Supporting the CIPR S*Pattern:

Background on MBSE, PBSE, and the ASELCM S*Pattern

Produced by:

INCOSE Critical Infrastructure Protection & Recovery Working Group

INCOSE MBSE Patterns Working Group

International Council on Systems Engineering

ET2016 Report-- Supporting Background Document V1.4.3

Contents

1 Introduction

1.1 Document Purpose and Context

The purpose of this document is to briefly summarize the general nature of three systems engineering topics, including how they are related to each other, their intended uses associated with the subjects and objectives of the INCOSE-IEEE Energy Tech 2016 Conference, and in particular the Critical Infrastructure Protection and Recovery (CIPR) S*Pattern (described elsewhere). For those interested in learning more about these subjects, further reference lists are included for each.

This document was originated following the Energy Tech 2016 Conference, in which these subjects were discussed and applied to describe its Track 1 and other discussions. As a part of the follow-up generation of a record of the conference, the INCOSE Critical Infrastructure Protection and Recovery (CIPR) Working Group and the INCOSE MBSE Patterns Working Group agreed that this short guide would be of value to those interested in learning to apply the related assets and methods, in support of the CIPR S*Pattern.

1.2 Intended Audience

This document does not assume technical knowledge of the conceptual methods and assets that it summarizes. Instead, it is intended for leaders, decision-makers, and others interested in a digestible executive overview of the subject matter addressed, and in guidance to additional references.

1.3 Document Scope

This document does not cover the CIPR S*Pattern, described elsewhere, but is pre-requisite background for that subject. This document is not a technical or detailed reference--it provides additional references for that purpose. This document does not describe the ET2016 Conference, or its sessions. It describes background reference information that was used in other conference materials for that purpose.

2 Overview, Proceeding from General to Specific

The Section 2 core of this document is organized around high level summaries of three subjects that are related to each other in the sense that they are each specializations of the preceding, in sequence. Together, they support the CIPR Pattern, described separately. Figure 1 notes the progressive specialization of:

Covered in this document:

1. Model-Based Systems Engineering (MBSE)
2. Pattern-Based Systems Engineering (PBSE) and the S*Metamodel, as an approach to MBSE
3. The Agile Systems Engineering Life Cycle Management (ASELCM) S*Pattern, described by PBSE

Separately documented:

4. The Critical Infrastructure Protection and Recovery (CIPR) S*Pattern, an application of the ASELCM S*Pattern

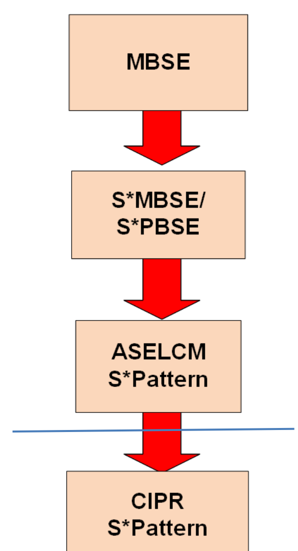


Figure 1: Four subjects, in order of increasing specialization

2.1 Model-Based Systems Engineering (MBSE)

2.1.1 Summary statement of general nature, intended use, value, current state, and trends

General Nature: Humanity has a powerful historical interest in understanding collections of interacting parts that we call "systems". This includes human-planned and constructed systems, other systems encountered in nature, and combinations of the two, at every scale from the sub-atomic, through machines the size of aircraft carriers or global networks, to cosmological systems of the universe. The relative degree and exploitation of that understanding has accelerated rapidly in the most recent three hundred years of the revolution in science, technology, engineering, and mathematics (STEM) knowledge, lifting the quality and possibility of human life, in a world of challenges and opportunities.

One of the essential keys to this revolution has been the development of "models"—representations of systems that provide powerful ways to analyze, understand, plan, and predict the behavior of systems encountered in nature and human-performed design. The Scientific Revolution and its growth in human understanding of natural laws are founded on the discovery and application of effective models. These express not only what we know, but also accountings of what we don't yet understand, what is at risk or uncertain, and perceived limitations.

Using visual diagrams, mathematics, tables of related information, or other data structure forms, these models may appear differently, across the domains of civil engineering, chemistry, mechanics, electrical engineering, biology, economic markets, distribution networks, transportation systems, and otherwise. The disparate knowledge and varying technical disciplines they support are unified by an underlying ability to apply the common science and engineering of general systems (to the extent that these are known), and this includes general systems models. In the space of domain specific models, we find the mechanical and civil engineer's structural blueprints and strength calculations, the electrical engineer's schematic diagrams and signal processing equations, the chemist's process flow diagrams and reaction equations, the biologist's signaling pathway diagrams and DNA maps, and many other forms of models.

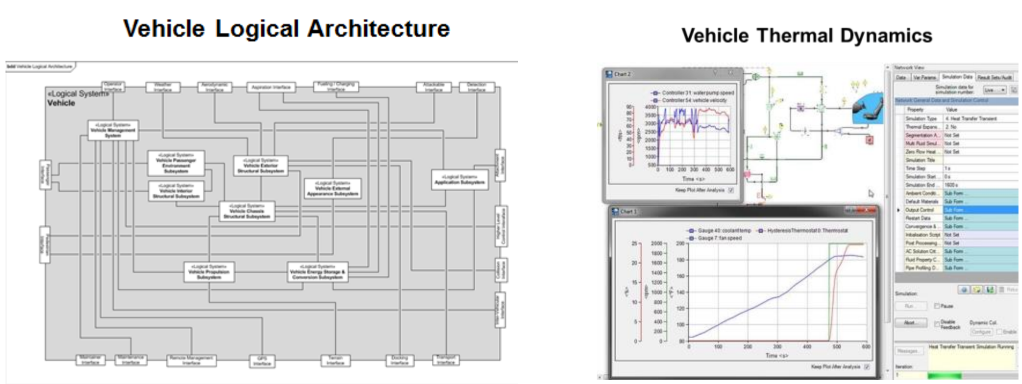


Figure 2: Models from Many Sources, Disciplines

Most engineering disciplines (e.g., CE, ME, ChE, EE) arose from combining (1) human efforts to plan, analyze, construct, use, and improve utilitarian systems with (2) growing scientific knowledge of specific domains (e.g., mechanics, chemistry, electrical science). The engineering disciplines thus arose somewhat separately from each other, each with its own domain-specific physical phenomena, related scientific knowledge, and terminology. Most of these disciplines have enjoyed hundreds of years of maturation, and each has developed their own successful models that are used to represent their subject matter, solve problems, and engineer improvements.

Systems Engineering (SE), concerned with general systems (collections of interacting parts) spanning all these domains, arose out of a different history, and much more recently (roughly fifty years) than other engineering disciplines. Still in its relatively early days, systems engineering did not originate alongside a physical science of general systems phenomena, with phenomena-specific models, like the other engineering disciplines. Instead, SE at first relied upon the use of prose (natural language) and other descriptions to accomplish some of its earliest victories. As shown in the References, these originally included organizing the broad framework of work processes and general concepts necessary to manage systems across their life cycles, integrating the work of other, domain-specific engineering disciplines.

Only more recently has SE moved to the use of generalized system models not unique to specific technologies. The resulting (and still young) approach is sometimes referred to as Model-Based Systems Engineering (MBSE). There is evidence of its (limited, scattered) use in serious systems practice going back nearly 40 years, but wider adoption rates have not accelerated until the last 10-20 years, and are still in progress. This has included the growth of a needed infrastructure of modeling methods and principles, several general systems modeling languages, automated toolsets, formal standards and practices, references, and educational efforts. The INCOSE Board of Directors recently declared the strategic objective of transformation of systems engineering to a model-based discipline. While seen as a sign of progress, it also illustrates that this growing model-based practice is not yet the widespread nature of all SE effort.

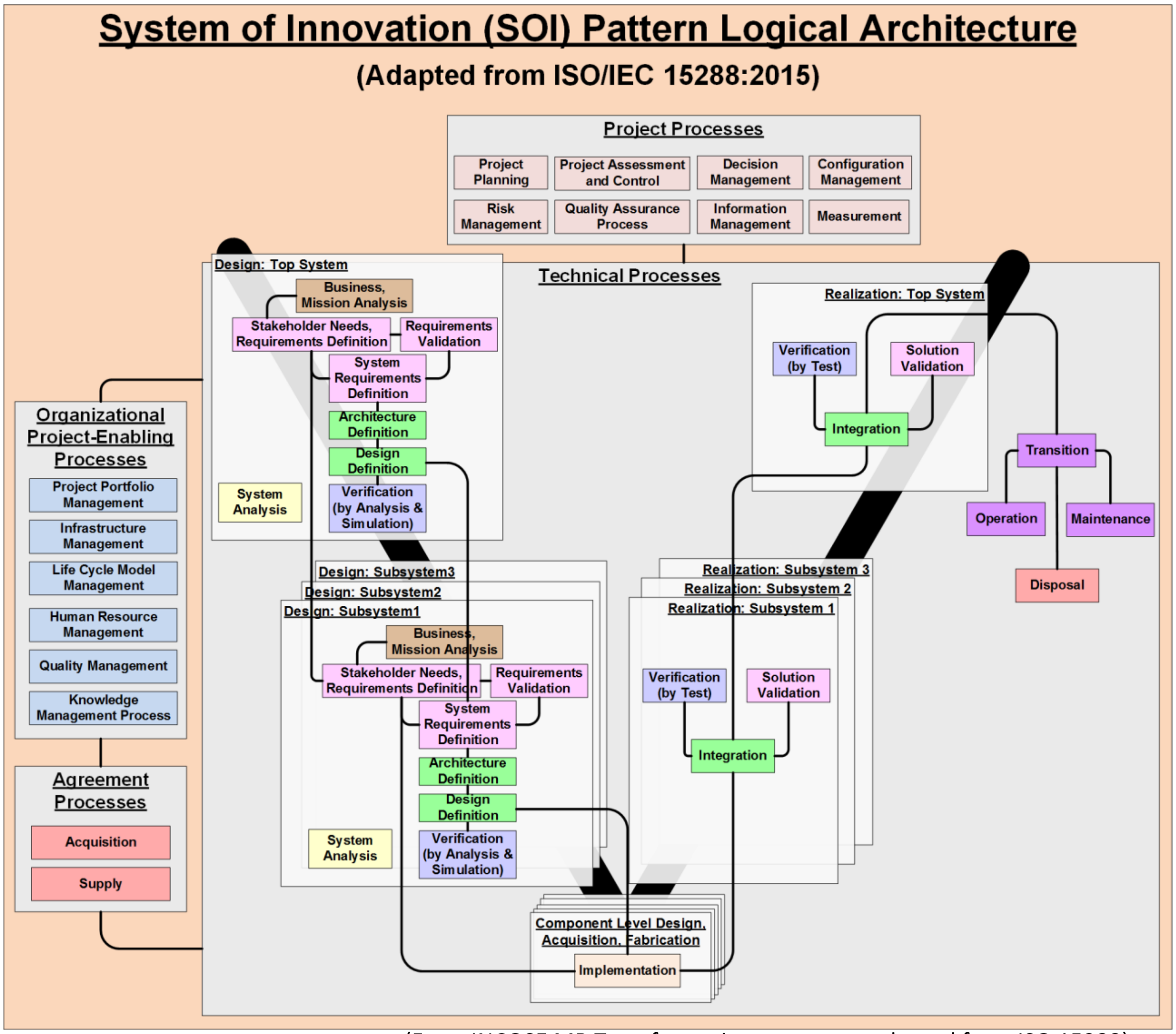
Intended MBSE Use, Value: Section 2.1 is less concerned with how MBSE is performed than with summarizing its intended use and value. While it is helpful to realize that all the values and intended uses of MBSE are based on more effective representations of systems than earlier alternative means, this still leaves us with the question: "effective to whom, and for what purpose?" To answer this, the earlier history of systems engineering is very helpful, because it has provided a general list of "system life cycle management processes", summarizing a list of cradle-to-grave activities that organizations must in some way perform, independent of the details of how, in order to conceive, plan, construct, use, analyze, retire, improve, or otherwise manage systems of any type, over their life cycles. These processes are illustrated by Figure 3. It is these System Life Cycle Management Processes that INCOSE has chosen (in the INCOSE SE Handbook and ISO 15288, of the References) to express an MBSE deployment strategic planning framework for those who would assess or plan uses of MBSE.

System models are information, and that information is intended to inform the processes shown in Figure 3. Although some of these purposes, and the people involved in them, are relatively technical in nature, in many other cases there are important decisions or other activities performed by relatively non-technical people and consideration. The scope of system models are not just technical matters, but include in particular consideration of the (often subjective) values of various non-technical stakeholders, including consumers, communities, senior leadership, and others. This is particularly evident when considering the entire life cycle of systems. Figure 4 indicates the many different types of stakeholders in the use of MBSE, as identified by the INCOSE MBSE Transformation, and it can be seen that some of the largest populations are non-technical people

Current State, Trends: Both the current state as well as trends of change of MBSE might be summarized across the following different dimensions, described here in the form of questions:

1. How widespread is current MBSE use, across organizations, domains, processes, and projects?
2. How impactful is current MBSE use, in what domains and processes, with what stakeholder impacts?
3. What differences in current MBSE approaches may be observed, and with what degree of use?
4. How integrated is current MBSE use, across processes, organizations, toolsets, enterprises?
5. Are the potential stakeholders / beneficiaries organized as a community?

6. What are the possible paths to MBSE adoption, and what adoption resources are available?
7. What obstacles to MBSE progress are identified, and with what plans to address them?
8. How difficult, costly, or time-consuming is MBSE adoption, compared to benefits, by area?



(From INCOSE MB Transformation resources; adapted from ISO 15288)

Figure 3: What are the uses of MBSE models?

Questions (1-2) above are the subject of attention of a current INCOSE Model-Based Transformation project (see References), but collected information on these to date is early and spotty enough that it should be considered anecdotal. The life cycle process of Figure 3 and the stakeholder groups of Figure 4 are being used by this INCOSE project as a means to further assess these. Growth in the accumulated number of MBSE instruction classes and enrollments (see References), attendance at INCOSE IW MBSE Workshops, publication of related papers and references, and installation of automated tooling may be judged as indicating positive growth, but don't yet provide a lot of insight into overall rates of use and impact, versus anecdotal samplings. Some reports (see ASSESS Reference cited) indicate that the rate of growth of at least key aspects of MBSE may currently be constrained by lack of available practitioner expertise.

A reasonable sense of the range of MBSE approaches (but not their relative degrees of use or outcomes) referenced by Question (3) above may be gained from the INCOSE survey of MBSE methodologies found in the References. However, it should be noted that survey is focused almost exclusively (as is much of MBSE attention) on technical methods, not other, less technical aspects of MBSE. The next section (2.2) addresses a specific approach to some of the MBSE challenges.

Question (4) above is one of the most commonly-referenced challenges that are noted by enterprises and practitioners—although not in the form of quantitative measurement. Instead, frustration is periodically expressed that different automated MBSE and other toolsets and methods are not perceived as integrated with each other to the degree desired, or the life cycle information about different systems is not seen as directly compatible across the organizations, methods, and tools of those areas. This continues to receive many years of efforts on related standards (see References on data exchange and access standards such as STEP, XMI, as well as shared frameworks of UPDM, UAF).

The stakeholder classes referred to in Question (5) above are listed in Figure 4. Only small subsets of these (e.g., the model authors of INCOSE, the technical standards bodies, or the tooling suppliers) are currently believed to be organized as a community to reflect those stakeholder interests within the overall MBSE transformational activity.

Population <-- Size (Log)	**Stakeholders in A Successful MBSE Transformation (showing their related roles and parent organizations)**	Industry & Gvmt. Initiatives	Orgs. Internalizing MBSE, Including Gvmt Contractors & Commercial	Vendors of MBSE Tooling and Services	Academia and Researchers	Technical Societies, Other Non-Technical Organizations	
Model Consumers (Model Users):							
****	Non-technical stakeholders in various Systems of Interest, who acquire / make decisions about / make use of those systems, and are informed by models of them. This includes mass market consumers, policy makers, business and other leaders, investors, product users, voters in public or private elections or selection decisions, etc.	X	X			X	
**	Technical model users, including designers, project leads, production engineers, system installers, maintainers, and users/operators.	X	X			X	
*	Leaders responsible to building their organization's MBSE capabilities and enabling MBSE on their projects	X	X			X	
Model Creators (including Model Improvers):							
*	Product visionaries, marketers, and other non-technical leaders of thought and organizations	X	X		X	X	
*	System technical specifiers, designers, testers, theoreticians, analysts, scientists	X	X		X	X	
*	Students (in school and otherwise) learning to describe and understand systems				X	X	
*	Educators, teaching the next generation how to create with models	X	X		X		
*	Researchers who advance the practice		X	X	X		
*	Those who translate information originated by others into models	X	X		X	X	
*	Those who manage the life cycle of models	X	X		X	X	
Complex Idea Communicators (Model "Distributors"):							
**	Marketing professionals	X	X	X		X	
**	Educators, especially in complex systems areas of engineering and science, public policy, other domains, and including curriculum developers as well as teachers	X	X	X	X		
**	Leaders of all kinds	X	X	X	X	X	
Model Infrastructure Providers, Including Tooling, Language and Other Standards, Methods:							
*	Suppliers of modeling tools and other information systems and technologies that house or make use of model-based information			X			
*	Methodologists, consultants, others who assist individuals and organizations in being more successful through model-based methods	X	X	X	X		
*	Standards bodies (including those who establish modeling standards as well as others who apply them within other standards)	X				X	
INCOSE and other Engineering Professional Societies							
*	As a deliverer of value to its membership					X	
*	As seen by other technical societies and by potential members					X	
*	As a great organization to be a part of					X	
*	As promoter of advance and practice of systems engineering and MBSE					X	

Figure 4: Who Are the Stakeholders in MBSE As An Approach?

(From INCOSE MB Transformation resources)

Questions (6) and (8) above are the subjects of a current project by the INCOSE MBSE Transformation, and a future related publication.

At least the identification of the (Question (7)) obstacles above is addressed through reports in (Friedenthal 2017) (Schindel 2016) (Li et al 2017) (Walsh 2017). More fundamental issues of basic organizational change management must also be considered a part of this landscape. For the reported obstacle of limited available practitioner expertise, Section 2.2 below addresses one approach. The expanded use of model representations as the basis for decision-making and other purposes across the life cycle management processes implies that these models must increasingly be trusted, so that a basis for understanding and managing trust in those models is required. This is the subject of the ASME Computational Model Verification and Validation standards and guideline making activity reported elsewhere here.

2.1.2 Related INCOSE and other entities and their activities

INCOSE and other entities, and their related activities concerned with MBSE or related model-based disciplines, include the following. Related references are listed in the References Section 3 later below.

1. The INCOSE / OMG MBSE Initiative, including its Challenge Teams, Activity Teams, and the annual INCOSE IW MBSE Workshop, in its tenth year, all in support of advancement of model-based systems engineering practice: http://www.omgwiki.org/MBSE/doku.php
2. The Object Management Group (OMG) Systems Engineering Domain Special Interest Group (SE DSIG): Industry standards-making in support of Systems Modeling Language (SysML), and including the Unified Architecture Framework (UAF) and (UPDM) predecessor Unified Profile for DoDAF and MODAF: http://syseng.omg.org/
3. ISO Technical Committee 184/Sub-Committee 5 (TC184/SC5), concerned with ISO/PAS 19450:2015 Automation systems and integration — Object-Process Methodology (OPM) systems modeling language and methodology.
4. Air Force Wright Aeronautical Laboratories Integrated Computer-Aided Manufacturing (ICAM) Architecture IDEF0 modeling language, described by NIST FIPS standard for Integration Definition for Function Modeling (IDEF0). The FIPS was later obsoleted by NIST, but use of IDEF0 continues, including as the basis of representation of the ISO15288 life cycle management processes in the current version of the INCOSE *Systems Engineering Handbook*: http://www.idef.com/idefo-function_modeling_method/
5. System Dynamics Society: System dynamics is a computer-aided approach to policy analysis and design. It applies to dynamic problems arising in complex social, managerial, economic, or ecological systems — literally any dynamic systems characterized by interdependence, mutual interaction, information feedback, and circular causality. http://www.systemdynamics.org/
6. NAFEMS is the International Association for the Engineering Modelling, Analysis and Simulation Community, focused on the practical application of numerical engineering simulation techniques such as the Finite Element Method for Structural Analysis, Computational Fluid Dynamics, and Multibody Simulation. http://www.nafems.org/
7. NAFEMS / INCOSE Systems Modeling & Simulation Working Group (SMSWG), concerned with standards for integrated representation of executable system simulations as a part of the systems modeling framework: http://www.omgwiki.org/MBSE/doku.php?id=mbse:smswg

8. INCOSE MBSE Initiative Modeling and Simulation Interoperability (MSI) Challenge Team http://www.omgwiki.org/MBSE/doku.php?id=mbse:modsim

9. The ASME Standards Committees on Verification and Validation in Computational Modeling and Simulation are concerned with standards as to the verification and validation of computational models: https://cstools.asme.org/csconnect/CommitteePages.cfm?Committee=100003367

2.2 MBSE Pattern-Based Systems Engineering (PBSE), and the S*Metamodel

2.2.1 Summary statement of general nature, intended use, value, current state, and trends

This section summarizes Pattern-Based Systems Engineering (PBSE) Methodology, a form of MBSE based on use of the S*Metamodel. In this approach, re-usable, configurable S*Models (which are MBSE models conforming to the S*Metamodel) are created, then used and re-used across a range of different system configurations or family members, and improved over time as the point of distillation of learning. These re-usable, configurable S*Models are called S*Patterns to emphasize their recurring use, and are model-based substantial extensions of earlier, pre-MBSE engineering patterns.

As shown in Figure 5, methodologies for systems engineering are concerned with both (1) the engineering process and (2) the information that is consumed and produced by that process. In comparison to a strong historical systems engineering emphasis on process, this methodology increases the relative emphasis on the information passing through that process, with favorable impacts on process outcomes. That *information* is in the form of explicit MBSE system models of stakeholder value, requirements, design, risk, and other aspects, comparable in many aspects to other MBSE methodologies (Estefan 2008), but also strengthened (by the S*Metamodel) in certain areas, and compatible with contemporary modeling languages and tools. The emphasis on that *information* is on description of the engineered system, not the system of engineering. PBSE extends this shift by focusing on how information is produced and consumed beyond a single project—it asks how we make use of what we previously learned:

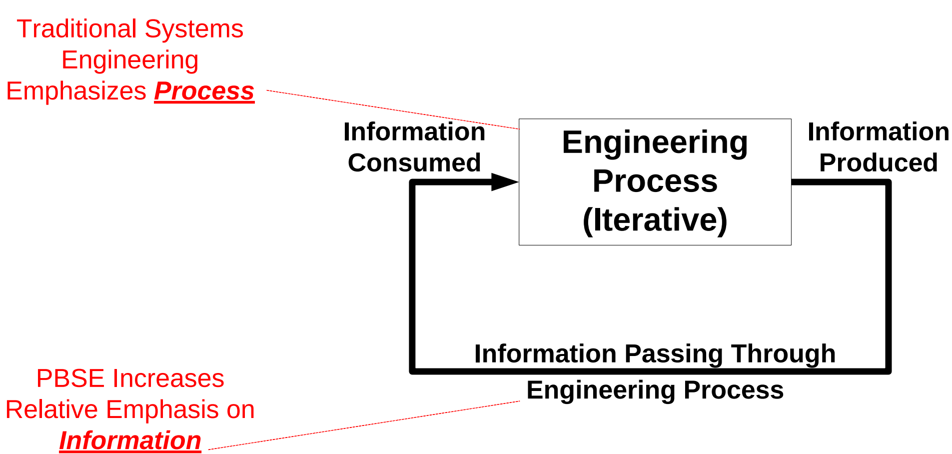


Figure 5: The Engineering Process Consumes and Produces Information, Iteratively

PBSE builds on and extends earlier work in patterns, through introduction of MBSE models (many historical engineering patterns were not explicit MBSE models), expansion of pattern scope to whole system families, platforms, and domains (as opposed to smaller-scale localized patterns), and foundation on a stronger MBSE metamodel to express systemic phenomena critical to engineering applications with clearer connection to scientific understanding of systems phenomena.

Benefits of the PBSE approach include:

1. Reduced recurring cost of modeling, along with shift of emphasis, from "learn how to model" (an abstract skill) to "learn *the* model (pattern)" already in use by an enterprise, a more concrete knowledge of the enterprise's products, simplifying the introduction of MBSE methods.
2. Very rapid generation of first draft configured system requirements, design FMEAs, and other historical systems engineering artifacts, of higher quality and completeness.
3. A detailed MBSE approach to Platform Management for system families and product lines.
4. Direct support of contemporary interests in product line engineering, reference architectures, architectural frameworks, ontologies.
5. A structured means of representing and accessing learning across organizations.
6. Strengthened semantic integration of system requirements with the rest of the model.
7. Rapid generation of D-FMEA, A-FMEA, P-FMEA, and FTA analyses of risks on a more systematic and complete basis, more deeply integrated with the rest of the system model.
8. Compatibility with contemporary modeling language standards.
9. Direct mapping into contemporary modeling tools, PLM information systems, and other COTS and enterprise systems, increasing the value of existing information technologies.
10. Deeper support for federated data across differing information systems, for integration with emerging open systems life cycle standard technologies.
11. Strong expression of fitness landscapes as the basis for selection, trades, improvements, decisions, innovations, configuration, and understanding of risk and failure.
12. Strong expression of life cycle and operational states of systems
13. Explication of the System Phenomenon as a real world-based science and math foundation for systems engineering, amenable to systems science, connected to historical math/science models of other engineering disciplines, and encouraging discovery and expression of systemic phenomena of higher-level systems.
14. Explication of key patterns in both Engineering Systems and Systems of Engineering, including the Embedded Intelligence (Management) Pattern, the Systems of Innovation Pattern generalization of ISO 15288, and its Agile Systems Life Cycle Pattern form.

*2.2.1.1 Introduction to the S*Metamodel*

Engineering disciplines such as ME, EE, ChE, CE, etc., are based upon underlying models of phenomena (mechanical, electrical, chemical, etc.) that are the fruits of physical sciences, mathematics, and philosophy. Newton's laws of motion, Maxwell's equations, and other underlying models describe aspects of the nature of subject systems, not engineering procedures for those systems, while opening up many procedural avenues that operate within the constraints of those underlying models of Nature. In a similar fashion, the S*Metamodel describes the underlying "systemic" aspects of systems of interest, based upon the fruits of science and mathematics. In the tradition of those same physical sciences (Gingerich 2004), these underlying models (whether specific to one technical discipline or systems in general) seek the "smallest model" capable of (verifiably) describing or explaining the phenomena of interest (Schindel 2011).

The rise of a number of MBSE methodologies (Estafan 2008) and system representation standards (ISO10303-233 2012) has provided many of the needed elements of that underlying "smallest model" framework, and the S*Metamodel builds on those, while adding some important missing and compressing other redundant aspects. Throughout, this is in the spirit of seeking out the smallest (simplest) verifiable model necessary to describe systems for purposes of engineering and science.

Figure 6 is a simplified summary of some of the key portions of the S*Metamodel. This diagram is not the sort that is produced in a related engineering project (illustrated in Figure 7), but instead is a representation of the underlying classes and relationships upon which those project-specific models are based. Those project-specific models may be in any modeling language (including but not limited to SysML, IDEF, or otherwise) and supported by any engineering tool or information system, as in **Figure 7**.

The conceptual awareness extract of **Figure 6** is a simplified representation of selected S*Metaclasses and key relationships that connect them, (the formal S*Metamodel is represented in UML) and summarizes some of the most important S*Metamodel concepts. Additional references are listed later.

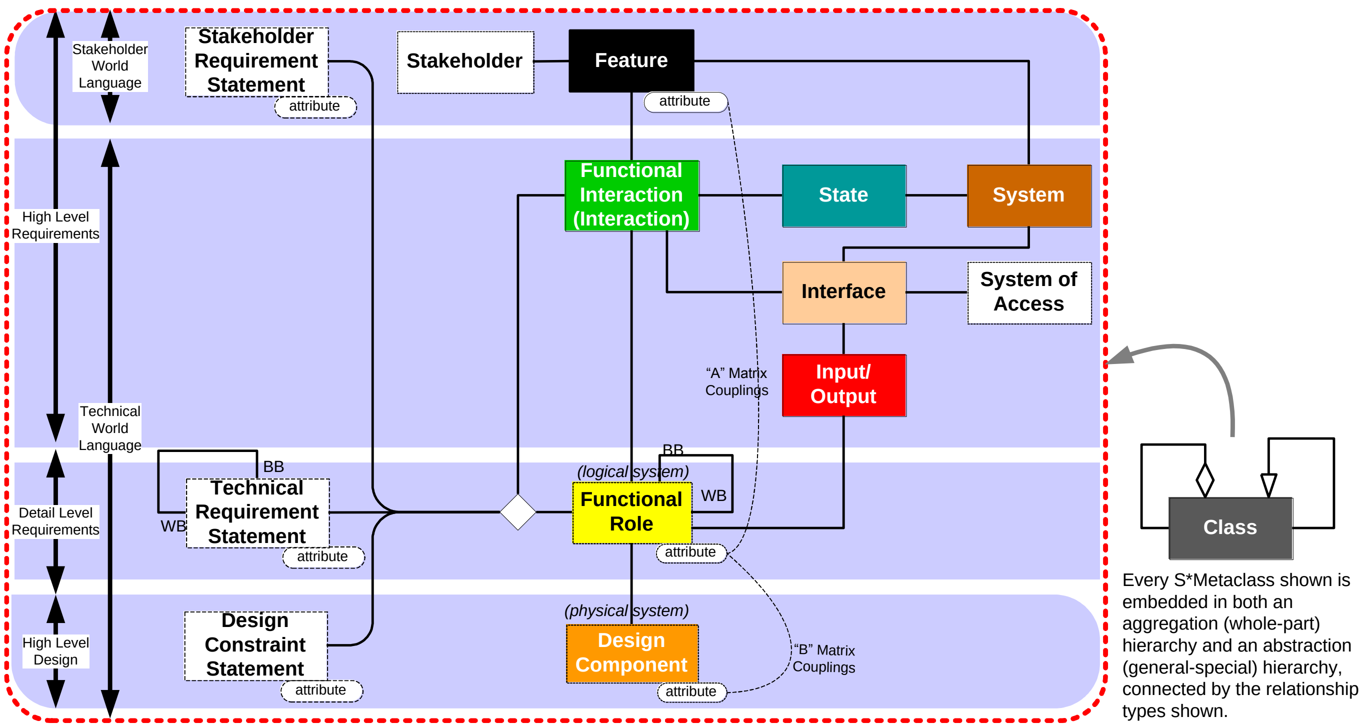


Figure 6: Summary of Some of the Key Portions of S*Metamodel

In reading **Figure 6**:

- The color coding of **Figure 6** provides informal reminder of stereotypes mapping in formal modeling languages (e.g., SysML), and related views--especially for non-technical views and viewers.
- A Feature is an aspect of the behavior or performance of a system that has stakeholder value, described in the concepts and terminology of that stakeholder, and serving as the basis of selection of systems or system capabilities by or on behalf of the Stakeholder. Features are parameterized by Feature Attributes, which have subjective stakeholder valuations.
- (Functional) Interaction means the exchange of energy, force, mass, or information between system components, each of which plays a (Functional) Role in that Interaction. This is the traditional systems engineering "function", performed by a given system.
- Functional Roles are described solely by their behavior, and parameterized by Role Attributes which have objective technical valuations.

- Requirements Statements are prose or other descriptions of the behavior of Functional Roles during Interactions. They are the prose descriptive equivalent to the Roles they describe, and are parameterized by Requirements Attributes which are identical to the related Role Attributes.

It is important to remember that the above concepts are underlying metamodel concepts. Practical S*Models are represented using whatever modeling languages and tools may be of value. **Figure 7** illustrates selected SysML views of a vehicle S*Model.

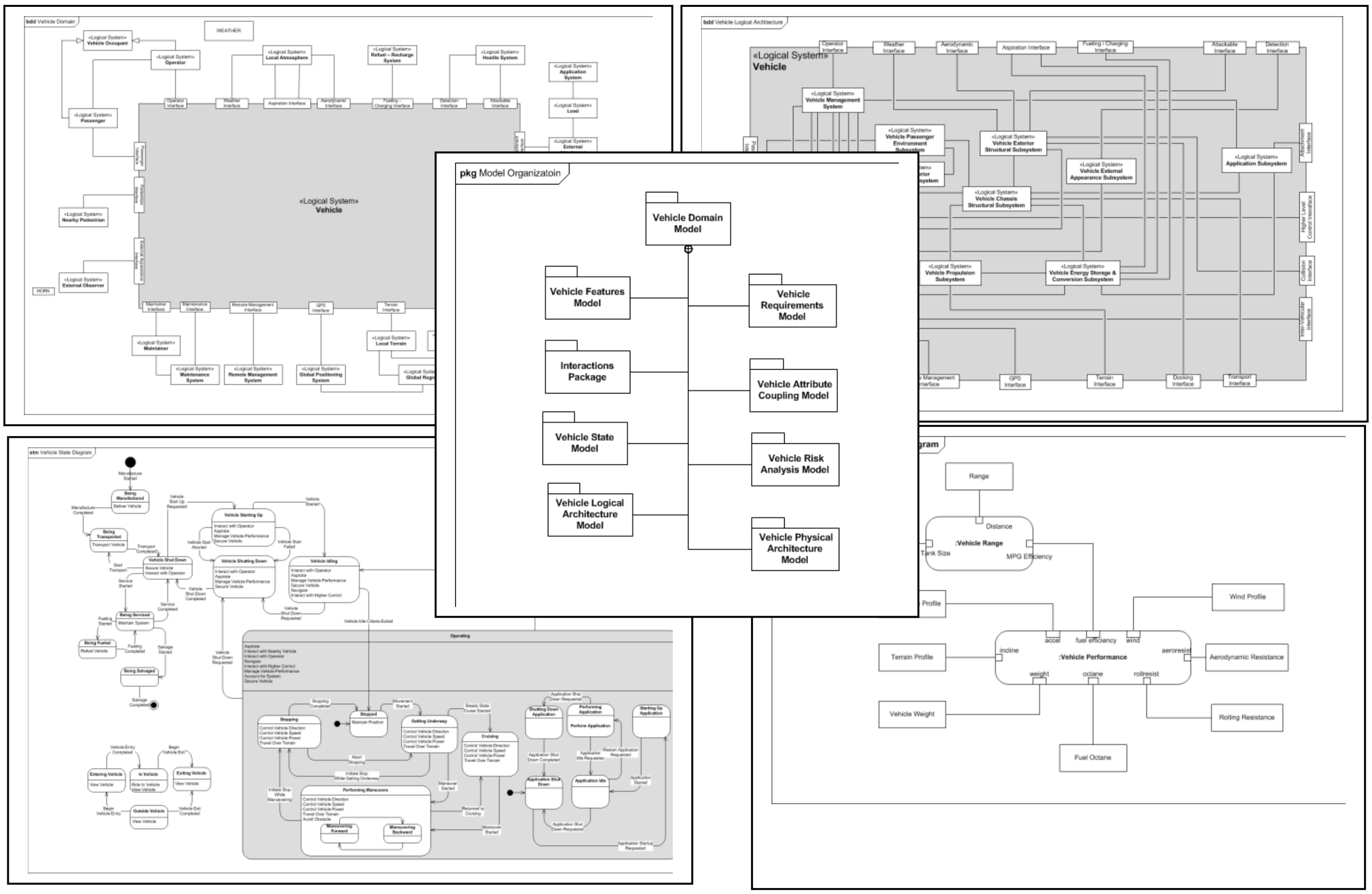


Figure 7: Example S*Model Extracts

*2.2.1.2 S*Models and S*Patterns*

S*Models are MBSE models conforming to the S*Metamodel (**Figure 6**). (That is, they contain Features, Interactions, Roles, States, Design Components, Interfaces, Requirements, Attributes thereof, couplings between them, etc.). S*Patterns are S*Models (with all their parts) that have been constructed to cover a system configuration space bigger than single system instances, and are sufficiently parameterized and abstracted to be configurable to more specific S*Models, and thereby reusable, as in **Figure 8** (Schindel and Smith 2002), (Schindel 2005a), (Bradley et al 2010), (Schindel, Peterson 2013).

Like S*Models, S*Patterns may be expressed in any system modeling language (e.g., SysML, IDEF, etc.) and managed in any COTS system modeling tool or repository, by means of an S*Metamodel mapping or profile for each such tool.

As also illustrated by the “up stroke” in **Figure 8**, discoveries are encountered during projects involving configured S*Models, and some of these cause improvements to be fed back to the S*Pattern, which thereby becomes a point of accumulation of all learning about what is known about the family of systems that pattern represents. This reduces the amount of “searching” required of future project users to take advantage of what is already known, and in particular reduces the likelihood of re-learning the same lessons by mistake and re-work. Notice that this “distillation and abstraction” process is quite different than simply accumulating a lot of separate “lessons learned” in a large searchable space—it is instead translating them into their foundational implications at the pattern level, for future users of the pattern, as a single point of learning well-known and accessible to distributed users. It is the S*Pattern equivalent of representation of scientific learning, and is also a model-based analogy to governance of prose engineering standards.

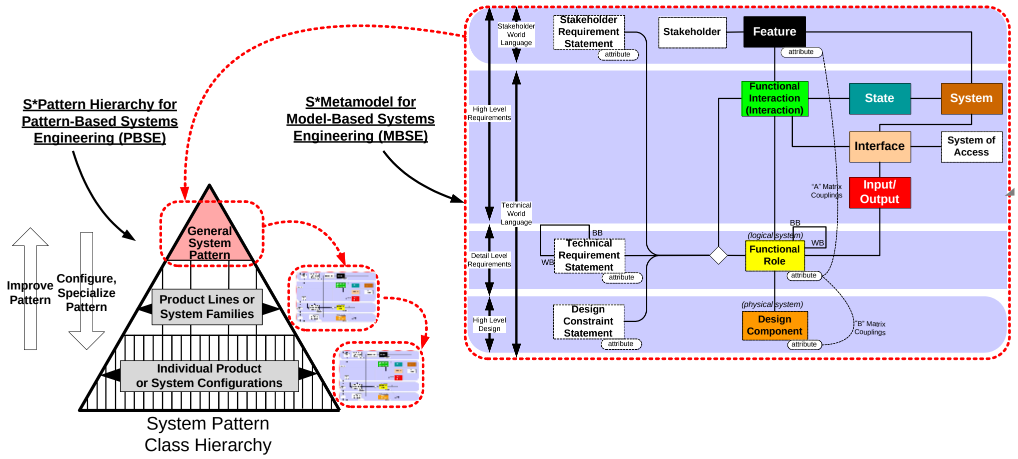


Figure 8: S*Patterns are S*Models of System Families, Configurable/Reusable as Models of Individual System Types

2.2.1.3 Impact on System Life Cycle Processes

The above subjects are entirely about the information flowing through the processes of Figure 5, and not about what those processes are. However, the nature of the information flowing through, described by the S*Metamodel, significantly improves the details of how those processes work—specifically, their efficiency, productivity, and effectiveness.

The highest level summary of the impact on the processes is summarized by the left side of **Figure 9**, which shows the separation of business processes into a Pattern Management Process, typically performed by very few people, managing the S*Pattern, and the Pattern Configuration Process, typically performed by a larger number of people spread across multiple system delivery projects or life cycle activities. This larger group’s work is made more efficient, productive, or effective, while the smaller group’s work is made more impactful. These impacts are basic to the nature of PBSE, and are caused by:

1. Expertise and work of a few expert Pattern Owners is leveraged across numerous Pattern Configuration users, making both more effective.
2. The more numerous pattern users in the Pattern Configuration Process “learn the model, not modeling”. For example, it is much more feasible for many automotive engineers to learn and effectively utilize their company’s Vehicle Pattern in individual projects than to expect them to learn how to model “from scratch” and perform modeling across numerous projects.

3. Pattern content is typically more complete than what would occur at first to an engineer or team on one project. For example, many S*Pattern requirements statements and failure modes and effects will not have occurred to project engineers, who will find them of immediate value.
4. A configured pattern is only a “first draft” of specifications on a project—as if an expert assistant was available to write a first draft. Nothing prevents the project from improving that draft, like any other draft. PBSE is not intended to turn over human thinking to a pattern or computer.

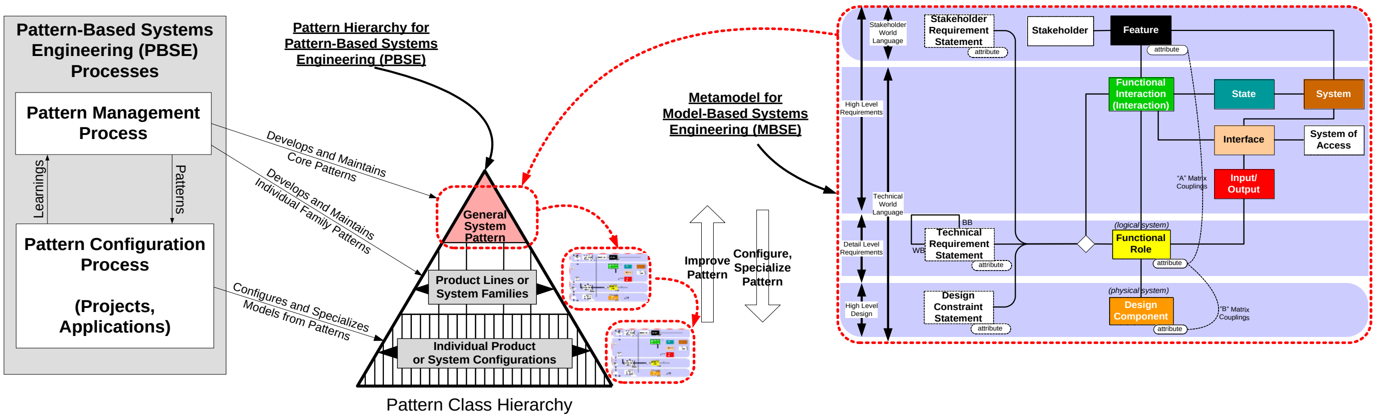


Figure 9: Separation of Pattern Management Process from Pattern Configuration (Project) Process

Both of the two general processes on the left side of **Figure 9** have their own MBSE forms of the ISO/IEC/IEEE 15288 standard Life Cycle Processes (ISO 15288 2015), enhanced in each case by the PBSE nature of the approach. This is summarized by Figure 10, which further details what Figure 5 only summarized. (For additional detail, each of the Process Areas shown has in turn been detail modeled using MBSE models of the processes.)

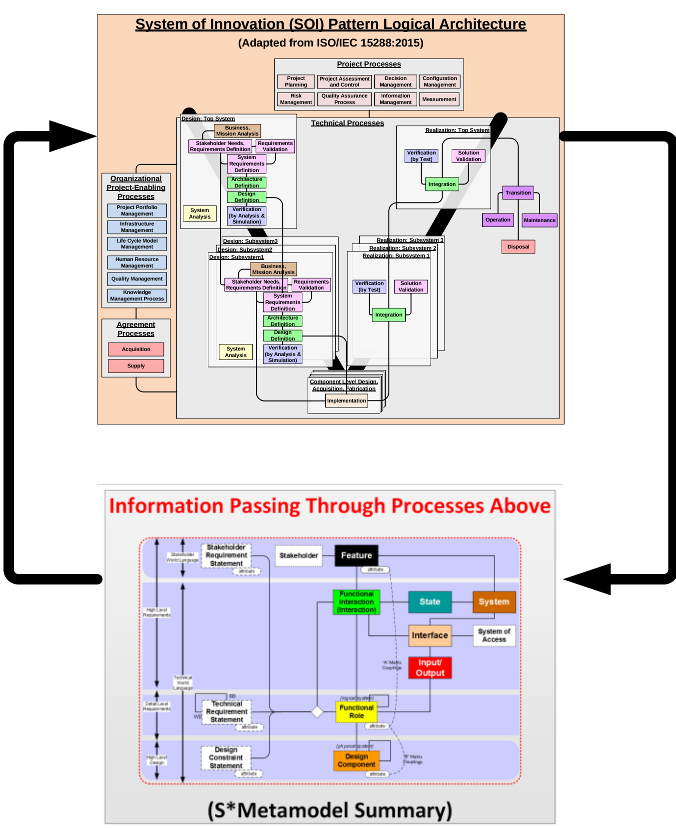


Figure 10: The ISO 15288 Processes Apply, Enhanced by PBSE (Compare to Figures 3, 5, 6)

2.2.1.4 Applications to Date

PBSE has been applied for about two decades, across a variety of domains in commercial, defense, and institutional environments. **Figure 11** lists some of these, and the references provide example content.

Medical Devices Patterns	**Construction Equipment Patterns**	**Commercial Vehicle Patterns**	**Space Tourism Pattern**
Manufacturing Process Patterns	**Vision System Patterns**	**Packaging Systems Patterns**	**Lawnmower Product Line Pattern**
Embedded Intelligence Patterns	**Systems of Innovation (SOI) Pattern**	**Consumer Packaged Goods Patterns (Multiple)**	**Orbital Satellite Pattern**
Product Service System Patterns	**Product Distribution System Patterns**	**Plant Operations & Maintenance System Patterns**	**Oil Filter Pattern**
Life Cycle Management System Patterns	**Production Material Handling Patterns**	**Engine Controls Patterns**	**Military Radio Systems Pattern**
Agile Systems Engineering Life Cycle Pattern	**Transmission Systems Pattern**	**Precision Parts Production, Sales, Engineering Pattern**	**Higher Education Experiential Pattern**

Figure 11: Examples of PBSE Applications to Date

2.2.2 Related INCOSE and other entities and activities

The general PBSE approach to enhanced MBSE is being shared through and explored by the members of the INCOSE MBSE Patterns Working Group, of the INCOSE-OMG MBSE Initiative, along with the S*Patterns Community. Such cross-industries groups continue to pursue a number of PBSE applications and projects (including joint projects with other INCOSE working groups) which are shared through the INCOSE MBSE Patterns Working Group's MBSE wiki / web site posted resources, reference, and information assets. Refer to the References below.

PBSE and its supporting S*Metamodel are tool-independent by intention. Substantially any COTS modeling, engineering, simulation, or PLM tool can be made to support PBSE, by the use of an S*Metamodel Map for the specific COTS tool. Such mappings have already been created for a number of third party COTS tools and information systems, both earlier and newer generation.

2.3 Agile Systems Engineering Life Cycle Management (ASELCM) S*Pattern

2.3.1 Summary statement of general nature, intended use, value, current state, and trends

(Adapted from: "Introduction to the Agile Systems Engineering Life Cycle MBSE Pattern", by Bill Schindel and Rick Dove, INCOSE IS2016.)

Engineered and other systems are under pressure to adapt, from opportunities or competition, predators, changing environment, and physical or cyberattack. Ability to adapt well enough as conditions change, especially in presence of uncertainty, is valued. Systems (including developmental and life cycle management) that adapt well enough, in time, cost, and effectiveness, are sometimes called "agile". As environmental change or uncertainty increase, agility can mean survival. Agile systems and agile systems engineering (ASE) are subjects of an INCOSE discovery project, described in the References. This section summarizes the underlying MBSE-based Agile Systems Engineering Life Cycle Pattern being used to capture, analyze, and communicate key aspects of systems being studied. More than an ontology, this model helps us understand necessary and sufficient conditions for agility, different approaches to it, and underlying relationships, performance couplings, and principles. This section introduces the framework, while specific findings about methods and practicing enterprises studied are be reported in other References.

ASE emphasis on learning. ASE emphasizes learning in the presence of uncertainty and change. Some may be human learning by individuals and teams, but we know that the accumulation of experience in an organization can take a number of forms, symbolized in **Figure 12**. Accordingly, part of the ASELCM project is examining how "information debt" can be managed to support sustained agility on a balanced basis, and this includes our investigation of MBSE-based Patterns in a role similar to their history in science—as repositories of accumulated (improving) knowledge about systems of interest. To the extent that we find that MBSE has a place within the practice of ASE, we know that MBSE processes can be made more agile by their retaining and efficiently re-using what we know as MBSE Patterns, thereby limiting MBSE overhead to new learning. Composable knowledge, like composable architecture, improves agility. Through learning, agile systems may accumulate formidable amounts of information, such as complex situation-based configuration rules, appearing intelligent, but in fact riding on a static accumulated base of learned knowledge:

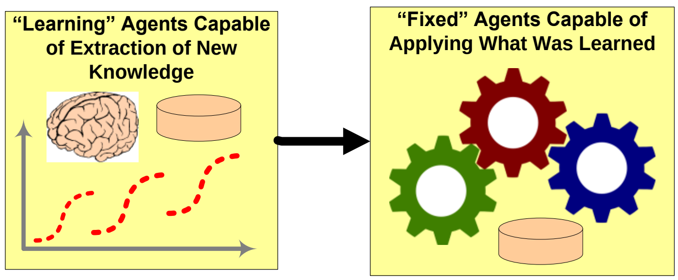


Figure 12: Experience Accumulates in Different Forms, for Later Uses

The ASELCM Domain Model: Key System Boundaries. The ASELCM Pattern establishes a set of system reference boundaries. Whether the systems of interest are small or large, human or inanimate, flying through the air or performing business processes, all these start with the S*Model definition of <u>System</u>, depicted in **Figure 13**:

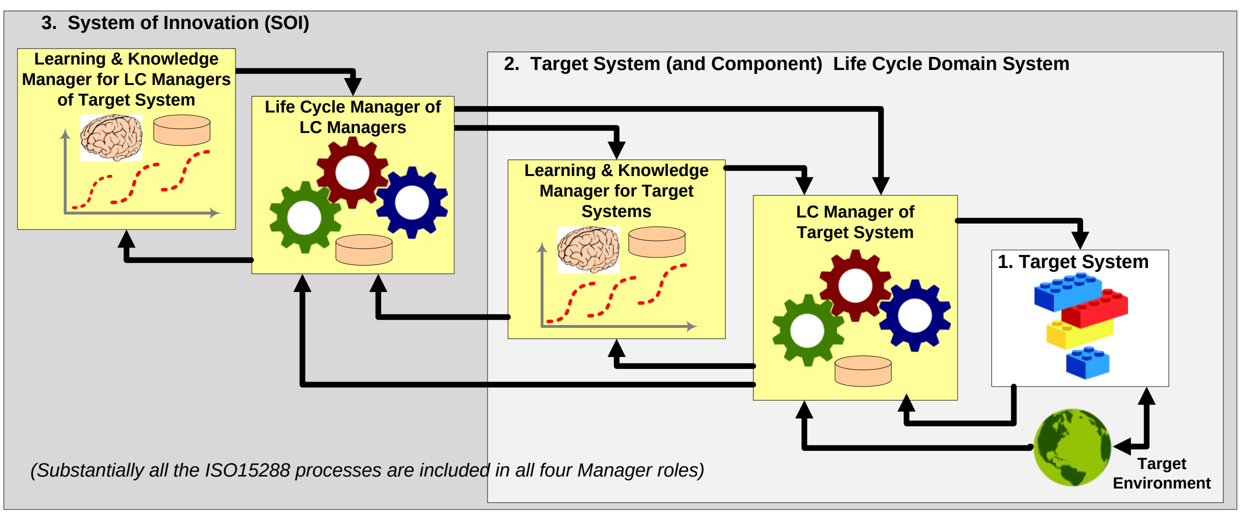


Figure 13: ASELCM Reference Boundaries--Systems 1, 2, and 3

This ASELCM Pattern particularly refers to <u>three major system reference boundaries</u>, and within those, six subsystem reference boundaries. These are all logical boundaries (defined by the behavior, not the identity, of systems), and are depicted by the iconic diagram of Figure 13:

System 1: The **<u>Target System (and Components)</u>:** (Refer to Figure 13.) The logical system of interest, which results from, or is subject to, innovation:

- Its behavior, characteristics, or performance are targets of the life cycle innovation (change, adaptation) process we'll introduce later.
- It is <u>potentially</u> agile. Assertion: for Systems Engineering (in System 2) to be fully agile, so must its target system (in System 1) also be agile—or else a competing SE system with an agile target system will out-perform it.
- Examples include aircraft, automobiles, telephones, satellites, the human immune system, software, restaurants, birds, and the health care delivery system.

System 2: The **<u>Target System (and Component) Life Cycle Domain System</u>:** (Refer to Figure 13) The logical system within which the Target System will exist during its life cycle, when "in service" or otherwise. This domain includes <u>*all actors with which the Target System will directly interact any time during its life cycle*</u>: This includes (among others) any system that directly manages the life cycle of an instance of a Target System (or a Component)—development, production and integration systems, maintenance and operations systems, and others. The System 2 model (Figure 13) recognizes three sub-systems besides the Target System:

<u>Target System Life Cycle Domain Actors</u>: All actors with which the Target System will directly interact during its life cycle—those in its operational domain (demanding agility) as well as all other direct actors. The next sub-system is a special case of those actors.

<u>LC Manager of Target System</u>: Manages all life cycle aspects of the Target System, as recognized by ISO 15288. Note that this is more than just development or systems engineering—it includes manufacturing or acquisition, operations, maintenance, security, configuration management, and all the ISO 10040 System Management Functional Areas (SMFAs). However, it manages only "already known" aspects of System 1 and its Target Environment—it does not include responsibility of learning new things about them, which is allocated elsewhere.

<u>Learning & Knowledge Manager for Target System (and Components)</u>: Responsible for learning new things about the Target System, its Components, and its Environment. This may include extraction of patterns or other knowledge from observations, planning experiments and extracting conclusions from their results, and other forms of learning. It also includes responsibility for accumulation and persistent memory of those learnings, and for providing the resulting knowledge for use by the LC Managers of the Target System.

Remember that these are logical (behavioral) roles. In realized physical systems, a single physical system may behave as both a Target System and a system that produces, modifies, reconfigures, or otherwise manages a Target System, by having roles from each allocated to it. For purposes of this logical roles description, they have been identified separately.

System 3: The **<u>System of Innovation</u>:** (Figure 13.) The logical system that includes System 1 and System 2, and that is additionally responsible for managing the life cycles of instances of any (System 2) Target System LC Manager. Recall that those System 2 Target System LC Managers include Target System development, production, integration, maintenance, operations, and other management systems.

Why are the learning capabilities of System 2 and System 3 differentiated from other capabilities in System 2 & 3 models? Especially for understanding two aspects of Agility:

- We want to explicitly understand what capabilities (Figure 12, right side) can exist for "agile movement within what is already known", for both System 2 and System 3, in nearly all of the ISO 15288 process areas, and . . .

- We also want to explicitly understand what is meant by “learning” (Figure 12, left side) in nearly all of the ISO 15288 process areas.

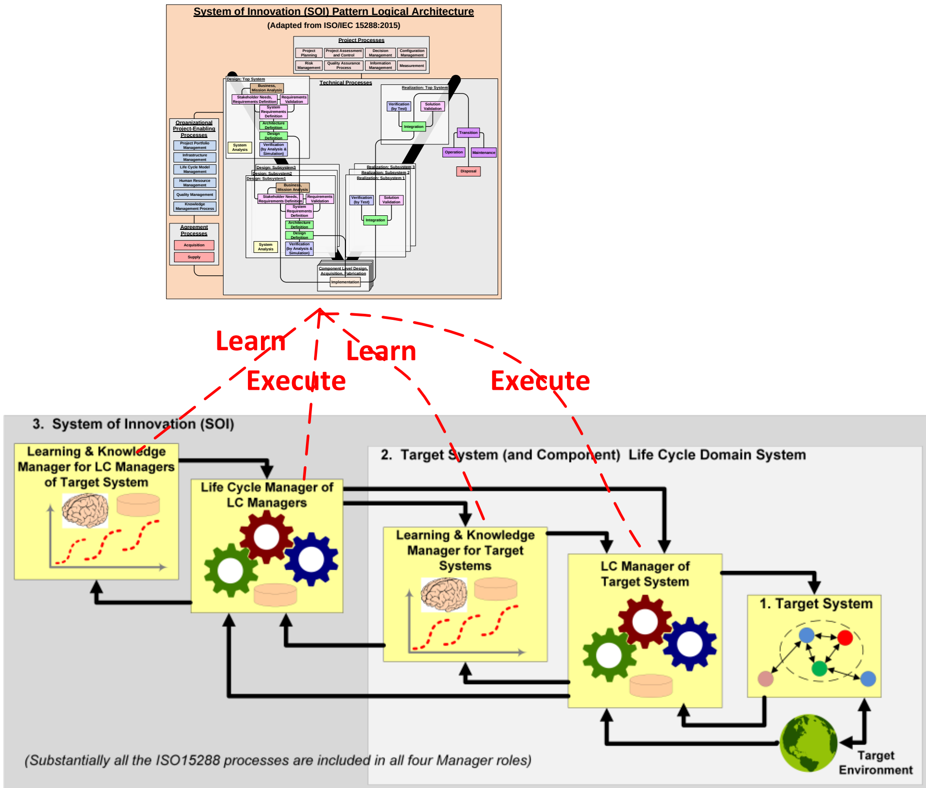


Figure 14: Four Instances of ISO15288 LC Management Processes Appear Across the ASELCM Pattern

How ISO 15288 LC Management Processes Participate in the ASELCM Pattern. There are four subsystems of ASELCM Systems 2 and 3 that deal in the life cycle management of other systems. The (system life cycle management) processes of ISO 15288 are therefore within the scope of those systems. Likewise, the processes of ASE, even if believed distinct from the ISO 15288 processes, are also within the scope of those ASELCM Pattern systems. The (agile or not) equivalents of the ISO15288 Technical Processes (see Figure 3) appear four times in the ASELCM Pattern, as shown in Figure 14.

The ISO 15288 and ASE models are alternate views of the same underlying reality, with ISO15288 traditionally emphasizing process management, and ASE processes typically emphasizing effective discovery, learning, and response in the presence of change and uncertainty, through rapid iteration. The configurability of the underlying pattern is driven by its Stakeholder Feature level optimization for different situations and different stakeholder objectives.

Additional ASELCM Pattern Content. The ASELCM Pattern as a whole contains more than this space limited summary describes— the References sample the additional scope of this S*Pattern:

- Stakeholder Features, Feature Attributes: System fitness (trade) space for S1, S2, S3
- Functional Interactions, Roles, Role Attributes: What actually happens
- States / Modes: Temporal aspects
- Attribute Couplings: Representing quantitative relationships, impacts, principles
- Configuration, Reconfiguration, and Adaptation

2.3.2 Related INCOSE and other entities and their activities

INCOSE and other entities, and their related activities, concerned with the Agile Systems Engineering Life Cycle Management (ASELCM) Pattern include the following. Related references are listed in the References Section 3 below.

1. The INCOSE Agile Systems & Systems Engineering Working Group: This INCOSE working group is concerned with agility of subject systems as well as agility of the engineering processes that accompany their life cycle. It is the lead INCOSE working group for the INCOSE Agile Systems Engineering Life Cycle Model Discovery Project.
   http://www.incose.org/ChaptersGroups/WorkingGroups/Transformational/agile-systems-se
2. The INCOSE MBSE Patterns Working Group: This working group of INCOSE is also part of the INCOSE/OMG MBSE Initiative, concerned with use of MBSE Patterns, referred to as Pattern-Based Systems Engineering (PBSE)—see Section 2.2. It is a participant in the INCOSE Agile Systems Engineering Life Cycle Model Discovery Project, with the responsibility of expression of what is learned by that project in the form of a formal MBSE Model: The ASELCM S*Pattern.
   http://www.incose.org/ChaptersGroups/WorkingGroups/Transformational/mbse-patterns
3. The INCOSE Health Care Working Group: This INCOSE working group is concerned with systems engineering in the health care domain. As a part of its overall activity, it conducts the annual INCOSE Agile Health Care Systems Conference, begun in 2015, specific to agility in that domain.
   http://www.incose.org/ChaptersGroups/WorkingGroups/Application/healthcare
4. The INCOSE Critical Infrastructure Protection and Recovery (CIPR) Working Group: This working group of INCOSE is concerned with systems engineering approaches to critical infrastructure, such as energy and communication utilities, transportation systems, water and food distribution, medical, and other infrastructure systems viewed from the standpoint of protecting them from disruption or recovering them when disrupted. This group participates in the annual INCOSE/IEEE/NASA Energy Tech Conference.
   http://www.incose.org/ChaptersGroups/WorkingGroups/Application/critical-infrastructure
5. The INCOSE Power and Energy Systems Working Group: This INCOSE working group is concerned with systems engineering of power and energy systems. This group produces, along with partners NASA and IEEE, the annual INCOSE/IEEE/NASA Energy Tech Conference.
   http://www.incose.org/ChaptersGroups/WorkingGroups/Application/power-energy-systems
6. The ASME Standards Committees on Verification and Validation in Computational Modeling and Simulation are concerned with standards as to the verification and validation of computational models. This is a formalization of the representation of group learning, and as such the V&V of these models is central to organized agility and the ASELCM Pattern. This organization conducts the annual ASME Symposium on Verification and Validation of Computational Models.
   https://cstools.asme.org/csconnect/CommitteePages.cfm?Committee=100003367

2.4 Applications to Critical Infrastructure Protection and Recovery (CIPR) S*Pattern

This is covered separately in the "Introduction to the Critical Infrastructure Protection and Recovery (CIPR) S*Pattern: Application to the Electrical Power Grid Pattern" reference document.

3 References:

3.1 MBSE References

General Systems Engineering References:

1. (Walden et al 2015) Walden, et al, eds., *Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities*, Fourth Edition, INCOSE TP-2003-002-04, International Council on Systems Engineering, San Diego, CA, 2015.
2. (ISO 2015) ISO/IEC 15288: 2015. "Systems engineering – System life cycle processes." Geneva: International Organization for Standardization.

MBSE Methodologies, Modeling Languages, and Related Standards:

3. (Estafan 2008) Estafan, J., INCOSE Survey of MBSE Methodologies, web site within INCOSE/OMG MBSE wiki: http://www.omgwiki.org/MBSE/doku.php?id=mbse:methodology
4. (MBSE wiki 2017) MBSE wiki, INCOSE/OMG MBSE Initiative: http://www.omgwiki.org/MBSE/doku.php
5. (Friedenthal et al 2015) Sanford Friedenthal, Alan Moore and Rick Steiner, *A Practical Guide to SysML, The Systems Modeling Language*,(Third Edition), MK/OMG Press, Elsevier, ISBN: 978-0-12-800202-5, 2015.
6. OMG web site for SysML: http://www.omgsysml.org/
7. OMG Formal Specifications Of SysML: retrieve from http://www.omg.org/spec/SysML/
8. OMG Formal Versions Of UPDM: retrieve from http://www.omg.org/spec/UPDM/
9. OMG documentation of Unified Architecture Framework (UAF) Profile, Version 1.0 - FTF Beta1, Release Date: August 2016, retrieve from http://www.omg.org/spec/UAF/1.0/Beta1/
10. (Hause et al 2016) Matthew Hause, Aurelijus Morkevicius, Graham Bleakley, "Transitioning UPDM to the UAF", Object Management Group, Reston, VA. March 23, 2016, retrieve from http://www.omg.org/news/meetings/tc/va-15/special-events/uaf-pdf/2_MH_GB_AM_Transitioning_UPDM_UAF_FD_MH.pdf
11. (Dori 2002) Dori, Dov (2002). *Object-Process Methodology – A Holistic Systems Paradigm*. Berlin, Heidelberg, New York: Springer Verlag. ISBN 3-540-65471-2.
12. Draft Federal Information Processing Standards Publication 183: "Integration Definition for Function Modeling (IDEF0)", December 21, 1993. (Note that this FIPS standard has been withdrawn by NIST.)
13. (Roberts et al 1983) Roberts, N., D. F. Andersen, R. M. Deal, M. S. Grant, & W. A. Shaffer. 1983. *Introduction to Computer Simulation: A System Dynamics Modeling Approach*. Reading MA: Addison-Wesley.
14. Functional Mock Up Interface (FMI) draft specification: https://en.wikipedia.org/wiki/Functional_Mock-up_Interface
15. Functional Mock Up Interface (FMI) for Model Exchange and Co-Simulation retrieve from: https://www.fmi-standard.org/downloads

MBSE Tooling and Related Standards:

16. ISO 10303-1:1994 "Industrial automation systems and integration -- Product data representation and exchange -- Part 1: Overview and fundamental principles", International Standards Organization, 1994.
17. ISO 10303 "Standard for the Exchange of Product model data (STEP - ISO 10303) Application Protocol 233 Systems engineering data representation", International Standards Organization.

18. ISO 10303 Overview, retrieve from: https://en.wikipedia.org/wiki/ISO_10303
19. OMG XML Metadata Interchange (XMI) Specification(Also Known As MOF 2 XMI Mapping: Retrieve from: http://www.omg.org/spec/XMI/
20. Open Services for Life Cycle Collaboration https://open-services.net/
21. OSLC Specification, retrieve from: http://open-services.net/wiki/core/Specification-3.0/OSLCCoreSpecificationsV3

MBSE Vision, Status, Plans, Road Ahead:

22. (Beihoff et al 2014) Beihoff et al, *A World In Motion: Systems Engineering Vision 2025*, International Council on Systems Engineering, San Diego, CA, June, 2014.
23. (Peterson et al 2017) Peterson, T., Schindel, W., Fuchs, J., "Model-Based Transformation: Planning and Assessment Instrument", INCOSE Corporate Advisory Board, , CA, Jan 2017.
24. (Hause 2016) Hause, M., "What is Missing in MBSE?", IEEE/INCOSE Energy Tech 2016, Cleveland, OH, Nov., 2016.
25. (Schindel 2016) Schindel, W., "Energy Tech 2016: Track 1 (MBSE) Data Collection—Using Model-Based Facilitation", IEEE-INCOSE Energy Tech, Cleveland, OH, Nov, 2016.
26. (Friedenthal 2017) Friedenthal, S., "Future Directions for SysML V2, INCOSE IW MBSE Workshop, INCOSE IW2017 MBSE Workshop, Torrance, CA, Jan, 2017.
27. (Peak 2017) Peak, R., "MBSE/SysML Education", INCOSE IW2017 MBSE Workshop, Torrance, CA, Jan, 2017.
28. (Peterson 2017) Peterson, T., "Transformation of Systems Engineering Into a Model-Based Discipline" (MBSE Workshop Closing Outbrief) INCOSE IW2017 MBSE Workshop, Torrance, CA, Jan, 2017.
29. (Walsh 2017) Walsh, J., "ASSESS–INCOSE: Addressing the Changing Role of Engineering Simulation", INCOSE IW2017 MBSE Workshop, Torrance, CA, Jan, 2017.
30. (Hightower 2017) Hightower, J., "Establishing Model Credibility Using Verification and Validation" (ASME Model V&V), INCOSE IW2017 MBSE Workshop, Torrance, CA, Jan, 2017.
31. (Li 2017) Li, C., et al, *Proc. of NASA JPL 2017 MBSE Symposium,* Pasadena, CA, Jan, 2017, retrieve from: https://drive.google.com/drive/folders/0B1_Bo9OVbkndUk1WVWRhWDhSNFk

3.2 PBSE References

The contextual significance of the following resources and references is best understood from their citation appearances in the material above. Some of the following may be downloaded from the (INCOSE PBSE Challenge Team, 2014) reference link provided below.

1. (Alexander et al, 1977) Alexander, C., Ishikawa, S., Silverstein, M., Jacobson, M., Fiksdahl-King, I., and Angel, S. *A Pattern Language*. Oxford University Press, New York, 1977.
2. (Bradley et al, 2010) J. Bradley, M. Hughes, and W. Schindel, "Optimizing Delivery of Global Pharmaceutical Packaging Solutions, Using Systems Engineering Patterns" Proceedings of the INCOSE 2010 International Symposium, 2010.
3. (Buckingham 1914) Buckingham, E., On Physically Similar Systems: Illustrations of the Use of Dimensional Analysis", *Phys. Rev*. 4(4): 345, Bibcode: 1914.
4. (Cloutier, 2008) Cloutier, R., *Applicability of Patterns to Architecting Complex Systems: Making Implicit Knowledge Explicit*. VDM Verlag Dr. Müller. 2008.

5. (Cook, Schindel, 2015) Cook, D., and Schindel, W., "Utilizing MBSE Patterns to Accelerate System Verification", to appear in Proc. of the INCOSE 2015 International Symposium, Seattle, WA, July, 2015.
6. (Estefan, 2008) Estefan, J. 2008. "Survey of Candidate Model-Based Systems Engineering (MBSE) Methodologies", rev. B. Seattle, WA, USA: International Council on Systems Engineering (INCOSE). INCOSE-TD-2007-003-02. Accessed April 13, 2015 at http://www.omgsysml.org/MBSE_Methodology_Survey_RevB.pdf
7. (Gamma et al, 1995) Gamma, E., Helm, R., Johnson, R., Vlissides, J. *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley Publishing Company, Reading, MA, 1995.
8. (Genesereth and Nilsson, 1987) Genesereth, M. R., & Nilsson, N., *Logical Foundations of Artificial Intelligence*, San Mateo, CA: Morgan Kaufmann Publishers, 1987.
9. (Gingerich 2004) Gingerich, O., *The Book Nobody Read: Chasing the Revolutions of Nicolaus Copernicus*, New York: Walker & Co. Publishing, 2004.
10. (Haskins, 2008) Haskins, C., "Using patterns to transition systems engineering from a technological to social context", *Systems Engineering*, Volume 11, Issue 2, pages 147–155, Summer 2008.
11. (INCOSE PBSE Challenge Team, 2014) MBSE wiki / web site of Patterns Challenge Team: http://www.omgwiki.org/MBSE/doku.php?id=mbse:patterns:patterns
12. (ISO10303-233 2012) ISO 10303-233:2012 "Industrial automation systems and integration -- Product data representation and exchange -- Part 233: Application protocol: Systems engineering", 2012.
13. (ISO 42010, 2011) ISO/IEC/IEEE 42010:2011 –"Systems and software engineering - Architecture description" 2011.
14. (ISO 26550, 2013) ISO/IEC 26550:2013, "Software and systems engineering -- Reference model for product line engineering and management", 2013.
15. (Levi, 2011) Levi, M., *Classical Mechanics with Calculus of Variations and Optimal Control*, American Mathematical Society, Providence, Rhode Island, 2014.
16. (Meyer and Lehnerd, 2011), Meyer, M., Lehnerd, A., *The Power of Product Platforms*, Free Press, 2011.
17. (Peterson and Schindel, 2014) Peterson, T., and Schindel, W., "Pattern-Based Systems Engineering: Leveraging Model-Based Systems Engineering for Cyber-Physical Systems", *Proc. of NDIA GVSETS Conference*, 2014.
18. (Rayleigh 1915) Lord Rayleigh, "The Principle of Similitude", *Nature*, 95(2368): 66-8, Bibcode: 1915Natur..95…66R.
19. (Schindel and Smith, 2002) Schindel, W., and Smith, V., "Results of applying a families-of-systems approach to systems engineering of product line families", SAE International, Technical Report 2002-01-3086, 2002.
20. (Schindel, 2005a) W. Schindel, "Pattern-Based Systems Engineering: An Extension of Model-Based SE", INCOSE IS2005 Tutorial TIES 4, 2005.
21. (Schindel, 2005b) W. Schindel, "Requirements statements are transfer functions: An insight from model-based systems engineering", *Proceedings of INCOSE 2005 International Symposium*, 2005.

22. (Schindel, 2006), “Feelings and Physics: Emotional, Psychological, and Other Soft Human Requirements, by Model-Based Systems Engineering”, *Proc. of INCOSE International Symposium*, 2006.
23. (Schindel, 2010), Schindel, W., “Failure Analysis: Insights from Model-Based Systems Engineering”, *Proc. of INCOSE International Symposium*, 2010.
24. (Schindel, 2011), Schindel, W., “What Is the Smallest Model of a System?”, *Proc. of the INCOSE 2011 International Symposium*, International Council on Systems Engineering (2011).
25. (Schindel, 2012) Schindel, W., “Integrating Materials, Process & Product Portfolios: Lessons from Pattern-Based Systems Engineering”, *Proc. of Society for the Advancement of Material and Process Engineering*, 2012.
26. (Schindel, Peterson, 2013) Bill Schindel, Troy Peterson, “Introduction to Pattern-Based Systems Engineering (PBSE): Leveraging MBSE Techniques”, in *Proc. of INCOSE 2013 International Symposium*, Tutorial, June, 2013.
27. (Schindel, 2013a), W. Schindel, “Systems of Innovation II: The Emergence of Purpose”, *Proceedings of INCOSE 2013 International Symposium*, 2013.
28. (Schindel, 2013b) “System Interactions: Making The Heart of Systems More Visible”, Proc. of INCOSE Great Lakes Regional Conference, 2013.
29. (Schindel, 2014) Schindel, W. “The Difference Between Whole-System Patterns and Component Patterns: Managing Platforms and Domain Systems Using PBSE”, INCOSE Great Lakes Regional Conference on Systems Engineering, Schaumburg, IL, October, 2014
30. (Schindel, Lewis, Sherey, Sanyal, 2015) Schindel, W., Lewis, S., Sherey, J., Sanyal, S., “Accelerating MBSE Impacts Across the Enterprise: Model-Based S*Patterns”, to appear in Proc. of INCOSE 2015 International Symposium, July, 2015.
31. (Sussman and Wisdom 2001) Sussman, G, and Wisdom, J., *Structure and Interpretation of Classical Mechanics,* Cambridge, MA: MIT Press, 2001.

3.3 ASELCM References

1. (Dove, Schindel, and Kenney 2017) Dove, R, W. Schindel, M. Kenney. 2017. Case study: agile SE process for centralized SoS sustainment at Northrop Grumman. *Proceedings International Symposium. International Council on Systems Engineering*. Adelaide, Australia, 17-20 July.
2. (Dove and Schindel 2017) Dove, R, W. Schindel. 2017. Case study: transition to Scaled-Agile systems engineering at Lockheed Martin’s Integrated Fighter Group. Unpublished working paper. www.parshift.com/s/ASELCM-04LMC.pdf
3. (Dove, Schindel, and Hartney 2017) Dove, R., Schindel, W., and Hartney, R., “Case Study: Agile Hardware/Firmware/Software Product Line Engineering at Rockwell Collins”, in *Proc. of 11th Annual IEEE International Systems Conference*, Montréal, Québec, Canada, April 24-27, 2017.
4. (Schindel 2017) Schindel, W., “Innovation, Risk, Agility, and Learning, Viewed as Optimal Control & Estimation”, *Proc. of INCOSE 2017 International Symposium*, Adelaide, Australia.
5. (Schindel and Dove 2016) Schindel, W., and Dove, R., 2016, ‘Introduction to the Agile Systems Engineering Life Cycle MBSE Pattern’, in *Proc. of INCOSE 2016 International Symposium*, Edinburgh, UK.
6. (Beihoff and Schindel 2012) Beihoff, B., and Schindel, W., “Systems of Innovation I: Models of Their Health and Pathologies”, *Proc. of INCOSE International Symposium*, 2012.
7. (Dove and LaBarge 2014) Dove, R, and LaBarge, R., “Fundamentals of Agile Systems Engineering—Part 1” and “Part 2”, in *Proc. of INCOSE International Symposium*, July, 2014.

8. (Dove 2015) Dove, R., INCOSE Agile Systems Engineering Life Cycle Model Discovery Project, project web site, loaded 2015 from: http://www.parshift.com/ASELCM/Home.html
9. (Dove, Scrapper, and Schindel 2016) Dove, R., Scrapper, C., and Schindel, W., "Agile Systems Engineering Process Features Collective Culture, Consciousness, and Conscience at SSC Pacific Unmanned Systems Group", in *Proceedings of INCOSE 2016 International Symposium,* 2016.
10. (Fowler and Highsmith 2001) Fowler, M., and Highsmith, J., "The Agile Manifesto". Dr. Dobb's Journal, August, 2001. www.drdobbs.com/open-source/the-agile-manifesto/184414755 .
11. (Haberfellner and de Weck 2005) Haberfellner, Reinhard and Olivier de Weck. 2005. "Agile SYSTEMS ENGINEERING versus AGILE SYSTEMS engineering" in *Proc. of INCOSE International Symposium*, Rochester, NY, 10-15 July. http://strategic.mit.edu/docs/3_59_INCOSE-2005-AGSEvsEAGS.pdf.
12. (Highsmith 2002) Highsmith, J., *Agile Software Development Ecosystems*. Addison-Wesley Professional, 2002.
13. (INCOSE Patterns Working Group 2015) INCOSE/OMG Patterns Working Group 2015, http://www.omgwiki.org/MBSE/doku.php?id=mbse:patterns:patterns
14. (Leffingwell 2011) Leffingwell, Dean, Agile *Software Requirements: Lean Requirements Practices for Teams, Programs, and the Enterprise*. Addison-Wesley 2011
15. (Schindel 2013) Schindel, W., "Systems of Innovation II: The Emergence of Purpose", in *Proceedings of INCOSE 2013 International Symposium* 2013

3.4 CIPR Reference

1. (Schindel 2017) "Introduction to the Critical Infrastructure Protection and Recovery (CIPR) S*Pattern: Application to the Electrical Power Grid Pattern", INCOSE CIPR Working Group, 2017.