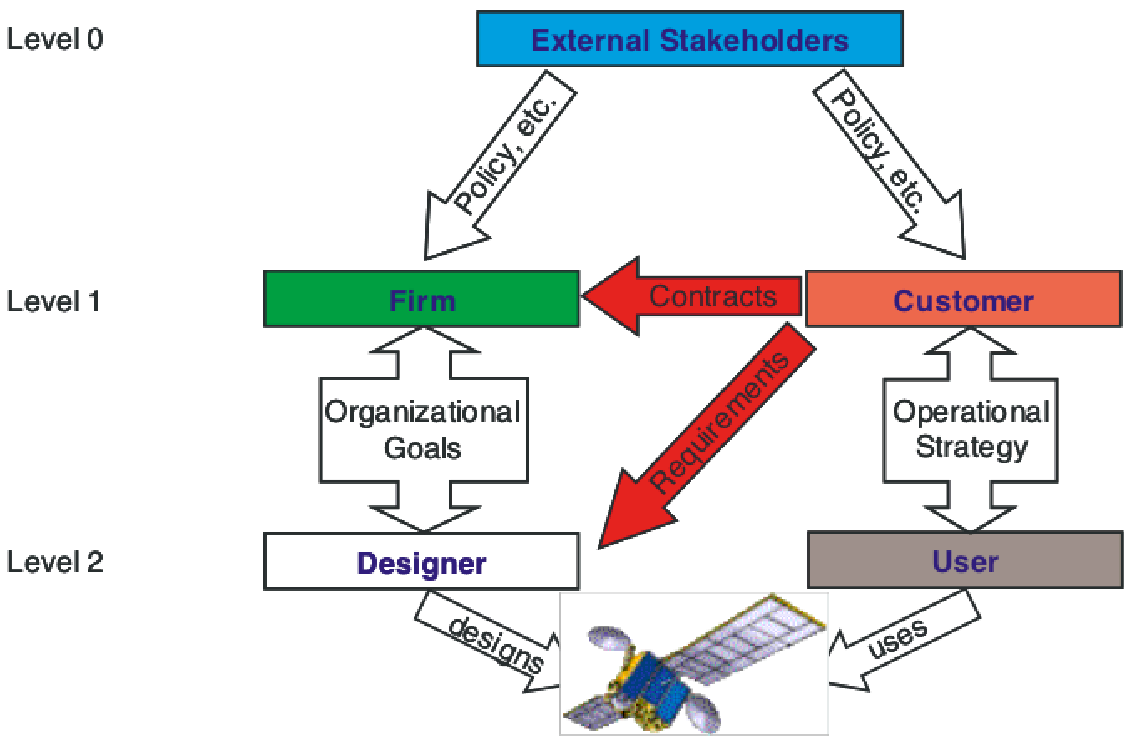


Figure 13 Stakeholder Framework

4.4. X TOS Need, Concept and Scope[32,35]

The X-TOS project was motivated by the need for improved predictions of drag on orbiting bodies. This drag is a strong function of the density of the upper atmosphere, which itself is a complex function of seasonal, solar cycle, and other conditions. The general purpose of the X-TOS mission is to collect information on the upper atmosphere to allow improved density predictions. The improvement of this forecasting ability will serve both the military and civilian communities. Militarily, it will provide data to permit more accurate modeling in three deficient areas – satellite tracking, close approach/collision avoidance, and orbiting body reentry prediction. From a civilian standpoint, improved reentry prediction will greatly enhance early warning capabilities for populated areas in the zone of impact.

A set of instruments has been developed by the AFRL to collect the necessary data on the upper atmosphere. A space vehicle is necessary to place these instruments in locations where they can collect data of interest to the scientists developing improved drag models.

The mission concept of the X-TOS project was set fairly narrowly, aided by the very specific needs of key stakeholders. Figure 14 shows a variety of space systems and instrument concepts that can collect information on the ionosphere. It was taken from and early study.[9] An *a priori* decision was made to concentrate on the a system that could use the instruments developed and built by the AFRL, which limited consideration to in-situ systems – systems that orbit through the regions of interest. Cost considerations also limited the number of vehicles considered to two. A programmatic decision was made to consider only independent missions – no possible sharing of a vehicle with other missions was considered.

The scope of the analysis of was also set fairly narrowly. Figure 15 shows the flow of information from the ionosphere itself to the ultimate users of the improved drag models. Ideally, the system would be optimized to meet those users needs. However, lack of maturity of the drag model and information distribution system precluded modeling of these aspects of the system of systems. Instead, the space segment up to the delivery of data to the ground was modeled, and the system was designed to optimize the delivery of data useful to the developers of the as-yet incomplete models.

The framework in Figure 13 was modified to capture the relationships for this project. The equivalent stakeholders were:

Designer:	Space System Design course students
User:	Air Force Research Lab (AFRL/Hanscom, Dr. John Ballenthin)
Firm:	Professors, staff
Customer:	Aerospace Corporation
External:	Eventual capability users (NORAD/USAF) and beneficiaries (public), existing space law and policy, policy setters (Congress).

The development was carried out in an academic (term project) environment. Although the results of the project were not actually pitched to the Aerospace corporation, such small science missions ultimately would have to be, so they were designated the customer. This project had the luxury of a well defined user who would ultimately make recommendations to the customer, so a single key decision maker (that user) was identified.

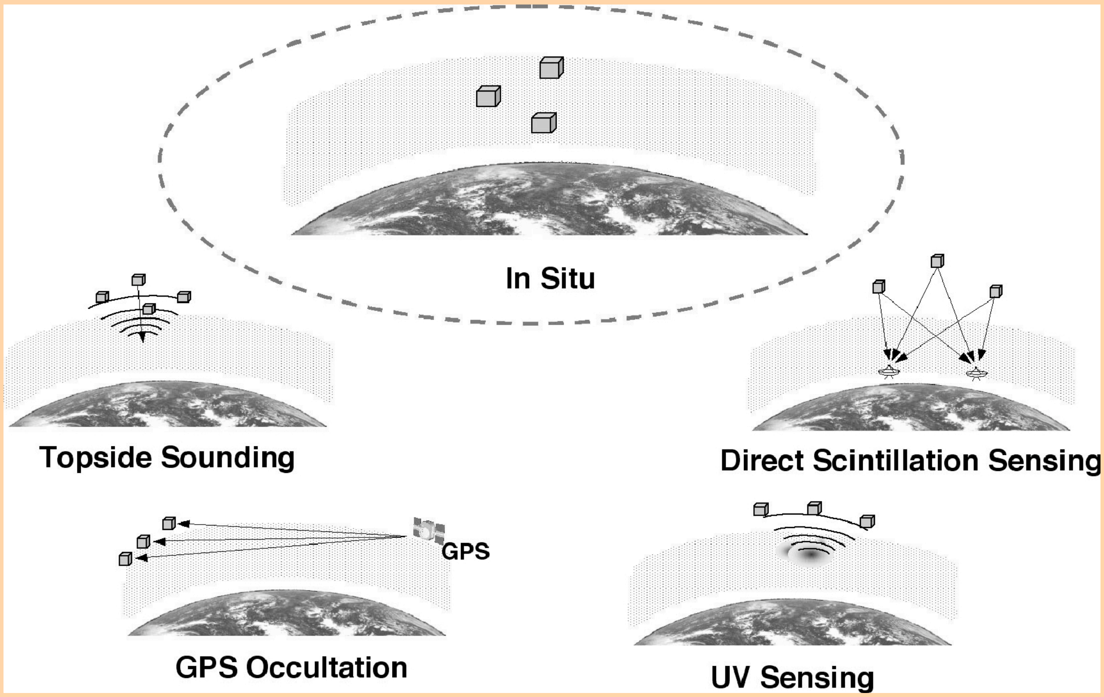


Figure 14 Techniques for collecting information about the ionsphere

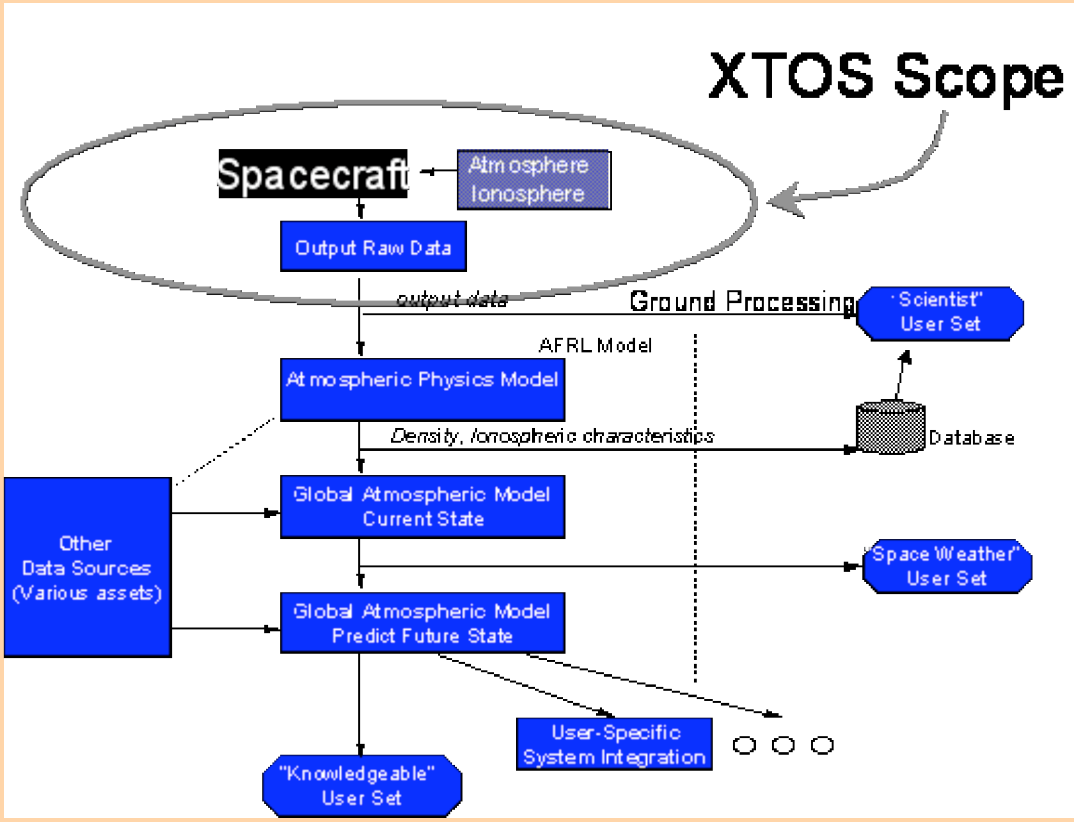


Figure 15 X-TOS system-of-systems with selected scope

4.5. Space Tug Needs, Concept and Scope[36]

The Spacetug project was motivated by the desire for a national infrastructure that would allow the observation, servicing, and moving of existing space assets. The users of such services are not currently well defined. Some missions can be defined based on existing assets and needs; others may emerge once such a capability exists. Among the potential developers and customers for such as system, there is a need to understand systems that maximize *potential* usefulness.

The mission concept of the Space Tug project was very open: a vehicle or vehicles capable of visiting a variety of orbits in near-earth space and performing unspecified jobs there, including changing the orbits of target vehicles. The scope of the analysis, on the other hand, was set fairly narrowly: only the vehicle bus was considered. Key issues involving the equipment and software necessary to perform orbital servicing and mating were considered in a separate study. This equipment was treated as a generic capability, which interfaced with the vehicle by having mass, and consuming power and communications bandwidth. Except for the communication issue, the command and control of the servicing equipment was also not considered. It was also assumed that existing infrastructure for launch, communication, and bus command and control would be used.

The stakeholder framework was challenged by the fact that there was no fixed user for the system. In this case, the customer (the Defense Advanced Research Projects Agency (DARPA), who was actually dedicating resources to the project) desired to create a capability of maximum usefulness to an uncertain user base. This situation was the opposite of the X-TOS project, which has a real user, but only a theoretical customer. The key decision maker in this case is the customer, acting as a surrogate for perceived future users. The development was carried out in an academic (summer project) environment. As this was a small research project, the Firm role was minimized. The equivalent stakeholders were:

Designer:	Project students and staff
User:	Unknown – commercial and government space users
Firm:	Host university and interested faculty
Customer:	DARPA
External:	Future direct beneficiaries (Military and civilian space users), indirect beneficiaries (US and world public), existing space law and policy, policy setters (Congress).

5. DEFINING THE TRADESPACE

5.1. Introduction[32]

Defining the trade space is the critical next step in the MATE-CON process. In this step, the user's (and possibly other stakeholders') *preference space* is defined. This preference space will be used to evaluate the members of a *design space*, which must also be defined now.

The preference space consists a number of *attributes* of the proposed system, and a set of *utility functions* that map the values of the attributes to user utilities. The attributes are functional descriptions of the outputs or outcomes of the working system, not physical features of the system itself. They must be carefully chosen to correctly represent the aspects of the system that the user cares about, and must have certain features (such as perceived independence) that will allow them to work as bases for a utility analysis. The utility functions are dimensionless representations of the relative desirability of various values of the attributes. Together, the attributes and utility functions define a wide range of functional system outcomes, evaluated in terms of their utility to the user(s).

The design space consists of a number of design variables that can be varied to define a large number of possible physical designs that might create the desired outcomes. Given the infinite choices facing a clean-sheet designer, the design space is a reduction of all possible design solutions to a tractable number of them. The selected design variables are referred to as the design vector; ideally these are the variables that are under the designers' or system engineers' control and have a large impact on the attributes of the system. The use of the term "design space" should not be interpreted to imply anything about the level of detail of the problem; a high-level architecture or a low-level component design can be considered equally well by the method.

Ultimately, the goal of the MATE process is to map the design space to a space of evaluated designs (the solution space or, often, the trades pace). This is accomplished by using simulation modeling to predict the attributes of all of the designs in the design space, and then using the preference space to understand the utility of the designs to the user. A rough view of this mapping is shown in Figure 16.

Two caveats are immediately in order. One is that the choices of attributes and design space define the bounds of the trade space. Incorrect choices will result in a trade space that does not reflect the users true needs, or does not contain the best solutions. Thus it is both important that the attributes and design vector be chosen carefully, and highly likely that these choices will need to be revisited as the project matures.

The other caveat is that the entire process represents an *approximation* of the users desires and the capability of various designs to fulfill them. The preference space approximately quantifies the users' needs; the design space represents a tractable subset of the infinite possible solutions; the simulation space is a tractable model of the proposed systems in action, and the solution space is the necessarily approximate result. At an early stage in the design or architecting process, these approximations are necessarily rather coarse. It is therefore vital that the trade space analyst understand these approximations, and understand the significance (or lack thereof) of the trade space results.

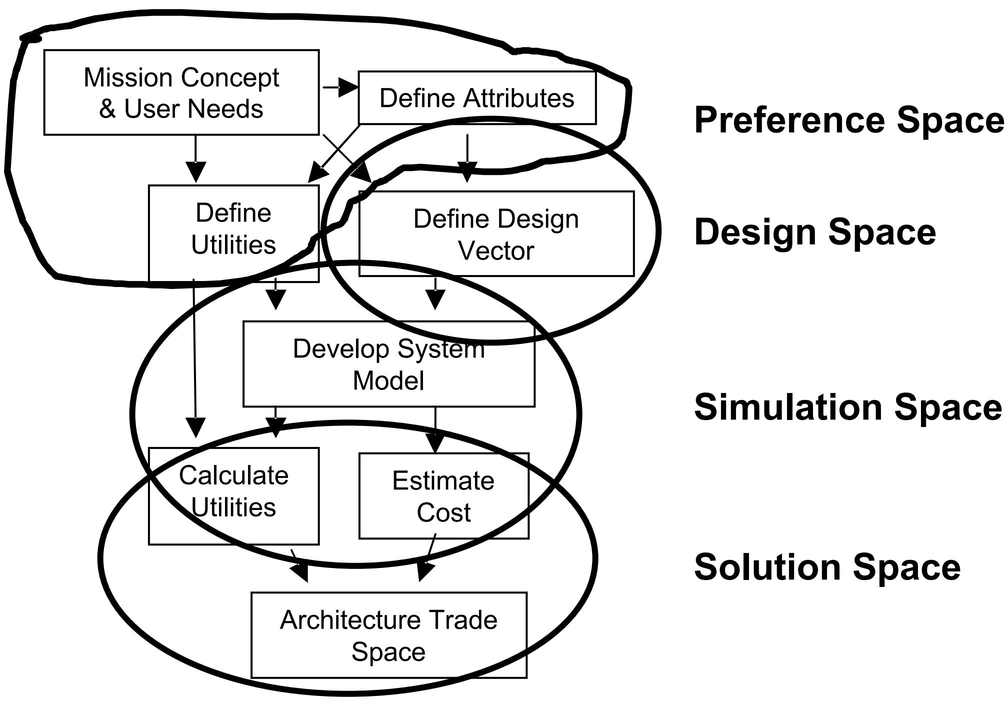


Figure 16 Design, preference, simulation and solution spaces

5.2. Defining Constraints[32]

A necessary step in defining the design and preference spaces is explicitly defining any hard constraints that may preclude choices of attributes or design variables. Constraints may be due to policy or political choices (for example, U.S government payloads must ride U.S. launch vehicles), funding constraints (maximum dollar or dollar-per-year amounts), hard customer preferences (consideration of only a subset of solutions that are in the customers interest to pursue), or other reasons.

It is important to make these constraints explicit in the development of the MATE model in order to understand their effects on the trade space. A will be seen later in this book, constraints can often have unintended negative consequences, driving up costs or precluding good solutions.

5.3. *Defining Attributes*[32]

What is an attribute?

Before continuing, the term "attribute" must be defined. An attribute is a decision maker-perceived metric that measures how well a decision maker-defined objective is met. Attributes have been described as "what the decision makers need to consider" and/or "what the user truly cares about." In practice, they must also be quantifiable, and capable of being predicted with reasonable fidelity by fairly high-level models. They can have natural or artificial units. All that matters is that the decision maker being assessed has a preference for different levels of that attribute in a well-defined context.

Attributes have a number of characteristics that must be explicitly determined through interactions with the decision maker. Attributes have a definition, units, range, and a direction of increasing value. All of these characteristics must be determined in order to properly design a system. The definition is incredibly important and must be determined *by the decision maker* to ensure that decision maker has a preference on the attribute. Units must be clarified in order to enable the Designer to accurately assess potential designs. The range is defined from the least-acceptable value (worst acceptable case) to the dream value (best case, above which delivers no additional value). Note that an attribute value at the least acceptable value is still acceptable. Lastly, when the range is defined, it also specifies the direction of increasing value—from worst to best case.

Table 1 contains examples of attributes drawn from case studies. The table shows the variety of possible attributes. The ATOS, BTOS, and XTOS systems (all designed to collect information on the earths ionosphere), and the Space Based Radar (from Spaulding[37]) have attributes that reflect scientists' or warfighters need for specific data collected. The attributes tend to be specialized and non-intuitive to the lay-person (and the designer!) and were determined with some difficulty. The Launch System[4] and Space Tug attributes are simple and relatively intuitive capabilities of the system, representing the customers' high-level interest in establishing national assets. The communication system attributes reflect the needs of the end-users of the communication network; they are standard network theory attributes, referred to in the original work as the GINA metrics. The generalize attributes from the INCOSE SE Handbook are interestingly similar.

Some of the examples in Table 1 were taken from work which did not use the MATE method. Instead, they specified *fixed* values of the attributes, which defined functional system requirements. This is generalizable – fixing the value of an attribute usually results in a functional requirement. This analogy should not, however, be taken too far; fixing all the attributes of a MATE study does not necessarily result in a complete set of functional requirements, nor can a good attribute set necessarily be developed simply by "floating" a set of functional requirements over a range of values.

Table 1: Examples of Attributes

ATOS:[9]	
	Equatorial Survey: presence of vehicle(s) in equatorial zone
	Equatorial Snapshot: complex function of relative vehicle positions to image ionosphere disturbances
	High Latitude Survey: Function of relative vehicle positions to map quasi-static ionosphere
BTOS:[43]	
	Mission Completeness: Combination of missions performed (AOA, EDP, turbulence).
	Spatial Resolution: Arc length of Earth between complete measurement sets.
	Revisit Time: Time between subsequent measurements of the same point above the Earth.
	Latency: Time delay from measurement to data reception by the end user.
	Accuracy: Measurement error in angle of arrival data from ground beacons.
	Instantaneous Global Coverage: percent of Earth's surface in view between subsequent measurements
XTOS:[44]	
	Data Life Span: Elapsed time between the first and last data points of the entire program
	Sample Altitude: Height above standard sea-level reference of a particular data sample
	Diversity of Latitudes Contained in Data Set
	Time Spent at the Equator: Time per day spent +/- 20 degrees off the equatorial
	Latency: The elapsed time between the collection of data and the start of transmission
Space Based Radar:[45]	
	Moving Object Tracking Area: Area in which a moving targets may be spotted
	Minimum Detectable Target Speed: Minimum speed for target to register as "moving"
	Image Resolution: Resolution of static imaging capability
	Image size: Area captured in static image
	Geo-location accuracy: Error ellipse of position information
	Gap Time: Time a target may go unobserved
	Center of Gravity Area
Launcher:[4]	
	Mass injected
	Injected speed (orbits attainable)
	Availability
Space Tug:[46]	
	Delta V change possible
	Capability of on-board equipment (grapplers, observation equipment, etc)
	Response Time
GINA Metrics:[38]	
	Signal Isolation: the ability to distinguish the desired signals from other information
	Information Rate: the rate at which information is generated or transmitted
	Information Integrity: the inverse of the error rate
	Information Availability: the probability that the generation/transmission will be successful
INCOSE General Attributes (detailed definition application dependent):[39]	
	Quantity
	Quality
	Coverage
	Timeliness
	Availability

The attributes will be used to determine the utility of the system to the user. In order to facilitate the use of formal utility theory (to be covered in following sections) the attributes should follow the following rules. According to Keeney and Raiffa,[40] a set of attributes must be complete, operational, decomposable, non-redundant, minimal, and perceived independent to ensure complete coverage of a decision maker's preferences (see Table 2). Operational means that the decision maker actually has preferences over the attributes. Decomposable means that they can be quantified. Non-redundant means none are double-counted. Minimal and complete are in tension, since Designer seeks to capture as many of the predominant decision metrics as possible, while keeping in mind the cognitive limitations in practice. (The human mind can typically only think about 7±2 objects simultaneously.[41]) The perceived-independent property is important for the utility independence axiom, described below, to hold. (The attributes need only be "perceived" independent; they do not need to actually be independent!) In practice, no set can be simply guaranteed to have all of these properties. The details of these restrictions and their consequences will be covered in greater depth in the section on utility theory.

Table 2 Characteristics of Attributes

Characteristics of attributes	Characteristics of a set of attributes	
Definition	• Complete	• Non-redundant
Units	• Operational	• Minimal
Range (worst→best)	• Decomposable	• Perceived-independent

<u>Determining Attributes</u>

The process of defining the attributes usually starts with preliminary interactions with possible users. Interviews, literature reviews, or other interactions with users, decision makers, and their work is necessary to understand the users needs, and imagine appropriate attributes. As part of this process, it is helpful if the interaction is two-way, so that the user or his or her representative understands the meaning and use of the attributes, and is ready for the utility interviews defined in the next section.

Ideally, attributes describe a function or output of a system. Thinking functionally is sometime difficult, especially for those with experience in traditional design methods. Functional thinking is key to defining concept-independent attributes that will not inherently bias later evaluations. That said, concept-independent attributes enable Designers more latitude in the design process, just as functional requirements enable more freedom than form requirements, but they are not absolutely required by the method.

Probe the needs that originated with the User and try to develop objective statements regarding these needs. The attributes will be quantifiable parameters that measure how well these objectives are met. A preliminary list of attributes allows the design team to begin to understand the modeling framework for the system.

Defining a set of attributes is a bit of an art. The examples in Table 1may provide a starting place for thinking about what the attributes may look like. Brainstorming with as many stakeholders as possible is a desirable first step. Using standard brainstorming technique, collect many possible attributes, and try to group similar ones together and eliminate weak ones. Attributes which cannot be quantified, or for which ranges cannot be established, are not useful. Attributes which have only one acceptable value are really constraints. Attributes which describe physical characteristics of the system rather than its functions or outcomes may be design variables (next section) or they may simply be misguided. Thinking in terms of "decision metrics" is valuable. An important question to ask the decision maker: "when deciding on a particular design, what are the characteristics that you would consider?" Those characteristics are often good attributes. Another method to define attributes is through a hierarchy of objectives. (See Keeney and Raiffa[40] and Smith, Levin et al.[42] for example frameworks.) <this needs to be hooked in concrete ways to the architecture/design framework?>.

In all cases, remember that the attributes should be appropriate to the level of analysis being carried out. Although there is no absolute limit to the number of attributes that can be handled, experience suggests that three to seven attributes is appropriate. Architecture studies should be concerned with a few of the highest-level functions of the system or systems. If brainstorming produces too many attributes, it is likely that the group is thinking at too detailed a level.*

Finalizing Attribute Definitions

The attributes will need to be iterated with stakeholders. They will be reevaluated in light of additional information that will emerge when the design vector is chosen and the attributes and design vector elements are correlated (see the next section). They also may need to be reevaluated or redefined as part of the process of formally quantifying them with multi-attribute utility theory. Finally, the results of the tradespace evaluation may require new attributes to be examined, and/or call into question the original choices. These iterations on the attribute definitions require progressively more work, so it is desirable to do the best job possible at each step.

Ideally, the attributes would be developed in full cooperation with the user. More typically, after the first interaction, the design team works on the attributes and returns to the users with a preliminary list. The user must critically assess if the proposed attributes accurately capture his or her needs. The team must also insure that the conditions in Table 2are met. The decision maker is also asked to provide or confirm a range for each attribute corresponding to the best case and the worst case. The best case is the best value for the attribute from which the user can benefit; a better level will not give more value. The worst case corresponds to the attribute value for which any further decrease in performance will make the attribute useless. These ranges define the domain where the single attribute preferences are defined. The attributes have to describe decision maker needs accurately in order to meaningfully assist the trade study. Iteration will almost certainly be required to find the right attribute set.

The final result is a finalized and mutually agreed upon list of user attributes including their definitions, ranges, units, and direction of increasing value.

---

* Nested utility functions are possible to capture more than six attributes, however nesting adds complication and requires a sophisticated MATE engineer.

A note on cost

Cost may be thought of as an attribute, but in the examples given here it is treated somewhat differently. It is clearly quantifiable, and has an obvious direction of preference (lower is better). However, defining upper and lower bounds on cost during concept exploration will be arbitrary and may be excessively restrictive. Financial resources tend to be controlled by different stakeholders than the technical attributes. In government systems, the user community may set the technical attributes, but the available funds will be controlled by the acquisition agencies and congress. Calculation of cost is independent of the calculation of technical attributes, usually using very different types of models. The cost estimates may be of considerably lower fidelity than the technical simulations, especially when new concepts are being considered. Finally, cost is a useful independent criterion against which to weigh the advantages and disadvantages of various levels of technical performance. For all these reasons, cost will be treated separately from the other attributes.

This does not mean that finding levels of funding which the customer is interested in providing is a bad idea. If the customer is very determined to keep costs below a certain level it can be included as a constraint in the trade space, and in any case it can be kept in mind when exploring the tradespace.

The expenditure of resources other than money may need to be included in the tradespace as well. If the system places extraordinary demands on a limited asset (e.g. the communications bandwidth of the TDRS system) then this resource burn should be included as either a component of cost or as an attribute. In the former case it needs to be converted into dollars. In the latter, it needs to be restated as a desirable characteristic, e.g. efficient transmission of information.

5.4. X-TOS Attributes[32]

The X-TOS attributes were determined by the needs of the science users. They needed a data set consisting of measurements from the predetermined instrument package (this was a constraint) collected over a period of time, at varying altitudes and latitudes, and transmitted with some latency to the ground. The scientists cared about all of these aspects of the data. This interest was quantified by brainstorming with the user a preliminary set of attributes:

Knowledge/Accuracy Altitude	(km)
Mission Lifetime	(months)
Time spent in region	(min)
Latitude range	(degrees)
Latency	(min)
# Simultaneous data pts	(integer)
Data Completeness	(%)
Pointing Accuracy	(degrees)
Pointing Control	(degrees)

These preliminary attributes suffered several of the weaknesses mentioned above. Some are not discriminating, e.g. the pointing accuracy was easily within any reasonable vehicle's capability. Others where not well posed for quantifying, e.g. time spent in various regions. Others were not functional, e.g. mission lifetime described the lifetime of a physical vehicle; the scientists were interested in the time from the collection of the first data point to the last, which could be collected by more than one vehicle or even system, hence "Data life span" below. The attributes were ultimately reduced to the following set, and upper and lower bounds set:

Table 3 X-TOS Attributes

Attribute	Units	Best	Worst
1) Data Life Span	(years)	11	0.5
2) Sample Altitude	(km)	150	1000
3) Diversity of Latitudes in Data Set	(degrees)	180	0
4) Time Spent in Equatorial Region	(hours/day)	24	0
5) Latency			
Scientific Mission	(hours)	1	120
Tech Demo Mission	(hours)	0.5	6

Data Life Span: Elapsed time between the first and last data points of the entire program, measured in years.
Sample Altitude: Height above standard sea-level reference of a particular data sample, measured in kilometers. (Data sample = a single measurement of all 3 instruments)
Diversity of Latitudes Contained in Data Set: The maximum absolute *change* in latitude contained in the data set, measured in degrees. The data set is defined as data taken between 150 – 1000 km.
Time Spent at the Equator: Time per day spent in the equatorial region defined as +/- 20 degrees off the equatorial. Measure in hours per day.
Latency: The maximum elapsed time between the collection of data and the start of transmission downlink to the communication network, measured in hours. This attribute does not incorporate delays to use.
Scientific Mission – Latency max and min for the AFRL model
Tech Demo Mission – Latency max and min for demonstration of now-casting capability.

There are some complications even in the final set. The sample altitude is a vector of values (one per data sample!) that must be reduced to be evaluated, and the latency has two different definitions for two potential stakeholders with incompatable needs. These difficulties will be addressed in the utility section to follow.

5.5. *Space Tug Attributes*

The space tug attributes were defined at a very high level. They reflect several decisions made at the front end of the trade study. The Space Tug is a hypothetical capability. Only the vehicle was considered in the first stage of the study. Key vehicle systems (such as grappling mechanisms) and operational details (such as software for rendezvous) were studied separately from the trade studies. The capabilities of a space tug vehicle determined to be useful to a potential user included:

Table 4 Space Tug Attributes

Attribute	Units	Best	Worst
1) Delta V capability	km/sec	40	>0
2) Equipment carrying capability	kg	5000	300
3) Response time	-	fast	slow

Delta-V capability: determines where the space-tug can go and how far it can change the orbits of target vehicles

Equipment carrying capability: mass of observation and manipulation equipment (and possibly spare parts, etc.) carried, which determines at a high level what it can do to interact with targets

Response time: how fast it can get to a potential target and interact with it in the desired way. This was initially considered only in a binary sense of fast (hours to days) or slow (weeks to months).

These were confirmed with the customer, but not initially iterated with him. At the conclusion of the first phase of the study, the customer expressed a desire to include launch systems and some operational details (storage and parking modes and locations) in the trade study. This required a rethinking of the attribute list, although it proved to be relatively minor. The updated attribute list included a quantified response time, in hours, from 1 (best) to 2160, or three months (worst).

5.6. *Defining the Design Space*[32]

What is a design space?

Once the attributes have been determined, the designers need to develop concepts to perform the mission, which are reflected in the construction of a design vector. The design vector focuses on those variables that have been identified to have significant impact on the specified attributes. A tension will exist between including more variables to explore a larger tradespace and the computational difficulty for actively exploring such a large space. Geometric growth of the trades pace results with increasing number of variables and the values over which they are enumerated. Computational considerations motivate keeping the list curtailed to only the key elements, while still maintaining the ability to keep the trade space as open as possible in order to explore a wide variety of architectures.

As a good general practice, a constants vector is also defined. This vector includes many potential design variables which are, for a variety of reasons, fixed for all analyses. These could be design variables that are assumed to be weak impact, variables reflecting the current economic and technical situation (that could conceivably change in the future but are not expected to impact the design), or any other variable that is not selected for the design vector. The constants vector might also include physical constants, constraints, and scoping assumptions for the model. Placing them in a constants vector allows them to be parametrically varied to assure that the models are not sensitive to them, varied to perform what-if scenarios, or converted to design variables quickly and easily.

Table 5 shows some example design vectors used in MATE analyses. The ATOS and BTOS vectors concentrate on the arrangements of swarms of small vehicles configured to collect data and maximize their respective attributes. Note that the ATOS vector contains no design vector elements concerning the design of the vehicles themselves; the performance of the swarms are only weakly dependent on the performance of the individual vehicles, so a nominal high-level vehicle design is placed in the constants vector. BTOS has only the highest level of vehicle concerns in the design vector: the configuration study relates to some high level options for which instruments and capabilities go on which vehicles. The XTOS design vector contains both orbital elements and high-level vehicle choices; the mission scenarios include the possibility of more than one vehicle. The space based radar design vector is a similar mix of vehicle and orbit variables, with constellation type including numbers of vehicles and orbit types. Finally, the Space Tug design vector is a high level description of the vehicle.

Table 5 Sample Design Vectors

ATOS: [9] Swarm perigee Swarm apogee # sats/swarm # subplanes/swarm # suborbits/subplane Yaw angle of subplanes Max sat separation (swarm diameter)
BTOS: [43] Circular orbit altitude (km) Number of Planes Number of Swarms/Plane Number of Satellites/Swarm Radius of Swarm (km) 5 Configuration Studies
XTOS: [44] Altitude of Apogee (km) Altitude of Perigee (km) Inclination (deg) Total Delta-V (m/s) Comm. Sys Type Antenna Gain Propulsion Type Power Sys Type Mission Scenario
Space Based Radar: [45] Scan Angle Technology Level Aperture Area Orbit Altitude Constellation type
Space Tug: [46] Mass of on-board equipment (grapplers, observation equipment, etc) Propulsion system Fuel load

Choosing a design vector

A set of variables that spans the desired space of possible solutions is proposed, usually by a brainstorming processes. The first list should be inclusive–the desire at this stage is to create a list from which the actual design vector will be reduced. Typically, design vector variables are descriptions of the *form* of the solution. For space vehicles, this might include vehicle types, subsystem choices, fuel loads, technologies used. For space systems, this might include orbits, operating and communications modes, ground and launch systems used, etc.

Like the attributes, choosing the design vector is something of an art. In general, however, it is more straightforward, as the design vector represents the physical characteristics of the system, which are easier to imagine and discuss than functional characteristics. Generally, more design vector elements can be used than attributes, although too many will make the simulations computationally intractable. The key to limiting the design vector is again selecting the right level of analysis.

The usual brainstorming process may produce many possible design vector elements, which must be reduced to a computationally tractable set. This is a process of reduction that can be carried out by a number of means. An effective technique is to map the proposed design variables against the attributes, and use educated guesswork or back-of-the-envelope modeling to estimate the likely impact of the design variables on the attributes. Design variables which have a strong impact on the attributes (and hence will have an strong impact on the user utilities) and which are actually under the control of the designer (and hence can be varied significantly) are desired. Eliminated variables can be left in the constants vector; later in the process, sensitivity analyses can be performed to validate the assumption that they only weakly impact the attributes.

Finalization of the design variables is necessary before code development can begin. Proposed design variables become finalized after the attributes have been finalized and an understanding of the dependencies between the design variables and attributes has been understood. In addition to the identity of the design variables, the values to be used also need to be picked at this point. Continuous variables (e.g. fuel load) need to be checked at a number of fixed values, which must be chosen, while discrete ones (e.g. mission scenario) need to be fully defined and quantified.

Updating the design vector

Experience has shown that the design vector is the least stable element in the trades space. As the modeling, and even the analysis, progress, design variables may prove irrelevant or non-discriminating. As often, sensitivity studies or changes in user preferences elevate variables consigned to the constants vector to design vector status. The enumeration of the design vector will almost always change somewhat, as sensitive regions of the trade space are identified that require more detailed looks. The model architecture should reflect this by being as modular as possible, and by including as many variables as practical in a constants vector rather than "hardwiring" the values into code.

5.7. X-TOS Design Vector[35]

The definition of the design vector begins with the consideration of user specified attributes (see Chapter 3). Since these attributes define user utility, and the objective of the designer is to maximize that utility it follows that the designer would choose a set of design variables that have a high degree of leverage in changing the values of these attributes. In the case of the X-TOS attributes, two key groups of variables emerged: the orbit(s) in which data would be taken, and the spacecraft(s) taking that data.

In the case of X-TOS orbits, three parameters were chosen: the altitude of apogee, the altitude of perigee and the orbital inclination. Of course these three parameters are not sufficient to fully specify a keplerian orbit; rather a total six orbital elements are needed. The remaining elements are not included in the design vector since they either do not provide leverage in changing utility or there is an obvious utility maximizing choice. For example, since only the latitude and altitude (not the longitude) of a particular data point is of interest to the user, the right ascension of the ascending node is not included. On the other hand, since the altitude and latitude range attributes are taken independently (i.e. the user is not expressing preferences for combinations of altitudes and latitudes) one would immediately choose the argument of perigee to align the line of nodes with line of apsides. Such a selection maximizes the time in the equatorial region without affecting the other attributes. These remaining elements are included in the constants vector.

Unlike the orbits, appropriate design variables used to describe the spacecraft are not readily apparent from the attributes. In general, the computational and modeling resources available will tend to reduce the scope of possible architectures. The X-TOS team decided to eliminate concepts such as tethers since sufficiently fast and accurate models of their behavior were not in hand and could not be constructed in the time allotted. After reducing the scope of possible satellites, to relatively small traditional designs using off the shelf technologies, key sub-system level trades were identified.

The final step in defining the design vector is to choose at which discrete levels to sample the continuous design variables. The designer needs to choose a sufficient diversity of levels to ensure coverage of the tradespace, yet balance that choice with the additional computational expense of more levels. Often the number of combinations of design variables grows geometrically with the number of levels per design variable. The key is to use the attributes and utility functions to help define interesting areas of the trade space. For example in X-TOS, the levels orbital parameters were chosen to ensure breadth in inclination and a preference for low altitudes. There is some degree of art to this choice since one does not want to eliminate high utility areas of the tradespace. In X-TOS, the Total Delta-V design variable was capped at 1000 m/s (a cap that was thought to be conservative). During the detailed design phase (MATE-CON) it was discovered that values of in excess of 1200 m/s. were tenable.

Table 6 X-TOS Design Vector

Design Variable	Levels	Justification
Altitude of Apogee (km)	200:50:350; 650:300:2000*	Emphasis on low altitude in utility function, therefore sample at a higher rate at low altitudes
Altitude of Perigee (km)	150:50:350*	Utility curve declines quite steeply between 150 and 350 km; will take a significant utility hit if spacecraft never flies below 350
Inclination (deg)	0; 30; 70; 90	Covers the possible range of inclinations
Total Delta-V (m/s)	200:100:1000*	The low end of the range is a high average value for low earth orbit satellites. The high end is an estimate of the optimistic (on the large side) estimate delta V allowed before the spacecraft mass will no longer accommodate small and medium sized US launch vehicles.
Comm. Sys Type	AFSCN; TDRSS	Discrete choice of systems available
Antenna Gain	High; Low	Discrete choice of systems available
Propulsion Type	Chemical; Hall	high-thrust at low efficiency vs. low-thrust at high efficiency
Power Sys Type	Solar; Fuel cells	Only body mounted solar considered due to prohibitive drag penalty of wings
Mission Scenario	Single; 2 Series; 2 Parallel	More than two satellites is computationally prohibitive since the number of possible multi-spacecraft mission grows as N^k where k is number of spacecraft in the mission scenario and N is number of combinations of the other (spacecraft and orbit related) design variables.

*The notation *low* : *inc* : *high* means from *low* to *high* in steps of *inc*.

5.8. Space Tug Design Vector[28]

The space tug design vector was also defined at a very high level. Per the assumptions made at the beginning of the study, only the characteristics of the vehicle were considered. A very wide tradespace was considered, so only the design choices likely to have first order effect on the attributes were considered. The final design vector was:

Table 7 Space Tug Design Vector

Design Variable	Units	Levels
1) Mass of on-board equipment	kg	300;1000;3000;5000
2) Propulsion type	-	Storable bi-prop; cryogenic; electric; nuclear
3) Fuel or reaction mass	kg	30;100;300;600;1200;3000;10000;30000;50000

Many other potential design variables, with weaker or less discriminating effects on the attributes, were placed in the constants vector. These included: bus systems (structure, thermal, non-propulsion power, and control and communications systems), which were reduced to a rule-of-thumb mass; the details of the propulsion system (I_{sp}, mass, and power), which varied between the various types of propulsion but were fixed for each type; and development and launch costs, which were build on rules of thumb. All of these were set to reasonable nominal values.

At the conclusion of the first phase of the study, the customer expressed a desire to include launch systems and some operational details (storage and parking modes and locations) in the trade study. This required a rethinking of the design vector, to include additional variables such as storage modes (ground vs. orbit), parking orbits, and launch options.

5.9. Preparation for modeling: Final attribute–design vector mapping

In order to structure the modeling stage of MATE, covered in the next sections, it is necessary to fully understand the anticipated relationship between the design variables and the attributes. Notional mappings of design variables to attributes allow for the conception of necessary modules for the model. This activity is done in parallel with the proposal and finalization of design variables since it helps prioritize design variables and pare down the proposed list.

A technique similar to Quality Function Deployment (QFD, also referred to as the House of Quality) is used to relate the attributes and design variables. It is very important to note that the purpose of the mapping is to anticipate how each design variable will impact the attributes, NOT to find the values of the design variables (and thus specify the design) as is the case when the method is used in its traditional way.

As shown in the examples, Attributes are listed on the rows, and Design variables are listed on the columns. Note cost is included as a special row, befitting its role as a special attribute. The degree of expected impact (how much the design variable is expected to affect the attribute) is rated as first order (9), second order (6), small (3) or zero (0). These rankings can be obtained by educated guesswork, back-of-the-envelope calculations, experience, or expert opinions. They are intended to help the process rather than provide solutions, so best-effort work here is expected and acceptable.

The central matrix gives a visual summary of the complexity of the calculations that will be necessary to compute the attributes given the design vector. A heavily-populated matrix tends to indicated a complex and highly coupled system; a sparsely-populated matrix a less complex one. Clumps of strong interactions (note the rows and columns may be rearranged at will to achieve this clumping) may indicate coupled physics and may suggest computational modules.

The rows and columns are summed. The sums on the attributes show how strongly they are affected by the design variables; a very low sum indicates the attribute is not sufficiently affected by the design vector and either the attribute is inappropriate or the design vector should be modified to more strongly affect it. The sums on the design variables indicate their impact on the attributes; again, a low number is a flag that either the design variable is non-discriminating, or that an attribute is missing. The latter can happen when the team's physical intuition for the problem (understanding that a design variable *should* affect the outcome) exceeds their functional intuition (understanding what function of the outcome would vary with the design variable).

As this step is informational, the process should be modified to suit the problem. First order effects can be accentuated by using a 9-3-1-0 scoring system instead of 9-6-3-0; this is useful for larger attribute-design vector sets where first order effects must be emphasized. If an interaction is truly unknown, (but suspected to be non-zero) a special notation can be made on the chart to indicate further study is required. The choice of attributes on rows and design variables on columns is fairly arbitrary (previous MATE studies have used the opposite convention to the one used here). The present convention is suggested so that a "house of quality" (see Reference 39, Appendix A) can be built over the design variables to study their interactions with each other.

5.10. X-TOS attribute–design vector mapping

Attributes / Design Vars	Perigee	Apogee	Delta-V	Propulsion	Inclination	Comm System	Ant. Gain	Power system	Mission Scenario	Total Impact
Data Lifespan	9	9	9	6	0	0	0	6	9	48
Sample Altitude	9	9	0	0	0	0	0	0	9	27
Diversity of Latitudes	0	0	0	0	9	0	0	0	9	18
Time at Equator	0	6	0	0	9	0	0	0	9	24
Latency	3	3	0	0	3	9	9	6	3	36
Total	21	27	9	6	21	9	9	12	39	
Cost	9	9	3	6	6	3	6	6	9	
Total w/Cost	30	36	12	12	27	12	15	18	48	

Figure 17 X-TOS attribute–design vector mapping

The chart above highlights the important dependencies. Data lifespan and sample altitude are determined primarily by the orbital mechanics and the delta-V capability necessary to maintain orbits. Diversity of latitudes and time at the equator are determined primarily by orbital inclination. Latency is primarily a function of the communication system. Mission scenario (how many vehicles are launched, when, into what orbits) affects most attributes very strongly.

The totals indicate that data lifespan is impacted by many of the design variables and may be the most discriminating of the attributes. (In hindsight, this proved to be the case). Diversity of latitudes, on the other hand, is impacted strongly by only two variables, and so will be easy to compute. It may well still be discriminating, as it has two strong interactions. The design variable totals show the orbital elements and mission scenario having more effect than the vehicle design parameters. Propulsion and power systems, in particular, look like they may have only weak effects. Intuition suggests that the propulsion system choice should have a stronger effect. In hindsight, its effect was diminished by the choice of delta-V as a design variable, instead of a more physical parameter such as fuel load. The power system affects data lifespan through its own lifespan, and latency via its ability to provide sufficient power; this suggests that power system modeling could be made very simple, concentrating only on these two aspects.

5.11. Space Tug attribute–design vector mapping

Attributes \ Design Vars	Propulsion System	Fuel Load	Equipment Mass	Total Impact
Delta-V	9	9	9	27
Response Time	9	1	1	11
Equipment Capability	0	0	9	9
Total	18	10	19	
Cost	9	9	9	
Total w/Cost	27	19	28	

Figure 18 Space tug attribute–design vector mapping

This rather simple map organizes the known interactions. Equipment capability is uniquely determined by equipment mass. Response time is primarily determined by the choice of propulsion system, with relatively weak interactions with the other design variables that were ultimately ignored. The delta-V calculation will be the most difficult, depending on all of the design variables.

NOTES AND REFERENCES

[1] Diller, N. P., “Utilizing Multiple Attribute Tradespace Exploration with Concurrent Design for Creating Aerospace Systems Requirements,” Master of Science Thesis in Aeronautics and Astronautics, Massachusetts Institute of Technology, June 2002.

[2] Ross, A. M., “Multi-Attribute Tradespace Exploration with Concurrent Design as a Value-Centric Framework for Space System Architecture and Design,” Master of Science Thesis in Aeronautics and Astronautics, Massachusetts Institute of Technology, June 2003.

[3] W. J. Fabrycky, Life Cycle Cost and Economic Analysis, Prentice-Hall, NJ, 1991, and W. J. Fabrycky, Engineering Economy, Pearson Education, Inc., NJ, 1991.

[4] NASA Systems Engineering Handbook, National Aeronautics and Space Administration SP-610S, June 1995.

[5] Murman, E., Allen, T., Bozdogan, K., Cutcher-Gershenfeld, J., McManus, H., Nightingale, E., Rebentisch, E., Shields, T., Stahl, F., Walton, M., Warmkessel, J., Weiss, S., and Widnall, S., *Lean Enterprise Value*, Palgrave, London, 2002.

[6] Ross, A. M., Diller, N. P., Hastings, D. E., and Warmkessel, J. M., “Multi-Attribute Tradespace Exploration as a Front-End for Effective Space System Design,” *Journal of Spacecraft and Rockets*, January 2004.

[7] Diller, N. P., "Utilizing Multiple Attribute Tradespace Exploration with Concurrent Design for Creating Aerospace Systems Requirements," SM, Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA, 2002.

[8] This section is modeled on, and has edited text from, Hugh L. McManus, Daniel E. Hastings, and Joyce M. Warmkessel, “New Methods for Rapid Architecture Selection and Conceptual Design,” *Journal of Spacecraft and Rockets.* < Vol and Number TBD)>

[9] McManus, H. L., and Warmkessel, J. M., “Creating Advanced Architectures for Space Systems: Emergent Lessons from New Processes,” accepted for publication in the *Journal of Spacecraft and Rockets,* < Vol and Number TBD)>

[10] Larson, W. J., and Wertz, J. R., *Space Mission Analysis and Design,* 3rd ed., Microcosm Press, El Segundo, CA., 1999.

[11] Mosher, T. J., Lao, N. Y., Davalos, E. T., and Bearden, D. A., “A Comparison of NEAR Actual Spacecraft Costs with Three Parametric Cost Models,” *Acta Astronautica*, Vol. 45, No. 4-9, Aug.-Nov. 1999, pp. 457-464.

[12] Saleh, J. H., Lamassoure, E., and Hastings, D. E. “Space Systems Flexibility Provided by On-Obit Servicing: Part l,” *Journal of Spacecraft and Rockets*, Vol 39. No. 4, July-Aug. 2002.

[13] Daniel E. Hastings, Annalisa L. Weigel, Myles A. Walton, “Incorporating uncertainty into conceptual design of space system architectures,” MIT Engineering Systems Division & Department of Aeronautics and Astronautic, unpublished.

[14] Portions of the text in this section are edited from Hugh L. McManus, Daniel E. Hastings, and Joyce M. Warmkessel, “New Methods for Rapid Architecture Selection and Conceptual Design,” accepted for publication in the *Journal of Spacecraft and Rockets.* < Vol and Number TBD)>

[15] Paragraph taken verbatim from McManus, H. L. and Schuman, T. E., “Understanding the Orbital Transfer Vehicle Trade Space,” AIAA Paper 2003-6370, Sept. 2003.

[16] Figures from McManus, H. L. and Schuman, T. E., “Understanding the Orbital Transfer Vehicle Trade Space,” AIAA Paper 2003-6370, Sept. 2003.

[17] Aguilar, J. A., and Dawdy, A., “Scope vs. Detail: The Teams of the Concept Design Center,” 2000 IEEE Aerospace Conference Proceedings, Big Sky, Montana, March 2000, Vol. 1, pp. 465-482.

[18] Unpublished figure from SPACETUG project final presentation. May have been used in other SpaceTug paper <check>.

[19] Boyars, A., Kusmik, W. A., and Yukish, M., “Collaborative Engineering Across Organizational Boundaries,” *Naval Engineers Journal*, Vol. 114, No. 3, Summer 2002, pp. 75-77.

[20] Budianto, I. A., and Olds, J. R., “A Collaborative Optimization Approach to Design and Deployment of a Space-Based Infrared System Constellation,” 2000 IEEE Aerospace Conference Proceedings, Big Sky, Montana, March 2000, Vol. 1, pp. 385-393.

[21] Stump, G., Simpson, T. W., Yukish, M., and Bennett, L., “Multidimensional Visualization and its Application to a Design-by-Shopping Paradigm,” AIAA Paper 2002-5622, 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Atlanta, GA, Sept. 2002.

[22] Smith, J. L., “Concurrent Engineering in the JPL Project Design Center,” Society of Automotive Engineers, Inc., Paper 98AMTC-83, 1998.

[23] Oxnevad, K. I., “The NPDT – The Next generation Concurrent Design Approach,” EUSEC 2000, 2nd European Systems Engineering Conference, Munich, Germany, Sept. 2000.

[24] Oxnevad, K. I., “A Concurrent Design Environment for Designing Space Instruments,” Proceedings of the 9th Thermal and Fluids Analysis Workshop, Ohio Aerospace Institute and NASA Lewis Research Center, Cleveland, Ohio, Sept. 1998.

[25] Smith, P. L., Dawdy, A. D., Trafton, T. W., Novak, R. G., and Presley, S. P., "Concurrent Design at Aerospace." *Crosslink*, Vol. 2, No. 1, 2001, pp. 4-11.

[26] Wirthlin, J. R., “Best Practices in User Needs/Requirements Generation,” Master of Science Thesis in System Design and Management, Massachusetts Institute of Technology, January 2000.

[27] Saleh, J. H., and Hastings, D. E., “On Flexibility in Design - Analyzing Flexibility of Space Systems,” AIAA Paper 2000-5098, AIAA Space 2000 Conference and Exposition, Long Beach, CA, Sept. 2000.

[28] Some text modified from McManus, H. L. and Schuman, T. E., “Understanding the Orbital Transfer Vehicle Trade Space,” AIAA Paper 2003-6370, Sept. 2003.

[29] Galabonva, K., Bounova, G., de Weck, O. and Hastings, D., “Architecting a Family of Space Tugs Based on Orbital transfer Mission Scenarios,” AIAA paper 2003-6368.

[30] Ross, A. M. and Diller, N. P., Multi-Attribute Trade Space with Concurrent Design MATE-CON: The MATE Short Book, unpublished.

[31] A. Ross, N. Diller, and D. Hastings, Multi-Attribute Tradespace Exploration with Concurrent Design for Space System Conceptual Design, AIAA 2003-1328, 41st Aerospace Sciences Meeting and Exhibit, 6 – 9 January 2003, Reno, Nevada

[32] This section includes edited material from Ross, A. M., “Multi-Attribute Tradespace Exploration with Concurrent Design as a Value-Centric Framework for Space System Architecture and Design,” Master of Science Thesis in Aeronautics and Astronautics, Massachusetts Institute of Technology, June 2003, and Diller, N. P., “Utilizing Multiple Attribute Tradespace Exploration with Concurrent Design for Creating Aerospace Systems Requirements,” Master of Science Thesis in Aeronautics and Astronautics, Massachusetts Institute of Technology, June 2002.

[33] Ulrich, K. T., and Eppingeer, S. D., *Product Design and Development*, Erwin McGraw-Hill, Boston, 2000, pp 20-23.

[34] Martin, J. N., *Systems Engineering Guidebook- A Process for Developing Systems and Products*, CRC Press, New York, 1997

[35] This section includes edited material from “X-TOS: 16.89 Final Design Report,” MIT Department of Aeronautics and Astronautics, May 2002.

[36] This section includes edited material from McManus, H. L., and Warmkessel, J. M., “Creating Advanced Architectures for Space Systems: Emergent Lessons from New Processes,” accepted for publication in the *Journal of Spacecraft and Rockets,* < Vol and Number TBD)>

[37] Spaulding, Timothy J., “Tools for Evolutionary Acquisition: A Study of Multi-Attribute Tradespace Exploration (MATE) applied to the Space Based Radar (SBR),” Master’s Thesis in Aeronautics and Astronautics, Massachusetts Institute of Technology, June 2003.

[38] Shaw, G. M., Miller, D. W., and Hastings, D. E., “Development of the Quantitative Generalized Information Network Analysis (GINA) Methodology for Satellite Systems,” *Journal of Spacecraft and Rocket*s, Vol. 38, No. 2, 2001, pp. 257-269.

[39] *INCOSE Systems Engineering Handbook: A "How To" Guide for All Engineers*, 2nd ed., International Council on Systems Engineering, Seattle, WA, 2002.

[40] Keeney, R. L. and Raiffa, H. *Decisions with Multiple Objectives--Preferences and Value Tradeoffs*. 2nd ed. Cambridge: Cambridge University Press, 1993. Chapters 1 and 4.

---

[41] Miller, G. A. (1956). "The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information." The Psychological Review **63**: 81-97.

[42] Smith, J. H., Levin, R. R., et al. (1990). An Application of Multiattribute Decision Analysis to the Space Station Freedom Program-Case Study: Automation and Robotics Technology Evaluation. Pasadena, CA, Jet Propulsion Laboratory: 190.

[43] Diller, N. P., and Warmkessel, J. M., "Applying Multi-Attribute Utility Analysis to Architecture Research for the Terrestrial Observer Swarm," Proceedings of the 20th Digital Avionics Systems Conference, Daytona Beach, FL, Oct. 2001, Vol. 1, pp. 4B6/1-4B6/9.

[44] Ross, A. M., Diller, N. P., Hastings, D. E., and Warmkessel, J. M., "Multi-Attribute Tradespace Exploration as a Front-End for Effective Space System Design," *Journal of Spacecraft and Rockets*, January 2004.

[45] Spaulding, Timothy J., "Tools for Evolutionary Acquisition: A Study of Multi-Attribute Tradespace Exploration (MATE) applied to the Space Based Radar (SBR)," Master's Thesis in Aeronautics and Astronautics, Massachusetts Institute of Technology, June 2003.

[46] McManus, H. L. and Schuman, T. E., "Understanding the Orbital Transfer Vehicle Trade Space," AIAA Paper 2003-6370, Sept. 2003.