THE AEROSPACE CORPORATION

Enterprise Development Strategy for Satellite Systems – How Important is the Ground Operations System?

Joseph Betser, Mary Rich, Sergio Alvarado, Phillip Schmidt, Jaime Milstein

Joe.Betser@aero.org
The Aerospace Corporation,
2350 East El Segundo Boulevard, M1-112
El Segundo, California 90245-4691, USA
+1-310-336-0577 +1-310-563-1178 fax

IFIP/IEEE Int'l Symposium On Integrated Management – IM 2005
Nice, France, 16-20 May, 2005

Keywords: Satellite Enterprises, Operations, Business Process Management, COTS, Customer Care, Requirements Analysis

Abstract

Space satellite enterprises are complex systems that often involve the application of leading-edge technologies in order to achieve the requirements and best performance feasible. A space program can take as long as decades from initial concept to full system deployment. Considerable technical and programmatic risks must be properly managed during the program lifetime in order to achieve success. This paper discusses these engineering challenges with a particular emphasis on the ground enterprise (segment) that operates and performs the mission of the satellite system (space segment). We pay particular attention to the software development challenges for the ground segment. These development projects engage sizable contractor teams that can acquire and build the ground system, the space segment, or both. This paper discusses future acquisition options based on experience gained from lessons learned on a number of space programs. The key finding of this paper is that, in our experience, the ground system carries the highest software development risks. These risks must be evaluated early on in the life of a program, since their accurate assessment must drive the acquisition strategy. A number of acquisition options must be considered and carefully evaluated. Decisions made in the embryonic phase of early program concept definition, have significant effects on the program later on. Appropriate acquisition strategy, architectural philosophy, and design tools are necessary in order to launch a successful program, as well as during proper follow through during the lifetime of the program.

0-7803-9087-3/05/$20.00 (C) 2005 IEEE

This paper begins with an overview of the challenges at hand. We provide a high level description of the nature of the ground segment versus the space segment. We describe the technical differences, as well as the "cultural" programmatic reasons that traditionally resulted in different treatment to these segment, ultimately to the determent of the ground systems. This causes the ground to not receive sufficient attention in order to manage the higher software engineering risk associated with the ground segment. After we establish these ground system development risks, which are both technical and programmatic, we look for ways to mitigate these risks. The approach is to increase the early emphasis on the ground system. This is accomplished by exploring ground-centric development and acquisition strategies. This is done by carefully evaluating a number of options to perform the development, and structure the acquisition teams as well as the associated contracting arrangements. In addition, we highlight the benefits of spiral, or evolutionary acquisitions as tools to reduce the software development risks for ground systems. We conclude with a number of recommendations that summarize the risk reduction strategy for ground systems for satellite systems.

The satellite mission and space vehicles (satellites) are the focus for most space enthusiasts. These space vehicles have the most demanding leading-edge requirements in terms of communication technology, sensor technology, launch technology, material technology, orbit management technology, and so on. These are the key drivers of technology readiness assessments. Failures during launch and early orbit are what gets media attention. At the same time, the hardware technical disciplines are quite mature, and the community is able to manage these risks relatively well.

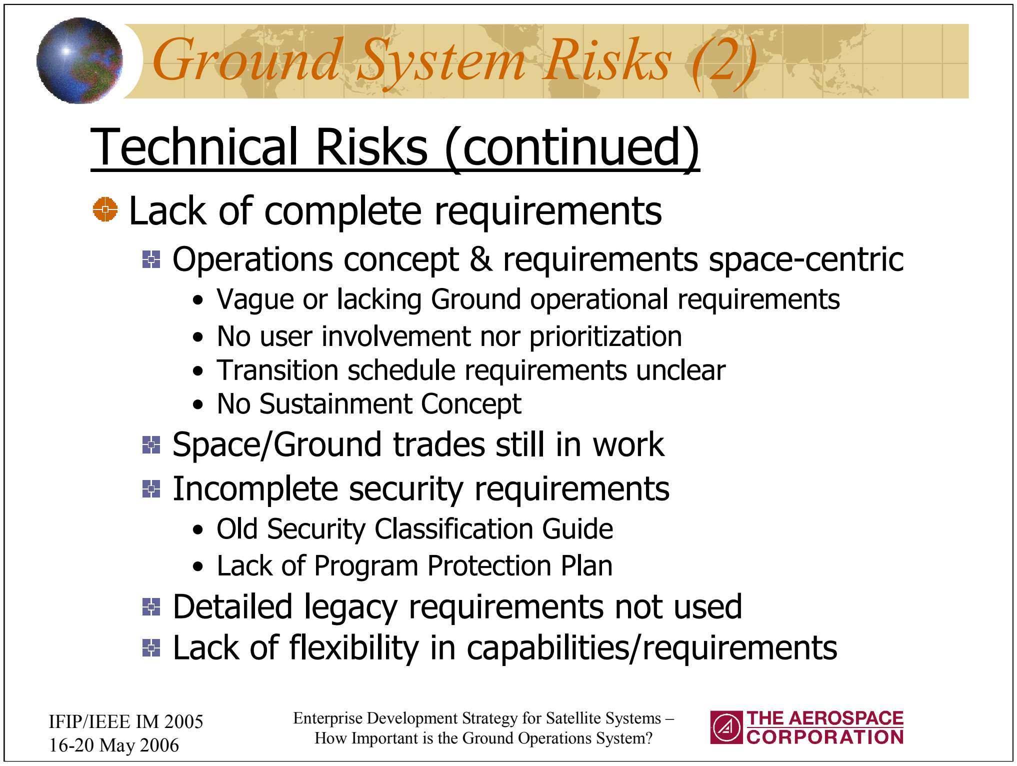
On the other hand, the ground system appears less exciting. It involves Commercial Off-The Shelf (COTS) enterprises with operations, control, and management software that is used to manage the space constellation as well as to manage the mission and the ground enterprise. This "support" role of the ground systems appears secondary to the space assets, and hence less attention has been paid to it traditionally. What we find from our experience in multiple programs, is that the ground segment often causes program challenges including schedule slips and cost overruns that threatened the success of the mission. A number of reasons account for this phenomenon. Often times requirements "leak" from the space segment to the ground segment. This causes mid-stream new requirements to the ground and presents the development with additional challenges. Software is a complex, abstract entity that is difficult to design, build, and maintain. These challenges include size, complex interfaces, legacy software, COTS, and sustainment after deployment. All of these factors increase the Total Ownership Cost (TOC).

Ground System Risks (1)

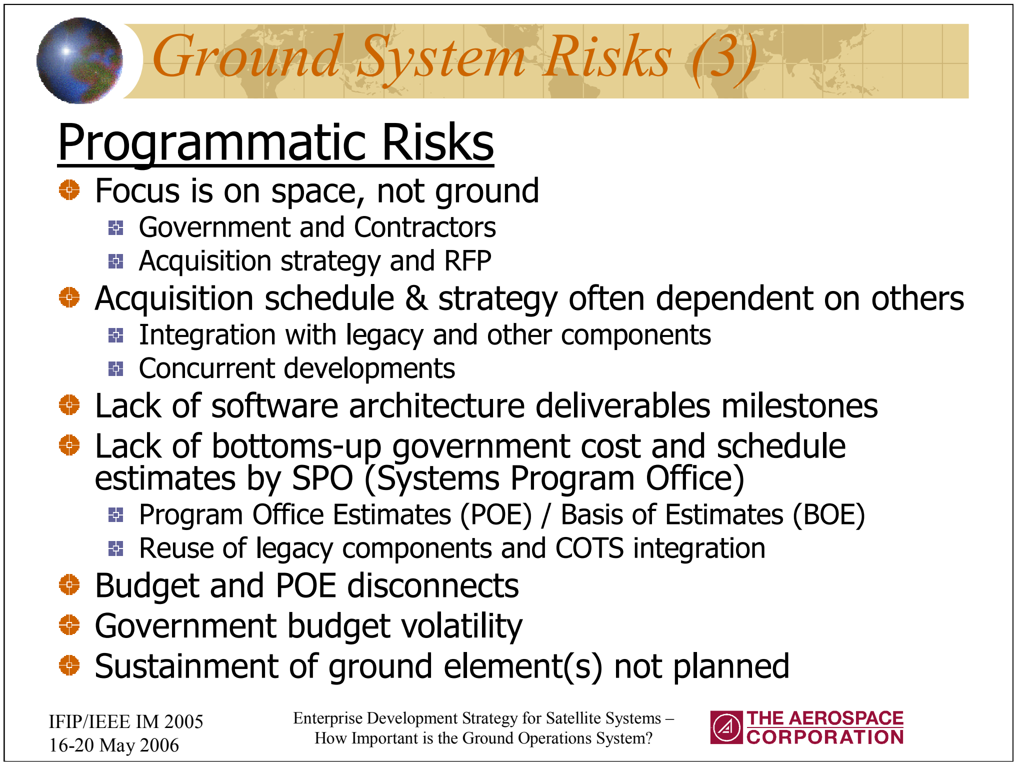
Technical Risks

- Large, complex software intensive systems
- COTS/Legacy use/reuse and integration not adequately planned
- Many and complex interfaces
- Lack of architecture requirements and definition
- Software Engineering & Architecture not part of overall system engineering equation
- Lack of architecture evaluation
  - Source selection and after Contract Award (CA)
  - Products and processes

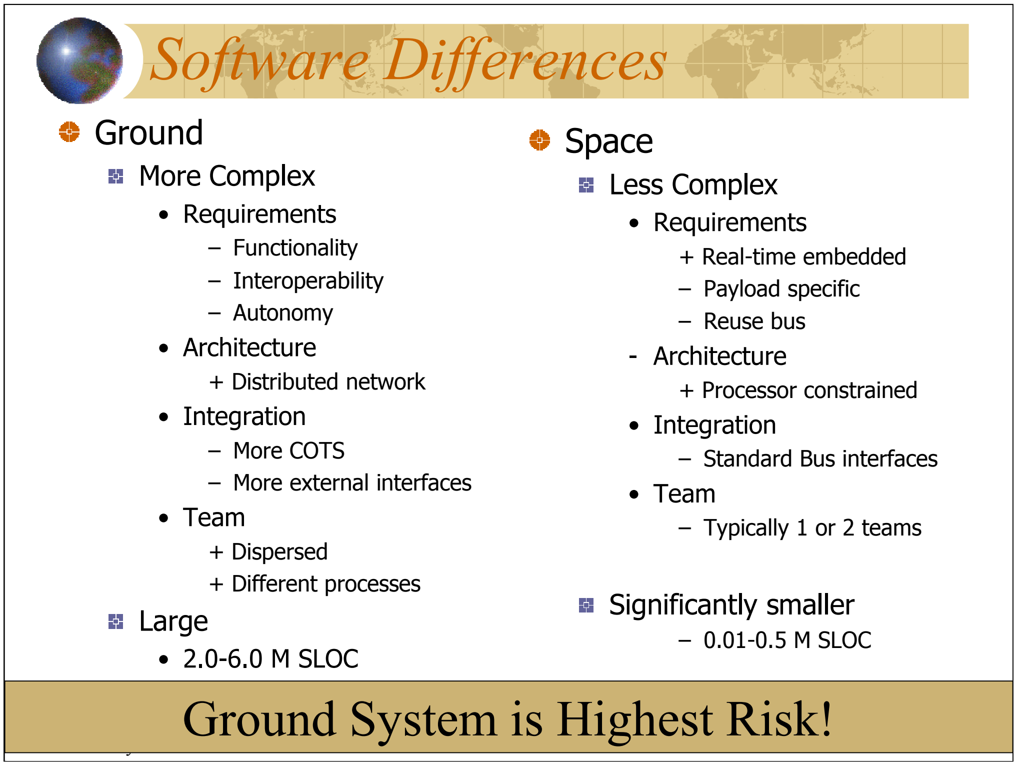
We divide the ground system risks into two classes of risks – Technical Risks and Programmatic Risks. Let us review a number of the common technical risks. The software systems for the ground segment often consist of millions of Source Lines Of Code (SLOC), of what is a complex system consisting of multiple modules and interfaces, including COTS that could be challenging to integrate. Often times the software architecture is poorly defined or entirely lacking. Hence the selection of the winning contractor to perform the work does not include a competent software architecture evaluation. In addition, there might be insufficient oversight of the performed work after contract award (CA), and not enough emphasis and rigor in requiring mature products and processes within the developing organizations.



Additional technical risks include incomplete requirements at the time of initial acquisition and early development. Again, the operations concept is space-centric, hence it lacks specificity in ground system requirements. End-users are often not consulted early on, hence the ground requirements are not well defined. The schedule for the transition is not thought out in advance, and the resulting requirements often fall short. The development often does not take into account the sustainment of the system once it has been deployed. In addition, often times development commences when trades between space and ground are incomplete, and requirements have not been allocated to the respective segments. Security requirements are very seldom complete at the time of initial acquisition and early development. This leads to a number of post-development design concepts, additional integration challenges, schedule delays, and budget overruns. The incorporation of legacy systems and technology almost always presents an integration challenge. Requirements for the legacy systems do not always exist in detail, and the integration with newer technologies is always a challenge, since the newer technologies did not exist at the time of original development of the legacy system, hence the difficulty in creating a working interface. Finally, it is costly and difficult to revisit and modify capabilities and requirements mid-stream, particularly in the event that solid software architecture and engineering principles had not been engaged in the original design.



The programmatic risks are very important to understand, as they are dominant drivers for any space program. Space programs are very costly. A mid-size program costs billions of dollars, and can span over a number of years. Hence, securing appropriate funding for a program is a challenging task, with multiple administrative, legal, and political constraints. Many of the funding challenges continue to haunt a program through its lifetime, thus exacerbating the multiple technical risks just outlined. Let us detail some of these programmatic risks. The acquisition is often focused on the space segments and not enough attention is paid to the challenging ground segment. Many dependencies exist on projects, technologies, and entities that are not controlled by the specific space program. Budget and schedule estimates are not always realistic. Often times they are driven by funding volatility. Sometimes space programs are optimistically funded in order to encourage a timely start and early consideration of all program requirements. Later on in the lifetime of the program, new budget requirements need to be managed. This scenario might be associated with some schedule slips that might further exacerbate the program challenges and risks. In addition, proper accounting for reuse of legacy and incorporation of COTS modules is at times optimistic. Lastly, the sustaiment is not considered part of the development, and at times might become prohibitively expensive because it did not receive the appropriate attention during the design and during the contractor evaluation and selection.



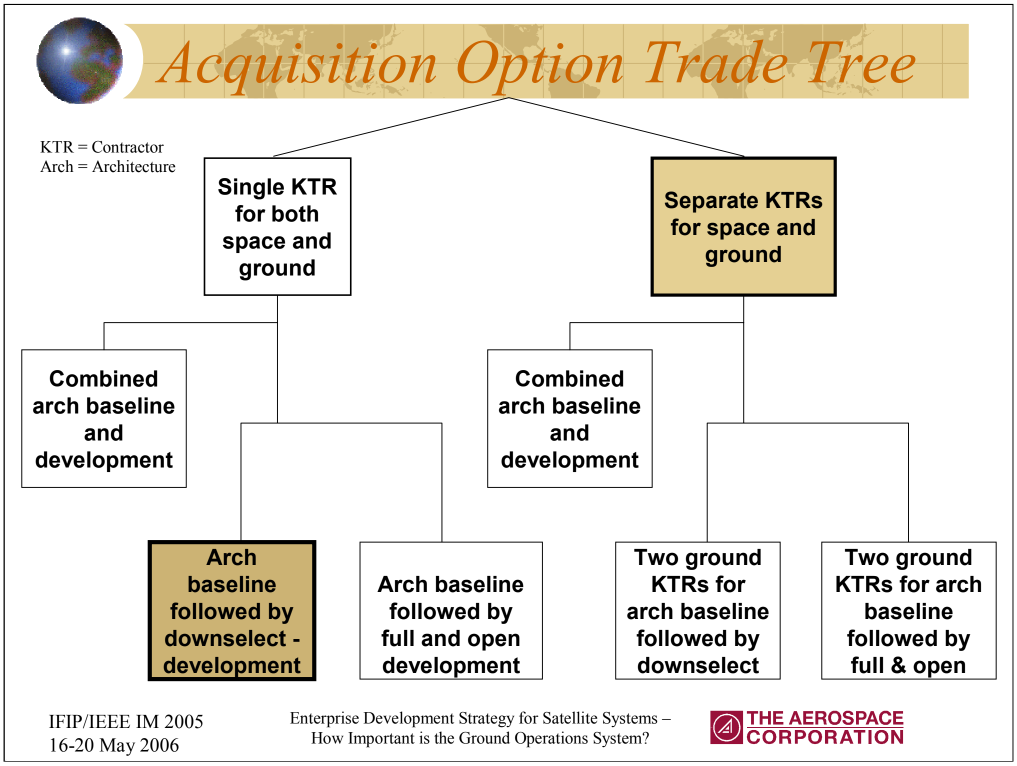
Let us compare the software elements of the space segment (satellite), versus those of the ground system. The ground software system is more complex than that of the space segment. It typically consists of a heterogeneous set of networked workstations with a distributed architecture. It interfaces with multiple users and mission specialists. It could be geographically dispersed and in most circumstances includes COTS and legacy components. Lastly, it is large, in that its size is measured in millions of Software Lines of Code (SLOC). In contrast, the space segment is much less than a million SLOC, it typically is specific to a single satellite type, and consists of a real-time embedded system that is constrained by processor speed and power budget. The development team for the satellite payload and bus software is typically smaller and more tightly-knit than that of the ground software.

Hence clearly, the ground presents a higher risk, and should be tackled with care, planning, and an architecture-centric approach Such an approach is described in [Schm2003]. The reader is referred to that paper for more detail on UML architectural modeling for architecture-centric approaches to evolutionary space program design and development.

0-7803-9087-3/05/$20.00 (C) 2005 IEEE

Let us examine how we could help manage these enterprise risks for a space program of moderate complexity. Let us assume that some legacy exists with which the new program would need to interface. Let us examine options to structure the acquisition and development, and review the advantages (Pros) and disadvantages (Cons) of each approach. We look at options from a single contract from start to finish for ground and space, to several contracts dividing pieces of the work among different contractors along different segments, as well as into separate initial contracts for study and development, with various combinations of down-select from competing teams to conduct the actual development of the space program.

The balance of this paper reviews such approaches. The diagram in the following page depicts the above option in a decision-tree (Option Trade Tree) structure.



The above Option Trade-Tree depicts some of the various combination threads of the acquisition development strategies outlined above.

For the purpose of a focused discussion, we will concentrate on the 2 shaded boxes in the option trade tree.

We use the next several charts to assess some of the major Pros and the Cons for each approach to help us devise an integrated acquisition strategy.

Two parallel contracts: one for ground & one for space and integration

- PROS
  - Use contractors respective strengths
  - Increased competition
  - Reduced ground risks: awards to most capable contractor
  - Incentivizes space/ground contractors appropriately
  - Model successfully used by commercial space
- CONS
  - Increased SPO requirements
    - PM & integration roles
    - Ground/space allocation
  - Increased potential for space/ground contractor misalignment
    - RFP (Request for Proposals) / contracts need to be well defined and flexible
      - Ensure cooperation in Interface definition, design, test, database calibration, etc.
    - High potential for ECP (Engineering Change Proposal) to ground after contract award
  - Reduced cost/schedule confidence
  - Space Contractor may lack integration capacity

Pros: Parallel contracts increase competition and reduce ground risks, since most capable ground contractor does the ground.

Cons: Greater difficulty in managing interfaces among segments and potential for misalignment between ground segment and space segment. These might turn prohibitively expensive in the long run. The lack of early requirements definition and integrated trade studies have the potential to cause long-lasting program difficulties.

0-7803-9087-3/05/$20.00 (C) 2005 IEEE

Two full contracts – first phase for requirements and architecture definition, then downselect

- PROS
  - Design before you buy
  - Prolonged competitive phase will result in contractors 'best' ideas
  - Requirements on contract
    - High cost/schedule confidence
  - Reduced cost risk
  - Better ground space trades
- CONS
  - 2 Source Selections (1 mini)
  - ~1 year added to schedule
  - Added SPO resources
  - Best value determination process needs to be clearly defined during competition
  - Additional cost for second phase 1 contractor

IFIP/IEEE IM 2005
16-20 May 2006

Enterprise Development Strategy for Satellite Systems – How Important is the Ground Operations System?

THE AEROSPACE CORPORATION

Pros: Compete the study phase that includes architecture definition and down select the most successful competitor team. This helps in having good requirements on the development contract, and a solid architecture to boot. In addition, better space-ground trades are conducted prior to design and development.

Cons: This comes at a cost of initial investment in budget and schedule, as well as in System Program Office (SPO) resources. However, the results of this initial investment can be very significant, as program success hinges on better embryonic design, using solid architecture and making good selection of contractor team to insure program success. This also requires more SPO involvement during the initial phase. The commitment of the customer SPO must be commensurate with the requirements of effective management prior-to and after the down-selection of the winning contractor team.

0-7803-9087-3/05/$20.00 (C) 2005 IEEE

Recognize all risks, and particularly the high risk driver coming from the ground system. The entire team, government, users, and contractors must recognize this. Study acquisition options thoroughly with a Pro/Con analysis. Always implement lessons learned from extensive experience in space system acquisition. Be reasonable, realistic, and flexible with cost, schedule, and system baselines. Must put great emphasis on capable software contractor team in team selection. MUST USE ARCHITECTURE-CENTRIC REQUIREMENTS AND PROCUREMENT THROUGHOUT DESIGN AND DEVELOPMENT [Schm2003]. Risk management continues throughout the life of the program!

Additional details and approaches are available at the GSAW web site above.

References:

[Schm2003] P. Schmidt, S. Alvarado, J. Rivera, J. Milstein, “Architectural-Centric Representation for Design Diversity and Program Evolution”, Ground System Architectures Workshop (GSAW 2003), 4-6 March 2003, Manhattan Beach, CA.

[Rich2003] M. A. Rich, S. Lazar, J. Betser, “Space Acquisition Strategy – Just How Important is the Ground Segment?”, Ground Systems Architecture Workshop (GSAW 2003), 4-6 March 2003, Manhattan Beach, CA.