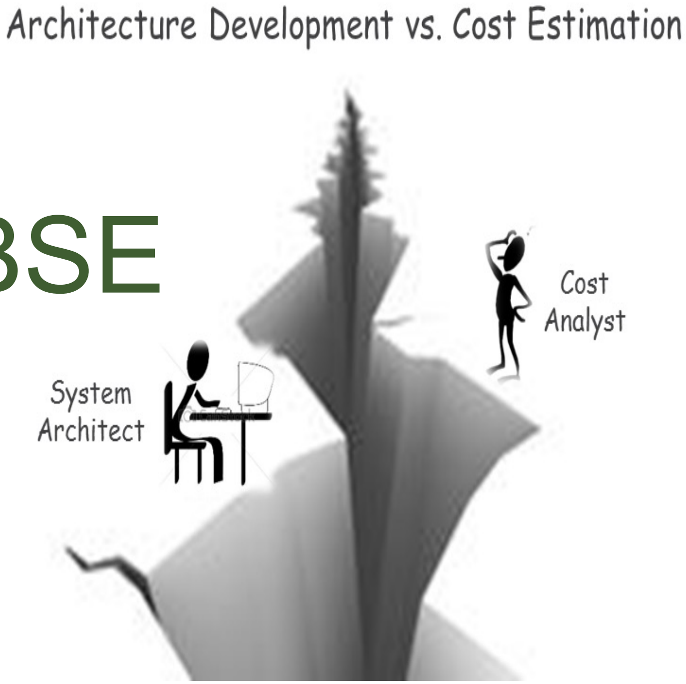
34th Annual INCOSE international symposium

hybrid event

Dublin, Ireland
July 2 - 6, 2024

The Convergence of COSYSMO Parametric Cost Estimation with MBSE



Barry Papke, Ricardo Valerdi, Gan Wang, Sean Densford

Hey! You’ve got your cost model in my architecture model!!

No! You’ve got your architecture model in my cost model!!

OMG SYSTEMS MODELING LANGUAGE SYSML™

CO SYS MO
CONSTRUCTIVE SYSTEMS ENGINEERING COST MODEL

Why Should Systems Engineers Care About Cost Estimating? Or About COSYSMO?

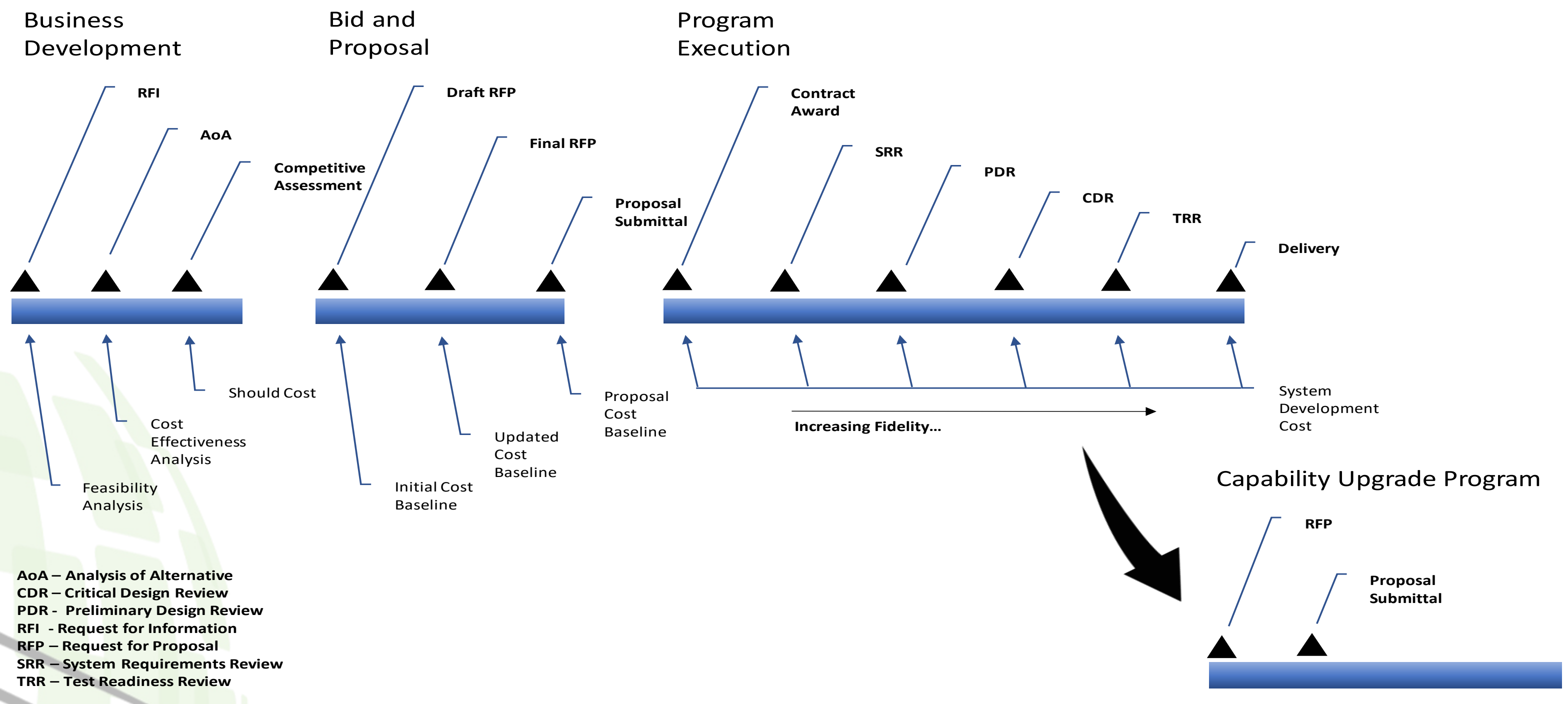
Concept Exploration
- Multiple concepts
- Investment decisions

Competitive Assessment
- Time critical
- Cost criteria affects design decisions

Proposal Gates
- Limited time and resources
- Early estimates based on high level concepts

Cost Volume Final Pricing
- Bottoms-up methods historically underestimate
- Time and labor intensive.

Engineering Change Proposals
- Design details often not available
- Limited time and resources

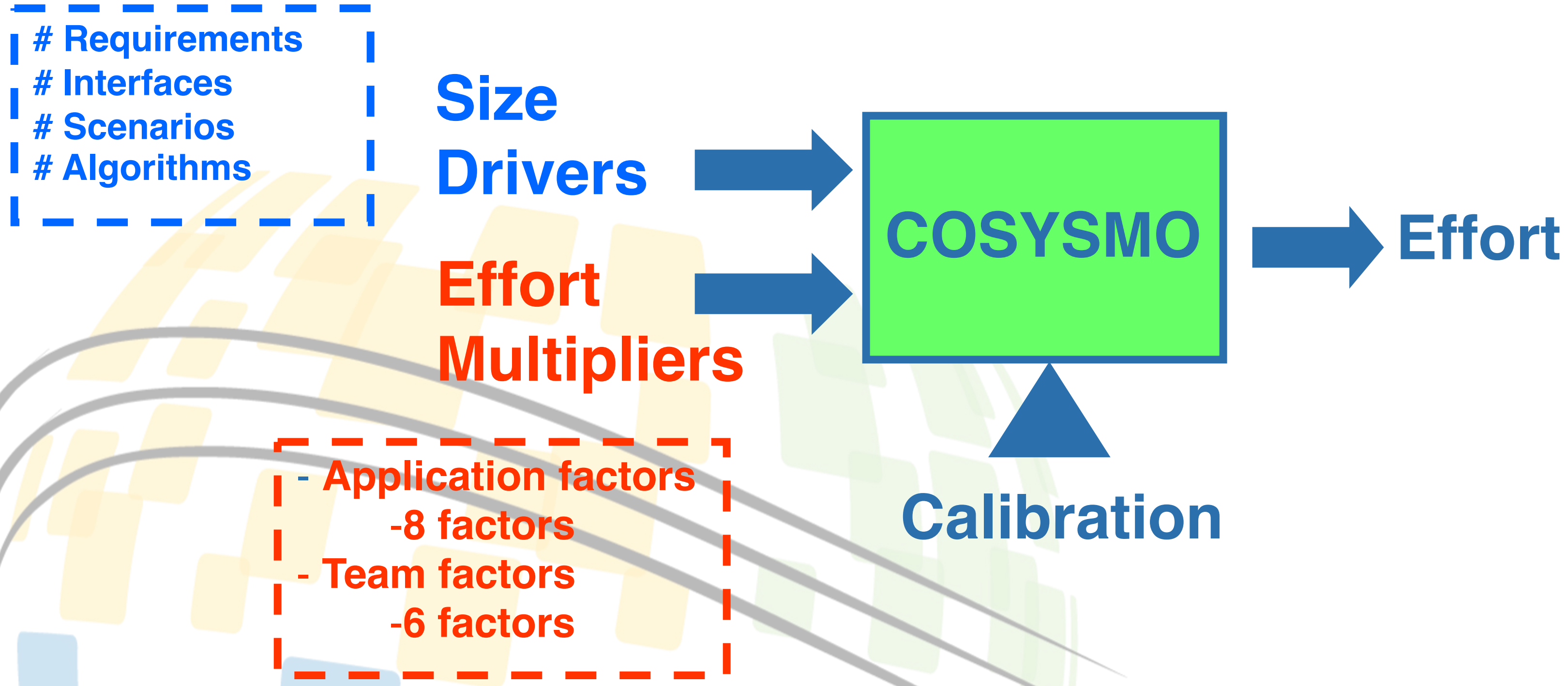


Cost baseline is part of the technical baseline throughout the project lifecycle!

A Quick Review of COSYSMO

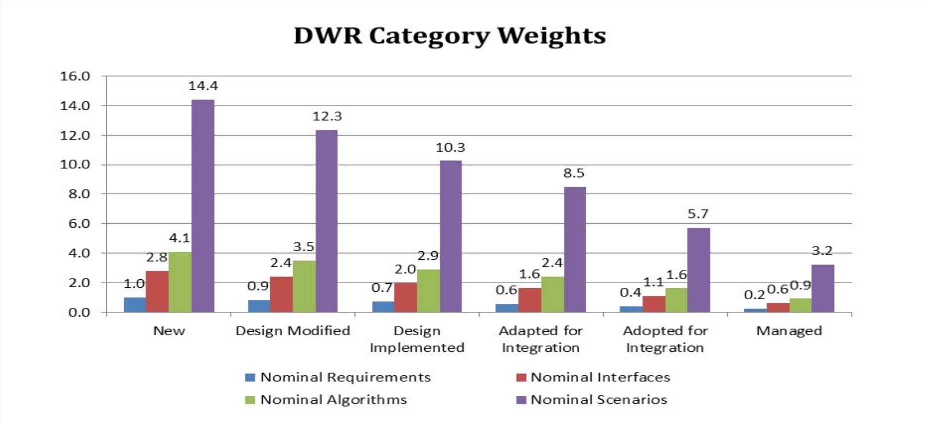
- History
  - 2000 COCOMO –Boehm
  - 2005 COSYSMO –Valerdi
  - 2007 SysML Standard - OMG
  - 2014 General Reuse Framework - Wang, Roedler, Pena, & Valerdi

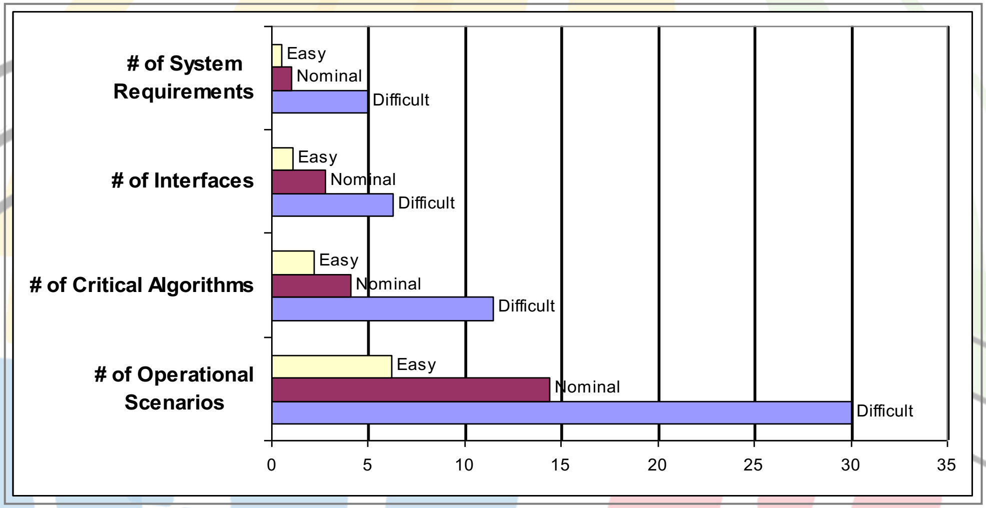
4 Size Drivers and 14 Cost Drivers

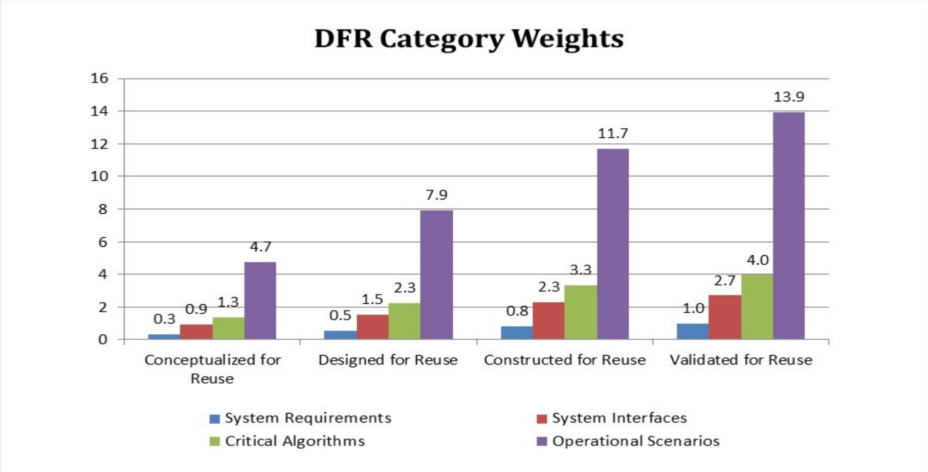


SIZE DRIVERS AND THE GENERAL REUSE FRAMEWORK

- The General Reuse Framework (GRF) extends COSYSMO with the concepts of "Development With Reuse (DWR)" and "Development For Reuse (DFR)."
  - Four Size Drivers
    - Requirements (REQ)
    - System Interfaces (IF)
    - Algorithms (ALG)
    - Operational Scenarios (SCN)
  - Three Difficulty Levels
  - Six DWR Reuse Levels
  - Four DFR Reuse Levels







REVIEW OF COSYSMO WITH THE GENERAL REUSE FRAMEWORK

The COSYSMO Cost Estimating Relationship (CER) with the GRF

$$\text{SE Effort} = (A_{DWR} \cdot \mathbf{SS}^{E_{DWR}} \cdot CEM_{DWR}) + (A_{DFR} \cdot \mathbf{SS}^{E_{DFR}} \cdot CEM_{DFR})$$

Where:

SE Effort = total systems engineering effort in person hours
SS = System Size (sum of each size driver count for DWR and DFR)
E = nonlinearity for the productivity curve, representing a diseconomy of scale
A = calibration constant, typically derived from historical project data
CEM = composite effort multiplier (cost drivers)

To elaborate the math, we get:

$$PH_{Total} = A_{DWR} \cdot \left[\sum_k \left(\sum_r w_r \left(w_{e,k} \Phi_{e,k} + w_{n,k} \Phi_{n,k} + w_{d,k} \Phi_{d,k} \right) \right) \right]^{E_{DWR}} \cdot CEM_{DWR}$$
$$+ A_{DFR} \cdot \left[\sum_k \left(\sum_q w_q \left(w_{e,k} \Psi_{e,k} + w_{n,k} \Psi_{n,k} + w_{d,k} \Psi_{d,k} \right) \right) \right]^{E_{DFR}} \cdot CEM_{DFR}$$

6 DWR Levels x 3 Difficulty Levels x 4 Size Drivers = 72 unique DWR SS Counts
4 DFR Levels x 3 Difficulty Levels x 4 Size Drivers = 48 unique DFR SS Counts

Background

- In 2017, we developed an approach to implement COSYSMO and the GRF in a SysML model.
- That work demonstrated the feasibility of the approach:
  - Simplied the identification and assignment of size driver properties
  - Automated the counting of size driver elements
  - Made the sizing data a property of the architecture model.
- Size driver counts could be exported to a COSYSMO cost model tool.

Convergence of COSYSMO with MBSE Today

- **This paper advances the previous work by:**
  - Updating the COSYSMO Size Driver Counting Rules specifically for SysML
  - Further Automating the Counting Process
  - Implementing the complete COSYSMO CER within the modeling tool (allowing full calculation of the system size and the total SE effort from within the model.)
  - Using Advanced Queries to review completeness, correctness, and consistency of the sizing driver assignments
- **The objective was to:**
  - Investigate how much of the COSYSMO cost estimation process could be implemented and automated in a SysML modeling tool
  - Evaluate the usability, benefits and impact
- **Disclaimer**
  - The implementation and the examples shown were developed and demonstrated with CAMEO Systems Model/CATIA Systems Architect:
  - The profile described is a proof of concept, not a product.
  - All features were created using out of the box capabilities of the modeling tool.

ORIGINALLY A DOCUMENT BASED APPROACH

- Developed before the release of SysML and the widespread adoption of MBSE, the sources for COSYSMO size drivers were document-based systems engineering artifacts:

  Counting rules reflected the subjective nature and wide variability of content in the source documents.

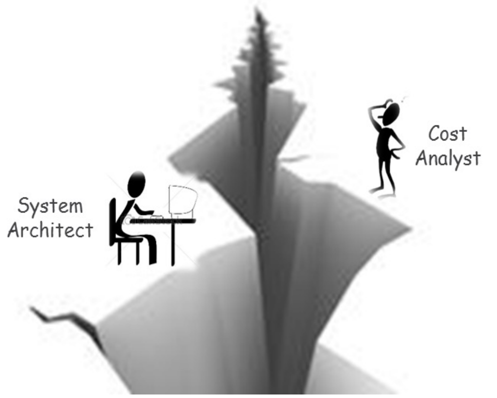
  Many of the documents referenced as size driver sources do not exist early in the program when the cost estimation activity is most needed.

  Counting and recording the size estimate was a manual process using spreadsheets.

  Sizing data was poorly maintained, managed separately from the design artifacts.

  Systems engineering continues to view parametric cost estimation as a pricing activity.

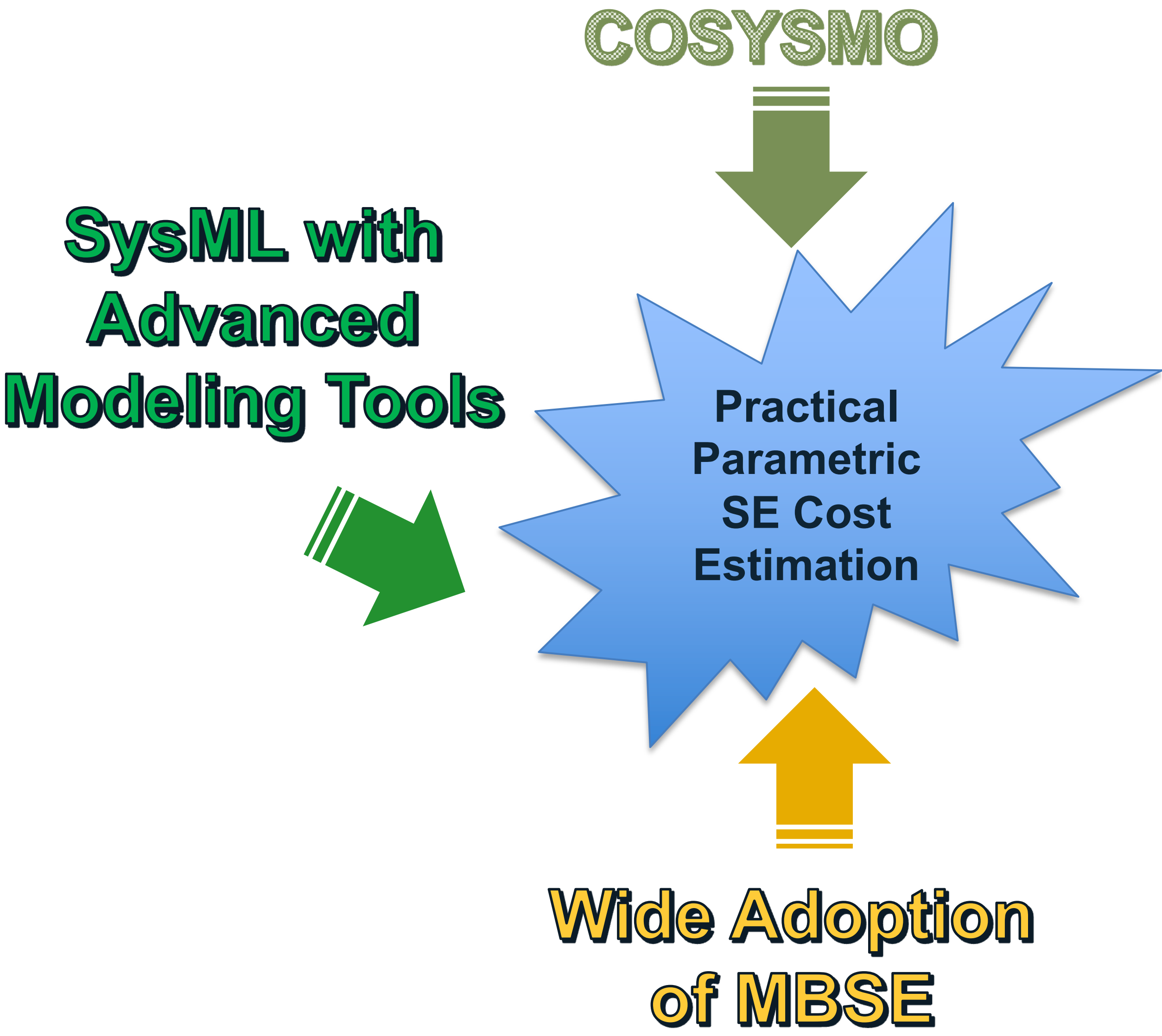
Architecture Development vs. Cost Estimation



Labor Intensive!

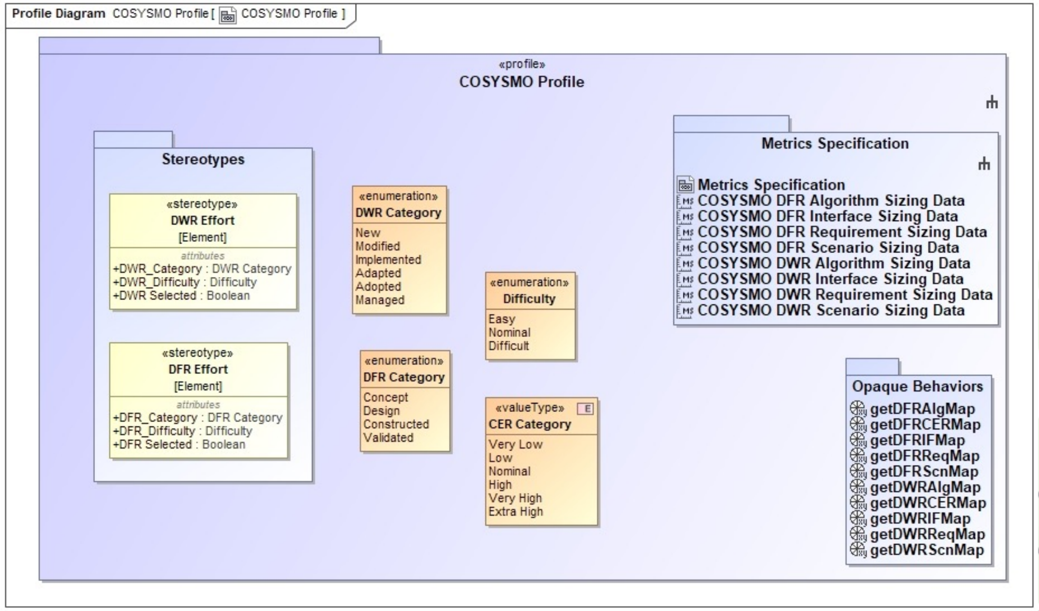
COSYMO AND THE AGE OF MBSE

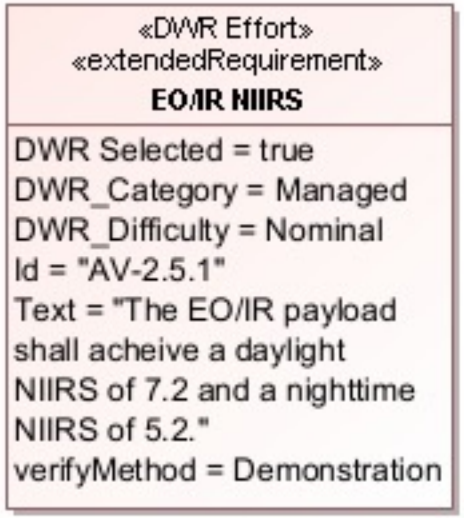
- SysML with advanced modeling tools:
  - Open standard systems modeling language
    - Provides a common, consistent set of model elements to represent each of the four size drivers
    - Provides a level of rigor, consistency and repeatability in the counting process.
  - Advanced tool features, such as custom tables that can execute complex queries using structured expressions and custom scripts
    - Can parse the model and automatically count the number of model elements in each of the 72 DWR and 48 DFR size driver categories.
- Wide Adoption of MBSE
  - Models are much more complete, much more rigorous and available much earlier in the project than documents.
  - Modeling tools enable the sizing data to become attributes of the architecture where it belongs!

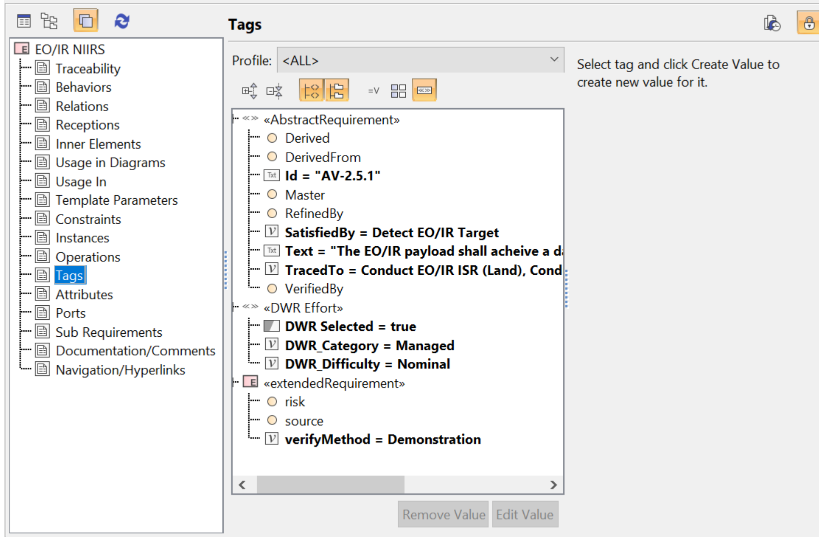


THE COSYSMO PROFILE

- The COSYSMO profile defines stereotypes for the DWR and DFR reuse levels and Difficulty level.
- When applied to a model element the, DWR, DFR and Complexity properties are implemented as Tag Values.
- The profile also include advanced scripts, structured expressions and Parametric Diagrams the automate the counting and calculation of system size and total SE effort.



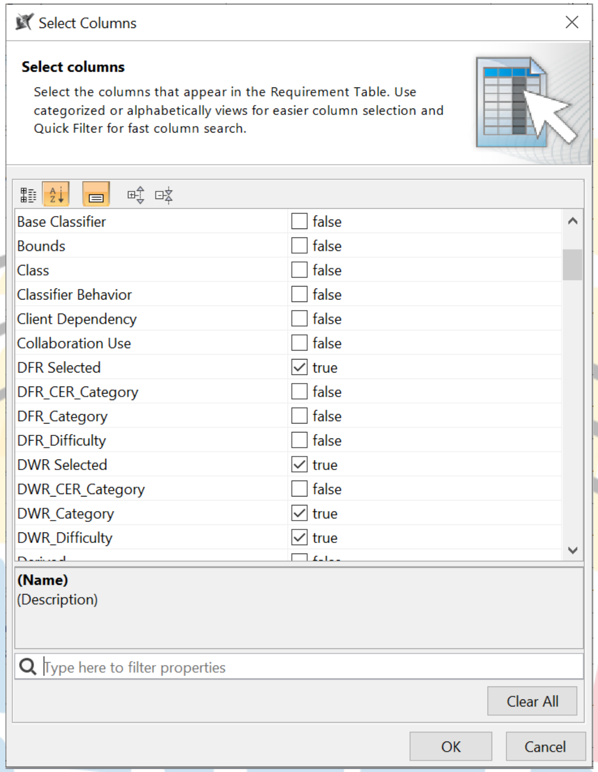


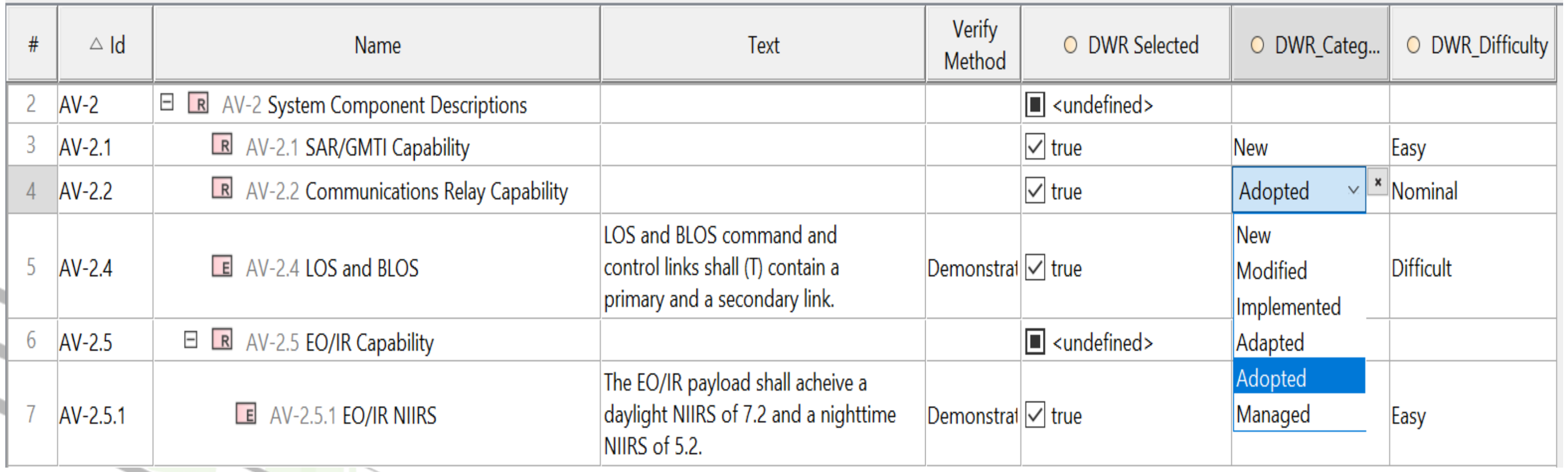


DASSAULT SYSTEMES

PROFILE FEATURES: SIMPLIFIED ASSIGNMENT OF REUSE CATEGORIES AND DIFFICULTY

- Features of the Generic Table in CAMEO made the assignment of DWR, DFR and Difficulty levels an efficient process.
  - In a Generic Table or a Requirements Table, create a New Custom Column for DWR, DFR and Difficulty Stereotypes.
  - The sizing analyst can first select the model elements to be included in the count and filter the table to show only selected requirements.
  - Clicking in the cell to present the enumerated list of possible values and then click on the value.

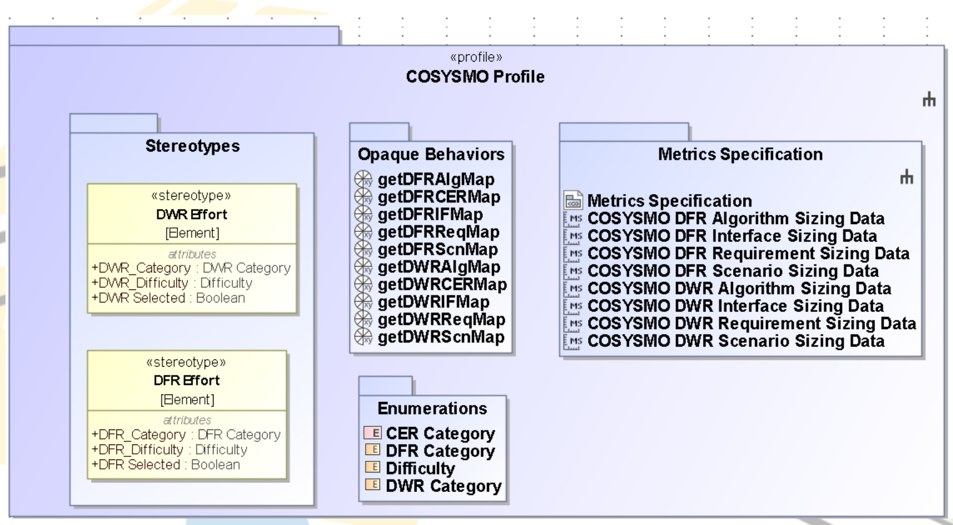


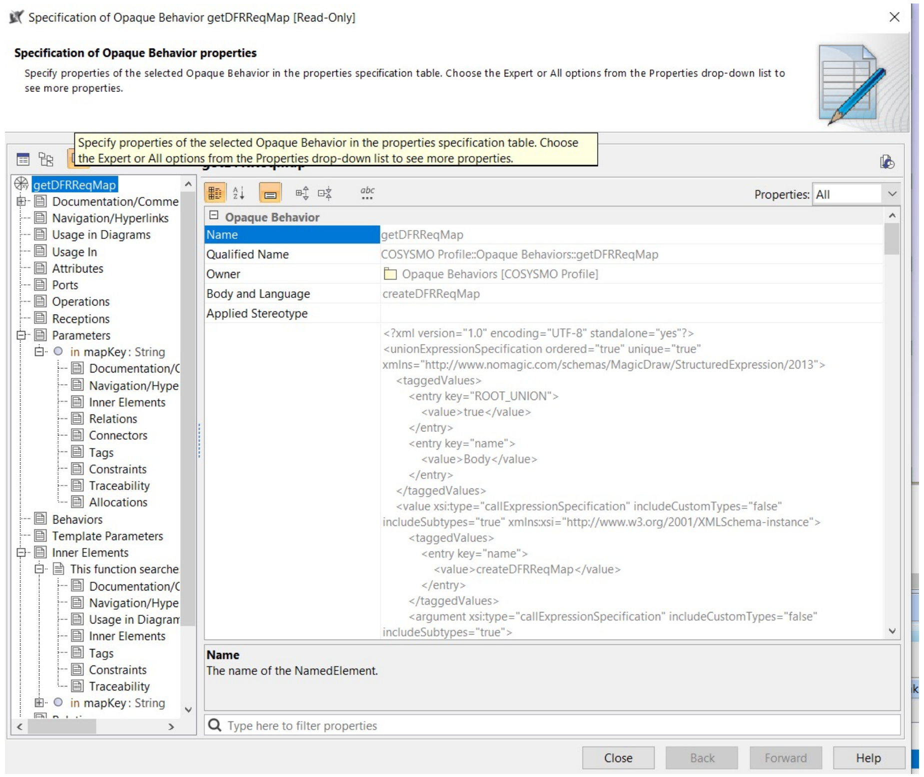


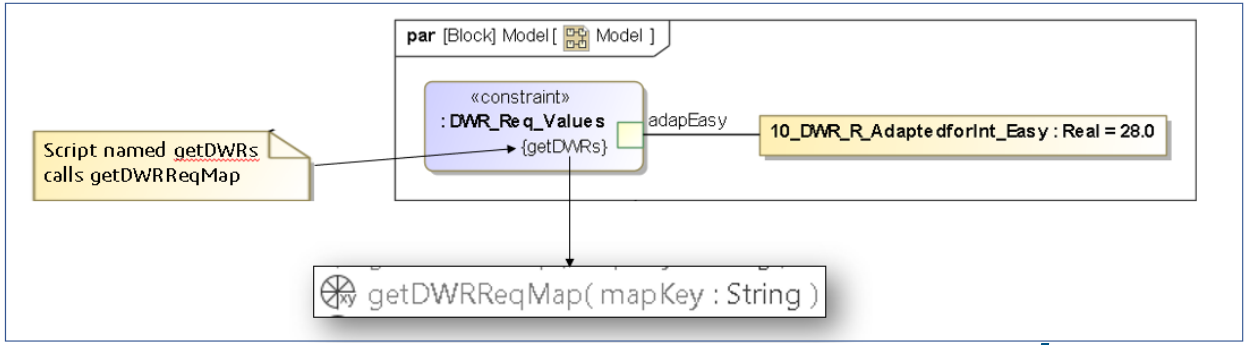
#	Id	Name	Text	Verify Method	DWR Selected	DWR_Categ...	DWR_Difficulty
2	AV-2	AV-2 System Component Descriptions			<undefined>		
3	AV-2.1	AV-2.1 SAR/GMTI Capability			true	New	Easy
4	AV-2.2	AV-2.2 Communications Relay Capability			true	Adopted	Nominal
5	AV-2.4	AV-2.4 LOS and BLOS	LOS and BLOS command and control links shall (T) contain a primary and a secondary link.	Demonstrat	true		Difficult
6	AV-2.5	AV-2.5 EO/IR Capability			<undefined>		
7	AV-2.5.1	AV-2.5.1 EO/IR NIIRS	The EO/IR payload shall acheive a daylight NIIRS of 7.2 and a nighttime NIIRS of 5.2.	Demonstrat	true		Easy

PROFILE FEATURES: AUTOMATION OF THE COUNTING PROCESS

- When executed, scripts parse the model, count the number of model elements in each of the DWR and DFR size driver categories and write the totals to their respective Size Driver Block Value Properties.
- The scripts are a combination of Stereotypes, Enumerations, and Opaque Behaviors.

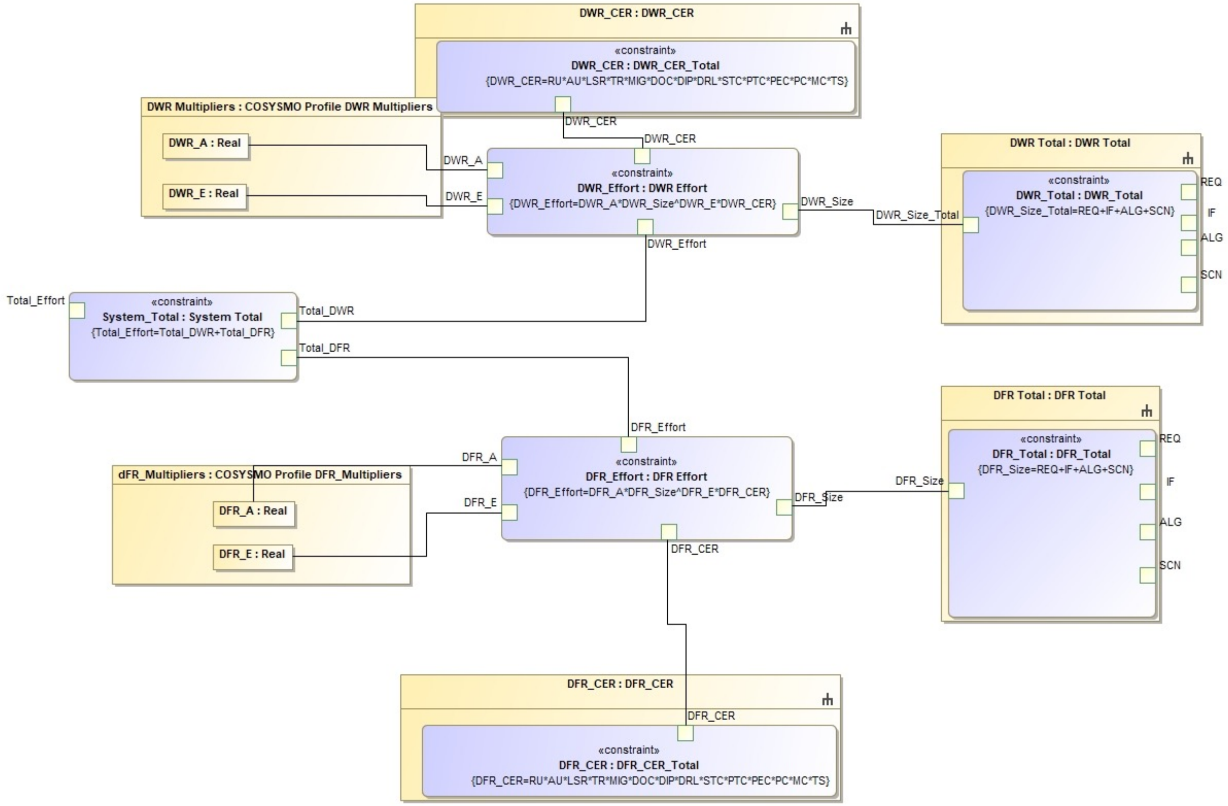


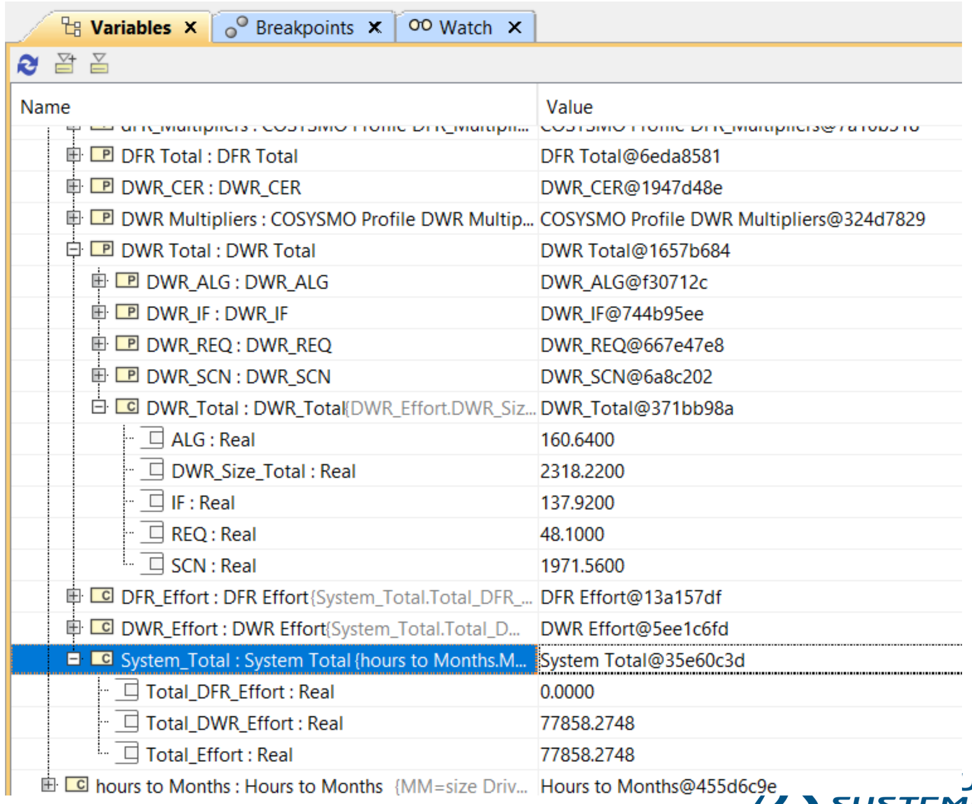




PROFILE FEATURES: AUTOMATIC CALCULATION OF SYSTEM SIZE AND TOTAL SE EFFORT

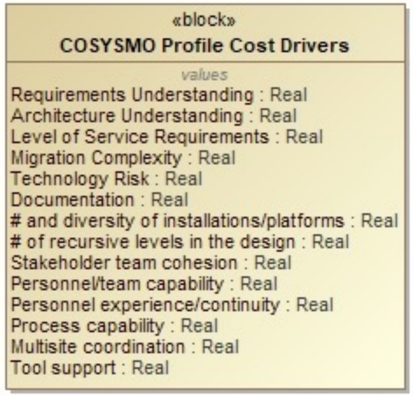
- The COSYSMO CER is implemented with Constraint Blocks in Parametric Diagrams.
- When the Parametric Diagram is executed, the profile calculates system size and SE effort.
- All properties at each level are displayed in the variables pane of the simulation window and saved as a sizing instance.





PROFILE FEATURES: THE COMPOSITE EFFORT MULTIPLIER IS IMPLEMENTED AS A BLOCK

- Cost Drivers are represented by a Cost Driver Block with Value Properties for the fourteen cost drivers.
- A Cost Driver stereotype allows the rating level to be applied to each cost driver value property.
- A custom script translates the rating level (Very Low to Extra High) into the numerical values that determine the Cost Effort Multiplier.



#	Name	DWR_CER_Category
1	# and diversity of installations/platforms	Very Low
2	# of recursive levels in the design	Low
3	Architecture Understanding	Nominal
4	Documentation	Nominal
5	Level of Service Requirements	Nominal
6	Migration Complexity	Nominal
7	Multisite coordination	Low
8	Personnel experience/continuity	Very High
9	Personnel/team capability	High
10	Process capability	High
11	Requirements Understanding	Very High
12	Stakeholder team cohesion	Nominal
13	Technology Risk	Nominal
14	Tool support	Extra High

	Cost Diver	Very Low	Low	Nominal	High	Very High	Extra High
1	Requirements Understanding	1.85	1.36	1.00	0.77	0.60	
2	Architecture Understanding	1.62	1.27	1.00	0.81	0.65	
3	Level of Service Requirements	0.62	0.79	1.00	1.32	1.74	
4	Migration Complexity			1.00	1.24	1.54	2.10
5	Technology Risk	0.70	0.84	1.00	1.32	1.74	
6	Documentation	0.82	0.91	1.00	1.13	1.28	
7	# and diversity of installations/platforms			1.00	1.23	1.51	2.00
8	# of recursive levels in the design	0.80	0.89	1.00	1.21	1.46	
9	Stakeholder team cohesion	1.50	1.22	1.00	0.81	0.66	
10	Personnel/team capability	1.48	1.22	1.00	0.81	0.66	
11	Personnel experience/continuity	1.46	1.21	1.00	0.82	0.67	
12	Process capability	1.46	1.21	1.00	0.88	0.77	
13	Multisite coordination	1.33	1.15	1.00	0.90	0.80	
14	Tool support	1.34	1.16	1.00	0.85	0.73	

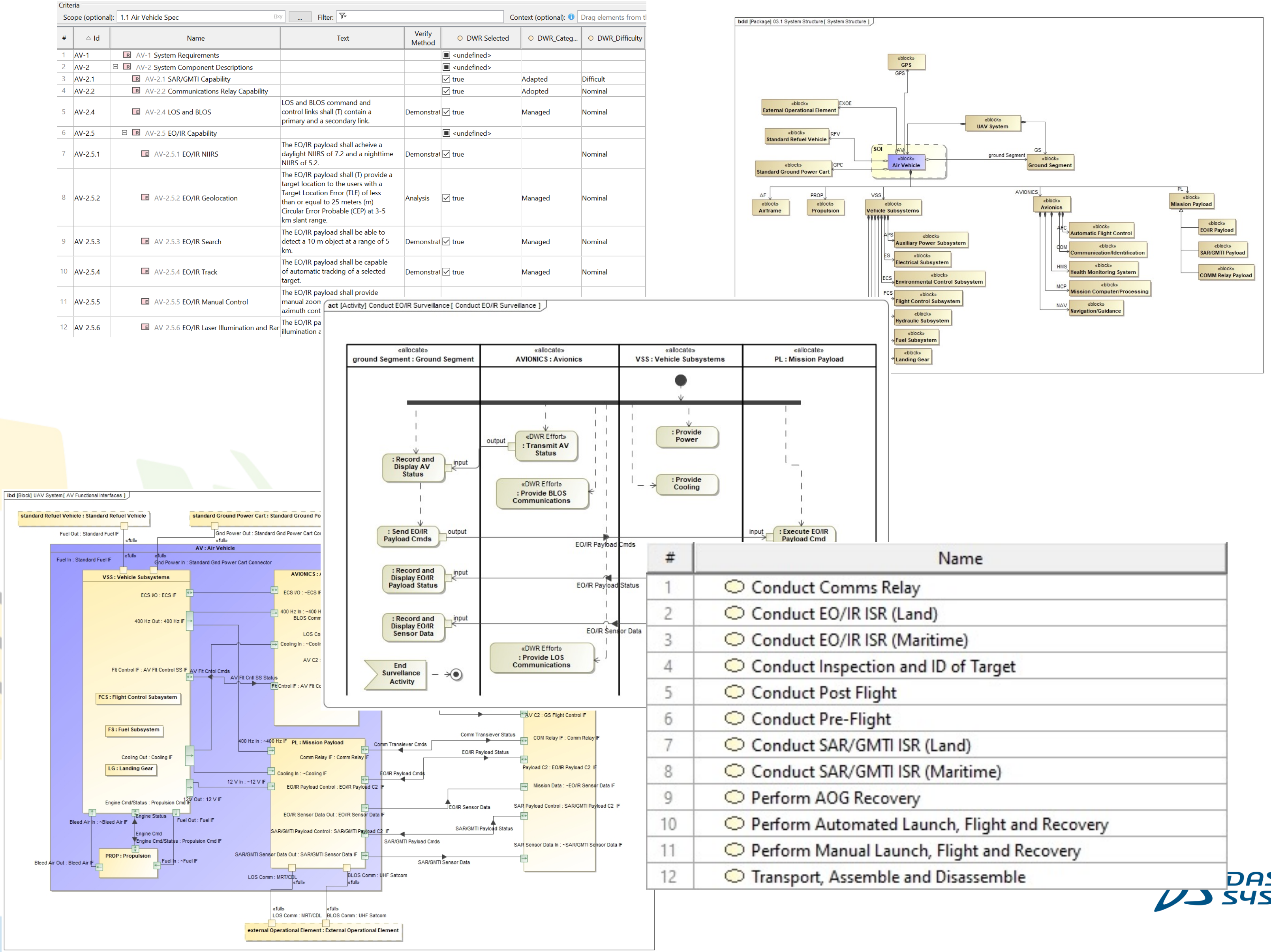
SO, WE MADE APPLYING SIZE DRIVER PROPERTIES EASY AND WE AUTOMATED THE COUNTING PROCESS!

BUT WHAT DO YOU COUNT?

DASSAULT SYSTEMES

UPDATING COSYSMO SIZE DRIVER COUNTING RULES FOR SYSML

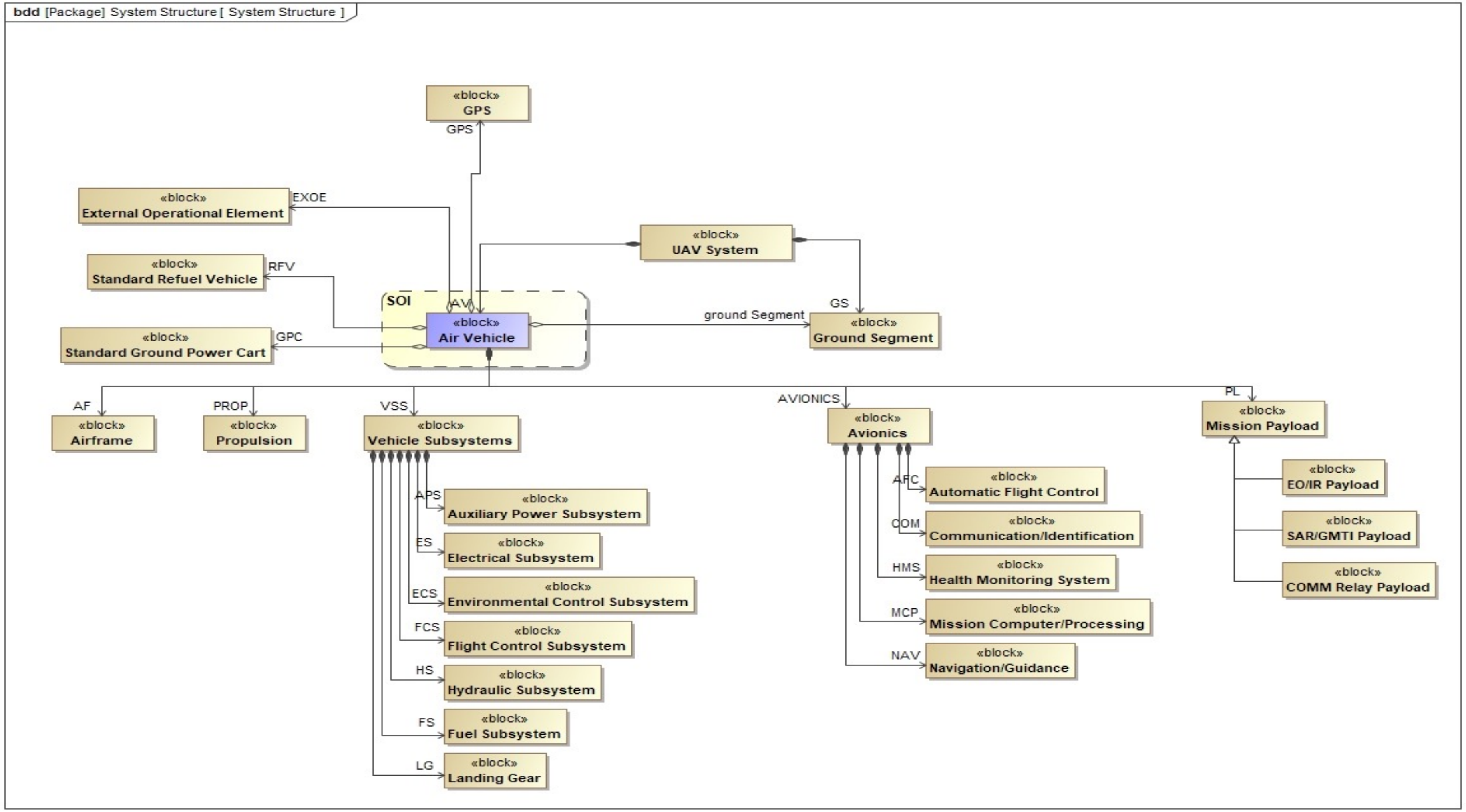
- While the mechanics of counting size drivers in a SysML model proved almost trivial, the question remains “which ones do you count?”
- In collaboration with Dr. Ricardo Valerdi and Dr. Gan Wang, we revised the size driver counting rules to be specific for counting in a SysML model.
- We selected the model elements that most closely corresponded to the original COSYSMO size driver definitions.



#	Name
1	Conduct Comms Relay
2	Conduct EO/IR ISR (Land)
3	Conduct EO/IR ISR (Maritime)
4	Conduct Inspection and ID of Target
5	Conduct Post Flight
6	Conduct Pre-Flight
7	Conduct SAR/GMTI ISR (Land)
8	Conduct SAR/GMTI ISR (Maritime)
9	Perform AOG Recovery
10	Perform Automated Launch, Flight and Recovery
11	Perform Manual Launch, Flight and Recovery
12	Transport, Assemble and Disassemble

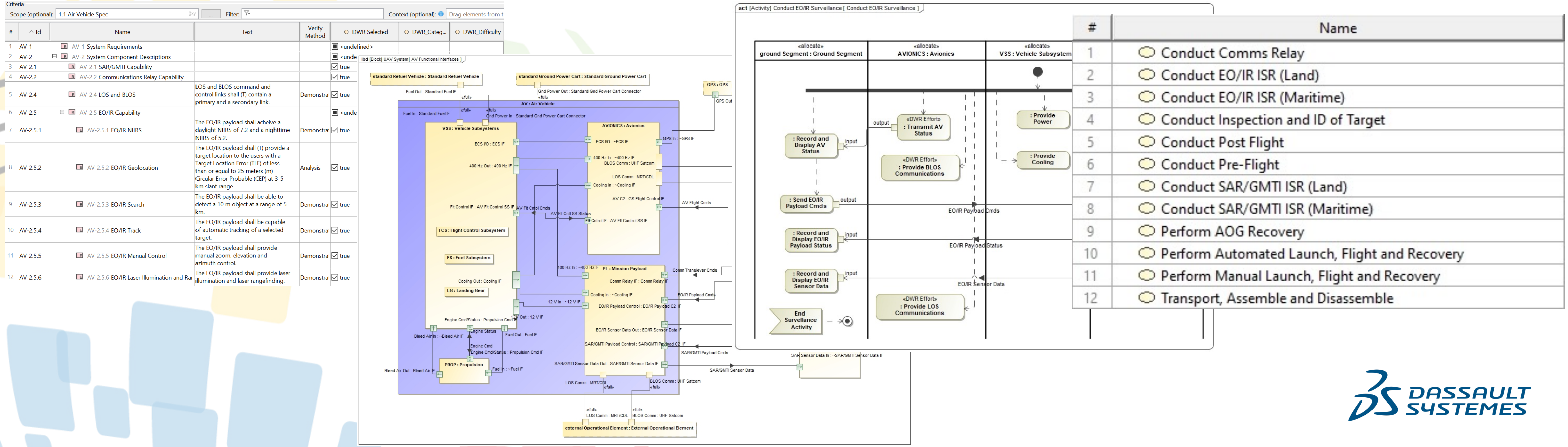
SAMPLE SYSTEM MODEL

The system of interest is the Air Vehicle segment of a larger UAV system.



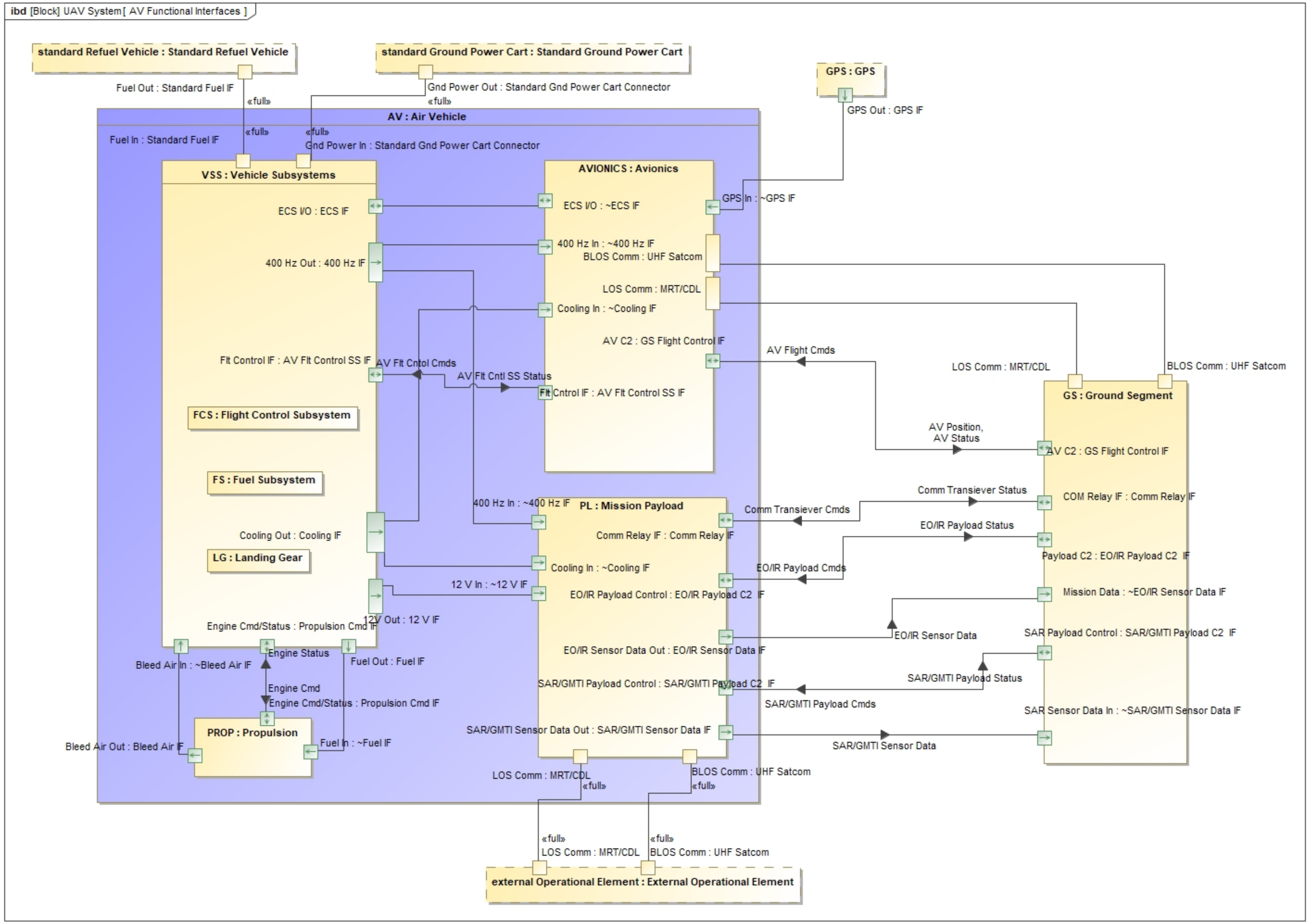
COUNTING SIZE DRIVERS IN SYSML

- **The paper provides full description of SysML size driver counting rules and required model content:**
  - **Requirements** - Requirements in the Systems Specification Package of the SOI.
  - **System Interfaces** - Connectors between the SOI and External Systems and between the Level 1 Subsystems in the System Internal Block Diagram.
  - **Algorithms** - Activities in Level 1 Subsystem Functions that will require development or modification of algorithms.
  - **Operational Scenarios** - Operational scenarios in CONOPS documents, enterprise architecture operational views, etc. (Create Use Cases as proxies for the operational scenarios.)



EXAMPLE: COUNTING SYSTEM INTERFACE (IF) IN SYSML

- Model all Level 1 internal and systems external interfaces with Ports, Interface Blocks and Connectors in the System Internal Block Diagram.
- Include Items Flows.



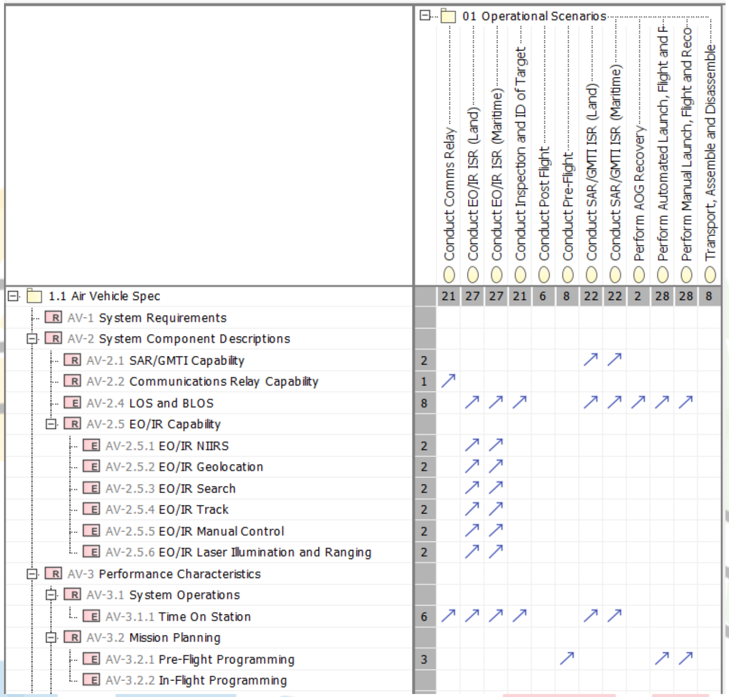
EXAMPLE: COUNTING SYSTEM INTERFACE (IF) IN SYSML

- In CAMEO, create a White Box ICD Table for the SOI.
- Each row in the table represents a Connector.
- The Port Features column shows the properties of the Interface Block that types the Port at each end of the connector.
- Add New Custom Columns for the DWR, DFR and Difficulty stereotypes.
- Reuse Levels and Difficulty are derived from the Port Features (i.e. Interface Blocks).

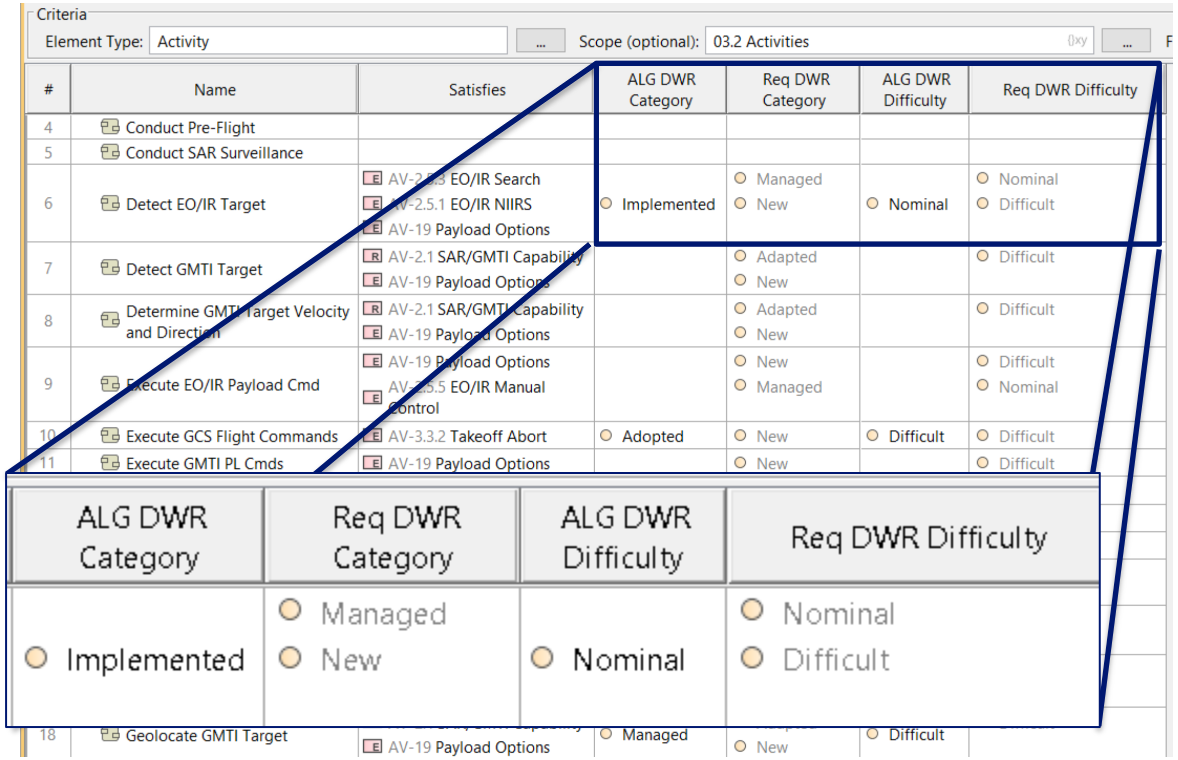
#	Part A	Port A	Port A Features	Item Flow	Port B	Port B Features	Part B	DWR Selected	DWR_Category	DWR_Difficulty
1	AF : Airframe	Engine Mount : Engine Mounti...			Engine Mount : ~Engine Moun...		PROP : Propul...	<undefined>		
2	PL : Mission Payload	Payload Mount : ~Payload Mo...			Payload Mount : Payload Mou...		PROP : Propul...	<undefined>		
3	VSS : Vehicle Subsystems	VSS Mount : ~VSS Mounting P...			VSS Mount : VSS Mounting Pr...		AF : Airframe	<undefined>		
4	VSS : Vehicle Subsystems	inout ECS I/O : ECS IF	in ECS Statu out ECS Cm		inout ECS I/O : ~ECS IF	out ECS Status : AV Flt Cntl SS St in ECS Cmds : Flight Control Cm	AVIONICS : A...	true	Modified	Nominal
5	VSS : Vehicle Subsystems	inout Flt Control IF : AV Flt Con...	inout AV Fli	AV Flt Cn AV Flt Cn	inout Flt Cntrol IF : AV Flt Cont...	inout AV Flight Cmds and Status	AVIONICS : A...	true	Modified	Difficult
6	VSS : Vehicle Subsystems	inout Engine Cmd/Status : Pro...	inout Engin	Engine Cr Engine St	inout Engine Cmd/Status : Pro...	inout Engine Commands : Engin	PROP : Propul...	true	Implemented	Nominal
7	VSS : Vehicle Subsystems	out 12V Out : 12 V IF	out 12 V Cu		in 12 V In : ~12 V IF	in 12 V Current : 12 V DC Current	PL : Mission P...	false		
8	VSS : Vehicle Subsystems	out 400 Hz Out : 400 Hz IF	out 400 Hz		in 400 Hz In : ~400 Hz IF	in 400 Hz Current : 400 Hz AC Cu	AVIONICS : A...	true	Adopted	Easy
9	VSS : Vehicle Subsystems	out 400 Hz Out : 400 Hz IF	out 400 Hz		in 400 Hz In : ~400 Hz IF	in 400 Hz Current : 400 Hz AC Cu	PL : Mission P...	true	Adopted	Easy
10	PROP : Propulsion	out Bleed Air Out : Bleed Air IF	out Cooling		in Bleed Air In : ~Bleed Air IF	in Cooling Air : Bleed Air	VSS : Vehicle ...	<undefined>		
11	VSS : Vehicle Subsystems	out Cooling Out : Cooling IF	out Cooling		in Cooling In : ~Cooling IF	in Cooling Air : Bleed Air	AVIONICS : A...	<undefined>		
12	VSS : Vehicle Subsystems	out Cooling Out : Cooling IF	out Cooling		in Cooling In : ~Cooling IF	in Cooling Air : Bleed Air	PL : Mission P...	<undefined>		
13	VSS : Vehicle Subsystems	out Fuel Out : Fuel IF	out Fuel : JF		in Fuel In : ~Fuel IF	in Fuel : JP4	PROP : Propul...	<undefined>		

USING ADVANCED QUERIES TO ENSURE ESTIMATE QUALITY

- While not part of the original COSYSMO counting rules, there are clear relationships between each of the size drivers.
- There are two fundamental relationships to review:
  - Traceability (Completeness and Correctness)
  - Consistency of DWR, DFR, and Difficulty assignments



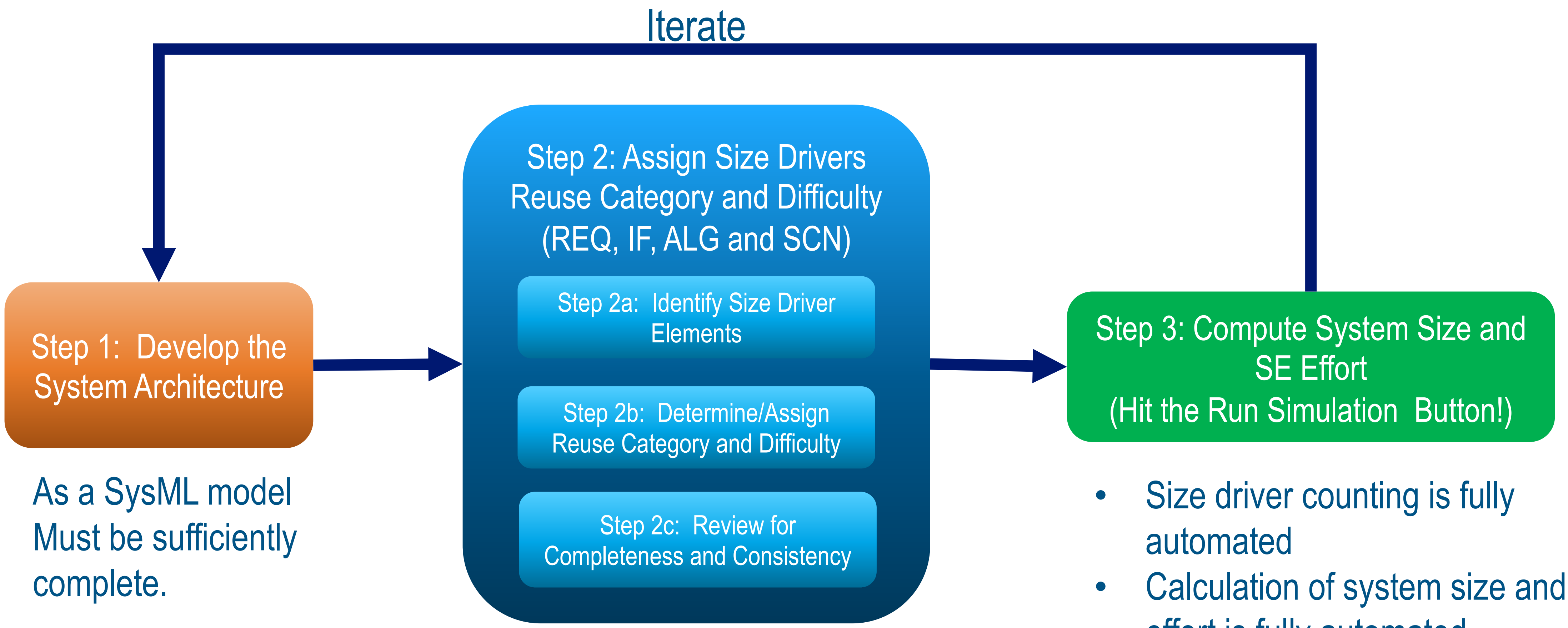
The matrix above shows traceability between Requirements and Operational Scenarios.



The table above shows DWR and Difficulty of Algorithms compared to the Requirements they Satisfy.

DASSAULT SYSTEMES

COSYSMO COST ESTIMATION PROCESS WITH MBSE



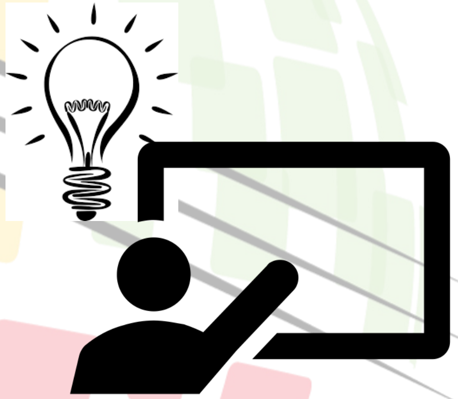
- As a SysML model
- Must be sufficiently complete.

- Simplified with less labor and more rigor
- Sizing properties are properties of the architecture model
- Cross cutting views enable quality review

- Size driver counting is fully automated
- Calculation of system size and effort is fully automated
- Results are saved within the architecture model or can be exported .

COSYMO with MBSE simplifies the labor intensive tasks and lets the Systems Engineer focus on the important things!

- **Tasks for the Systems Engineer**
  - Design the System
  - Identify Sizing Elements
  - Determine Reuse Category and Complexity
  - Analyze Results



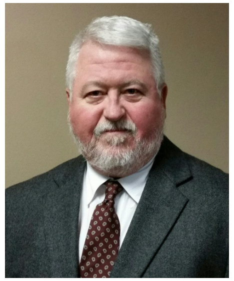
- **Tasks for MSBE Toolset**
  - Maintain Sizing Data as part of the System Architecture
  - Provide efficient User Interface to apply Sizing Parameters
  - Automates Counting
  - Calculates System Size and SE Effort
  - Provide Cross Cutting Views for Analysis

SUMMARY, CONCLUSION AND NEXT STEPS

- The Tool and the Method
  - The COSYSMO profile simplifies the assignment of size driver properties and automates the counting and calculations.
  - The rigor and structure of SysML, along with sound modeling practices, enable a more repeatable counting process.
  - The semantic relationships between model elements provides strong traceability between size driver elements.
  - Advanced queries in tables and matrices enable a review of completeness, correctness and consistency of size driver assignments.
- The SE Process
  - The convergence of COSYSMO with MBSE makes the sizing activity a responsibility of the system engineer/architect.
  - The size driver properties become properties of the architecture model as opposed to data maintained separately and disconnected from the system architecture.
  - Requires sufficient model content and fidelity and some specific modeling patterns.
  - Enables the project to perform quick cost trades earlier in the lifecycle.
  - The ability to access the sizing data for each size driver, for each DWR/DFR category, allows the cost analyst to review and validate the size estimate.
- Next Steps
  - Developing cost model calibration data for SysML Models
  - Industry use and feedback to validate new counting rules
  - Case Study with an Actual SOI

Questions?

About the Authors



Barry Papke. Barry is the MBSE Special Project Lead for CATIA Magic. He has thirty-five years of systems engineering and operations analysis experience in the aerospace and defense industry across the entire systems engineering life cycle from concept development through integration, test and post-delivery support. Throughout his ca-reer, he has been actively involved in application of model-based methods including requirements management, enterprise architecture, cost estimation, system design, and operations analysis. He is a member of the INCOSE Agile and Security Working Groups and the MBSE Initiative.

Ricardo Valerdi. Ricardo is a Professor and Head of the Department of Systems & Industrial Engineering at the University of Arizona. He is the original de-veloper of the COSYSMO model and teaches courses in systems engineering and sports analytics. He is a Fellow of INCOSE and received his Ph.D. from the University of Southern California.

Gan Wang. Gan is Vice President for Systems Engineering Ecosystem at Dassault Systèmes. He leads the engagement and collaboration with the systems engineering communities, including industry associations and standardization bodies. His specialty spans systems engineering processes, architecture, design reuse methodology, cost estimating and analysis, and decision support methods. Gan is an ESEP and a Fellow of INCOSE and received his Ph.D. from the University of Virginia.

Sean Densford. Sean holds an Associates in Music Education, Bachelor's in Electrical Engineering, and a Master's in Systems Engineering. He has worked over a decade in Electromagnetics, Special Test Equipment, Test and Model Based Systems Engineering. He is an OMG Certified Systems Modeling Language Professional.

34th Annual INCOSE international symposium

hybrid event

Dublin, Ireland
July 2 - 6, 2024

www.incose.org/symp2024
#INCOSEIS