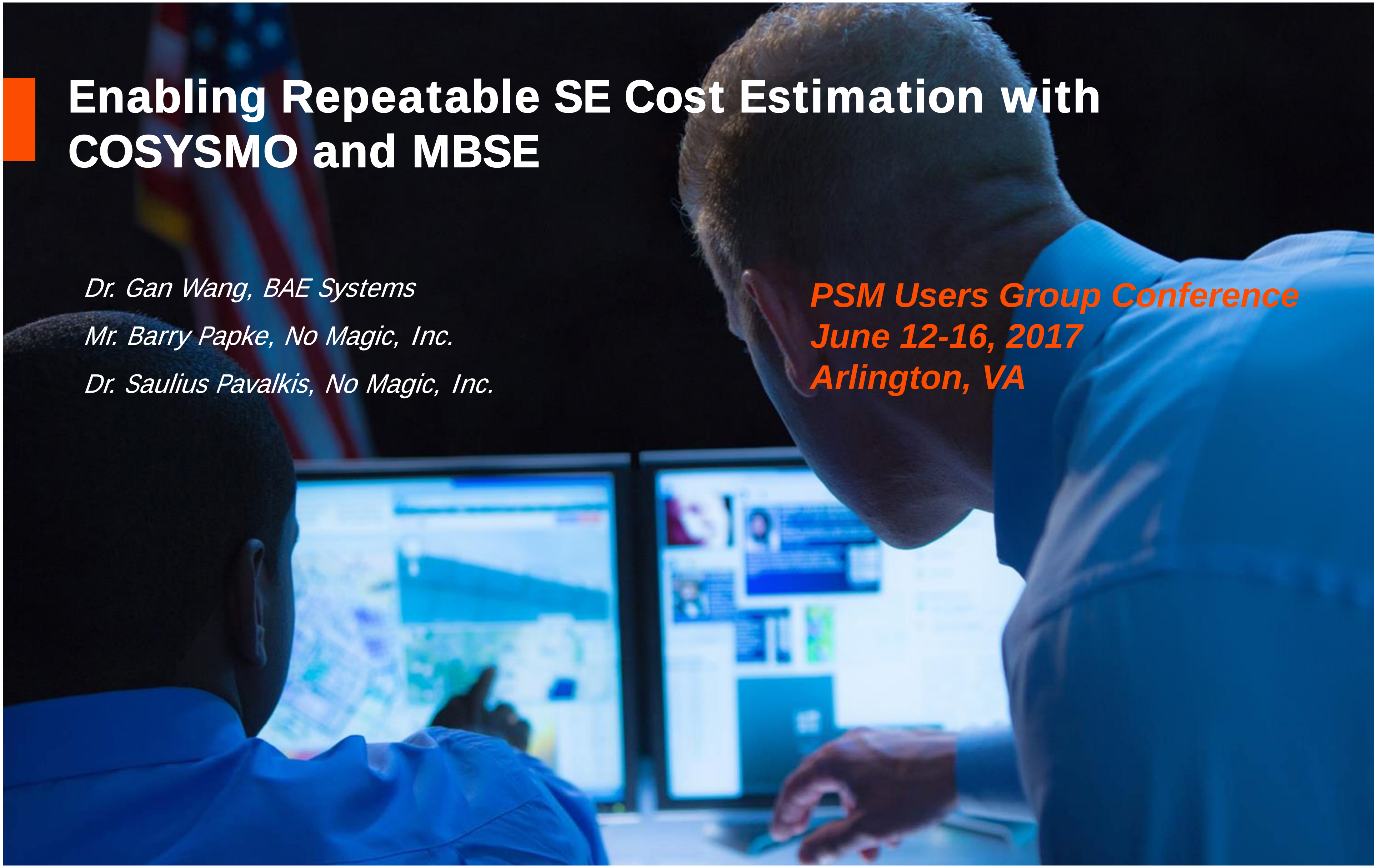


Enabling Repeatable SE Cost Estimation with COSYSMO and MBSE

Dr. Gan Wang, BAE Systems

Mr. Barry Papke, No Magic, Inc.

Dr. Saulius Pavalkis, No Magic, Inc.

PSM Users Group Conference
June 12-16, 2017
Arlington, VA

No Magic

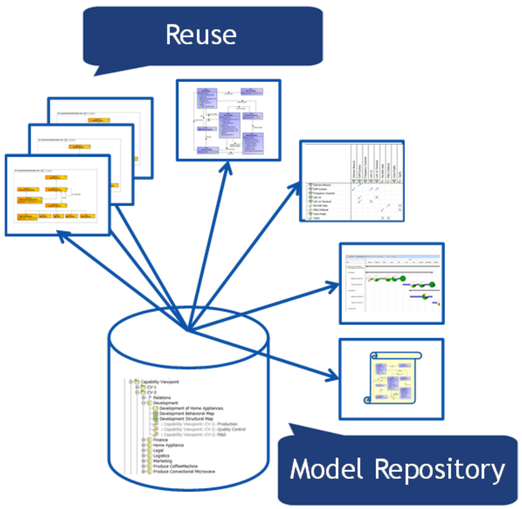
BAE SYSTEMS
INSPIRED WORK

Introduction

- Parametric cost estimating should be a natural extension of Model-Based Systems Engineering
- COSYSMO can be seamlessly integrated into the SysML-based modeling environment
- A use case: integrating COSYSMO with MagicDraw
- Benefits and future work

What do we mean when we say MBSE?

Model Based Systems Engineering



Language

Methodology

Tool

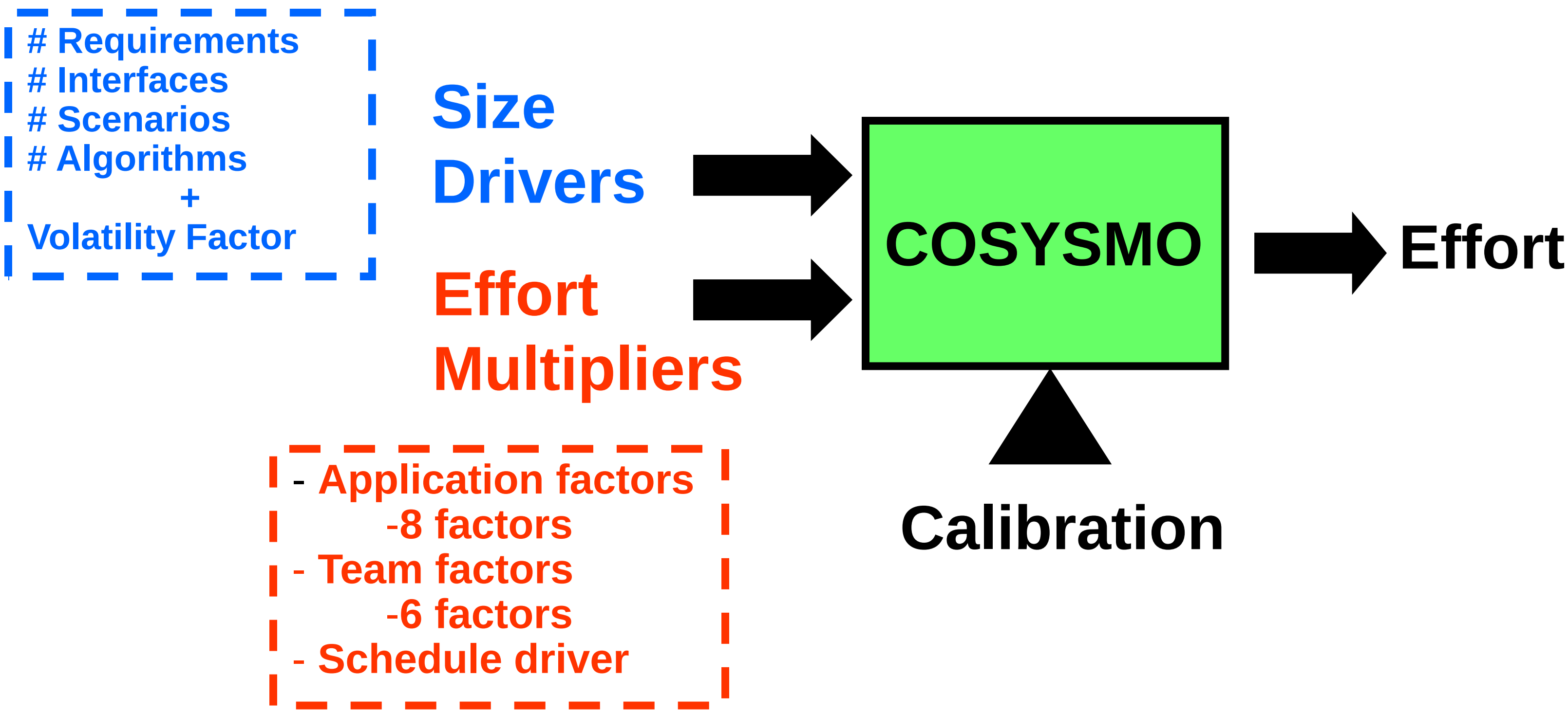
- Model/data repository provides a single source of truth!
  - Cost model is another model
  - Estimate is another piece of data within repository

No Magic

BAE SYSTEMS
INSPIRED WORK

COSYSMO – Parametric Cost Model for Systems

4 Size Drivers and 14 Cost Drivers



Basic Cost Estimating Relationship (CER)

$$PH_{NS} = A \cdot SS^{E} \cdot CEM$$

- PH_{NS} = systems engineering effort in person-hours under nominal schedule
- SS = system size, determined by the four size drivers
- CEM = composite effort multiplier, determined by the fourteen cost drivers
- A = productivity constant, typically derived from historical project data
- E = nonlinearity for the productivity curve, representing a diseconomy of scale

Four Size Drivers

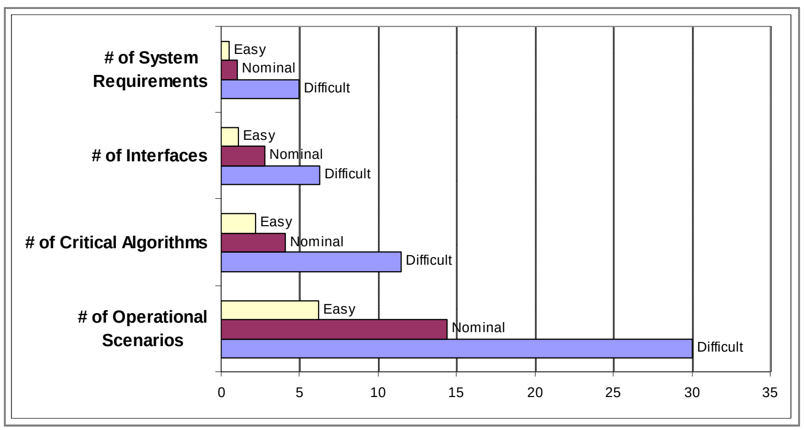
1. Number of System Requirements
2. Number of System Interfaces
3. Number of Critical Algorithms
4. Number of Operational Scenarios

Each weighted by: 1) Levels of complexity; 2) Degrees of reuse

Driver Counting / Classification Rules

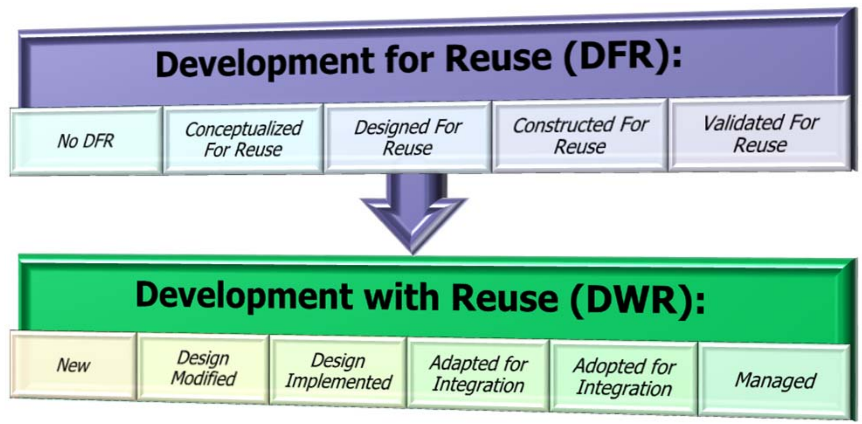
Level of Complexity

- "Easy"
- "Nominal"
- "Difficult"



Degrees of Reuse

- "Generalized Reuse Framework"

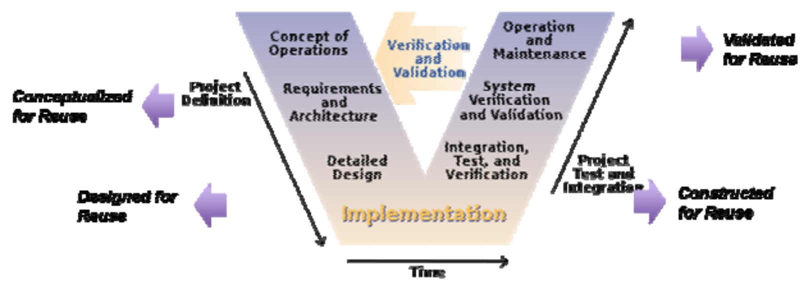


No Magic

BAE SYSTEMS
INSPIRED WORK

Development for Reuse (DFR) Process

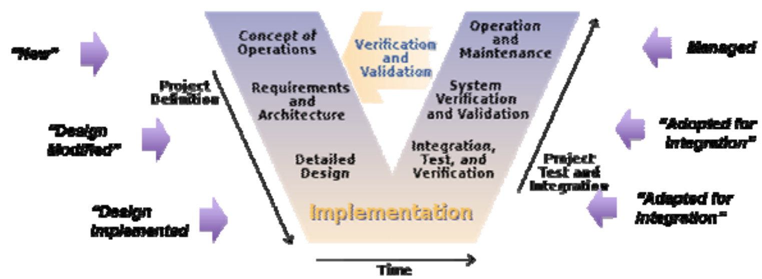
Category	Required Activities	Delivering (for reuse)
No DFR	• N/A	• Little / accidental
Conceptualized For Reuse	• Analysis • Architecture development	• Functional & Logical architecture
Designed For Reuse	• Analysis • Architecture • System design	• Physical design of system
Constructed For Reuse	• Design • Build • Unit test	• Implemented system or component
Validated For Reuse	• Design • Build • System test	• Validated and deployed system or component



Activity-based Model

Development with Reuse (DWR) Process

Category	Required Activities	Leveraging (existing)
New	• Develop anew • Revamp of existing	• New concept
Design Modified	• Design & implement from logical architecture	• Logical/functional architecture
Design Implemented	• Implement from design • Build-to-print	• Physical design of system
Adapted for Integration	• Adapt from existing implementation • Tailor to integrate	• Built system or component
Adopted for Integration	• Integrate per instructions • V&V testing	• Build system or component
Managed	• Manage • Inspect	• Integrated & verified system or component



Activity-based Model

COSYSMO 3.0 with the Generalized Reuse Framework

Total Project Effort = DWR Effort + DFR Effort

$$PM_{DWR+DFR} = A_1 \cdot \left[\sum_k \left(\sum_r w_r \left(w_{e,k}\Phi_{e,k} + w_{n,k}\Phi_{n,k} + w_{d,k}\Phi_{d,k} \right) \right) \right]^{E_1} \cdot CEM_1$$

$$+ A_2 \cdot \left[\sum_k \left(\sum_q w_q \left(w_{e,k}\Psi_{e,k} + w_{n,k}\Psi_{n,k} + w_{d,k}\Psi_{d,k} \right) \right) \right]^{E_2} \cdot CEM_2$$

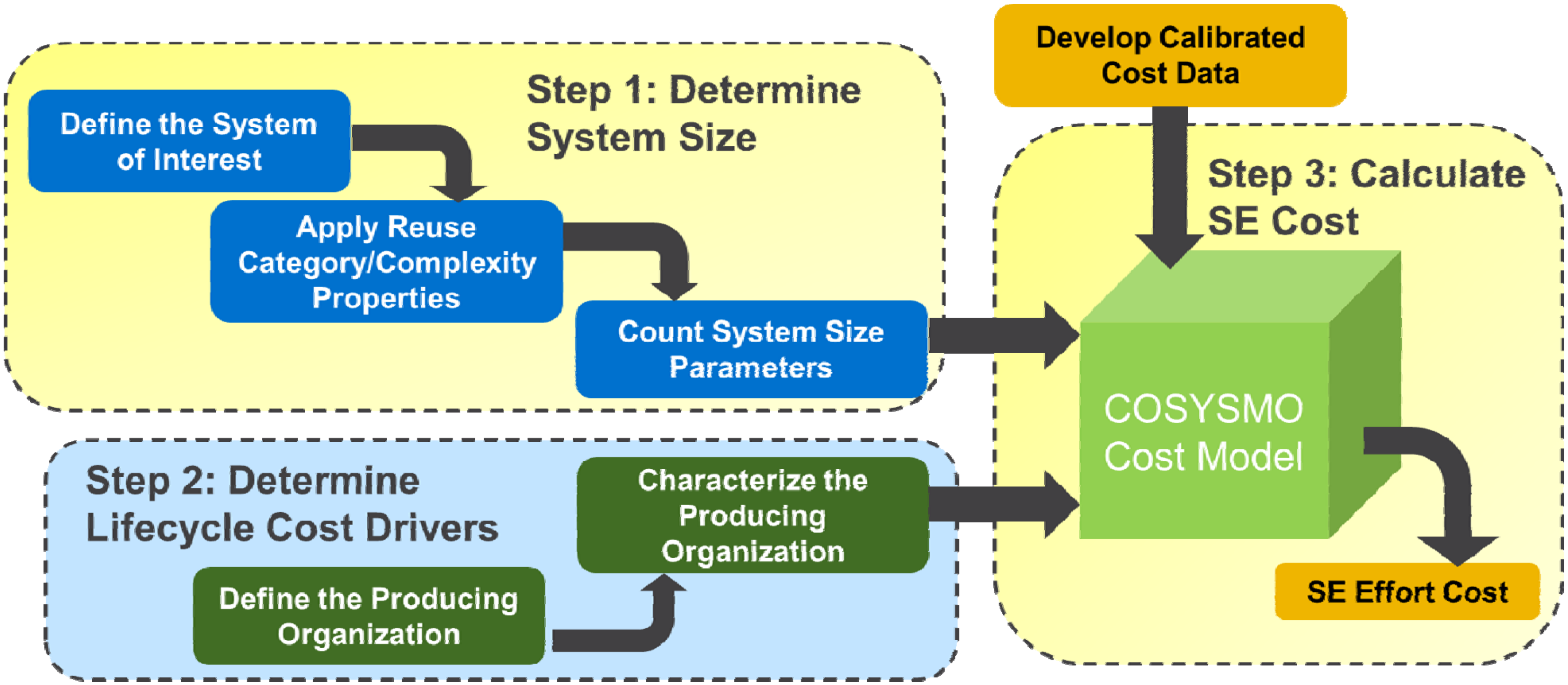
Where:

- $\mathbf{PM_{DWR}}$ = effort in Person Hours/Months (Nominal Schedule)
- $\mathbf{A_1}$ = DWR constant derived from historical project data
- **k** = {REQ, IF, ALG, SCN}
- **r** = {***New, D. Modified, D. Implemented, Adapted for Int., Adopted for Int., Managed***}
- w_r = weight for defined levels of size driver reuse
- w_x = weight for "easy", "nominal", or "difficult" size driver
- Φ_x = quantity of "k" size driver
- $\mathbf{E_1}$ = represents diseconomy of scale in DWR
- $\mathbf{CEM_1}$= composite effort multiplier for DWR

Where:

- $\mathbf{PM_{DFR}}$ = effort in Person Hours/Months (Nominal Schedule)
- $\mathbf{A_2}$ = DFR constant derived from historical project data
- **k** = {REQ, IF, ALG, SCN}
- **q** = {***No DFR, Conceptualized, Designed, Constructed, Validated***}
- w_q = weight for defined levels of size driver reuse
- w_x = weight for "easy", "nominal", or "difficult" size driver
- Φ_x = quantity of "k" size driver
- $\mathbf{E_2}$ = represents diseconomy of scale in DFR
- $\mathbf{CEM_2}$ = composite effort multiplier for DFR

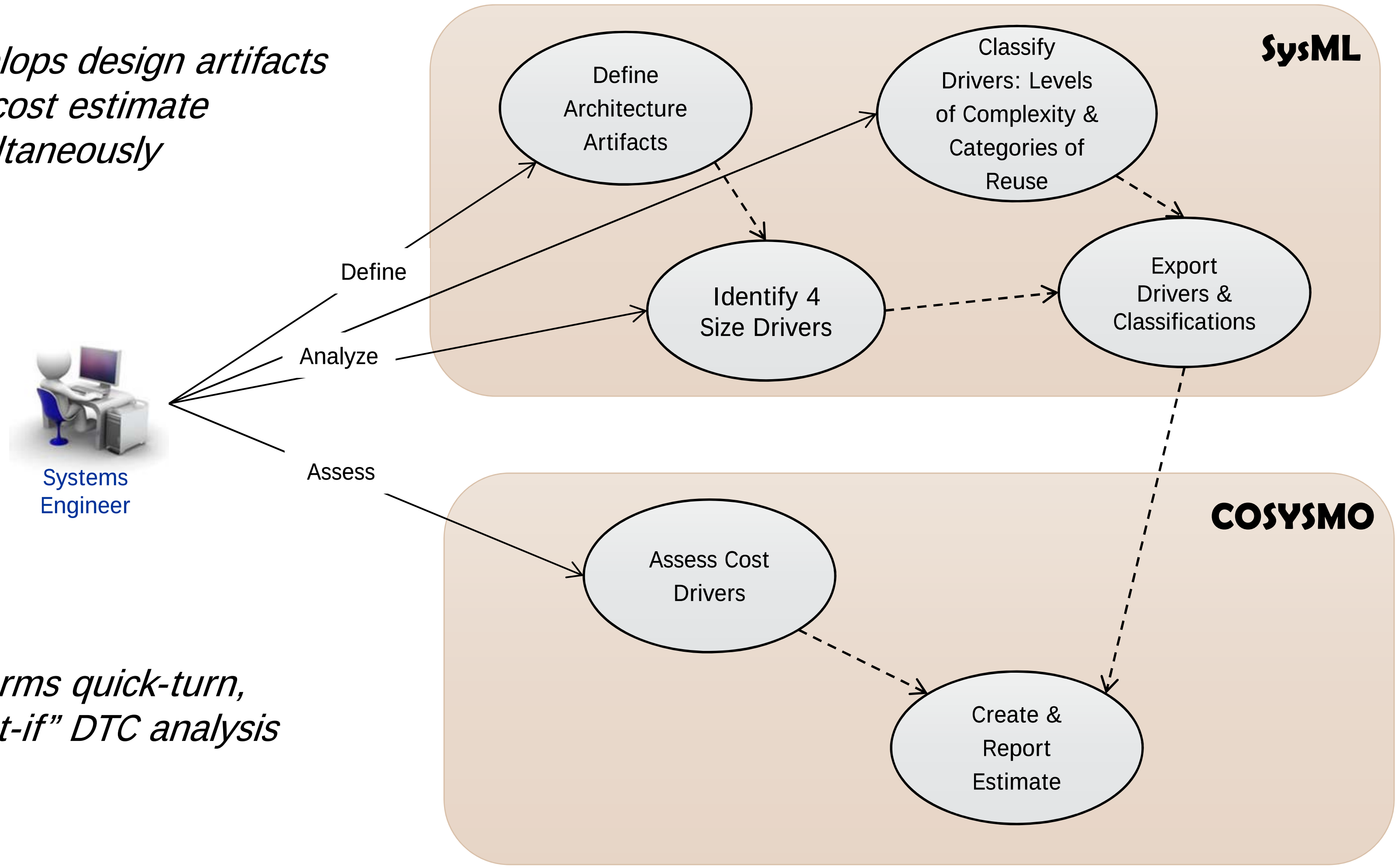
COSYSMO Cost Modeling Process



These steps can be naturally achieved in an MBSE modeling environment…

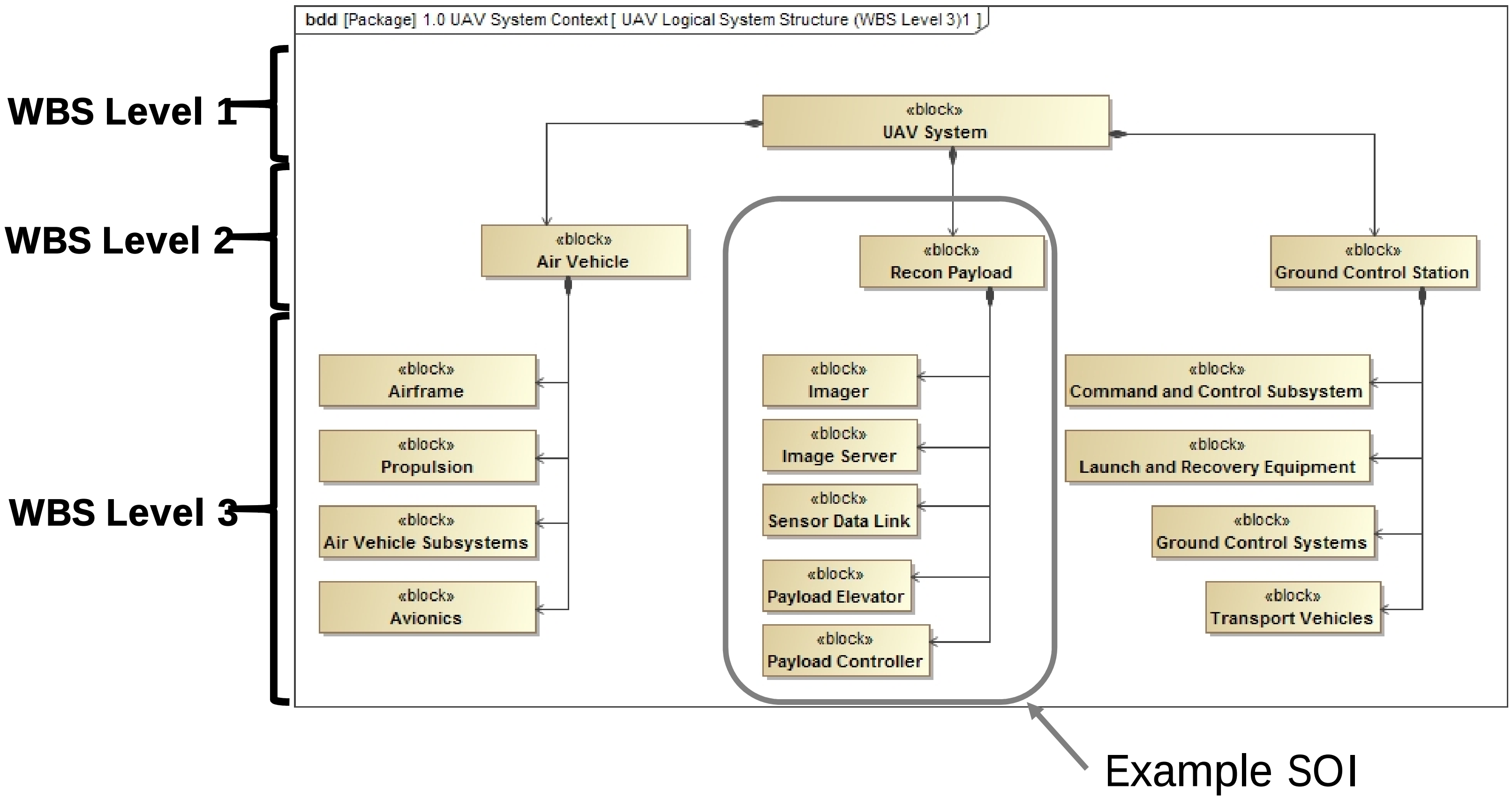
An Integrated COSYSMO-SysML Modeling Use Case

- *Develops design artifacts and cost estimate simultaneously*



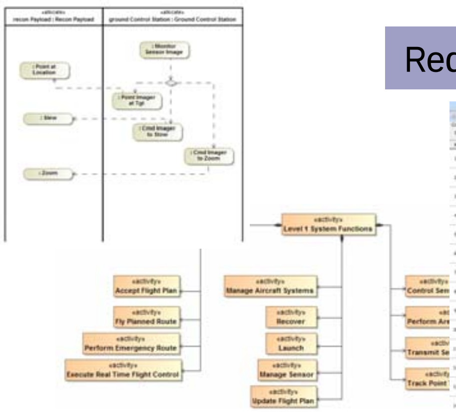
- *Performs quick-turn, "what-if" DTC analysis*

Define Scope of the "System of Interest"

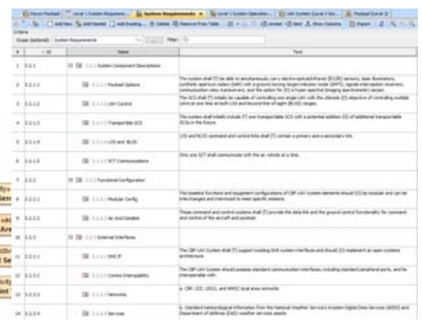


Identify Size Drivers While Defining System Architecture

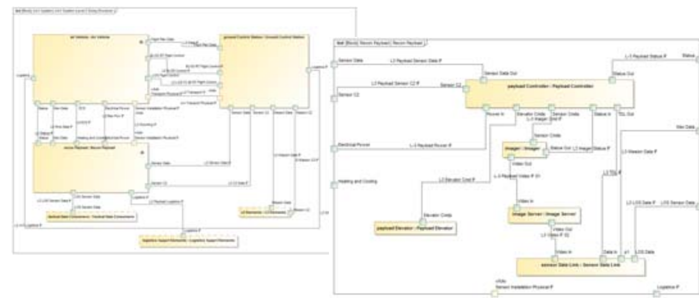
Algorithms



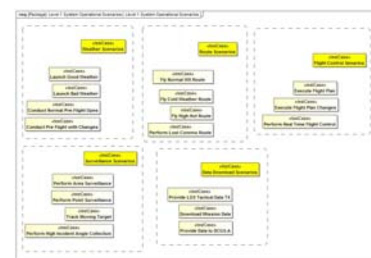
Requirements



Interfaces



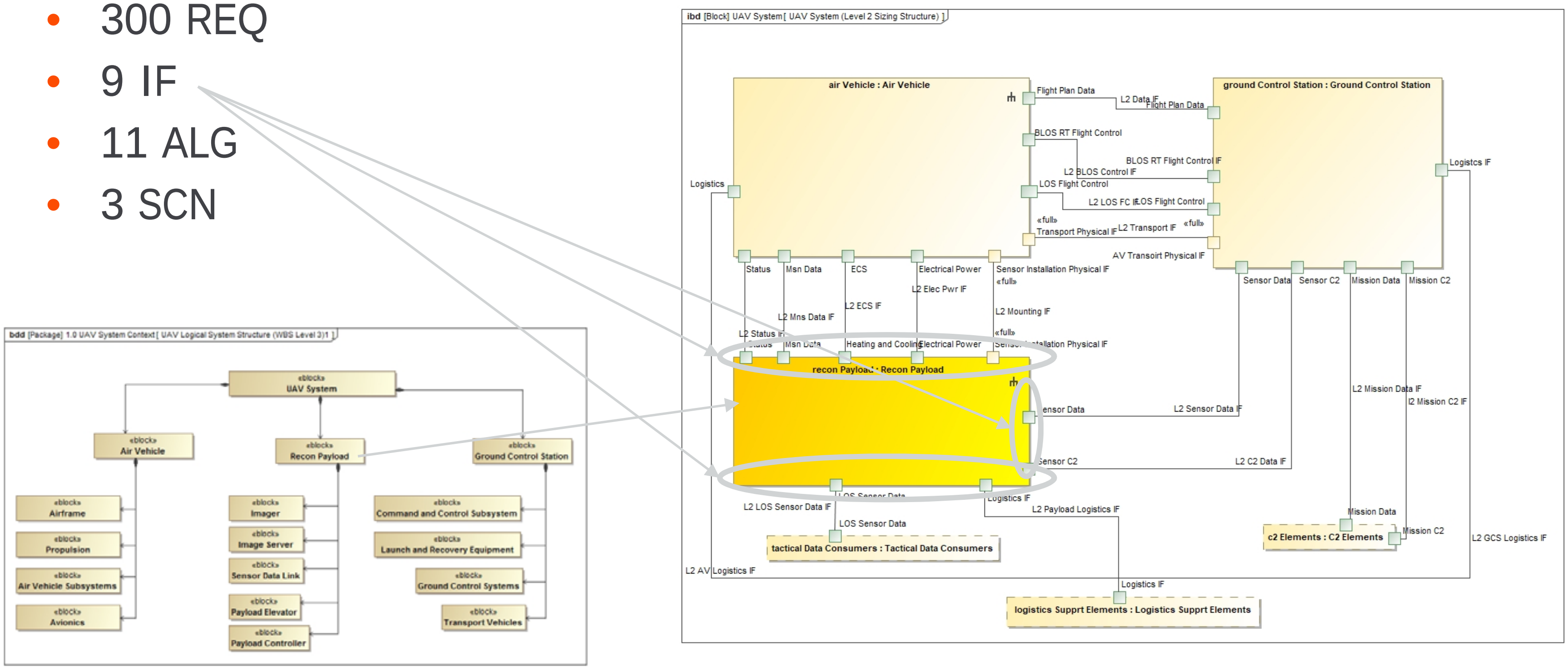
Scenarios



This is the natural systems engineering design process.

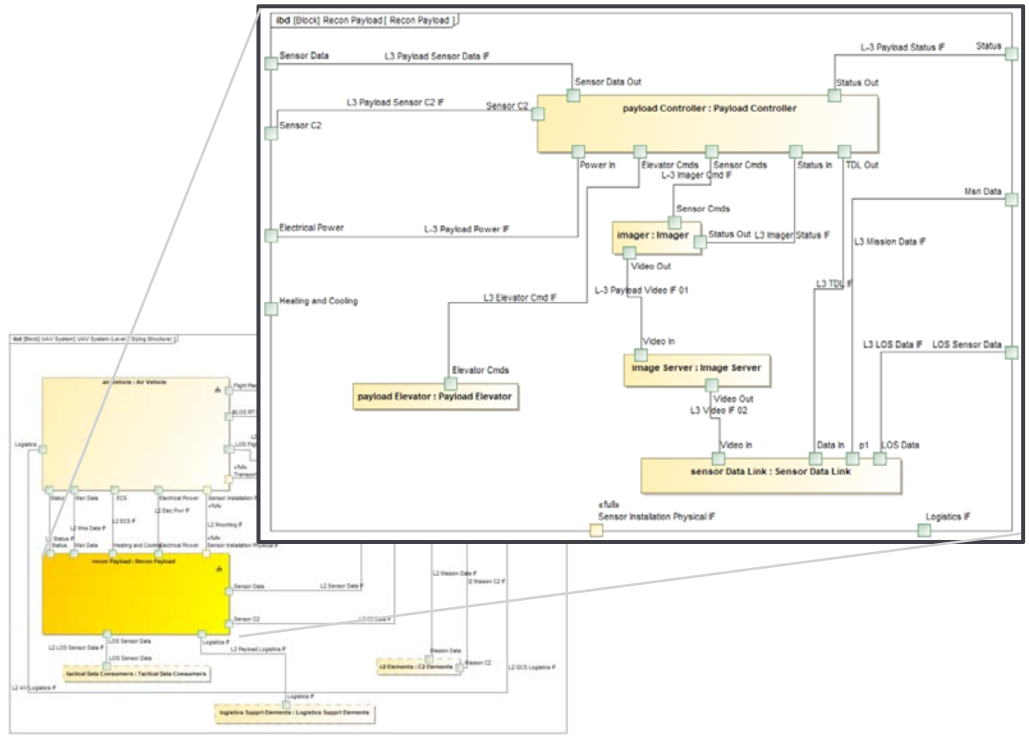
Maintain the Levels of Abstraction, Consistency and Traceability

- Example SOI Context Level
  - 300 REQ
  - 9 IF
  - 11 ALG
  - 3 SCN



Maintain the Levels of Abstraction, Consistency and Traceability

- Example SOI Level 1
  - Payload Controller
    - 8 IF
    - 250 REQ
    - 8 ALG
    - 1 SCN
  - Payload Elevator
    - 2 IF
    - 100 REQ
    - 2 ALG
    - 2 SCN
  - Imager
    - 3 IF
    - 300 REQ
    - 6 ALG
    - 3 SCN
  - Image Server
    - .....
  - Sensor Data Link
    - .....



The level of abstraction chosen for the sizing estimate directly affects the quantity of sizing elements.

The key is to maintain consistency with the approach used across projects and with that used for calibration.

No Magic

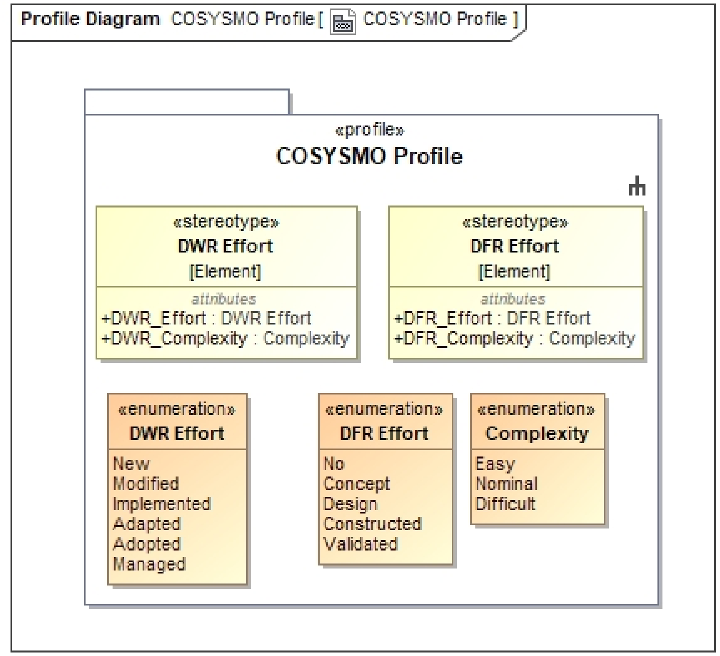
BAE SYSTEMS
INSPIRED WORK

Applying Reuse Categories and Levels of Complexity

- Reuse category and complexity are simply properties of the model elements being considered
- There are multiple methods to assign properties to model elements in SysML
- Different tools provide different capabilities for defining model element properties and calculating model metrics
- The approach shown in this presentation was selected because it took advantage of advanced tool features in the tool (MagicDraw™) that resulted in the lowest total effort to assign and count sizing elements

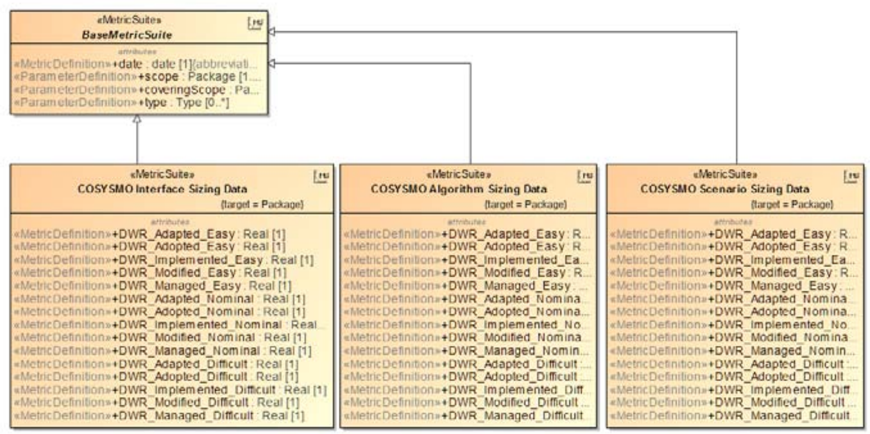
Create COSYSMO Profile and Metrics Rules (Non-Recurring)

- COSYSMO sizing elements are created as new stereotype elements:
  - Stereotypes are a core SysML feature
  - Defined in a Profile Package
- Metrics rules and measurements are a tool specific feature:
  - Multiple methods exist to determine the numbers of each sizing element
- The COSYSMO profile and metrics set are created once as a separate project and reused:
  - The are applied (reused) on each new system project when generating sizing estimates



5 DFR Reuse Categories
6 DWR Reuse Categories
3 Levels of Complexity for each (Easy, Nominal, Difficult)
4 Sizing Elements Types (REQ, ALG, SCN and IF)

A potential of 132 individual pieces of sizing data:
(5+6) x 3 x 4 = 132



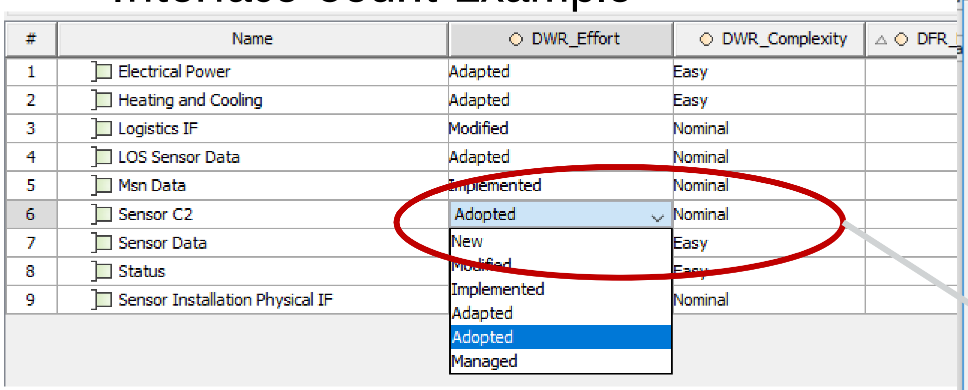
No Magic

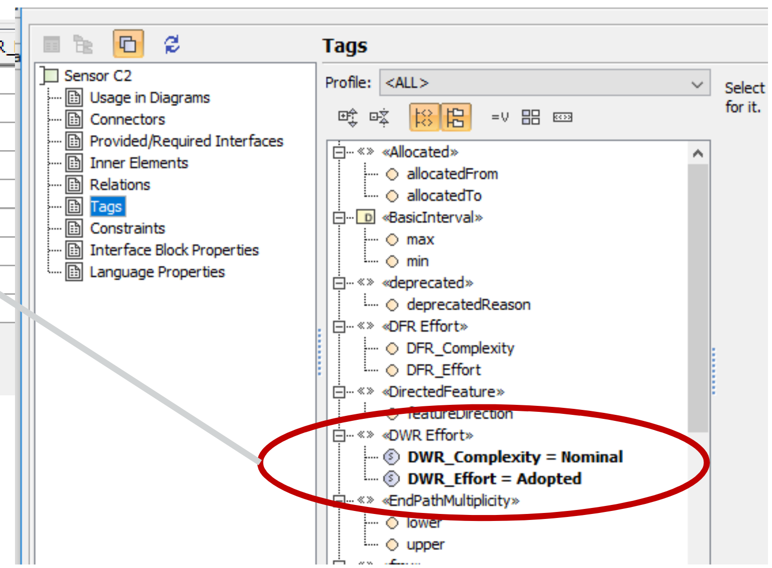
BAE SYSTEMS
INSPIRED WORK

Apply Reuse Categories

- Application of re-use category and complexity is a trivial effort:
  - Create generic table and select element type and package scope.
  - Select the new stereotypes from the “show columns” pull-down.
  - Select the cell in the table and apply the reuse category and complexity.
  - Once selected, the tool applies the properties as tag values to the model element.

Interface Count Example





Properties selected in the table are actual properties of the model element.

Apply Reuse Categories

- The process of applying re-use categories and complexity level is repeated for each of the four sizing categories (REQ, IF, ALG and SCN)
- If requirements are managed in an external requirements management tool, sizing metrics for requirements can be easily calculated by applying properties in that tool, and using spreadsheets or other applications to sum each category.

Criteria

Element Type: Activity | Scope (optional): L2 System Functions

#	Name	Allocated To	DWR_Effort	DWR_Complexity
1	enter area stare mode	Recon Payload	Adopted	Nominal
2	Monitor Sensor Status	Recon Payload	Adapted	Nominal
3	perform gyro alignment	Recon Payload	Adopted	Nominal
4	Point at Location	Recon Payload	Adapted	Nominal
5	record metadata	Recon Payload	Implemented	Nominal
6	record streaming image	Recon Payload	Adopted	Easy
7	Report Sensor Status	Recon Payload	Adapted	Nominal
8	Slew	Recon Payload	Adapted	Nominal
9	Store Search Plan	Recon Payload	Implemented	Nominal
10	Stow for landing	Recon Payload	Adopted	Nominal
11				

Criteria

Element Type: Test Case | Scope (optional): System Operational Scenarios

#	Name	DWR_Effort	DWR_Complexity
1	Payload Stow and Deploy Scenarios	Adapted	Nominal
2	Sensor Visibility Scenarios	Adapted	Nominal
3	Target Tracking Scenarios	Adapted	Difficult

Criteria

Element Type: Full Port, Proxy Port | Scope (optional): Recon Payload

#	Name	DWR_Effort	DWR_Complexity
1	Electrical Power	Adapted	Easy
2	Heating and Cooling	Adapted	Easy
3	Heating and Cooling	Modified	Nominal
4	LOS Sensor Data	Adapted	Nominal
5	Msn Data	Implemented	Nominal
6	Sensor C2	Adopted	Nominal
7	Sensor Data	Adopted	Easy
8	Status	Adopted	Easy
9	Sensor Installation Physical IF	Modified	Nominal

Run the Metrics Tool and Calculate Sizing Element Counts

- Run the metrics tool to generate a metrics table with counts for each reuse category/complexity combination.
  - Separate tables are create for each sizing element type (REQ, IF, ALG and SCN)
  - Metrics tables can be exported to Excel for input to the cost model.
- Depending on the tool, other methods may be available to determine sizing counts:
  - Export of element tables and count externally.

UAV System Model Metric

#	Date	DWR_ Adapted_ Easy	DWR_ Adopted_ Easy	DWR_ Implemented_ Easy	DWR_ Modified_ Easy	DWR_ Managed_ Easy	DWR_ Adapted_ Nominal	DWR_ Adopted_ Nominal	DWR_ Implemented_ Nominal	DWR_ Modified_ Nominal	DWR_ Managed_ Nominal	DWR_ Adapted_ Difficult	DWR_ Adopted_ Difficult	DWR_ Implemented_ Difficult	DWR_ Modified_ Difficult	DWR_ Managed_ Difficult
1	2017.05.26 17.30	0	1	0	0	0	4	3	2	0	1	0	0	0	0	0
2	2017.05.26 17.30	0	1	0	0	0	4	3	2	0	1	0	0	0	0	0
3	2017.05.26 17.31	0	1	0	0	0	4	2	1	0	1	1	0	0	0	0

Criteria

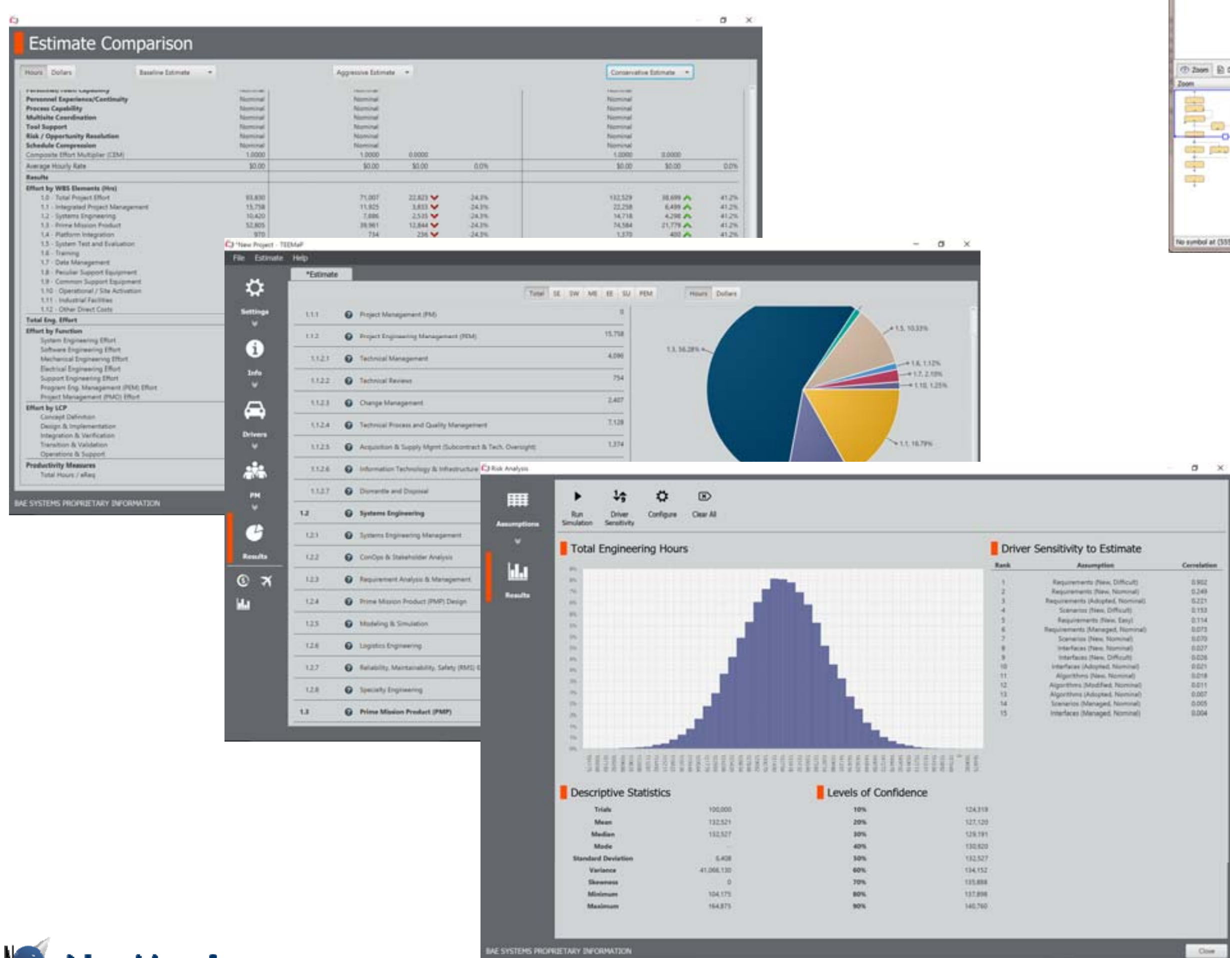
Metric Suite: COSYSMO Algorithm Sizing Data … Scope (optional): Drag elements from the Model Brow … Filter:

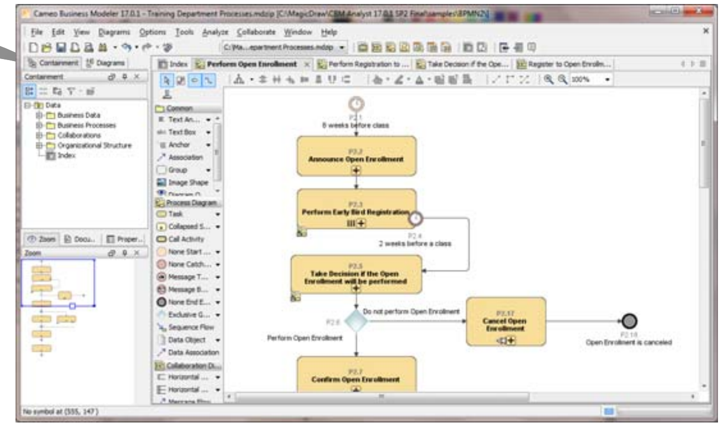
#	Date	Documentation	DWR_ Adapted_ Easy	DWR_ Adopted_ Easy	DWR_ Implemented_ Easy	DWR_ Modified_ Easy	DWR_ Managed_ Easy	DWR_ Adapted_ Nominal	
1	2017.05.26 17.30	Estimate for initial ROM	0.00	1.00	0.00	0.00	0.00	4.00	3.00	2.
2	2017.05.26 17.30	Updated based on changes from XYZ	0.00	1.00	0.00	0.00	0.00	4.00	3.00	2.
3	2017.05.26 17.31	Updated for submittal gate review	0.00	1.00	0.00	0.00	0.00	4.00	2.00	1.

The metrics tables shows the history of metrics calculations.

A documentation column can be added to record rational and other data for each metric calculation.

Analyze Estimate Result





Rapid "what-if" analysis roundtrips

Conclusion

- Parametric cost estimating can be seamlessly integrated into Model-Based Systems Engineering:
  - Sizing data becomes a property of each model element
  - The model repository provides a single source of truth
  - Libraries can be created to maintain and revise reuse category and complexity consistent with changes in project lifecycles
  - Rigor of MBSE with SysML and the automated features of the tools provide a practical and efficient mechanism to develop sizing models of legacy systems in order to establish a calibration baseline
- Feasibility of COSYMO integrated into a SysML-based modeling environment demonstrated with MagicDraw:
  - All methods shown are existing features of the SysML language or the toolset
- Feasibility of tool-tool interfaces enabled by reusable profiles and templates:
  - Create once, reused again and again
  - Can be applied to new SysML models

Key Benefits

- Seamless integration of cost estimation with the system design and modeling process:
  - Providing consistency and traceability.
    - Sizing data becomes a property of the model element.
  - Enabling rapid-turnaround “what-if” architecture trade analysis
  - Promoting Design-To-Cost.
  - Enabling design reuse.
  - Economic impact early in system lifecycle and an integral part of architecture
  - Culture change for systems engineers:
    - Shift of mindset and right behavior in design
    - Systems engineering for economic goals
  - Application of Model Based Systems Engineering – LET THE TOOLS DO THE WORK

Next Steps

- Design Patters and Guidelines for Sizing Estimation
  - Develop guidelines and standards for levels of abstraction, design patters and identification of model elements that should be included or excluded from the sizing counts.
- Design Patters and Guidelines for Developing Calibration Data
  - Develop guidelines and standards for modeling existing/delivered systems with known cost data for calibration of cost model equations.
- Tool-Tool Data Exchange
  - Develop an export/report format that can be used as direct input to the calibrated cost estimation tools.
- Commercial Plugin
  - Develop a tool plug-in or template that can be obtained as an off-the-shelf module with all required COSYSMO model stereotypes, pre-defined counting metrics and export formats to further automate the sizing collection and estimation process.

No Magic

BAE SYSTEMS
INSPIRED WORK

Thank You

Dr. Gan Wang, BAE Systems

Mr. Barry Papke, No Magic, Inc.

Dr. Saulius Pavalkis, No Magic, Inc.