

Types of Systems

Types of Systems

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 Author: Rick Adcock, **Contributing Authors:** Brian Wells, Scott Jackson

This article forms part of The Nature of Systems knowledge area (KA). It provides various perspectives on system classifications and types of systems, expanded from the definitions presented in What is a System?.

The modern world has numerous kinds of systems that influence daily life. Some examples include transport systems; solar systems; telephone systems; the Dewey Decimal System; weapons systems; ecological systems; space systems; etc. Indeed, it seems there is almost no end to the use of the word “system” in today’s society.

This article considers the different classification systems which some systems science authors have proposed in an attempt to extract some general principles from these multiple occurrences. These classification schemes look at either the kinds of elements from which the system is composed or its reason for existing.

The idea of an engineered system is expanded. Four specific types of engineered system context are generally recognized in systems engineering: product system, service system, enterprise system and system of systems.



Contents

System Classification

Types of Engineered System

Products and Product Systems

Services and Service Systems

Enterprises and Enterprise Systems
Systems of Systems
Applying Engineered System Contexts
References
Works Cited
Primary References
Additional References

System Classification

A taxonomy is "a classification into ordered categories" (Dictionary.com 2011). Taxonomies are useful ways of organizing large numbers of individual items so their similarities and differences are apparent. No single standard systems taxonomy exists, although several attempts have been made to produce a useful classification taxonomy, e.g. (Bertalanffy 1968) and (Miller 1986).

Kenneth Boulding (Boulding 1956), one of the founding fathers of general system theory, developed a systems classification which has been the starting point for much of the subsequent work. He classifies systems into nine types:

1. Structures (Bridges)
2. Clock works (Solar system)
3. Controls (Thermostat)
4. Open (Biological cells)
5. Lower organisms (Plants)
6. Animals (Birds)
7. Man (Humans)
8. Social (Families)
9. Transcendental (God)

These approaches also highlight some of the subsequent issues with these kinds of classification. Boulding implies that physical structures are closed and natural while social ones are open. However, a bridge can only be understood by considering how it reacts to traffic crossing it, and it must be sustained or repaired over time (Hitchins 2007). Boulding also separates humans from animals, which would not fit into more modern thinking.

Peter Checkland (Checkland 1999, 111) divides systems

into five classes: natural systems, designed physical systems, designed abstract systems, human activity systems and transcendental systems. The first two classes are self-explanatory.

- **Designed abstract systems** - These systems do not contain any physical artifacts but are designed by humans to serve some explanatory purpose.
- **Human activity systems** - These systems are observable in the world of innumerable sets of human activities that are more or less consciously ordered in wholes as a result of some underlying purpose or mission. At one extreme is a system consisting of a human wielding a hammer. At the other extreme lies international political systems.
- **Transcendental systems** - These are systems that go beyond the aforementioned four systems classes, and are considered to be systems beyond knowledge.

Checkland refers to these five systems as comprising a “systems map of the universe”. Other, similar categorizations of system types can be found in (Aslaksen 1996), (Blanchard 2005) and (Giachetti 2009).

Magee and de Weck (Magee and de Weck 2004) provide a comprehensive overview of sources on system classification such as (Maier and Rechtin 2009), (Paul 1998) and (Wasson 2006). They cover some methods for classifying natural systems, but their primary emphasis and value to the practice of systems engineer is in their classification method for human-designed, or man-made, systems. They examine many possible methods that include: degree of complexity, branch of the economy that produced the system, realm of existence (physical or in thought), boundary, origin, time dependence, system states, human involvement / system control, human wants, ownership and functional type. They conclude by proposing a functional classification method that sorts systems by their process (transform, transport, store, exchange, or control), and by the entity on which they operate (matter, energy, information and value).

Types of Engineered System

The figure below is a general view of the context for any potential application of an engineered system life cycle.

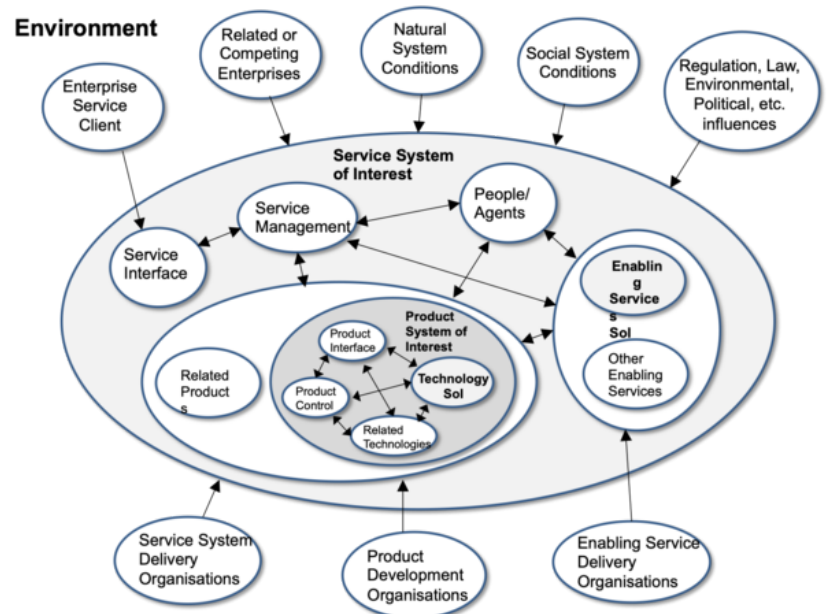


Figure 1: General types of Engineered System of Interest (Sol) (SEBoK original)

Figure 1 shows four general cases of system of interest (Sol) which might be the focus of a life cycle.

- A technology focused product system Sol embedded within one or more integrated products,
- An integrated multi-technology product system Sol used directly to help provide a service,
- An enabling service system Sol supporting multiple service systems
- A service system Sol created and sustained to directly deliver capability.

Products and Product Systems

The word product is defined as "a thing produced by labor or effort; or anything produced" (Oxford English Dictionary). In a commercial sense a product is anything which is acquired, owned and sustained by an organization and used by an enterprise (hardware, software, information, personnel, etc.).

A product system is an engineered system in which the focus of the life cycle is to develop and deliver products to an acquirer for internal or external use to directly support the delivery of services needed by that acquirer.

A product systems life cycle context will describe a technology focused Sol plus the related products, people and services with which the Sol is required to interact. Note, the people associated with a product system over

its life (e.g, operators, maintainers, producers, etc.) sit outside of the product SoI, since they are not delivered as part of the product. However, to develop a successful product, it is essential to fully understand its human interfaces and influences as part of its context. The product context will also define the service systems within which it will be deployed to help provide the necessary capability to the acquiring enterprise.

In a product life cycle, this wider context defines the fixed and agreed relationships within which the SoI must operate, and the environmental influences within which the life cycle must be delivered. This gives the product developer the freedom to make solution choices within that context and to ensure these choices fit into and do not disrupt the wider context.

A product life cycle may need to recommend changes to enabling services such as recruitment and training of people, or other infrastructure upgrades. Appropriate mechanisms for the implementation of these changes must be part of the agreement between acquirer and supplier and be integrated into the product life cycle. A product life cycle may also suggest changes in the wider context which would enhance the product's ownership or use, but those changes need to be negotiated and agreed with the relevant owners of the systems they relate to before they can be added to the life cycle outputs.

A more detailed discussion of the system theory associated with product systems can be found in History of Systems Science and an expansion of the application of systems engineering to service systems in the Product Systems Engineering KA in Part 4.

Services and Service Systems

A service can be simply defined as an act of help or assistance, or as any outcome required by one or more users which can be defined in terms of outcomes and quality of service without detail to how it is provided (e.g., transport, communications, protection, data processing, etc.). Services are processes, performances, or experiences that one person or organization does for the benefit of another, such as custom tailoring a suit; cooking a dinner to order; driving a limousine; mounting a legal defense; setting a broken bone; teaching a class; or running a business's information technology infrastructure and applications. In all cases, service involves deployment of knowledge and skills (competencies) that one person or organization has for

the benefit of another (Lusch and Vargo 2006), often done as a single, customized job. To be successful, service requires substantial input from the client and related stakeholder, often referred to as the co-creation of value (Sampson 2001). For example, how can a steak be customized unless the customer tells the waiter how the customer wants the steak prepared?

A service system (glossary) is an engineered system created and sustained by an organization that provides outcomes for clients within an enterprise. A service system context contains the same kinds of system elements as a product system context but allows greater freedom for what can be created or changed to deliver the required service.

A service system life cycle may deliver changes to how existing products and other services are deployed and used. It may also identify the need to modify existing products or create new products, in which case it may initiate a related product life cycle. In most cases the service developer will not have full freedom to change all aspects of the service system context without some negotiation with related system element owners. In particular, people and infrastructure are part of the service context and changes to how system elements are used to provide desired outcomes are part of the service life cycle scope.

The description of product system context above might be viewed as a special case of a service system context in which a specific product is created and integrated into a fixed service system by an organization and used by an enterprise directly related to the organization to provide a capability.

In a general service system context, it is not necessary to deliver all hardware or software products to the service provider. In some cases, some of the hardware, software or human elements may be owned by a third party who is not responsible for the service directly but provides enabling outputs to a number of such services. In other cases, the whole service may be provided by an organization that is completely separate to the enterprise which needs the service. Nor is it necessary for the exact versions of products or enabling services to be defined and integrated prior to service delivery. Some service system elements can be selected and integrated closer to the point of use. To allow for this late configuration of a service system, it will contain some method of discovery by which appropriate available elements can be found, and an overall service

management element to implement and direct each instance of the service system. The use of a service system approach gives greater freedom for acquirers in how they obtain and support all of the capital equipment, software, people, etc. in order to obtain the capabilities needed to satisfy users.

Services have been part of the language of systems engineering (SE) for many years, either as a way to describe the context of a product-focused life cycle or to describe commercial arrangements for the 'outsourcing' of product ownership and operation to others. The use of the term service system in more recent times is often associated with software configurable and information intensive systems, i.e.,

...unique features that characterize services - namely, services, especially emerging services, are information-driven, customer-centric, e-oriented, and productivity-focused. (Tien and Berg 2003, 13)

A more detailed discussion of the system theory associated with service systems can be found in History of Systems Science and an expansion of the application of systems engineering to service systems in the Service Systems Engineering KA in Part 4.

Enterprises and Enterprise Systems

An enterprise is one or more organizations or individuals sharing a definite mission, goals, and objectives to offer an output such as a product or service.

An enterprise system consists of a purposeful combination (network) of interdependent resources (e.g., people; processes; organizations; supporting technologies; and funding) that interact with each other (e.g., to coordinate functions; share information; allocate funding; create workflows; and make decisions) and their environment(s), to achieve business and operational goals through a complex web of interactions distributed across geography and time (Rebovich and White 2011).

Both product and service systems require an enterprise system to create them and an enterprise to use the product system to deliver services, either internally to

the enterprise or externally to a broader community.

Enterprise systems are unique, compared to product and service systems, in that they are constantly evolving; they rarely have detailed configuration controlled requirements; they typically have the goal of providing shareholder value and customer satisfaction, which are constantly changing and are difficult to verify; and they exist in a context (or environment) that is ill-defined and constantly changing.

While an enterprise system cannot be described using the general system context above, an enterprise may wish to create a model of the capabilities and services it needs to achieve its strategy and goals. Such a model can be extended to describe a baseline of service system and product system contexts related to its current capabilities, and to proposed future capabilities. These are referred to as enterprise architectures or enterprise reference architectures.

A more detailed discussion of the system theory associated with service systems can be found in History of Systems Science and an expansion of the application of systems engineering to service systems in the Enterprise Systems Engineering KA in Part 4. The notion of enterprises and enterprise systems also permeates Part 5 Enabling Systems Engineering.

Systems of Systems

A product, service or enterprise context can be defined as a hierarchy of system elements, with the additional definition of which elements are part of a SoI solution, which form the related problem context and which influence any life cycle associated with that context.

The additional concepts of Systems of Systems (SoS) or Federations of Systems (FoS) is used for some contexts. In terms of the general description in Figure 1 above, this would apply to any life cycle context in which elements within the SoI have independent life cycle relationships. This concept could apply to any of the life cycle contexts above, although it is of particular relevance to the service and enterprise contexts.

It is important to understand that the term SoS is an addition to the general concept of system hierarchy that applies to all systems. Maier examined the meaning of System of Systems in detail and used a characterization approach which emphasizes the independent nature of the system elements (Maier 1998, 268). Maier describes

both independence in how a system element operates (e.g. an element in the SoI also has its own separate mission or is part of another SoI) and in how an element is developed or sustained (e.g. an element is made available, modified or configured by a different organization to the one responsible for the rest of the SoI).

There are advantages to being able to have elements shared across a number of engineered systems and to being able to quickly create solutions to problems by combining existing engineered systems. As the technology to enable integration of independent systems becomes more common, this SoS approach becomes a common aspect of many SE life cycles.

Wherever system elements in an engineered system context have any degree of independence from the SoI life cycle, this adds a further complexity; specifically, by constraining how the resulting engineered system can be changed or controlled. This dimension of complexity affects the management and control aspects of the systems approach.

A more detailed discussion of the different system grouping taxonomies developed by systems science can be found in Part 4 Applications of Systems Engineering and an expansion of the ways we deal with SoS complexity can be found in the Systems of Systems KA in Part 4.

Applying Engineered System Contexts

From the discussions of product and service contexts above, it should be clear that they require similar systems understanding to be successful and that the difference between them is more about the scope of life cycle choices and the authority to make changes than about what kinds of systems they are.

These contexts are presented here as generalizations of the system engineering approach. All real projects may have both product and service system dimensions to them. In this general view of engineered systems, there is always an enterprise system directly interested in the service system context and either directly owning and operating any product systems and enabling services or gaining access to them as needed. This enterprise system may be explicitly involved in initiating and managing an engineering system life cycle or may be

implicit in the shared ownership of a problem situation. Any engineered system context may have aspects of the SoS independence discussed above. This may be part of the context in the wider system or environment or it may be related to the choice of elements within the Sol.

A real SE life cycle typically combines different aspects of these general contexts into a unique problem and solution context and associated acquirer and supplier commercial relationships. These must be identified by that life cycle as part of its SE activities. More details of these different life cycle contexts are given in part 2 and their applications to SE practice are expanded upon in Part 4.

A good example of a general description of the above is given by Ring (1998), who defines the overall context as the Problem Suppression System, describes a cycle by which an enterprise will explore its current needs, uses these to identify one or more life cycle interventions and relevant organizations, then conduct and deliver those life cycles and integrate their outputs into the PSS; the enterprise can then review the results in the environment and begin the cycle again.

This general systems approach is described in part 2 and used as a focus to identify areas of foundational knowledge. The current practices of SE described in the rest of the SEBoK reference these foundations as appropriate.

References

Works Cited

- Aslaksen, E.W. 1996. *The Changing Nature of Engineering*. New York, NY, USA: McGraw-Hill.
- Bertalanffy, L. von. 1968. *General System Theory*. New York, NY, USA: Brazillier.
- Blanchard, B.S., and W.J. Fabrycky. 2005. *Systems Engineering and Analysis*, 4th ed. Prentice-Hall International Series in Industrial and Systems Engineering. Englewood Cliffs, NJ, USA: Prentice-Hall.
- Boulding, K. 1956. "General systems theory: The skeleton of science" *Management Science*, vol. 2, no. 3, Aprilpp.197-208, 1956; reprinted in *General Systems, Yearbook of the Society for General Systems Research*, vol. 1, 1956.

- Checkland, P.B. 1999. *Systems Thinking, Systems Practice*. Chichester, UK: John Wiley & Sons Ltd.
- Dictionary.com, s.v. "Taxonomy." Accessed 3 December 2014. Available at: <http://dictionary.reference.com/browse/taxonomy>.
- Encyclopedia Britannica, s.v. "Service Industry." Accessed 3 December 2014. Available at: <http://www.britannica.com/EBchecked/topic/535980/service-industry>.
- DeRosa, J. K. 2005. "Enterprise systems engineering." *Air Force Association, Industry Day*, Day 1, 4 August 2005, Danvers, MA, USA.
- Giachetti, R.E. 2009. *Design of Enterprise Systems: Theory, Architectures, and Methods*. Boca Raton, FL, USA: CRC Press.
- Hitchins, D. 2007. *Systems Engineering: A 21st Century Systems Methodology*. Hoboken, NJ, USA: Wiley.
- Lusch, R.F. and S. L. Vargo (Eds). 2006. *The Service-Dominant Logic of Marketing: Dialog, Debate, and Directions*. Armonk, NY, USA: ME Sharpe Inc.
- Magee, C.L. and O.L. de Weck. 2004. "Complex System classification". Proceedings of the 14th Annual International Symposium of the International Council on Systems Engineering, Toulouse, France, 20-24 June, 2004.
- Maier, M. W. 1998. "Architecting principles for systems-of-systems". *Systems Engineering*, vol. 1, no. 4, pp. 267-84.
- Maier, M., and E. Rechtin. 2009. *The Art of Systems Architecting*, 3rd Ed. Boca Raton, FL, USA: CRC Press.
- Miller J. G. 1986. "Can Systems Theory Generate Testable Hypothesis?: From Talcott Parsons to Living Systems Theory" *Systems Research*, vol. 3, pp. 73-84.
- Paul, A.S. 1998. "Classifying systems." Proceedings of the 8th Annual International Council on Systems Engineering International Symposium, Vancouver, BC, Canada, 26-30 July, 1998.
- Rebovich, G., and B.E. White (eds.). 2011. *Enterprise Systems Engineering: Advances in the Theory and Practice*. Boca Raton, FL, USA: CRC Press.
- Ring, J., 1998. "A value seeking approach to the

engineering of systems." Proceedings of the IEEE Conference on Systems, Man, and Cybernetics. p. 2704-2708.

Sampson, S.E. 2001. *Understanding Service Businesses*. New York, NY, USA: John Wiley.

Tien, J.M. and D. Berg. 2003. "A case for service systems engineering." *Journal of Systems Science and Systems Engineering*, vol. 12, no. 1, pp. 13-38.

Wasson, C.S. 2006. *System Analysis, Design and Development*. Hoboken, NJ, USA: John Wiley and Sons.

Primary References

Checkland, P. B. 1999. *Systems Thinking, Systems Practice*. Chichester, UK: John Wiley & Sons.

Magee, C. L., O.L. de Weck. 2004. "Complex system classification." Proceedings of the 14th Annual International Council on Systems Engineering International Symposium, Toulouse, France, 20-24 June 2004.

Rebovich, G., and B.E. White (eds.). 2011. *Enterprise Systems Engineering: Advances in the Theory and Practice*. Boca Raton, FL, USA: CRC Press.

Tien, J.M. and D. Berg. 2003. "A case for service systems engineering". *Journal of Systems Science and Systems Engineering*, vol. 12, no. 1, pp. 13-38.

Additional References

None.

< Previous Article | Parent Article | Next Article >

SEBoK v. 2.10, released 06 May 2024

Retrieved from

"https://sandbox.sebokwiki.org/index.php?title=Types_of_Systems&oldid=71215"

This page was last edited on 2 May 2024, at 22:03.