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

Model-Based System Patterns for Automated Ground Vehicle Platforms

Troy Peterson
Booz Allen Hamilton
troy.peterson@incose.org

Bill Schindel
ICTT System Sciences
schindel@ictt.com

Agenda

- Automated Ground Vehicle Context
- Model Based Systems Engineering
- Pattern Based Systems Engineering
- AGV Diversity and Complexity
- AGV Model-Based Pattern
  - Vehicle Pattern
  - Embedded Intelligence Pattern
- Follow on Work
- Conclusions

Automated Ground Vehicle Context

Automated Ground Vehicle (AGV) platforms

- Are Rapidly and dramatically increasing in complexity
- Can change the way we develop, manage and interact with systems
- Fit the National Science Foundation (NSF) definition of Cyber Physical Systems (CPS) as "engineered systems that are built from, and depend upon, the seamless integration of computational algorithms and physical components". With challenges that are both significant and far-reaching.

AGV Benefits	AGV Challenges
Efficiency / Adaptability	Efficiency / Adaptability
Safety / Trustworthiness/ Security	Safety / Trustworthiness/ Security
Situational Awareness / Data Available	Situational Awareness / Data Available
Undiscovered Opportunities	Undiscovered Risks

Is Autonomy Accelerating Innovation

- The need to manage the inherent complexity within AGVs is very relevant today and the complexity is only accelerating.
- Google's driverless cars are expected to hit the markets between 2017 and 2020.
- Automakers including GM, Ford, Nissan, Volkswagen, Mercedes Benz, Volvo, and Volvo Truck all have autonomous programs.

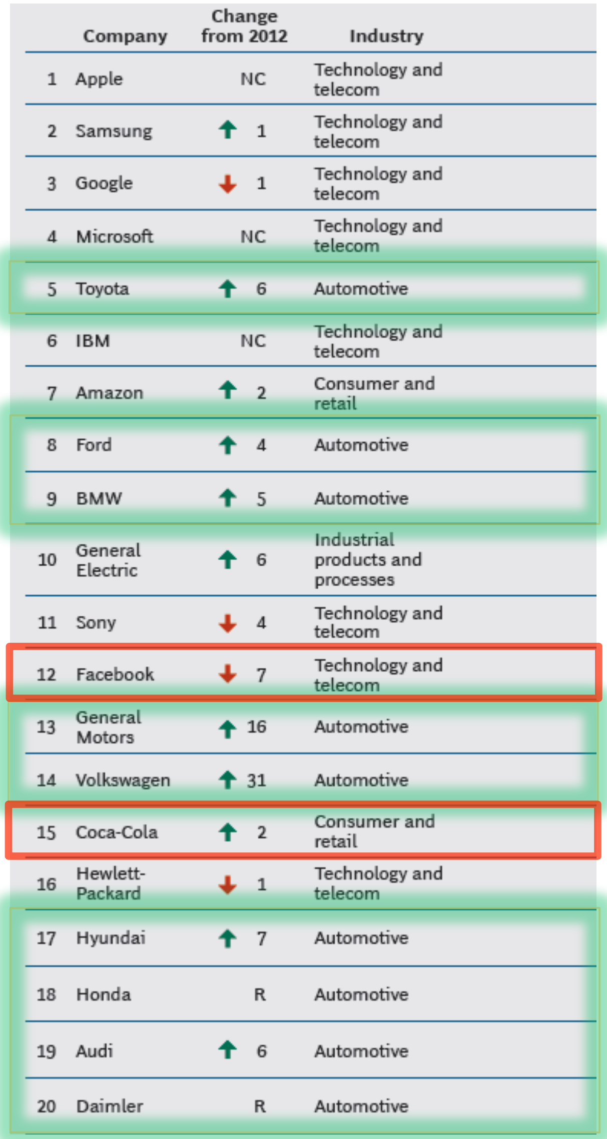
$$\begin{tikzcd}
\text{AGV Competition} \arrow[d, bend left=60, "+"] \arrow[d, phantom, "\circlearrowleft \text{R}"] \\
\text{AGV Innovation} \arrow[u, bend left=60, "+"]
\end{tikzcd}$$

AGV Innovation Acceleration

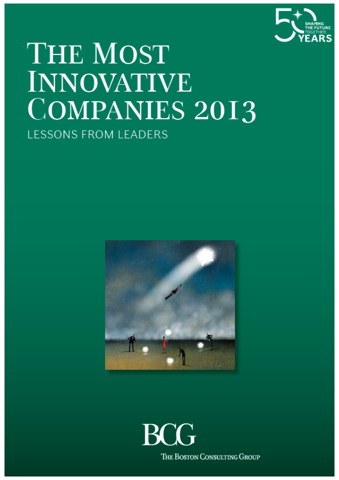
The Boston Consulting Group's 2013 report on the most innovative companies noted that for the first time, there were more automakers than tech companies listed the top 20 (BCG, 2013).

One surprise, not noted in the study however, was how many of these companies were involved with ground vehicle automation.

For the first time, there are more automakers than tech companies in the top 20.



	Company	Change from 2012	Industry
1	Apple	NC	Technology and telecom
2	Samsung	↑ 1	Technology and telecom
3	Google	↓ 1	Technology and telecom
4	Microsoft	NC	Technology and telecom
5	Toyota	↑ 6	Automotive
6	IBM	NC	Technology and telecom
7	Amazon	↑ 2	Consumer and retail
8	Ford	↑ 4	Automotive
9	BMW	↑ 5	Automotive
10	General Electric	↑ 6	Industrial products and processes
11	Sony	↓ 4	Technology and telecom
12	Facebook	↓ 7	Technology and telecom
13	General Motors	↑ 16	Automotive
14	Volkswagen	↑ 31	Automotive
15	Coca-Cola	↑ 2	Consumer and retail
16	Hewlett-Packard	↓ 1	Technology and telecom
17	Hyundai	↑ 7	Automotive
18	Honda	R	Automotive
19	Audi	↑ 6	Automotive
20	Daimler	R	Automotive


THE MOST INNOVATIVE COMPANIES 2013
LESSONS FROM LEADERS
50 YEARS
BCG
THE BOSTON CONSULTING GROUP

Model Based Systems Engineering

- The systemic complexity of AGVs demands a systems engineering approach.
- It requires a systems paradigm which is interdisciplinary, applies the requisite physics-based and mathematical models to represent them – it requires modeling.
- INCOSE defines Model-Based Systems Engineering (MBSE) as "the formalized application of modeling to support system requirements, design, analysis, verification and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases…"
- The INCOSE MBSE wiki notes that "Modeling has always been an important part of systems engineering to support functional, performance, and other types of engineering analysis."
- MBSE is often discussed as being composed of three fundamental elements – tool, language and method. This third element, method, has not always been given proper consideration.

INCOSE SE Vision 2020 - http://www.omgwiki.org/MBSE/doku.php?id=start

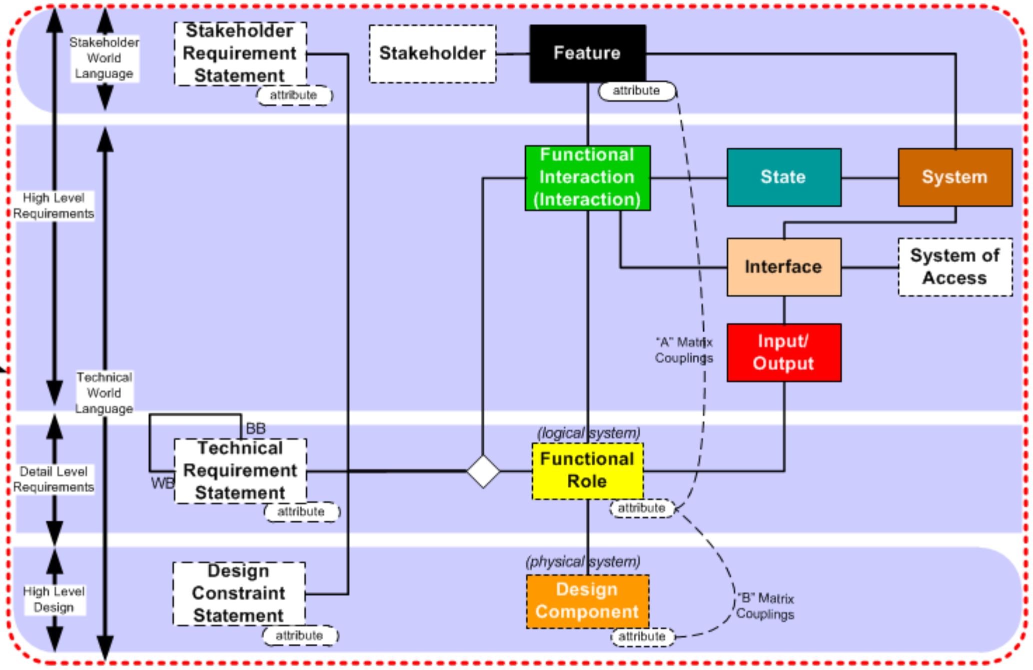
Pattern Based Systems Engineering

- As a Model-Based Systems Engineering (MBSE) methodology, Pattern-Based Systems Engineering (PBSE) addresses complex systems, with a reduction in modeling effort.
- Gains are possible because projects using PBSE get a “learning curve jumpstart” from an existing model-based pattern and its previous users, rapidly gaining the advantages of its content.
- The term “pattern” appears repeatedly in the history of design, such as civil architecture, software design and systems engineering. While these are all loosely similar in the abstract the PBSE methodology referred to by this paper, based on S*Models and S*Patterns which are distinguished by:
  - <u>S*Patterns are Model-Based</u>: Patterns represented by formal system models, and specifically those which are re-usable, configurable models based on the underlying S*Metamodel.
  - <u>Scope of S*Patterns</u>: Patterns which will usually cover entire systems, not just smaller-scale element design patterns within them. For this reason, the typical scope of an S*Pattern applications may be thought of as re-usable, configurable models of whole domains or platform.

The S* Model

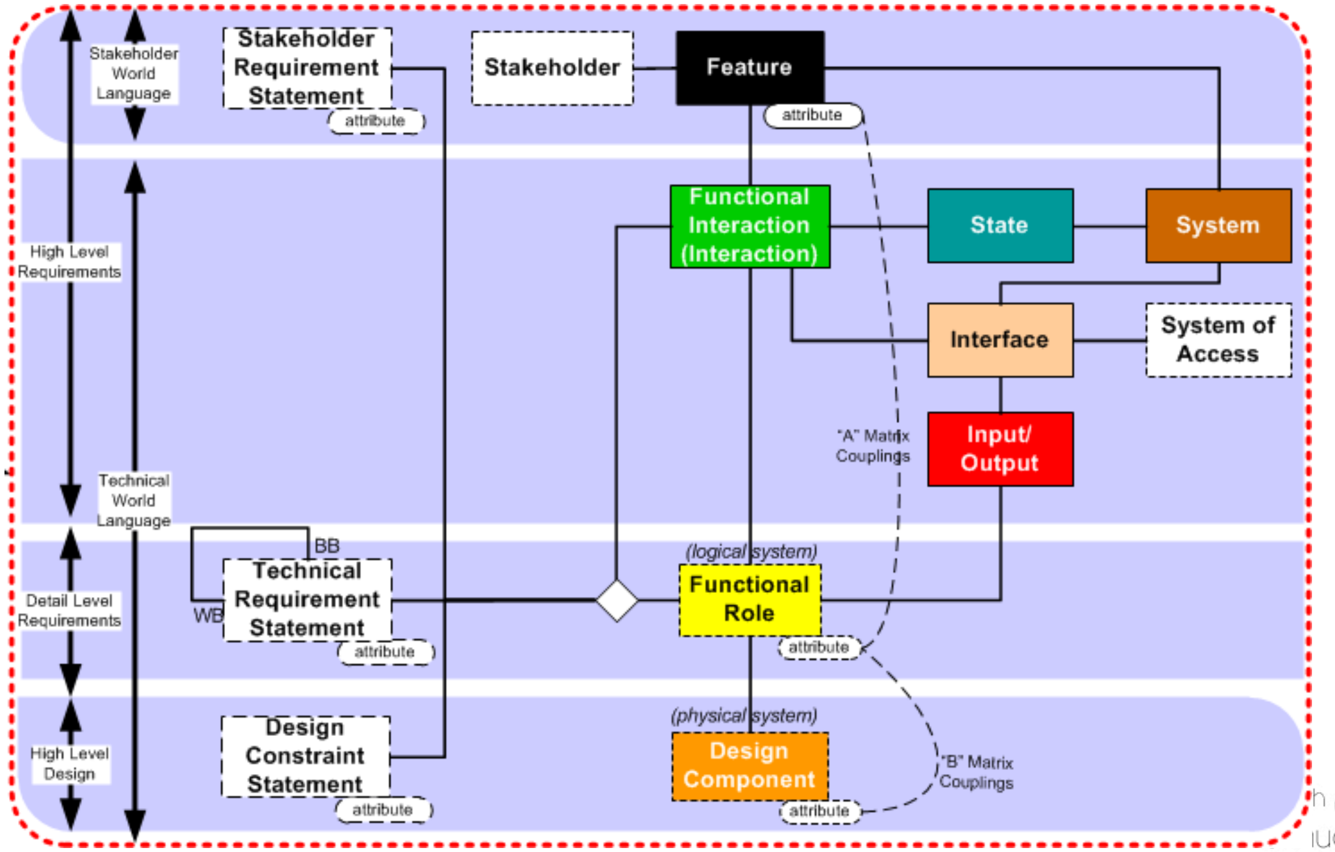
- **An S* Model** is a description of all those important things, and the relationships between them.
- Typically expressed in the "views" of some modeling language (e.g., SysML™).
- The S* Metamodel: The smallest set of information sufficient to describe a system for systems engineering purposes.
- Includes not only the physical Platform information, but all the extended system information (e.g., requirements, risk analysis, design trade-offs & alternatives, decision processes, etc.):

Metamodel for Model-Based Systems Engineering (MBSE)



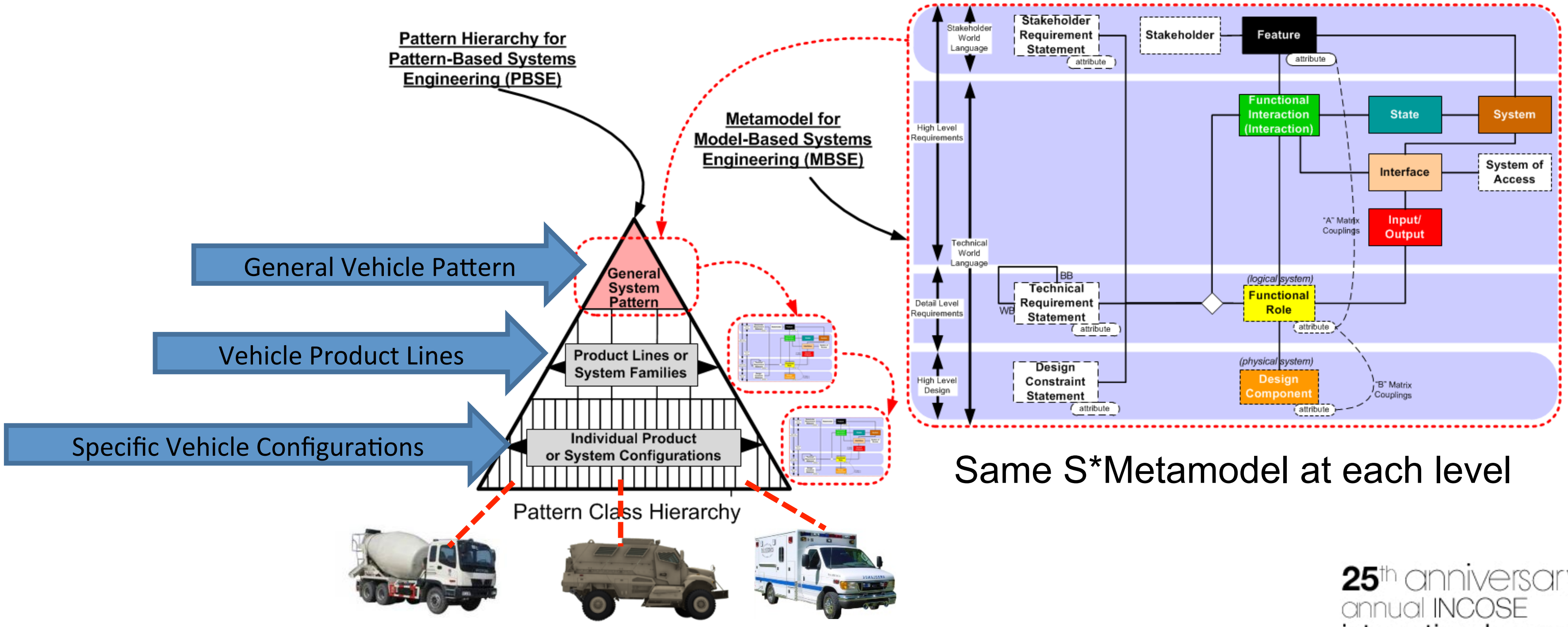
Interactions: The Heart of the S* Metamodel

- The S*Metamodel is focused on the very physical Interactions that are the basis of all the observed laws of the physical sciences, and which we assert are at the heart of the definition of System (a collection of interacting components).



The S* Pattern

- An **S* Pattern** is a configurable, re-usable S* Model. It is an extension of the idea of a Platform (which is a configurable, re-usable design) or Enterprise / Industry Framework.
- The Pattern includes not only the physical Platform information, but all the extended system information (e.g., pattern configuration rules, requirements, risk analysis, design trade-offs & alternatives, decision processes, etc.):

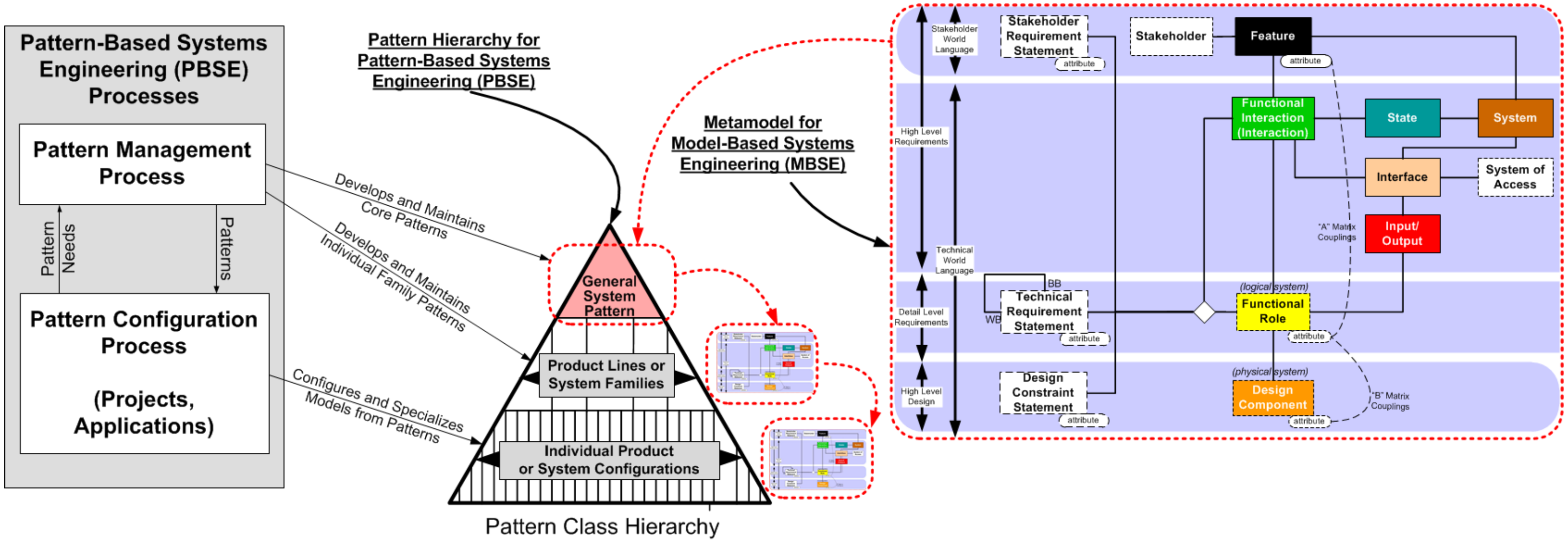


Pattern Hierarchy for Pattern-Based Systems Engineering (PBSE)
Metamodel for Model-Based Systems Engineering (MBSE)
General Vehicle Pattern
Vehicle Product Lines
Specific Vehicle Configurations
General System Pattern
Product Lines or System Families
Individual Product or System Configurations
Pattern Class Hierarchy
Stakeholder World Language
Stakeholder Requirement Statement
attribute
Stakeholder
Feature
High Level Requirements
Functional Interaction (Interaction)
State
System
Interface
System of Access
"A" Matrix Couplings
Input/ Output
Technical World Language
BB
WB
Technical Requirement Statement
(logical system)
Functional Role
Detail Level Requirements
High Level Design
Design Constraint Statement
(physical system)
Design Component
"B" Matrix Couplings

Same S*Metamodel at each level

PBSE S* Metamodel

- Fundamental to PBSE is the use of the S*Metamodel, a relational / object information model intended to describe the “smallest possible model” necessary for the purposes of performing systems engineering or science.
- It provides the semantics to describe requirements, designs, and other information such as verification, failure analysis, etc.
- Specifically, an S*Pattern is a re-usable, configurable S*Model of a family of systems (product line, set, ensemble etc.)


Pattern-Based Systems Engineering (PBSE) Processes
Pattern Management Process
Pattern Needs
Patterns
Pattern Configuration Process
(Projects, Applications)
Pattern Hierarchy for Pattern-Based Systems Engineering (PBSE)
Develops and Maintains Core Patterns
Develops and Maintains Individual Family Patterns
Configures and Specializes Models from Patterns
General System Pattern
Product Lines or System Families
Individual Product or System Configurations
Pattern Class Hierarchy
Metamodel for Model-Based Systems Engineering (MBSE)
Stakeholder World Language
High Level Requirements
Technical World Language
Detail Level Requirements
High Level Design
Stakeholder Requirement Statement
attribute
Stakeholder
Feature
Functional Interaction (Interaction)
State
System
Interface
System of Access
"A" Matrix Couplings
Input/ Output
BB
WB
Technical Requirement Statement
(logical system)
Functional Role
(physical system)
Design Constraint Statement
Design Component
"B" Matrix Couplings

A summary view of the S* metamodel and Pattern Hierarchy and Process

Definitions of some S* Metamodel Classes

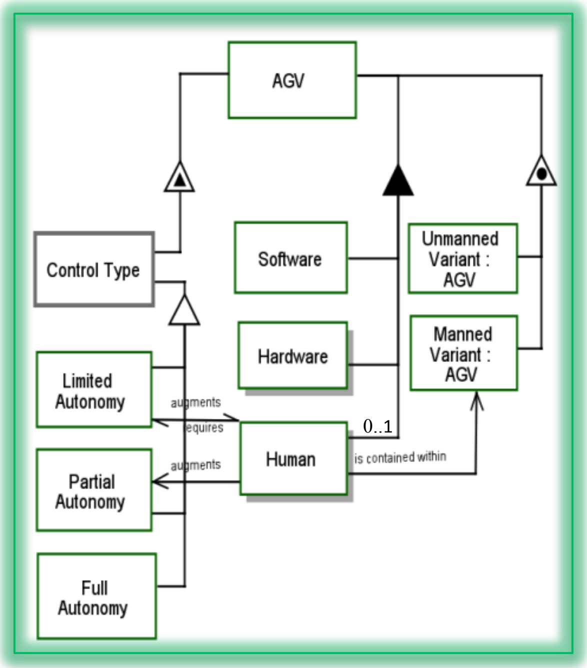
- **System**: A collection of interacting components. Example: Vehicle; Vehicle Domain System.
- **Stakeholder**: A person or other entity with something at stake in the life cycle of a system. Example: Vehicle Operator; Vehicle Owner; Pedestrian
- **Feature**: A behavior of a system that carries stakeholder value. Example: Automatic Braking System Feature; Passenger Comfort Feature Group
- **Functional Interaction (Interaction):** An exchange of energy, force, mass, or information by two entities, in which one changes the state of the other. Example: Refuel Vehicle; Travel Over Terrain
- **Functional Role (Role):** The behavior performed by one of the interacting entities during an Interaction. Example: Vehicle Operator; Vehicle Passenger Environment Subsystem
- **Input-Output:** That which is exchanged during an interaction (generally associated with energy, force, mass, or information). Example: Fuel, Propulsion Force, Exhaust Gas

Definitions of some S* Metamodel Classes

- **System of Access:** A system which provides the means for physical interaction between two interacting entities. Examples: Fueling Nozzle-Receptacle; Grease Gun Fitting; Steering Wheel; Dashboard; Brake Peddle
- **Interface:** The association of a System (which "has" the interface), one or more Interactions (which describe behavior at the interface), the Input-Outputs (which pass through the interface), and a System of Access (which provides the means of the interaction). Examples: Operator Interface; GPS Interface
- **State:** A mode, situation, or condition that describes a System's condition at some moment or period of time. Example: Starting; Cruising; Performing Maneuvers
- **Design Component:** A physical entity that has identity, whose behavior is described by Functional Role(s) allocated to it. Examples: Garmin Model 332 GPS Receiver; Michelin Model 155 Tire
- **Requirement Statement:** A (usually prose) description of the behavior expected of (at least part of) a Functional Role. Example: "The System will accept inflow of fuel at up to 10 gallons per minute without overflow or spillage."

AGV Diversity and Complexity

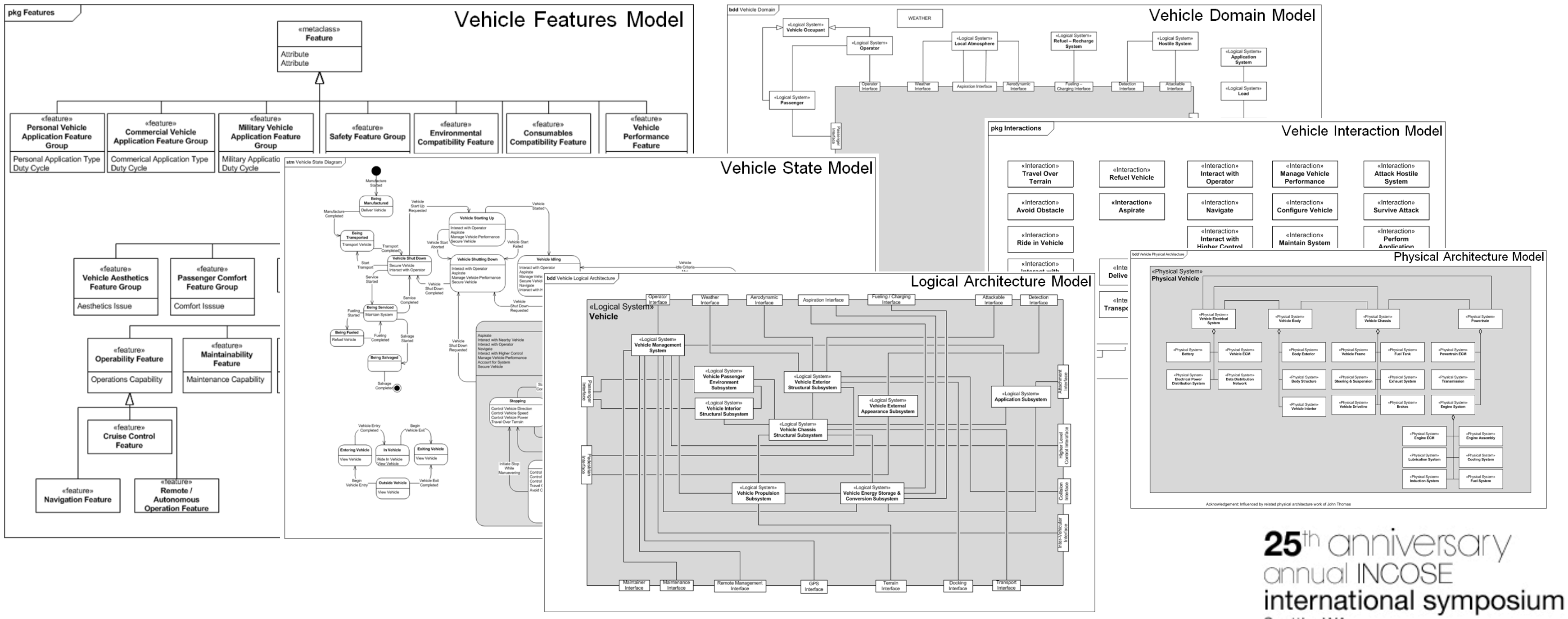
- Autonomous ground vehicles are often broadly categorized as manned and unmanned and by their level or type of control (limited, partial or full)
- These variations when combined with scale and primary function difference make the broad class of automated ground vehicles especially diverse
- Furthermore, AGV's can have algorithms and sensors on-board, off-board, at a central station, embedded within local infrastructure or otherwise



AGV exhibits Control Type.
AGV consists of Software, Hardware, and 0 to 0 Humans.
Hardware is physical.
Human is physical.
Human is contained within Manned Variant.
Human augments Partial Autonomy.
Human augments Limited Autonomy.
Full Autonomy is a Control Type.
Partial Autonomy is a Control Type.
Limited Autonomy is a Control Type.
Limited Autonomy requires Human.
Unmanned Variant is instance of an AGV.
Manned Variant is instance of an AGV.

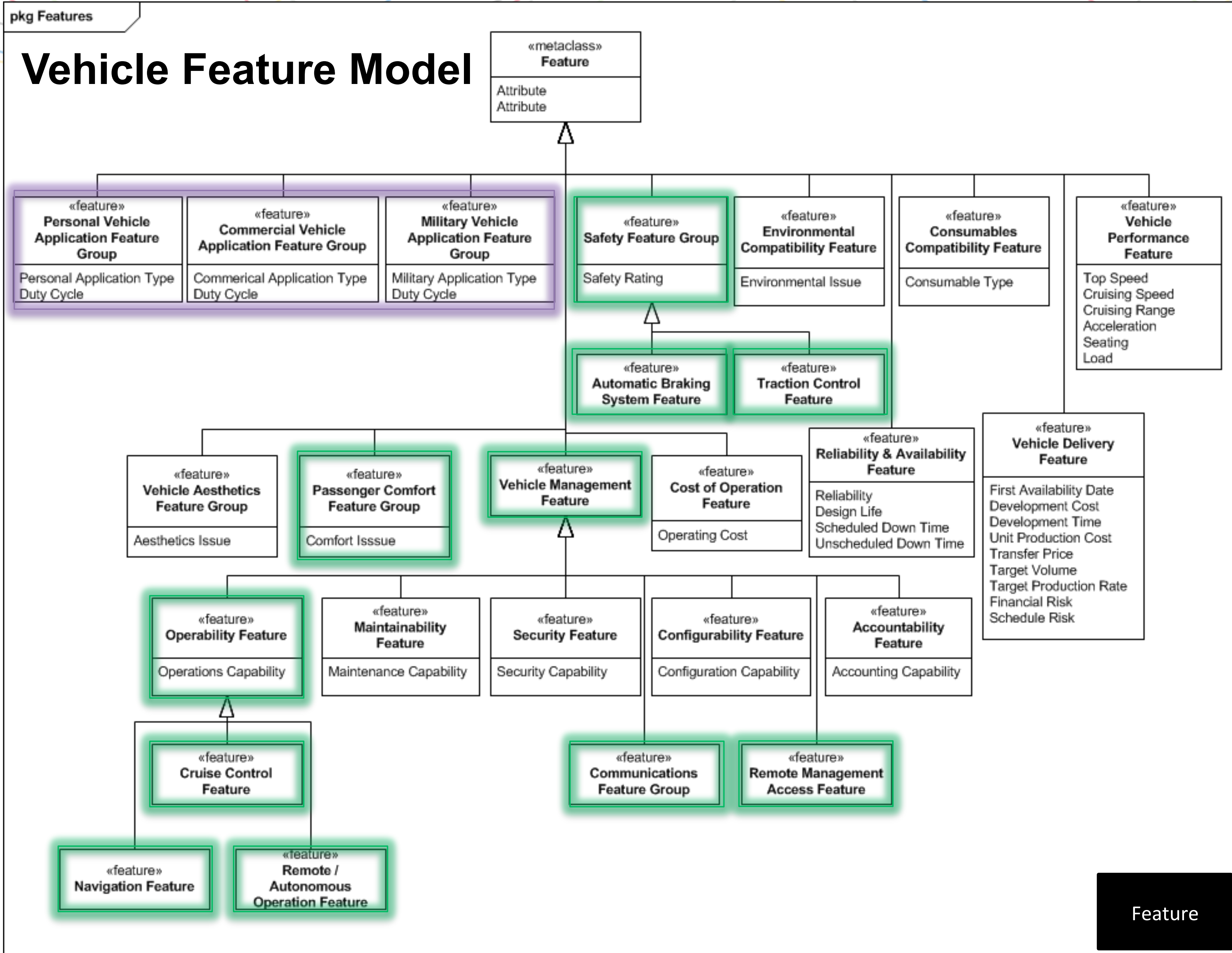
Vehicle Pattern

- Given the variety of applications, the variability in the high level design and implementation of such systems is not surprising. The general vehicle pattern could represent any one of these diverse sets of platforms.
- The amount of information and configurability of this model through the application of PBSE permits rapid specialization/configuration through selection of required feature sets



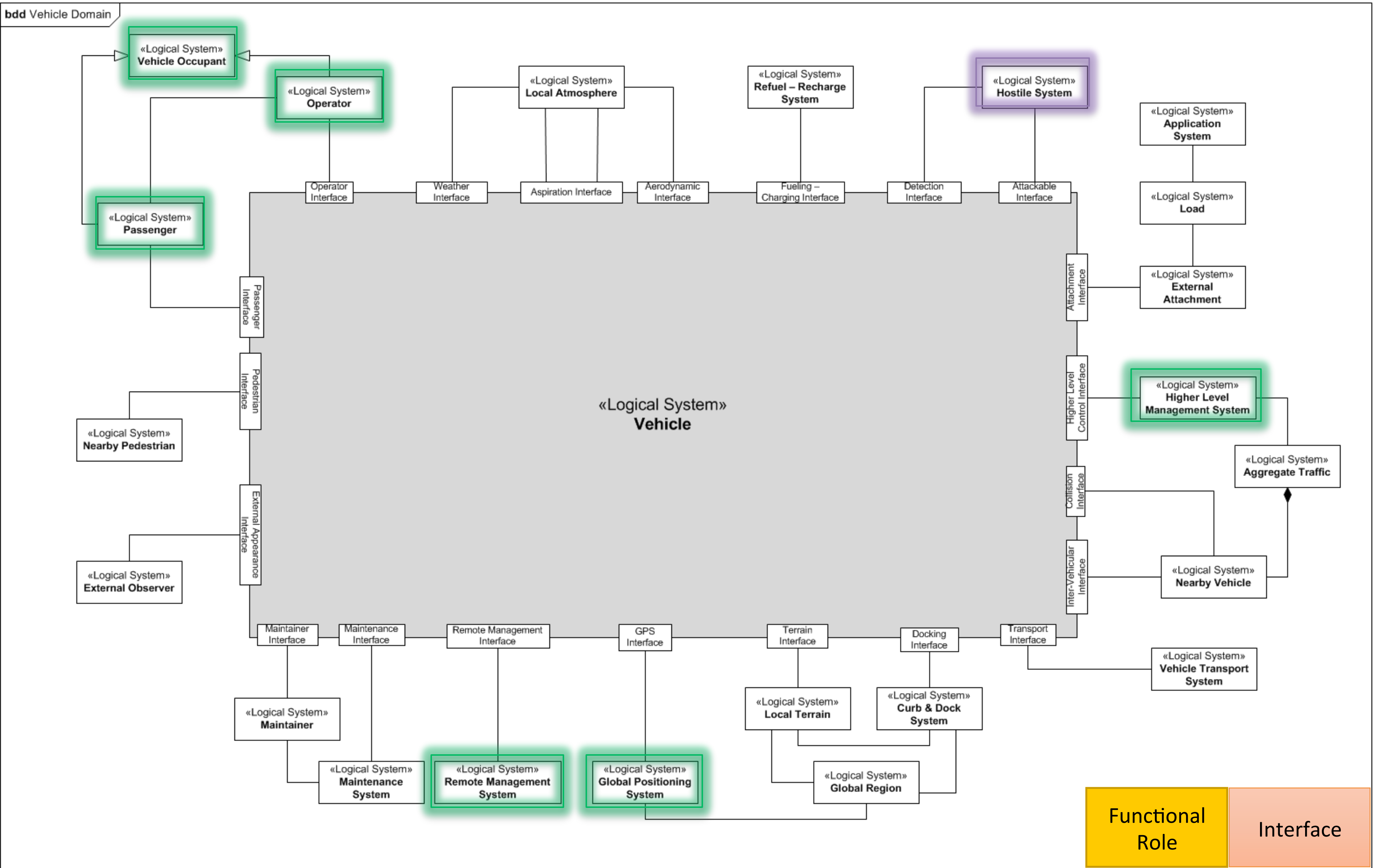
Vehicle Features Model
Vehicle Domain Model
Vehicle Interaction Model
Vehicle State Model
Logical Architecture Model
Physical Architecture Model

Vehicle Feature Model

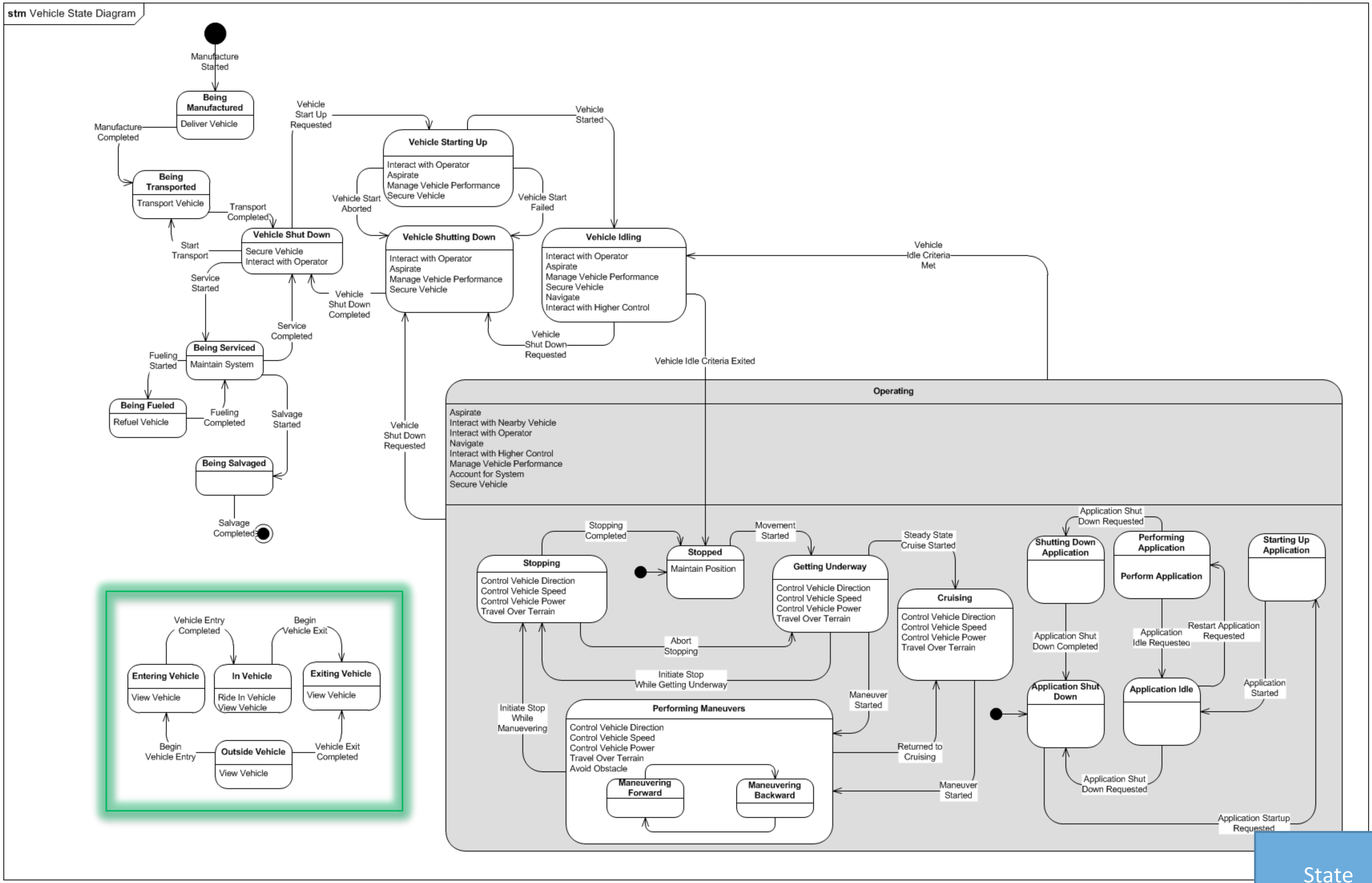

pkg Features
«metaclass» Feature
Attribute
Attribute
«feature» Personal Vehicle Application Feature Group
Personal Application Type
Duty Cycle
«feature» Commercial Vehicle Application Feature Group
Commerical Application Type
Duty Cycle
«feature» Military Vehicle Application Feature Group
Military Application Type
Duty Cycle
«feature» Safety Feature Group
Safety Rating
«feature» Environmental Compatibility Feature
Environmental Issue
«feature» Consumables Compatibility Feature
Consumable Type
«feature» Vehicle Performance Feature
Top Speed
Cruising Speed
Cruising Range
Acceleration
Seating
Load
«feature» Automatic Braking System Feature
«feature» Traction Control Feature
«feature» Reliability & Availability Feature
Reliability
Design Life
Scheduled Down Time
Unscheduled Down Time
«feature» Vehicle Delivery Feature
First Availability Date
Development Cost
Development Time
Unit Production Cost
Transfer Price
Target Volume
Target Production Rate
Financial Risk
Schedule Risk
«feature» Vehicle Aesthetics Feature Group
Aesthetics Issue
«feature» Passenger Comfort Feature Group
Comfort Isssue
«feature» Vehicle Management Feature
«feature» Cost of Operation Feature
Operating Cost
«feature» Operability Feature
Operations Capability
«feature» Maintainability Feature
Maintenance Capability
«feature» Security Feature
Security Capability
«feature» Configurability Feature
Configuration Capability
«feature» Accountability Feature
Accounting Capability
«feature» Cruise Control Feature
«feature» Communications Feature Group
«feature» Remote Management Access Feature
«feature» Navigation Feature
«feature» Remote / Autonomous Operation Feature

Feature

Vehicle Domain Model

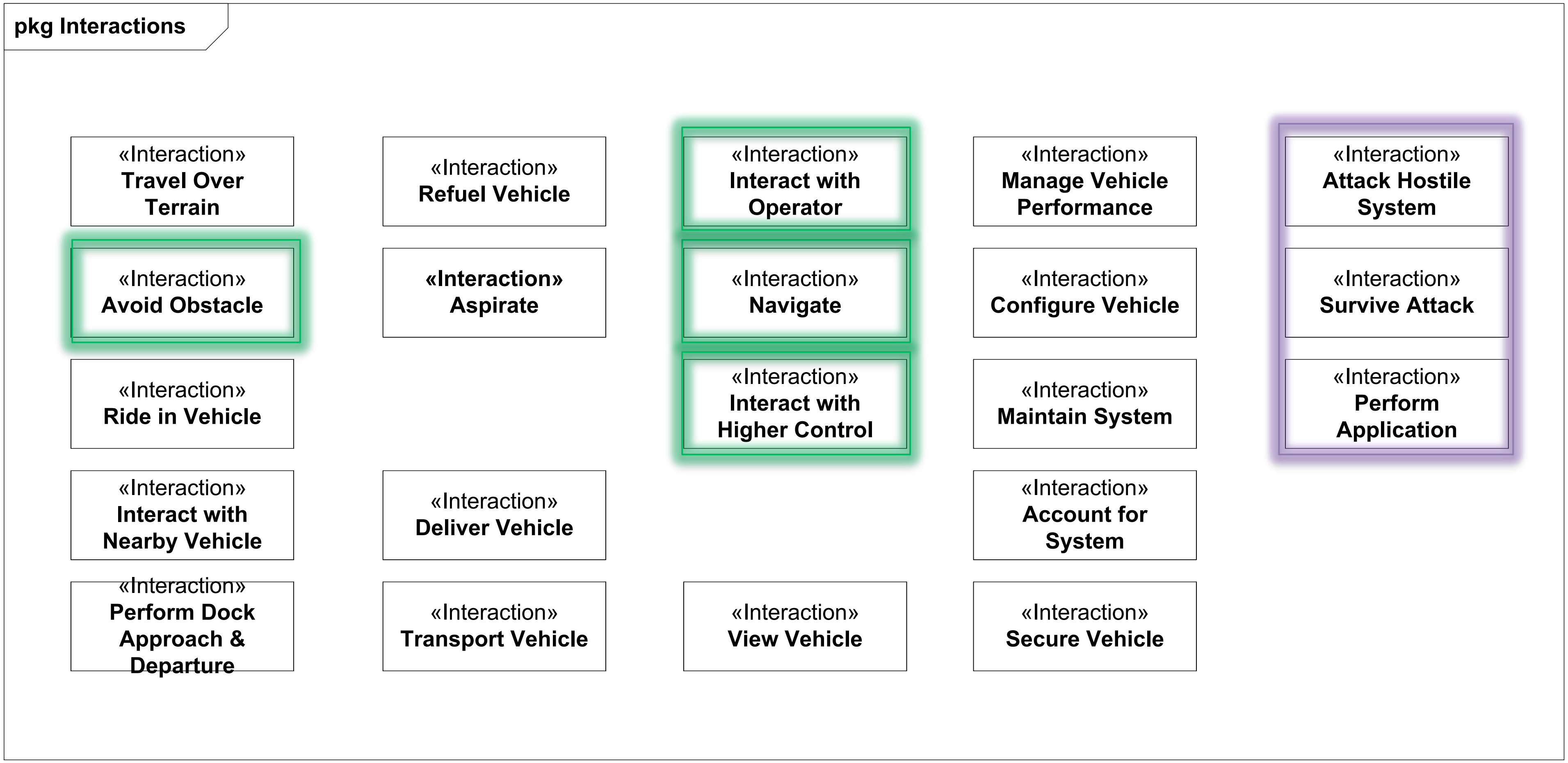

bdd Vehicle Domain
«Logical System» Vehicle Occupant
«Logical System» Operator
«Logical System» Passenger
«Logical System» Local Atmosphere
«Logical System» Refuel – Recharge System
«Logical System» Hostile System
«Logical System» Application System
«Logical System» Load
«Logical System» External Attachment
«Logical System» Higher Level Management System
«Logical System» Aggregate Traffic
«Logical System» Nearby Vehicle
«Logical System» Vehicle Transport System
«Logical System» Nearby Pedestrian
«Logical System» External Observer
«Logical System» Maintainer
«Logical System» Maintenance System
«Logical System» Remote Management System
«Logical System» Global Positioning System
«Logical System» Local Terrain
«Logical System» Curb & Dock System
«Logical System» Global Region
«Logical System» Vehicle
Operator Interface
Weather Interface
Aspiration Interface
Aerodynamic Interface
Fueling – Charging Interface
Detection Interface
Attackable Interface
Attachment Interface
Higher Level Control Interface
Collision Interface
Inter-Vehicular Interface
Passenger Interface
Pedestrian Interface
External Appearance Interface
Maintainer Interface
Maintenance Interface
Remote Management Interface
GPS Interface
Terrain Interface
Docking Interface
Transport Interface
Functional Role
Interface

Vehicle State Model



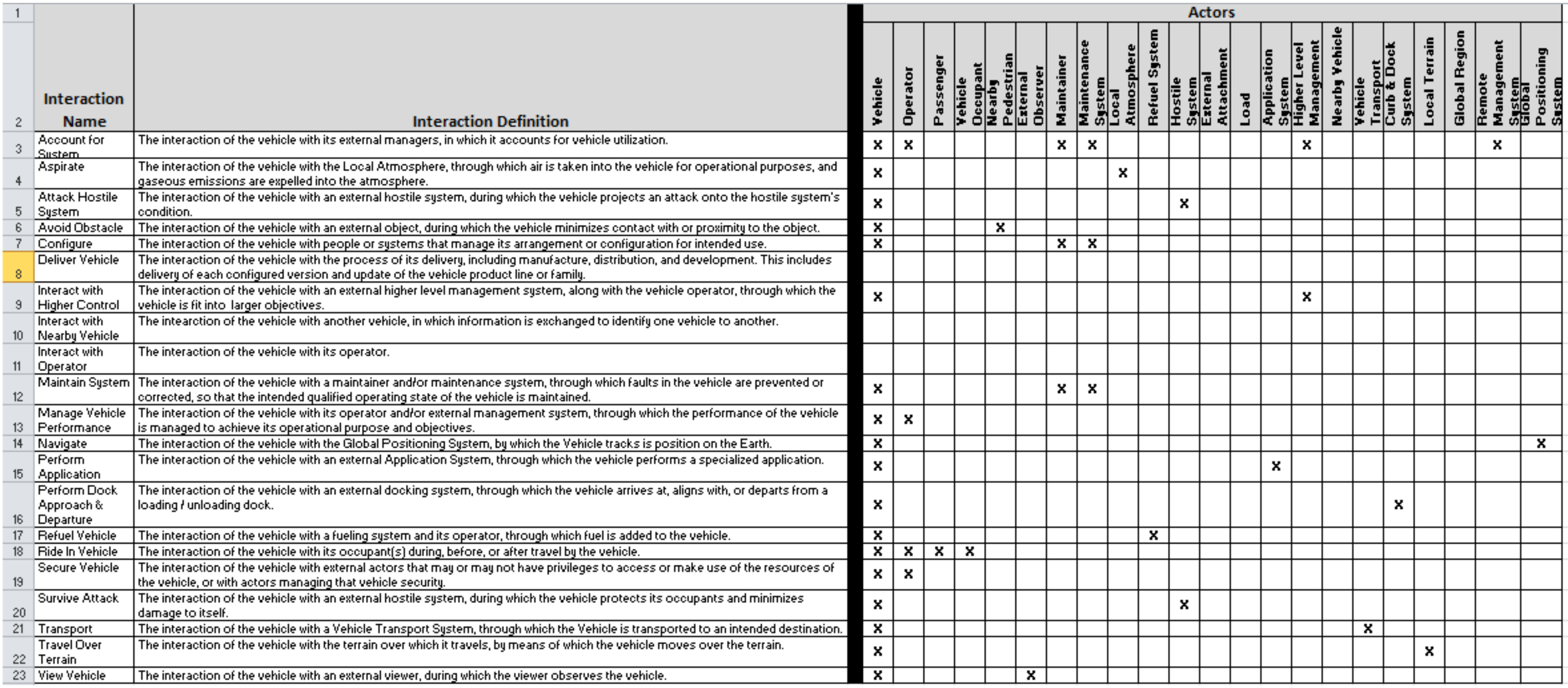
State

Vehicle Interaction Model


pkg Interactions
«Interaction» Travel Over Terrain
«Interaction» Avoid Obstacle
«Interaction» Ride in Vehicle
«Interaction» Interact with Nearby Vehicle
«Interaction» Perform Dock Approach & Departure
«Interaction» Refuel Vehicle
«Interaction» Aspirate
«Interaction» Deliver Vehicle
«Interaction» Transport Vehicle
«Interaction» Interact with Operator
«Interaction» Navigate
«Interaction» Interact with Higher Control
«Interaction» View Vehicle
«Interaction» Manage Vehicle Performance
«Interaction» Configure Vehicle
«Interaction» Maintain System
«Interaction» Account for System
«Interaction» Secure Vehicle
«Interaction» Attack Hostile System
«Interaction» Survive Attack
«Interaction» Perform Application

Functional Interaction

Vehicle Interactions Matrix

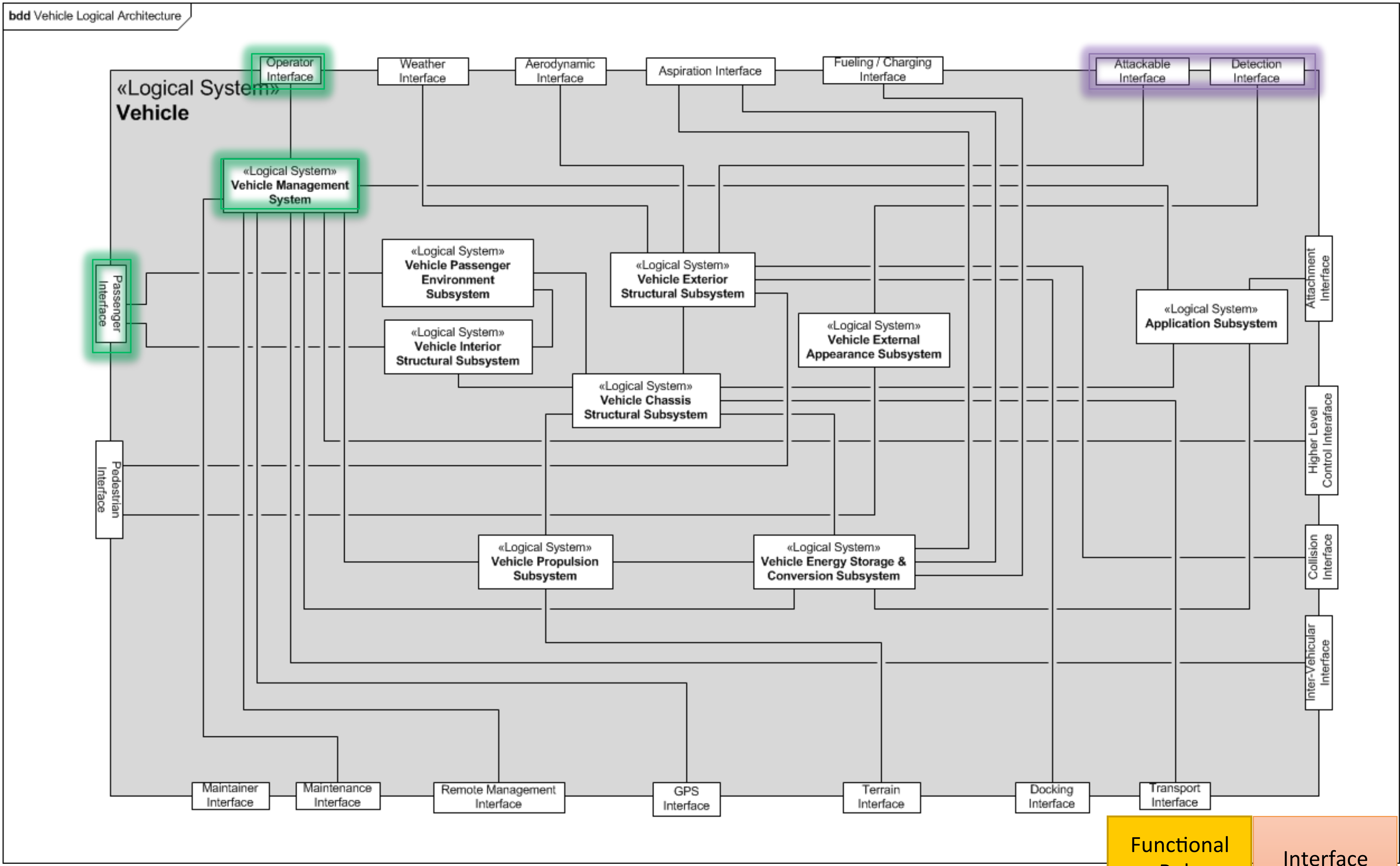


1			Actors																					
2	Interaction Name	Interaction Definition	Vehicle	Operator	Passenger	Vehicle Occupant	Nearby Pedestrian	External Observer	Maintainer	Maintenance System	Local Atmosphere	Refuel System	Hostile System	External Attachment	Load	Application System	Higher Level Management	Nearby Vehicle	Vehicle Transport	Curb & Dock System	Local Terrain	Global Region	Remote Management System	Global Positioning System
3	Account for System	The interaction of the vehicle with its external managers, in which it accounts for vehicle utilization.	X	X					X	X							X						X	
4	Aspirate	The interaction of the vehicle with the Local Atmosphere, through which air is taken into the vehicle for operational purposes, and gaseous emissions are expelled into the atmosphere.	X								X													
5	Attack Hostile System	The interaction of the vehicle with an external hostile system, during which the vehicle projects an attack onto the hostile system's condition.	X										X											
6	Avoid Obstacle	The interaction of the vehicle with an external object, during which the vehicle minimizes contact with or proximity to the object.	X				X																	
7	Configure	The interaction of the vehicle with people or systems that manage its arrangement or configuration for intended use.	X						X	X														
8	Deliver Vehicle	The interaction of the vehicle with the process of its delivery, including manufacture, distribution, and development. This includes delivery of each configured version and update of the vehicle product line or family.																						
9	Interact with Higher Control	The interaction of the vehicle with an external higher level management system, along with the vehicle operator, through which the vehicle is fit into larger objectives.	X														X							
10	Interact with Nearby Vehicle	The intearction of the vehicle with another vehicle, in which information is exchanged to identify one vehicle to another.																						
11	Interact with Operator	The interaction of the vehicle with its operator.																						
12	Maintain System	The interaction of the vehicle with a maintainer and/or maintenance system, through which faults in the vehicle are prevented or corrected, so that the intended qualified operating state of the vehicle is maintained.	X						X	X														
13	Manage Vehicle Performance	The interaction of the vehicle with its operator and/or external management system, through which the performance of the vehicle is managed to achieve its operational purpose and objectives.	X	X																				
14	Navigate	The interaction of the vehicle with the Global Positioning System, by which the Vehicle tracks is position on the Earth.	X																					X
15	Perform Application	The interaction of the vehicle with an external Application System, through which the vehicle performs a specialized application.	X													X								
16	Perform Dock Approach & Departure	The interaction of the vehicle with an external docking system, through which the vehicle arrives at, aligns with, or departs from a loading / unloading dock.	X																	X				
17	Refuel Vehicle	The interaction of the vehicle with a fueling system and its operator, through which fuel is added to the vehicle.	X									X												
18	Ride In Vehicle	The interaction of the vehicle with its occupant(s) during, before, or after travel by the vehicle.	X	X	X	X																		
19	Secure Vehicle	The interaction of the vehicle with external actors that may or may not have privileges to access or make use of the resources of the vehicle, or with actors managing that vehicle security.	X	X																				
20	Survive Attack	The interaction of the vehicle with an external hostile system, during which the vehicle protects its occupants and minimizes damage to itself.	X										X											
21	Transport	The interaction of the vehicle with a Vehicle Transport System, through which the Vehicle is transported to an intended destination.	X																X					
22	Travel Over Terrain	The interaction of the vehicle with the terrain over which it travels, by means of which the vehicle moves over the terrain.	X																		X			
23	View Vehicle	The interaction of the vehicle with an external viewer, during which the viewer observes the vehicle.	X					X																

Functional Role

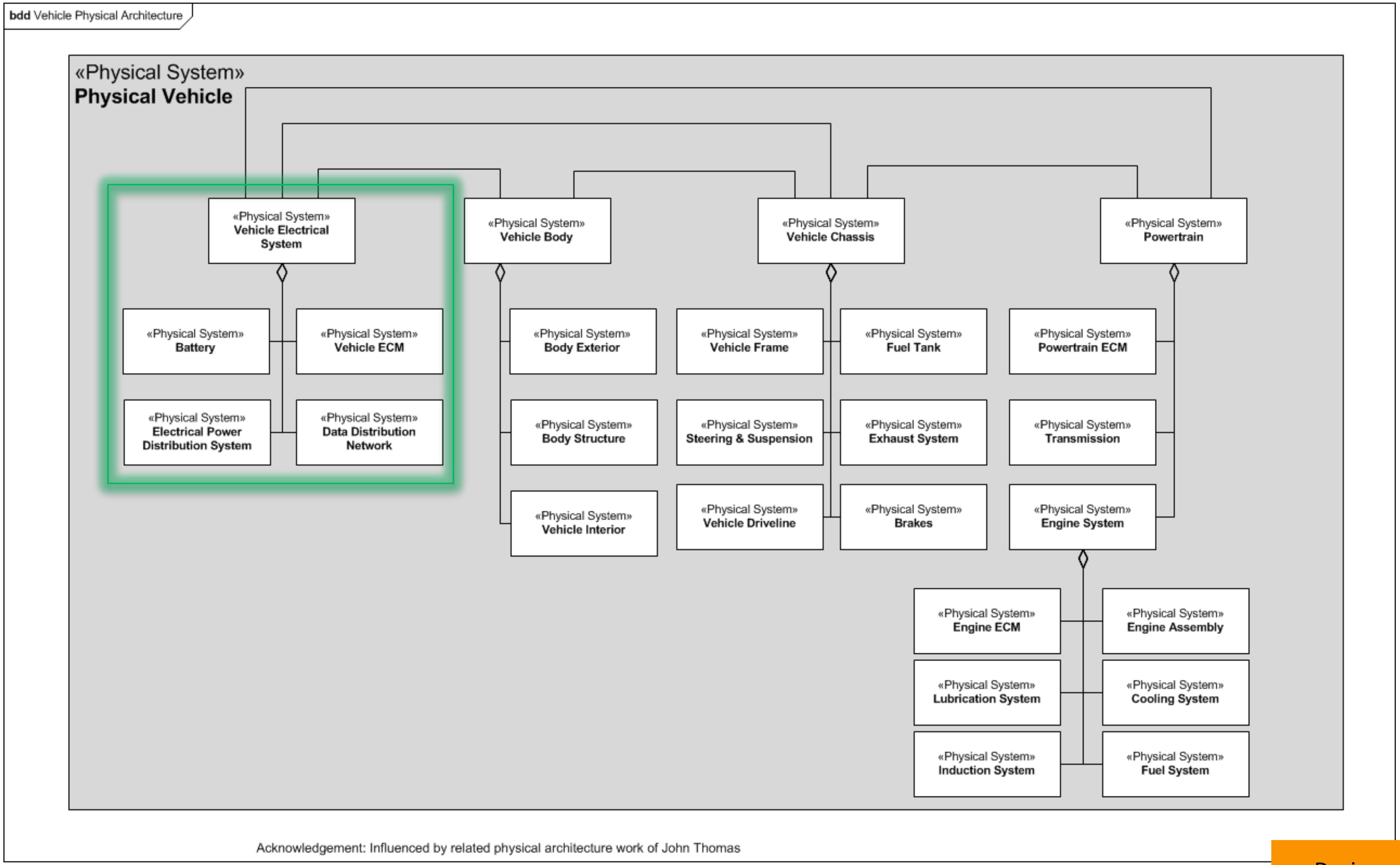
Functional Interaction

Vehicle Logical Architecture Model


bdd Vehicle Logical Architecture
«Logical System» Vehicle
Operator Interface
Weather Interface
Aerodynamic Interface
Aspiration Interface
Fueling / Charging Interface
Attackable Interface
Detection Interface
«Logical System» Vehicle Management System
Passenger Interface
«Logical System» Vehicle Passenger Environment Subsystem
«Logical System» Vehicle Interior Structural Subsystem
«Logical System» Vehicle Exterior Structural Subsystem
«Logical System» Vehicle External Appearance Subsystem
«Logical System» Application Subsystem
Attachment Interface
«Logical System» Vehicle Chassis Structural Subsystem
Higher Level Control Interaface
Pedestrian Interface
«Logical System» Vehicle Propulsion Subsystem
«Logical System» Vehicle Energy Storage & Conversion Subsystem
Collision Interface
Inter-Vehicular Interface
Maintainer Interface
Maintenance Interface
Remote Management Interface
GPS Interface
Terrain Interface
Docking Interface
Transport Interface

Functional Role	Interface

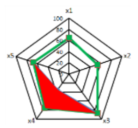
Vehicle Physical Architecture Model


bdd Vehicle Physical Architecture
«Physical System» Physical Vehicle
«Physical System» Vehicle Electrical System
«Physical System» Vehicle Body
«Physical System» Vehicle Chassis
«Physical System» Powertrain
«Physical System» Battery
«Physical System» Vehicle ECM
«Physical System» Electrical Power Distribution System
«Physical System» Data Distribution Network
«Physical System» Body Exterior
«Physical System» Body Structure
«Physical System» Vehicle Interior
«Physical System» Vehicle Frame
«Physical System» Steering & Suspension
«Physical System» Vehicle Driveline
«Physical System» Fuel Tank
«Physical System» Exhaust System
«Physical System» Brakes
«Physical System» Powertrain ECM
«Physical System» Transmission
«Physical System» Engine System
«Physical System» Engine ECM
«Physical System» Engine Assembly
«Physical System» Lubrication System
«Physical System» Cooling System
«Physical System» Induction System
«Physical System» Fuel System
Acknowledgement: Influenced by related physical architecture work of John Thomas

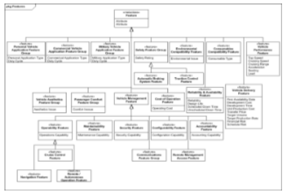
Design Component

Vehicle Multi Domain Matrix

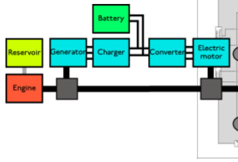
Features

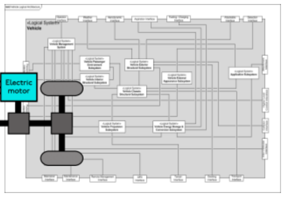


Requirement X

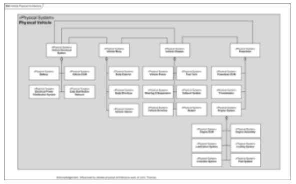


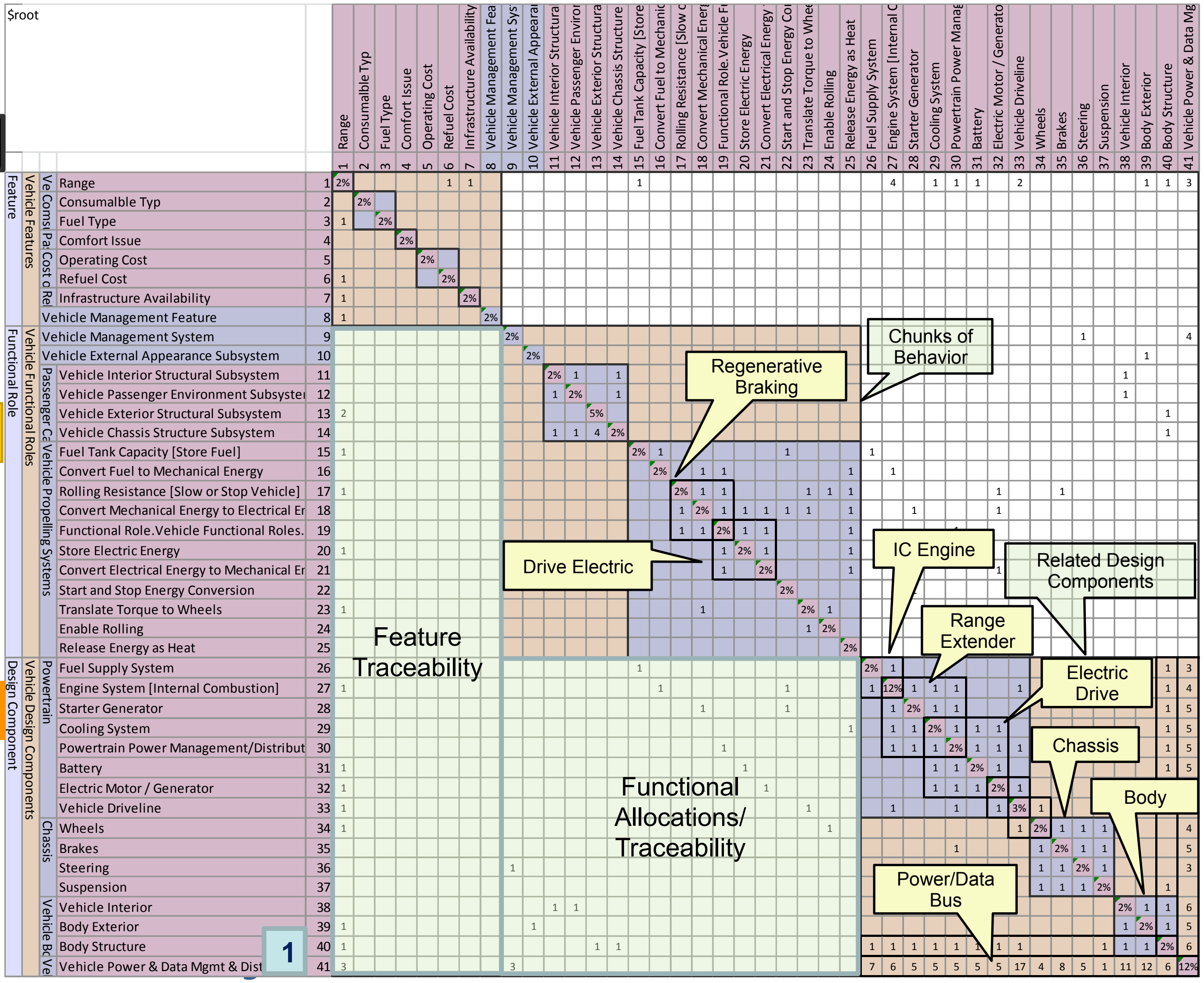
Functional Role





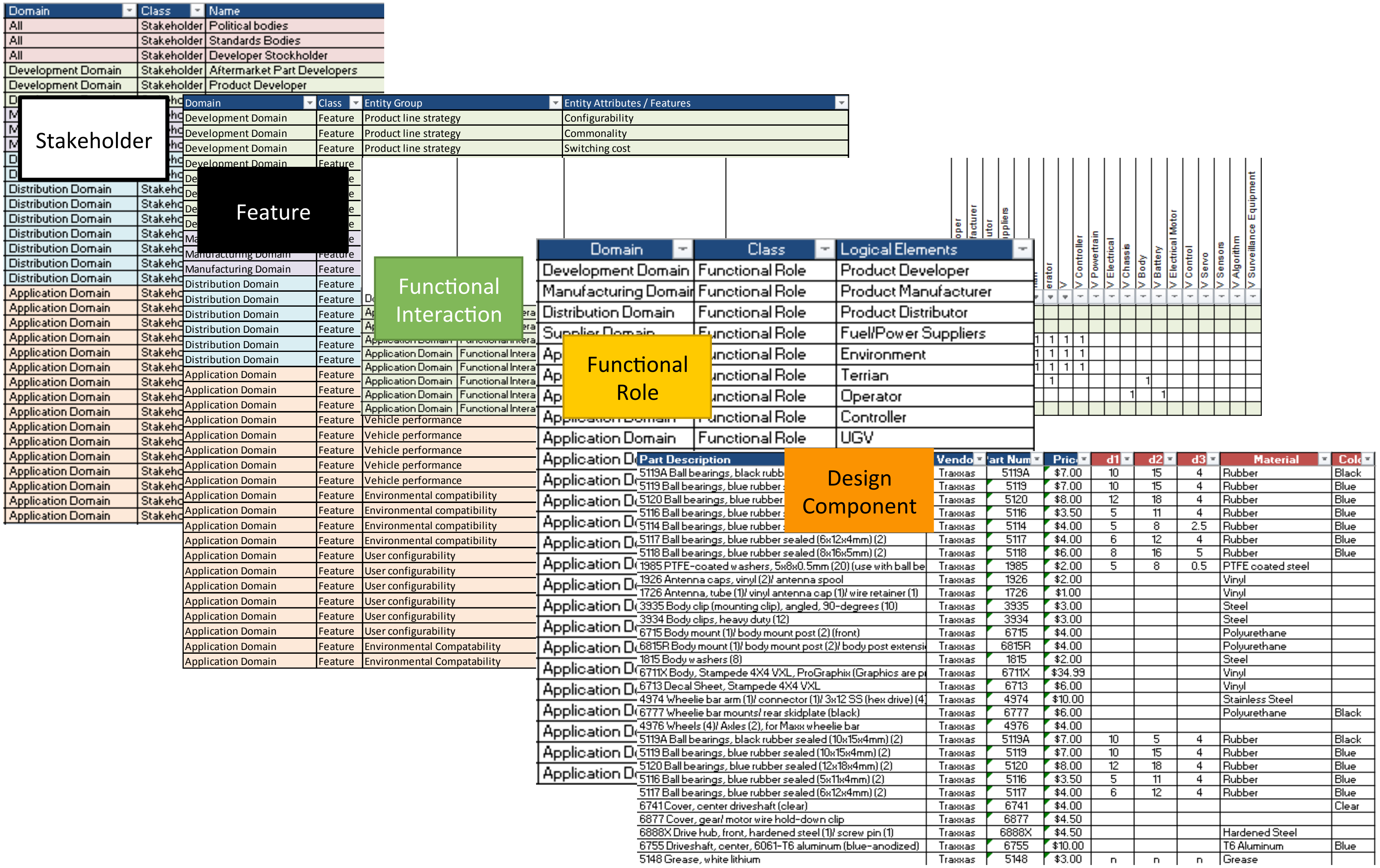
Design Component




$root
Feature
Vehicle Features
Ve
Consu
Pa
Cost o
Re
Functional Role
Vehicle Functional Roles
Passenger Co
Vehicle Propelling Systems
Design Component
Vehicle Design Components
Powertrain
Chassis
Vehicle Bo
Ve
1
Feature Traceability
Regenerative Braking
Chunks of Behavior
Drive Electric
IC Engine
Related Design Components
Range Extender
Electric Drive
Chassis
Body
Power/Data Bus
Functional Allocations/ Traceability

		1 Range	2 Consumable Typ	3 Fuel Type	4 Comfort Issue	5 Operating Cost	6 Refuel Cost	7 Infrastructure Availability	8 Vehicle Management Fea	9 Vehicle Management Sys	10 Vehicle External Appeara	11 Vehicle Interior Structura	12 Vehicle Passenger Enviro	13 Vehicle Exterior Structura	14 Vehicle Chassis Structure	15 Fuel Tank Capacity [Store	16 Convert Fuel to Mechanic	17 Rolling Resistance [Slow c	18 Convert Mechanical Ener	19 Functional Role.Vehicle Fu	20 Store Electric Energy	21 Convert Electrical Energy	22 Start and Stop Energy Co	23 Translate Torque to Whe	24 Enable Rolling	25 Release Energy as Heat	26 Fuel Supply System	27 Engine System [Internal C	28 Starter Generator	29 Cooling System	30 Powertrain Power Manag	31 Battery	32 Electric Motor / Generato	33 Vehicle Driveline	34 Wheels	35 Brakes	36 Steering	37 Suspension	38 Vehicle Interior	39 Body Exterior	40 Body Structure	41 Vehicle Power & Data Mg
Range	1	2%					1	1								1												4		1	1	1		2						1	1	3
Consumable Typ	2		2%																																							
Fuel Type	3	1		2%																																						
Comfort Issue	4				2%																																					
Operating Cost	5					2%																																				
Refuel Cost	6	1					2%																																			
Infrastructure Availability	7	1						2%																																		
Vehicle Management Feature	8	1							2%																																	
Vehicle Management System	9									2%																											1					4
Vehicle External Appearance Subsystem	10										2%																													1		
Vehicle Interior Structural Subsystem	11											2%	1		1																								1			
Vehicle Passenger Environment Subsyste	12											1	2%		1																								1			
Vehicle Exterior Structural Subsystem	13	2												5%																											1	
Vehicle Chassis Structure Subsystem	14											1	1	4	2%																										1	
Fuel Tank Capacity [Store Fuel]	15	1														2%	1						1				1															
Convert Fuel to Mechanical Energy	16																2%		1	1						1		1														
Rolling Resistance [Slow or Stop Vehicle]	17	1																2%	1	1				1	1	1							1			1						
Convert Mechanical Energy to Electrical Er	18																	1	2%	1	1	1	1	1		1			1				1									
Functional Role.Vehicle Functional Roles.	19																	1	1	2%	1	1				1					1											
Store Electric Energy	20	1																		1	2%	1				1																
Convert Electrical Energy to Mechanical Er	21																			1		2%				1							1									
Start and Stop Energy Conversion	22																						2%						1													
Translate Torque to Wheels	23	1																	1					2%	1																	
Enable Rolling	24																							1	2%																	
Release Energy as Heat	25																									2%																
Fuel Supply System	26															1											2%	1													1	3
Engine System [Internal Combustion]	27	1															1						1				1	12%	1	1	1			1							1	4
Starter Generator	28																		1				1					1	2%	1	1										1	5
Cooling System	29																									1		1	1	2%	1	1	1								1	5
Powertrain Power Management/Distribut	30																			1								1	1	1	2%	1	1	1							1	5
Battery	31	1																			1									1	1	2%	1								1	5
Electric Motor / Generator	32	1																				1								1	1	1	2%	1								
Vehicle Driveline	33	1																						1				1			1		1	3%	1							
Wheels	34	1																							1									1	2%	1	1	1				4
Brakes	35																														1				1	2%	1	1				5
Steering	36									1																									1	1	2%	1				3
Suspension	37																																		1	1	1	2%			1	
Vehicle Interior	38											1	1																										2%	1	1	6
Body Exterior	39	1									1																												1	2%	1	5
Body Structure	40	1												1	1												1	1	1	1	1	1	1	1				1	1	1	2%	6
Vehicle Power & Data Mgmt & Dist	41	3								3																	7	6	5	5	5	5	5	17	4	8	5	1	11	12	6	12%

Patterns Challenge Team AGV content



Domain	Class	Name
All	Stakeholder	Political bodies
All	Stakeholder	Standards Bodies
All	Stakeholder	Developer Stockholder
Development Domain	Stakeholder	Aftermarket Part Developers
Development Domain	Stakeholder	Product Developer
D	hc	
M	hc	
M	hc	
M	hc	
D	hc	
D	hc	
Distribution Domain	Stakehc	
Distribution Domain	Stakehc	
Distribution Domain	Stakehc	
Distribution Domain	Stakehc	
Distribution Domain	Stakehc	
Distribution Domain	Stakehc	
Distribution Domain	Stakehc	
Application Domain	Stakehc	
Application Domain	Stakehc	
Application Domain	Stakehc	
Application Domain	Stakehc	
Application Domain	Stakehc	
Application Domain	Stakehc	
Application Domain	Stakehc	
Application Domain	Stakehc	
Application Domain	Stakehc	
Application Domain	Stakehc	
Application Domain	Stakehc	
Application Domain	Stakehc	
Application Domain	Stakehc	
Application Domain	Stakehc	
Application Domain	Stakehc	
Application Domain	Stakehc	

Domain	Class	Entity Group	Entity Attributes / Features
Development Domain	Feature	Product line strategy	Configurability
Development Domain	Feature	Product line strategy	Commonality
Development Domain	Feature	Product line strategy	Switching cost
Development Domain	Feature		
De	e		
De	e		
De	e		
De	e		
Ma	e		
Manufacturing Domain	Feature		
Manufacturing Domain	Feature		
Distribution Domain	Feature		
Distribution Domain	Feature		
Distribution Domain	Feature		
Distribution Domain	Feature		
Distribution Domain	Feature		
Distribution Domain	Feature		
Application Domain	Feature		
Application Domain	Feature		
Application Domain	Feature		
Application Domain	Feature	Vehicle performance	
Application Domain	Feature	Vehicle performance	
Application Domain	Feature	Vehicle performance	
Application Domain	Feature	Vehicle performance	
Application Domain	Feature	Vehicle performance	
Application Domain	Feature	Environmental compatibility	
Application Domain	Feature	Environmental compatibility	
Application Domain	Feature	Environmental compatibility	
Application Domain	Feature	Environmental compatibility	
Application Domain	Feature	User configurability	
Application Domain	Feature	User configurability	
Application Domain	Feature	User configurability	
Application Domain	Feature	User configurability	
Application Domain	Feature	User configurability	
Application Domain	Feature	User configurability	
Application Domain	Feature	Environmental Compatability	
Application Domain	Feature	Environmental Compatability	

D	
Ap	era
Ap	era
Application Domain	Functional Intera
Application Domain	Functional Intera
Application Domain	Functional Intera
Application Domain	Functional Intera
Application Domain	Functional Intera
Application Domain	Functional Intera

Domain	Class	Logical Elements
Development Domain	Functional Role	Product Developer
Manufacturing Domai	Functional Role	Product Manufacturer
Distribution Domain	Functional Role	Product Distributor
Supplier Domain	Functional Role	Fuel/Power Suppliers
Ap	nctional Role	Environment
Ap	nctional Role	Terrian
Ap	nctional Role	Operator
Application Domain	Functional Role	Controller
Application Domain	Functional Role	UGV
Application D		
Application D		
Application D		
Application D		
Application D		
Application D		
Application D		
Application D		
Application D		
Application D		
Application D		
Application D		
Application D		
Application D		
Application D		
Application D		

	erator	V	V Controller	V Powertrain	V Electrical	V Chassis	V Body	V Battery	V Electrical Motor	V Control	V Servo	V Sensors	V Algorithm	V Surveillance Equipment
1	1	1	1											
1	1	1	1											
1	1	1	1											
	1						1							
						1		1						

Part Description	Vendo	art Num	Pric	d1	d2	d3	Material	Colc
5119A Ball bearings, black rubb	Traxxas	5119A	$7.00	10	15	4	Rubber	Black
5119 Ball bearings, blue rubber	Traxxas	5119	$7.00	10	15	4	Rubber	Blue
5120 Ball bearings, blue rubber	Traxxas	5120	$8.00	12	18	4	Rubber	Blue
5116 Ball bearings, blue rubber	Traxxas	5116	$3.50	5	11	4	Rubber	Blue
5114 Ball bearings, blue rubber	Traxxas	5114	$4.00	5	8	2.5	Rubber	Blue
5117 Ball bearings, blue rubber sealed (6x12x4mm) (2)	Traxxas	5117	$4.00	6	12	4	Rubber	Blue
5118 Ball bearings, blue rubber sealed (8x16x5mm) (2)	Traxxas	5118	$6.00	8	16	5	Rubber	Blue
1985 PTFE-coated washers, 5x8x0.5mm (20) (use with ball be	Traxxas	1985	$2.00	5	8	0.5	PTFE coated steel	
1926 Antenna caps, vinyl (2)/ antenna spool	Traxxas	1926	$2.00				Vinyl	
1726 Antenna, tube (1)/ vinyl antenna cap (1)/ wire retainer (1)	Traxxas	1726	$1.00				Vinyl	
3935 Body clip (mounting clip), angled, 90-degrees (10)	Traxxas	3935	$3.00				Steel	
3934 Body clips, heavy duty (12)	Traxxas	3934	$3.00				Steel	
6715 Body mount (1)/ body mount post (2) (front)	Traxxas	6715	$4.00				Polyurethane	
6815R Body mount (1)/ body mount post (2)/ body post extensi	Traxxas	6815R	$4.00				Polyurethane	
1815 Body washers (8)	Traxxas	1815	$2.00				Steel	
6711X Body, Stampede 4X4 VXL, ProGraphix (Graphics are pı	Traxxas	6711X	$34.99				Vinyl	
6713 Decal Sheet, Stampede 4X4 VXL	Traxxas	6713	$6.00				Vinyl	
4974 Wheelie bar arm (1)/ connector (1)/ 3x12 SS (hex drive) (4	Traxxas	4974	$10.00				Stainless Steel	
6777 Wheelie bar mounts/ rear skidplate (black)	Traxxas	6777	$6.00				Polyurethane	Black
4976 Wheels (4)/ Axles (2), for Maxx wheelie bar	Traxxas	4976	$4.00					
5119A Ball bearings, black rubber sealed (10x15x4mm) (2)	Traxxas	5119A	$7.00	10	5	4	Rubber	Black
5119 Ball bearings, blue rubber sealed (10x15x4mm) (2)	Traxxas	5119	$7.00	10	15	4	Rubber	Blue
5120 Ball bearings, blue rubber sealed (12x18x4mm) (2)	Traxxas	5120	$8.00	12	18	4	Rubber	Blue
5116 Ball bearings, blue rubber sealed (5x11x4mm) (2)	Traxxas	5116	$3.50	5	11	4	Rubber	Blue
5117 Ball bearings, blue rubber sealed (6x12x4mm) (2)	Traxxas	5117	$4.00	6	12	4	Rubber	Blue
6741 Cover, center driveshaft (clear)	Traxxas	6741	$4.00					Clear
6877 Cover, gear/ motor wire hold-down clip	Traxxas	6877	$4.50					
6888X Drive hub, front, hardened steel (1)/ screw pin (1)	Traxxas	6888X	$4.50				Hardened Steel	
6755 Driveshaft, center, 6061-T6 aluminum (blue-anodized)	Traxxas	6755	$10.00				T6 Aluminum	Blue
5148 Grease, white lithium	Traxxas	5148	$3.00	n	n	n	Grease	

Embedded Intelligence (EI) Pattern

- Many S*Patterns are discovered and expressed through PBSE, but the EI Pattern is of particular importance to the subject of AGVs.
- the EI Pattern returns to the perspective of Norbert Weiner, who first coined the term "cybernetics" to refer to the study of control and communication in both living and human-engineered systems. (Weiner, 1965).
- 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.
- EI Pattern describes the individual elements and overall systemic framework of embedded intelligence on a total system; agents may be information technology, human, hybrid, or other forms of management.
- The four types of Embedded Intelligence Pattern functional roles that arise are the Managed System, Management System, System of Users and System of Access.

Embedded Intelligence (EI) Pattern

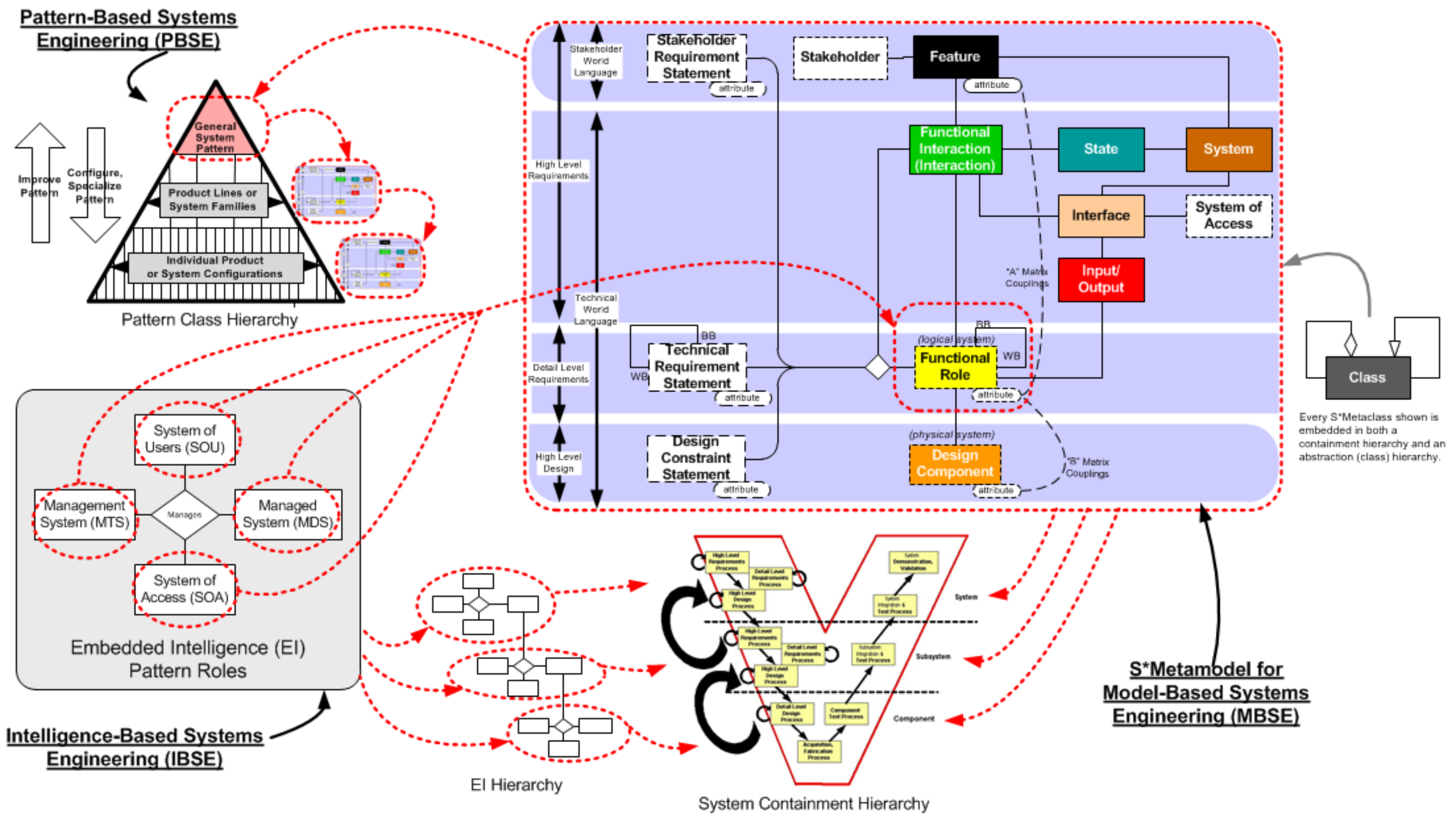
- **Managed System (MDS):** Any system behavior whose performance, configuration, faults, security, or accounting are to be managed--referred to as System Management Functional Areas (SMFAs) or in ISO terminology fault, configuration, accounting, performance, security (FCAPS). MDS is the "controlled plant".

- **Management System (MTS):** The roles of performing management (active or passive) of any of the SMFAs of the managed system. This roles may be played by automation technology, human beings, or hybrids thereof, to accomplish regulatory or other management purposes. MTS is the "controller".

- **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.

- **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


Pattern-Based Systems Engineering (PBSE)
Improve Pattern
Configure, Specialize Pattern
General System Pattern
Product Lines or System Families
Individual Product or System Configurations
Pattern Class Hierarchy
System of Users (SOU)
Management System (MTS)
Manages
Managed System (MDS)
System of Access (SOA)
Embedded Intelligence (EI) Pattern Roles
Intelligence-Based Systems Engineering (IBSE)
EI Hierarchy
Stakeholder World Language
Stakeholder Requirement Statement
attribute
Stakeholder
Feature
High Level Requirements
Functional Interaction (Interaction)
State
System
Interface
System of Access
"A" Matrix Couplings
Input/ Output
Technical World Language
Detail Level Requirements
BB
WB
Technical Requirement Statement
(logical system)
Functional Role
High Level Design
Design Constraint Statement
(physical system)
Design Component
"B" Matrix Couplings
Class
Every S*Metaclass shown is embedded in both a containment hierarchy and an abstraction (class) hierarchy.
System
Subsystem
Component
System Containment Hierarchy
S*Metamodel for Model-Based Systems Engineering (MBSE)

States and Situation Resolution Cycles

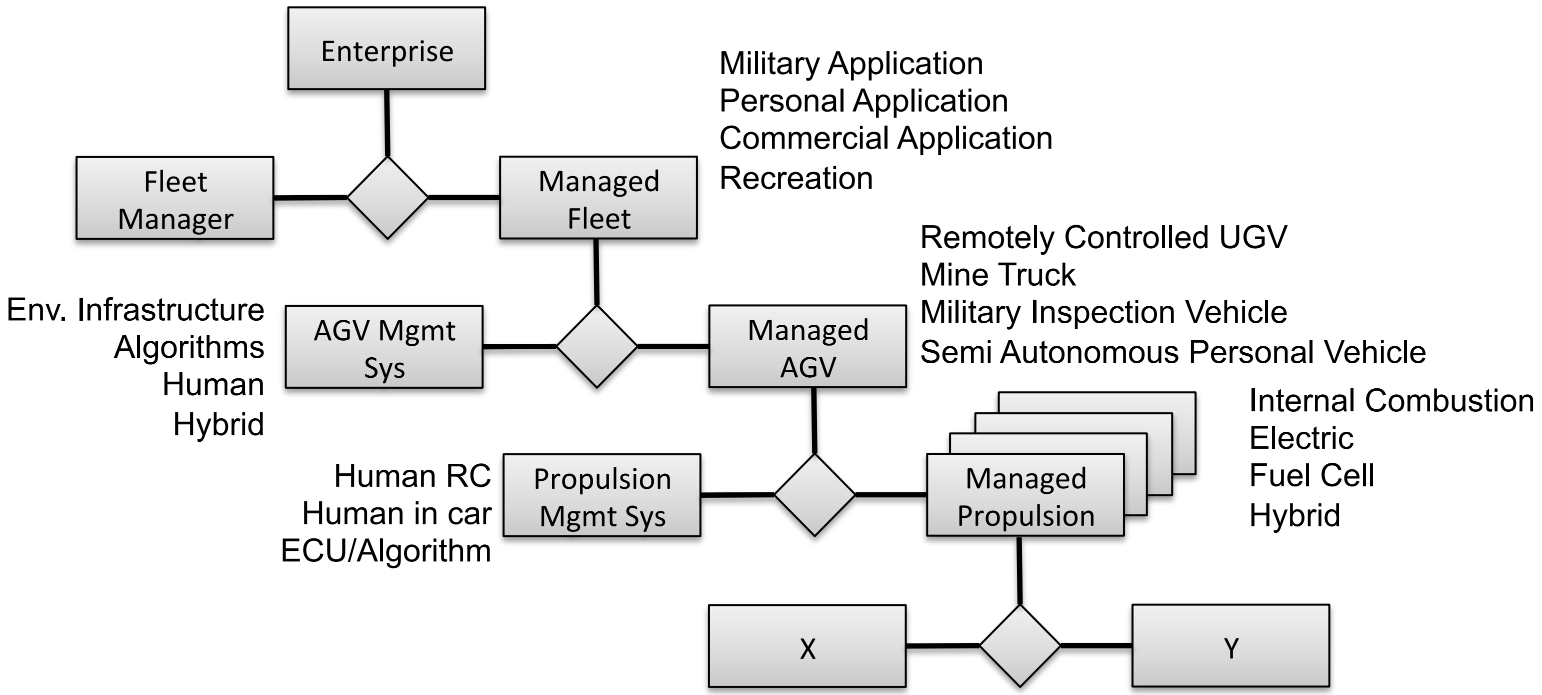
System stability over time requires a form of system regulation to “resolve” various “situations” that may occur from time to time, driving the managed system back to a “normal” or nominal state.

- **Major Mission Resolution Cycles**: These proceed through a series of mission states, from mission initiation to fulfilment, including planning.
- **Minor Use Case Resolution Cycles:** These similarly resolve various situational use cases.
- **Resolution of Faults:** These may include the recognition, diagnosis, repair, and recovery from system faults.
- **Resolution of Service Requests:** These may include resolution of requests for such services as re-configuration, security, or other situations.

If a system is capable of traveling a situation resolution cycle trajectory and recognizing that such a situation has arisen in the first place the the system is said to be “Situationally Aware”.

High Level AGV EI Pattern

- Diagram of AGV EI Pattern



This management hierarchy continues downward, but also upward—there is management at higher levels, including traffic control, fleet management, site management, warehouse management, mission control, etc.

Vehicle and EI Pattern Integration

- The integration of the vehicle and EI pattern provides a very powerful and compact way to model the diversity of AGVs including application, manned and unmanned, level and types of control
- The integration is a straight forward exercise given both are compliant with the S* metamodel to form S* patterns.
- For AGVs this permits the configuration and reuse of the general vehicle pattern and EI pattern to cover the diverse range of platforms within the AGV domain.
- Configuration of a specialized AGV can be accomplished specifically thorough the selection of features to ultimately configure systems solutions.
- The next thing the PBSE Challenge Team will do is to incorporate feature selection following the precepts of ISO 26550 "Software and Systems Engineering - Reference Model for Product Line Engineering and Management".

Vehicle Pattern Configuration Chart

- Unlike the diversity of physical implementation across these AGV platforms, whether manned or unmanned, wheeled or tracked, mine, city or wartime environments the means and allocation of control can have a very high level of commonality well represented by control patterns--more specifically the Embedded Intelligence (EI) Pattern.

Vehicle Management Capability	Remotely Controlled AGV	Inspection / Mining AGV	Fully Autonomous AGV
Faults Management Situation	Few	Several	Many
Configuration Management	0	Several	Many
Accounting Management	Few	Several	Many
Performance Management	Few	Several	Many
Security Management	0	0	Many

Follow on Work

Elaboration of this work will be performed within a sub-team of the INCOSE PBSE Challenge Team whose focus is to model an Automated Ground Vehicle Platform. More specifically the sub-team will focus on the follow items:

- Expand the depth of both the vehicle and embedded intelligence pattern to build an Automated Ground Vehicle Platform Pattern. Initial work has focused on limited autonomy in an unmanned remotely controlled platform. In elaborating the model the sub-team will expand and detail feature selection and optioning using the precepts of Product Line Engineering (ISO 26550)
- Use the AGV model to further demonstrate the use of Design Structure Matrix (DSM) and Network Analysis as aids in architecting engineered systems. More specifically to aid with modularization, partitioning, visualizing allocations across Multi-Domain Matrices (MDMs) and investigating key network metrics and their ability to aid in architecting systems.
- Determine the applicability of ongoing standardization efforts affecting AGVs. In specific either SARTRE semi-automated truck platooning system or the AUTomotive Open Systems ARchitecture (AUTOSAR) Chassis Control Functional Architecture.
- Explore the needs and opportunities to incorporate precepts of the ISO26262 functional safety standard into this pattern.

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

Conclusions

- The systemic complexity of AGVs demands a systems engineering approach
- It requires a systems paradigm which is interdisciplinary, leverages principals common to all complex systems and applies the requisite physics-based and mathematical models to represent them.
- For the approach discussed in this paper, the "methodology" includes not only process, but more significantly the very concept of the underlying information those processes produce and consume, independent of modeling language and tools.
- PBSE provides a data model and framework that is both holistic and compact and is well suited to address the complexity of AGVs
- PBSE and the EI pattern establish patterns of adaptive and hierarchical control which can be leveraged as a framework for engineering trusted systems.
- The Embedded Intelligence Pattern explicitly represents the logical roles which enable planned evolution and limits architectural lock in, effectively reducing switching costs and speeding technology integration.

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. (BCG, 2013) "The Most Innovative Companies 2013 – Lessons from Leaders" 2013 BCG Global Innovators Survey, BCG Analytics.
3. (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.
4. (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).
5. (Cloutier, 2008) Cloutier, R., Applicability of Patterns to Architecting Complex Systems: Making Implicit Knowledge Explicit. VDM Verlag Dr. Müller. 2008.
6. (Estafan, 2008) Estafan, J. 2008. Survey of model-based systems engineering (MBSE) methodologies. INCOSE MBSE Initiative.
7. (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.
8. (ICTT, 2013) Abbreviated Systematica Glossary, ICTT System Sciences, 2013.
9. (INCOSE Handbook, 2014) INCOSE Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities, Version 4, International Council on Systems Engineering (2014).
10. (INCOSE Patterns Team, 2014) INCOSE MBSE Initiative: Patterns Challenge Team 2013-14 Web Site: http://www.omgwiki.org/MBSE/doku.php?id=mbse:patterns:patterns
11. (ISO 15288, 2014) ISO/IEC 15288: Systems Engineering—System Life Cycle Processes. International Standards Organization (2014).
12. (ISO 26262, 2011) ISO 26262 "Road Vehicles—Functional Safety", 2011.
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.TS
15. (NHTSA, 2013), "2012 Motor Vehicle Crashes: Overview", National Highway Safety Administration , U.S. Department of Transportation, 2013.
16. (NSF, 2014)"NSF-Funded Joint Efforts", at www.nsf.gov/funding/pgm_summ.jsp?pims_id=503286
17. (Schindel, 2005a) Schindel, W., "Pattern-Based Systems Engineering: An Extension of Model-Based SE", INCOSE IS2005 Tutorial TIES 4, (2005).

References

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, 2011b) Schindel, W. “What Is the Smallest Model of a System?”, Proc. of the INCOSE 2011 International Symposium, International Council on Systems Engineering (2011).
21. (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
22. (Schindel, 2012a) Schindel, W. “Introduction to Pattern-Based Systems Engineering (PBSE)”, INCOSE Finger Lakes Chapter Webinar, April 26, 2012.
23. (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
24. (Schindel, 2013a) Schindel, W. “Interactions: At the Heart of Systems”, INCOSE Great Lakes Regional Conference on Systems Engineering, W. Lafayette, IN, October, 2013.
25. (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
26. (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.
27. (Schindel, Peterson, 2014) “Pattern Based Systems Engineering – Leveraging Model Based Systems Engineering for Cyber-Physical Systems”, Proc. of 2014 NDIA Ground Vehicle Systems Engineering and Technology Symposium, Systems Engineering Technical Session, August 12-14, 2014, Novi, MI.
28. (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).
29. (Weiner, 1965) Norbert Weiner, Cybernetics: Control and Communication in the Animal and the Machine, Cambridge, MA, MIT Press, 1965.
30. US Census, 2009 - https://www.census.gov/prod/2011pubs/acs-15.pdf

Model-Based System Patterns for Automated Ground Vehicle Platforms

Abstract. Automated Ground Vehicle (AGV) platform research and engineering is proliferating across commercial, military, and consumer applications. Beyond diversity of form and application, AGVs can be manned or unmanned, and exhibit a broad range of automated control, from partial to full autonomy, making these vehicles strikingly diverse.

This paper reports on application of Pattern-Based Systems Engineering (PBSE) to representation of automated ground vehicle platforms. PBSE is based upon reusable, configurable S*Models conforming to the S*Metamodel, expressed in any modeling language and toolset. The INCOSE MBSE Initiative Patterns Challenge Team has been practicing PBSE across applications, reported in this and other IS2015 papers.

A specialized class of Cyber-Physical Systems, AGVs are subject to intense interest, creating new opportunities, risks, and complexities. To address the diversity and complexity of these systems, the Embedded Intelligence (EI) Pattern, another S*Pattern, is being applied by the team to illustrate its applicability to an AGV Platform Pattern.

Biographies

Troy Peterson is a Chief Engineer and Booz Allen Fellow operating as a firm-wide resource supporting top priority programs. Prior to Booz Allen, he worked at Ford Motor Company and operated his own engineering consulting business. He serves on the MSU Mechanical Engineering Department Advisory Board and INCOSE's Corporate Advisory Board. He's also a past president of INCOSE's Michigan Chapter and he co-leads the Patterns Challenge Team of the INCOSE MBSE Initiative. Troy is also INCOSE's AD for Systems Engineering Transformation

William D. (Bill) Schindel is president of ICTT System Sciences. His engineering career began in mil/aero systems with IBM Federal Systems, included faculty service at Rose-Hulman Institute of Technology, and founding of three systems enterprises. Bill co-led a 2013 project on the science of Systems of Innovation in the INCOSE System Science Working Group. He co-leads the Patterns Challenge Team of the INCOSE MBSE Initiative

Misc Backup Slides

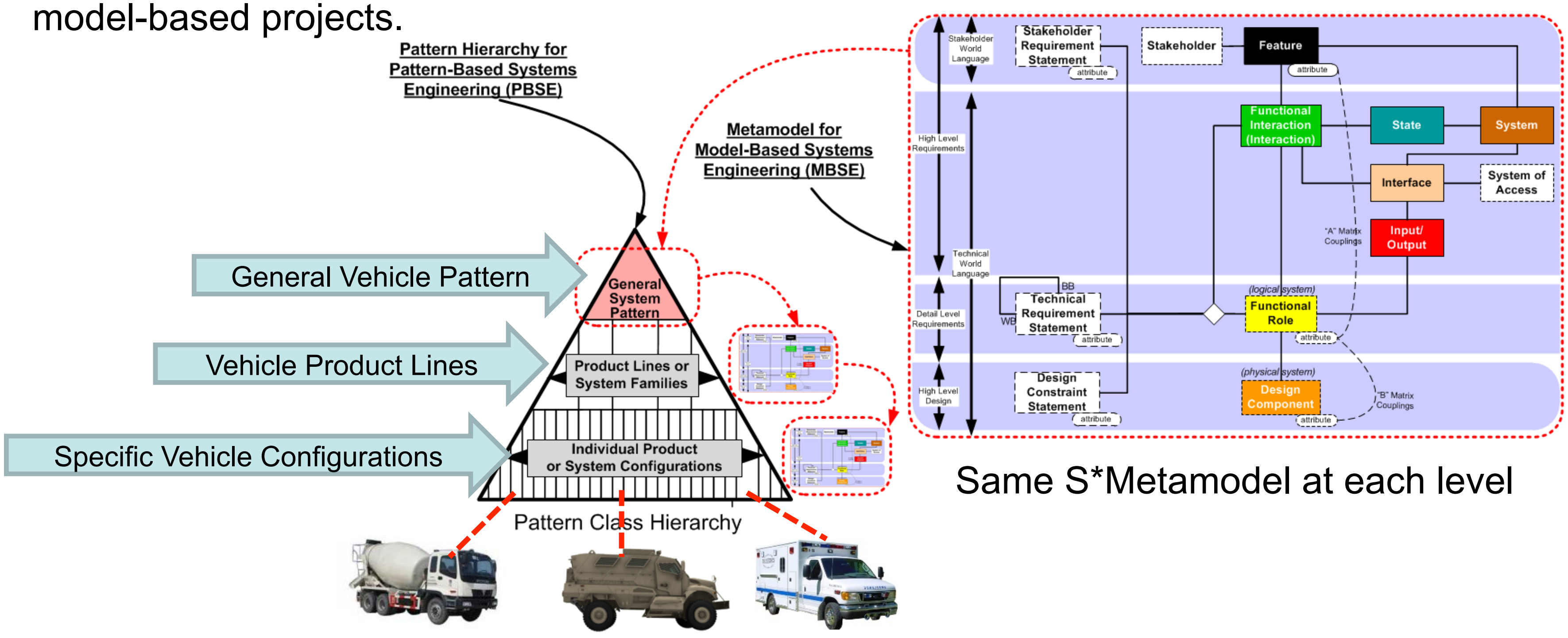
- Text

Concept Summary: Pattern-Based Systems Engineering (PBSE)

– By including the appropriate S* Metamodel concepts, these can readily be managed in (SysML or other) preferred modeling languages and MBSE tools—the ideas involved here are not specific to a modeling language or specific tool.

– The order-of-magnitude changes have been realized because projects that use PBSE rapidly start from an existing Pattern, gaining the advantages of its content, and feed the pattern with what they learn, for future users.

– The "game changer" here is the shift from "learning to model" to "learning the model", freeing many people to rapidly configure, specialize, and apply patterns to deliver value in their model-based projects.

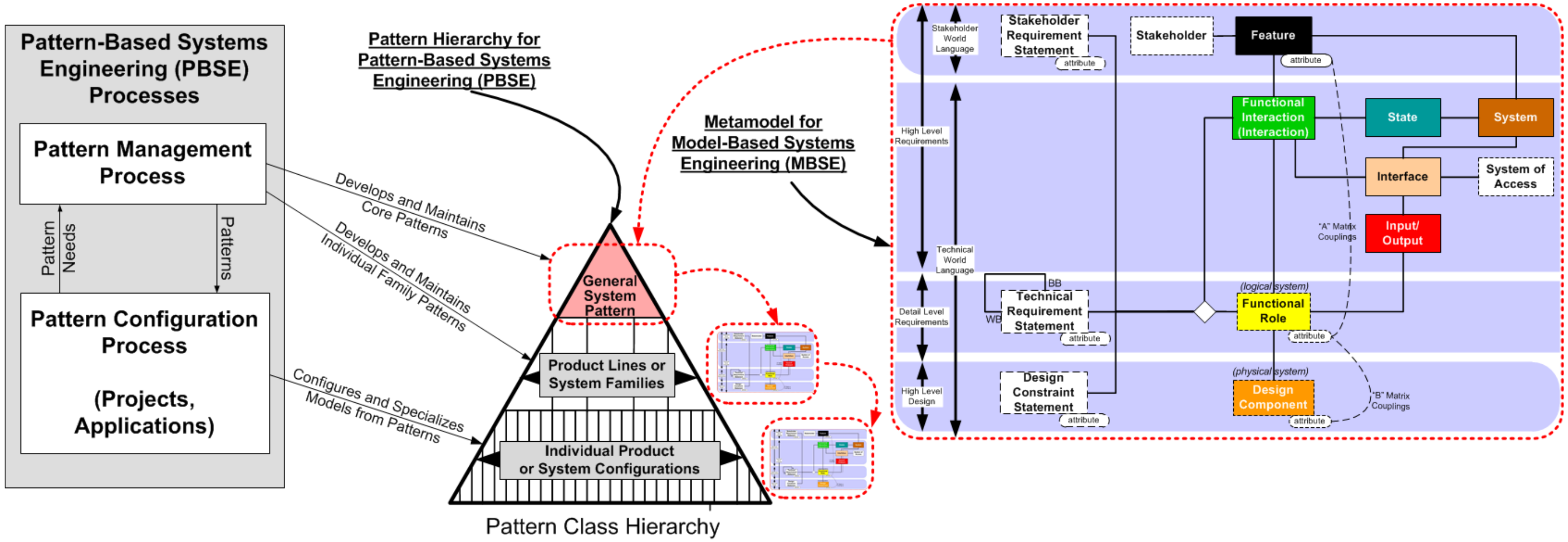


Pattern Hierarchy for Pattern-Based Systems Engineering (PBSE)
Metamodel for Model-Based Systems Engineering (MBSE)
General Vehicle Pattern
Vehicle Product Lines
Specific Vehicle Configurations
General System Pattern
Product Lines or System Families
Individual Product or System Configurations
Pattern Class Hierarchy
Stakeholder World Language
High Level Requirements
Technical World Language
Detail Level Requirements
High Level Design
Stakeholder Requirement Statement
attribute
Stakeholder
Feature
Functional Interaction (Interaction)
State
System
Interface
System of Access
"A" Matrix Couplings
Input/ Output
BB
WB
Technical Requirement Statement
(logical system)
Functional Role
Design Constraint Statement
(physical system)
Design Component
"B" Matrix Couplings

Same S*Metamodel at each level

Concept Summary: Pattern-Based Systems Engineering (PBSE)

- PBSE provides a specific technical method for implementing:
  - Platform Management
  - Enterprise or Industry Frameworks
  - System Standards
  - Experience Accumulation for Systems of Innovation
  - Lean Product Development & IP Asset Re-use



Pattern-Based Systems Engineering (PBSE) Processes
Pattern Management Process
Pattern Needs
Patterns
Pattern Configuration Process (Projects, Applications)
Pattern Hierarchy for Pattern-Based Systems Engineering (PBSE)
Develops and Maintains Core Patterns
Develops and Maintains Individual Family Patterns
Configures and Specializes Models from Patterns
General System Pattern
Product Lines or System Families
Individual Product or System Configurations
Pattern Class Hierarchy
Metamodel for Model-Based Systems Engineering (MBSE)
Stakeholder World Language
High Level Requirements
Technical World Language
Detail Level Requirements
High Level Design
Stakeholder Requirement Statement
attribute
Stakeholder
Feature
Functional Interaction (Interaction)
State
System
Interface
System of Access
Input/ Output
"A" Matrix Couplings
BB
WB
Technical Requirement Statement
(logical system)
Functional Role
(physical system)
Design Constraint Statement
Design Component
"B" Matrix Couplings

Representing System Patterns: The S* Metamodel Framework

- What is the smallest amount of information we need to represent pattern regularities?
  - Some people have used prose to describe system regularities.
  - This is better than nothing, but usually not enough to deal with the spectrum of issues in complex systems.
- We use S* Models, which are the minimum model-based information necessary:
  - This is not a matter of modeling language—your current favorite language and tools can readily be used for S* Models.
  - The minimum underlying information classes are summarized in the S* Metamodel, for use in any modeling language.
- The resulting system model is made configurable and reusable, thereby becoming an S* Pattern.

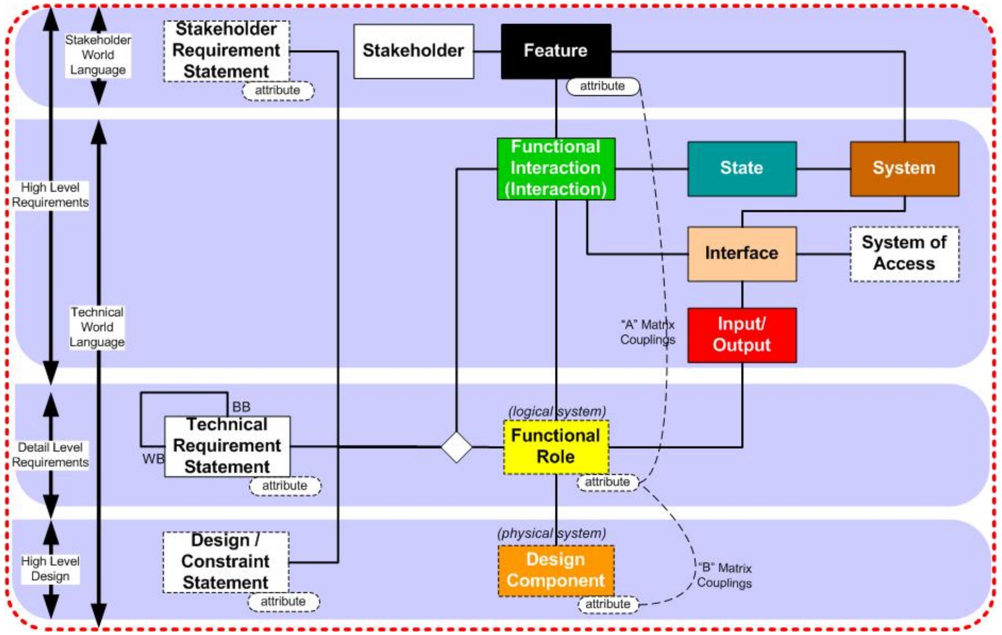
Representing System Patterns: The S* Metamodel Framework

- A metamodel is a model of other models;
  - Sets forth how we will represent Requirements, Designs, Verification, Failure Analysis, Trade-offs, etc.;
  - We utilize the (language independent) S* Metamodel from Systematica™ Methodology:

- The resulting system models may be expressed in SysML™, other languages, DB tables, etc.

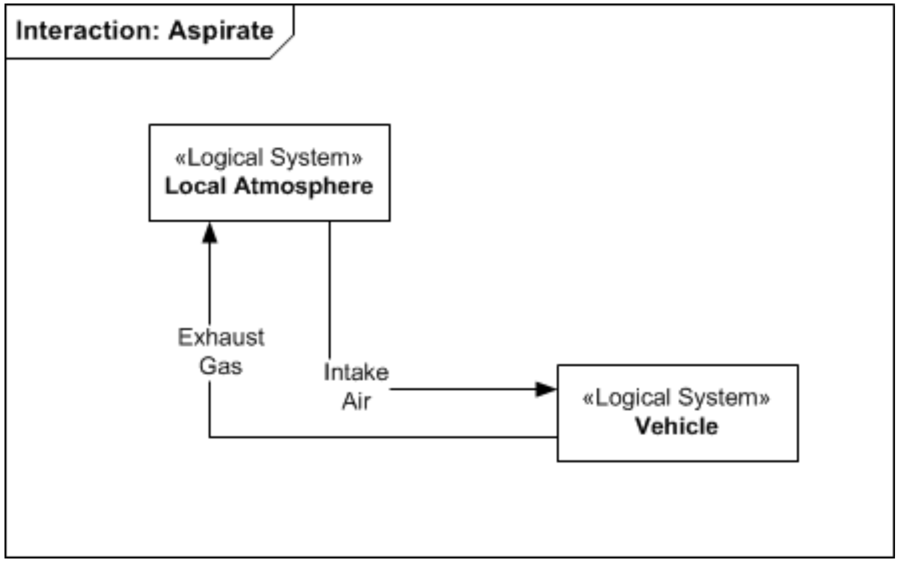
- Has been applied to systems engineering in aerospace, transportation, medical, advanced manufacturing, communication, construction, other domains.

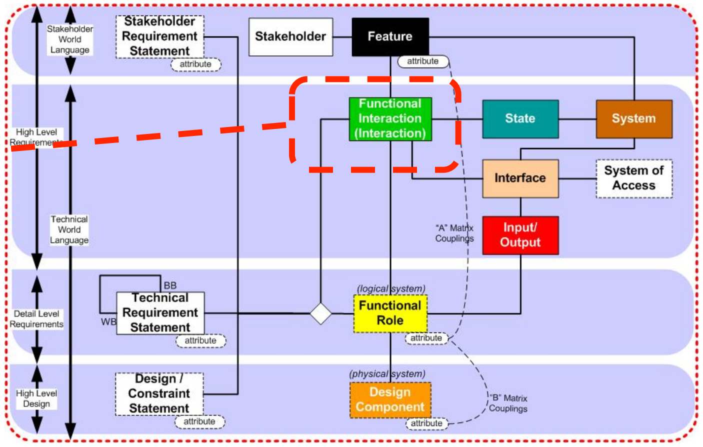
Simple summary of detailed S* Metamodel.



Physical Interactions: At the heart of S* models

- S* models represent Interactions as explicit objects:
  - Goes to the heart of 300 years of natural science of systems as a foundation for engineering, including emergence.
  - All physical laws of science are about interactions in some way.
  - All functional requirements are revealed as external interactions (!)





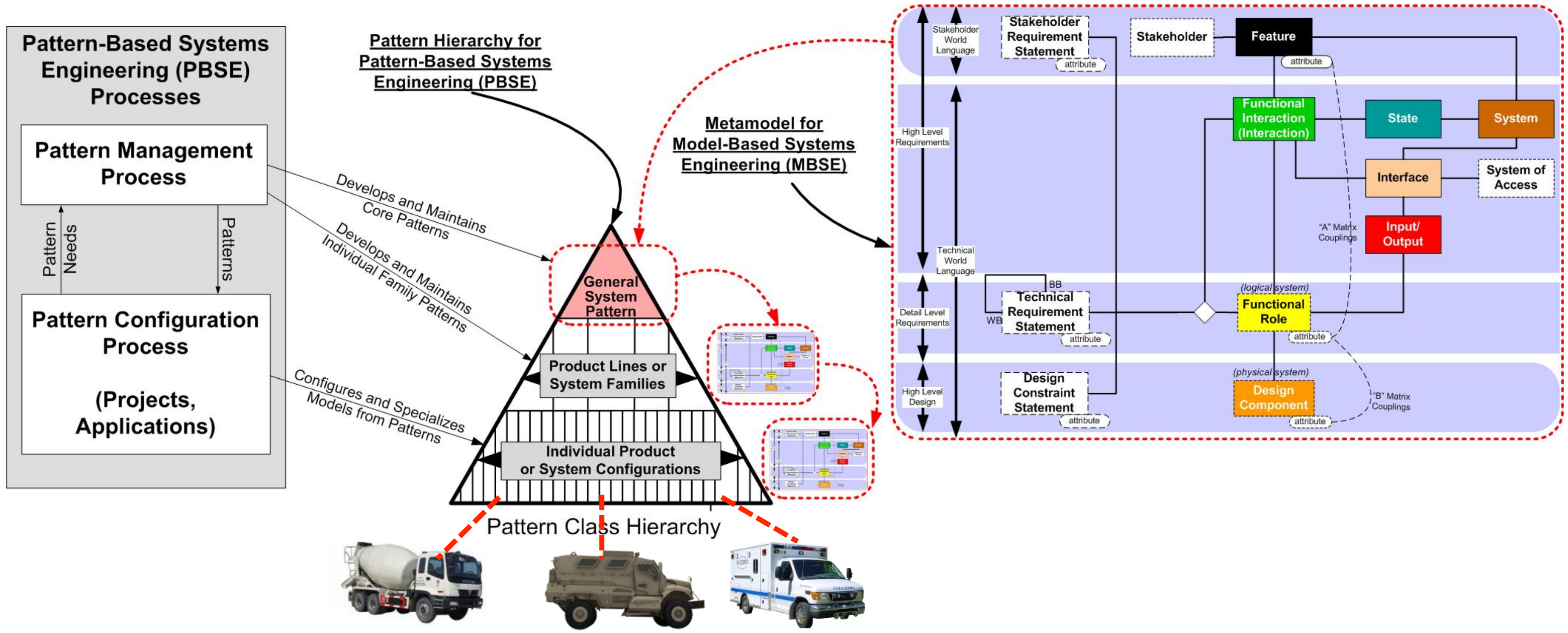
- Other Metamodel parts: See the Vehicle Pattern example.

Pattern-based systems engineering (PBSE)

- Model-based Patterns:
  - In this approach, Patterns are reusable, configurable S* models of families (product lines, sets, ensembles) of systems.
  - A Pattern is not just the physical product family—it includes its behavior, decomposition structure, failure modes, and other aspects of its model.
- These Patterns are ready to be configured to serve as Models of individual systems in projects.
- Configured here is specifically limited to mean that:
  - Pattern model components are populated / de-populated, and
  - Pattern model attribute (parameter) values are set
  - both based on Configuration Rules that are part of the Pattern.
- Patterns based on the same Metamodel as “ordinary” Models

Pattern-based systems engineering (PBSE)

- Pattern-Based Systems Engineering (PBSE) has two overall processes:
  - **<u>Pattern Management Process</u>**: Creates the general pattern, and periodically updates it based on application project discovery and learning;
  - **<u>Pattern Configuration Process</u>**: Configures the pattern into a specific model configuration (e.g., a new product) for application in a project.



Pattern-Based Systems Engineering (PBSE) Processes
Pattern Management Process
Pattern Needs
Patterns
Pattern Configuration Process
(Projects, Applications)
Pattern Hierarchy for Pattern-Based Systems Engineering (PBSE)
Develops and Maintains Core Patterns
Develops and Maintains Individual Family Patterns
Configures and Specializes Models from Patterns
General System Pattern
Product Lines or System Families
Individual Product or System Configurations
Pattern Class Hierarchy
Metamodel for Model-Based Systems Engineering (MBSE)
Stakeholder World Language
Stakeholder Requirement Statement
attribute
Stakeholder
Feature
attribute
High Level Requirements
Functional Interaction (Interaction)
State
System
Interface
System of Access
"A" Matrix Couplings
Input/ Output
Technical World Language
BB
Technical Requirement Statement
WB
attribute
Detail Level Requirements
(logical system)
Functional Role
attribute
(physical system)
High Level Design
Design Constraint Statement
attribute
Design Component
attribute
"B" Matrix Couplings

We'll discuss examples from both processes in this tutorial.

3. Selecting Solutions

More Informed Trade-offs

PBSE and Trades

Feature Space

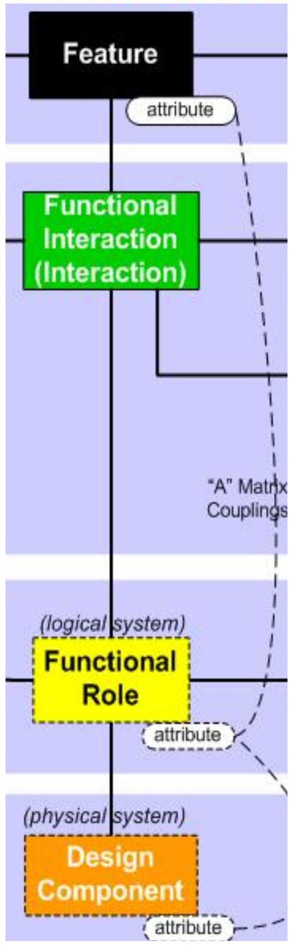
- Makes explicit all stakeholder needs
- Quantifies value impact through attributes
- Contains the entire trade space

Functional Role / Logical Architecture

- Logical, independent of design
- Describes the system's behavioral structure
- Formally models subsystems/design components
- Houses performance data (range, cost, weight etc.)
- Supports modeling of multiple physical architectures

Design Components

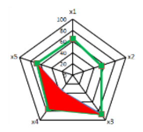
- Contains subsystem and technology options
- Design component options populate the logical architecture to create system configurations
- Contains part numbers, option names etc.
- Models the physical architecture



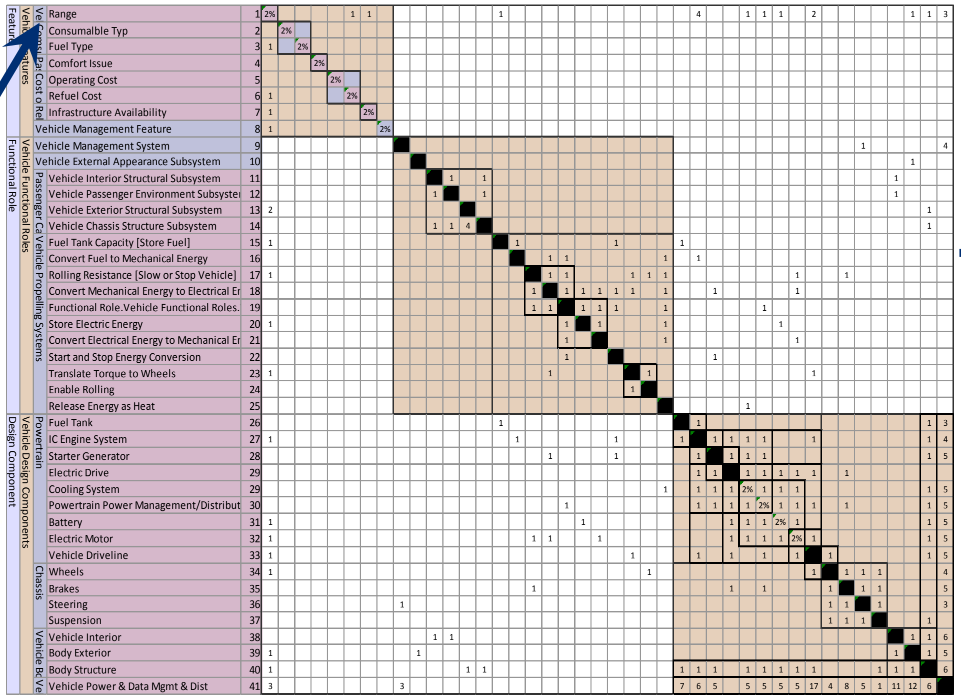
4. Design for Change
Architecture Management

Impact Analysis

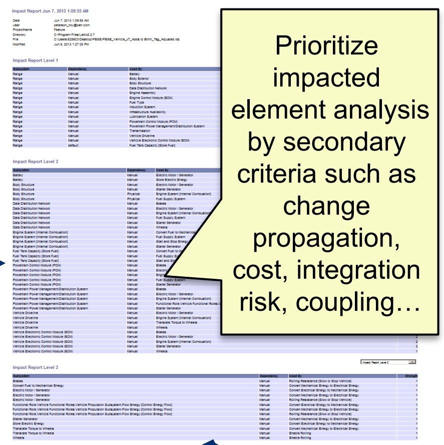
- ***Product Line/System Families/Platforms***: The common system pattern which enable rapid specialization or configuration of individual products / systems configurations i.e. product variants. Change impact analysis can aid in determining which elements remain a part of the family pattern, which are unique and which should become flexible.



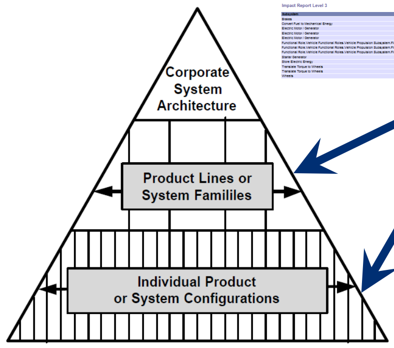
Δ Range


Vehicle Features
Functional Role
Design Component
Vehicle Functional Roles
Vehicle Design Components
Vehicle Propelling Systems
Powertrain
Chassis
Range
Consumable Typ
Fuel Type
Comfort Issue
Operating Cost
Refuel Cost
Infrastructure Availability
Vehicle Management Feature
Vehicle Management System
Vehicle External Appearance Subsystem
Vehicle Interior Structural Subsystem
Vehicle Passenger Environment Subsystem
Vehicle Exterior Structural Subsystem
Vehicle Chassis Structure Subsystem
Fuel Tank Capacity [Store Fuel]
Convert Fuel to Mechanical Energy
Rolling Resistance [Slow or Stop Vehicle]
Convert Mechanical Energy to Electrical Er
Functional Role.Vehicle Functional Roles.
Store Electric Energy
Convert Electrical Energy to Mechanical Er
Start and Stop Energy Conversion
Translate Torque to Wheels
Enable Rolling
Release Energy as Heat
Fuel Tank
IC Engine System
Starter Generator
Electric Drive
Cooling System
Powertrain Power Management/Distribut
Battery
Electric Motor
Vehicle Driveline
Wheels
Brakes
Steering
Suspension
Vehicle Interior
Body Exterior
Body Structure
Vehicle Power & Data Mgmt & Dist

Generate impact report of realized / modeled uncertainties



Prioritize impacted element analysis by secondary criteria such as change propagation, cost, integration risk, coupling…

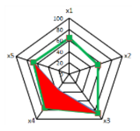

Corporate System Architecture
Product Lines or System Famililes
Individual Product or System Configurations

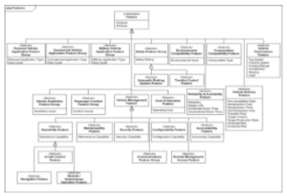
Address uncertainty as high up in the pattern as possible to leverage across the portfolio

1. deWeck, Oli, Strategic Engineering: Designing Systems for an Uncertain Future, Flexible Product Platforms: Framework and Case Study
2. Kalligeros K., de Weck O., de Neufville R., Luckins A., "Platform Identification using Design Structure Matrices", *Sixteenth Annual International Symposium of the International Council On Systems Engineering (INCOSE)*, Orlando, Florida, 8 - 14 July 2006

Vehicle Example MDM

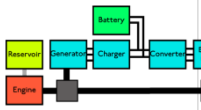
Features

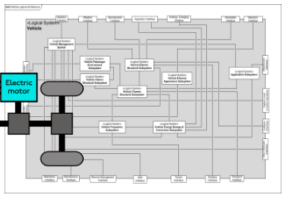




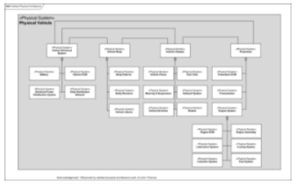
Requirement X

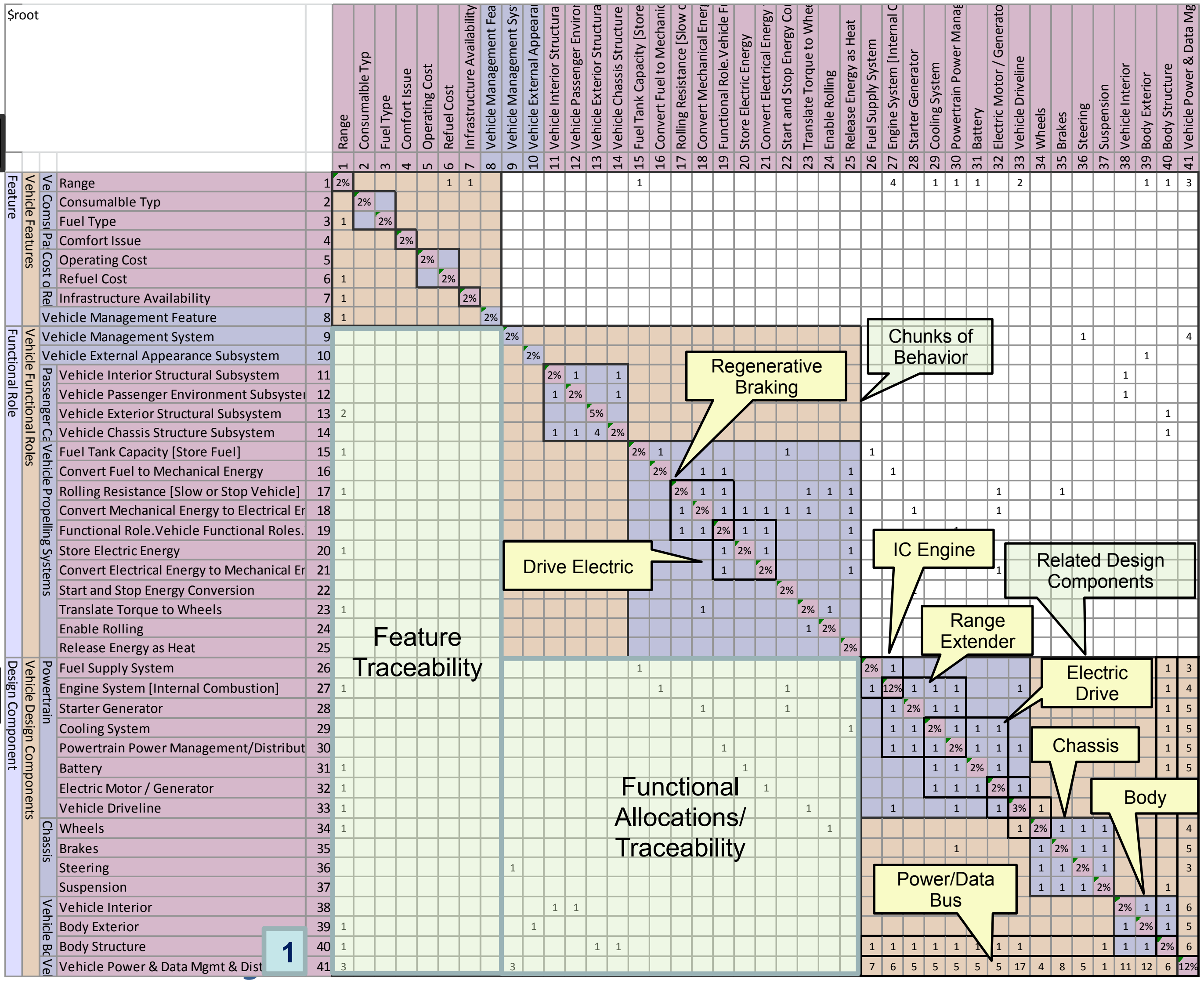
Functional Roles





Design Components




$root
Feature
Vehicle Features
Ve
Consu
Pa
Cost o
Re
Functional Role
Vehicle Functional Roles
Passenger Ca
Vehicle Propelling Systems
Design Component
Vehicle Design Components
Powertrain
Chassis
Vehicle Bo
Ve
Chunks of Behavior
Regenerative Braking
Drive Electric
IC Engine
Related Design Components
Range Extender
Electric Drive
Chassis
Body
Power/Data Bus
Feature Traceability
Functional Allocations/ Traceability

		1 Range	2 Consumalble Typ	3 Fuel Type	4 Comfort Issue	5 Operating Cost	6 Refuel Cost	7 Infrastructure Availability	8 Vehicle Management Fea	9 Vehicle Management Sys	10 Vehicle External Appeara	11 Vehicle Interior Structura	12 Vehicle Passenger Enviro	13 Vehicle Exterior Structura	14 Vehicle Chassis Structure	15 Fuel Tank Capacity [Store	16 Convert Fuel to Mechanic	17 Rolling Resistance [Slow c	18 Convert Mechanical Ener	19 Functional Role.Vehicle Fu	20 Store Electric Energy	21 Convert Electrical Energy	22 Start and Stop Energy Co	23 Translate Torque to Whee	24 Enable Rolling	25 Release Energy as Heat	26 Fuel Supply System	27 Engine System [Internal C	28 Starter Generator	29 Cooling System	30 Powertrain Power Manag	31 Battery	32 Electric Motor / Generato	33 Vehicle Driveline	34 Wheels	35 Brakes	36 Steering	37 Suspension	38 Vehicle Interior	39 Body Exterior	40 Body Structure	41 Vehicle Power & Data Mg
Range	1	2%					1	1								1												4		1	1	1		2						1	1	3
Consumalble Typ	2		2%																																							
Fuel Type	3	1		2%																																						
Comfort Issue	4				2%																																					
Operating Cost	5					2%																																				
Refuel Cost	6	1					2%																																				
Infrastructure Availability	7	1						2%																																			
Vehicle Management Feature	8	1							2%																																		
Vehicle Management System	9									2%																											1				4
Vehicle External Appearance Subsystem	10										2%																													1		
Vehicle Interior Structural Subsystem	11											2%	1		1																									1			
Vehicle Passenger Environment Subsyster	12											1	2%		1																									1			
Vehicle Exterior Structural Subsystem	13	2												5%																											1	
Vehicle Chassis Structure Subsystem	14											1	1	4	2%																											1	
Fuel Tank Capacity [Store Fuel]	15	1														2%	1						1				1															
Convert Fuel to Mechanical Energy	16																2%		1	1						1		1															
Rolling Resistance [Slow or Stop Vehicle]	17	1																2%	1	1				1	1	1								1			1							
Convert Mechanical Energy to Electrical Er	18																	1	2%	1	1	1	1	1		1			1				1										
Functional Role.Vehicle Functional Roles.	19																	1	1	2%	1	1				1					1											
Store Electric Energy	20	1																			1	2%	1				1																	
Convert Electrical Energy to Mechanical Er	21																			1			2%				1								1										
Start and Stop Energy Conversion	22																						2%						1														
Translate Torque to Wheels	23	1																1					2%	1																		
Enable Rolling	24																							1	2%																		
Release Energy as Heat	25																									2%																	
Fuel Supply System	26															1											2%	1																1	3
Engine System [Internal Combustion]	27	1															1						1				1	12%	1	1	1			1						1	4
Starter Generator	28																		1				1					1	2%	1	1												1	5
Cooling System	29																										1		1	1	2%	1	1	1								1	5
Powertrain Power Management/Distribut	30																			1									1	1	1	2%	1	1	1							1	5
Battery	31	1																				1									1	1	2%	1								1	5
Electric Motor / Generator	32	1																					1								1	1	1	2%	1							
Vehicle Driveline	33	1																							1				1			1			1	3%	1						
Wheels	34	1																									1									1	2%	1	1	1				4
Brakes	35																	1													1				1	2%	1	1				5
Steering	36									1																									1	1	2%	1				3
Suspension	37																																			1	1	1	2%			1	
Vehicle Interior	38											1	1																									2%	1	1	6
Body Exterior	39	1									1																												1	2%	1	5
Body Structure	40	1													1	1												1	1	1	1	1	1	1	1				1	1	1	2%	6
Vehicle Power & Data Mgmt & Dist	41	3								3																		7	6	5	5	5	5	5	17	4	8	5	1	11	12	6	12%

1