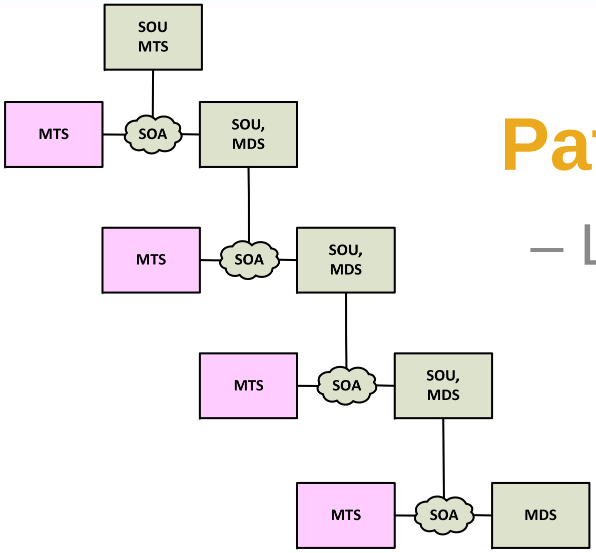
V1.2.4

Bill Schindel, ICTT System Sciences
schindel@ictt.com

Troy Peterson, Booz Allen Hamilton
peterson_troy@bah.com



Pattern-Based Systems Engineering

– Leveraging Model-Based Systems Engineering for Cyber-Physical Systems

GVSETS

Abstract

As a network of interacting elements, cyber-physical systems (CPS) provide tremendous opportunities to advance system adaptability, flexibility and autonomy. However, they also present extremely complex and unique safety, security and reliability risks. The Department of Defense is seeking methods to deliver and support trusted systems and manage risks associated with mission-critical functionality. Technical thought leaders have discussed the need to address 10:1 more complex systems with 10:1 reduction in effort, using people from a 10:1 larger community than the "systems expert" group.

This presentation briefly summarizes the approach of Pattern-Based Systems Engineering (PBSE), which leverages the power of Model-Based Systems Engineering (MBSE) to rapidly deliver these benefits to the larger systems community. This order-of-magnitude improvement is especially necessary to address the rapidly increasing complexity of today's and future cyber-physical systems.

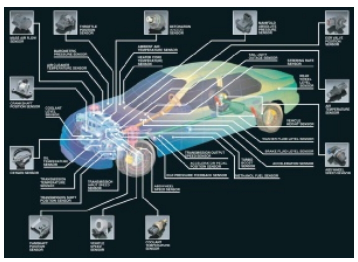
While applying PBSE expresses many patterns, this paper introduces the Embedded Intelligence (EI) Pattern, particularly relevant to cyber-physical systems such as autonomous ground vehicles.

Contents

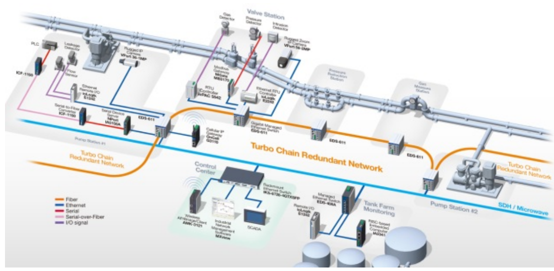
Introduction

- NSF on Cyber-Physical Systems (CPS):

"Engineered systems that are built from, and depend upon, the seamless integration of computational algorithms and physical components"[1]







- Intertwine computational elements with physical entities, within aerospace, automotive, energy, healthcare, manufacturing, other sectors.

- New opportunities from Cyber-Physical Systems:
  - Enhanced system value: Functionality, performance, productivity
  - Higher sophistication of behavior, adaptation
  - Emergent combinations at S-o-S level, not all expected
  - Improved life cycle management
  - Opportunities for integrators, higher level networks
  - Technology renewal in previously mature sectors
  - Higher rates of change, increased upgrade opportunity

Cyber-Physical Systems

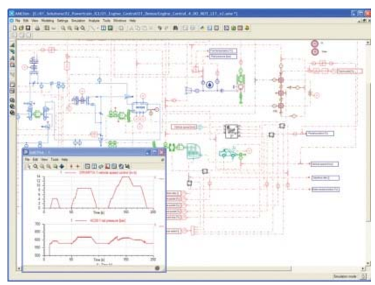
- New challenges, risks from Cyber-Physical Systems:

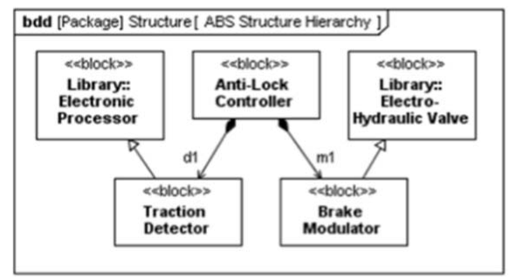
  - Emergent system risks, unpredicted consequences
  - Opens unsecured new lines of attack to opponents
  - Increased complexity, reduced confidence
  - New and more complex failure modes, multi-level consequences
  - Development in general, and verification in particular

Model-Based Systems Engineering (MBSE)

- INCOSE on Model-Based Systems Engineering (MBSE):
  – "the formalized application of modeling to support system requirements, design, analysis, verification and validation activities, beginning in the conceptual phase and continuing throughout the later life cycle phases …"[2]

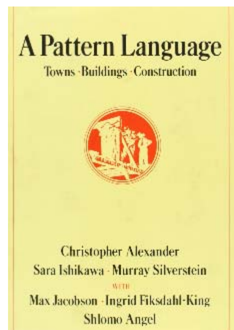
INCOSE
International Council on Systems Engineering

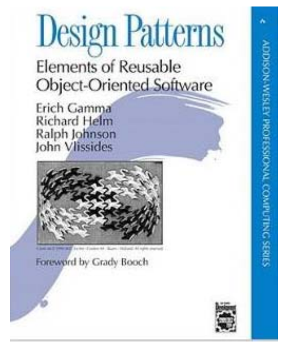


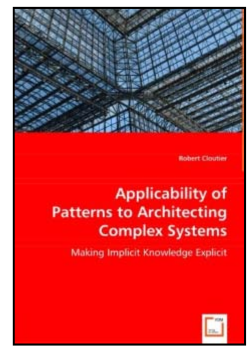


- MBSE is a natural partner for CPS engineering & life cycle management:
  – Explicit descriptions become more important as complexity and risks grow.
  – MBSE emerges in part out of information systems, which are a key <u>subset</u> of CPS--the scope of the model is enlarged and systems-level.
  – The subset of a CPS which is IT-based can potentially have its software generated directly from models, enhancing consistency.

- The term “pattern” appears repeatedly in the history of design, such as civil architecture[6], software design[7], and systems engineering[8]:

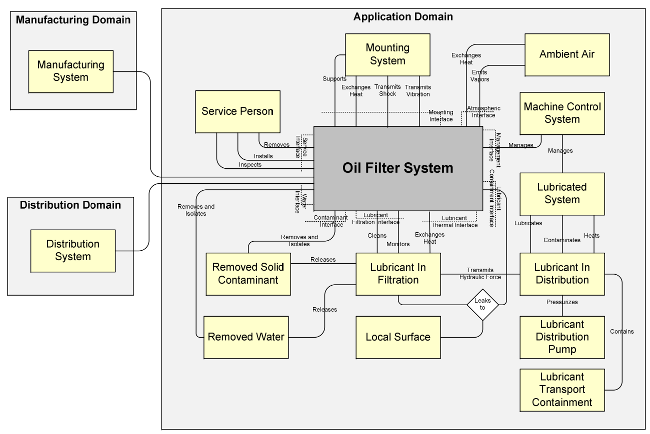






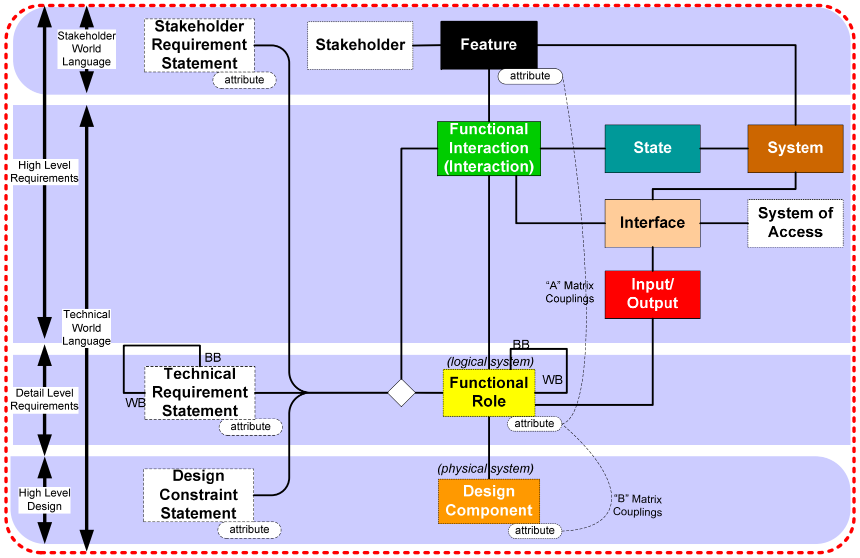
- Those “patterns” represent regularities that repeat, modulo some variable aspects, across different instances in space and time.
- However, when we refer to “PBSE” in this presentation, we will mean the use of S*Patterns . . . .

- S*Patterns are Model-Based:
  - We are referring to patterns represented by formal system models.



  - Many of the historical "design patterns" were not based on formal system models.
  - S*Patterns are not dependent on any single system modeling language, and are readily expressed in SysML, IDEF, or other formal modeling languages.

- S*Patterns are Model-Based:
  - Independent of the specific modeling language, S*Models always conform to the underlying S*Metamodel:



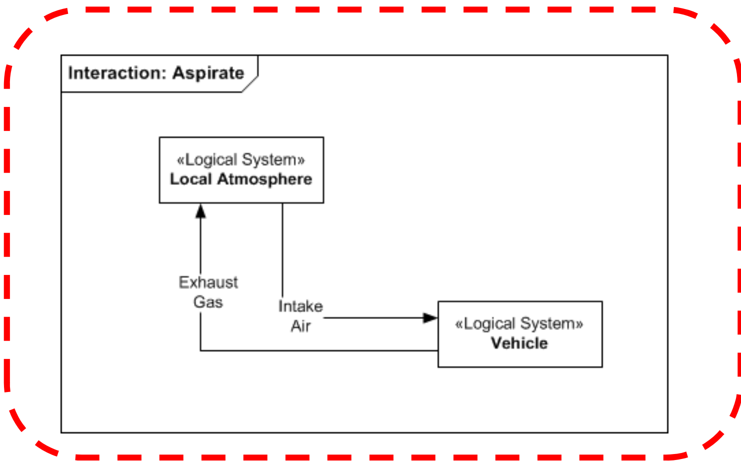
Summary view of S*Metamodel

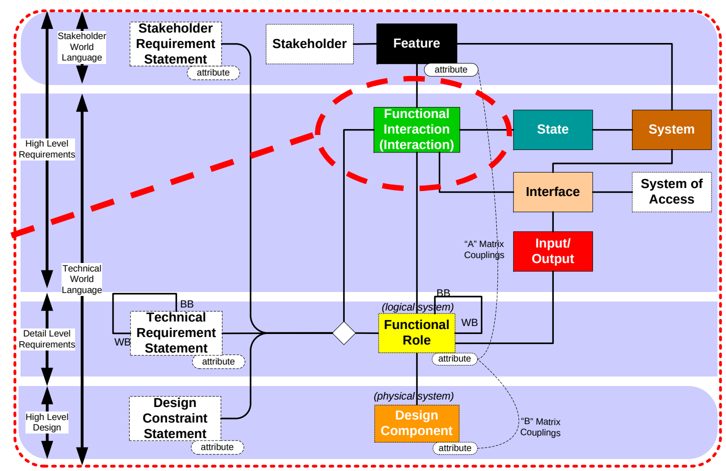
  - The S*Metamodel is the smallest model sufficient to the purposes of engineering and science.[19]

Pattern-Based Systems Engineering (PBSE)

SYSTEMS ENGINEERING AND INTEGRATION

- The S*Metamodel explicates Physical Interactions:
  - Interactions: state-impacting exchange of energy, force, mass, or information:

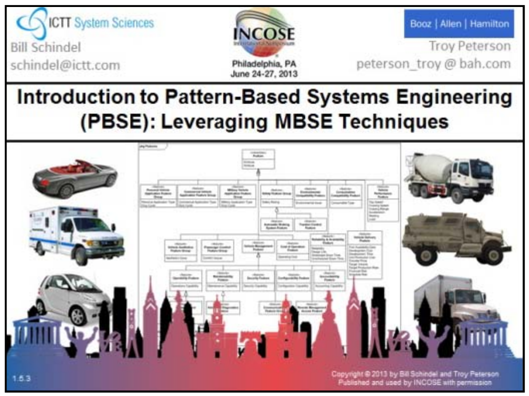




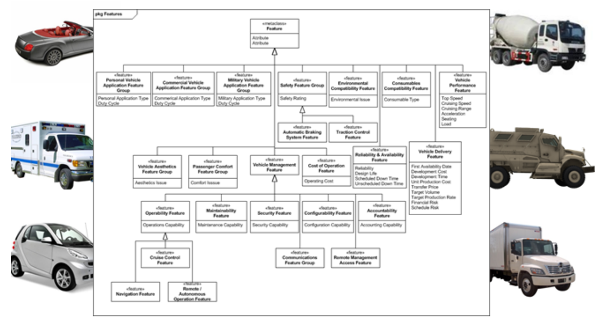
Summary view of S*Metamodel

  - Such interactions are the basis of substantially all the laws (patterns, regularities) of the physical sciences.
  - Systems Engineering should have as strong a foundation as the other engineering disciplines.

- The <u>scope</u> of S*Patterns are "<u>Whole Systems</u>":
  - An S*Pattern is effectively a formal model of a <u>platform</u> system, or a whole system domain:



A vehicle pattern in SysML

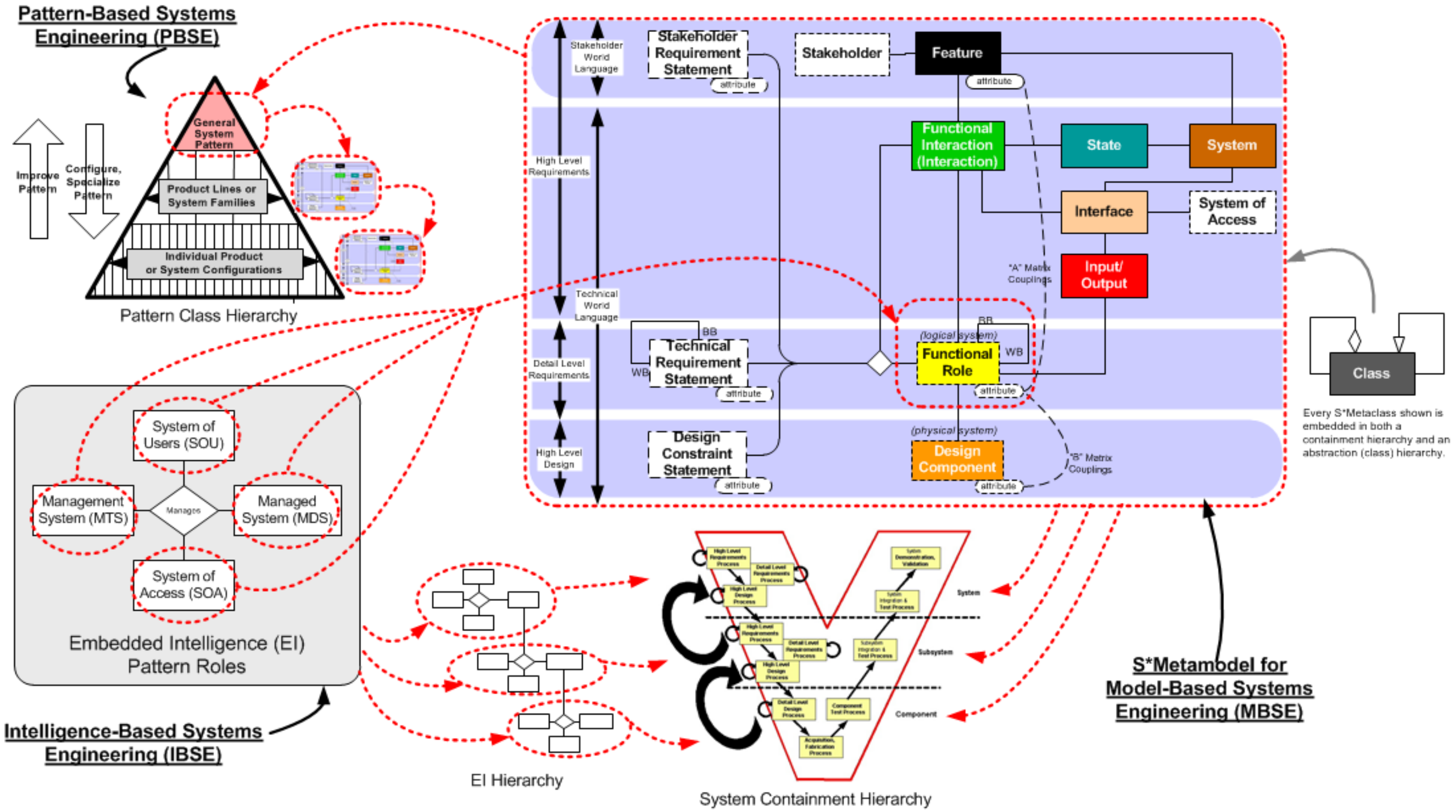


  - Historical "design patterns" were most frequently about smaller repeating component or subsystem patterns, used as deemed applicable.
- The <u>scope</u> of S*Patterns includes system requirements, designs, and other S*Model information such as verification, failure analysis, etc. [9, 10, 11, 12, 13, 14, 15]

Pattern-Based Systems Engineering (PBSE)

SYSTEMS ENGINEERING AND INTEGRATION

- S*Patterns are formally configurable, using automated algorithms, portable across numerous third-party COTS engineering tools and databases, to rapidly generate many specific system requirement/design configurations (including failure mode analyses) from desired platform features:



Embedded Intelligence (EI) Pattern

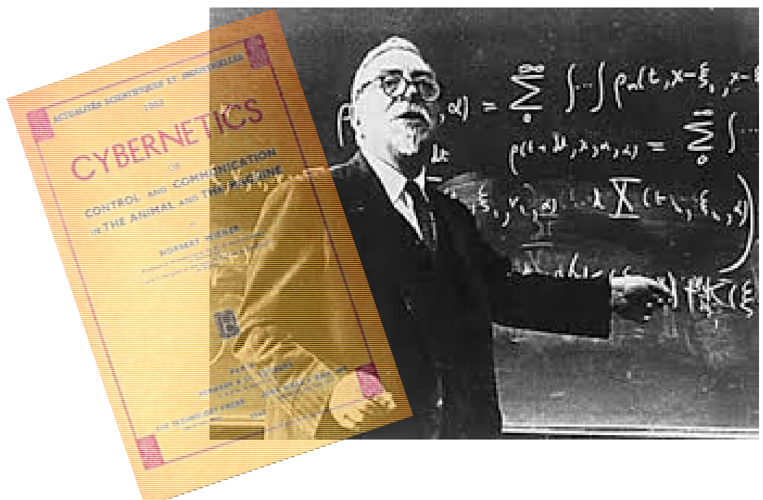
SYSTEMS ENGINEERING AND INTEGRATION

- We have applied S*PBSE over several decades, across a range of domains:
  - Carrier grade telecommunications, engines and power systems, automotive and off road heavy equipment, mil/aero, medical devices, pharmaceutical manufacturing, consumer products, and advanced manufacturing systems.[16] [17] [18]
- Many different S*Patterns have emerged from this, but:
  - We will focus here on the Embedded Intelligence (EI) Pattern.
  - Fundamental to Cyber-Physical Systems (CPS) at all levels.
  - Extends the typical use of the term "embedded" to refer to only lower-level electronic controls embedded in machines.
  - Extended in the EI Pattern to refer to intelligence "embedded" at all levels, from molecular regulation to global and larger systems.
  - The central pattern being the combination of Cyber and Physical.

Embedded Intelligence (EI) Pattern

SYSTEMS ENGINEERING AND INTEGRATION

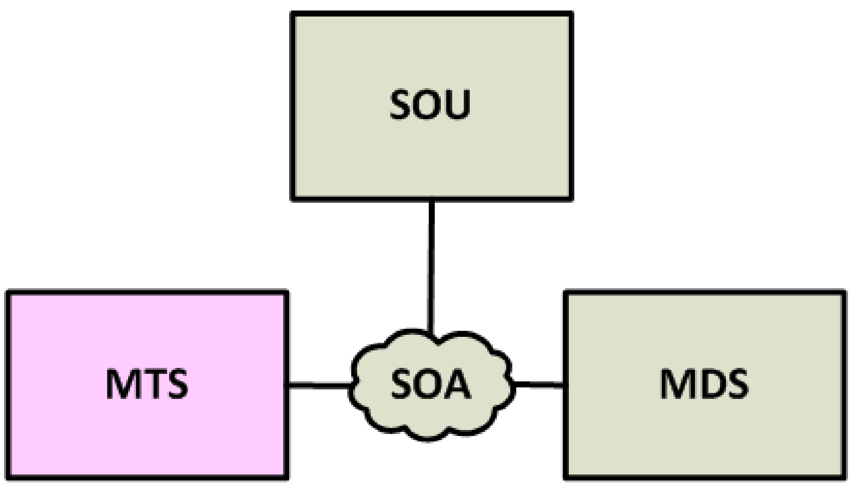
- The EI Pattern returns to the perspective of Norbert Wiener, who first coined the term "cybernetics" to refer to the study of communication and control in living and human-engineered systems[20]:

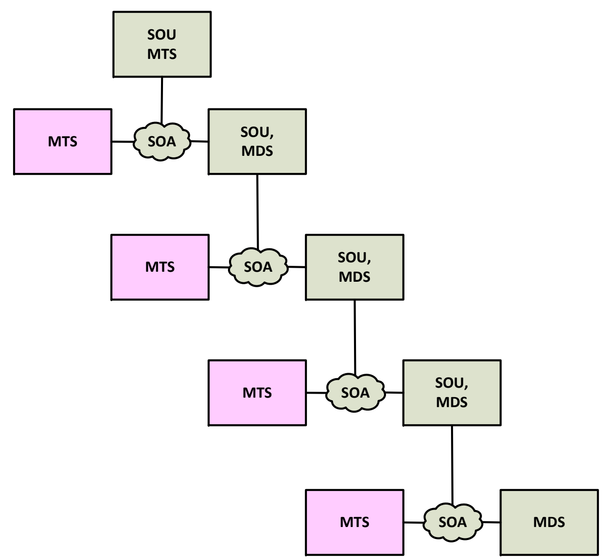


- Especially appropriate if we are interested in Cyber-Physical Systems – but now we are interested in more than just feedback and control performance (studied by Wiener) . . .

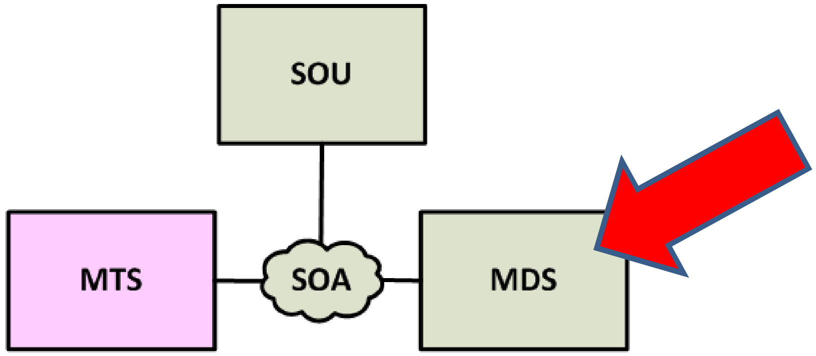
Embedded Intelligence (EI) Pattern

- The EI Pattern is an S*Pattern that emerges to describe intelligence in explicit models of evolving systems in the natural and man-made world:
  - Also referred to as the Management System Pattern.[21]
  - Concerned with the emergence of four roles, emergent at multiple hierarchical levels:



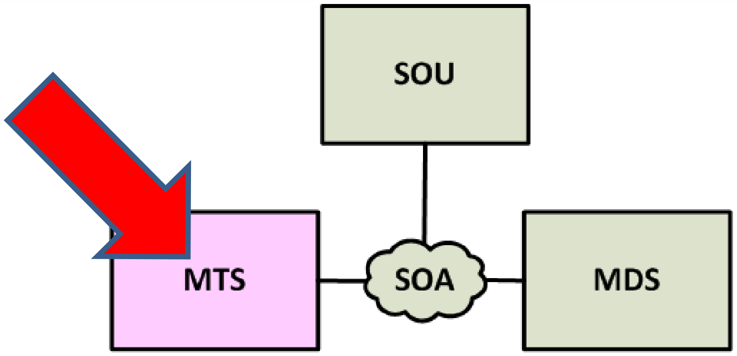


Embedded Intelligence (EI) Pattern



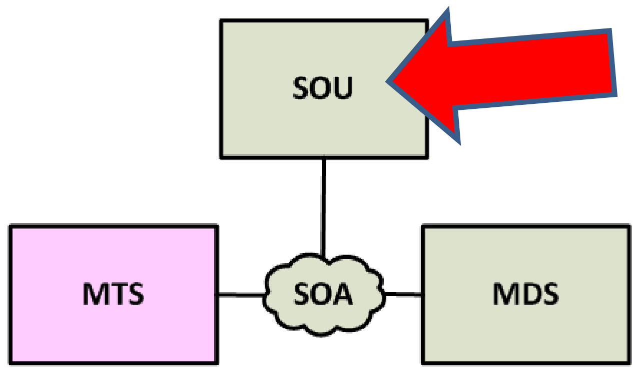
- Managed System (MDS): Any system behavior whose performance, configuration, faults, security, or accounting are to be managed--referred to as System Management Functional Areas (SMFAs) or in ISO terminology fault, configuration, accounting, performance, security (FCAPS).
- These are the roles played by the so-called "physical systems" in a cyber-physical system, providing physical services such as energy conversion, transport, transformation, or otherwise.

Embedded Intelligence (EI) Pattern



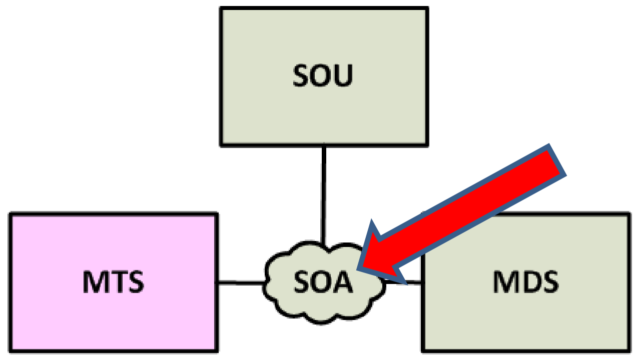
- Management System (MTS): The roles of performing management (active or passive) of any of the SMFAs of the managed system.
- These are so-called "cyber" roles in a cyber-physical system, and may be played by automation technology, human beings, or hybrids thereof, to accomplish regulatory or other management purposes.

Embedded Intelligence (EI) Pattern



- System of Users (SOU): The roles played by a system which consumes the services of an managed system and/or management system, including human system users or other service-consuming systems at higher levels.

Embedded Intelligence (EI) Pattern

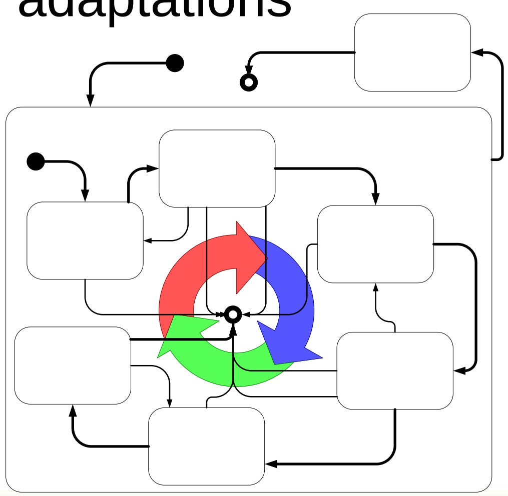


- System of Access (SOA): The roles providing a means of interaction between the other EI roles.
- Engineered sensors, actuators, the Internet, and human-machine interfaces have contributed greatly to the emergence of the “Internet of Things”..

Embedded Intelligence (EI) Pattern

SYSTEMS ENGINEERING AND INTEGRATION

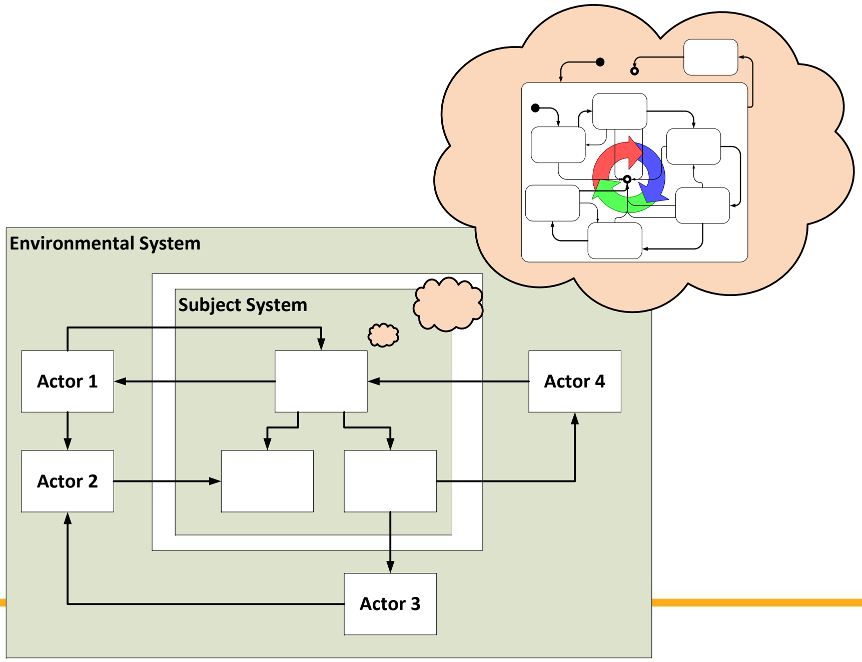
- The State Model portion of the EI Pattern provides insight into the nature of the "regulatory" role of embedded intelligence.
- These show numerous "situation resolution cycles" that drive the managed system to nominal states, when various situations are encountered:
  - Major mission cycles, from mission start to completion
  - Fault resolution cycles, other lesser or minor situation resolution cycles
  - Configuration change cycles, including adaptations
  - Fulfillment of requests for services
  - Security condition resolution cycles
  - Other situation resolution cycles
- Specific or general situations



Sample EI Situation Resolution Cycle

Embedded Intelligence (EI) Pattern

- A system that is capable of not only traversing a situation resolution cycle, but also <u>recognizing</u> that a triggering situation has arisen in the first place is said to be “Situationally Aware”:
  - If a human operator control panel has a “mode switch”, the system relies on the human to be aware of situations, launching the appropriate cycles
  - More advanced systems recognize these situations autonomously—also leading to EI Attention Model recognition of finite system resources.



Applications

- Because the EI Pattern describes all intelligent systems, it has many applications; among these . . .
  1. Identification of gaps or weaknesses in Cyber-Physical Systems; know "how a CPS really works" across situations.
  2. EI describes systems with human operators, autonomous systems, and hybrids; it helps us plan migration from manual to autonomous versions.
  3. Formal identification of mission family "envelopes" in which a system can perform in a resilient, reconfigurable fashion.
  4. Improve the interoperability of tools, departments, and suppliers, through shared semantic representations.
  5. Improved acquisition specification/proposal alignment, leveraging rapidly configurable and auditable model-based EI Patterns.
  6. Hierarchical intelligence, from vehicle subsystems, through vehicles, to vehicle fleet management systems.

Conclusions

1. As a model-based systems engineering approach, S*PBSE and the EI Pattern are well-suited to address CPS challenges.
2. S*PBSE provides a data model and framework that are both holistic and compact.
3. It addresses the core system science needed in designing CPS by making Interactions, the heart of systems, explicit.
4. The EI Pattern itself captures the central notion of general embedded intelligence in all its forms.
5. Both are essential in establishing patterns of adaptive hierarchical control as a framework for engineering trusted systems.

References

1. "NSF-Funded Joint Efforts", at www.nsf.gov/funding/pgm_summ.jsp?pims_id=503286
2. "The Most Innovative Companies 2013 – Lessons from Leaders" 2013 BCG Global Innovators Survey, BCG Analytics.
3. INCOSE SE Vision 2020
4. http://www.omgwiki.org/MBSE/doku.php?id=start
5. http://www.omgwiki.org/MBSE/doku.php?id=mbse:patterns:patterns
6. Christopher Alexander, Sara Ishikawa, Murray Silverstein, Max Jacobson, Ingrid Fiksdahl-King, and Shlomo Angel. A Pattern Language. Oxford University Press, New York, 1977.
7. Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides. Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley Publishing Company, Reading, MA, 1995.
8. Robert Cloutier. Applicability of Patterns to Architecting Complex Systems: Making Implicit Knowledge Explicit. VDM Verlag Dr. Müller. 2008.
9. Bill Schindel, Troy Peterson, "Introduction to Pattern-Based Systems Engineering (PBSE): Leveraging MBSE Techniques", in Proc. of INCOSE 2013 Great Lakes Regional Conference on Systems Engineering, Tutorial, October, 2013.
10. W. Schindel, "System Interactions: Making The Heart of Systems More Visible", in Proc. of INCOSE Great Lakes 2013 Regional Conference on Systems Engineering, October, 2013.
11. Abbreviated Systematica Glossary, Ordered by Concept, V 4.2.2, ICTT System Sciences, 2013.
12. W. Schindel, "The Impact of 'Dark Patterns' On Uncertainty: Enhancing Adaptability In The Systems World", in Proc. of INCOSE Great Lakes 2011 Regional Conference on Systems Engineering, Dearborn, MI, 2011.
13. W. Schindel, "Failure Analysis: Insights from Model-Based Systems Engineering", in Proceedings of INCOSE 2010 Symposium, July 2010.
14. W. Schindel, "Pattern-Based Systems Engineering: An Extension of Model-Based SE", INCOSE IS2005 Tutorial TIES 4, (2005).
15. W. Schindel, "Requirements Statements Are Transfer Functions: An Insight from Model-Based Systems Engineering", in Proc. of INCOSE 2005 International Symposium, (2005).
16. W. Schindel, and V. Smith, "Results of Applying a Families-of-Systems Approach to Systems Engineering of Product Line Families", SAE International, Technical Report 2002-01-3086 (2002).
17. J. Bradley, M. Hughes, and W. Schindel, "Optimizing Delivery of Global Pharmaceutical Packaging Solutions, Using Systems Engineering Patterns", in Proc. of the INCOSE 2010 International Symposium (2010).
18. W. Schindel, "Integrating Materials, Process & Product Portfolios: Lessons from Pattern-Based Systems Engineering", in Proc. of 2012 Conference of Society for the Advancement of Material and Process Engineering, 2012.
19. W. Schindel, "What Is the Smallest Model of a System?", in Proc. of the INCOSE 2011 International Symposium, International Council on Systems Engineering (2011).
20. Norbert Weiner, Cybernetics: Control and Communication in the Animal and the Machine, Cambridge, MA, MIT Press, 1965.
21. W. Schindel, and V. Smith, "Results of Applying a Families-of-Systems Approach to Systems Engineering of Product Line Families", SAE International, Technical Report 2002-01-3086 (2002).

Authors

Bill Schindel is president of ICTT System Sciences (www.ictt.com), a systems engineering company. His 40-year engineering career began in mil/aero systems with IBM Federal Systems, Owego, NY, included service as a faculty member of Rose-Hulman Institute of Technology, and founding of three commercial systems-based enterprises. He has led and consulted on improvement of engineering processes within automotive, medical/health care, manufacturing, telecommunications, aerospace, and consumer products businesses. Schindel earned the BS and MS in Mathematics. At the 2005 INCOSE International Symposium, he was recognized as the author of the outstanding paper on Modeling and Tools. Bill is an INCOSE CSEP, and Crossroads of America INCOSE chapter President.

Troy Peterson is a Chief Engineer and Booz Allen Fellow managing a portfolio of the firm's work in Michigan. His areas of expertise include strategy, systems engineering and analysis and product and enterprise development. He has led several international teams in the delivery of complex systems and has instituted numerous methodologies to speed development cycles. His experience spans across industry, non-profit and academic environments supporting numerous top priority programs. He completed Graduate Studies at MIT, obtained a MS in Technology Management from RPI and holds a BS in ME from MSU. Troy also is an ASQ Certified Six Sigma Black Belt, PMI PMP and INCOSE CSEP.