SMARTSTYLE PROJECT
Sardi Shehu

Organizational Requirements

Operational Requirements

Definition: Operational requirements are those statements that "identify the essential capabilities, associated requirements, performance measures, and the process or series of actions to be taken in effecting the results that are desired in order to address mission area deficiencies, evolving applications or threats, emerging technologies, or system cost improvements [1]." The operational requirements assessment starts with the Concept of Operations (CONOPS) and goes to a greater level of detail in identifying mission performance assumptions and constraints and current deficiencies of or enhancements needed for operations and mission success. Operational requirements are the basis for system requirements.

Background

A key process in the concept development phase is analysis to define the operational requirements of the system. Operational requirements are typically prepared by a team of users, user representatives, developers, integrators, and MITRE SEs and are based on the identified user need or capability gaps (see the Operational Needs Assessment article). Establishing operational requirements forms the basis for subsequent system requirements and system design in the system design and development phase.

The operational requirements focus on how the users will operate the system, including interfaces and interoperability with other systems. The requirements establish how well and under what conditions the system must perform. The operational requirements should answer these questions:

> Who is asking for this requirement? Who needs the requirements? Who will be operating the system?
>
> What functions/capabilities must the system perform? What decisions will be made with the system? What data/information is needed by the system? What are the performance needs that must be met? What are the constraints?
>
> Where will the system be used?
>
> When will the system be required to perform its intended function and for how long?
>
> How will the system accomplish its objective? How will the requirements be verified?

Process

The operational requirement definition process includes the following activities:

- Identify stakeholders who will or should have an interest in the system throughout its entire life cycle.
- Elicit requirements for what the system must accomplish and how well. Doing this in the form of operational scenarios and/or use cases can be particularly helpful in discussions with end users.
- Define constraints imposed by agreements or interfaces with legacy or co-evolving enabling systems.
- Establish critical and desired user performance: thresholds and objectives for operational performance parameters that are critical for system success and those that are desired but may be subject to compromise in order to meet the critical parameters. To assess the feasibility of meeting performance, consider, suggest (if appropriate), and help formulate prototypes and experiments to determine whether near-term capabilities can satisfy users' operational performance needs. Results of the prototypes can help determine an evolutionary strategy to meet critical performance. Additionally, technology assessments can help gauge when desired performance might be met in the future.
- Establish measures of effectiveness and suitability: measures that reflect overall customer/user satisfaction (e.g., performance, safety, reliability, availability, maintainability, and workload requirements) [2]. Many of these measures will be used for the test and evaluation lifecycle building phase.

This process is consistent with the standard process for determining any level of requirements. See the SEG Requirements Engineering topic for a discussion on eliciting, analyzing, defining, and managing requirements and discussions on the characteristics of "good" requirements (e.g., concise, necessary, attainable, verifiable, and traceable, implementation free, evolvable). It is also important to establish a requirements baseline that is kept under configuration control (see the SEG's Configuration Management topic). Together with the rationale, this provides an established and complete audit trail of decisions and changes that were made. The configuration baseline will also identify and manage the trade-offs of satisfying near-term requirements versus allocating requirements over the evolution of a system.

Documentation

The operational requirements are captured in a document, model, or specification (e.g., user requirements document, operational requirements document, or capabilities development document). The type of document depends on the type of acquisition and the customer organization, (e.g., Department of Defense, Federal Aviation Administration, Department of Homeland Security, Internal Revenue Service, or other government agency). Whatever name and form these documents take, they provide a basic framework for the articulation and documentation of operational requirements that all stakeholders will use. The complexity of the intended system and its operational context will govern the required level of detail in the operational requirements document. Examples and formats of these documents are found in the references.

Best Practices and Lessons Learned

The following tips from active systems engineering practitioners may help you work through the concept development phase and the operational requirements development process.

a) Work with the end users early and often. Be sure to fully understand their mission, operational domain, and most important, their constraints. It is helpful to talk their "language." This allows for an easier exchange of ideas, resolution of conflicts, etc. Participate in training and exercises that the user community is involved in to get a first-hand perspective of the operational environment.

b) Create mutually beneficial interactions. Determine the users' needs by using mutually beneficial studies or analyses, including modeling and simulation, prototypes, and demonstrations where appropriate. These help the users justify and defend capability needs while providing the acquisition organization with requirements and CONOPS to start system development, testing, and fielding.

c) Organize your thinking before engaging users. It is often difficult for users to develop requirements from scratch. Draft your understanding of their requirements prior to engaging with them and create a straw man for discussion. This provides a good starting point for discussion. They will tell you if it is wrong. Demonstrations of your understanding using executable models or prototypes of capabilities will help both you and the users to engage on the operational needs and realm of the possible.

d) Help users understand new technology. Provide users with suggestions on how they might employ a new technology. Often, users cannot see past how they do business today. Introducing them to technology might help them break loose from their thought processes, develop new processes, and possibly rethink or refine some requirements. Consider the use of prototypes to help demonstrate possibilities and show users the technical aspects of a potential solution as they identify their operational needs and consider gives-and-takes based on solution feasibility and constraints.

e) Explain technology limitations clearly and simply. Clearly and simply explain the limitations of the technology to users, including maturity and associated risk. This helps ensure that requirements are achievable, secures their buy-in on what is possible, and stimulates them to think about how to use what they can get. Again, consider using prototypes or experiments of capabilities that can help bring technology issues to the forefront with users.

f) Engage users throughout the process. It is important to stay engaged with the user community through the system development process. Break down barriers and overcome incorrect and bad perceptions. Ensure that users are involved in the decision-making process. Ensure they are involved in subsequent decision-making that concerns tradeoffs affecting operational utility or performance. Keep users apprised of schedule and capability impacts. This builds trust, cooperation, facilitates quick turn times on questions, and helps ensure that the users' needs and objectives are met.

g) Make user satisfaction a priority. Make customer/user satisfaction a key metric for your program.

h) Make delivery to the users a primary driver. Getting capabilities to users early and often is the best strategy. The users have a mission to satisfy, and every capability that can help them is needed as soon as feasible. Evolve the deliveries over time based on priorities, associated capability, and feasibility of implementation.

i) Build a foundation of trust. The greatest likelihood of making good decisions occurs when the users and acquisition communities trust each other.