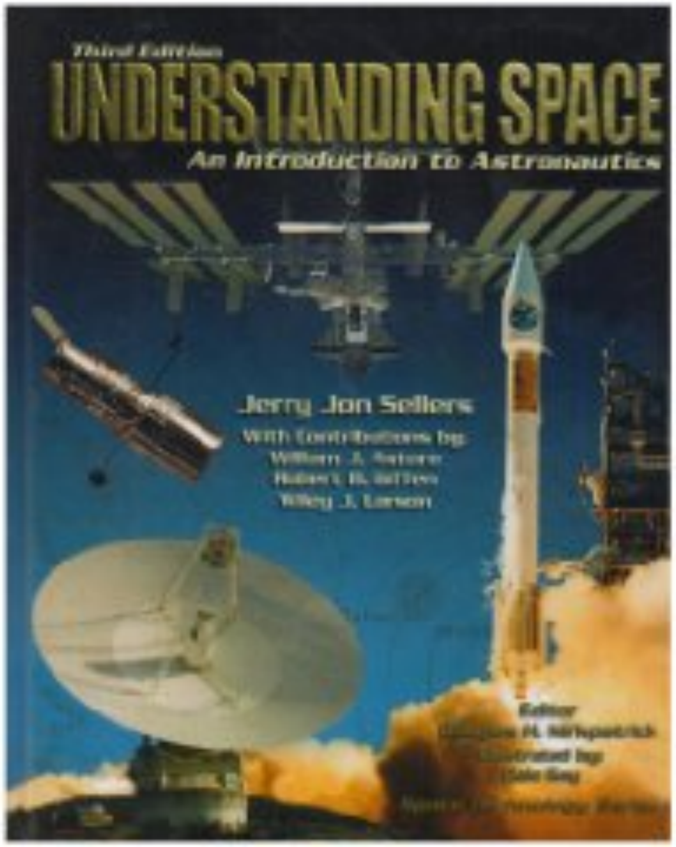
Utah State University

Mechanical & Aerospace Engineering

Introduction to Systems Engineering



Material Taken from

Understanding Space: An Introduction to Astronautics, 3rd Edition (Space Technology) by Jerry Jon Sellers, William J. Astore, Robert B. Giffen and Wiley J. Larson (Sep 8, 2005)

ISBN-13: 978-0073407753 ISBN-10: 0073407755

Wiley Interscience

Chapters 11, 15

What is Systems Engineering?

A System Is …

Simply stated, a system is an integrated composite of people, products, and processes that provide a capability to satisfy a stated need or objective.

Systems Engineering Is…

Systems engineering consists of two significant disciplines: the *technical knowledge domain* in which the systems engineer operates, and *systems engineering management*.

It is an interdisciplinary approach that encompasses the entire technical effort, and evolves into and verifies an integrated and life cycle balanced set of system people, products, and process solutions that satisfy customer needs.

What is Systems Engineering? (2)

Systems Engineering Management Entails…

Systems engineering management is accomplished by integrating three major activities:

- Development phasing that controls the design process and provides baselines that coordinate design efforts,

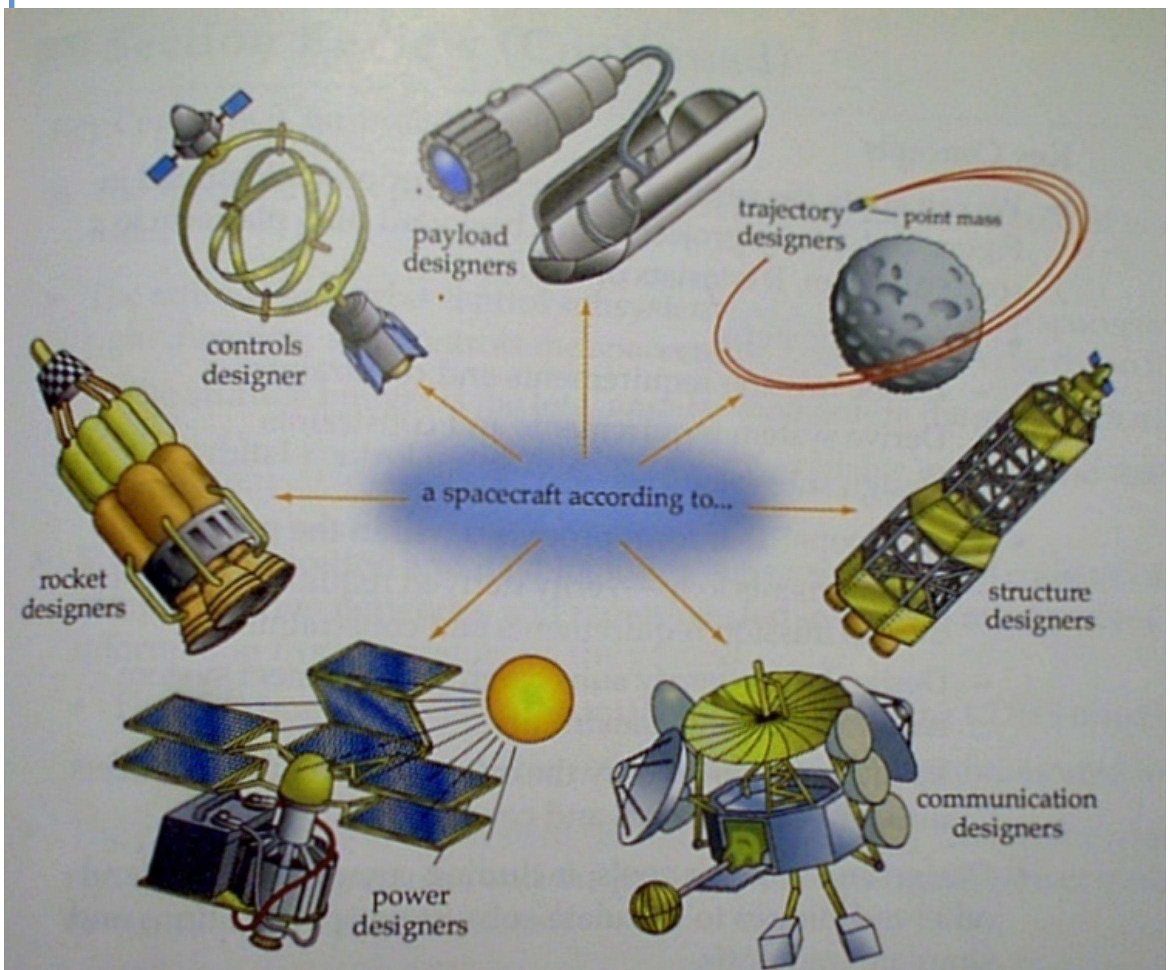
- A systems engineering process that provides a structure for solving design problems and tracking requirements flow through the design effort, and

- Life cycle integration that involves customers in the design process and ensures that the system developed is viable throughout its life.

The Systems Engineering Process

- **The systems engineering** process is a top-down comprehensive, iterative and recursive problem solving process, applied sequentially through all stages of development, that is used to:
- **Transform needs and requirements** into a set of system product and process descriptions (adding value and more detail with each level of development),
- **Generate information for decision makers**
- **Provide input for the next level of development.**

Systems Engineering Applied to the Space System Design Process

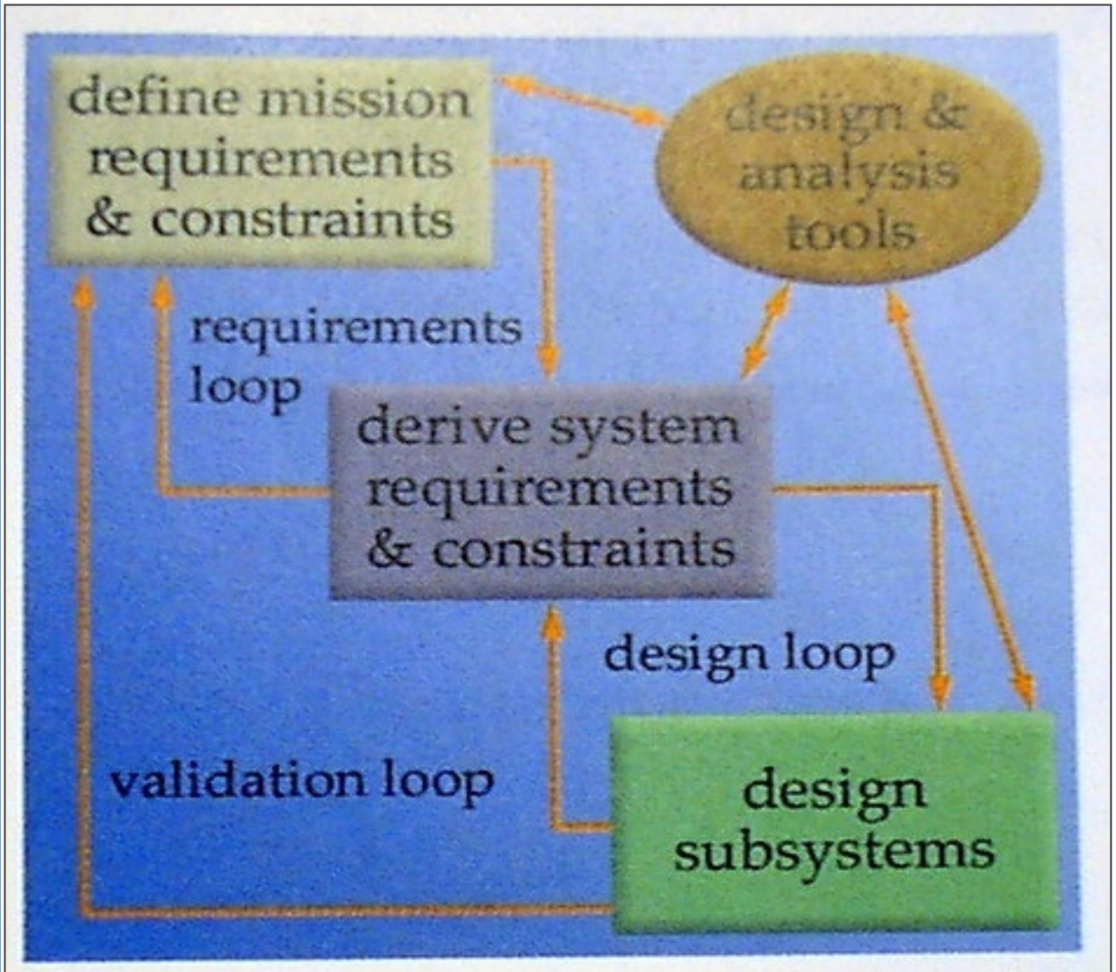


... "a spacecraft according to ...

- **Sometimes individual subsystem designers get so focused on their subsystem designs that they lose sight of the overall mission objectives and requirements**

- **Good systems engineering coordinates the activities of disciplinary groups with disparate design objectives**

Systems Engineering Applied to the Space System Design Process (2)



- Systems Engineering is a fundamental process that can be used to design anything from a backyard grill to a crewed-space platform.

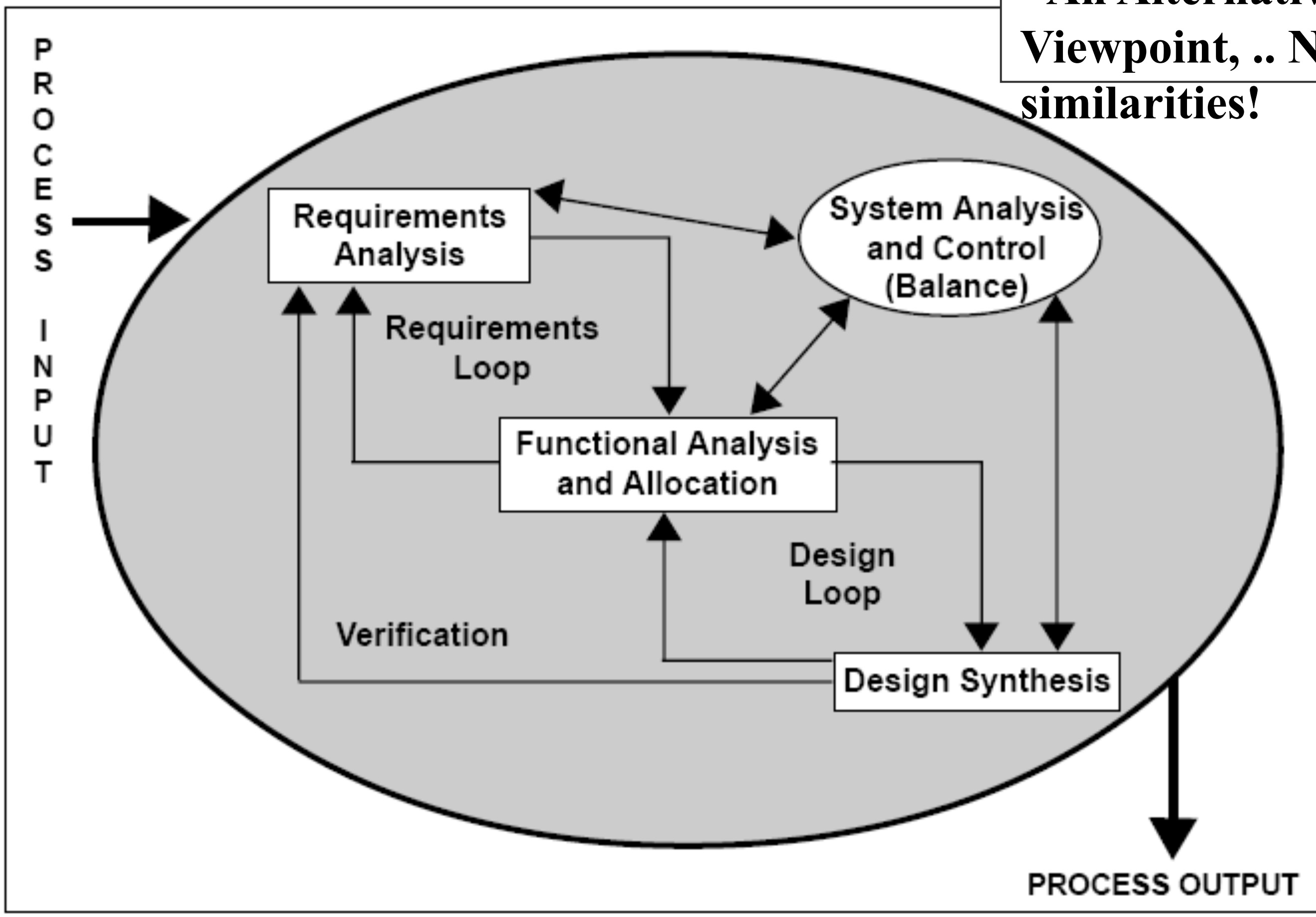
- Each step utilizes established design and analysis tools.

- The process is iterative.

- Between process steps there are "feedback loops" to review decisions made in previous steps.

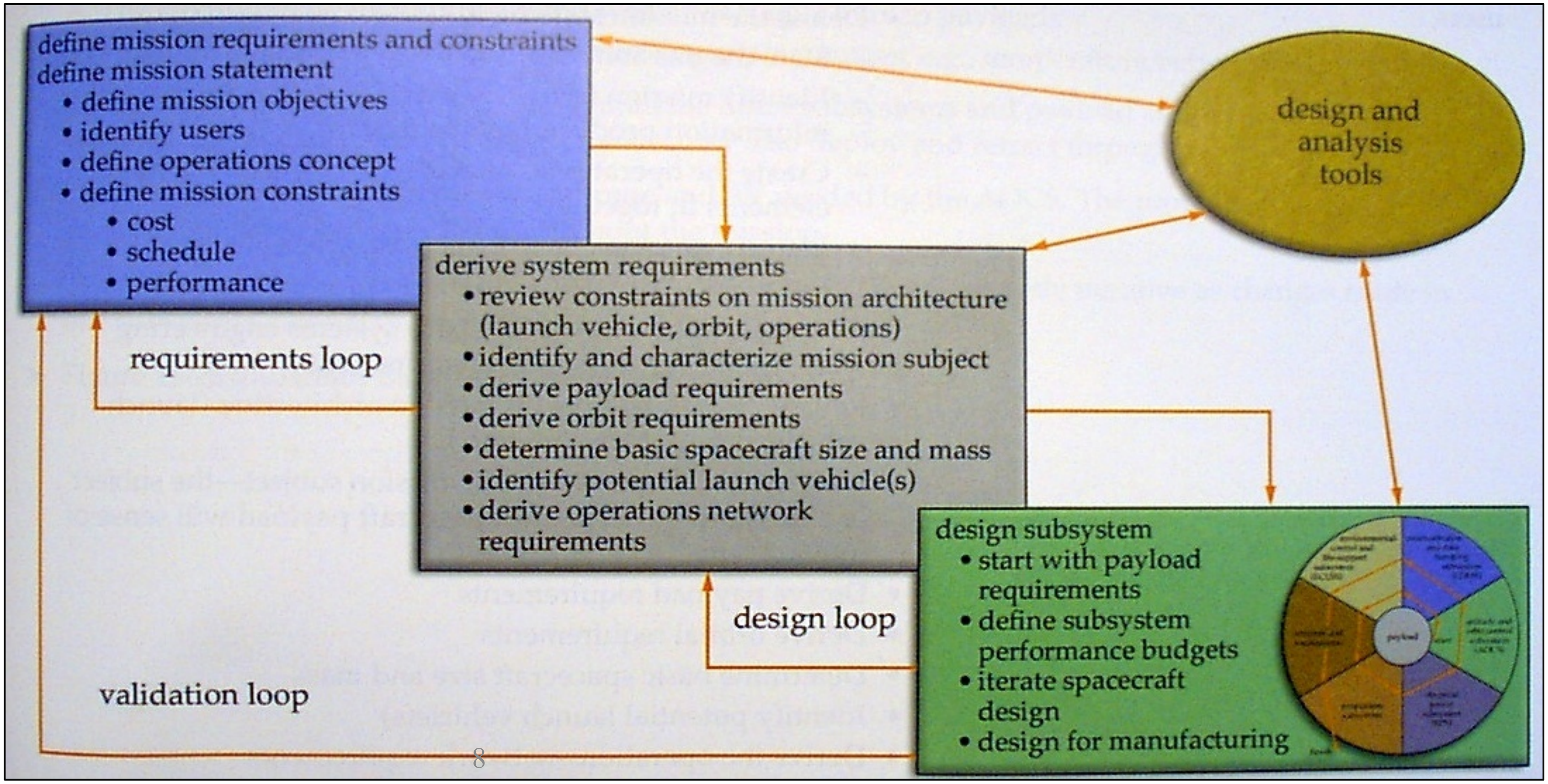
The Systems Engineering Process (4)

• An Alternative Viewpoint, .. Notice the similarities!



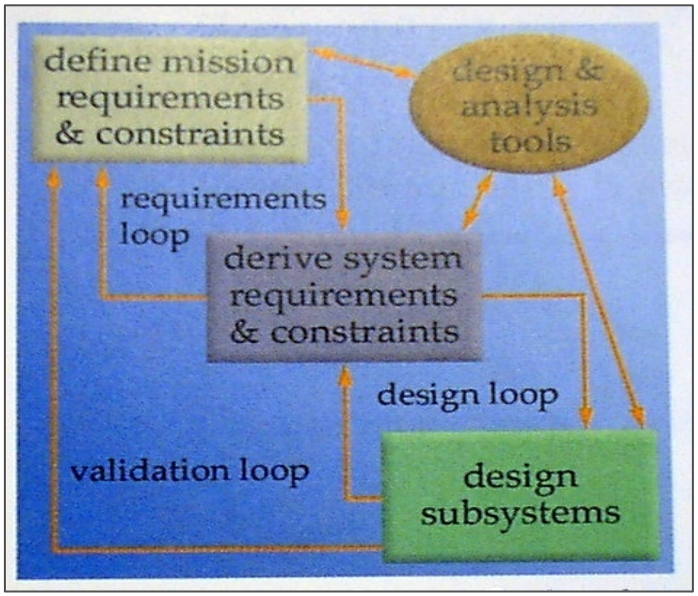
The Systems Engineering Process

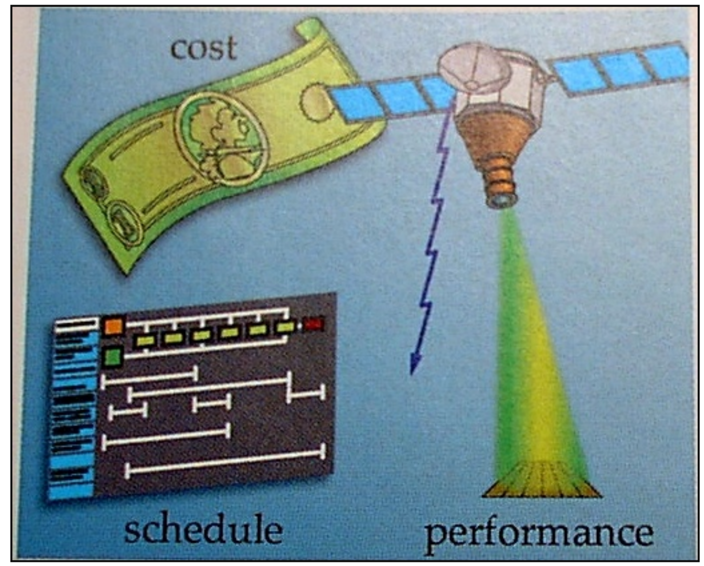
Systems Engineering Applied to the Space System Design Process



• **By following a well-defined process, systems engineers design spacecraft that meet mission requirements while staying within budget and conforming to constraints**

Systems Engineering Applied to the Space System Design Process (3)





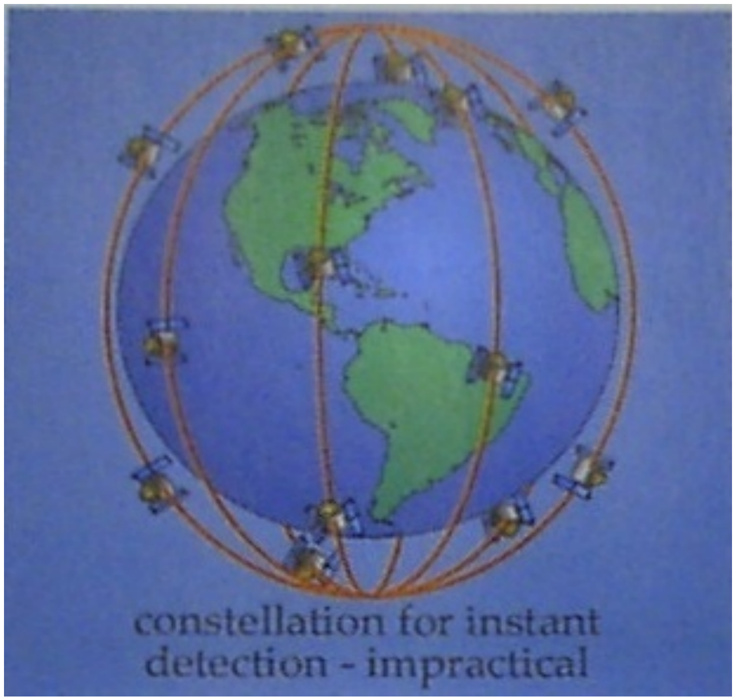
Cost, Schedule, Performance

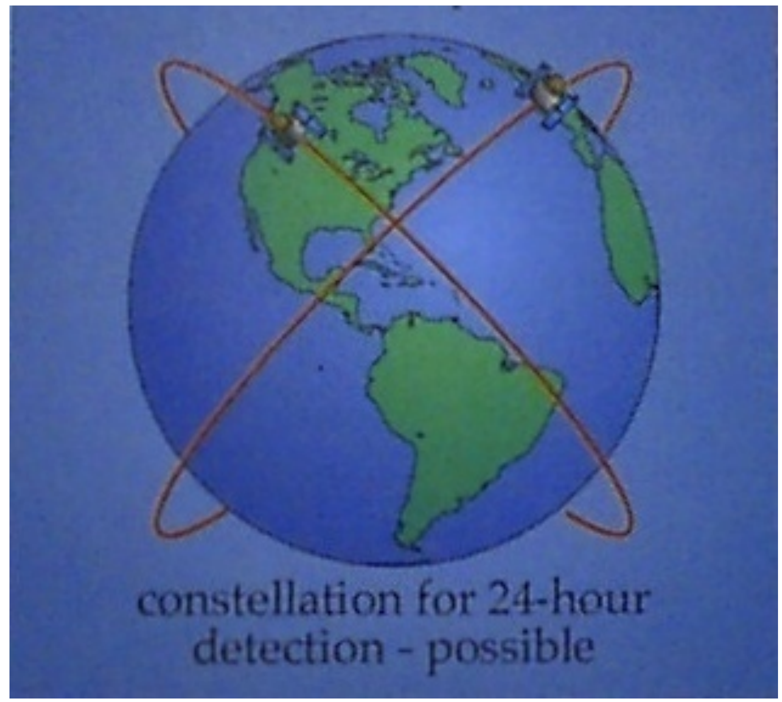
- **3-D trade space that mission must operate within.**

- **Systems engineers continually trade competing objectives to achieve well-balanced solution -- "optimal" solution often not-achievable**

Systems Engineering Applied to the Space System Design Process (5)

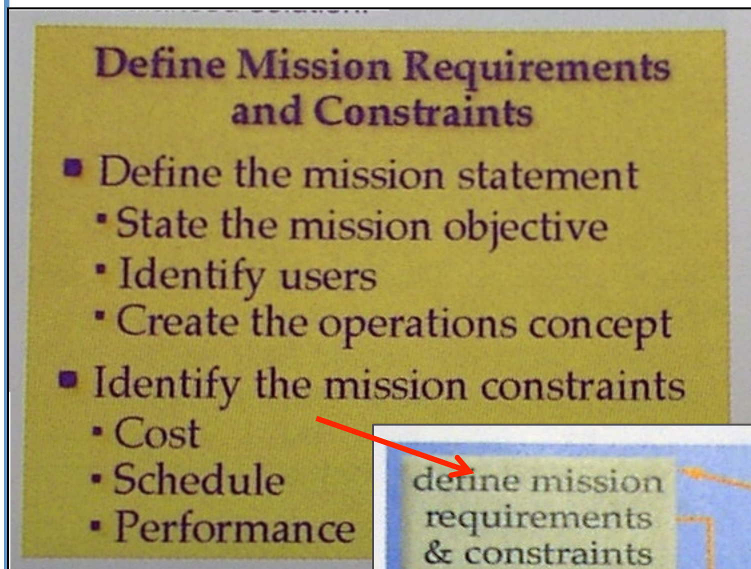
Trading Requirements

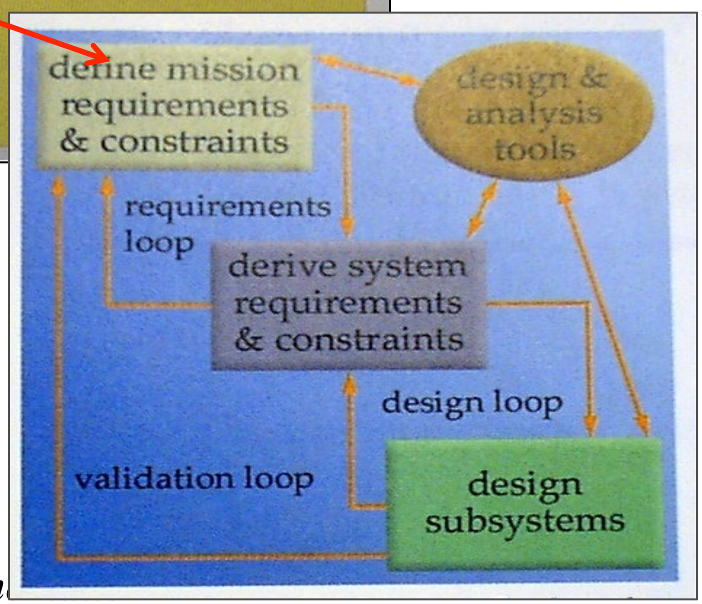




- **By trading-off mission requirements versus system-level requirements, an infeasible mission (too complex or to expensive or both) may become feasible and affordable**

Systems Engineering Applied to the Space System Design Process (4)

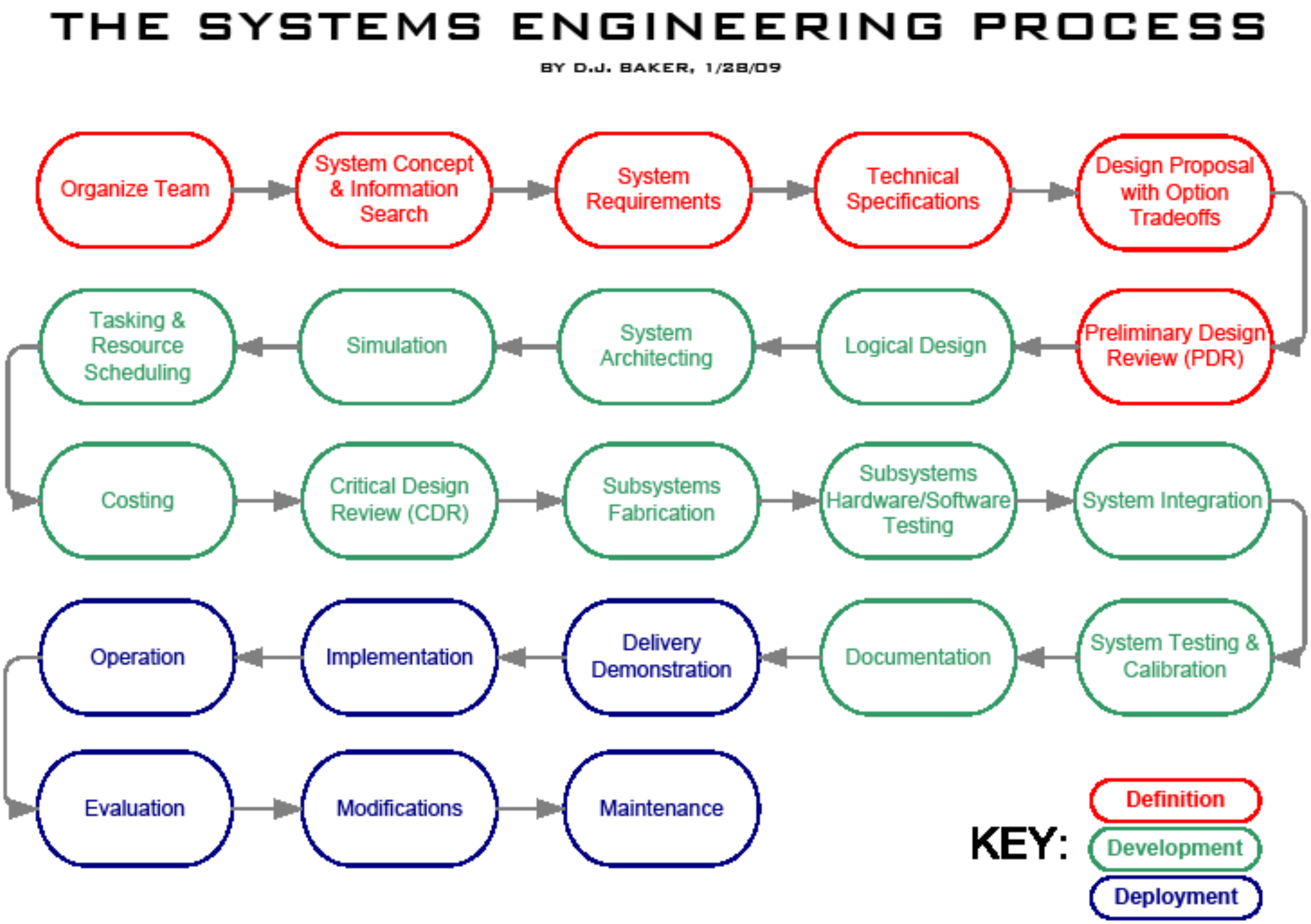




- **First Phase in design process is to define the mission requirements, Objectives, and constraints.**

- **Often documented and detailed in the mission "Objectives and Requirements Document." *(ORD)***

The "Baker" Chart



Requirements Analysis

Requirements analysis involves defining customer needs and objectives in the context of planned customer use, environments, and identified system characteristics to determine requirements for system functions.

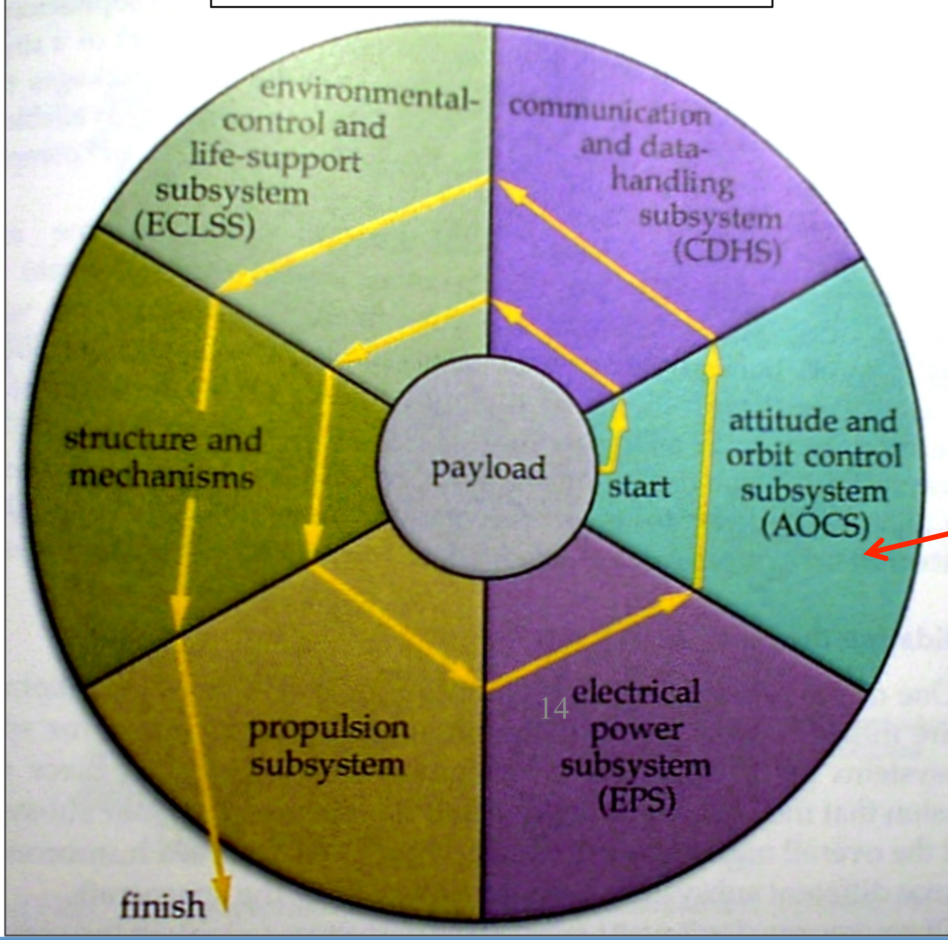
Requirements analysis is conducted iteratively with functional analysis to optimize performance requirements for identified functions, and to verify that synthesized solutions can satisfy customer requirements.

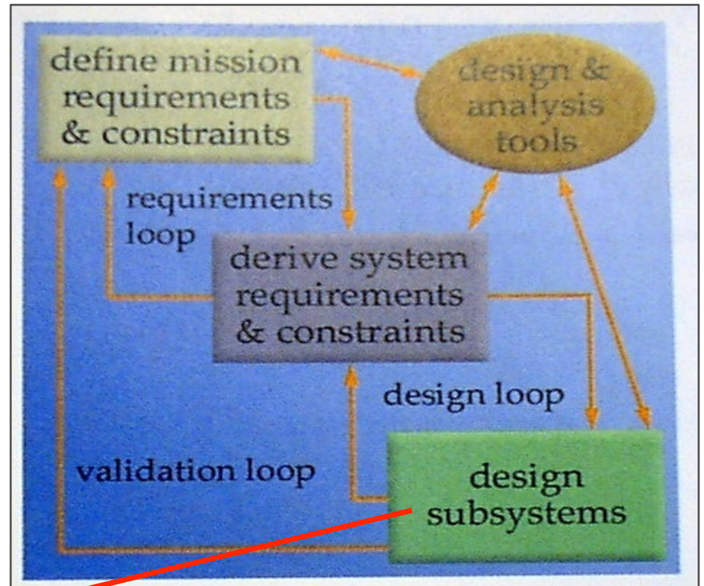
In general, Requirements Analysis should result in a clear understanding of:

- *Functions: What the system has to do,*
- *Performance: How well the functions have to be performed,*
- *Interfaces: Environment in which the system will perform, and*
- *Other requirements and constraints.*

Systems Engineering Applied to Sub-System Design Process (1)

Subsystems Design



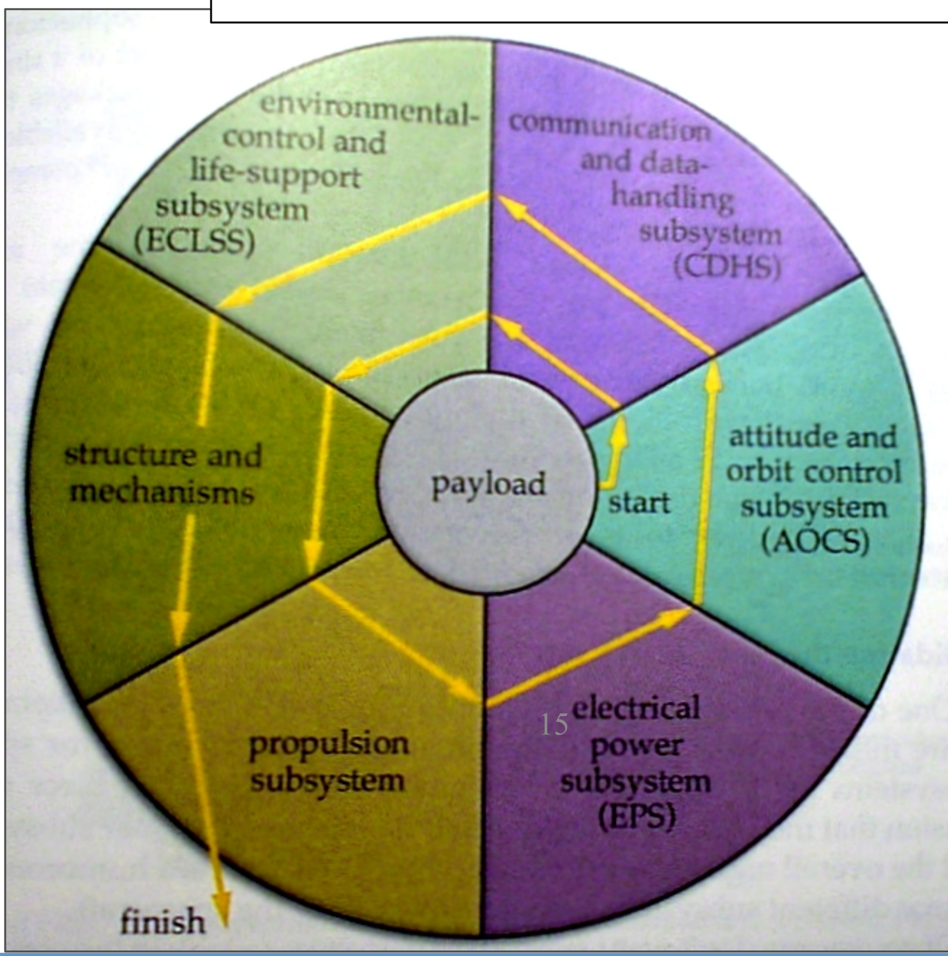


- **Subsystem Design Process follows a distinct order and development hierarchy**

- ***Hmmmm .. Why is the propulsion System last on this chart?***

Systems Engineering Applied to the Sub-System Design Process (2)

Subsystems Design Revisited



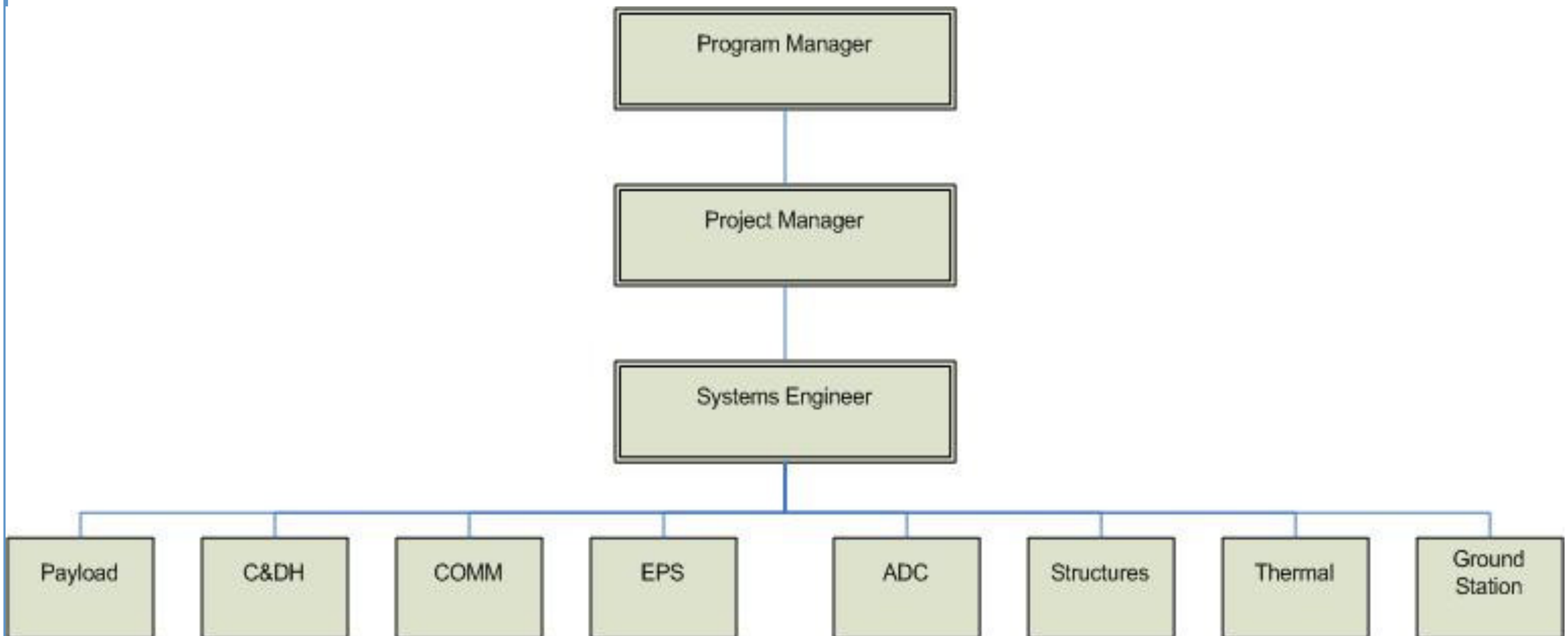
• **Subsystem Design chart shows the Interdependence of all of the Spacecraft subsystems.**

• ***When the design of one sub-system is modified,* then it typically become s necessary to adjust the designs of *Some or all* of the other sub systems.**

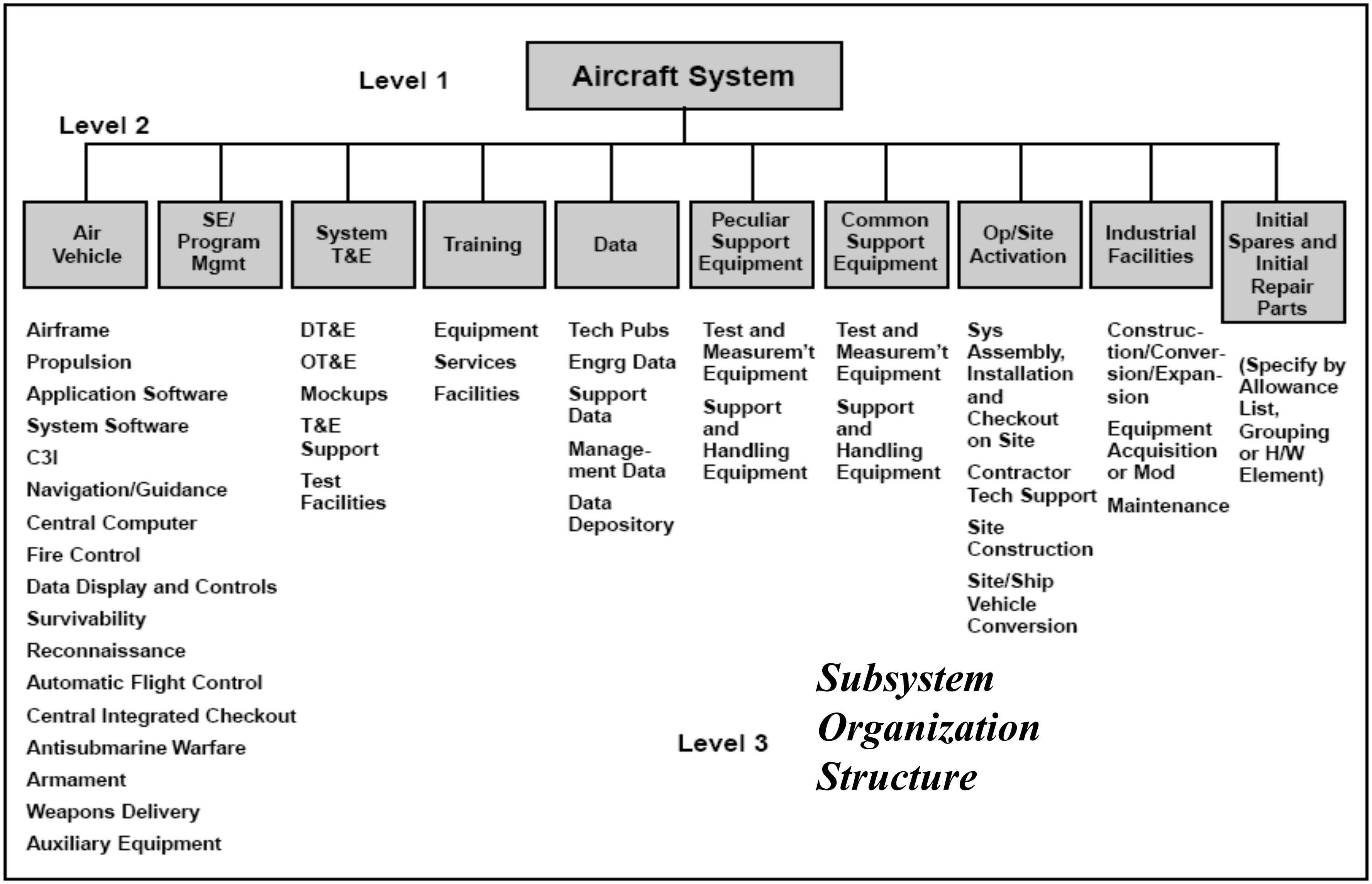
• ***In extreme cases, the payload sometimes needs to be modified as the result of a mandated sub-system Change.***

Program Management

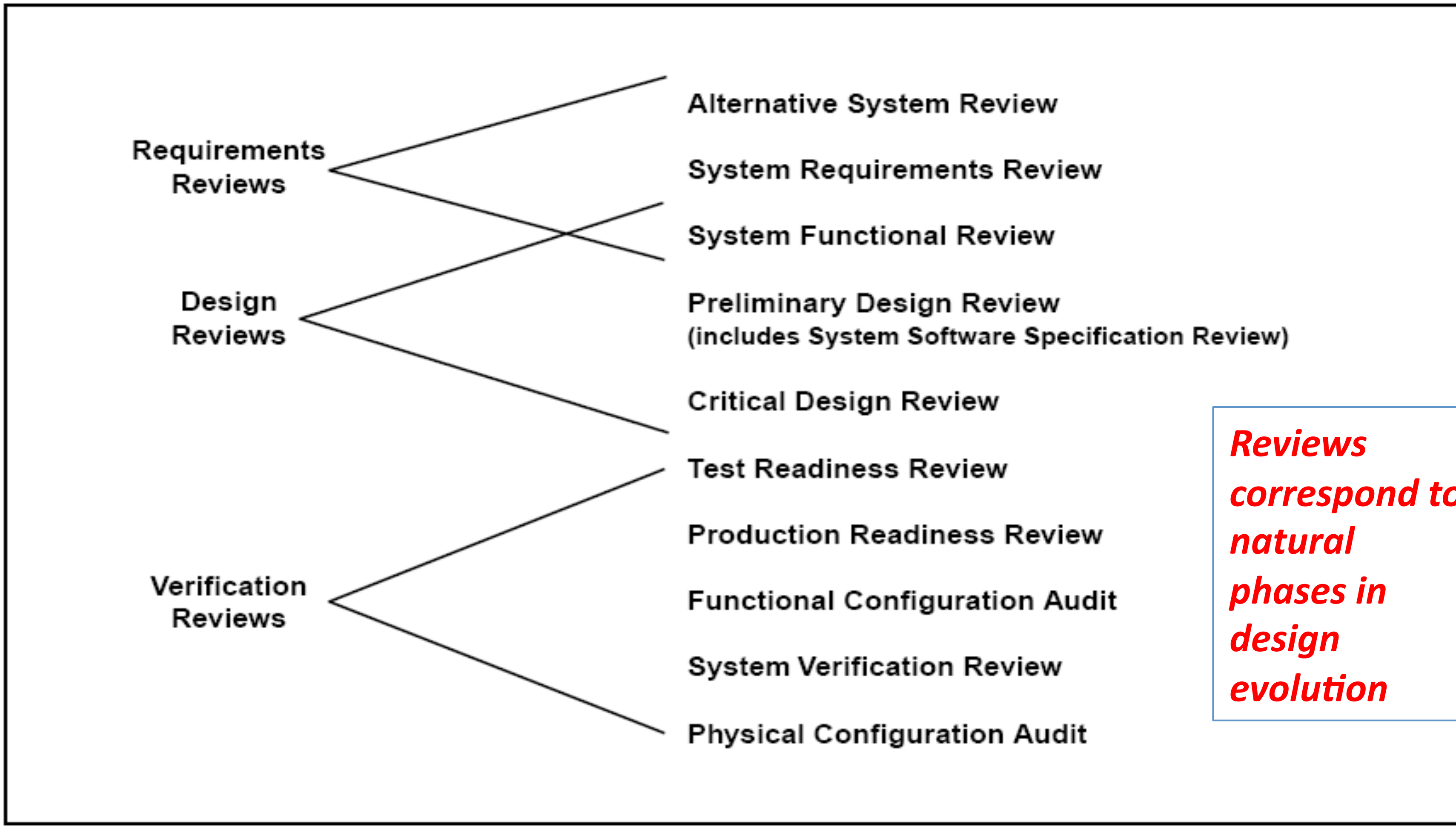
Typical Spacecraft Program Management Structure
(The bottom row is subsystem team leads)



Program Management (2)



The Documentation and Review Process



Typical System-Level Technical Reviews

Reviews correspond to natural phases in design evolution

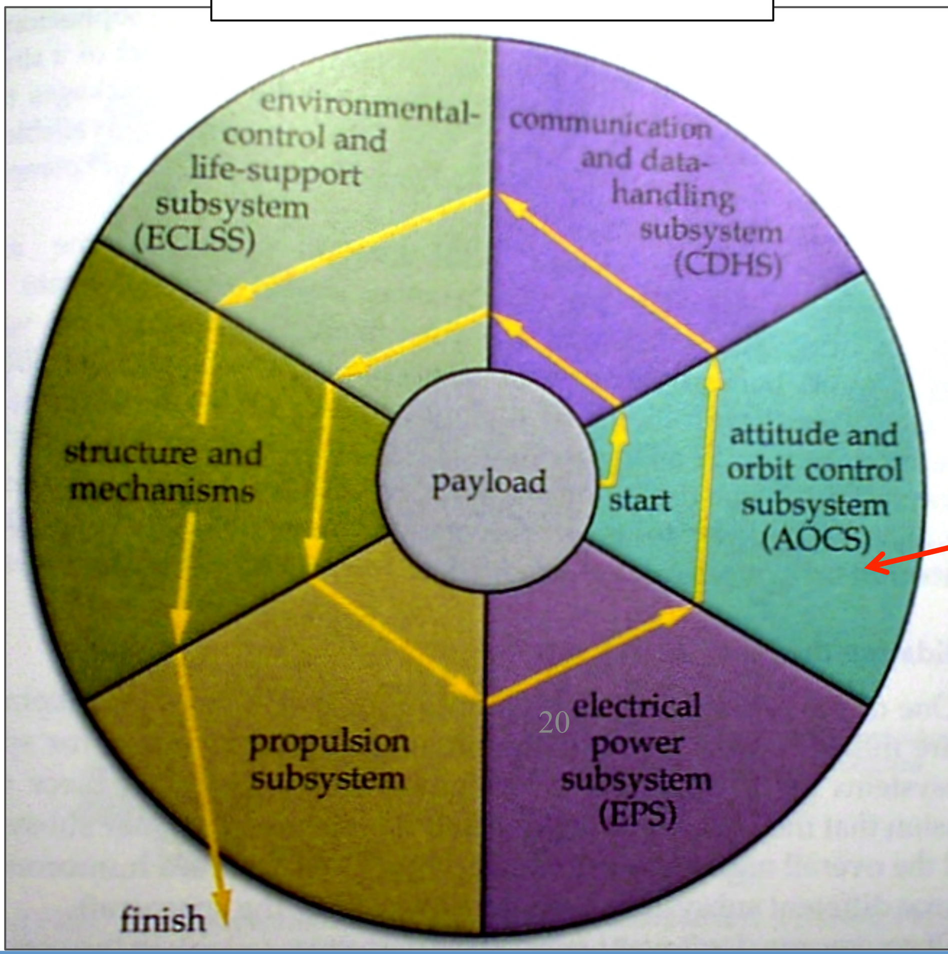
Systems Engineering II: Design Tools

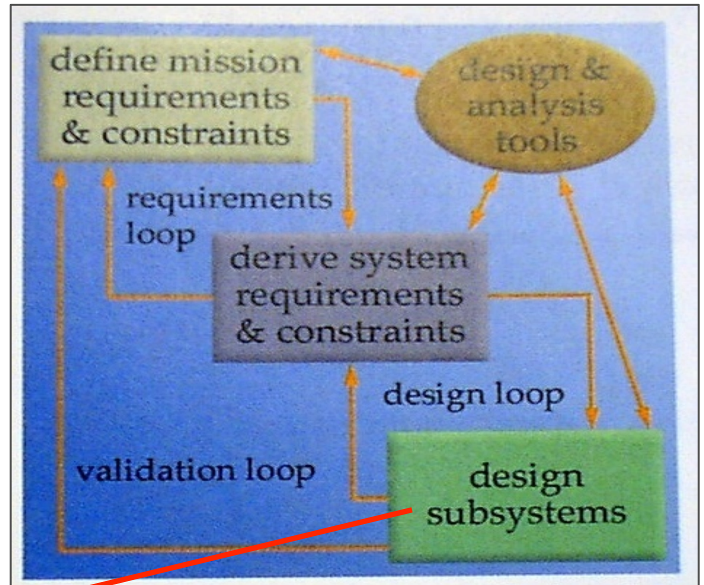
Sellers: Chapters 11, 15 + Material
From Auburn University Lunar Excavator
Design Course, Courtesy of David Beale

This section provides examples of systems engineering tools which may be needed during the design process.

Systems Engineering Applied to Sub-System Design Process (1)

Subsystems Design





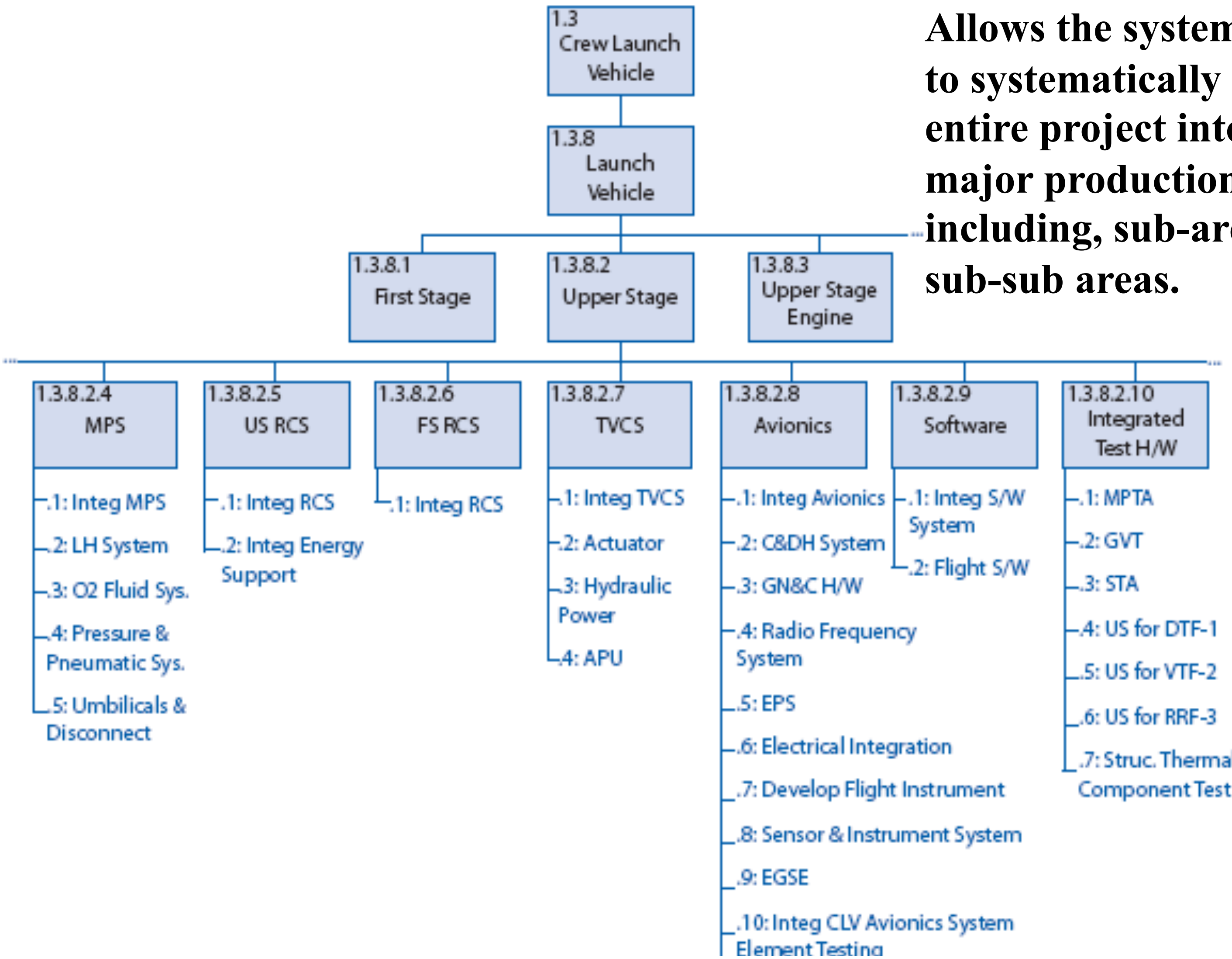
- **Subsystem Design Process follows a distinct order and development hierarchy**

- ***Hmmmm .. Why is the propulsion System last on this chart?***

Systems Engineering Tools

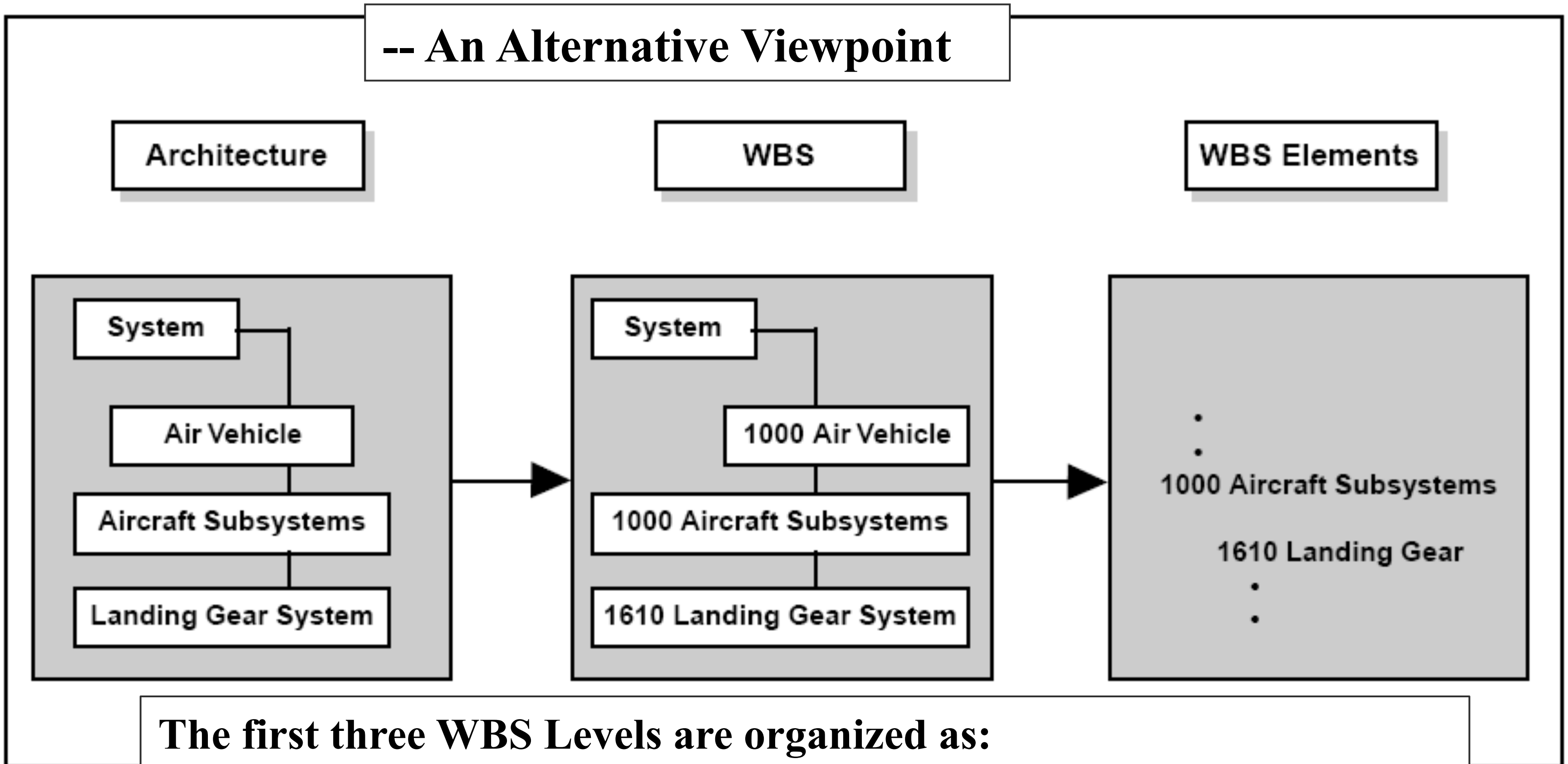
SE Function	Tool	Phase(s) where Tool Applied
Input Derived Requirements		
Architecture/Design	Product Breakdown Structure, Trade Studies	A, B
Concept of Operations		
Interfacing	Interface Control Document	A, B, C
	Modeling and Simulation	
Mission Environment		
Resource Budgets	Mass, Power, Cost, Link budgets	A, B, C
Risk Management	Failure Mode Analysis	B, C
Configuration Management		
Management Functions	Work Breakdown Structure, Gantt Chart, SEMP	throughout

Product Breakdown Structure



Allows the systems engineer to systematically divide an entire project into a set of major production areas including, sub-areas, and sub-sub areas.

Work Breakdown Structure (2)

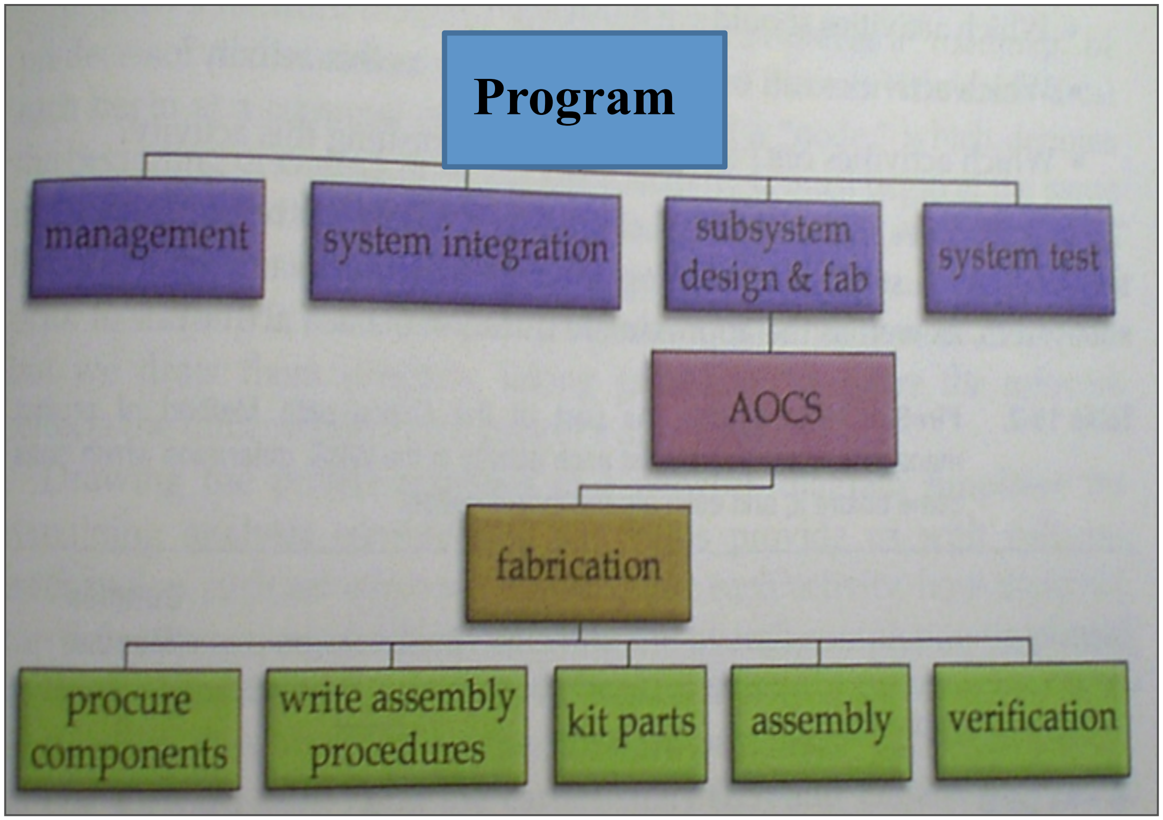


The first three WBS Levels are organized as:
Level 1 – Overall System
Level 2 – Major Element (Segment)
Level 3 – Subordinate Components (Prime Items)

UtahState UNIVERSITY

Mechanical & Aerospace Engineering

Work Breakdown Structure (WBS)



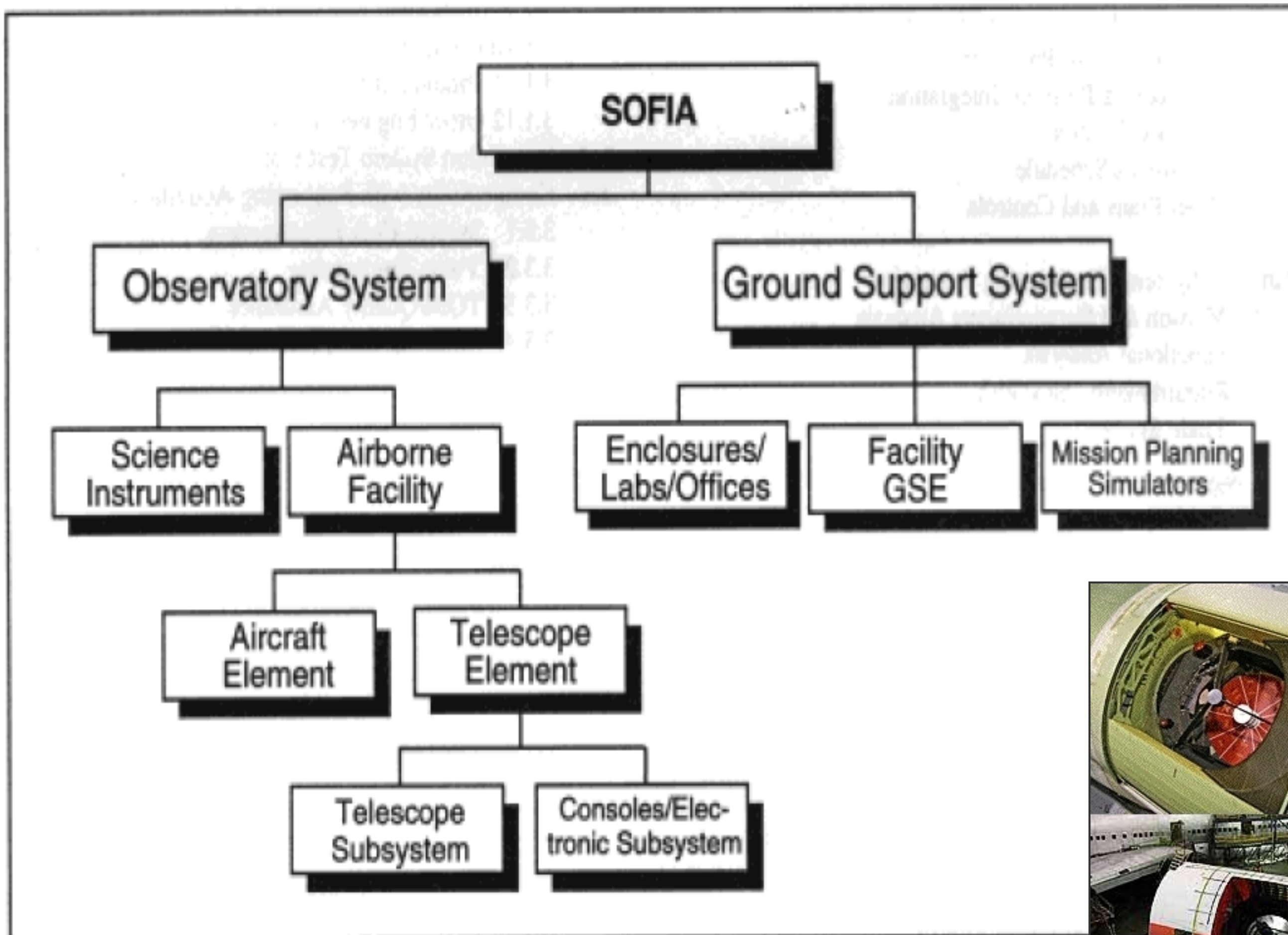
Fundamental Management Tool

-- WBS allows the systems engineer to systematically divide an entire project into a set of major tasks, sub-tasks, and sub-sub tasks.

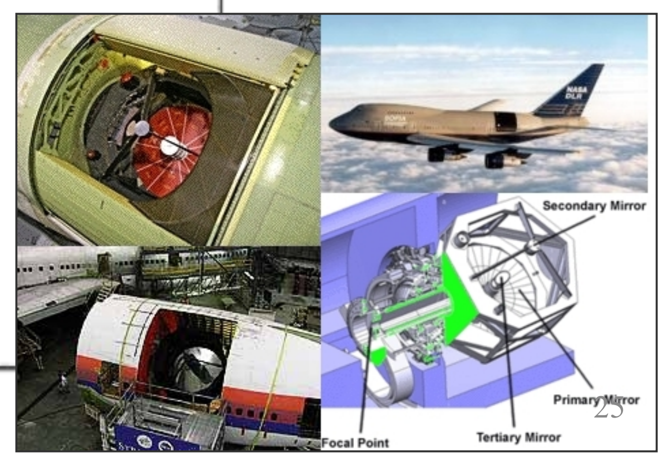
-- In this example, the tasks for fabrication of the attitude and orbit control system (AOCS) are broken into 5 sub-tasks. (Level 1 WBS)

-- Each sub-tasks can be further sub-divided (Level 2 WBS)

Product Breakdown Structure (2)



PBS for the SOFIA infrared telescope



Work Breakdown Structure (3)

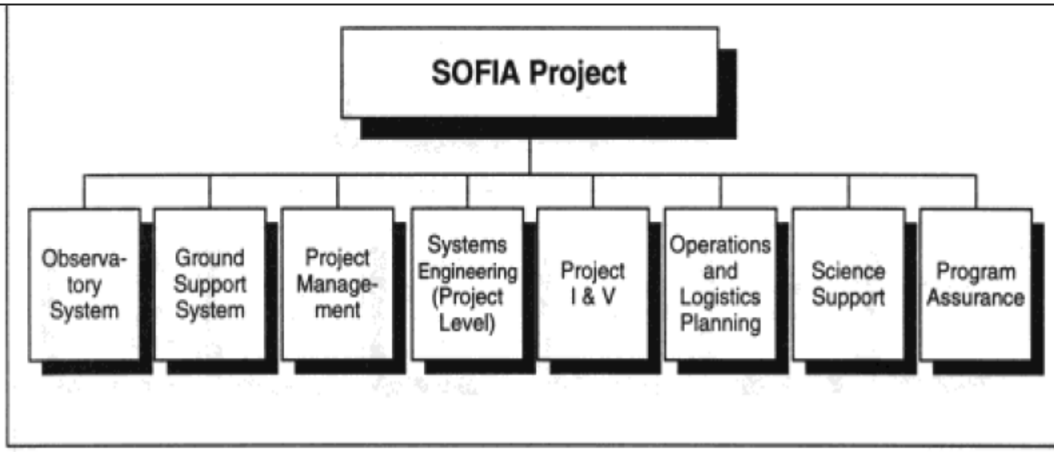
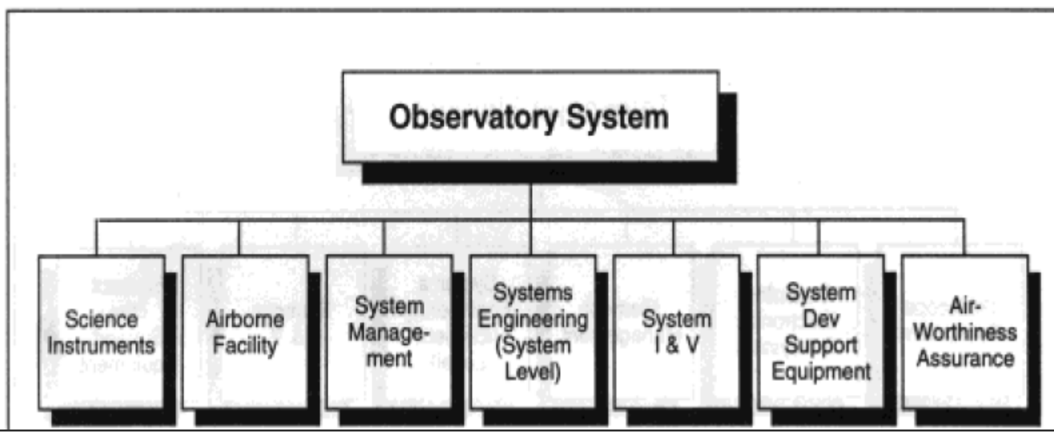
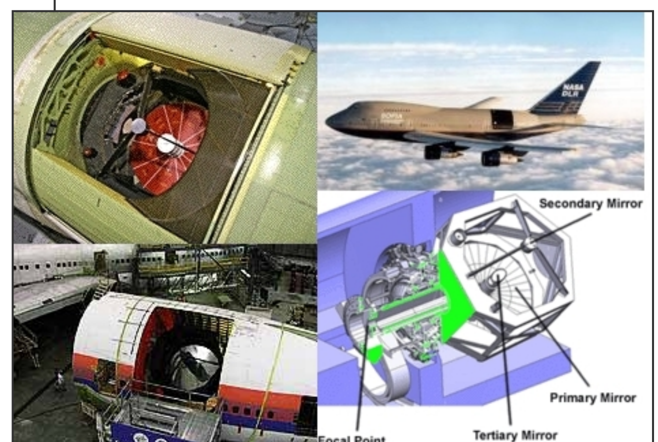


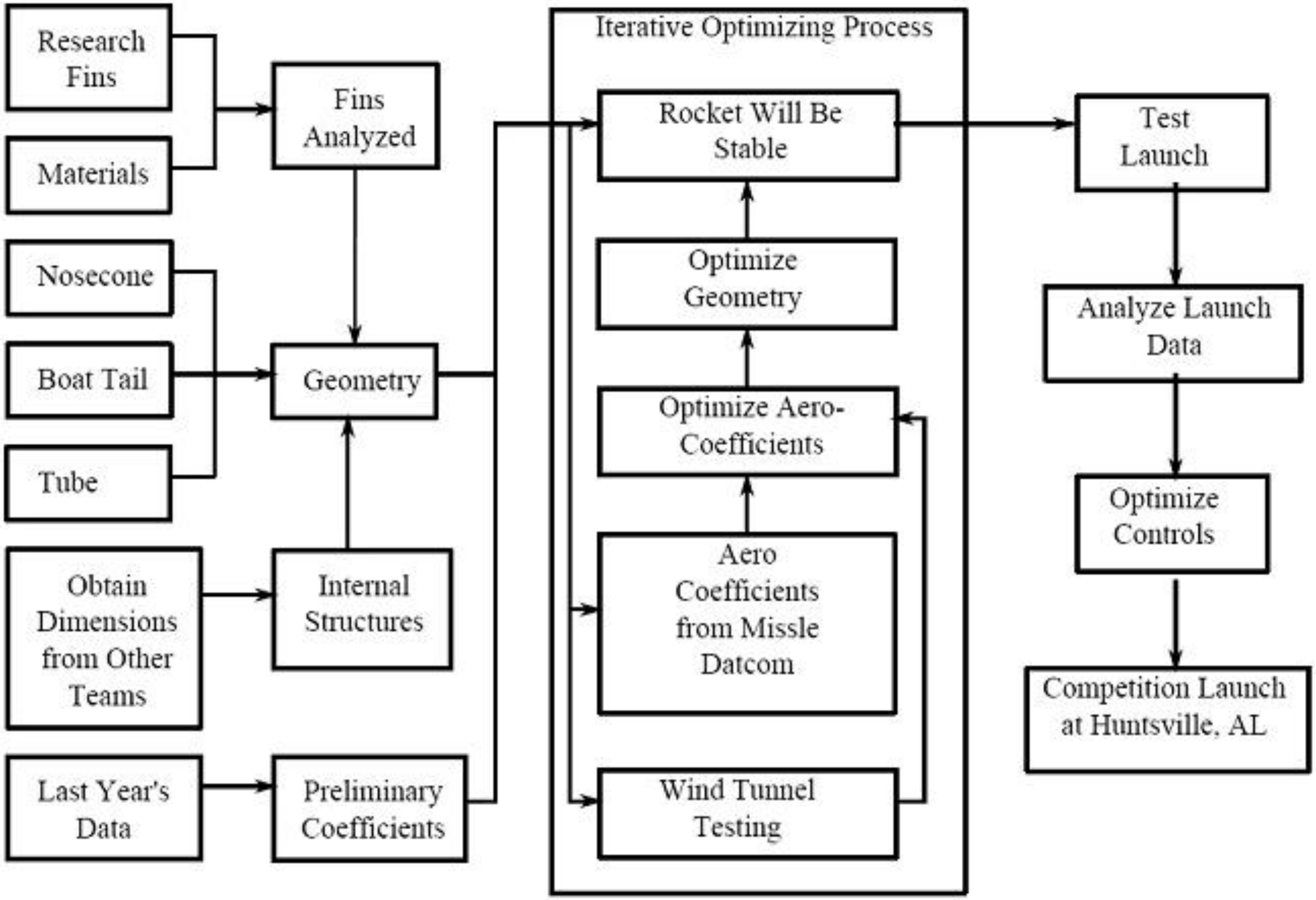
Figure B-2 — SOFIA Project WBS (*Level 3*).



WBS for SOFIA Project

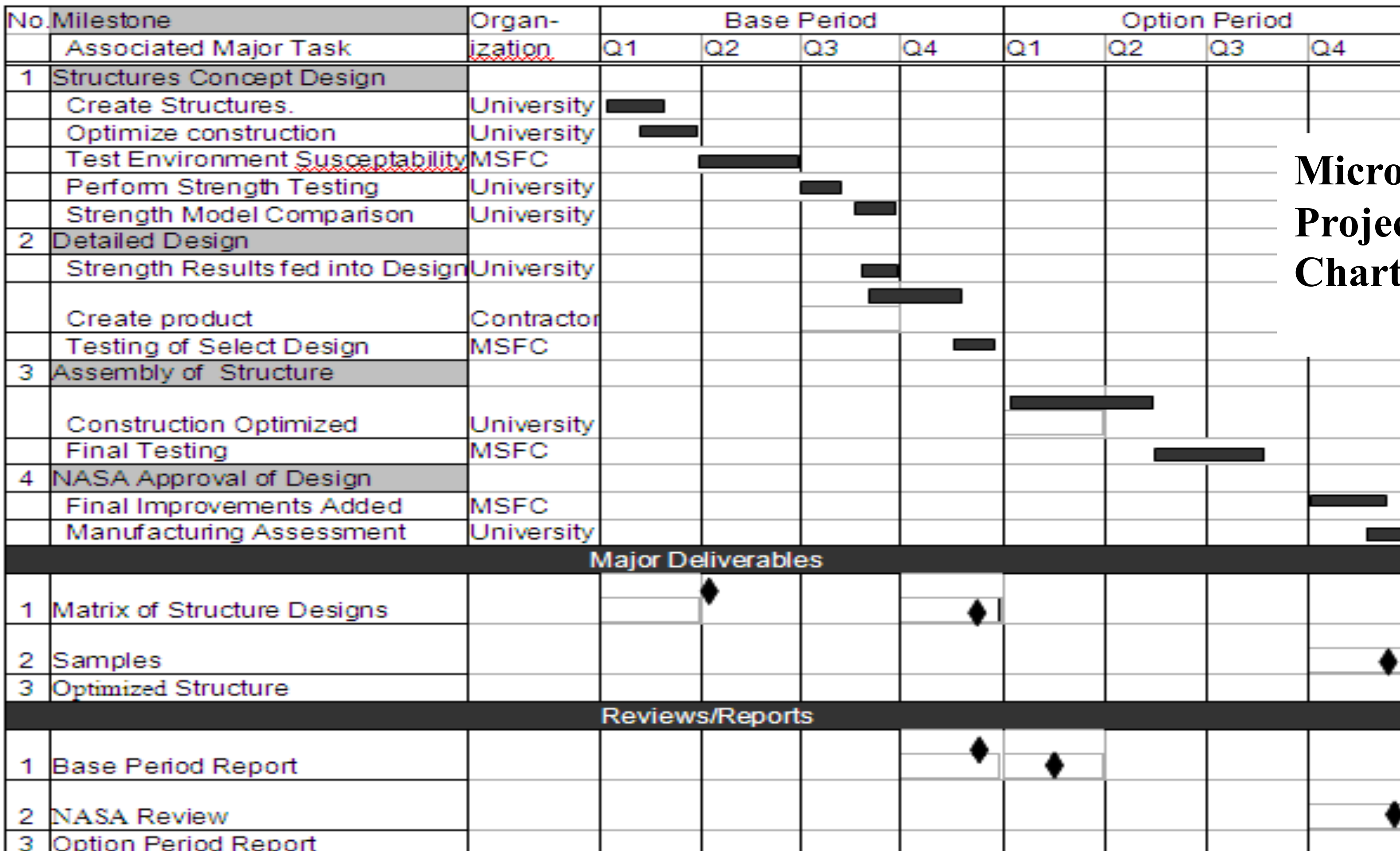


USU Chimaera WBS, 2008-2009



Gantt Chart

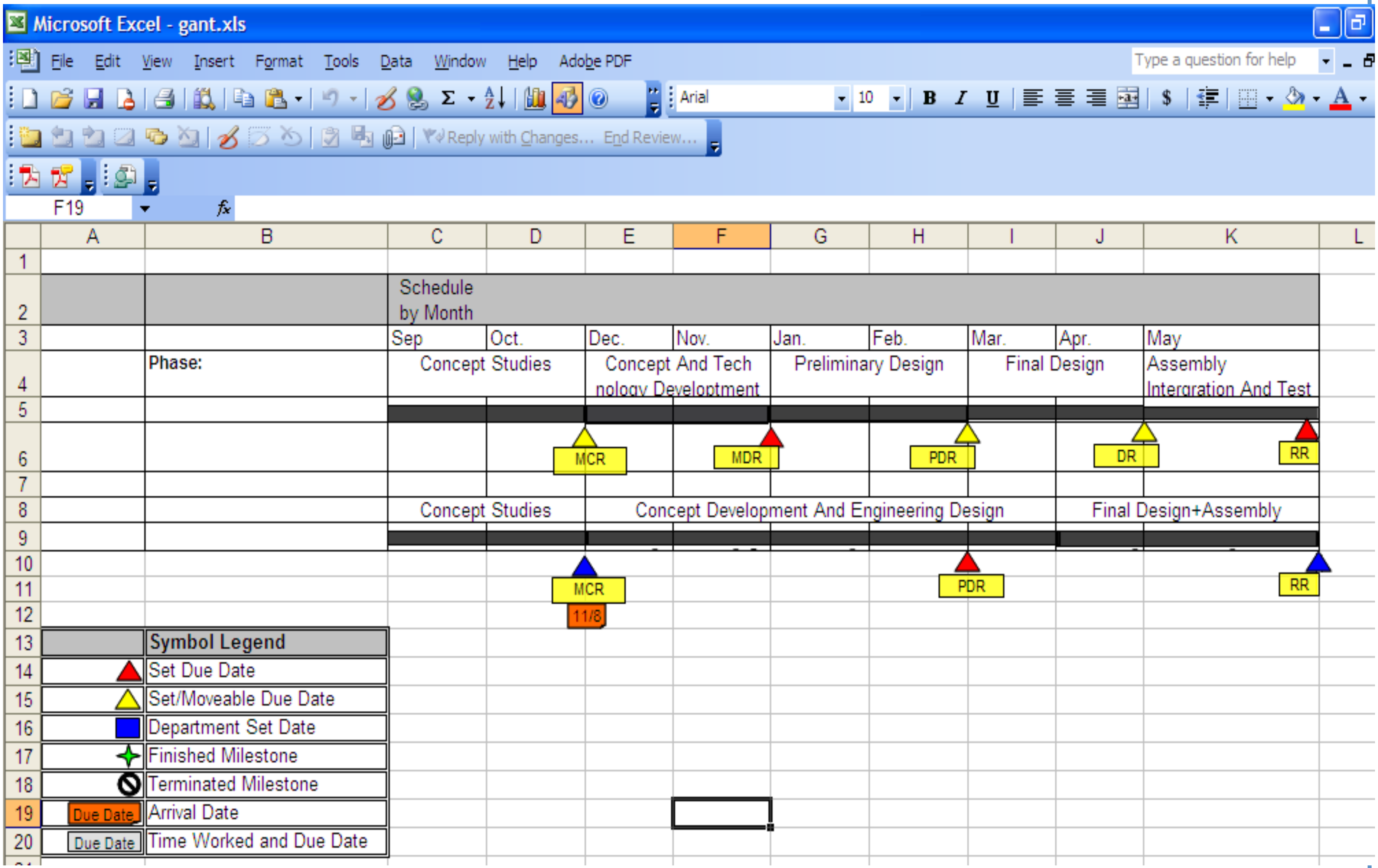
Bar chart that can be used to allot time to tasks, schedule reviews, and date milestones .. Complements WBS



Microsoft Project Chart

1-2 Gantt Chart showing schedule of tasks

Microsoft EXCEL Gantt Chart



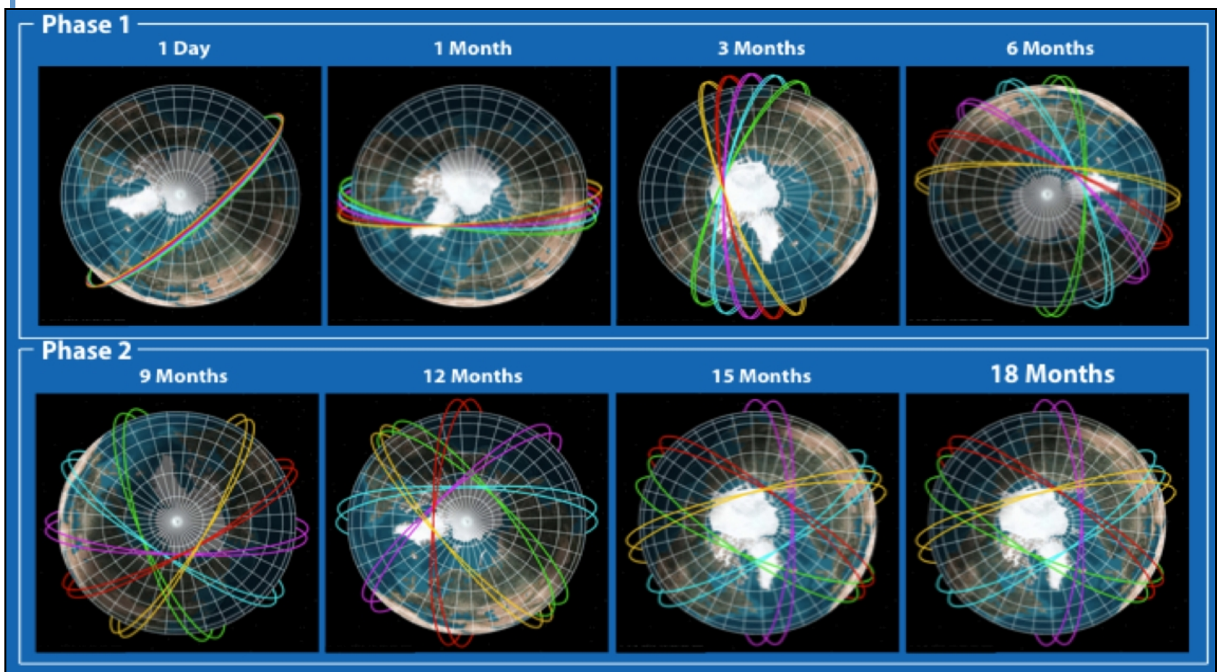
Concept of Operations (CONOPS)

Short Verbal or graphic statement, in broad outline, of a commander's assumptions or intent in regard to an operation or series of operations.

The concept of operations frequently is embodied in campaign plans and operation plans; in the latter case, particularly when the plans cover a series of connected operations to be carried out simultaneously or in succession.

The concept is designed to give an overall picture of the operation. It is included primarily for additional clarity of purpose. Also called commander's concept or CONOPS.

CONOPS Example

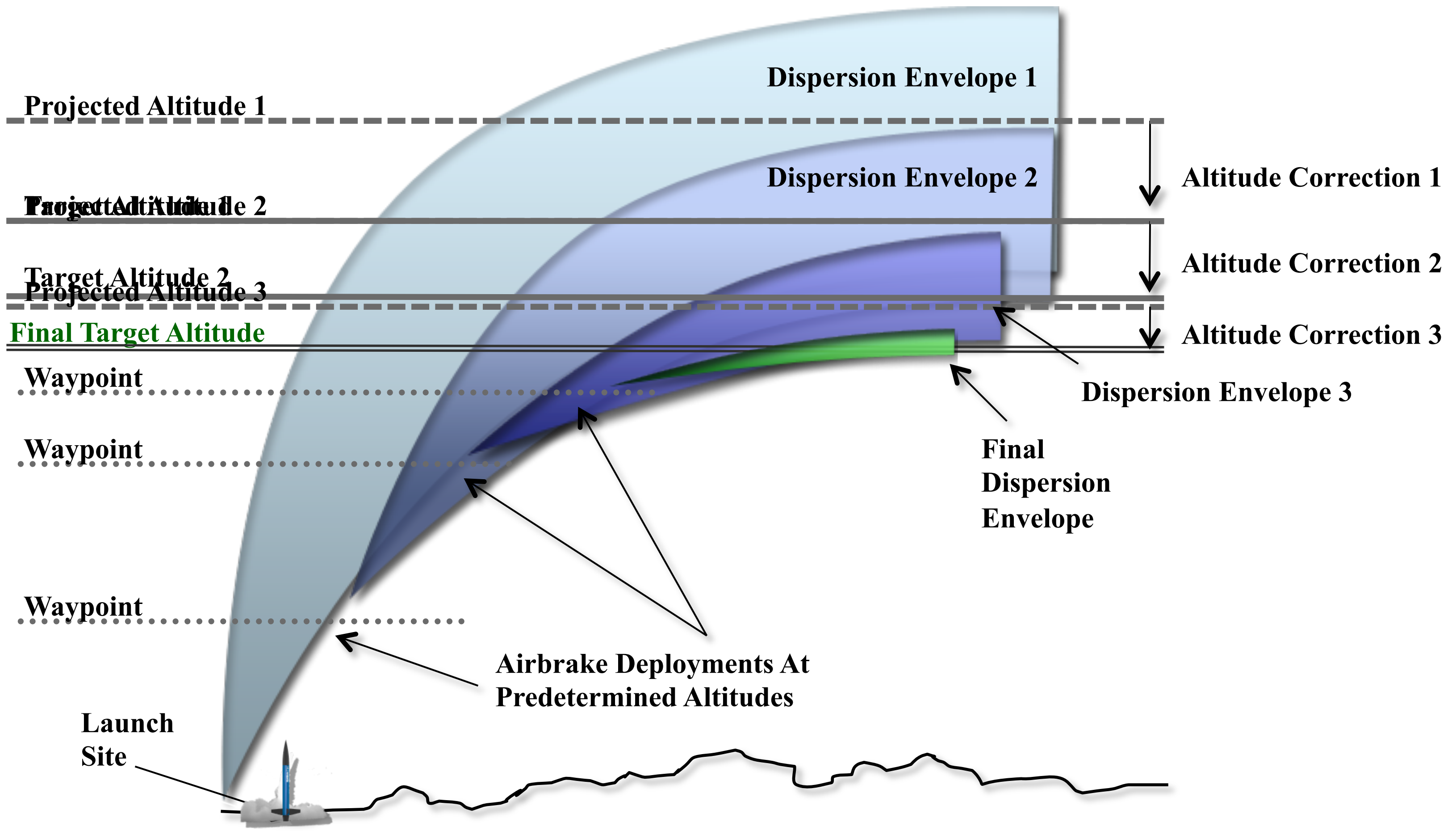


• As magnetosphere processes evolve during a geomagnetic disturbance, *HiDEF* E-field observations provide a detailed map

• Constellation will utilize natural RAAN precession to transform cluster from initially densely packed “sting of pearls” to a globally distributed sensor cluster

• HiDEF mission proposed for in-situ simultaneous E-field measurements
using constellation of pico-satellites

CONOPS Example2

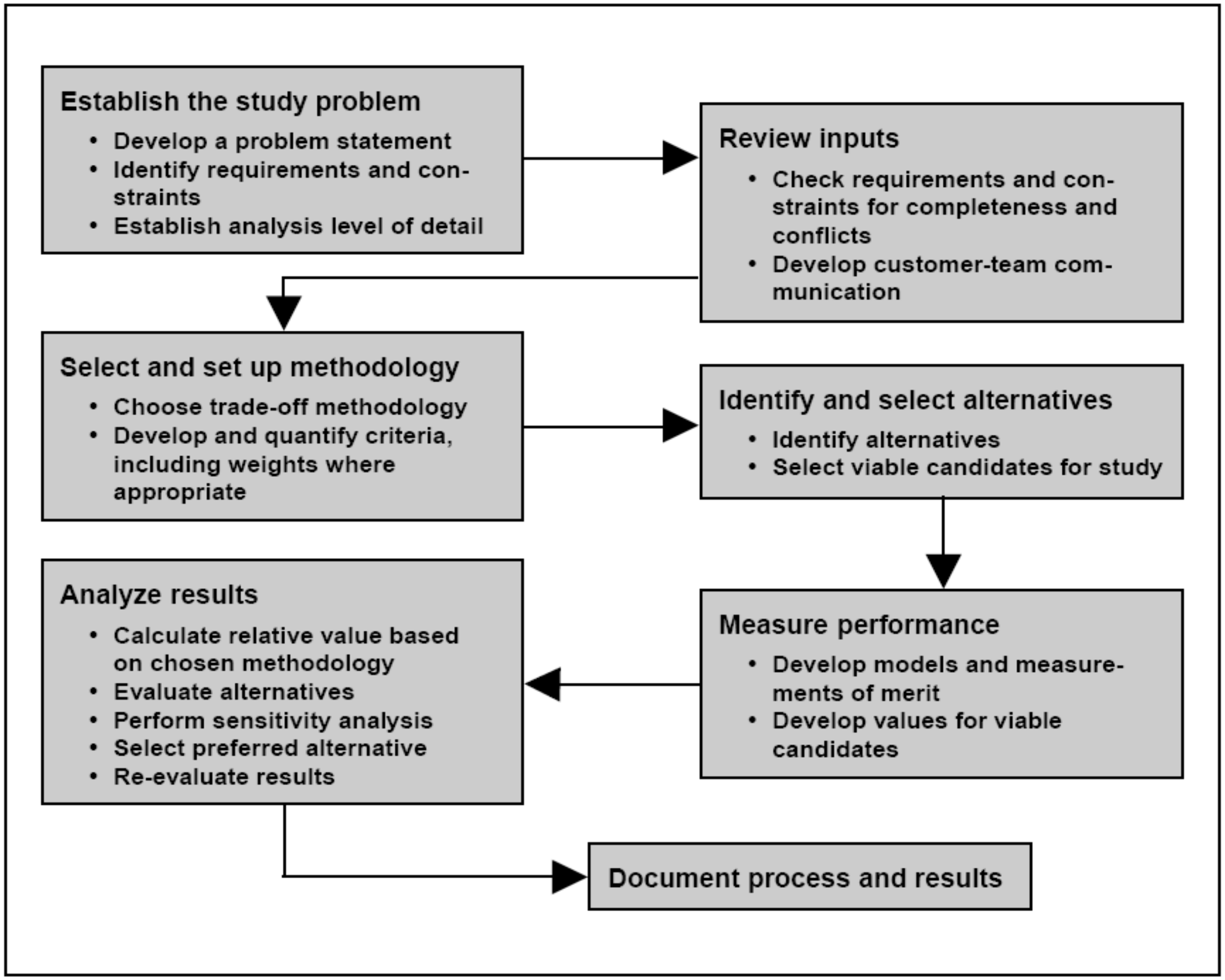


- Launch "high" and then control maximum altitude through successive application of airbrakes until desired energy state is reached
- Inertial navigation used during powered ascent and a Kalman filter used for navigation during coasting stage-both navigation algorithms were student designed and implemented

Trade Studies

- **Trade study is a tool used to help choose the best solution among alternatives.**
- **Numerical values are given based on weight factors and a normalization scale for the evaluation criteria.**
- **Evaluation criteria are important factors that are included intrade study.**
- **Weight factors are used to dictate how important the evaluation criteria are relative to each other.**
- **The choice of weight factors and normalization scale are extremely important to this process.**
- **Normalization scale creates a constant interval scale that allows us to set a numerical for each of the evaluation criteria (e.g. cost, mass, volume, power consumption legacy, ease of use).**

Trade Studies (2)



Trade Study Process

Steps to a trade study

1. Define the problem.
2. Define constraints on the on the solutions.
3. Find 3-5 solutions
4. Define evaluation criteria.
5. Define weight factors
6. Define normalization scale
7. Populate trade matrix
8. Rank the solutions

Trade Studies (3)

Decision Factors / Alternatives	Range Wt. = 2.0		Speed Wt. = 1.0		Payload Wt. = 2.5		Weighted Total
	U	W	U	W	U	W	
Transport System 1	.8	1.6	.7	.7	.6	1.5	3.8
Transport System 2	.7	1.4	.9	.9	.4	1.0	3.3
Transport System 3	.6	1.2	.7	.7	.8	2.0	3.9
Transport System 4	.5	1.0	.5	.5	.9	2.25	3.75

Key: U = Utility value
W = Weighted value

Sample Pugh Decision Trade matrix

•*Decision matrix:* a decision-support tool allowing decision makers to solve their problem by evaluating, rating, and comparing different alternatives on multiple criteria ... Finding a "best" design

•Prevents a team from "falling in love" with a flawed design or one not meeting all design constraints or objectives

•Communication tool; builds consensus

Trade Studies (4)

The Pugh Evaluation Process

Phase I

1. **Criteria:** The list of evaluation criteria is developed through team discussion. A benchmark or datum is selected, usually the "best" existing product. If no comparable product exists, one of the new concepts (selected at random) can serve as datum.

2. **Design concepts**: Original design concepts are brainstormed by individuals or small teams.

3. **Evaluation matrix:** Each design concept is discussed and evaluated against the datum. Through the discussion, new concepts emerge; they are added to the matrix and evaluated.

4. **Round 1 results:** The results of the first round are evaluated, and the top-ranking concept is selected as the datum for the next round. During an incubation period, the teams improve the original design concepts by borrowing ideas and components from each other, as well as through additional creative thinking. Then Steps 3 and 4 are repeated with these improved, synthesized designs (further rounds).

Phase II

5. **Better designs:** The weakest designs are dropped; the improvement process is continued for additional rounds with fewer but increasingly better concepts. During the process, the strong, surviving concepts are engineered to more detail; the criteria are expanded and further refined. The weak points of the concepts are being eliminated. The team gains insight into the entire problem and solution.

6. **Superior concept:** The process converges to a strong consensus concept that cannot be overturned by a "better idea." The team is committed to this superior design and wants to see it succeed.

(Lumsdaine *et al.*, 2006)

Trade Studies (4)

Study Example – Comparison of Controllers for CubeSat

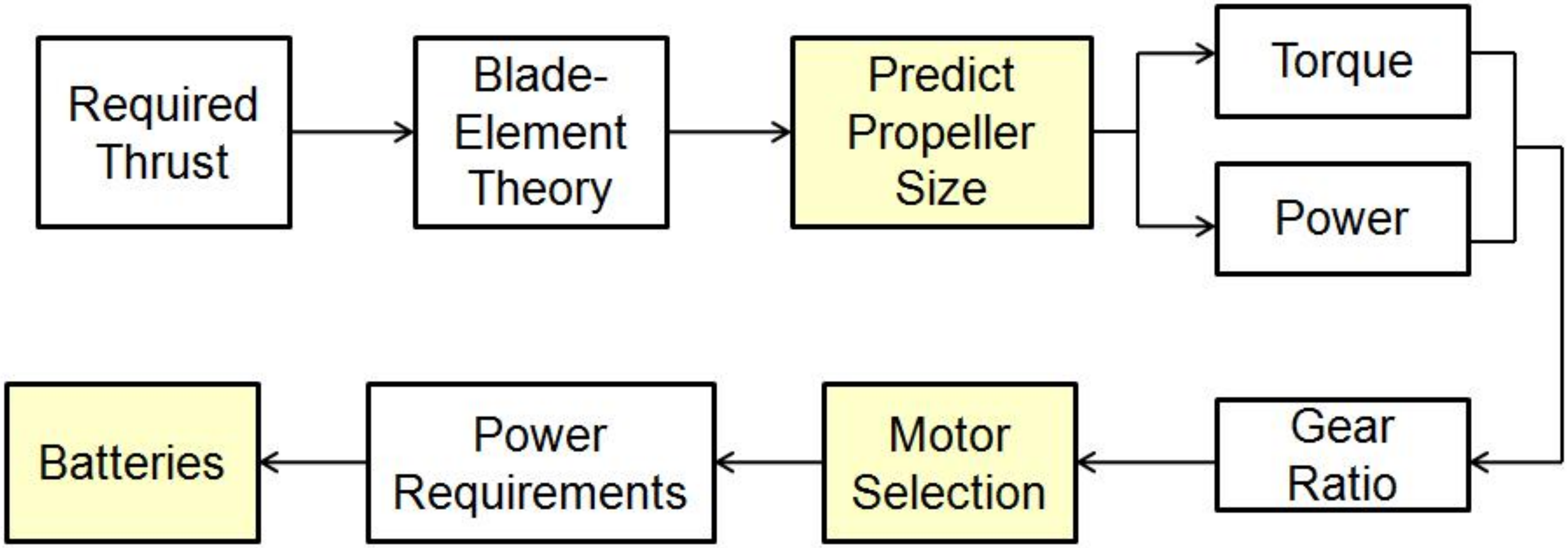
Microcontroller vs. FPGA Trade Study

MCU vs. Antifuse FPGA Trade Study V1.0					
Criteria	**Weight (%)**	**Microcontroller**	**Grade**	**Antifuse FPGA**	**Grade**
Radiation Tolerance	30%	Logical	2	Physical (rad hard by design)	5
Programming Language	20%	C	4	VHDL or Verilog	2
Power consumption	15%	16.5 mW	4	<16.5 mW	5
Cost per unit	10%	$15.05	4	$30	2
Initial Cost	5%	$0.00	5	$500	2
In Flight Programmable	5%	Yes	5	No	1
CubeSat Legacy	15%	Extensive	3	Unknown	1
Average Score			**3.8571**		**2.57143**
Weighted Score			**3.35**		**3.15**

Engine Selection Trade Study

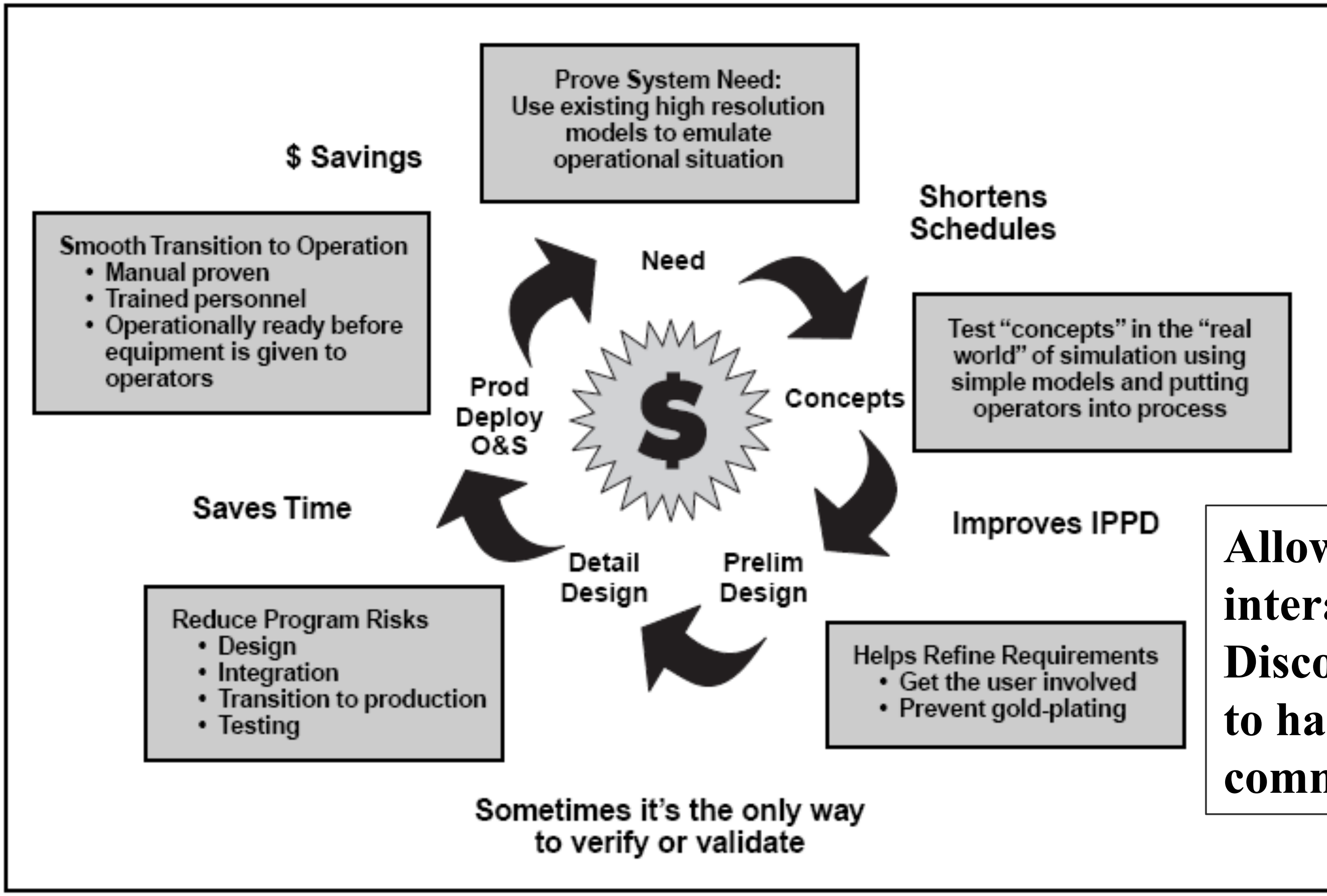
Define scope of trade study
↓
Initial research and analysis
↓
1st elimination round
↓
In-depth research and analysis
↓
2nd elimination round
↓
Final 4 engines: quad chart & matrix rating
↓
Final selection

Maneuvering System – Initial Trades (2)



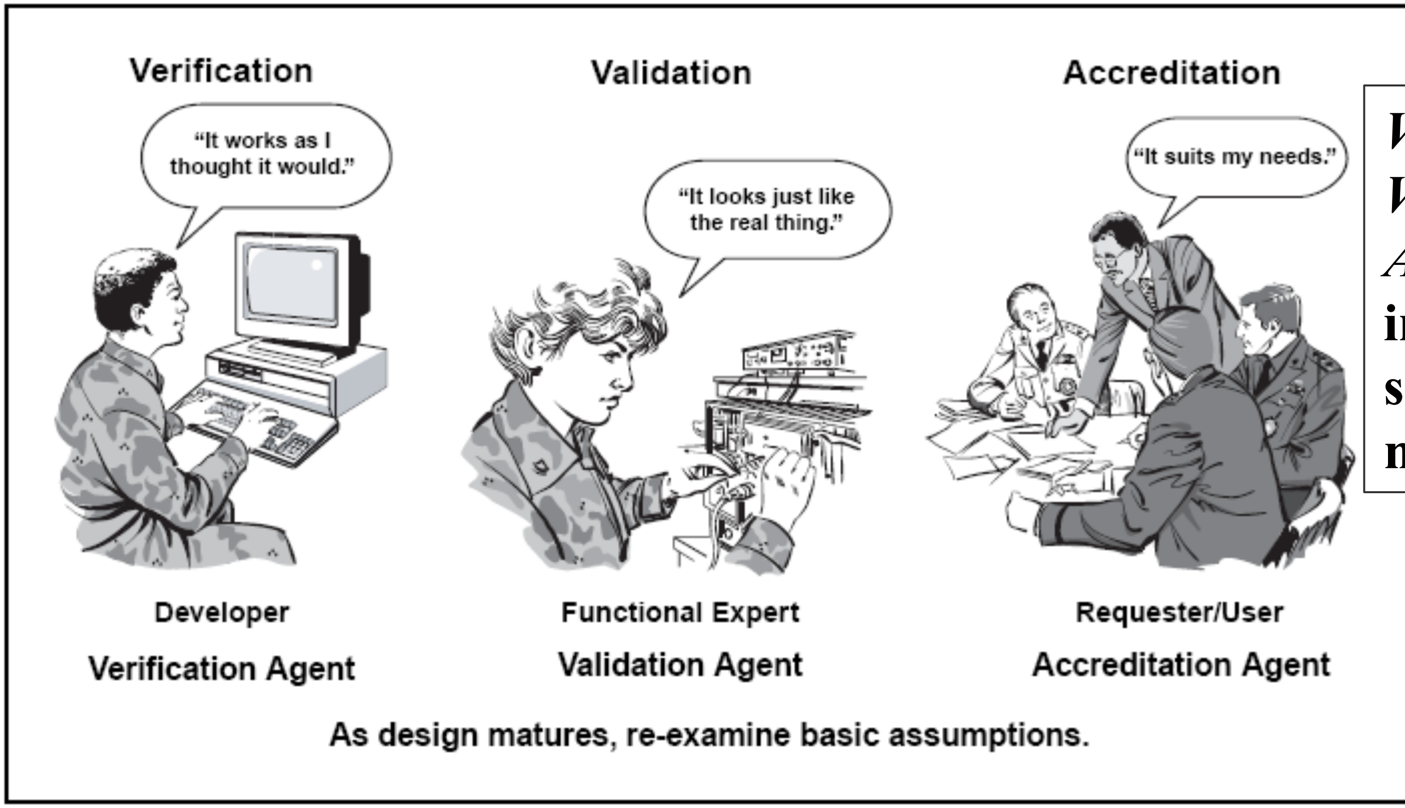
- Rotor, Drive Mechanisms, and Power Component Selection Process
- Brushless DC-motors, direct propeller drive
- Matched Electronic Speed Controllers
- Batteries Selected to Meet Brake Power Requirements

Modeling and Simulation



Allows complex interactions to be Discovered prior to hardware commitments

Modeling and Simulation (2)

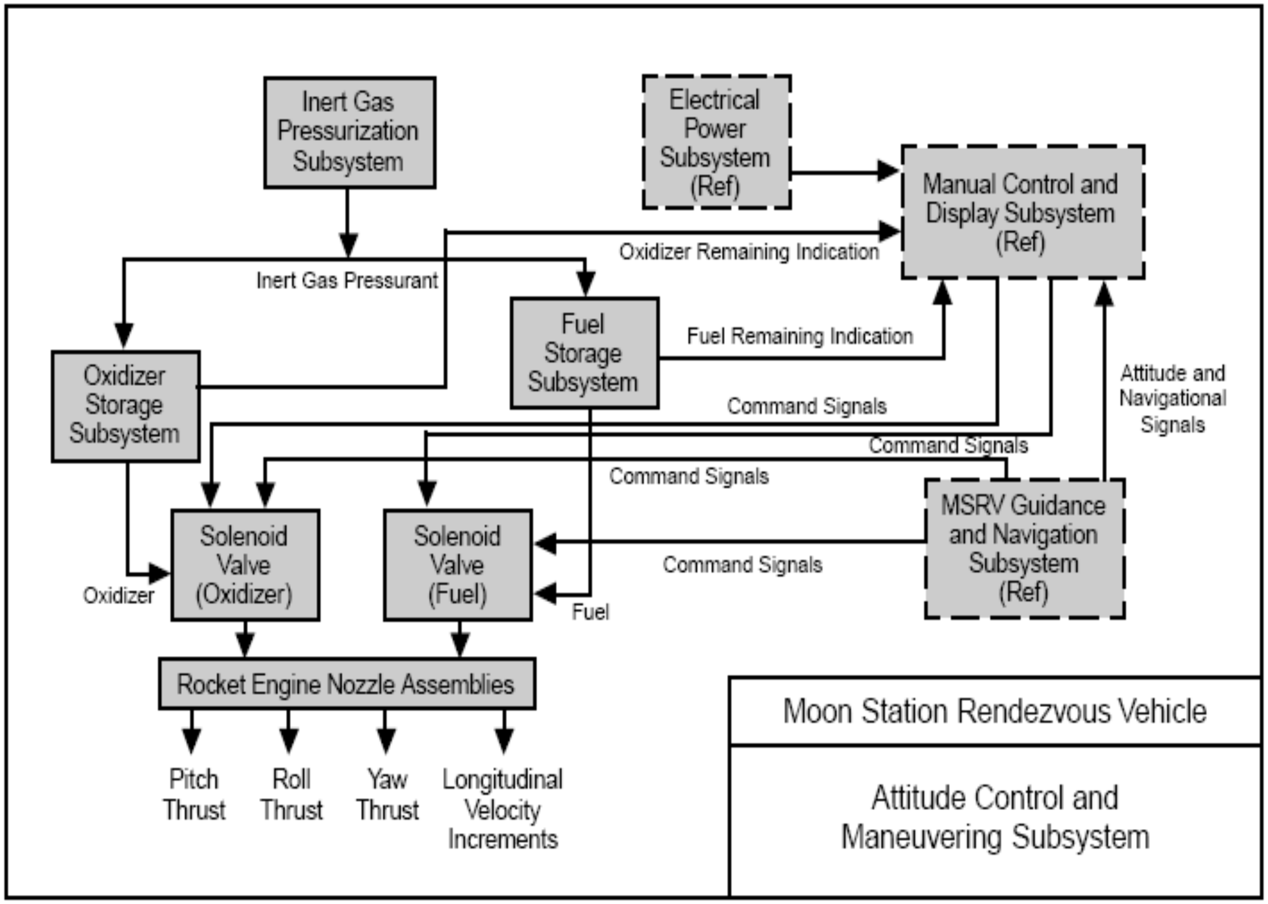


***Verification, Validation, and Accreditation* are integral part of simulation and modeling process**

UtahState UNIVERSITY

Mechanical & Aerospace Engineering

Functional Block Diagrams



Schematic Block Diagram Example

Schematic Block Diagram (SBD) depicts hardware and software components and their interrelationships.

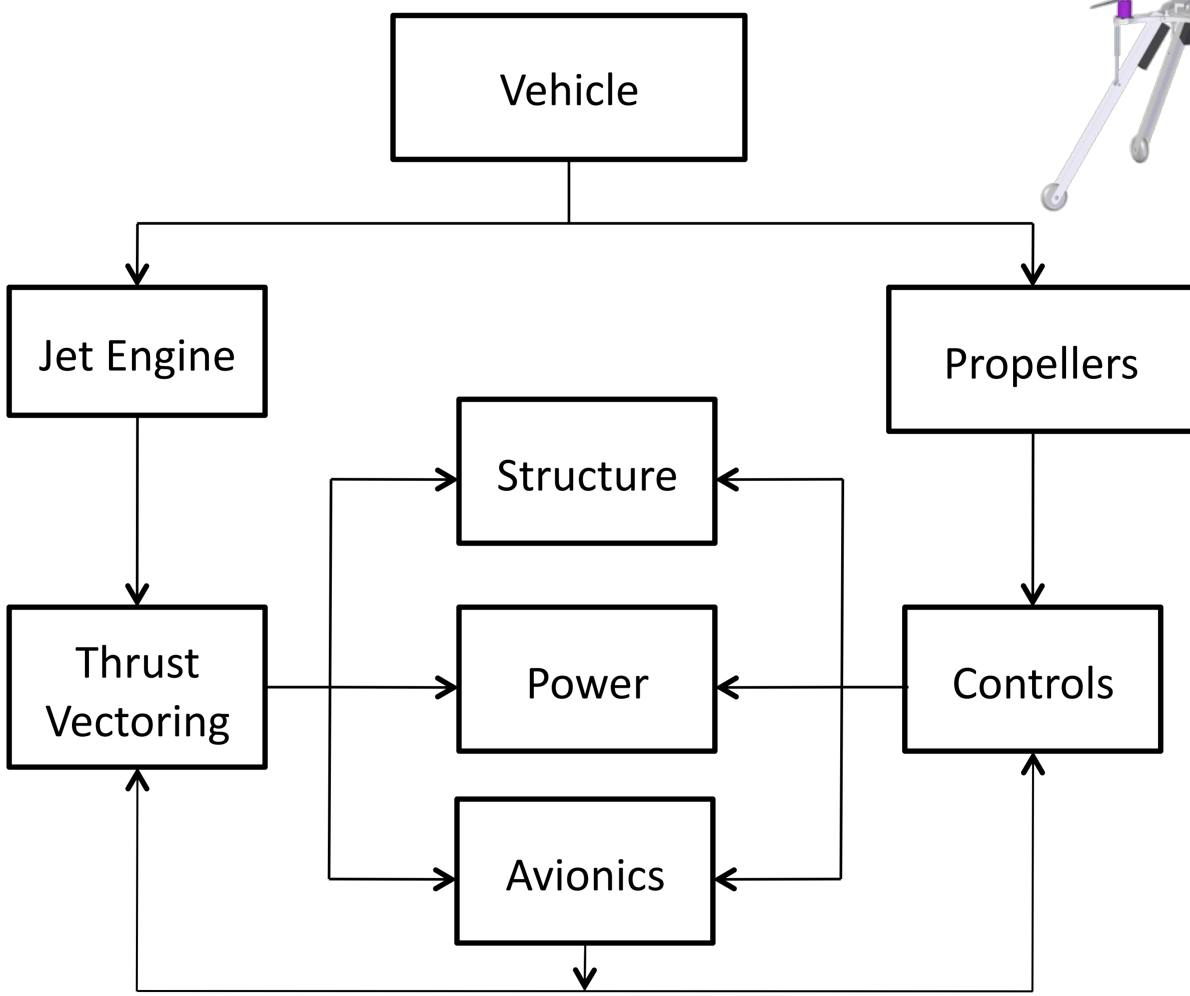
Developed at successively lower levels as analysis proceeds to define lower-level functions within higher-level requirements.

Useful for developing Interface Control Documents *(ICD's)*

UtahState UNIVERSITY

Mechanical & Aerospace Engineering

LPSLRV Functional Elements

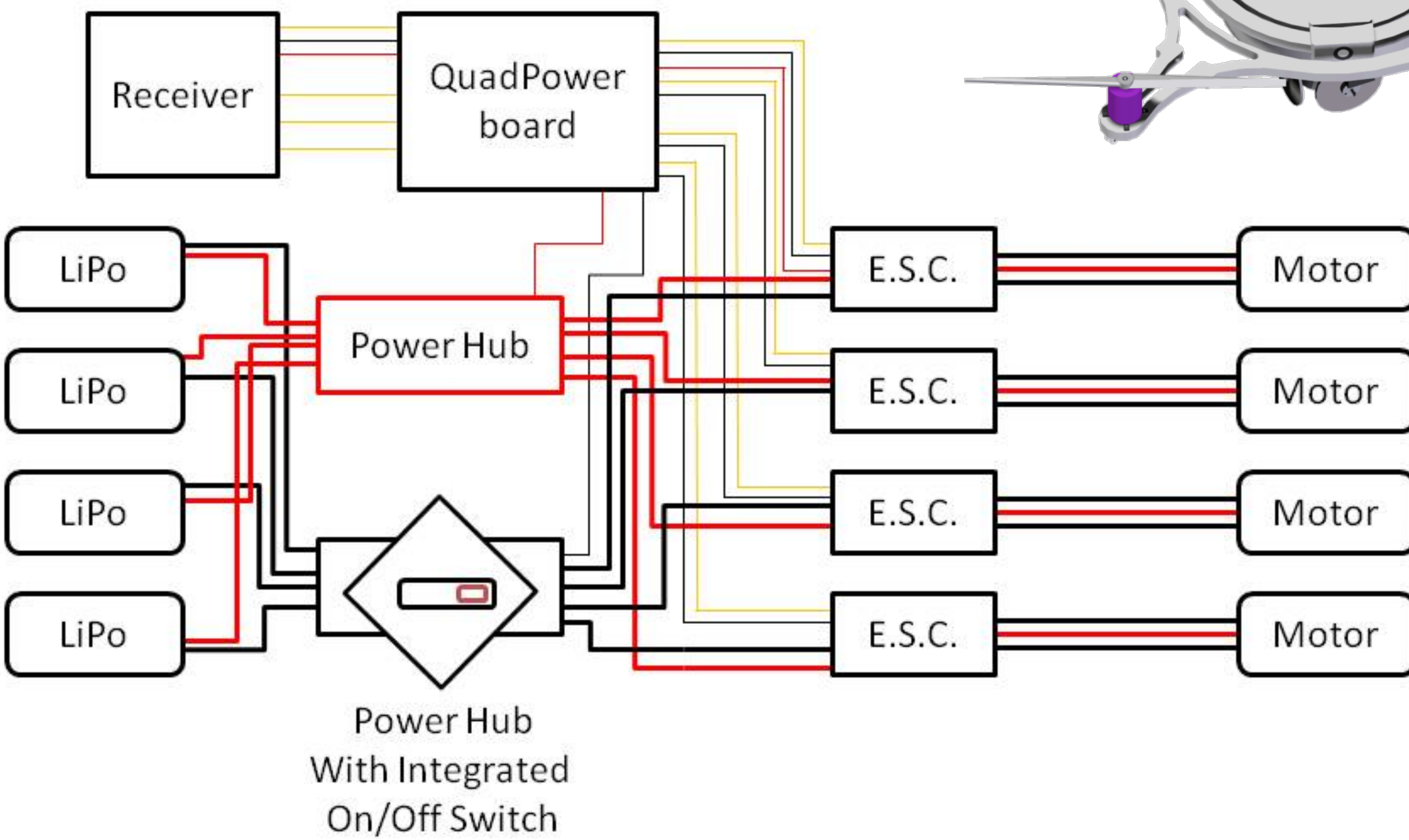


UtahState UNIVERSITY

Mechanical & Aerospace Engineering

Power Distribution System

Outer Platform
Power System Distribution Diagram



Interface Control Document (ICD)

-- ICD's define how the block within the SBD schematic are actually "connected"

--Interface control documents are a key element of systems engineering as they define and control interface(s) of a system, and bound its requirements.

-- The purpose of the ICD is to communicate all possible inputs to and all potential outputs from a system for some potential or actual user of the system.

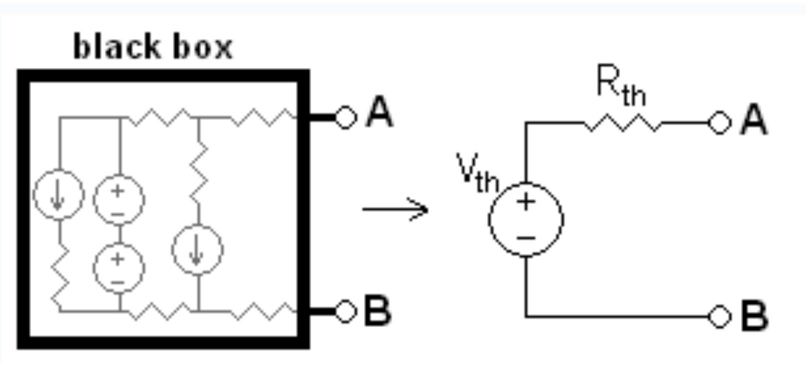
-- An ICD should only describe the interface itself, and not the characteristics of the systems which use it to connect -- The function and logic of those systems should be described in their own design documents.

Interface Control Document (2)

-- Allows Disparate groups to work integrate sub-systems without complete working knowledge of what is inside of the “black box

-- In this way, independent teams can develop the connecting systems which use the interface specified, without regard to how other systems will react to data and signals which are sent over the interface.

-- An adequately defined ICD will allow one team to test its implementation of the interface by simulating the opposing side with a simple communications simulator.



UtahState UNIVERSITY

Mechanical & Aerospace Engineering

Interface Control Document (3)

Example ICD

ANALOG SIGNAL NAME(INTER-MODULE)	WHAT SYSTEM	HOW COMMUNICATED TO C&DH	Voltage Range	PIN #'s	CONNECTOR	COMMENT
Power Voltages						
Gnd	EPS	NA	0V			signal ground
3.3V regulated supply	EPS	CDH MCU ADC (PF0)	3.3V			C&DH power
5.0V regulated Supply	EPS	CDH MCU ADC (PF1)	5V			TNC power, XCVR power
unregulated supply	EPS	CDH MCU ADC (PF2)	3.3-4.25			nothing directly uses
EPS Voltages						
bat1	EPS	via I2C from ADC1 on EPS	3.7V	I2C		
bat2	EPS	via I2C from ADC1 on EPS	3.7V	I2C		
solar cell output	EPS	via I2C from ADC1 on EPS	2.5V	I2C		the entire array
unregulated supply	EPS	via I2C from ADC1 on EPS	3.3-4.25V	I2C		nothing directly powered from this
EPS Currents (sent as voltage)						
bat1 charging current	EPS	via I2C from ADC3 on EPS		I2C		
bat 1 discharging current	EPS	via I2C from ADC3 on EPS		I2C		
bat2 charging current	EPS	via I2C from ADC3 on EPS		I2C		
bat2 discharging current	EPS	via I2C from ADC3 on		I2C		

Interface Control Document (4)

Example ICD

TNC Interface V1.0				
Pin			**Description**	**MCU Pin**
1	CTS	Clear to Send	RS-232 level flow control signal out of the TNC. Indicates whether the TNC is allowing or holding off data input on pin 3.	PE4
2	RXD	Receive Data	RS-232 level data out of the TNC.	PE0
3	TXD	Transmit Data	RS-232 level data into the TNC.	PE1
4	RTS	Request to Send	RS-232 level flow control signal into the TNC. Indicates the MCU wants to send data to the TNC.	PE3
5	GND	Ground	Common signal and frame ground.	GND
6	DCD		No connection.	N/C

Table 4: MCU/TNC Interface

Power and Mass Budget Analysis

C&DH Mass Budget V1.0			
Part	Mass (kg)	Quantity	Mass Total
Memory	0.0030	1	0.0030
ATmega2561L	0.0010	2	0.0020
3V relays	0.0003	4	0.0012
I2C ADC's	0.0010	3	0.0030
I2C GPIO	0.0010	1	0.0010
Circuit Board	0.0250	1	0.0250
Crystal	0.0010	1	0.0010
Miscellaneous	0.0100	1	0.0100
Thermistors	0.0010	8	0.0080
Total	**0.0433**		**0.0542**
Contingency			**10%**
Total Plus Cont.			**0.0596**

Table 5: C&DH Mass Budget

Weight and Power growth are major enemies of any spacecraft

Power and Mass Budget Analyses Insure spacecraft growth is bounded and eventually mandates comes in “under weight” and “overpowered”

Example

UtahState UNIVERSITY

Mechanical & Aerospace Engineering

Power and Mass Budget Analysis (2)

C&DH Power Budget

Example

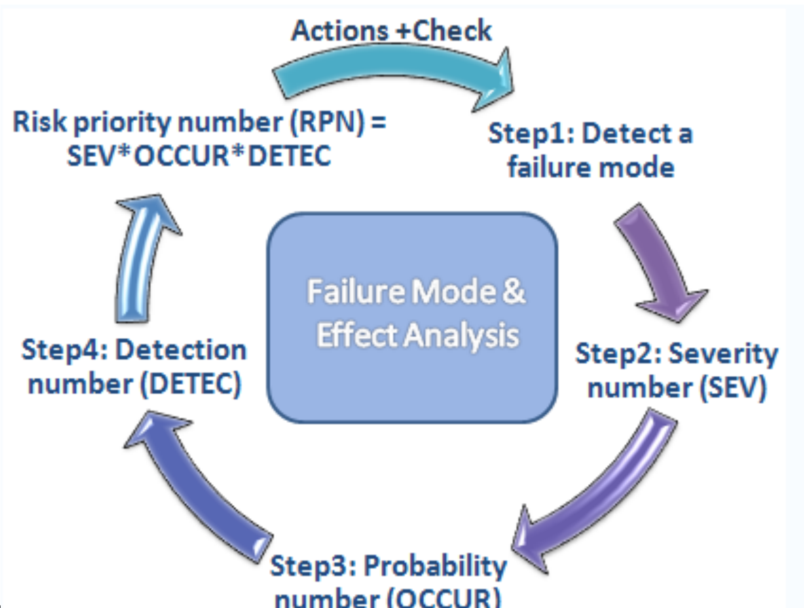
C&DH Power Budget V1.0										
		Voltage Range			Current	Power Max	Power Mode			
Part	Quantity	Min	Max	V used	mA	mW	Safe	Idle	Normal	Transmit
Memory	1	2.7000	3.6000	3.0000	4.0000	12.0000	0.1000	0.1000	1.0000	1.0000
ATmega2561L	2	1.7000	5.5000	3.0000	5.5000	33.0000	16.5000	16.5000	16.5000	16.5000
3V relays	4	1.0000	5.0000	3.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
I2C ADC's	3	2.7000	5.0000	3.0000	0.2250	2.0250	1.0125	1.0125	2.2050	2.2050
I2C GPIO	1	2.3000	5.5000	3.0000	0.1040	0.3120	1.0000	1.0000	1.0000	1.0000
Circuit Board	1	~	~	~	~	0.0000	0.0000	0.0000	0.0000	0.0000
Crystal	1	~	~	~	~	0.0000	0.0000	0.0000	0.0000	0.0000
Miscellaneous	1	~	~	~	~	0.0000	0.0000	0.0000	0.0000	0.0000
Thermistors	8	1.0000	5.5000	3.0000	0.3333	7.9992	8.0000	8.0000	8.0000	8.0000
Total					**10.1623**	**55.3362**	**26.6125**	**26.6125**	**28.7050**	**28.7050**
Contingency						**15%**	**15%**	**15%**	**15%**	**15%**
Total Plus Cont.						**63.6366**	**30.6044**	**30.6044**	**33.0108**	**33.0108**

Table 6: C&DH Power Budget

Failure Modes and Effects Analysis (FMEA)

-- A failure modes and effects analysis (FMEA) is a procedure for analysis of potential failure modes within a system for classification by severity or determination of the effect of failures on the system.

--FMEA provides an analytical approach, when dealing with potential failure modes and their associated causes.



***Failure mode*:** The manner by which a failure is observed; it generally describes the way the failure occurs.“

***Failure effect*:** Immediate consequences of a failure on operation, function or functionality, or status of some item

"Failure Mode Criticality Analysis (FMCA) *(1)*

Example Sources of Risk

In the "identify" activity, checklists such as this can serve as a reminder to analysts regarding areas in which risks have been identified previously.

- Unrealistic schedule estimates or allocation
- Unrealistic cost estimates or budget allocation
- Inadequate staffing or skills
- Uncertain or inadequate contractor capability
- Uncertain or inadequate vendor capability
- Insufficient production capacity
- Operational hazards
- Issues, hazards, and vulnerabilities that could adversely affect the program's technical effort
- Unprecedented efforts without estimates
- Poorly defined requirements
- No bidirectional traceability of requirements
- Infeasible design
- Inadequate configuration management
- Unavailable technology
- Inadequate test planning
- Inadequate quality assurance
- Requirements prescribing nondevelopmental products too low in the product tree
- Lack of concurrent development of enabling products for deployment, training, production, operations, support, or disposal

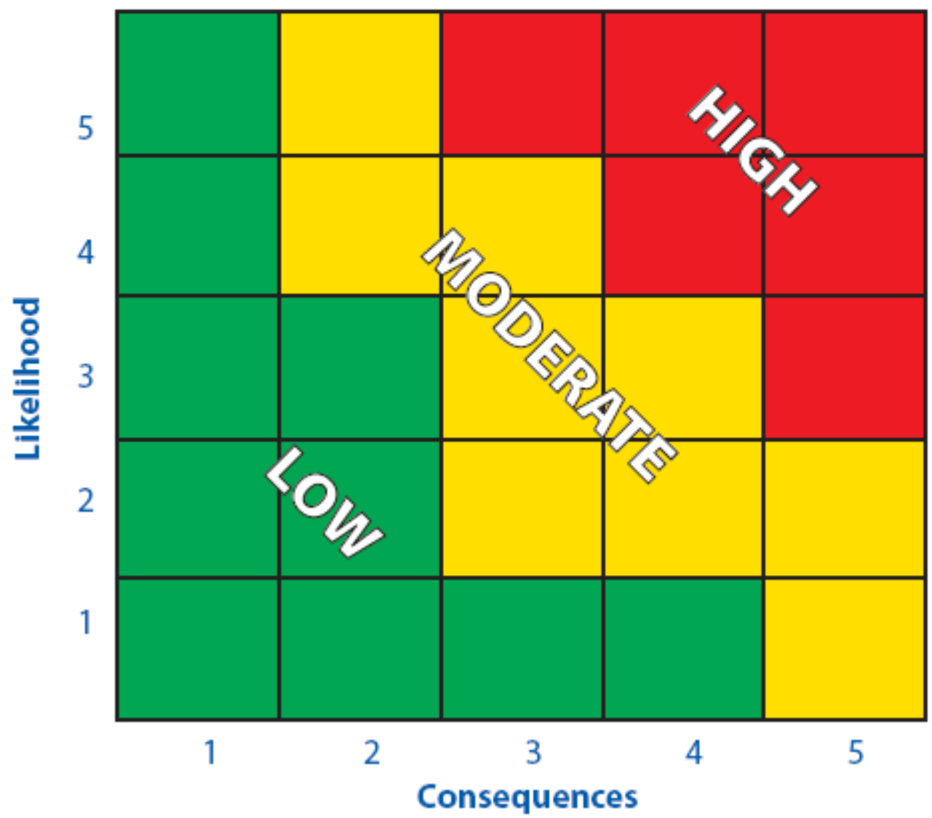
"Failure Mode Criticality Analysis (FMCA) *(2)*

Hazard Assessment Matrix

Risk matrices provide assistance in managing and communicating risk.

Qualitative and semi-quantitative measures of likelihood with similar measures of consequences.

Track the status and effects of risk-handling efforts,
And precisely Communicate risk status information.



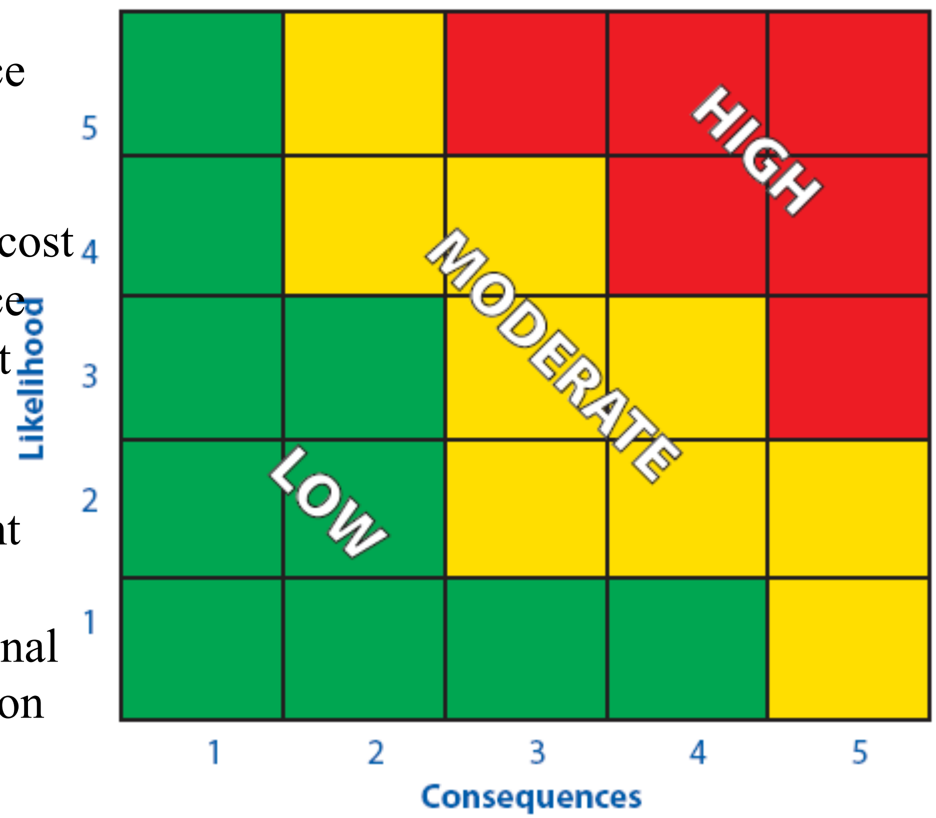
"Failure Mode Criticality Analysis (FMCA) (2)

Hazard Assessment Matrix

Low (Green) Risk: Low potential for cost increase, schedule disruption, or performance degradation. .. acceptable risk.

Moderate (Yellow) Risk: May cause some cost increase, schedule disruption, or performance degradation. Special action and management attention may be required to handle risk.

High (Red) Risk: Likely to cause significant cost increase, schedule disruption, or performance degradation Significant additional action and high-priority management attention will be required to handle risk.



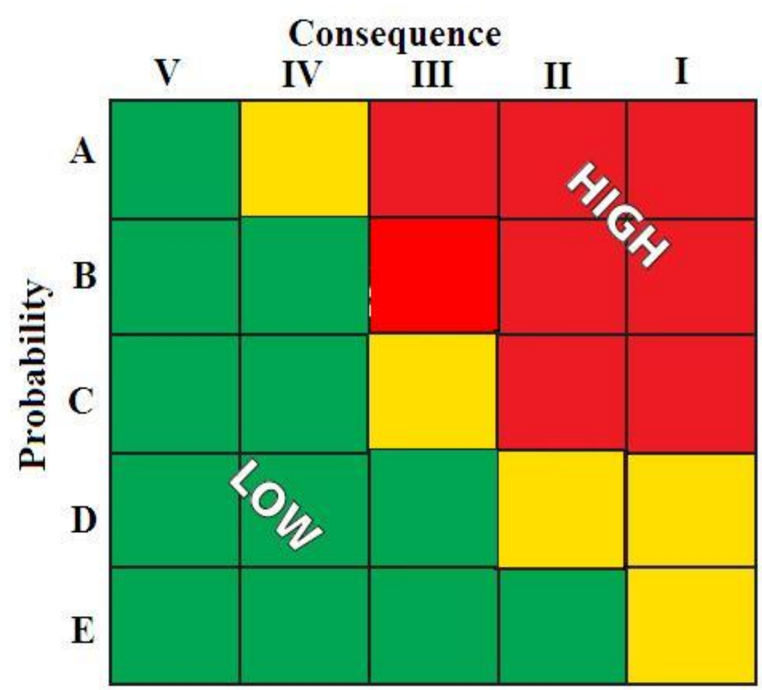
Flight Safety Hazard Assessment Matrix

Risk Consequence:

I	-	*Catastrophic, Loss of Vehicle, Death of Crew*
II	-	*Severe, Significant Damage to Vehicle, Injury to Crew*
III	-	*Minor Damage to Vehicle, Potential for Minor Crew Injury Loss of Mission Objectives*
IV	-	*Nuisance, Most Missions Objectives Accomplished*
V	-	*No Mission Impact*

Failure Probability:

A	-	*Likely (1 -- 1/10)*
B	-	*Probable (1/10 -- 10^{-2})*
C	-	*Unlikely (10^{-2} -- 10^{-3})*
D	-	*Very Unlikely (10^{-3} --10^{-5})*
E	-	*Remote < 10^{-5}*



Items in Yellow require NASA management Waiver, Shuttle flies with III,C & management waiver

Risk Assessment

Example Hazard Tracking List

Hazard Level	Hazard	Causes	Preventative Measures
16	Engine Failure, causing an inability to keep vehicle in air	Debris Weather Temperature	Screen on jet intake, Check flying conditions, Pre-flight checklist Pre-flight and in-flight systems check
9	Human Injury	Burns from Jet Engine Exhaust Blowing debris Low-Voltage Electrical shock	Wear protective equipment, Designate "Keep out" zones, No power during maintenance, Follow manufacturer's recommendations, Follow checklists
8	Electronics Failure, causing a loss of power to rotors	Communication loss Communication interference Electrical shorting	Pre-flight and in-flight systems check
8	Vibration Effects, causing the vehicle to become unstable or components to become loose	Rotors rotating near Resonance	Pre/Post assembly testing
4	Fuel Leakage, forcing the time of the mission to be reduced	Bad seal on Fuel Tank, Improper filling of Fuel Tank	Quality check, Pre-flight checklist

Test Checklist

DAY OF TEST (typical)

0400 Mx Prep

0600 Crew Brief

07:30 Aircraft "Crew Ready"

07:45 Aircrew Step to Aircraft

08:15 Engine Start

08:45 Taxi

09:00 Takeoff

THINGS TO THINK ABOUT

- GO-NO-GO ITEMS
- WEATHER
- Equipment STATUS
- Contingency Options
- LIMFACS (bandwidth, test site availability, range availability)
- CONTROL ROOM POSITIONS
- COMM SETUP

Rapid Prototyping

Rapid prototyping (RP) can be defined as a group of techniques used to quickly fabricate a scale model of a part or assembly using three-dimensional computer aided design (CAD) data.

RP has obvious use as a vehicle for visualization. In addition, RP models can be used for testing, such as when an airfoil shape is put into a wind tunnel. RP models can be used to create male models for tooling, such as silicone rubber molds and investment casts. In some cases, the RP part can be the final part, but typically the RP material is not strong or accurate enough.

Rapid Prototyping (2)

The reasons of Rapid Prototyping are

To increase effective communication.
To decrease development time.
To decrease costly mistakes.
To minimize sustaining engineering changes.
To extend product lifetime by adding necessary features and eliminating redundant features early in the design.

Rapid Prototyping decreases development time by allowing corrections to a product to be made early in the process. By giving engineering, manufacturing, marketing, and purchasing a look at the product early in the design process, mistakes can be corrected and changes can be made while they are still inexpensive.

Eight Rules for Prototyping

1 Recognize That Ideas Are Cheap – Given the connected, Internet-savvy world in which we live, ideas have become cheap and they will probably become cheaper with time. The expense lies in testing and verifying what has economic value.
A great prototype is often the best way to start a dialogue with potential customers and test your idea's value.

2 Start with a Paper Design – You may be eager to start coding or designing the electronics too quickly. Fight the urge. Writing code without real consideration for several design factors leads to heartache and a lot of rework. Start with a simple paper design. For a user interface or Web software prototype, a paper design is efficient and effective for quickly working through the functionality.
You can get peers and, hopefully, customers to give feedback on where images, text, buttons, graphs, menus, or pull-down selections are located. Paper designs are inexpensive and more valuable than words.

Eight Rules for Prototyping (2)

3 Put in Just Enough Work – Know your objectives and stick to them. There are two good reasons to prototype: the first is to test the feasibility of a hardware or software architecture, and the second is to create a demonstration and gain customer feedback so you can price and put a value on your innovation. Keep these objectives in mind and be careful not to fall in love with the process. Prototyping is fun and innovators love to tinker, but you want to invest just enough time and work to meet the objectives.

4 Anticipate for Multiple Options – Design your prototype with modularity in mind. Great prototypes are often modular, which means you can quickly adapt them to meet customers' unforeseen needs. Customers ultimately decide how to use your product, not you. Design in options for expansion, performance, packaging, and lower cost.

Eight Rules for Prototyping (3)

5 Design for Reuse in the Final Product – The ideal situation is to design a prototype you can produce and distribute in high volume. Not many prototyping tools can deliver on this promise. Typically you give up performance for design flexibility. Look for prototyping tools that make it possible for you to scale your prototype from lab to market.

6 Avoid Focusing on Cost Too Early – For hardware designs, a potential time sink and pitfall is getting caught up in endless cost optimization analysis during the early stages of your prototype design. Cost is always important, but your goal with a prototype is to be within striking distance of a profitable design. Initially, focus on proving the value of your innovation, and design with modularity in mind. While frustrating, your design may follow many paths that do not ultimately lead to value. Focus on securing your first set of customers and then work on cost optimization.

Eight Rules for Prototyping (4)

7 Fight "Reversion to the Mean" – When prototyping, the tendency is to develop something easy rather than develop something that has a "wow" factor. Stay true to your vision and make sure your prototype captures the original thought of your innovation.

8 Ensure You Can Demonstrate Your Prototype – Your prototype should be easy to demonstrate. With customers, venture capitalists (VCs), and potential employees, you want to start strong and show the most amazing capabilities first. Do not build up to a crescendo. Most people's attention spans are limited to less than 60 seconds. In presentations, whether they are for a new employee or a VC, get to the demonstration as fast as possible. If the demonstration is amazing, all else falls into place.

http://zone.ni.com/devzone/cda/pub/p/id/579?metc=mtnxdy

Keys to Holding a Successful Meeting

- Meetings are essential to any team effort, be it designing a rocket System, or launching a new cosmetic product

- Done properly, meetings can quickly disseminate information, solve problems, create consensus, and get everyone "on the same page"

- Done improperly, meetings can bog down, cause dissention, delay, and sometimes cripple a project.

- Every meeting must a specific purpose – before arranging a meeting one need to think precisely about what it is that needs to be accomplished.

Keys to Holding a Successful Meeting (2)

- **Typical Meeting Purposes"**

  Brainstorming new ideas
  Developing an idea or plan
  Having a progress update
  Technical interchange
  Considering options and making a collective decision
  Selling something to a potential buyer
  Building a relationship with somebody

There may be a mixture of objectives and desired outcomes for a particular meeting, however, primary objectives should kept clearly in mind and those should prioritized above others.

Keys to Holding a Successful Meeting (3)

1. **Invite the right people. Make sure these people attend.**

2. **Start with a clear objective for the meeting. Particularly with routine meetings, it's tempting to hold the meeting because it's "checking a box", but what are you really trying to accomplish? People don't actually bond very much in unproductive meetings that lack clear objectives.**

3. ***Set up a written agenda in advance.* As you build the agenda, get real about how long it will take to address each topic. As a guideline, assume that if the goal is to make a decision, it will take four times longer than if the goal is to simply provide a status report.**

Keys to Holding a Successful Meeting (4)

4. **Formally track problem-solving and decision-making discussions. If everyone is in same room, use a flipchart or whiteboard, otherwise use electronic recording media. Appoint someone to take notes at the beginning of the meeting. Formally archive meeting notes in a data base with access to participating team members.**

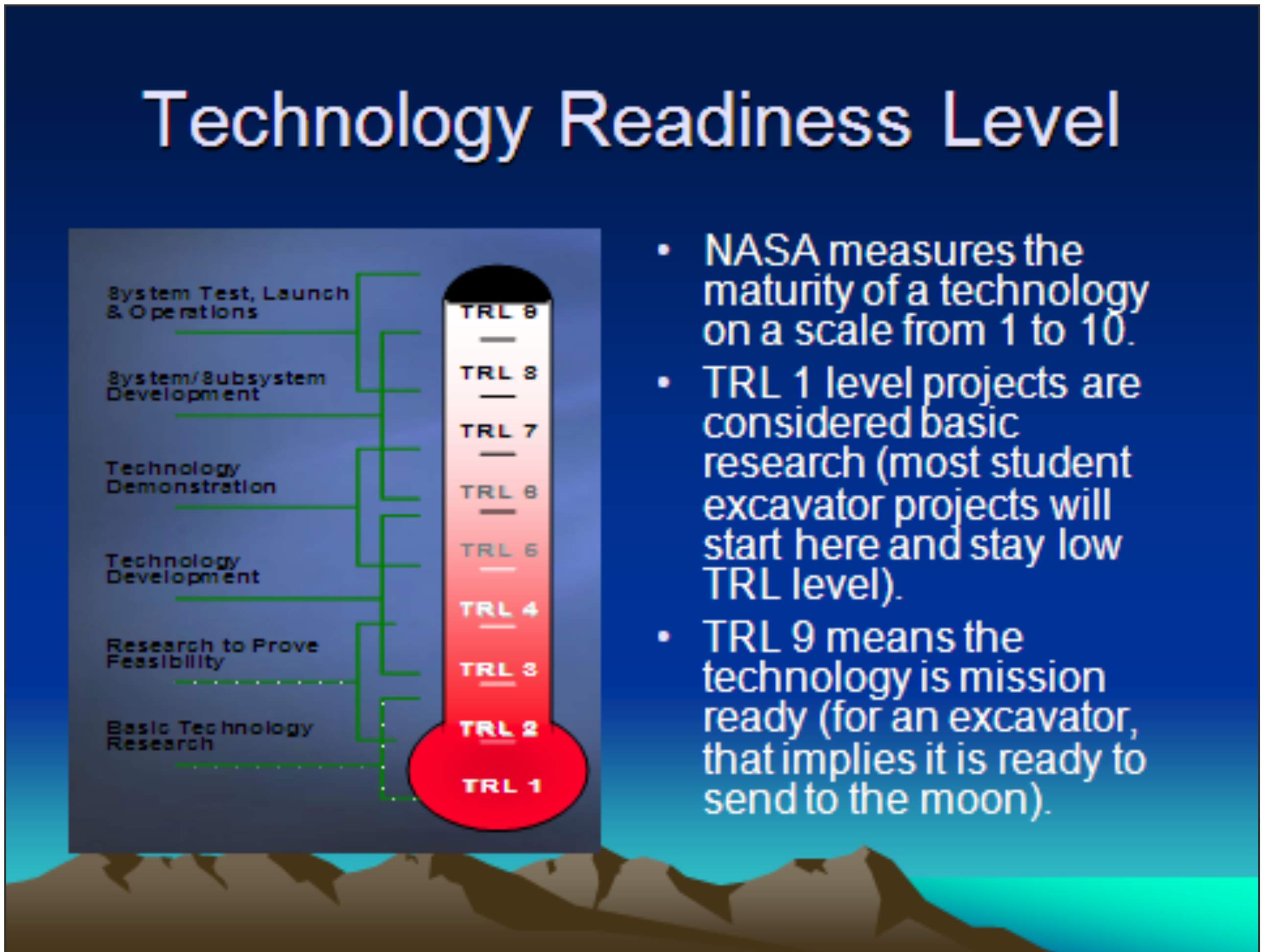
5. **Formal Tracking Tools:**
   a. Action Items – Requests for Action (RFA)
   Who is assigned action?
   When is action due?
   Who are action's "customers"
   b. Information Items – Requests for Information (RFI)
   Who provided the information and verification?
   When is action due?
   Who needs the information

Keys to Holding a Successful Meeting (5)

6. **Log and Track Action Items.. Don't let people "off the hook" require that action forms be formally CLOSED.**

7. **End each meeting with a "consensus" check. Is everyone clear on assigned actions, and due dates. FORMALLY set a tentative time and date for a follow-up meeting, and who needs to be in attendance at this meeting. Log that follow up meeting time.**

Technology Readiness Levels (TRL)



- **Designing sub-systems using high TRL components is a good way to reduce or mitigate programmatic risk.**

- **High TRL systems have "heritage" and offer increased reliability and (hopefully) enhanced ease of integration.**

Technology Readiness Levels (2)

- **Cardinal Sub-system Design Rules:**

  Integrate when can (high TRL)

  Design and fabricate when you must

  Low TRL sub-systems require significant testing and evaluation before integration

  Low TRL's can "fight" each other and have potential to seriously impact overall design budget and schedule!

- **High TRL systems have "heritage" and offer increased reliability and (hopefully) enhanced ease of integration.**

Conceptual Technology Readiness Levels, 1-5

Technology Readiness Level	Description
1. Basic principles observed and reported.	Lowest level of technology readiness. Scientific research begins to be translated into technology's basic properties.
2. Technology concept and/or application formulated.	Invention begins. Once basic principles are observed, practical applications can be invented. The application is speculative and there is no proof or detailed analysis to support the assumption. Examples are still limited to paper studies.
3. Analytical and experimental critical function and/or characteristic proof of concept.	Active R&D is initiated. This includes analytical studies and laboratory studies to physically validate analytical predictions of separate elements of the technology. Examples include components that are not yet integrated or representative.
4. Component and/or breadboard validation in laboratory environment.	Basic technological components are integrated to establish that the pieces will work together. This is relatively "low fidelity" compared to the eventual system. Examples include integration of "ad hoc" hardware in a laboratory.
5. Component and/or breadboard validation in relevant environment.	Fidelity of breadboard technology increases significantly. The basic technological components are integrated with reasonably realistic supporting elements so that the technology can be tested in simulated environment. Examples include "high fidelity" laboratory integration of components.

Prototype and Deployment Technology Readiness Levels, 6-9

Technology Readiness Level	Description
6. System/subsystem model or prototype demonstration in a relevant environment.	Representative model or prototype system, which is well beyond the breadboard tested for level 5, is tested in a relevant environment. Represents a major step up in a technology's demonstrated readiness. Examples include testing a prototype in a high fidelity laboratory environment or in a simulated operational environment.
7. System prototype demonstration in an operational environment.	Prototype near or at planned operational system. Represents a major step up from level 6, requiring the demonstration of an actual system prototype in an operational environment. Examples include testing the prototype in a test bed aircraft.
8. Actual system completed and qualified through test and demonstration.	Technology has been proven to work in its final form and under expected conditions. In almost all cases, this level represents the end of true system development. Examples include developmental test and evaluation of the system in its intended weapon system to determine if it meets design specifications.
9. Actual system proven through successful mission operations.	Actual application of the technology in its final form and under mission conditions, such as those encountered in operational test and evaluation. Examples include using the system under operational mission conditions.

(continued)


Questions??