25th Annual INCOSE International Symposium (IS2015)
Seattle, July 13-16, 2015

A Systems Engineering Approach to Architecture Development

David A. Di Pietro
NASA Goddard Space Flight Center, Mail Code 599, Greenbelt MD 20771
Phone: 301 286-0240, E-mail: david.a.dipietro@nasa.gov, ddipietro@erols.com

Abstract. Architecture development is often conducted prior to system concept design when there is a need to determine the best-value mix of systems that works collectively in specific scenarios and time frames to accomplish a set of mission area objectives. While multiple architecture frameworks exist, they often require use of unique taxonomies and data structures. In contrast, this paper characterizes architecture development using terminology widely understood within the systems engineering community. Using a notional civil space architecture example, it employs a multi-tier framework to describe the enterprise level architecture and illustrates how results of lower tier, mission area architectures integrate into the enterprise architecture. It also presents practices for conducting effective mission area architecture studies, including establishing the trade space, developing functions and metrics, evaluating the ability of potential design solutions to meet the required functions, and expediting study execution through the use of iterative design cycles.

Preface

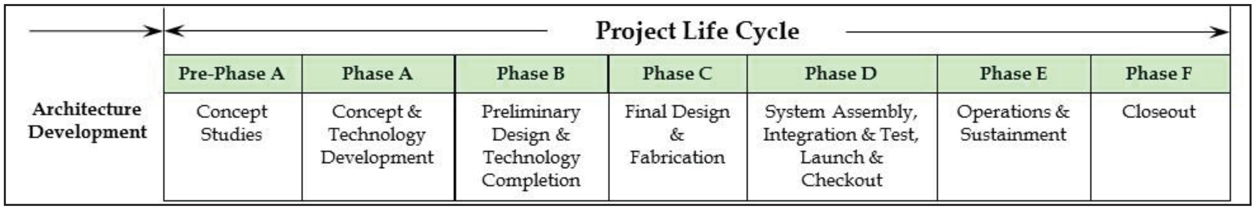
The terms “architecture” and “architecture framework” are used frequently within government and industry, but they are often used with context dependent definitions that can be associated with multiple product hierarchy layers and varying time frames. For example, “architecture” may be used to describe existing hardware and software such as a computer processor instruction set, a computer, or a network of computers. Or, it may be used to describe a collection of candidate systems such as spacecraft or launch vehicles that may be developed for operation in the far-term future. Similarly, “architecture framework” has a wide range of definitions. For example, ref. (a) includes a survey of over 60 architecture frameworks (enterprise, defense, information technology, software, automotive, business, etc.), such as the Department of Defense Architecture Framework (DoDAF), The Open Group Architecture Framework (TOGAF), the Federal Enterprise Architecture Framework (FEAF), etc., with varying scope and taxonomies. Such a wide range of definitions can lead to differing perspectives among architecture development teams (ADTs), their customers, and their stakeholders, particularly when teams span multiple organizations. These differing perspectives can slow ADT progress and cause variations in product content and fidelity.

This paper provides techniques to help ADTs, their customers, and their stakeholders quickly gain a collective understanding of what an architecture is when planning for the far-term future and illustrates how such an architecture might get developed for space systems. It describes a fundamental systems engineering approach that enables architecture development without need for special training in the use of more complex and abstract models and data structures such as those associated with DoDAF. It uses a multi-tier framework to describe the enterprise level architecture and illustrates how results of lower tier mission area architectures (MAAs) are developed and integrated into the enterprise level architecture. This paper is based on original work developed for the Department of Defense (ref. (b)), adapted for use at NASA (ref. (c)), and presented at ref. (d). As the techniques discussed here use widely understood methods and terminology, their use is not limited to space architecture development.

Architecture Studies - Beginning Thoughts

Architecture studies are conducted prior to Pre-Phase A of the project life cycle as shown in Fig. 1, adapted from ref. (e), to inform planners on recommended capabilities and investment profiles prior to conducting concept design studies for a particular mission. They can be conducted at either mission area or mission levels, and their scope is broader and shallower than the scope of concept design studies conducted in both Pre-Phase A and Phase A.

Mission area architecture studies typically are conducted to determine the best-value mix (number, capability, rough cost, etc.) of MAA assets that works collectively in specific scenarios and time frames to accomplish a set of mission area objectives. Results are used to inform planners on recommended capabilities and investment profiles across the mission area. Alternatively, mission architecture studies are conducted to determine the best approach to meet objectives for a particular mission. Mission architecture studies are conducted when little is known of a mission and when significantly different mission approaches exist, such as for a first-time expedition to study a moon of Saturn.[1] Their scope is narrower and deeper than that of MAAs, and their results are used to inform planners on the most cost effective approach for a particular mission. Allowing for some exceptions, mission architectures typically will address nearer term capabilities than will MAAs.



	Project Life Cycle						
	Pre-Phase A	Phase A	Phase B	Phase C	Phase D	Phase E	Phase F
Architecture Development	Concept Studies	Concept & Technology Development	Preliminary Design & Technology Completion	Final Design & Fabrication	System Assembly, Integration & Test, Launch & Checkout	Operations & Sustainment	Closeout

Figure 1: Architecture Development Precedes Concept Design in the Project Life Cycle

As the potential range of perspectives for the term, “architecture”, is significant (e.g., whether it implies design, or whether it implies only building codes or behaviors) it is useful to begin with an objective definition such as in ref. (h) that conveys the basic intent.

Architecture is defined in ref. (h) as: 1) “the art or science of building; *specif*: the art or practice of designing and building structures and esp. habitable ones”, 2) “formation or construction as or as if the result of conscious act”, 3) “architectural product or work”, 4) “a method or style of building”. Reference (h) further describes Architect, from Latin “architectus”, from Greek: “*architekton* master builder”, as: 1) “one who designs buildings and superintends their construction”, 2) “one who plans and achieves a difficult objective” (e.g., a military victory).

[1] An example mission architecture study is discussed in refs. (f) and (g).

From these definitions, it is clear architecting typically involves some level of design, but what level of design, and is design all there is to it? What does an architecture look like, and what does it do? To answer these questions, a notional civil space (NCS) enterprise architecture, or "NCS architecture" is introduced. We also will need a common view of 1) the core elements of the NCS architecture, and 2) the NCS architecture, including its constituent MAAs.

Describe What an MAA is and How it Integrates into the NCS Architecture Framework

Core Elements of the NCS Architecture. The core elements of the NCS architecture[2] are as below. Each element pertains to specific period in time, or "epoch".

1) The set of functional capabilities that characterizes the actual or forecast capabilities of NCS physical assets and human command and control (C2) entities. Includes "what" capability will be delivered along with measures of performance (MOPs), e.g., quality, quantity, timeliness, interoperability, and robustness (QQTIR).[3]

2) The set of NCS physical assets, both hardware and software, that is, or is forecast to be, available along with their interconnectivities. Shows "how" architecture functional capabilities will be delivered.

3) The set of NCS human C2 operator and decision maker entities available along with their interconnectivies.[4]

4) The concept of operations (CONOPS) that identifies how NCS physical assets and human C2 entities will be employed in time sequence to meet a defined mission. Used to evaluate effectiveness as a function of environment and scenario.

5) The set of constraints, i.e., rules, policies and standards, protocols, etc., that constrain the use of NCS physical assets and human C2 entities.

NCS Architecture Framework Example. The NCS architecture framework is established by functional decomposition, a standard systems engineering technique which enables vertical flowdown of guidance and identification of horizontal interfaces within and among architectures. Figure 2 shows an example that highlights a partial functional decomposition (shaded boxes) for the Space Access mission area for a specific epoch.

In this example, Tier 0 of the framework represents the enterprise level as it includes functions and assets applicable to the entire NCS architecture. At Tier 1, Tier 0 functions are allocated to mission area functions, e.g., "Space Access" (read as "provide Space Access capabilities"). At Tier 2, Tier 1 functions are allocated to sub-mission area functions, e.g., "Space Access" allocates functions to "Spacelift / Payload Transportation". At Tier 3, Tier 2 functions are allocated to more detailed functions, e.g., "Spacelift / Payload Transportation" allocates

[2] The NCS architecture is notional and for illustration only. No such architecture has been defined.
[3] This is a minimum set of metrics
[4] Automated C2 assets are considered part of the physical assets

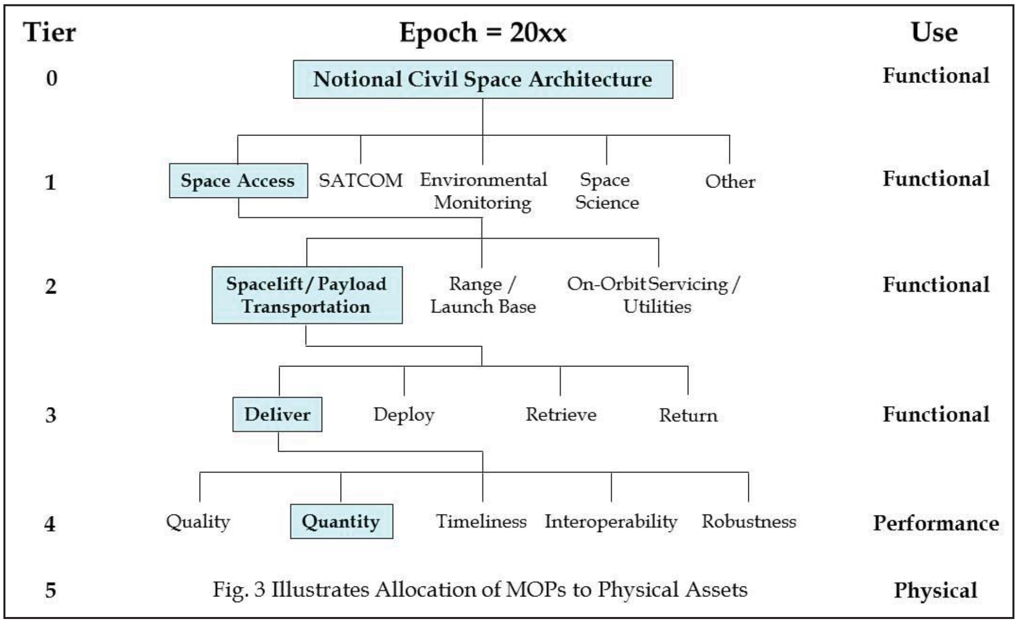


Figure 2: NCS Architecture Framework Highlighting Space Access Mission Area

functions to “Deliver”, “Deploy”, “Retrieve”, and “Return”. At this point in the Space Access functional decomposition, mission area functions have been characterized to a point where QQTIR metrics and MOPs may be established. For example, at Tier 4, an example quantity metric for the “Deliver” function might be “x payloads of y,000 kg to z,000 km circular orbit at i° inclination”. These metrics become MOPs when numerical values are assigned, e.g., “2 payloads of 2,000 kg to 400 km circular orbit at 51.6° inclination”. At Tier 5, Tier 4 MOPs are allocated to physical assets and human C2 entities. The number of tiers can vary among (or within) mission areas as needed to sufficiently characterize each mission area. When multiple stakeholder organizations participate in an MAA development, Fig. 2 can be viewed as having a third dimension, with the allocation of functions to stakeholders going into the page. Table 1

Table 1: Functional Decomposition for Space Access Mission Area (Epoch = 20xx)

Tier 0	Tier 1	Tier 2	Tier 3	Tier 4
1.0 Provide NCS capabilities	1.1 Provide Space Access capabilities	1.1.1 Provide Spacelift / Payload Transportation capabilities	1.1.1.1 Provide capability to **deliver** payload(s) to orbit	**1.1.1.1.1 Quality**
				1.1.1.1.2 Quantity
				1.1.1.1.3 Timeliness
				1.1.1.1.4 Interoperability
				1.1.1.1.5 Robustness
			1.1.1.2 Provide capability to **deploy** payload(s) on orbit	**1.1.1.2.1 Quality**
				1.1.1.2.2 Quantity
				1.1.1.2.3 Timeliness
				1.1.1.2.4 Interoperability
				1.1.1.2.5 Robustness
			1.1.1.3 Provide capability to **retrieve** payload(s) on orbit	Continue as done for 1.1.1.1
			1.1.1.4 Provide capability to **return** payload(s) from orbit	Continue as done for 1.1.1.1
		1.1.2 Provide Range / Launch Base capabilities	Continue as done for 1.1.1	Continue as done for 1.1.1.1
		1.1.3 Provide On-Orbit Servicing / Utilities capabilities	Continue as done for 1.1.1	Continue as done for 1.1.1.1

provides a tabular view of the functional decomposition example in Fig. 2 for the Space Access mission area. The decomposition can be continued for the balance of mission areas to represent the full NCS architecture.

Architecture Guidance and Constraints. Guidance and constraints applicable to all mission areas are reflected in functions assigned at Tier 0. For example, Tier 0 may reflect environmental policy associated with power or fuel sources, orbital debris, or planetary protection. It also may include interoperability standards as well as criticality categories that drive the level of robustness (i.e., fault tolerance) needed. Criticality categories might include specific levels of fault tolerance required for assets and capabilities that pertain to: 1) human survival, 2) specific mission operational capabilities, and 3) specific technology capabilities. A fault in this context means loss of capability for any reason (component failure, hostile action, etc.). The severity of a potential fault can depend on the severity of the threat. Tier 1 adds guidance unique to each Tier 1 mission area.

Figure 3 shows the physical view of the Space Access MAA. It includes example assets (i.e., "nodes") that might be assigned to provide capabilities that meet the MOPs from the functional decomposition. In Fig. 3, LEO, MEO, and GEO correspond to low, medium, and geosynchronous Earth orbits, respectively. Assets beyond GEO have been excluded.

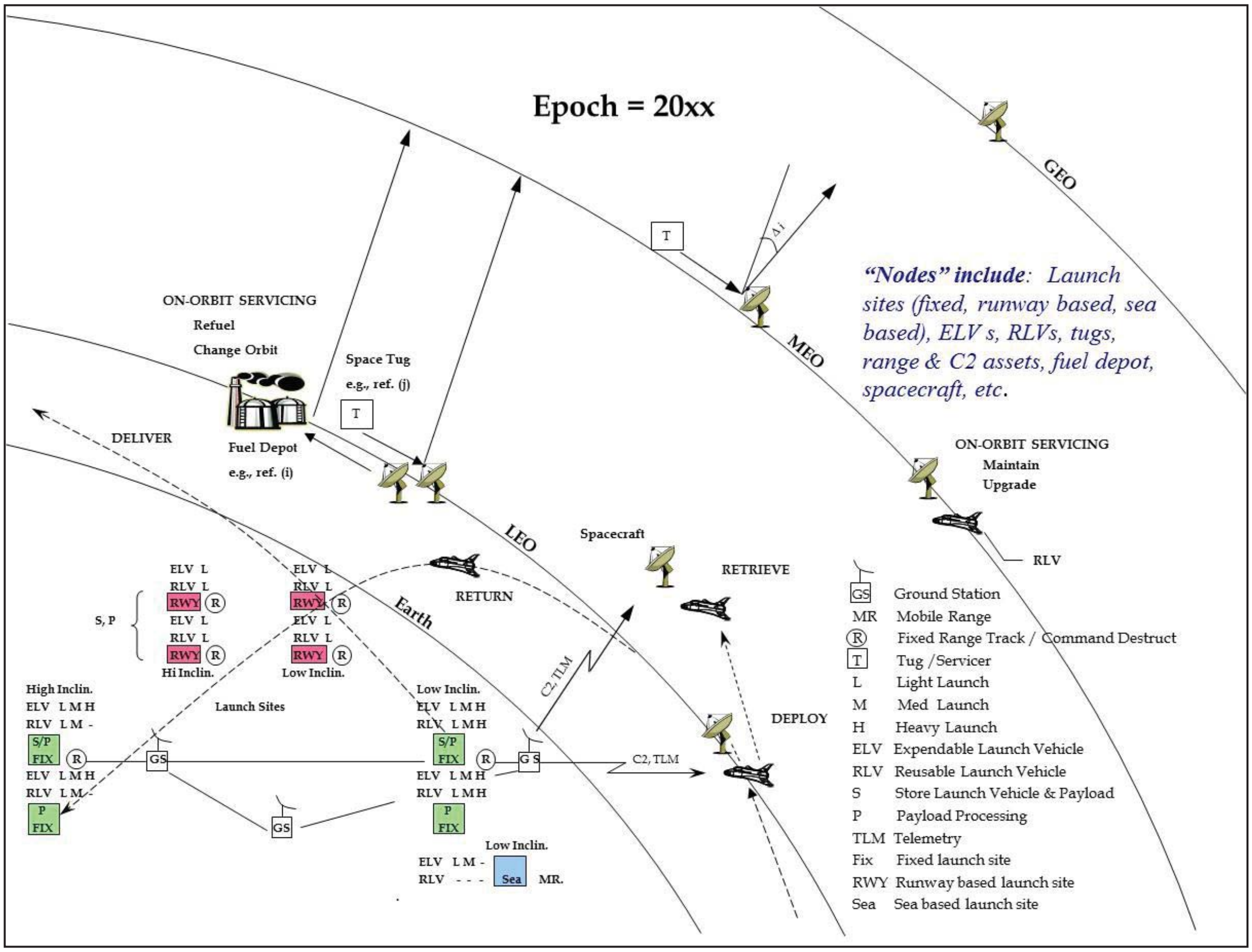


Figure 3: Physical View for Space Access Mission Area Architecture Assets

While the scale of the NCS architecture is large, it may exist as part of an even larger notional collection of architectures that crosses domains and stakeholders as depicted in Fig. 4 under a notional authority structure. Integration with these adjacent architectures may introduce additional considerations for, or impose additional constraints on, the NCS architecture. The NCS architecture may exist adjacent to informal architectures and collections of systems as well. Such informal entities also may impose constraints on the NCS architecture.

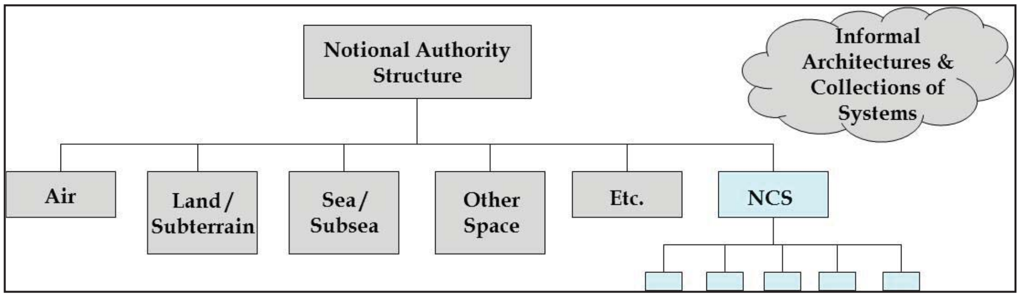

Notional Authority Structure
Informal Architectures & Collections of Systems
Air
Land / Subterrain
Sea / Subsea
Other Space
Etc.
NCS

Figure 4: NCS Architecture May be Influenced by Other Notional Architectures

Level of Design Work Conducted in MAA Development. MAA technical analysis typically is limited to first principles at the system (e.g., spacecraft or launch vehicle) level. Subsystem level details typically are deferred to later phases. For example, considering the space tug used to service and maneuver spacecraft in Fig. 3, the ADT might determine the tug mass to the first order via use of the rocket equation (ref. k) accompanied by other simple scaling relationships such as dry mass to propellant mass ratio. Use of the rocket equation presumes knowledge of the propulsion subsystem efficiency (i.e., specific impulse) which is important in vehicle level sizing and performance. While the efficiency also would indicate the type of propulsion subsystem used (e.g., solid propellant, monopropellant, bi-propellant), the ADT typically would not design the propulsion subsystem tanks, lines, and nozzles during an MAA study. Teams conducting mission level architecture studies, such as discussed in refs. (f) and (g), may need to do some subsystem design analysis, though typically not to the level performed in concept design.

Measures of Effectiveness (MOEs). MOEs typically address effectiveness at the architecture level and differ from MOPs. Where MOPs might pertain to sizing nodes for spacelift, range, and on-orbit servicing functions, MOEs might pertain to how well such nodes combine to meet an operational scenario at the MAA level. MOEs typically need to be decomposed into measurable terms in order to be useable by ADTs. Developing and refining MOEs requires early and continued customer and user engagement.

Architecture Scenarios and Environments. MAAs are developed using specific scenarios that include the driving operational cases at the MAA level. In conjunction with scenarios, MAAs also must consider the environments under which specific systems within the architecture will be developed and operated. Examples of environments include: 1) stable / cooperative vs. unstable / uncooperative governments, 2) stable vs. unstable budgets, 3) contested vs. uncontested space operations, 4) orbital debris, and 5) space weather, among others. A key enabler for conducting NCS architecture level (i.e., Tier 0) effectiveness analysis across multiple MAAs is the use of consistent scenarios and environments for a given epoch.

Interface Identification. During MAA development, horizontal interfaces are likely to emerge within or among MAAs. For example, the transmit data rate and frequency from a remote sensing node in an Environmental Monitoring MAA might interface with a ground station node developed for a satellite communications (SATCOM) MAA. These interfaces can be highlighted on the functional decomposition. In addition, some physical interfaces may need to be standardized to enable interoperability. Horizontal integration analyses across MAAs validates interface compatibility.

Mission Area CONOPS Development and Use. Each MAA has at least one CONOPS that applies to a particular scenario, environment, and epoch. The CONOPS is used to evaluate MAA effectiveness and assess satisfaction of user needs. The CONOPS is specific to the architecture design. For example, an operational scenario likely will be met differently by a CONOPS using reusable launch vehicles (RLVs) and on-orbit servicing than by a CONOPS using only expendable launch vehicles (ELVs).

NCS Architecture CONOPS Development and Use. An NCS architecture level CONOPS, used to evaluate the effectiveness of the NCS architecture, may be of interest if the NCS MAAs have significant operational interrelationships. At Tier 0, a CONOPS would be developed using scenarios, environments, and epochs consistent with those evaluated for each MAA. NCS architecture effectiveness would be periodically evaluated using this CONOPS with a fixed or "frozen" architecture design. Results would highlight areas of the NCS architecture that underperform, for example, due to a capability gap or overlap or due to an interface incompatibility. These areas may emerge as candidates for study in the next cycle of MAA designs.

NCS Architecture Framework Use. The NCS architecture framework has several uses, e.g., it:

- Provides for structured flowdown of policy and guidance into MAAs
- Allows synthesis of Tier 0 (enterprise) architecture from constituent MAAs for a given epoch
- Facilitates identifying Tier 0 CONOPS and evaluating Tier 0 architecture effectiveness
- Enables horizontal and cross-organization interface integration within or among MAAs
- Exposes gaps and overlaps usable to identify the need for follow-on MAA studies
- Clearly highlights whether studies are for: 1) one mission area across all QQTIR metrics, or 2) all mission areas for only one metric, e.g., timeliness
- Establishes a common lexicon for functions, metrics, and products
- Provides coherent context and relationships among architecture elements

Describe an Effective Approach for Developing an MAA

With the NCS architecture framework and MAAs described, the following sections summarize the basics of an effective approach usable by ADTs to conduct MAA architecture development.

Terms of Reference (TOR). One of the first tasks for the organization conducting the MAA development is to develop a TOR and gain customer approval. The TOR clearly identifies the "who, what, where, why, when" of both the study process and products and should include resources, participants, roles, and responsibilities.

The TOR typically will include: 1) background on the study problem, including any relationship of the present study to relevant past studies, 2) a concise and clear problem statement, and 3) the study scope and product depth, to include functional boundaries (e.g., include spacelift, but exclude on-orbit servicing), stakeholders, domains, and epoch (Fig. 5). The TOR also typically will include: 4) mission area guidance, including relevant policy directives, 5) guidance for establishing MOEs, 6) definitions for key unique terms, 7) assumptions, constraints, and ground rules (e.g., scenarios and environments, technology readiness date, policy, cost, direction to exclude specific systems or stakeholders, or direction to use a specific data source), and 8) guidance on how to select the recommended architecture (e.g., single, best value architecture within a specific cost constraint).

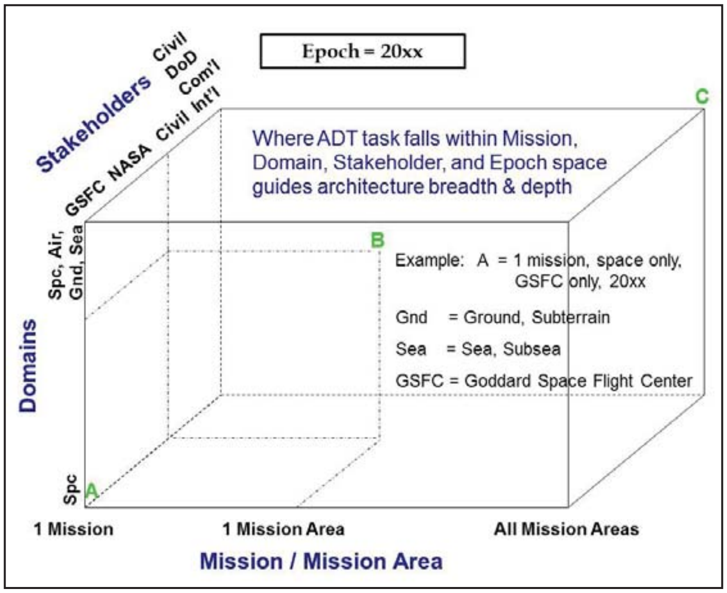

Epoch = 20xx
Stakeholders
GSFC NASA Civil Int'l Com'l DoD Civil
Where ADT task falls within Mission, Domain, Stakeholder, and Epoch space guides architecture breadth & depth
C
B
Example: A = 1 mission, space only, GSFC only, 20xx
Gnd = Ground, Subterrain
Sea = Sea, Subsea
GSFC = Goddard Space Flight Center
Domains
Spc, Air, Gnd, Sea
Spc
A
1 Mission
1 Mission Area
All Mission Areas
Mission / Mission Area

Figure 5: Architecture Scope and Depth Considerations for TOR

Deceptively difficult to develop, a strong TOR is central to conducting an effective MAA study and can save ADTs significant time. Conversely, a weak TOR can leave the ADT to define such key aspects as study purpose, scope, depth, epoch, and deliverables while concurrently designing the MAA. This not only can potentially overload an ADT and delay product delivery, it also can yield a study product that is inconsistent with the customer's view.

Current, Mid-Term, and Far-Term Architectures. Figure 6 shows architectures are developed for three epochs: current, mid-term, and far-term.[5] The current, mid-term, and far-term architectures are referred to as the "as-is", "to-be", and "should-be" architectures, respectively. The "as-is" architecture is associated with current (assumed as FY 2010 in Fig. 6) systems and capabilities. The "should-be" architecture is associated with systems and capabilities the ADT determines to be needed in the far-term epoch of FY 2035 at full operational capability. The "to-be" (planned) architecture is associated with systems and capabilities that would result from proceeding on the planned development path (i.e., includes efforts funded in any year of the current budget) until the mid-term epoch of FY 2020.

[5] Adapted from model used by ref. (l)

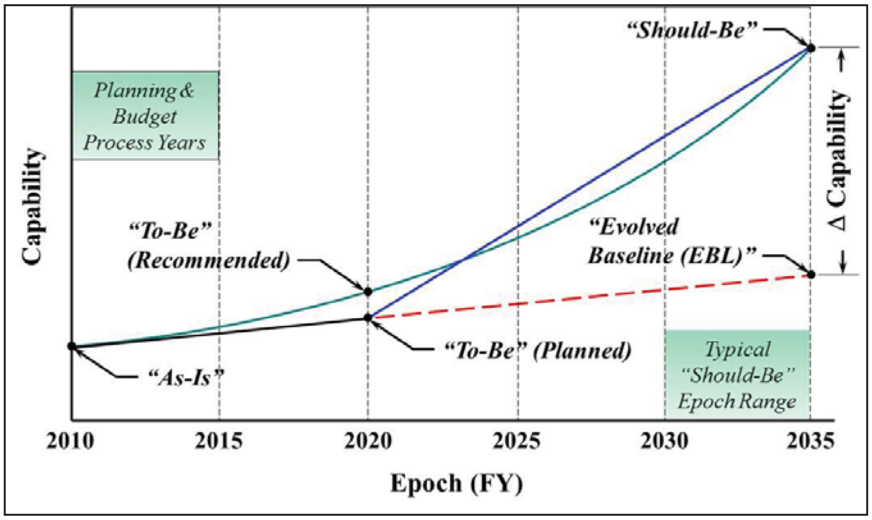


Figure 6: "As-Is", "To-Be", "Should-Be", and "Evolved Baseline" MAAs

Technologies are incorporated into these architectures according to technology readiness date guidance in the TOR. The Evolved Baseline, or EBL, is a linear extrapolation of the "to-be" (planned) architecture out to the far-term epoch. That is, the EBL assumes no breakthroughs afforded by new technologies, new operational doctrine, or new policies.

Figure 6 shows achieving the "should-be" capability from the "to-be" (planned) capability would require a marked change at the mid-term epoch. To avoid relying on such an occurrence, the ADT compares the "should-be" architecture capabilities to EBL capabilities and translates needed improvements into recommended near-term program and budgetary adjustments. The result yields a "to-be" (recommended) architecture that enables more gradual and continuous improvement toward the "should-be" capability.

Study Epoch is a Pivotal Parameter. Selection of the far-term epoch in Fig. 6 shapes MAA study conduct, as it guides the breadth and depth of analysis needed. As an example, consider two far-term epochs, one set 25 years in the future and the other set 15 years in the future. An MAA with a far-term epoch set at 25 years typically will be broader and shallower than one with a far-term epoch set at 15 years. Given the extended timelines associated with developing some space systems, setting the far-term epoch at 15 years implies ADT discussions are likely to a) be more influenced by near-term budget and program considerations, and b) have narrower scope and to envision more specific solutions.

A potential advantage in setting the far-term epoch at 25 years includes enabling the ADT to engage in a broad range of candid, impartial architecture designs relatively free of near-term budgetary and programmatic considerations. A potential risk is the epoch may be viewed as too far into the future to enable credible planning given the changes that may occur in the environment (e.g., domestic, international, threat) and in technology within 25 years. Reciprocal considerations apply for setting the far-term epoch at 15 years, where a potential advantage is results are more likely to be relevant for near-term budget and programmatic considerations. A potential risk is results may reflect only a relatively minor scaling of present capabilities and may not consider new technologies, doctrine, and policies that enable major capability enhancements.

Selection of a 25 year far-term epoch, as in Fig. 6, tends to be more consistent with the role of MAA studies, whereas selection of 15 years tends to be more consistent with mission architecture studies. Figure 6 allows that a typical far-term (“should-be”) epoch might be set between 20-25 years for MAAs. Recognizing the environment and technology will evolve over a 20-25 year period, MAA results should be periodically revisited to verify they remain relevant.

Design Cycle Process for MAA Studies

The design cycle process is a structured approach based on a standard SE technique for conducting system requirements development, design, and analysis in successive iterations. It applies well to MAA developments, which are inherently exploratory and uncertain. Other process models, such as waterfall or ad-hoc iterative, exist for conducting design efforts, but they are less effective for tasks that are developmental or that have high uncertainty. A waterfall (i.e., linear, unidirectional) process is more effective for tasks that are well understood, and an ad-hoc iterative process can be difficult to keep synchronized, particularly with large or distributed teams with members from multiple stakeholder organizations.

The design cycle process helps maintain synchronization of assumptions, trades and analyses on uncertain efforts by bringing products to a common, coherent reference point in each cycle while providing discrete opportunities for coherent stakeholder and management review. It accelerates team learning by starting architecture design earlier than possible when using a process with only a single design phase.[6] Doing this helps surface “unknown unknowns” early, where unknown unknowns often emerge only as a byproduct of design work and cannot be planned for in advance. Accelerating team learning also enables needed changes to be made early in the study, when there is still time available to evaluate, and benefit from, the effects of those changes.

Further, developing products during each cycle facilitates systems level integration and enables coherent waypoints for ADT reference in future cycles. It also documents ADT results, methods, findings, and rationale while they are current and unaltered by subsequent discussions and decisions. This avoids the need to reconstruct work after an extended study, improves the final report, and reduces the effort required to produce the final report. Having a comprehensive ADT report also will accelerate the work of future ADTs that revisit the MAA.

A general MAA study plan, such as in Fig. 7, with major activities and milestones should be outlined early along with the details of the first design cycle. The details of subsequent design cycles can be planned later within general plan constraints. However, having a standard schedule template for design cycle tasks is recommended at the outset of the study. Having a standard template allows design cycles to be moved and tailored, to a minor extent, relatively easily within general plan constraints. A template also provides a repeatable work flow that becomes increasingly familiar as the team progresses. Conversely, due to the high degree of uncertainty associated with MAA developments, it is neither practical nor recommended to develop a detailed, line item by line item, schedule for the entire study at the outset. The overhead in maintaining and controlling such a schedule can lead to valuable time lost.

[6] Using a single design phase, such as done with a waterfall approach, would tend to require more planning prior to starting design as the ADT would need to develop a successful MAA on its first, and only, attempt at MAA design.

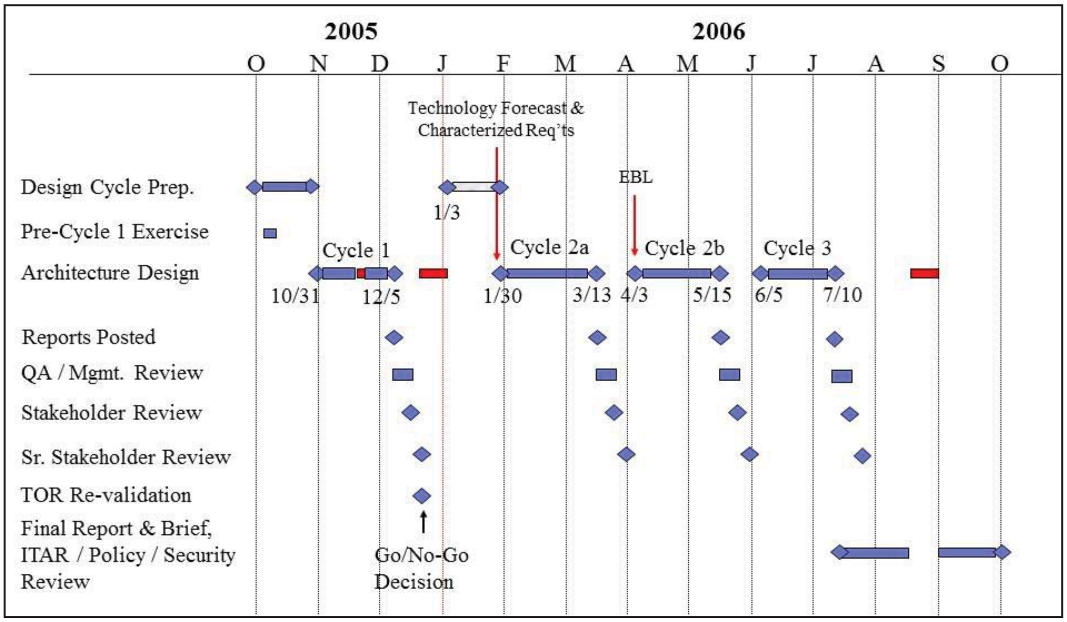


Figure 7: Example 12-Month MAA Study Design Cycle Template

Design Cycle Approach Overview. MAA development is conducted in three cycles as shown in Fig. 7 for a CY 2005/2006 example with "pre-design" products available in advance as discussed below. The second cycle is conducted in two parts. ADTs objectively and impartially investigate the full trade space in each cycle, rather than using each cycle to investigate separate parts of the trade space. Each cycle is followed by a set of reviews, including internal quality assurance (QA) and management reviews and external reviews with stakeholders, senior stakeholders, and customers. Feedback from those reviews guides subsequent design cycles. Rather than retrofitting the results of current cycles into the results of past cycles, results and learning are just applied to subsequent cycles. Red shading in Fig. 7 denotes periods of no planned work.

Cycle 1. Cycle 1 is conducted as a pathfinder cycle for first-time MAAs. Its purpose is to help the ADT learn and assess readiness for MAA design. During Cycle 1, the ADT learns whether the MAA: a) "requirements"[7] are characterized in a form usable for analysis, b) metrics are compatible with the MAA modeling tools, c) modeling tools can analyze the design to provide the MAA desired product set, and d) desired product set suffices to answer the problem statement in the TOR. During Cycle 1, ADTs analyze a limited number of architectures that span the trade space to exercise thought paths needed in future cycles. Surrogates (i.e., estimates) may be used for the "requirements" and the technology forecast if the final set of "requirements" and the technology forecast are not yet available. As the ADT learns at a high rate in Cycle 1, the plan in Fig. 7 allows time between Cycles 1 and 2a to affect any needed changes resulting from that learning. Also, the end of Cycle 1 presents an opportunity to review

[7] In this paper, the term, "requirements", when in quotes is used in an informal context to reflect classes of user needs used to guide evaluation of architectures within the trade space. This differs substantially from the context of requirements in project management wherein system level requirements are not formally baselined until the System Requirements Review near the end of Phase A for a specific, final mission concept design that meets technical and programmatic (including cost and schedule) constraints.

and re-validate the TOR (e.g., to confirm the problem is well posed and the scope is realistic) and to confirm the study should be continued. With a concerted effort, Cycle 1 can provide high value learning essential to the success of subsequent cycles. However, it should be kept brief, and results should not be used for budgetary or programmatic planning.

Cycles 2a & 2b. These cycles are used to conduct comprehensive investigations for a broad range of candidate architectures and to determine the most promising architectures across the trade space. Cycles 2a and 2b typically address the same scope. However, Cycle 2b leverages the learning from Cycle 2a to repackage architecture candidates for improved effectiveness. The capability of the EBL is evaluated in Cycle 2b as well.

Cycle 3. Cycle 3 is used to refine the analysis on the few most promising representative architectures of the trade space and to recommend a single architecture based on the criteria in the TOR.

Pre-Design Products. Pre-Design products are draft products developed before Cycle 1 to help accelerate the start of Cycle 1. They include: 1) the functional decomposition (through performance metrics), 2) generic scalable physical nodes prepared for modeling use, including governing equations and relationships, 3) generic "threads" (described further below), 4) the types of modeling tools available to analyze the nodes, 5) the technology forecast, to the degree readily available in plans and roadmaps, 6) a summary of known mission area guidance and relevant studies, and 7) MOEs previously used or identified for the mission area. They also may include data collection templates that support development of the technology forecast and the "as-is", "to-be" (planned), and EBL architectures. This assumes user needs will be supplied by stakeholders without need for a separate data collection template.

The need for data collection from stakeholders, particularly for construction of the "as-is", "to-be" (planned), and EBL architectures, should be evaluated due to stakeholder and ADT time and effort required. Data collected should be limited to essential information. The ADT should understand generally how the data collected will be reduced and used. This can be challenging for first-time MAA studies, as what data will be needed and how the data will be used may not become substantially evident until the completion of Cycle 1.

Architecture Trade Case Matrix. Table 2 leverages the format of Table 1 to show an example architecture trade case matrix for the Space Access mission area. The analysis of individual nodes combines to determine the performance and effectiveness of each thread. Threads contain all the nodes needed to deliver an end-to-end service. For example, "deliver" payload to orbit includes nodes for: launch base, ground station, range, launch vehicle, and human C2 assets and entities. Analyses of individual threads combine to determine the performance and effectiveness of the MAA. ADTs assign combinations of threads to a range of candidate MAAs. Following selection of the final MAA solution after Cycle 3, the functional decomposition for that solution may be transferred into the NCS architecture functional decomposition in the format of Table 1.
Details on the conduct of each design cycle and on trade space evaluation methods are beyond the scope of this paper. Techniques for multi-attribute trade space exploration, utility analysis, and value analysis are discussed in other work, e.g., ref. (m).

Table 2 Architecture Trade Case Matrix (Space Access Example for Epoch = 20xx)

Architecture Solution (How's) => Functions / MOPs (What's)	1a	1b	1c	2a	2b	2c	3a	3b	3c	4a	4b	4c	5a	5b	5c	6a	6b	6c
Provide Space Access Capabilities	**All ELV**			**Mix ELV / RLV**			**All RLV w/Tugs**											
Provide Spacelift / Payload Transportation Capabilities																		
- Deliver																		
- Quality										**Each "architecture" is a composite of several "threads" designed to meet MOPs** **Architecture #1** represents an all ELV solution where threads 1a, 1b, & 1c might include light, medium, & heavy ELVs, respectively. **Architecture #3** represents an all RLV solution with tugs, where threads 3a, 3b, & 3c might include light RLVs, medium RLVs, & medium tugs, respectively.								
- Quantity																		
- Timeliness																		
- Interoperability																		
- Robustness																		
- Deploy (QQTIR as above)																		
- Retrieve (QQTIR as above)																		
- Return (QQTIR as above)																		
Provide Range / Launch Base Capabilties										Expand as done for Spacelift / Payload Transportation								
Provide On-Orbit Servicing / Utilities Capabilities										Expand as done for Spacelift / Payload Transportation								

Figure 8 illustrates how classes of MOPs might appear in a trade space. Point B, for example, depicts the need for light "Deliver" and "Deploy" capability, medium "Return" capability, and medium "On-Orbit Servicing" capability. These needs would be mapped to specific nodes within threads. When stakeholder needs group closely together in the trade space, they may be able to be evaluated effectively as a single need to reduce the required number of analysis cases.

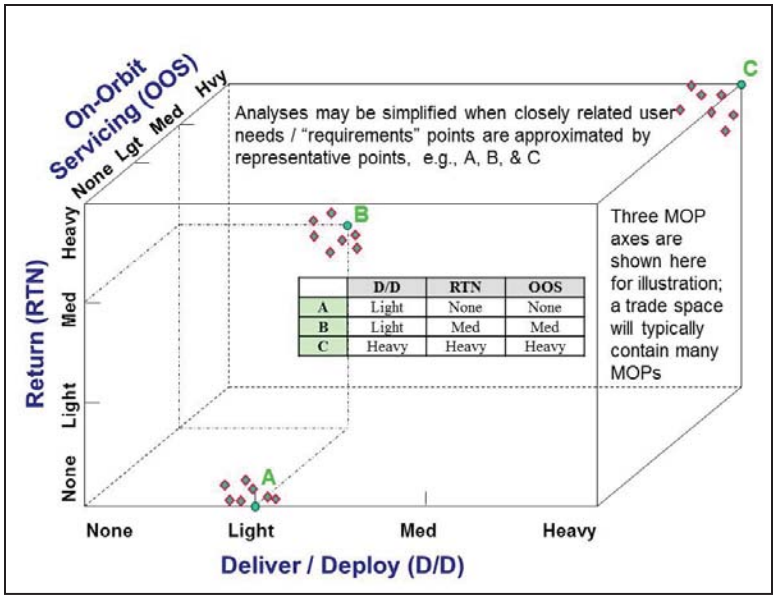

On-Orbit Servicing (OOS)
None
Lgt
Med
Hvy
Analyses may be simplified when closely related user needs / "requirements" points are approximated by representative points, e.g., A, B, & C
C
B
A
Three MOP axes are shown here for illustration; a trade space will typically contain many MOPs
Return (RTN)
None
Light
Med
Heavy
D/D
RTN
OOS
A
Light
None
None
B
Light
Med
Med
C
Heavy
Heavy
Heavy
None
Light
Med
Heavy
Deliver / Deploy (D/D)

Figure 8: Example Space Access "Requirements" Trade Space

Typical MAA Design Cycle Products. Table 3 shows typical design cycle products (reports) developed by an ADT organized using three primary subteams (operations, systems, analysis) that functionally report to an architecture systems engineer (ASE) who functionally reports to the ADT lead. All products are delivered at the end of each cycle with some exceptions. The user needs / "requirements" and technology forecast are delivered no later than the start of Cycle 2a, and the EBL is delivered no later than the start of Cycle 2b along with the "as-is" and "to-be"

Table 3: Typical MAA Design Cycle Products[8]

No	Product	Cycles	Subteam
1	Functional Decomposition (incl. MOPs / Interface "Requirements")	1, 2a, 2b, 3	Systems
2	User Needs / "Requirements" Classes & Bounding Cases	2a, 2b, 3	Operations
3	Tradespace & Trade Case Matrix	1, 2a, 2b, 3	Systems
4	"as-is", "to-be" (planned), & EBL architectures	2b, 3	Systems
5	Technology Forecast	2a, 2b, 3	Systems
6	Architecture Alternative Point Designs	1, 2a, 2b, 3	Systems
7	Scenarios	1, 2a, 2b, 3	Operations
8	Future Environments / Threat Assessment	1, 2a, 2b, 3	Operations
9	MOEs	1, 2a, 2b, 3	Analysis
10	Performance / Utility Analyses	1, 2a, 2b, 3	Analysis
11	CONOPS	1, 2a, 2b, 3	Operations
12	Vulnerability Assessment	1, 2a, 2b, 3	Analysis
13	Doctrine / Policy Assessment	1, 2a, 2b, 3	Operations
14	Work Breakdown Structure	1, 2a, 2b, 3	Analysis
15	Cost Analysis	1, 2a, 2b, 3	Analysis
16	Risk Assessment	1, 2a, 2b, 3	ASE
17	Subteam Technical Reports	1, 2a, 2b, 3	All Subteams
18	Systems Engineer Report	1, 2a, 2b, 3	ASE

(planned) architectures. Each of these products typically is assembled by the ADT after receiving stakeholder responses to data collection requests. Surrogates may be used for each of these products in prior cycles.

ADT reports, begun in Cycle 1 and refined in later cycles, provide the official study record of what team did, how the team did it, and what the team found for present (and future) team use. Their technical descriptions, including analysis methods and example calculations, enable effective technical integration across subteams, system level review and integration, and independent and external reviews. Briefings, often too cursory to be used effectively for the same purposes, are developed exclusively from the approved reports. ADT reports also provide the foundation to engage stakeholders and customers with coherent products at the end of each cycle. They position stakeholders and customers to effectively steer the study and to take incremental ownership of the product enroute to selection of the final recommended architecture.

Closing Thoughts

This paper describes a fundamental systems engineering approach that can help ADTs, their customers, and their stakeholders quickly gain a collective understanding of the core elements of an enterprise architecture and visualize how constituent MAAs are integrated into an enterprise architecture when planning for the far-term future. It also illustrates how an MAA might get developed for space systems without need for specialized training in the use of more complex and abstract methods. As the methods used in this paper are based on widely understood systems engineering techniques and terminology, they should be readily usable by a wide range of teams within government and industry and should have application beyond space architecture development.

[8] Product list is adapted from the National Security Space Architect (NSSA) Architecting Guide, October 2000

References

a) iso-architecture.org. 2014. “Survey of Architecture Frameworks.” Last accessed 3 September 2014. http://www.iso-architecture.org/ieee-1471/afs/frameworks-table.html.

b) Di Pietro, David A. 2009. “National Security Space (NSS) Architecture Development Process (Draft).” Briefing to Mr. J. Rouge, Director National Security Space Office, Washington, DC (US). Presented and approved 9 February.

c) Di Pietro, David A. 2009. “Space Mission Architecture Development, Notional Framework and Approach.” Training course presented at the NASA Goddard Space Flight Center, Greenbelt, MD (US). 30 September.

d) Di Pietro, David A. 2014. “A Systems Engineering Approach to Architecture Development.” Briefing presented (http://www.dtic.mil/ndia/2014system/2014system.html) at the National Defense Industry Association 17th Annual Systems Engineering Conference, Springfield, VA (US). October 2014.

e) NASA Headquarters. 2012. NASA Procedural Requirements (NPR) 7120.5E w/Changes 1-10. *NASA Space Flight Program and Project Management Requirements*, Figure 2-5. Washington, DC (US): Office of the Chief Engineer.

f) Razzaghi, Andrea I., Di Pietro, David A., Simon-Miller, Amy A., and Spencer, John R. 2007. *Enceladus Flagship Mission Concept Study*, Revision 1, Edited for Public Release. NASA Goddard Space Flight Center, Greenbelt, MD (US). 2 November.

g) Razzaghi, Andrea I., Di Pietro, David A., Quinn, David A., Simon-Miller, Amy A., and Tompkins, Stephen D. 2008. “Mission Concepts for Studying Enceladus.” Paper presented at Space Technology & Applications International Forum, Albuquerque, NM (US). 14 February.

h) G. & C. Merriam Co. 1975. *Webster’s New Collegiate Dictionary.* Springfield, MA (US).

i) Wikipedia. 2014. “Propellant_depot.” Last accessed 5 October 2014. http://en.wikipedia.org/wiki/Propellant_depot.

j) Wikipedia. 2014. “Space_tug.” Last accessed 5 October 2014. http://en.wikipedia.org/wiki/Space_tug.

k) Brown, Charles D. 1998. *Spacecraft Mission Design.* 2nd Edition. Equation 3.21. Reston, VA (US): American Institute of Aeronautics and Astronautics.

l) Hagemeier, Hal E. 2009. Deputy Director, National Security Space Office. Washington, DC (US).

m) Rader, Andrew A., Ross, Adam M., and Fitzgerald, Matthew E. 2014. "Multi-Epoch Analysis of a Satellite Constellation to Identify Value Robust Deployment across Uncertain Futures." Paper presented (http://seari.mit.edu/publications.php#2015) at AIAA Space 2014, San Diego, CA, August 2014.

Biography

Mr. Di Pietro is the Senior Engineer for Advanced Concepts and Architectures at the NASA Goddard Space Flight Center where he has led, or served as mission systems engineer on, over 20 advanced space mission designs. He has served in interagency assignments with the Defense Department in space architecture development and with the State Department in defense trade controls. His prior assignments include Delta II Launch Service Manager (NASA), Space Station Freedom Thermal Control System Manager (NASA), and aircraft advanced design engineer (Naval Air Systems Command). He has M.S. and B.S. degrees in Aerospace Engineering and is a retired Colonel, US Air Force Reserve.