Team Capability

> The printable version is no longer supported and may have rendering errors. Please update your browser bookmarks and please use the default browser print function instead.

Lead Authors: *Dick Fairley,* ***Contributing Authors:*** *Alice Squires, Art Pyster, Heidi Davidz*

The capability of a team to perform systems engineering (SE) depends on having competent personnel, adequate time, sufficient resources and equipment, and appropriate policies and procedures (Torres and Fairbanks 1996).

The team should have a charter. Staff must be proficient in the needed competencies and must work together with the right attitude, under the right organization, and with appropriate tools, training, and processes such as configuration management and peer review.

Those responsible for the team attaining the desired capability need to organize, staff, develop, and assess the team. Techniques for pilot projects, post-mortem analysis, and lessons learned can be applied as well.

☐

Contents

Organizing the Team

Project teams, and the roles of systems engineers within those teams, depend on factors such as the nature, size, and scope of the project, the organization's preferred way of organizing teams, and external constraints such as a larger program in which the project may be embedded. Options range from a dedicated team of systems engineers, to Integrated Product Teams, to teams that include other kinds of engineers that perform systems engineering.

Systems engineers and SE teams may play the roles of technical leads, consultants, or advisers; this influences the ways in which SE teams are organized. In some organizations, systems engineers and SE teams provide technical leadership; they perform requirements analysis and architectural design, conduct trade studies, and allocate requirements and interfaces to the various elements of a system. In addition, they work with component specialists, develop integration plans and perform system integration, verification, and validation. Depending on the scope of effort, they may also install the system and train the operators and users; provide ongoing services to sustain the system; and retire/replace an aged system. Systems engineers may be housed within a functional unit of an organization and assigned, in matrix fashion, to projects and programs, or they may be permanently attached to a project or program for the duration of that endeavor. They may be organized based partially on their domain of expertise, such as finance or telecommunications. For additional information on organizational options see Determining Needed Systems Engineering Capabilities in Businesses and Enterprises.

In other cases, one or more systems engineers may provide consulting or advisory services, as requested, to projects and programs. These engineers may be dispatched from a central pool within an organization, or they may be hired from an outside agency.

An SE team can be organized by job specialization, where each SE team member (or each SE sub-team) plays a different role; for example, requirements engineering, system architecture, integration, verification and validation, field test, and installation and training In this case the various job specializations are

typically coordinated by a lead systems engineer.

Alternatively, an SE team can be organized by subsystem where each SE team member (or SE sub-team) performs the previously indicated functions for each of the subsystems with a top-level team to coordinate requirements allocation, interfaces, system integration, and system verification and validation.

Ideally, roles, responsibilities, and authority will be established for each project or program and used to determine the optimal way to organize the team. Sometimes, however, an *a priori* organizational, project, or program structure may determine the structure, roles, responsibilities, and authority of the SE team within a project or program; this may or may not be optimal.

Within a project, a systems engineer or SE team may occupy a staff position subordinate to the project manager, as indicated in Figure 1 or conversely, the SE team may provide the authoritative interface to the customer with the project manager or management team, serving in a staff capacity, as indicated in Figure 2. In both cases, SE and project management must work synergistically to achieve a balance among product attributes, schedule, and budget. Eisner (2008) lays out various approaches to organizing systems engineers. For additional information see Systems Engineering and Project Management.
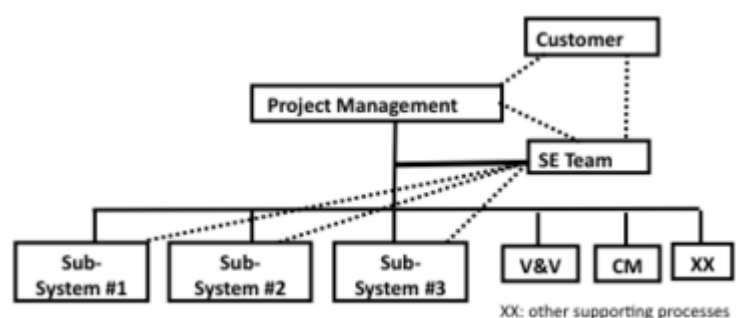

Figure 1. SE Team Subordinate to Project Management.
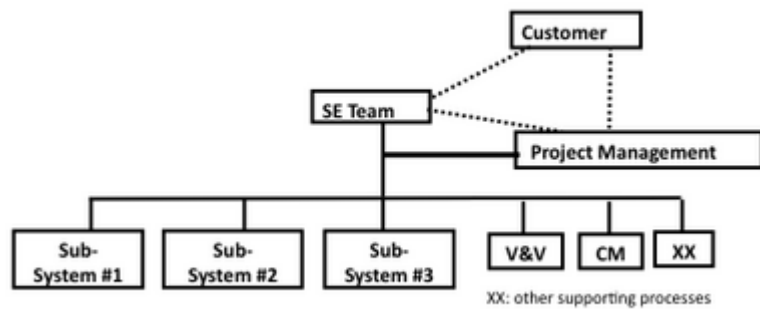(SEBoK Original)

Figure 2. Project Management Subordinate to Systems Engineering. (SEBoK Original)

In scaling up to the program level, the considerations portrayed in Figures 1 and 2 can be generalized so that a top-level SE team provides coordination among the subordinate projects. In this case, each project has an SE team, and within each project the SE team members can be organized in either of the ways indicated in the figures. When scaling up to programs, each of the sub-systems in Figures 1 and 2 are separate, coordinated projects.

The models presented in Figures 1 and 2 can be scaled down to smaller projects, where an individual systems engineer performs the SE activities, either in the subordinate position of Figure 1 or the superior position of Figure 2. In this case, there is a single subsystem (i.e., the system) and the supporting functions may be provided by the systems engineer or by supporting elements of the larger organization.

The roles to be played by members of a SE team are influenced by the structures adopted as part of the organizational strategy of the business in which the team is operating (see Systems Engineering Organizational Strategy). In Product Centered Organizations, for example, an Integrated Product Team (IPT) is assigned to each element of the system breakdown structure (SBS). Each IPT consists of members of the technical disciplines necessary to perform systems engineering functions for that element of the system.

At the program level there is a top-level IPT commonly called a SE and integration team (SEIT), whose purpose is to oversee all of the lower level IPTs. Some specialists, such as reliability and safety engineers, may be assigned to a team to cover all elements within a given level of the SBS. These teams are sometimes called Analysis and Integration teams (AITs), and are created at various levels of the SBS as needed.

Organizing communication and coordination among a group of systems engineers should follow the well-known *7 ± 2* rule because the number of communication paths among *N* engineers is *N(N-1)/2*; i.e., the number of links in a fully connected graph (Brooks 1995). There are 10 communication paths among 5 engineers, 21 among 7 engineers, and 36 among 9 engineers. An SE team of more than 10 members (45 paths) should be organized hierarchically with a top-level team leader. Sub-teams can be partitioned by product subsystem or by process work activities (analysis, design, integration).

Staffing the Team

Once the organizational structure of the SE team is understood, the team can be staffed. As noted in Enabling Individuals, competency of an individual is manifest in the knowledge, skills, abilities, and attitudes needed for the individual to perform a specific task efficiently and effectively. Different levels of competency may be needed in different situations. Competencies include occupational competence, social competence, and communication competence. Competent systems engineers, for example, have SE knowledge, skills, and ability; engage in systems thinking; possess emotional intelligence; and have good communication and negotiation skills. In addition, competent systems engineers are typically competent within specific domains (e.g. aerospace, medicine, information technology) and within specific process areas of systems engineering (e.g., requirements, design, verification and validation). (See Part 3, Systems Engineering and Management for more information on specific process areas.) The article on Roles and Competencies includes a summary of SE competency models. Based on the context, these competency models are tailored to match the needs of each project. The roles within the team are defined, and competencies are linked to the roles. The lists of competencies given in those models are most often distributed among the members of a SE team. It is not often that a single individual will possess the full list of competencies given in these models.

In addition to individual competencies to perform SE roles, the collective SE competencies needed by a team depend on additional factors including the domain, the stakeholders, the scope of the effort, criticality of outcome, new initiative versus enhancement, and the responsibilities and authority assigned to the team. For example, collective SE competencies needed to develop the IT enterprise architecture for a small company are

quite different from those needed to develop the architecture of an aircraft which is engineered and manufactured in a distributed fashion around the world.

To determine the collective set of competencies an SE team needs to conduct a project or program, perform the following steps:

1. Identify the context, to include:
    1. domain
    2. stakeholders
    3. organizational culture
    4. scope of effort
    5. criticality of the product, enterprise endeavor, or service
    6. new initiative or sustainment project
2. Clarify the responsibilities, authority, and communication channels of the systems engineering team
3. Establish the roles to be played by systems engineers, and other project personnel as determined by context, responsibilities, and authority
4. Determine the required competencies and competency levels needed to fill each of the systems engineering roles
5. Determine the number of systems engineers needed to provide the competencies and competency levels for each role
6. Determine the availability of needed systems engineers
7. Adjust based on unavailability of needed systems engineers
8. Organize the systems engineering team in a manner that facilitates communication and coordination within the SE team and throughout the project or program
9. Consult stakeholders to ask "What are we missing?"

Competency models and skills inventories, such as INCOSE (2010) and Curtis et al. (2001), can be used as checklists to assist in determining the needed competencies and competency levels for a product, enterprise, or service. (See Roles and Competencies.)

When the needed competencies, competency levels, and capacities have been determined, one of two situations will arise: In the optimal situation, the number of

systems engineers who have the needed competencies and competency levels to fill the identified roles will be available; or, they will be unavailable or cannot be provided because of insufficient funding. For example, a new initiative may need a lead engineer, a requirements engineer, a systems architect and a systems integrator-tester to accomplish systems engineering tasks. Budgetary constraints may indicate that only two of the four roles can be supported. Compromises must be made; perhaps the system architect will be the lead engineer and the requirements engineer will also be assigned the tasks of system integration and testing despite lacking the desired level of skill and experience (i.e., competency level) in integration and testing.

Developing the Team

Before a team that performs SE can be effective, it needs to establish its own identity, norms, and culture. The well-known four stages of *"forming, storming, norming, performing"* (Tuckman 1965, 384-399) indicate that a SE team needs time to form, for the members to get to know and understand each other as well as the tasks to be performed, and to work out how best to work together. It is also important that care is taken to ensure, to the greatest extent possible, assignment of roles and responsibilities that would allow SE team members to satisfy their individual goals (Fraser 2010).

The *cost and time to cohesion* can be minimized by good selection and management of the SE team, consistent training across the business so that team members have a common framework of understanding and language for their work, good "infostructure" to allow easy and useful sharing of information, and shared behavioral norms and values. Conversely, in cross-site, inter-company and international SE teams, more time must be allowed for team formation. SE teams are more effective if attention is given to ensuring that each member's work satisfies their individual goals as well as the team and organizational objectives (Fraser 2010).

According to Stephenson and Weil (1992), capable people are:

> *those who know how to learn; are creative; have a high degree of self-efficacy, can apply competencies in novel as well as familiar situations; and work well with others. In comparison to competency, which involves the acquisition of*

> *knowledge and skills, capability is a holistic attribute.*

The results of a survey by Steward Hase (2000) concluded that the following are significant contributors to the human elements of capability:

- Competent People
- Working in Teams
- Visible Vision and Values
- Ensuring Learning Takes Place
- Managing the Complexity of Change
- Demonstrating the Human Aspects of Leadership
- Performing as Change Agents
- Involving People in Change
- Developing Management Talent
- Committing to Organizational Development

These attributes of human capability apply to all members of an organization, including systems engineers, both as individuals and as members of project teams.

DeMarco and Lister (1999) discuss "teamicide" techniques by which management, perhaps unintentionally, practices *sure fire techniques to kill teams*. Teamicide techniques include

- physical separation of team members
- fragmentation of time
- unrealistic schedules
- excessive overtime

Methods for developing and improving SE capabilities within teams include building cohesive teams, conducting pilot projects, participating in and studying post-mortem analyses, and preparing and examining lessons learned. Members of a cohesive systems engineering team have a strong sense of commitment to the work and to the other team members. Commitment creates synergy, which results in performance greater than the sum of the performance of the individual team members.

Some key indicators of a cohesive systems engineering team (Fairley 2009, 411) are:

- clear understanding of systems engineering roles and

responsibilities
- shared ownership of systems engineering work products
- willingness of systems engineers to help one another and to help other project members
- good communication channels among systems engineers and with other project elements
- enjoyment of working together

Negations of these indicators—the hallmarks of a dysfunctional team—are:

- confusion of systems engineering roles and responsibilities
- protective ownership of systems engineering work products
- unwillingness to help one another
- absence of good communications among systems engineers and with other project elements
- personal dislike of one or more other systems engineering team members

Techniques for building and maintaining cohesive systems engineering teams include:

- an appropriate number of systems engineering team members
- a correct mix of systems engineering competencies
- celebration of project milestones
- team participation in off-site events
- social events that include family members

Assessing the Team

Performance evaluation is most often conducted for individuals. Robbins (1998, 576) states the historic belief that individuals are the core building blocks around which organizations are built. However, it is also important to assess the team's capability and performance. To design a system that supports and improves the performance of teams, including SE teams, Robbins offers four suggestions:

1. Tie the SE team's performance and the overall project team's results to the organization's goals
2. Begin with the team's customer and the work process

the team follows to satisfy customer's needs

3. Measure both team and individual performance and compare them to organizational norms and benchmarks
4. Train the team to create its own measures

Robbins' approach can be applied in the context of SE:

1. Tie the SE and overall project team's results to the project's and the organization's goals. Use measures that apply to goals the team must achieve. For SE in particular, the team effort should be tied to the product or service which the organization seeks to deliver. The end product for the SE team should not be only the SE work products but the delivered products and services provided by the project. For more information on general SE assessment, see Systems Engineering Assessment and Control.
2. Consider the SE team's customers and more broadly the key stakeholders and the work processes that the SE team follows to satisfy customer needs. SE customers and stakeholders can be internal or external; the internal customers of systems engineering are the other project elements that depend on systems engineering work products and services, which can be evaluated for on-time delivery of quantity and quality. The process steps can be evaluated for waste and cycle time; i.e., efficiency and effectiveness.
3. Assess both individual and team performance. Define the roles of each SE team member in terms of the tasks that must be accomplished to produce the team's work products. For more information on individual assessment, see Assessing Individuals.
4. Finally, have the team define its own measures of achievement of goals. This helps all members of the team to understand their roles, while also building team cohesion.

As an example, NASA's Academy of Program/Project and Engineering Leadership (APPEL) provides a service where team performance is assessed and interventions are provided to the team for specific gaps in performance (NASA 2011). This performance enhancement service increases a project's probability of success by delivering the right support to a project team at the right time. APPEL offers the following

assessments:

- Project/Team Effectiveness — Measures effectiveness of a team's behavioral norms
- Individual Effectiveness — Measures effectiveness of an individual's behavioral norms
- Project/Team Process Utilization — Measures the extent of a team's utilization of key processes
- Project/Team Knowledge — Covers topics that NASA project personnel should know in order to perform in their jobs

The APPEL approach can be applied to assessing the performance of a SE team and individual systems engineers.

Further Techniques for Building Team Capability

Further techniques for developing SE capabilities within teams include conducting pilot projects, preparing post-mortem analyses, and participating in and studying lessons learned.

Pilot Projects

Pilot projects are an effective mechanism by which SE teams can build team cohesion, acquire new skills, and practice applying newly acquired skills to projects and programs. Pilot projects can be conducted for the sole purpose of skills acquisition, or they can be conducted to determine the feasibility of a proposed approach to solving a problem. Feasibility studies and acquisition of new team skills can be combined in proof-of-concept studies. Primary inhibitors to conducting SE pilot projects are the time required and diversion of personnel resources.

Post-Mortem Analysis

A post-mortem analysis identifies areas for improvement of SE performance in future projects and programs. Inputs to a post-mortem analysis include:

- personal reflections and recollections of project personnel and other stakeholders;
- email messages, memos, and other forms of

communication collected during a project or program;

- successful and unsuccessful risk mitigation actions taken; and
- trends and issues in change requests and defect reports processed by the change control board.

Team participation in a post-mortem analysis allows SE team members to reflect on past efforts, which can lead to improved team capabilities for future projects or, if the present team is being disbanded, improved individual ability to participate in future systems engineering teams.

Inhibitors for effective post-mortem analysis include failure to allocate time to conduct the analysis, failure to effectively capture lessons-learned, failure to adequately document results, reluctance of personnel to be candid about the performance of other personnel, and negative social and political aspects of a project or program. Mechanisms to conduct effective post-mortem analyses of SE projects include using a third-party facilitator, brainstorming, Strength-Weakness-Opportunity-Threat (SWOT) analysis, fishbone (Ishikawa) diagrams, and mind mapping.

Lessons Learned

Lessons learned in SE can be both positive and negative. Experiences gained and documented from past projects and programs can be an effective mechanism for developing and improving the capabilities of a team that performs SE tasks. Studying past lessons learned can aid in team formation during the initiation phase of a new project. Lessons learned during the present project or program can result in improved capabilities for the remainder of the present project and for future projects. Inputs for developing and documenting SE lessons learned include results of past post-mortem analyses plus personal recollections of the team members, informal *war stories*, and analysis of email messages, status reports, and risk management outcomes. Inhibitors for developing and using SE lessons learned include failure to study lessons learned from past projects and programs during the initiation phase of a project, failure to allocate time and resources to developing and documenting lessons learned from the present project or program, and reluctance to discuss problems and issues.

References

Works Cited

Brooks, F. 1995. *The Mythical Man-Month*. Anniversary Edition. Reading, MA, USA: Addison Wesley.

Curtis, B., W.E. Hefley, and S.A. Miller. 2001. *People Capability Maturity Model (P-CMM)*, version 2.0. Pittsburgh, PA, USA: Software Engineering Institute (SEI). CMU/SEI-2001-MM-01. Accessed April 24, 2013. Available: http://www.sei.cmu.edu/library/abstracts/reports/01mm001.cfm.

DeMarco, T., and T. Lister. 1999. *Peopleware: Productive Projects and Teams*, 2nd ed. New York, NY, USA: Dorset House.

Eisner, H. 2008. *Essentials of Project and Systems Engineering Management*, 3rd ed. Hoboken, NJ, USA: John Wiley & Sons.

Fairley, R.E. 2009. *Managing and Leading Software Projects*. Hoboken, NJ, USA: John Wiley & Sons.

Fraser, D. 2010. *Relationships Made Easy: How to Get on with The People You Need to Get on with...and Stay Friends with Everyone Else*. Worchestershire, UK: HotHive Books.

Hase, S. 2000. "Measuring Organisational Capability: Beyond Competence." Paper presented at Future Research, Research Futures: Australian Vocational Education and Training Research Association (AVETRA) Conference (2000). Accessed September 14, 2011. Available: http://www.avetra.org.au/abstracts_and_papers_2000/shase_full.pdf.

INCOSE. 2010. *Systems Engineering Competencies Framework 2010-0205*. San Diego, CA, USA: International Council on Systems Engineering (INCOSE). INCOSE-TP-2010-003.

NASA. 2011. *Academy of Program/Project and Engineering Leadership (APPEL), NASA APPEL Performance Enhancement*. Accessed September 15, 2011. Available: http://appel.nasa.gov/team/request-support/.

Robbins, S.P. 1998. *Organizational Behavior: Concepts,*

Controversies, Applications, 8th ed. Upper Saddle River, NJ, USA: Prentice Hall. p. 576.

Stephenson, J. and S. Weil. 1992. *Quality in Learning: A Capability Approach in Higher Education*. London, UK: Kogan Page.

Torres, C., and D. Fairbanks. 1996. *Teambuilding: The ASTD Trainer's Sourcebook*. New York, NY, USA: McGraw-Hill.

Tuckman, B. 1965. "Developmental Sequence in Small Groups." *Psychological Bulletin*. 63 (6): 384-99.

Primary References

Brooks, F. 1995. *The Mythical Man-Month*. Anniversary Edition. Reading, MA, USA: Addison Wesley.

Curtis, B., W.E. Hefley, and S.A. Miller. 2001. *People Capability Maturity Model (P-CMM)*, Version 2.0. Pittsburg, PA, USA: Software Engineering Institute (SEI). CMU/SEI-2001-MM-01. Accessed on June 8, 2012. Available at http://www.sei.cmu.edu/library/abstracts/reports/01mm0 01.cfm.

DeMarco, T. and T. Lister. 1999. *Peopleware: Productive Projects and Teams*. 2nd ed. New York, NY, USA: Dorset House.

Eisner, H. 2008. *Essentials of Project and Systems Engineering Management*. 3rd ed. Hoboken, NJ, USA: John Wiley & Sons.

Fairley, R.E. 2009. *Managing and Leading Software Projects*. Hoboken, NJ, USA: John Wiley & Sons.

Hase, S. 2000. "Measuring Organisational Capability: Beyond Competence". Paper presented at Future Research, Research Futures: Australian Vocational Education and Training Research Association (AVETRA) Conference (2000). Accessed on June 8, 2012. Available at http://www.avetra.org.au/abstracts_and_papers_2000/sh ase_full.pdf.

INCOSE. 2010. *Systems Engineering Competencies Framework 2010-0205*. San Diego, CA, USA: International Council on Systems Engineering (INCOSE). INCOSE-TP-2010-003.

NASA. 2011. *Academy of Program/Project and*

Engineering Leadership (APPEL), NASA APPEL Performance Enhancement. Accessed on May 2, 2014. Available at http://appel.nasa.gov/team/request-support/.

Torres, C. and D. Fairbanks. 1996. *Teambuilding: The ASTD Trainer's Sourcebook*. New York, NY, USA: McGraw-Hill.

Additional References

Fasser, T. and D. Brettner. 2002. *Management for Quality in High Technology Enterprises.* New York, NY, USA: Wiley.

INEEL 2004. *A Project Management and Systems Engineering Structure for a Generation IV Very High Temperature Reactor.* Idaho Falls, ID, USA: Idaho National Engineering and Environmental Laboratory, NEEL/CON-04-02175. Accessed on September 14, 2011. Available at http://www.inl.gov/technicalpublications/Documents/2808490.pdf.

< Previous Article | Parent Article | Next Article >
SEBoK v. 2.10, released 06 May 2024