Requirements, Risk, and Trades

MAE 4160, 4161, 5160

V. Hunter Adams, PhD

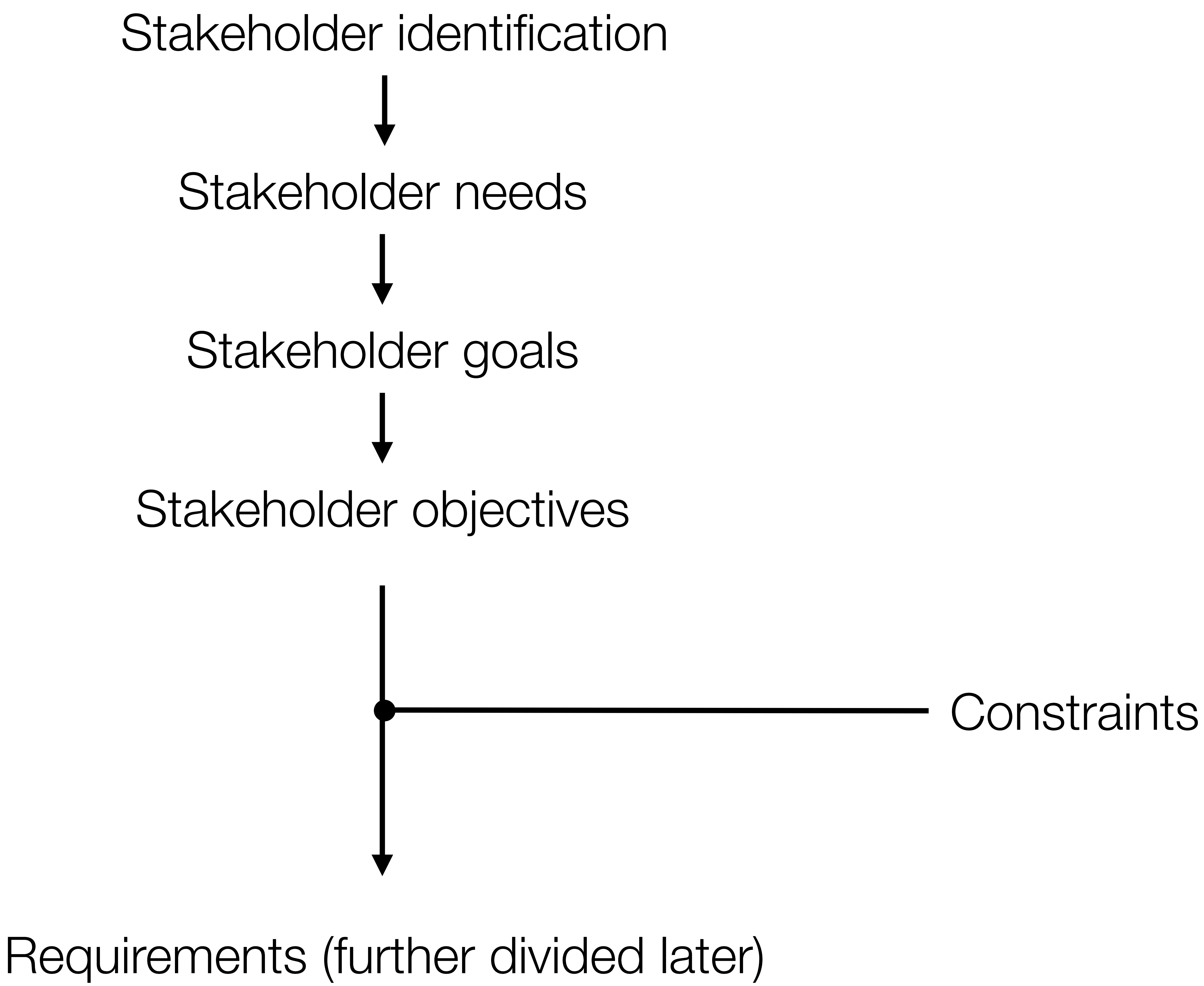
Today's topics:

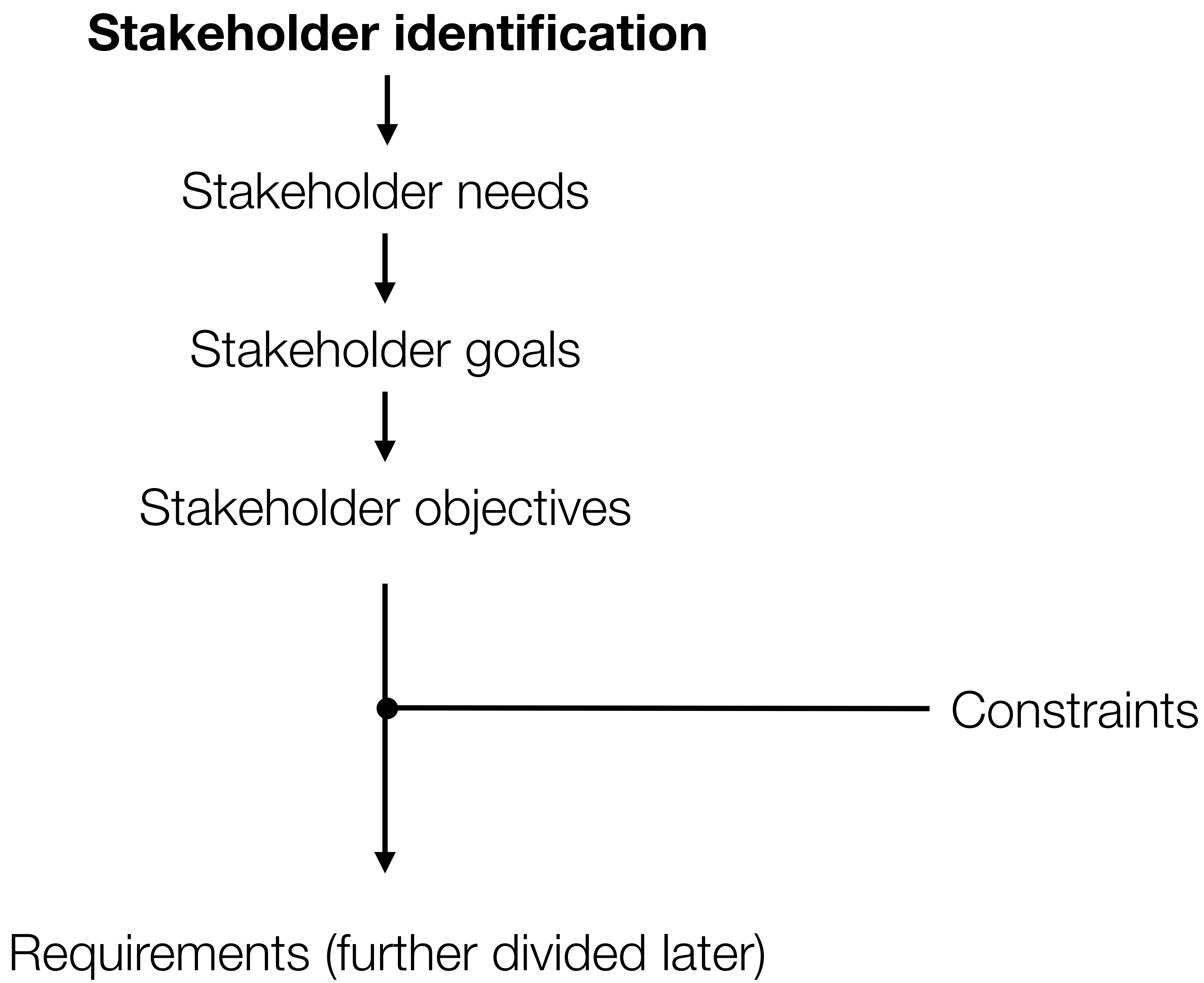
- Identifying stakeholders
- Identifying stakeholder needs, goals, and objectives
- Technical requirements definition process
- Types of requirements
- How to write valid requirements
- Requirements verification
- Trade studies
- Risk

It all starts with **stakeholders**!

Stakeholder: Any group or individual that is affected by or has a stake in the product or project.

The success of a system depends entirely on satisfying stakeholders. It is not about maximizing performance or minimizing cost, it is about satisfying stakeholder expectations.


Stakeholder identification
Stakeholder needs
Stakeholder goals
Stakeholder objectives
Constraints
Requirements (further divided later)


Stakeholder identification
Stakeholder needs
Stakeholder goals
Stakeholder objectives
Constraints
Requirements (further divided later)

Stakeholder identification

TABLE 4.1-1 Stakeholder Identification throughout the Life Cycle

Life-Cycle Stage	Example Stakeholders
Pre-Phase A	NASA Headquarters, NASA Centers, Presidential Directives, NASA advisory committees, the National Academy of Sciences
Phase A	Mission Directorate, customer, potential users, engineering disciplines, safety organization
Phase B	Customer, engineering disciplines, safety, crew, operations, logistics, production facilities, suppliers, principle investigators
Phase C	Customer, engineering disciplines, safety, crew, operations, logistics, production facilities, suppliers, principle investigators
Phase D	Customer, engineering disciplines, safety, crew, operations, training, logistics, verification team, Flight Readiness Board members
Phase E	Customer, system managers, operations, safety, logistics, sustaining team, crew, principle investigators, users
Phase F	Customer, NASA Headquarters, operators, safety, planetary protection, public

+contractors, media, regulatory agencies, congress . . .

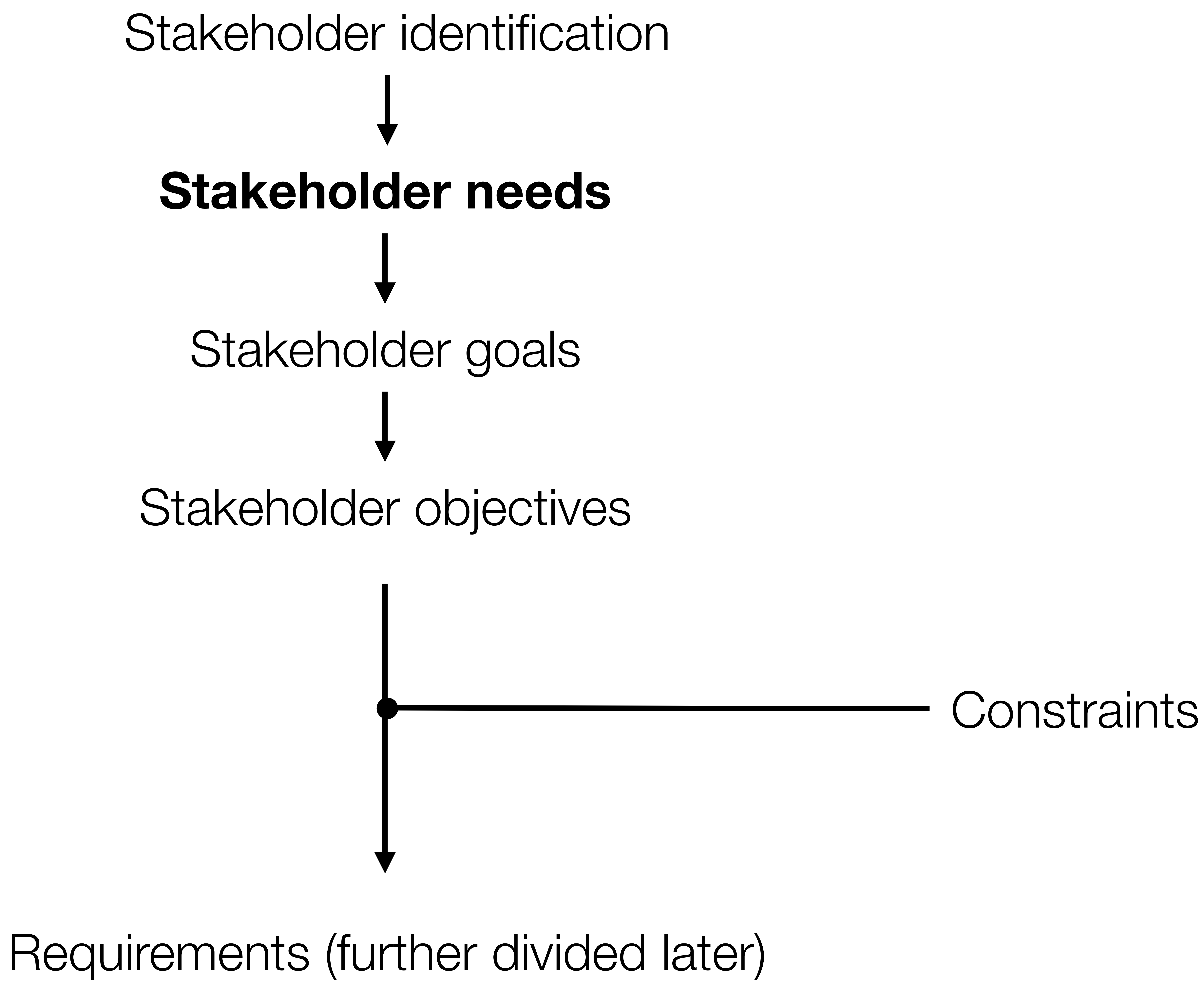
Stakeholder identification

TABLE 4.1-1 Stakeholder Identification throughout the Life Cycle

Life-Cycle Stage	Example Stakeholders
Pre-Phase A	NASA Headquarters, NASA Centers, Presidential Directives, NASA advisory committees, the National Academy of Sciences
Phase A	Mission Directorate, customer, potential users, engineering disciplines, safety organization
Phase D	Customer, engineering disciplines, safety, crew, operations, training, logistics, verification team, Flight Readiness Board members
Phase E	Customer, system managers, operations, safety, logistics, sustaining team, crew, principle investigators, users
Phase F	Customer, NASA Headquarters, operators, safety, planetary protection, public

Who is funding the project? Who is architecting the project? Who is building the project? Which regulatory agencies affect the project? Who will use the data from the project? Who else has opinions that matter?

+contractors, media, regulatory agencies, congress . . .


Stakeholder identification
Stakeholder needs
Stakeholder goals
Stakeholder objectives
Constraints
Requirements (further divided later)

Stakeholder needs

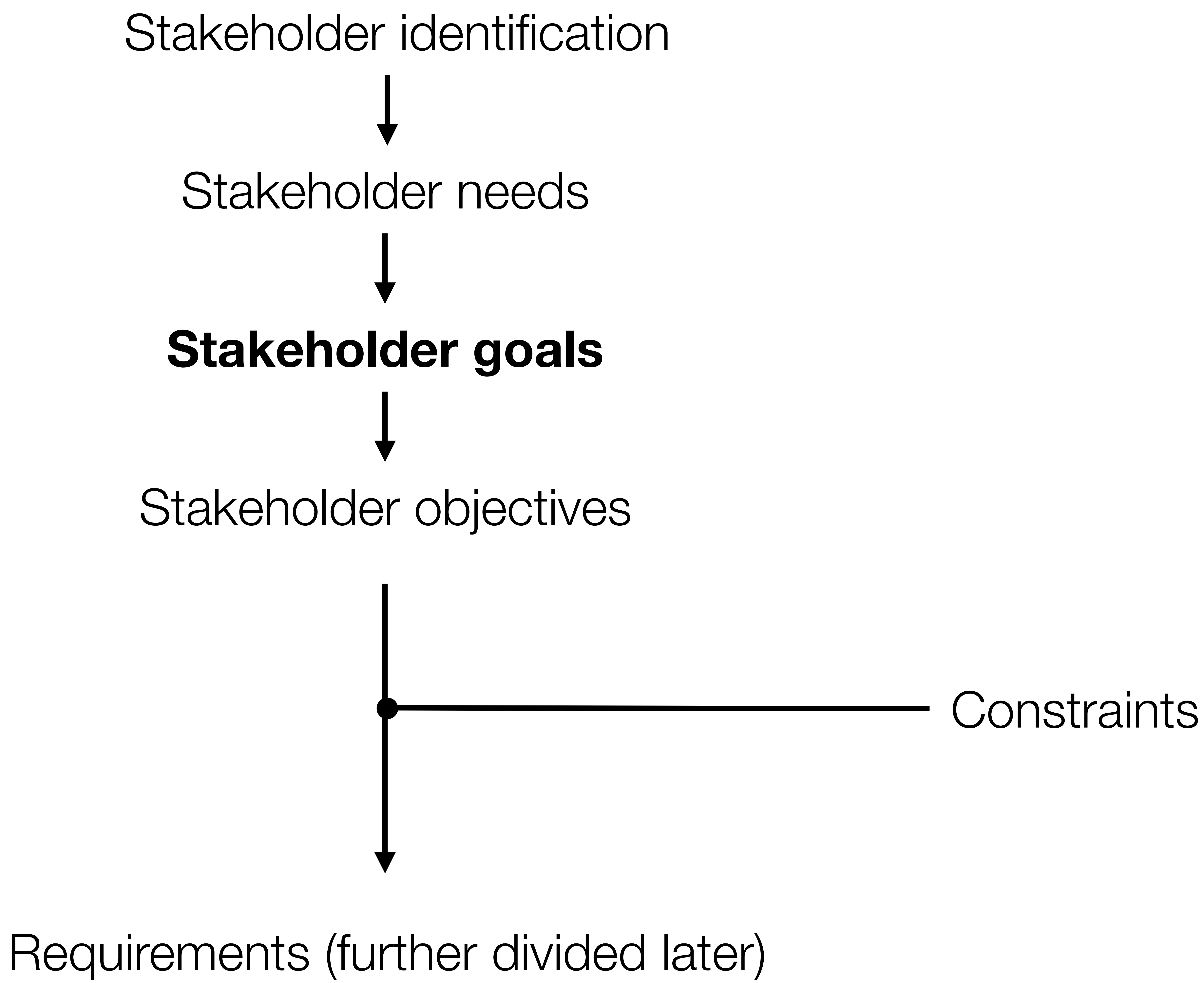
<u>Needs</u>: A single statement that drives everything else. It should relate to the problem that the system is supposed to solve but not be the solution. The need statement is singular. Trying to satisfy more than one need requires a trade between the two, which could easily result in failing to meet at least one, and possibly several, stakeholder expectations.

Monitor changes in the Earth's surface. - Landsat

Counter Soviet military threat. - Apollo

- Drives everything else
- Related to strategic or business plan
- *Not* a definition of the system or solution
- Explains why the project exists from a stakeholder point of view.
- Does not change much during the life of the project.

Can you come up with any other examples?

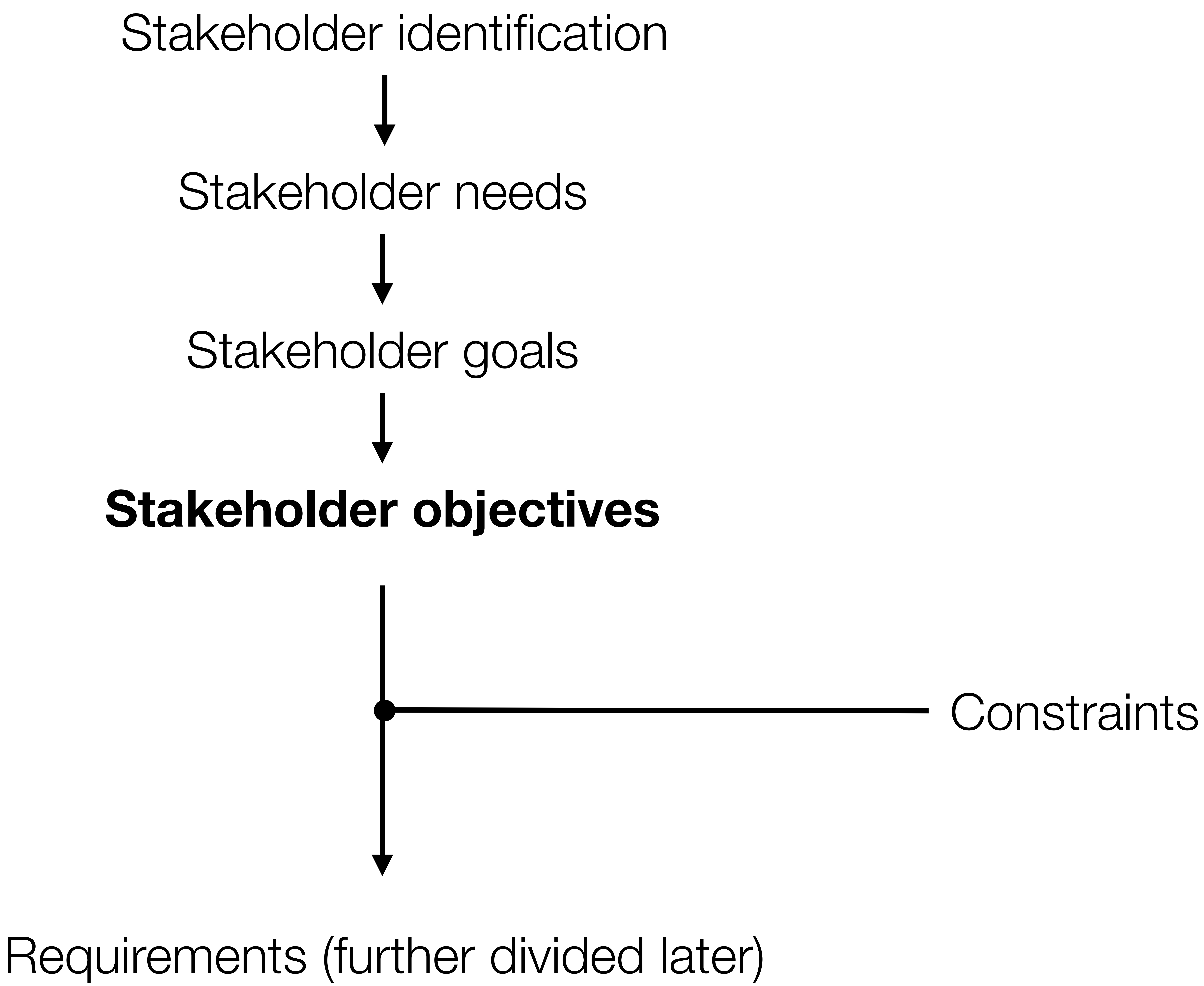

Stakeholder identification
Stakeholder needs
Stakeholder goals
Stakeholder objectives
Constraints
Requirements (further divided later)

Needs become goals

<u>Goals</u>: An elaboration of the need, which constitutes a specific set of expectations for the system. Goals address the critical issues identified during the problem assessment. Goals need not be in a quantitative or measurable form, but they should allow us to assess whether the system has achieved them.

The primary goal is to observe the early universe, at an age between 1 million and a few billion years. - James Webb

The goal is to continue the acquisition, archival, and distribution of multi-spectral imagery affording global, synoptic, and repetitive coverage of the Earth's land surfaces at a scale where natural and human- induced changes can be detected, differentiated, characterized, and monitored over time. - Landsat Data Continuity Mission


Stakeholder identification
Stakeholder needs
Stakeholder goals
Stakeholder objectives
Constraints
Requirements (further divided later)

Goals become objectives

<u>Objectives</u>: Specific target levels of outputs the system must achieve. Each objective should relate to a particular goal. Generally, objectives should meet four criteria.

1. They should be specific enough to provide clear direction, so developers, customers, and testers will understand them. **They should aim at results and reflect what the system needs to do but not outline how to implement the solution.**
2. They should be **measurable, quantifiable, and verifiable.** The project needs to monitor the system's success in achieving each objective.
3. They should be **aggressive but attainable**, challenging but reachable, and targets need to be realistic. Objectives "To Be Determined" (TBD) may be included until trade studies occur, operations concepts solidify, or technology matures. Objectives need to be feasible before requirements are written and systems designed.
4. They should be results-oriented focusing on desired outputs and outcomes, not on the methods used to achieve the target **(what, not how)**. It is important to always remember that objectives are not requirements. Objectives are identified during pre-Phase A development and help with the eventual formulation of a requirements set, but it is the requirements themselves that are contractually binding and will be verified against the "as-built" system design.

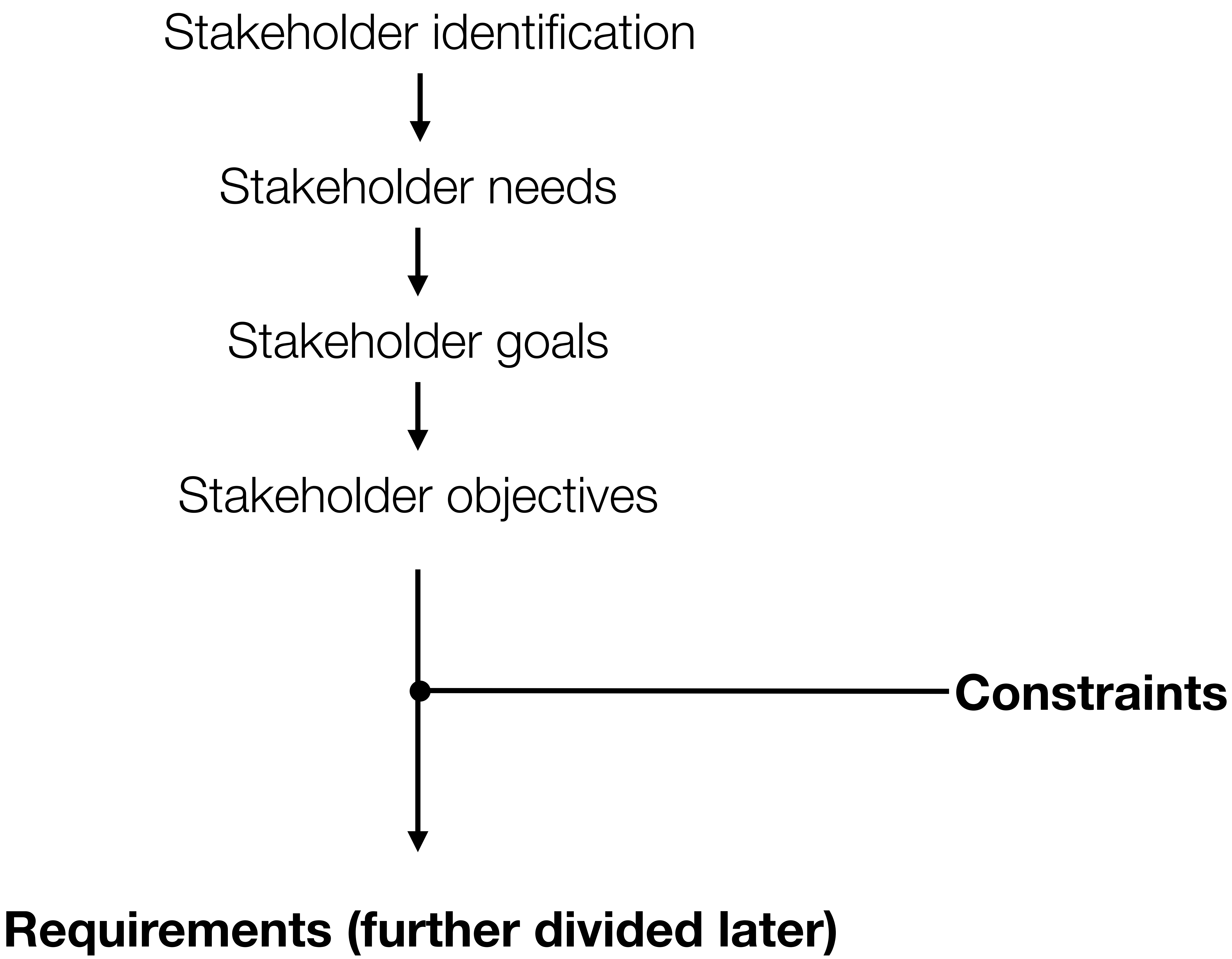
Goals become objectives

- *Collect and archive moderate resolution (circa 30 m ground sample distance) multispectral image data affording seasonal coverage of the global landmass for a continuous period of not less than 5 years.*
- *Ensure that LDCM data are sufficiently consistent with data from the earlier Landsat missions in terms of acquisition geometry, calibration, coverage characteristics, spectral characteristics, output product quality, and data availability to permit studies of land cover and land use change over multi-decadal periods.*
- *Distribute LDCM data products to the general public on a nondiscriminatory basis and at a price no greater than the incremental cost of fulfilling a user request.*

Goals become objectives

- May be somewhat fuzzy/imprecise
- Should specify **what** the system is supposed to do, without specifying **how** the system will do it.
- Requirements are derived from these objectives (and elsewhere) and are not fuzzy at all.

For your projects, you are given objectives. You are tasked with writing requirements based on these objectives and other constraints. Let’s discuss how to do that.


Stakeholder identification
Stakeholder needs
Stakeholder goals
Stakeholder objectives
Constraints
Requirements (further divided later)

What is a requirement?

<u>Requirements</u>: Specify the system in terms of **what the system must accomplish** and **what constraints it must satisfy**. Requirements specify the problem, not the solution. They are:

- Unambiguous
- Concise
- Measurable
- Unique
- Consistent
- Isolated

Requirements definition and concept definition are linked processes and occur simultaneously. It is an iterative process in which vague stakeholder needs are refined into specific requirements.

What is a requirement?

<u>Requirements</u>: Specify the system in terms of **what the system must accomplish** and **what constraints it must satisfy**. Requirements specify the problem, not the solution. They are:

- Unambiguous
- Concise
- Measurable
- Unique
- Consistent
- Isolated

Requirements definition and concept definition are linked processes and occur simultaneously. It is an iterative process in which vague stakeholder needs are refined into specific requirements.

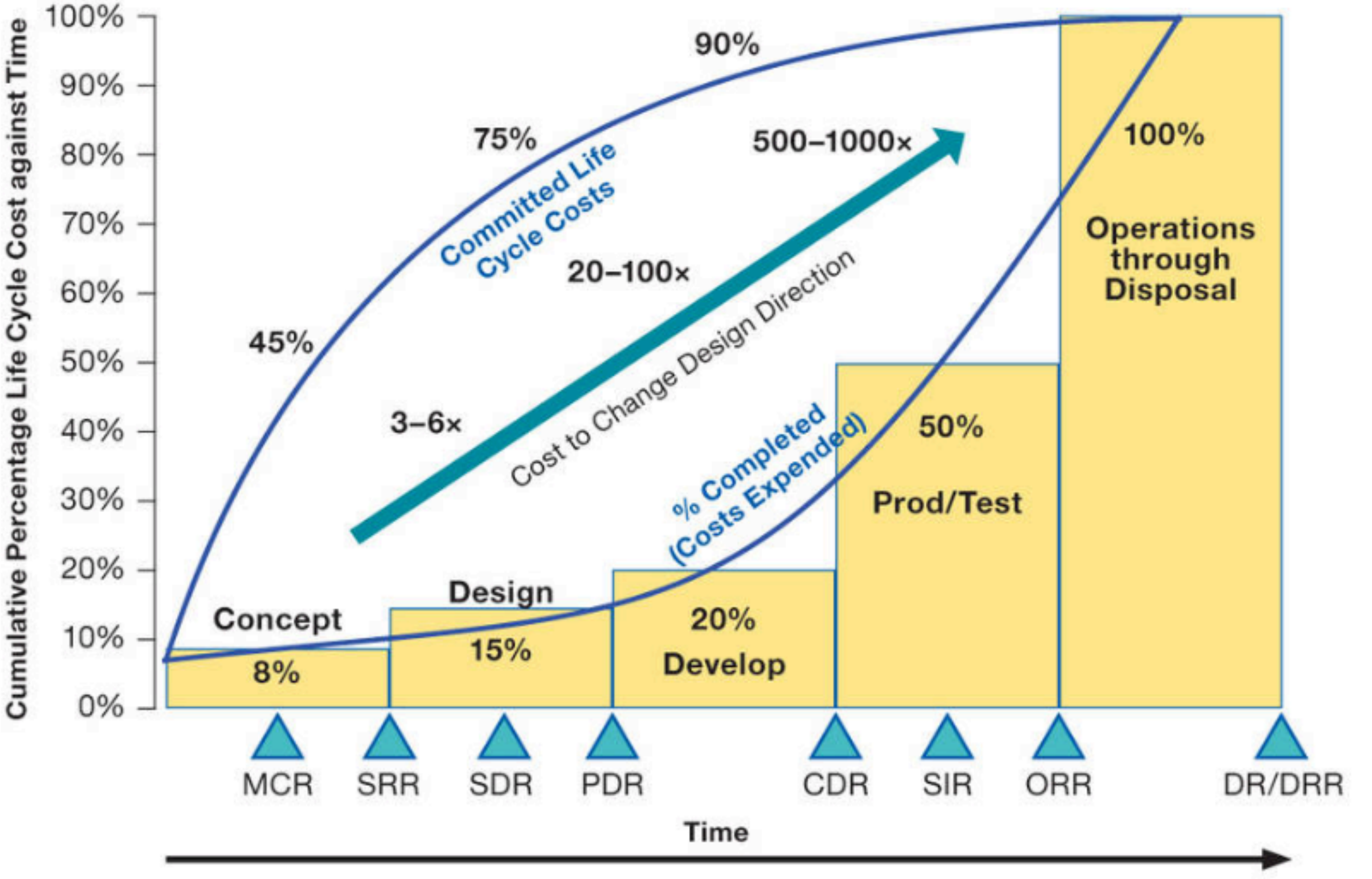
↑

Competing concepts compared via trade studies. More on that later

Why are requirements important?

- Requirements problems are the biggest cause of project problems.
- Requirements define what is to be done, how well, and under what constraints. If you get the requirements wrong, the hardware will be wrong.
- *We cannot solve a problem until we've agreed what the problem is, and how success is measured*.

Why is it important to get requirements right?



MCR	Mission Concept Review	CDR	Critical Design Review
SRR	System Requirements Review	SIR	System Integration Review
SDR	System Definition Review	ORR	Operational Readiness Review
PDR	Preliminary Design Review	DR/DRR	Decommissioning/Disposal Readiness Review

Adapted from INCOSE-TP-2003-002-04, 2015

FIGURE 2.5-1 Life-Cycle Cost Impacts from Early Phase Decision-Making

Where do requirements come from?

High-level requirements

↓

System functional requirements

↓

System performance requirements

↓

Subsystem requirements

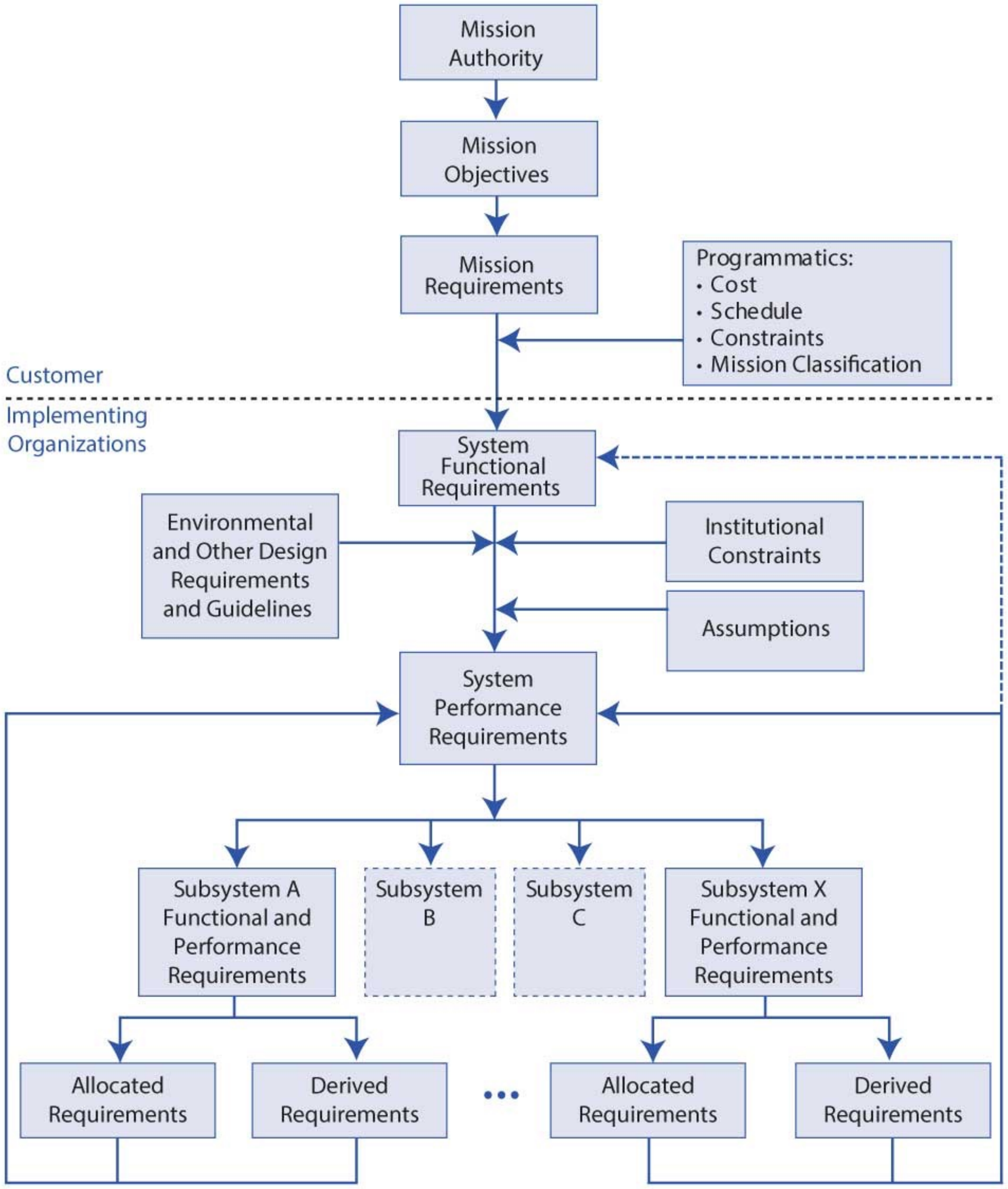


FIGURE 4.2-3 The Flowdown of Requirements

Where do requirements come from?

High-level requirements

↓

System functional requirements

↓

System performance requirements

↓

Subsystem requirements

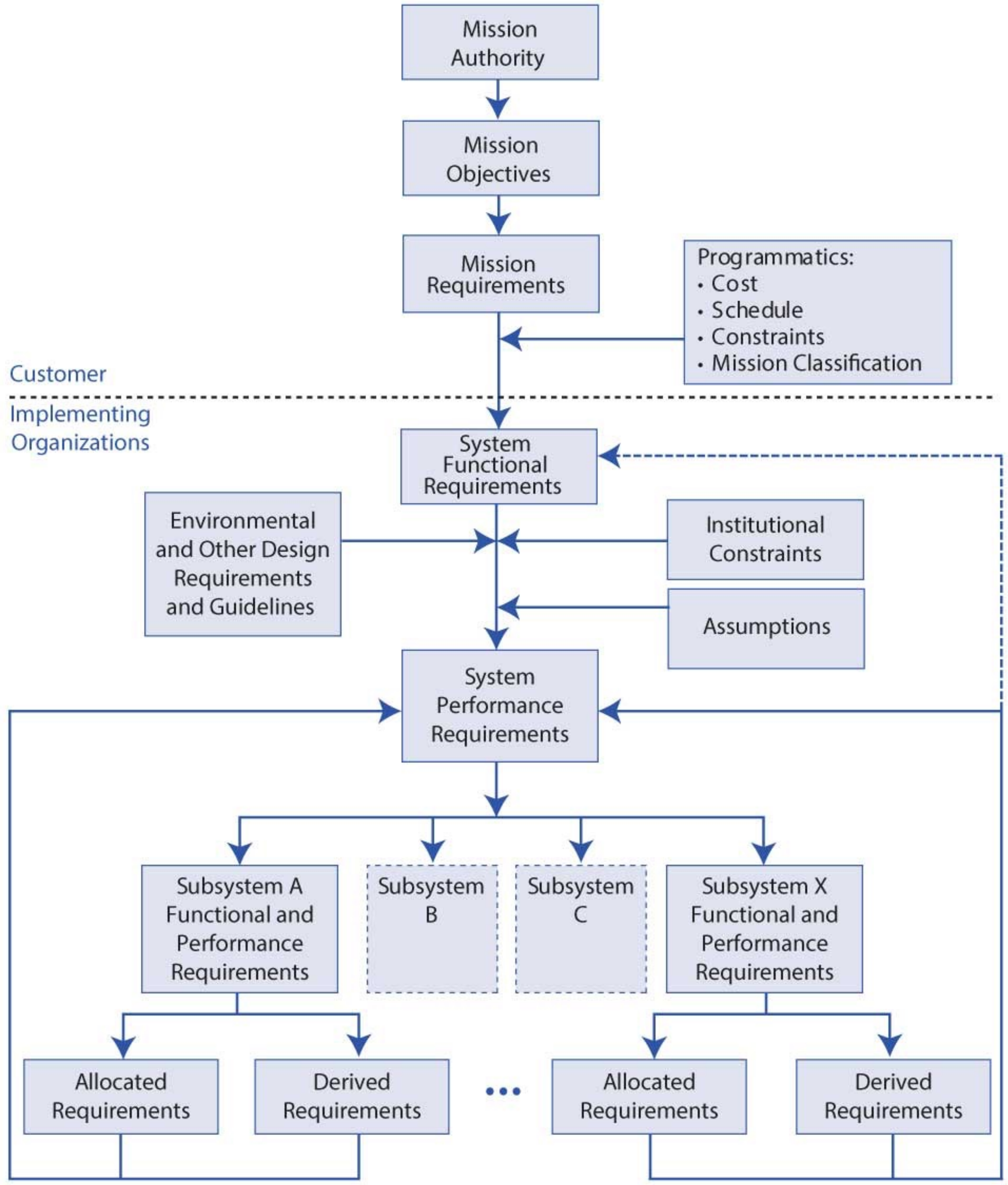


FIGURE 4.2-3 The Flowdown of Requirements

High-level requirements

High-level requirements come from stakeholder needs, concept of operations, constraints, and regulations. They exist in order to understand the technical problem to be solved, the scope of that problem, and the design boundary. Identifying high-level requirements includes:

1. Defining constraints that the design needs to adhere to or that limit how the system will be used. The constraints typically cannot be changed based on trade-off analyses. *T**he system shall cost less than 10M. The JWST wet mass shall not exceed 6,159 kg.***
2. Identifying those elements that are already under design control and cannot be changed. This helps establish those areas where further trades will be made to narrow potential design solutions.
3. Identifying external and enabling systems with which the system should interact and establishing physical and functional interfaces (e.g., mechanical, electrical, thermal, human, etc.). ***The JWST Observatory shall meet the interface requirements to the Launch Segment defined in the Application to Use Ariane (DUA) IRD (JWST-IRD-003674). The operational JWST shall utilize the Deep Space Network.***
4. Defining functional and behavioral expectations for the range of anticipated uses of the system as identified in the ConOps. The ConOps describes how the system will be operated and the possible use-case scenarios. ***The JWST shall orbit the second Lagrange point (L2) of the Sun-Earth system. The operational JWST System shall have at least one two-way communication contact between the Observatory and Ground Segment in a 24 hour period.***

High-level requirements

High-level requirements come from stakeholder needs, concept of operations, constraints, and regulations. They exist in order to understand the technical problem to be solved, the scope of that problem, and the design boundary. Identifying high-level requirements includes:

Examples of places to look for high-level requirements:

- NPR 8020.12C “Planetary protection provisions for robotic extraterrestrial missions”
- NPR 8705.2A “Human-rating requirements for space systems”
- Launch vehicle payload/user requirements
- Standards-based requirements
- Regulations (e.g. FCC)
- System boundaries and external interfaces (docking with ISS?)

High-level requirements

High-level requirements come from stakeholder needs, concept of operations, constraints, and regulations. They exist in order to understand the technical problem to be solved, the scope of that problem, and the design boundary. Identifying high-level requirements includes:

Examples of places to look for high-level requirements:

A brief aside on types of requirements

- NPR 8020.12C “Planetary protection provisions for robotic extraterrestrial missions”
- NPR 8705.2A “Human-rating requirements for space systems”
- Launch vehicle payload/user requirements
- Standards-based requirements
- Regulations (e.g. FCC)
- System boundaries and external interfaces (docking with ISS?)

Functional requirements

Functional requirements define **what** functions need to be performed to accomplish the objectives.

- *The Thrust Vector Controller shall provide vehicle control about the pitch and yaw axes.*

Performance requirements

Performance requirements define **how well** the system needs to perform the functions. These are generally derived from system/subsystem level functional requirements.

- *The TVC shall gimbal the engine a maximum of 9 degrees, ± 0.1 degree.*
- *The TVC shall gimbal the engine at a maximum rate of 5 degrees/second ± 0.3 degrees/second.*
- *The TVC shall provide a force of 40,000 pounds, ± 500 pounds.*
- *The TVC shall have a frequency response of 20 Hz, ± 0.1 Hz*

Interface requirements

Requirements that specify the functional or structural interfaces among subsystems.

- *The power subsystem shall provide 12V DC at up to 1.5 A to the payload.*
- *The launch-vehicle upper stage shall provide the spacecraft with positive detection of separation via a +5 V signal on pin 8 of the umbilical connector.*

Customer requirements

These will include product expectations, mission objectives, operational concerns, and/or measures of effectivity and suitability. It may require careful analysis to extract functions, and success criteria are generally provided.

- *The rock-abrasion tool shall be capable of grinding away some part of the surface of any rock encountered on the surface of Mars to permit scientists to analyze a portion that has not been exposed to weathering.*

Design requirements

These are requirements derived from process specifications (e.g. MIL STDs), or internal best practices (tolerances, trade-secret guidelines, design for manufacturability, etc.). These are often associated with "design for X."

- *All control loops shall demonstrate at least 20 deg. phase margin by analysis.*
- *Mechanisms shall be designed with torque margin in compliance with MIL-STD 1540-D.*

Verification requirements

Requirements that specify the way in which verification must proceed—test requirements, analysis methodologies, etc. (We'll go over verification in some detail a bit later).

- *The rover wheel-bearing life tests shall include no accelerated component testing.*
- *Corrosion testing shall be performed within the temperature range 10-120 deg. C.*

Where do requirements come from?

High-level requirements

↓

System functional requirements

↓

System performance requirements

↓

Subsystem requirements

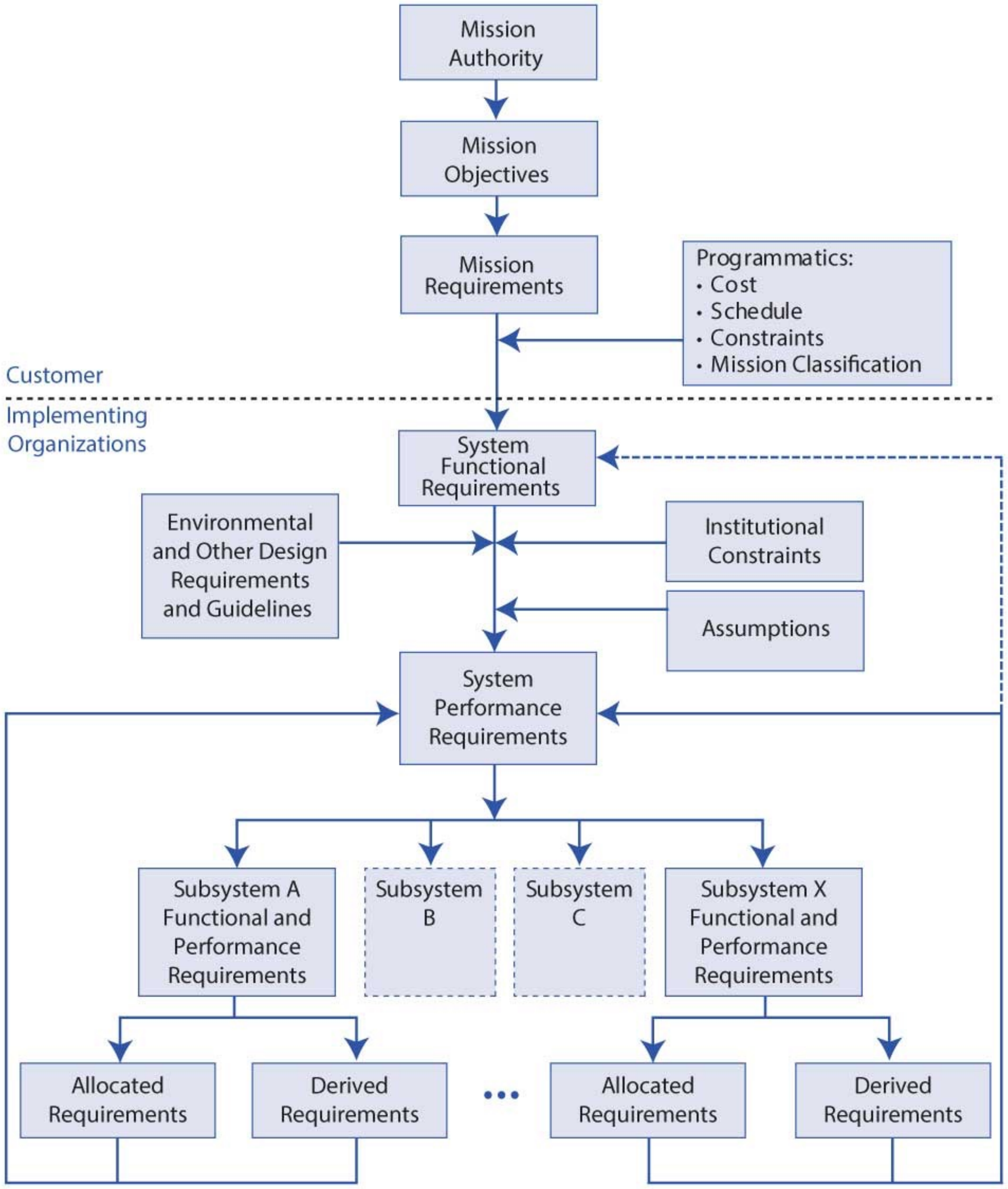


FIGURE 4.2-3 The Flowdown of Requirements

Where do requirements come from?

High-level requirements

↓

System functional requirements

↓

System performance requirements

↓

Subsystem requirements

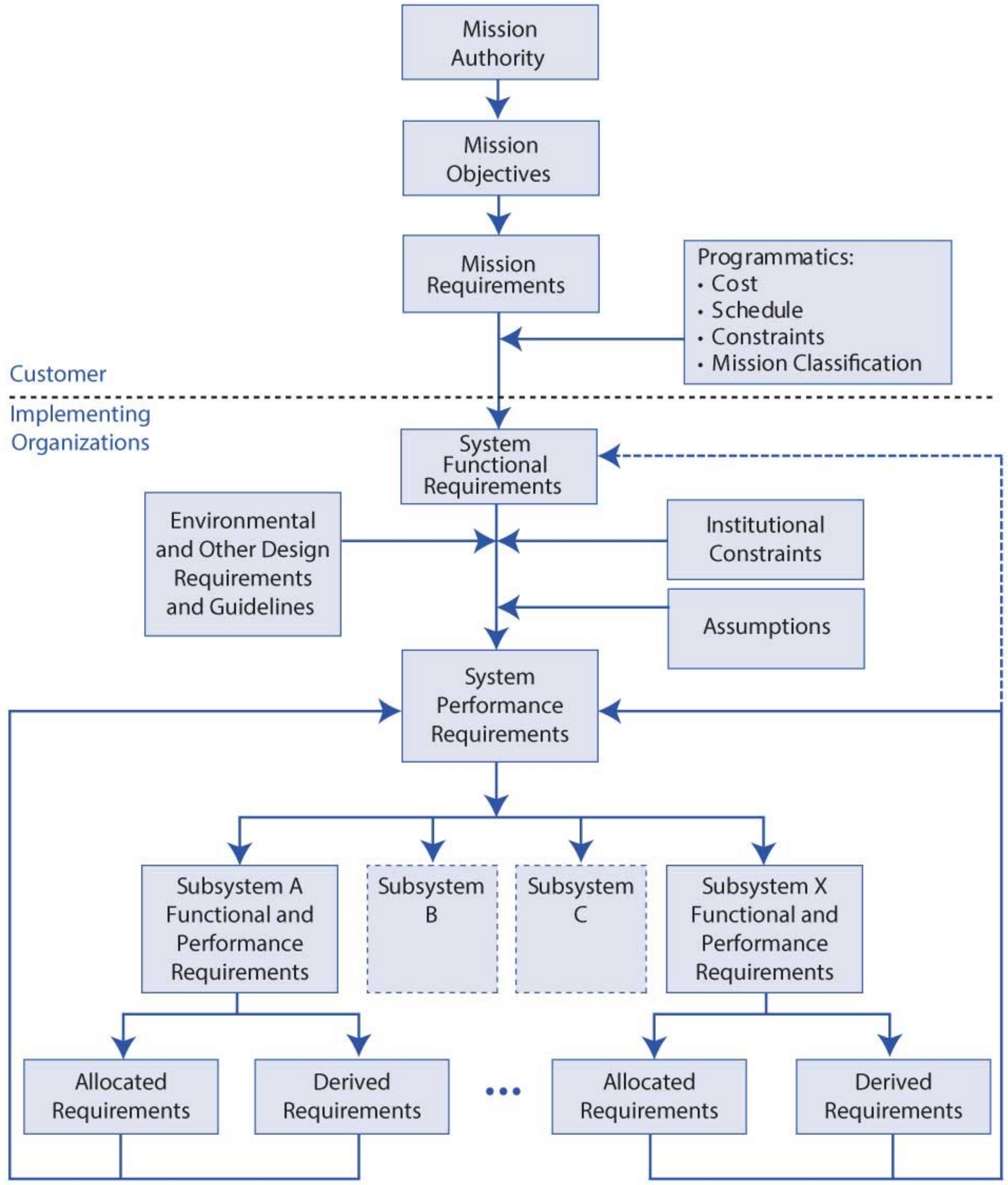


FIGURE 4.2-3 The Flowdown of Requirements

Where do requirements come from?

High-level requirements

↓

System functional requirements

↓

System performance requirements

↓ **Flowdown, Allocation, Derivation**

Subsystem requirements

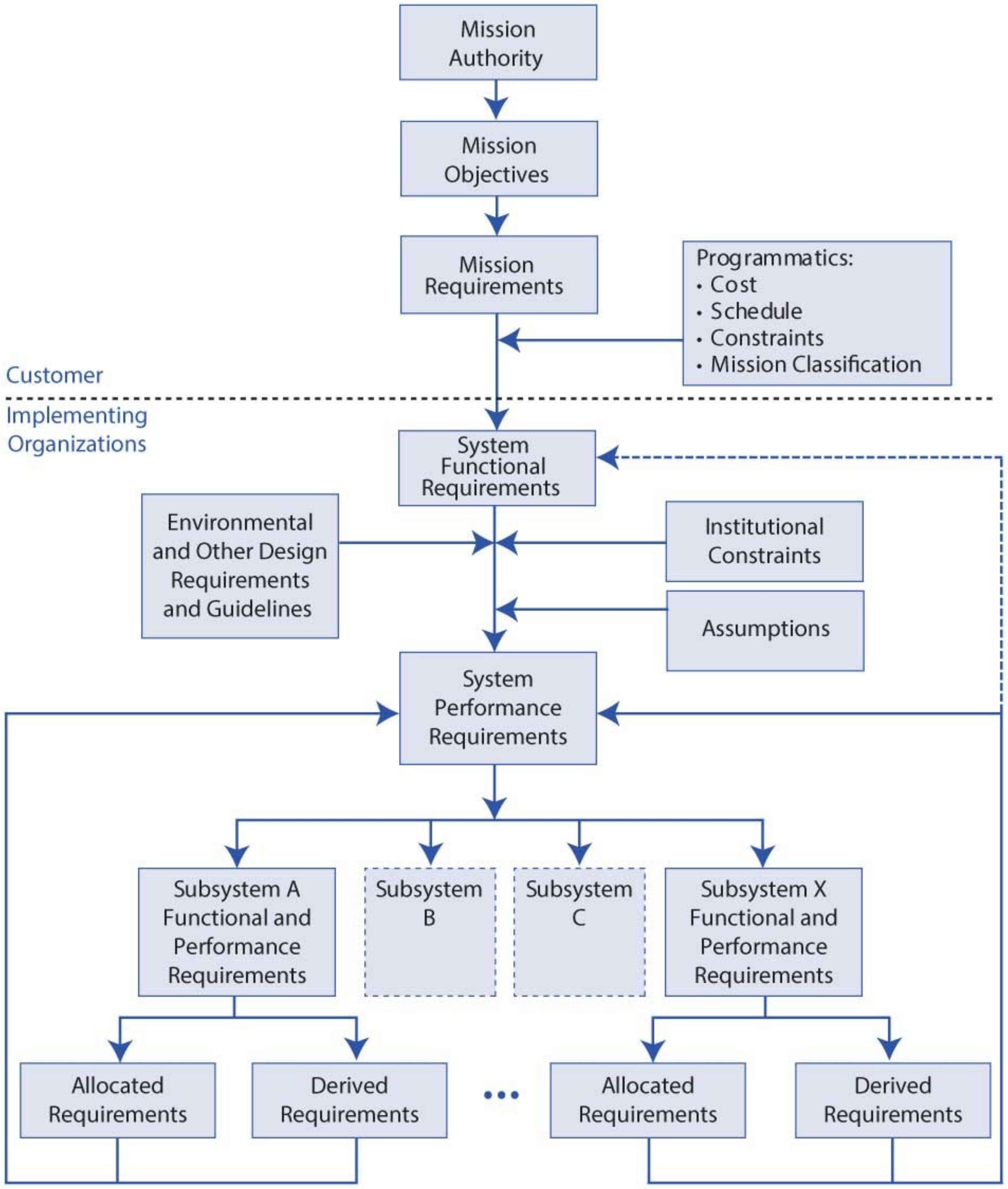


FIGURE 4.2-3 The Flowdown of Requirements

Flowdown requirements

A direct transfer of a requirement from the system-level to a subsystem-level, since a subsystem provides that capability.

- E.g., Requirements for spacecraft communications may be entirely flowed-down from the spacecraft system requirements to the spacecraft communications subsystem requirements

Allocated requirements

Allocation is a quantitative apportionment from a higher level to a lower level and for which the unit of measure remains the same. Examples include mass, power, or pointing.

- E.g., A 1,000 kg spacecraft may allocate 200 kg, 500 kg, and 300 kg to three separate subsystems.

Derived requirements

Derived requirements are apportionments that depend on a specific implementation.

- SYS 1.4 “The launch vehicle shall be capable of taking off vertically from any NATO aircraft carrier.”
- STRUCT 1.0 “The launch vehicle gross lift-off weight shall be less than 10,000 kg.” [SYS 1.4]
- ENG 1.0 “The sea-level standard-day thrust shall be no less than 209,000 N.” [SYS 1.4]

Crew Exploration Vehicle (CEV) Requirements Distribution, example

- **System Performance Requirements Document**
  - The CEV System shall provide two-way voice communications during crewed operations.
- **Flight Segment Performance Requirements Document**
  - The flight segment shall provide voice communications to the ground through TDRSS.
  - Other derived requirements.
- **Flight Vehicle Contract End Item Specification Part I**
  - The flight vehicle shall provide four omni directional antennas.
  - The flight vehicle shall provide two S-Band transponders.
  - The flight vehicle shall provide a switch matrix to allow connection of each S-Band transponder with each omni directional antenna.
  - Other derived requirements.
- **Flight Vehicle Contract End Item Specification Part II**
  - The communications subsystem shall provide the equipment specified in drawing CEVFV-COM1-234.
  - The communications subsystem shall be wired as shown in drawing CEVFV-COM2-234.
  - The communications subsystem equipment shall be mounted on the avionics pallets as shown in drawing CEVFV-COM2-235.
  - Other derived requirements.

System Functional Performance Requirements

Segment Functional Performance Requirements

Element Design Requirements

Element Fabrication Requirements

How do we write requirements?

Writing Requirements

Preferred verb: “Shall”

- Anything else (should, ought) implies a soft requirement, one to which the system will not be held during the verification phases

Good requirements are **valid**. Validity implies

- Unambiguous
- Isolated
- Concise
- Measurable
- Unique
- Consistent

Writing Requirements

Grammar establishes the flow of requirements.

- Single sentence
- The subject is a system, element, subsystem, component, etc., which establishes the functional level at which the requirement is relevant
- The vert often implies the type of verification (test, inspect, analyze, etc.)
- The object of the verb is often a **Technical Performance Measure (TPM)**

Which is good? Which is bad?

- The rover drive system shall weigh less than 5 kg.
- The weight of the rover harness shall be less than 1 kg.

Writing Requirements

Unambiguous

- Unambiguous requirements are free of words and phrases such as “reasonable,” “acceptable,” “minimize,” and “where applicable.”
- Unambiguous requirements are not a matter of opinion, and cannot be misinterpreted.
- Quantitative requirements are often unambiguous, but qualitative ones can also be valid.

Which is good? Which is bad?

- The rover shall be very fast.
- The rover shall be capable of collecting 3 surface samples in less than 75 minutes.

Writing Requirements

Isolated

- Each “shall" statement belongs in a separate, unique requirement (no conjunctions)
- Constraining each paragraph to contain no more than one “shall” allows one to take full advantage of the viewing, reporting, and traceability functions of requirements-management tools
- Isolation allows full traceability, discrete referencing, and one-to-one verification cross-referencing.

Which is good? Which is bad?

- The rover shall weigh less than 10 kg and shall operate when tethered to a 28V nuclear power source.
- The rover shall weigh less than 10 kg.

Writing Requirements

Measurable

- Each requirement will be verified (by test, analysis, inspection, etc.)
- If the requirement cannot be verified, it cannot be tested.
- A measurable requirement is the only type that can be verified.

Which is good? Which is bad?

- The rover shall be robust to failures.
- The rover shall be single-fault tolerant.

Writing Requirements

Concise

- Do not include explanations, definitions, or other information unrelated to the specification; use a glossary, a list of acronyms, etc. in the documentation instead.

Which is good? Which is bad?

- The rover shall be able to work in the dark, e.e. 0.2 cd (where “cd” refers to candela, one lumen per steradian)
- The rover shall be capable of operating at light levels below 0.2cd

Writing Requirements

Unique

- It is easy in long documents created by teams of people to identify the same requirement multiple times in slightly different forms
- The work to be done is deciding which version of the requirement to retain and which to delete

Which one should we keep?

- The rover's rubber seals shall function within the range 10-120 deg. C [Europa science spec]
- The rover's rubber seals shall function after having been exposed to temperatures in the range 15-50 deg. C [Launch vehicle ICD]

Writing Requirements

Consistency

- Incorrect slowdown, or slowdown from different high-level requirements may lead to similar specs that differ quantitatively
- The requirements are invalid until the inconsistency is resolved

Example of an inconsistency

- The rover’s rubber seals shall function within the range 10-120 deg. C [Europa science spec]
- The rover's rubber seals shall function after having been exposed to temperatures in the range -20-50 deg C [Launch vehicle ICD]

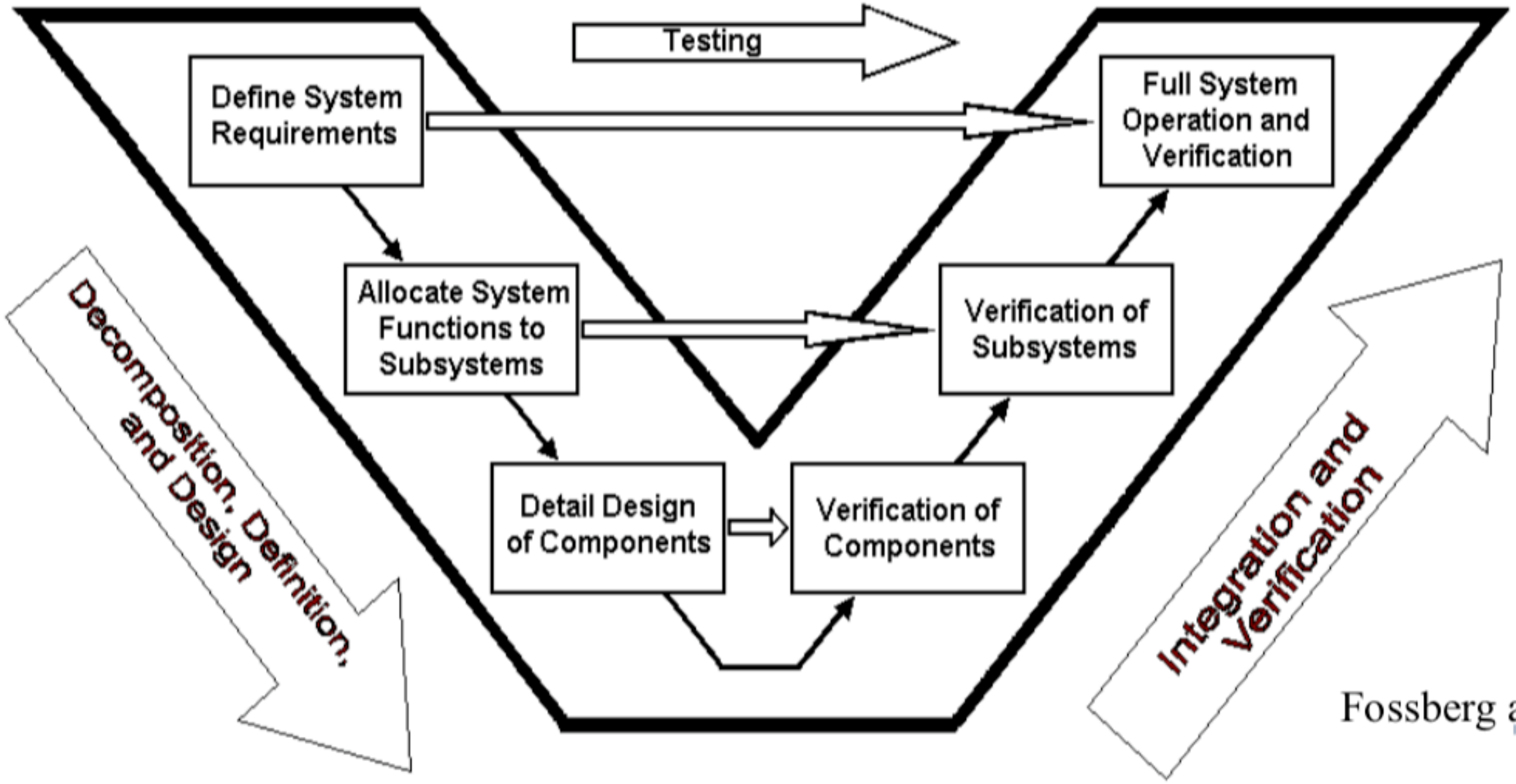
How do we manage requirements?

- Requirements must be **traceable** (which are the parent requirements? which are the children?)
- Traceability can be maintained through a **verification cross-reference matrix**

TABLE 4.2-2 Requirements Metadata

Item	Function
Requirement ID	Provides a unique numbering system for sorting and tracking.
Rationale	Provides additional information to help clarify the intent of the requirements at the time they were written. (See "Rationale" box below on what should be captured.)
Traced from	Captures the bidirectional traceability between parent requirements and lower level (derived) requirements and the relationships between requirements.
Owner	Person or group responsible for writing, managing, and/or approving changes to this requirement.
Verification method	Captures the method of verification (test, inspection, analysis, demonstration) and should be determined as the requirements are developed.
Verification lead	Person or group assigned responsibility for verifying the requirement.
Verification level	Specifies the level in the hierarchy at which the requirements will be verified (e.g., system, subsystem, element).

How do we verify requirements?

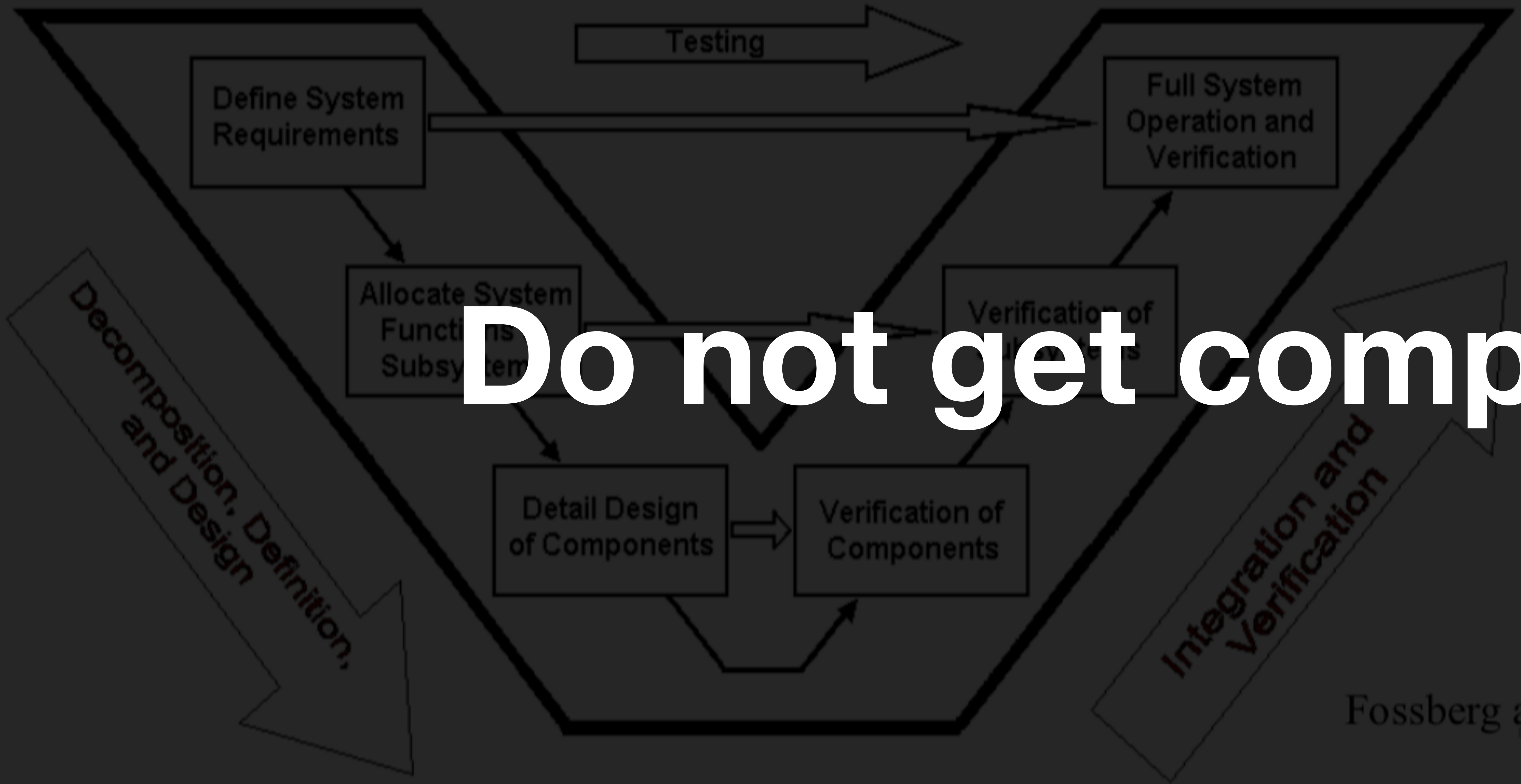


Fossberg and Mooz

Verify via:

- Test
- Inspection
- Analysis
- Demonstration
- Similarity

How do we verify requirements?



Fossberg and Mooz

Verify via:

- Test
- Inspection
- Analysis
- Demonstration
- Similarity

Do not get complacent.

Trade Studies

- Many alternatives, including the **Pugh Matrix**

Criteria	Reference	Alternative 1	Alternative 2	Alternative 3
Cost	0	+	+	-
Coverage	0	0	0	-
Latency	0	+	+	+
Avg Revisit time	0	+	+	0
Max Revisit time	0	+	0	0
Spatial resolution	0	0	-	0
Risk	0	+	0	-

What alternative should be selected?

- Define a set of alternatives
- Define a set of selection criteria (cost, risk, complexity, mass, etc.)
- Choose a reference option
- Compare all options with the reference option across all criteria. Place a + if the option is better than reference, 0 if same, - if worse
- Compute the score for each alternative (score = # '+' * '-' # . . .), adding weights if necessary
- Choose the alternative with the highest score
- Eliminate poor alternatives, add new ones, eliminate non-distinguishing criteria, iterate.

Table 4. Science Value Matrix

Science value for architectural options - Ratings approach: 0 = "Architecture does not address science objective", 10= "Architecture completely fulfills objective". "Completely" is subjective, particularly since very few areas of research are ever complete	Relative Category Science Value	Goal Science Value Relative in Category	Cassini (for reference)	Saturn orbiter with E high speed flybys	Simple Enceladus Orbiter	Simple Enceladus Orbiter (shorter ops)	Enhanced Enceladus Orbiter	Simple Enceladus Orbiter (2a) + freeflying magnetometer	Simple Enceladus Orbiter (2a) + semi-hard seismic network	Simple Enceladus Orbiter (2a) + impactor	High Performance orbiter	High Performance orbiter + semi-hard seismic network	High Performance orbiter + instrumented lander/hopper	Titan-Enceladus Connection	Sample plume from Saturn orbit (nuclear), ~2 km/s sampling velocity, 250K samples	Sample plume from Saturn orbit (solar), ~2 km/s sampling velocity, 250K samples	Sample plume from Saturn orbit, ~4 km/s sampling velocity, no temperature control	5a with enhanced payload (2a w/no laser altimeter)
Study option designator			0	1a	2a	2b	2c	2d	2e	2f	3a	3b	3c	4a	5a	5b	5c	5d
Nature of Enceladus; cryovolcanic activity	6		2.8	4.7	5.3	4.9	6.5	5.2	5.6	5.6	6.9	7.2	7.7	6.4	6.9	6.9	5.9	7.7
Physical conditions at the plume source		4	2.5	4.8	4.7	4.5	5.9	4.7	5.2	5.5	6.2	6.3	6.8	5.9	6.3	6.3	4.9	7.5
Chemistry of the plume source		4	2.7	5.2	5.9	5.6	7.5	5.9	5.9	6.1	7.7	7.7	8.3	7.3	8.5	8.5	7.2	9.0
Presence of biological activity		1	0.0	2.4	3.0	2.8	5.3	3.0	3.0	3.9	5.1	5.1	6.4	4.3	8.3	8.3	6.7	8.3
Plume dynamics and mass loss rates		2	4.5	5.0	5.5	4.8	6.3	5.5	5.5	5.5	7.5	7.5	8.0	6.3	6.7	6.7	6.3	7.0
Origin of south polar surface features		2	3.0	4.3	6.0	5.3	6.7	5.7	7.0	6.0	7.3	8.3	9.0	6.7	4.3	4.3	4.3	6.0
Internal structure and chemistry of Enceladus	4		2.4	4.0	5.5	4.8	6.9	6.9	7.2	6.2	7.2	8.3	8.2	6.7	4.8	4.8	4.5	6.0
Internal structure		3	1.0	2.0	4.7	4.0	6.7	7.0	7.7	6.0	7.0	8.7	7.3	6.7	2.3	2.3	2.3	3.0
Presence, physics, and chemistry of the ocean		4	2.8	4.6	6.1	5.0	7.8	7.4	7.6	6.9	7.8	8.8	9.0	7.3	7.0	7.0	6.5	7.6
Tidal dissipation rates and mechanisms		3	3.0	4.3	5.3	5.0	6.0	6.7	7.3	5.7	6.3	8.0	7.7	5.7	2.3	2.3	2.3	5.0
Chemical clues to Enceladus' origin and evolution		2	2.9	5.2	5.6	5.4	6.8	5.9	5.6	5.6	7.4	7.4	8.5	6.9	7.8	7.8	7.1	8.8
Geology of Enceladus	3		3.0	4.7	5.7	5.0	7.3	5.7	6.7	6.3	7.7	8.7	9.0	7.0	3.0	3.0	3.0	4.7
Nature, origin and history of geological features		4	3.0	4.7	5.7	5.0	7.3	5.7	6.7	6.3	7.7	8.7	9.0	7.0	3.0	3.0	3.0	4.7
System Interaction	2		3.8	3.5	3.9	3.4	5.7	4.3	3.9	4.1	5.9	5.9	6.0	5.7	4.3	4.3	4.1	4.9
Plasma and neutral clouds		4	4.0	2.3	2.7	1.7	5.7	3.7	2.7	3.3	6.0	6.0	6.0	5.7	1.7	1.7	1.7	2.3
E-ring		4	4.0	4.7	4.7	4.7	5.3	4.7	4.7	4.5	5.5	5.5	5.5	5.3	7.0	7.0	6.7	7.0
satellites		2	3.0	3.7	4.7	4.3	6.3	4.7	5.0	4.8	6.3	6.5	6.8	6.7	4.3	4.3	4.0	5.7
Other satellite science	2		3.0	2.3	3.0	3.0	4.5	3.0	3.0	3.0	5.0	5.0	5.0	6.8	0.3	0.3	0.2	0.7
Nature of Titan's geological processes		4	3.0	1.3	1.3	1.3	3.3	1.3	1.3	1.3	4.7	4.7	4.7	8.7	0.0	0.0	0.0	0.0
Surfaces and interiors of Rhea, Dione, and Tethys		4	3.0	3.3	4.7	4.7	5.7	4.7	4.7	4.7	5.3	5.3	5.3	5.0	0.7	0.7	0.3	1.3
Preparation for follow-on missions	1		2.0	3.0	5.0	4.7	6.7	5.0	6.7	7.0	7.3	8.0	9.0	6.3	2.7	2.7	2.7	4.3
Nature of potential landing sides		4	2.0	3.0	5.0	4.7	6.7	5.0	6.7	7.0	7.3	8.0	9.0	6.3	2.7	2.7	2.7	4.3
Category value by Architecture, summed			17.0	22.2	28.3	25.7	37.6	30.0	33.0	32.2	39.9	43.0	44.9	39.0	22.0	22.0	20.3	28.2
Category Value-weighted, summed, normalized			0.85	1.21	1.49	1.35	1.93	1.59	1.71	1.64	2.04	2.20	2.28	1.96	1.36	1.36	1.23	1.66
Normalized to Reference Architecture			1.00	1.44	1.76	1.60	2.28	1.88	2.03	1.94	2.41	2.60	2.70	2.32	1.61	1.61	1.45	1.97
Sum					Key=	Low	Mid	High										

Risk

Risk: A measure of the probability and severity of adverse effects.

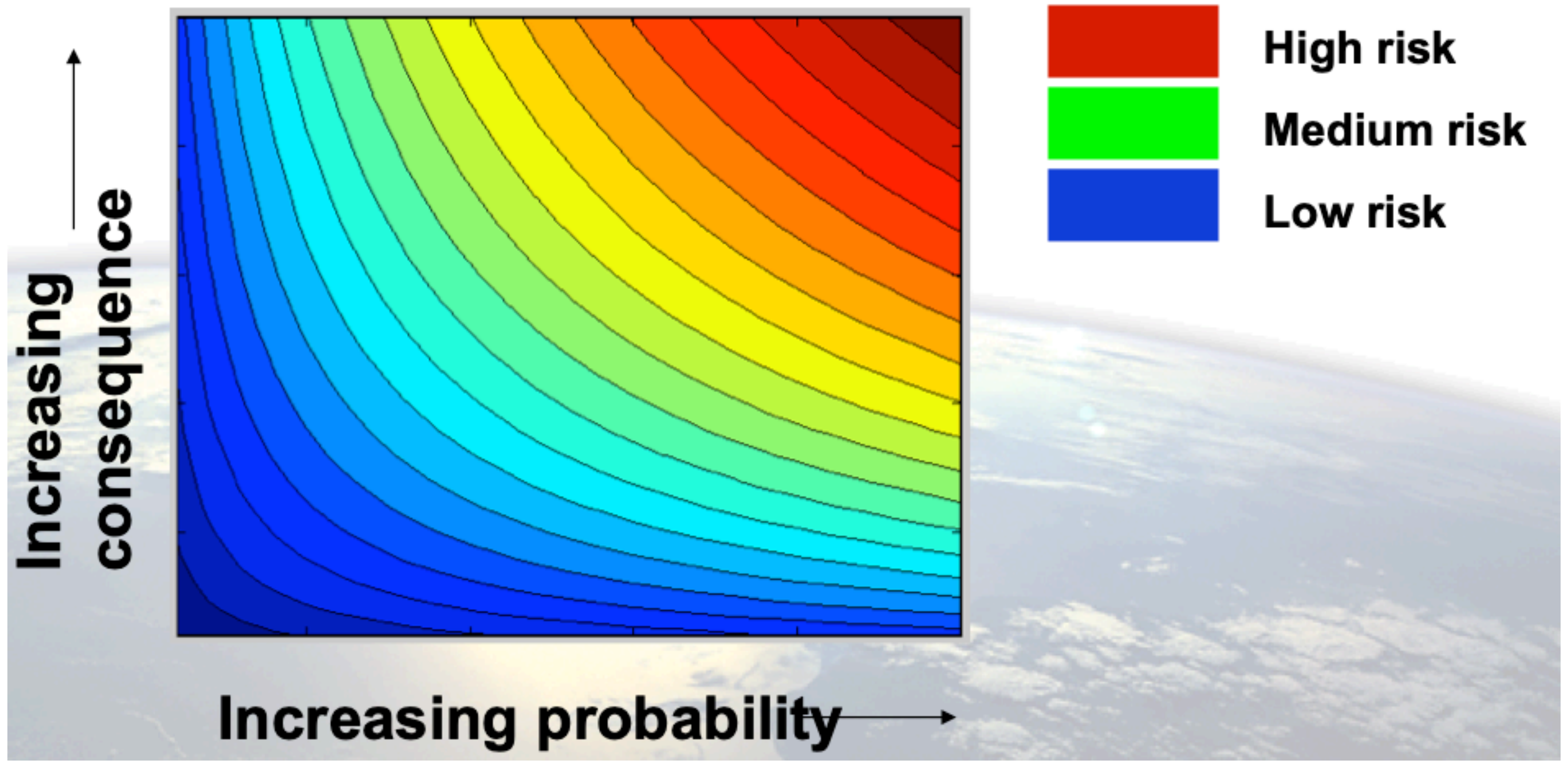
Reliability: The ability of a system or component to perform its required functions under the stated conditions for a specified period of time.

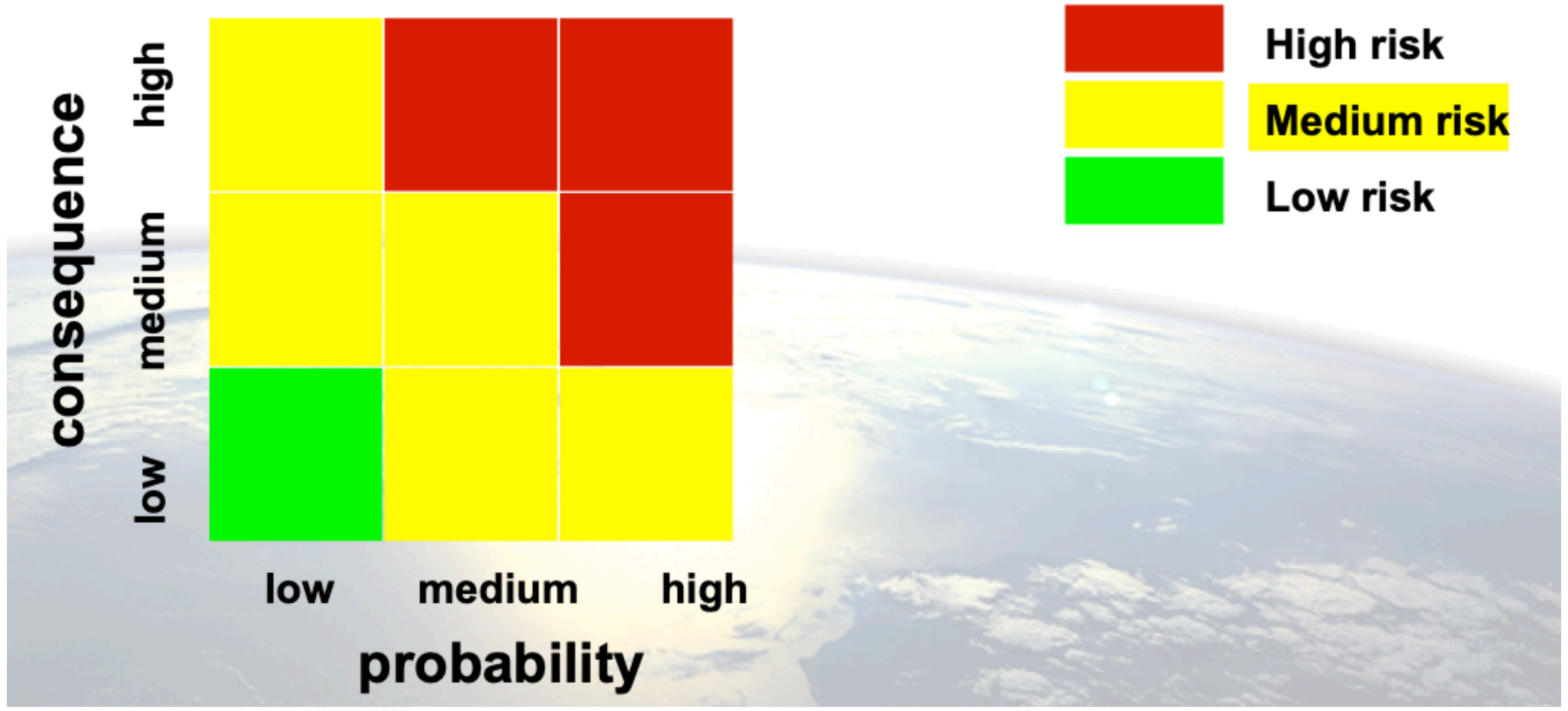
Opportunity: A measure of the probability and the benefit of beneficial effects.

When you're doing risk analysis, you're asking yourself the following questions:

- What can go wrong?
- What it the likelihood that it would go wrong?
- What are the consequences if it goes wrong?

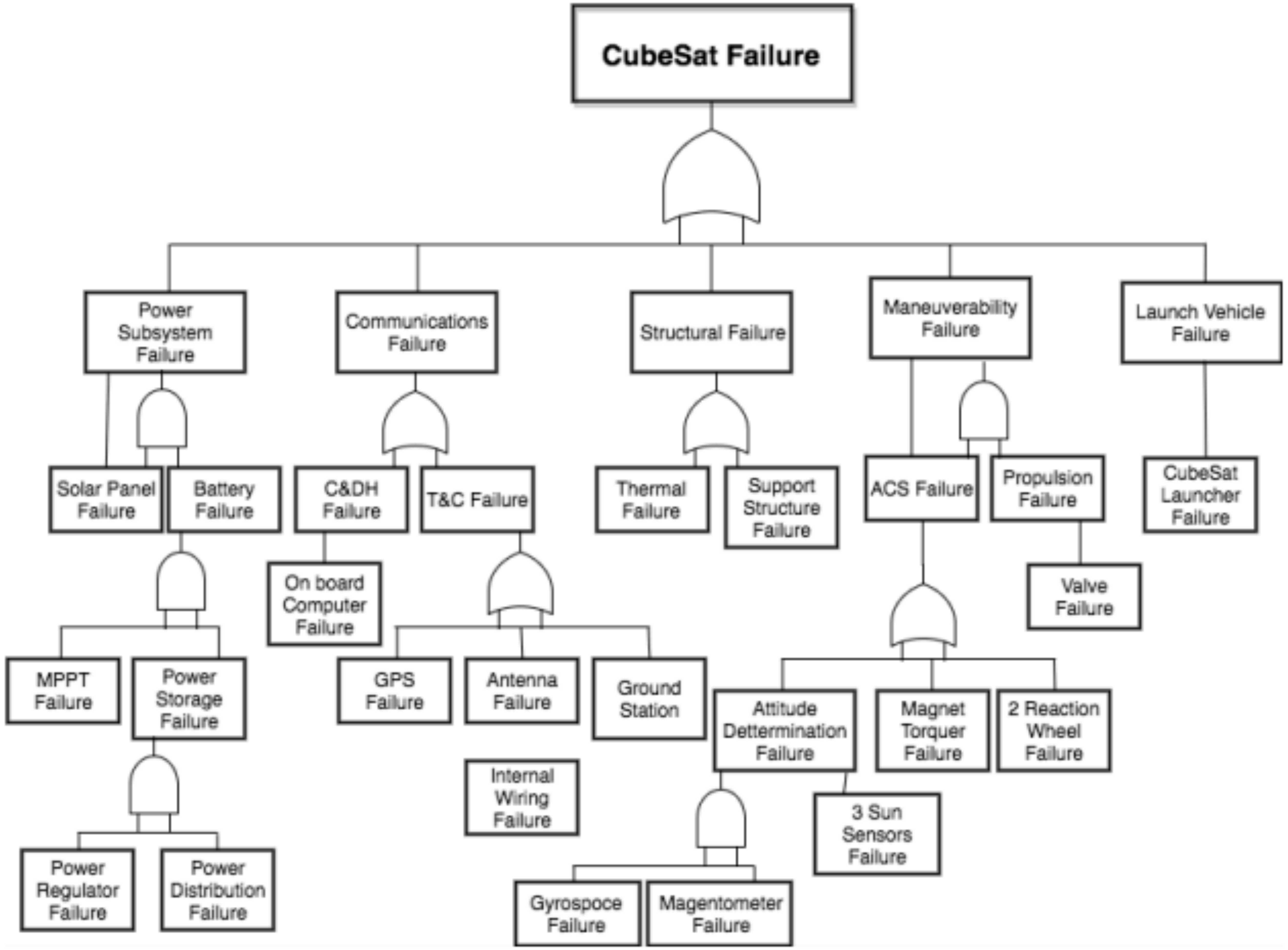
Risk





Stoplight charts provide a systematic means of classifying the risk of various options relative to one another.

Risk



Potential faults are organized into fault trees.