NASA's Space Launch System: Systems Engineering Approach for Affordability and Mission Success

Dr. John J. Hutt[1]
NASA Marshall Space Flight Center, Huntsville, AL, 35811
Dr. Josh Whitehead[2]
NASA Marshall Space Flight Center, Huntsville, AL, 35811
Dr. John Hanson[3]
NASA Marshall Space Flight Center, Huntsville, AL, 35811

NASA is working toward the first launch of the Space Launch System, a new, unmatched capability for deep space exploration with launch readiness planned for 2019. Since program start in 2011, SLS has passed several major formal design milestones, and every major element of the vehicle has produced test and flight hardware. The SLS approach to systems engineering has been key to the program's success. Key aspects of the SLS SE&I approach include: 1) minimizing the number of requirements, 2) elimination of explicit verification requirements, 3) use of certified models of subsystem capability in lieu of requirements when appropriate and 4) certification of capability beyond minimum required capability.

I. Introduction

NASA is currently developing and building the Space Launch System (SLS), a new, unmatched launch capability for deep space human and robotic exploration, shown in Figure 1. SLS is designed to evolve from an initial capability of at least 70 metric tons (mt) to low Earth orbit (LEO) – greater than any current launch vehicle – to a capability of at least 130mt to LEO, greater than any vehicle in history. NASA is developing that unprecedented capability in a constrained budget environment and under a relatively flat funding profile by employing existing subsystem designs and hardware. NASA selected a super-heavy-lift design based on the powerful and proven space shuttle RS-25 liquid propellant engines and solid rocket boosters (SRBs), together with a new core stage design. The Program also faces several unique challenges. SLS design, manufacturing and operations involves complex interactions of systems developed at three primary NASA field centers, with support from several other NASA field centers. Each of these centers manage prime contracts in conjunction with in-house activities that involve civil servants and support contracts.

SLS, due to its high energy demands and mass limitations, has high system sensitivity to subsystem interactions. For example, a change in a structural component to address low margins on structural failure can cause the vehicle structural dynamic response to interfere with the vehicle control system designed by another organization. These interactions could lead to late software and/or hardware redesigns or lead to system failure if the change is not addressed in the final design. Further, the use of existing subsystem designs can lead to unexpected problems due to changes in the level of interactions with the new system. Systems Engineering is critical to timely and cost-effective SLS development. The SLS Program (SLSP) has established a Systems Engineering and Integration (SE&I) Operating Model to effectively engineer the system through development, certification, and operations. Adapting general systems engineering principles, SLS SE&I has made significant contributions to the program's success. This paper will discuss key features of the SLS SE&I process.

---

[1] SLS Lead Systems Engineer, NASA, Marshall Space Flight Center.
[2] SLS System Verification and Validation Lead, NASA, Marshall Space Flight Center.
[3] SLS Alternate Lead Systems Engineer, NASA, Marshall Space Flight Center, AIAA Associate Fellow.

American Institute of Aeronautics and Astronautics

Figure 1. Artist rendering of SLS rollout from Vehicle Assembly Building, Kennedy Space Center.

II. The Scope of SLS Systems Engineering and Integration

Due to the breadth of the field and differences in terminology used in the industry, it is important to define the specific scope of SLS SE&I. SLS defines scope for Systems Engineering as a specific discipline within SE&I. This definition does not rigidly conform to commonly used references for systems engineering guidance but rigorously implements the fundamental principles. For SLS, the role of SE&I is all engineering required to:

1) Define the SLS system to meet requirements
2) De-compose the system into hardware and software end items with assigned functionality
3) Manage the technical, cost and schedule interactions of the allocated end items
4) Integrate the end item designs into a certified system design
5) Integrate the end item hardware and software into a flight-certified system
6) Support the operation of the system.

End items are parts of the system that can be assigned to a specific team within SLS for design and development. The key to end item definition is that the end item can be developed with a manageable degree of interaction with the other end items. In the case of SLS, end items are assigned to the Program element offices: Liquid Engines, Boosters, Stages, Spacecraft/Payload Integration and Evolution (SPIE), and SE&I. The following sections will discuss how SLS SE&I performs those roles to achieve both affordability and mission success.

III. Executing SLS SE&I Scope to Meet Objectives

A. Defining the SLS System

As described above, the SE&I process begins with the definition of the system that will meet the requirements allocated to SLS. The primary requirement for SLS is to deliver mass to low Earth orbit (LEO) and to various deep space escape trajectories. Traditionally, this system involves trade studies of launch vehicle configurations to select a configuration for a balanced solution of technical capability, risk, development cost and schedule, and operations cost. SLS performed these trade studies with a focused consideration of the existing hardware, designs and infrastructure from the Space Shuttle Program and other related U.S. space programs.

Generally, traditional trade study processes were applied with the SLS difference being in how risk, cost, and schedule were affected by existing capability. Selection of design features based on existing systems was an initial objective, because the Program determined that the reduced development risk of using existing hardware, such as propulsion subsystems, would outweigh the increased risk that the new vehicle could not as easily be optimized for capability and cost. The trade studies confirmed this, and SLS was selected to begin with existing RS-25 engines from the Space Shuttle Program, the 5-segment SRB from the Shuttle and Ares programs, the Shuttle External Tank (ET)

American Institute of Aeronautics and Astronautics

manufacturing infrastructure, and a modified Delta Cryogenic Second Stage (DCSS) from the existing Delta vehicle program. Figure 2 shows the major components of the SLS Block 1 configuration.
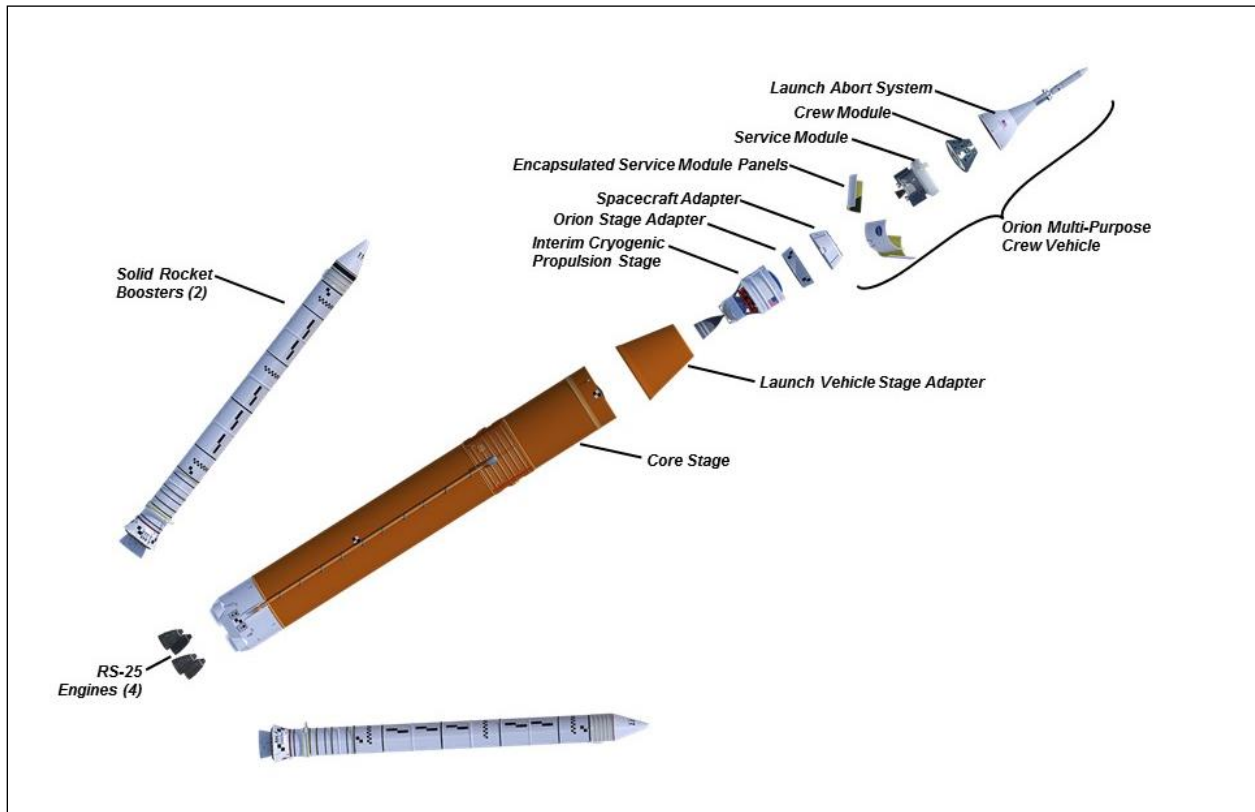

Figure 2. Major components of SLS Block 1 configuration.

B. System Decomposition

SLS processes differ from traditional SE&I processes due to the way SLS implements the decomposition of the system into manageable hardware and software end items. The vehicle configuration trade studies above consisted of assessments of the SLS vehicle's capability to perform the functions necessary to achieve the allocated requirements. Traditionally, the next step would have been to capture this first level of functions as requirements and then begin the same process for each of these requirements. Specifically, more detailed functions necessary to execute these "functional requirements" would be defined. If these more detailed functions were to be performed by the integrated system, they would be assigned as "derived functional requirements." In cases where these sub-functions can be allocated to specific hardware/software within the system, they would be assigned to this system as allocated functional requirements.

The SLS process is a simplified version of this traditional process. For SLS, the process focus is to identify the most direct way to implement the needed function. By doing this, the functional decomposition becomes the foundation for understanding the maturity of the system throughout the lifecycle. In other words, the integrated system maturity is based on an understanding of how well the system can perform its necessary functions at each lifecycle milestone. As shown in Figure 3, there are several ways to more directly implement necessary functions in lieu of assigning them as "derived" or "allocated" functional requirements.
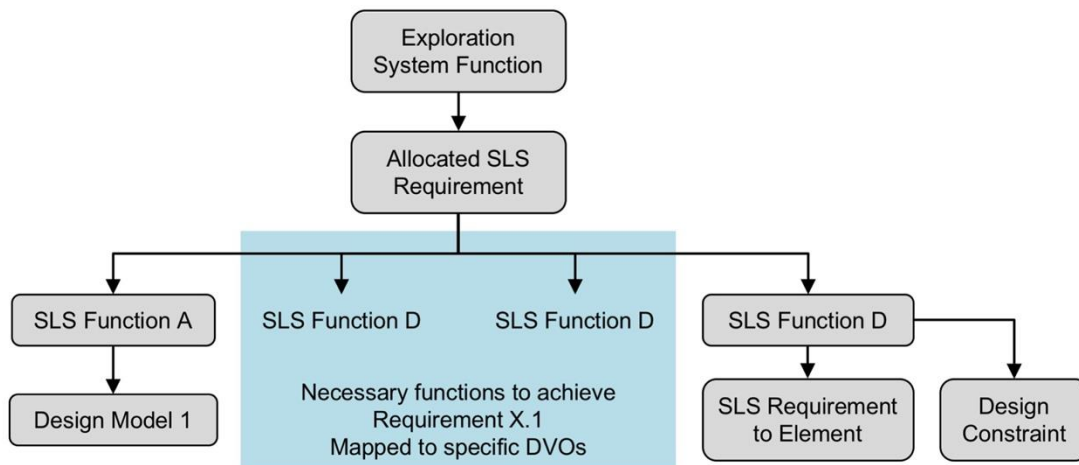
Figure 3. SLS requirements approach to reduce explicit derived requirements while meeting allocated requirements.

These include: 1) modeling the behavior of the system to show it meets the function; 2) direct verification of the function in support of the "parent" requirement through satisfaction of Detailed Verification Objectives (DVOs); 3) assignment of a design constraint that meets the function, often in conjunction with allocating part of the function to an end-item as an allocated requirement. Design constraints include any specific documented design decisions that can be confirmed directly without the need to perform a verification activity. A simple analogy is house design.

There is no need to have a requirement such as, "the house shall have a roof." The roof is simply a design feature (or constraint) that supports the function of protecting the contents from the weather. It can be confirmed to have a roof by simply looking at the blueprints. Extending this analogy, how can the roof be determined to be adequate? This again can be evaluated by details of the blueprint such as pitch and materials selected. None of these parameters would require any formal verification unless some type of non-typical roof is used that may require analysis or even test to confirm it will work. In this case, a specific requirement could be added to confirm the function.

Likewise, SLS system decomposition is implemented as a mix of these three implementations and formally assigned requirements. The advantage of this approach is that it can reduce documentation content without changing the technical work needed or the evidence that the system meets its requirements.

An example of SLS direct functional verification is the job of separating the SRBs from the core stage (CS). SLS has no formal requirement for this separation function. However, the boosters are designed to operate for the first two minutes of the eight-minute ascent to Earth orbit and must be discarded at the end of their burn. Although there is no derived requirement, the delivered mass performance requirement of SLS includes a DVO to verify that the separation function can be performed to allow the vehicle to deliver its mass to space.

The use of modeling and design constraints is best described using a single example because the two are often effectively used together. As mentioned above, in SLS, a design constraint is any part of the technical baseline that is not directly associated with a formal requirement in a "shall" statement, i.e. a requirement such as "the engine shall be capable of X." This includes a wide variety of information, such as design drawings, parameters in interface control documents (ICDs), and information in architecture design description documents. By this definition, models also fall under this description but, due to their unique characteristics, are addressed separately within SLS.

Philosophical debates often arise as to whether design decisions should be captured as requirements in cases where the customer is not only specifying the function to be performed but is directing a specific design to be applied. This is the case within SLS, where heritage subsystems are used to avoid the development cost of a new subsystem. SLS applied an objective approach to this question. For any case where a design decision is applied and it can be confirmed simply by examining configuration records, it is not assigned a requirement. It is simply documented within the technical baseline without a shall statement as simply a statement of fact. This is illustrated for the SLS Thrust Vector Control (TVC) function, as shown in Figure 4.

American Institute of Aeronautics and Astronautics
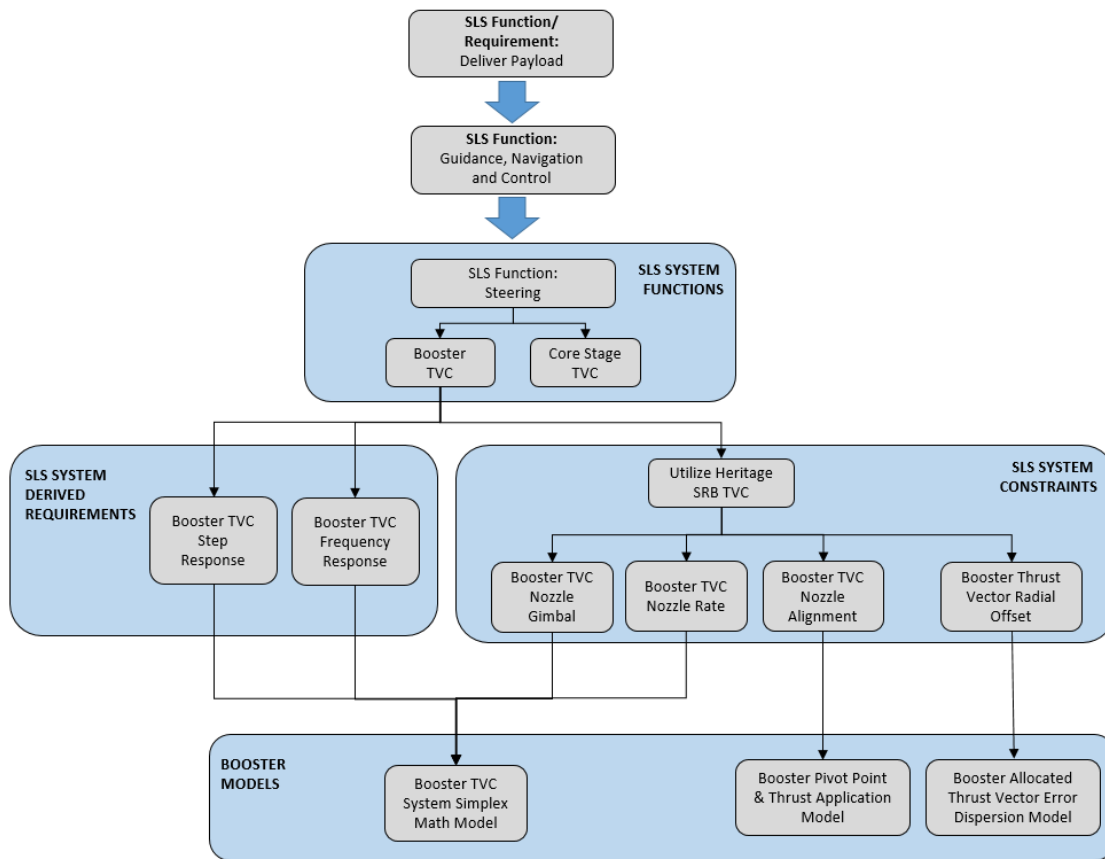
Figure 4. SLS Booster Thrust Vector Control Requirements Approach

The SLSP System Specification states: "Utilize STS (Space Transportation System) heritage thrust vector control (TVC) for boosters". Since the system is now constrained to use the capability of this system, it would be redundant to allocate the required capability of this system to match the design's known capability. Whenever practical, SLS encourages avoiding this redundancy.

However, the capability of the system must be formally captured to provide certification evidence that the system can perform its allocated requirements. SLS captures this capability as a Design Math Model (DMM) of the behavior of the system. This model would have been the same model used to verify the derived requirement. This subtle difference eliminates the definition and approval of the new requirement, the associated DVO, and the verification closure. If there were few opportunities to apply this approach, work-hour savings would not be significant enough to pursue this approach. In reality, complex systems such as the SLS afford many opportunities to apply this approach.

For example, it is common – and it can be counter-productive to establish – ICD parameters such as a bolt pattern as a requirement. This can result in the verification objective that the ICD match the drawings on each side of the interface. Proper implementation on configuration management (CM) establishes this match when each item is approved at the configuration control board, and audits and technical reviews are used to check for any mistakes in the configuration.

Establishing this check as a verification objective fulfilled by inspection is a redundant, but limited, check on the CM system that occurs far too late in the design cycle to be effective. If these two parameters did not match prior to verification, it would be impossible to properly test and analyze the interface. In addition, attempts to avoid using design decisions in concert with requirement development can lead to unneeded work trying to derive a verifiable requirement that will result in the design solution that already exists.

An example would be attempting to establish a requirement for the vehicle outer mold line (OML). If the requirement stated, "The system OML shall result in a maximum vehicle drag of value X", the result would involve

American Institute of Aeronautics and Astronautics

much more work than simply using good design practice for the OML and showing that the vehicle performs adequately based on the drag characteristics of this design determined by test and analysis.

For a system to perform its top-level functions, each part in the system enables the system to perform its functions. Each part operates as a decomposed sub-function and must be designed and certified to perform its respective sub-function. The parts and their sub-functions are highly coupled in a launch vehicle, and allocation of these functions to each piece of hardware and software in the system must consider what can be verified at the hardware component level and what must be verified at each higher level of the system assembly and at the total system level.

As stated earlier, the SLS system is divided into four major hardware/software Program elements – Engines, Boosters, Stages, and SPIE. The SLS system decomposition provides an allocated set of requirements and design constraints that allow these elements to be certified for their portion for the SLS system functionality. These element designs and certifications cannot be done in an independent and decoupled manner because of the complex interactions of each element that affect the functionality of the system. Managing this complex interaction is the key to efficient systems engineering. Classic systems references typically do not discuss this in great detail. Prime examples for launch vehicles are structural integrity and structural dynamics of the integrated system. It is not possible to allocate to the Program element its contribution to the structural integrity and the dynamic flex modes of the vehicle, because it is a function of the overall design itself. Estimates can be made, but the system complexity is too great to sufficiently characterize the element contribution without a substantially mature design. For this reason, iteration between the element design and the system level loads and environment definition is necessary. The functional decomposition foundation provides a flexible structure to manage these iterations.

C.   Managing System Interactions

With the system decomposed into functions and assigned for either system or element-level implementation, the next step is for each organization to focus on the development of its allocated design responsibility in parallel efforts. Since the system is highly coupled, a mechanism must exist to assess the interactions that arise within each development activity that might affect another parallel development effort such as the structural design example above. A matrix of the SLS elements and the technical disciplines involved in the hardware/software development is shown in Figure 5.

SLS Program Office		SLS Chief Engineer Organization	Systems Engineering	Integrated Avionics and Software	Vehicle Management	Structures and Environments	Propulsion	Production	Test	Safety & Mission Assurance
Program Manager	SEI Office Manager	Program Chief Engineer	Lead Systems Engineer	Lead Engineer	Discipline Lead Engineer	Discipline Lead Engineer	Discipline Lead Engineer	Discipline Lead Engineer	Discipline Lead Engineer	Chief S&MA Officer
Stages Manager	SEI Lead	Stages Chief Engineer	ELSE	EDLE	EDLE	EDLE	EDLE	EDLE	EDLE	ECSO
Booster Manager	SEI Lead	Booster Chief Engineer	ELSE	EDLE	EDLE	EDLE	EDLE	EDLE	EDLE	ECSO
Liquid Engines Manager	SEI Lead	Engines Chief Engineer	ELSE	EDLE	EDLE	EDLE	EDLE	EDLE	EDLE	ECSO
Spacecraft & Payload Integration Manager	SEI Lead	Spacecraft & Payload Integration CE	ELSE	EDLE	EDLE	EDLE	EDLE	EDLE	EDLE	ECSO

Figure 5. SLS Program and Engineering structure

The SLS Program performs the SE&I activity as a government in-house-led activity based at the Marshall Space Flight Center (MSFC) in Huntsville, AL. Although SLS development involves substantial work from other NASA Centers and space industry contactors, the MSFC Engineering Directorate and Departments along with the MSFC Safety & Mission Assurance (S&MA) Directorate, are the primary sources that perform this function for the SLS Program. The boxes above in orange represent the SLS Program office. The boxes in red, yellow, and blue are matrixed support from the MSFC Engineering Directorate. Each color represents a key difference in the role with SE&I. The boxes in green represent the unique role S&MA Directorate. As shown in red, the Chief Engineer is the chief designer of the vehicle. The SLS Chief Engineer is matrixed from the MSFC Engineering Directorate and also has the authority to represent the MSFC Chief Engineer as the System Technical Authority. The yellow and blue boxes represent engineering leads that are accountable their respective applied engineering disciplines to the chief engineer. The distinction between the blue and yellow boxes will be discussed later.

The matrix structure illustrates the SLS mechanism for identifying and addressing interactions within a hardware item (horizontal integration) and interactions between the hardware items (vertical integration).   This organizational construct is effective because the technical disciplines have the tools and expertise to characterize all known

American Institute of Aeronautics and Astronautics

interactions in the system within their disciplines (vertical) and the SLS Systems Engineering (SE) organization has access to common data regarding element interactions (horizontal). The "E" prefix indicates the corresponding element lead for each discipline who is responsible for the element engineering for the respective discipline and accountable to the System level discipline lead for identifying affects beyond their element,

A simple hypothetical example would be the Propulsion Discipline's need to reduce feedline pressure drop. An increase in feedline diameter would reduce the pressure drop, but it would likely increase mass, which would have to be evaluated for the impact on payload performance by Engineering's Vehicle Management Discipline. In addition, the increased diameter would have to be changed in drawings by the Engineering Structures and Environments Discipline, which could cause interference with other hardware. All of these interactions must be considered before any such change is implemented within the system.

In reality, the Discipline Lead Engineers (DLEs) are responsible for a group of related engineering specialties, or true disciplines, in the sense that proficiency in these specialties requires focused specialty training and/or experience. To assign a DLE position to each specialty would result in an organizationally flat reporting structure to the SLS Chief Engineer. Since all of the engineering specialties are interrelated, there are multiple ways to group the DLEs. In SLS, the most straightforward way to group these DLEs was to map the DLEs to the MSFC institutional organization structure. The Engineering and Safety and Mission Assurance (S&MA) Directorates at MSFC provide both the engineers that do the direct engineering for SLS and the technical discipline insight into the prime contractor tasks. By assigning the DLE to represent specific organizational units, the DLE has clear lines of responsibility that match discipline technical authority of the respective institutional organizational manager and has the authority to speak for these managers regarding the technical adequacy of the work performed.

The SLS Chief Engineer, who approves the technical adequacy of the entire system, solicits input from each engineering organizational unit involved in the design, prior to this approval. In addition, the SMA Directorate approves the adequacy of the design according to agency expectations for SMA requirements and procedures. MSFC assigns a Chief Safety Officer (CSO) to SLS, who represents the center SMA organization. This organization also provides direct engineering tasks in support of the design. The matrix therefore has an SMA DLE column that represents both the role of the CSO and the role of the DLE to execute design tasks assigned to SMA. The matrix structure serves multiple purposes. It is a basis for specific task assignments for the design of the vehicle as led by the SLS Chief Engineer, as well as the programmatic tasks from the SLS Program office. It provides the mechanism to address the interactions in the system. And finally, it provides the structure for the MSFC Chief Engineer and MSFC CSO.

Although, there is nothing fundamentally unique in this concept, the SLS implementation has some unique features to make the model more effective. One of these key features is that systems engineering (SE) is considered a discipline whose leader is a peer to the specialized discipline lead. Known as the Lead Systems Engineer (LSE), this position is also a direct report to the MSFC Chief Engineer. Additionally, the LSE is matrixed and directly accountable to the Program office through the SLS SE&I manager, who has responsibility for the programmatic details of the SE&I effort. The SE column along with the Integrated Avionics and Software (IAS) are shown in yellow because of their unique dual-accountability role to the Chief Engineer and the SE&I manager.

The LSE has the responsibility for the flow of technical information, so that all interactions are accounted for. To use a sports analogy, the LSE operates much like a football quarterback, as a leader among peers, to implement the direction of the SLS Chief Engineer as the coach. The LSE is expected to have broad understanding of the system design and engineering disciplines. The LSE applies this broad knowledge to ensure that, as the design detail increases through the development process, all interactions are accounted for at an acceptable level of technical, schedule, and cost risk.

Because there are countless ways that subsystems can interact, the communication of hardware, software, and discipline experts is the only practical way to adequately address these interactions. Although, research and technology development continues to improve the ability to simulate all these interactions, it is not currently feasible to completely do so. However, the LSE facilitates a communication process that relies very heavily on key processes and discipline models to effectively evaluate the interactions. This must all be integrated with program cost and schedule impacts.

As mentioned above, the other column that has dual accountability in the SLS organizational matrix is Integrated Avionics and Software (IAS). The matrixed lead engineer of this column not only has responsibility for the engineering design task with respect the avionics and software but also has direct responsibility to deliver the flight software for SLS. The IAS lead engineer functions as both a DLE accountable to the SLS Chief Engineer and is matrixed to the SLS SE&I Manager for direct guidance with respect to programmatic priorities.

American Institute of Aeronautics and Astronautics

Assigning these disciplines at the System level and at the decomposed element level leads to a two-dimensional responsibility matrix. This matrix provides a basis for assessing discipline-to-discipline effects at any end item-level of the system (System, element, subsystem, component, part), as well as the end item-to-end item interaction through any discipline. The element offices (Figure 6) have the responsibility of designing, developing and delivering the specific hardware end items.

MARSHALL SPACE FLIGHT CENTER
SPACE LAUNCH SYSTEM PROGRAM OFFICE
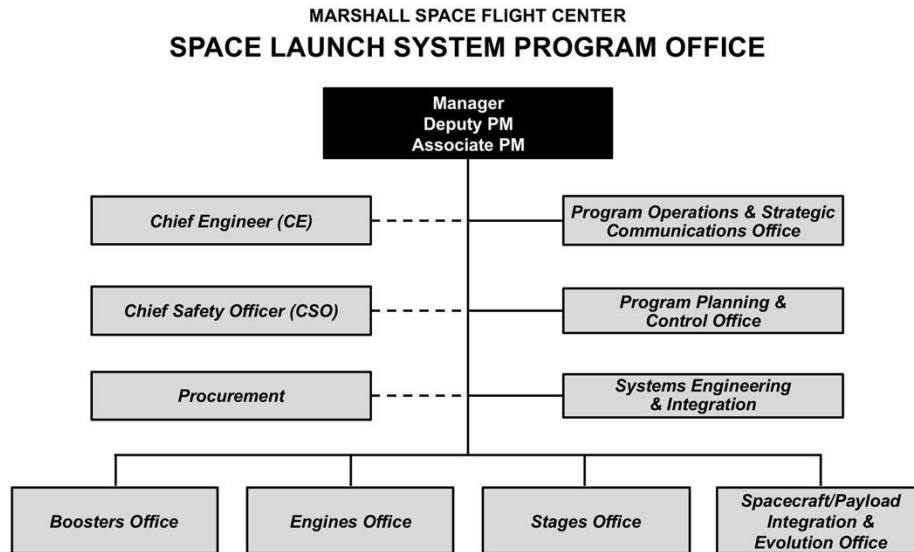
Figure 6. SLS Program organization showing element offices.

The SLS Program element offices have allocated budgets, allocated technical requirements, and allocated programmatic scope. They implement primarily through large industry prime contracts. Interactions with other parts of the system can have very significant impacts on the contracted activity. Careful assessment and early identification of interactions is not only critical to minimize time and effort of technical work but also to minimize unplanned contract changes that can create additional burden on schedule and resources. Design activities at the element level are assessed by each element DLE. A key part of this assessment is determining potential impacts to the system that would propagate through the interactions under that specific discipline's scope.

When a change to the SLS System baseline is needed or may be needed as a result of an element design update, SLS implements a task team process to assess and define any System-level changes to accommodate the effect that was driven out by the element design process. This process works in the same manner when an element impact is identified by System-level design activities. This process, in concert with rigorous configuration management, is key in efficiently managing system interactions during the development process.

The SLS task team process is simply a methodical way to address an integrated effect in the System. The process is executed to develop the information needed to determine if a change to the SLS configuration-controlled baseline is needed, and, if so, the specific content of that change. In the latter case, the change is then processed to the appropriate SLS Configuration Control Board.

When a potential change is identified, the most affected DLE proposes a task team for approval by the SLS LSE. That task team provides a high-level description of the potential change and identifies which DLEs and which elements may be affected by the change. The LSE then ensures all affected parties are identified and solicits concurrence and a team member from each affected party. The task team member as assigned then represents the element manager or DLE in completion of the task. The task is completed when the all team members concur by signature that adequate information is available to proceed to a formal decision. If all parties concur that no changes are needed, the LSE has authority to close the task based on the captured information from the task team.

In the event that a no-change recommendation results in an increase in program risk, the LSE elevates the decision to the management level that has risk acceptance authority. In the case where the task team recommends a change, the recommendations are presented to the SLS Chief Engineer. Depending on the extent of the impacts, the recommendation is also presented to SLS Program management for a decision to proceed to a formal baseline change at an SLS configuration control board.

For the task team to address potential impacts of an interaction, many different types of activities may be needed. These may be as simple as checks of baseline information, such as an integrated drawing or may instead be a detailed trade study, where major design changes are analyzed for best options to address the impacts. These trade studies must not only address technical solutions but must be assessed for impacts to program costs and schedule. In addition, the relative risk posture of each option must be assessed. Providing detailed cost and schedule impacts is not practical in all cases due to the amount of effort required by prime contractors to provide accurate estimates. However, high-fidelity resource estimates are not always needed. These estimates only need to be accurate enough for relative comparisons of options and often lower fidelity estimates are adequate.

The task team process relies on the team working together to reach consensus on recommendations. SLS certainly recognizes the potential subjective nature of decision recommendations and understands that, at times, the team will not be able to reach consensus. When this occurs, the task team recommendation includes the dissenting opinion, which will follow the NASA process for elevation of this opinion to appropriate levels for a final decision.

Generally, any time any new information is put under SLS configuration control, it constitutes a change to SLS baseline. Even planned information, such as the first release of a system-level drawing, can cause unanticipated impacts. For this reason, the SLS standard process is to implement a task team for all board changes, including the introduction of a first version of a planned document. While it is the default process, the board chair always has the option to forgo this pre-coordination process when the change is well enough understood to proceed to a formal change request, or, in some cases, straight to a directive change of the baseline.

As mentioned earlier, managing a complex system interaction often requires multiple changes to the technical baseline. The role that each organization within the matrix plays in configuration is critical. The key aspect in configuration management that becomes amplified in a highly coupled system is configuration consistency. This is a critical and often overlooked aspect of configuration management. There is a natural tendency when a change is being evaluated to "get it right". However, "get it consistent" is just as critical. As discussed above, a change to a drawing can ripple into changes in lower level drawings, ICDs, system mass, structural integrity, and dynamic behavior, as well as many other parts of the baseline. Therefore, it is necessary to acknowledge all identified interactions in formal baseline changes in order not to lose the identified interaction driven out by the task team process.

For timely and effective capture of identified interactions, SLS implements a flexible, broad view of configuration management. Where practical, SLS attempts to change all affected baseline information simultaneously in a single change directive. This not only reduces resources required to manage change, but it also encourages complete evaluation of all interactions. In practice, this cannot always be done because some effects take more work to fully characterize the interaction. In these cases, however, SLS rigorously captures all open work identified by the task teams where another part of the baseline is affected, but not all details are not yet completely defined.

D. Design Certification of the Integrated System

As pieces of the integrated design are finalized in the parallel process described in Subsection C above, these individual pieces will enter into their certification process where they demonstrate they have met the design requirements that were allocated to them as described in Subsection B above. This design certification phase of the program must also be conducted with continued careful consideration of the System interactions that could affect the validity of a completed certification effort.

This activity is further complicated by the fact that large and complex subsystem designs are typically certified without actually building and flight-testing them. There is no practical way to safely and cost-effectively fly a subsystem for the sole purpose of certifying the design. For this reason, the subsystem design is certified by test-verifying key hardware and certifying the integrated System by analysis. While continuing to improve, analysis methods will remain for the foreseeable future only approximations of the physical behavior of a launch system. Certification of the system by analysis increases the importance of an effective SE&I process.

NASA experience with complex launch systems has led to a robust understanding of key aspects related to understanding the risks associated with certification by analysis. The SLS Design Model (DM) Log builds on this experience and is an integral part of the SLS design process. The DM Log is managed as part of the SLS configuration baseline and addresses some key challenges for launch system certification. It must ensure that models used for the certification analysis are adequate. In addition, the analysis must stay consistent with the design as it matures. When the SLS completes its certification, all models that are critical for declaring compliance to SLS allocated requirements will be baselined within the DM Log. In addition, a set of metadata will be captured within the log that will document the evidence that the models have been rigorously evaluated as the primary evidence that the SLS will fly successfully. Arguably, the most critical attribute within this metadata is model validation.

For the first flight of the SLS, the model validation evidence will be used in lieu of comparison to a complete system test. When possible, this evidence will include a comparison of the model by using it to analyze similar systems. Also, the System models will be compared with test data from parts of the SLS, including tests of the SLS elements. When using the models in this fashion, the interactions within the system will differ from that of the actual SLS. Every affected discipline at both the element- and System-level must assess these representative cases to determine if potential System interactions are adequately simulated in these representative cases. Once the models have been validated, they are then used to analyze the SLS system for a set of bounding cases and disciplines that will prove each detailed verification objective is met. As the flight hardware is manufactured, data from flight hardware testing will be used to increase the confidence in these models prior to flight and post-flight to increase the confidence in the models for the next flight.

E. Flight Certification of the System

As discussed above, the system design for the SLS will be certified before flight. Following the certification of the System Design, a great deal of new information about the system will be defined and managed. Data from actual flight unit builds of hardware and specific mission details are the major sources of this information. SLS SE&I manages the development of this new information in the same functional framework as was used in the definition and design phases of the Program. Figure 7 illustrates the SLS SE&I data structure that allows the progression of the System to and beyond flight certification.

ESD Requirement A	Certification	Assembly and Checkout	Mission Specific	Hazards	Post Flight
SLS Function 1	DVO –1	N/A	MR-1	HC-1	FDA-1
SLS Function 2	DVO –2	OAR-2	LCC-1	N/A	FDA-2
SLS Function 3	Model-1	OAR-3	N/A	HC-1	FDA-3

Figure 7. SLS SE&I data structure

The benefit of this structure is that it provides a basis to assess any new data against the existing data previously used to certify the System to ensure that it is consistent with the certified design. It also allows the design data, unit-build data, mission-specific data, and post-flight data to be assessed and accessed in a structured manner. This data management structure allows for effective queries of key information about the system. Shown in the second column of Figure 7 is a simplified representation of design certification data, such as design data, allocated requirements, DVOs, DM models and verification compliance data.

The third column represents assembly and checkout data such as operational assembly and checkout requirements (OARs), closure information, and non-conformance identification and resolution. This allows a function by assessment of available data from assembly and checkout and indicates that the system is coming together as designed.

The fourth column represents mission-specific data, including mission requirements (MRs), launch commit criteria (LCC), flight rules and mission-specific analysis/assessments. This facilitates a function-by-function assessment of the mission to ensure that the mission will be performed within the systems certified design capability.

The fifth column represents all data with respect to hazards including hazard cause (HC), design and operational controls and control verification evidence.

The sixth column is post-flight data analysis (FDA) to confirm that the system performed the mission within its certified capability. Notably, in some cases, the function is assured for the life cycle based on design certification alone. This is noted in Figure 7 as Non-Applicable (N/A). The development of this data will utilize the same matrix structure and SE&I processes used to certify the system design.

F. Supporting the Operation of the Vehicle

The SLS SE&I structure and processes were established with the complete system life cycle in mind. The SLS system will be operated by its exploration systems sister programs, Ground Systems Development & Operations (GSDO), based at NASA Kennedy Space Center, and the Orion Multi-Purpose Crew Vehicle (MPCV), based at Johnson Space Center. GSDO has the Ground Systems operations function and Orion is responsible for the flight operations function. The SLS SE&I Program will support the operations using the matrix structure in Figure 5 by performing real-time evaluation of the system behavior to provide feedback to the mission management team. In addition, the SE&I team will perform post-flight evaluation of the SLS for each Discipline and element. This evaluation of the system will be used to assess how well SLS performed the mission, and the data will be used to update and increase the confidence in all the engineering used to certify SLS for the next flight.

IV. Summary

The SLS Program's systems engineering approach described above has made a significant contribution to the program's success to date. Facing several challenges, the Program has adapted general SE&I principles to effectively engineer the System through its development and is proceeding to design certification, and into flight operations. These processes have successfully brought the SLS through its Critical Design Review (CDR). This was the first CDR of a U.S. government developed crewed space launch system since the Space Shuttle in the 1970s. Along the way, SLS has solved many technical challenges resulting from complex hardware interactions that the SLS structure was designed to accommodate. Every major element has completed development testing and is in production of flight hardware and software. Through the continued efforts of the SLS SE&I team, the first launch of what will be the world's most powerful, capable launch vehicle is now planned for 2019.