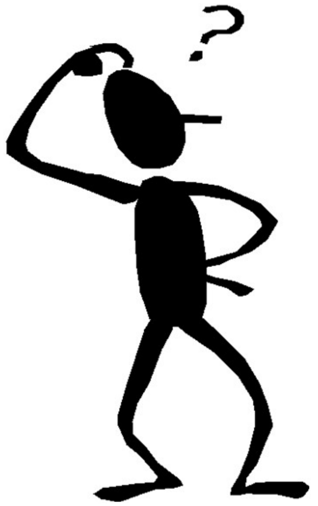
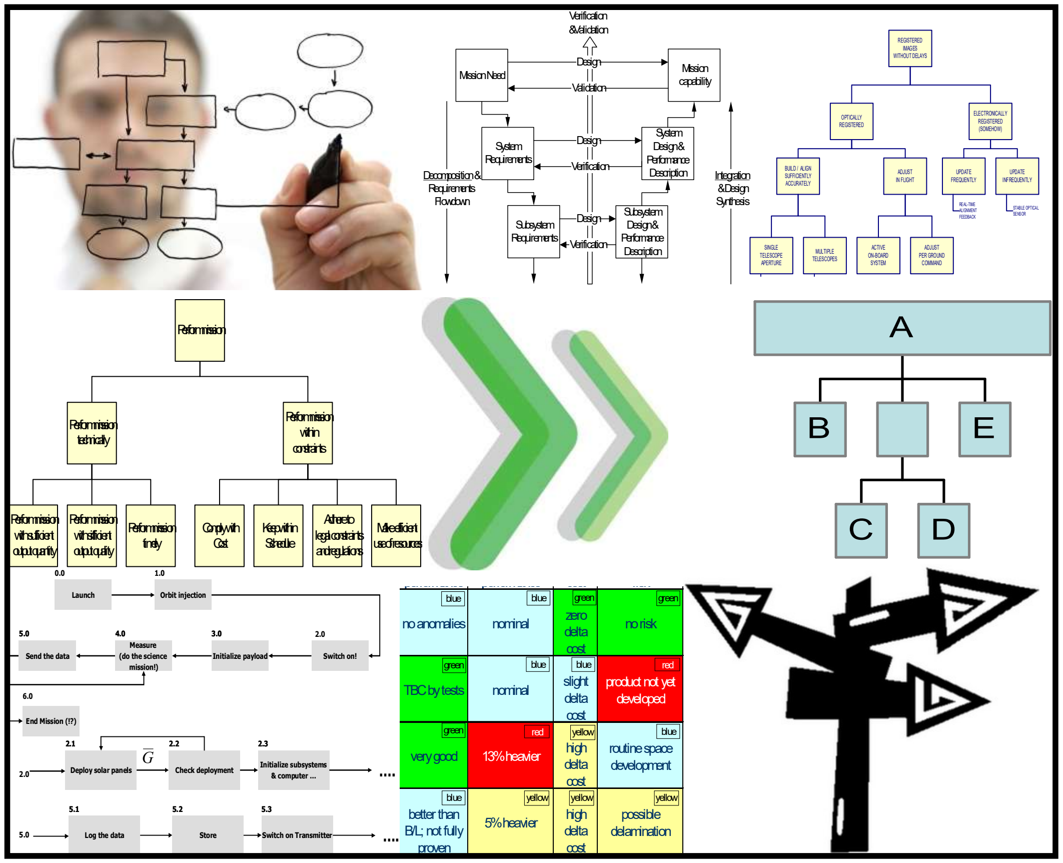
Systems Engineering and Aerospace Design (AE3211-I)

Systems Engineering Methods

Prof. Dr. Eberhard Gill, Space Systems Engineering



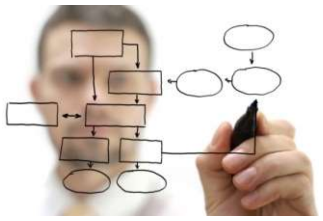


Today you'll learn to …

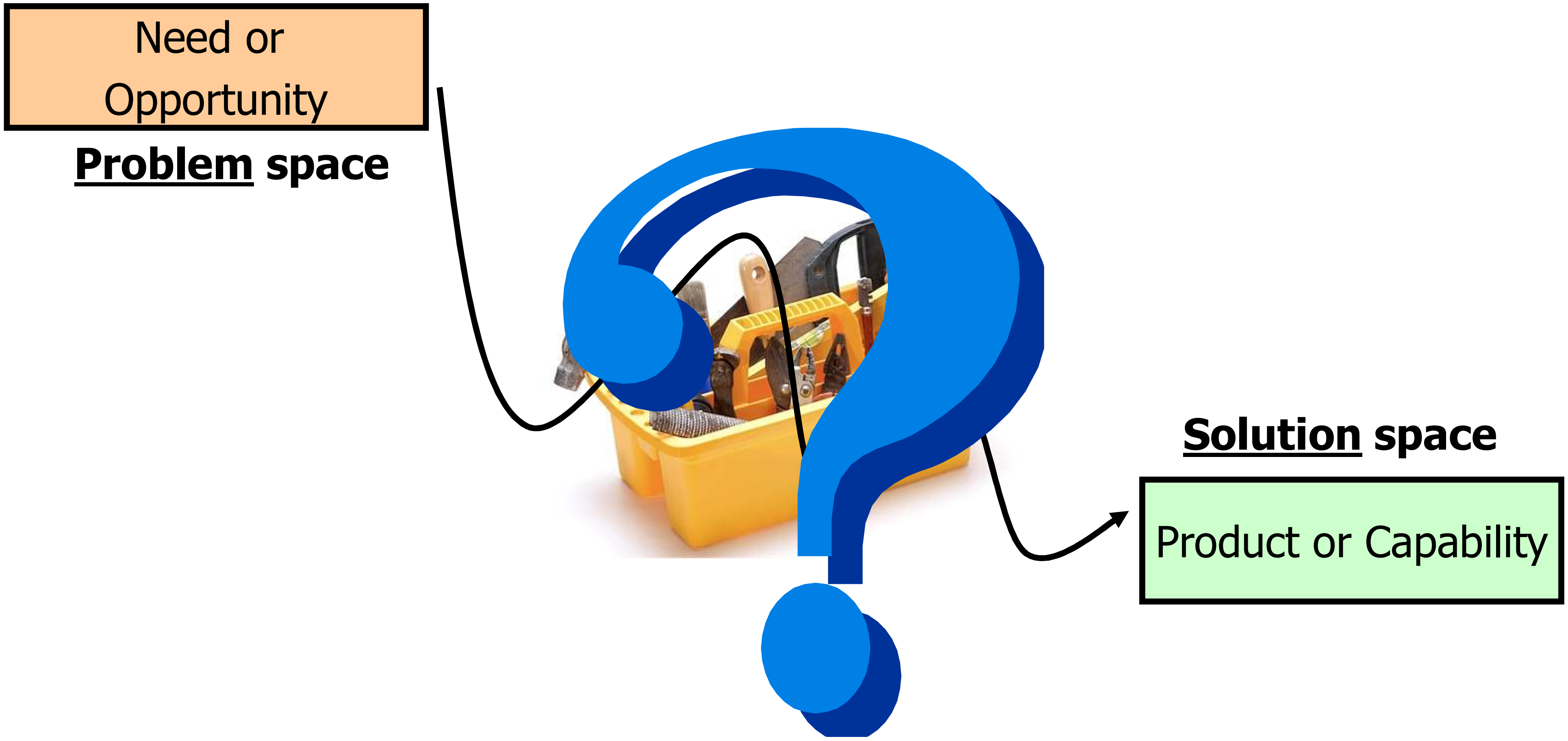
1. **discover** systems engineering elements introduced in previous courses and **integrate** them into the systems engineering framework

2. **apply** systems engineering methods you will need in the Design Synthesis Exercise.

Contents



How to design a complex product?

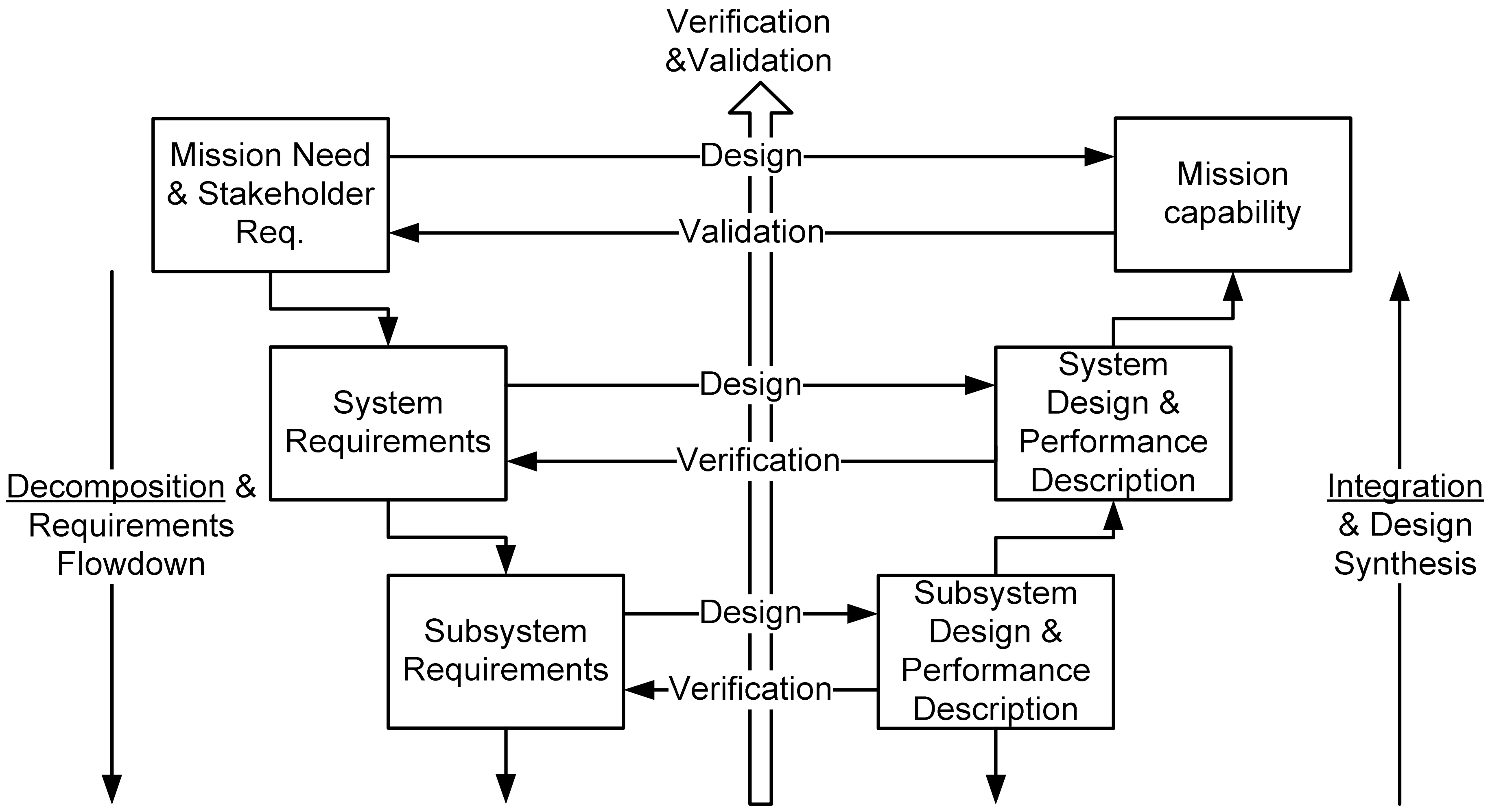


What you will need…

- Process
  - The action of **taking something through a set of steps** to convert it from one form to another.
  - Examples: *Systems Engineering process (V diagram), Verification process*
- Method
  - A method is a **specific approach to treat a problem**.
  - A method may be applied in a certain process using a certain tool.
  - Examples: *Analysis, Requirement discovery, configuration management*
- Model
  - Something which **describes and represents a characteristic** behavior or system.
  - Examples: *Software orbit model, satellite flight model*

Adequate processes, methods, and models are needed to efficiently go from the problem to its solution.

The Overall Systems Engineering Process



The *V model* emphasizes the dual components of <u>decomposition</u> and <u>integration</u> in the product development process.

Hints and Pitfalls

The methods discussed in this lecture contain **valuable experience** from legions of engineers.

General hints

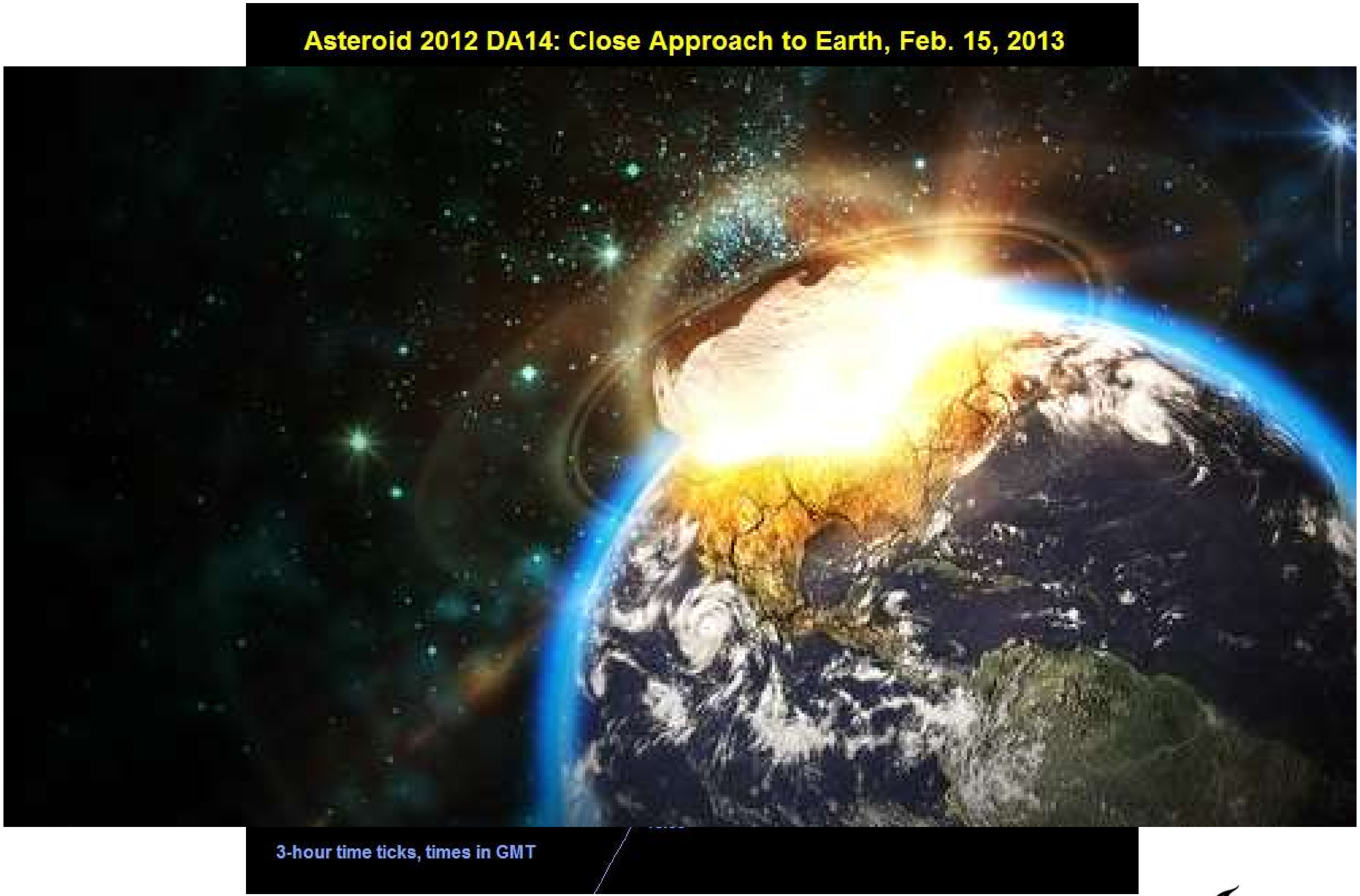
- Keep problem (what?) and solution (how?) apart!
- Think systems!

Some pitfalls

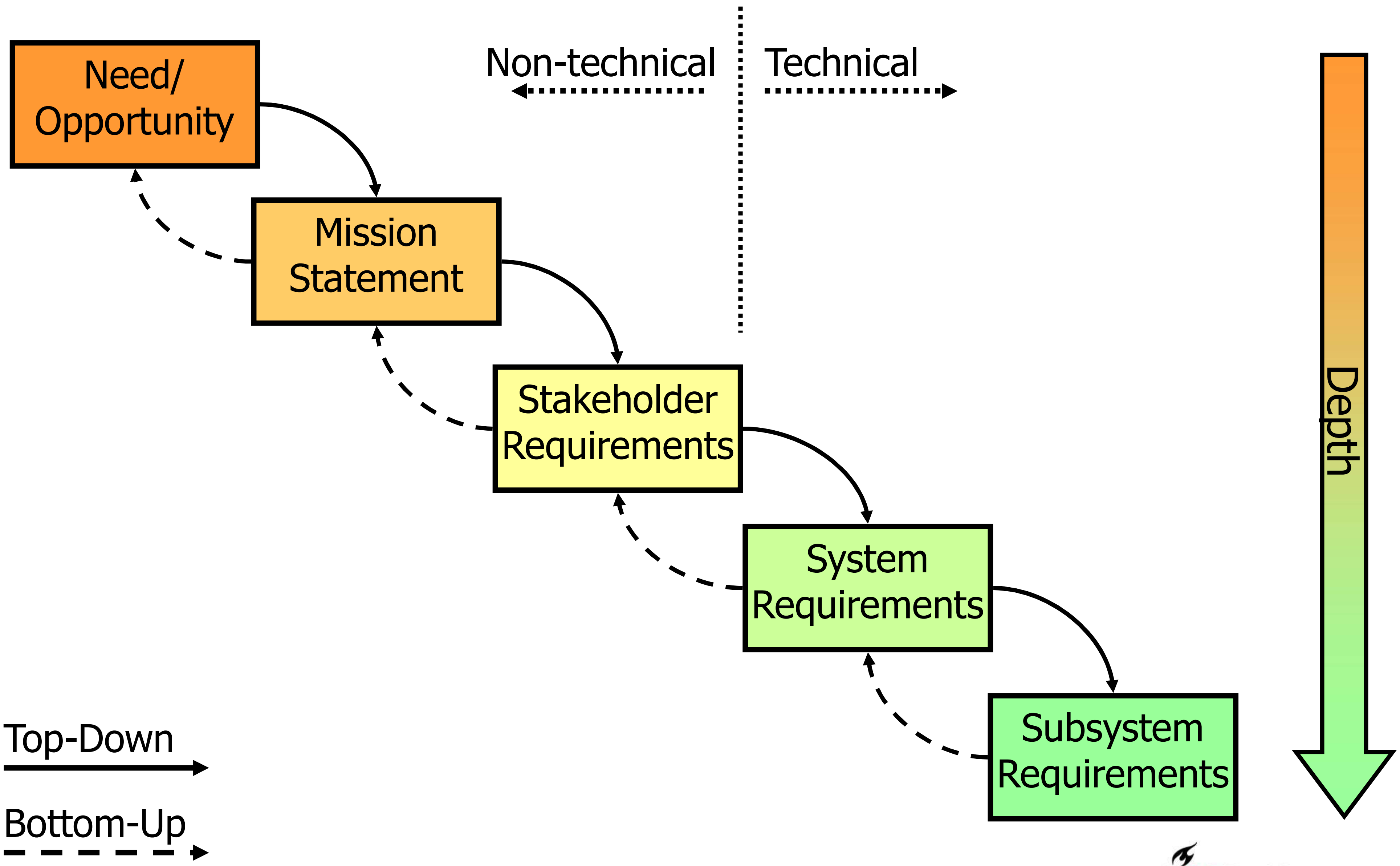
- ~~Using methods as cookbook without context and concept~~
- ~~Using methods for their own sake~~

In the end, you are assessed for your **product**, not for your process!

A Discovery Tour: From Needs to Technical Requirements


Asteroid 2012 DA14: Close Approach to Earth, Feb. 15, 2013
3-hour time ticks, times in GMT

A Discovery Tour: From Needs to Technical Requirements



A Discovery Tour: From Needs to Technical Requirements – Some good examples

Need statement

- *Mankind must be protected from extinction by the threat of celestial bodies impacting Earth.*

Mission statement Save-The-World (STW)

- *The United Nations will protect mankind from being extinguished by impacts from celestial bodies from 2030 onwards.*

Stakeholder requirement (for space agencies)

- ***STW-Ast-04** Impacts of celestial bodies on Earth, causing potential extinction of mankind, shall be avoided by deflecting those bodies from collision with the Earth with a minimum distance from the Earth's surface of 36000 km $\pm$ 5000 km (1σ).*

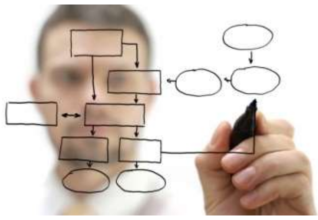
System requirement

- ***STW-Sys-A04-5** A space system shall provide a total velocity increment to the celestial body before reaching one lunar distance of the Earth.*

Subsystem requirement

- ***STW-Sys-A04-5-Deflect-Perf.1.2** The space system shall be equipped with a nuclear explosive of minimum of <tbd> Mt.*

Mission Need Statements



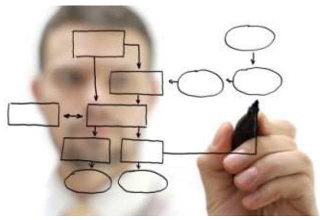
- What is it?
  - It describes, in general terms, need and mission in single, clear and concise sentences.
  - It describes what the product does, NOT what it is or how it is done.
- Why do we need this?
  - It gives you and your customer a common view on the task.
  - If your product satisfies the statement, the customer is happy.

Example of a mission statement:

Google™

Organize the world's information and make it universally accessible and useful

Requirements



Definition*

A condition which must be complied with

Purpose

- Defines characteristics, capabilities and constraints of your system
- Forms the basis for design, manufacture, test, and operations

Types

1. **Stakeholder requirement**: The need of each stakeholder must be captured in at least one stakeholder requirement. This must be formulated in the language of the stakeholder.
2. **System requirement**: It describes the system in general technical terms at a high level.

Pay special attention to those:

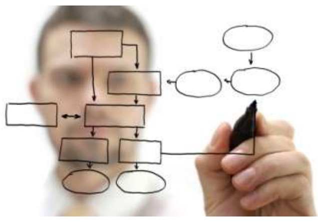
- **Killer requirement**: A requirement which drives the design to an unacceptable extent.
- **Driving requirement**: A requirement which drives the design more than average.
- **Key requirement**: A requirement which is of primary importance for the customer.

* ESA standard ECSS-P-001

Requirements on requirements

1. **Each requirement shall have a unique identifier.**
2. This identifier shall not be re-used in the event the requirement is deleted.
3. A requirement shall be unambiguous, use straight forward and simple words and phrases.
4. The inclusion of essential, design-to, requirements shall be complete.
5. **Each sentence shall include only one requirement.**
6. Terms vague or not verifiable as "**or, etc., goal, relevant, necessary, appropriate, shall include, but not be limited to**" shall not be used.
7. A requirement shall be unique (e.g. shall be included only once in a specification).
8. The content of a requirement shall not be interpretable in more than one way.
9. **A requirement shall be verifiable.**
10. A requirement shall be consistent, i.e. it shall be ensured that there is no conflict between any two requirements in a specification.
11. **The content of a requirement shall not include the reason for the requirement or a description of the design.** Rationales and design descriptions shall be documented separately.
12. The requirement shall be complete, i.e. it shall be ensured that all functions and performance items are defined, which are required during the foreseen life cycle of the system.

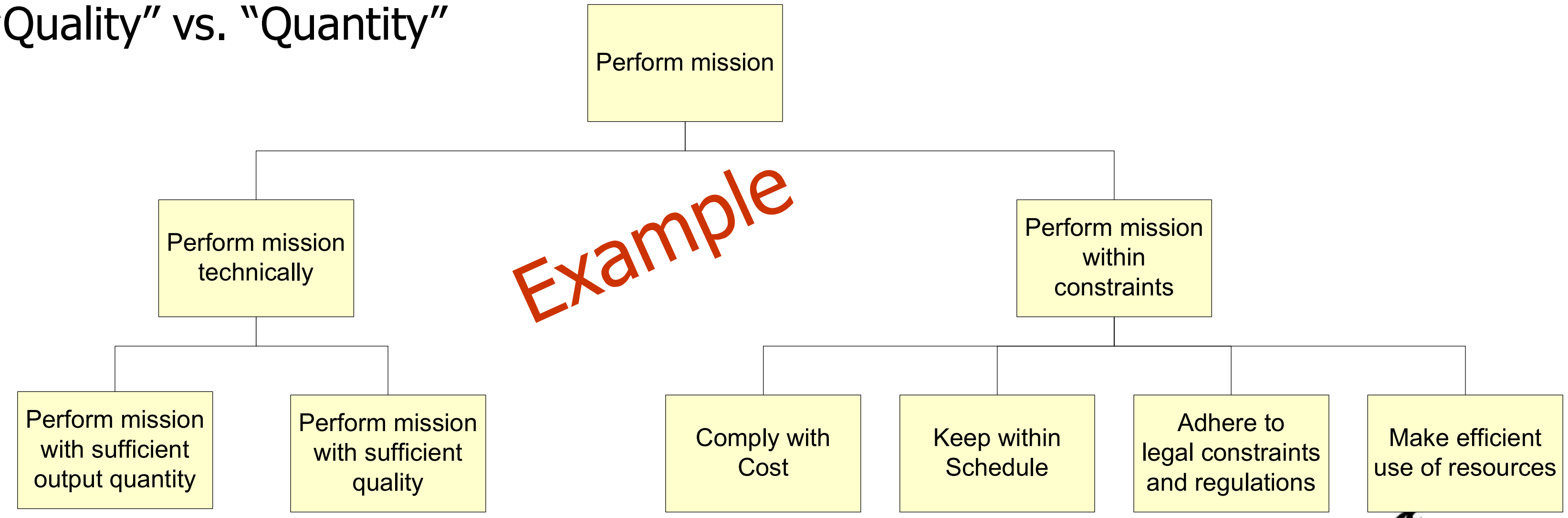
Requirements Discovery Tree



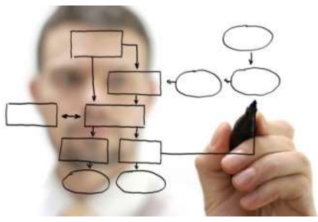
- Basics
  - A requirements discovery tree is a tool to establish - in a structured way - complete and consistent requirements on subsequently lower levels.
  - The process you follow in that way is called requirement breakdown or flow-down.

- Approach: an **AND tree**
  - Every element is the sum of the elements under it and nothing more.
  - If done properly, no requirement will be forgotten.

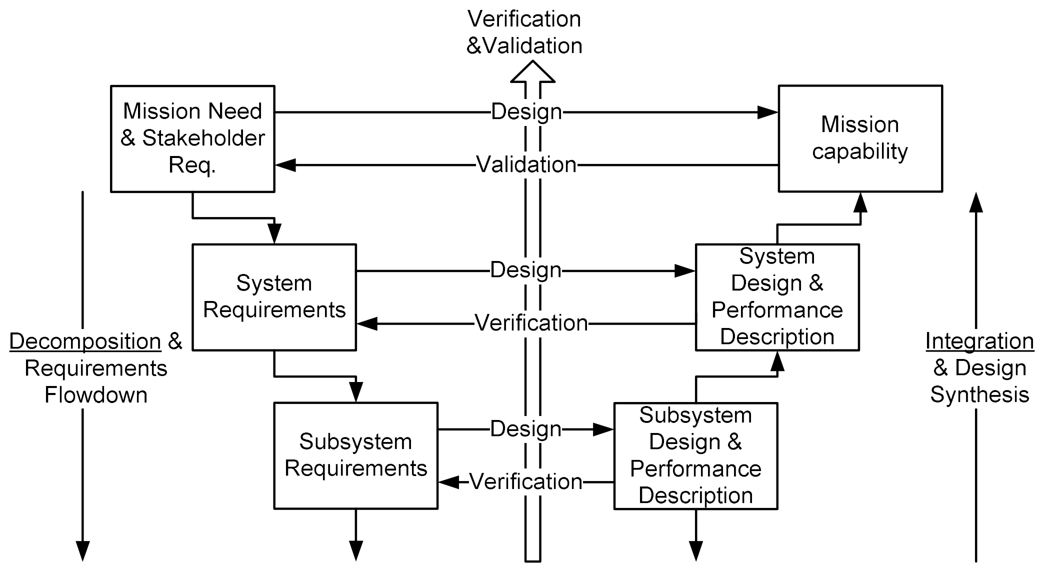
- Basic guideline: Split requirements up in two parts (dichotomy)
  - E.g. "Functional requirements" versus "Constraints", or
  - "Quality" vs. "Quantity"


Perform mission
Perform mission technically
Perform mission within constraints
Example
Perform mission with sufficient output quantity
Perform mission with sufficient quality
Comply with Cost
Keep within Schedule
Adhere to legal constraints and regulations
Make efficient use of resources

Requirements Verification



- Requirements are not only important during the top-down design process.
- You also need to verify them during **integration**.



Verification &Validation
Mission Need & Stakeholder Req.
Design
Validation
Mission capability
System Requirements
Design
Verification
System Design & Performance Description
Decomposition & Requirements Flowdown
Integration & Design Synthesis
Subsystem Requirements
Design
Verification
Subsystem Design & Performance Description

There are 4 methods to verify requirements:

1. **Inspection** (*Example: "… shall be 27 cm long."*)
2. **Analysis** (*Example: "… shall have a probability of … ."*)
3. **Demonstration** (*Example: "… shall take less than one hour."*)
4. **Test** (*Example: "… shall be less than 2.5 W."*)

Selecting the best verification method depends on the requirement itself and the project.

Functions

General

- In engineering, a function describes the requested **behavior** of a system.
- When you formulate functions, always use a ***verb***.

Example of a functional requirement:

*GNC-FCT-12.5 The system shall **provide** Cartesian position information.*

Functional analysis comprises:

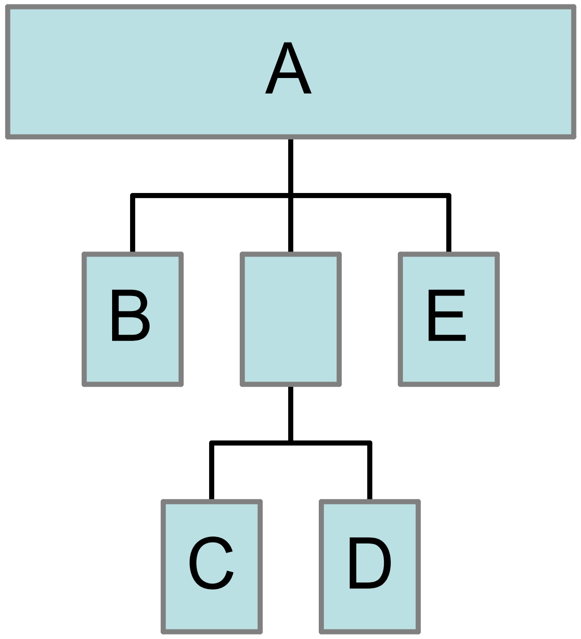
1. Functional Breakdown Structure (FBS) (this lecture)
2. Functional Flow Block diagram (FFBD) (this lecture, see also AE2101)
3. N2 chart (see AE2101)

When you know your requirements, you ask first "what the system shall do" before you ask "how the system shall do that".

GNC: Guidance, Navigation, Control
FCT: Function

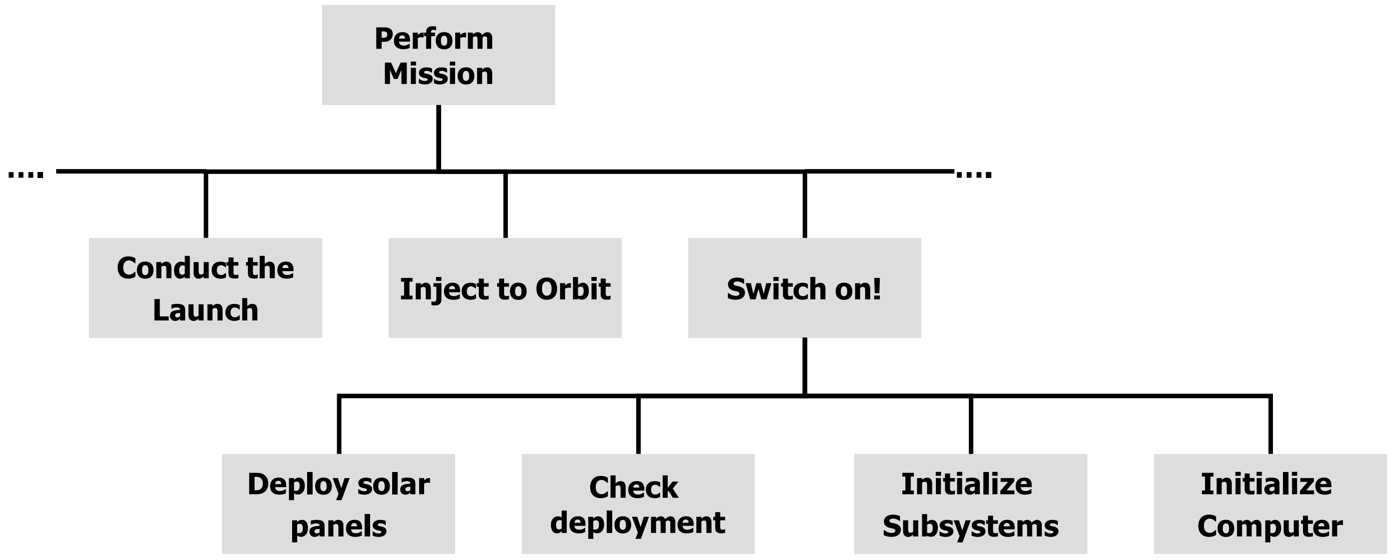
Functional Break-down Structure (FBS)

- Logical grouping according to a predetermined set of criteria
  - An FBS is an **AND** tree, that is:
    - Each element of the tree = the sum of the elements below it and nothing more
    - A = B + (C + D) + E
    - Gives a complete picture of the system or product in terms of functions



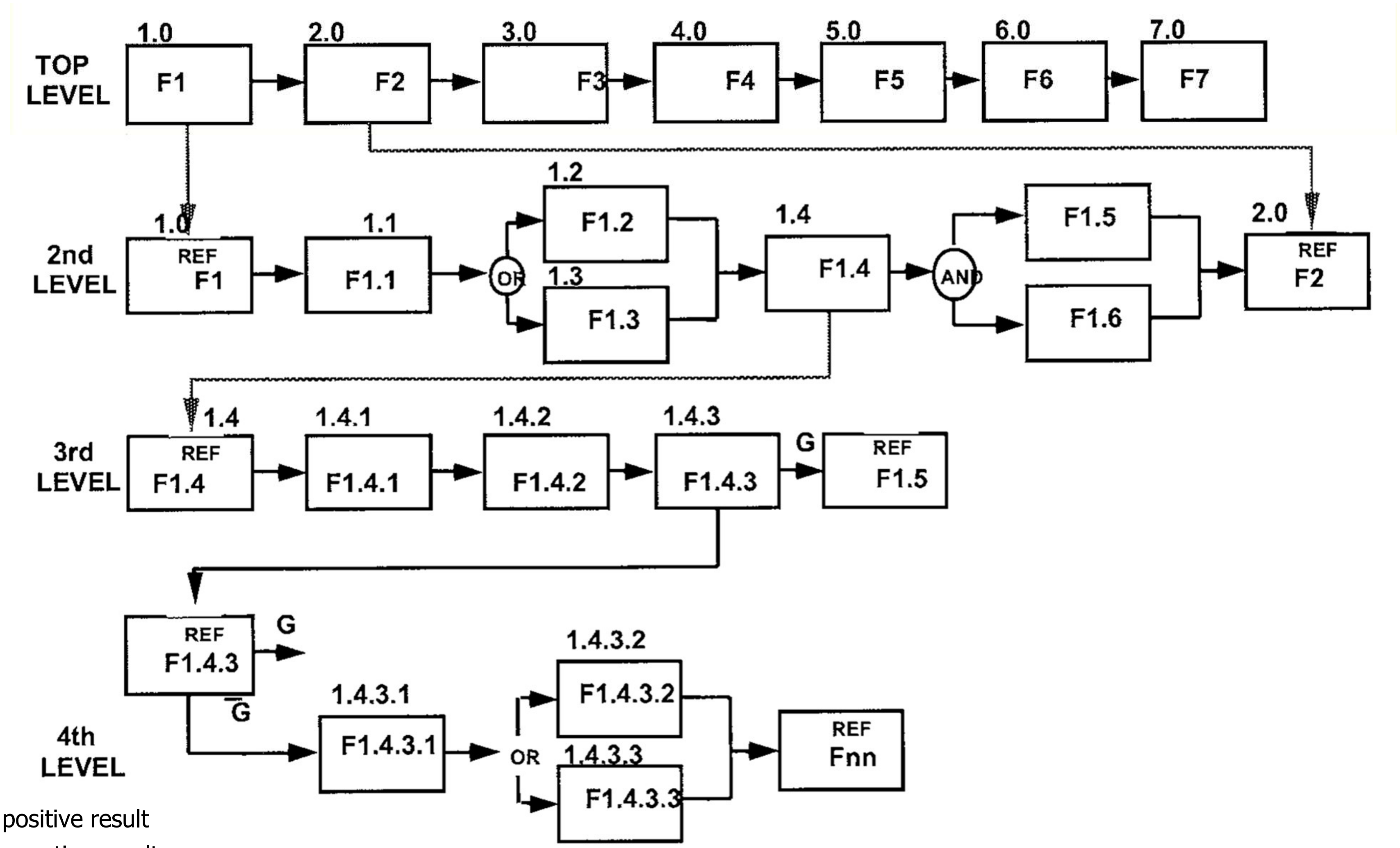
The FBS says nothing about the sequence!

Example *Functional Break-down Structure (FBS)*



Functional Flow Block Diagram (FFBD)

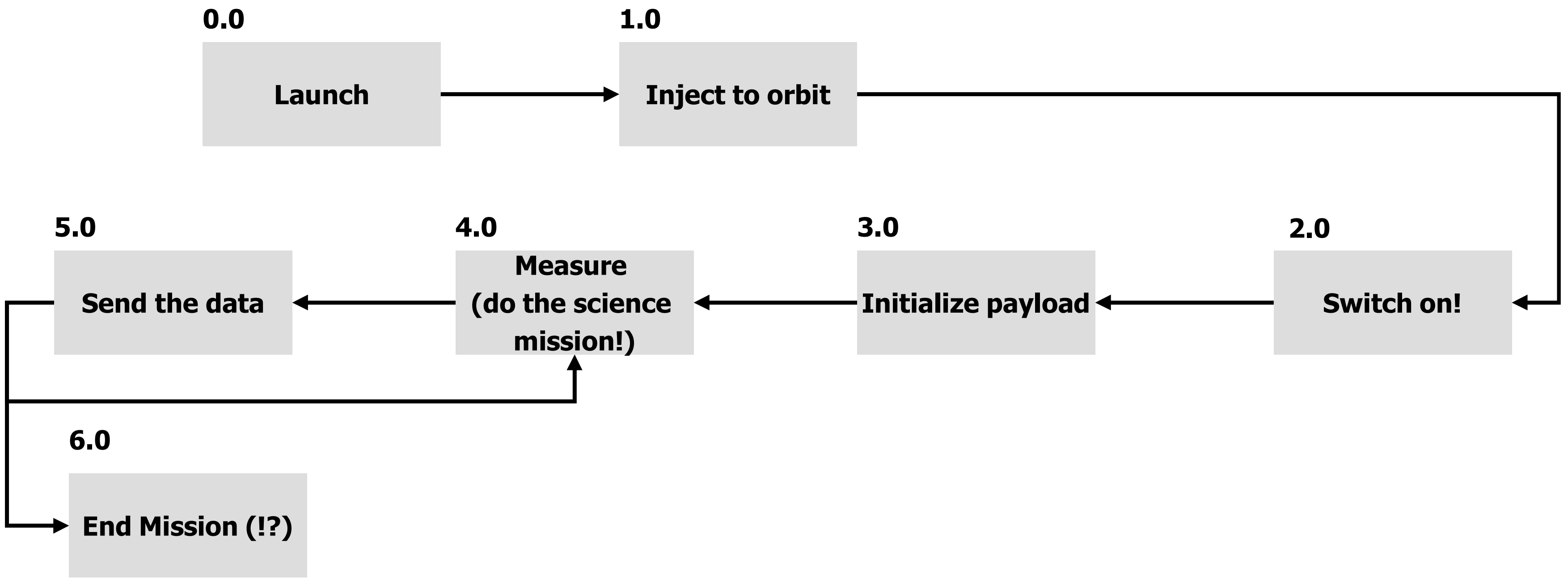
A Functional Flow Block Diagram is a multi-tier, time-sequenced, step-by-step flow diagram of a system's functional flow.

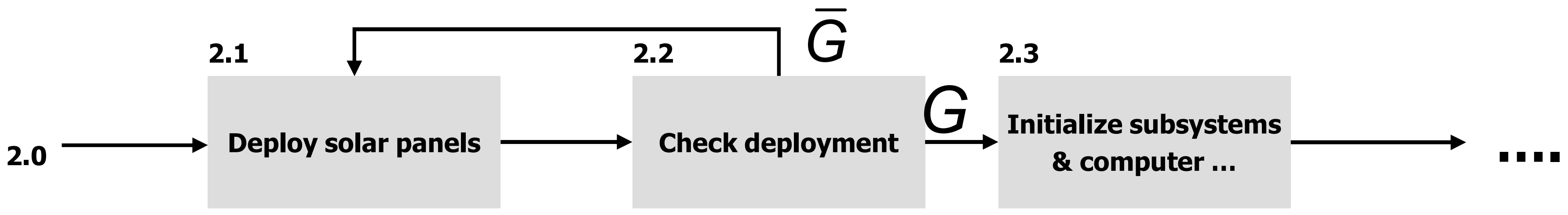


G: positive result

Ḡ: negative result

Example *Functional Flow Block Diagram (FFBD)*

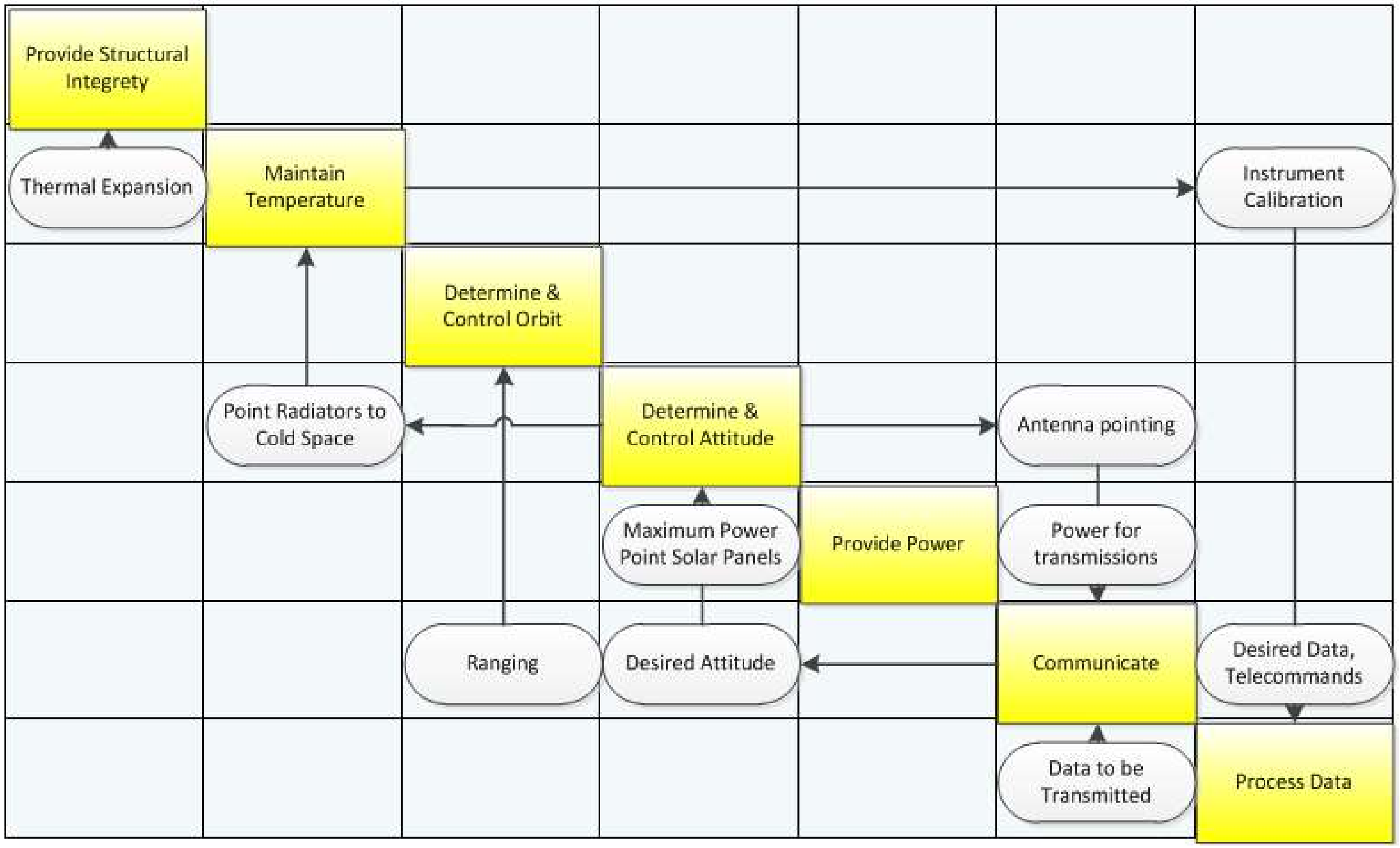




G: positive result
Ḡ: negative result

Homework Exercise

Go back to AE2111 II to recall how you can use an N2 chart to **structure functions** and **describe interfaces**.



Design Trades – Description

Once requirements ("What?") are defined, we have to find, analyse and select a certain concept ("How?") which can best fulfil these requirements. This process is called **design trade**.

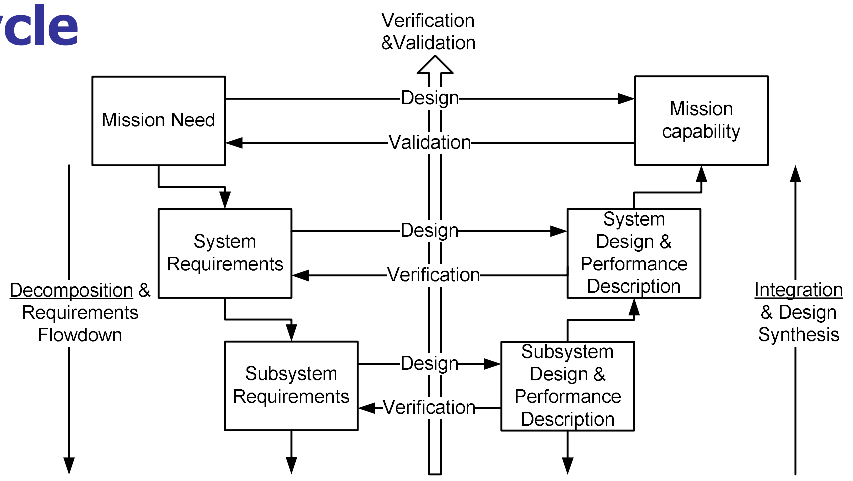
Design trades (design concept selection) consists of three parts:

1. Find **all** design options (Use a Design Option Tree)
2. Do **trade-offs** between all options (Follow the trade-off process)
3. **Record** and present the results (Document process, assumptions, findings)

Design Trades are Everywhere

Design trades occur

- throughout the **entire development cycle**


Verification &Validation
Mission Need
Design
Validation
Mission capability
System Requirements
Design
Verification
System Design & Performance Description
Decomposition & Requirements Flowdown
Subsystem Requirements
Design
Verification
Subsystem Design & Performance Description
Integration & Design Synthesis

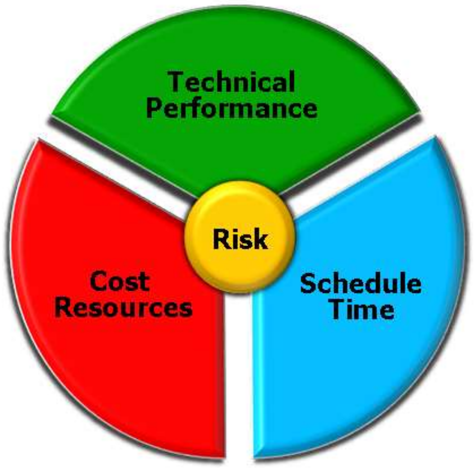
- at **all levels** of the system hierarchy

System	e.g. Space mission
Segment	e.g. Ground segment
Element	e.g. Satellite
Subsystem	e.g. ACS
Assembly	e.g. Sun sensor
Component	e.g. PCB

ACS: Attitude Control System
PCB: Printed Circuit Board

Design Trades - Motivation

- Design trades lead to **decisions on implementation**.
- These decisions may have a crucial impact on your project for:



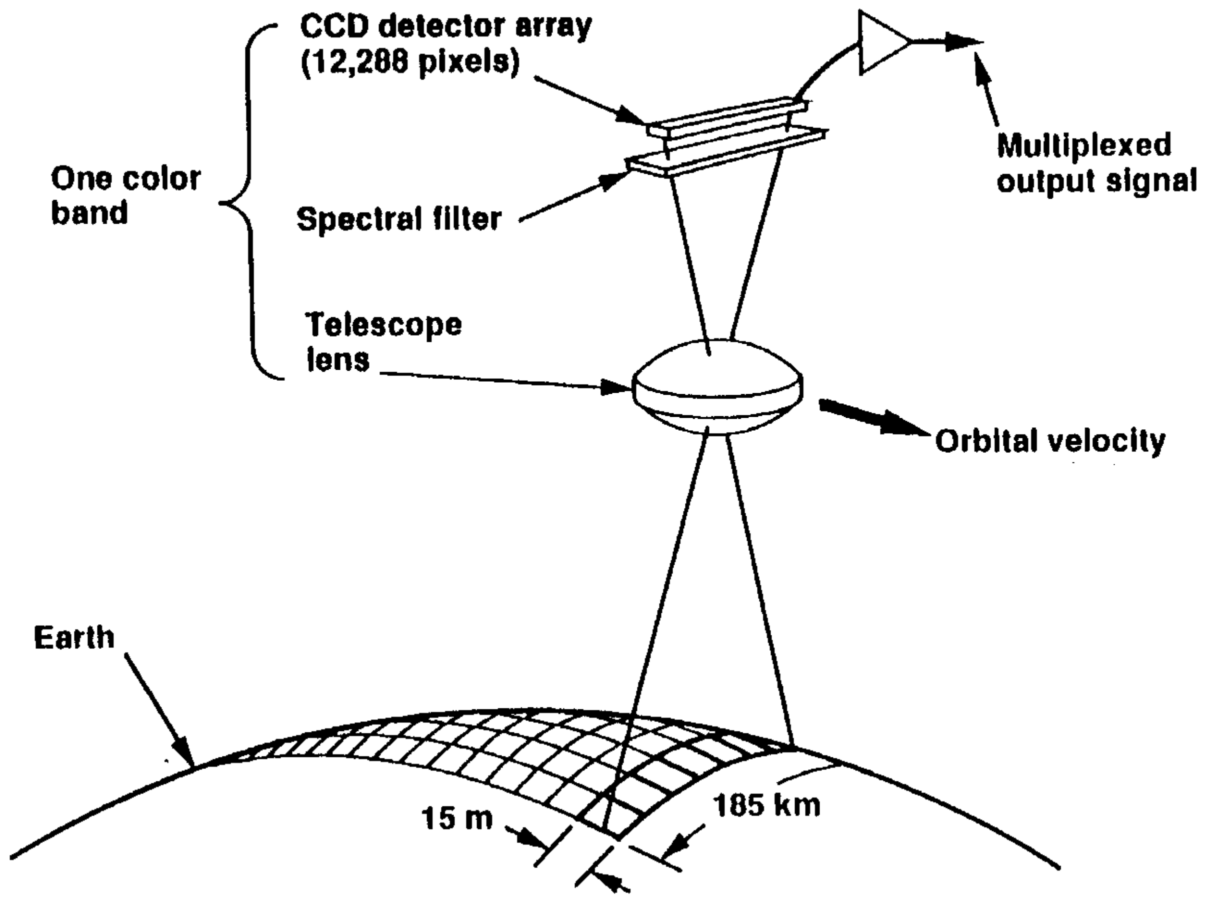
⇨ Being able to do design trades **properly** is crucial!

In the beginning of a design project, only few but crucial trades need to be done. The more you progress, the more trades you need to do and the less relevant they are.

Design Trades – Generating Concepts

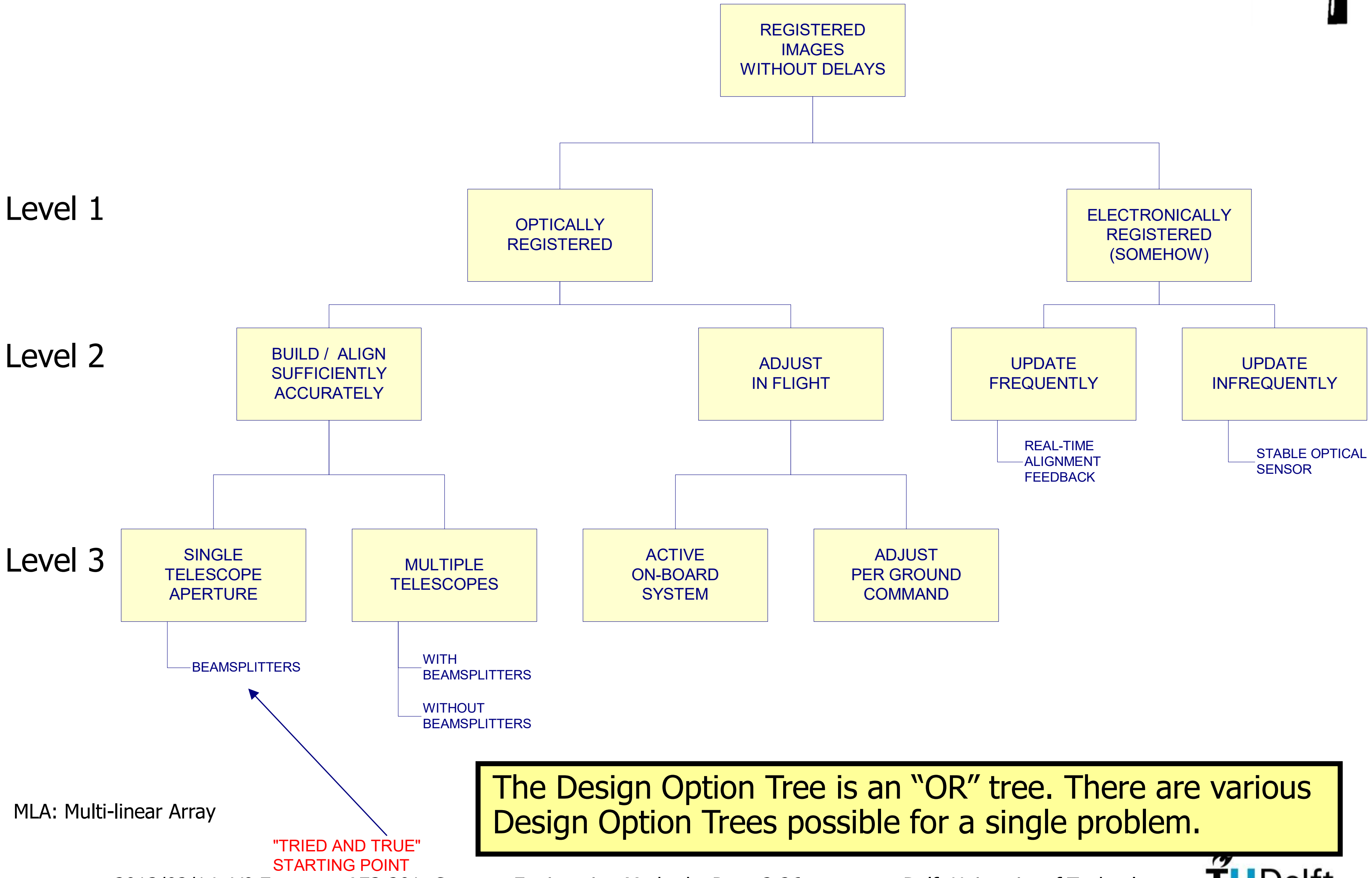
Generate concepts using a **Design Option Tree** (trade tree)!

Example: *Multi-spectral Linear Array (MLA)*



<u>Problem</u>: Which concepts are there to register images in several spectral bands that must be fitted to each other in real-time with acceptable accuracy?

Design Option Tree (Example for MLA)


REGISTERED IMAGES WITHOUT DELAYS
Level 1
OPTICALLY REGISTERED
ELECTRONICALLY REGISTERED (SOMEHOW)
Level 2
BUILD / ALIGN SUFFICIENTLY ACCURATELY
ADJUST IN FLIGHT
UPDATE FREQUENTLY
UPDATE INFREQUENTLY
REAL-TIME ALIGNMENT FEEDBACK
STABLE OPTICAL SENSOR
Level 3
SINGLE TELESCOPE APERTURE
MULTIPLE TELESCOPES
ACTIVE ON-BOARD SYSTEM
ADJUST PER GROUND COMMAND
BEAMSPLITTERS
WITH BEAMSPLITTERS
WITHOUT BEAMSPLITTERS
"TRIED AND TRUE" STARTING POINT

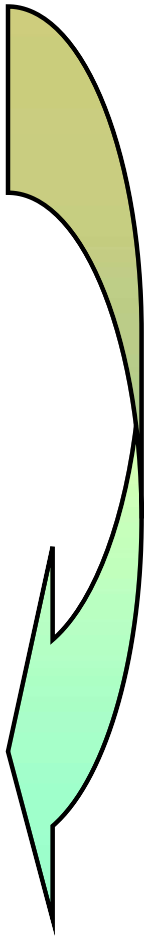
The Design Option Tree is an "OR" tree. There are various Design Option Trees possible for a single problem.

MLA: Multi-linear Array

Hints on Design Option Trees

- Design Option Tree (DOT) must be **complete** to be useful!
  - Each element has one or more descendant and nothing else
  - Impossible to omit anything if rigorous logic is applied
  - Every possibility to be included:
    1. “Tried and true” solutions
    2. Reasonable alternatives
    3. Apparently silly alternatives
    4. Things that can be named, but for which no concept is known yet
  - Use of "anything else" as temporary placeholder
    - Keeps you from getting stuck during thinking process
    - Must eventually be replaced by specifics or be removed

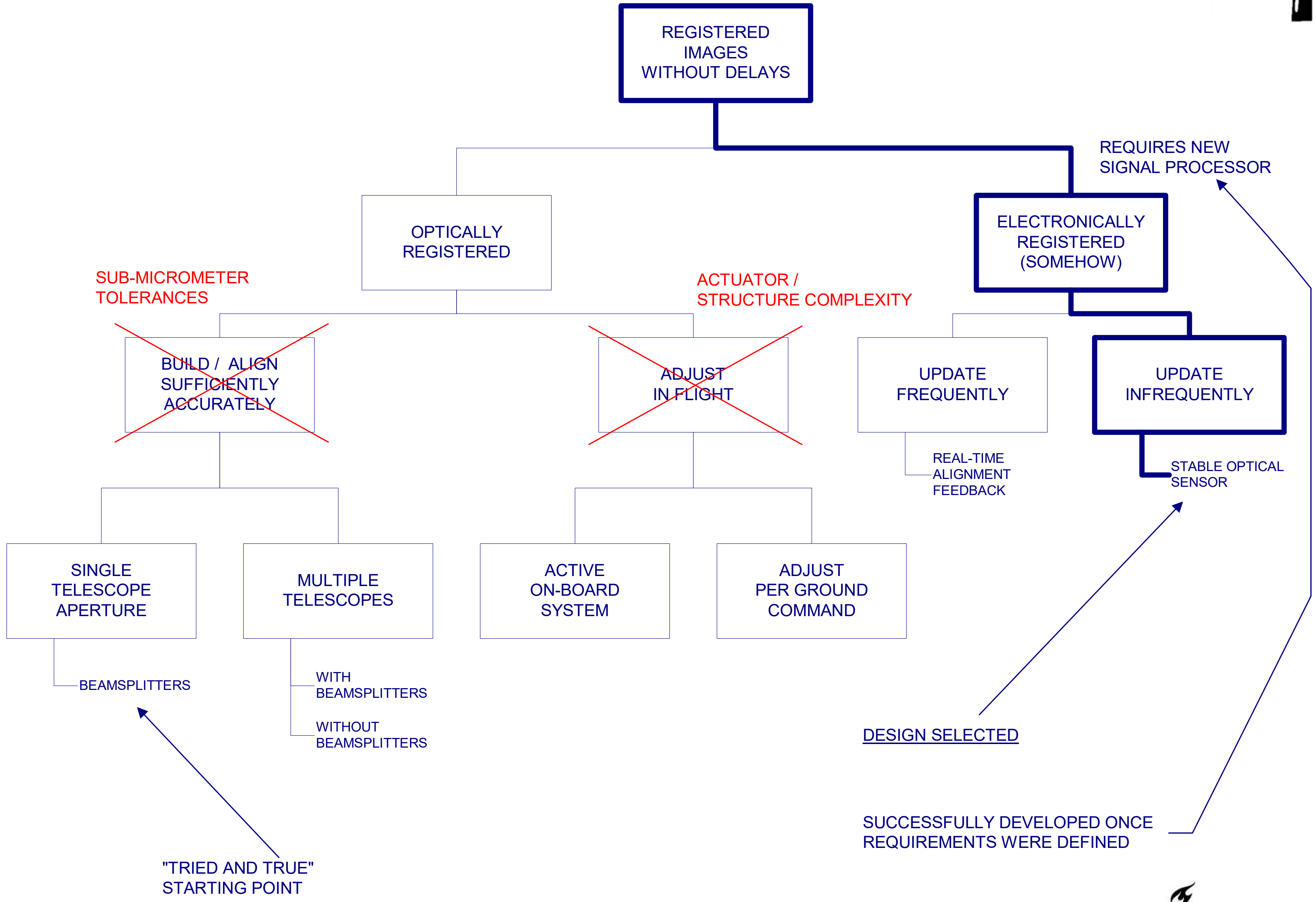
Establish DOT



Evaluate DOT

- Importance of looking for alternatives to "tried and true" depends on how successful the usual solution is
- Use scientific method to disprove the usefulness of options
  - Test against killer requirements, laws of physics, etc.
  - A "no-go" decision prunes all descendants from tree

Evaluation using a Design Option Tree


REGISTERED IMAGES WITHOUT DELAYS
OPTICALLY REGISTERED
ELECTRONICALLY REGISTERED (SOMEHOW)
REQUIRES NEW SIGNAL PROCESSOR
SUB-MICROMETER TOLERANCES
ACTUATOR / STRUCTURE COMPLEXITY
~~BUILD / ALIGN SUFFICIENTLY ACCURATELY~~
~~ADJUST IN FLIGHT~~
UPDATE FREQUENTLY
UPDATE INFREQUENTLY
REAL-TIME ALIGNMENT FEEDBACK
STABLE OPTICAL SENSOR
SINGLE TELESCOPE APERTURE
MULTIPLE TELESCOPES
ACTIVE ON-BOARD SYSTEM
ADJUST PER GROUND COMMAND
BEAMSPLITTERS
WITH BEAMSPLITTERS
WITHOUT BEAMSPLITTERS
DESIGN SELECTED
SUCCESSFULLY DEVELOPED ONCE REQUIREMENTS WERE DEFINED
"TRIED AND TRUE" STARTING POINT

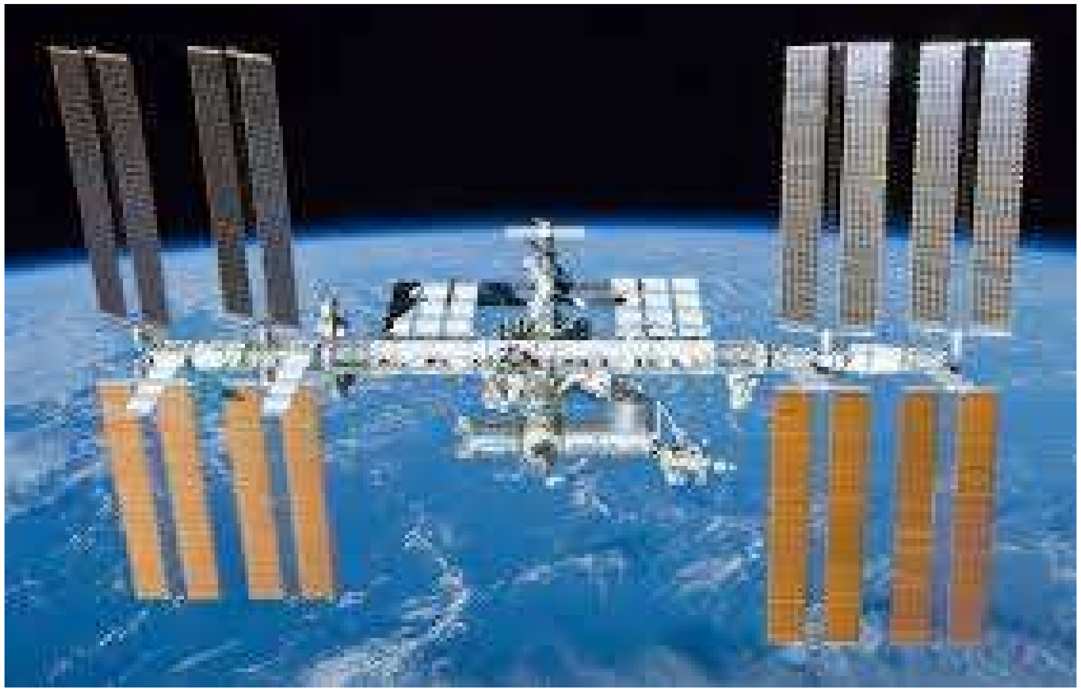
Exercise *Design Option Tree*

The International Space Station needs permanent communications capability to the ground segment.

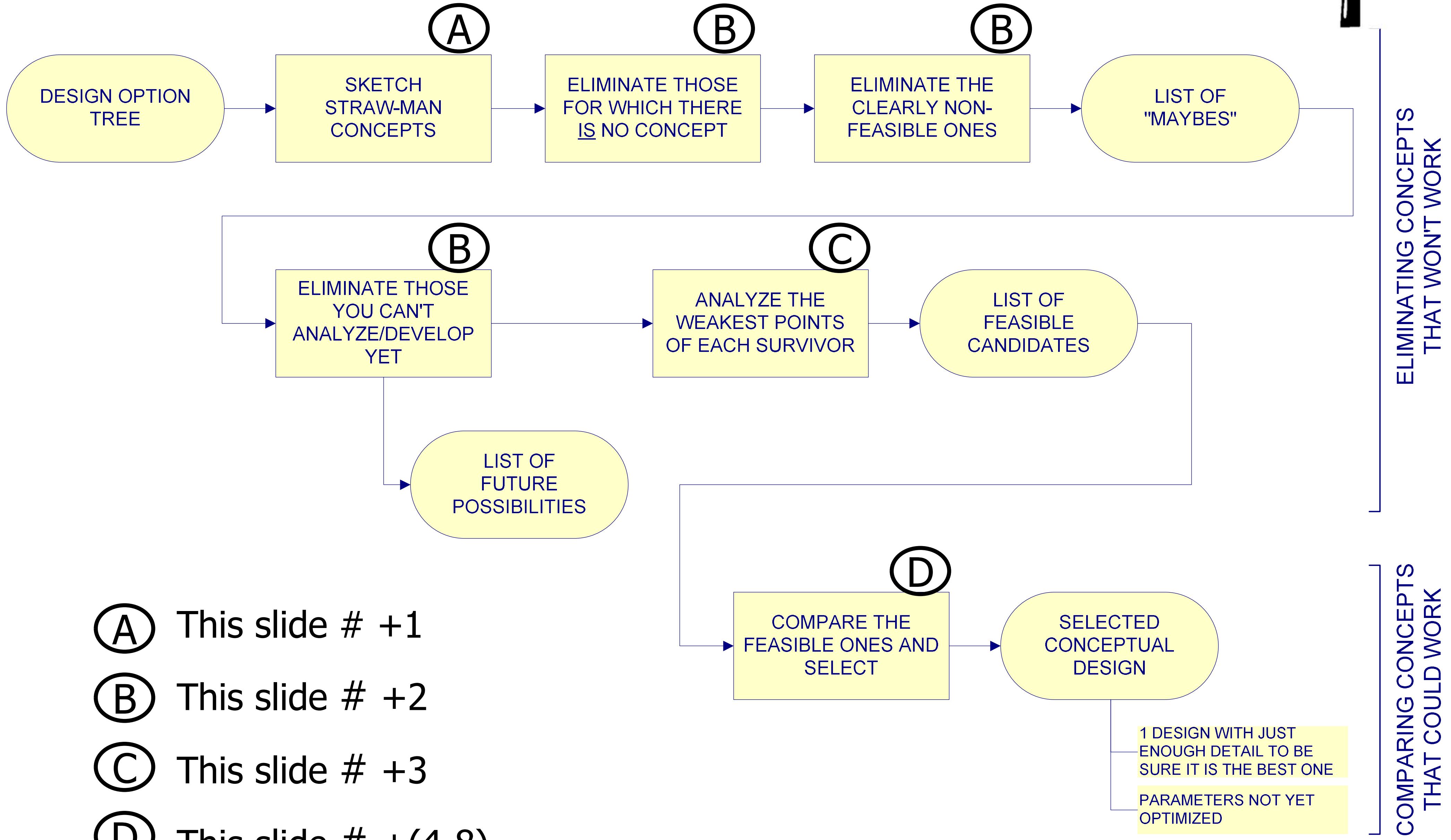
Develop a Design Option Tree to level 1 with respect to either

- Communications technology, or
- Communications architecture

and identify the non-recommended options.



Trade-off process



A
B
B
DESIGN OPTION TREE
SKETCH STRAW-MAN CONCEPTS
ELIMINATE THOSE FOR WHICH THERE IS NO CONCEPT
ELIMINATE THE CLEARLY NON-FEASIBLE ONES
LIST OF "MAYBES"
B
C
ELIMINATE THOSE YOU CAN'T ANALYZE/DEVELOP YET
ANALYZE THE WEAKEST POINTS OF EACH SURVIVOR
LIST OF FEASIBLE CANDIDATES
LIST OF FUTURE POSSIBILITIES
D
COMPARE THE FEASIBLE ONES AND SELECT
SELECTED CONCEPTUAL DESIGN
1 DESIGN WITH JUST ENOUGH DETAIL TO BE SURE IT IS THE BEST ONE
PARAMETERS NOT YET OPTIMIZED
ELIMINATING CONCEPTS THAT WON'T WORK
COMPARING CONCEPTS THAT COULD WORK

Ⓐ	This slide # +1
Ⓑ	This slide # +2
Ⓒ	This slide # +3
Ⓓ	This slide # +(4-8)

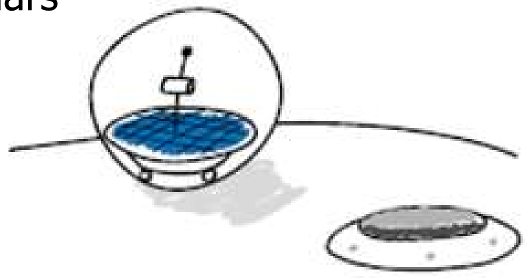
Trade-off Process - Strawman concepts

- **Specific** concepts to think about
  - Representatives of each major branch of the design option tree
  - Enough detail and realism to enable to see what the questions are

- Use **familiar** concepts, e.g.
  - "Tried and true" designs
  - Customer's favourite design

- Use **new** concepts
  - Many sources of ideas available (Experts, invited people, media, etc.)
  - Experience, general curiosity, everything you have ever seen

Examples of strawman concepts of how to move on Mars



Trade-off Process - Eliminating Concepts

- Eliminate the **non-concepts**
  - Things named for the sake of completeness, but for which no implementation can be found

- Eliminate the obviously **non-feasible** ones
  - Things forbidden by laws of nature or that grossly violate practicality

- Eliminate concepts that **can't be analyzed or developed** yet
  - Concepts, that might be workable, but are not worth pursuing now (e.g. new technology you don't have resources to develop now, too difficult to analyse within your schedule)

Write down eliminated concepts that can't be analyzed or developed yet and why you have eliminated them!

Trade-off Process – Analyze Weakness of Survivors

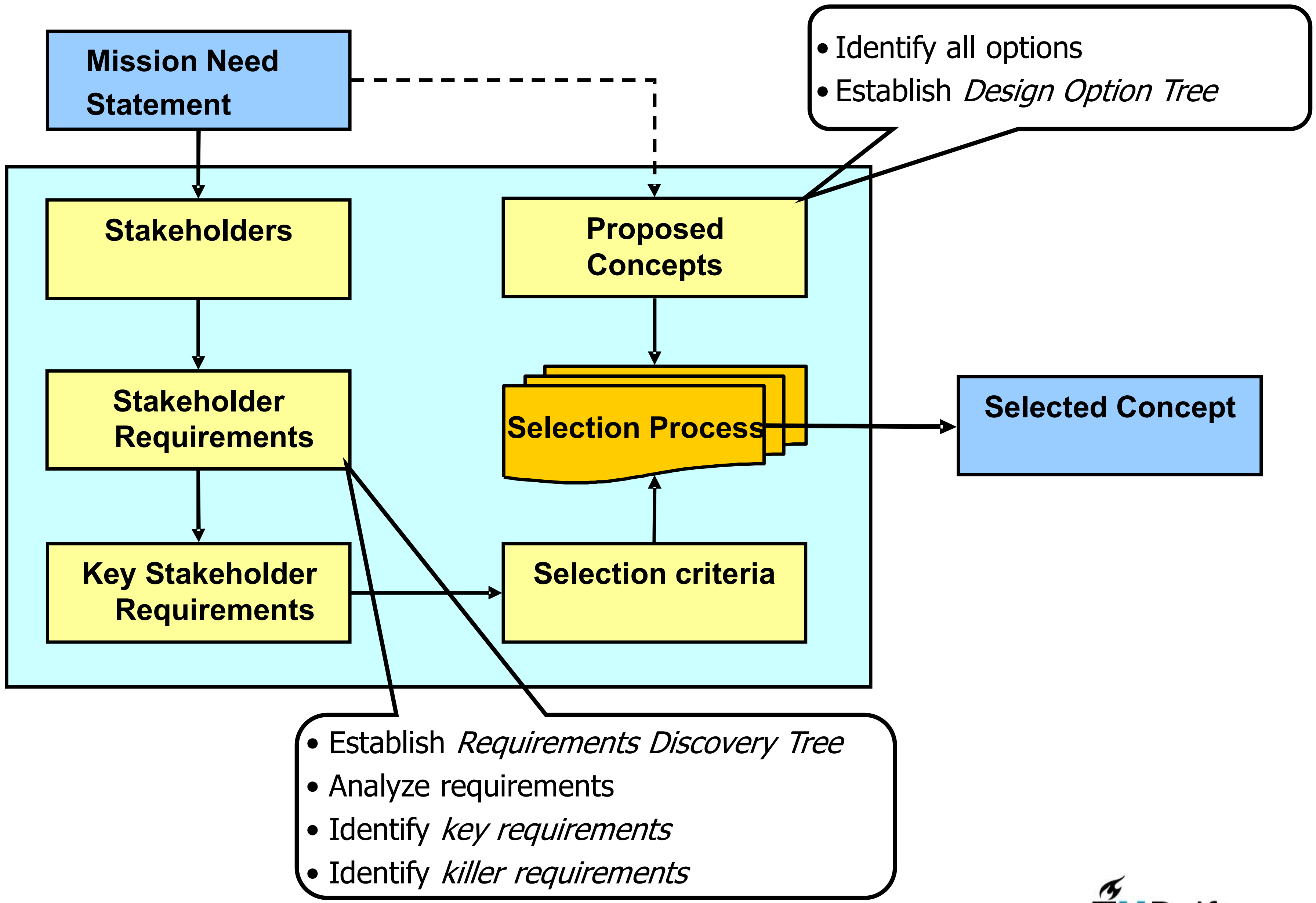
- Analyse the weakest points of each surviving candidate
  - Understand the **governing principles**
  - Find the things most likely to **keep this concept from working**
  - Gather required data and **do analysis** to try to prove it can't work

- *Examples:*
  - *Concept X looks too heavy*
  - *Analyse weight of X*
  - *Eliminate X as infeasible, if weight exceeds some hard constraint*

- Will give you insight you need for the final selection
  - Eliminates all but the truly feasible concepts
  - If *X* is feasible, include weight among the trade-off criteria
  - May identify unexpected trade-off driving requirements or issues

Focusing on the weak points guides you through the selection process efficiently.

High-level Selection Process



- Identify all options
- Establish *Design Option Tree*

- Establish *Requirements Discovery Tree*
- Analyze requirements
- Identify *key requirements*
- Identify *killer requirements*

Design Trades – Selection Criteria

Define **selection criteria** from 3 sources:

1. Killer or key requirements identified in initial requirements analysis
2. Other system-level requirements in which candidate concepts differ significantly
3. Additional selection criteria from analyzing strawman design

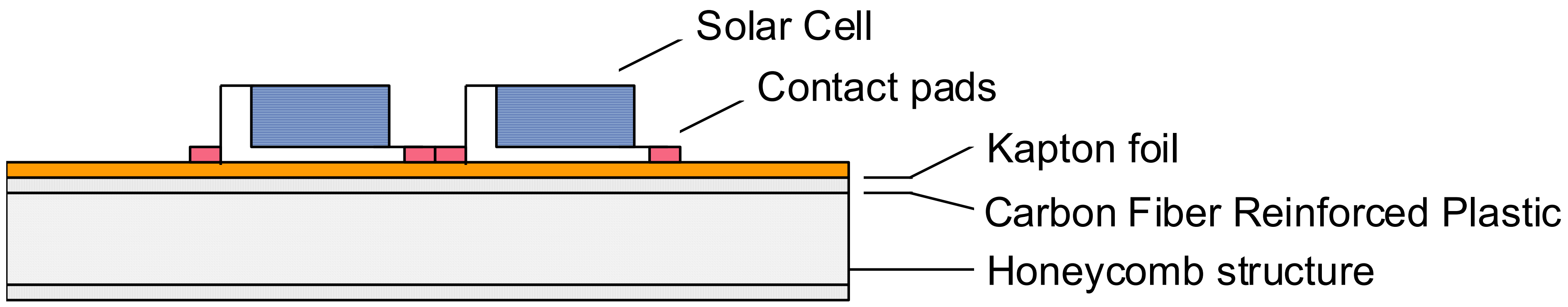
Define **weights** of selection criteria:

- Each selection criterion receives a weight indicating its importance.

Define a limited number of selection criteria, e.g. 3-5.

Example on Selection Criteria

Solar arrays need isolation between solar cells and supporting structure!



Selection criterea (selection weights):

1. Isolation performance (3/5)
2. Mass performance (4/5)
3. Development cost (1/5)
4. Development risk (4/5)

Comparison and Selection: Graphical Trade-off

Trade-off summary table (matrix) with the following characteristics:

- Options in left hand column; criteria in top row
- **Column width** proportional to the weight of criteria
- **Reason for judgement** summarised in the box
- Score **classes defined clearly**, e.g.
  - Excellent
  - Good
  - Correctable deficiencies
  - Unacceptable
- **Colour** used for oral presentation (labels convey information on black/white copy; beware of colour-blind!)

Other than Graphical Trade-off, comparison and selection can be done using a classical approach (numerical scoring scheme; Pugh matrix) or the Analytical Hierarchy Process (AHP; pair-wise comparison).

Example *Graphical Trade-off Table*

CRITERION / OPTION	isolation performance	mass performance	dev. cost	development risk
CFRP skins + 50 μm Kapton film (current B/L)	blue no anomalies	blue nominal	green zero delta cost	green no risk
CFRP skin + alternative film	green TBC by tests	blue nominal	blue slight delta cost	red product not yet developed
Kevlar composite skin	green very good	red 13% heavier	yellow high delta cost	blue routine space development
Kevlar composite skin + CFRP doubler	blue better than B/L; not fully proven	yellow 5% heavier	yellow high delta cost	yellow possible delamination

green	excellent, exceeds requirements	yellow	Correctable deficiencies
blue	Good, meets requirements	red	Unacceptable

CFRP:	Carbon Fiber Reinforced Plastic
B/L:	Baseline
TBC:	To be confirmed

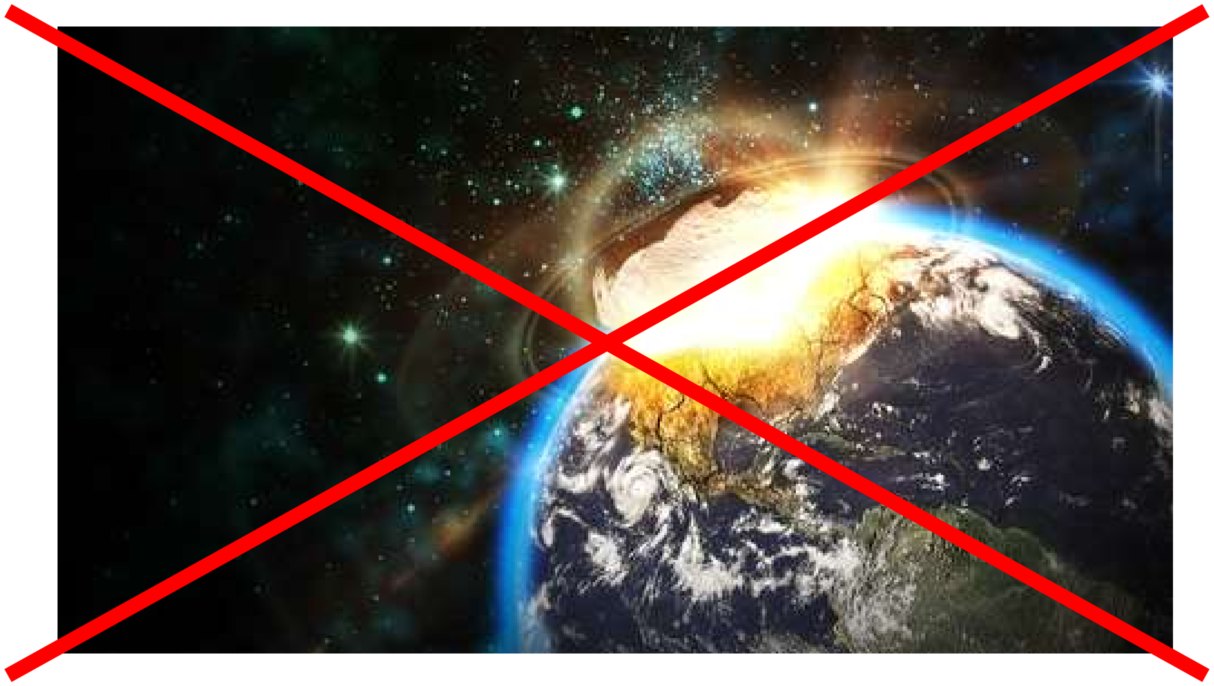
TU Delft

Exercise *Graphical Trade-Off Table*

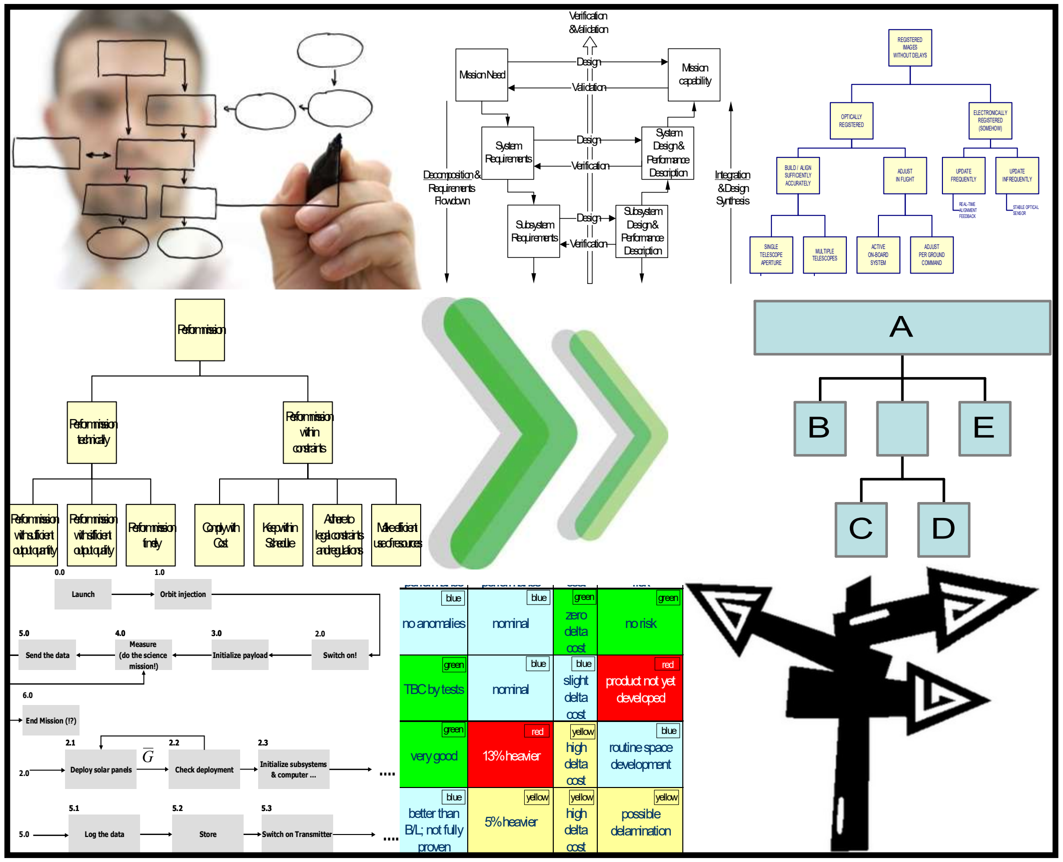
Develop a Graphical Trade-Off Table for the *Save-The-World* mission discussed earlier. Use a 2x2 table with the following concepts:

1. Gravitational Tractor
2. Nuclear Explosive

To this end, develop 2 selection criteria first, then identify their weights, finally fill the table. What is your result?



What you have learned



Homework:

- Do the exercises of this lecture
- Re-visit the lecture "Analysis of Requirements" on course AE1201
- Recall the use of N2 charts from the course AE2101

Appendix:

- Literature on Systems Engineering Methods

Literature

- Wertz J.R., Larson W.J.; Space Mission Analysis and Design; Third Edition; Microcosm, Inc. (1999)
  The classics on Space Mission Analysis and Design with excellent content.