

Requirements engineering for human activity systems

Jonnro Erasmus Pr. Eng.
Council for Scientific and Industrial Research
Email: jonnro.erasmus@gmail.com
Tel: +27 82 923 6478

Copyright © 2014 by Jonnro Erasmus. Published and used by INCOSE with permission.

Abstract. The paper explores the use of requirements engineering techniques and principles for the description of social systems. The scope is limited to only defining requirements for the system-of-interest, thus excluding any attempt to design the system. The requirements engineering domain is briefly investigated to identify the applicable techniques and principles. Systems theory and soft systems methodology are investigated to understand how social systems are currently described and analysed. To demonstrate how requirements engineering and social systems may be unified, the example of an illustrative health care system is used. The paper concludes with the finding that it is a valid approach to use requirements to describe desired characteristics of social systems, for the purpose of giving context and enabling the systems engineering of elements of social systems.

Introduction

We perceive something as a problem because it isn't currently as we want it to be (Soanes and Stevenson 2008). For example, poverty is considered a problem because poor people can't afford what is considered basic needs, such as health care, food and sanitation. However, as is done far too often in business endeavours, the problem is not fully understood and defined before possible solutions are considered (Gibson Jr. and Dumont 1995). Furthermore, it has been shown that one of the leading causes for project failure is incomplete requirements (Pinto and Mantel, Jr. 1990) (Hooks 1993) (Kasser and Schermerhorn 1994) (Carson 2001). Thus, it can be argued that initiatives to overcome prevalent social challenges have a very low chance of success if the actual problem is not fully understood and analysed first.

Good engineering practice dictates that the requirements of the eventual solution should be defined, developed, decomposed and understood at an appropriate level of detail before the functional and physical characteristics of the solution are defined (U.S. DOD 1969). The practice of requirements engineering aims to ensure that such requirements are properly developed and managed, by applying well defined rules and techniques (Hull, Jackson and Dick 2011). For example, a requirement should not be unrealistic and each requirement must be accompanied by criteria that define when that requirement is satisfied (Tran and Kasser 2005). This paper explores whether the techniques of requirements engineering can be applied to the definition of social systems, in the hope that such definition will increase our understanding of the challenges and what it will take to overcome them.

In the first part, the practice of requirements engineering is studied to identify the relevant theory and techniques that may be applied to social systems. Secondly, social systems are investigated to understand the similarities between such systems and the more familiar product systems. Finally, through the application of an example, the paper concludes with findings and recommendations.

Requirements Engineering

Hull et al. define requirements engineering as “the subset of systems engineering concerned with discovering, developing, tracing, analyzing, qualifying, communicating and managing requirements that define the system at successive levels of abstraction (Hull,

Jackson and Dick 2011).” From this definition, it is clear that requirements engineering does not only concern the development of requirements, but covers the entire lifecycle of requirements from the initial identification to the eventual validation thereof.

It is a common misconception that requirements are only an input into the design and development process and that the product is evaluated according to those original input requirements. However, new requirements are constantly discovered and derived during the development, realisation, utilisation and disposal of products. Figure 1 illustrates that the design process actually transforms input requirements into output requirements, to be satisfied by the production, utilisation and support of the product.

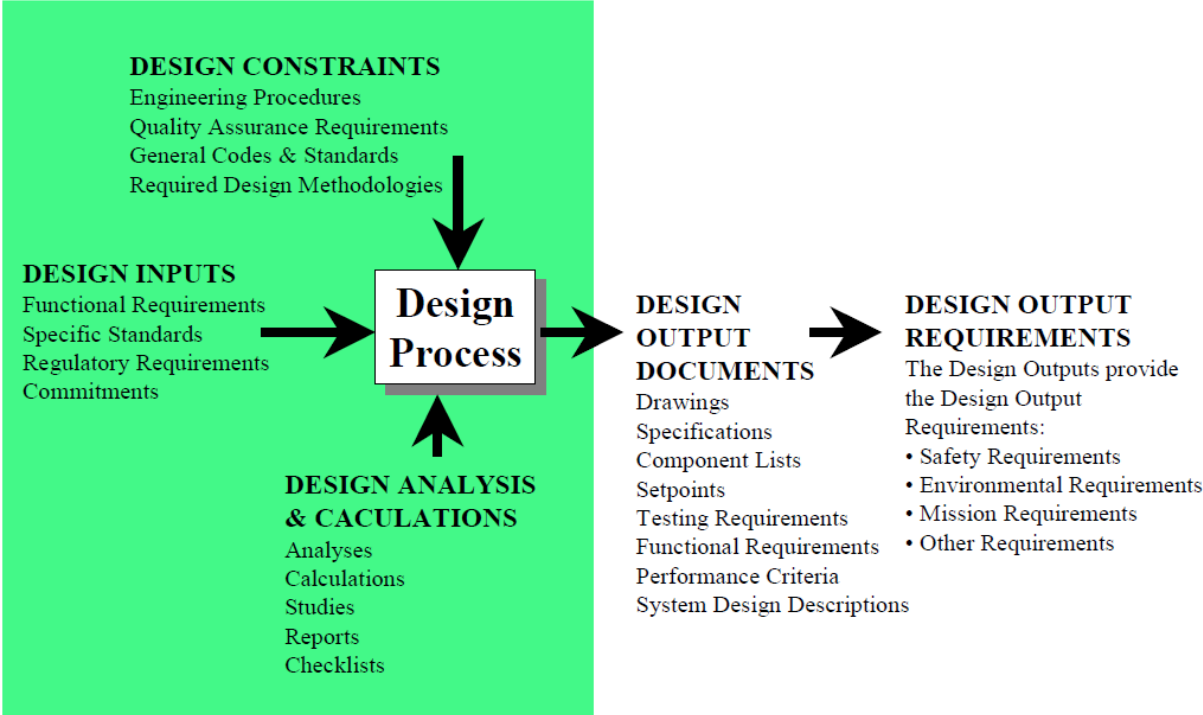


Figure 1: Design input and output requirements (U.S. Department of Energy 2003)

The set of requirements is not only subject to change, but it also grows throughout the lifecycle of the system. Requirements engineering is commonly divided into two parts, corresponding to the generation of requirements and the management thereof. Figure 2 shows a hierarchical decomposition of requirements engineering, with lower level activities to illustrate the differences.

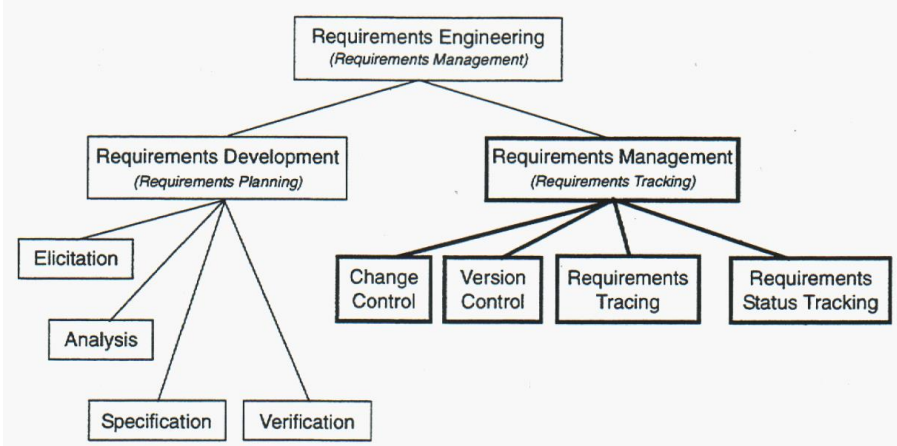


Figure 2: Decomposition of the requirements domain (Edwards, Harris and Eaton 2001)

Requirements analysis. Analysis is the act of partitioning something complex in smaller parts, to improve understanding of those parts and how they relate to each other (IEEE 1998). Therefore, requirements analysis is the activity which partitions input requirements and produces simpler output requirements with relationships between those requirements. Practically, requirements analysis does not occur only once during the systems engineering process, but is performed every time a need is stated or a new requirement is derived during the engineering design effort (Blanchard 2003). The output from requirements analysis is system requirements, including functional, performance, physical, operational and support requirements. These system requirements are allocated to system elements during the functional analysis and allocation activity of the systems engineering process (U.S. DOD 1994).

Requirements management. Requirements are constantly generated by the systems engineering process, whether due to emerging properties of the system or from derived requirements, throughout the system life cycle. Additionally, requirements will also often change as the design definition increases over time, resulting in different versions of a requirement. It is critically important to organise and control the requirements defined and changed by the requirements analysis activities. Some of the requirements management activities are shown on the right-hand side of Figure 2. The following is a more complete list of activities included in requirements management:

- Plan for how requirements will be managed throughout the product life cycle;
- Evaluate necessary changes and control the impact thereof;
- Version control of individual requirements to ensure consistent use of the current requirements;
- Control changes to the set of requirements to evaluate and control changes to the product configuration;
- Maintain the traceability from the stakeholder requirements to the eventual configuration item which satisfies that requirement;
- Tracking and updating the status of individual requirements to show progress towards a satisfactory product (Edwards, Harris and Eaton 2001).

Requirements traceability. The tracing of requirements, as part of the list of activities above, refers to the act of establishing relationships between requirements and the eventual product. The objective is to show how each of the input requirements are satisfied by one or more system element of the product or service. The relationships between parent and child requirements are defined during the decomposition of requirements and then the requirements are allocated to functions and configuration items of the product. These relationships should be documented, whether in a simple table or using a relational database, to allow for reporting and review in the format of a requirements traceability matrix.

Requirements quality. In order to maximise the effectiveness of requirements engineering, it is important to ensure that all requirements, whether input or output, are defined and captured properly. The following five rules may be applied to ensure that each requirement is a good requirement:

- Need: it should be clear what the need is and why it exists;
- Verification: the requirement should include criteria for how satisfaction of that requirement can be proven;

- **Attainable:** the requirement should be possible within the allocated time and budget; if it is unknown whether the requirement is attainable, the necessary studies should be done;
- **Clarity:** a requirement should represent a single need and should be clear and concise;
- **Traceable:** all design output requirements should support an original stakeholder requirement and stakeholder requirements should be linked to a stakeholder (Requirements Working Group of INCOSE 1993).

Requirements structure. To improve the quality of requirements, it is worthwhile to use consistent language to write requirements. For requirements stated in natural language, Mavin et al. suggests the following generic syntax: <optional preconditions> <optional trigger> the <system name> shall <system response> (Mavin, et al. 2009). More specialised syntaxes are also offered for requirements that are ubiquitous, event-drive, state-drive, etc. Using such syntaxes is not only useful when actually defining requirements, but can also be used to parse a set of requirements to determine the completeness.

Furthermore, design output requirements should also be accompanied with information regarding the intent of that requirement. This complete description of the foundation of a requirement is called the design rationale and consists of the following two parts (Karl Stum 2002):

- **Design narrative:** a written description of the intent of the requirement and that which it attempts to achieve;
- **Design basis:** the calculations, assumptions and decisions which led to the output requirement, often referring to analysis and calculation reports.

Types of requirements. Due to the different ways requirements are originated, several requirement types exist as inputs to and outputs from the systems engineering process. Typically, the following requirements are elicited, derived and defined during the design of a complex product:

- **Stakeholder requirements:** express the expectations of the various stakeholders (business, owner, user, regulator, etc.) as good requirement statements (Hull, Jackson and Dick 2011);
- **System requirements:** specifies what the expected functional, performance and physical characteristics of the product (IEEE 2005);
- **Functional requirements:** a type of system requirement which specifies what the system should be capable of (IEEE 2005).
- **Non-functional requirements:** specifies physical characteristics of the system, such as mass, size, reliability, corrosiveness, etc.

Verification and validation. Having properly defined requirements enables the evaluation of the design and product to determine whether it complies to requirements. However, to perform such verification activities, it should be very clear how compliance to a requirement will be shown. It is advisable to create a requirements verification matrix listing the requirement statement, success criteria, method of verification and special tools necessary to perform the verification (NASA 2007). This information will also allow for proper planning and scheduling of the verification activities to be performed throughout the development and realisation of the product.

Measures of effectiveness. It is often found that multiple solutions may satisfy the input requirements. To select the alternative to be developed further and eventually realised, the stakeholder goals should be elicited and captured, over and above the definition of stakeholder requirements (NASA 2007). These goals and targets may be defined as measures of effectiveness, stating some desired characteristics of the system-of-interest. INCOSE defines measures of effectiveness as the quantification of how well a system, product or process achieves the objectives thereof (INCOSE 2010). These measures of effectiveness are usually expressed in a range, where the base requirement denotes the minimum and some ideal state the maximum. In addition to allowing direct comparison of alternative solutions, these measures of effectiveness may also improve the chances of actually satisfying customer expectations, by striving towards targets beyond the imperatives of the specified requirements. The following four steps may be followed to derive measures of effectiveness from requirements:

1. Consider the system intent to identify the performance that is of primary importance;
2. Extract from the requirement set those requirements most relevant to the primary performance output;
3. Identify the functional and physical characteristics of the system that may be traded off against the performance target; and
4. Identify possible reduction in effect on the environment.

Value model. Trade-offs between conflicting design goals and alternative designs may occur very frequently during the development of a product. To facilitate this decision making process, it may be advisable to create a model based on the value to be gained from pursuing some of the goals. Naturally, this value is derived from the perceived risks and opportunities inherent to trade-offs and allows for consistent and informed decision making. Such models are often named decision support frameworks or value models.

Table 1: Illustrative value model of a railway crossing system

Measure of Effectiveness	Min	Max	Priority	Points	Weight
Rail-road safety, APAs	1	0	2	80	24
Investment cost, \$k - RC	250k	150k	6	20	6
Sustainment cost, \$k/20 years - RC	250k	100k	4	40	12
Investment cost, \$k - RTA	20k	5k	9	1	0
Sustainment cost, \$k/20 years - RTA	20k	5k	8	1	0
Reduction from track speed, %	50	0	3	60	18
Reduction from road speed, %	50	0	7	10	3
Availability, %	99	1	5	20	6
Fail-safe reliability, 0-1	0.999	1	1	100	30
Aesthetic appeal, SAAI 1-10	1	10	10	1	0

Table 1 shows an illustrative value model of a railway crossing system (Halligan, Effectiveness evaluation and decision making 2014). The measures of effectiveness were identified to be those requirements that most drive the architectural design of the system. The “min” field of each MOE represents the bare minimum that the system must perform to satisfy the specified requirement. The “max” field represents an ideal goal, though it may be unattainable, that is strived towards to increase the value of the system to one or more of its

stakeholders. The priority and points columns are scored subjectively, based on stakeholder expectations. The weight is calculated to determine the relative importance of the individual MOEs. A design can then be subjected to this value model to calculate a relative score of how it compares to other alternatives, or to identify design characteristics that can be improved to increase overall system value.

Systems Theory

General systems theory is the attempt to describe the complexity of practice through theoretical model-building, to allow for improved understanding (Boulding 1956). It studies the relationships between elements of a system to allow for the application of mathematics and scientific disciplines to express the system in a language more understandable and expressible. The term “system” covers a very wide spectrum, from tiny biological systems to galaxies, or even abstract systems such as mathematical models. Most systems share some common characteristics though, such as the following:

- Systems are abstracts of reality;
- Systems have structures, defined by its parts;
- Systems have behaviour, such as the processing of material, information or energy;
- The parts of the system have functional and structural relationships (Baianu 2011).

Social systems theory. Social systems theory is a specialisation and attempts to apply the general systems theory to the domain of human activity systems (Laszlo and Krippner 1998). Thus, it is the study of highly complex systems, such as the economy, by analysing the components that interact in the system. Modern social systems theory deals with the societal processes such as entropy, information processing and control in society (Bailey 1994). Social systems differ from the more familiar engineered systems most markedly due to the extent of human activity in social systems. However, it is difficult to truly distinguish between the two, as most engineered systems have some extent of human interaction, which must be catered for in the system design. As the amount of human activity in a system increases, the difficulty to differentiate between the system-of-interest and its environment also increases (Jordan 1998). Social systems theory is mostly concerned with understanding the decision making of the actors present in the system.

Soft Systems Methodology. The general theory of systems, in its original incarnation, did not succeed in its quest to unify the sciences and provide a language to combine the problems of the different disciplines (Checkland, *Soft Systems Methodology: A Thirty Year Retrospective* 2000). Soft systems methodology was thus formulated in an attempt to avoid the reductionism of natural science, by rather trying to evaluate human activity systems with its inherent complexity (Checkland, *Systems thinking, systems practice* 1981). Although not often used any more, the methodology originally consisted of seven steps, as follows:

1. Entering the problem situation;
2. Expressing the problem situation;
3. Formulating root definitions of relevant systems;
4. Building conceptual models of human activity systems;
5. Comparing the models with the real world;
6. Defining changes that are desirable and feasible;
7. Taking action to improve the real world situation (Checkland, *Systems thinking,*

systems practice 1981).

This methodology aims to make the good principles of systems engineering more applicable to less technical systems, such as enterprises and communities. This paper has a similar objective, though it is focussed entirely on the use of requirements engineering to describe the desired characteristics of a social system. Therefore, this paper may be positioned close to step two of the seven step methodology, but the purpose is to define what we expect of the system, irrespective of what the system currently achieves. This will allow systems engineers a more familiar point of entry to contribute to the design of the system and making the necessary changes.

System of Systems. Boulding referred to successive system levels to explain how larger systems are composed of smaller systems. He starts at the lowest level and refers to the geography and anatomy of the universe, then works up to level nine, named the transcendental level, with systems of unknowns and unknowables (Boulding 1956). Social systems are found at level eight in this hierarchy and are defined as a set of roles tied together with channels of communication. Therefore, social systems consist of lower level systems, each with its own purpose and characteristics. This corresponds closely to the collaborative system of systems category, as defined by the US Department of Defence (Office of the Deputy Under Secretary of Defense for Acquisition and Technology, Systems and Software Engineering 2008). In this type of system of systems, the central players collectively decide how to provide or deny service, thereby providing some means of enforcing and maintaining standards. Sage and Cuppan offer the following five characteristics of all systems of systems, regardless of the category:

- A system of systems is composed of systems that are independently useful;
- The component systems may be acquired and operated independent of the system of systems;
- The component systems are often geographically dispersed;
- The system of systems displays emerging behaviour that do not reside in any of its component systems;
- Systems of systems continuously evolve over time and are never complete or fully formed (Sage and Cuppan 2001).

The DoD Systems of Systems Engineering document offers guidance on the engineering of systems of systems, but Kossiakoff et al. argues that the basic tools that we have in systems engineering may not be sufficient to engineer a collaborative system of systems (Kossiakoff, et al. 2011). Next I present how one tool at least may be useful for social systems.

Requirements of a human activity system

Systems theory and soft systems methodology certainly provides a way to consider the relationships internal to a system and to its environment, but it does not describe the desired system as a set of requirements that can be analysed and allocated. The aim here is to make use of requirements engineering techniques when considering a human activity system.

As an example, a generic and illustrative health care system will be considered. The World Health Organisation states that a well-functioning health system responds to a population's needs and expectations by doing the following:

1. Improving the health of individuals, families and communities;
2. Defending the population against health threats;

3. Protecting people against the financial consequences of ill-health;
4. Providing equitable access to people-centred health care; and
5. Enable people to participate in decisions affecting their health and health system (World Health Organization 2010).

While the key components are rather unspecific, they do at least represent a very basic set of requirements of a good health care system. Various requirements analysis activities may then be applied in an attempt to improve the quality of the requirements. As an example, by performing requirements parsing analysis, we can improve the completeness and structure of the requirements and ensure that each requirement at least contains an actor, action and object of the action (Halligan, Requirements quality metrics: The basis of informed requirements engineering management 1993). Table 2 shows the results of applying a requirements structural template to the second of the five requirements.

Table 2: Requirements parsing template

Element of requirement	Text
Actor	The health care system
Conditions of action	
Action	shall defend
Object of action	the population
Constraints of action	
Source of object	
Destination of action	against health threats
Other	

When evaluated with this template, the quality of the requirement is fairly good. However, the conditions and constraint of action are glaringly absent. It does not give indication of how well or under what circumstances it defends people against health threats. The same exercise may be performed with the other four requirements and the same result will be found. By adding clarifying constraints to the action, we may vastly improve the requirement in terms of clarity and specificity, thus improving our understanding of what is expected from a health care system.

In 2000, the United Nations agreed to a set of Millennium Development Goals, targeting realistic improvement across several different social systems. The following seven indicators related to health systems can be extracted from the millennium development goals:

1. Under-5 mortality (all causes);
2. Maternal mortality;
3. HIV prevalence;
4. Measles vaccination coverage by 12 months of age;
5. Deliveries attended by skilled health professionals;
6. Tuberculosis (TB) treatment success rate under DOTS;
7. General government health expenditure (GGHE) per capita;
8. Private expenditure on health per capita;
9. Density of health workforce by 1000 population;

10. Smoking prevalence (15 years of older) (Human Metrics Network 2012).

For HIV prevalence, the objective is to have halted and begun to reverse the spread of HIV/AIDS by 2015 (United Nations 2010). This objective provides some performance criteria, in terms of magnitude and duration. An improved requirement may be specified, for example: The health care system shall halt and reverse the spread of HIV/AIDS by 2015. Thus, the original requirement will be separated into several requirements for the different health threats. Such improved requirements may lead to new insights of how the health care system should function and the roles of the different actors therein.

The requirements analysis activity may be continued by defining a value model for the health care system. Such a model may assist with the various decisions to be made about the design of the health care system. Table 3 shows an example of such a value model, by using the ten health care related goals of the millennium development goals.

Table 3: Illustrative health care system value model

Goal	Min	Max	Priority	Points	Weight
Under-5 mortality (all causes)	40%	0%	2	100	18
Maternal mortality	30%	0%	1	90	16
HIV prevalence	50%	0%	4	60	11
Measles vaccination coverage by 12 months of age	50%	100%	7	80	14
Deliveries attended by skilled health professionals	60%	100%	8	80	14
Tuberculosis (TB) treatment success rate under DOTS	30%	100%	3	50	9
General government health expenditure (GGHE) per capita	\$10k	0	10	10	2
Private expenditure on health per capita	\$10k	0	6	30	5
Density of health workforce by 1000 population	1	10	5	20	4
Smoking prevalence (15 years of older)	70%	0	9	40	7

The minimum values are entirely illustrative, but would represent the bare minimum that is expected of the health care system. The maximum is the ideal situation and the other fields may be calculated to allow for trade-offs and decision making. The priority and points fields are also illustrative, to allow calculation of weights and completion of the model. Such a model, with agreed values, can be used to make decisions regarding the configuration of the health care system.

Conclusion

Several techniques and methods exist to understand and analyse human activity systems, but these techniques rely on analysis of the complexity of the system to identify possible improvements to be made. This paper shows that requirements engineering techniques and processes may be used to first understand what is expected of a social system, before attempting to design a system that will meet those expectations. It further shows that system effectiveness evaluation may be used, by defining measures of effectiveness for the system, to make trade-offs between alternative solutions. Such trade-offs do not have to be a simple selection between multiple options, but the value model may be used to intelligently compare the alternatives and see how different design characteristics may be incorporated in other solutions. Furthermore, the value system may be used to inform which design adjustments of a possible solution is most likely to result in system effectiveness improvements.

The following insights were gained from applying requirements engineering to a human activity system:

- During the definition of system requirements, it is crucially important to clearly define the scope of the system and the interfaces to its environment.
- It may be beneficial to manage the entire lifecycle of the requirements, perhaps indicating the successful societal improvement and to identify the changes responsible for the success.
- As social systems evolve over time, it may be found that new requirements emerge, whether in the form of new possibilities or some other cause.
- Defining the requirements of a social system and establishing a value model are effective ways of reaching agreement between the various stakeholders about what is of greatest value in such a system.

Future work. If requirements are only defined, it will not have a direct impact on any of our social systems. However, it may improve our understanding of these systems and more importantly, what we want from these systems. It may also enable further analysis and perhaps even improve the ability to apply other engineering techniques to social systems. For example, by defining weighted values to the goals of a system, it is possible to apply operations research techniques to calculate the closest-to-ideal trade-off between a few design parameters.

References

- “Complex Systems Theory and Biodynamics.” In *Complexity, Emergent Systems and Complex Biological Systems*, by Prof Dr I. C. Baianu, edited by Prof Dr I. C. Baianu. 2011.
- Bailey, Kenneth D. *Sociology and the New Systems Theory*. Albany, N.Y.: State University of New York, 1994.
- Blanchard, Benjamin S. *System Engineering Management*. Hoboken, New Jersey: John Wiley & Sons, Inc., 2003.
- Boulding, Kenneth E. “General systems theory: The skeleton of science.” *Management Science* (INFORMS) 2, no. 3 (1956): 197–208.
- Carson, R. S. “Keeping the Focus During Requirements Analysis.” *Proceedings of the 11th International Symposium of the International Council on Systems Engineering*. Melbourne, Australia: INCOSE, 2001.
- Checkland, Peter. “Soft Systems Methodology: A Thirty Year Retrospective.” *Systems Research and Behavioral Science* (John Wiley & Sons, Ltd.), November 2000: S11-S58.
- . *Systems thinking, systems practice*. Hoboken: John Wiley, 1981.
- Edwards, H C, Pam W Harris, and Shelley M Eaton. *SIERRA Requirements Management Process Overview*. SAND Report, Sandia Corporation, Albuquerque: Sandia National Laboratories, 2001.
- Gibson Jr., G. Edward, and Peter Richard Dumont. *Project Definition Rating Index (PDRI) for Industrial Projects*. Research Report, Construction Industry Institute, Austin: Construction Industry Institute, 1995.
- Halligan, Robert J. “Effectiveness evaluation and decision making.” *Systems Engineering course notes*. Project Performance (Australia), 2014.
- . “Requirements quality metrics: The basis of informed requirements engineering management.” *Complex Systems Engineering Synthesis and Assessment Technology Workshop*. Calvados: Naval Surface Warfare Center, 1993. 9-14.

- Hooks, Ivy. "Writing good requirements." *Proceedings of the 3rd NCOSE International Symposium*. Arlington, Virginia: INCOSE, 1993.
- Hull, Elizabeth, Ken Jackson, and Jeremy Dick. *Requirements Engineering*. London, England: Springer-Verlag London Limited, 2011.
- Human Metrics Network. *Country Health Information Systems Assessment for UN Commission on Information and Accountability Priority Countries*. White paper, Geneva: World Health Organization, 2012.
- IEEE. *IEEE 1362-1998 Guide for Information Technology - System Definition - Concept of Operations (ConOps) Document*. IEEE Standard, The Institute of Electrical and Electronics Engineers, Inc., The Institute of Electrical and Electronics Engineers, Inc., 1998.
- IEEE. *IEEE Std 1220-2005 Standard for Application and Management of the systems engineering process*. IEEE Computer Society, The Institute of Electrical and Electronics Engineers, New York: The Institute of Electrical and Electronics Engineers, Inc., 2005.
- INCOSE. *Systems Engineering Handbook: A Guide for System Life Cycle Processes*. Version 3.2. Edited by C. Haskins. San Diego, California: International Council on Systems Engineering, 2010.
- Jordan, J. Scott, ed. *Systems Theories and A Priori Aspects of Perception*. Amsterdam: Elsevier Science B.V., 1998.
- Karl Stum, P.E. "Design Intent and Basis of Design: Clarification of Terms, Structure, and Use." *ASHRAE Technical Committee 7.9 - Building Commissioning*. Honolulu, 2002.
- Kasser, J. E., and R. Schermerhorn. "Determining Metrics for Systems." *The 4th Annual International Symposium of the NCOSE*. San Jose, CA: INCOSE, 1994.
- Kossiakoff, Alexander, William N. Sweet, Samuel J Seymour, and Steven M Biemer. *Systems engineering principles and practice*. Hoboken, New Jersey: Wiley-Interscience, 2011.
- Laszlo, Alexander, and Stanley Krippner. "Systems Theories: Their Origins, Foundations and Development." In *Systems Theories and A Priori Aspects of Perception*, by J. Scott Jordan, 47-76. Amsterdam: Elsevier Science B.V., 1998.
- Mavin, Alistair, Philip Wilkinson, Adrian Harwood, and Mark Novak. "Easy Approach to Requirements Syntax (EARS)." *17th IEEE International Requirements Engineering Conference*. Atlanta: IEEE, 2009. 317-322.
- NASA. *NASA Systems Engineering Processes and Requirements*. Office of the Chief Engineer, NASA, 2007.
- . *Systems Engineering Handbook*. Washington, D.C.: National Aeronautics and Space Administration, 2007.
- Office of the Deputy Under Secretary of Defense for Acquisition and Technology, Systems and Software Engineering. *Systems Engineering Guide for Systems of Systems*. Guide, Washington, DC: US Department of Defence, 2008.
- Pinto, J. K., and S. J. Mantel, Jr. "The causes of project failure." *IEEE Transactions on Engineering Management* (IEEE Technology Management Council) 37, no. 4 (1990): 269 to 276.
- Requirements Working Group of INCOSE. "Writing Good Requirements." Edited by Ivy Hooks. *Third International Symposium of INCOSE*. Houston: International Council on Systems Engineering, 1993.
- Sage, Andrew P., and Christopher D. Cuppan. "On the Systems Engineering and Management of Systems of Systems and Federations of Systems." *Information, Knowledge, Systems Management* 2, no. 4 (2001): 325 - 345.

- Soanes, Catherine, and Angus Stevenson, . *Concise Oxford English Dictionary*. Eleventh Edition. Oxford: Oxford University Press, 2008.
- Tran, Xuan-Linh, and Joseph E. Kasser. "Towards improving the recognition and correction of poor requirements." Edited by P. A. Lindsay. *SETE 2005: Systems Engineering/Test and Evaluation Conference*. Brisbane, Australia: Systems Engineering Society of Australia, 2005. 2-13.
- U.S. Department of Energy. *Configuration Management*. Washington, D.C.: U.S. Department of Energy, 2003.
- U.S. DOD. *MIL-STD-499 (USAF) System Engineering Management*. Military Standard, Washington, D.C.: U.S. Department of Defense, 1969.
- U.S. DOD. *MIL-STD-499B Systems Engineering*. Military Standard, Washington D.C.: U.S. Department of Defense, 1994.
- United Nations. *Millennium Development Goals: At a Glance*. White Paper, Geneva: UN Department of Public Information, 2010.
- World Health Organization. *Key components of a well functioning health system*. White Paper, Geneva: WHO, 2010.

Biography

Jonnro Erasmus has worked as an industrial and systems engineer in the electricity, transport and manufacturing industries. He has also had the responsibility of managing interdependent risks between major construction projects of both public and private enterprises. He specialises in the use of requirements and functional analysis to identify risks and opportunities in business systems and engineering organisations. Jonnro holds a bachelor of engineering from the University of Pretoria and a master of engineering from the University of Johannesburg. He is also a registered professional engineer with the Engineering Council of South Africa.