Fifteenth Annual International Symposium of the
International Council On Systems Engineering (INCOSE)
10 July to 15 July 2005, Rochester, NY

A Framework for Understanding Uncertainty and its Mitigation and Exploitation in Complex Systems

Dr. Hugh McManus
Metis Design, 222 Third St.
Cambridge MA 02142
hmcmanus@metisdesign.com

Prof. Daniel Hastings
MIT E40-257 77 Massachusetts Ave.
Cambridge MA 02140
hastings@mit.edu

Abstract. A framework to aid in the understanding of uncertainties and techniques for mitigating and even taking positive advantage of them is presented. The framework is an attempt to clarify the wide range of uncertainties that affect complex systems, the risks (and opportunities) they create, the strategies system architects can use to mitigate (or take advantage) of them, and the resulting system attributes. Current and developing methods for dealing with uncertainties are projected onto the framework to understand their relative roles and interactions.

Introduction

Many types of uncertainty affect the design and operation of complex systems. Mature techniques exist for some classes of uncertainties, e.g. rolling up component reliabilities to calculate system reliability, and mitigating problems with redundancy. These techniques have recently been pushed forward into preliminary and even conceptual design trades.[1] Techniques are emerging for many other classes of uncertainty, e.g. budget and policy instability[2,3] and the effects of non-collocated teams during design.[4] Uncertainty is not always a negative to be mitigated; robust, versatile and flexible systems not only mitigate uncertainties, they can also create additional value for users.

The current environment of rapidly changing technologies and markets on the commercial side, and rapidly changing technologies, threats, needs, and budgets on the defense side, has created a need for better understanding of these classes of uncertainties and their effects on complex airspace systems. This problem is recognized at a national level, and "robust", "flexible", or "evolutionary" systems and designs have been called for. Unfortunately, tools for handling these classes of uncertainties are immature, and methods for flexible or evolutionary designs are in their infancy. This represents both a need and an opportunity for growth in the systems engineering community.[5]

The wide range of types of uncertainties and possible responses to them make unified discussions of the problem difficult. In particular, discussion of desired advanced system characteristics such as robustness, flexibility, and adaptability is plagued by poorly defined terminology. This difficulty is particularly acute when teaching both the basic problems and the emerging techniques to students of complex system design. As an aid to discussion and teaching, a framework is presented.

The framework is intended to clarify, and structure the discussion of, the problem of handling uncertainties in complex engineering systems. Definitions for terminology are included which are clearer in the context of the framework than in isolation. Existing methodologies are mapped onto the framework, making their relative roles and their relationships more explicit. The overlap (or lack thereof) of the available methods can be seen, as can areas of future need and opportunity. The framework is also used in graduate engineering education, as a mechanism for unifying a variety of material on these problems in a new class in Space Systems Architecture given in the fall of 2004.

Framework

The global problem of dealing with uncertainty is first broken into four categories, which are conceptually very different. Simplistically, **Uncertainties** lead to **Risks or Opportunities,** which are handled technically by **Mitigations or Exploitations,** which hopefully lead to desired **Outcomes**.

- **Uncertainties** are things that are not known, or known only imprecisely. They may be characteristics of the universe (e.g. statistical processes) or characteristics of the design process (e.g. information not yet collected); in either case they are factual. Many uncertainties are measurable, although some are not (e.g. future events). They are value neutral; they are not necessarily bad. Their causes are numerous and not addressed here.
- **Risks** are pathologies created by the uncertainties that are specific to the program in question. They are often quantified as (probability of uncertain event)*(severity of consequences). In addition to technical failure, cost, schedule, political, market, and user need shift risks need to be considered. Risk has a negative connotation, but uncertainty may also create **opportunity**, which we put in the same category.
- **Mitigations** are technical approaches to risk minimization; we use the word **exploitations** for technical approaches to value or opportunity enhancement. They are not necessarily good things in and of themselves; on the contrary they are often expensive and must be justified by their effects on outcomes.
- **Outcomes** are attributes of the system that a user may find valuable, specifically those which quantify or at least characterize its interaction with uncertainties.

Taxonomies

There are many causes of uncertainty, many types of risk, many approaches to mitigation. Therefore, under each category is a decomposition or taxonomy. The elements are not specific uncertainties, risks, or approaches, but rather broad types or classes. The intention is that that the elements be well-defined, and that they be as unique and independent as possible. Completeness is not attempted, although the elements should be sufficient for a discussion of issues facing complex space systems and systems-of-systems. Figure 1 shows the framework in its current form, with the four categories and their elements.

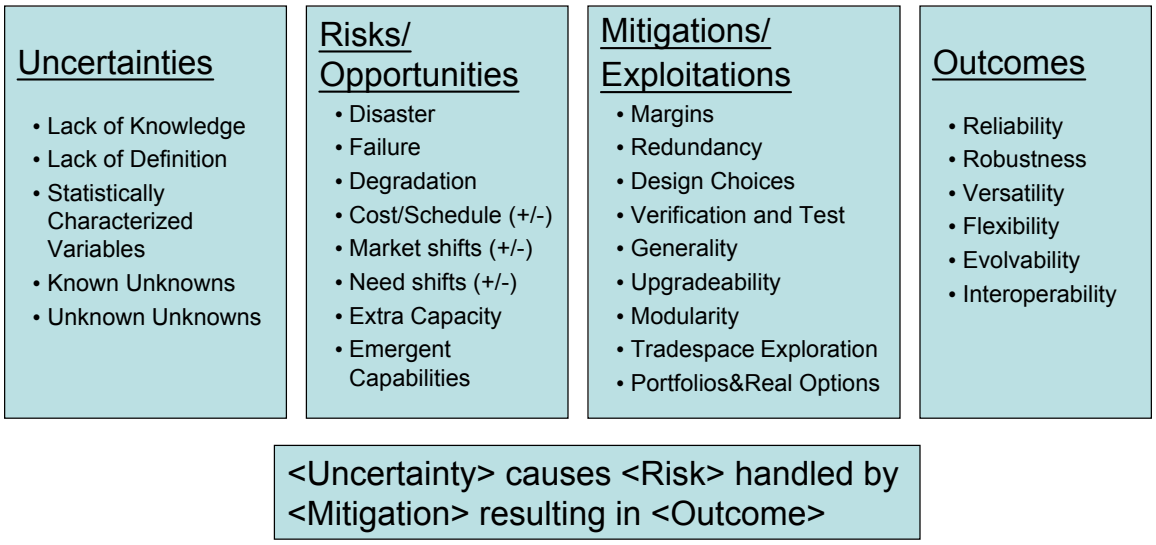

Figure 1. Framework for handling uncertainties and their effects.

Uncertainties:

Uncertainties are things that are not known, or known only imprecisely. There is no value judgment in stating that something is uncertain – it may be worse or better than expected. Uncertainties are factual and measurable; things are known, or not known, or known to a quantifiable degree or within quantifiable bounds.

The causes of uncertainty are numerous, specific to the system, environment, or context under study, and well addressed in the literature. Here, we will consider only the broadest categories of types of uncertainties; the examples will include some specific uncertainties, but in general the naming of uncertainties and their causes is left to the reader.

We will consider uncertainties from the point of view of the system architect or designer. Thus we make the important distinction that these uncertainties exist within the knowledge base of this person or (more probably) organization. They are not static, but will evolve over time as more information is collected. Two overarching classes of uncertainties exist, from the point of view of a system architect or designer working on a specific technical product:

- **Lack of knowledge:** *Facts that are not known, or are known only imprecisely, that are needed to complete the system architecture in a rational way.* This knowledge may simply need to be simply collected, or it may need to be created. It may even be unknowable, or knowable only at some time in the future. As this is written, the authors do not have fatigue properties of 7075-T6 aluminum handy, but they could either obtain it, or design a test program to do so. We cannot know

what material may have replaced this alloy in 2040, however, or at least not until somewhat closer to that distant date. Early in development there are (and should be!) many of these uncertainties; they must be systematically reduced at the appropriate time.

- **Lack of definition:** *Things about the system in question that have not been decided or specified*. Again, this is not a bad thing early in a program. A current challenge is to avoid defining too much about a system too early, both in terms of defining (bad) requirement and in over-specifying the nature of the solution before any work has been done. Again, these uncertainties must be systematically reduced at the appropriate time. Order is terribly important in reducing this unknown; the High Speed Civil Transport (HSCT) at the time of its cancellation had rivet spacings specified, but not fuselage materials or markets.

Within both of the above classes, the uncertainties can have one of several flavors. The types below are really three points in a continuum, from well-characterized statistical variation to complete lack of knowledge:

- **Statistically characterized (random) variables/phenomena:** *Things that cannot always be known precisely, but which can be statistically characterized, or at least bounded.* The fatigue properties of 7075-T6 aluminum fall nicely into this category. So do most environmental variables (weather, space environment, etc.). A strong characterization would be to know the statistical distribution of the possible values, to a known confidence level; a weaker characterization would be to know at least the bounds of the possible values. This type of uncertainty can be handled by powerful analytical techniques. Indeed, much of the science of Risk Analysis is dedicated to statistically characterizing uncertainties of various types which may lead to risks. An expanded definition of this important type is in Appendix A.

- **Known Unknowns:** *Things that it is known are not known*. They are at best bounded, and may have entirely unknown values. With time and/or effort, many items in this class may be characterized statistically. More frequently they are handled qualitatively or at best semi-analytically. Often, risk management plans are based on identifying Known Unknowns, and dealing with them essentially *ad hoc*. Future budgets, future adversaries, the performance of new technologies, and the like fall in this category.

- **Unknown Unknowns:** *Gotchas*. By definition not known. Some are hopeless to even contemplate (asteroid strikes vehicle). But, as the current Secretary of Defense might put it, we know there are unknown unknowns out there, which gives us some (difficult to quantify) motivation for applying conservative mitigation strategies. With experience, even unknown unknowns may be reduced to statistically characterized variables, e.g. large civil engineering structures have

very high margins based on the high probability that some time in 100+ years something strange WILL happen.[*]

Risks and Opportunities:

These are the consequences of the uncertainties to a program or system. The word "risk" emphasizes the down side, although there are also potential opportunities in the application of uncertainties to a system. Generally, risk can be quantified by considering (probability of problem)x(severity of problem), where "problem" is an undesirable resolution of an uncertainty. Conversely, an opportunity may exist if an uncertainty is resolved in a way favorable to the system. The math is the same: (probability of an event)x(value of the event). Again, we will not go into specific risks here, but will list general types

- **Disaster:** *System causes harm.* The converse, having the system prove to be a paradigm-changing "killer app" by random chance, is probably not worth contemplating.[†]

- **Failure / emergent capabilities:** *System does not work*; conversely, system works unexpectedly well, and/or for purposes not originally envisioned.

- **Degradation / unexpected capacity:** *System works, but not up to initial expectations*; conversely, system exceeds expectations.

- **Funding, cost, or schedule deviations:** *Program (to produce system) gets in one of several kinds of trouble*; conversely, is early or under budget. McNutt[6] showed that, at least in military projects, a variety of forces tend to make the uncertainty unsymmetrical – programs are almost never early or under budget.

- **Market shifts (+/-):** *System works, but need for its services has changed from assumed level*. This can be good or bad.

- **Need shifts (+/-):** *System works, but function desired from the system has changed from that for which it was designed.* Again, this can be bad (system may no longer meet needs) or good (need for system increases and/or new needs that the system can serendipitously satisfy are found).

---

[*] Two friends of one of the authors are alive now not because the designers of the Golden Gate Bridge had any clue that a mob of 100,000+ people would get stuck on it during its 50th anniversary party, but because they applied a very conservative load factor on the assumption that SOMETHING odd would happen sometime.

[†] Although it can happen. E-bay was founded by a graduate student who put together a website for exchanging *collectable candy dispensers* as a favor to his girlfriend.

Mitigations and Exploitations:

These are technical or programmatic things you do to avoid or manage risks, and/or exploit opportunities. They are not necessarily good things in and of themselves; on the contrary they are often expensive and must be justified by their effects on **outcomes**. This list is typical of strategies used or in consideration for aerospace systems; there are doubtlessly others.

- **Margins:** *Designing systems to be more capable, to withstand worse environments, and to last longer than "necessary."* Programmatically, to budget more money and take more time than one's initial estimate. All systems and programs do this, for very good reasons. Explored in depth on the technical side by Thunnissen.[7]

- **Redundancy:** *Including multiple copies of subsystems (or multiple copies of entire systems) to assure that at least one works.* Often, no extra capacity if all (sub) systems do work – redundant systems are unused if unnecessary. Requires overhead of "cross-wiring." Common.

- **Design Choices:** *Choosing design strategies, technologies, and/or subsystems that are not vulnerable to a known risk.*

- **Verification and Testing:** *Testing after production to drive out known variation, bound known unknowns, and surface unknown unknowns.* Testing occurs at all levels (component to system) and is needed to uncover bad components, check and bound known failure modes, and uncover unexpected ones (bad interfaces, unexpected failure modes) at great expense and some risk. Currently required for most systems.

- **Generality:** *Using Multiple-function (sub)systems and interfaces, rather than specialized ones.* Common example is general-purpose processor/computer rather than specialized chip or machine. Bus interfaces, or fully switched networks connecting redundant elements (instead of the minimum interconnection necessary to swap in for a dead unit), are examples of "general" interfaces. Common on the ground, less so in flight vehicles.

- **Serviceability/Upgradeability:** *(Sub) systems that can be modified to improve or change function.* Obvious difficult in non-retrievable vehicles like satellites although there are options even in this case (software, swarm components). Combining **general** hardware with upgradeable software is very powerful.[‡]

[‡] The Galileo Jupiter mission suffered from non-general hardware in its antenna (no reverse function) but upgradeable software and general capability in its other flight systems saved the mission (the computer could do data compression, not originally required, and data could be streamed through the low-gain antenna, intended for command data only).

- **Modularity, open architectures, and standard interfaces:** *Functions grouped into modules and connected by standard interfaces in such a way that they can "plug and play."* Not independent strategies, but greatly helps redundancy, generality, upgradeability, and makes testing easier (sometimes).

- **Trade Space Exploration:** *Analyzing or simulating many possible solutions under many possible conditions.* Using simulation modeling and (usually) massive computer power, provides a picture of how the system will respond to variations in both conditions and design choices, favorable or unfavorable.[8,9]

- **Portfolios and Real Options:** *Emerging technique originating in the financial world.* Allows program strategy of carrying various design options forward and trimming options in a rational way as more information becomes available and/or market conditions change. May also be useful for operations planning.

Outcomes:

These are the desired attributes of the system that quantify or at least characterize its interaction with uncertainties. There is a great deal of confusion as to the definition of these terms; below is a proposed set of reasonably concise definitions consistent with the dictionary definitions of the words and those used by advanced treatments on the subjects. Note that at least some of them can be applied to both systems (hardware) and programs – a robust hardware can work in bad weather, a robust program can survive a funding cut.

- **Reliability:** *Probability that the system will do the job it was asked to do (i.e. will work)*. Closely related to, though not the same as, some other –ilities (e.g. availability). This is the goal of most currently used risk management methods used in aerospace.[10,11]

- **Robustness:** *Ability of the system to do its basic job in unexpectedly adverse environments.* Well understood for non-aerospace products; aerospace products tend to be designed for *expected* adverse environments already, leading to minor ambiguities. Worse is the common tendency to use this word for any of the attributes on this list.

- **Versatility:** *Ability of the system, as built/designed, to do jobs not originally included in the requirements definition, and/or or to do a variety of required jobs well.* Often confused or combined with **Robustness** and **Flexibility** in discussions.

- **Flexibility:** *Ability of the system to be modified to do jobs not originally included in the requirements definition.* The modification may be in the design, production, or operation of the system; each has a unique flavor. These get tangled up when one considers system of systems, e.g. modifying (in

design/production) a vehicle and inserting it in a swarm, which changes the swarm's function (in service). Salah *et al.* have considered the resulting confusion, and clarified the role of flexibility in space system design.[12] This is a current area of intense research interest.[13,14,15]

- **Evolvability:** *Ability of the system to serve as the basis of new systems (or at least generations of the current system) to meet new needs and/or attain new capability levels.* An area of intense interest. [16,17,18] The US Air Force has declared evolutionary acquisition (not possible unless the systems are evolvable) to be their preferred approach to the acquisition of new systems.[19]

- **Interoperability**: Ability of the system to "play well with others," both with systems it was originally designed to work with, and with future systems. May be desirable in and of itself; also enhances versatility, flexibility and evolvability of systems of systems.

Current Methods

In an ideal world, methods would exist to collect knowledge of all the uncertainties facing a potential system, calculate all risks and opportunities implicit in these uncertainties, model the effects of all mitigation and exploitation strategies, and achieve all of the desirable system attributes. To illustrate *current* capabilities and practices, we project a number of them on the framework. The examples have little in common except that they deal with uncertainty in some way. Placing them in the framework clarifies what uncertainty is handled, in what way, and to what end. It also clarifies the relations between the studies, and the possible synergies between the techniques developed.

Most of the field of Risk Analysis (as defined for the purpose of this study as the contents of the journal of that name) is concerned with converting known unknowns such as the risk of a nuclear power plant accident or a plane falling on a house into statistically characterized variables that map one-to-one with "disaster" type risks. Sometimes design options are assessed in light of this risk. This well-populated, field occupies the upper-left corner of our framework, as shown in Figure 2.

Aerospace engineering practice in reliability and lifetime analysis takes well-characterized risks and mitigates them to achieve system reliability. Robustness is sometimes a goal; expected environments are always designed for; whether this constitutes "robust design" by our definition depends on how conservative the environment definitions are. The traditional mitigation techniques tend to be margins or redundancy.[11] Two recent studies examine different aspects of common engineering practice.

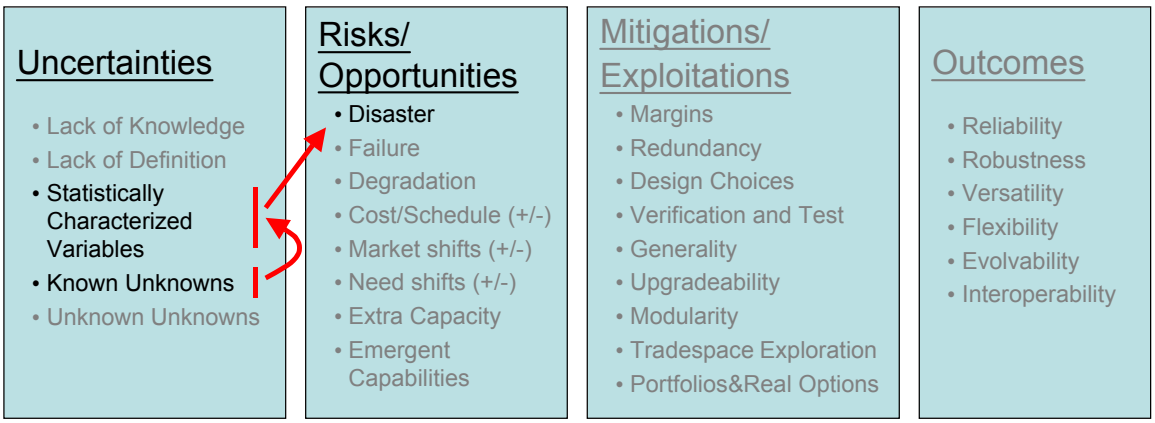

Figure 2. Risk Analysis

A recent study looked at the issue of design margins for conceptual and preliminary design.[7] The study explored in depth a standard practice in aerospace product development. The issue is the "fuzziness" of design at these stages; there is lack of knowledge (of use requirements, environments, etc.) and a lack of firm definition of the final form of the solution. A "best guess" solution would be at severe risk of failure or cost/schedule trouble (e.g. late rework); this is mitigated by the use of appropriately conservative design margins. The result is a solution that is robust to a range of requirements, environments, and final solution geometries. This rather linear walk through the framework is shown in Figure 3.

A technique for including component reliability effects in the conceptual design of space systems[1] deals with component and sub-system reliabilities as statistically characterized variables. Typically, components which are expected to have some chance of failure, but which are not fully characterized, are handled as "known unknowns", with estimated (and parametrically studied) reliabilities. The method calculates the chances of system failure or degradation, and allows designers to mitigate them, primarily through redundancy. The outcome is a reliable system. Figure 4 is also a linear walk through the framework, but handling a different set of uncertainties, in a different way. Note the possible synergy between this and the above method; component redundancy and margin could be traded to satisfy both known random variation and design immaturity by a hypothetical integrated process.

Risk Management (as defined by aerospace systems engineering guides and texts, e.g. Ref 10 as well as the contents of the journal of that name) is the process of uncovering the uncertainties that might pose a threat to a program, assessing the corresponding risks, and applying mitigations. It covers a bit more of our framework, as shown in Figure 5. However, as it is applied early in a design process, it is often qualitative or experienced-based. The analysis is usually used to avoid problems; it is rarely used to target opportunities. Risk management has historically been applied to the technical aspects of uncertainty;[11] applying it more broadly to include team dynamics, management, and other social factors is a subject of current research.[20]

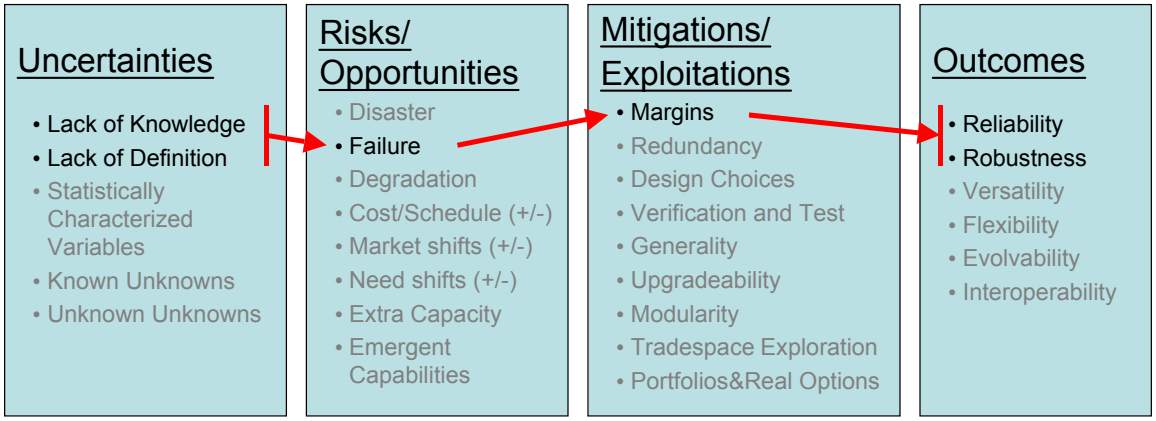

Figure 3. Use of margins in conceptual and preliminary design.

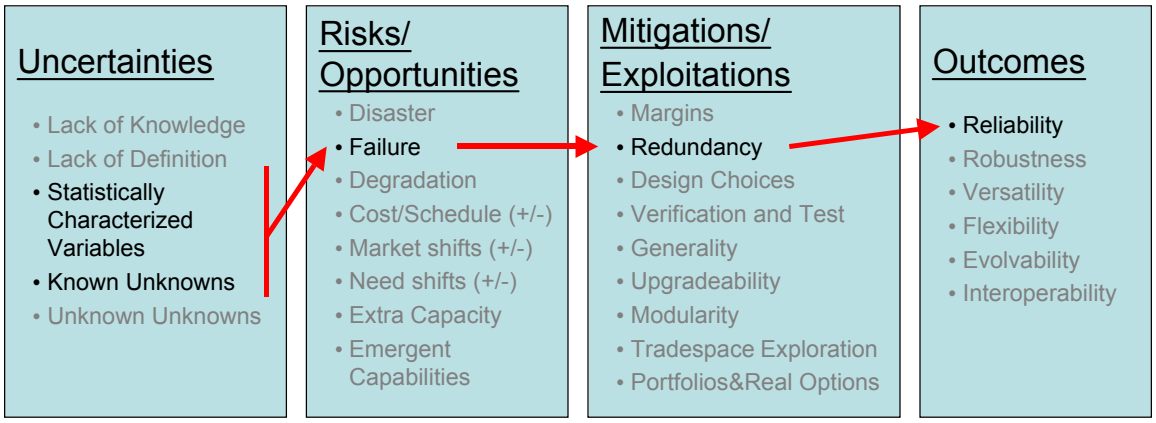

Figure 4. Reliability Engineering.

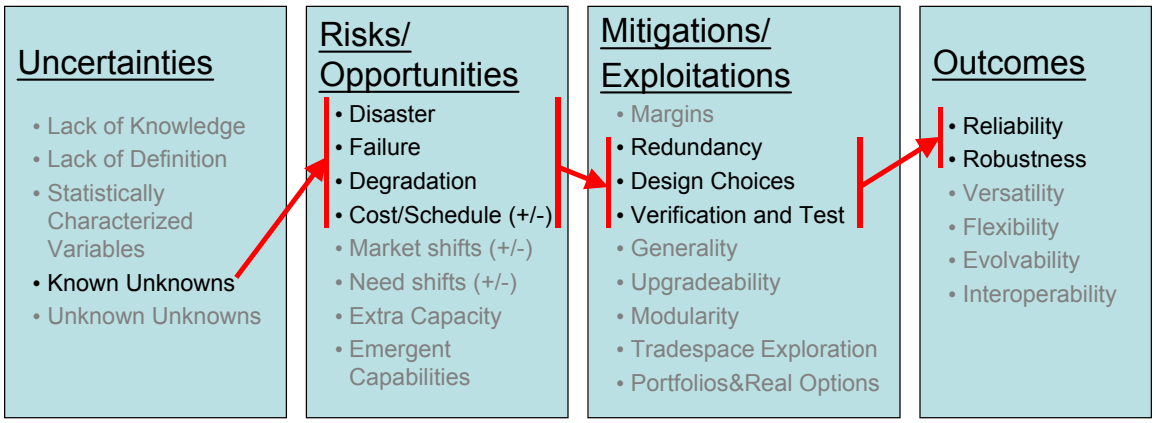

Figure 5. Risk Management.

Application of these methods to the "positive" side of the uncertainties is limited to date. Even simple engineering analyses could be used in this way, to (for example) quantify extra capacity or margin, but in general there is no motivation in traditional processes for creating "excess" capacity once requirements are met. In a similar fashion, risk analysis and management methods could be used to look for extra value opportunities, but in general are not. A major component of this problem is the use of fixed requirements or objective functions that do not reward upside capabilities, as opposed to bias in the methods themselves.

That said, traditional approaches to reliability and robustness (e.g margins and redundancy) often do add substantial extra value, if only serendipitously. Historically, systems such as the B-52 bomber (an extraordinarily flexible system), communication and scientific exploration satellites (which routinely exceed their planned lifetimes), and well-built civil infrastructure (which finds uses in systems not imagined by the builders) have, through their high margins and redundant systems or structures, provided a great deal of extra value to their builders and owners (and their descendants!). Intentionally providing extra value under uncertainty, as part of the system design, is the current challenge.

Emerging Capabilities

Multi-Attribute Tradespace Exploration (MATE) and similar techniques are powerful tools for considering uncertainty in conceptual design.[8,21] MATE is a tool for analyzing systems architectures with the goal of maximizing or trading various system attributes, rather than meeting specific requirements. It is not itself an uncertainty analysis tool, but rather allows the technical analysis of system concepts including the effects of uncertainties in various parameters. Sometimes tradespace analysis, even without explicit uncertainty analysis, can reveal the vulnerability of a proposed system, or conversely the opportunity for extra value. Figure 6 shows a tradespace for an orbital transfer vehicle, with cost on the vertical axis and utility (mostly determined by the ability to impart delta-V on other vehicles) on the horizontal.[22] Each point is an evaluated architecture. At a glance, one can see that architectures in group A are probably robust and have potential for further value creation. If, for example, the user's requirements change unexpectedly, there are "nearby" architectures of similar cost that can accommodate the user. A system using the architectures at B, on the other hand, would get in severe cost trouble if the user's demands increased; the only advantage to these architectures is the potential for savings if the users needs *decreased*.

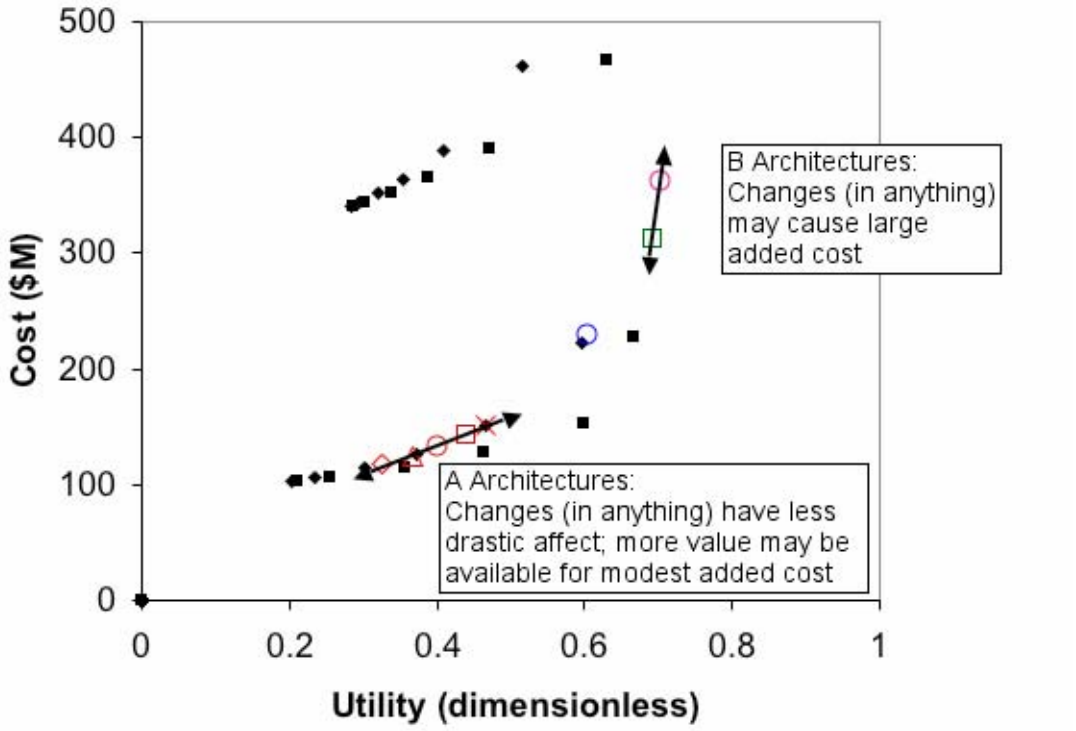

Figure 6. Tradespace (for orbital transfer vehicles) showing gross impact of uncertainties on two families of architectures.

Uncertainty can also be considered explicitly. Figure 7 shows a result from Reference 9. Various candidate architectures for a space-based broadband communication system were analyzed using the Generalized Information Network Analysis (GINA) method, and their predicted performance in terms of lifetime subscriber hours provided and cost plotted as the diamonds in Figure 7. Taking advantage of the ability of these methods to rapidly assess large numbers of architectures, a Monti-Carlo study was done. Many analyses were performed, varying both modeling assumptions (representing the lack of definition in these conceptual designs) and assumptions about the future market for this service (a lack of knowledge issue). The results of this study are shown as "error ellipses" around the mean-value diamonds. In this case, the uncertainties are large.

If the effects of uncertainty on the performance of systems can be quantified, the associated risk may be quantified, and mitigations applied.[9] In some cases, portfolio or real options methods may be used as risk management tools for complex systems. These techniques are particularly well suited for types of uncertainties that are difficult to quantify early in a program but become better known as time passes. In particular, they allow more systematic characterization of uncertainties such as market and needs shifts, budget uncertainties,[2] or policy mandates,[3] during the lifetime of programs. Using tradespace analysis, with or without portfolio theory, to mitigate abstract risks early in the design process is illustrated on the framework in Figure 8.

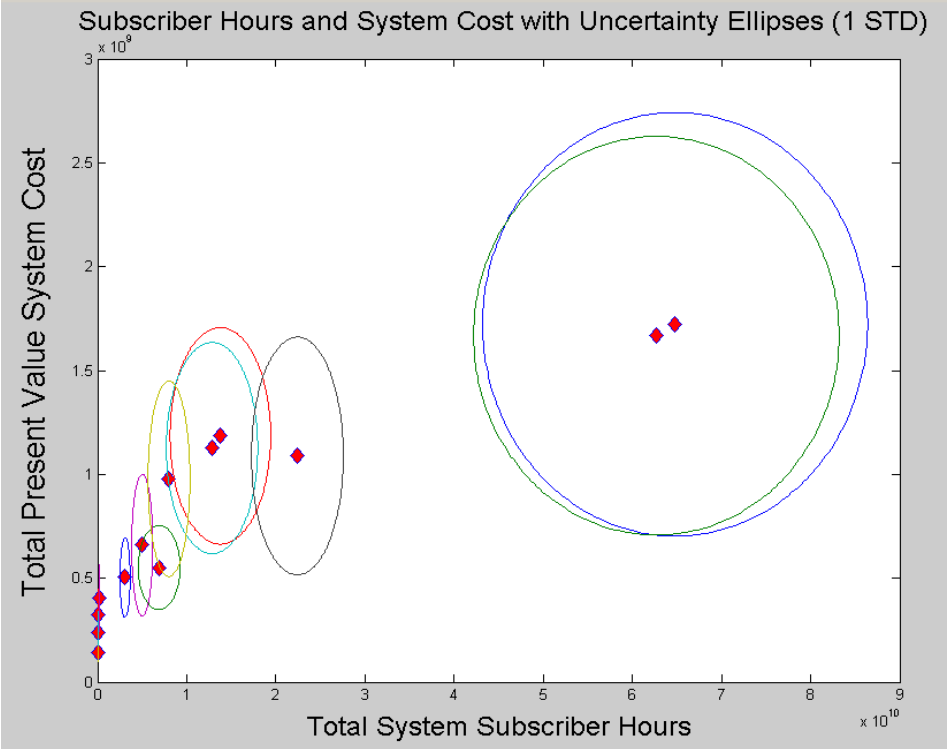

Figure 7. GINA analysis of broadband communication system including uncertainty (from Reference 9).

Portfolio and real options method are intrinsically designed (from their origins in the world of finance) to explore the trade of upside opportunities vs. possible additional risk. They can be used to design systems with desirable attributes such as versatility. The concept of versatility is explored,[12] and its value in aerospace systems defined,[15] by Salah *et al.* Some work has been directed at the design of systems (e.g. orbital transfer vehicles) the sole purpose of which is to enhance the flexibility of other systems.[13,14] The use of tradespace and financial methods to build flexibility into systems is shown in Figure 9.

The recent emphasis in "evolutionary acquisition" has focused attention on systems that can be adapted based on needs that will only be defined at a later time. Again, tradespace exploration is a tool that can evaluate multiple architectures base on not only their suitability for current needs, but their potential adaptability in the future. Preliminary work using this tool set is underway. The method has been applied on a trial basis to a number of systems.[16,17,18] Conceptually, this gives the system architect a tool to deal with true unknown-unknowns, as illustrated in Figure 10.

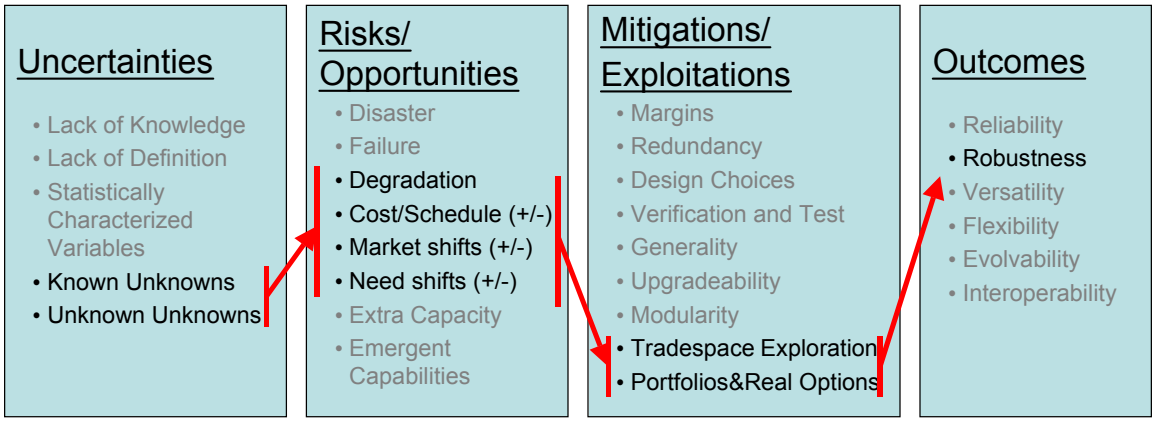

Figure 8. Use of tradespace exploration to deal with uncertainty, including "unknown-unknown" future events

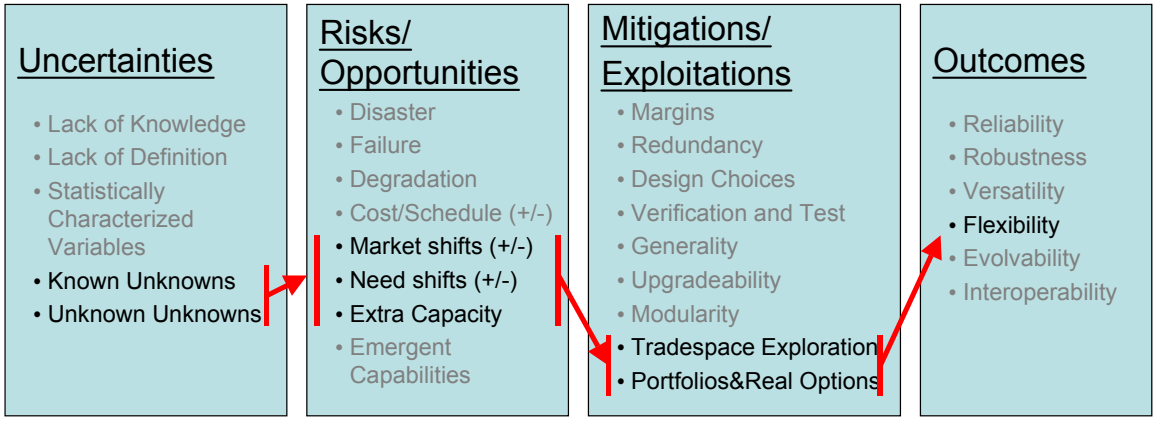

Figure 9. Achieving flexibility in uncertain environments.

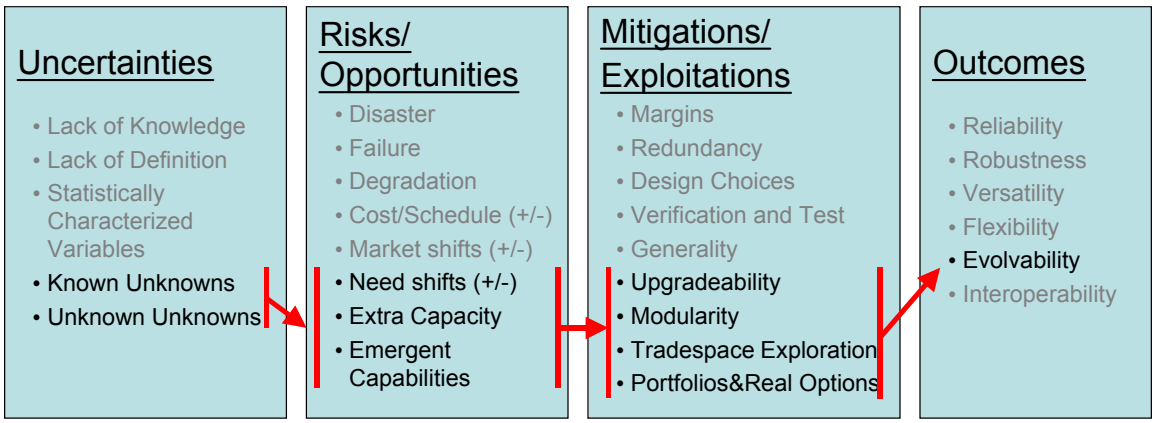

Figure 10. Preliminary techniques for defining evolutionary architectures.

A final example, shown Figure 11, looks at a much more speculative question: the use of a wireless "bus" system for satellite components. This would allow not only cable-free vehicles, but also architectures that would include multiple vehicles integrated to perform a single function without the overhead of dedicated inter-vehicle communications systems. Here, we enter the framework "in the middle." We have a mitigation (a modular, open architecture) that allows a flexible, evolvable systems. It is interoperable almost by definition, and it may (if implemented correctly) also allow rapid development of new capabilities. But what is it for? Conceivably, it could be used to address component risk (by adding new modules to a "swarm" to replace broken ones) but this is unlikely to pay off. Much more interesting is its ability to mitigate the risk that the user needs will shift and not be met by the existing system. Basic shifts in user needs are usually caused by "unknown unknowns" such as unexpected new missions and adversaries in defense, or new markets or sources of competition for civil systems. A capability (through launch of new sub-systems to join a wireless swarm) that would allow rapid adaptation to many classes of unanticipated problems would be very valuable. A similar idea, applied to docking rather than wireless components, and mitigating the important known risk of launch vehicle failure, is studied by Brown.[23]

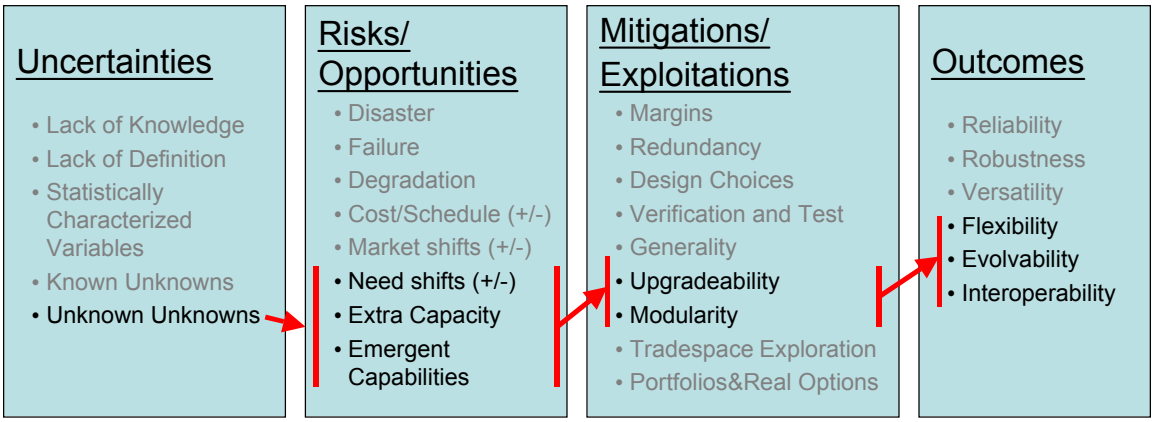

Figure 11. Dealing with unknown unknowns with modular, upgradeable systems

Summary

A framework is presented that can be used to understand the issues of uncertainty in complex system architecture determination. In the context of the framework, definitions of potentially confusing terms can be made, especially for system attributes such as robustness, flexibility, and evolveability. It is hoped that the framework and definitions will clarify discussion, and aid in teaching.

Current and developing methods are mapped on the framework. Mature, quantitative methods (e.g. Risk Analysis) were found to occupy the "upper left corner" of the framework, allowing known uncertainties and unsubtle risks to be quantified. Engineering practice uses straightforward techniques (e.g. margins and redundancy) to mitigate less well-characterized risks. In the "middle" of the framework, mature but more qualitative or approximate methods deal with known but less well-characterized uncertainties, allowing programs to develop mitigating strategies. In all of these cases, the aim is to avoid the downside of uncertainty.

Driven by demand to deal with "unknown unknowns" in shifting commercial markets and security threats, new methods are emerging. They deal with the mitigation of less well-understood uncertainties (e.g. funding, policy, or management), and also the exploitation of uncertainty to potentially increase future value. Powerful tools such as tradespace exploration, portfolio theory, and real options theory may allow not only risk mitigation, but the exploitation of uncertain environments through the fielding of versatile, flexible, and/or evolvable systems. These techniques occupy the "bottom right" of the framework. Ideally, the entire framework should be covered by tools; this would allow the system architect to work with any sort of uncertainty. The new tools must mature and enter general practice to achieve this ideal state. There is a great deal of work to be done before this can happen.

Appendix A – Empirical Uncertainty

The most widely used formalism for classifying uncertainty is probability. In order to be meaningful, any probability, classical or Bayesian, must pass what is known as the c*larity test*. To conduct the clarity test for a given probability, imagine a clairvoyant who knows all, and ask yourself whether such a person could either say unambiguously whether the event has occurred, or could give an exact value. Although this may sound trivial, it forces the necessary clarity for the probability to be meaningful. For example, "What is the price of gasoline?" does not pass the clarity test. This would have to be refined to something like "What was the price of gasoline at the Shell Station on Mass Ave in Cambridge at noon on January 4, 2001?" Only if it passes the clarity test is a probability worth trying to determine.

Let us define *empirical* quantities as measurable properties of real world systems, which must pass the clarity test. We can now discuss uncertainty in empirical quantities, which can arise from many sources. Seven sources relevant to systems architecting are:

- **Statistical variation:** Arises from random error in direct measurements of a quantity because of imperfections in measuring instruments and techniques.
- **Systematic error and subjective judgment:** Arises from biases in measurement apparatus & experimental procedure as well as from key assumptions by the experimenter.
- **Linguistic imprecision:** As described above. For example, phrases such as "fairly likely" and "highly improbable" give rise to uncertainty. Defining something so it passes the clarity test should get rid of this.
- **Variability:** When there is a natural frequency distribution associated with a variable, such as the weight of newborn children in Washington, DC over a year.
- **Randomness:** Describes quantities which must be viewed as random. One type of randomness is inherent: for example, in principle (specifically the Heisenberg Uncertainty Principle), the position and velocity of an electron cannot be known simultaneously. There are other quantities that although not technically random must be treated as such, because we cannot compute them accurately enough (e.g., weather prediction is very sensitive to initial conditions).
- **Disagreement:** arises from different technical interpretations of same data, as well as from different stakeholder positions in the outcome.
- **Approximations:** Examples include numerical (finite difference) approximations to equations and model reduction by approximation (e.g., spherical cows).

Appendix B: Systems Engineering and the Framework

The systems engineering process as captured in the NASA Systems Engineering Handbook[24] contains the following uncertainty management steps:

- **Pre-Phase A-Advanced Studies:** Risk estimates
- **Phase A-Preliminary Analysis:** Consider alternative design concepts, including: feasibility and risk studies and advanced technology requirements.
- **Phase B-Definition:** Prepare a Risk Management Plan; establish the…verification requirements matrix; Perform and archive trade studies; Reviews
- **Phase C-Design:** Refine verification plans; Perform and archive trade studies; Reviews
- **Phase D-Development:** Develop verification procedures; perform verifications; Reviews
- **Phase E-Operations:** Upgrade the system; Reviews

The early phases (pre-A to B) concentrate on risk studies and risk management plans.

Although the exact form of these plans is not specified, typically they follow the path shown in Figure 5: known unknowns are assessed in terms of the probability that they will create failure or suboptimal performance, and mitigated, mostly with design choices.

Alternate design choices are considered; and later trade studies are performed and "archived" in the intermediate phases (A to C). Again, these can vary from very simple considerations of alternatives that have a minimal role in risk mitigation or uncertainty exploitation, to full trade space studies. Typically they are simple studies to find optimal performance solutions. In the extreme case, they may approach the trade space explorations that can be used to understand uncertainty effects as shown in Figure 8, but this not the current practice.

Verification plays a major role in mitigating the downside of uncertainty in current practice. Planning and requirements definition begins in Phase B and continues through Phase D. Verification can be used to mitigate all risks to failure or lack of performance, obtaining reliable performance to requirements, often at a high price. This course of action is shown in Figure 12. The usual function of reviews is also to assure reliable attainment of requirements.

The only explicit consideration of the upside of uncertainty is upgrading the system in Phase E. This is currently done ad-hoc, although sometimes upgradability is designed into the system (e.g., the Hubble telescope).

Not stated explicitly in the systems engineering framework, but standard practice in preliminary and final design and engineering of most systems, is the role of margins and redundancy to achieve reliability and robustness as shown in Figures 3 and 4.

This brief exploration of a typical systems engineering framework shows that current practice concentrates on mitigating the downside of uncertainty to assure performance. Current practice does not preclude more advanced approaches (e.g. trade space exploration) but does not call for or encourage it. In particular, exploiting the upside of uncertainty to achieve flexible, evolvable, or adaptable system does not have an explicit place in current systems engineering practice.

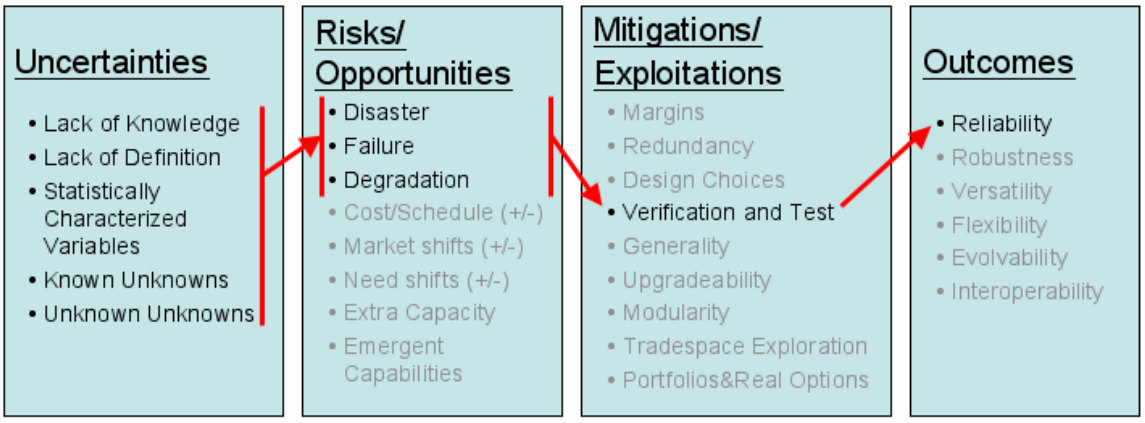

Figure 12. Traditional use of verification and test to mitigate downside uncertainty.

Biographies

Prof. Daniel Hastings is the Co-Director of the MIT Engineering Systems Division and a Professor of Aeronautics and Astronautics & Engineering Systems at MIT. His research has concentrated on issues related to spacecraft-environmental interactions, space propulsion, and more recently space systems engineering and space policy. He served as Chief Scientist of the Air Force from 1997 to 1999, and has led several national studies on government investment in space technology. He is a Fellow of the AIAA and a member of the International Academy of Astronautics.

Dr. Hugh McManus is a Senior Special Projects Engineer at Metis Design, working with MIT's Lean Aerospace Initiative and the Space Systems, Policy, and Architecture Research Consortium. He has an eclectic background that includes published work in aerospace systems development and analysis, structures and materials research and development, and educational innovation, done in both university and industry settings. He is an Associate Fellow of the AIAA.

References

[1] Benjamin, J. L., and Paté-Cornell, M. E., "A Risk Chair for Concurrent Design Engineering: a Satellite Swarm Illustration," *Journal of Spacecraft and Rockets*, Vol. 41, No. 1, 2004, pp. 51-59.

[2] Weigel, A. L., and Hastings, D. E., "Measuring the Value of Designing for Uncertain Future Downward Budget Instabilities," *Journal of Spacecraft and Rockets*, Vol. 41, No. 1, 2004, pp. 111-119.

[3] Weigel, A. L., and Hastings, D. E., "Evaluating the Cost and Risk Impacts of Launch Choices," *Journal of Spacecraft and Rockets*, Vol. 41, No. 1, 2004, pp. 103-110.

[4] Garber, R., and Paté-Cornell, M. E., "Modeling The Effects of Dispersion of Design Teams on System Failure Risk," *Journal of Spacecraft and Rockets*, Vol. 41, No. 1, 2004, pp. 60-68.

[5] deNeufville, R, "Uncertainty Management for Engineering Systems Planning and Design," Engineering Systems Symposium, MIT, Cambridge, MA, 2004.

[6] McNutt, R.T., "Reducing DoD Product Development Time: The Role of the Schedule Development Process," Doctoral Thesis, Massachusetts Institute of Technology, 1998.

[7] Thunnissen D. P., "A Method for Determining Margins in Conceptual Design," *Journal of Spacecraft and Rockets*, Vol. 41, No. 1, 2004, pp. 85-92.

[8] McManus, H. L., Hastings, D. E. and Warmkessel, J. M., "New Methods for Rapid Architecture Selection and Preliminary Design," *Journal of Spacecraft and Rockets,* Vol. 41, No. 1, 2004, pp. 10-19.

[9] Walton, M. A, and Hastings, D. E., "Applications of Uncertainty Analysis Applied to Architecture Selection of Satellite Systems" *Journal of Spacecraft and Rockets,* Vol. 41, No. 1, 2004, pp. 75-84.

[10] NASA Systems Engineering Handbook, SP-610S, June 1995, section 4.6.

[11] Walton, Myles, A., "Managing Uncertainty in Space Systems Conceptual Design Using Portfolio Theory," Doctoral Thesis in Aeronautics and Astronautics, June 2002, Chapter 3.

[12] Saleh, Joseph H., Hastings, D. E., and Newman, D. J., "Flexibility in System Design and Implications for Aerospace Systems," *Acta Astronautica,* Vol. 53, 2003, pp. 927-944.

[13] J. Saleh, E. Lamassoure and D. Hastings, "Space Systems Flexibility provided by On-Orbit Servicing I", *Journal of Spacecraft and Rockets*, Volume 39, Number 4, pp. 551-560, 2002.

[14] E. Lamassoure, J. Saleh and D. Hastings, "Space Systems Flexibility provided by On-Orbit Servicing II", *Journal of Spacecraft and Rockets*, Volume 39, Number 4, pp. 561-570, 2002.

[15] Saleh, J.H., Marais, K. S., Hastings, D. E., and Newman, D. J., "The Case for Flexibility in System Design," INCOSE 2002 International Symposium, July-August 2002, Las Vegas, NV.

[16] Roberts, Christopher J., "Architecting Evolutionary Strategies Using Spiral Development for Space Based Radar," Masters Thesis in Technology and Policy, Massachusetts Institute of Technology, June 2003.

[17] Spaulding, T. J, "Tools for Evolutionary Acquisition: A Study of Multi-Attribute Tradespace Exploration (MATE) applied to the Space Based Radar (SBR)," Masters theses in Aeronautics and Astronautics, Massachusetts Institute of Technology, June 2003.

[18] Derleth, Jason E., "Multi-Attribute Tradespace Exploration and its Application to Evolutionary Acquisition," Master's Thesis in Aeronautics and Astronautics, Massachusetts Institute of Technology, June 2003.

[19] Wolfowitz, P., Memorandum: "Defense Acquisition," Office of the Deputy Secretary of Defense, 30 Oct 2002.

[20] See, for example, Paté-Cornell, M. E., and Dillon, R., "Advanced Programmatic Risk Analysis for Programs of Dependent Projects Involving Critical Systems and Unmanned Space Mission Illustration," Proceedings of PSAM5: International Conference on Probabilistic Safety Assessment and Management, Vol. 1, Osaka, Japan, Nov. 2000, pp. 203-209, and Garber, R., and Paté-Cornell, M. E., "Modeling The Effects of Dispersion of Design Teams on System Failure Risk," *Journal of Spacecraft and Rockets*, Vol. 41, No. 1, 2004, pp. 60-68.

[21] Ross, A. M., Diller, N. P., Hastings, D. E., and Warmkessel, J. M., "Multi-Attribute Tradespace Exploration with Concurrent Design as a Front-End for Effective Space System Design," *Journal of Spacecraft and Rockets*, Vol. 41, No. 1, 2004, pp. 20-28.

[22] McManus, H. L. and Schuman, T. E., "Understanding the Orbital Transfer Vehicle Trade Space," AIAA Paper 2003-6370, Proceedings of AIAA Space 2003, Long Beach, CA, Sept. 2003.

[23] Brown, Owen, "Reducing Risk of Large Scale Space Systems Using a Modular Archietcture," Defense Advanced Research Projects Agency.

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