



SYSTEMS ENGINEERING NEWSLETTER

brought to you by

Project Performance International (PPI)

SyEN 61 – January 30, 2018

SyEN is an independent free newsletter containing informative reading for the technical project professional, with scores of news and other items summarizing developments in the field, including related industry, month by month. This newsletter and a newsletter archive are also available at www.ppi-int.com.

Systems engineering can be thought of as the problem-independent, solution technology-independent, life-cycle-oriented principles and methods, based on systems thinking, for defining, performing, and controlling the engineering effort within a technical project. The approach aims to maximize the benefit delivered to the enterprise, as influenced by the needs and values of the applicable stakeholders.

If you are presently receiving this newsletter from an associate, you may wish to receive the newsletter directly in future by signing up for this free service of PPI, using the form at www.ppi-int.com. If you do not wish to receive future Systems Engineering Newsletters, please unsubscribe by clicking on the link at the bottom of this email.

We hope that you find this newsletter to be informative and useful. Please tell us what you think. Email us at syen@ppi-int.info.

The views expressed in externally authored articles are those of the author(s), and not necessarily those of PPI or its professional staff.

IN THIS EDITION

Quotations to Open On

[Read More...](#)

Feature Article

- Object-Process Methodology, OPM ISO 19450 – OPCloud and the Evolution of OPM Modeling Tools, *by Dov Dori, Ahmad Jbara, Natali Levi, and Niva Wengrowicz*

[Read More...](#)

Article

- Leadership Values and Behaviors in Lean Organizations, *by Rainer Grau*
- Integrating Program Management and Systems Engineering, *by Dr. Ralph Young*

[Read More...](#)

Systems Engineering News

- Should Systems Engineering be Integrated into High School Curricula?
- A Systems Thinking Approach to Leadership
- A Systems Engineer as A Music Maestro
- Future Directions for SysML v2, *by Sanford Friedenthal and Larry Johnson*
- Object Management Group Issues RFP for SysML v2

[Read More...](#)

Featured Organization

- INCOSE's Critical Infrastructure Protection and Recovery (CIPR) Working Group

[Read More...](#)

Conferences and Meetings

[Read More...](#)

Some Systems Engineering-Relevant Websites

[Read More...](#)

Systems Engineering Publications

- Model-Based Systems Engineering with OPM and SysML
- Production Systems Engineering
- A Practical Guide to SysML: The Systems Modeling Language
- Wind Plant Organization and Systems Engineering Newsletter
- Architecting Spacecraft with SysML: A Model-based Systems Engineering Approach
- The Toyota Way to Service Excellence: Lean Transformation in Service Organizations

[Read More...](#)

Systems Engineering Tools News

- GENESYS 5.0
- Object Process Methodology

[Read More...](#)

Education and Academia

- Systems Engineering at the Wroclaw University of Science and Technology, Wroclaw, Poland
- Positions Available at the University of Luxembourg, Software Verification and Validation Lab

[Read More...](#)

Standards and Guides

- ISO/PAS 19450:2015: Automation Systems and Integration -- Object-Process Methodology

[Read More...](#)

Definitions to Close On

[Read More...](#)

PPI and CTI News

[Read More...](#)

Upcoming PPI and CTI Participation in Professional Conferences

[Read More...](#)

PPI and CTI Events

[Read More...](#)

QUOTATIONS TO OPEN ON

“Perform systems engineering like the casinos run their businesses. Make decisions on the basis of maximizing the value that will be delivered by the engineering, on the balance of probabilities, notwithstanding that any individual decision can produce a bad result.”

Robert John Halligan

“The important thing is to never stop questioning.”

Albert Einstein

“Companies cannot afford to just fix employees’ weaknesses, because fixing weaknesses only helps people prevent failure. It’s within the strengths that lie the true opportunities for growth and world-class performance.”

Marcus Buckingham

FEATURE ARTICLE

Object-Process Methodology, OPM ISO 19450 – OPCloud and the Evolution of OPM Modeling Tools

by

Dov Dori^{1,2}, Ahmad Jbara^{1,3}, Natali Levi¹, and Niva Wengrowicz¹

¹ Technion, Israel Institute of Technology

² Massachusetts Institute of Technology

³ Netanya Academic College

Abstract

We provide an overview of Object-Process Methodology – OPM ISO 19450, focusing on its underlying philosophy and principles. We then describe a recent experience of Volvo Cars Corporation in search of a modeling language, which culminated in the adoption of OPM. The focus of the paper moves to describing the evolution of past, present, and future OPM modeling tools with a real-life complex example of an OPM model of a system prepared with a leading Gas Turbine manufacturer for Gas Turbine Designing, aimed at cutting to one third the time to respond to an RFP. We conclude with a partial list of industrial applications and outlook for OPCloud development and its envisioned role in advancing MBSE.

Web sites: <http://esml.iem.technion.ac.il/> and <https://www.opcloud.tech/>

Email: dori@mit.edu

Copyright © 2017 by Dov Dori. All rights reserved.

Introduction

Object-Process Methodology, OPM (Dori, 2002; 2016), [ISO 19450 \(ISO 19450, 2015\)](#), is a holistic formal yet intuitive conceptual modeling approach to the development of complex socio-technical systems and knowledge management. OPM is both a language and a methodology. The OPM language part is defined by the specification of its syntax, semantics, and ontology. The OPM methodological part is the specification of recommended approach to using the OPM language for modeling complex systems, and is therefore applicable to model-based systems engineering (MBSE) and to capturing and managing scientific, engineering, and any other kind of knowledge.

The OPM language is *bimodal*—it uses both **graphics**—a single diagram kind, Object-Process Diagram (OPD), and natural language **text**—Object-Process Language (OPL), to represent knowledge at various, interconnected **levels of detail** about the major aspects of any system: function, structure, and behavior.

Due to its compact, general ontology, its single diagram kind, in stark contrast with UML (OMG UML 2.2, 2009) and SysML (OMG SysML 1.5, 2017), its bimodal representation, its ease of learning and implementation, and its generality, OPM is most suitable to serve as the underlying modeling language for MBSE.

The graphical and the textual OPM modalities are semantically equivalent and represent the same model. A set of hierarchically structured, interrelated Object-Process Diagrams (OPDs) constitutes the graphical model, and a set of automatically generated sentences in a subset of the English language constitutes the textual model expressed in Object-Process Language (OPL). In a graphical-visual model, each OPD consists of OPM elements, depicted as graphic symbols, sometimes with label annotation. The OPD syntax specifies the consistent and correct ways to manage the arrangement of those graphic elements. Using OPL, OPM generates the corresponding textual model for each OPD in a manner that retains the constraints of the graphical model. Since the syntax and semantics of OPL are a subset of English natural language, domain experts easily understand the textual model.

OPM as ISO 19450 (2015). In December 2015, the International Organization for Standardization (ISO) recognized OPM as [ISO 19450](#). Since then, leading industries in the domains of aerospace, car manufacturing, home appliance manufacturing, insurance, and defense have adopted OPM and are applying it for such objectives as managing knowledge, developing a new generation of product systems, and technology road mapping.

As noted in ISO 19450, OPM notation supports the conceptual modeling of systems via formal syntax and semantics. This formality serves as the basis for model-based systems engineering (MBSE) in general, including systems architecting, engineering, development, life cycle support, communication, and evolution. OPM facilitates a common view of the system under construction, test, integration, and daily maintenance, providing for working in a multidisciplinary environment. Furthermore, the domain-independent nature of OPM opens system modelling to the entire scientific, commercial and industrial community for developing, investigating, and analyzing manufacturing and other industrial and business systems within their specific application domains. This enables companies to merge and provide for interoperability of different skills and competencies into a common intuitive yet formal framework. Moreover, using OPM, companies can improve their overall, big-picture view of the system's functionality, flexibility in assignment of personnel to tasks, and managing exceptions and error recovery. System specification is extensible for any necessary detail, encompassing the functional, structural, and behavioral aspects of a system. As noted, especially since the recognition of OPM by ISO as ISO 19450, there is growing interest and demand for using OPM among large, international Fortune 500 companies, including technology, insurance, banking, and large-scale national agencies. Some request to remain in the shadow for now, as they deem it a competitive advantage to use OPM. The following case in point (Törmänen et al. 2017) exemplifies this trend.

In Search of a Modeling Language: The Volvo Experience. Working in Volvo Cars Corporation, Törmänen et al. (2017) searched for a language to generate an information model that takes into account different contexts of car design, from user requirements to optimization requirements. They required that the language be intuitive, self-explainable, easily learned by new users, and general, so it can model all the aspects of the product. They checked various systems engineering modeling languages including IDEF0 (Defense Acquisition University Press, 2001), [UML](#), [SysML](#) and [Modelica](#). They found that IDEF0 works for processes but not for structural diagramming. UML and SysML were found to be flexible multi-purpose languages, but had a “big drawback” of not being intuitive for non-experts, and their models “easily become quite messy.” Modelica can model behavior and structure, but it requires non-experts to make considerable effort to learn and understand. They then note: *“The Eureka moment came when we ran into Object Process Methodology (OPM) with its universal minimal ontology describing the world with things and things that happen, i.e. stateful objects that exist and processes that can transform them. It is much leaner than SysML and thereby becomes easier to understand and practice for newcomers.”* (p. 8).

The Evolution of OPM Modeling Tools

While it is possible to use a modeling language with a pencil and paper (or basic drawing software), it was clear from the outset that to do serious modeling and analysis, a computer-aided software engineering (CASE) tool that supports the language is mandatory. In this section, we describe the evolution of the three main tools developed for OPM modeling: the past OPA CASE Tool, the current OPCAT, and the future OPCLoud.

OPA CASE Tool – A Historical Perspective

As early as 1998, only three years after the first paper on OPM (Dori, 1995), then called OPA for Object-Process Analysis, was published, two undergraduate students at the Technion took the initiative and developed in Visual Basic an OPM modeling tool called **OPA CASE Tool**. Figure 1 shows a screenshot with the graphic user interface (GUI) of OPA CASE Tool version 1.7. The tool was quite rudimentary, with a single level of detail, and it did not generate OPL. The model shows the process **OPM Modeling**, which requires the two parts of **OPM – OPM Language** and **OPM Methodology**. The **Modeler** is the (human) agent and the result is the **OPM Model**, which comprises of **OPD Set** – the graphic modality of the model, and **OPL Spec** – the textual modality of the model. We use the same OPM model to demonstrate the next two OPM modeling tools, OPCAT and OPCLoud.

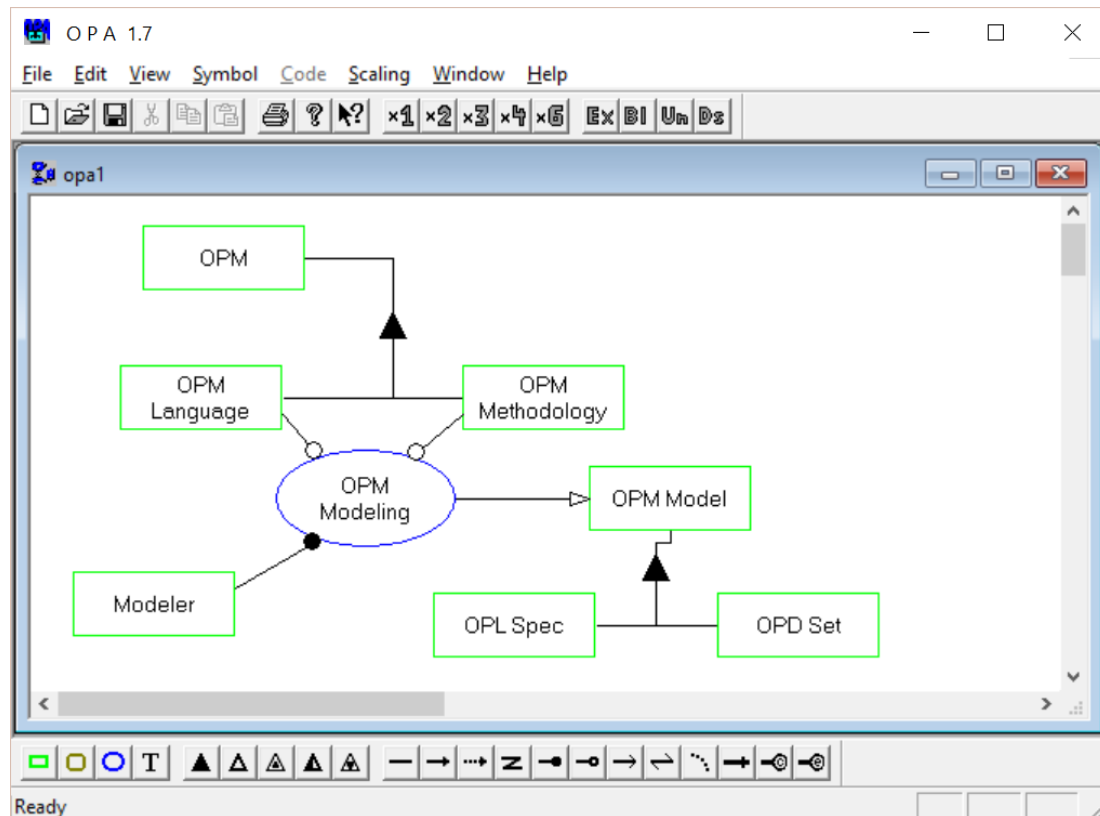


Figure 1. A screenshot of OPA CASE Tool 1.7 (1998)

OPCAT – The Horsepower of OPM Modeling

A couple of years after the introduction of OPA CASE Tool, we started developing OPCAT, short for Object-Process CASE Tool. Presently, OPM modelers use OPCAT (Dori et al., 2010), a freely [web-available](#) Java-based OPM modeling software desktop tool¹.

OPCAT was first presented in July 1998 at ECOOP'98 (Sturm & Dori, 1998). It may be interesting to read that the philosophy of OPM was already in place then:

“The underlying observation of the Object-Process paradigm is that everything in the universe of interest is either a process or an object. This opens the door for the possibility of modeling a system using a single model that faithfully defines and describes both its structure and behavior. These two major aspects of any system are represented without suppressing one another. Structural relations - primarily aggregation, generalization and characterization - and procedural relations, which model the behavior of the system over time, are seamlessly intertwined to provide a comprehensive understanding of the system.”

¹ OPCAT is downloadable freely from <http://esml.iem.technion.ac.il/opcat-installation/>

A few months later, we presented OPCAT in OOPSLA'98 (Dori & Sturm, 1998). Since UML, Unified Modeling Language (OMG UML 2008), had been adopted a year earlier, we discussed the main difference between the OPM approach and that of UML, namely the model multiplicity of UML vs. the model singularity in OPM:

“Object-Process Methodology (OPM) is a system development approach that integrates structure and behavior of the system within a single unifying model. The conventional wisdom has been that there is an inherent dichotomy between object- and process-oriented approaches, and that it is not possible to combine these two essential aspects of any system into one coherent integral frame of reference. This misconception has accompanied systems analysis to the extent that even the accepted UML standard (Booch and Rumbaugh, 1995, 1996) maintains the separation between structure and behavior, and spreads analysis activities across no less than eight types of models that use different diagram types.”

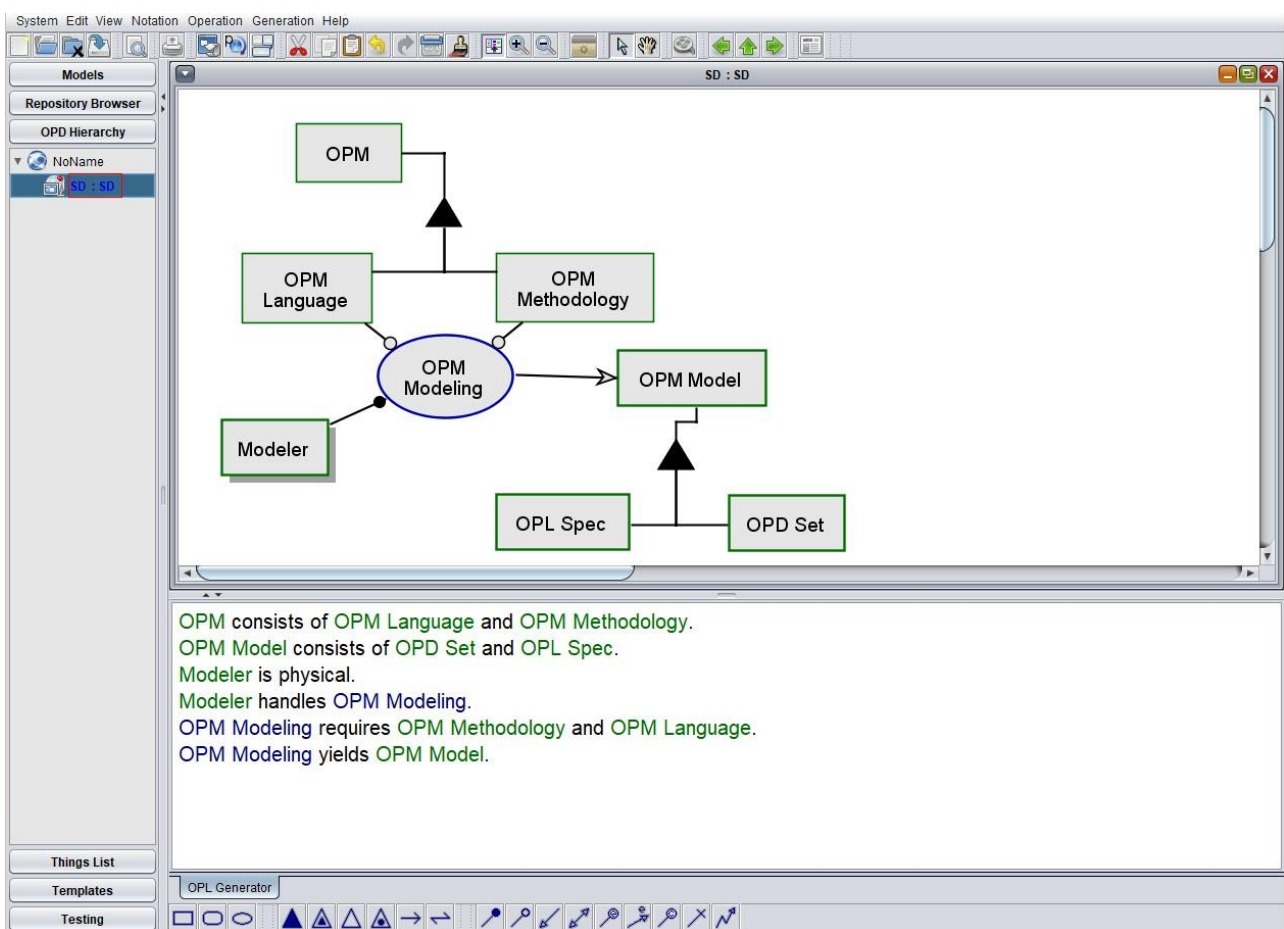


Figure 2. The GUI of OPCAT showing the process OPM modeling and the involved object set

Figure 2 shows the GUI of OPCAT 4.2 using the same system model as in Figure 1. Figure 3 is an OPM model of **Gas Turbine Designing** in a large corporation whose motivation was to reduce the **Proposal Production Rate** in response for an **RFP** from the current **6-18 months** to **4-6 months**. Figure 3 shows detail level 1 (SD1), where the process **Gas Turbine Designing** is in-zoomed, exposing four

subprocesses: **Researching**, **Conceptual Designing**, **Detailed Designing**, and **Design for Manufacturing**.

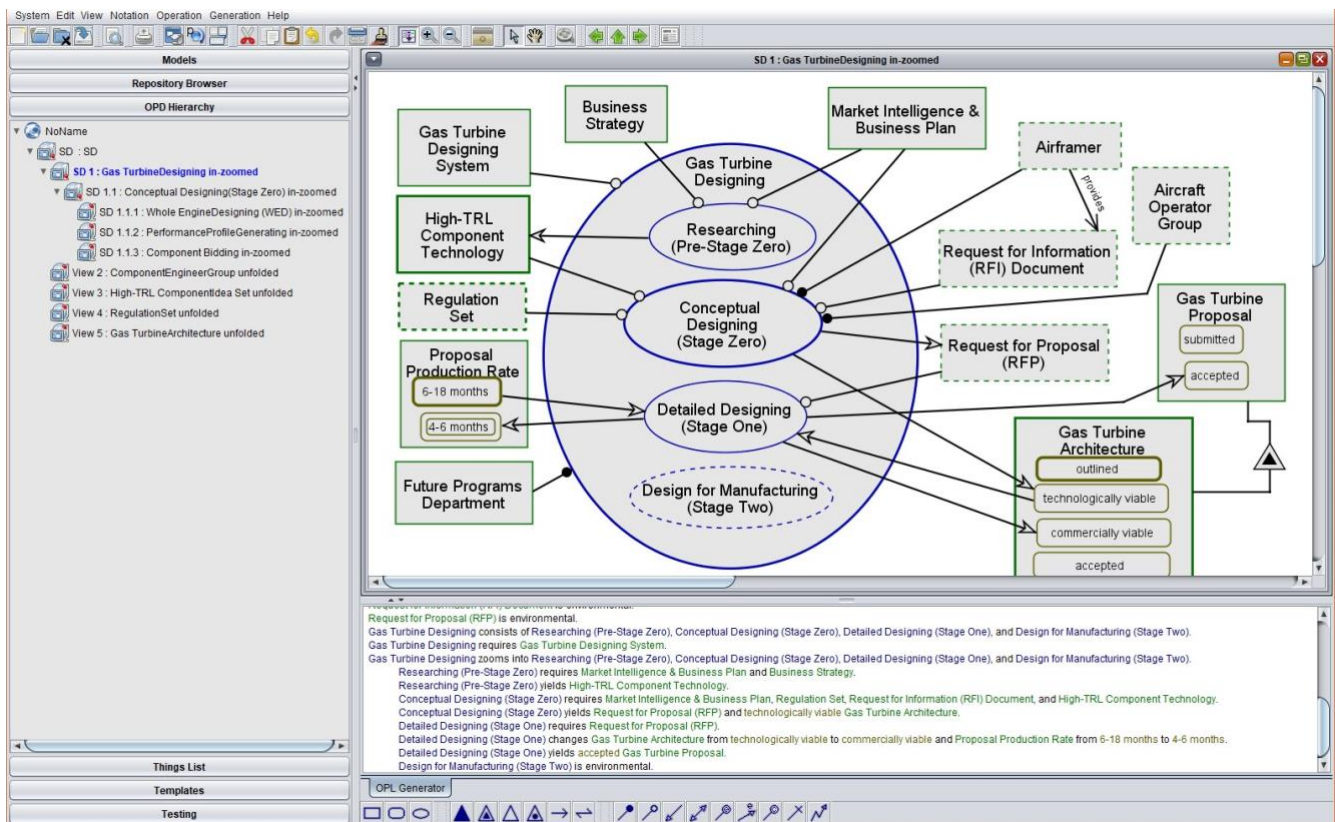


Figure 3. OPM model of Gas Turbine Designing – Detail level 1 (SD1) – Gas Turbine Designing in-zoomed

The automatically generated OPL sentence expressing the benefit of the system is in line 3 from the bottom of Figure 3:

Detailed Designing (Stage One) changes Gas Turbine Architecture from technologically viable to commercially viable and Proposal Production Rate from 6-18 months to 4-6 months.

This OPL sentence also expresses the fact that the same process changes **Gas Turbine Architecture** from **technologically viable** to **commercially viable**, as also expressed in the OPD.

OPCAT enables visual modeling and model simulation via concurrent, synchronous, and discrete time execution (Yaroker et al., 2013). The execution enables understanding the system's behavior and the detection of modeling errors by analyzing the mechanisms underlying the system under study. As an example, Figure 4 shows the process **Conceptual Designing (Stage Zero)**, which appeared as a subprocess in Figure 3, in-zoomed during OPCAT's animated simulation, showing **Whole Engine Designing (WED)** currently executing, as marked by the dark purple. This process is currently generating the object **Structural & Mechanical Requirements Set**. At the bottom of Figure 4, we can see the lifespan graph – a graph showing the state of each object and process at each step of the simulation. Currently, the animated simulation is at step 2 and **Whole Engine Designing (WED)** is marked as being active.

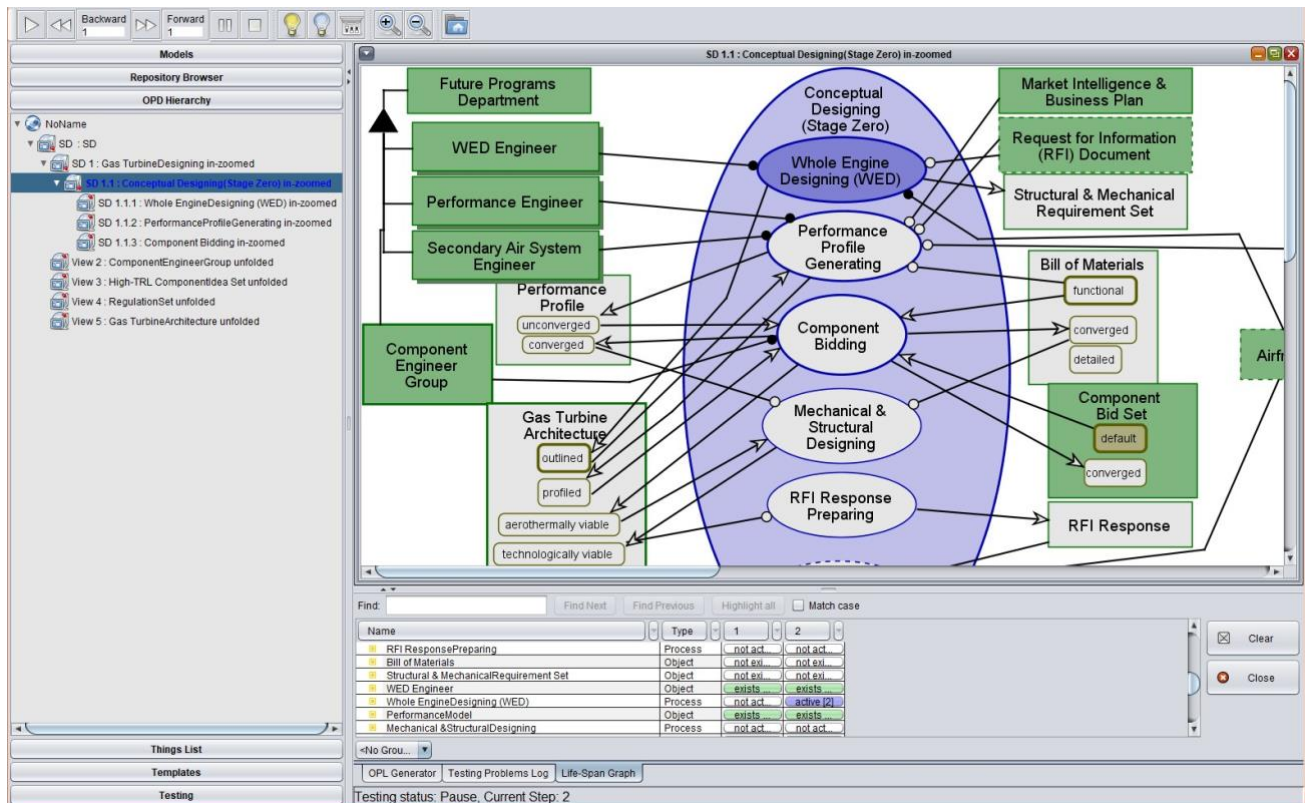


Figure 4. The process Conceptual Designing (Stage Zero) from Figure 3 in-zoomed during OPCAT's animated simulation, showing Whole Engine Designing (WED) currently executing

OPCloud – The Even Brighter Future of OPM Modeling

While OPCAT has been evolving and serving the OPM community for well over a decade, it is time for moving to the next generation of MBASE modeling tools, which is Cloud based. Indeed, last year we started developing OPCloud (<https://www.opcloud.tech/>), a collaborative OPM modeling tool, which is invoked by a URL so it requires no program installation. Figure 5 presents the current OPCloud GUI with the same model of OPM Modeling shown in OPCAT in Figure 2 and in OPA CASE Tool in Figure 1.

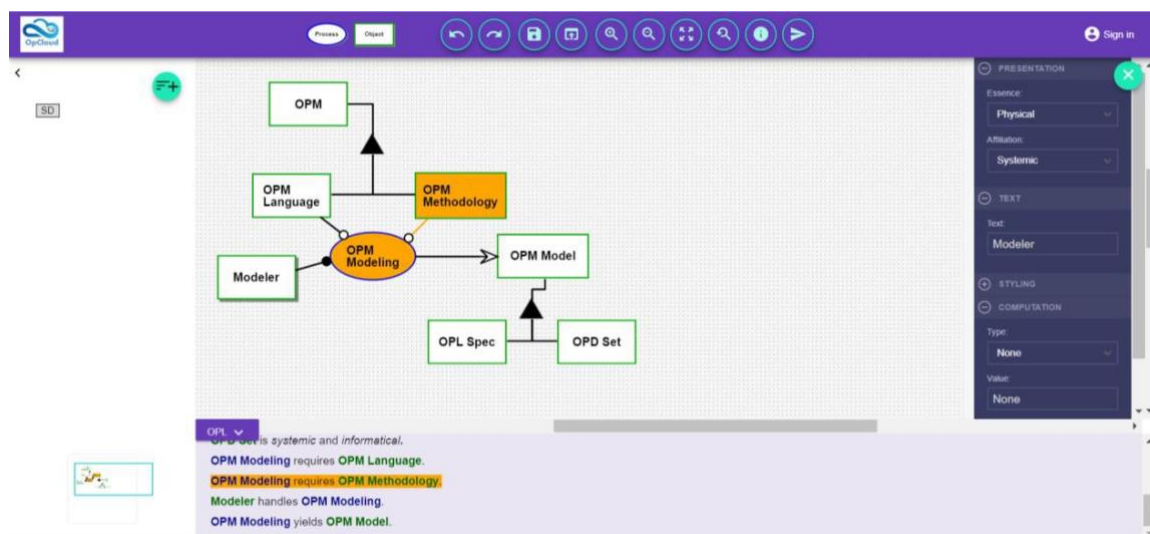


Figure 5. The current OPCloud GUI

Besides being cloud-based and enabling real-time collaboration, OPCloud has many novel features that distinguish it from OPCAT. One such feature is exemplified in Figure 5: While hovering over an OPL sentence, the OPD construct (a set of things, i.e., objects and processes, connected by a link) is highlighted. In Figure 5, the highlighted sentence causes the process OPM Modeling and the object OPM Methodology, as well as the instrument link connecting them, to also be highlighted in the OPD. The reverse shall be true as well: Hovering over a link shall highlight the OPL sentence **OPM Modeling requires OPM Methodology** and also present (or utter) it in a note in the OPD.

OPM uniquely distinguishes between physical and informatical things (objects or processes), enabling the generation of executable code from the informatical things and relations, alongside simulation of the hardware components, which the software expressed informatically in the model, controls. After conceptual modeling, the same OPCloud framework shall enable moving seamlessly back and forth between conceptual modeling and detailed design by gradually refining the system and introducing into it computational elements. The simulated execution shall combine the auto-generated software with the modeled hardware components, providing for a complete, system-level emulation. Developers will be able to gradually replace modeled hardware components by real prototypical parts, and the software modules that control them will evolve to reflect required needs as they are revealed from the hardware operation.

Epilogue – Testimonials

In response to a few questions, Major (Retired) Rafael Vila, United States Air Force, wrote on October 2, 2017:

How I learned about OPM?

As part of my duties as a Joint Test Officer in 2008, I was required to follow the Department of Defense Architecture Framework (DoDAF, Version 1.5) in support of test and evaluation activities. An essential part of this work was interpreting DoDAF artifacts to develop effective interoperability testing strategies. DoDAF was complex, and project managers consistently struggled to deliver valuable architectural data. ... As a result I began to explore alternatives, and in 2009 and I ran into Professor Dori's Object Process Methodology (OPM) text.

My experience with OPM, how it is, or can be, applied in industry or in academia, and specifically at my place or work

Almost 10 years later, I still ... apply OPM as a simple analysis method to consistently identify and model top level functionality, supporting instruments, and agents required to deliver mission critical outcomes.

What is the potential contribution of OPM to MBSE?

OPM can bring much needed simplicity and structure to improve the acquisition of IS. ... OPM can begin to bridge the gap between thinkers, doers, and talkers—their respective inputs are critical, but their interactions must be optimized.

What are the challenges in adopting MBSE in general and OPM in particular? Is OPM easier or more challenging than other approaches?

OPM is simple, elegant, and powerful in the right hands. The biggest challenge facing OPM is breaking into governments and private entities as Free and Open Source Software (FOSS).

Ricardo Moraes, a Systems Engineer, Aircraft Safety and Security for Embraer S/A Commercial Aviation Company, wrote on Sept. 26, 2017:

My experience with OPM, how it is, or can be, applied in industry?

I conduct an OPM application in the study scenario on the air management system and the experience was very good. OPM helped with Generalization-specialization and inheritance structural relations in the brainstorm and abstraction level of the functionalities analysis that help us understand in “big picture” and contributes to the robustness of the architecture.

What is the potential contribution of OPM to MBSE?

OPM contributions to MBSE in early phases ... to quickly understand and provide analysis and solutions. OPM increases productivity ... by minimizing unnecessary manual transcription of concepts when coordinating the work of a multidisciplinary team.

What are the challenges in adopting MBSE in general and OPM in particular?

I see OPM as complementary for brainstorming. OPM makes the context diagrams much easier than SysML and brings a global view where System Architects do not worry about details. However, OPM has just a few notations and this can create difficulty in sustaining the models during the development phases.

Summary and Conclusions

Since the first paper on OPM (Dori, 1995), two books (Dori 2002; 2016) and dozens of papers have been published in a large variety of multidisciplinary human-made socio-technical industrial and information systems, as well as biological systems. Domains that have applied OPM include molecular biology (Somekh et al., 2014), medicine (Wachs et al., 2014), satellite communication software development (Dori & Thipphayathethana, 2015), socio-technical systems (Osorio et al., 2011), data warehousing (Dori et al., 2008), agent technology (Sturm et al., 2010), and aerospace (Mordecai et al., 2016).

This partial list demonstrates the universality of the OPM underlying philosophy of minimalism and employing the smallest possible set of conceptual building blocks: Stateful objects and processes that transform them. Small is beautiful, and if we aspire to stand a chance in tackling the design and lifecycle support of ever more complex systems, the use of a simple yet powerful conceptual modeling language is a must. Equally important is the availability of an excellent modeling tool that acts as a smart apprentice of the system engineering employing MBSE. This principle is guiding us as we are developing OPCloud, and we expect it to be functional and available before the end of 2018. ISO 19450 OPM provides a complete guide to tool vendors, and we hope to see more OPM modeling tools developed in the near future, as OPM usage keeps increasing and becoming the conceptual modeling language and methodology of choice.

List of Acronyms Used in this Paper

<u>Acronym</u>	<u>Explanation</u>
CASE	Computer-Aided Software Engineering
MBSE	Model-Based Systems Engineering
OPM	Object-Process Methodology
OPD	Object-Process Diagram
OPL	Object-Process Language
OPCAT	Object-Process CASE Tool
OPCloud	Object-Process Cloud (CASE Tool)
OMG	Object Management Group
SysML	Systems modeling Language
UML	Unified Modeling Language

References

Dori, D. Object-Process Analysis: Maintaining the Balance between System Structure and Behavior. Journal of Logic and Computation, 5, 2, pp. 227-249, 1995.

Dori, D. [Object-Process Methodology – A Holistic Systems Paradigm](#), Springer Verlag, Berlin, Heidelberg, New York, 2002.

Dori, D. [Model-Based Systems Engineering with OPM and SysML](#), Springer, New York, 2016.

Dori, D. and Sturm, A. OPCAT– Object-Process Case Tool: an Integrated System Engineering Environment, LNCS 1543, pp 555-556, 1998.

Dori, D. Feldman, R. and Sturm, A. From conceptual models to schemata: An object-process-based data warehouse construction method. *Information Systems* 33 (6), pp. 567-593, 2008.

Dori, D., Linchevski, C. and Manor, R. OPCAT – A Software Environment for Object-Process Methodology Based Conceptual Modelling of Complex Systems. *Proc. 1st International Conference on Modelling and Management of Engineering Processes*, University of Cambridge, Cambridge, UK, Heisig, P., Clarkson, J., and Vajna, S. (Eds.), pp. 147- 151, July 19-20, 2010.

Dori, D. and Sturm, A. OPCAT– Object-Process Case Tool: An Integrated System Engineering Environment, [LNCS 1543](#), pp 555-556, 1998.

Dori D. and Thippayathethana, S. [Model-Based Guidelines for User-Centric Satellite Control Software Development](#). *International Journal of Satellite Communications and Networking*, 34: pp. 295–319, 2015.

[ISO 19450](#) Automation systems and integration – Object-Process Methodology. 2015. <https://www.iso.org/obp/ui/#iso:std:iso:pas:19450:ed-1:v1:en>. Retrieved June 2, 2016.

Mordecai, Y., Orhof, O., and Dori, D. [A Conceptual Modeling Framework for Interconnectivity and Interoperability](#). *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 1, 2016.

OMG SysML 1.5. Object Management Group, Systems Modeling Language Specification, Version 1.5, 2017. <http://www.omg.org/spec/SysML/About-SysML/>. Accessed Oct. 29, 2017.

OMG UML 2.2. Object Management Group, Unified Modeling Language Specification, Version 2.2, 2009. <http://www.omg.org/spec/UML/2.2/About-UML/>. Accessed Aug. 24, 2017.

Sturm, A. and Dori, D. Integrated System Engineering Environment with OPCAT - Object-Process CASE Tool. In J. Dockx, *Reflections on a Demonstration Chair*, *Proc. European Conference on Object Oriented Programming (ECOOP'98)*, July 1998. <http://ecoop98.vub.ac.be/demonstrations.html#D9>.

Sturm, A. Dori, D., and Shehory, O. An Object-Process-Based Modeling Language for Multi-Agent Systems. *IEEE Transactions on Systems, Man, and Cybernetics – Part C: Applications and Reviews*, 40 (2) pp. 227-241, 2010.

Somekh, J., Haimovich, G., Guterman, A., Dori, D., and Choder, M. Conceptual Modeling of mRNA Decay Provokes New Hypotheses. [PLOS ONE, Sept. 2014. PLoS ONE 9\(9\): e107085. doi:10.1371/journal.pone.0107085.](#)

Törmänen, M., Hägglund, A. Rocha, T. and Drenth, E. Integrating Multi-Disciplinary Optimization into the Product Development Process using Model-Based Systems Engineering (MBSE). NAFEMS World Congress, Stockholm, Sweden, June 2017.

Wachs, J. Frenkel, B. and Dori, D. Operation Room Tool Handling and Miscommunication Scenarios: An Object-Process Methodology Conceptual Model. Artificial Intelligence in Medicine, 62(3) pp. 153-163, 2014.

Yaroker, Y., Perelman, V. and Dori, D. An OPM Conceptual Model-Based Executable Simulation Environment: Implementation and Evaluation. Systems Engineering, 16(4), pp. 381-390, 2013.

ARTICLE

Leadership Values and Behaviors in Lean Organizations

by

Rainer Grau

Digitec Galaxus AG

Version 2.0 December 10, 2018

Abstract

Lean and agility are recognized success factors of companies. In particular, when we consider the largest companies by market cap², we find organizations that have implemented agile and lean consistently from top management throughout the entire organization. Research concerning lean and agility identifies that the company's culture and leadership³ are important prerequisites to establishing lean thinking and agility. An important question is: what concrete aspects of leadership support, foster, and consolidate change towards an agile and lean culture? This article discusses lean and agile principles and derived specific good practices and methods as building blocks for leadership values and behaviors in lean organizations

Email: rg@juopera.com

² See [1]. <http://www.visualcapitalist.com/chart-largest-companies-market-cap-15-years/>

³ See [2].

Introduction

It is widely accepted that it is not possible to create or form a specific company culture by direct means and activities. A company culture emerges as result of behaviors, measures, actions, rules, and conditions cultivated over time that are executed and active within a human work system that is, by definition, a complex system. Members of management are, by definition, more prominent in companies than employees without management responsibilities. Therefore, members of management occupy a position of power, provide a leadership role, and play an important part in the process of cultural change.

Leadership in lean and agile companies implies that members of management have to adapt behaviors and have to apply new or modified measures that support cultural change. This article discusses lean and agile principles and derived specific good practices and methods as building blocks for leadership values and behaviors in lean organizations.

It is important to point out that selection and implementation of good practices and methods does not by itself ensure success. The results from using good practices and methods always depends on the context of the specific situation.

Needs, Values, Principles and Practices

Starting with the agile manifesto [3], continuing with the extreme programming movement by Kent Beck [4], and supported and developed towards the spine model [5], the agile community has evolved an intellectual model that defines the concepts of needs, values, principles, and practices. The spine model explains this intellectual model as follows (citation from <http://spinemodel.info/explanation/introduction/>):

“Once you understand the reason [the human work] system exists in the first place, and the reason you want to be a part of the system (Needs), you can decide what to optimize for (Values). Once you have decided what you are optimizing for, you can decide what ecological levers are going to get you there (Principles). Once you have that, then decide how you are going to do it (Practices). And once you have done that, decide if any refinement is needed (Tools).”

Every specific implementation of this intellectual model of needs, values, principles, practices, and tools depends on the specific human work system. That is, every human work system will have its own specific needs, values, principles, practices, and tools. This is independent of agile or lean approaches and is valid for any human work system.

Over time a very stable and consistent set of values, principles, practices, and tools has been observed in practice that support agile and lean thinking. While this is not a fixed and officially acknowledged set, nevertheless it can be treated as common agreement in the lean and agile community.

Examples of values sets can be found in process frameworks including Scrum [18] (Focus, Courage, Openness, Commitment, Respect), SAFe [17] (Alignment, Code Quality, Transparency, Program

Execution), and Extreme Programming (Simplicity, Communication, Feedback, Respect, Courage). Interesting as well are the value definitions of companies that identify themselves as agile and lean. For example, the official values of Google are:

- We want to work with great people
- Technology innovation is our lifeblood
- Working at Google is fun
- Be actively involved; you *are* Google
- Don't take success for granted
- Do the right thing; don't be evil
- Earn customer and user loyalty and respect every day
- Sustainable long-term growth and profitability are key to our success
- Google cares about and supports the communities where we work and live

Based on such values, we can identify a relevant and consistent subset of values that address agile and lean thinking. In this article, we refer to this set as the core agile and lean value set (or just core value set) as including the following:

- Focus
- Courage
- Openness
- Commitment
- Respect
- Communication
- Feedback
- Transparency

The discussion in this article is based on these values. Values do not suggest advice or recommendations for specific behaviors. Rather, values “...are the qualities we believe we should optimize for in order to meet the Needs [of the human work system]. They can be used as measuring sticks when deciding how to apply Principles”. (Citation from the Spine Model)

To keep focus in this article, the values openness, respect, and communication are explicitly omitted.

Company Values and Their Relationship to Leadership Values and Behaviors

An active and living core value set is a desired property of an agile and lean company. As stated above, there are no direct measures to create these values. There is no guarantee that a specific measure will create or improve directly a specific value. Instead, a consistent set of measures influences the evolution of the value as a whole. The reason is that a human work system is a complex system with no direct relation between a measure and an effect. Following the terminology of the spine model, “measures” is a synonym for the term “good practices”. In the following discussion, we will use the term *good practices*.

It is the responsibility of the leading individuals in a company to agree upon, implement, support, and anchor good practices. The term “leading individuals” is selected intentionally. Leading individuals often are members of management, but not all leading persons are members of management. Leading persons provide examples, engaging in change in an outstanding way. Leadership is executed by leading individuals.

In the situation in which a company is in an active transformation towards agile and lean, the transformation is an organizational change project driven by a change team provided by leading individuals in the company. In this case, change is addressed explicitly. Leadership is more important and relevant when a company claims itself to be agile or the change project officially is declared as finished successfully. An ongoing investment into the company culture is required to anchor, improve, and preserve the current state.

Leadership drives and anchors agile and lean values beyond explicit change activities as part of the daily work. Leadership in agile and lean organizations is the engagement by leading individuals to agree upon, implement, support, and anchor good practices that foster agility and lean as part of the daily work. Accordingly, we can state:

Leadership values are identical to the agile and lean core values set. Commitment to live and represent the agile and lean core values is an extra responsibility of individuals having a leadership role in an organization.

Leadership behavior at its highest level is the engagement by leading individuals to agree upon, implement, support, and anchor good practices that foster agility and lean as part of the daily work based on intrinsic motivation.

Leadership behavior includes setting a positive example.

Based on the core value set, the most important value in leadership is commitment; commitment to invest continually in anchoring the core values in the organization.

From Core Values to Good Practices

This article focusses on leadership behavior based on the core values: Focus, Courage, Commitment, Feedback, and Transparency. As stated above leadership behavior is the application of specific good practices as invest into core values.

Following the idea of the spine model, principles follow values as a set of general beliefs one can use to effect a desired change to a system. The discussion of principles behind the core values goes beyond the scope of this article. For more information about principles, we refer you to the spine model. This article focusses on leadership behavior as recommendations concerning what good practices in the form of methods and techniques that support one or more of the core values. The following sections discuss good practices to foster specific values.

Leadership Behaviors that Foster Focus

Maintaining Focus is one of the most difficult values. Focus is continually in conflict with ambitious company goals. Ambitious company goals strive to reach as many goals in a given period of time as possible. A typical error is to start too many activities simultaneously. Task switching and reducing the quality of the outcomes, resulting in future rework, are typical effects of this behavior⁴.

Leadership behavior strives to limit the work in progress (WIP) in the system. Ideally every employee and every team in an organization is actively working on not more than two distinct work packages (project, business epic, activity, whatever term a company uses) at one time. Kanban⁵ systems support this behavior. Visual Kanban boards for teams and individuals are a technique that allows adaptation of the work flow towards a continuous flow with the least amount of waste in human work systems. Requirements work packages should be as small as possible. A principle behind this is to treat every work package as a minimum viable product (MVP)⁶.

The second important behavior is to establish a pull system instead of a push system. Capacity consumption and planning is delegated to the teams. Teams pull work as soon as they finish a work package according to a well-defined Kanban work in progress limit. To establish this behavior, management support is essential. A typical management error is to push work into the system assuming this will encourage progress.

⁴ For more information see [6].

⁵ Kanban is a lean method to visualize and optimize flow in a system.

⁶ For more information see [7].

WIP limit and pull are the most important good practices to be supported by leadership. This includes 1) Delegating capacity planning responsibility to teams to optimize team work; and 2) Balancing teams with the objective of minimizing the number of work in progress (WIP) items in the entire human work system.

In summary, recommended good leadership practices in to foster focus are:

- Restrict work in progress (WIP) for teams and individuals to the minimal possible size. Ideally, every employee and every team should be actively working on not more than two distinct work packages.
- Work packages are as small as possible to establish a continuous work flow instead of a sequence of activities.
- Establish the human work system as a continuous flow. Any activities (meetings, planning, decision making) are aligned and synchronized along this continuous flow.
- Teams and individuals pull work packages according to their WIP limit.

Leadership Behaviors that Foster Courage

Courage is often communicated by management and as often it is misused. An encouraged employee decides to experiment with solutions or processes that are not covered by or are explicitly against company rules. Specific (non-agile) company cultures punish misbehavior, for example in employee performance ratings, or misuse this behavior for peer blaming because of potential career options.

The most important leadership behavior to foster courage is the principle behind a living failure culture. A living failure culture analyzes a failure; it identifies the root cause and learns by defining corrective action. Failures are treated as opportunities to learn while minimizing the number of failures within the human work system.

To foster courage, a core good practice is to set a good example, i.e., communicating one's own failures in an active and transparent way including executing actions to analyze and learn. Specific actions to analyze a failure are retrospectives. Retrospective techniques are well known and documented⁷. Concrete actions are manifold. One type of action is offering training and education options, either external or prepared and executed internally. For internal training and education establishing and fostering communities of practices (CoP) are a good practice as recommended by the Large Enterprise Scaled Scrum Framework LeSS⁸.

⁷ For more information see [19], [20] and [21]

⁸ See [8]. <https://less.works/less/framework/index.html>

Another good practice is to establish an on-employee level exchange with other companies on level of principles and good practices. Community of practices are a perfect base to establish and foster exchange under companies.

In summary, recommended good leadership practices to foster courage are:

- Leaders share their own failures in an active and transparent way, including executing actions to analyze and learn from failures.
- Providing training and education options.
- Establishing and fostering communities of practices (synonym: guilds).
- Establishing and fostering external out of company exchanges concerning principles and good practices.

Leadership Behaviors that Fostering Commitment

In many companies, one of the most discussed topics is the “gap”. Gaps are recognized and identified in many manifestations, for example, the gap between business and IT, the gap between management and the work force, the gap between goals and capabilities, the gap between project portfolio and available capacity, and many others.

A gap is a symptom for missing feedback (→ see section feedback), transparency (→ see section transparency) and call for action. An important question for leadership to answer is how to design a call for action. Call for action comes in two varieties, extrinsic and intrinsic. Agile and lean communities state that intrinsic motivation (=call for action) is the most important driver for a human being to do something. Research in Psychology and Neuroscience observe a well-set balance between extrinsic and intrinsic motivation as a key driver⁹. Based on these observations, principles and specific good practices can be derived.

A well-agreed principle is to work with vision and mission statements. The problem is to communicate, align, and bind vision and mission statements to specific activities at the employee level. Feedback and transparency are values that support disseminating vision and mission statements throughout an organization. Feedback and transparency are required, but in many cases, not sufficient. Additional activities are required to represent and communicate vision and mission throughout the organization in the form of specific guidelines for a target audience (i.e., an organizational unit).

⁹ See [11] and [12].

One principle for such activities is to communicate on eye-level. Members of the management and leading individual have to offer a relationship based on respect and eye-level. A relationship based on respect and eye-level communication fosters a mutual understanding of the needs, a feedback culture based on openness and transparency, and the respect of the vis-à-vis. These are enablers that create intrinsic and extrinsic motivation depending of the content of communication. Eye-level communication is the means to close gaps. The content of the communication is the mechanism to create the well-set balance of extrinsic and intrinsic motivation.

In summary, recommended good leadership practices to foster commitment through communication on eye-level are:

- Establish a regular One-day-with-X event. An arbitrarily selected employee accompanies a leading individual (or manager) one day in all her activities and vice versa.
- Strategic development invites employees from all organization units and at all levels of hierarchy.
- Top management provides a regular vision and mission roadshow including Q&A sessions for all departments and employees on a half year or even quarterly basis.
- Official company-wide chat sessions with leading individuals or managers on strategy, vision, mission, values, principles, or even specific projects.
- Leading individuals or managers work in staged activities at the employee level for one or more days – for example, as cashier in a retail shop, or as a bank branch office with direct customer contact.

Leadership Behaviors that Foster Feedback

Feedback is a well-defined and established value. Principles are found for example in the agile manifesto (“At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly”), XP (“Rapid Feedback”) or in queueing theory (“Fast feedback enables smaller queues; Use fast feedback to make learning faster and more efficient; Whenever possible make feedback local”).

Feedback has distinct dimensions. Feedback dimensions include the created customer value, the (professional excellence of the) development process, the personal engagement, and others. It is the responsibility of management to establish a holistic feedback system that addresses the most important feedback dimensions. Transparency (➔ see section transparency) is a required prerequisite to establish a holistic feedback system. Leadership behavior is the communicated desire to receive open and transparent feedback and to apply means to demonstrate that open and transparent feedback is desired on a regular basis. Concrete actions are then to guide and coach teams to define, establish, and execute

feedback activities in a regular cadence. Specific good practices include retrospectives in different forms and for different actions, and teams providing feedback on S.M.A.R.T.¹⁰ goals.

In addition to feedback concerning organization development, feedback on personal development is as important. Agile organizations distinguish between team development and personal development. Team development addresses organization goals to implement vision and mission (→ see section commitment). Personal development addresses capabilities and competences of individuals in areas such as self-competences (self-discipline, flexibility, autonomy, professional education...), social competences (communication, cooperation, conflict behavior ...) or leadership competences (delegation, coaching, responsibility, alignment, setting a good example). Core principles in leadership for personal development are decoupling of personal development from the annual personal review process and a feedback process that includes 360-degree feedback elements.

Over all good practices for feedback exist on many different levels and address different aspects in an organization. Good practices strive to establish a holistic feedback system that incorporates all levels and aspects in a sufficient manner.

Recommended good leadership practices to foster feedback include:

- Invest into (→ see section transparency) and (→ see section commitment)
- Require feedback for yourself on a regular (institutionalized) basis to provide an example and to communicate feedback results in a natural way.
- Establish coaching for teams to support in the definition of S.M.A.R.T. goals.
- Establish an active retrospective culture by demanding retrospectives for concrete actions, projects, or aspects.
- Invest in personal development in cooperation with the human resource department (perhaps better named the human development department).
- Decouple personal development from the wage and salary process
- Experiment and strive for 360-degree feedback as the most important element in personal development.

¹⁰ S.M.A.R.T. goals are specific, measurable, achievable, results-focused, and time-bound

Leadership Behaviors that Foster Transparency

The traditional behavior in companies is to allow access to only that information required to execute all activities as defined by a specific job description. For example, a customer service agent has access to information that is required to answer customer calls and to get feedback about her performance (calls per hour) and her call quality (customer satisfaction feedback). This behavior often results in tunnel vision, a limited view on vision and mission of the organization and in finger-pointing to collaborating organizational units in the event of problems.

An agile and lean principle to foster transparency is to allow access to as much information as possible to every individual in the organization. Ideally, restriction exists only where compliance or legal aspects apply or where the objective risk of damage or loss is extreme. In many companies the subjective risk is taken as the rationale to hide information. The internal publishing – for example – of information about the realization of a new and competitive feature offered to customers is a typical case for a subjective risk. Google for example keeps all employees informed about a new feature as soon as possible and trusts employees not to communicate externally before an agreed point in time.

This behavior is even harder to implement within an organization in the area of performance indicators. Performance indicators are typically used to rate the performance of a team or an individual. A lean and agile interpretation of performance indicators is the valuation of the organizational power, capability, and ability. Leadership in respect to performance indicators motivates every employee and every team to employ performance indicators as benchmarks against competition or an organizations goal and deriving actions to improve the human work system towards the benchmarks. In this way heuristic thinking, an “optimize the whole” thinking is supported, and a tunnel vision or limited view attitude avoided.

This transparency includes two qualities: 1) to allow access to as much information as possible to every individual in an organization, and 2) to establish a mindset to work positively with the information, i.e., to utilize the information as benchmark of the current state and an opportunity to identify options for improvement of this state aligned to vision and mission of the organization, rather than to harm the organization.

Recommended good leadership practices to foster transparency include:

- Clearly define the set of limitations that provide restricted access. Clarify why any specific information unit is under restricted access.
- Define and communicate benchmarks against competition or derived from company vision and mission (in the form of S.M.A.R.T. goals)
- Delegate the development of performance indicators to teams and support the definition performance indicators aligned to S.M.A.R.T. goals.

- Give access to the reporting system to all staff members.
- Establish behaviors as listed under “commitment”.

Summary

It is widely accepted that it is not possible to create or form a company culture by direct means and actions. A company culture emerges as the result of behaviors, measures, actions, rules, and conditions that are executed and active within a human work system that is, by definition, a complex system.

It is also widely accepted that leadership behavior influences company culture. This article identifies behaviors that strive to comprise a company culture with agile and lean thinking. The starting point is the discussion of needs, values, principles, and practices as defined in the spine model. A small set of relevant core values is identified as the desired properties of an agile and lean company. From this core value set, the article identifies and discusses specific leadership behaviors and good practices that apply the values in action.

Essential is the finding that these behaviors are interrelated. It is misleading to select a single good practice and to expect a specific impact as effect. It is as well a property of an organization to select an appropriate set of good practices and to experiment with different combinations of good practices. Feedback concerning company culture is essential to observe whether the applied good practices influence company culture as desired. A core leadership behavior is to make all this happen.

List of Acronyms Used in this Paper

<u>Acronym</u>	<u>Explanation</u>
CEO	Chief Executive Officer
WIP	work in progress
SAFe	the Scaled Agile Framework
MVP	minimum viable product
MMP	minimum marketable product
CoP	community of practice (or guild)
Intrinsic motivation	people’s spontaneous tendencies to be curious and interested
Extrinsic motivation	when an activity is done in order to attain some separable outcome
S.M.A.R.T.	Acronym for specific, measurable, achievable, results-focused, and time-bound

Acknowledgements

A sincere thank you to SyEN Editor Dr. Ralph Young for his extensive and very helpful collaboration with me.

References

- [1] Chart: The Largest Companies by Market Cap Over 15 Years. <http://www.visualcapitalist.com/chart-largest-companies-market-cap-15-years/>. Copyright by Visual Capitalist.
- [2] Al-Najem, Mohamad, Dr. Hom Dhakal, and Professor Nick Bennett. "The Role of Culture and Leadership in Lean Transformation: A Review and Assessment Model." International Journal of Lean Thinking, Volume 3, Issue Number 1, 1 Jun 2012, Pages 119-138.
- [3] The Agile Manifesto, <http://agilemanifesto.org/>.
- [4] Beck, Kent and Cynthia Andres. Extreme Programming Explained: Embrace Change, Addison-Wesley; 2nd edition (November 16, 2004), ISBN-13: 978-0321278654.
- [5] Spine Model, <http://spinemodel.info/>, © 2017 Spine Model is licensed under a Creative Commons Attribution-NoDerivatives 4.0 International License.
- [6] Reinersten, Donald G. The Principles of Product Development Flow. Celeritas Publishing, 29 May 2009, ISBN-13: 978-1935401001.
- [7] Ries, Eric, The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses, Publisher: Currency, 17 Oct 2017, ISBN-13: 978-1524762407.
- [8] The Large Enterprise Scaled Scrum Framework. <https://less.works/>, © 2014 ~ 2017 The LeSS Company B.V. All Rights Reserved.
- [9] Lean Leadership of Senior Management, Research Paper, OPX-Institut, BTW: NL8530.24.418B01. Available at <http://www.vanassen.info/wp-content/uploads/Research-paper-Lean-Leadership-of-Senior-Management.pdf>.
- [10] Larman, Craig and Bas Vodde. Scaling Lean and Agile Development. Addison Wesley, 8 Dec 2008. ISBN-13: 978-0321480965.
- [11] Ryan, Richard M. and Edward L. Deci, Intrinsic and Extrinsic Motivations: Classic Definitions and New Directions, University of Rochester, Contemporary Educational Psychology 25, 54–67 (2000).
- [12] Di Domenico, Stefano I. and Richard M. Ryan. The Emerging Neuroscience of Intrinsic Motivation: A New Frontier in Self-Determination Research, Frontiers Human Neuroscience, 24 March 2017. Available at <https://doi.org/10.3389/fnhum.2017.00145>

- [13] Ambler, Scott. *Disciplined Agile Delivery*. IBM Press, 23 May 2012. ISBN-13: 978-0132810135.
- [14] Appelo, Jurgen. *Management 3.0*. Addison Wesley, 28 Dec 2010. ISBN-13: 978-0321712479.
- [15] Pflaeging, Niels. *Organize for Complexity: How to Get Life Back Into Work to Build the High-Performance Organization*. Betacodex Publishing, 19 March 2014. ISBN-13: 978-0991537600.
- [16] Strathausen, Roger. *Leading When You're Not the Boss: How to Get Things Done in Complex Corporate Cultures*. 1st ed. Edition. Apress. ISBN-13: 978-1484217474.
- [17] SCALED AGILE, INC, <http://www.scaledagileframework.com/>, Copyright © 2010-2017 Scaled Agile, Inc.
- [18] Schwaber, Ken, *Agile Software Development with Scrum*, Prentice Hall; (18 Feb 2002)
Language: English, ISBN-13: 978-0130676344.
- [19] Ester Derby, Diana Larsen, and Ken Schwaber, *Agile Retrospectives: Making Good Teams Great*, Pragmatic Bookshelf, August 5, 2006; ISBN-13: 978-0977616640
- [20] Luis Goncalves and Ben Linders, *Getting Value out of Agile Retrospectives – A Toolbox of Retrospective Exercises*, June 4, 2014; ISBN-13: 978-1304789624
- [21] Norman L. Kerth, *Project Retrospectives: A Handbook for Team Reviews*, Dorset House, February 1, 2001, ISBN-13: 978-0932633446

ARTICLE

Integrating Program Management and Systems Engineering

by

Dr. Ralph R. Young, Editor, SyEN

This month we provide a summary of Chapter 6, How Integration Works in Programs, in *Integrating Program Management and Systems Engineering* (IPMSE), a collaboration of the International Council on Systems Engineering (INCOSE), the Project Management Institute (PMI), and the Consortium for Engineering Program Excellence (CEPE) at the Massachusetts (USA) Institute of Technology (MIT).

Chapter 6 is the first of six chapters in Part II, Building Capabilities to Effectively Execute Programs. The task in this chapter is to provide a framework, called the “Integration Framework”, which defines the key factors that enable effective utilization of program management and systems engineering disciplines to drive higher performance, stronger team engagement, and customer satisfaction.

The framework highlights how various factors work in concert to produce outcomes in complex behavioral networks. Since integration is a multi-faceted concept, this framework explains the core dimensions and observed practices, organizational approaches, and skillsets in use by successful programs with high levels of integration. The framework emerged from the multiyear research activities described in the book's introduction.

The integration of program management and systems engineering is defined as a reflection of the organization's ability to combine program management and systems engineering practices, tools and techniques, experience, and knowledge in a collaborative and systematic approach in the face of challenges, in order to be more effective in achieving common goals and objectives in complex program environments.

The Integration Framework encompasses six main dimensions, shown in Figure 6-1 in "The Book".

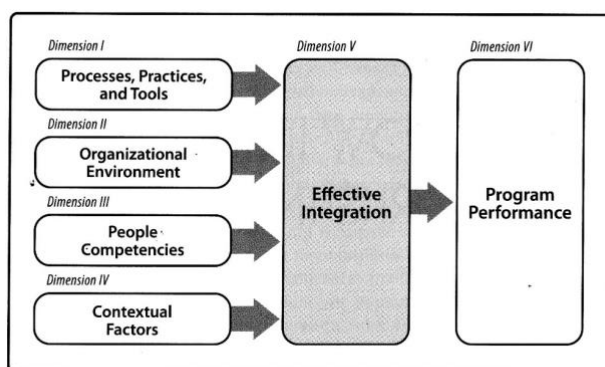


Figure 6-1: The Integration Framework for program management and systems engineering

The center of the framework, figuratively and literally, is 'effective integration'. On the right side is the program performance dimension. Multiple findings from the research indicated that greater integration leads to improved program performance. (The evidence for this finding is discussed in detail in Chapter 12 of the book.)

On the left side of the framework are four dimensions that when combined contribute to greater integration between the program management and systems engineering disciplines. These dimensions are processes, practices, and tools; organizational environment; people competencies; and contextual factors.

In each dimension shown in Figure 6-1 (provided in the book) there are several variables and associated elements. Subsections in the book examine each dimension in more detail, define the dimension, and highlight some of the key insights that emerged during the research on integration.

One objective of this framework is to characterize integration as an organizational and behavioral competence through the elaboration of the multiple integration factors. Evidence from the research suggests that the individual elements, such as processes, practices, and tools from program management and systems engineering areas, are just one piece of a larger puzzle of organizational capabilities.

The Effective Integration Dimension is a result of a combination of these four dimensions. It is important to understand that effective integration comprises three distinct elements, rapid and effective decision making, effective collaborative work, and effective information sharing.

Integration is not simply the combination of standards, practices, tools and techniques, and defining roles and responsibilities. Instead, it is a broad concept that encompasses multiple dimensions of the organization and the program, and it is grounded in the three elements.

Figure 6-8 in The Book is a graphic of the complete Integration Framework, including all dimensions and key elements. As noted above, in each of the dimensions shown in Figure 6-1, there are several variables and associated elements. There is a discussion of each dimension that describes how the specific variables and associated elements were evolved as well as the related research that was applied.

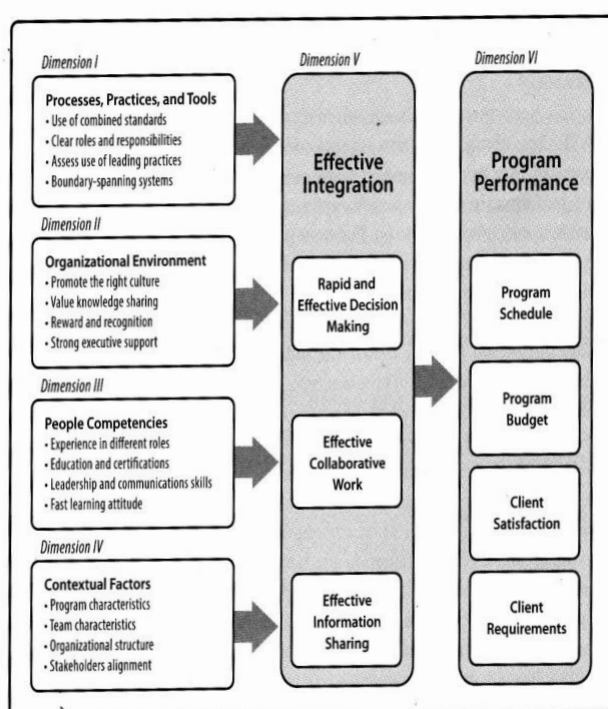


Figure 6-8: The complete Integration Framework with all dimensions and key elements

Interestingly, none of the professionals interviewed during the research indicated that their organizations had clearly defined objectives related to integration itself! Organizations measure success based on schedule, budget, and scope.

Research shows that one of the key challenges in the program management domain is benefits realization. Unlike projects that have defined deliverables whose value is generally linked to customer satisfaction, scope, budget, and time, programs are designed to deliver a broad array of business benefits. Those benefits can range from new capabilities to financial return to opening new markets. But those benefits are often delivered throughout the life of the program which can last for many years. It is impossible also

to define specific goals relating to the development of integration (i.e., its associated capabilities, and how it is to be measured) and then putting the right processes and measures in place to capture those benefits.

The analysis of the Phase IV research found a positive correlation between greater levels of integration and better program performance. Of two groups, one with greater levels of integration and another with lesser levels of integration, the group with greater levels of integration also had higher levels of program performance.

The value of these insights to you might best be determined by giving thoughtful attention to the following questions:

1. Why is adopting a standard (or multiple standards) for practices, tools, and techniques valuable for improving integration?
2. Why is it important to define clear goals and objectives for integration?
3. Using the framework to evaluate your own organization, in which dimensions is your organization strongest and weakest?
4. What are the gaps within your organization highlighted by the Integration Framework that you believe, if closed, would help to improve the level of integration between systems engineering and program management?
5. Who are the key stakeholders inside your organization who can provide additional insight to support evaluation of the organizational and behavioral competencies that support integration?

Acknowledgement

The tables are reproduced from Integrating Program Management and Systems Engineering: Methods, Tools, and Organizational Systems for Improving Performance, Eric Rebentisch, Editor-in-Chief, Published by John Wiley Sons, Inc. Copyright 2017 by John Wiley & Sons, Inc. All rights reserved.

Should Systems Engineering be Integrated into High School Curricula?

In a study conducted by Intel involving 1000 teenagers, it was discovered that most high school students do not enter engineering in university because 'they do not know what engineers do'. 63% of students between the ages of 13 and 18 had never even considered engineering as a career choice in spite of the fact that their general opinions of engineering were generally positive. In light of this, it is no wonder that the Da Vinci School for Science and Arts sought to develop an engineering curriculum that would incorporate some of the problem solving and design problems found in various engineering fields in order to increase the student rate of buy-in for studying engineering. The 10th grade course devised incorporated a curriculum comprising the following units:

- General introduction to engineering
- An overview of the engineering process
- Further in-depth introductions to Mechanical, Civil, Electrical, Bioengineering and Industrial & Manufacturing Engineering
- A Conclusion in Materials Studies.

In a survey carried out after the first implementation of the curriculum, more than 50% claimed to have definitely understood what engineering involved. In fact, 71% of students stated that they grasped the Engineering Design Process. On the other side, only 22% agreed that they understand how studying engineering helps address real world issues and only 26% stated that they understood the relationships among the five core engineering disciplines studies. Of the skills that were learned, the strongest included working effectively with others (78%) and applying the Engineering Design Process to solving problems (61%) while the weakest area was, by a landslide, the ability to write documents in appropriate style for Engineering, with a 26% outcome. On one hand the study proves that the curriculum was valuable since 55% indicated that they definitely or possibly would study engineering in university. On the other hand, the study proves that the curriculum may have been lacking since fundamental aspects of engineering were missed by the students. This was suggested by the poor results in the understanding of how engineering disciplines are integrated, the understanding of how engineering solves real-world problem and the poor performance in engineering writing skills. This supports the view that engineering may not always be introduced or taught in the best way, even in the very early introductions to the field. What is noticeable is that the weaker-performed aspects of the course could have improved by teaching some fundamental principles of Systems Engineering, seeing that Systems Engineering is a multi-disciplinary approach to solving complex problems in the real world. Systems Engineering should be included in any engineering

program - even at the high school level - to bring the multi-disciplinary aspects together and describe a structured, low-risk approach to solving problems of high complexity and uncertainty. In addition, the ability to write coherently and effectively is a fundamental part of defining the problem - outlining the requirements that needs to be done *before* the problem is solved. One wonders if the separation between problem and solution is ever distinguished clearly enough in any engineering course. Although no course in engineering at a high school level will ever be complete - it certainly will be improved with a sound introduction to the Systems Engineering Approach.

[More information](#)

A Systems Thinking Approach to Leadership

In today's world where complexity and uncertainty adorn so many of the projects in our organizations, it is essential that components playing a role in the execution of a project are contributing effectively. One of the most important components, of course, are the people; and the members of a project team and organization are often left feeling discontented in their roles as the work focuses on navigation rules and procedures which are often not in accordance with real project progress - i.e. organizations commonly follow famous Pareto phenomena where 80% of activities done only lead to 20% of the results. Furthermore, sometimes the common project management activities of better planning, analysis and monitoring do not yield any improvements in the project and enterprise results.

The constant state of busyness experienced by so many in recent years arises from the obsessiveness with doing *something*, often no matter what it is. The role of leadership in an organization is to influence the individual and team in order to work towards organizational goals in a way that aligns their personal aspirations to the work done by the individual and the team. What is common in most organizations is the 'don't stop, just keep going - we need to do *work*' sentiment propagated by people in leadership positions and instead of the pause-and-reflect perspective which leads to measuring twice but only cutting once. What is most commonly experienced of course is measuring once but cutting twice- the far more expensive and time-consuming route. The issue here is that 'measuring' is not usually viewed as part of the work. This leaves workers feeling frustrated, discouraged and either frustrated or apathetic as the incessant work seems to bring little value. By over-simplifying the complexity of human behavior in search of a linear cause and effect explanation - the root cause leading to dissatisfactory results is often missed. That arises in the shorter lunch breaks and longer work hours solution to try to increase the amount of valuable output.

Peter Senge describes Systems Thinking as providing the toolset to unpack the interrelationships in complex systems such as societies. In Sydney Finkelstein's article on Why Smart Executives Fail, his examination of over 50 of the world's most well-known business failures resulted from people in traditional leadership positions in the corporation not accepting the reality of what was going on in the business. In other words, not adopting a Systems View of the project or enterprise situation but a muffled and truncated version of reality. In addition to looking at the reality, Systems Thinking encourages a constant assessment

of changes in the environment in order to keep the mental model as accurate to reality as possible and to ensure that the system is resilient and pre-emptive. The Systems Thinking Approach to leadership focuses on synthesis over analysis, reality over destiny, effectiveness over excessiveness and putting people first.

[More information](#)

A Systems Engineer as A Music Maestro

In an article titled The Art of Science of Systems Engineering aimed at defining the scope of Systems Engineering, NASA compares Systems Engineering to an orchestra and its ability to play a symphony. Although most people understand what music is, fewer can play an instrument and each instrument requires a different set of skills and expertise in order to play it. Some musicians spend their lifetime mastering one instrument but in an orchestra, different instruments need to be played in unison to perform the symphony. If the symphony is the system, musicians apply science to their music by translating notes on a piece of paper into music. The conductor leads the process to ensure there is co-ordination and harmony to produce a beautiful piece of music. In particular, a Systems Engineer is a lot like a music maestro:

- Just as a maestro needs to know about aspects of music such as pitch, rhythm and dynamics so the Systems Engineer needs to understand the fundamentals of mathematics, physics and other sciences
- Just as a maestro needs to have mastered one or more musical instruments so does the Systems Engineer needs to have mastered a technical discipline and learn about multiple disciplines
- The System Architect is to Systems Engineering as the composer is to the symphony
- The maestro interprets the composer's music in the same way that a System's Engineer must interpret objectives and requirements to form a formidable design
- Just as the maestro must ensure that integrity of the composer is maintained so must the designer's technical integrity be demonstrated in the final product
- Overall, the maestro is responsible for the total of the delivery and success of the performance as is the Systems Engineering responsible for the delivery of the product

Like playing a symphony, Systems Engineering is holistic and integrative and is a combination of an art and a science. Technical Leadership is the art that balances technical knowledge, problem solving, communication and creativity and Systems Management is the science that focuses on rigorous and efficient execution throughout development and operation. For success, Technical Leadership and Systems Management need to be seamlessly blended to create a result worth writing Mozart about.

Read the original article [here](#).

Future Directions for SysML v2

by

Sanford Friedenthal, Model Based Systems Engineering (MBSE) Consultant and Chair of Object Management Group's (OMG) Systems Engineering (SE) Special Interest Group

and

Larry Johnson, OMG Vice President and Technical Director

Abstract

SysML v1 was adopted in 2006, and has been a key enabler of model-based systems engineering (MBSE). Since that time, much has been learned about applying MBSE with SysML. This presentation provides an overview of the requirements for the next generation of SysML v2 to provide capabilities that address the limitations of SysML v1, and enable the evolving practice of MBSE. The links below provide access to both a YouTube video of the presentation and the slides for the presentation.

Status Update on SysML v2 Requirements as of December 8, 2018

The SysML v2 RFP was issued at the Object Management Group ([OMG](#)) meeting in Burlingame, California on December 8, 2018. This culminates a 1.5 year effort to develop the requirements for the next generation systems modeling language, which is intended to improve the precision, expressiveness, and usability over SysML v1. A complementary RFP entitled the 'SysML v2 API and Services RFP' is also being developed to enable interoperability between SysML modeling tools and other model-based engineering tools. This RFP is planned to be presented at the next OMG meeting in March, 2018, and issued at the following OMG meeting in June, 2018. The capabilities provided by SysML v2 should enable improved effectiveness of MBSE and broader adoption by the systems engineering community.

Highlights of this paper

Note: that the presentation on the Future Directions for SysML v2 begins about seven minutes into the [video](#) after the introduction to the OMG by Larry Johnson

- Model-based systems engineering (MBSE) is an evolving systems engineering practice that addresses many of the limitations of using document-based artifacts to capture information about the system. Model-based artifacts can be more consistent, precise, traceable, and reusable than document-based artifacts.
- A model of the system (i.e., the system model) is the over-arching model-based artifact that contains information about the specification, design, analysis, and verification of the system under

development. To fully leverage the value of MBSE, the system model must be integrated with other engineering models including electrical, mechanical, software, test, and analysis models.

- SysML is a general-purpose modeling language that represents multiple views of the system in a system model that include the system requirements, behavior, structure, and parametric constraints.
- SysML has evolved from SysML v1 to SysML v1.5 over the past ten years based on user and tool vendor feedback.
- The requirements for the next generation of SysML (v2) are based on experiences with applying MBSE with SysML v1, and are intended to provide significant enhancements for both the developers and consumers of system models. This includes improved expressiveness, precision, usability, and interoperability.
- The requirements for SysML v2 are specified in two separate request-for-proposals (RFPs) to be issued by the Object Management Group (OMG). The SysML v2 RFP specifies the requirements for the modeling language, and the SysML v2 API and Services RFP specifies the requirements for an Application Programming Interface (API) to provide standard access to the system model.

The slide deck is available [here](#).

Object Management Group Issues RFP for SysML v2

The Object Management Group® (OMG®), an international, open membership, not-for-profit technology standards consortium, issued a [Request for Proposal \(RFP\) for SysML® v2](#), which will improve the precision, expressiveness, and usability relative to SysML v1. In order to influence the future of SysML, both members of OMG and non-members must submit a Letter of Intent to show their interest in participating in the process by September 24, 2018. In order to become a submitter, individually or as part of a submission team, companies must become members of OMG by the initial submission deadline of November 4, 2019.

[More information](#)

FEATURED ORGANIZATION

INCOSE's Critical Infrastructure Protection and Recovery (CIPR) Working Group

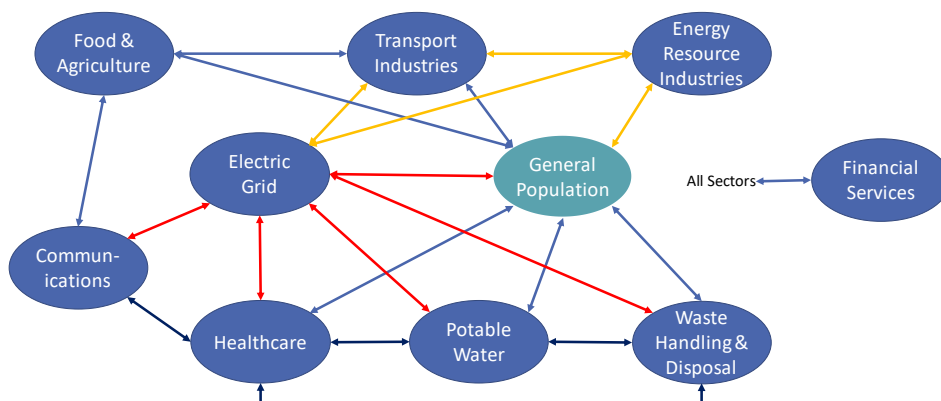
Our Critical Infrastructure

We take our modern societies for granted. Our ability to buy groceries in a city of millions, drink clean water, travel freely and power our homes is without equal in history. Our critical infrastructures that enable these services are so reliable that we hardly think about them. They enable the growth of civilization to support billions of people, trillion dollar economies, and a transition to an age dominated by the exchange of information.

The critical infrastructure domains enabling modern life include those listed below. There are more domains than listed here, but perhaps these represent those that we deal with daily, or nearly so.

- Communications & Information Technologies
- Electrical & Energy production and distribution
- Financial Services
- Food and Agriculture
- Healthcare
- Transportation
- Water storage, treatment and distribution
- Waste handling and disposal (water, refuse, hazardous)

Each of these infrastructures are networks that inextricably intertwine with the other domain networks. They affect one another even if not physically connected. The figure below shows an oversimplified view of the network of networks.



Our infrastructure networks are so tightly coupled that a failure or glitch in one network node will ripple through the other networks. For example, if potable water services are lost in a community, then most community-based services, (e.g. hospitals, emergency services) will also be lost or severely degrade in a matter of a few hours or days. The communities might not recognize the full brunt of such an event if they are well-supported by their neighboring communities.

The Threats

Critical infrastructures are vulnerable to manmade and natural events that can cause disruption for extended periods, resulting in societal disruptions and loss of life. Hurricanes, earthquakes and tsunamis are examples of devastating natural events that destroy regional infrastructure. The affected communities can take months or years to recover. When there are local unaffected regions nearby, the affected communities and their population can survive and return to normalcy far more quickly than those that are isolated. The logistical details and availability financing are key enablers affecting the rate of recovery. However, there are certain threats that can have such widespread and debilitating impacts that sufficient logistics and financing are no longer possible.

Three threats are known to endanger critical infrastructure so pervasively that a large-scale incident could make our infrastructure unavailable for a period of a month or more – mostly likely much more. These threats include Electro-Magnetic Pulse (EMP) weapons, Geo-Magnetic Disturbances (GMD) and cyber-attacks on IT and industrial control systems. These threats, sometimes referred to as the “triple threats” can cause damage on regional, continental or even global scales such that obtaining help from neighboring regions is unlikely since all neighboring regions may be equally and significantly damaged.

EMP has been known as a threat to critical infrastructure since the 1960’s when high altitude atmospheric tests of nuclear weapons were conducted by the US and the USSR. We learned that a single bomb detonated at the optimum altitude can destroy electric power grids that are over 1000 miles distant. Since that time, EMP weapons were placed in country arsenals and their offensive use included in the military doctrines of the US, NATO, Russia, China, India, Pakistan, Iran, Israel, North Korea, and others. These are considered by some as the weapons of first resort in a major conflict due to their widespread devastating impact.

The result of EMP is not destruction caused by blast, heat or radiation. Rather, the nuclear blast takes place in the upper atmosphere inducing three types of electromagnetic pulses known as the E1, E2 and E3. The E1 pulse has a nanosecond rise time with extreme electromagnetic fields that can generate high currents in conductors (e.g. transmission lines) and thereby destroy electrical components. The E2 is similar in energy and pulse characteristics as a lightning strike, and already addressed in system designs. The E3 has a slower rise time, but lasts dozens of seconds with the ability to couple to long wires, also generating sufficient currents to destroy electrical components. The technology to protect against this

threat is available with moderate cost. The primary issues needing to be resolved have more to do with policy than technology.

GMD is caused by large Coronal Mass Ejections (CMEs) from the sun interacting with the Earth's atmosphere. The chance that one such event can have sufficient magnitude to destroy electric grid components throughout a continent, or even with significant global effect is about 10% per decade. We are overdue since the last one of such great magnitude hit the Earth in 1859. These interactions produce a pulse similar to the E3 with similar destructive effects. CMEs and the resulting GMDs happen regularly, but tend to be of smaller to moderate size. In late 2015, a moderate GMD disrupted air traffic control over Sweden for several hours. Today, studies of premature loss of electrical grid equipment has been linked to even moderate GMD. In the last few decades, moderate storms were credited with taking down the electrical grid in the US Northeast and in Canada in 1979 and again in 1989. The larger storms appear to have a periodicity of about 100 years. The Earth narrowly escaped being hit by a large CME in 2012. It missed us by only a week. The threat is considered so real that the US Government recently issued a US Space Weather policy and action plan.

Cyber-attack threats are typically thought about in terms of loss of data and privacy. However, the threat to critical infrastructure is not about data, but loss of visibility and control. Cyber-attacks on industrial control systems pose a more serious physical threat that can result in direct loss of life through destruction of equipment. Tests such as Aurora, conducted by the Idaho National Laboratory, demonstrated that a cyber-attack on a control system can be used to destroy heavy industrial equipment. In this test a heavy generator was caused to self-destruct. Stuxnet is another example where control systems were spoofed to show that systems were operating within normal parameters (loss of visibility), while the centrifuges were being driven erratically (loss of control), resulting in self-destruction. News reports of the last couple of years revealed the ability for nations to infiltrate the electrical systems of others, enabling an attack that disabled power distribution. The same kinds of control systems are used throughout the world's critical infrastructure, but cyber security of these devices is known to be weak.

These threats, when considering their effects on critical infrastructure, cause complex systems problems needing immediate coordinated attention across traditional domain and governmental boundaries. For example, a US President issues Presidential Policy Directive PPD-21 that addresses "a national unity of effort to strengthen and maintain secure, functioning, and resilient critical infrastructure." This includes an imperative to "implement an integration and analysis function to inform planning and operations decisions regarding critical infrastructure." The CIPR working group will seek to support this and other policies with international application and reach.

CIPR Purpose

INCOSE, as the premier professional society for systems engineering, can provide significant contributions toward critical infrastructure protection and recovery. We need to ask ourselves the question, "Is INCOSE

willing and able to contribute to resolving problems of national and international importance?” This question is most importantly answered by each INCOSE member.

The purpose for the Critical Infrastructure Protection and Recovery (CIPR) Working Group (WG) is to provide a forum for the application, development and dissemination of systems engineering principles, practices and solutions relating to critical infrastructure protection and recovery against manmade and natural events causing physical infrastructure system disruption for periods of a month or more.

This working group will work within INCOSE and with external organizations sharing similar interests and goals. We will promote and apply systems engineering principles with emphasis on policy, analysis and concepts useful to understand, protect and recover existing operational infrastructure, and to provide strategies, standards and concepts for more resilient approaches. Specific areas of development include the following:

- Characterizing the event events capable of causing infrastructure disruption for periods of a month or more, to include all aspects of their characteristics and impacts.
- Developing reference models for infrastructure systems.
- Understanding socio-technical factors related to CIPR.
- Understanding the overarching structure, behaviors and inter-connectedness among the critical infrastructure domains.
- Characterizing interactions among infrastructure systems under various degraded states of operation.
- Examining possible conceptual and design solutions, and related information.
- Offering strategies for verification and validation of solutions.

[More information](#)

CONFERENCES AND MEETINGS

For more information on systems engineering related conferences and meetings, please proceed to [our website](#).

SOME SYSTEMS ENGINEERING-RELEVANT WEBSITES

Object Process Methodology Life Cycle Model

This site provides an overview of object process methodology life cycle model.

<http://www.freetutes.com/systemanalysis/sa2-object-oriented-methodology.html>

SysML Tutorials for Model-Based Systems Engineering

A list of online tutorials that teach SysML and Model - Based Systems Engineering (MBSA) in a manner that is completely independent of tools. Quotations are available on requests and onsite or online courses available.

<http://sysmlforum.com/sysml-resources/tutorials/>

WorldWideLearn

A website containing information on various career choices including what it means to study the degree searched, what a career in education for that field entails, what degree programs are available and information on possible career options, certification and licensure in the field. In particular, this link goes to the Systems Engineering page which gives information on the comprising majors and the most important aspects of the field. WorldWideLearn aims to provide prospective students with the resources they need to make an informed decision.

<https://www.worldwidelearn.com/online-education-guide/engineering/systems-engineering-major.htm/>

Systems Engineering Standards: A Summary

A link to a .pdf free download summarising standards and models in Systems Engineering. Standards covered include: MIL-STD-499, ANSI/EAI 632, IEEE 1220, ISO/IEC 15288 and CMMI.

<http://www.acqnotes.com/Attachments/Systems%20Engineering%20Standards%20A%20Summary.pdf>

Wayne University College of Engineering Industrial and Systems Engineering Current and recent projects

A list of current and research projects undertaken by or involving the Industrial and Systems Engineering department at the College of Engineering. Projects showcased include ones related to healthcare, design tools and human factors in engineering. An interesting site to discover the application of Systems Engineering as well as to see the calibre of work and the great projects undertaken by the university and its students.

<http://engineering.wayne.edu/ise/research/current.php>

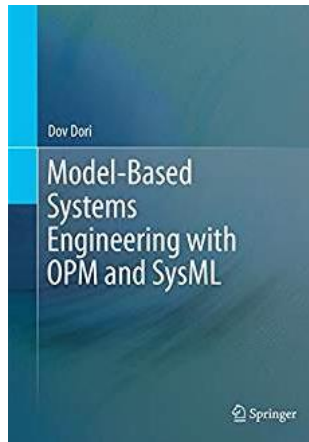
Writing Guidelines for Engineering and Science

Writing guidelines is designed to help people in engineering and science to write about their work. Site contains exercises on grammar, punctuation, style, definitions and other topics to improve competence in English. Visitors to the site are welcome to print and distribute any resources on the site, as long as credit to the site is given. This is a great source to improve one's scientific writing technique in order to convey the message as close to the intended meaning and to facilitate understanding so that the work to which the writing applies is honoured.

<https://www.craftofscientificwriting.com/>

SYSTEMS ENGINEERING PUBLICATIONS

Model-Based Systems Engineering with OPM and SysML



[Image source](#)

by

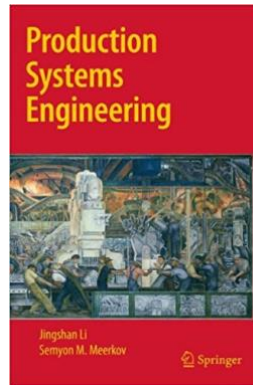
Dov Dori

Book Description (from the Amazon web site):

Model-Based Systems Engineering (MBSE), which tackles architecting and design of complex systems through the use of formal models, is emerging as the most critical component of systems engineering. This textbook specifies the two leading conceptual modeling languages, OPM—the new ISO 19450, composed primarily by the author of this book, and OMG SysML. It provides essential insights into a domain-independent, discipline-crossing methodology of developing or researching complex systems of any conceivable kind and size. Combining theory with a host of industrial, biological, and daily life examples, the book explains principles and provides guidelines for architecting complex, multidisciplinary systems, making it an indispensable resource for systems architects and designers, engineers of any discipline, executives at all levels, project managers, IT professional, systems scientists, and engineering students.

[More Information](#)

Production Systems Engineering



[Image source](#)

by

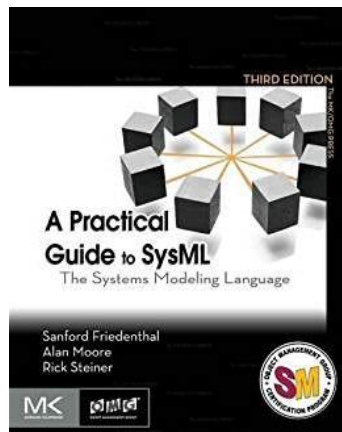
Jingshan Li and Semyon M. Meerkov

Book Description (from the Amazon web site):

Production Systems Engineering (PSE) is an emerging branch of systems engineering intended to uncover fundamental principles of production systems and utilize them for analysis, continuous improvement, and design. This volume is the first ever textbook devoted exclusively to PSE. It is intended for senior undergraduate and first year graduate students interested in manufacturing. The development is first principle-based rather than recipe-based. The only prerequisite is elementary probability theory; however, all necessary probability facts are reviewed in an introductory chapter. Using a system-theoretic approach, this textbook provides analytical solutions for the following problems: mathematical modeling of production systems, performance analysis, constrained improvability, bottleneck identification and elimination, lean buffer design, product quality, customer demand satisfaction, transient behavior, and system-theoretic properties. Numerous case studies are presented. In addition, the so-called PSE Toolbox, which implements the algorithms developed, is described. The volume includes numerous case studies and problems for homework assignment.

[More Information](#)

A Practical Guide to SysML: The Systems Modeling Language



[Image source](#)

by

Sanford Friedenthal, Alan Moore, and Rick Steiner

Book Description (from the Amazon web site):

A Practical Guide to SysML, Third Edition, fully updated for SysML version 1.4, provides a comprehensive and practical guide for modeling systems with SysML. With their unique perspective as leading contributors to the language, Friedenthal, Moore, and Steiner provide a full description of the language along with a quick reference guide and practical examples to help you use SysML.

The book begins with guidance on the most commonly used features to help you get started quickly. Part 1 explains the benefits of a model-based approach, providing an overview of the language and how to apply SysML to model systems. Part 2 includes a comprehensive description of SysML that provides a detailed understanding that can serve as a foundation for modeling with SysML, and as a reference for practitioners. Part 3 includes methods for applying model-based systems engineering using SysML to specify and design systems, and how these methods can help manage complexity. Part 4 deals with topics related to transitioning MBSE practice into your organization, including integration of the system model with other engineering models, and strategies for adoption of MBSE.

- Learn how and why to deploy MBSE in your organization with an introduction to systems and model-based systems engineering
- Use SysML to describe systems with this general overview and a detailed description of the Systems Modeling Language
- Review practical examples of MBSE methodologies to understand their application to specifying and designing a system

- Includes comprehensive modeling notation tables as an appendix that can be used as a standalone reference

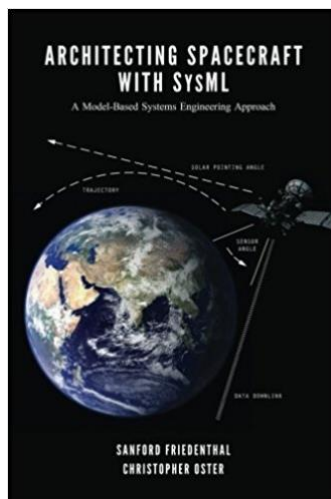
[More Information](#)

Wind Plant Organization and Systems Engineering Newsletter

The *Wind Plant Optimization and Systems Engineering* newsletter covers the latest research, technology, and software development activities about systems engineering. Topics range from multi-disciplinary design analysis and optimization of wind turbine sub-components to wind plant optimization and uncertainty analysis to concurrent engineering and financial engineering.

[More information](#)

Architecting Spacecraft with SysML: A Model-based Systems Engineering Approach



[Image source](#)

by

Sanford Friedenthal and Christopher Oster

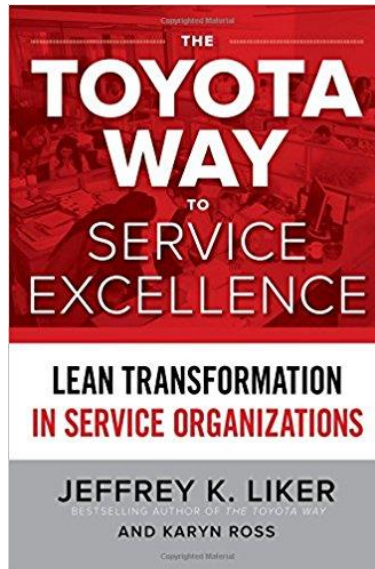
Book Description (from the Amazon web site):

This book provides a straightforward guide to develop an architecture model of a Small Satellite using the Systems Modeling Language (SysML®). SysML is a general-purpose modeling language used to specify and architect systems. Model-based Systems Engineering (MBSE) is intended to produce an integrated system model using SysML which reflects multiple views of the system to specify the interaction and interconnection of its components, and their functions, states, interfaces, and performance and physical characteristics. The system model can enhance quality, reuse, and shared understanding of the system.

This book can be used by instructors and students to learn how to apply MBSE with SysML, as well as practitioners of MBSE and organizations as a reference approach for their application.

[More information](#)

The Toyota Way to Service Excellence: Lean Transformation in Service Organizations



by

Jeffrey Liker and Karyn Ross

Book Description (from the Amazon Web site):

The world's bestselling Lean expert shows service-based organizations how to go Lean, gain value, and get results—The Toyota Way.

A must-read for service professionals of every level, this essential book takes the proven Lean principles of the bestselling *Toyota Way* series and applies them directly to the industries where quality of service is crucial for success. Jeff Liker and Karyn Ross show you how to develop Lean practices throughout your organization using the famous 4P model. Whether you are an executive, manager, consultant, or frontline worker who deals with customers every day, you'll learn how take advantage of all Lean has to offer.

With this book as your guide, you'll gain a clear understanding of Lean and discover the principles, practices and tools needed to develop people and processes that surprise and delight each of your customers. These ground-tested techniques are designed to help you make continuous improvements in your services, streamline your operations, and add ever-increasing value to your customers. Fascinating case studies of Lean-driven success in a range of service industries, including healthcare, insurance, financial services, and telecommunications, illustrate that Lean principles and practices work as well in services as they do in manufacturing.

Drawn from original research and real-world examples, *The Toyota Way to Service Excellence* will help you make the leap to Lean.

[More information](#)

SYSTEMS ENGINEERING TOOLS NEWS

GENESYS 5.0

Vitech Corporation's GENESYS 5.0 offers model-centric approach to systems engineering across a project team. The collaboration emphasis of GENESYS provides tools for model based systems engineering to:

- collect, import, and manage requirements
- examine use cases, define system behavior, and perform functional analysis
- create and modify architectures
- test models with on-board simulation enabling virtual prototyping from the earliest stages of system definition
- use pre-populated templates to generate industry standard documentation or create the user's own reports with drag-drop functionality.

The new features in GENESYS 5.0 are described [here](#).

Object Process Methodology

If you have never been introduced to Object Process Methodology (OPM), I would strongly suggest that you download and listen to the INCOSE Webinar no 85.

OPM has been developed at the Enterprise Systems Modeling Laboratory (ESML) of the Technion Israel Institute of Technology. The ESML is an international center of knowledge and research in conceptual modeling that focuses on the advancement of OPM as a vehicle to model, analyze, simulate, and verify complex systems in all domains of science and engineering.

In this webinar, Prof. Dov Dori presents this simple yet very powerful methodology by posing and answering 10 questions:

Question 1: What is needed to describe the universe?

Answer: Describing the universe requires things and relations among them.

Question 2: What **is** a thing or what can it **do**?

Answer: A thing can either **exist** or **happen**.

Question 3: What are the things that *exist* in the world?

Answer: **Objects** *exist*. They are **static** – *time-independent*.

Question 4: What are the things that ***happen in the*** world?

Answer: **Processes** *happen*. They are **dynamic** – *time-dependent*.

Question 5: How do **objects** and **processes** *relate* to each other?

Answer: **Processes** *happen* to **objects**.

Question 6: What does a **process** do when it happens to an **object**?

Answer: The **process** *transforms* the **object**.

Question 7: How does a process ***transform*** an object?

Answer: *Transforming means*

- *Creating* an object or
- *destroying* an object or
- *affecting* an object.

Question 8: How does a process ***affect*** an object?

Answer:

- A **process** *affects* an **object** *by changing its state*.
- Hence, **objects** *must be stateful* – *they must have states*.

Question 9: What are the two major aspects of any system?

Answer:

Structure – the static aspect: ***What*** the system is made of.

- *Time-independent*.

Behavior – the dynamic aspect: ***How*** the system changes over time.

- *Time-dependent*.

Question 10: What third aspect is specific to man-made systems?

Answer:

Function – the utilitarian, subjective aspect:

- **Why** is the system built?
- For **whom** is the system built?
- **Who** benefits from operating the system?

Prof. Dori then proceeds to explain the **The Object-Process Theorem: Stateful objects, processes, and relations among them constitute a necessary and sufficient universal ontology**, before illustrating how system function, structure and behavior are expressed bi-modally, in graphics and equivalent text in a **single** model, using a single kind of diagram.

[More information](#)

EDUCATION AND ACADEMIA

Systems Engineering at the Wroclaw University of Science and Technology, Wroclaw, Poland

The Wroclaw University of Science and Technology (WUST) is the most prominent university in south-west Poland. The program in Systems Engineering (SE) was launched in 2011 by the Faculty of Computer Science and Management, one of 16 faculties comprised by WUST. WUST offers 44 fields of study with SE belonging to the newest group of them.

The full-time undergraduate (bachelor) and graduate (master) studies in SE in the Polish language are offered in the periods of seven and three semesters, respectively. Within undergraduate and graduate interdisciplinary programs, students must complete respectively 2400 and 945 hours of courses equivalent to 210 and 90 credits (ECTS) as well as write a relevant degree thesis under the supervision of a faculty member. The programs consist of lectures and practical activities (laboratories, tutorials, seminars, and projects). The general aim of Bachelor and Master Programs is to provide students with essential knowledge and skills for designing, testing, and maintaining complex interdisciplinary technological, production, or service systems, e.g., cyber-physical systems.

See <http://pwr.edu.pl/en/admission/admission-2017-2018> and <http://wiz.pwr.edu.pl/studenci/programy-ksztalcenia/> for more details.

Positions Available at the University of Luxembourg

Software Verification and Validation Lab

There are some exciting Research Associate (RA) and PhD positions at the SnT Centre, University of Luxembourg (UL). Computer Science @ UL is ranked 78th worldwide according to the ranking.

The salaries are highly competitive and the projects are run in collaboration with industry partners in the automotive, satellite, legal, and financial domains, among others. Luxembourg is a highly international location with top infrastructure, a wide range of international schools, and a universal quality health care system. For further details see: svv.lu.

Relevant fields:

- Model-driven engineering
- Test automation, including search-based testing
- Program analysis for security vulnerability analysis
- Applications of Natural Language Processing and Machine Learning to software engineering

The RA positions require a PhD degree in one of the relevant fields. Excellent communication skills in English are a must.

STANDARDS AND GUIDES

ISO/PAS 19450:2015: Automation Systems and Integration -- Object-Process Methodology

ISO/PAS 19450:2015 specifies Object-Process Methodology (OPM) with detail sufficient for enabling practitioners to utilize the concepts, semantics, and syntax of Object-Process Methodology as a modelling paradigm and language for producing conceptual models at various extents of detail, and for enabling tool vendors to provide application modelling products to aid those practitioners.

While ISO/PAS 19450:2015 presents some examples for the use of Object-Process Methodology to improve clarity, it does not attempt to provide a complete reference for all the possible applications of Object-Process Methodology.

[More Information](#)

Editor's notes:

- Chapter 24 of Dov Dori's book cited above provides a summary of this standard.

- INCOSE's Webinar 85, Object-Process Methodology – the new ISO 19450 Standard: Principles and MBSE Applications, is available for INCOSE members.

DEFINITIONS TO CLOSE ON

Methodology is the systematic, theoretical analysis of the methods applied to a field of study. It comprises the theoretical analysis of the body of methods and principles associated with a branch of knowledge. Typically, it encompasses concepts such as paradigm, theoretical model, phases and quantitative or qualitative techniques.

A methodology does not set out to provide solutions - it is, therefore, not the same as a method. Instead, a methodology offers the theoretical underpinning for understanding which method, set of methods, or best practices can be applied to specific case, for example, to calculate a specific result.

The *methodology* is the general research strategy that outlines the way in which research is to be undertaken and, among other things, identifies the methods to be used in it. These *methods*, described in the methodology, define the means or modes of data collection or, sometimes, how a specific result is to be calculated. *Methodology* does not define specific methods, even though much attention is given to the nature and kinds of processes to be followed in a particular procedure or to attain an objective.

Methodology and *method* are not interchangeable. In recent years however, there has been a tendency to use *methodology* as a "pretentious substitute for the word *method*". Using *methodology* as a synonym for *method* or *set of methods* leads to confusion and misinterpretation and undermines the proper analysis that should go into designing research.

[More information](#)

PPI AND CTI NEWS

New Training Locations for 2018

We at PPI and CTI are delighted to announce exciting new locations for our public systems engineering training course programs for 2018. The new locations are New York and Portland, Oregon in the United States, and Rome and Paris in Europe, with others in the planning phase. We look forward to exploring the new cities with you!

Making a Difference Through Delegate Feedback

Delegate feedback from our courses plays a major role at PPI. Apart from triggering any short-term action that is needed, this feedback provides major input to PPI's annual two-week Learning Improvement

Workshop. This year's Learning Improvement Workshop was held in Melbourne over 8 to 19 January, during which senior staff met in Melbourne to examine every aspect of every course against delegate feedback and other opportunities for improving learning outcomes. A little time was available for relaxation; pictured here are Melbourne-based Robert Halligan (Managing Director), Cape town-based Alwyn Smit (Principal Consultant) and Brisbane-based Clive Tudge (Principal Consultant) seen at Southbank Promenade overlooking the Yarra River. Meanwhile the engine behind PPI's training, Production, led by Bekara Freeman, has been hard at work implementing the many improvements that have been devised.



New Faces at PPI

PPI has recently welcomed a new team member, Benjamin Bryant, who will assist PPI in the role as Marketing Executive. Benjamin comes to PPI with experience in previously very niche marketing projects. Benjamin has a diverse background that we are sure will contribute positively to our communications with whoever we interface with.

René King has joined the PPI team in an engineering research and development role. René holds a BSc (Mechanical Engineering) and an MSc (Systems Engineering) from the University of the Witwatersrand, Johannesburg, South Africa. René will be based in Johannesburg, South Africa.

A New PPI Course in the Making for the Infrastructure Sector

PPI is pleased to announce a collaborative effort with Brazilian company Engeflux Engenharia de Sistemas Ltda to bring systems engineering training specifically to the infrastructure sector worldwide.

This new training course will be delivered by George Sousa PhD (Industrial & Systems Engineering). George co-founded Engeflux in 2006 and has been serving as its executive director. Engeflux is a boutique research, development, training and consulting firm that works on systems thinking and engineering-related undertakings. Its purpose is to help leaderships develop successful enterprises. Engeflux offers a holistic approach for coordinating the complete transformation cycle of complex systems, connecting systems thinking and modeling in a team effort for the conception, development, production and utilization of high performance enterprises.

UPCOMING PPI AND CTI PARTICIPATION IN PROFESSIONAL CONFERENCES

PPI will be participating in the following upcoming events.

INCOSE IS2018

7 – 12 July 2018

Washington, DC, USA

PPI AND CTI EVENTS

Systems Engineering 5-Day Courses

Upcoming locations include:

- Las Vegas, NV, United States of America
- Amsterdam, the Netherlands
- Melbourne, Australia

Requirements Analysis and Specification Writing 5-Day Courses

Upcoming locations include:

- Las Vegas, NV, United States of America
- Amsterdam, the Netherlands
- Pretoria, South Africa

Systems Engineering Management 5-Day Courses

Upcoming locations include:

- Stellenbosch, South Africa
- Amsterdam, the Netherlands
- Pretoria, South Africa

Requirements, OCD and CONOPS in Military Capability Development 5-Day Courses

Upcoming locations include:

- Amsterdam, the Netherlands
- Melbourne, Australia
- Pretoria, South Africa

Architectural Design 5-Day Course

Upcoming locations include:

- Pretoria, South Africa
- Adelaide, Australia
- London, United Kingdom

Software Engineering 5-Day Courses

Upcoming locations include:

- Amsterdam, the Netherlands
- Melbourne, Australia
- London, United Kingdom

Human Systems Integration Public 5-Day Courses

Upcoming locations include:

- Sydney, Australia
- Melbourne, Australia

CSEP Preparation 5-Day Courses (Presented by Certification Training International, a PPI company)

Upcoming locations include:

- Laurel, MD, United States of America
- London, United Kingdom
- Las Vegas, NV, United States of America

Kind regards from the SyEN team:

Robert Halligan, Editor-in-Chief, email: rhalligan@ppi-int.com

Dr. Ralph Young, Editor, email: ryoung@ppi-int.com

Suja Joseph-Malherbe, Managing Editor, email: smalherbe@ppi-int.com

Project Performance International

2 Parkgate Drive, Ringwood North, Vic 3134 Australia Tel: +61 3 9876 7345 Fax: +61 3 9876 2664

Tel Brasil: +55 11 3958 8064

Tel UK: +44 20 3608 6754

Tel USA: +1 888 772 5174

Web: www.ppi-int.com

Email: contact@ppi-int.com

Copyright 2012-2018 Project Performance (Australia) Pty Ltd, trading as Project Performance International.

Tell us what you think of SyEN. Email us at syen@ppi-int.info.