V1.4.2

25th anniversary annual INCOSE international symposium
Seattle, WA
July 13 - 16, 2015

INCOSE 2015

# Accelerating MBSE Impacts Across the Enterprise: Model-Based S*Patterns

William D. Schindel, Stephen A. Lewis, Jason J. Sherey,
ICTT System Sciences
schindel@ictt.com, lewis@ictt.com, sherey@ictt.com,

Saumya K. Sanyal
K2 Firm, LLC
sksanyal@k2firm.com

**Abstract:** Model-Based Systems Engineering (MBSE) methods can directly address "silos" problems. This paper reports on work by the INCOSE MBSE Initiative Patterns Challenge Team, focusing on Pattern-Based Systems Engineering (PBSE) using model-based system patterns based on the S*Metamodel, speeding and improving multiple SE processes.

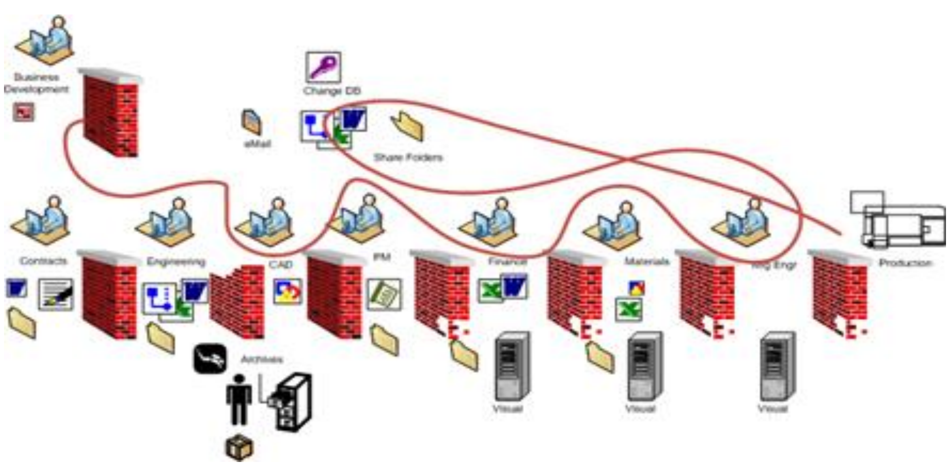
Distinctive are (1) the configurable, model-based nature of the patterns (not all historical patterns work has been model-based), (2) the technical scope of the models, encompassing requirements, design, failure mode, verification, other aspects, (3) the system scope of the models, encompassing whole systems, configurable product lines, and platforms, not just libraries of components, (4) the diverse and integrating cross-enterprise domains of the patterns, encompassing products, innovation processes, manufacturing, packaging / distribution, and other domains, and (5) the ability to enable a variety of COTS modeling languages and tools, PLM, and other enterprise information systems to integrate support of management and application of S*Patterns across enterprises.

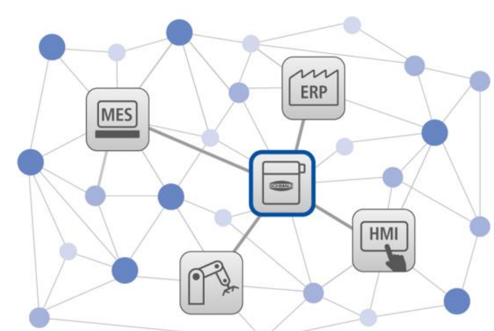
# Contents

# Business challenges and opportunities

- Competitive pressure cause departmental optimization
- Business Systems are designed and marketed to departments
- Interactions across departments are not well understood
- Each business systems have disparate methodology

Enterprise: Model Based S* Patterns provide a means to model, understand and evolve business systems within the Enterprise as the context



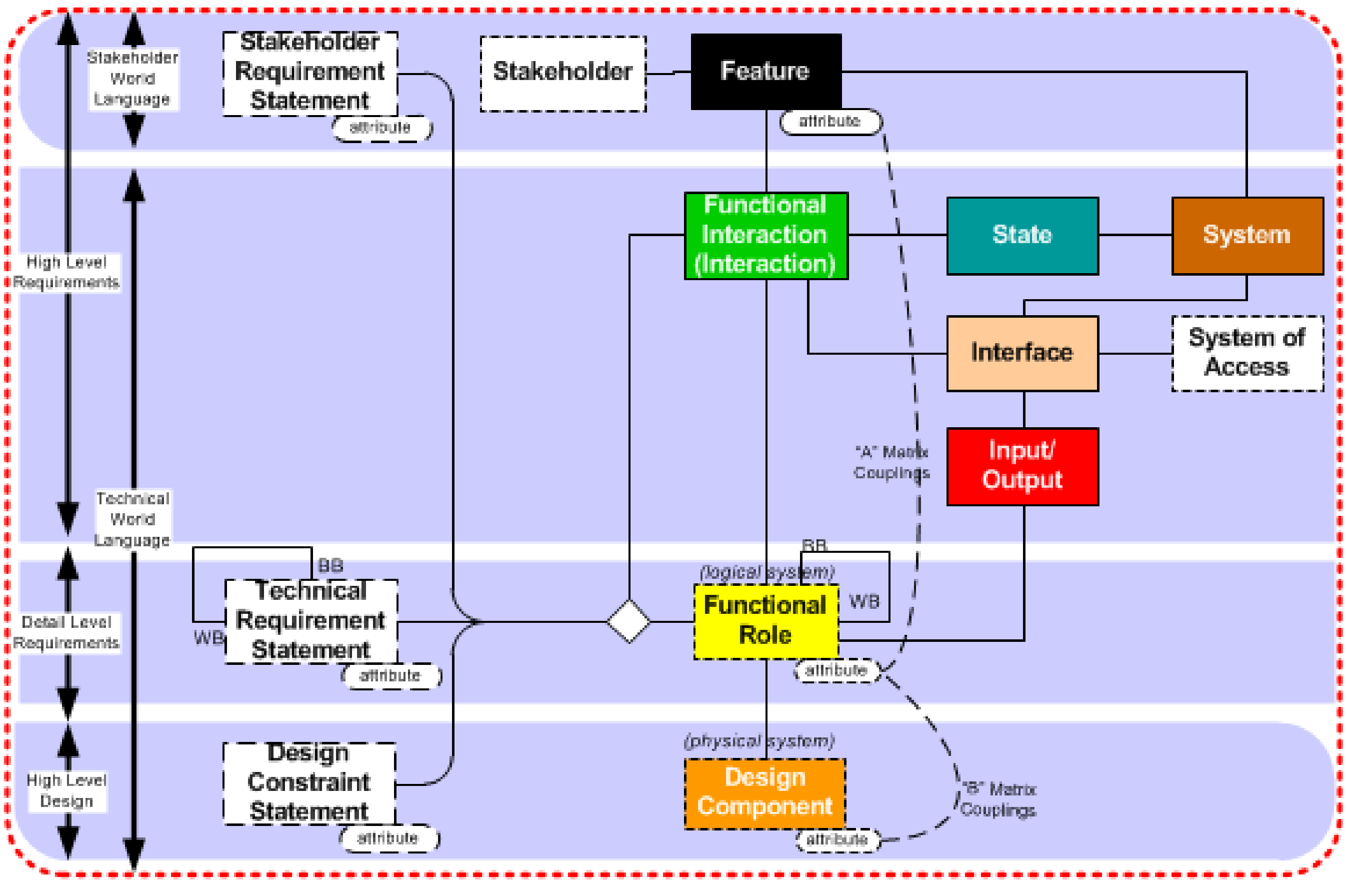



From Industry 4.0 ready: Vacuum solutions for the intelligent factory

# Representing system patterns with MBSE models

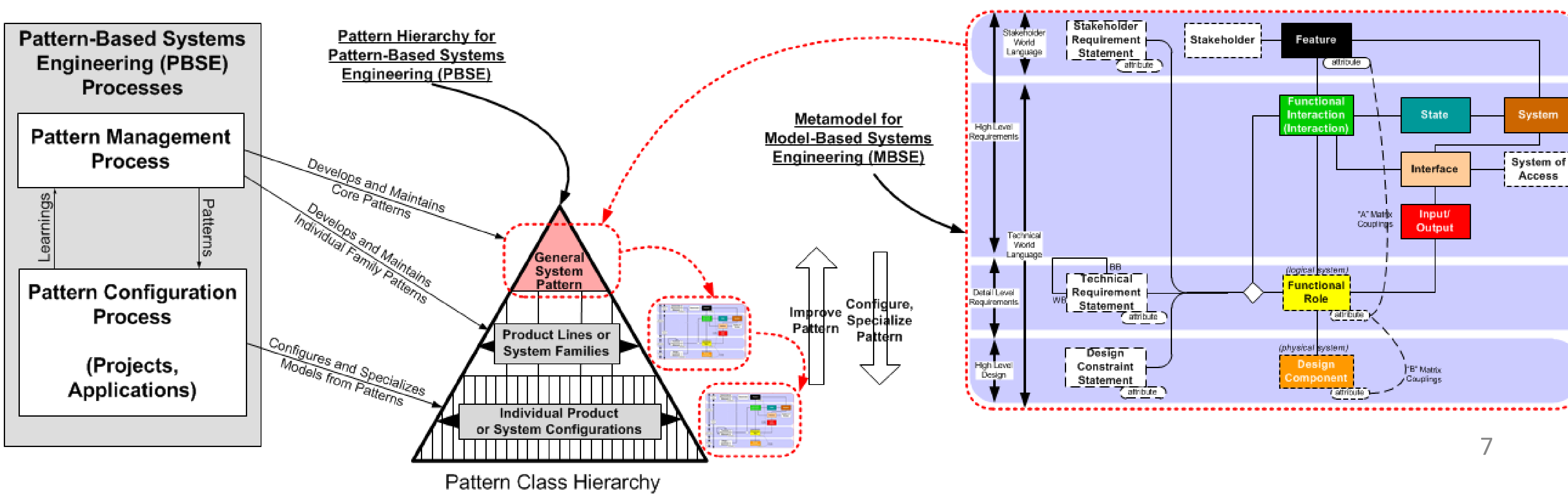
- The INCOSE Patterns Challenge Team of the MBSE Initiative was formed in 2013 to pursue practical use and awareness of system patterns of a particular type—called S*Patterns . . .

1. S*Models are MBSE models based on the S*Metamodel:
   - Provides explicit semantic meaning for S*Models
   - Includes some key systems concepts long established in science and engineering, but not always found explicitly in contemporary MBSE Models
   - A summary extract of some of the most important aspects:



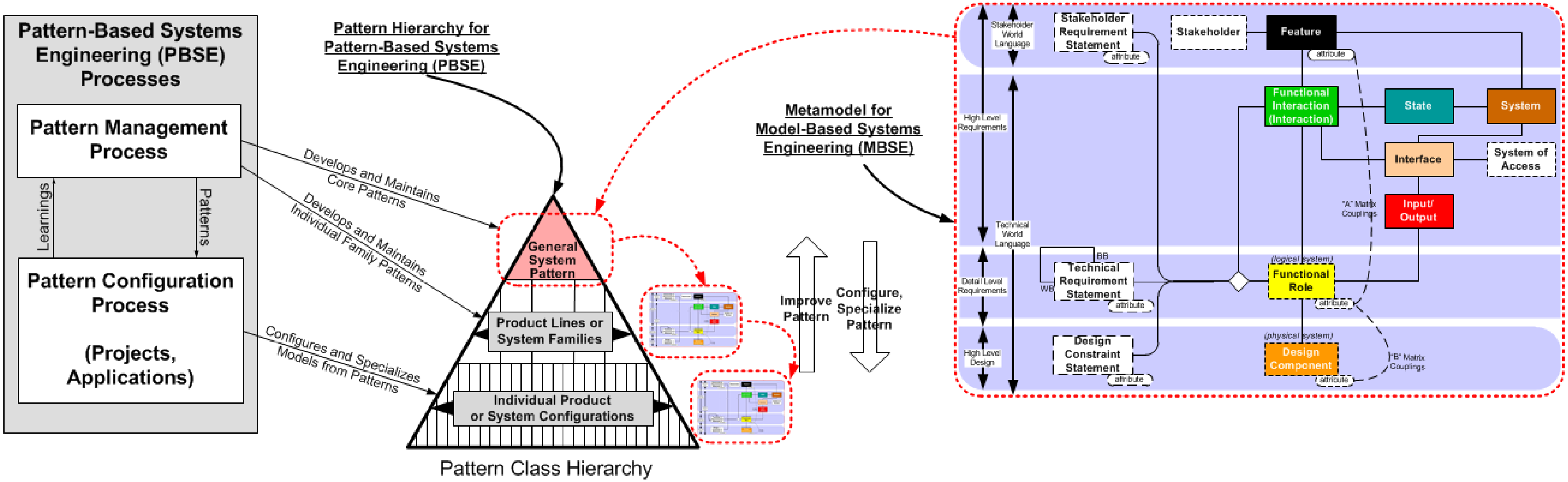
2. S*Patterns are configurable, re-usable S*Models:

- Earlier engineering patterns were not always based on the use of explicit MBSE Models.
- An S*Pattern may be thought of as a model of a family of systems, a platform, or a product line—an extended architectural framework.
- Once an S*Pattern has been created for an enterprise, it may be used during delivery projects to rapidly create high-grade S*Models.
- Typically an order of magnitude faster than creating a new model, and configured for the specific needs at hand:



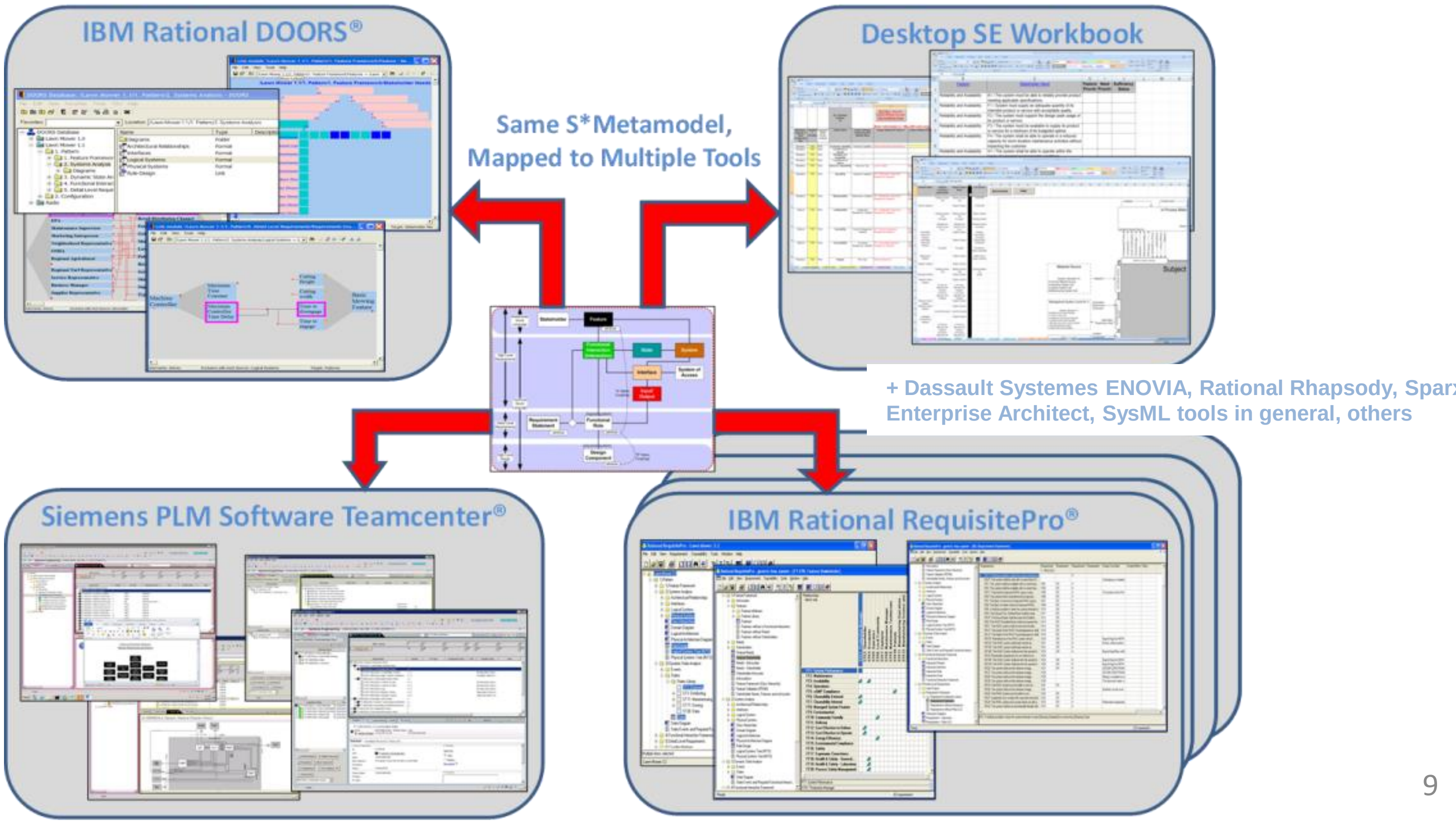
# 3. S*Models and S*Patterns are <u>independent of any specific modeling language</u>:

- Typically expressed using any of a variety of the popular standard or third-party contemporary modeling languages.
- A formal mapping into each such language helps (e.g., a profile).
- SysML is common but not required for S*Models, S*Patterns.
- Strengthens the semantics of existing languages in key areas required for pattern representation in engineering & science.

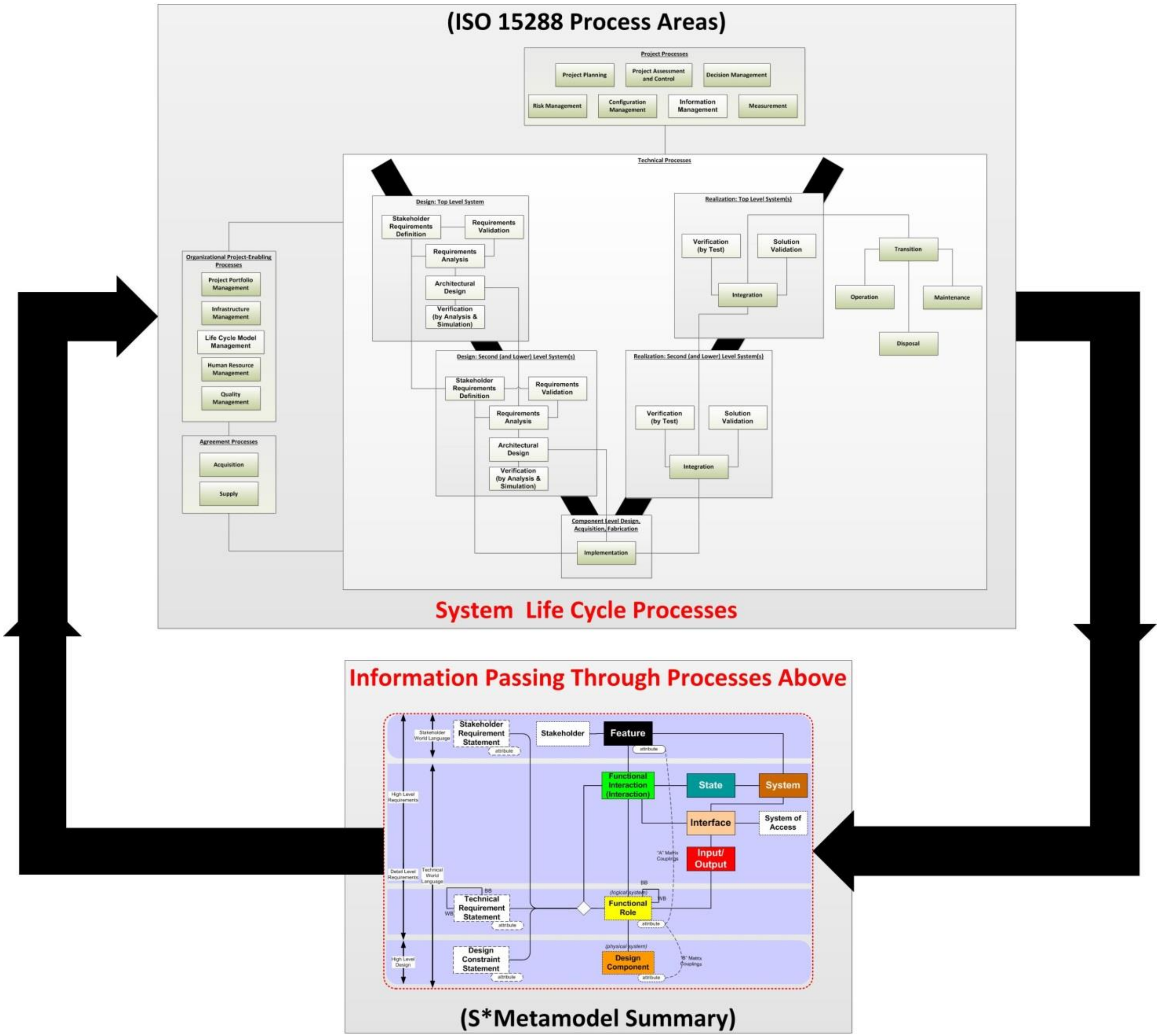


# 4. S*Models and S*Patterns are independent of any specific software tool or information system:

- May be authored, stored, or managed using a variety of popular third-party COTS modeling tools, information systems.
- This paper illustrates S*Models and S*Patterns in several third party COTS modeling tool, requirements database, and PLM systems already in use.
- Supported by a formal mapping into the schema of each such repository.



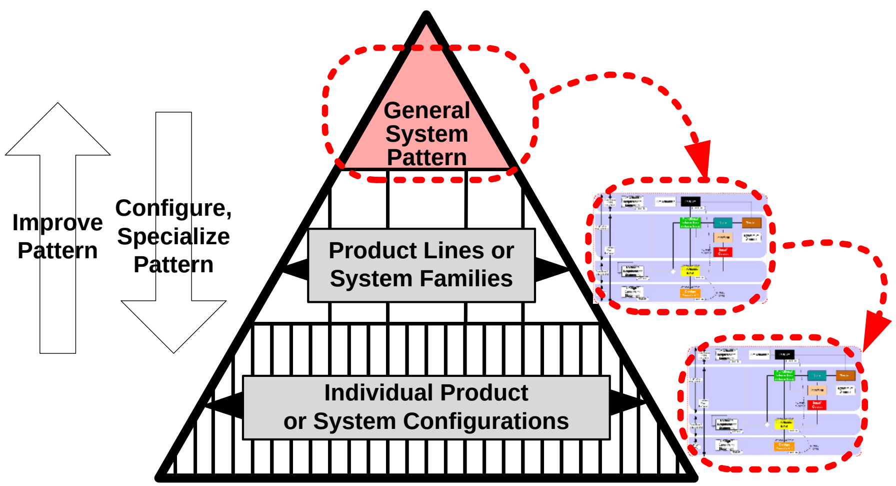
# 5. SE Processes consume and produce information:



5. SE Processes <u>consume and produce information</u>:
   - Systems Engineering has a tradition of extensive description of process and procedure.
   - In describing SE, less ink is usually devoted to describing that produced/consumed information than the related processes and procedures.
   - Compare this to the amount of description of underlying relationships of physics, chemistry, or electromagnetic phenomena, versus the related engineering procedures of ME, ChE, or EE.
   - Now that we are making more use of explicit system models—closer to the language of science and mathematics—we suggest a shift in this balance is in order.
   - The idea is that the SE process should be primarily performed to drive trajectories in (modeled) configuration space.

**6. The processes of SE, even MBSE, most often presented, conceived, or practiced as if each engineering project is "starting from scratch":**

- And yet, in nearly all cases we are starting from extensive prior experience, in the heads of the team.
- Much traditional SE guidance is typically offered on discovery, synthesis, and analysis of stakeholders, requirements, architectures, allocations, trade-spaces, risks and failure modes, etc.—in a context that might suggest first-time study of the system of interest.
- But what about formal guidance about use of what we already know?

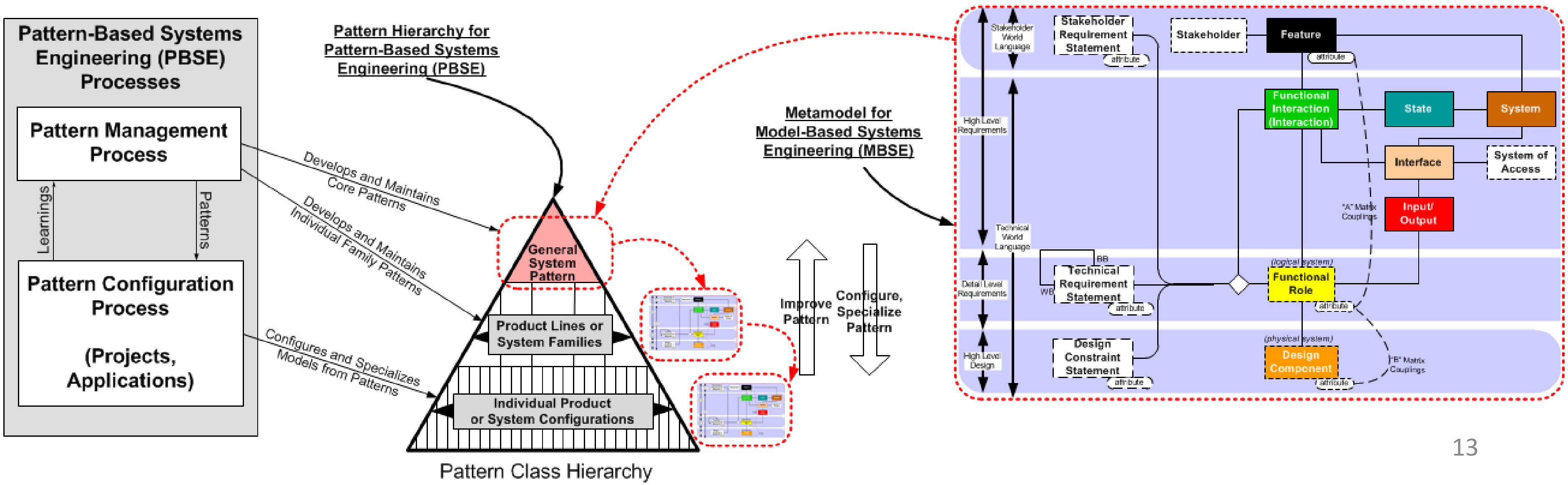



Pattern Class Hierarchy

- Recent progress with Product Line Engineering shows a rebalancing.

MBSE Initiative Patterns Challenge team IS2015 papers illustrate use of MBSE Patterns in:

- autonomous ground vehicles
- automated safety critical system test
- optimization of design review assignment
- and cross-functional enterprise dependencies in product manufacturing businesses.



# Integrating S*Patterns, at enterprise and lower levels

- Agricultural silos are designed to minimize unwanted external interactions harmful to stored silage:

- The “silos” metaphor is an infamous description invoked to describe certain organizational pathologies.
- This can be an unfair attack on the hard-working staff in those areas, when it is really an emergent aspect of the overall enterprise system.
- Dealing with this situation on a system basis provides a more constructive way to engage.

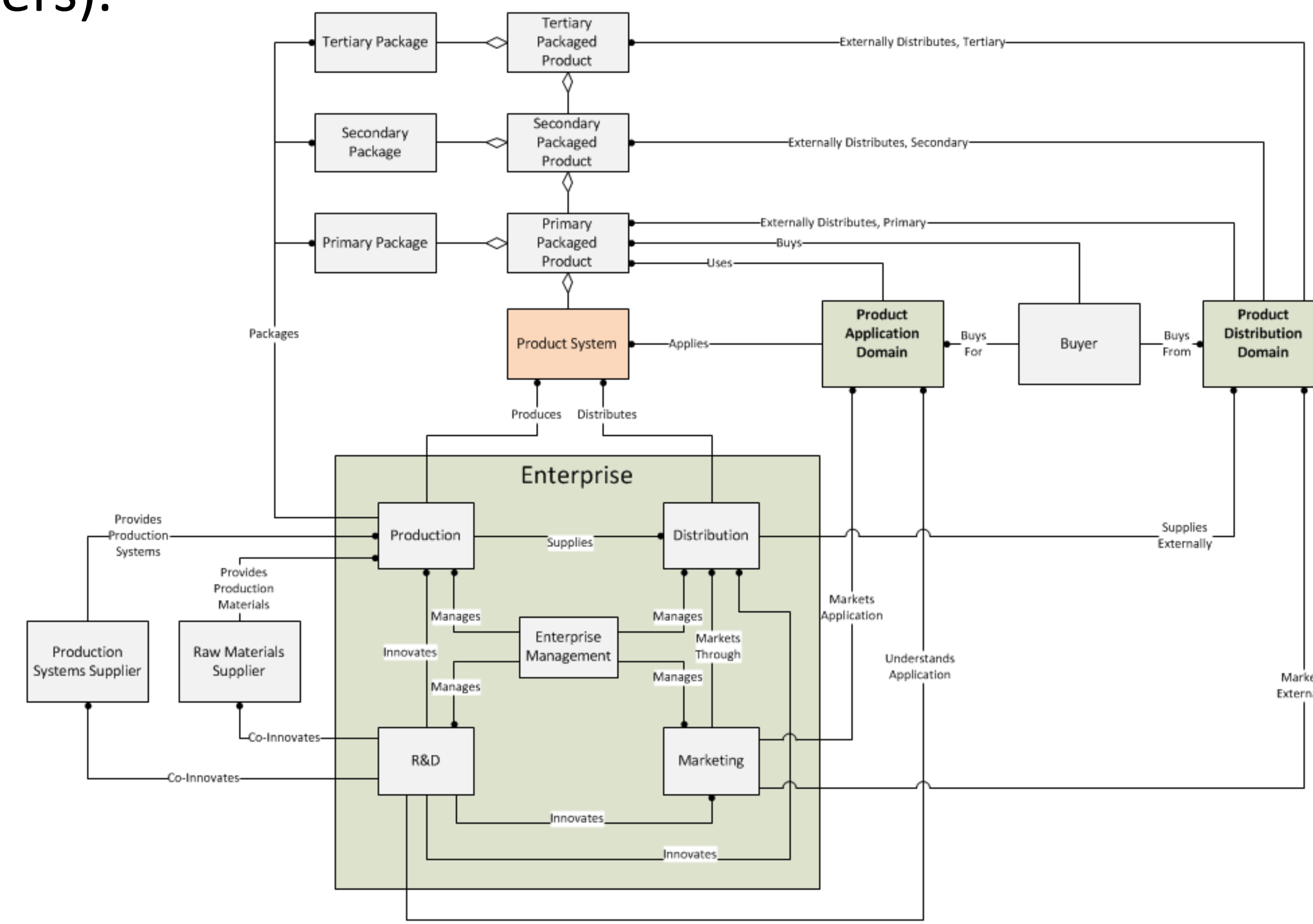
# Integrated models, at different levels

- Enterprise System Model, aligning the following:
- Product Application Domain Model
- Manufacturing System Model
- Distribution System Model
- Service and Support System Model
- System of Innovation Model
- Other enterprise subsystem models

For a given project, each S*Model is configured from its respective S*Pattern.

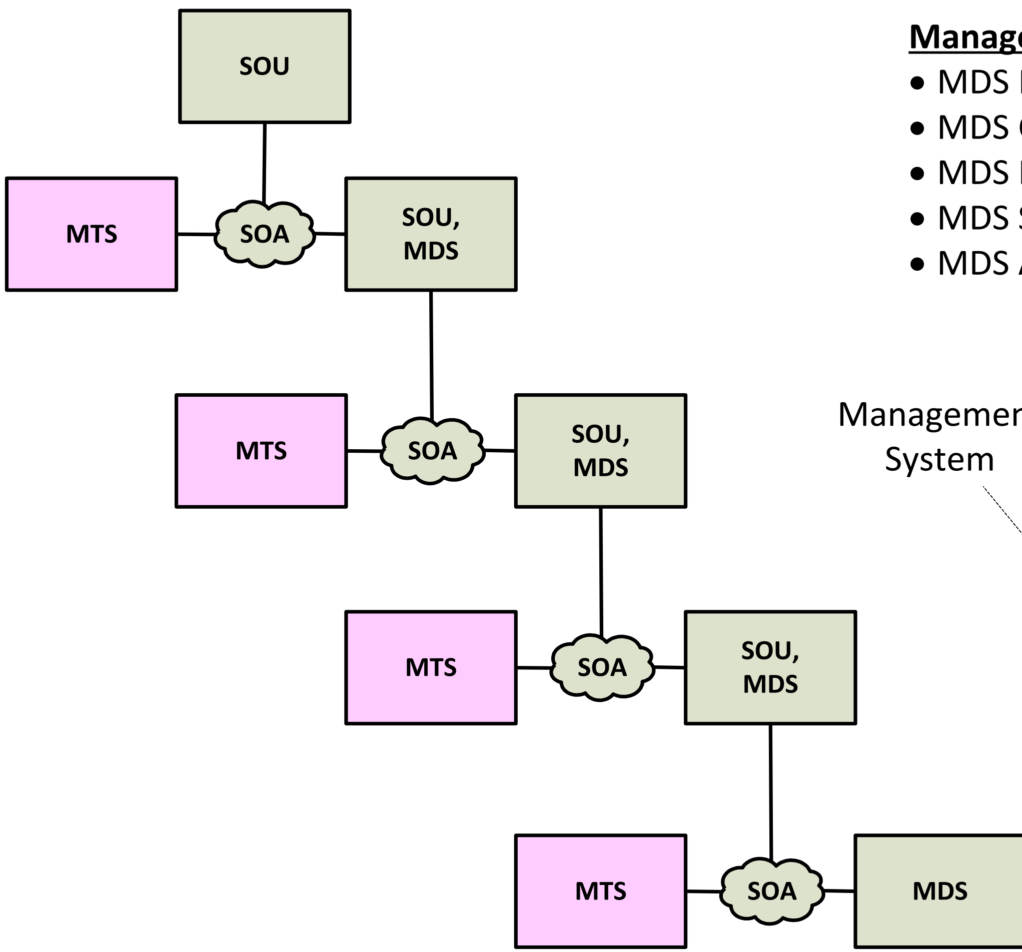
We are interested in interactions (or their lack) between enterprise functional areas, along with external actors:

- S*Interactions are exchanges of information, mass flows, energy, forces
- An overall enterprise behavior emerges, as seen by external actors (e.g., customers).

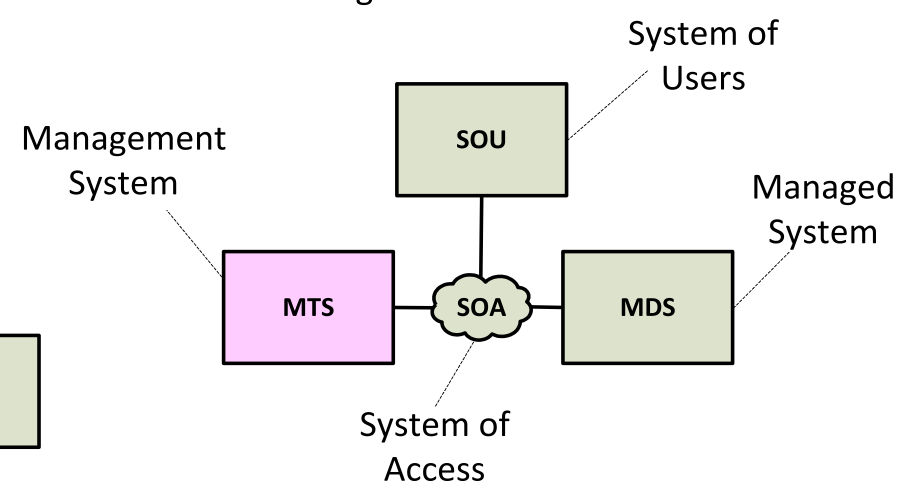


# Inclusion of the Embedded Intelligence (EI) Pattern (aka Management Systems Pattern)

- The hierarchy of (human and automated) Management Systems, below and above the Enterprise level:

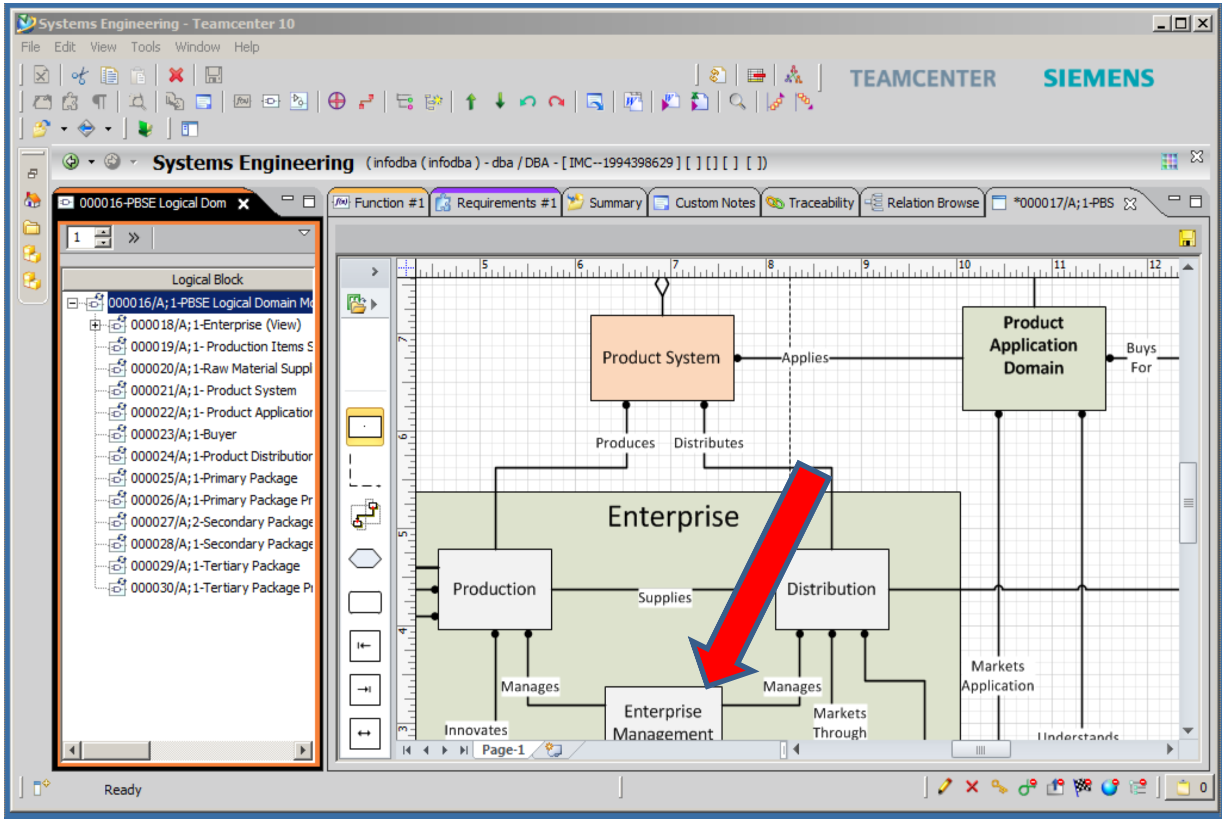



**Management of:**
- MDS Performance
- MDS Configuration
- MDS Faults
- MDS Security
- MDS Accounting



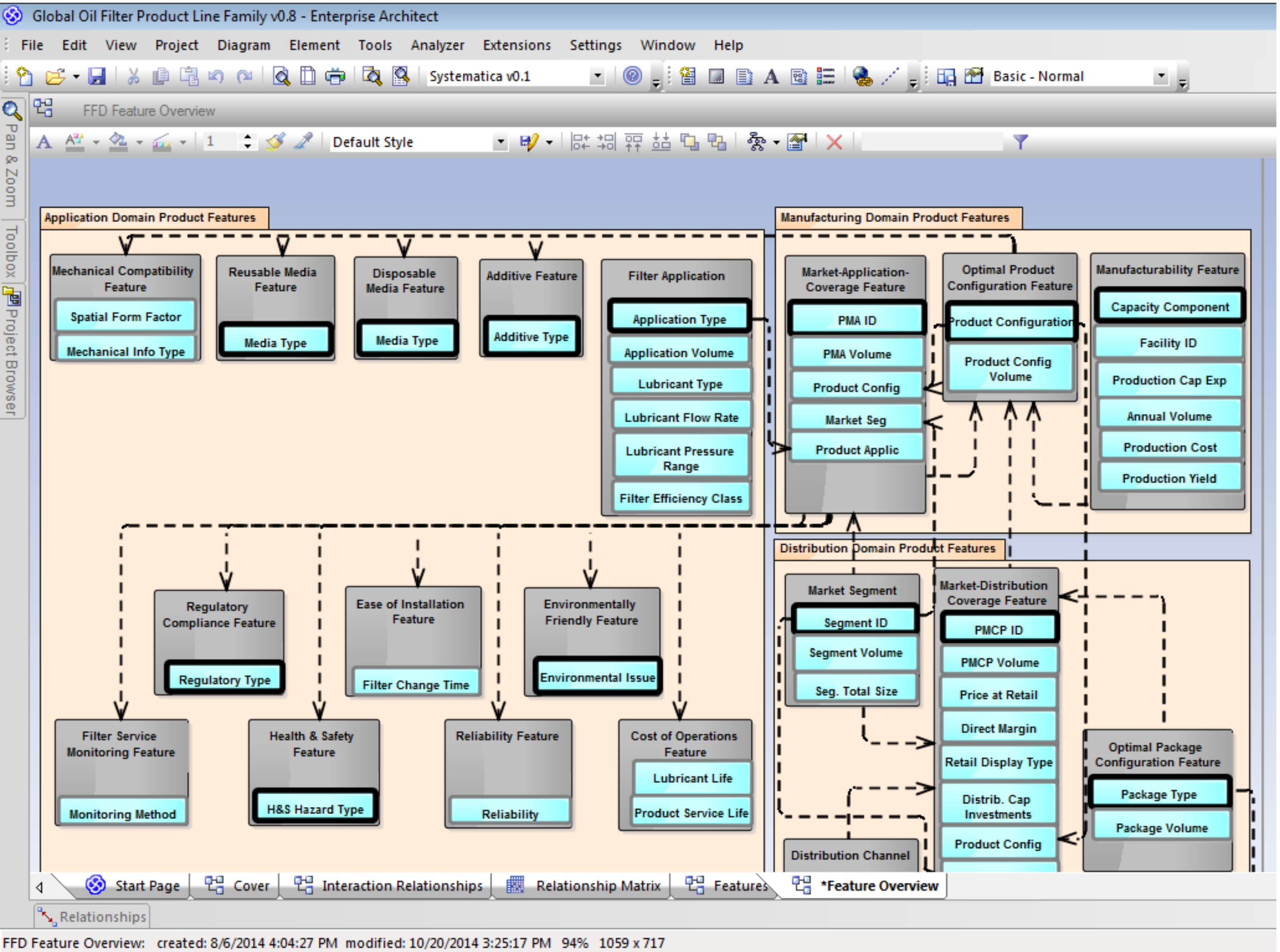
Explicitly modeling and managing the Enterprise level system, for successful enterprise projects:

1. <u>Pattern Management Process</u>: Creates and improves the re-usable Enterprise S*Pattern, in appropriate modeling tool.
2. <u>Pattern Configuration Process</u>: Configures and applies the pattern, for each major enterprise project. Can be managed in a PLM or other system, using an S*Pattern Configuration Agent:



# The product application domain pattern

- An S*Pattern that describes the enterprise's products, platform, product line, in service in its intended application or other domain



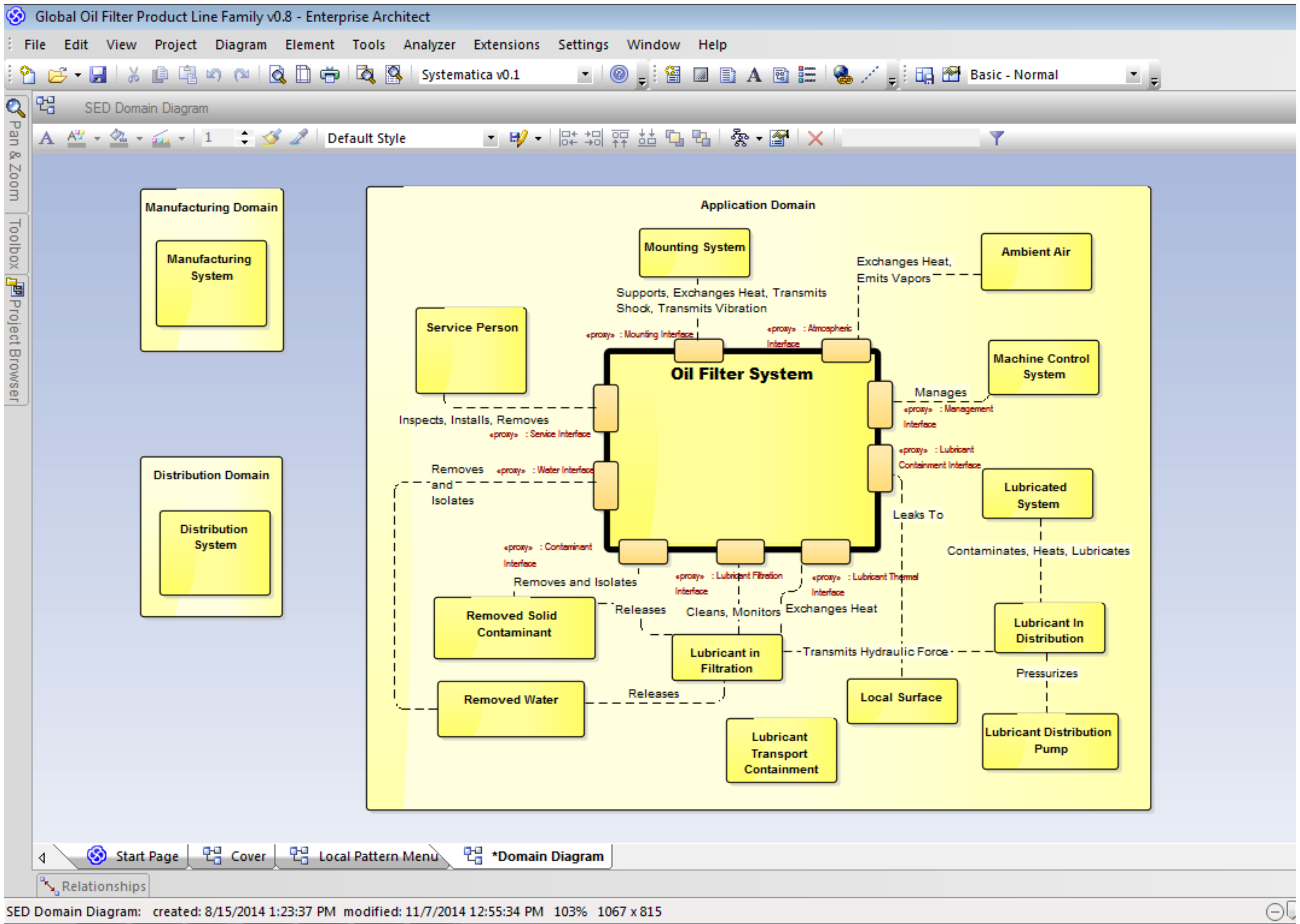

**Oil Filter Product Line: Stakeholder Feature Model**

Pattern is modeled and maintained in a modeling tool.

When configured for a given project, may reside in a PLM System.

# The product application domain pattern

- An S*Pattern that describes the enterprise's products, platform, product line, in service in its intended application or other domain



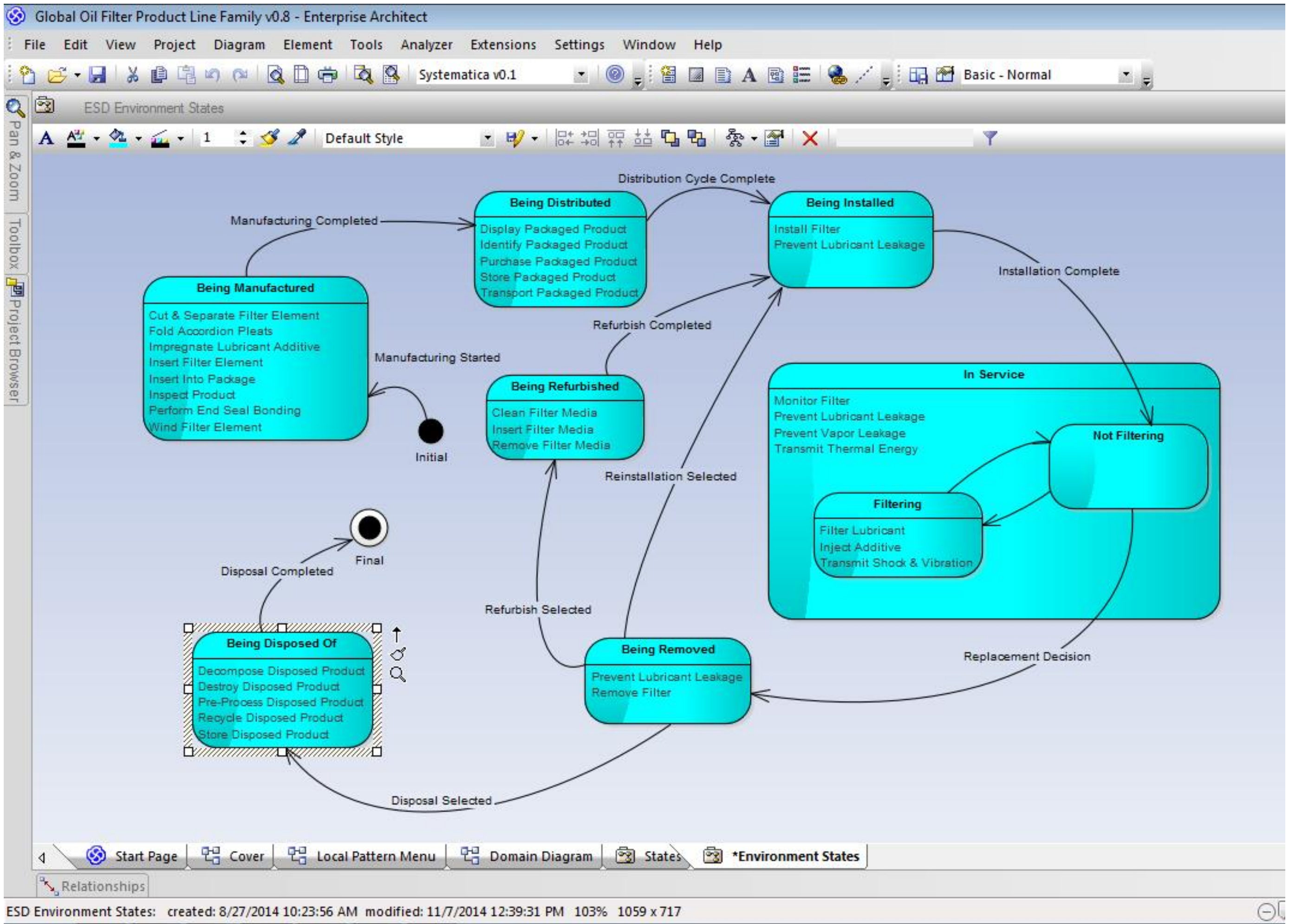

**<u>Oil Filter Product Line</u>: Domain Model**

Pattern is modeled and maintained in a modeling tool.

When configured for a given project, may reside in a PLM System.

# The product application domain pattern

- An S*Pattern that describes the enterprise's products, platform, product line, in service in its intended application or other domain




**<u>Oil Filter Product Line</u>: State Model**

Pattern is modeled and maintained in a modeling tool.

When configured for a given project, may reside in a PLM System.

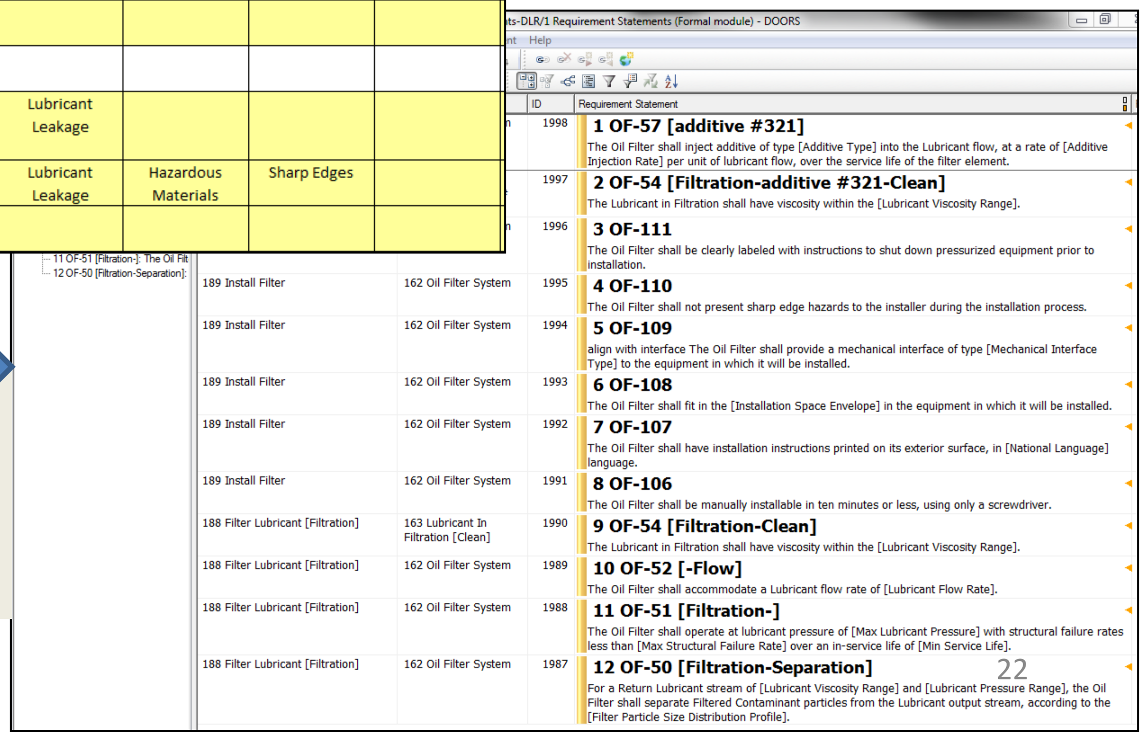
# The product application domain pattern

- An S*Pattern that describes the enterprise's products, platform, product line, in service in its intended application or other domain

| Mandatory, Optional, or Other Configuration Rule | Populate? (Yes/No) | Feature Name | Feature Attribute Primary Key (PK) Attribute Name | Feature Attribute PK Value #1 | Feature Attribute PK Value #2 | Feature Attribute PK Value #3 | Feature Attribute PK Value #4 | Feature Attribute PK Value #5 | Feature Attribute PK Value #6 |
|---|---|---|---|---|---|---|---|---|---|
| One per Filter Application Type | Yes | Filter Application Feature | Application Type | Consumer Automotive | | | | | |
| Mandatory for Oil Filter | Yes | Mechanical Compatibility Feature | -- | | | | | | |
| Mandatory for Oil Filter | Yes | Cost of Operation Feature | -- | | | | | | |
| Mandatory for Oil Filter | Yes | Reliability Feature | -- | | | | | | |
| One Per Additive Type | Yes | Additive Feature | Additive Type | additive #321 | | | | | |
| Optional | No | Disposable Filter Media Feature | Media Type | | | | | | |
| Optional | No | Reusable Filter Media Feature | Media Type | | | | | | |
| Optional | No | Filter Service Monitoring Feature | -- | | | | | | |
| One Per Environmental Issue | Yes | Environmentally Friendly Feature | Environmental Issue | Solid Waste Disposal | Gaseous Emissions | Lubricant Leakage | | | |
| One Per Regulatory Issue | Yes | Regulatory Compliance | Regulatory Issue | Solid Waste Disposal | Gaseous Emissions | Lubricant Leakage | Hazardous Materials | Sharp Edges | |
| One Per Health & Safety Issue | Yes | Health & Safety | H&S Hazard Type | Hazardous Materials | Sharp Edges | | | | |

Consumer Automotive
Commercial Automotive
Fixed Based Engine Syste
Harsh Environment
High Temperature Enviror
Cold Environment

**Oil Filter System**: Selecting Product Features during the Pattern Configuration Process (in PLM System, for example)

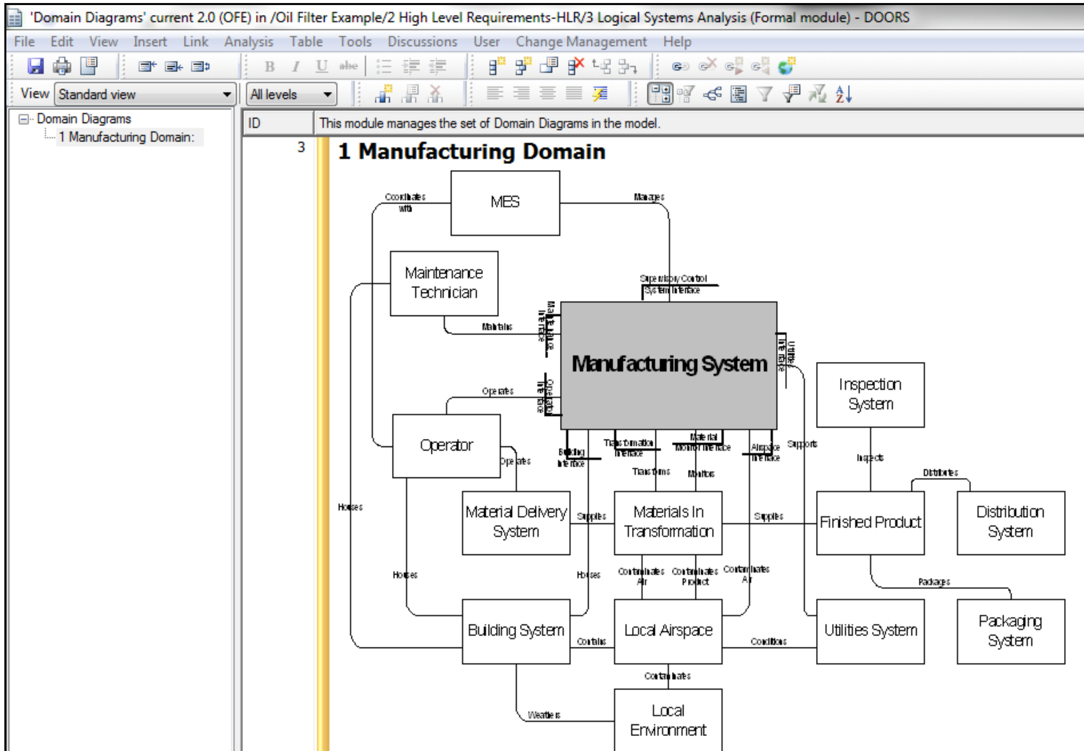



| | | ID | Requirement Statement |
|---|---|---|---|
| | | 1998 | **1 OF-57 [additive #321]** The Oil Filter shall inject additive of type [Additive Type] into the Lubricant flow, at a rate of [Additive Injection Rate] per unit of lubricant flow, over the service life of the filter element. |
| | | 1997 | **2 OF-54 [Filtration-additive #321-Clean]** The Lubricant in Filtration shall have viscosity within the [Lubricant Viscosity Range]. |
| | | 1996 | **3 OF-111** The Oil Filter shall be clearly labeled with instructions to shut down pressurized equipment prior to installation. |
| 189 Install Filter | 162 Oil Filter System | 1995 | **4 OF-110** The Oil Filter shall not present sharp edge hazards to the installer during the installation process. |
| 189 Install Filter | 162 Oil Filter System | 1994 | **5 OF-109** align with interface The Oil Filter shall provide a mechanical interface of type [Mechanical Interface Type] to the equipment in which it will be installed. |
| 189 Install Filter | 162 Oil Filter System | 1993 | **6 OF-108** The Oil Filter shall fit in the [Installation Space Envelope] in the equipment in which it will be installed. |
| 189 Install Filter | 162 Oil Filter System | 1992 | **7 OF-107** The Oil Filter shall have installation instructions printed on its exterior surface, in [National Language] language. |
| 189 Install Filter | 162 Oil Filter System | 1991 | **8 OF-106** The Oil Filter shall be manually installable in ten minutes or less, using only a screwdriver. |
| 188 Filter Lubricant [Filtration] | 163 Lubricant In Filtration [Clean] | 1990 | **9 OF-54 [Filtration-Clean]** The Lubricant in Filtration shall have viscosity within the [Lubricant Viscosity Range]. |
| 188 Filter Lubricant [Filtration] | 162 Oil Filter System | 1989 | **10 OF-52 [-Flow]** The Oil Filter shall accommodate a Lubricant flow rate of [Lubricant Flow Rate]. |
| 188 Filter Lubricant [Filtration] | 162 Oil Filter System | 1988 | **11 OF-51 [Filtration-]** The Oil Filter shall operate at lubricant pressure of [Max Lubricant Pressure] with structural failure rates less than [Max Structural Failure Rate] over an in-service life of [Min Service Life]. |
| 188 Filter Lubricant [Filtration] | 162 Oil Filter System | 1987 | **12 OF-50 [Filtration-Separation]** For a Return Lubricant stream of [Lubricant Viscosity Range] and [Lubricant Pressure Range], the Oil Filter shall separate Filtered Contaminant particles from the Lubricant output stream, according to the [Filter Particle Size Distribution Profile]. |

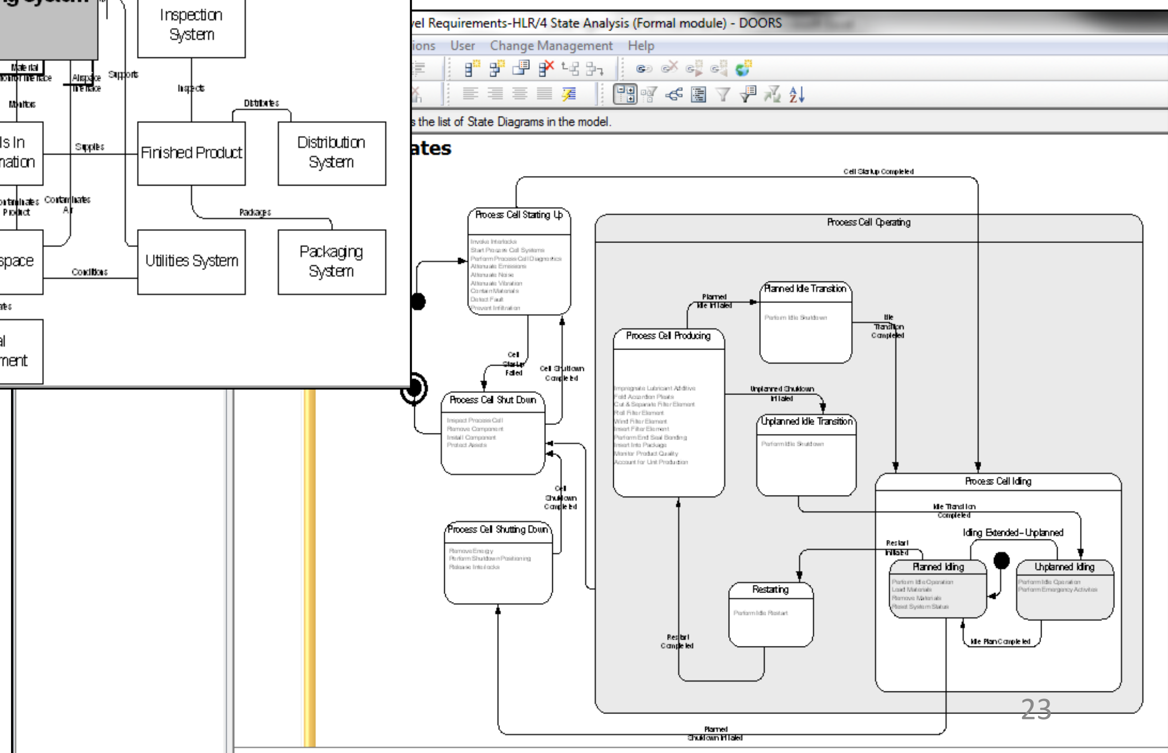
**Oil Filter System:** Resulting Auto-Configured Requirements, after Pattern Configuration Process (in Requirements System, for example)

# The manufacturing system pattern

- An S*Pattern that describes the enterprise's production systems, during the intended use or other parts of their life cycles



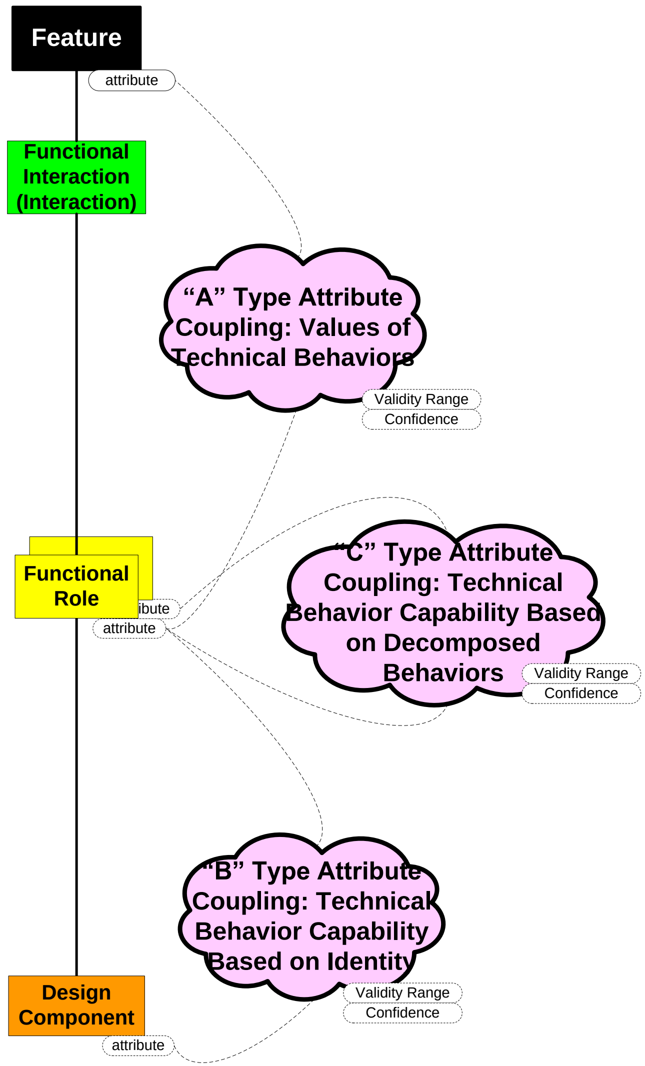

**Oil Filter Manufacturing System: Domain Model**




**Oil Filter Manufacturing System: State Model**

# The manufacturing system pattern

- An S*Pattern that describes the enterprise’s production systems, during the intended use or other parts of their life cycles



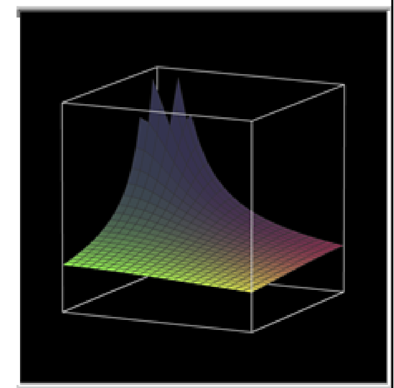

**Oil Filter Manufacturing System:**

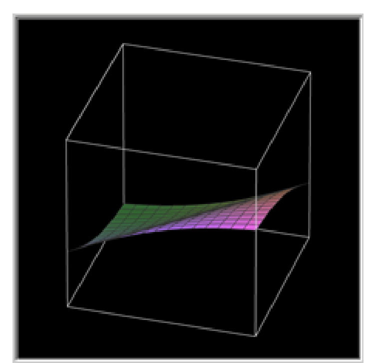
**Detail Bonding Interaction**

**Attribute Coupling Model**

Unit Throughput as a function of Heat Time and Spray Time

X-Axis (Horizontal 1):
Heat Time
Y-Axis (Horizontal 2):
Spray Time
Z-Axis (Vertical):
Unit Throughput





Additive Life as a function of Heat Time and Spray Time

X-Axis (Horizontal 1):
Heat Time
Y-Axis (Horizontal 2):
Spray Time
Z-Axis (Vertical):
Additive Life

# The System of Innovation pattern

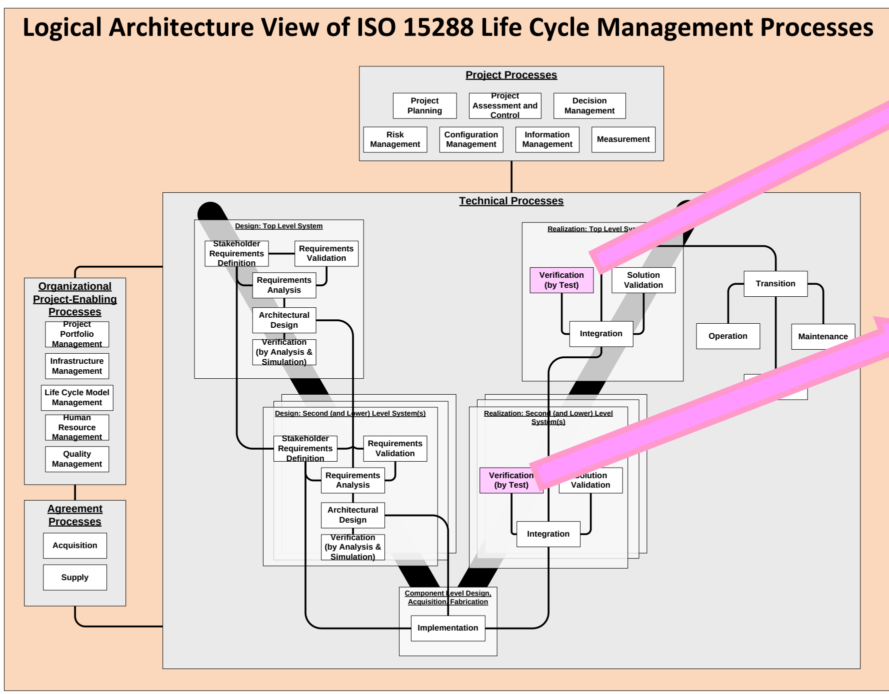
- The enterprise subsystem responsible for creating new instance configurations of all the other systems:
  - Product System
  - Manufacturing System
  - Distribution System
  - Service and Support System
  - Other enterprise subsystems

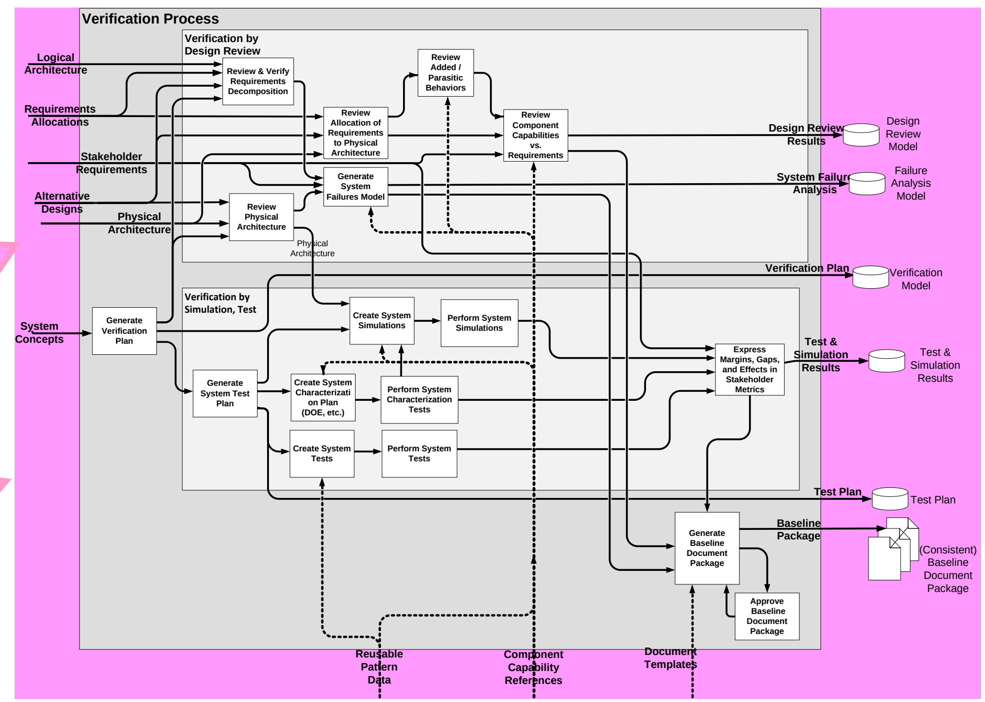
- Includes product R&D, but also manufacturing process development, equipment engineering, distribution, service, and other aspects.

# The System of Innovation pattern

- Includes a formal S*Model of  ISO15288 processes, along with their subsystem details:
  - Tailored to explicitly represent MBSE and PBSE aspects
  - Managed as an S*Pattern and configured as an S*Model for each project.

**<u>System of Innovation</u>: Logical Architecture, from ISO15288**

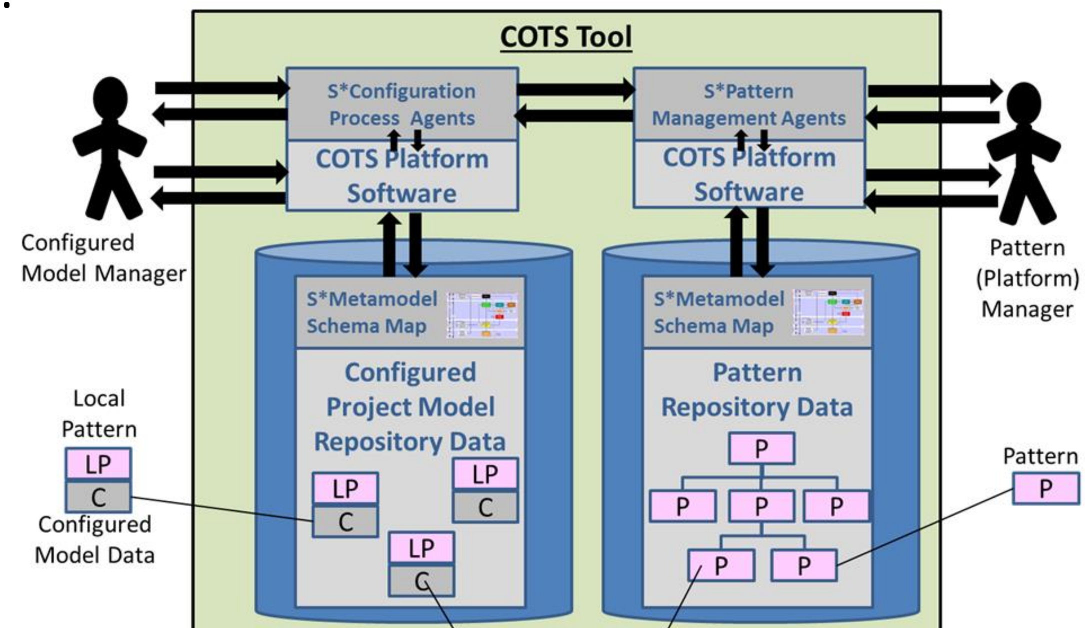


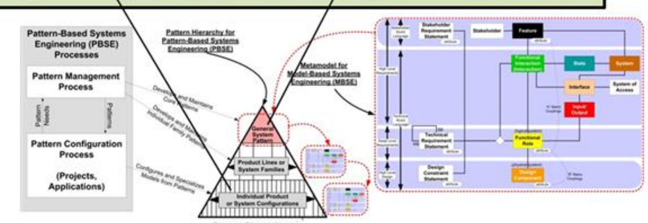




**<u>System of Innovation</u>: Example Verification by Test subsystem**

# Human and Information Systems Agents Enable the System of Innovation

- S*Metamodel schema map (profile) is provided for each modeling tool and engineering, manufacturing, or enterprise information system,
- So they can uniformly represent project-specific configured S*Models and generalized S*Patterns.
- S*Configuration Process agents likewise provide a unified approach to configuring S*Models from S*Patterns:

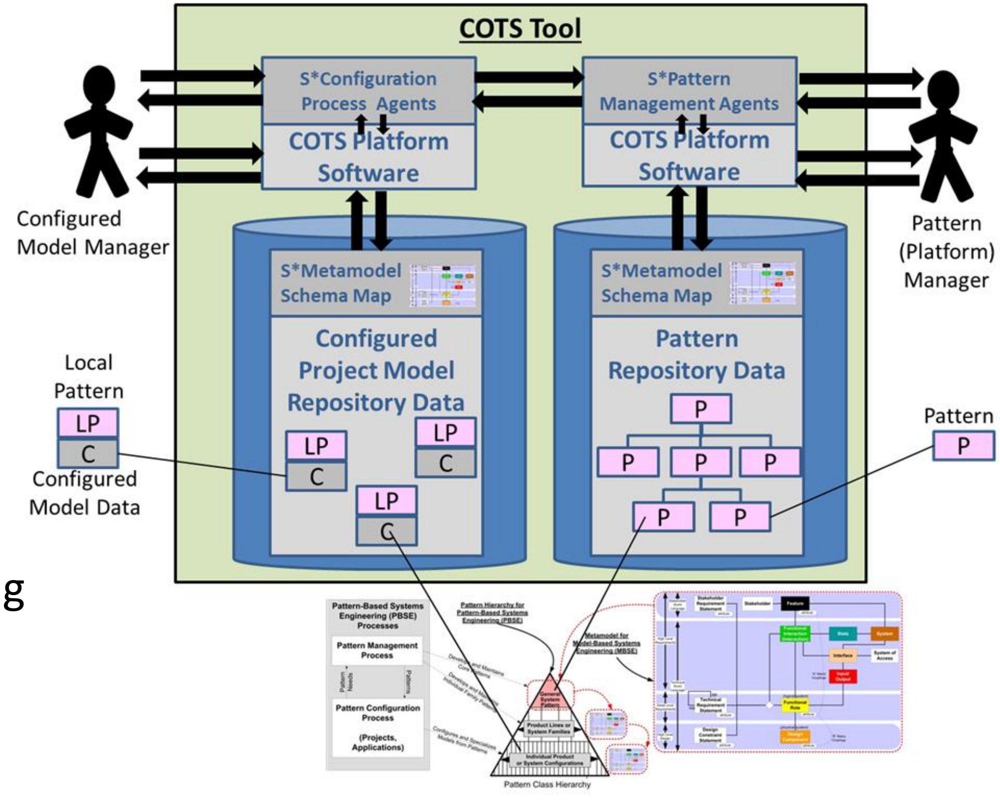



Existing COTS Engineering & Modeling Tools, PLM Systems Can All Support Common Underlying S*Metamodel, Innovation Processes

# Human and Information Systems Agents Enable the System of Innovation

- Many third-party COTS tools and information systems provide some means of data exchange among them, using standards-based or other types of exchange interfaces.
- Open standards for information exchange or federation are likewise emerging.
- The approach described here extends this by providing a deeper underlying semantic compatibility between these existing systems, while still taking advantage of those emerging exchange and transport interfaces.
- This is more than an information technology approach, as it also aligns the semantics of how human users of these systems conceive of the information they manage.




Existing COTS Engineering & Modeling Tools, PLM Systems Can All Support Common Underlying S*Metamodel, Innovation Processes

# Summary and Conclusions

1. MBSE and PBSE not only apply across the enterprise—they can directly address enterprise-level challenges that arise out of interactions of lower-level enterprise subsystems.
2. The expressive power of Models is further leveraged when they do not have to be developed "from scratch" for each project, but can be derived from Patterns that also accumulate learning as it occurs, becoming a new form of IP, increasing the agility of the enterprise.
3. This changes the perspective of individuals from "learn modeling" to "learn the model" (referring to the enterprise's MBSE pattern IP)—a different perspective from the more popular "learn how to model" movement.
4. In addition to improving the power and capabilities of individuals, existing and in-service engineering modeling and simulation tools, databases, and PLM systems likewise have their power increased when they are enabled to accommodate the stronger semantics of the S*Metamodel.

# References


1. (Alexander 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. (Berg 2014) Berg, E., "Affordable Systems Engineering: An Application of Model-Based System Patterns To Consumer Packaged Goods Products, Manufacturing, and Distribution", at INCOSE IW2014 MBSE Workshop, 2014.
3. (Bradley, Hughes, Schindel 2010) Bradley, J., Hughes, M. and Schindel, W., "Optimizing Delivery of Global Pharmaceutical Packaging Solutions, Using Systems Engineering Patterns" Proceedings of the INCOSE 2010 International Symposium (2010).
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. (Dove, LaBarge 2014) Dove, R., LaBarge, R., "Fundamentals of Agile Systems Engineering—Part 1" and "Part 2", INCOSE IS2014, July, 2014.
7. (Dove, Schindel 2015) Dove, R., and Schindel, W., "Agile Modeling and Modeling Agile Systems", to appear at INCOSE IW2015 MBSE Workshop, Torrance, CA, January 24, 2015.
8. (Estafan 2008) Estafan, J. 2008. Survey of model-based systems engineering (MBSE) methodologies. INCOSE MBSE Initiative.
9. (Gamma et al 1995) Gamma, E., Helm, R., Johnson, R., and Vlissides, J., Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley Publishing Company, Reading, MA, 1995.
10. (INCOSE Handbook 2014) INCOSE Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities, Version 4, International Council on Systems Engineering (2014).
11. (INCOSE Patterns Team 2014) INCOSE/OMG MBSE Initiative: Patterns Challenge Team 2013-14 Web Site: http://www.omgwiki.org/MBSE/doku.php?id=mbse:patterns:patterns
12. (ISO 15288 2014) ISO/IEC 15288: Systems Engineering—System Life Cycle Processes. International Standards Organization (2014).
13. (ISO 26550 2013) ISO/IEC 26550 "Software and Systems Engineering—Reference Model for Product Line Engineering and Management", 2013.
14. (ISO 42010 2011) ISO/IEC/IEEE 42010 "Systems and Software Engineering—Architecture Description", 2011.
15. (Nolan, Pickard, Russell, Schindel 2015) Nolan, A., Pickard, A., Russell, J., Schindel, W., "When two is good company, but more is not a crowd", to appear in Proc. of the INCOSE 2015 International Symposium, Seattle, WA, July, 2015.
16. (Peterson, Schindel 2015) Peterson, T., Schindel, W., "Unmanned Ground Vehicle Platforms and Model-Based System Patterns: An Example", to appear in Proc. of the INCOSE 2015 International Symposium, Seattle, WA, July, 2015.

# References


17. (Schindel 2005a) Schindel, W., “Pattern-Based Systems Engineering: An Extension of Model-Based SE”, INCOSE IS2005 Tutorial TIES 4, (2005).
18. (Schindel 2005b) Schindel, W. “Requirements statements are transfer functions: An insight from model-based systems engineering”, Proceedings of INCOSE 2005 International Symposium, (2005).
19. (Schindel 2010) Schindel, W., “Failure Analysis: Insights from Model-Based Systems Engineering”, INCOSE International Symposium, Chicago, 2010.
20. (Schindel 2011a) Schindel, W. “Innovation as Emergence: Hybrid Agent Enablers for Evolutionary Competence” in Complex Adaptive Systems, Volume 1, Cihan H. Dagli, Editor in Chief, Elsevier, 2011
21. (Schindel 2011b) Schindel, W. “What Is the Smallest Model of a System?”, Proc. of the INCOSE 2011 International Symposium, International Council on Systems Engineering (2011).
22. (Schindel 2011c) Schindel, W., “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
23. (Schindel 2012a) Schindel, W. “Introduction to Pattern-Based Systems Engineering (PBSE)”, INCOSE Finger Lakes Chapter Webinar, April 26, 2012.
24. (Schindel 2012 b) Schindel, W., “Integrating Materials, Process, & Product Portfolios: Lessons from Pattern-Based Systems Engineering”, in Proc. of Society for Advancement of Materials and Process Engineering (SAMPE), 2012
25. (Schindel 2013a) Schindel, W. “Interactions: At the Heart of Systems”, INCOSE Great Lakes Regional Conference on Systems Engineering, W. Lafayette, IN, October, 2013.
26. (Schindel 2013b) Schindel, W., “Systems of Innovation II: The Emergence of Purpose”, Proceedings of INCOSE 2013 International Symposium (2013).
27. (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
28. (Schindel 2015a) Schindel, W., “Maps or Itineraries? A Systems Engineering Insight from Ancient Navigators”, to appear in Proc. of the INCOSE 2015 International Symposium, Seattle, WA, July, 2015.
29. (Schindel 2015b) Schindel, W., “System Life Cycle Trajectories: Tracking Innovation Paths Using System DNA”, to appear in Proc. of the INCOSE 2015 International Symposium, Seattle, WA, July, 2015.
30. (Schindel, Beihoff 2012) Schindel W., and Beihoff, B., “Systems of Innovation I: Models of Their Health and Pathologies”, Proc. of INCOSE International Symposium, 2012.
31. (Schindel, Peffers, Hanson, Ahmed, Kline 2011) Schindel, W., Peffers, S., Hanson, J., Ahmed, J., Kline, W., “All Innovation is Innovation of Systems : An Integrated 3-D Model of Innovation Competencies ”, Proc. of ASEE 2011 Conference, American Association for Engineering Education, (2011).
32. (Schindel, Peterson 2013) Schindel, W., and Peterson, T. “Introduction to Pattern-Based Systems Engineering (PBSE): Leveraging MBSE Techniques”, in Proc. of INCOSE 2013 International Symposium, Tutorial, June, 2013.
33. (Schindel, 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).