16.842 Fundamentals of Systems Engineering

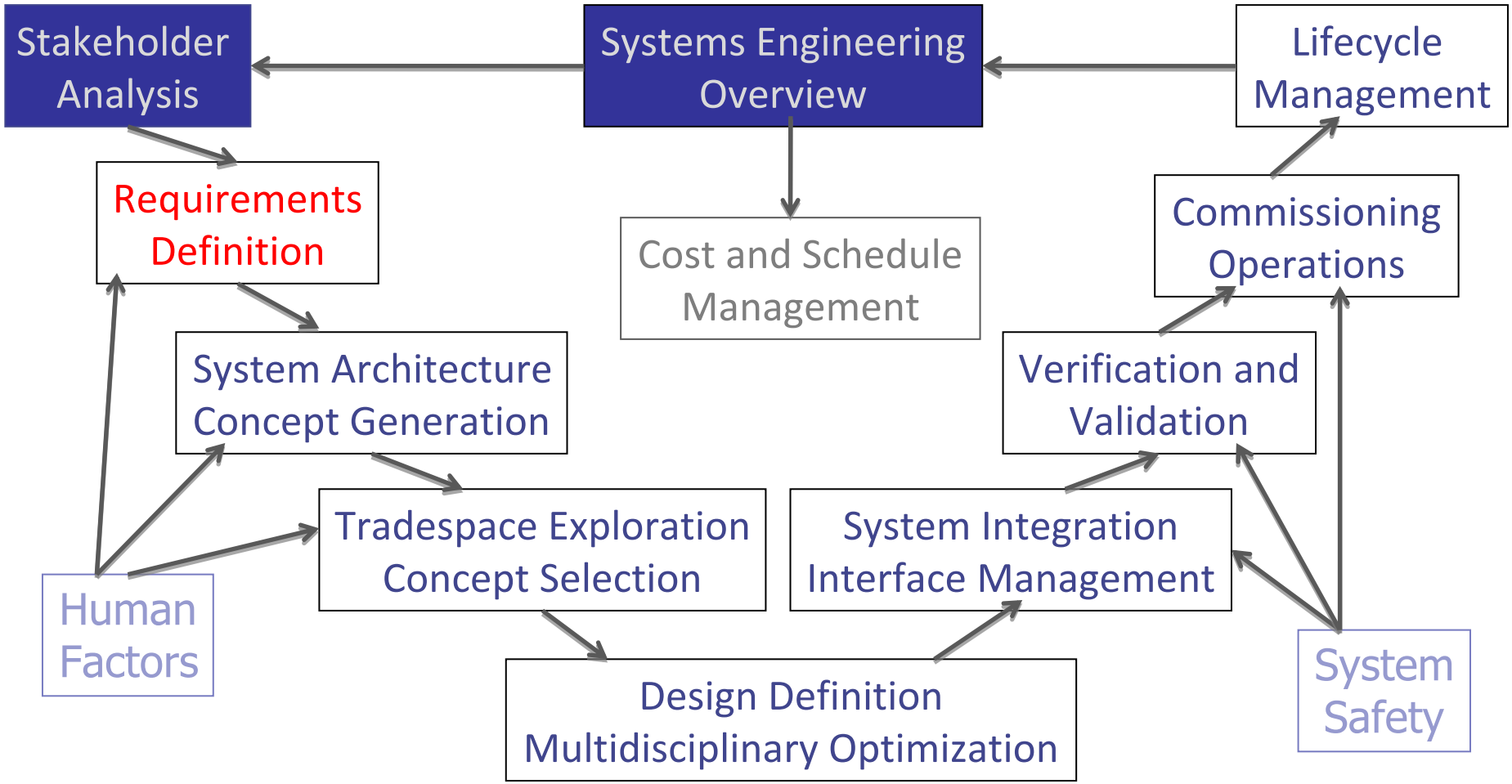
Session 3 Fall 2009

Requirements Definition

How we should (attempt to) specify exactly what is needed before we start designing

Prof. Olivier de Weck

V-Model – Sept 25, 2009



Overview

- What are requirements?
  - Definition, Example, Evolution, Standards
- NASA Requirements Process
- Challenges of Requirements Definition
  - Flowdown and Allocation
  - Validation and Verification
  - Writing good requirements

Requirements Definition

- Requirements describe the necessary functions and features of the system we are to conceive, design, implement and operate.
  - Performance
  - Schedule
  - Cost
  - Other Characteristics (e.g. lifecycle properties)
- Requirements are often organized hierarchically
  - At a high level requirements focus on what should be achieved, not how to achieve it
  - Requirements are specified at every level, from the overall system to each hardware and software component.
- Critically important to establish properly

DC-3 Requirements

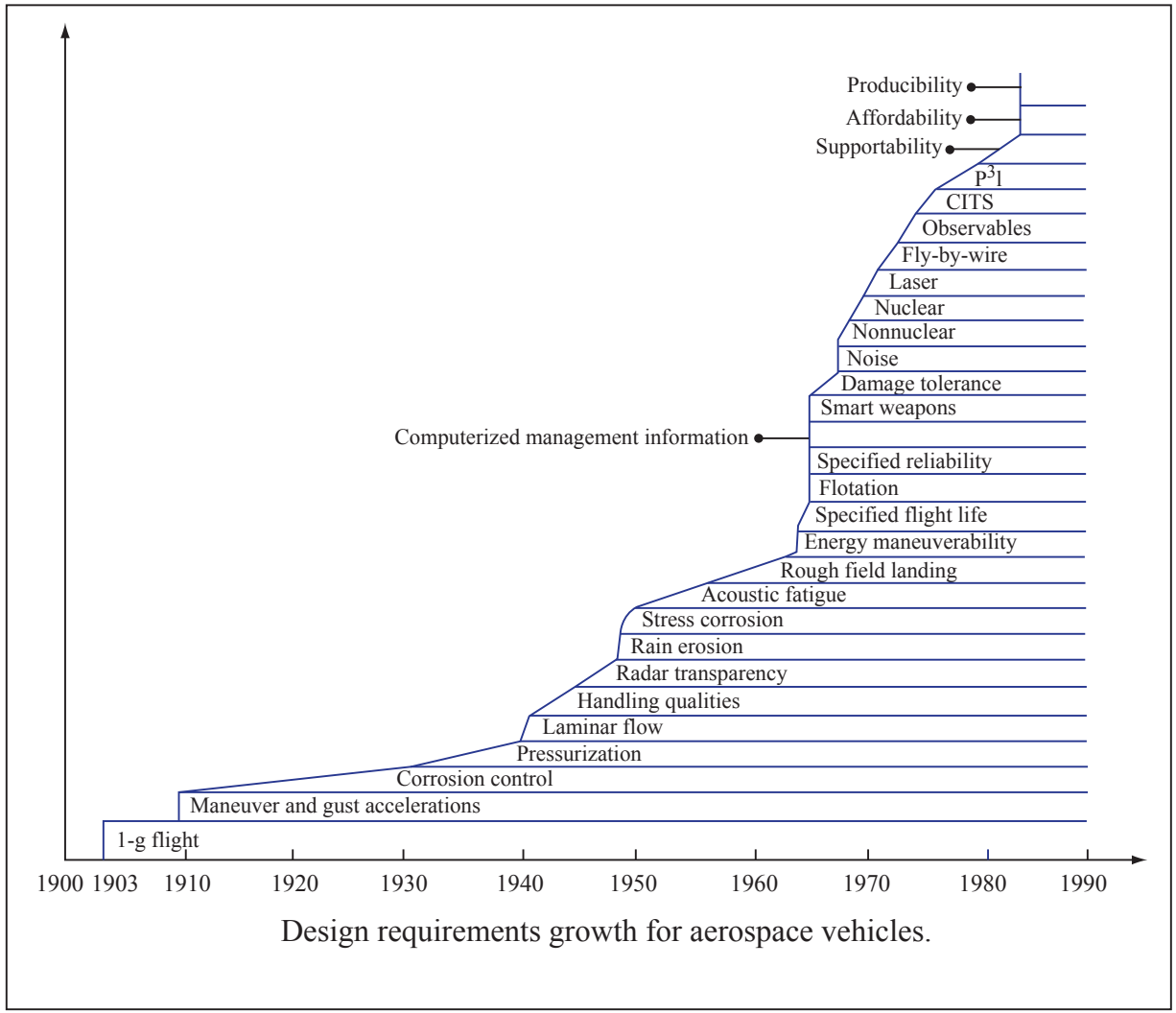
1st flight: 17 Dec 1935
Over 10,000 built

- Requirements based on desired improvements to DC-2
- Very simple
  - 3 page RfP (McDonnell Museum)
  - "Marathon" phone call between Smith and Douglas
- Key Requirements
  - Range: 1000 miles
  - Cruise Speed: 150 mph
  - Passengers: 20-30
    - Depending on configuration
  - Twin Engines
  - "Rugged and Economical"

Image by jfhweb on Flickr.

Requirements Explosion

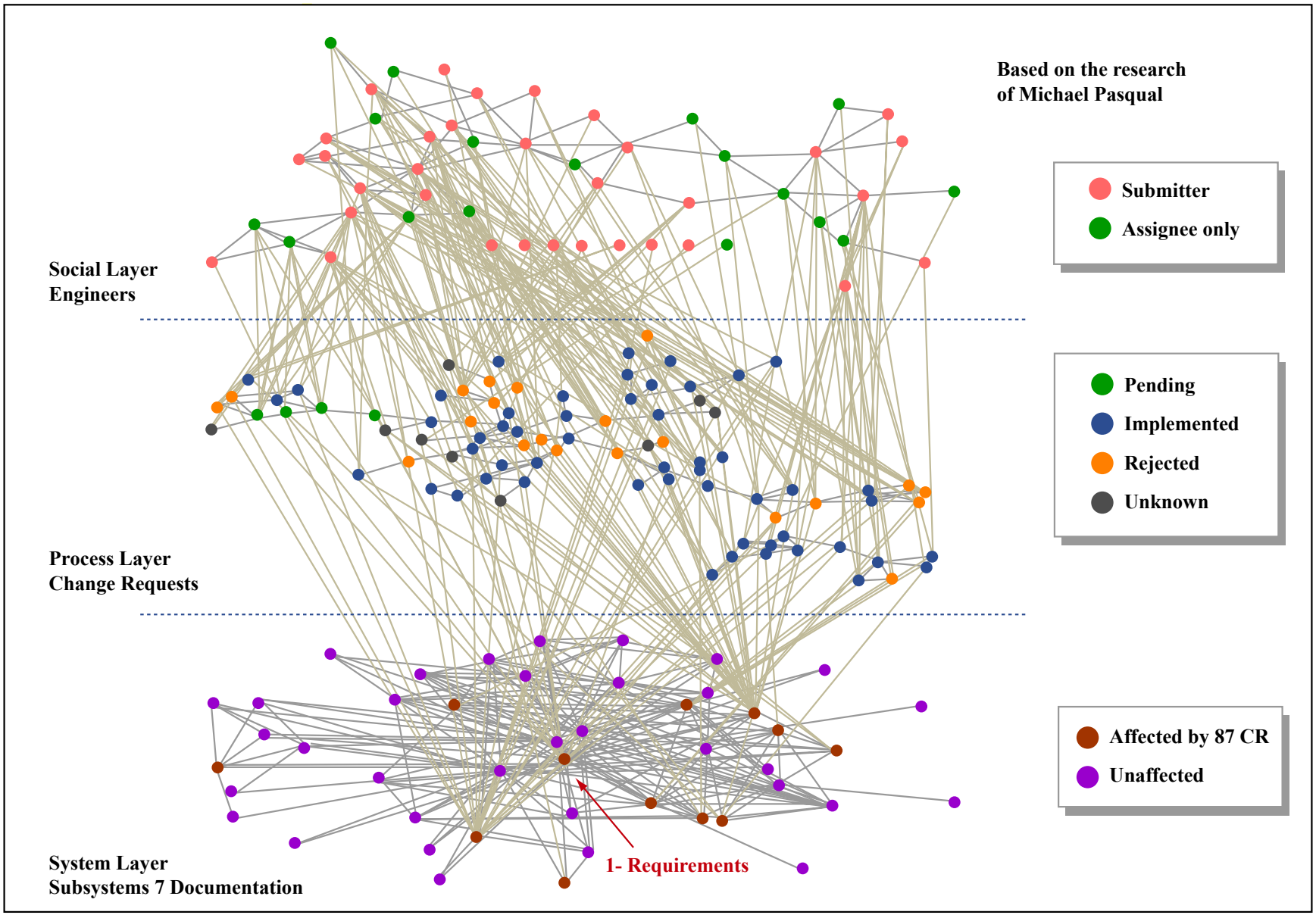
More and more requirements were added as systems grew in performance and complexity



Design requirements growth for aerospace vehicles.

Source:
AIAA MDO TC
White Paper, 1991

Requirements are not static



Data from Raytheon IDS

Requirements Standards

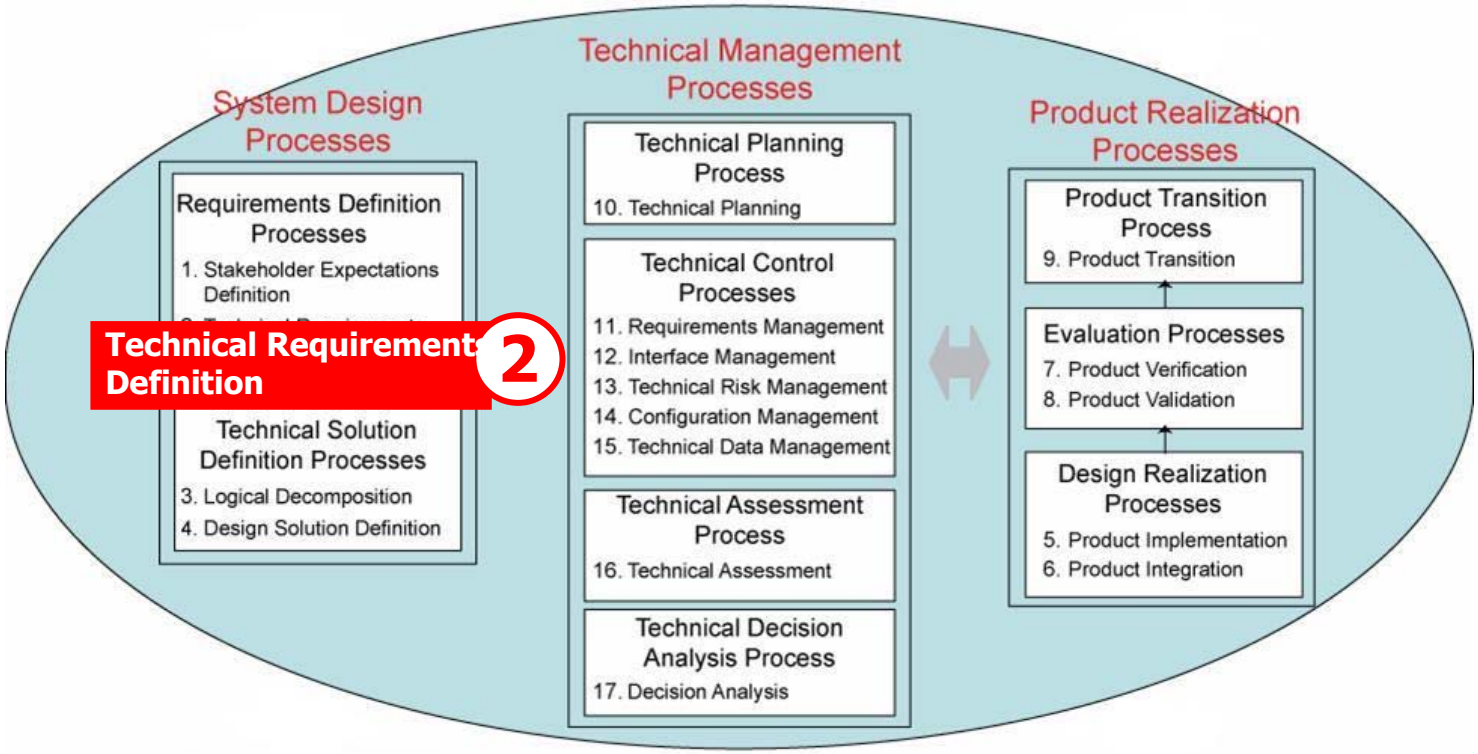
- NASA Systems Engineering Handbook
  - NASA/SP-2007-6105
    - Section 4.2 (pp. 40-48) – Technical Requirements Definition
    - Section 6.2 (pp. 131-135) – Requirements Management
    - Appendix C (pp. 279-281) – How to write a good Requirement
    - Appendix D (pp. 282-283) – Requirements Verification Matrix
- International Council of Systems Engineering (INCOSE)
  - Systems Engineering Handbook, Version 3.1
  - Requirements Working Group
    - http://www.incose.org/practice/techactivities/wg/rqmts/
- ISO/IEC 15288 (IEEE STD 15288-2008)
  - Systems and software engineering —
  - System life cycle processes
    - 6.4.1 Stakeholder Requirements Definition Process

Overview

- What are requirements?
  - Definition, Example, Evolution, Standards
- NASA Requirements Process
- Challenges of Requirements Definition
  - Flowdown and Allocation
  - Validation and Verification
  - Writing good requirements

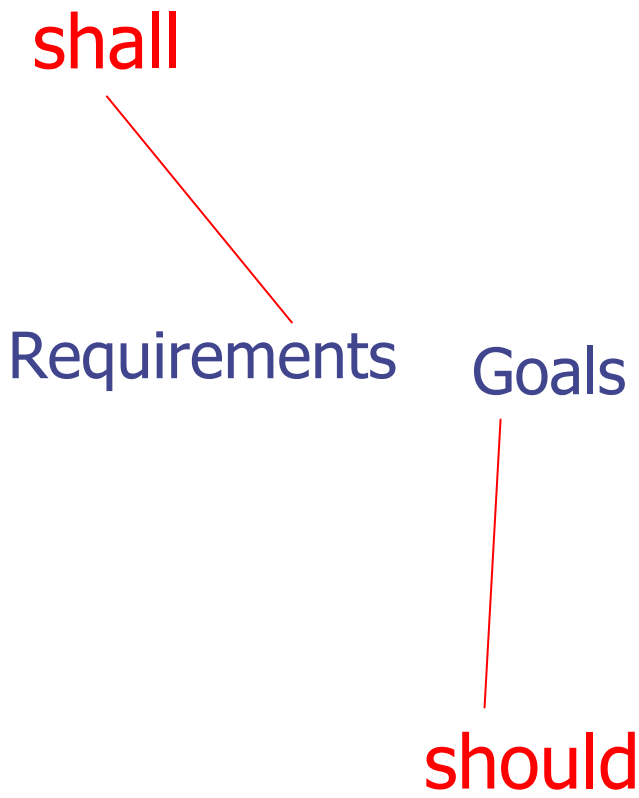
Technical Requirements Definition Process

SE Engine



- **Requirement 16 (Section 3.2.2.1) "The Center Directors or designees shall establish and maintain a process, to include activities, requirements, guidelines, and documentation, for definition of the technical requirements from the set of agreed upon stakeholder expectations for the applicable WBS model."**

Technical Requirements



Purpose of Technical Requirements Definition

- The Technical Requirements Definition Process
  - Is used to transform the baselined stakeholder expectations (input) into unique, quantitative, and measurable technical requirements (output)
- Requirements
  - Come in many flavors
  - Should be expressed as well-written "shall" statements that can be used for defining a design solution for the WBS model end product and related enabling products

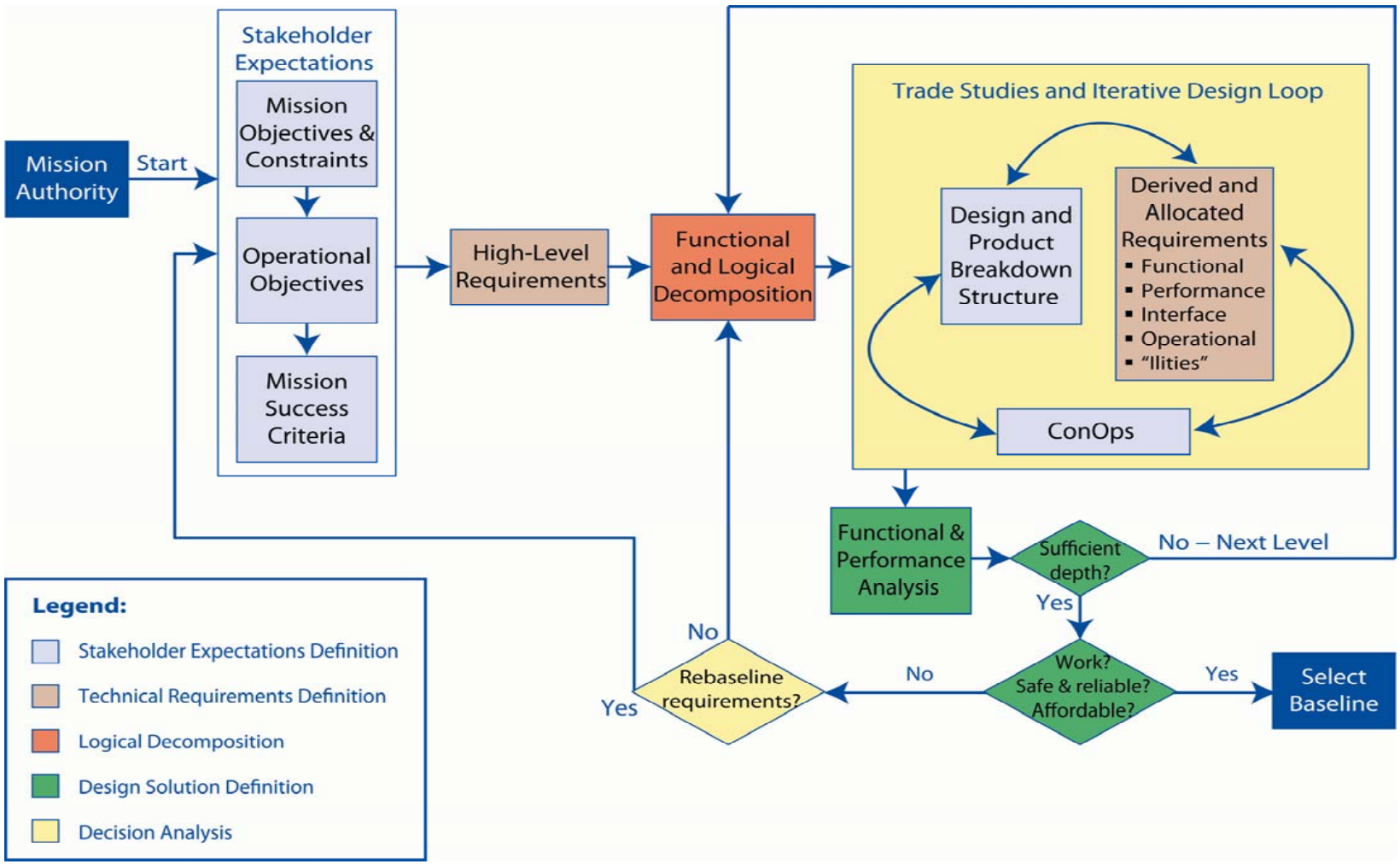
Importance of Technical Requirements Development (1/2)

- **Establishes the basis for agreement between the stakeholders and the developers on what the product is to do**
- **Reduces the development effort because less rework is required to address poorly written, missing, and misunderstood requirements.**
  - Forces the relevant stakeholders to consider rigorously all of the requirements before design begins
  - Careful review can reveal omissions, misunderstandings, and inconsistencies early in the development cycle
- **Provides a basis for estimating costs and schedules**
  - The description of the product to be developed as given in the requirements is a realistic basis for estimating project costs and can be used to evaluate bids or price estimates

Importance of Technical Requirements Development (2/2)

- **Provides a baseline for verification**
  - Organizations can develop their validation and verification plans much more productively from a good requirements document.
  - The requirements document provides a baseline against which compliance can be measured.
  - The requirements are also used to provide the stakeholders with a basis for acceptance of the system.
- **Facilitates transfer of the product to new users or new machines.**
- **Serve as a basis for later enhancement or alteration of the finished product.**

Interrelationships Among the System Design Processes



SP-2007-6105, Figure 4.01

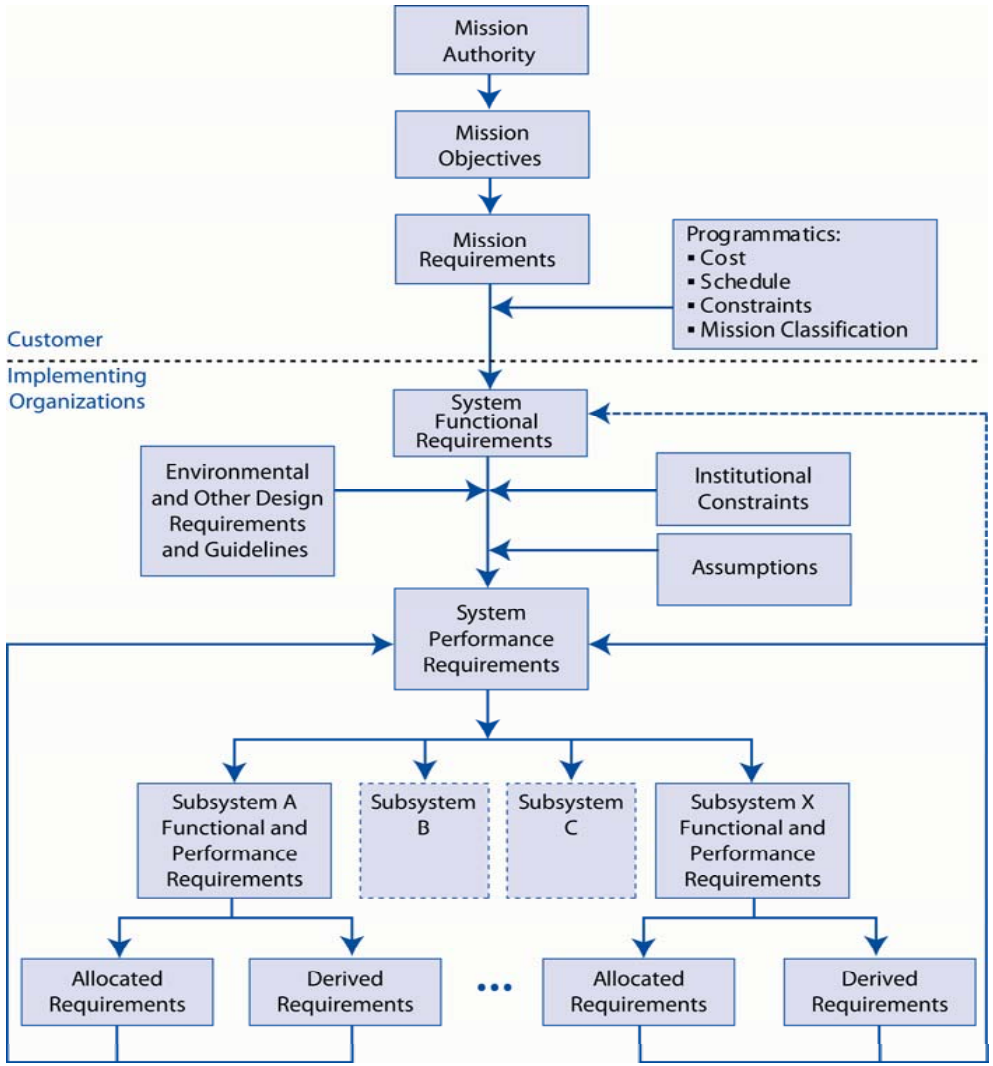
Types of Requirements

- **Functional Requirements define what functions need to be done to accomplish the mission objectives**
  - Example: The Thrust Vector Controller (TVC) shall provide vehicle control about the pitch and yaw axes.
    - This statement describes a high level function that the TVC must perform.
    - Statement has form of Actor – Action Verb – object acted on
- **Performance Requirements define how well the system needs to perform the functions**
  - Example: The TVC shall gimbal the engine a maximum of 9 degrees, +/- 0.1 degree
- **Constraints are requirements that cannot be traded off with respect to cost, schedule or performance**
  - Example: The TVC shall weigh less than 120 lbs.
- **Interface Requirements**
  - Example: The TVC shall interface with the J-2X per conditions specified in the CxP 72262 Ares I US J-2X Interface Control Document, Section 3.4.3.
- **Environmental requirements**
  - Example: The TVC shall use the vibroacoustic and shock [loads] defined in CxP 72169, Ares 1 Systems Vibroacoustic and Shock Environments Data Book in all design, analysis and testing activities.
- **Other -illities requirement types described in the SE Handbook include: human factors, reliability requirements, and safety requirements.**

Attributes of Acceptable Requirements

- A complete sentence with a single "shall" per numbered statement
- Characteristics for Each Requirement Statement:
  - Clear and consistent – readily understandable
  - Correct – does not contain error of fact
  - Feasible – can be satisfied within natural physical constraints, state of the art technologies, and other project constraints
  - Flexibility – Not stated as to how it is to be satisfied
  - Without ambiguity – only one interpretation
  - Singular – One actor-verb-object requirement
  - Verify – can be proved at the level of the architecture applicable
- Characteristics for pairs and sets of Requirement Statements:
  - Absence of redundancy – each requirement specified only once
  - Consistency – terms used consistent
  - Completeness – usable to form a set of "design-to" requirements
  - Absence of conflicts – not in conflict with other requirements or itself

Requirements Decomposition, Allocation and Validation



- Requirements are decomposed in a hierarchical structure starting with the highest level requirements.
- These high-level requirements are decomposed into functional and performance requirements and allocated across the system.
- These are then further decomposed and allocated among the elements and subsystems. This complete set of design-to requirements is achieved.
- At each level of decomposition (system, subsystem, component, etc.), the total set of derived requirements must be validated against the stakeholder expectations or higher level parent requirements.

Source: SE HB Figure 4.2-3

Requirement = Metric + Value*

- To be effective, requirements should have an associated metric plus a target value
- Values can be continuous (100 mph), discrete/logical (meets standards), qualitative (pleasing to most people)**
- More quantification helps clarify intent and ensure success
  - Requirements should be testable
- For functional requirements, the metric should be directly related to the delivered external process

- *** note this is a different use of the word "value"**
- **** as long as they can be verified**

19

Formulation of Metrics

- May be marginal, absolute, probabilistic
  - = X% improvement in _____
  - = X value of _____
  - = X value of _____ with 90% confidence
- Tradition is metric based on benefit/performance (with cost*, schedule and risk assessed later)
- Current practice is metric based on benefit/performance and cost (with schedule and risk assessed later)
- "Ideal" would be metrics which include benefit/performance, schedule, cost and risk

*** cost can include liens on resources in addition to $**

20

What are Requirements for ...

- **Automobile ?**
- **Data link ?**
- **Copy machine ?**
- **Golf club ?**
- **Dishwasher?**
- **Helicopter ?**

21

Monitoring a Requirement

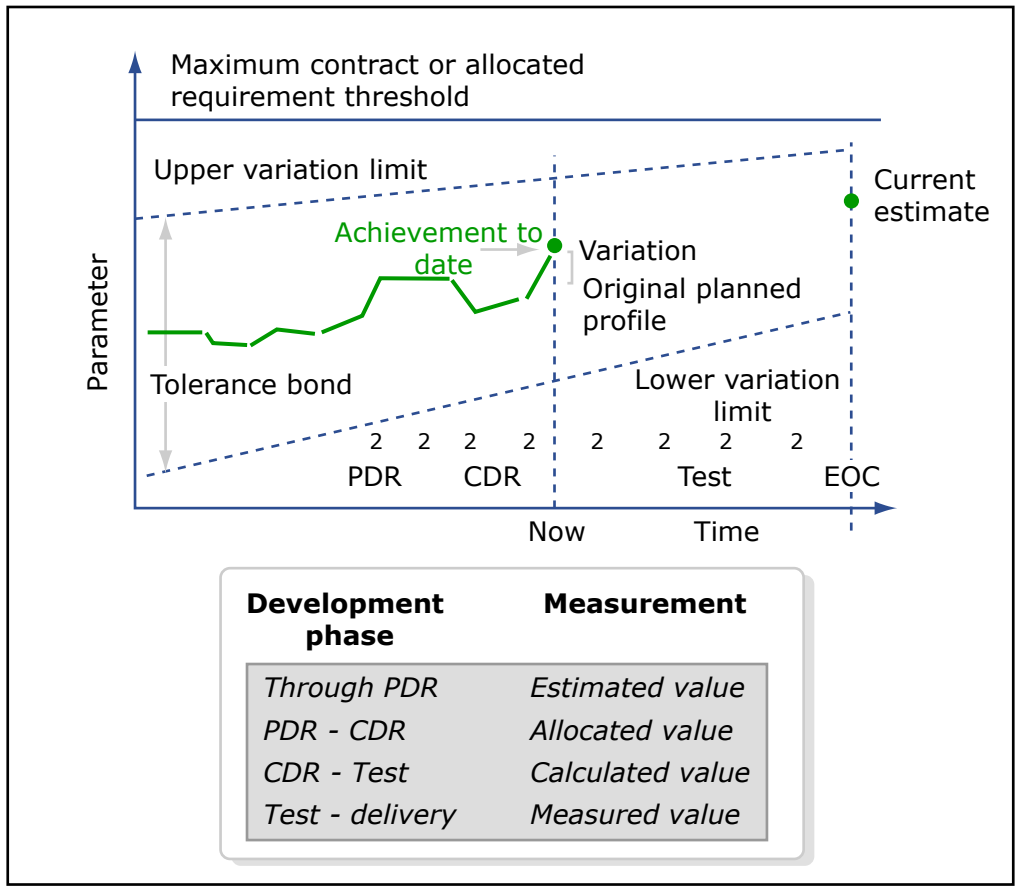
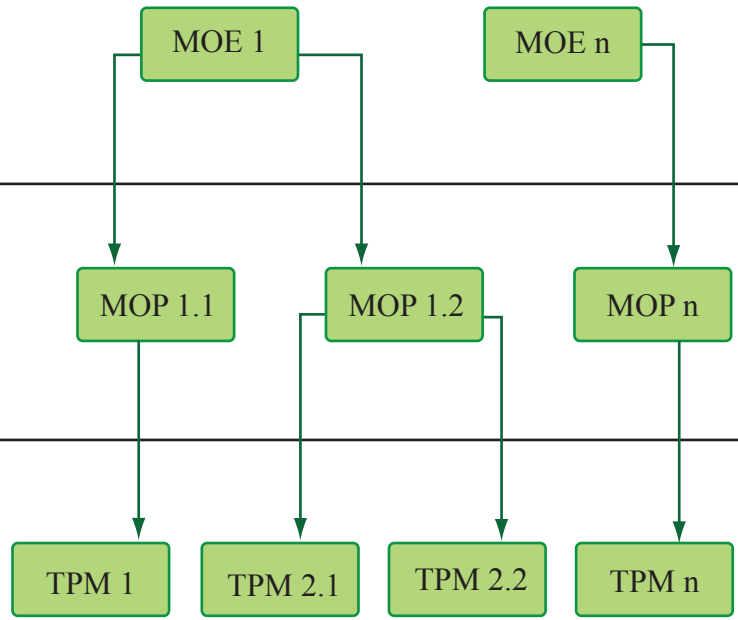


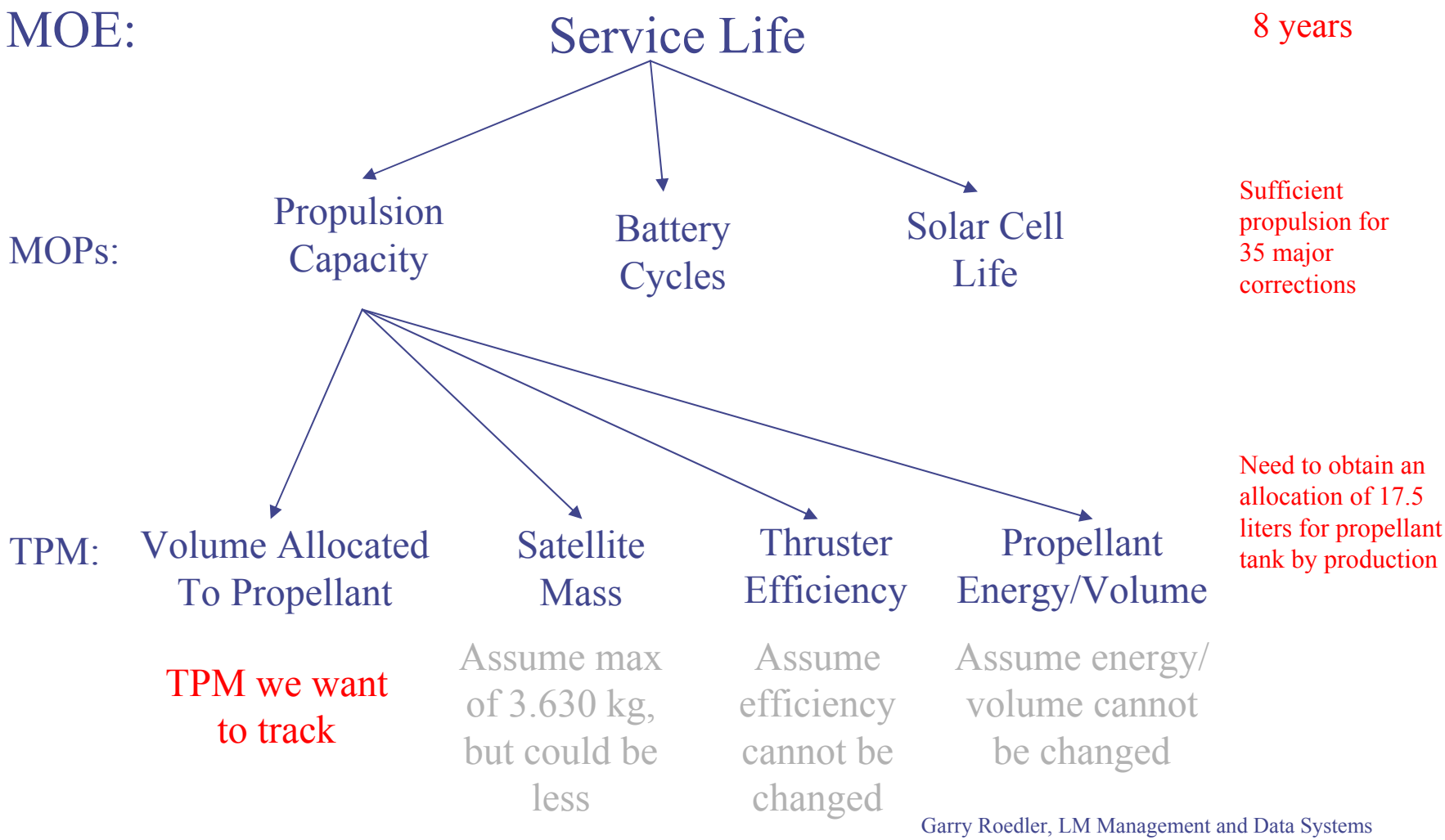
Image by MIT OpenCourseWare.

MOE, MOP, TPM Relationship

	Qualitative	
Measures of Effectiveness	*Derived from Stakeholder expectation statements; deemed critical to success of the system.*	
	Quantitative	
Measures of Performance	*Broad functional and performance requirements combinations; means of requirments combinations; means of assuring meeting the associated MOE.*	
	Significant Qualifier	
Technical Performance Measures	*Key performance or technical attribute Measurable. Progress profile can be established and monitored.*	

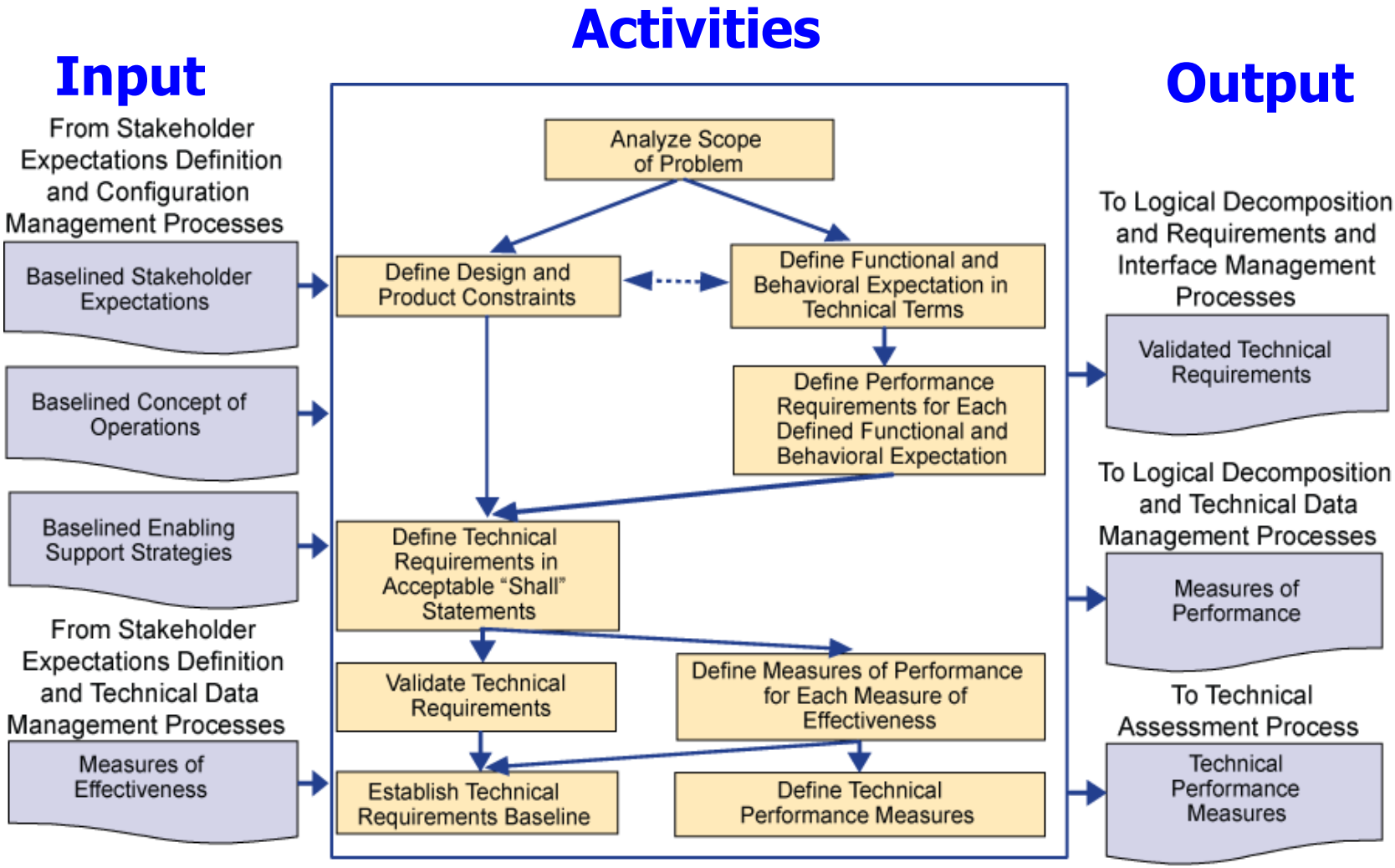


Example of MOE/MOP/TPM Relationship



Garry Roedler, LM Management and Data Systems

Technical Requirements Definition Best Practice Process Flow Diagram



Overview

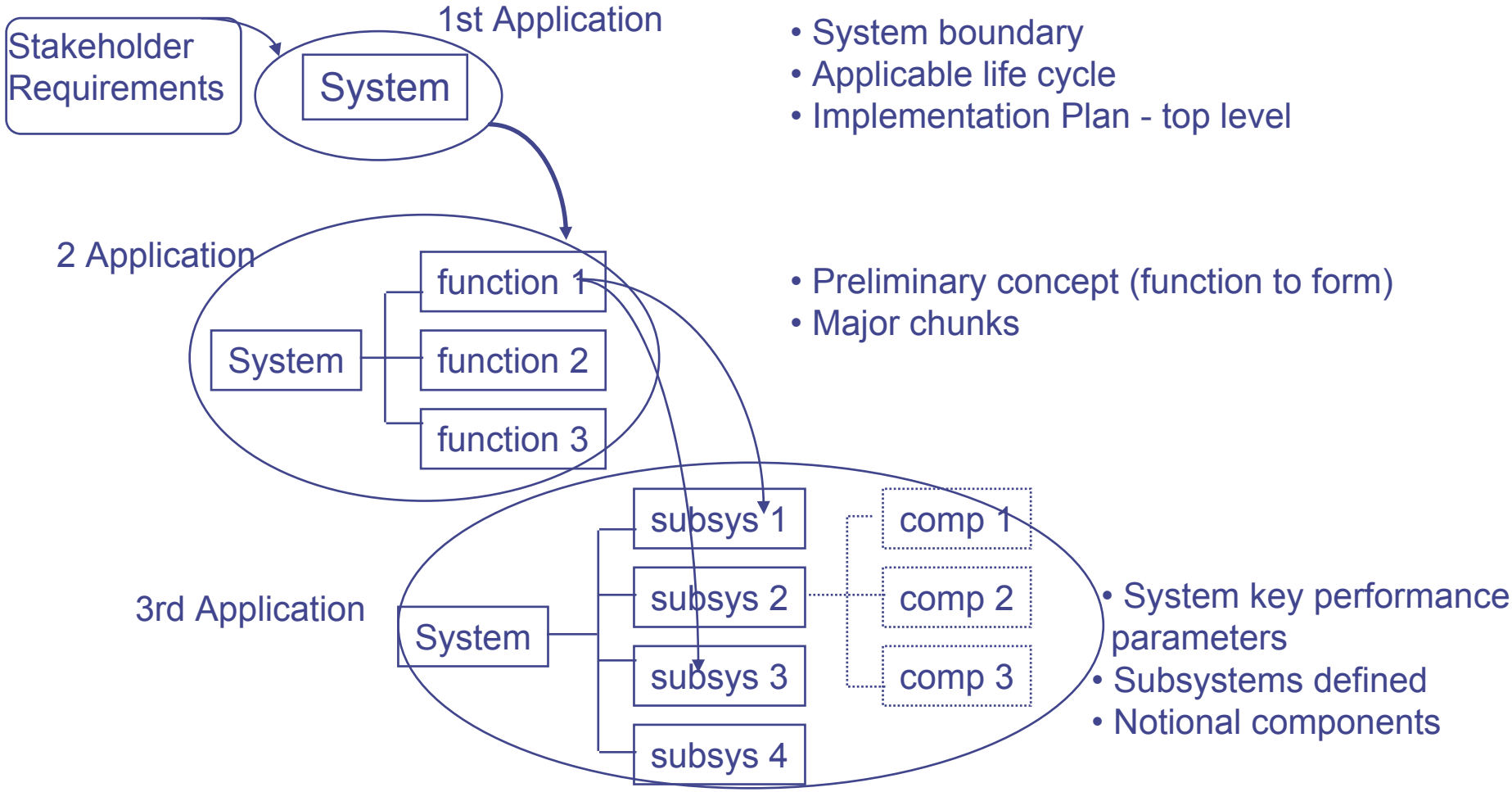
- What are requirements?
  - Definition, Example, Evolution, Standards
- NASA Requirements Process
- Challenges of Requirements Definition
  - Flowdown and Allocation
  - Validation and Verification
  - Writing good requirements

Requirements Allocation

- Decompose system requirements into lower levels of design.
  - Define all the lower level functions which must be performed to satisfy the requirement
  - Create architecture of sub-components to provide those functions
- Allocate a level of performance to each lower level function
  - Specify interface requirements to other sub-systems
- Closure - Ensure that satisfaction of the set of requirements at the lower level will guarantee satisfaction of the higher level requirement.

Ref: Isoperformance

Requirement Allocation Process



 Difficult to decompose reqts to lower levels while staying solution neutral

Common Problems

- Writing implementations (How) instead of requirements (What)
  - Forces the design
  - Implies the requirement is covered
- Using incorrect terms
  - Use “shall” for requirements
  - Avoid “support”, “but not limited to”, “etc”, “and/or”
- Using incorrect sentence structure or bad grammar
  - Use “The system shall be capable of….” followed by single predicate

Common Problems continued

- Writing unverifiable requirements
  - E.g., minimize, maximize, rapid, user-friendly, easy, sufficient, adequate, quick
- Missing requirements
  - Requirement drivers include

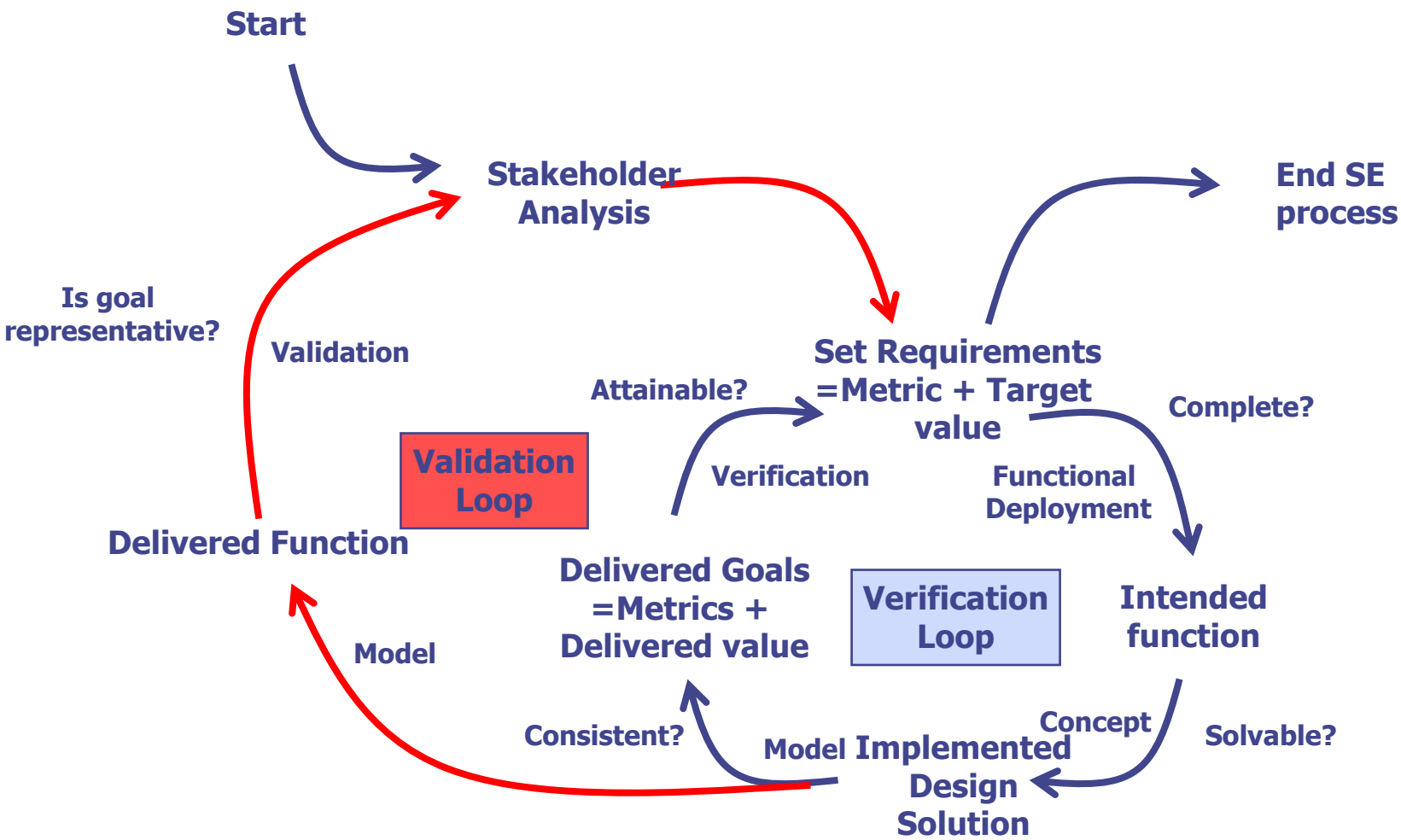
Functional	Performance	Interface
Environment	Facility	Transportation
Training	Personnel	Reliability
Maintainability	Operability	Safety

- Requirements only written for "first use"
- Over-specifying

Verification

- Every requirement must be verified to ensure that the proposed design actually satisfies the requirement by
  - Examination,
  - Test,
  - Demonstration, or
  - Analysis
- Requirement documentation specifies the development phase and method of verification

Verification and Validation Loops



32

Questions ?

16.842 Fundamentals of Systems Engineering

Fall 2009