
# Systems Engineering Part 1

**Jesse Olson**

**University Nanosatellite Program Manager**

**AFRL/RV**

# Agenda

- What is the point of this Expert Area Telecon (EAT)?
- What is Systems Engineering?
- The Systems Engineering Process
- Verification & Validation
- NASA Systems Engineering
- UNP Systems Engineering
- Moving Forward
- Want to learn more?

# References

- Significant credit to Sam Baxendale, author of the NS-10 Systems Engineering EAT series from which this heavily drew. Also references:
  - “UNP NS11 User’s Guide”, AFRL/RV, 2022
  - “Applied Space Systems Engineering”, Larson/Kirkpatrick/Sellers/Thomas/Verma, 2009
  - “Space Mission Engineering”, Wertz/Everett/Puschell, 2011
  - “The NASA Systems Engineering Handbook”, NASA, 2007
  - “INCOSE Systems Engineering Handbook”, INCOSE, 2010
  - “Michigan Tech MEPIV Lecture: An Introduction to Systems Engineering”, King, 2019
  - “UNP NS9 EAT: Systems Engineering”, Straight, 2016
  - “ISO/IEC 15288 IEEE Systems Engineering Standard”, IEEE, 2015
  - “ECSS-E-10A European Systems Engineering Standard”, ESA, 2018

# What is the point of this EAT?

**There is only so much that can be covered in an hour**

- Systems Engineering can take a whole career and/or multiple graduate degrees to truly understand
- This EAT is just the first of a series intended to introduce students to Systems Engineering

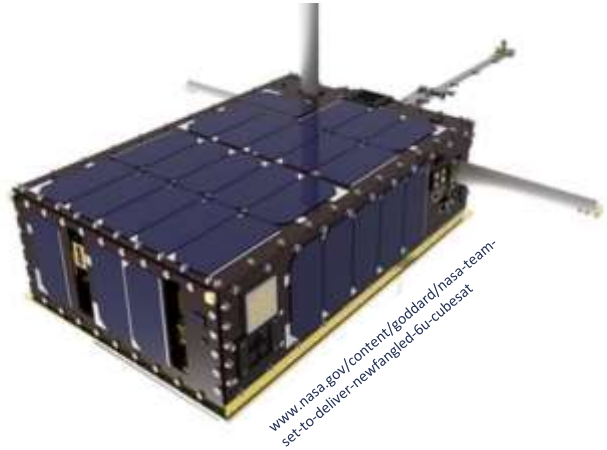
**It is assumed that most undergraduate UNP students have not been exposed to Systems Engineering before**

- It is usually the content of a senior-level course in traditional engineering curriculums, even then it is often an afterthought

**There are many ways to define and apply Systems Engineering**

- This and future EATs should help define UNP's take on Systems Engineering


??




Encourage students to view UNP from a high-level Systems Engineering perspective

# What is Systems Engineering?


"A methodical, **multi-disciplinary** approach for the design, realization, technical management, operations, and retirement of a combination of elements that function together to produce the **capability required to meet a need** – or in other words, a system. Systems engineering is the art and science of **developing an operable system capable of meeting requirements** within often **opposed constraints**."


"Systems Engineering is an **interdisciplinary** approach and means to enable the **realization of successful systems**. It **focuses on defining customer needs and required functionality** early in the development cycle, documenting requirements, then proceeding with design synthesis and system validation while considering the complete problem: **performance, cost, schedule,** operations, training & support, manufacturing, test, and disposal."


"The M.S. Systems Engineering program at Johns Hopkins University provides in-depth knowledge and technical skills that address the needs of engineers and scientists engaged in all aspects of analysis, design, integration, production, and operation of modern systems. The systems engineering process coordinates and leads the **translation of an operational need into a system designed to meet that need**. It integrates the inputs of **all the required technical disciplines** into a coordinated effort that meets established **performance, cost, and schedule goals**."

Systems Engineering is a vast discipline applied across numerous areas of industry and academia that consider the entire life cycle of an engineered product

# What is Systems Engineering?

**What is a system?**

- INCOSE: “A system is a construct or collection of different elements that together produce results not obtainable by the elements alone.”
- IEEE: “A set of interrelated components working together as an integrated whole to achieve some common objective.”

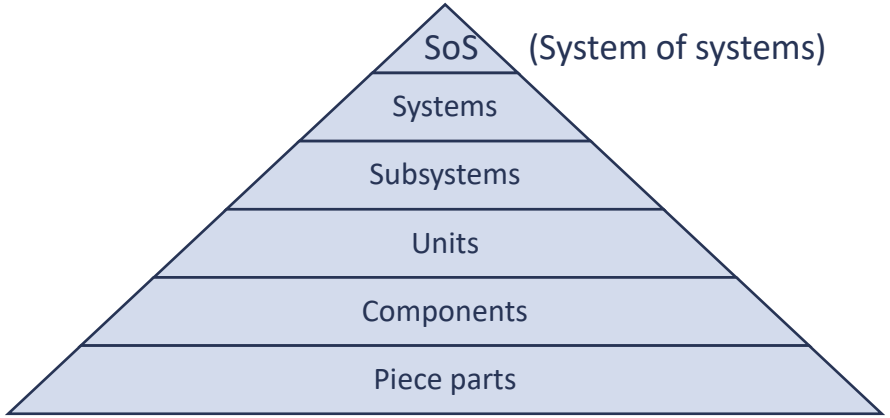
**Why do we need Systems Engineering?**

- Informal SE has been practiced throughout time (Pyramids, Great Wall of China, etc.)
- Formal SE started with the emergence of complex engineering systems in the 20th century (aircraft, automobiles, lunar program, etc.)
- Modern systems are complex, typically comprising of thousands of parts with millions of potential interactions

**How does Systems Engineering differ from traditional engineering disciplines (e.g. mechanical, electrical, etc.)?**

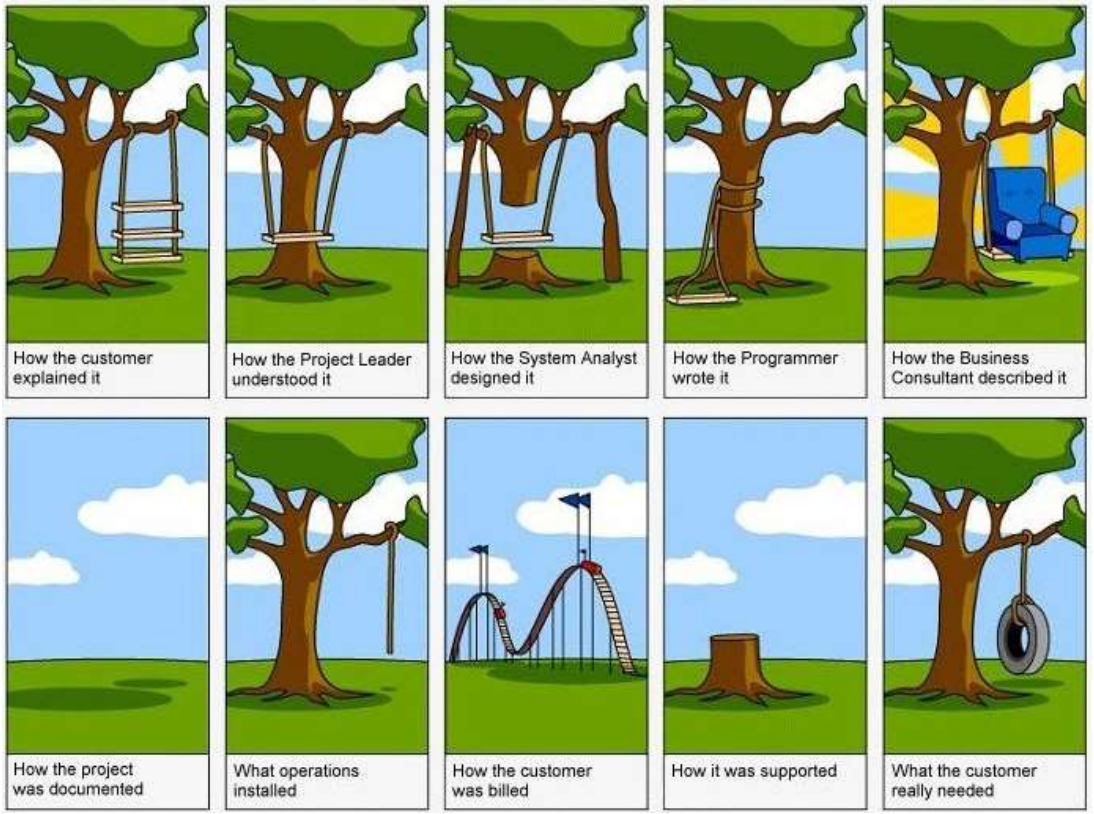
- SE focuses on the system as a whole, emphasizing its total operation, considering both internal and external interactions
- SE bridges all the traditional engineering disciplines together to lead an idea from concept through operation that reflects the users’ needs

www.pinterest.com/pin/471471273190367 68/



# What is Systems Engineering?


www.pinterest.co.uk/pin/277956608238483646/



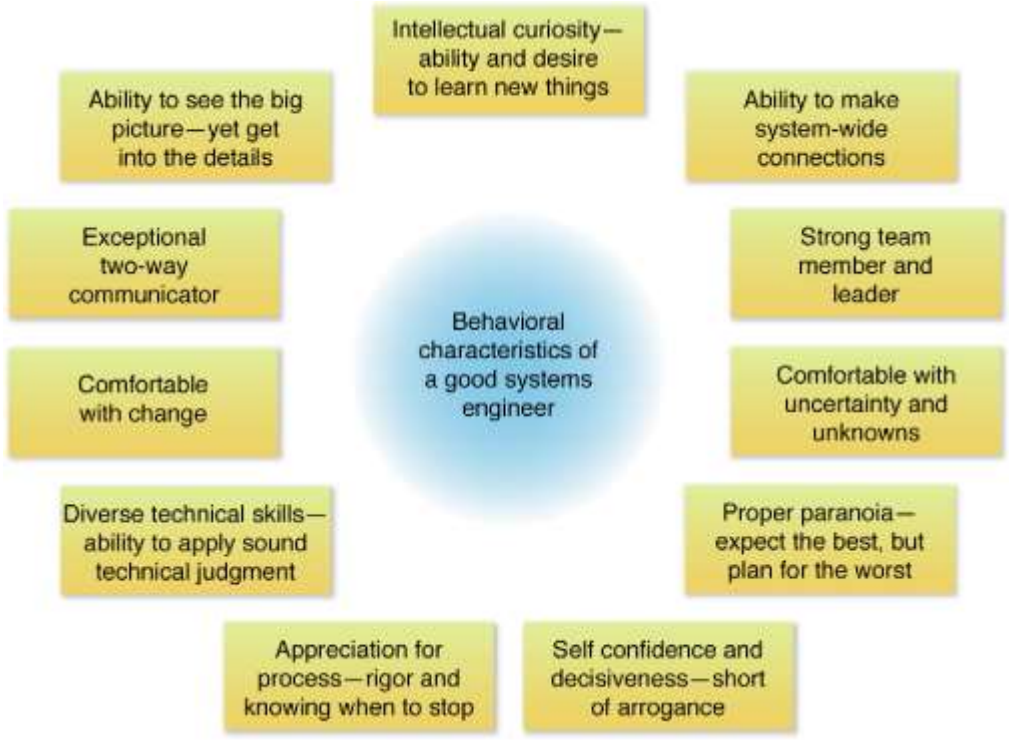

endigit.com/2019/1/how-create-requirements-document-and-nail-your-project

Who can prevent this from happening?
**Systems Engineers!**

# What is Systems Engineering?

## Effective Systems Engineers…

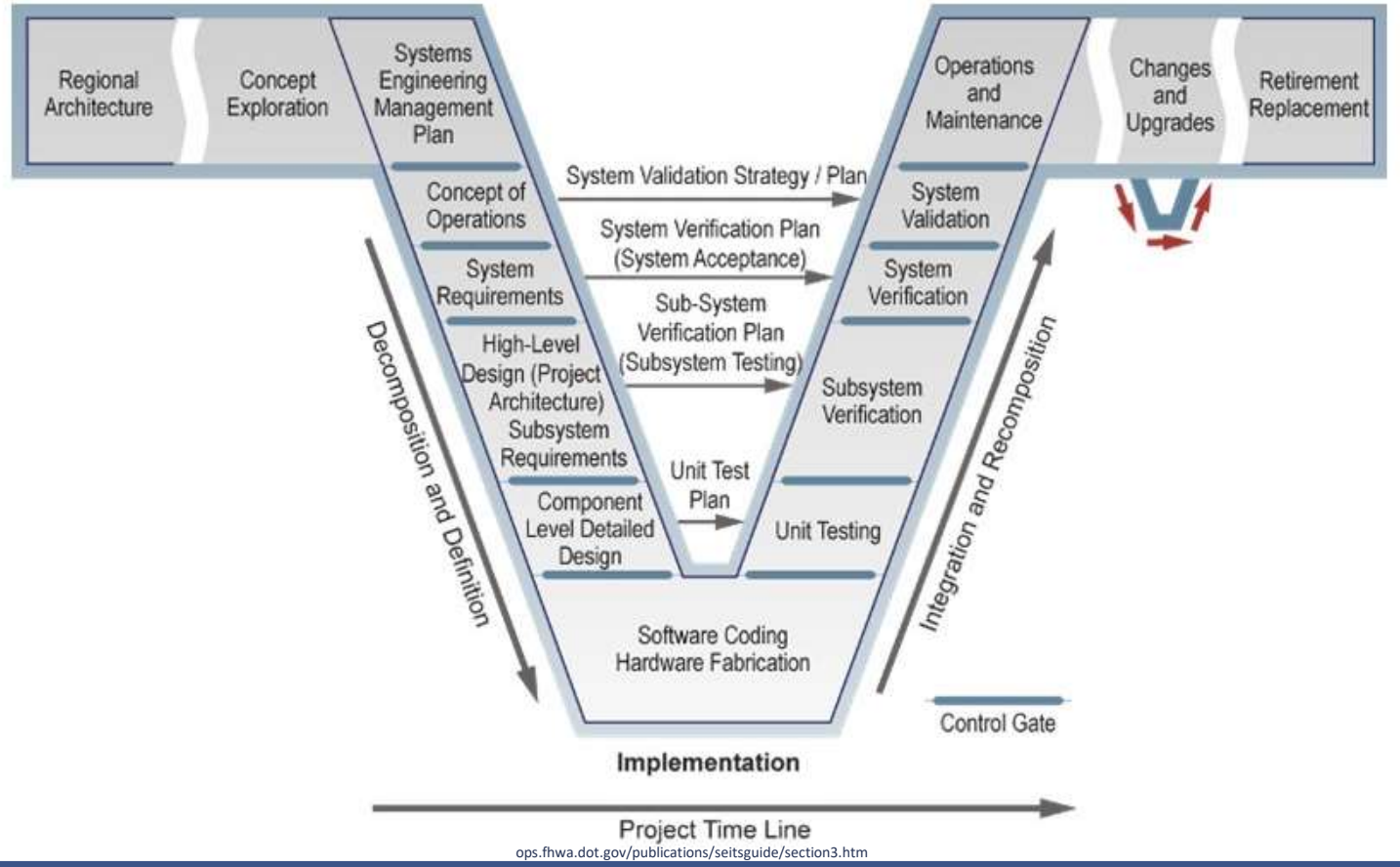
- Know that all programs start and end with people
- Are masters of decision making in environments plagued by incomplete or conflicting information and objectives
- Understand that system development processes are not truly "linear." They can plan for and mitigate the effect of design iterations/rework on technical and programmatic outcomes
- Are the ultimate authority on how the system's technical and programmatic integrity can be simultaneously achieved
- Decide which technical risks are worth undertaking and which are not
- Do not necessarily seek the "best performing" system, instead they seek the "best balance" of critical attributes
- Measure their effectiveness not by a program's spectacular successes, but rather by the absence of program difficulties




www.nasa.gov/pdf/311199main_Art_and_Sci_of_SE_SHORT_1_20_09.pdf

Systems engineers lead the translation and realization of an operable system that meets requirements within imposed constraints

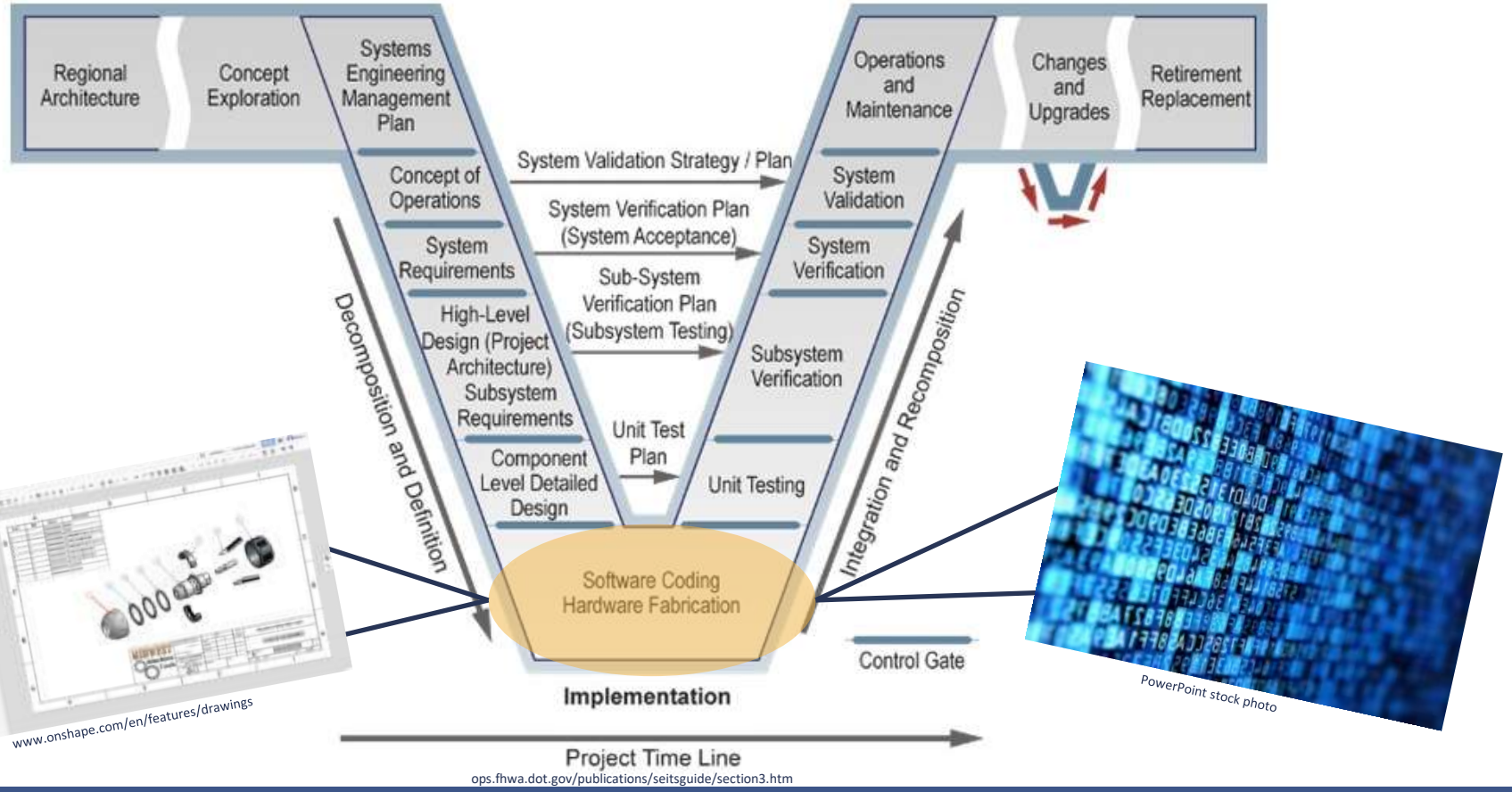
# What is Systems Engineering?




ops.fhwa.dot.gov/publications/seitsguide/section3.htm

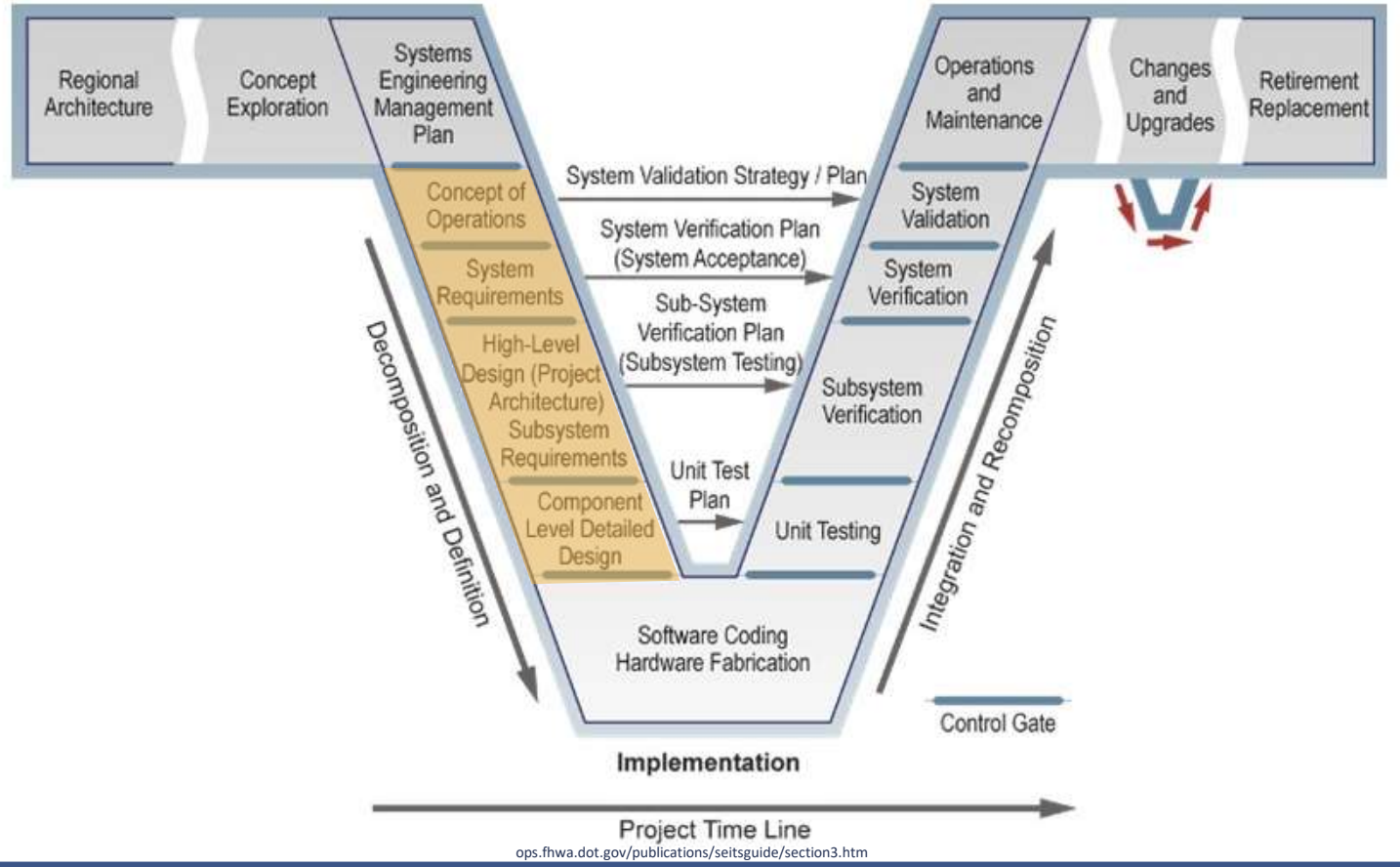
The infamous “V-Model of Systems Engineering”

# What is Systems Engineering?




As engineers, it may be tempting to jump right to the bottom of the “V” (Hint: don’t)

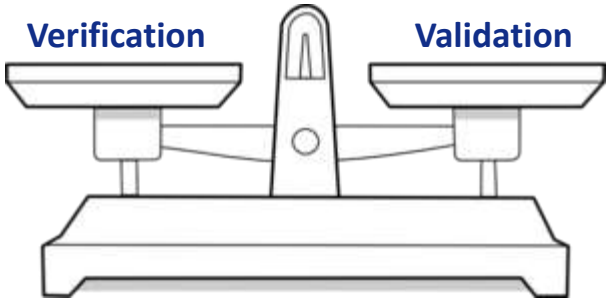
# What is Systems Engineering?




ops.fhwa.dot.gov/publications/seitsguide/section3.htm

True design occurs on the left side of the "V", before you ever open SolidWorks or Altium

# Verification and Validation




*"Neither the world's greatest design, poorly implemented – nor a poor design, brilliantly implemented – is worth having." – Ryschkewitsch et al, 2009*

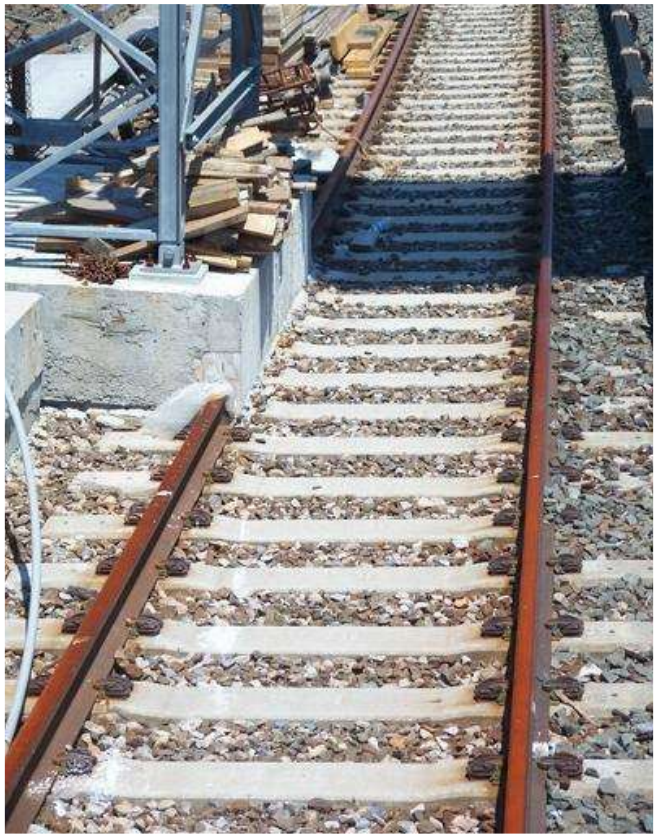
The true meanings of **Verification** & **Validation** are often convoluted in the real world, here is a definition UNP uses….

| | Verification | Validation |
|---|---|---|
| **Short & sweet** | A process to ensure you built the system right | A process to ensure you built the right system |
| **Detailed (IEEE)** | The process of evaluating a system to determine whether the products of a given development phase **satisfy the conditions imposed at the start** of that phase | The process of evaluating a system to determine whether it **satisfies the stakeholders** of that system |
| **Primary tool (UNP)** | RVM | Mission Design Document |
| **Questions to ask** | • Does it meet the requirements you imposed at the start of the project?<br>• Do the units, subsystems and system operate as intended by their design specifications? | • Do the stakeholders like it?<br>• Did you miss any requirements?<br>• Do operability issues render the system useless? |

# Verification and Validation

www.dailymail.co.uk/femail/article-11113259/Hilarious-photos-major-building-fails-including-stairs-nowhere.html

www.pinterest.com/pin/303148618661258479/



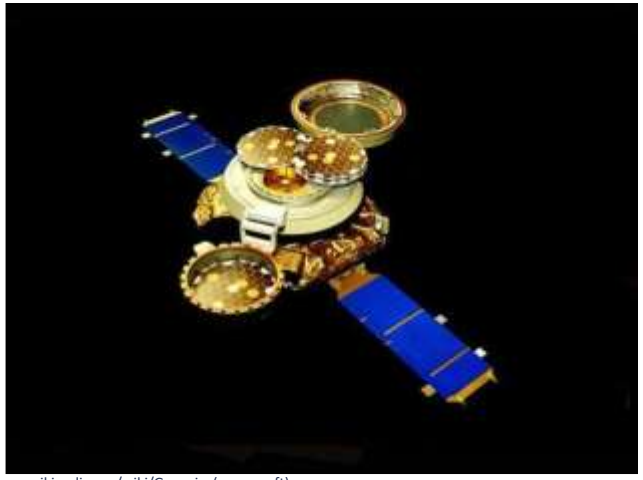
www.constructionchemicals.co.uk/blog/2015/02/28/479/

www.constructionchemicals.co.uk/blog/2015/02/28/479/

www.ebaumsworld.com/pictures/37-construction-fails/86377488/?image=86377506

Example verification issues (not building the system right)

# Verification and Validation



en.wikipedia.org/wiki/Genesis_(spacecraft)

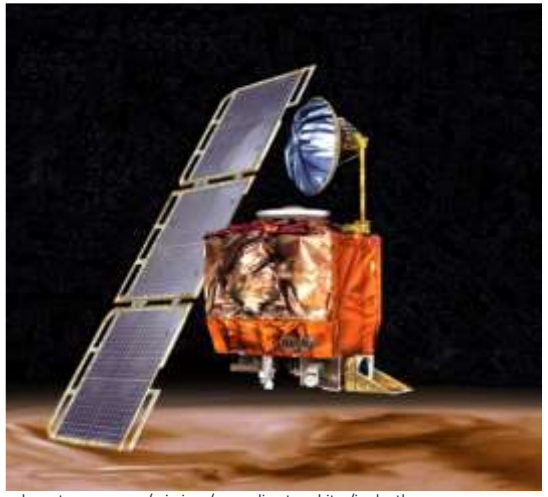
**NASA GENESIS Mission**

- After completing its sampling phase, the spacecraft's return capsule made a hard landing due to parachute deployment failure
- Highly suspected failure can be traced back to an accelerometer installed backwards during assembly
- Some data was retrieved, but not all –the mission cost nearly $300M

www.nasa.gov/mission_pages/hubble/main/index.html

**NASA Hubble Space Telescope Mission**

- Primary mirror shape requirement was correct
- Specialized test apparatus used to verify the requirement was flawed, lens were actually 1 mm askew
- Issue was not realized until after launch, costing $1.5B



solarsystem.nasa.gov/missions/mars-climate-orbiter/in-depth

**Mars Climate Orbiter**

- Software controlling the thrusters failed to convert units from English to metric
- The spacecraft entered Mars' atmosphere instead of orbit and burned up
- Complete loss –the mission cost nearly $200M

Example verification issues (not building the system right)

# Verification and Validation


www.bloomberg.com/news/features/2017-04-19/silicon-valley-s-400-juicer-may-be-feeling-the-squeeze?

**Juicero Juice Press Machine**

- Silicon Valley start-up raised $120M to take advantage of the “healthy diet” market
- Spent years engineering an exquisite machine that pressed bags of pre-made juice, reminiscent of Keurig
- The machine cost consumers $700 (which was still a loss to the company), and required a subscription for the juice bags at ~$50/month
- Turned out the bags could be squeezed by hand.

**Google Glass**

- It wasn’t clear what practical functions or problem it was trying to solve
- It cost $1,500 for consumers
- Product deployment left consumers wondering if it was still a prototype or not in the first two years on the market
- Do you want someone walking into a public bathroom with these on?

www.wired.com/story/google-glass-2-is-here/

Example validation issues (not building the right system)

# Verification and Validation

www.walklondon.com/london-attractions/millennium-footbridge.htm

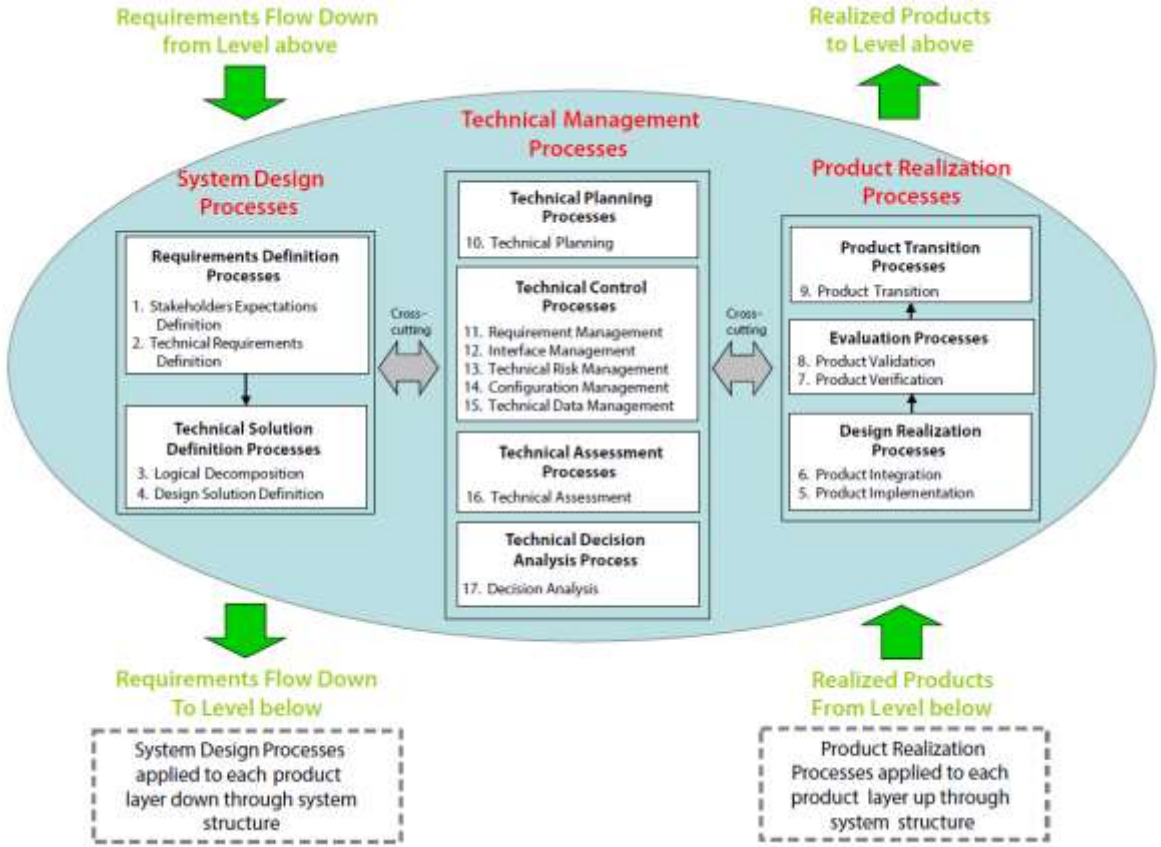

makeagif.com/gif/london-millennium-bridge-OIR23e

**London Millennium Bridge**

- Fancy new London bridge opened in 2000
- During the opening ceremony, over 1000 people started walking across the bridge only to be terrified by its unexpected lateral motion
- The movement was so severe people lost balance
- It was later concluded that lateral deck movement encourages pedestrians to walk in sync, whose combined force resonates with the bridge deck
- The bridge cost $32M initially, modifications cost $9M
- **Unexpected failure mode and missing requirement**

Example validation issues (not building the right system)

# NASA Systems Engineering

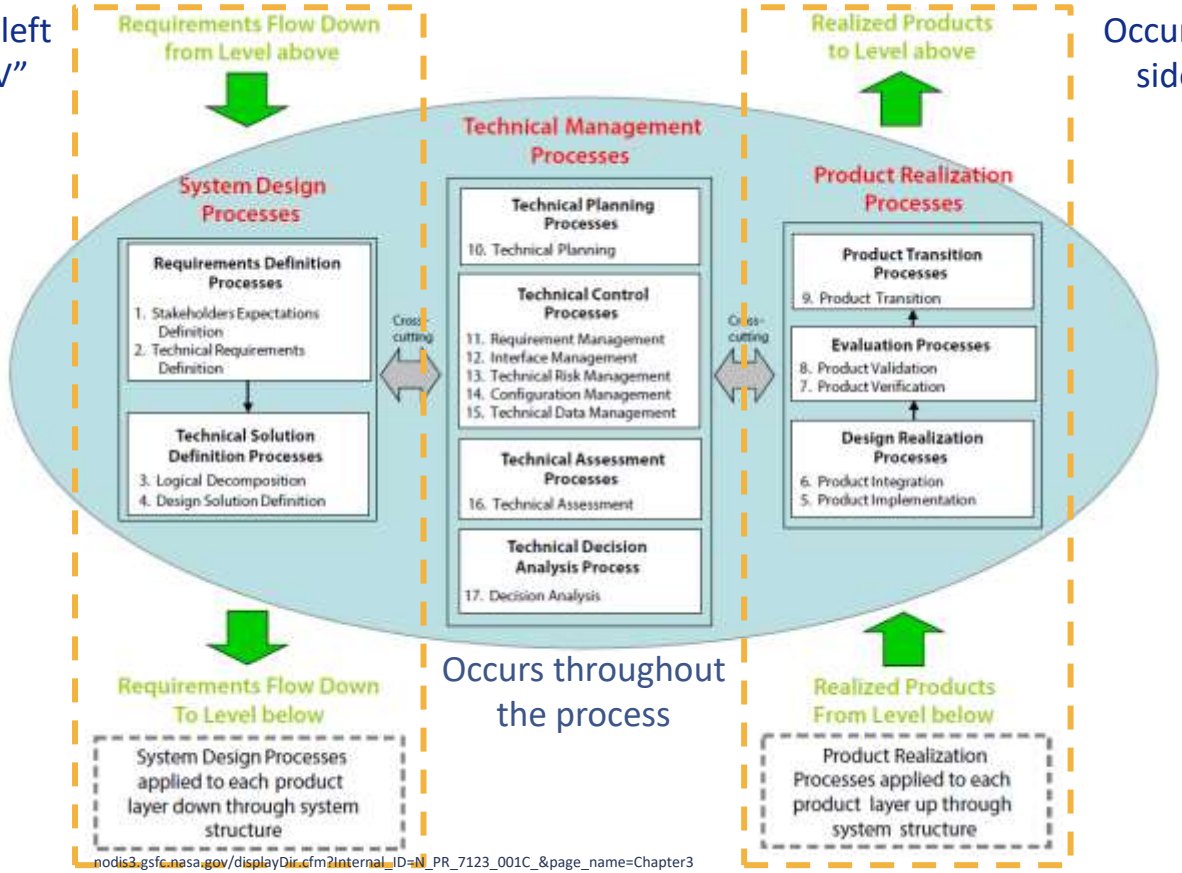



nodis3.gsfc.nasa.gov/displayDir.cfm?Internal_ID=N_PR_7123_001C_&page_name=Chapter3

Helpful guidance, but not necessarily a directive. Makes a lot of assumptions and is intended for much higher levels of mission assurance than UNP

# NASA Systems Engineering

Occurs on the left side of the "V"




Occurs throughout the process

Occurs on the right side of the "V"

nodis3.gsfc.nasa.gov/displayDir.cfm?Internal_ID=N_PR_7123_001C_&page_name=Chapter3

Helpful guidance, but not necessarily a directive. Makes a lot of assumptions and is intended for much higher levels of mission assurance than UNP

# UNP Systems Engineering

**So how is this relevant to UNP?**

# UNP Systems Engineering

AFRL Image

## Education

- Systems engineering training
- Workforce development
- Foundation for all UNP decisions

AFRL Image

## Technology

- Innovative, low-cost technology development
- Motivation for government and industry sponsors
- DoD relevant

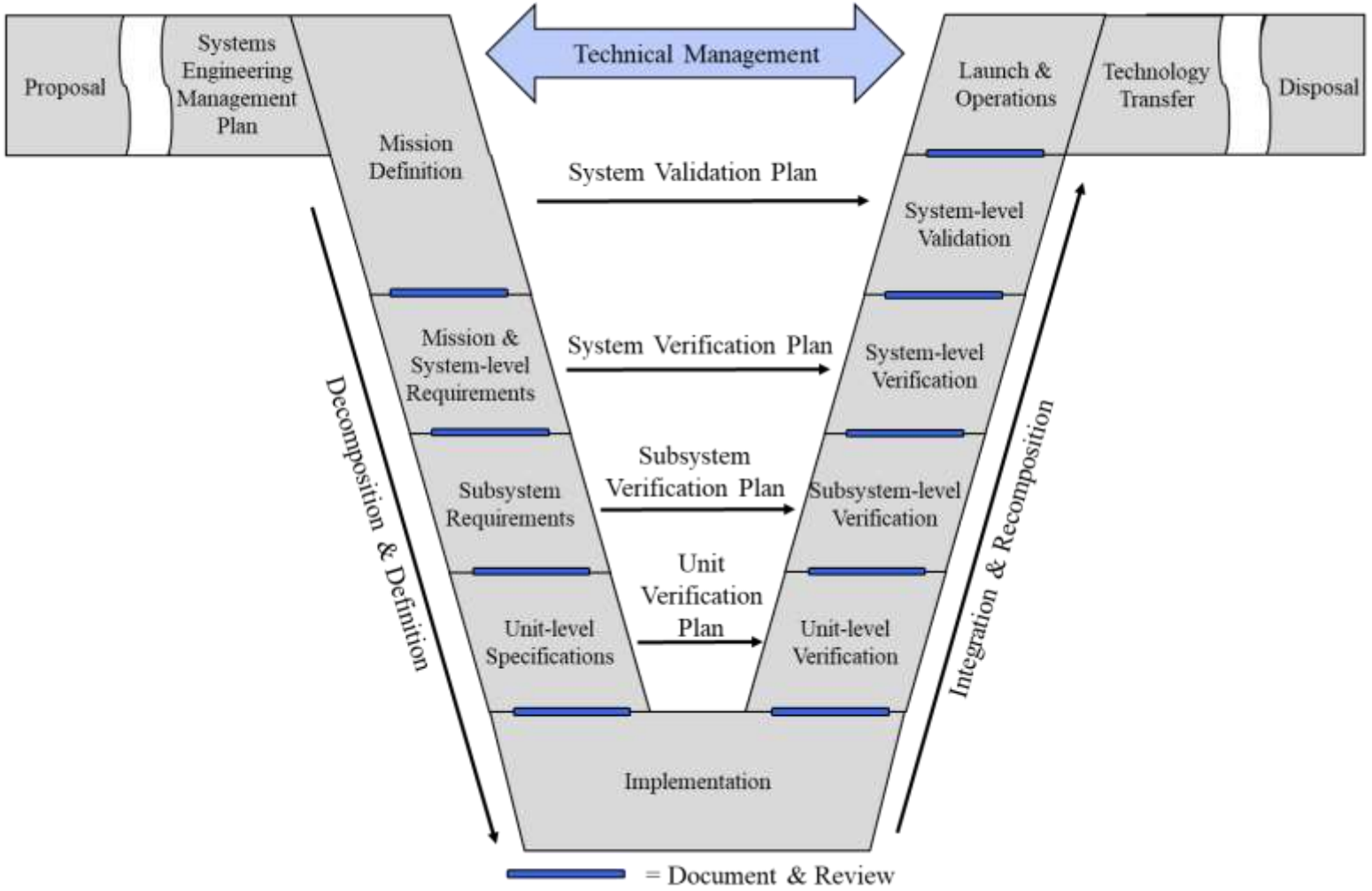
AFRL Image

## University Development

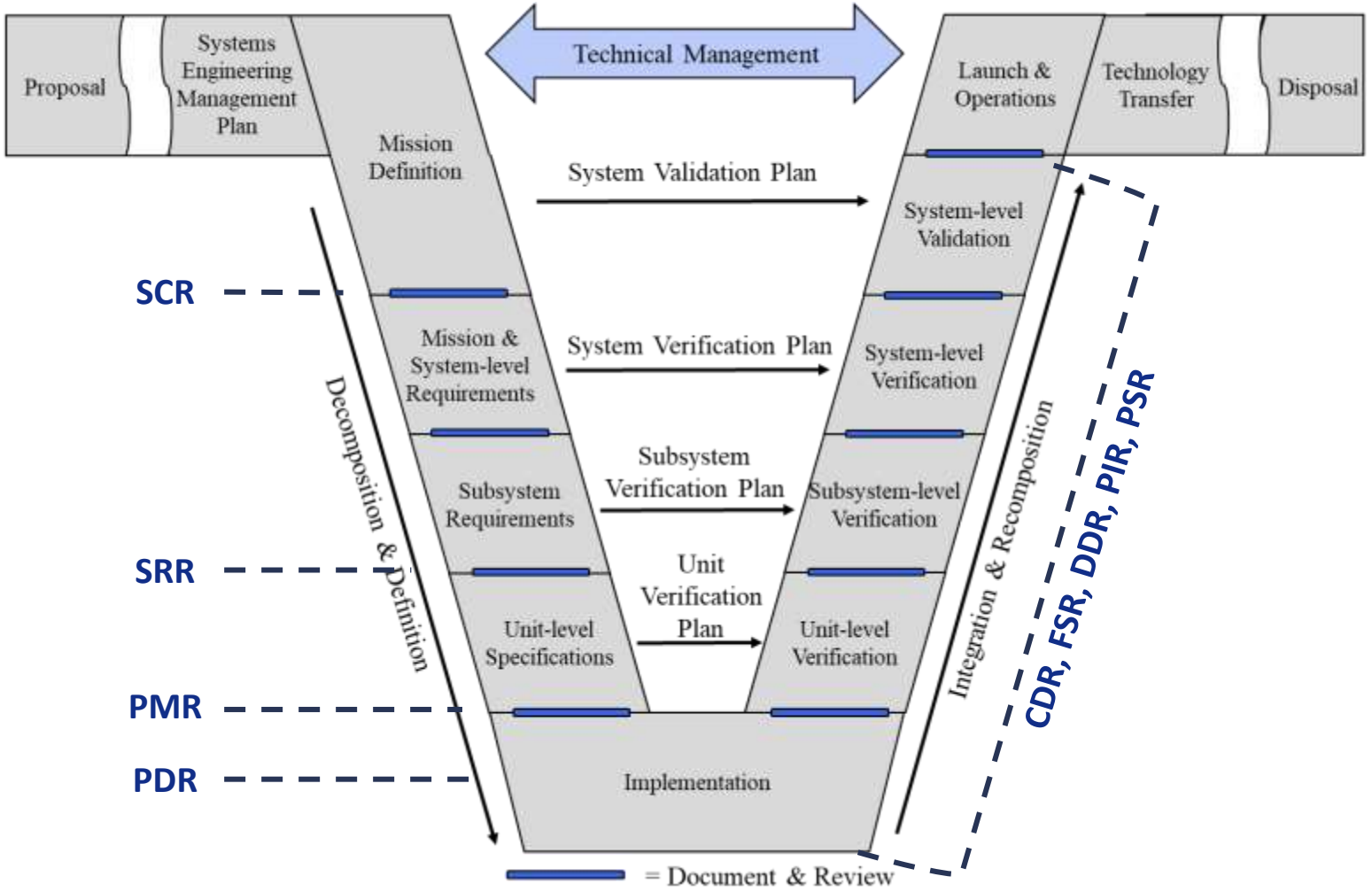
- Develop space hardware laboratories
- Support university PIs

UNP is a systems engineering educational exercise

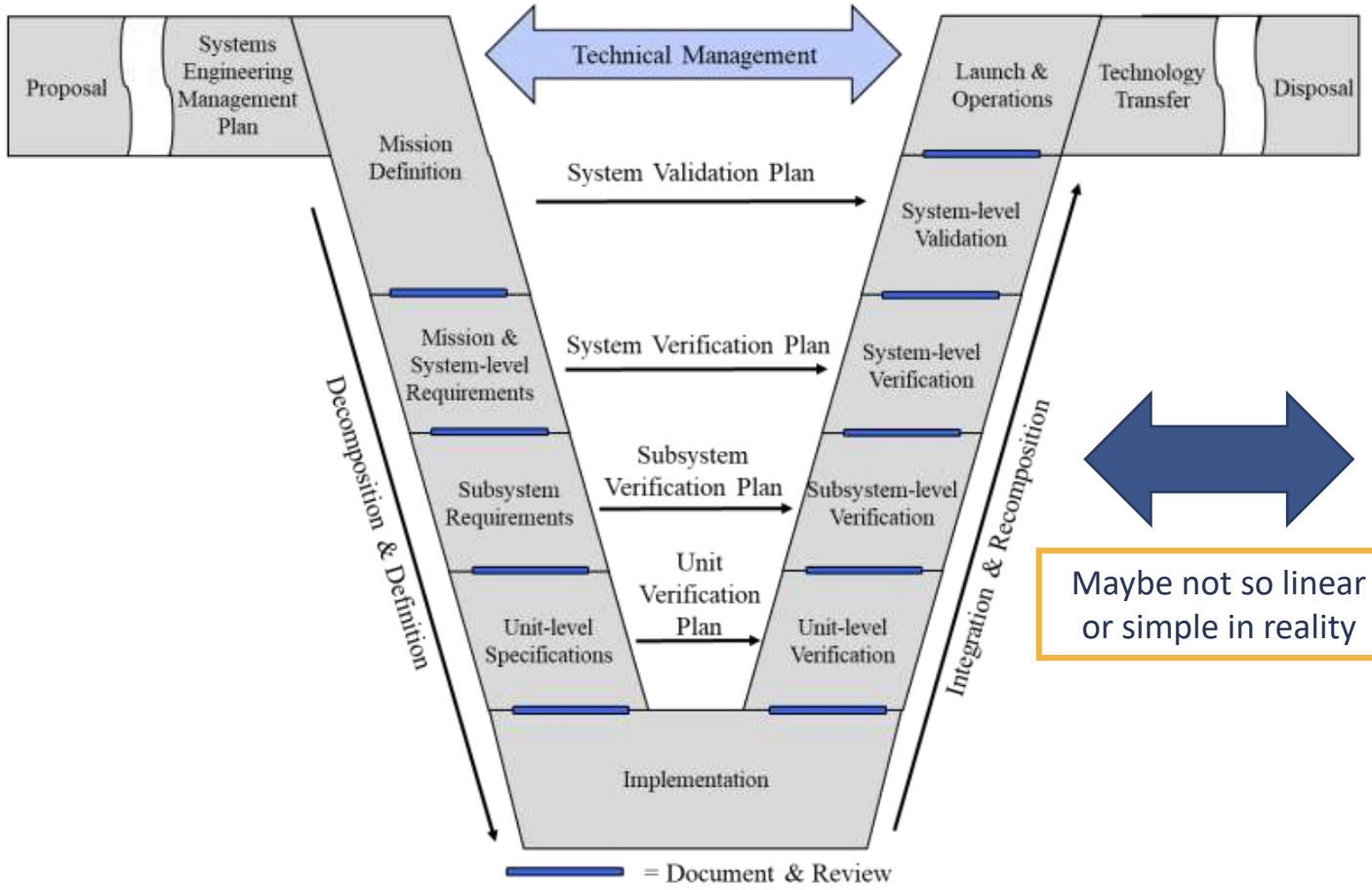
# UNP Systems Engineering



# UNP Systems Engineering

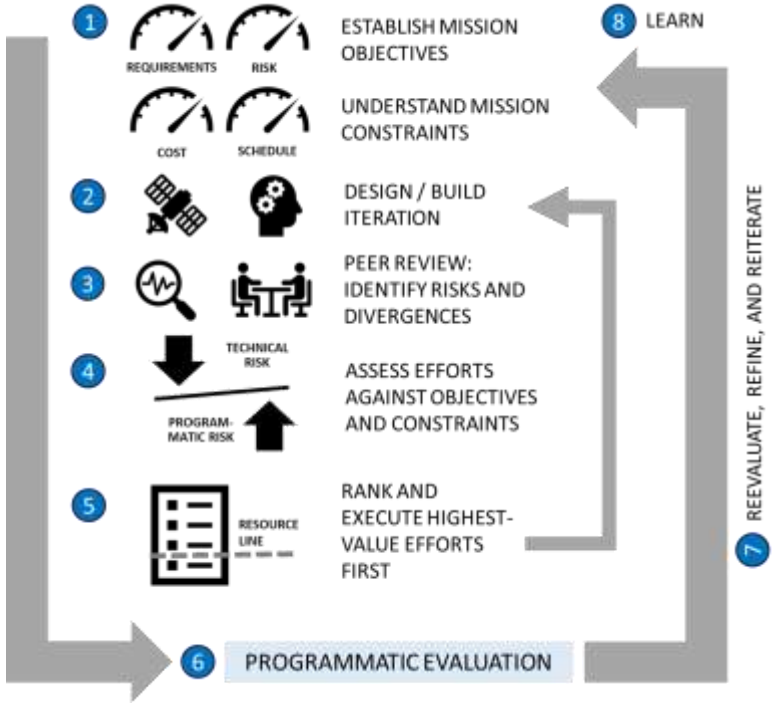


# UNP Systems Engineering





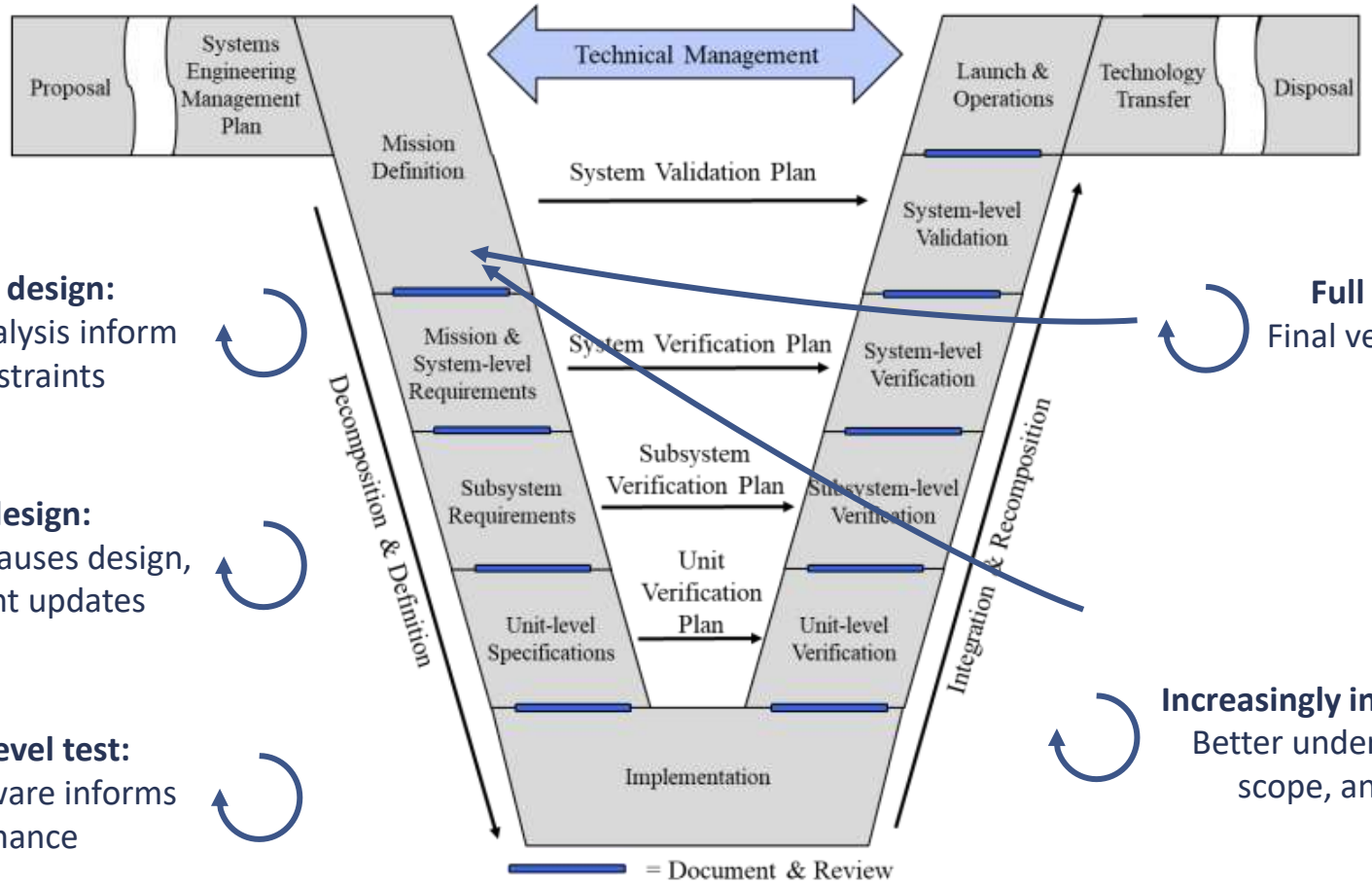

Maybe not so linear or simple in reality




Small Satellite Portfolio Approach

(AFRL Image)

# UNP Systems Engineering




**Concept/mission design:**
Feasibility studies/analysis inform concept and constraints

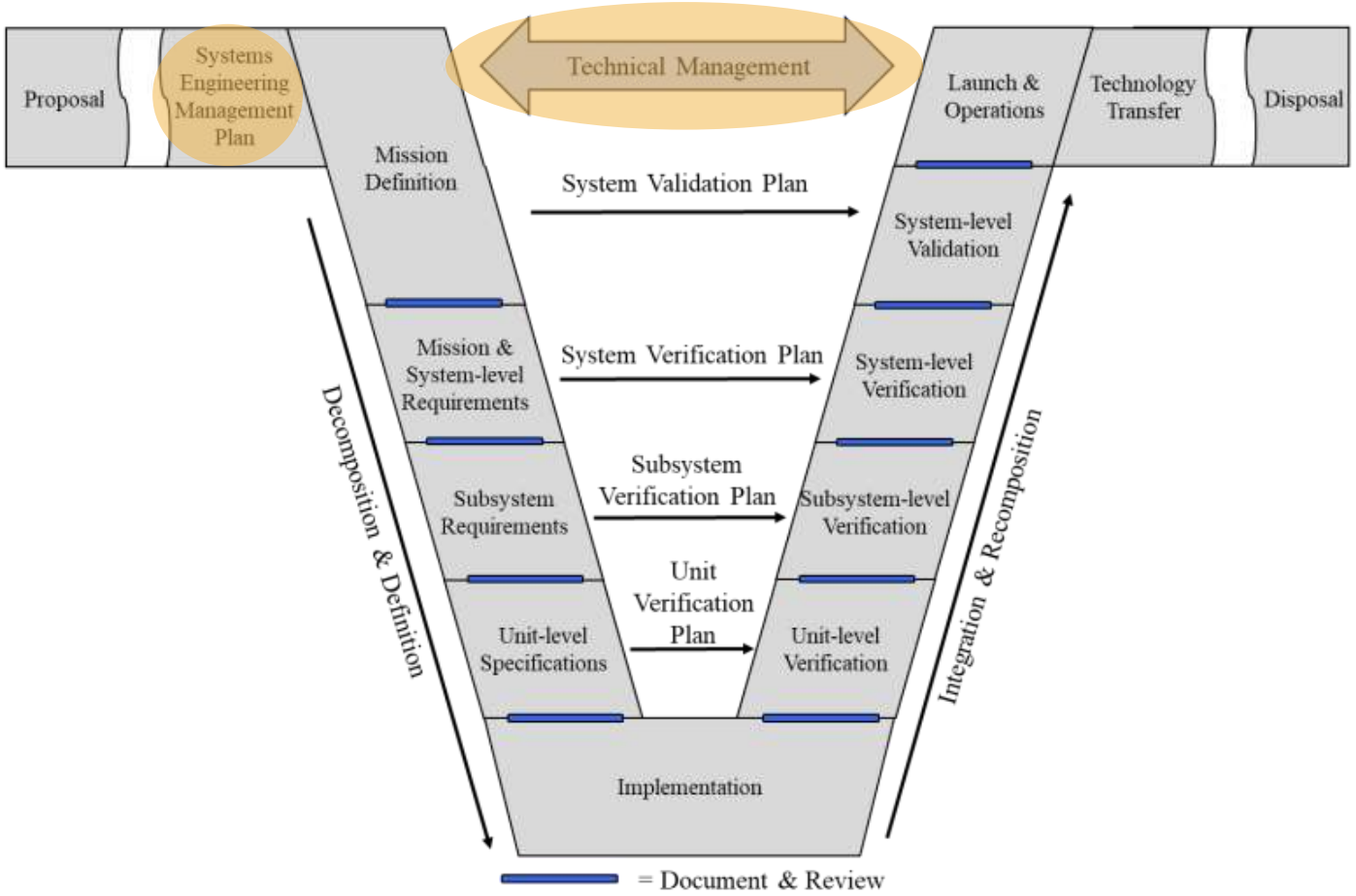
**Definition and design:**
Better understanding causes design, scope, and constraint updates

**Unit/subsystem level test:**
Hands-on with hardware informs actual performance

**Full system testing:**
Final vetting of scope and capability

**Increasingly integrated HW/SW testing:**
Better understanding causes design, scope, and constraint updates

# UNP Systems Engineering



# UNP Systems Engineering

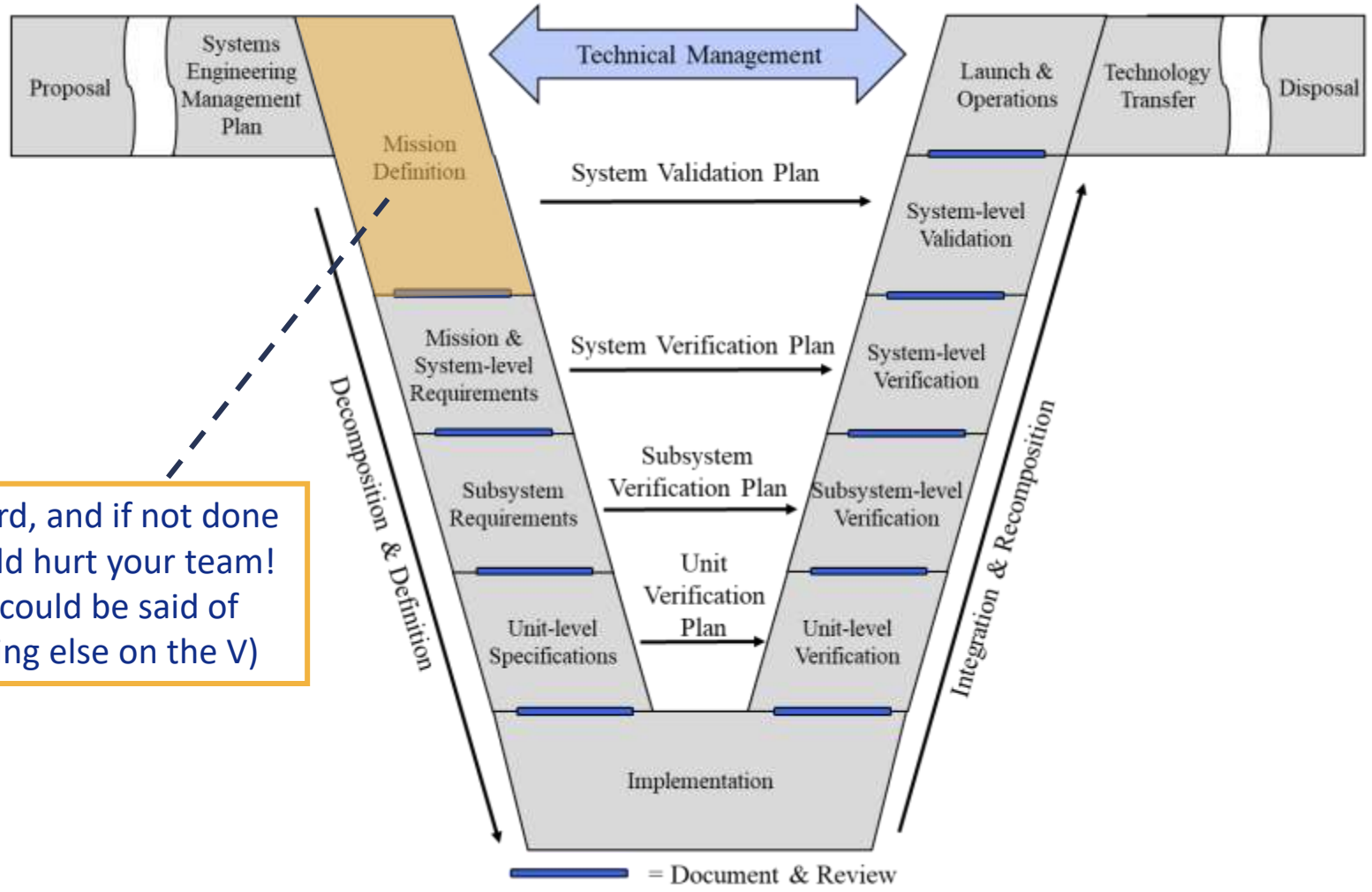
**Where to begin?**
- Some of this material was covered at Kickoff
- The UNP User's Guide has a lot of material discussing critical items teams should be thinking about (i.e. read the User's Guide)

**Some questions each team should ask themselves right now:**
- What is our organizational structure?
- What are the norms/expectations for members our organization?
- What are the roles of each team member (e.g. PI, PM, CE, Team Leads, etc.)? Where is their overlap?
- How will our team operate around university schedules (e.g. breaks, summer, final exam periods, etc.)?
- How do we recruit the talent we need?
- How do we keep team members accountable (e.g. what stops someone from just not showing up?)?
- How do we train new members?
- What is our risk management approach?
- What is our configuration management approach?
- What is our quality assurance approach?
- UNP has a systems engineering process, but what is specifically ours? Do we document it?
- How do we track our personnel, schedule, and cost budgets?
- Should we document our answers to all of the questions above?
- And much more…..

**How do other organizations do this?**
- NASA, ESA, and many other large organizations have SEMPs that are hundreds of pages long –they can be found on Google, check them out
- UNP does not necessarily require a SEMP as a deliverable, so do not take this as a directive to make a massive SEMP document
- I encourage each team to reflect on what is best for you (some UNP teams are very large, some are very small, some are like a club, some are a formal class)

At the beginning of a project, it is critical to establish a systems engineering/project management plan

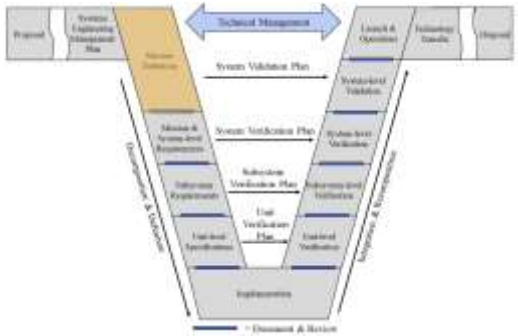
# UNP Systems Engineering




This is hard, and if not done right could hurt your team! (same could be said of everything else on the V)

# UNP Systems Engineering

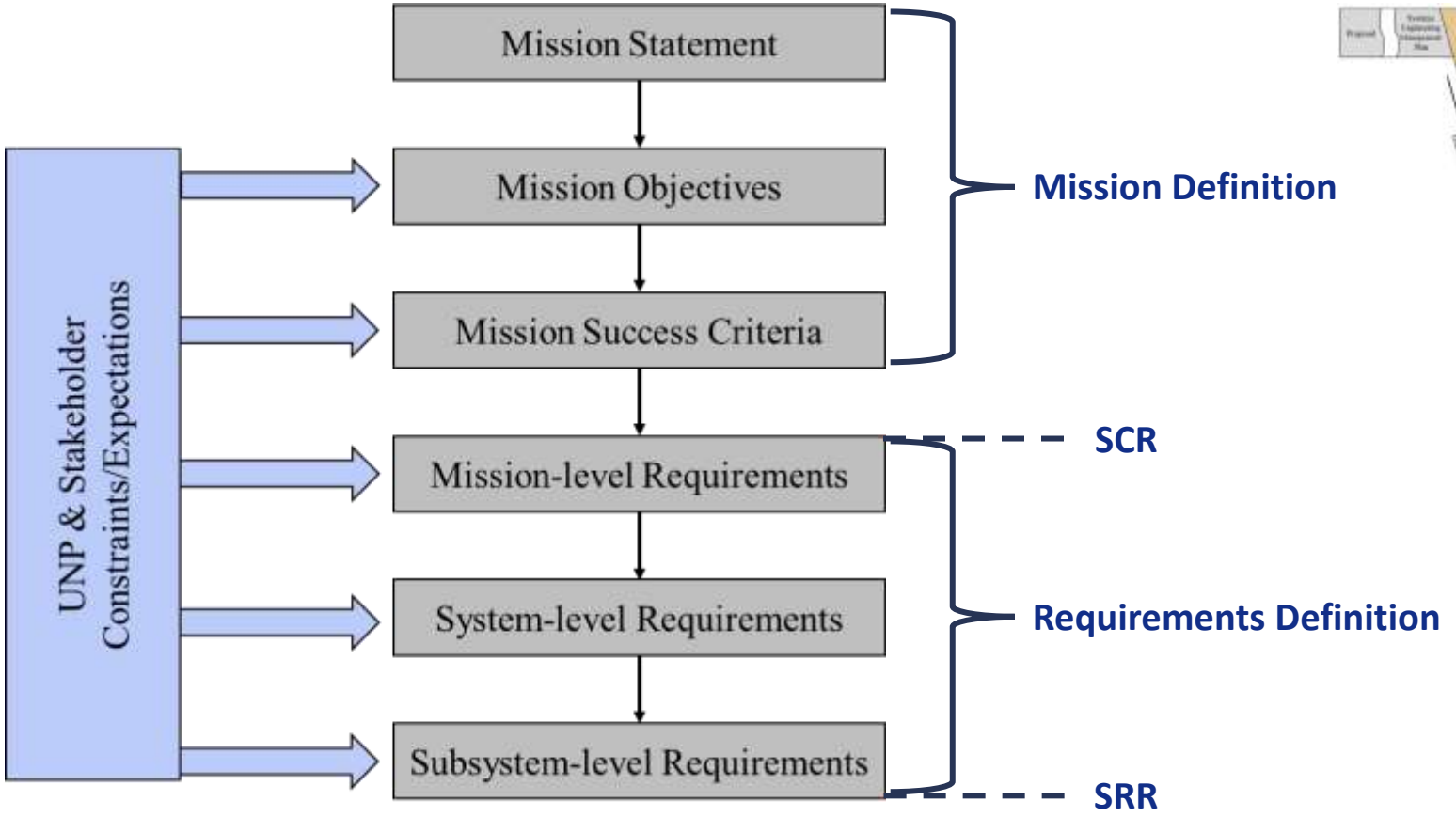
- Mission Design Document
  - Mission Overview
  - Experiment Plan
  - CONOPS



**This deliverable is discussed in the UNP User's Guide**

The Mission Design Document is the primary product of the Mission Definition Process

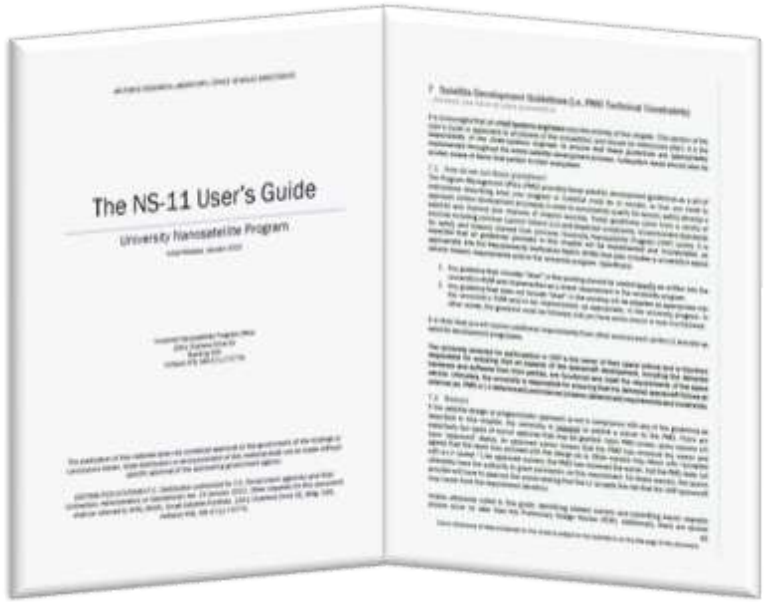
# UNP Systems Engineering



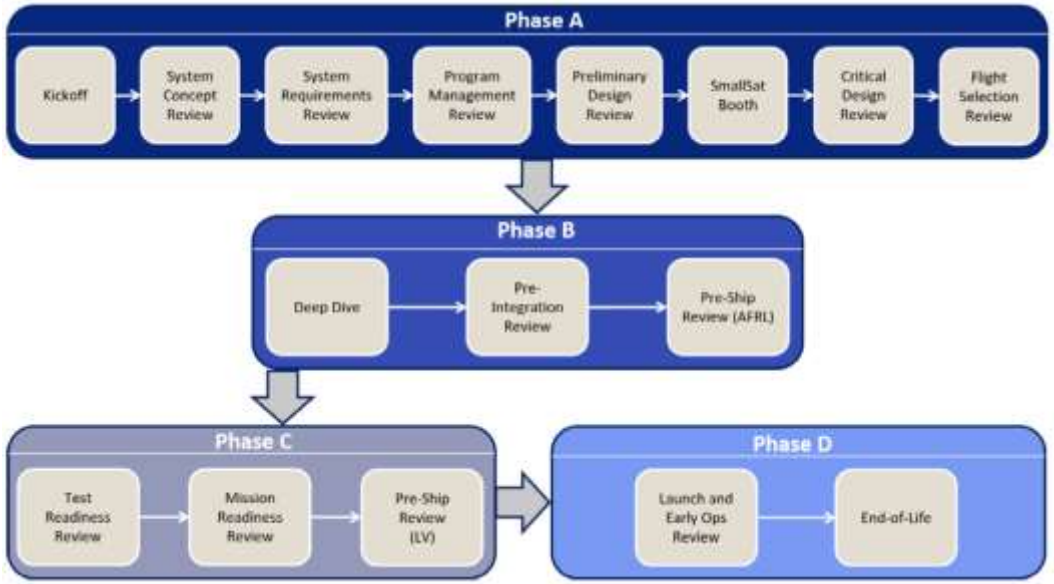

In a very broad sense, this is the decomposition flow down UNP teams should aim for

**it is okay to have some mission/system-level requirements defined before SCR*

# UNP Systems Engineering

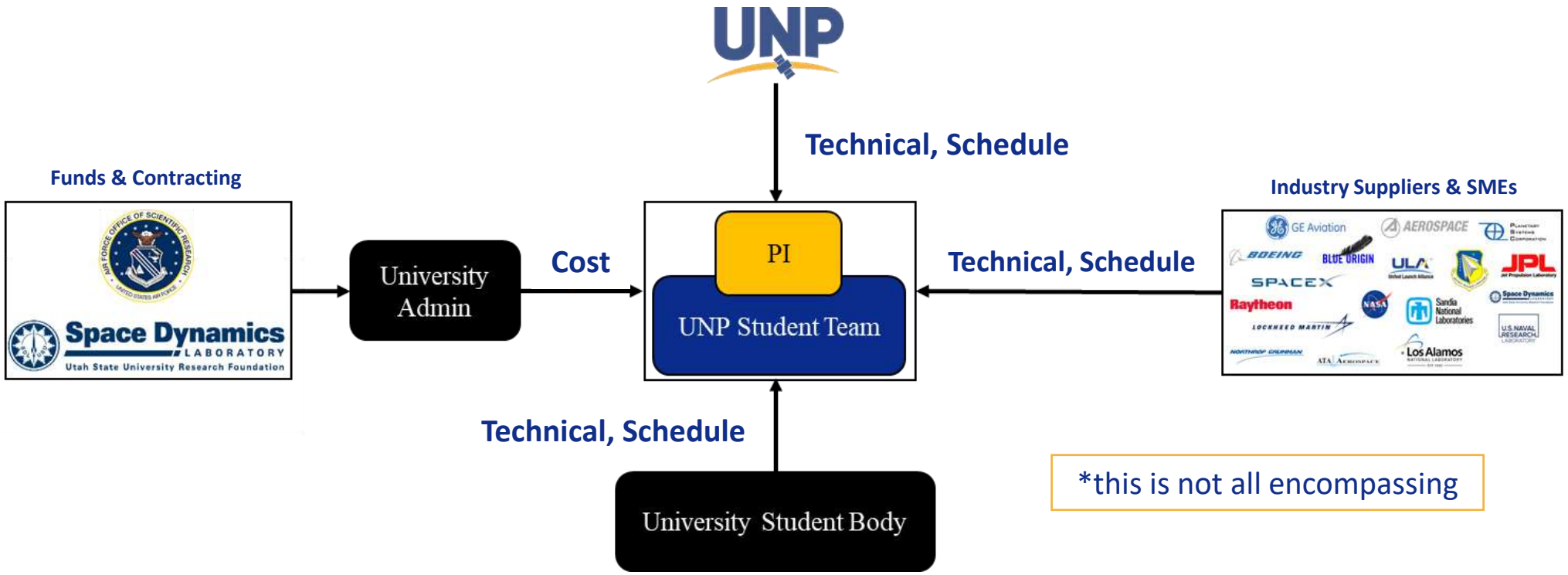



$




UNP imposes numerous technical, cost, and schedule constraints on your team

# UNP Systems Engineering



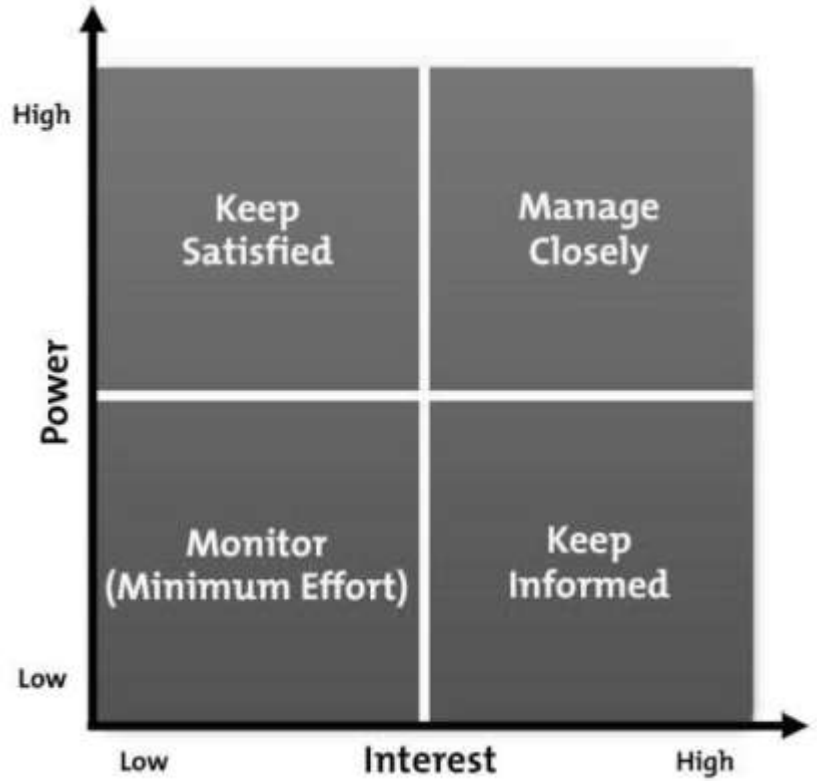

*this is not all encompassing

There are constraints from other sources as well

# UNP Systems Engineering

**Stakeholder**

- IEEE “A stakeholder is any person or organization who will be impacted by the system at any phase in the life cycle”
- UNP UG: “Could already be identified or be parties seen to benefit from a mission’s data. Merely stating AFRL as a customer does not adequately capture who the customer is. A name, office, or a program would be examples of a particular customer at AFRL. The UNP PMO does not qualify as a customer of a program’s data.”
- A stakeholder can help, damage, make, or kill a system




www.projectmanagement.com/wikis/368897/Stakeholder-Analysis--using-the-Power-Interest-Grid

Stakeholders can be very helpful for determining why your system matters

# UNP Systems Engineering

**Constraint**

- NASA: “A condition that is to be met. Constraints, in conjunction with the CONOPS, help identify how the system should be operated to achieve a mission’s objectives
- Not to be confused with risk! (Risk Management is the subject of a future EAT)

**Mission Statement**

- UNP: “A multiple-sentence statement of the entire mission’s purpose, usually focusing on the scientific or technological goals of the mission.”
- Can be viewed as the mission’s elevator pitch

**Mission Objective**

- SMAD: “Broad statements of what the system must do to be useful”

**Mission Success Criteria**

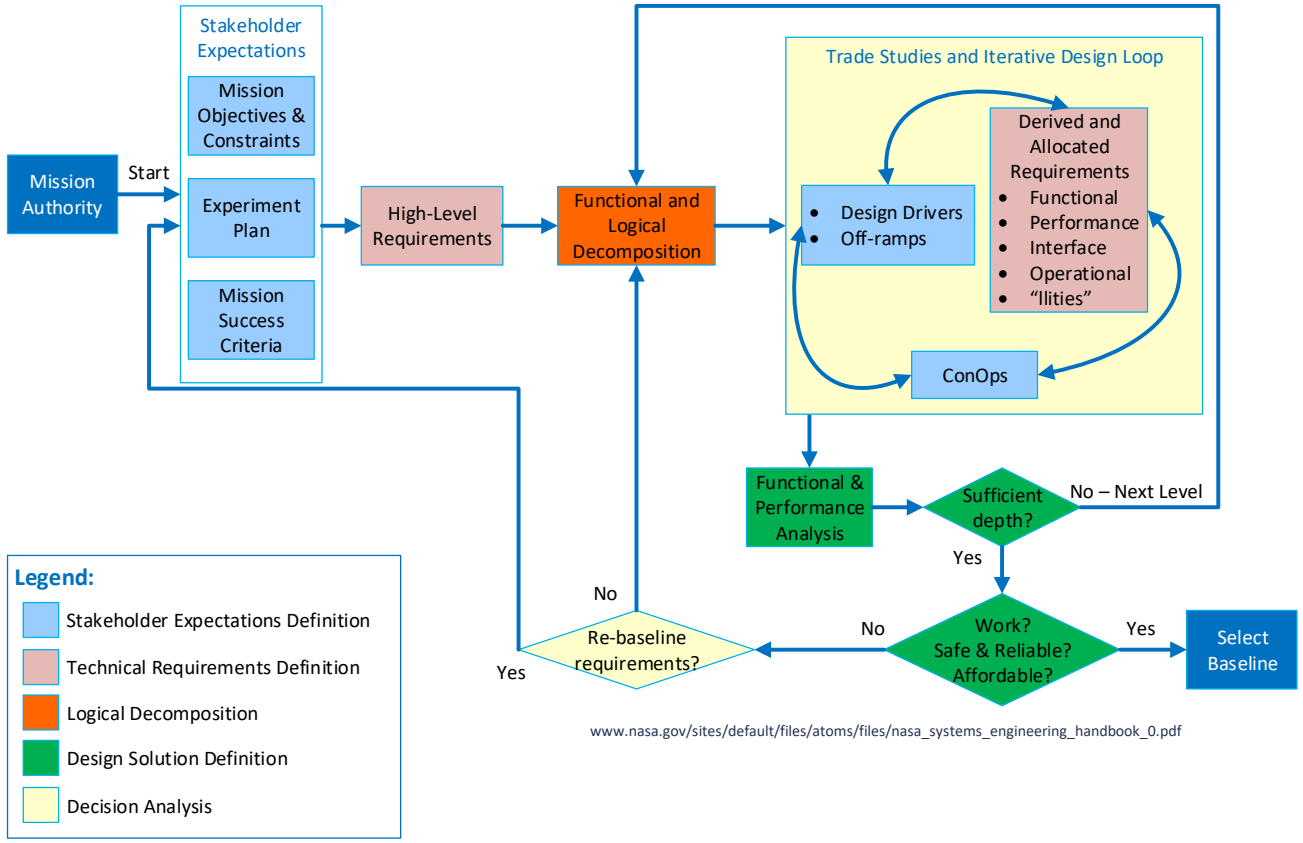
- While Mission Objectives state “what the system must do,” Mission Success Criteria state “how well the system must do it”
- Your mission’s pass/fail criteria. All minimum success criteria must be met to “pass” your mission

**Concept of Operations (CONOPS)**

- See the UNP User Guide for detailed description
- IEEE: “A document describing the characteristics of a proposed system from the viewpoint of an individual who will use the system. It shall communicate quantitative and qualitative system characteristics to all stakeholders.”
- Combined with Mission Objectives and Success Criteria, it is your “contract” with primary stakeholders

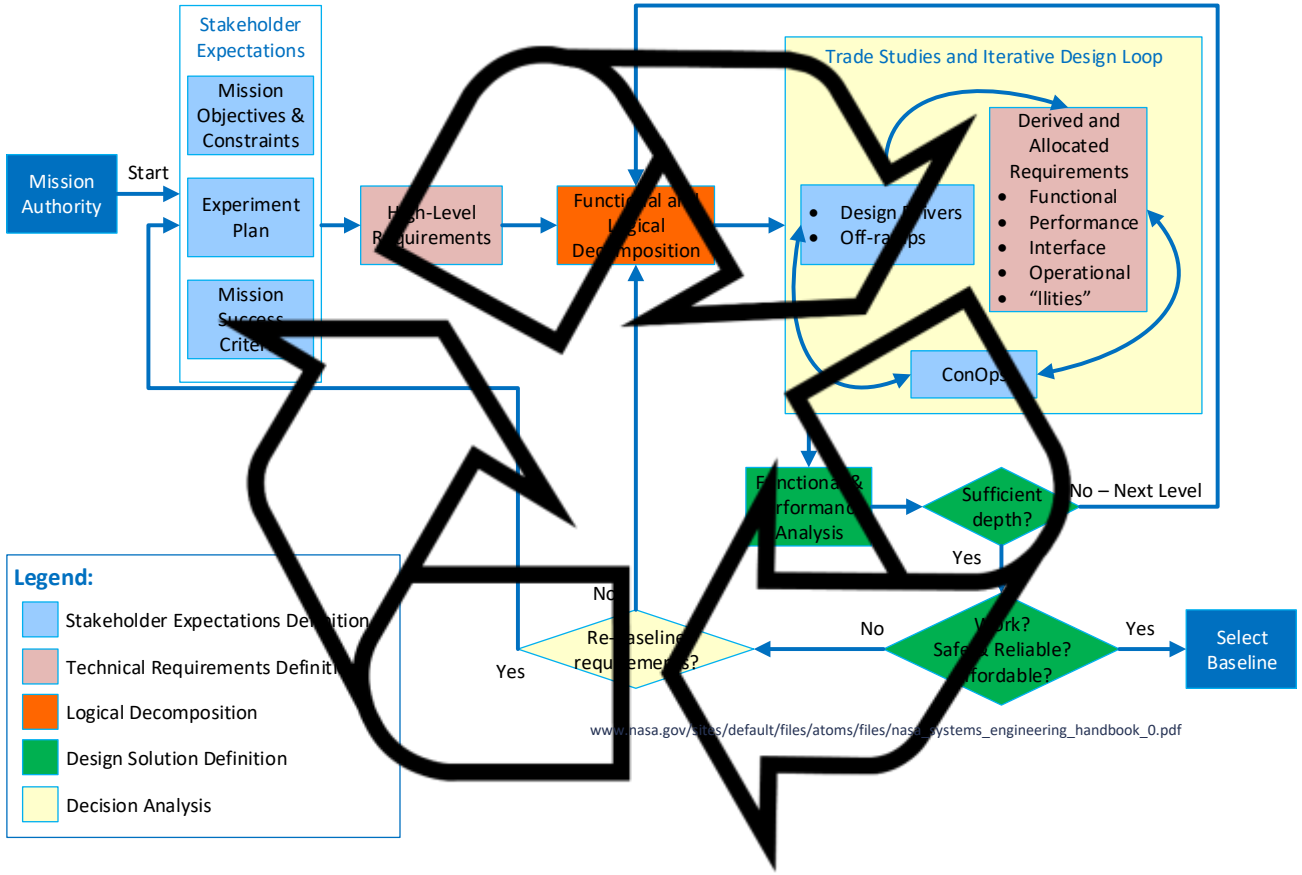
Some key definitions

# UNP Systems Engineering



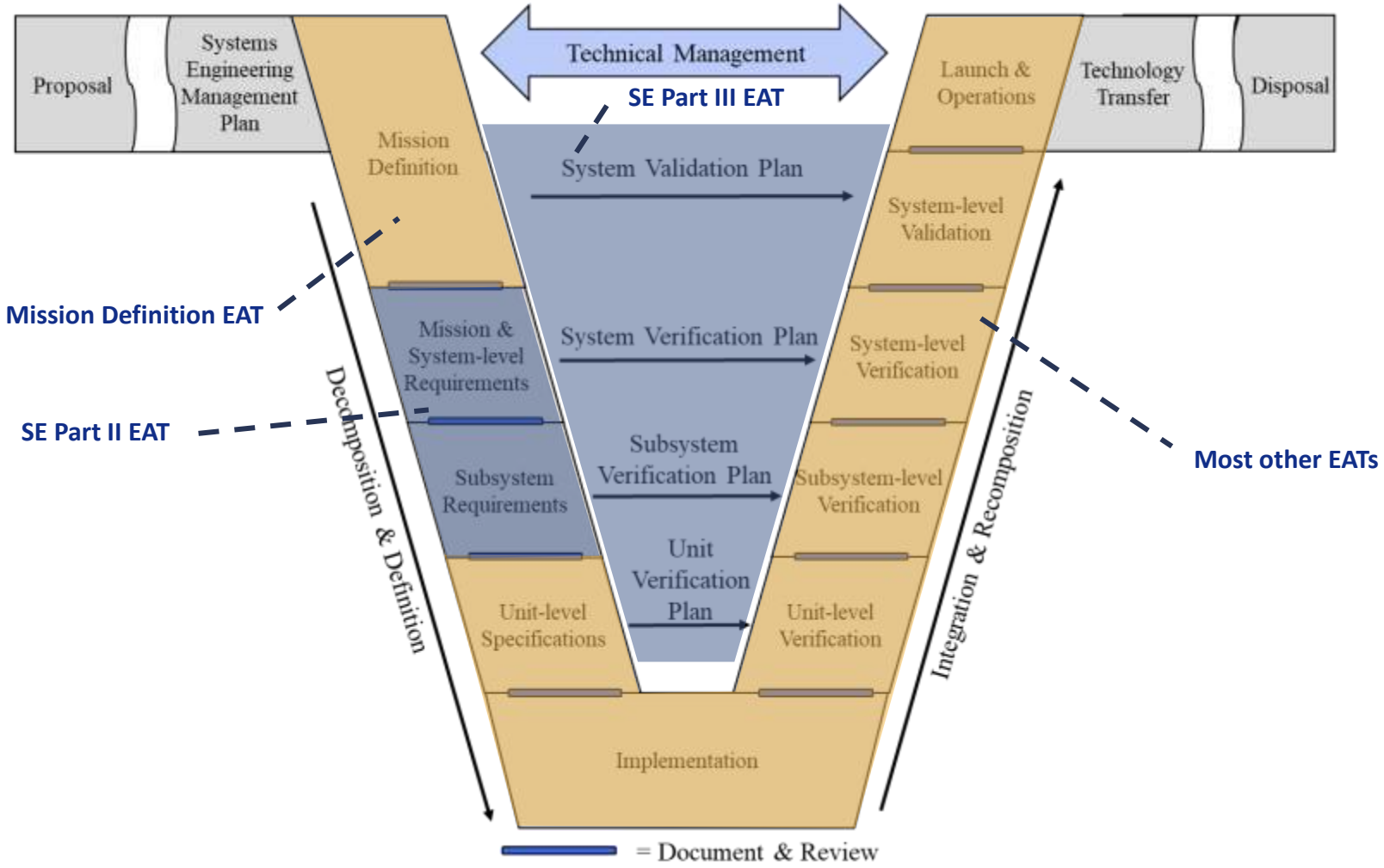

Here is a more detailed overview of the "Mission Definition Process"

# UNP Systems Engineering

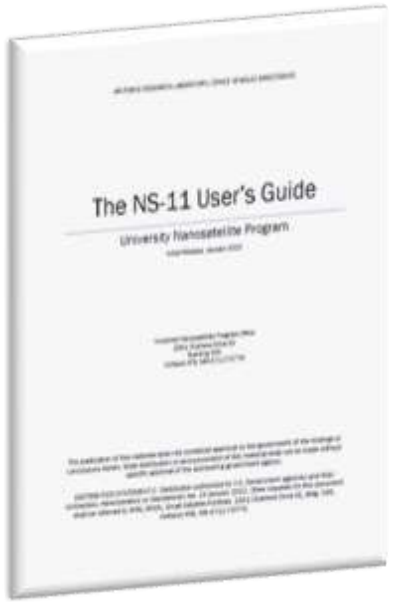



Successful teams iterate through this cycle many times before SCR/SRR

# UNP Systems Engineering



# Want to Learn More?




**Read the UNP User's Guide!**

**Stick around for future EATs!**

**Ask questions!**

**Read the supporting material**

**Supporting material**

- Most of this presentation copied from "UNP NS10 Systems Engineering Part1 EAT" by Sam Baxendale
- "UNP NS11 User's Guide", AFRL/RV, 2022
- "Applied Space Systems Engineering", Larson/Kirkpatrick/Sellers/Thomas/Verma, 2009
- "Space Mission Engineering", Wertz/Everett/Puschell, 2011
- "The NASA Systems Engineering Handbook", NASA, 2007
- "INCOSE Systems Engineering Handbook", INCOSE, 2010
- "Michigan Tech MEPIV Lecture: An Introduction to Systems Engineering", King, 2019
- "UNP NS9 EAT: Systems Engineering", Straight, 2016
- "ISO/IEC 15288 IEEE Systems Engineering Standard", IEEE, 2015
- "ECSS-E-10A European Systems Engineering Standard", ESA, 2018
- And more! *Not an exhaustive list*