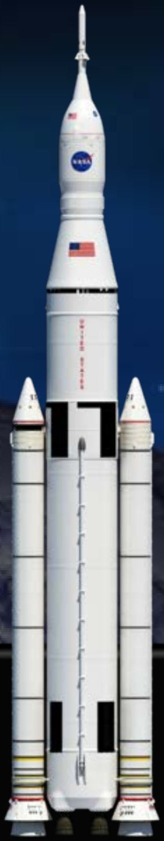
National Aeronautics and Space Administration

Engineering Elegant Systems: Engineering at the System Level

29 March 2019

Michael D. Watson, Ph.D.

Consortium Team
UAH
George Washington University
Iowa State
Texas A&M
Dependable System Technologies, LLC
Multidisciplinary Software Systems Research Corporation (MSSRC)
Missouri University of S&T
University of Michigan
AFRL Wright Patterson



Space Launch System

Outline

- **Understanding Systems Engineering**
  - Postulates
  - Hypothesis
  - Principles
- **Systems Engineering Domain**
  - System Integration
    - System State Variables
      - Goal Function Tree
      - State Analysis Model
    - System Value Model
    - System Integrating Physics
    - System Autonomy
    - Multidisciplinary Design Optimization (MDO)
    - Engineering Statistics
    - Methods of System Integration
  - Discipline Integration
    - Sociological Concepts in Systems Engineering
    - Information Flow
    - System Dynamics
- **Summary**

Understanding Systems Engineering

Motivation

- **System Engineering of Complex Systems is not well understood**

- **System Engineering of Complex Systems is Challenging**
  - System Engineering can produce elegant solutions in some instances
  - System Engineering can produce embarrassing failures in some instances
  - Within NASA, System Engineering does is frequently unable to maintain complex system designs within budget, schedule, and performance constraints

- **"How do we Fix System Engineering?"**
  - Michael D. Griffin, 61st International Astronautical Congress, Prague, Czech Republic, September 27-October 1, 2010
  - Successful practice in System Engineering is frequently based on the ability of the lead system engineer, rather than on the approach of system engineering in general
  - The rules and properties that govern complex systems are not well defined in order to define system elegance

- **4 characteristics of system elegance proposed as:**
  - System Effectiveness
  - System Efficiency
  - System Robustness
  - Minimizing Unintended Consequences

Consortium

- **Research Process**
  - Multi-disciplinary research group that spans systems engineering areas
  - Selected researchers who are product rather than process focused

- **List of Consortium Members**
  - Michael D. Griffin, Ph.D.
  - Air Force Research Laboratory – Wright Patterson, Multidisciplinary Science and Technology Center: Jose A. Camberos, Ph.D., Kirk L. Yerkes, Ph.D.
  - Doty Consulting Services: John Doty, Ph.D.
  - George Washington University: Zoe Szajnfarber, Ph.D.
  - Iowa State University: Christina L. Bloebaum, Ph.D., Michael C. Dorneich, Ph.D.
  - Missouri University of Science & Technology: David Riggins, Ph.D.
  - NASA Langley Research Center: Peter A. Parker, Ph.D.
  - Texas A&M University: Richard Malak, Ph.D.
  - Tri-Vector Corporation: Joey Shelton, Ph.D., Robert S. Ryan, Kenny Mitchell
  - The University of Alabama in Huntsville: Phillip A. Farrington, Ph.D., Dawn R. Utley, Ph.D., Laird Burns, Ph.D., Paul Collopy, Ph.D., Bryan Mesmer, Ph.D., P. J. Benfield, Ph.D., Wes Colley, Ph.D., George Nelson, Ph.D.
  - The University of Colorado – Colorado Springs: Stephen B. Johnson, Ph.D.
  - The University of Michigan: Panos Y. Papalambros, Ph.D.
  - The University of Texas, Arlington: Paul Componation, Ph.D.
  - The University of Bergen: Erika Palmer

- **Previous Consortium Members**
  - Massachusetts Institute of Technology: Maria C. Yang, Ph.D.
  - Stevens Institute of Technology – Dinesh Verma
  - Spaceworks – John Olds (Cost Modeling Statistics)
  - Alabama A&M – Emeka Dunu (Supply Chain Management)
  - George Mason – John Gero (Agent Based Modeling)
  - Oregon State – Irem Tumer (Electrical Power Grid Robustness)
  - Arkansas – David Jensen (Failure Categorization)

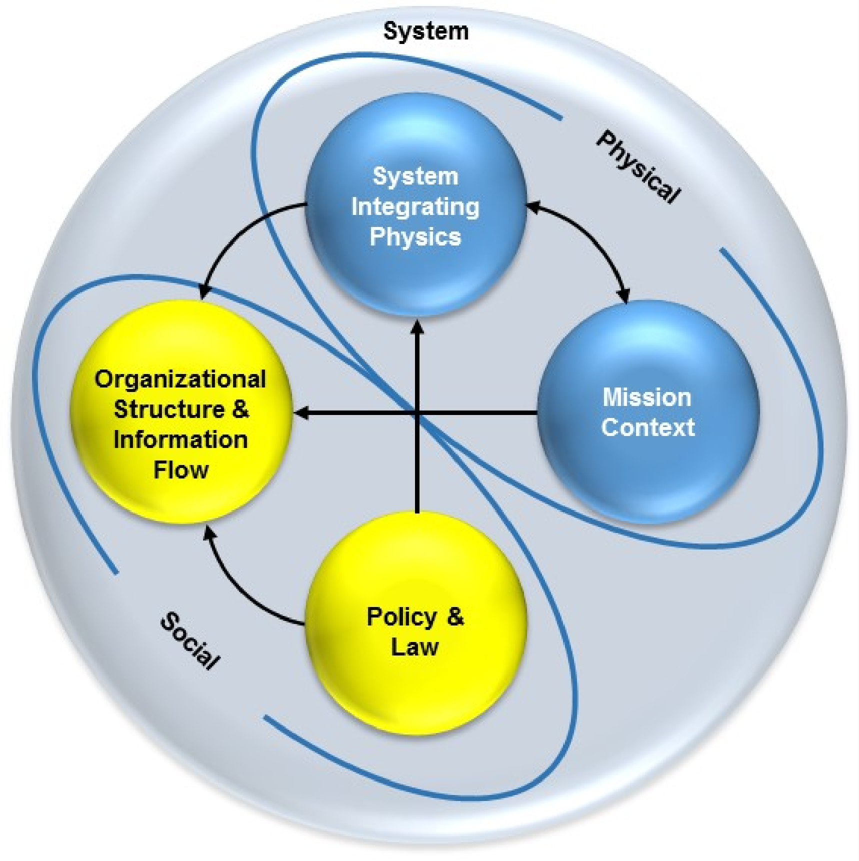
~50 graduate students and 15 undergraduate students supported to date

Understanding Systems Engineering

- **Definition – System Engineering is the engineering discipline which integrates the system functions, system environment, and the engineering disciplines necessary to produce and/or operate an elegant system.**
  - Elegant System - A system that is robust in application, fully meeting specified and adumbrated intent, is well structured, and is graceful in operation.

- **Primary Focus**
  - System Design and Integration
    - Identify system couplings and interactions
    - Identify system uncertainties and sensitivities
    - Identify emergent properties
    - Manage the effectiveness of the system
  - Engineering Discipline Integration
    - Manage flow of information for system development and/or operations
    - Maintain system activities within budget and schedule

- **Supporting Activities**
  - Process application and execution
    - Processes organize the engineering



Systems Engineering Postulates

System Integration (physical/logical system)
Discipline Integration (social system)
Both System and Discipline Integration

- **Postulate 1: Systems engineering is system specific and context dependent in application**
- **Postulate 2: The Systems Engineering domain consists of subsystems, their interactions among themselves, and their interactions with the system environment**
- **Postulate 3: The function of Systems Engineering is to integrate engineering disciplines in an elegant manner**
- **Postulate 4: Systems engineering influences and is influenced by organizational structure and culture**
- **Postulate 5: Systems engineering influences and is influenced by budget, schedule, policy, and law**
- **Postulate 6: Systems engineering spans the entire system life-cycle**
- **Postulate 7: Understanding of the system evolves as the system development or operation progresses**
- **Postulate 7 Corollary: Understanding of the system degrades during operations if system understanding is not maintained.**

Systems Engineering Principles

- **Principle 1: Systems engineering integrates the system and the disciplines considering the budget and schedule constraints**
- **Principle 2: Complex Systems build Complex Systems**
- **Principle 3: A focus of systems engineering during the development phase is a progressively deeper understanding of the interactions, sensitivities, and behaviors of the system, stakeholder needs, and its operational environment**
  - Sub-Principle 3(a): Mission context is defined based on understanding of the stakeholder needs and constraints
  - Sub-Principle 3(b): Requirements and models reflect the understanding of the system
  - Sub-Principle 3(c): Requirements are specific, agreed to preferences by the developing organization
  - Sub-Principle 3(d): Requirements and design are progressively elaborated as the development progresses
  - Sub-Principle 3(e): Hierarchical structures are not sufficient to fully model system interactions and couplings
  - Sub-Principle 3(f): A Product Breakdown Structure (PBS) provides a structure to integrate cost and schedule with system functions
  - Sub-Principle 3(g): As the system progresses through development, a deeper understanding of the organizational relationships needed to develop the system are gained.
  - Sub-Principle 3(h): Systems engineering achieves an understanding of the system's value to the system stakeholders
  - Sub-Principle 3(i): Systems engineering seeks a best balance of functions and interactions within the system budget, schedule, technical, and other expectations and constraints.

Systems Engineering Principles

- **Principle 4: Systems engineering has a critical role through the entire system life-cycle**
  - Sub-Principle 4(a): Systems engineering obtains an understanding of the system
  - Sub-Principle 4(b): Systems engineering defines the mission context (system application)
  - Sub-Principle 4(c): Systems engineering models the system
  - Sub-Principle 4(d): Systems engineering designs and analyzes the system
  - Sub-Principle 4(e): Systems engineering tests the system
  - Sub-Principle 4(f): Systems engineering has an essential role in the assembly and manufacturing of the system
  - Sub-Principle 4(g): Systems engineering has an essential role during operations, maintenance, and decommissioning
- **Principle 5: Systems engineering is based on a middle range set of theories**
  - Sub-Principle 5(a): Systems engineering has a physical/logical basis specific to the system
  - Sub-Principle 5(b): Systems engineering has a mathematical basis
  - Sub-Principle 5(c): Systems engineering has a sociological basis specific to the organization(s)
- **Principle 6: Systems engineering maps and manages the discipline interactions within the organization**
- **Principle 7: Decision quality depends on system knowledge present in the decision-making process**
- **Principle 8: Both Policy and Law must be properly understood to not overly constrain or under constrain the system implementation**

Systems Engineering Principles

- **Principle 9: Systems engineering decisions are made under uncertainty accounting for risk**
- **Principle 10: Verification is a demonstrated understanding of all the system functions and interactions in the operational environment**
- **Principle 11: Validation is a demonstrated understanding of the system's value to the system stakeholders**
- **Principle 12: Systems engineering solutions are constrained based on the decision timeframe for the system need**
- **Principle 13: Stakeholder expectations change with advancement in technology and understanding of system application.**
- **Principle 14: The real physical system is the perfect model of the system**
  - Kullback-Liebler Information shows the actual system is the ideal information representation of the system
    - $I(f,g) = \int f(x)\log(f(x))\,dx - \int f(x)\log(g(x|\theta))\,dx = 0$

System Engineering Hypotheses

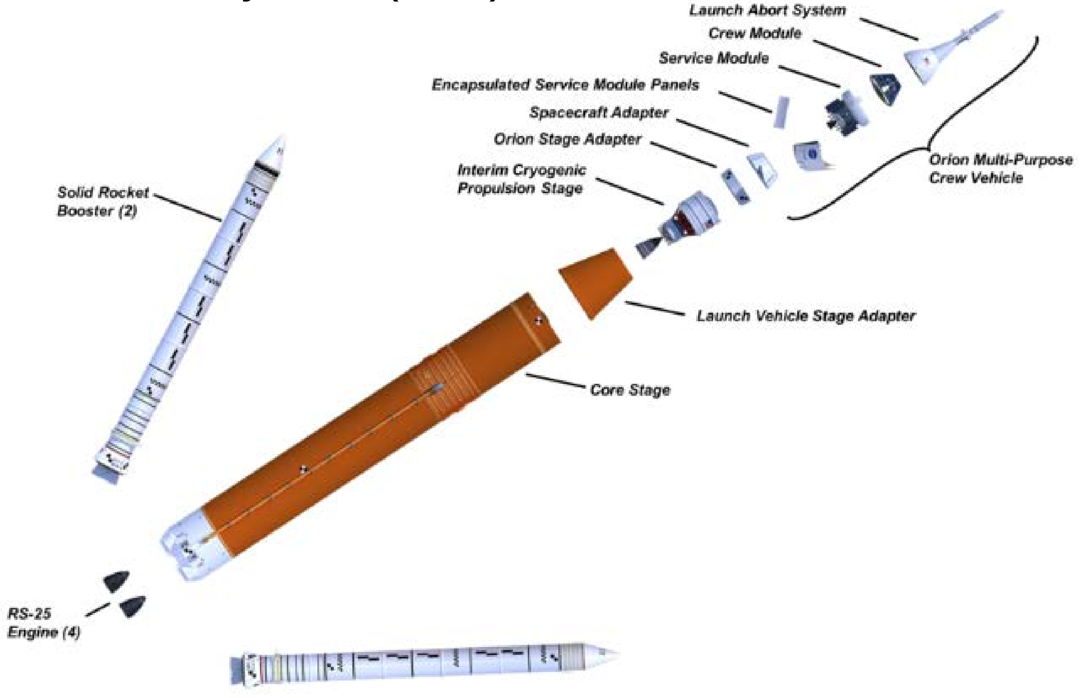
- **Hypothesis 1: If a solution exists for a specific context, then there exists at least one ideal Systems Engineering solution for that specific context**
  - Hamilton's Principle shows this for a physical system
    - $\int_{t_1}^{t_2}(\delta T - \delta V + \delta W)dt = 0$
- **Hypothesis 2: System complexity is greater than or equal to the ideal system complexity necessary to fulfill all system outputs**
- **Hypothesis 3: Key Stakeholders preferences can be accurately represented mathematically**

Mathematical Basis of Systems Engineering: Mathematical Category Theory

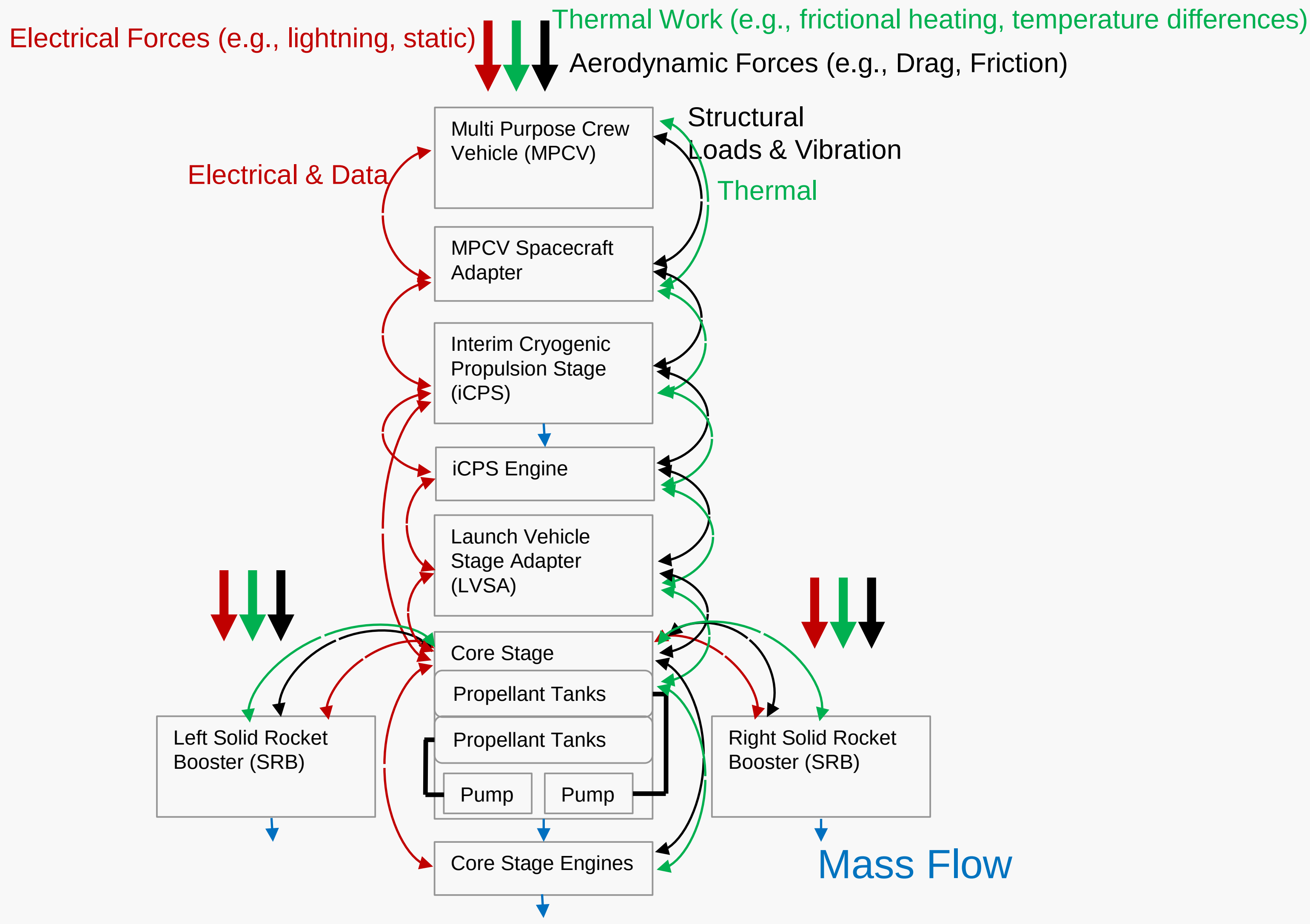
System Representations

- **Systems are comprised of 2 basic structures**
  - Postulate 2: The Systems Engineering domain consists of subsystems, their interactions among themselves, and their interactions with the system environment
  - Components
  - Relationships among components
    - Physical
    - Logical
  - Relationships with the environment
    - Physical

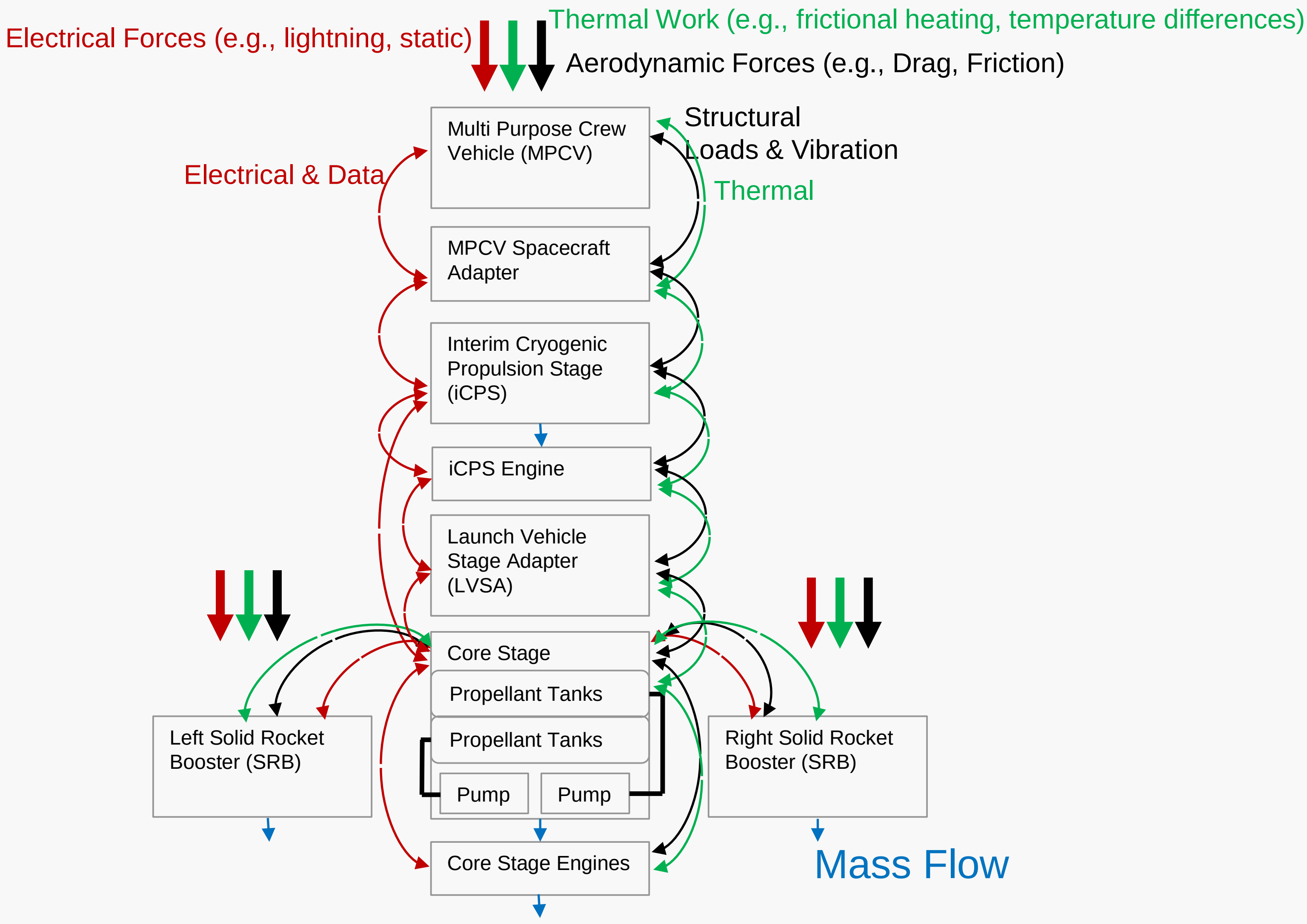
Major Components of the NASA Space Launch System (SLS)



Rocket Physical and Logical Relationships



Rocket as a Mathematical Category



Mathematical Category

A Mathematical Category consists of

- Objects (i.e., system components): a,b,c,…
- Arrows (i.e., system relationships between components and the environment): f,g,…

A Mathematical Category has properties

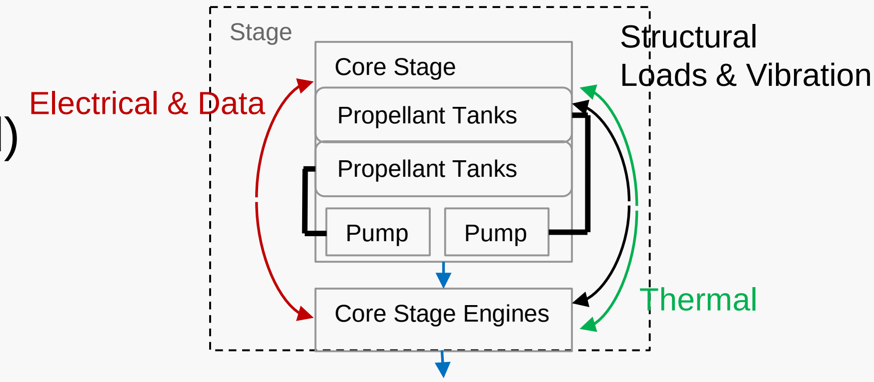
- Domain/Codomain
  - f: a → b where a is the domain of f and b is the codomain of f
- Identify Relationship
  - $id_a = 1_a$: a → a

$$\begin{tikzcd} a \arrow[loop right] \end{tikzcd}$$

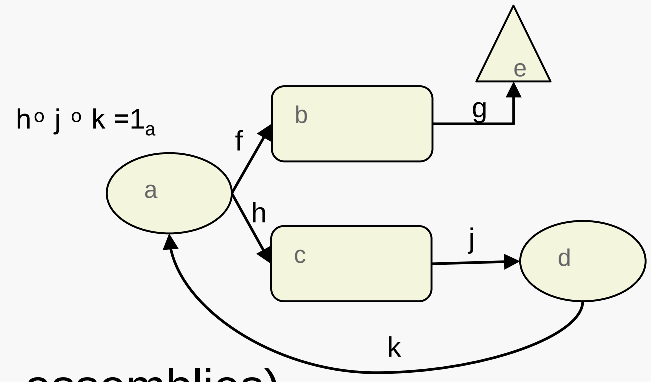
- Associativity
  - $f \circ (g \circ h) = (f \circ g) \circ h$
- Composition
  - Composition can be performed by various mathematical operations (i.e., addition, subtraction, multiplication, division)
  - $a \xrightarrow{f} b \xrightarrow{g} c = a \xrightarrow{f \circ g} c$

$$\begin{tikzcd} & b \arrow[dr, "g"] & \\ a \arrow[ur, "f"] \arrow[rr, "f \circ g"'] & & c \end{tikzcd}$$

Mathematical Category Types



- **Category Types**
  - Category of Sets
  - Category of Arrows (objects are implied)
  - Category of Groups
  - Category of Categories
  - Universal Category
  - Category of Small Categories
  - Abelian Categories



- **Objects within a category can be**
  - Objects (i.e., individual parts or components)
  - Sets (i.e., sets of individual parts)
  - Groups
  - Smaller Categories (i.e., stages, subsystems, assemblies)

- **Directed Graphs**
  - Directed graphs, when they meet the property conditions, are a form a mathematical category

Mathematical Category Transformations

Functors

- Mathematical morphisms between categories, F: $A \xrightarrow{F} C$
- Creates a mapping from one category to another
- Includes composition in the mapping

Natural Transformations

- Transformation is the same among all objects
- Is commutative
- If invertible, then is a ‘natural equivalence’ or ‘isomorphism’

Isomorphism

- If the relationships (arrows) are invertible between two objects, then the objects are isomorphic, $a \cong b$
  - $a \xrightarrow{f} b \xrightarrow{g} a$, f = g’, g = f’
- Categories can be isomorphic, A$\cong B$
  - The objects can be different, but the relationships between the objects of the two categories are preserved
    - i.e., different copies of the same system are isomorphic
    - Or, two different designs of the same system type may be isomorphic (e.g., different automobile makes with similar models)

Mathematical Category Transformations

Co-cones/Co-limits

- Co-cone
  - A common codomain for Functors operating on Category C

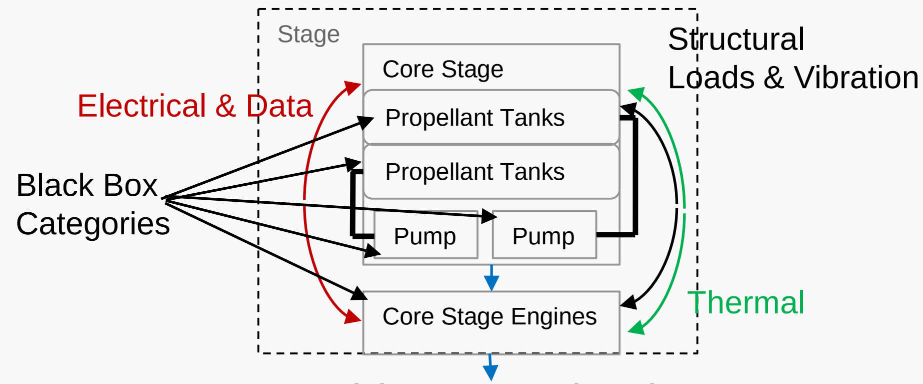
$$
\begin{tikzcd}
F_1 \arrow[dr, "\tau_1"'] & F_2 \arrow[l, "f_1"'] \arrow[r, "f_2"] \arrow[d, "\tau_2"] & F_3 \arrow[dl, "\tau_3"] \\
& c \in C &
\end{tikzcd}
$$

- Co-limit
  - The limit of the Co-cone defining the conditions where all Functors and mappings to objects of the Category, C, are included

https://www.nasa.gov/content/goddard/nasa-engineer-set-to-complete-first-3-d-printed-space-cameras/

Systems Engineering Application

Black Box
- Since a Category may contain smaller Categories, then an engineering 'black box' is a Category treated as an object within a larger Category



System Completeness
- The mathematical structure of the system Category provides a mechanism to construct a completeness proof for a given system

System Specification
- The System objects and relationships form the basis of the system requirements
- The Category must contain the correct and complete objects and relationships
  - Variations result in a system different than intended

System Assembly
- Co-cones and co-limits define the assembly operations needed to construct the system Category
- The Functors map parts from the parts category(s) to the system category
  - The parts may map to sub-categories (i.e., assemblies and subsystems) within the system category
- The limits define what must be included at each step of the assembly in order to be complete

Methods of System Integration

Goal: Techniques to Enable Integrated System Design and Assessments by the Systems Engineer

System State Variables

Goal: Utilize system state variables to understand the interactions of the system in relation to system goals and system execution

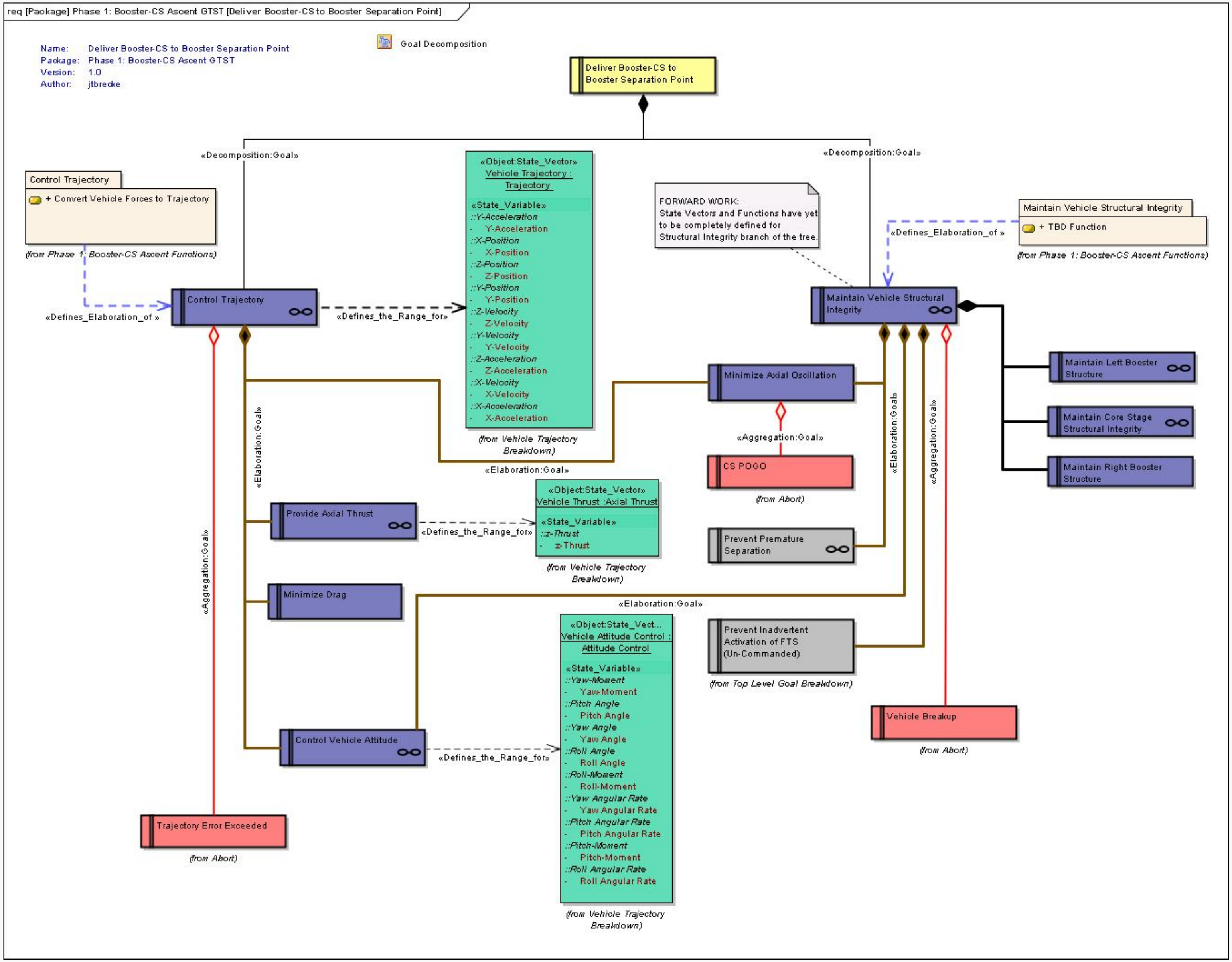
System State Models

- **System Stage Models represent the system as a whole in terms of the hardware and software states that the system transitions through during operation**

- **Goal Function Tree (GFT) Model**
  - "Middle Out" model of the system based on the system State Variables
  - Shows relationship between system state functions (hardware and software) and system goals
  - Does not contain system physical or logical relationships and is not executable

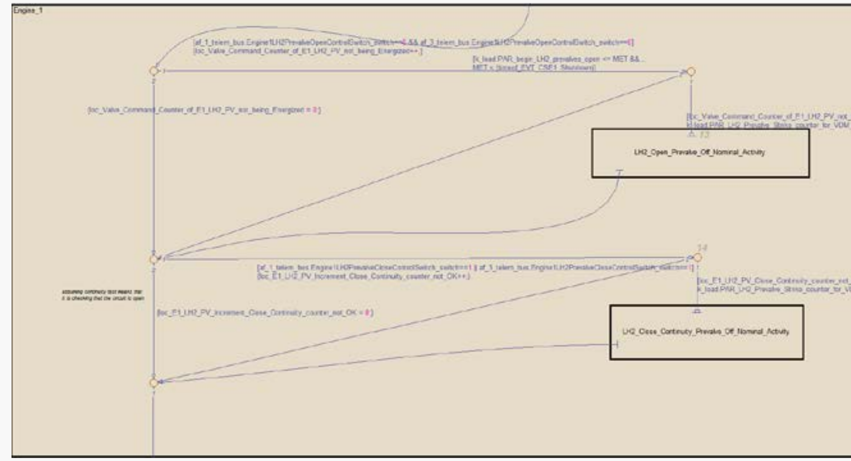
- **System State Machine Model**
  - Models the integrated State Transitions of the system as a whole (i.e., hardware states and software states)
  - Confirms system functions as expected
    - Checks for system hazardous, system anomalies, inconsistent state progression, missing states, improper state paths (e.g., short circuits in hardware and/or software design)
    - Confirms that the system states progress as stated in the system design
  - Executable model of system

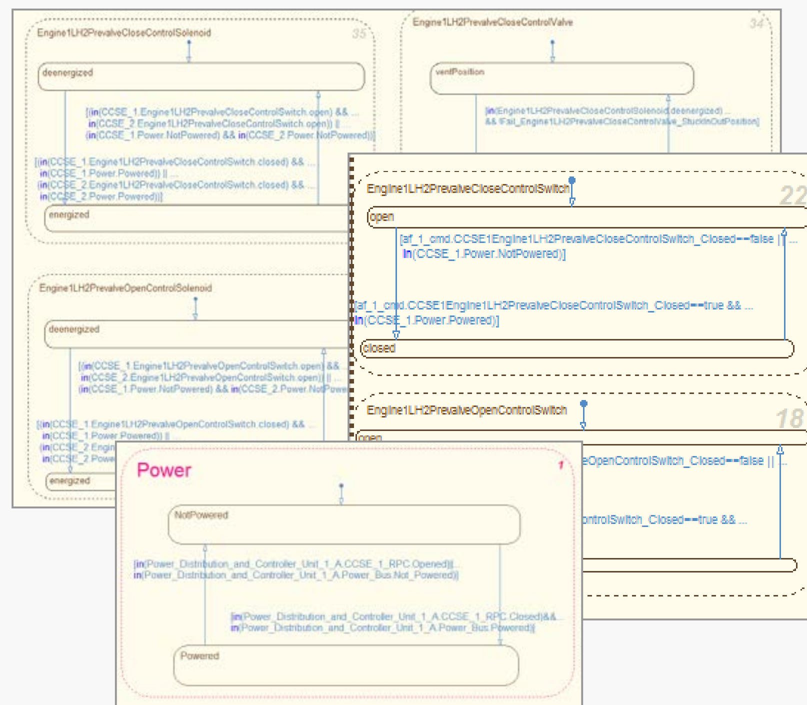
Booster – CS Ascent GFT



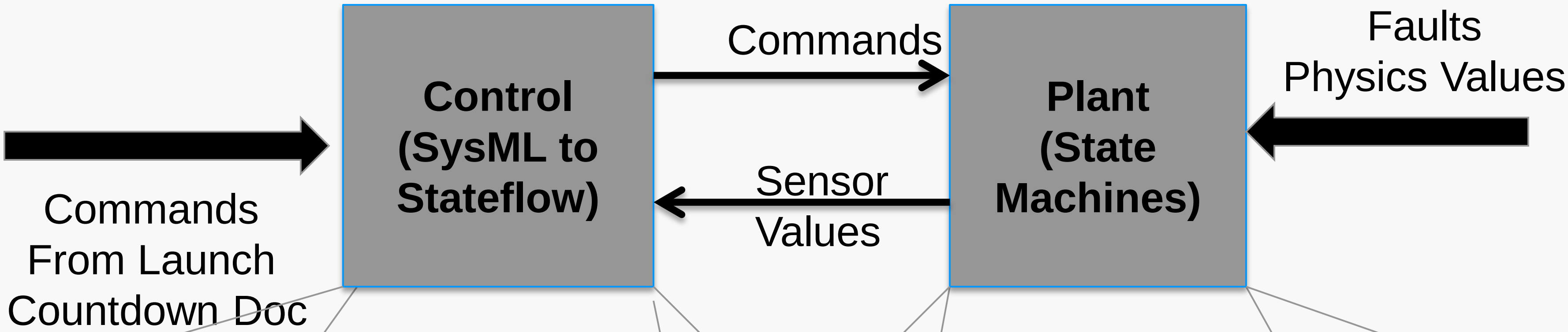
System State Machine Model

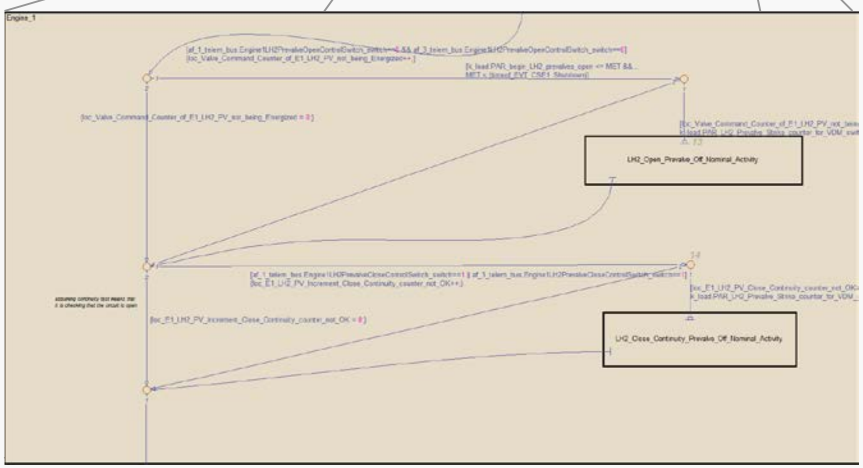
- **The state analysis model is split into two main components:**
  - Manager software model
  - System Plant
- **Modeled using MATLAB Stateflow**
  - Allows the software model to look like the SysML Activity Diagrams
  - Allows the SystembPlant to be modeled as State Machines
  - Allows those two models to interact with each other within the MATLAB environment
    - Facilitates the ability to generate custom analysis tools
- **Reads in command sequence to execute model**

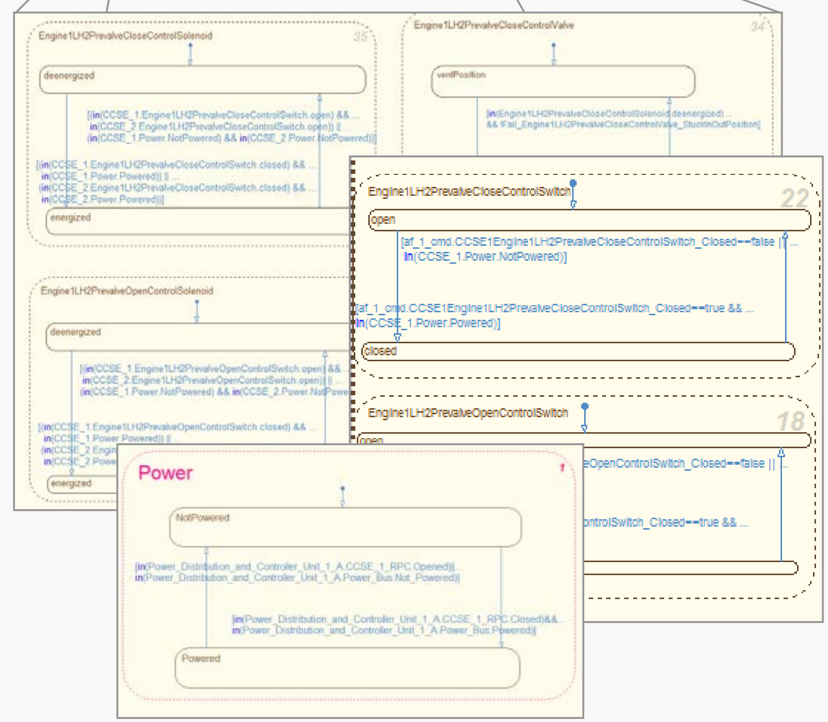




State Analysis Model for SLS M&FM







- 14% of R12 modeled
- Over 7,200 Transitions in the Vehicle and Software
- Over 3,500 States in the Vehicle

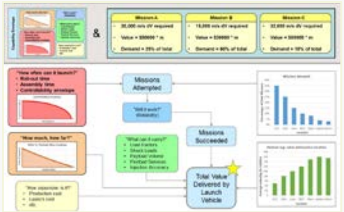
System Value

Goal: Utilize system state variables to understand the interactions of the system in relation to system goals and system execution

System Value Model

- **A System Value Model is a mathematical representation of Stakeholders Preferences (Expectations) for the system**
  - The basic structure is straight forward
  - The sociology/psychology of representing the Preferences can be a challenge
- **The System Value Model is the Basis of System Validation!!!**
  - The Requirements and Design Models form the basis of System Verification
  - The System Value Model forms the basis of System Validation
- **Constructing an SLS Value Model to compare to System Validation results**
  - Can expand to Integrated Stack with input from MPCV and GSDO
- **System Value model also provides basis for a measure of System Robustness**
  - How many mission types are supported by the system?

	Status	Gradient	Value
Efficiency	90%	150,000	135,000
Weight	700	-130	-91,000
Reliability	1500	2	3,450
Maintainability	7.8	-340	-2,652
Maintenance Cost	500	-1	-250
Support Equipment	12	-15	-180
Manufacturing Cost	700	-1	-700
Design Value		$	43,668

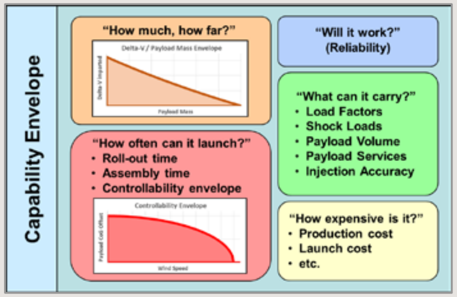


$$\pi = f_{aircraft} \; x_1, x_2, \dots, x_n$$

$$v_e = \sum_{j=1}^{m}\left(\sum_{i=1}^{n}\frac{\partial \pi}{\partial x_i}\cdot\frac{\partial x_i}{\partial y_j}\, y_j\right)$$

$$v_t = \sum_{k=1}^{p}\left(\sum_{j=1}^{m}\frac{\partial v_e}{\partial y_j}\cdot\frac{\partial y_j}{\partial z_k}\, z_k\right)$$

Mapping System Capability to Value



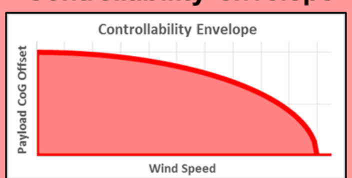
&

Mission A
- 20,000 m/s dV required
- Value = $50000 * m
- Demand = 25% of total

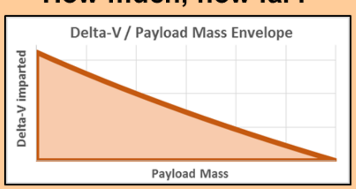
Mission B
- 15,000 m/s dV required
- Value = $30000 * m
- Demand = 60% of total

Mission C
- 32,000 m/s dV required
- Value = $80000 * m
- Demand = 15% of total

"How often can it launch?"
- **Roll-out time**
- **Assembly time**
- **Controllability envelope**



"How much, how far?"



"How expensive is it?"
- Production cost
- Launch cost
- etc.

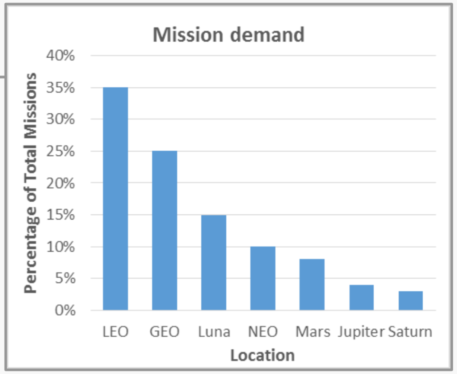
Missions Attempted

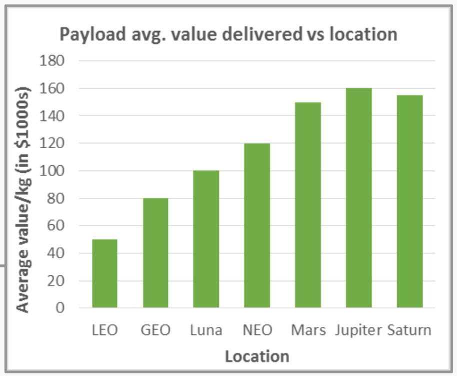
"Will it work?" (Reliability)

Missions Succeeded

"What can it carry?"
- Load Factors
- Shock Loads
- Payload Volume
- Payload Services
- Injection Accuracy

Total Value Delivered by Launch Vehicle





Launch Vehicle Value Model

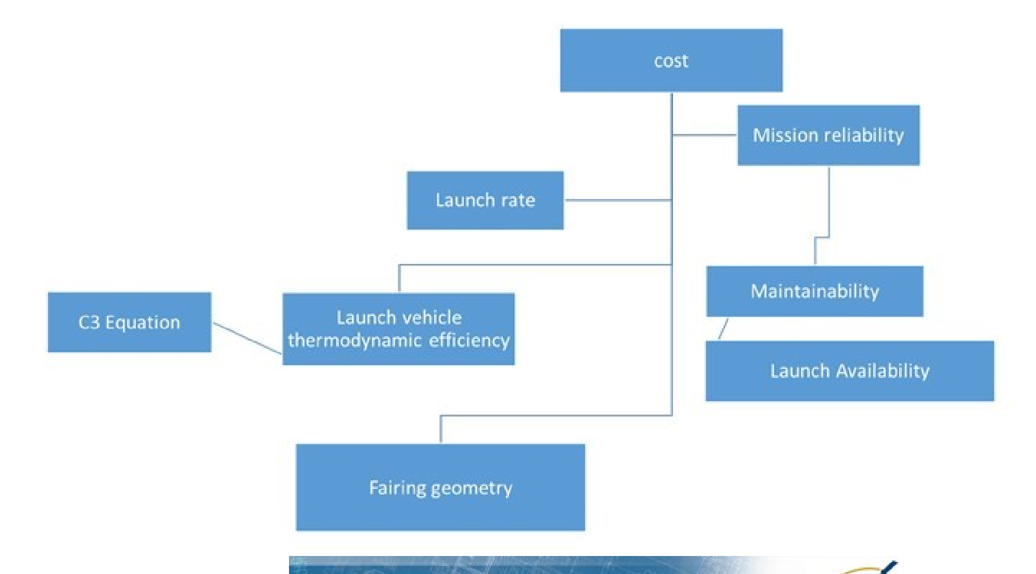
- **Launch Vehicle Value related to impact to national GDP**
- **Rockets are thermodynamic systems, there thermo-economics can be applied**

$$\dot{C}_T = \sum_i c_{ei}\, \dot{\epsilon}_i + \sum_n \dot{Z}_n$$

$$c_{ei} = \frac{\frac{\$}{kg}}{J/kg} \rightarrow \left(\frac{propellant\ cost}{exergy}\right) = \$/J$$

$$\dot{\epsilon}_i = \frac{kg}{yr}\left(\frac{J}{kg}\right) \rightarrow \left(\frac{mass}{year}\right) * HHV = \frac{J}{yr}.$$

$$\dot{Z}_n = L_R * unit\ cost + \frac{manufactoring\ base\ cost}{yr}$$



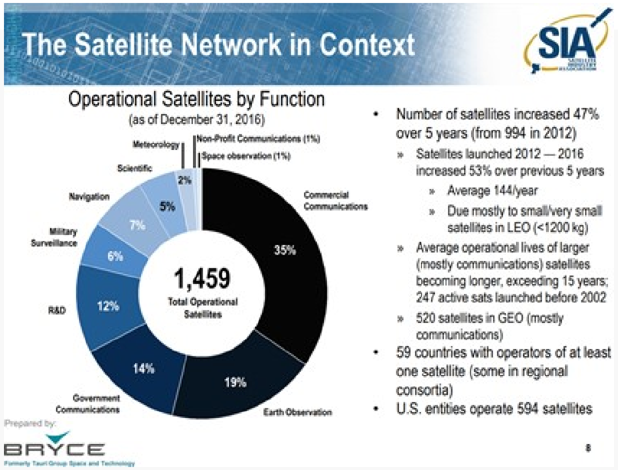
Mission Reliability is an important value

$$R_{mission}\ (R_m) = R_{launch} * A_O * R_{flight}$$

Value to Satellite Industry can be used

Δdiameter:	diameter:	Satellite Benefit (value of payload)	$ Value (Billions)
	4	Commerical Communications:	$45.69
1	5	Optical Sensing:	$24.80
3	8	Interplanetary Missions:	$6.53
2	10	Astronomical Telescope:	$1.31

Human Exploration:		(measured by using % of US GDP)		$ Value:	
National Renown:		0.06	=	$	1,116,000,000,000.00
Extended Science		0.1	=	$	1,860,000,000,000.00
Technological Gains:		0.056	=	$	1,041,600,000,000.00
Medical Advances:		0.1	=	$	1,860,000,000,000.00



Launch Vehicle Value Model

- **Launch Vehicle Value based on 3 factors (currently)**
  - Value is not cost!!!! It includes cost.
  - Industry Value

Launch Vehicle Value	=	Benefit -Ct	
Value to Scientific Uses	=		$63,008,431,752.36
Value to Commerical Services	=		$20,584,576,252.36
Value to Resource Mining	=		($2,252,876,665.31)
Value to Human Exploration	=	$	2,936,540,961,752.35
Total Value	=		$3,017,881,093,091.75

  - Mission Reliability (96%)
    - $V_2 = (R_m)(Value\ of\ Satellite\ Benefit)$
    - $V_L = (1 - R_m)(Value\ of\ Satellite\ Benefit)$
    + Unit Cost + Satellite Cost

V2(Commerical Communication)	=	$43,859,739,840.00
V2(Optical Sensing	=	$23,809,573,056.00
V2(Interplanetary)	=	$6,265,677,120.00
V2(Astronomical Telescope)	=	$1,253,135,424.00
total V2	=	$75,188,125,440.00

Value Lost from Failed Mission		
V(L)(Commerical Communication	=	$7,995,274,012.95
V(L)(Optical Sensing	=	$6,538,291,607.03
V(L)(Interplanetary)	=	$3,732,182,001.85
V(L)(Astronomical Telescope)	=	$2,930,436,400.37
total Value Lost	=	$21,196,184,022.20

  - Payload Accommodation
    - $V_3 = \Delta diameter * \left(\frac{\Delta value\ of\ payload}{meter}\right)$

Satellite Benefit ($B)	4 meters	5 meters	8 meters	10 meters
Commerical Communications:	$45.69	$45.69	$45.69	$45.69
Optical Sensing:	$12.40	$24.80	$24.80	$24.80
Interplanetary Missions:	$2.61	$5.87	$6.53	$6.53
Astronomical Telescope:	$0.00	$0.00	$0.13	$1.31
Total:	$60.70	$76.36	$77.15	$78.32

Launch Vehicle Value	Value
Revenue Value (V1)	$3,017,881,093,091.75
Mission Reliability Value (V2)	$66,294,779,977.80
Payload Size Value (10m Fairiing) (V3)	$78,320,964,000.00

System Physics and System Integrating Physics

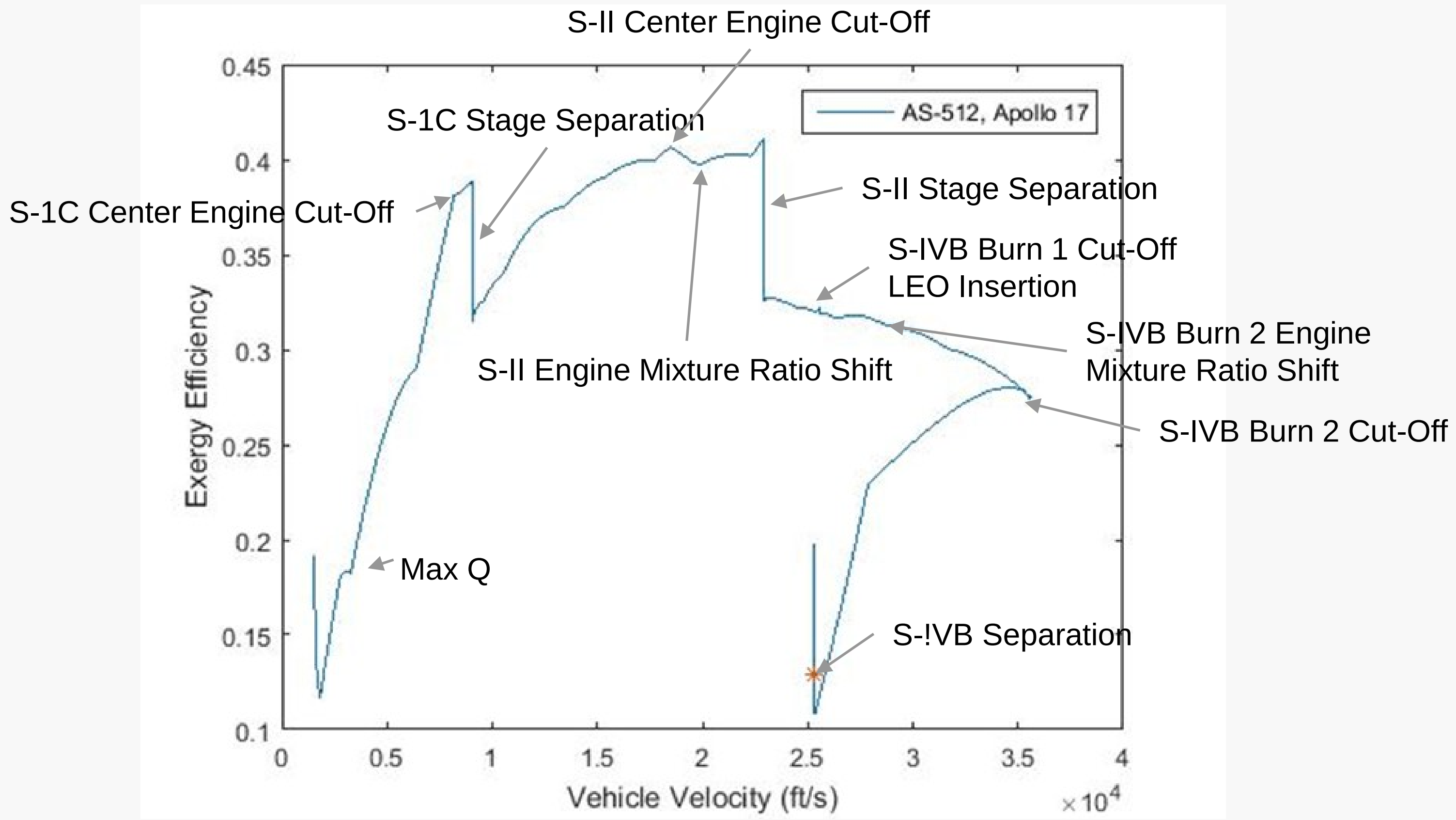
Goal: Utilize the key system physics to produce an elegant system design

System Integrating Physics

- **Consortium is researching the significance of identifying and using the System Integrating Physics for Systems Engineering**
  - First Postulate: Systems Engineering is Product Specific.
  - States that the Systems are different, and therefore, the Integrating Physics for the various Systems is different
- **Launch Vehicles**
  - Thermodynamic System
- **Spacecraft**
  - Robotic
    - Integrated through the bus which is a thermodynamic system
      - Each Instrument may have a different integrating physics but integrates with the bus thermodynamically
  - Crew Modules
    - Integrated by the habitable volume (i.e., ECLSS)
      - A thermodynamic system
  - Entry, Descent, and Landing (EDL)
    - Integrated by thermodynamics as spacecraft energy is reduced in EDL
- **Other Thermodynamic Systems**
  - Fluid Systems
  - Electrical Systems
  - Power Plants
  - Automobiles
  - Aircraft
  - Ships
- **Not all systems are integrated by their Thermodynamics**
  - Optical Systems
  - Logical Systems
    - Data Systems
    - Communication Systems
  - Biological Systems
- **System Integrating Physics provides the engineering basis for the System Model**

Launch Vehicle System Exergy Efficiency

$$\Delta m_{prop} \sum_{stage} \left(h_{prop} + \frac{V_e^2}{2} \right) - X_{des} = \left(M_{vehicle,final} \frac{V_{vehicle,final}^2}{2} - M_{vehicle,initial} \frac{V_{vehicle,initial}^2}{2} \right) + \left(\frac{G M_E M_{vehicle,initial}}{r_{altitude,initial}} - \frac{G M_E M_{vehicle,final}}{r_{altitude,final}} \right).$$



Spacecraft Integration Model

- $\Delta \boldsymbol{m}_{\boldsymbol{propellant,engine}} \left(\boldsymbol{h}_{\boldsymbol{prop,engine}} + \frac{\boldsymbol{V}_{\boldsymbol{e,engine}}^{\boldsymbol{2}}}{\boldsymbol{2}} \right) + \Delta \boldsymbol{m}_{\boldsymbol{propellant,thruster}} \left(\boldsymbol{h}_{\boldsymbol{prop,thruster}} + \frac{\boldsymbol{V}_{\boldsymbol{e,thruster}}^{\boldsymbol{2}}}{\boldsymbol{2}} \right) + \sum_t (\sigma A e (T_{radiator}^4 -$

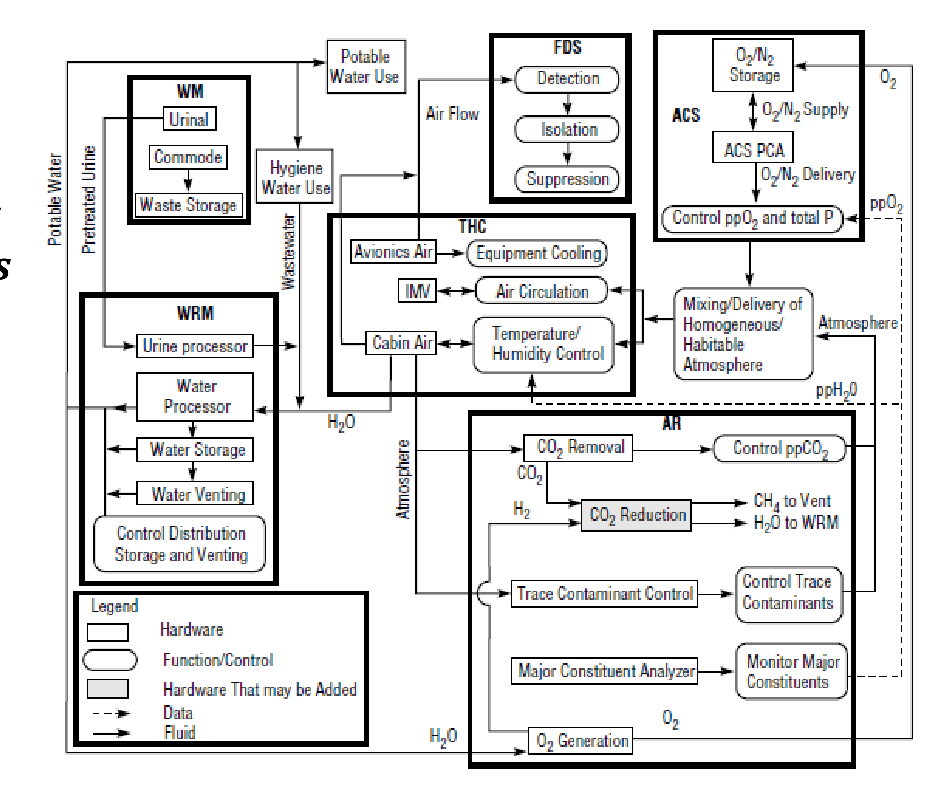
Crew Module Exergy Balance: ISS ECLSS

- $$\Delta X_{ECLSS} = \sum_{process} m_{fluid}\left(\left(h_{final}-h_{cabin}\right) - T_{cabin}\left(s_{final}-s_{cabin}\right) + \left(\frac{V_{final}^2}{2}\right)\right) - \sum_{process} m_{fluid}\left(\left(h_{initial}-h_{cabin}\right) - T_{cabin}\left(s_{initial}-s_{cabin}\right) + \left(\frac{V_{initial}^2}{2}\right)\right) = \sum\left(1-\frac{T_{cabin}}{T_{crew}}\right)Q_{crew} - \sum\left(\frac{T_{cabin}-T_{coolant}}{T_{coolant}}\right)Q_{TMS} + \sum W_{EPS} - P_{cabin}\left(Vol_{final}-Vol_{initial}\right) + m_{in}\left[\sum(h_{in} -\right.$$

- $\Delta X_{ECLSS} = \Delta X_{ACS} + \Delta X_{AR} + \Delta X_{THC} + \Delta X_{WRM} + \Delta X_{WM}$

- ***Incorporate chemcial process product and reactant enthalpy and entropy terms***

- **Calculate Efficiency**



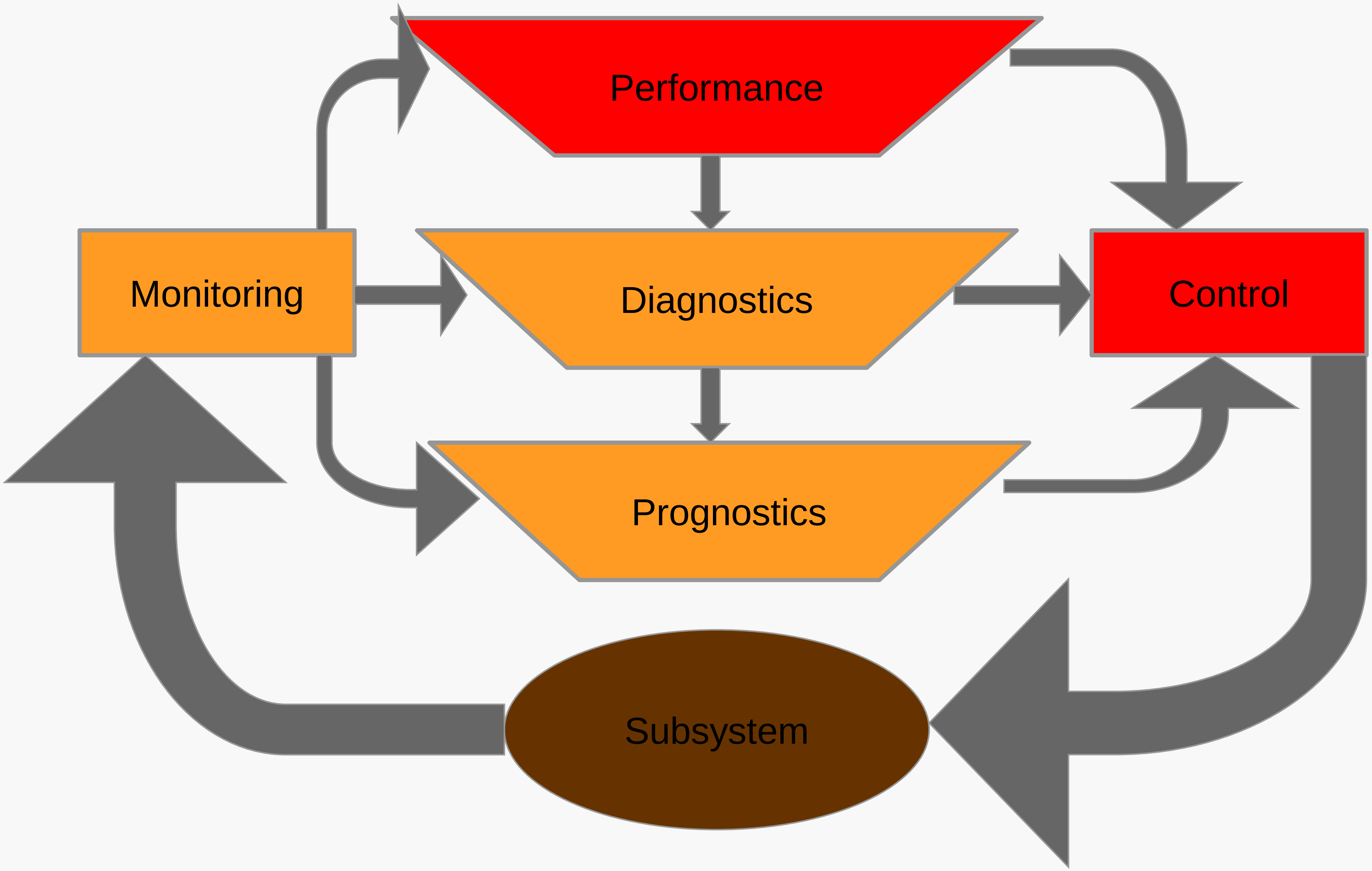
System Autonomy

Goal: Establish system interfaces to provide autonomy algorithms with system information necessary and sufficient to manage system

Autonomy in Context: What and Why?

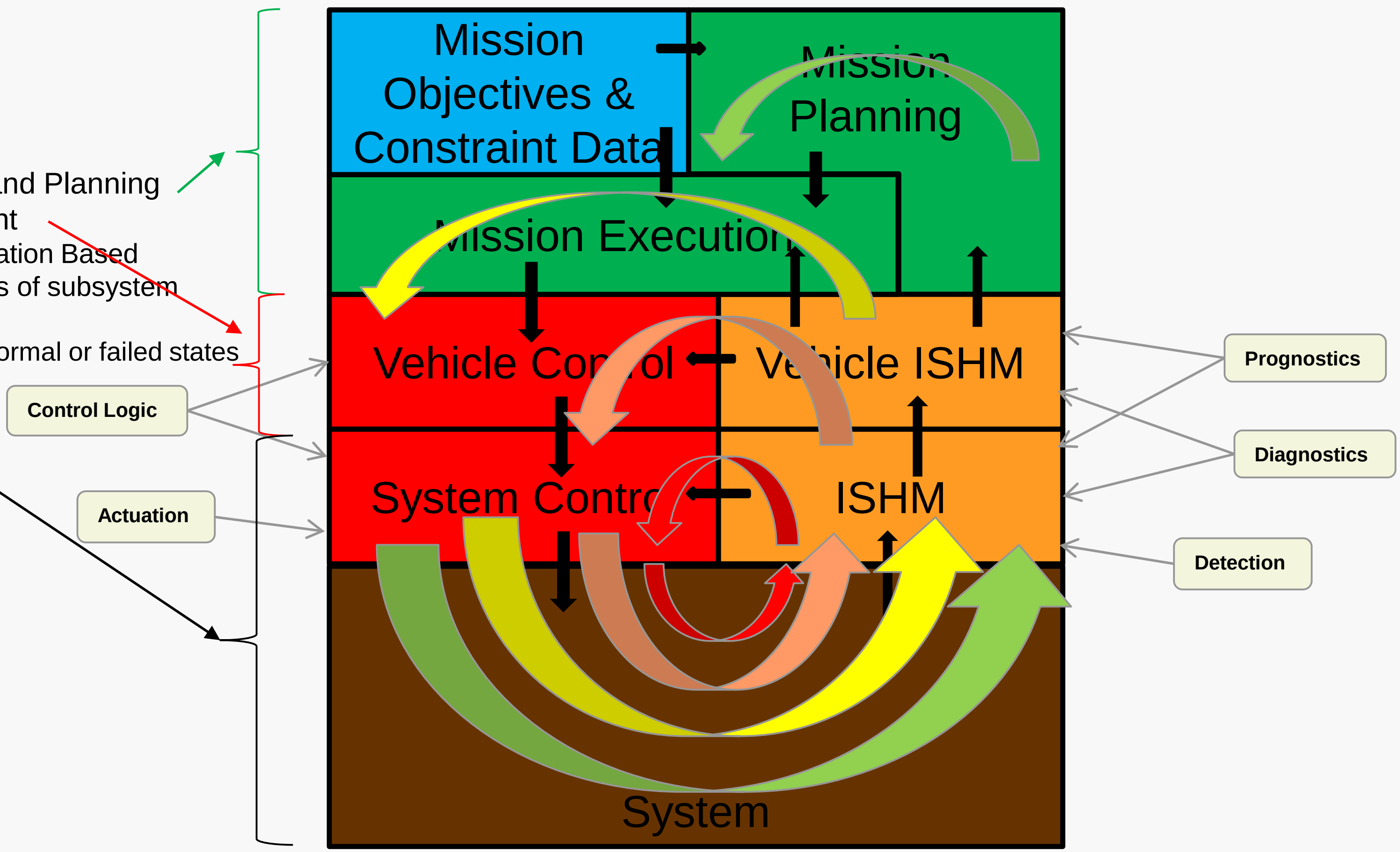
- **Spacecraft and Surface System Autonomy is the enabling capability for Human Exploration beyond Lunar Sortie Missions**
  - Autonomy is necessary for complex system operations
  - Timely response to unplanned or unscheduled events
- **Propulsion, Structure, Thermal Conditioning, ECLSS, Electrical Power, Avionics, RCS, Communication are all understood sufficiently to allow engineered solutions to be reliably produced**
  - Challenges do exist in terms of Space Environmental Effects, efficiency, compact size
    - Radiation Hardened computer processors needed
  - Physics and demonstrated solutions are available from which to engineer a vehicle
- **Operations are sufficiently understood for terrestrial based execution, not on-board execution**
  - Manual operations provide a rich knowledge base of planning and execution processes
  - Manual operations have a generic template (derived from Apollo/Saturn) applied uniquely to each spacecraft
  - Terrestrial based manual operations will not support operations beyond 5 light minutes from Earth
- **Autonomous Operations are essential to Human Exploration of the Solar System**

Subsystem Management Functions for System Control



Autonomy System Stack

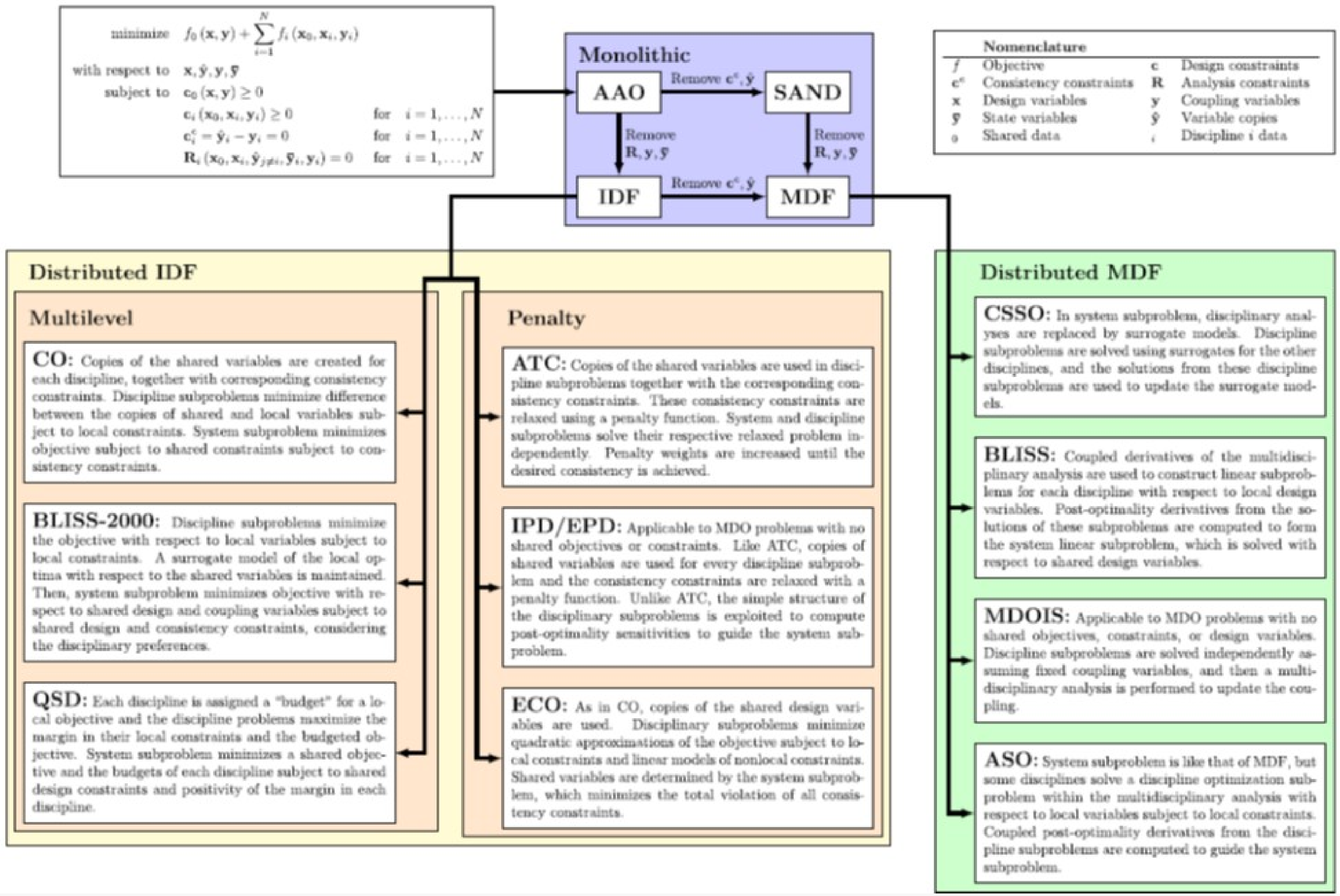
- **3 Levels**
  - Mission Execution and Planning
  - Vehicle Management
    - Subsystem Integration Based
    - Physics form basis of subsystem interactions
      - Form basis of normal or failed states
  - Subsystem Level
    - Physics based



System Design and Optimization

Goal: Apply system design and optimization tools to understand and engineer system interactions

Multidisciplinary Design Optimization



Martins, J. R R. A., Lambe, A. B., “Multidisciplinary Design Optimization: A Survey of Architectures”, AIAA Journal, Vol. 51,No. 9, September 2013, pp 2049 – 2075

Engineering Statistics

Goal: Utilize statistical methods to understand system uncertainties and sensitivities

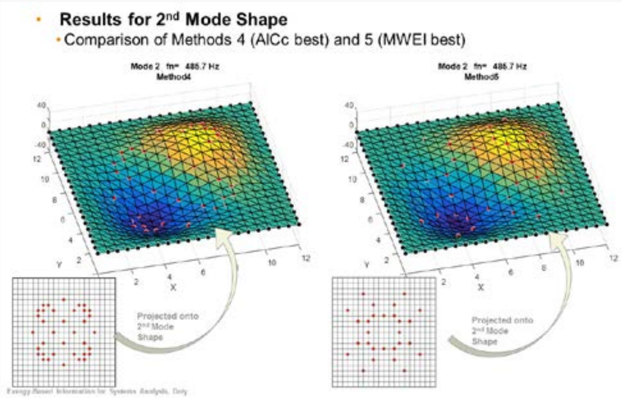
Systems Engineering makes use of Frequentist Approaches, Bayesian Approaches, Information Theoretic Approaches as appropriate

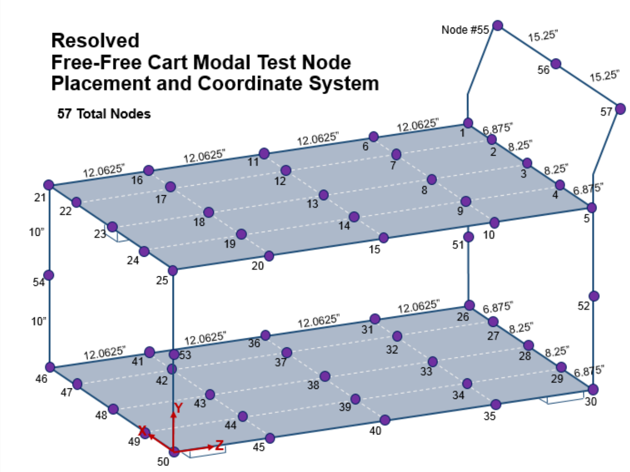
Optimal Sensor Information Configuration

- **Applying Akaike Information Criteria (AIC) corrected (AICc) to assess sensor coverage for a system**

$$AICc(F) = -2\left(I^{KL}(F|G)\right) + 2K + \frac{2K(K+1)}{n-K-1}$$

- **Two Views of Information Content**
  - AIC Information
    - Information is viewed as the number of meaningful parameters
      - Parameters with sufficient measurements to be reasonable estimates
  - Fisher Information Matrix
    - Defines information as the matrix of partial second derivatives
      - Information is the amount of parameters with non zero values (so provides an indication of structure)
      - This value converges to a maximum as the number of parameters goes to infinity
      - Does not contain an optimum, always increases with added parameters
- **AIC/AICc has an adjustment factor to penalize sensor arrangements where:**
  number of sensors < 3x(number of measurements)
- **Provides an optimization tool for use with System Models**





Flat Plate FEA Analysis and Akaike Information Criterion (AIC)

- **Results**
  - Sources of AICc:
    - Bias: ∝ mean square error (MSE)
    - Total Corrections: “penalties” for over-fitting

MWEI ‘Best’ is worst bias and $(\boldsymbol{AICc})_{\boldsymbol{Local}}$

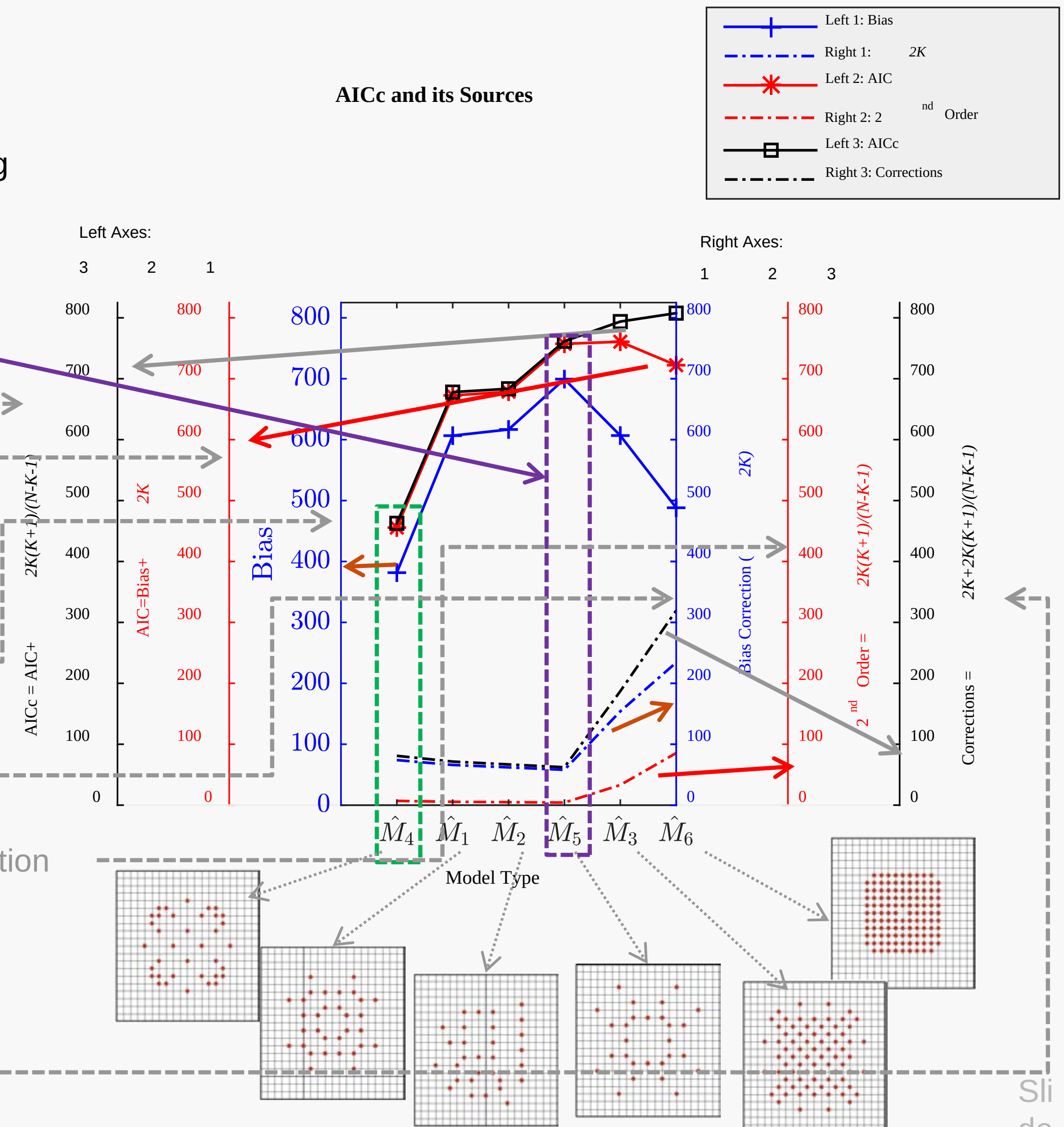
$$\boldsymbol{AICc} = \boldsymbol{N}\log(\boldsymbol{MSE}) + \boldsymbol{2K} + \frac{\boldsymbol{2K(K+1)}}{\boldsymbol{N-K-1}}$$

Bias

Bias Correction

Small Sample Correction

Total Corrections



Sli
de
-

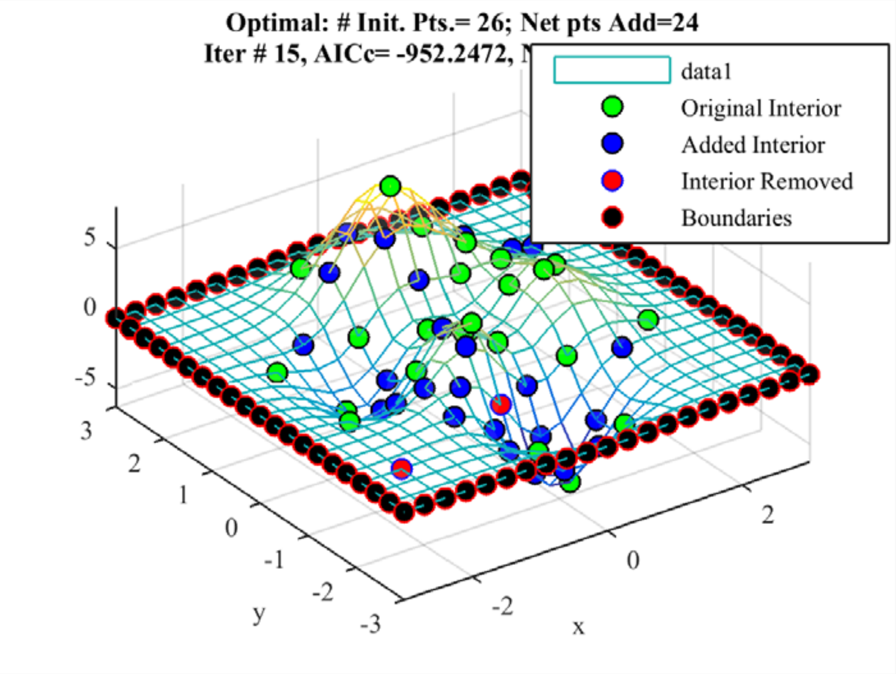
Verification Process

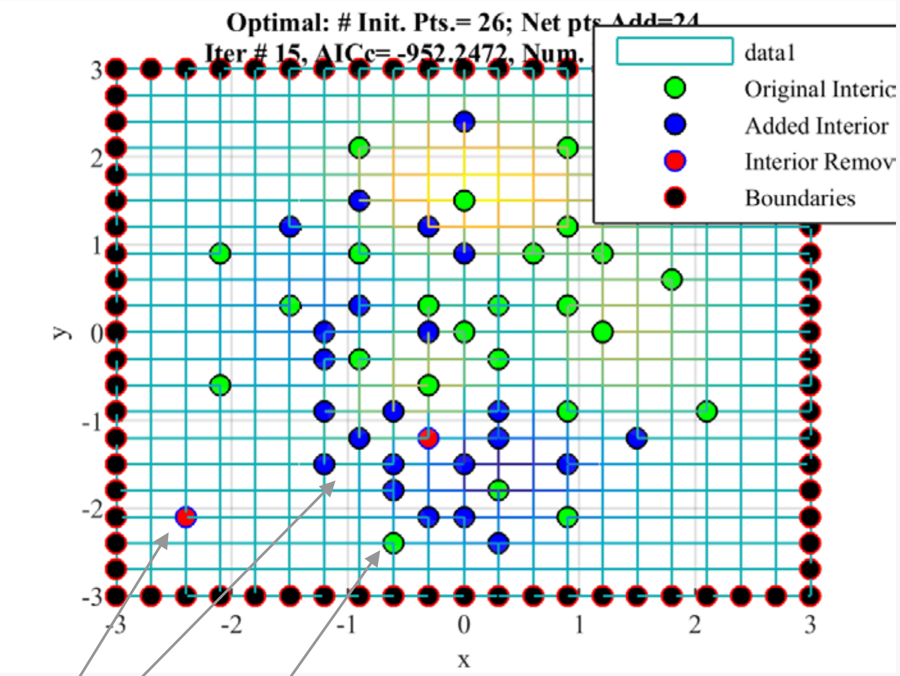
Method 1: 'Intelligent' Guess

- **Final Solution:**

Overlaid on Peaks

Projected to XY plane





Note:

2 initial guesses 'removed' (red)
NEW points added (blue)
MOST initial guesses 'survive' (green)

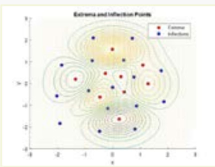
Sensor Location

- **Sensor Placement is determined by locations of highest residual error**
  - Indicates lowest level of information about the system
- **System model allows determination of highest residual error location**
  - Must properly model physics of the system to be measured and associated interactions
  - Placing the first sensor here changes the information available and biases all other locations
  - Provides keystone for locating sensors appropriately
- **Provides an objective method to determine proper sensor measurement locations**

Methods of System Integration

Goal: System Design and Analysis

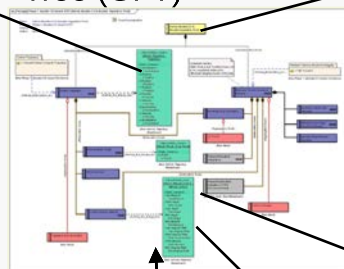
System Models Contain an Understanding of the System



Engineering Statistics

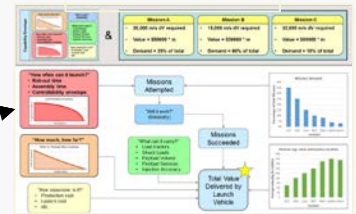
State Variables

Goal Function Tree (GFT)



Goals

Value Model



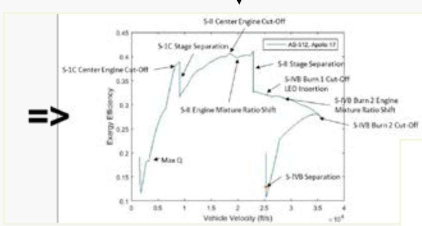
System Functions & State Variables

System Functions & State Variables

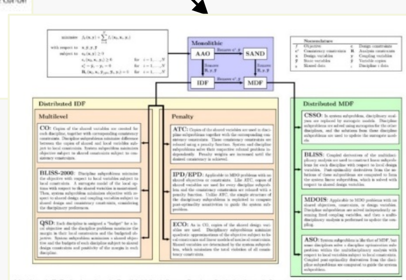
System State Transition Model

Discipline Physics Models

=>



System Integrated Physics Model (System Exergy)



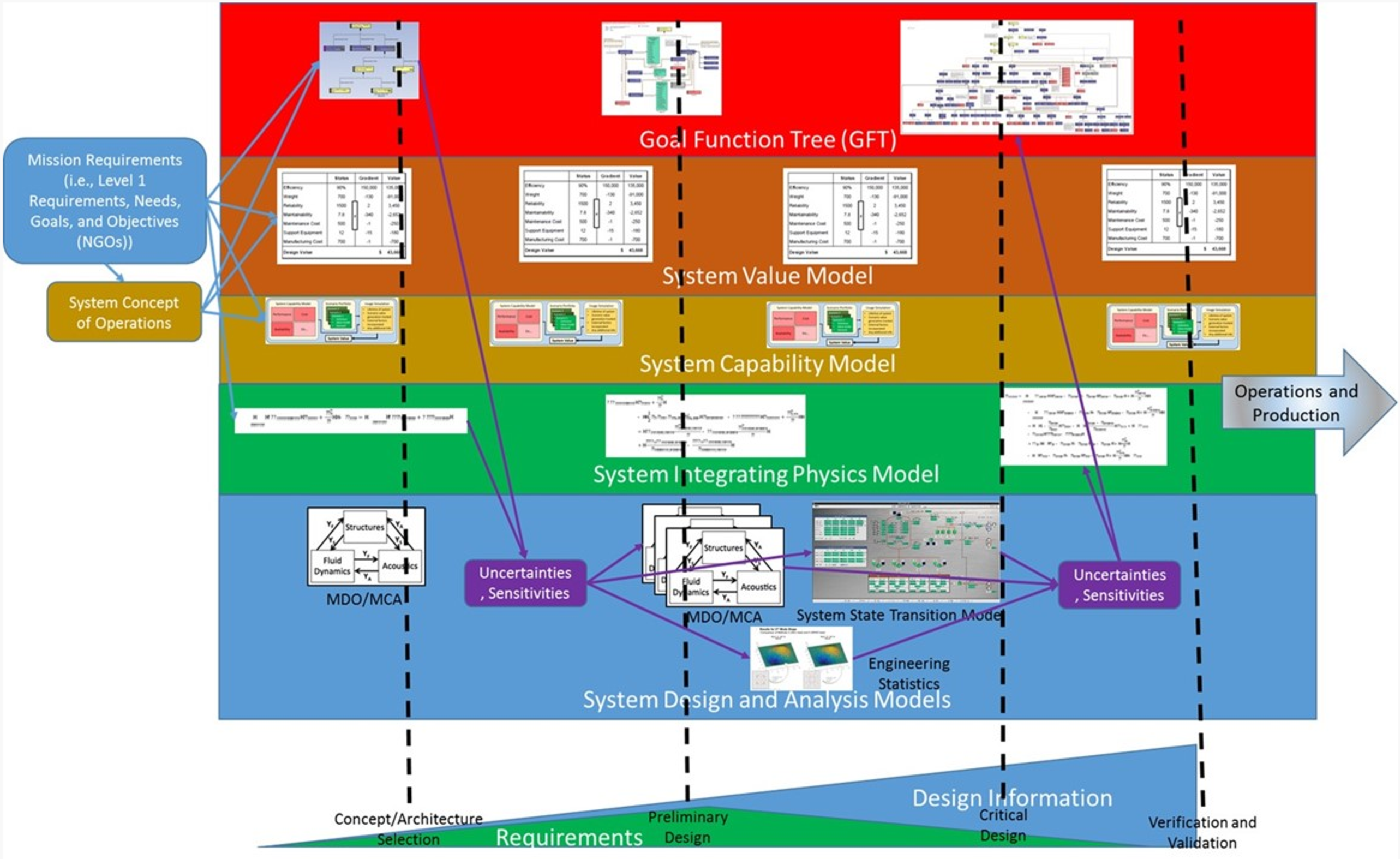
Martins, J. R R. A., Lambe, A. B., "Multidisciplinary Design Optimization: A Survey of Architectures", AIAA Journal, Vol. 51,No. 9, September 2013, pp 2049 – 2075

Multidisciplinary Design Optimization (MDO)

- Allow systems engineers to:
  - Define system functions based on the system state variables
  - Understand stakeholders expectations on system value (i.e., capabilities)
  - Integrate discipline engineering models into a system level physics based model (e.g., system exergy)
  - Design and Analyze system responses and behaviors at the System level

- MagicDraw Enterprise (SysML)
- Matlab
- Matlab StateFlow
- Microsoft Excell

System Design and Integration



Methods of Engineering Discipline Integration

Goal: Understand How Organizational Structures influence Design and Operations Success of Complex Systems

Sociological Concepts in Systems Engineering

- **Specification of Ignorance is important in the advancement of the understanding of the system**
- **Consistent use of Terminology is important for Communication within the Organization**
- **Opportunity Structures**
  - Provide opportunity to mature ideas
    - Task teams, working groups, communities of practice, etc.
- **Socially Expected Durations will exist about the project**
- **Both Manifest and Latent Social Functions exist in the organization**
- **Social Role Sets**
  - Individuals have a set of roles for their position
- **Cultural Subsets will form**
  - i.e., disciplines can be a subset within the organization
  - Insider and Outsider attitudes can form
    - Be Aware of the Self-Fulfilling Prophecy, Social Polarization
- **Reconsiderations Process (i.e., Reclama Process)**
  - Provides ability to manage social ambivalence
  - Must be able to recognize social beliefs that may be contributing to the disagreement
  - Helps to avoid putting people in to social dysfunction or complete social anomie
    - Conformity
    - Innovation
    - Ritualism
    - Retreatism
    - Rebellion

Information Flow

- **Information Flow through a program/project/activity is defined by Information Theory**
  - Organizational communication paths
  - Board Structure
- **Decision Making follows the First Postulate**
  - Decision Process is specific to the decision being made
  - Tracked 3 SLS CRs, with 3 separate task team processes, all had equally rated effectiveness

SLS SE&I MANAGEMENT STRUCTURE

June 9, 2014 version

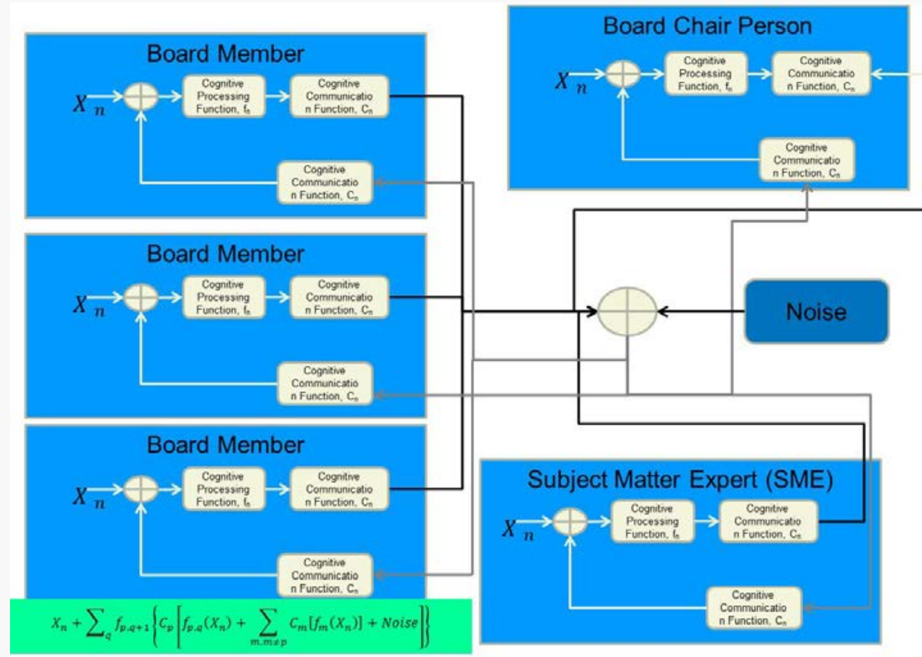
( ) = OPR   [ ] = ORGANIZATIONS MAPPED TO DISCIPLINE

SLS PROGRAM OFFICE ORGANIZATION	CHIEF ENGINEERS OFFICE ORGANIZATION	Systems Engineering (EV01) [EV70, EE12]	Vehicle Management (EV40) [EV40]	Structures & Environments (StE) (EV30) [EV30, ER40, ES21, ES22]	Propulsion (ER01) [ALL ER EXCEPT ER40]	Production (EM01) [ALL EM]	Integrated Avionics and Software (ES01) [ALL ES EXCEPT ES21,ES22]	Operations (EO01) [ALL EO, ES10]	Test (ET01) [ALL ET]	S&MA (QD01) [ALL QD]
SLS Program Manager SLS Program Deputy Manager SLS Associate Program Manager Assistant PM Procurement	Program Chief Engineer Program Deputy Chief Engineer SE&I Technical Manager Assistant CE for Affordability Tech. Assist. Cross Program Integ. Tech. Assist. Ext. Interface Integ.	LSE: EV01 Alt: EV70 Alt: EV73	DLE: EV40 Alt: EV40	DLE: EV30 Alt: EV30	DLE: ER01 Alt: ER51 Alt: ER24	DLE: EM03 Alt: EM03 Alt: EM03	DLE: ES30 Alt: ES01	DLE: EO04 Alt: EO04	DLE: ET10 Alt: ET10	Program CSO Deputy CSO QD02 SE&I S&MA Lead QD35
Stages Element Manager Stages Deputy Element Manager - Avionics Manager - Core Stage Manager - Integration Manager	Stages Chief Engineer Stages Deputy Chief Engineer Stages Deputy CE - Avionics Stages Deputy Chief Engineer - Test	EV70 Alt: EV71	EDLE: EV41	EDLE: EV34	EDLE: ER22	EDLE: EM03 Alt: EM32	EDLE: ES12	EDLE: EO40	EDLE: ET10	QD33
Booster Element Manager Booster Deputy Element Manager - Control Systems Manager - Assem & Struct Systems Manager - Motor/BSM ASM - Booster CEI/Interface Mgr	Booster Chief Engineer Booster Deputy Chief Engineer	ER50	EDLE: EV40	EDLE: ER40	EDLE: ER51	EDLE: EM03	EDLE: ES12	EDLE: EO40		QD31
Engines Element Manager Engines Deputy Element Manager	Engines Chief Engineer Engines Deputy Chief Engineer	ER20	EDLE: EV43	EDLE: ER41	EDLE: ER21	EDLE: EM03	EDLE: ES12	EDLE: ER21		QD32
Spacecraft/Payload Integration and Evolution (SPIE) Office Manager SPIE Deputy Manager	SPIE CE SPIE Deputy CE	EV70 Alt: EV70	EDLE: EV41	EDLE: EV30	EDLE: ER23	EDLE: EM03	EDLE: ES10	EDLE: EO40	EDLE: ET30	QD22
	SPIE CE SPIE Deputy CE				EDLE: ER01 Alt: ER21	EDLE: EM03				QD31

- **Margin is maintained by the Organization, not in the margin management tables**
  - Biased Information Sharing
  - Margin Management is focused on Managing the Disciplines (informed by the System Integrating Physics)
- **SLS Organizational Structure was defined by the LSE as a recommendation to the Chief Engineer and the Program Manager**

Decision Structure Information Flow

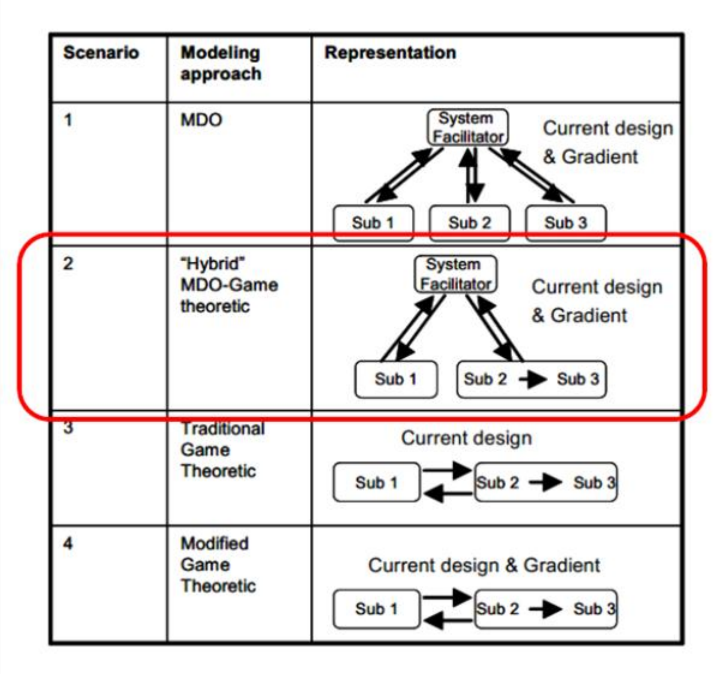
- **Information Theory Model**
  - Information Theory can be used to understand decision making structures and information flow
  - $\bar{I} = H = -\sum_n p_n \log p_n$
- **Practitioner's Guidance**
  - Understand and define the scope of each needed decision body



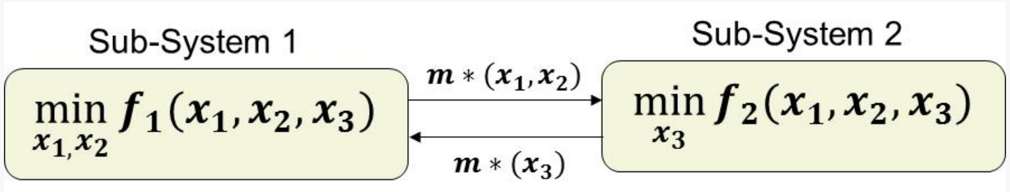
  - Ensure that each decision body has all affected or contributing disciplines represented, including understanding of the types and magnitudes of uncertainties affecting decisions within that decision body's scope, but no more
    - $H(p_1, p_2, \dots, p_n, q_1, q_2, \dots q_m) \geq H(p_1, p_2, \dots, p_n)$
  - Minimize the number of decision bodies based on scope. The efficiency of the structure decreases with distributed and overlapping scopes.
    - $H(S, D, X, Y, Z) \leq H(S) + H(D) + H(X) + H(Y) + H(Z)$

SLS Organizational Structure Modeling

- **Interviewed 12 Marshall engineers/designers (w/J. Shelton)**
  - Understand strategies used to integrate subsystems with each other
- **Common strategy across subsystems – margins**
  - Keep some percentage of a parameter in “back pocket” as hedge for future negotiations
  - Biased Information Sharing
  - (Here, “margins” different from “safety margin”)
- **How does maintaining a margin affect optimality of the final design?**
  - Model as simple 2 Player System with 3 design parameters
  - 15 problem test suite

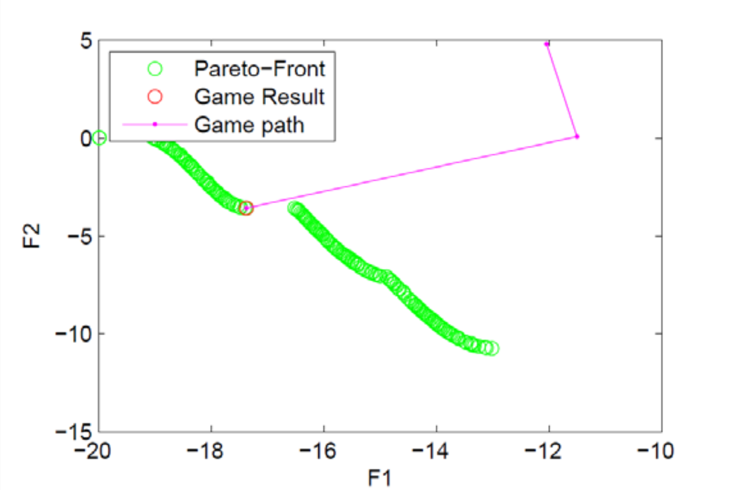


Scenario	Modeling approach	Representation
1	MDO	System Facilitator; Current design & Gradient; Sub 1, Sub 2, Sub 3
2	"Hybrid" MDO-Game theoretic	System Facilitator; Current design & Gradient; Sub 1, Sub 2, Sub 3
3	Traditional Game Theoretic	Current design; Sub 1, Sub 2, Sub 3
4	Modified Game Theoretic	Current design & Gradient; Sub 1, Sub 2, Sub 3

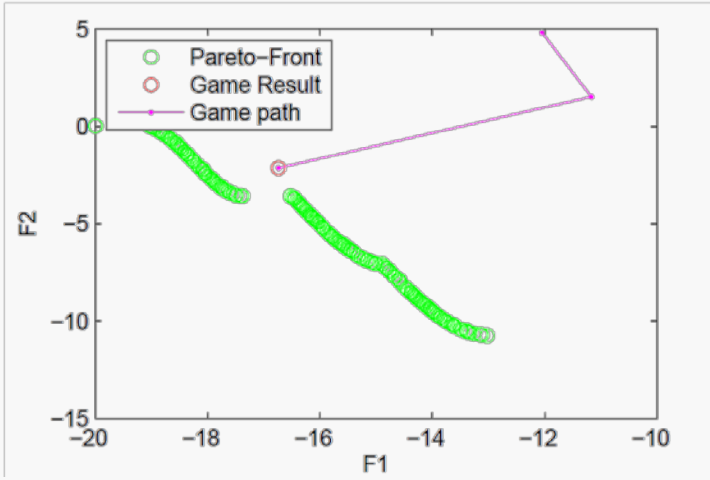


Simulation Results

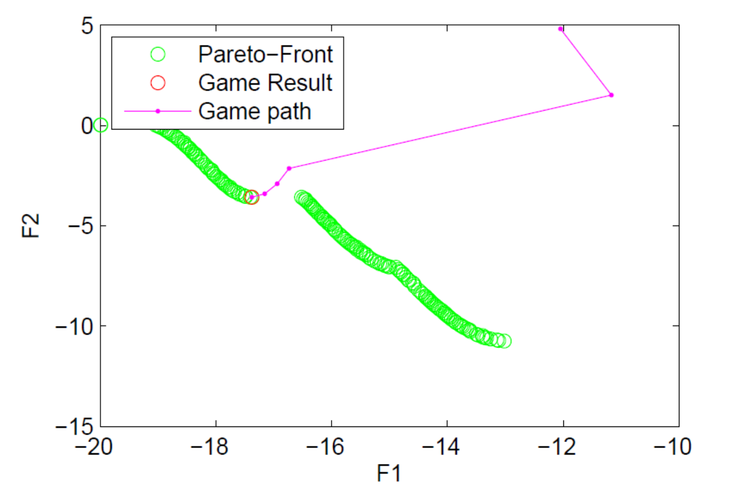
No margin : $m = 1$



Static margin, m= 1.3

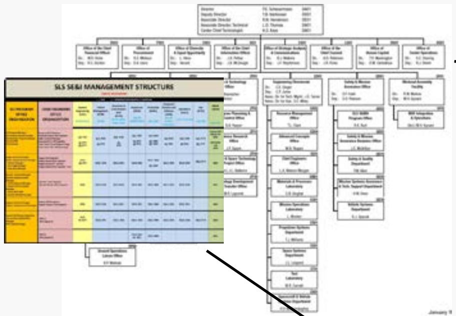


Descending margin, m=1.3−.1$*i$ until m=1



- No margin condition reaches optimality quickest
- Descending margin still reaches optimal, but requires more iterations
- Margins are an issue
  - Interviews highlight real-world consequences
  - Simulations quantify extent of the problem
  - Still possible to achieve optimal design with descending margin, but takes additional time to achieve

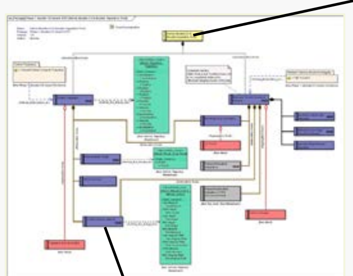
Discipline Integration Models



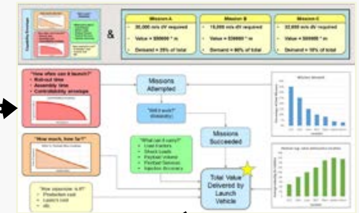
Organizational Structure & Mapping

Organizational Values

Goal Function Tree (GFT)



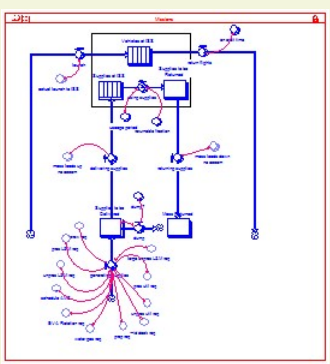
Value Model



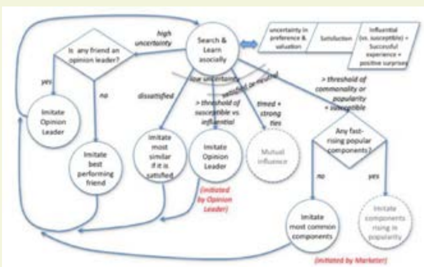
Goals

Value Attributes

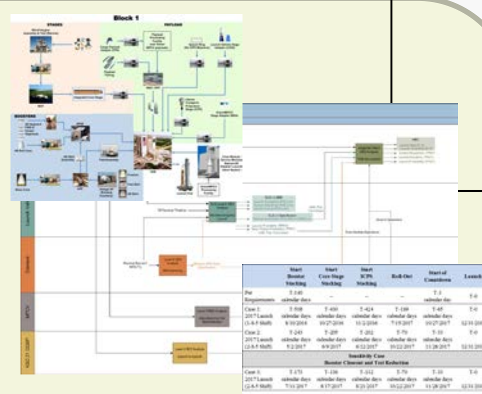
System Functions



System Dynamics Model



Agent Based Model (ABM)



Discrete Event Simulation

- Allow systems engineers to:
  - Understand information flow through the development and/or operations organization
  - Integrate discipline information into a system level design
  - Analyze information flow, gaps, and blind spots at the System level

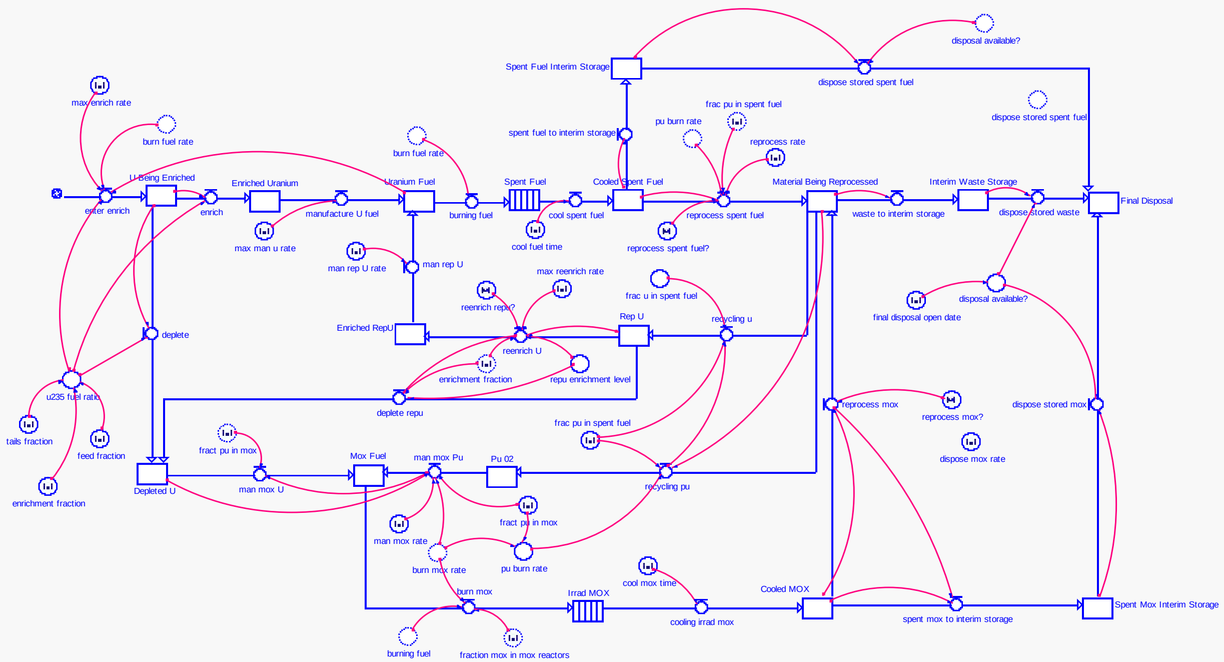
- MagicDraw Enterprise (SysML)
- Matlab
- Matlab StateFlow
- JAVA
- Anylogic
- Extend

System Dynamics

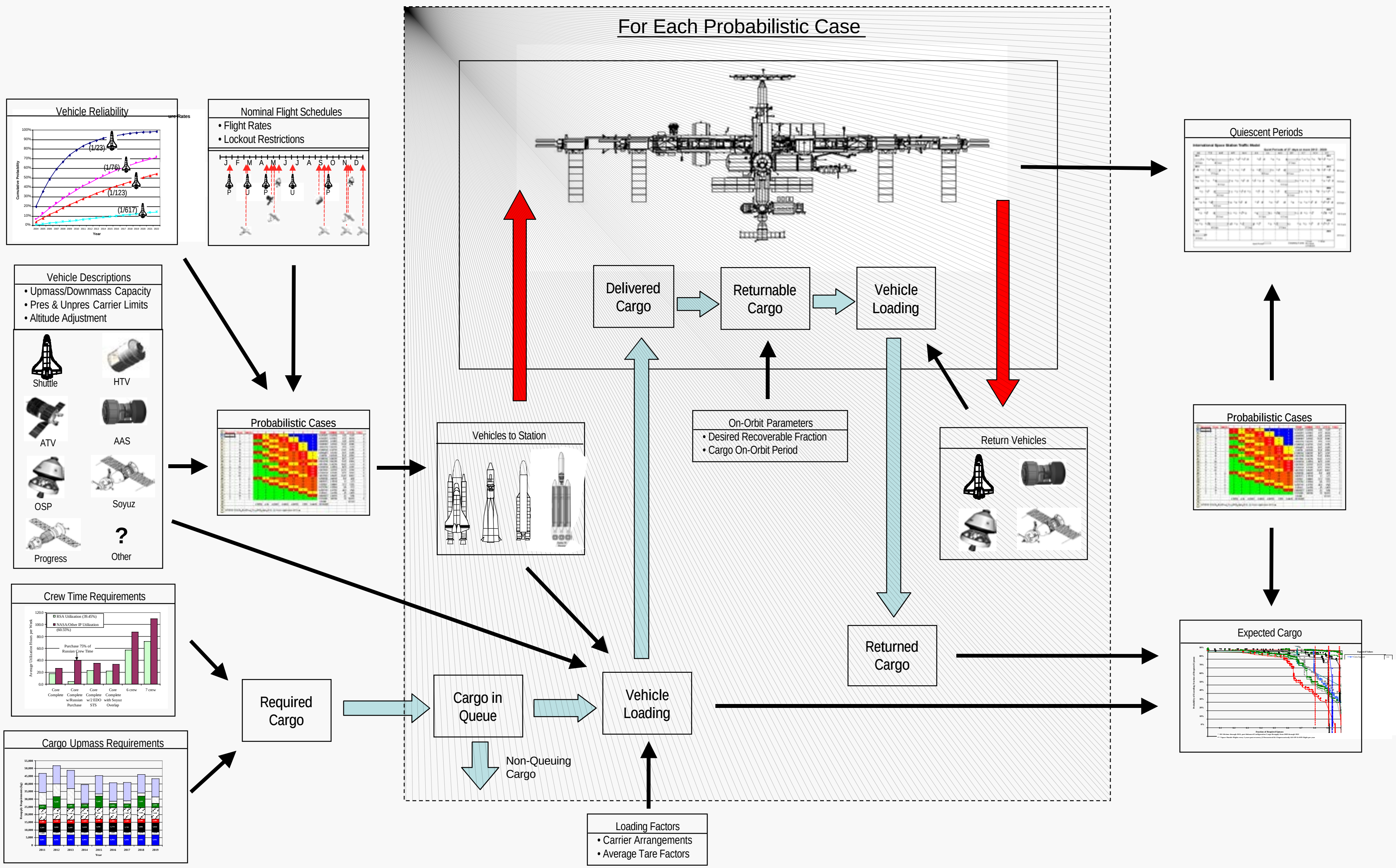
Goal: Understand how information about the system flows through the organization and into the design and operations

Tools and Methodologies

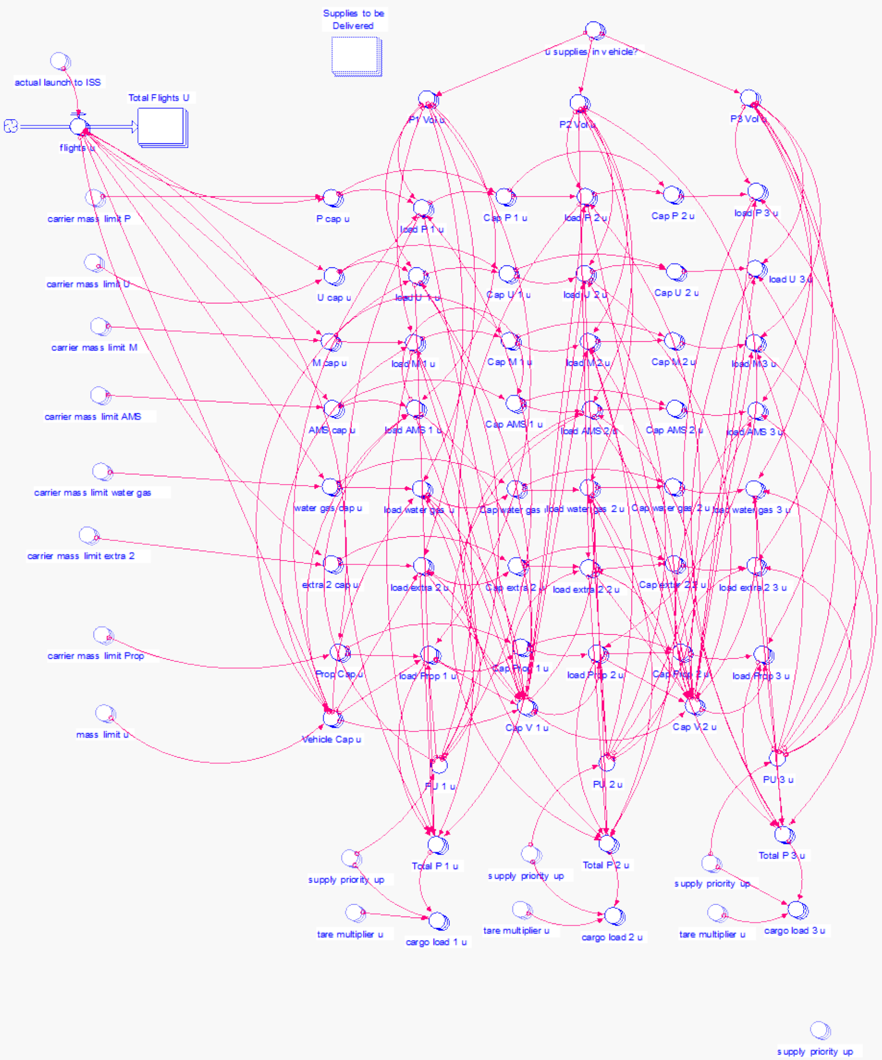
- Tools and techniques have been developed using the System Dynamics methodology that make it possible to efficiently decompose complex systems and to quickly set-up and test models of system operation.
- Tools promote understanding through visual diagramming and modeling.

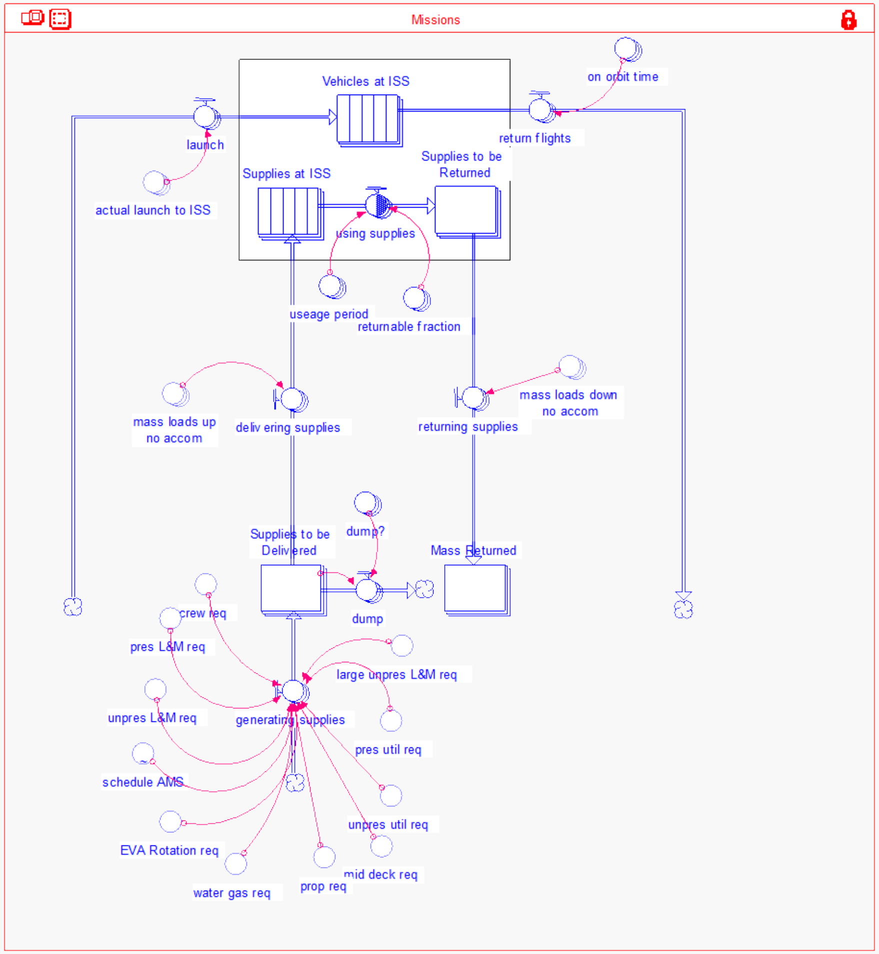


STS-ISS Transportation / Operation Analysis



ISS-STS Transportation / Operations Analysis





Policy and Law Assessments

Goal: Understand How Policy and Law Constrain the Design and Operations of a System and How the System Engineer Should Interpret These Constraints

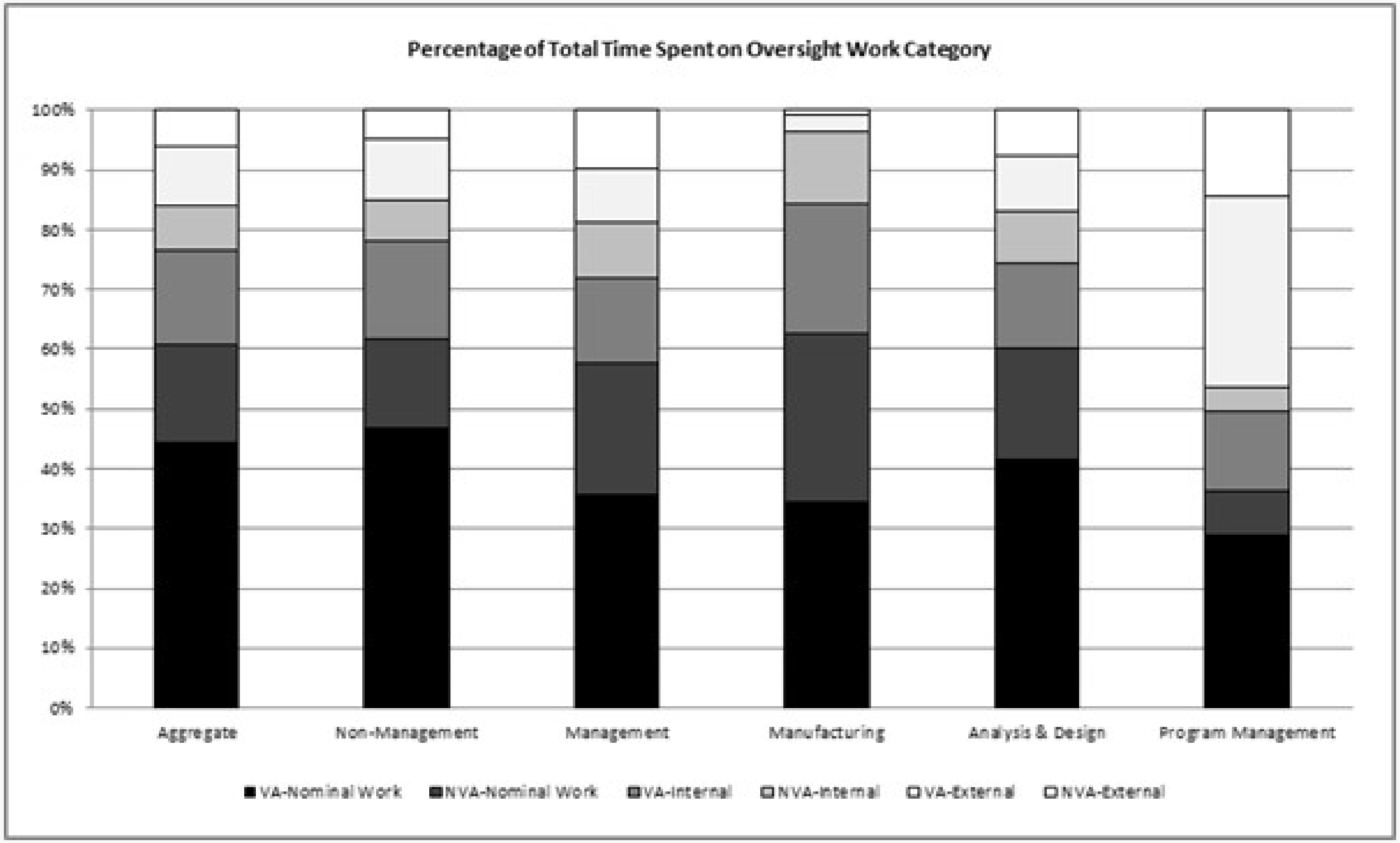
Impact of Government Oversight Time Allocation Study

- Motivation: Industry and government leaders agree that government oversight leads to cost growth, but there is less agreement on how much and through what mechanisms.

- Research Plan:
  - Build an empirical basis for measuring the extent and nature of the impact of oversight
  - Non-invasive “Time Allocation Study:” Statistically valid aggregated observations of how engineers actually spend their time throughout a product’s life cycle.
    - Part One: Collect time-recall diaries to develop a composite list of activities performed
    - Part Two: Survey Population over several months at random times per day to accurately observe amount of time spent on activities

- Data collection is complete and analysis is in process
  - Most non-value added oversight is internal company driven
  - Government generated insight/oversight is a small % of work done (< 10%)
  - Corporate Communication and Administrative work drive non-value added work from viewpoint of practicing systems engineers within the company

Percentage of total time spent on each oversight category



Brainard, S. M., Zsajnfarber, Z., "Understanding the burden of government oversight on engineering work: adding empirical data to the debate", submitted to Space Policy

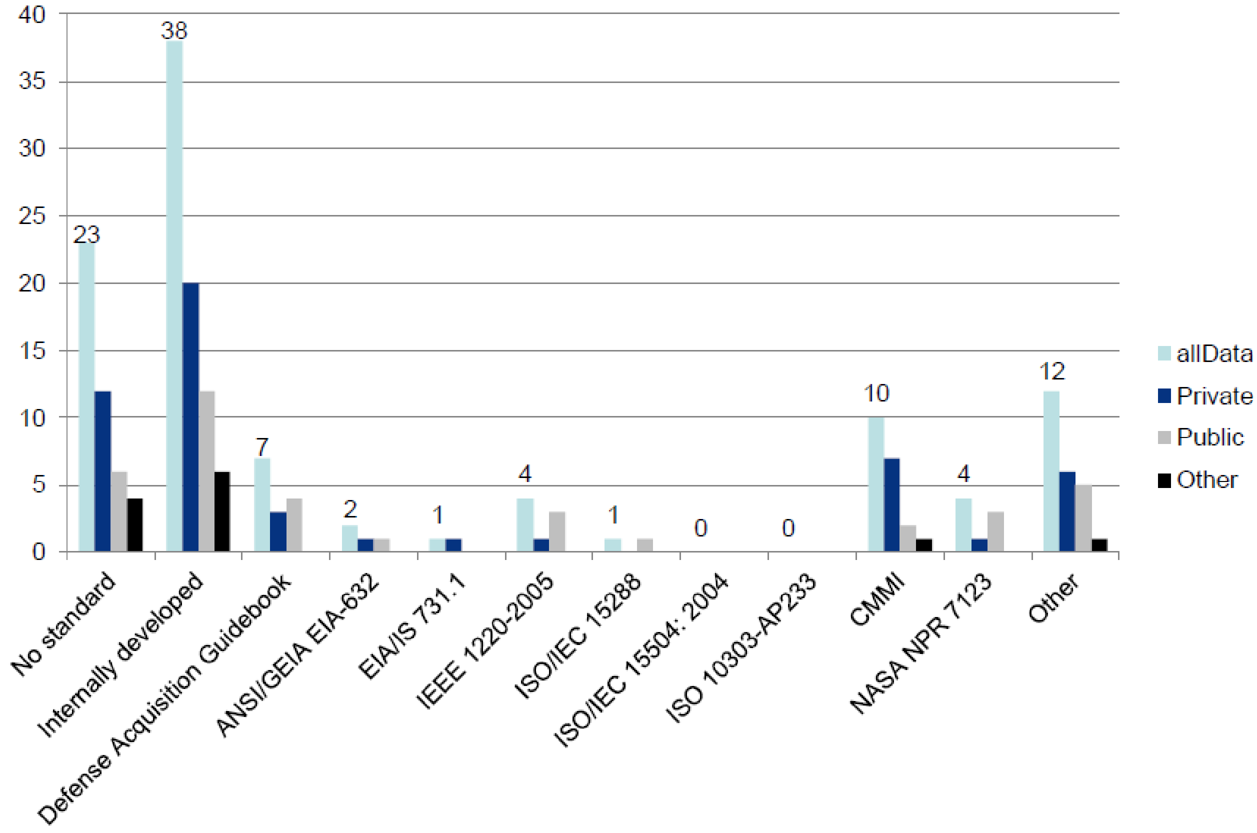
System Engineering Supporting Activities

Process Application and Execution for the Specific System

Processes

- **Well defined in NASA/SP-2007-6105 Rev1, NASA Systems Engineering Handbook**
- **SEMP is essential to capture appropriate application of processes to the specific system**
  - Process application is specific to the system being developed
    - Tailoring is not a special exception, it is the norm

System Engineering Standards in Practice



UAH SE Consortium - Comparing the Relationship between Systems Engineering Process and Project Success in Commercial and Government Research and Development Efforts, 2012 – 2014.

ORIGINAL NASA STUDY AND NEW STUDY COMMERCIAL FOCUSED PROJECTS Correlation of 0.4 or greater noted Project Success and System Engineering Processes	1. Stakeholder Expectations Definition	2. Technical Requirements Definition	3. Logical Decomposition	4. Design Solution	5. Product Implementation	6. Product Integration	7. Product Verification	8. Product Validation	9. Product Transition	10. Technical Planning	11. Requirements Management	12. Interface Management	13. Technical Risk Management	14. Configuration Management	15. Technical Data Management	16. Technical Assessment	17. Decision Analysis
Technical success relative to initial req.									.4	.4							.4
Technical success relative to similar projects						.7	.6		.6								
On schedule relative to original project plan			.4							.6		.4					
On schedule relative to similar projects										.4		.4					
On budget relative to original project plan										.5		.5	.5	.4			
On budget relative to similar projects										.4			.4				
Satisfaction with project management process		.5		.5						.5							
Overall project success (organization view)						.6			.5								
Overall project success (stakeholder view)						.4							.5				

Agriculture
Aerospace
Defense and security
Transportation
Communications
Electronics
Energy
Infrastructure

Processes with > 3 Correlations ≥ .4

Processes with < 3 Correlations ≥ .4

Original Study Correlations

UAH SE Consortium - Comparing the Relationship between Systems Engineering Process and Project Success in Commercial and Government Research and Development Efforts, 2012 – 2014.

ORIGINAL NASA STUDY AND NEW STUDY GOVERNMENT FOCUSED PROJECTS: CHECK

Correlation of 0.4 or greater noted Project Success and System Engineering Processes

	1. Stakeholder Expectations Definition	2. Technical Requirements Definition	3. Logical Decomposition	4. Design Solution	5. Product Implementation	6. Product Integration	7. Product Verification	8. Product Validation	9. Product Transition	10. Technical Planning	11. Requirements Management	12. Interface Management	13. Technical Risk Management	14. Configuration Management	15. Technical Data Management	16. Technical Assessment	17. Decision Analysis
Technical success relative to initial req.	.5		.6				.4		.5		.5	.4	.5			.4	.5
Technical success relative to similar projects	.5		.6	.5	.5		.5			.5	.4					.5	.5
On schedule relative to original project plan	.4	.5								.4						.5	
On schedule relative to similar projects	.5	.4			.4					.5			.5			.5	.5
On budget relative to original project plan	.4						.4			.6						.6	.4
On budget relative to similar projects	.4	.4					.5			.5						.5	
Satisfaction with project management process	.5	.5			.4		.4			.6			.6			.6	.6
Overall project success (organization view)	.5					.5		.5	.5								
Overall project success (stakeholder view)	.6				.4					.4						.5	.4

Processes with > 3 Correlations ≥ .4

Processes with < 3 Correlations ≥ .4

Original Study Correlations

Summary

- **Discussed approach to Engineering an Elegant System**
- **Systems Engineering Framework and Principles**
  - System Integration
  - Engineering Discipline Integration
- **Several methods and tools are available for conducting integrated system design and analysis**
  - System Integration
    - System State Variables
      - Goal Function Tree
      - State Analysis Model
    - System Value Model
    - System Integrating Physics
    - System Autonomy
    - Multidisciplinary Design Optimization (MDO)
    - Engineering Statistics
  - Discipline Integration
    - Sociological Concepts in Systems Engineering
    - Information Flow
    - Systems Thinking (Cognitive Science)
    - Policy and Law
    - System Dynamics Modeling
- **Systems Engineering Approach defined in two documents**
  - "Engineering Elegant Systems: Theory of Systems Engineering"
  - "Engineering Elegant Systems: The Practice of Systems Engineering"
  - Send requests for documents to: michael.d.Watson@nasa.gov