Starting at the Top

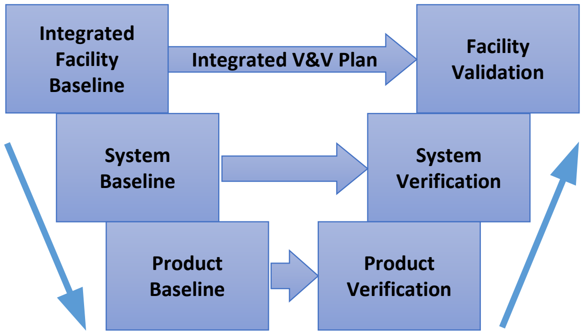
Client and stakeholder requirements are the starting point for development efforts. These upper-level requirements must be analyzed and interpreted to form the requirements baselines for the integrated facility, system and components. A baseline is a configuration controlled set of approved requirements allocated to the physical elements of the facility (including software). These baselines are established sequentially and used to control change and reduce risk.

The first baseline establishes the facility-level behaviors, performance and constraints that are necessary for its design and validation. This baseline provides the criteria for facility acceptance in measurable engineering terms.

The integrated facility baseline is used as the starting point for subsequent analyses to identify required systems and to derive the associated system requirements. These system-level requirements form the baseline that will be used to drive system design and the subsequent verification activities.

The system baseline provides the input necessary to enable detailed design. The needed components are identified and specified in design. The collection of this information forms the product baseline used to procure, fabricate, construct and integrate components, structures, systems and facilities.

Requirements of each baseline form the basis for integrated Verification and Validation (V&V) planning. The following diagram illustrates the concepts that development is a top-down process followed by V&V activities that demonstrate compliance with requirements of each baseline.



Useful References

International Organization for Standardization (2008). ISO/IEC/IEEE 15288 *Systems and software engineering - Life cycle processes – System life cycle processes*.

International Organization for Standardization (2011). ISO/IEC/IEEE 29148 *Systems and software engineering - Life cycle processes - Requirements engineering*.

International Council On Systems Engineering (2011). *Systems Engineering Handbook: A Guide For System Life Cycle Processes and Activities*.

International Council On Systems Engineering (2012). *Guide for the Application of Systems Engineering in Large Infrastructure Projects.*

Requirements Working Group, International Council On Systems Engineering (2012). *Guide For Writing Requirements*.

This Leaflet

This leaflet is part of a series intended as a brief introduction to the application of systems engineering approaches to infrastructure projects. It was developed by the International Council On Systems Engineering (INCOSE) Infrastructure Working Group in the interest of aiding industry.

For further information about the application of systems engineering in large infrastructure projects, including a Guide applicable to the Construction project stage, go to www.incose.org and look for publications.

INCOSE is a not-for-profit membership organization founded to develop and disseminate the interdisciplinary principles and practices that enable the realization of successful systems.

INCOSE
INFRASTRUCTURE
WORKING GROUP

Applying Systems Engineering to Industrial & Infrastructure Projects

Managing Requirements for Design

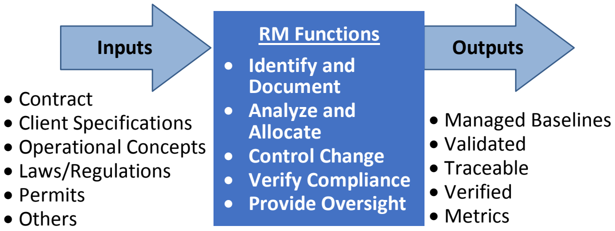
The Need

Large industrial facilities and infrastructure projects are subject to the requirements of a multitude of stakeholders. Requirements originate from the contract, laws and regulations, operational concepts, site conditions, external interfaces and utilities, industrial codes and standards, operator needs, the public interest and other sources.

Projects often suffer several types of problems caused by inadequately managed requirements. For example, contractual and regulatory commitments are overlooked. Requirements do not flow down to subcontractors or suppliers. Requirements are incomplete, unclear, not regularly updated, and subject to widely varying interpretations. Projects have difficulty demonstrating requirements were met and the responsibility for managing the requirements is often unclear. Solving these and other issues drive the need for a more formal approach to managing requirements.

Requirements Management as a Process

The figure below provides a simple context diagram showing requirements management as a process. It gathers input from authorized sources to produce managed baselines of validated and traceable requirements. The process gathers evidence that requirements were met. Metrics revealing requirements issues are also produced. The process functions that are necessary to achieve a viable process are shown in the center box.



Inputs:
- Contract
- Client Specifications
- Operational Concepts
- Laws/Regulations
- Permits
- Others

RM Functions:
- Identify and Document
- Analyze and Allocate
- Control Change
- Verify Compliance
- Provide Oversight

Outputs:
- Managed Baselines
- Validated
- Traceable
- Verified
- Metrics

INCOSE-PI-2015-003-1.0, Version 1.0. Authored by Mike deLamare.

Requirements Management Functions

Effective requirements management needs to ensure the following activities are well executed:

Identify and Document

Capture why the end product or system needs to exist in terms of value provided to the end user. Elicit and document requirements from the client and stakeholders. Gain their agreement on the written statements and on the measures of successful completion. Ensure regulatory requirements are understood by all parties.

Analyze and Allocate

Derive explicit measurable requirements useful for design that are directly traceable to providing the system's value within constraints (e.g., cost, regulatory, and social constraints). Continue derivation and decomposition such that derived requirements can be allocated to system and components. Link requirements and their supporting bases to provide two-way traceability. Demonstrate requirement validity through analysis, authority or other demonstration of need. Valid requirements are those that are necessary, clear, achievable and verifiable.

Control Change

Analyze requirements changes for the full impact, approve change through a prescribed process and appropriate levels of authority. Understand the impact before committing to the change. Ensure approved changes are propagated to all affected items.

Verify Compliance

Verify requirements are met and that the resulting system actually provides the needed value in the real environment of use. Engineering should determine test conditions and acceptance criteria during design to enable test planning.

Provide Oversight

Manage the resulting information and the processes being implemented. The depth of requirements managed by a program needs to be set. Usually it is sufficient to manage to the level of subcontracts or procured items.

Develop metrics useful to show the quality of the requirements management activities. Integrate requirements across systems to ensure a balanced solution and compatible interfaces.

Requirement Topics (Areas)

Requirements have natural groupings and topics. These grouping are based on how the law is organized, or natural principles, technical specialization or best practices. These groupings can be used to organize requirements for easier management. More importantly, these groupings need to be recognized up front so that they are not overlooked. It is extremely important to carefully gather and analyze these requirements. A few example examples include:

- Mission-derived
- Environmental Protection
- Government Regulations
- Industry Codes and Standards
- Natural Phenomenon Hazards
- Operational and Induced Environments
- Materials
- Electromagnetic Compatibility
- Nuclear, Chemical & Industrial Safety,
- System Safety and Security
- Fire Protection
- Human Factors
- Reliability, Availability, and Maintainability
- Others such as the operating environment, technologies applied, and client standards.

Requirements Precedence

One key concept is the idea that requirements have a recognized precedence. When there is a conflict, the more authoritative requirement should dominate the selected solution. Precedence is based on the authority of its giver as recognized by law, contract or approved process. Clearly law has the higher precedence than a contract since any part of a contract that violates the law is invalid. The contract typically has the next level of authority. This is followed by those documents approving agreements with regulators or with contractually recognized external organizations.

Precedence is also affected by development sequence as system requirements are decomposed to establish incremental baselines in a top-down manner. For example, system requirements are derived based on the client's needs. Subsequently, components requirements are derived to meet the system's needs. Thus, system requirements have a higher precedence.

A project-specific hierarchy showing the precedence level should be developed to help people apply these concepts.

Characteristics of Good Requirements

Well-written requirements are essential for project success, including the following key characteristics. Refer to the INCOSE Systems Engineering Handbook for more discussion.

Necessary

State only what is determined to be necessary for achieving the client's mission within regulatory constraints. These statements are derived through analytical means to the depth necessary for design, procurement, implementation and verification.

Clear

Convey what must be achieved in a manner that can be understood by those who are expected to implement the requirement, without having to ask the author what was meant.

Achievable

Confirm with the implementer that the requirement can be affordably achieved either by previously developed means, or within a reasonable period of development.

Traceable

Ensure derived requirements can be traced to a user need or a higher-level specification, and to a supporting analytical basis. Derived requirements need to also trace to the implementing item.

Verifiable

Requirements must be stated in a manner that compliance can be objectively confirmed. Typical methods of confirmation include analysis, inspection, demonstration and test.

Complete

A set of requirements needs to be complete, such that if all are met then the resulting system will successfully achieve the client's need for the system. In addition, the needs of other stakeholders will be addressed to the agreed extent, and the regulations will be met.

Implementation Free

State *what* is required and *how well* it needs to be done without bias for *how* it will be done. The design team should be allowed to choose the best means of accomplishing the requirements. This helps to provide stable requirements and to control cost.