NAVAL
POSTGRADUATE
SCHOOL

System Cost Modeling and SysML Integration in Model-Based Systems Engineering

Ray Madachy, Naval Postgraduate School
David Jacques, Air Force Institute of Technology

INCOSE San Diego Chapter Meeting
January 18, 2017

Agenda

- Introduction
- Cost Modeling Overview
- SysML Integration
- Case Study: Remote Targeting System UAS
- Case Study: TBM Identification and Elimination: 3-Tier UAS SoS
- Conclusions and Future Work

Introduction

- System Qualities Ontology, Tradespace and Affordability (SQOTA) Project with DoD Systems Engineering Research Center (SERC)
- Joint research by NPS and AFIT to incorporate methods in case studies for assessing impacts of requirements changes and scenario variations in MBSE tools, Modeling and Simulation (M&S) environments.
- Focus on translations between models/tools in MBSE, specifically mapping architectural elements into behavior/performance analysis and cost model inputs.
  - SysML, DoDAF, Monterey Phoenix, parametric cost models, M&S environments
- Initial application to UAV Intelligence, Surveillance and Reconnaissance (UAS ISR) mission involving heterogeneous teams of autonomous and cooperative agents.
- AFIT develop mission CONOPS, Architectures and provide modeling support.
- NPS provide cost modeling expertise, tools and modeling support.
- Approach
  - Develop operational and system architectures to capture sets of military scenarios.
  - Develop the architectures in MBSE environments.
  - Design and demonstrate UAS ISR tradespace in MBSE and/or M&S environments .
  - Develop cost model interfaces for components of the architectures in order to evaluate cost effectiveness in an uncertain future environment.

Overview

- Total Ownership Cost (TOC) modeling to enable affordability tradeoffs with other ilities
  - Integrated costing of systems, software, hardware and human factors across full lifecycle operations
  - Combine with other MBSE architecture-based behavior and performance analysis
- Current shortfalls for ilities tradespace analysis
  - Models/tools are incomplete wrt/ TOC phases, activities, disciplines, SoS aspects
  - No integration with physical design space analysis tools, system modeling, or each other
- Cost estimation can be improved by using the same architectural definitions for cost model inputs, without the need for independent cost modeling expertise and effort expenditure.
- Developing translation rules and constructs between MBSE methods, performance analysis and cost model inputs.
- Demonstrating tool interoperability and tailorability

Case Study Method

- Use various MBSE methods and tools to evaluate behavior and performance analysis in the face of requirements changes and System of System (SoS) architectural variations.
- Develop operational and system architectures to capture sets of UAS military scenarios for cooperative swarms with 3 UAS group sizes
- Transition the architectures to MBSE environments.
  - SysML diagrams and executable activity models using Innoslate and CORE
- Develop cost model interfaces for components of the architectures in order to evaluate cost effectiveness in an uncertain future environment.
  - XML model files parsed automatically to extract cost model inputs
- Design and demonstrate UAS ISR tradespace including cost in integrated MBSE environment with executable models of architectures

UAS Mission Summaries

- Single UAS Search and Target Tracking (Simple Mission)

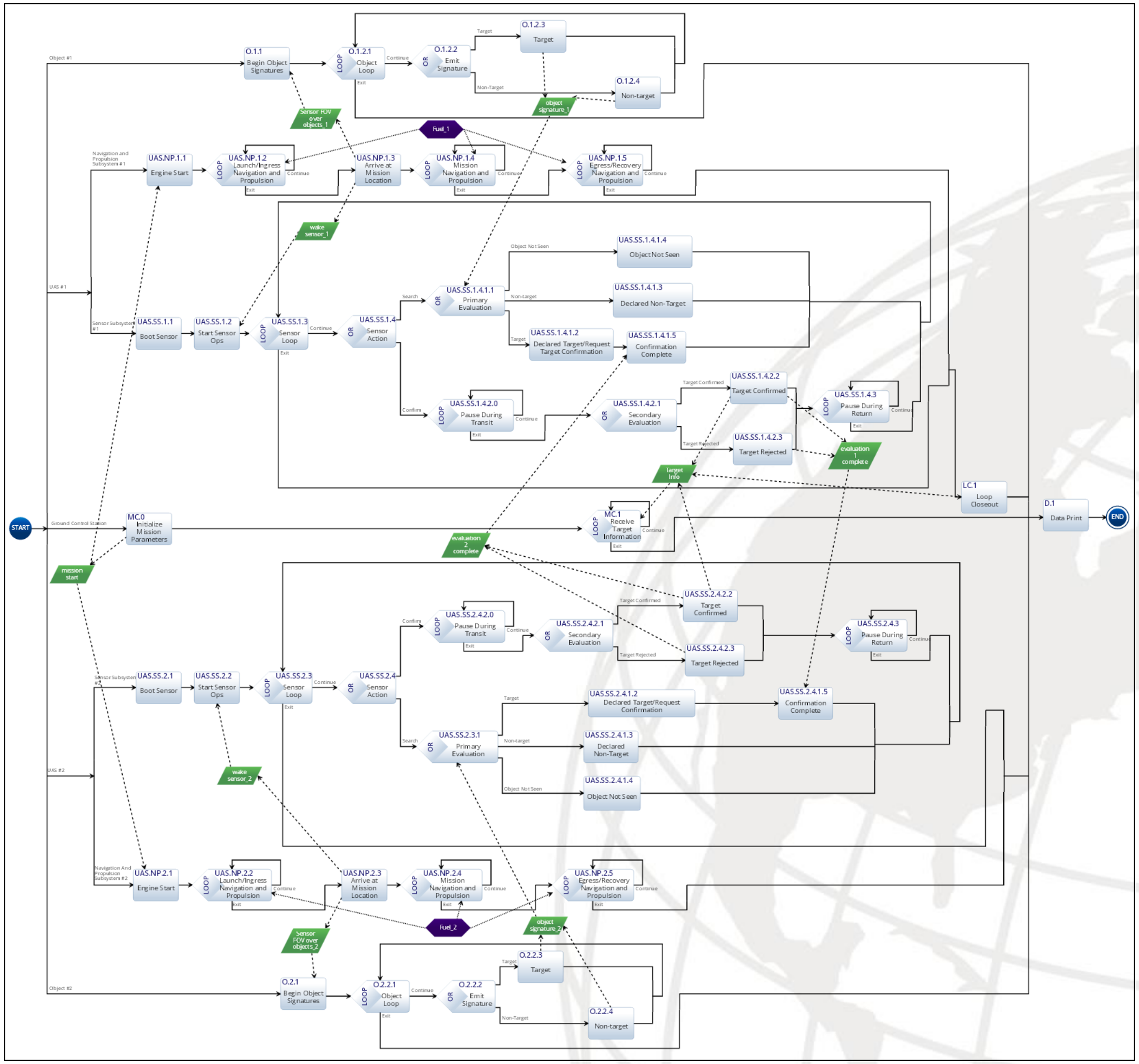
- UAS Pair Search and Target Tracking
- Find, Fix and Finish Terrorist Leadership (1)
- Find, Fix and Finish Terrorist Leadership (2)
- Mobile Missile Launcher Monitoring (1)
- Mobile Missile Launcher Monitoring (2)

Single UAS Simple Mission Threads

- Launch
- Navigation and flight
- Search and target ID including evaluation
- Target tracking
- Return/recovery

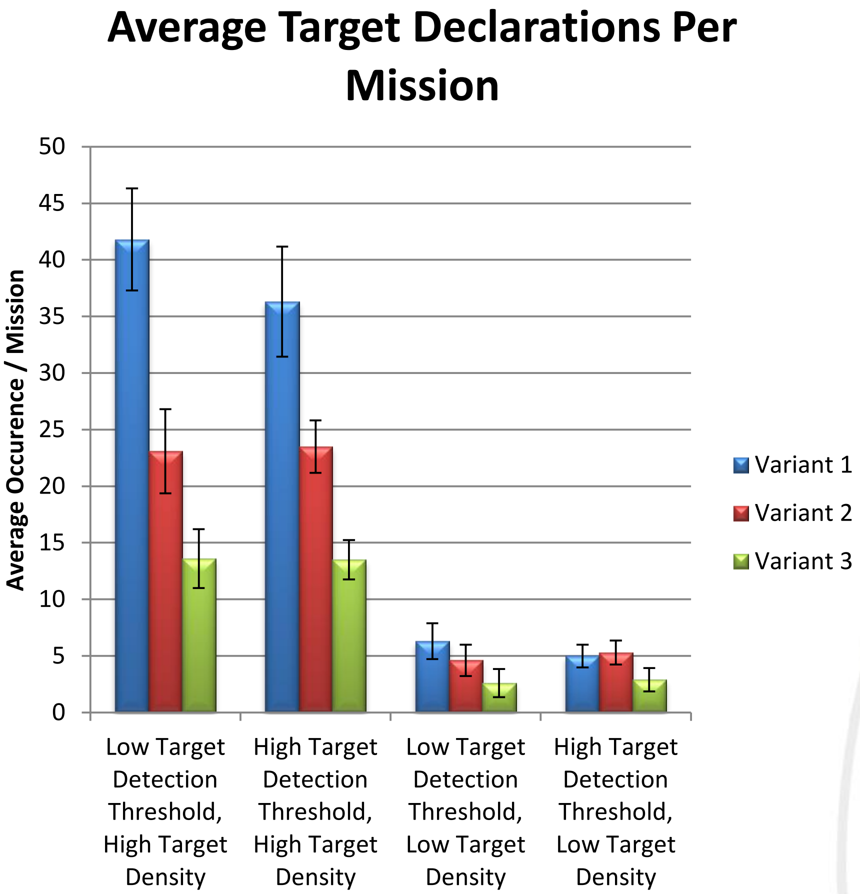
- *Enumeration of these in MBSE models constitutes primary size input for Constructive Systems Engineering Cost Model (COSYSMO)*

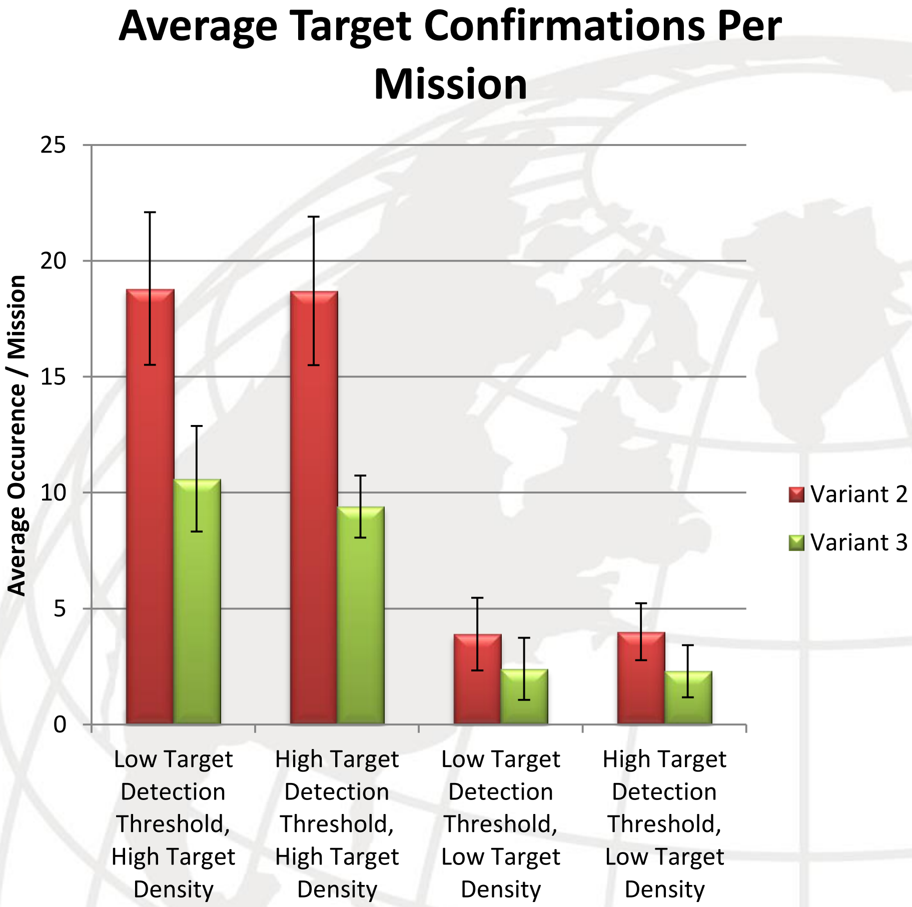
Example Activity Model (OV-5b) for Simple UAS Mission



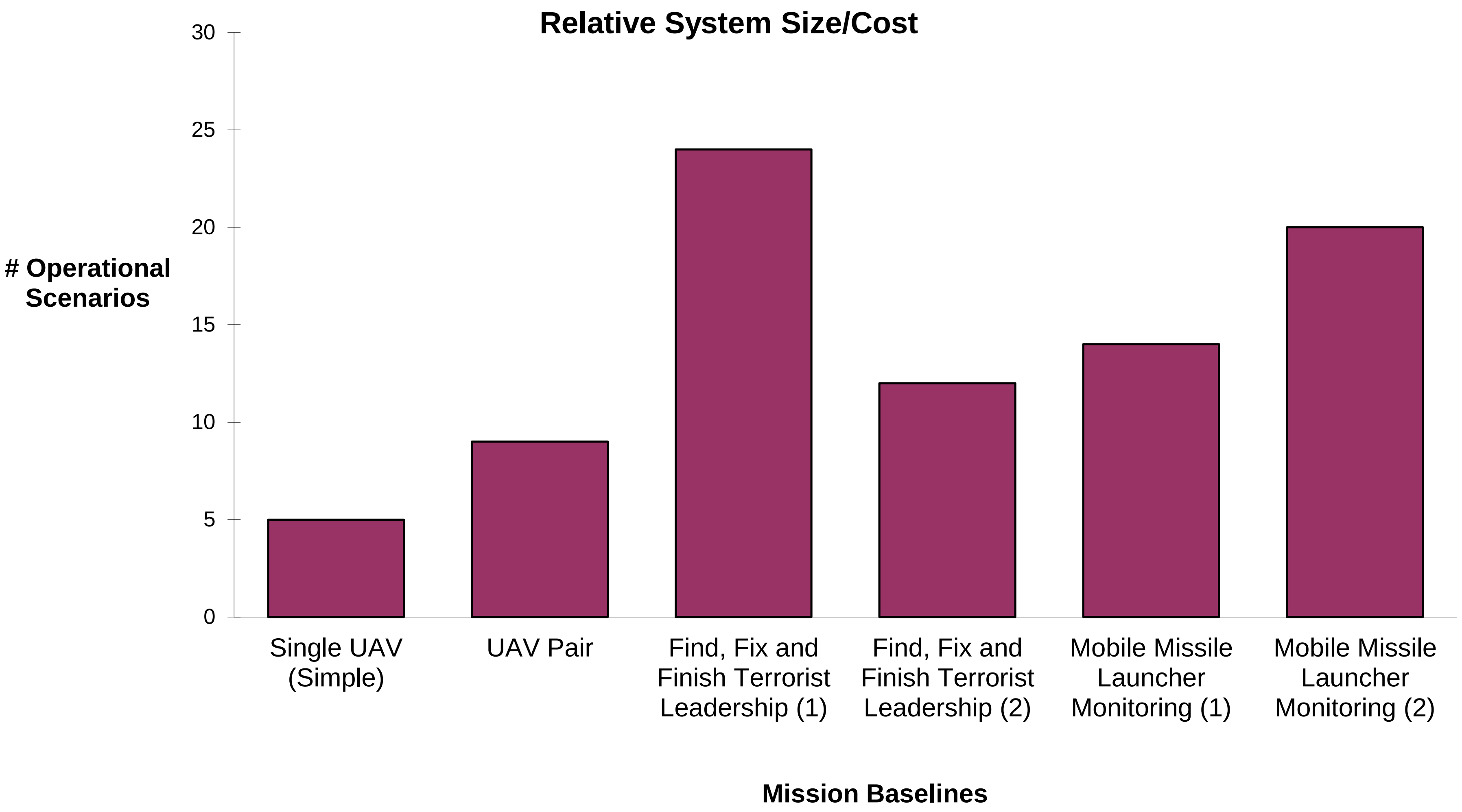
NAVAL POSTGRADUATE SCHOOL

Example Measures of Effectiveness for UAV Mission from Simulation





UAV Mission
Nominal Cost Comparisons



Agenda

- Introduction
- ➡ Cost Modeling Overview
- SysML Integration
- Case Study: Remote Targeting System UAS
- Case Study: TBM Identification and Elimination: 3-Tier UAS SoS
- Conclusions and Future Work

Parametric Effort Formula for Constructive Cost Models

$$Effort = A * Size^{B} * \prod_{i=1}^{N} EM_i$$

Where

- *Effort* is in Person-Months (PM)
- *A* is a constant derived from historical project data
- *Size* is a measure of the work product
- *B* is an exponent for the diseconomy of scale
- EM_i is an effort multiplier for the i^{th} cost driver. The geometric product of *N* multipliers is an overall Effort Adjustment Factor (EAF) to the nominal effort.

Constructive - A user understands why the model gives the estimate it does, and gains a better understanding of the job being estimated through using the cost model.

COSYSMO Effort Equation

$$Effort = A \cdot \left(\sum_{k} (w_{e,k}\Phi_{e,k} + w_{n,k}\Phi_{n,k} + w_{d,k}\Phi_{d,k}) \right)^{B} \cdot \prod_{i=1}^{14} EM_i$$

Where:

PM$_{NS}$ = effort in Person Months (Nominal Schedule)

A = calibration constant derived from historical project data

k = {Requirements, Interfaces, Algorithms, Scenarios}

w$_x$ = weight for “easy”, “nominal”, or “difficult” size driver

Φ_x = quantity of “k” size driver

B = represents diseconomies of scale

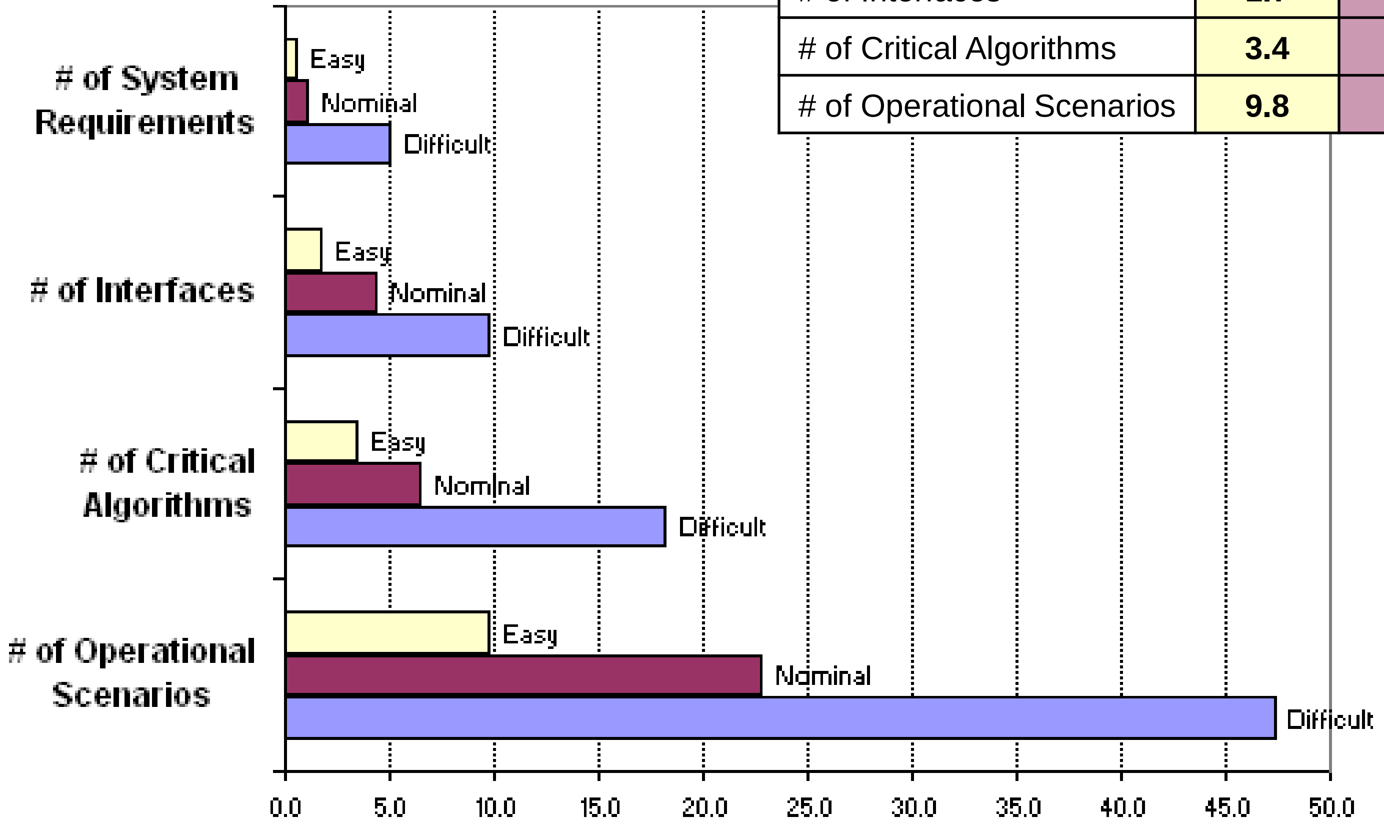
EM$_i$ = effort multiplier for the j_{th} cost driver. The geometric product results in an overall effort adjustment factor to the nominal effort.

COSYSMO Size Inputs

Size Type	Description
Requirements	The number of requirements for the system-of-interest at a specific level of design. Requirements may be functional, performance, feature, or service-oriented.
Interfaces	The number of shared physical and logical boundaries between system components or functions (internal interfaces) and those external to the system (external interfaces).
Algorithms	The number of newly defined or significantly altered functions that require unique mathematical algorithms to be derived in order to achieve the system performance requirements.
Operational Scenarios (Threads)	Operational scenarios that a system must satisfy, including nominal and off-nominal threads.

Size Driver Weights

	Easy	Nominal	Difficult
# of System Requirements	**0.5**	**1.00**	**5.0**
# of Interfaces	**1.7**	**4.3**	**9.8**
# of Critical Algorithms	**3.4**	**6.5**	**18.2**
# of Operational Scenarios	**9.8**	**22.8**	**47.4**



Cost Driver Ratings and Effort Multipliers

	Very Low	Low	Nominal	High	Very High	Extra High	EMR
Requirements Understanding	1.87	1.37	1.00	0.77	0.60		3.12
Architecture Understanding	1.64	1.28	1.00	0.81	0.65		2.52
Level of Service Requirements	0.62	0.79	1.00	1.36	1.85		2.98
Migration Complexity			1.00	1.25	1.55	1.93	1.93
Technology Risk	0.67	0.82	1.00	1.32	1.75		2.61
Documentation	0.78	0.88	1.00	1.13	1.28		1.64
# and diversity of installations/platforms			1.00	1.23	1.52	1.87	1.87
# of recursive levels in the design	0.76	0.87	1.00	1.21	1.47		1.93
Stakeholder team cohesion	1.50	1.22	1.00	0.81	0.65		2.31
Personnel/team capability	1.50	1.22	1.00	0.81	0.65		2.31
Personnel experience/continuity	1.48	1.22	1.00	0.82	0.67		2.21
Process capability	1.47	1.21	1.00	0.88	0.77	0.68	2.16
Multisite coordination	1.39	1.18	1.00	0.90	0.80	0.72	1.93
Tool support	1.39	1.18	1.00	0.85	0.72		1.93

EMR = Effort Multiplier Ratio

Average Effort Distribution Across EIA 632 Fundamental Processes

EIA 632 Fundamental Process	Percent
Acquisition & Supply	7%
Technical Management	17%
System Design	30%
Product Realization	15%
Technical Evaluation	31%

System Requirements

Number of System Requirements

This driver represents the number of requirements for the system-of-interest at a specific level of design. The quantity of requirements includes those related to the effort involved in system engineering the system interfaces, system specific algorithms, and operational scenarios. Requirements may be functional, performance, feature, or service-oriented in nature depending on the methodology used for specification. They may also be defined by the customer or contractor. Each requirement may have effort associated with is such as V&V, functional decomposition, functional allocation, etc. System requirements can typically be quantified by counting the number of applicable shalls/wills/shoulds/mays in the system or marketing specification.

Note: some work is involved in decomposing requirements so that they may be counted at the appropriate system-of-interest.

Easy	Nominal	Difficult
- Simple to implement	- **Familiar**	**- Complex to implement or engineer**
- Traceable to source	**- Can be traced to source with some effort**	**- Hard to trace to source**
- Little requirements overlap	**- Some overlap**	**- High degree of requirements overlap**

System Interfaces

Number of System Interfaces

This driver represents the number of shared physical and logical boundaries between system components or functions (internal interfaces) and those external to the system (external interfaces). These interfaces typically can be quantified by counting the number of external and internal system interfaces among ISO/IEC 15288-defined system elements.

Easy	Nominal	Difficult
- Simple message	**- Moderate complexity**	**- Complex protocol(s)**
- Uncoupled	**- Loosely coupled**	**- Highly coupled**
- Strong consensus	**- Moderate consensus**	**- Low consensus**
- Well behaved	**- Predictable behavior**	**- Poorly behaved**

System-Specific Algorithms

Number of System-Specific Algorithms

This driver represents the number of newly defined or significantly altered functions that require unique mathematical algorithms to be derived in order to achieve the system performance requirements. As an example, this could include a complex aircraft tracking algorithm like a Kalman Filter being derived using existing experience as the basis for the all aspect search function. Another example could be a brand new discrimination algorithm being derived to identify friend or foe function in space-based applications. The number can be quantified by counting the number of unique algorithms needed to realize the requirements specified in the system specification or mode description document.

Easy	Nominal	Difficult
-Algebraic	- Straight forward calculus	- Complex constrained optimization; pattern recognition
- Straightforward structure	- Nested structure with decision logic	- Recursive in structure with distributed control
- Simple data	- Relational data	- Noisy, ill-conditioned data
- Timing not an issue	- Timing a constraint	- Dynamic, with timing and uncertainty issues
- Adaptation of library-based solution	- Some modeling involved	- Simulation and modeling involved

Operational Scenarios (Threads)

Number of Operational Scenarios

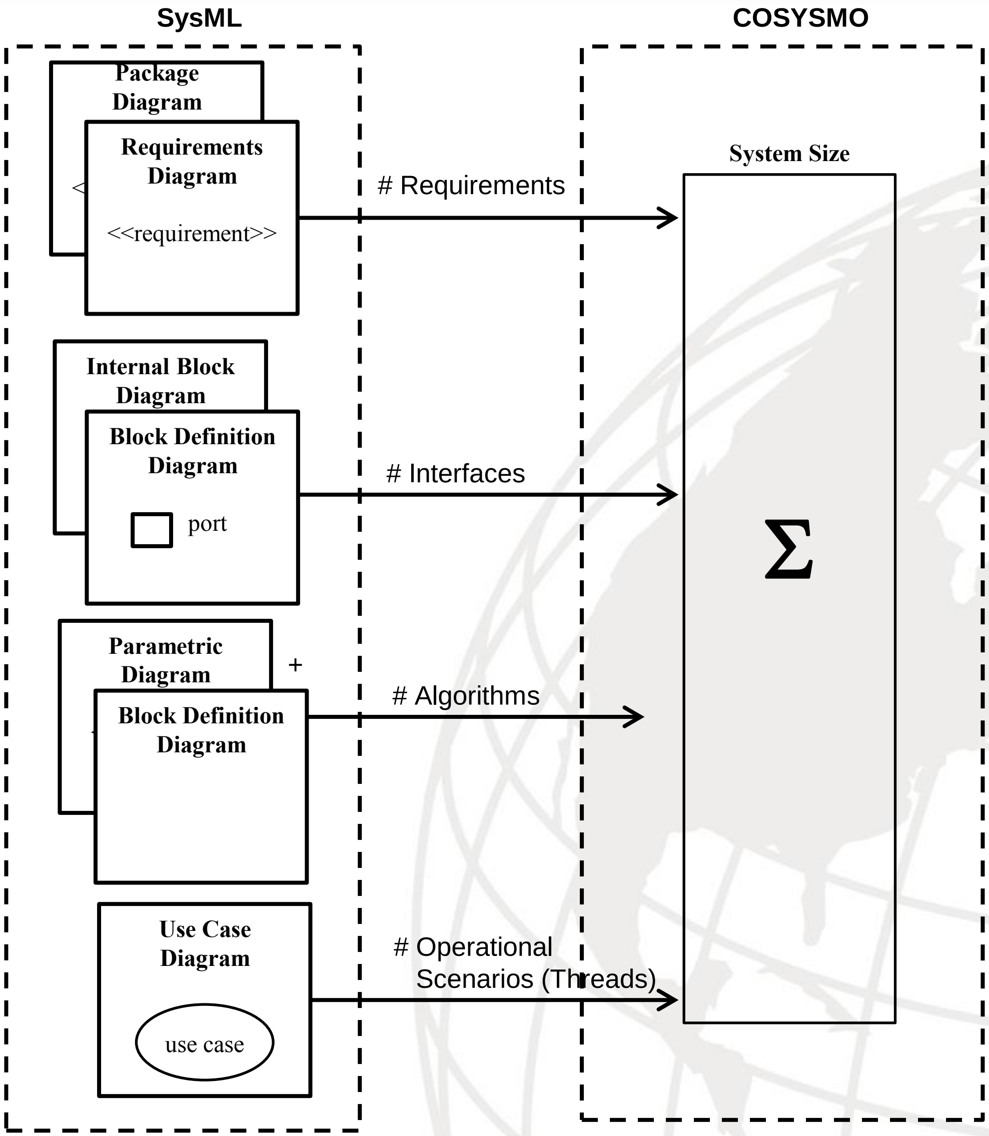
This driver represents the number of operational scenarios that a system must satisfy. Such scenarios include both the nominal stimulus-response thread plus all of the off-nominal threads resulting from bad or missing data, unavailable processes, network connections, or other exception-handling cases. The number of scenarios can typically be quantified by counting the number of system test thread packages or unique end-to-end tests used to validate the system functionality and performance or by counting the number of use cases, including off-nominal extensions, developed as part of the operational architecture.

Easy	Nominal	Difficult
- Well defined	**- Loosely defined**	**- Ill defined**
- Loosely coupled	**- Moderately coupled**	**- Tightly coupled or many dependencies/conflicting requirements**
- Timelines not an issue	**- Timelines a constraint**	**- Tight timelines through scenario network**
- Few, simple off-nominal threads	**- Moderate number or complexity of off-nominal threads**	**- Many or very complex off-nominal threads**

Agenda

- Introduction
- Cost Modeling Overview
- ➡ SysML Integration
- Case Study: Remote Targeting System UAS
- Case Study: TBM Identification and Elimination: 3-Tier UAS SoS
- Conclusions and Future Work

SySML to COSYSMO Mapping



SysML

COSYSMO

Package Diagram

Requirements Diagram

<<requirement>>

Internal Block Diagram

Block Definition Diagram

port

Parametric Diagram

+

Block Definition Diagram

Use Case Diagram

use case

# Requirements

# Interfaces

# Algorithms

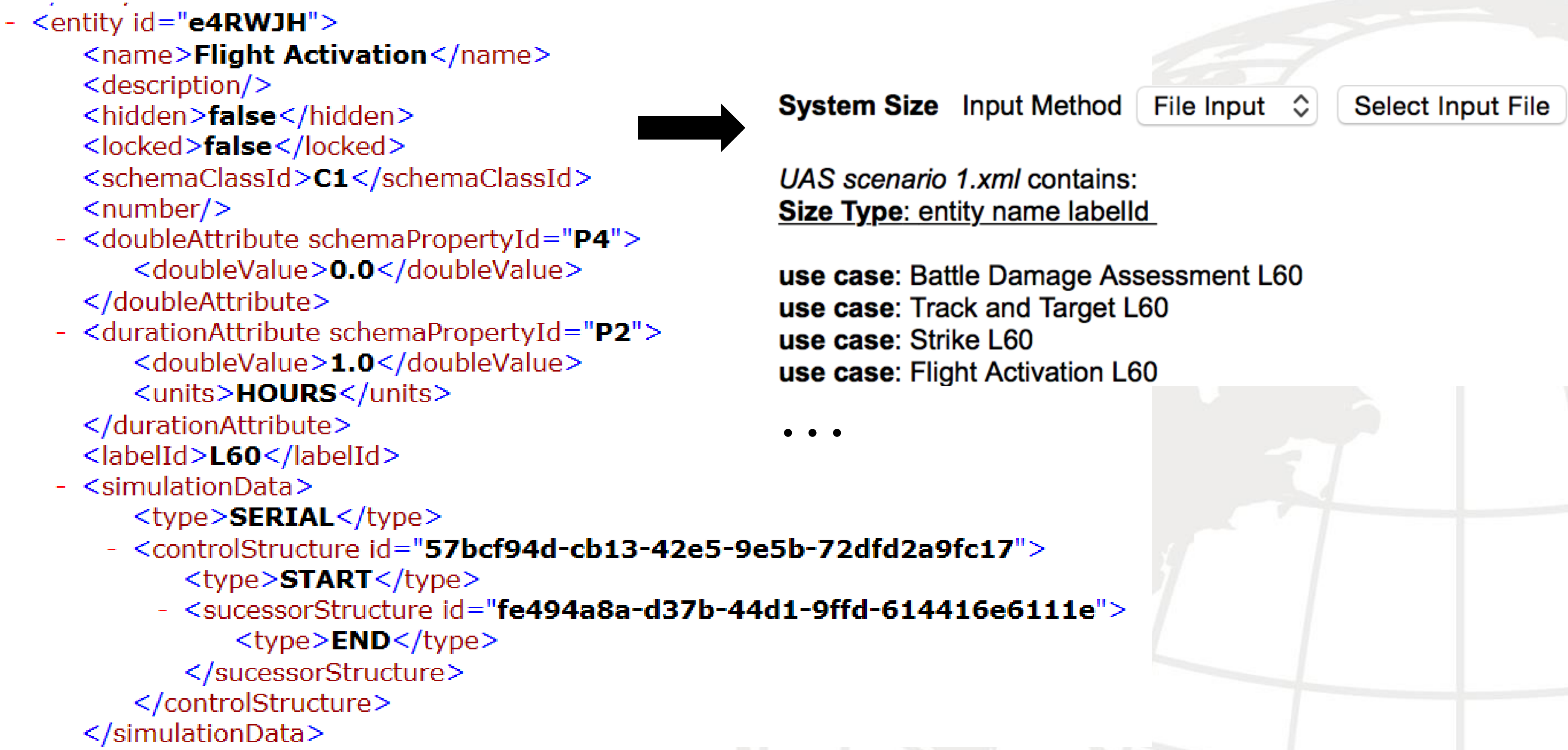
# Operational Scenarios (Threads)

System Size

Σ

XML Interface Processing

• • •



```
- <entity id="e4RWJH">
      <name>Flight Activation</name>
      <description/>
      <hidden>false</hidden>
      <locked>false</locked>
      <schemaClassId>C1</schemaClassId>
      <number/>
    - <doubleAttribute schemaPropertyId="P4">
          <doubleValue>0.0</doubleValue>
      </doubleAttribute>
    - <durationAttribute schemaPropertyId="P2">
          <doubleValue>1.0</doubleValue>
          <units>HOURS</units>
      </durationAttribute>
      <labelId>L60</labelId>
    - <simulationData>
          <type>SERIAL</type>
        - <controlStructure id="57bcf94d-cb13-42e5-9e5b-72dfd2a9fc17">
              <type>START</type>
            - <sucessorStructure id="fe494a8a-d37b-44d1-9ffd-614416e6111e">
                  <type>END</type>
              </sucessorStructure>
          </controlStructure>
      </simulationData>
```

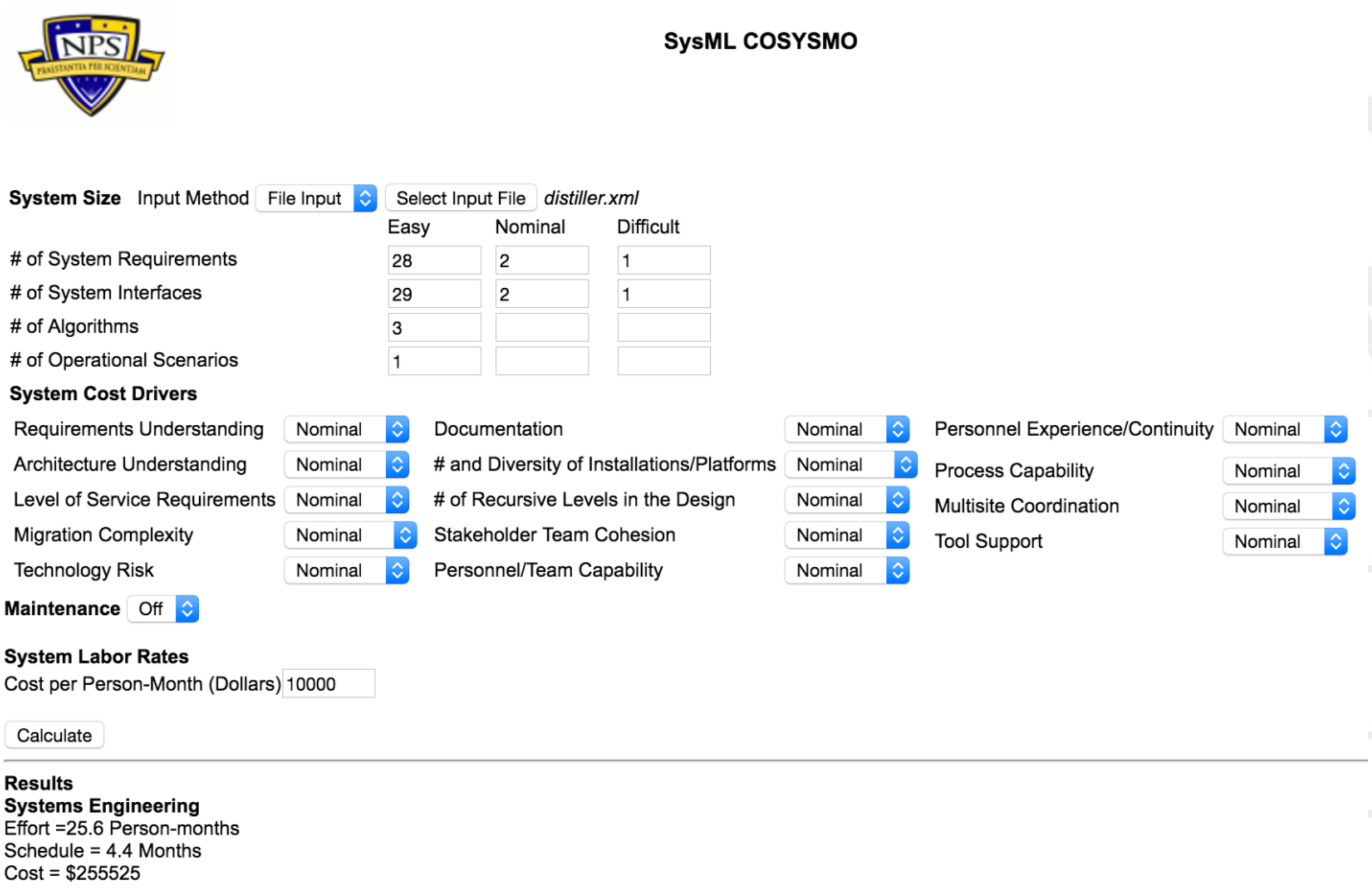
UAS scenario 1.xml contains:
Size Type: entity name labelId

use case: Battle Damage Assessment L60
use case: Track and Target L60
use case: Strike L60
use case: Flight Activation L60

• • •

• • •

Example COSYSMO Estimate



SysML COSYSMO

System Size Input Method File Input Select Input File *distiller.xml*

	Easy	Nominal	Difficult
# of System Requirements	28	2	1
# of System Interfaces	29	2	1
# of Algorithms	3		
# of Operational Scenarios	1		

System Cost Drivers

Requirements Understanding	Nominal	Documentation	Nominal	Personnel Experience/Continuity	Nominal
Architecture Understanding	Nominal	# and Diversity of Installations/Platforms	Nominal	Process Capability	Nominal
Level of Service Requirements	Nominal	# of Recursive Levels in the Design	Nominal	Multisite Coordination	Nominal
Migration Complexity	Nominal	Stakeholder Team Cohesion	Nominal	Tool Support	Nominal
Technology Risk	Nominal	Personnel/Team Capability	Nominal		

Maintenance Off

System Labor Rates

Cost per Person-Month (Dollars) 10000

Results

Systems Engineering

Effort =25.6 Person-months

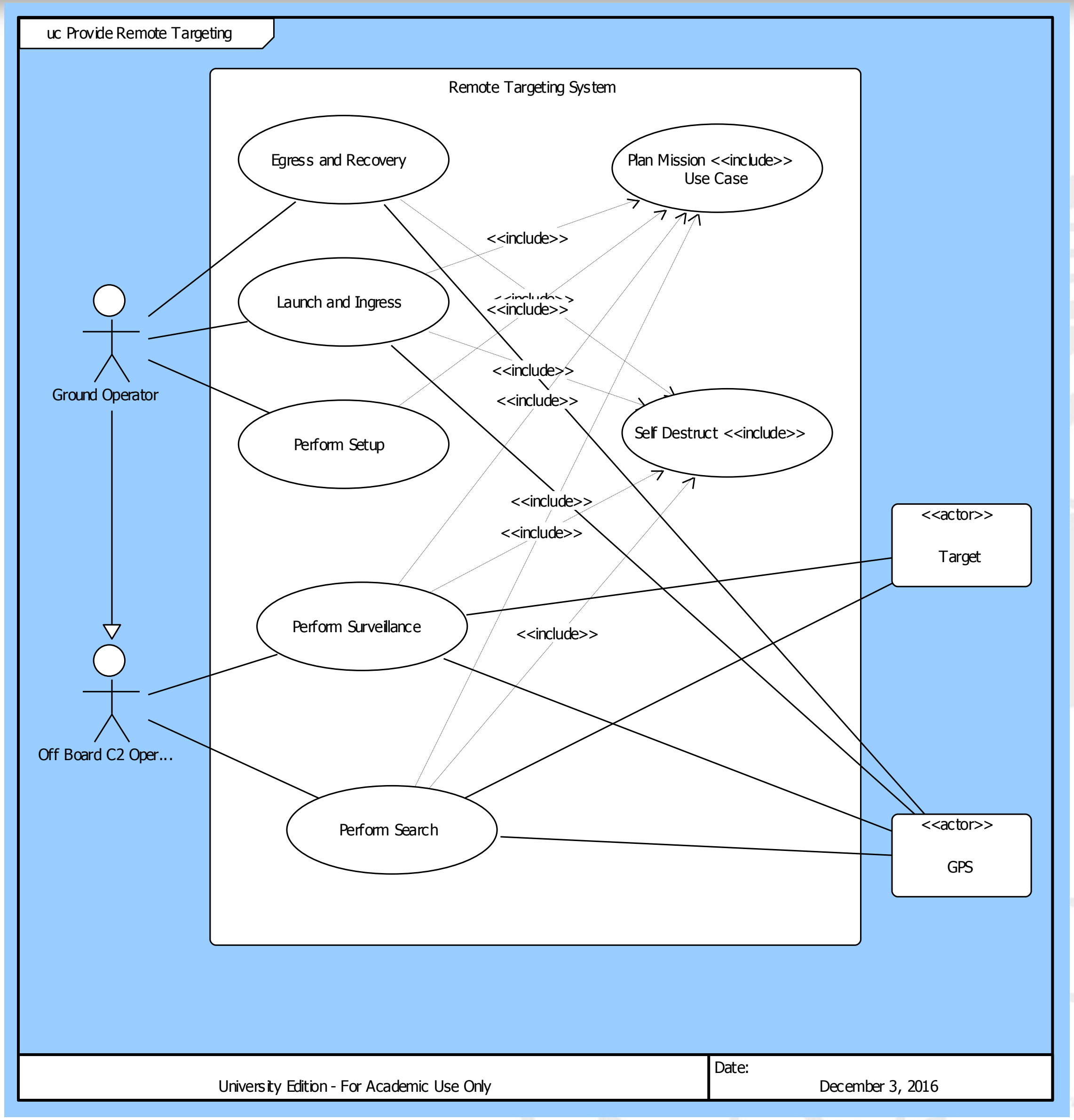
Schedule = 4.4 Months

Cost = $255525

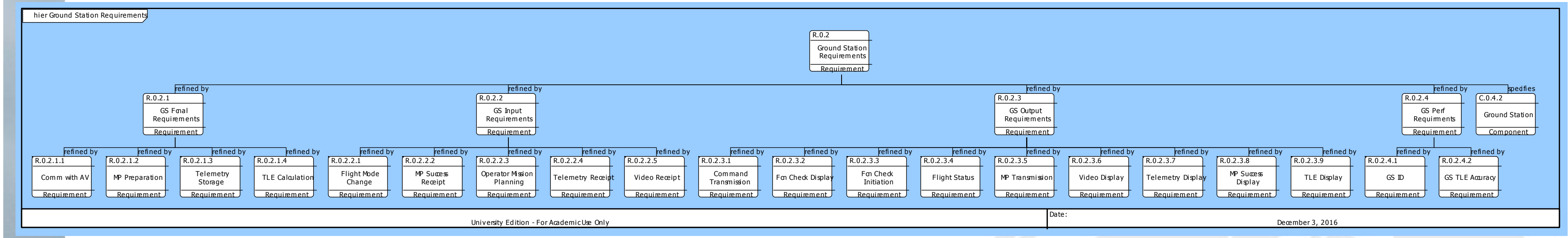
Agenda

- Introduction
- Cost Modeling Overview
- SysML Integration
- ➡ Case Study: Remote Targeting System UAS
- Case Study: TBM Identification and Elimination: 3-Tier UAS SoS
- Conclusions and Future Work

RTS Scenarios (Use Cases)



RTS Requirements



hier Ground Station Requirements

R.0.2 Ground Station Requirements Requirement

refined by R.0.2.1 GS Fcnal Requirements Requirement

refined by R.0.2.1.1 Comm with AV Requirement

refined by R.0.2.1.2 MP Preparation Requirement

refined by R.0.2.1.3 Telemetry Storage Requirement

refined by R.0.2.1.4 TLE Calculation Requirement

refined by R.0.2.2 GS Input Requirements Requirement

refined by R.0.2.2.1 Flight Mode Change Requirement

refined by R.0.2.2.2 MP Success Receipt Requirement

refined by R.0.2.2.3 Operator Mission Planning Requirement

refined by R.0.2.2.4 Telemetry Receipt Requirement

refined by R.0.2.2.5 Video Receipt Requirement

refined by R.0.2.3 GS Output Requirements Requirement

refined by R.0.2.3.1 Command Transmission Requirement

refined by R.0.2.3.2 Fcn Check Display Requirement

refined by R.0.2.3.3 Fcn Check Initiation Requirement

refined by R.0.2.3.4 Flight Status Requirement

refined by R.0.2.3.5 MP Transmission Requirement

refined by R.0.2.3.6 Video Display Requirement

refined by R.0.2.3.7 Telemetry Display Requirement

refined by R.0.2.3.8 MP Success Display Requirement

refined by R.0.2.3.9 TLE Display Requirement

refined by R.0.2.4 GS Perf Requirments Requirement

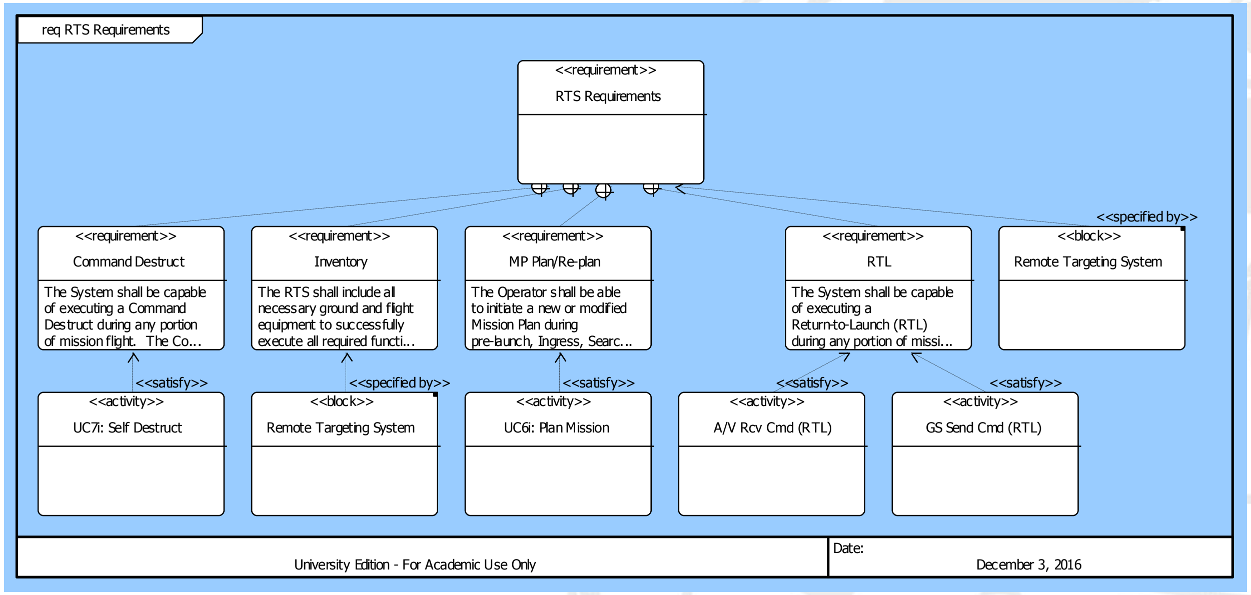
refined by R.0.2.4.1 GS ID Requirement

refined by R.0.2.4.2 GS TLE Accuracy Requirement

specifies C.0.4.2 Ground Station Component

University Edition - For Academic Use Only

Date: December 3, 2016



req RTS Requirements

<<requirement>> RTS Requirements

<<requirement>> Command Destruct

The System shall be capable of executing a Command Destruct during any portion of mission flight. The Co...

<<requirement>> Inventory

The RTS shall include all necessary ground and flight equipment to successfully execute all required functi...

<<requirement>> MP Plan/Re-plan

The Operator shall be able to initiate a new or modified Mission Plan during pre-launch, Ingress, Searc...

<<requirement>> RTL

The System shall be capable of executing a Return-to-Launch (RTL) during any portion of missi...

<<specified by>>

<<block>> Remote Targeting System

<<satisfy>>

<<activity>> UC7i: Self Destruct

<<specified by>>

<<block>> Remote Targeting System

<<satisfy>>

<<activity>> UC6i: Plan Mission

<<satisfy>>

<<activity>> A/V Rcv Cmd (RTL)

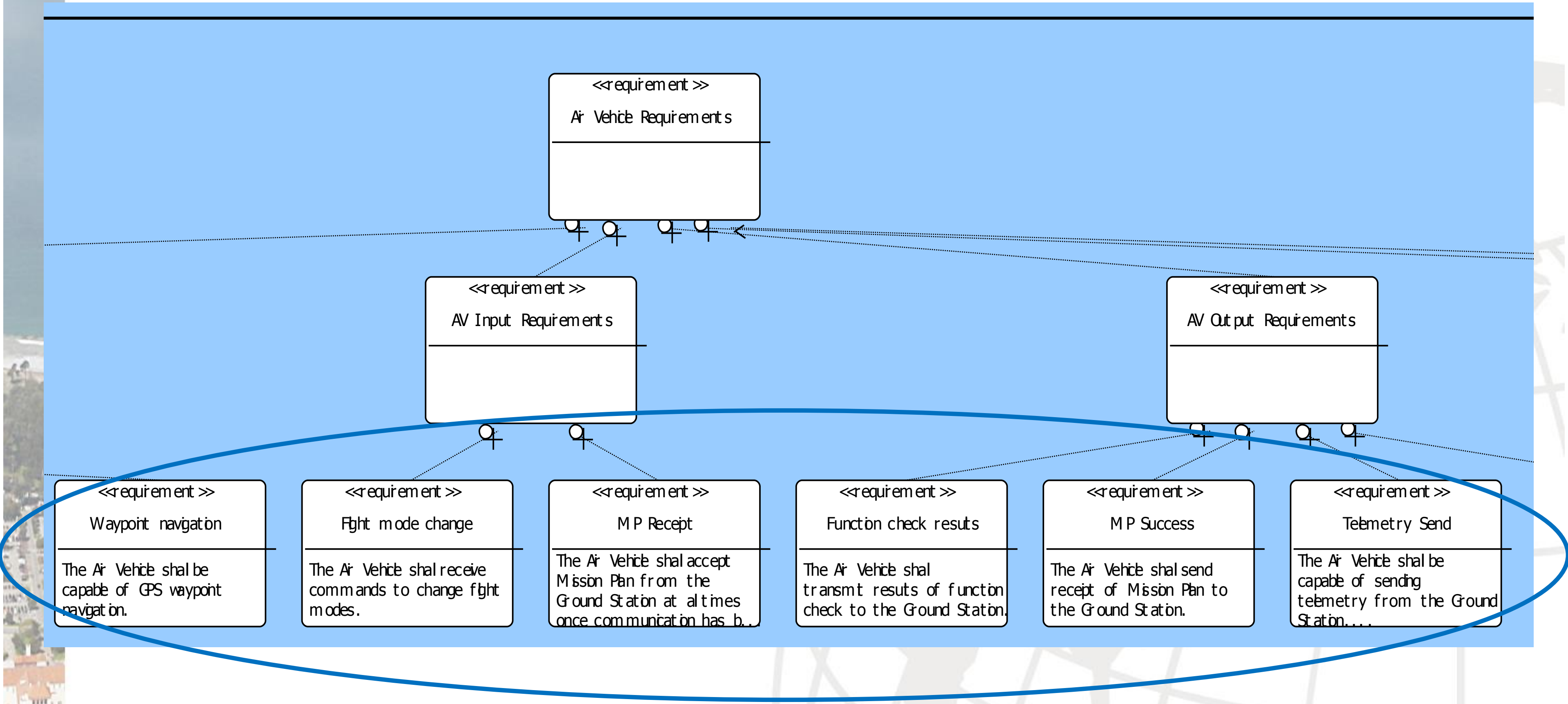
<<satisfy>>

<<activity>> GS Send Cmd (RTL)

University Edition - For Academic Use Only

Date: December 3, 2016

Requirements Decomposition Level for Costing



<<requirement>>
Air Vehicle Requirements

<<requirement>>
AV Input Requirements

<<requirement>>
AV Output Requirements

<<requirement>>
Waypoint navigation

The Air Vehicle shall be capable of GPS waypoint navigation.

<<requirement>>
Flight mode change

The Air Vehicle shall receive commands to change flight modes.

<<requirement>>
MP Receipt

The Air Vehicle shall accept Mission Plan from the Ground Station at all times once communication has b...

<<requirement>>
Function check results

The Air Vehicle shall transmit results of function check to the Ground Station.

<<requirement>>
MP Success

The Air Vehicle shall send receipt of Mission Plan to the Ground Station.

<<requirement>>
Telemetry Send

The Air Vehicle shall be capable of sending telemetry from the Ground Station....

Use Case Level for Costing

Perform Surveillance

Description: This Use Case covers surveillance activities

Preconditions: Target has been identified and Air Vehicle has entered Surveillance mode

Primary Flow:

1. Air Vehicle transmits telemetry to Ground Station(s)
2. Ground Station(s) receives and displays flight data
3. Ground Station(s) stores telemetry data
4. Air Vehicle loiters over target
5. Air Vehicle continues video transmission to Ground Station and Off-Board C2
6. Ground Station(s) receives and displays video transmission
7. Operator and Off-Board C2 monitor video and flight data
8. Ground station(s) calculate target coordinates based on video and telemetry
9. Ground station(s) displays target coordinates
10. Operator initiates RTL
11. Ground Station sends RTL command to Air Vehicle
12. Air Vehicle enters RTL mode

Alternate Flow: At any time:

a. If bad vehicle health, Operator enters RTL command on Ground Station
b. Ground Station sends RTL command to Air Vehicle
c. Air Vehicle enters RTL mode

At any time:

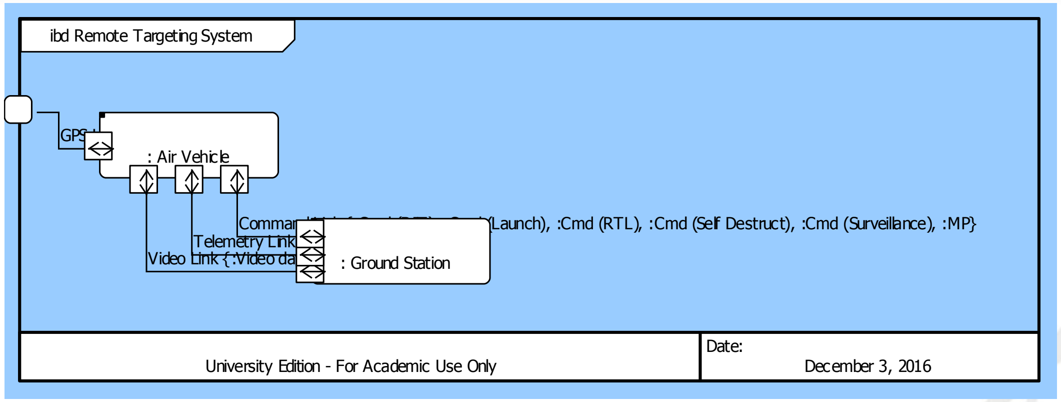
a. Operator initiates <<include>> Plan Mission Use Case
b. Vehicle ingresses to new Search Insertion point

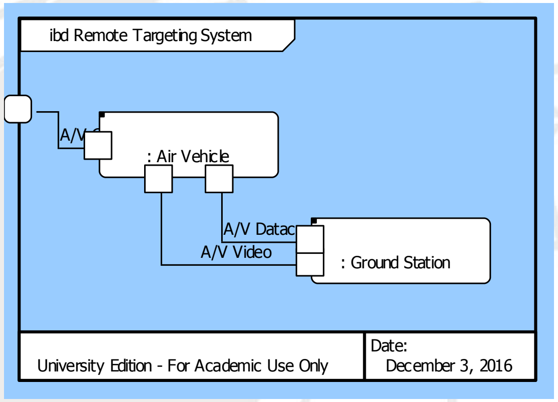
At any time:

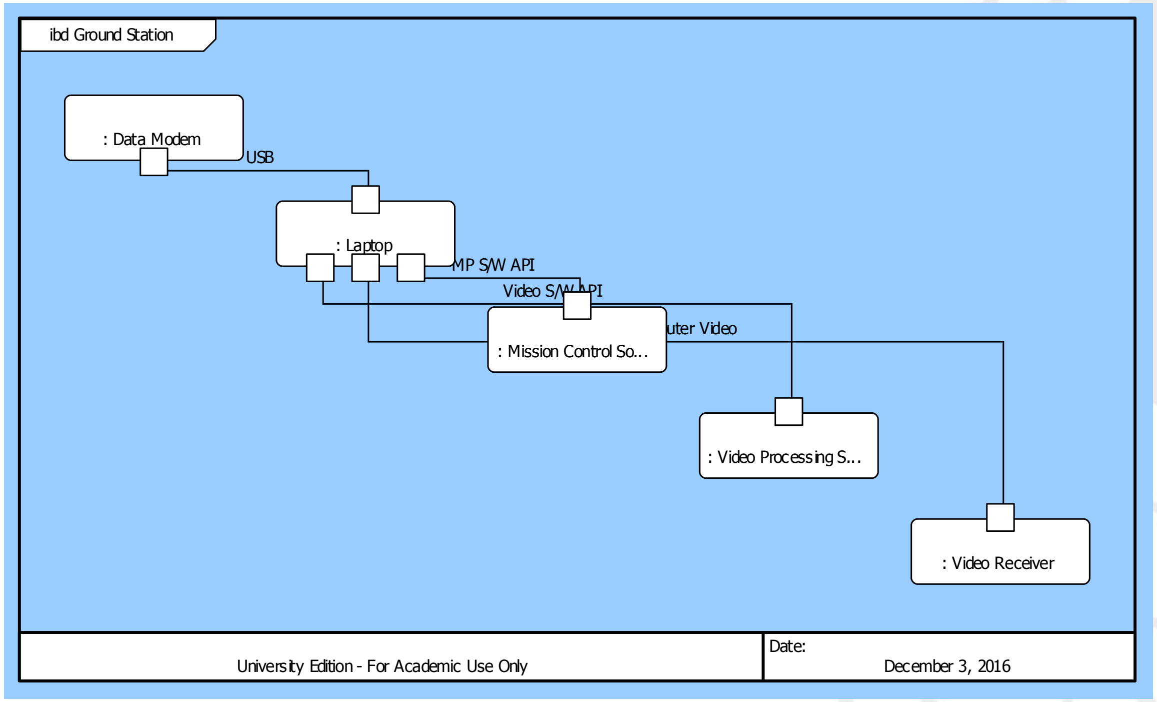
a. If vehicle compromise is evident, execute <<include>> Self Destruct Use Case

Postconditions: Air Vehicle is loitering over the target for > 10 minutes and target coordinates are calculated and displayed on Ground Station(s); Air Vehicle enters RTL mode

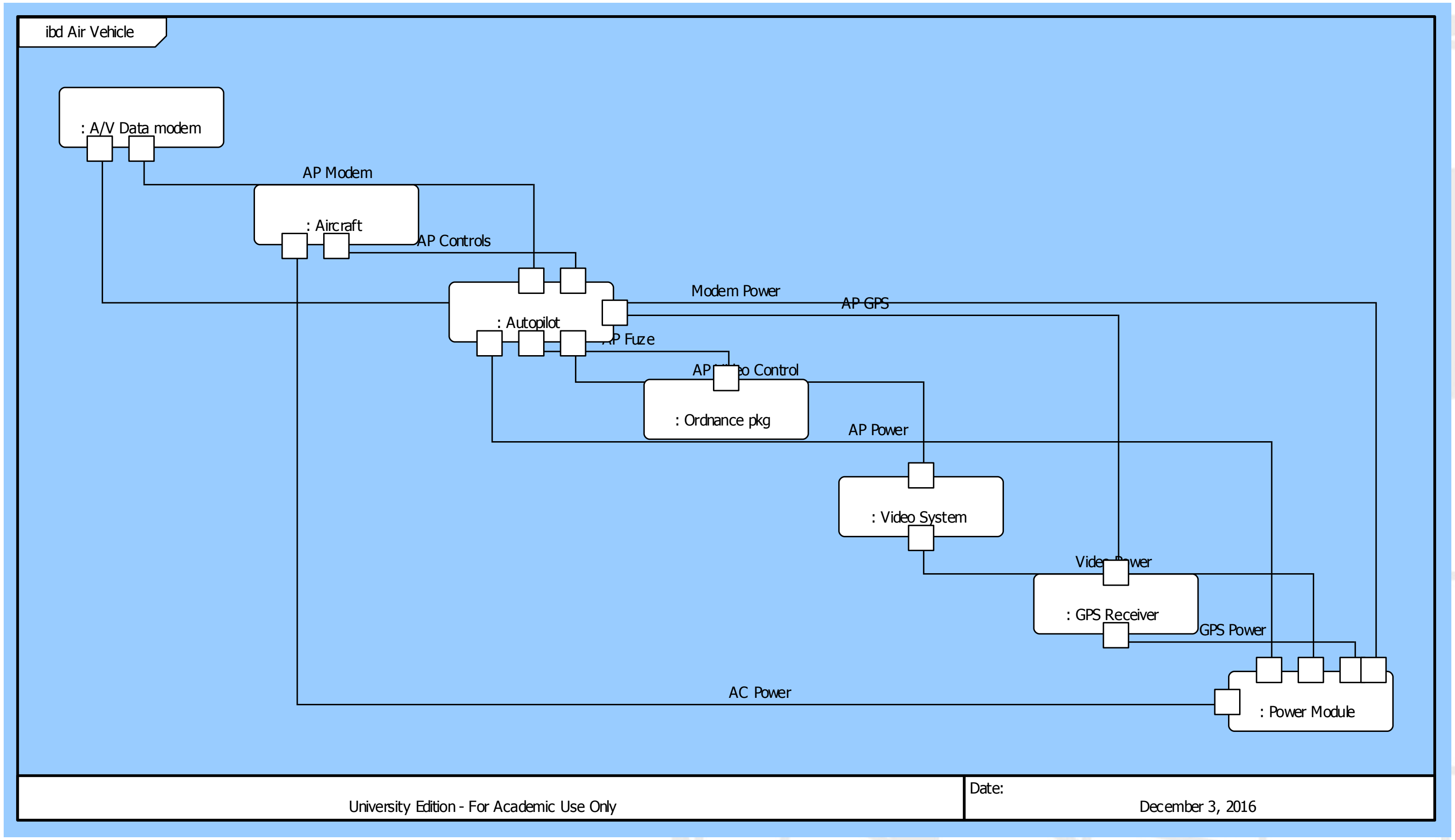
RTS Interfaces (Ports) (1/2)


ibd Remote Targeting System
GPS
: Air Vehicle
Comman
(Launch), :Cmd (RTL), :Cmd (Self Destruct), :Cmd (Surveillance), :MP}
Telemetry Link
Video Link { :Video da
: Ground Station
University Edition - For Academic Use Only
Date:
December 3, 2016

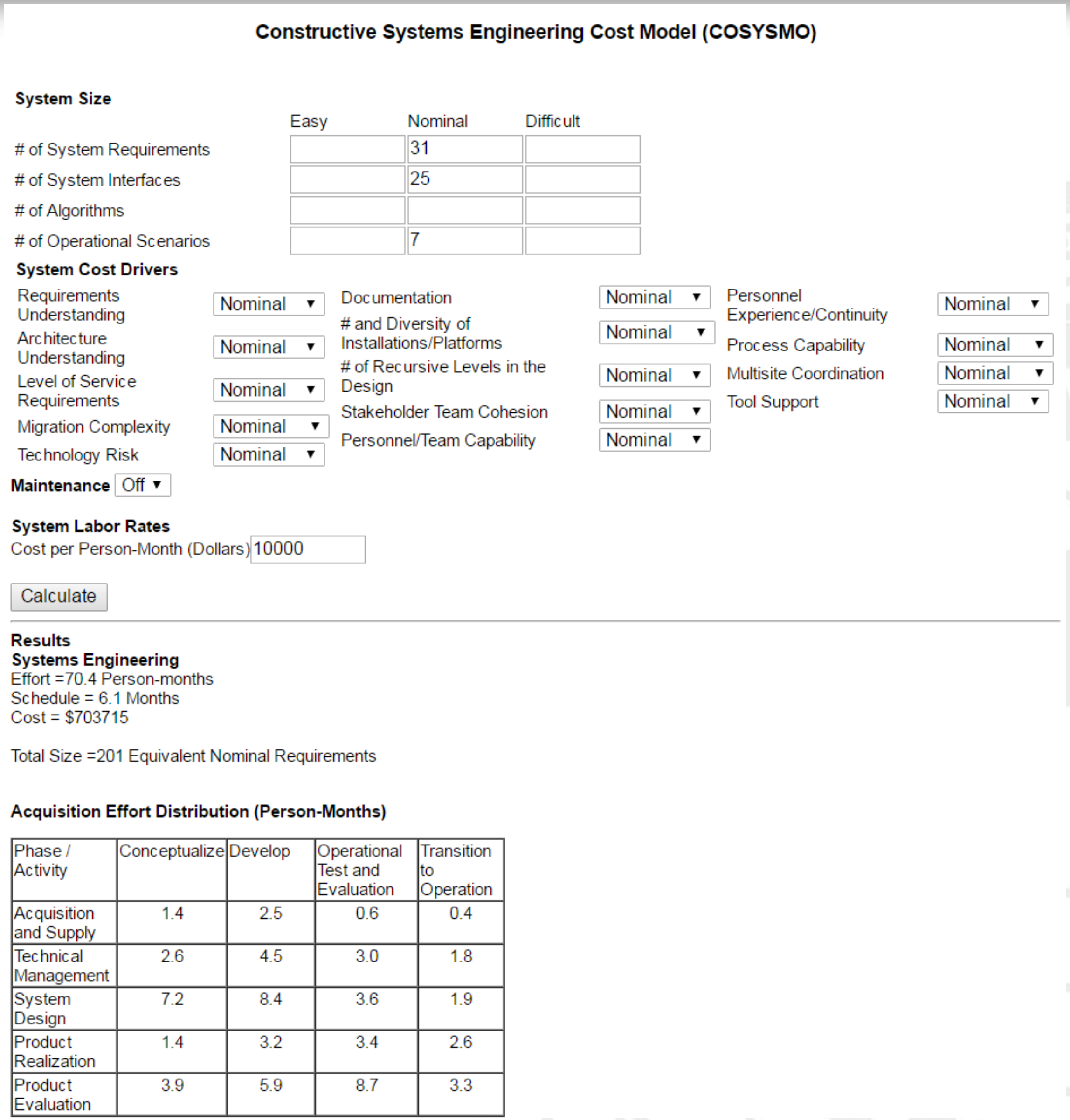




RTS Interfaces (Ports) (2/2)



RTS Nominal SE Cost Estimate



Constructive Systems Engineering Cost Model (COSYSMO)

System Size

	Easy	Nominal	Difficult
# of System Requirements		31	
# of System Interfaces		25	
# of Algorithms			
# of Operational Scenarios		7	

System Cost Drivers

Requirements Understanding	Nominal ▼	Documentation	Nominal ▼	Personnel Experience/Continuity	Nominal ▼
Architecture Understanding	Nominal ▼	# and Diversity of Installations/Platforms	Nominal ▼	Process Capability	Nominal ▼
Level of Service Requirements	Nominal ▼	# of Recursive Levels in the Design	Nominal ▼	Multisite Coordination	Nominal ▼
Migration Complexity	Nominal ▼	Stakeholder Team Cohesion	Nominal ▼	Tool Support	Nominal ▼
Technology Risk	Nominal ▼	Personnel/Team Capability	Nominal ▼		

Maintenance Off ▼

System Labor Rates

Cost per Person-Month (Dollars) 10000

Calculate

Results

Systems Engineering

Effort =70.4 Person-months

Schedule = 6.1 Months

Cost = $703715

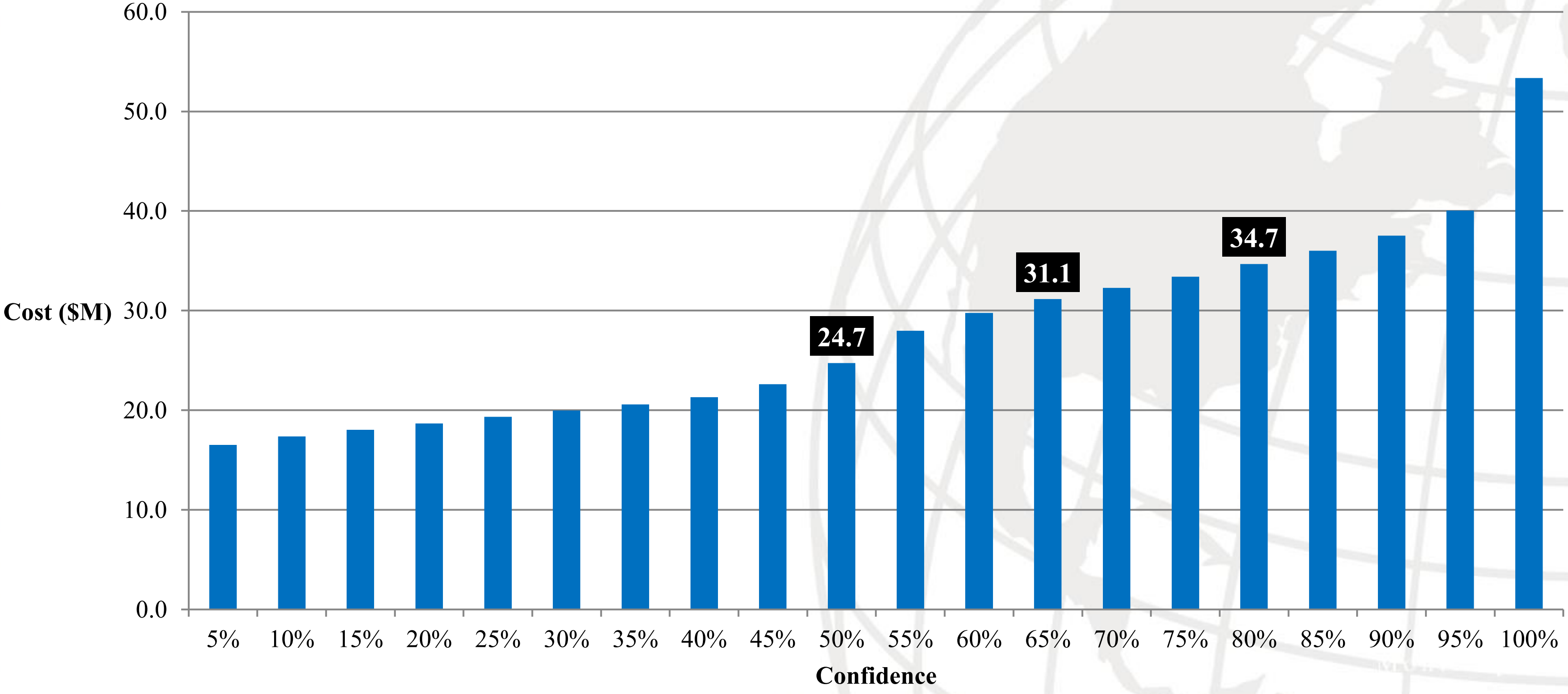
Total Size =201 Equivalent Nominal Requirements

Acquisition Effort Distribution (Person-Months)

Phase / Activity	Conceptualize	Develop	Operational Test and Evaluation	Transition to Operation
Acquisition and Supply	1.4	2.5	0.6	0.4
Technical Management	2.6	4.5	3.0	1.8
System Design	7.2	8.4	3.6	1.9
Product Realization	1.4	3.2	3.4	2.6
Product Evaluation	3.9	5.9	8.7	3.3

Extrapolation for RTS Full Lifecycle Cost Distribution

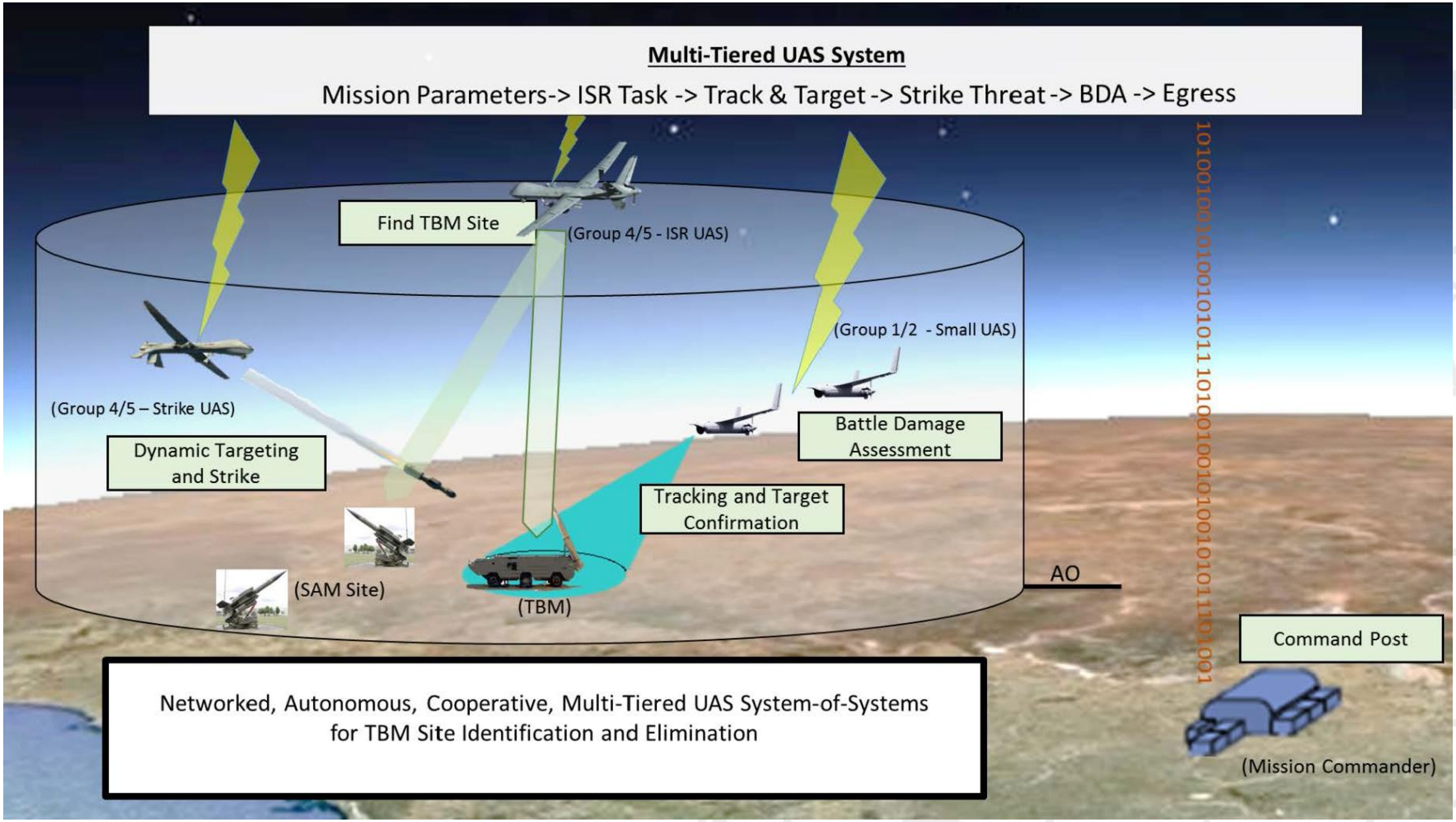
Input Table				
Variable	**Units**	**Optimistic**	**Expected**	**Pessimistic**
Systems Engineering Hours (From COSYSMO)	Hours	9476	10652	16284
SE Conversion Factor	Hours / Hour	0.150	0.125	0.100
Labor Rate	$ / Hour	100	110	125
Bill of Materials	$M	2	3	7
Travel	Percentage	2.5%	3.5%	6.0%
G&A Percentage	Percentage		10%	
ODC Percentage	Percentage		10.0%	
Fee	Percentage		10.0%	
MR	Percentage		10.0%	
Calibration Factor	No Units		1.3	



Agenda

- Introduction
- Cost Modeling Overview
- SysML Integration
- Case Study: Remote Targeting System UAS
- ➡ Case Study: TBM Identification and Elimination: 3-Tier UAS SoS
- Conclusions and Future Work

3-Tier UAS SoS CONOPS (OV 1)



Multi-Tiered UAS System

Mission Parameters-> ISR Task -> Track & Target -> Strike Threat -> BDA -> Egress

Find TBM Site

(Group 4/5 - ISR UAS)

(Group 1/2 - Small UAS)

(Group 4/5 – Strike UAS)

Dynamic Targeting and Strike

Battle Damage Assessment

Tracking and Target Confirmation

(SAM Site)

(TBM)

AO

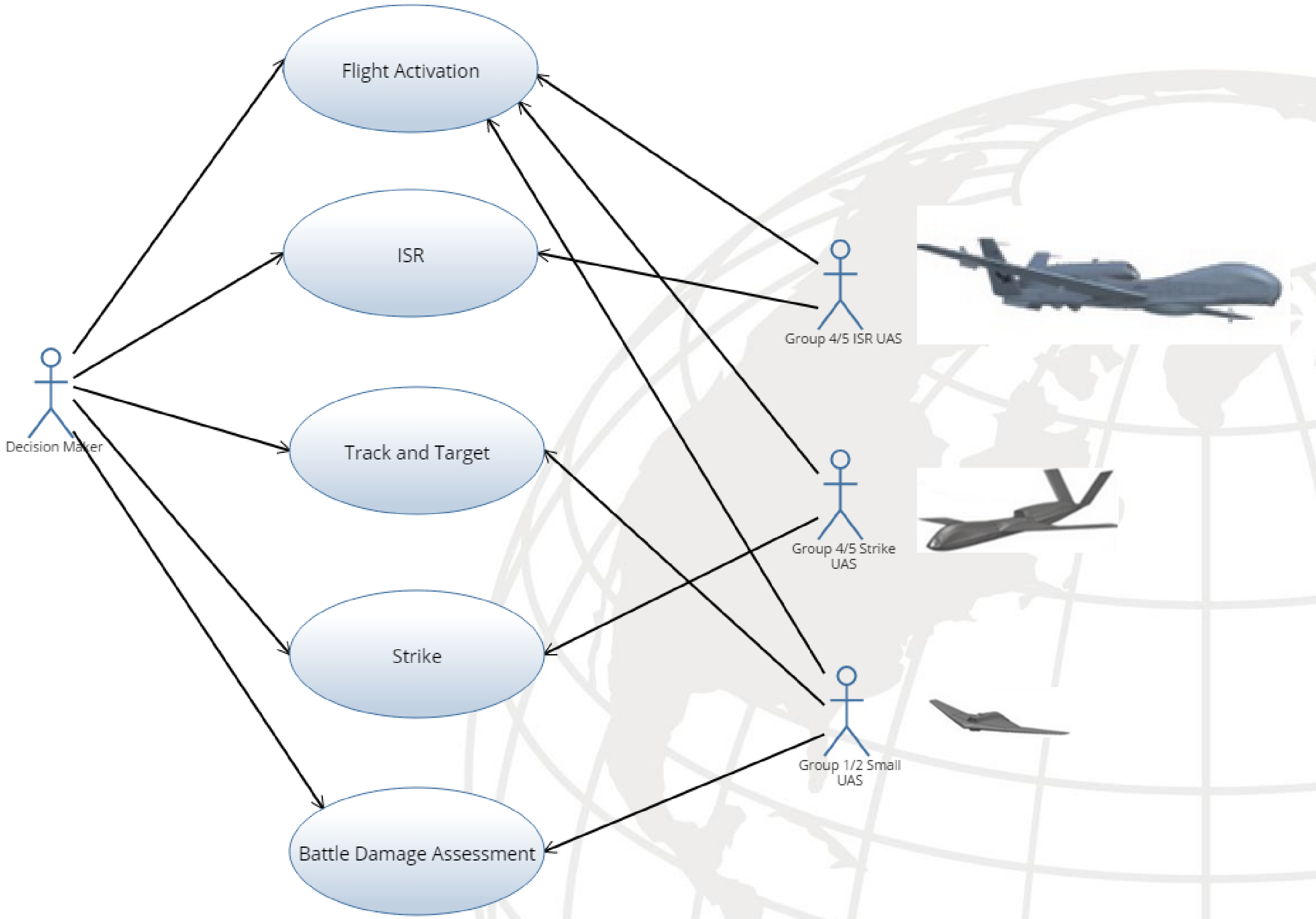
Command Post

(Mission Commander)

1010010010100101010111 1010010010100101011101001

Networked, Autonomous, Cooperative, Multi-Tiered UAS System-of-Systems for TBM Site Identification and Elimination

TBM 3-Tier UAS Scenarios (Use Cases with Threads)



Measures of Performance

1. Target Acquisition Pct

$$\mathit{Target\ Acquisition\ (Percentage)} = \frac{\mathit{Target\ Positively\ Acquired}}{\mathit{Total\ number\ of\ Targets\ encountered}} \times 100\%$$

2. False Alarm Pct

$$\mathit{False\ Alarm\ (Percentage)} = \frac{\mathit{False\ Target\ Acquired}}{\mathit{Total\ number\ of\ targets\ declared\ in\ area}} \times 100\%$$

3. Time-to-Strike

$$\mathit{Time\ to\ strike} = \mathit{Bomb\ launched\ Time} - \mathit{Target\ Acqusition\ Time}$$

4. Target Destruction Pct

$$\mathit{Target\ Destruction\ (Percentage)} = \frac{\mathit{Target\ Destroyed}}{\mathit{Total\ number\ of\ Targets\ encountered}} \times 100\%$$

Architectural Variants

Design Parameters	Variants		Effects
Decision-Capability	Manual C2	Autonomous C2	1. Speed of decision making 2. Quality of decision making
Sensor Capability	Normal Sensor	High End Sensor	1. Target acquisition 2. False Alarm
Number of Strike UAS deployment	1 x Strike UAS	2 x Strike UAS	1. Time-to-strike 2. Target destruction

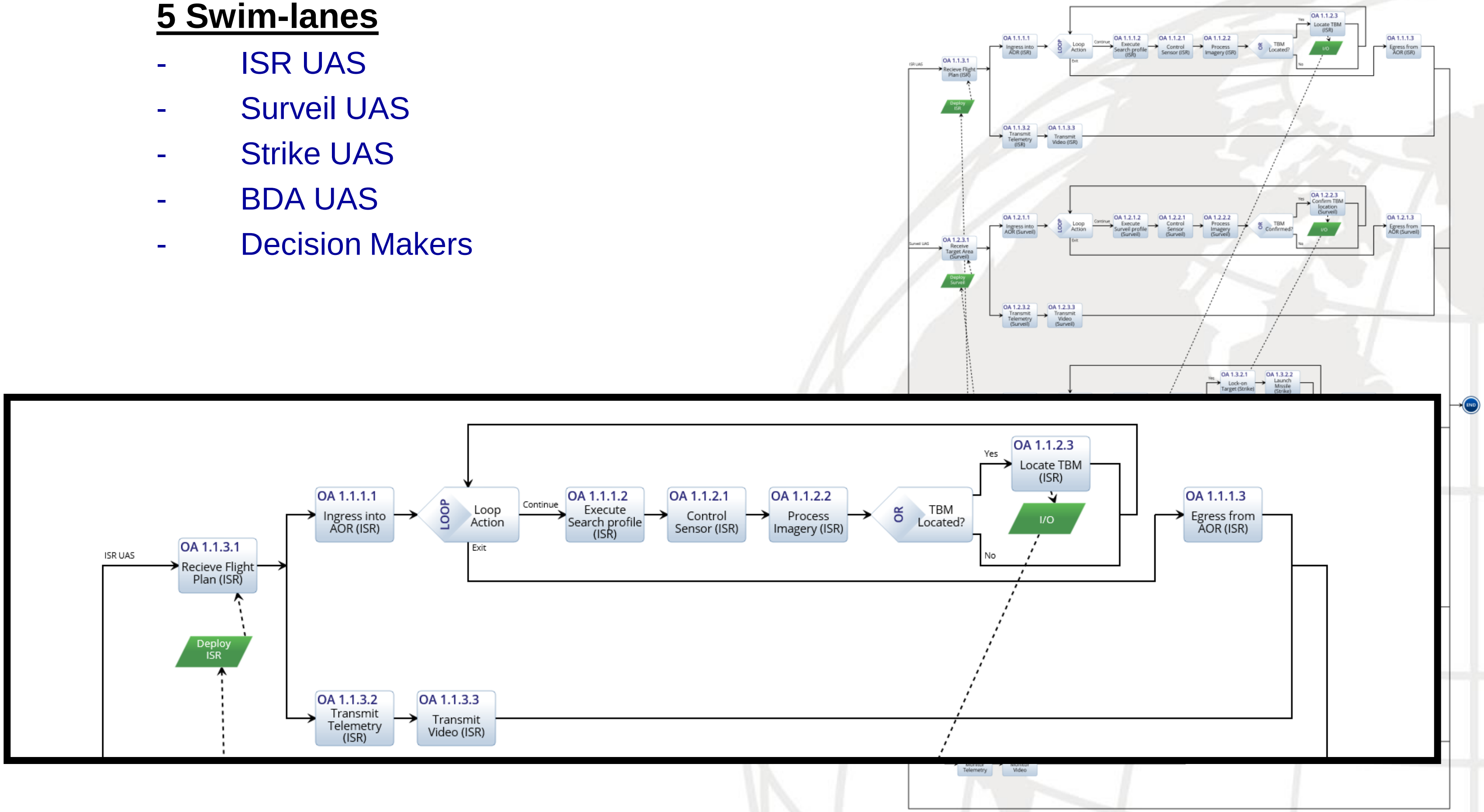
NAVAL POSTGRADUATE SCHOOL

Executable Activity Model

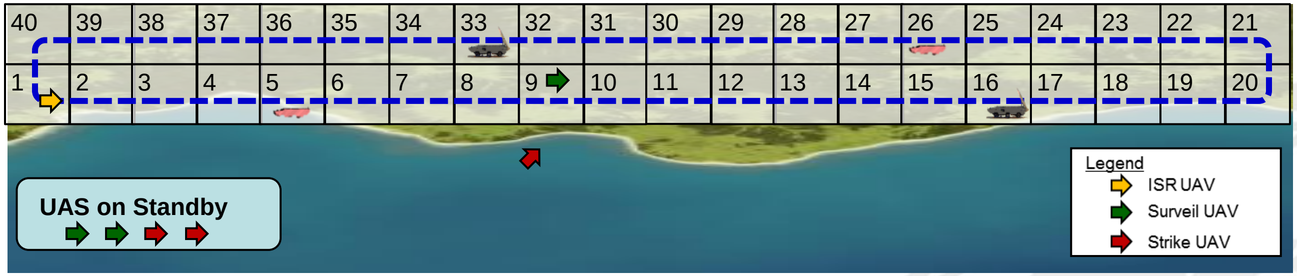
OV 5b: Operational Activity Model

5 Swim-lanes

- ISR UAS
- Surveil UAS
- Strike UAS
- BDA UAS
- Decision Makers



Simulation Scenario for Activity Model



Threat Assessment shows possible TBM deployment within Area of Operations (AO)
During each run, 2 x Targets and 2 x False targets randomly deployed over the 40 grids

Simulation Scenario

- 1 x ISR UAS deployed to conduct ISR [marked by ⇨]. Follow anti-clockwise search pattern over AO.
- When potential target are located, small UAS are deployed to Confirm and track target. Simulation limited to 2 x Surveil UAS [marked by ➡].
- Strike UAS deploy to strike target, once target confirmed [marked by ➡].
- Small UAS to conduct BDA [marked by ➡].

Total of 50 runs carried out per cycle, generating 100 targets and 100 false targets.
Total of 50 cycles executed as part of Monte Carlo simulation for each scenario.

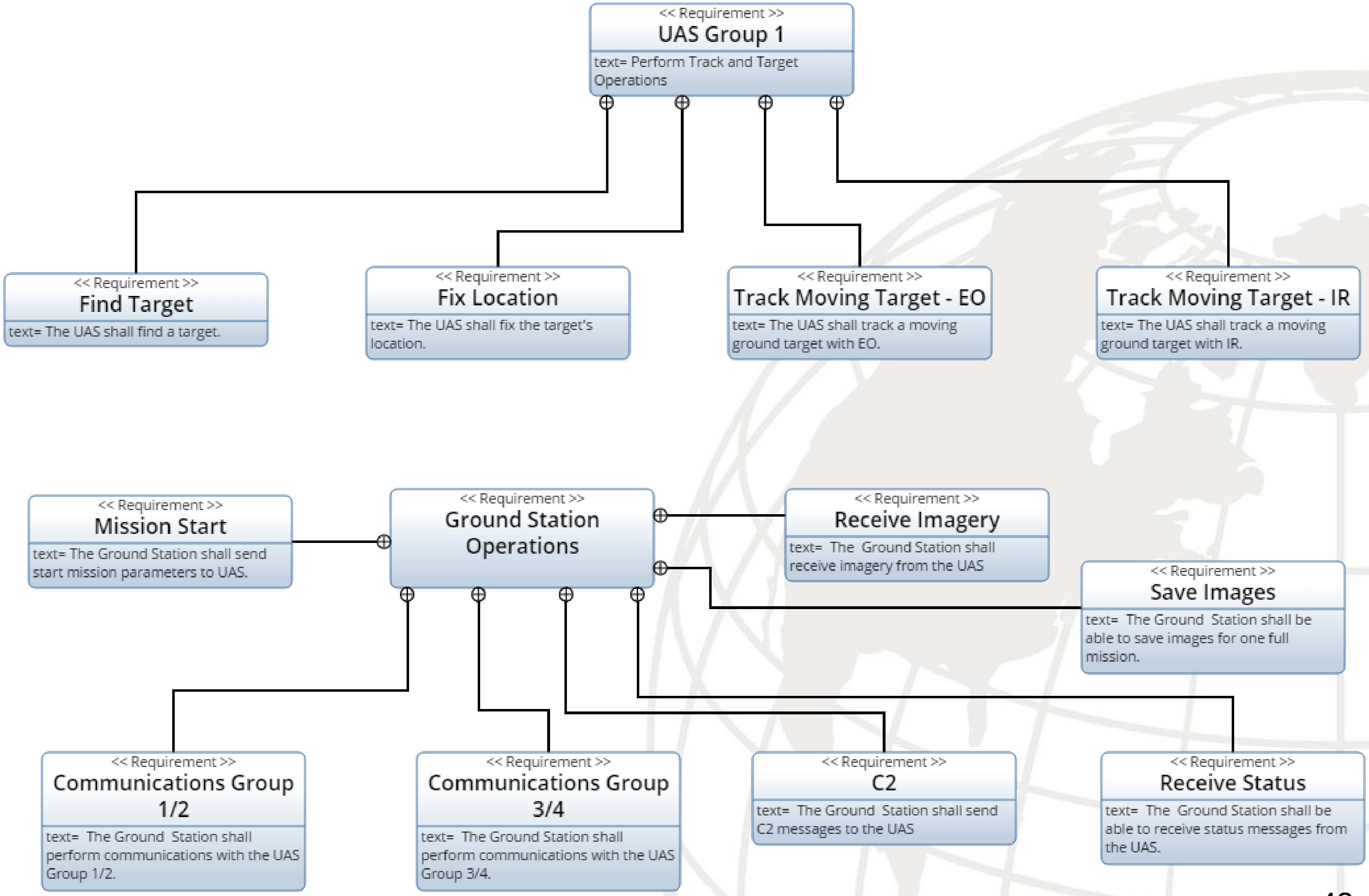
Total of 8 Simulation Scenarios

	Centralized Manual C2	Autonomous C2 Operations
Normal ISR Sensor	1 x Strike UAS	1 x Strike UAS
	2 x Strike UAS	2 x Strike UAS
High End ISR Sensor	1 x Strike UAS	1 x Strike UAS
	2 x Strike UAS	2 x Strike UAS

Summary of Results

MOP	Design Parameters	Simulation Results	Pct Improvement
Target Acquisition Percentage	Type of Sensor	High: **85.5%**	**61.5%** improvement over Normal Sensor
		Normal: **52.9%**	
False Alarm Percentage	Type of Sensor	High: **0.4%**	**95.6%** improvement over Normal Sensor
		Normal: **9.6%**	
Time-to-Strike	Type of C2	Autonomous: **91.2 mins**	**9.8%** improvement over Manual C2
		Manual: **100.1 min**	
	Number of Strike UAS	1 x Strike UAS: **94.6 min**	**2.1%** improvement over 2 x Strike UAS
		2 x Strike UAS: **96.9 min**	
Target Destruction Percentage	Type of Sensors	High: **75.1%**	**62.2%** improvement over Normal Sensor
		Normal: **46.3%**	
	Number of Strike UAS	1 x Strike UAS: **54.8%**	**21.7%** improvement over 2 x Strike UAS

TBM 3-Tier UAS Example Requirements



<< Requirement >>
UAS Group 1
text= Perform Track and Target Operations

<< Requirement >>
Find Target
text= The UAS shall find a target.

<< Requirement >>
Fix Location
text= The UAS shall fix the target's location.

<< Requirement >>
Track Moving Target - EO
text= The UAS shall track a moving ground target with EO.

<< Requirement >>
Track Moving Target - IR
text= The UAS shall track a moving ground target with IR.

<< Requirement >>
Mission Start
text= The Ground Station shall send start mission parameters to UAS.

<< Requirement >>
Ground Station Operations

<< Requirement >>
Receive Imagery
text= The Ground Station shall receive imagery from the UAS

<< Requirement >>
Save Images
text= The Ground Station shall be able to save images for one full mission.

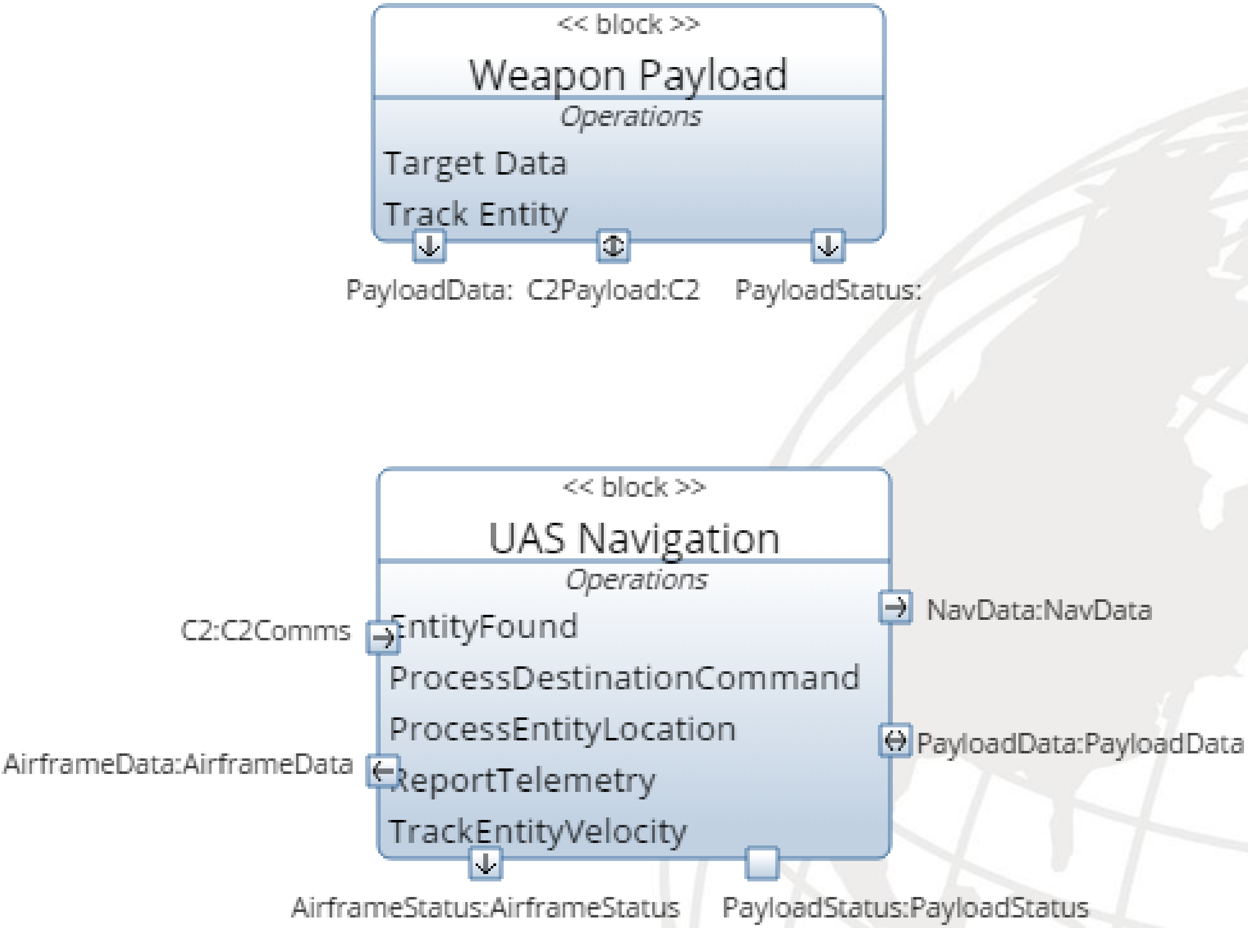
<< Requirement >>
Communications Group 1/2
text= The Ground Station shall perform communications with the UAS Group 1/2.

<< Requirement >>
Communications Group 3/4
text= The Ground Station shall perform communications with the UAS Group 3/4.

<< Requirement >>
C2
text= The Ground Station shall send C2 messages to the UAS

<< Requirement >>
Receive Status
text= The Ground Station shall be able to receive status messages from the UAS.

TBM 3-Tier UAS Example Interfaces (Ports)



<< block >>

Weapon Payload

Operations

Target Data

Track Entity

PayloadData: C2Payload:C2 PayloadStatus:

<< block >>

UAS Navigation

Operations

EntityFound

ProcessDestinationCommand

ProcessEntityLocation

ReportTelemetry

TrackEntityVelocity

C2:C2Comms

AirframeData:AirframeData

NavData:NavData

PayloadData:PayloadData

AirframeStatus:AirframeStatus

PayloadStatus:PayloadStatus

Agenda

- Introduction
- Cost Modeling Overview
- SysML Integration
- Case Study: Remote Targeting System UAS
- Case Study: TBM Identification and Elimination: 3-Tier UAS SoS
- Conclusions and Future Work

Systems Engineering Cost Model Sizing Correlation in MBSE Tools

Requirements
The number of requirements for the system-of-interest at a specific level of design.

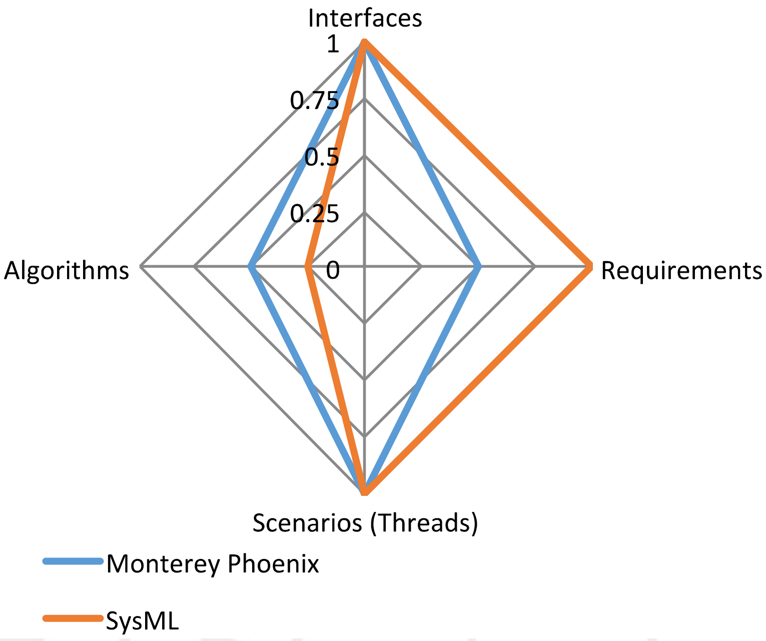
Interfaces
The number of shared physical and logical boundaries between system components or functions (internal interfaces) and those external to the system (external interfaces).

Algorithms
The number of newly defined or significantly altered functions that require unique mathematical algorithms to be derived in order to achieve the system performance requirements.

Operational Scenarios
Operational scenarios that a system must satisfy, including nominal and off-nominal threads.

These size drivers are further weighted for complexity levels.

System Size Correlation Strength



Conclusions and Future Work

- Have demonstrated architectural tradespaces with simpler UAS swarm models for further elaboration on more complex mission scenarios
- We have found a strong correspondence between SysML constructs and system size measures of requirements, interfaces, algorithms, and operational scenarios.
  - Still comparing approaches for complex algorithm representations in SysML
  - Require additional attributes for modeling complexity levels of size drivers
- Continue transcribing all UAS architectural variations into SysML for cost tradeoffs to evaluate with other Measures of Effectiveness
  - Expanded mission sets to include heterogeneous UAS teams and more complex scenarios
- Apply method and case study with other MBSE tools, evaluate and compare
  - More detailed modeling to support thread, requirements, functions, algorithms and interface definitions
- Develop guidelines with examples for practitioners on modeling decomposition levels of detail
- Continue essential research on integration of MBSE methods and tools
  - SysML, Monterey Phoenix (MP), DoDAF, COSYSMO, COCOMO, COQUALMO
  - Further tool integration with methods for file input, REST API web service and organic cost computation within SysML tools

References

- R. Madachy, *Systems Engineering Cost Estimation Workbook*, Naval Postgraduate School, October 2015
- D. Jacques and R. Madachy, "Model-Centric UAV ISR Analysis," presented at Systems Engineering Research Center, 7th Annual SERC Sponsor Research Review, Washington, DC, December 3, 2015.
- Maj. Ryan Pospisal (DTRA/A9, Kirtland AFB), "Application of Executable Architectures in Early Concept Evaluation", M.S. thesis, AFIT, December 2015
- Monica Farah-Stapleton, "Resource Analysis Based On System Architecture Behavior", Ph.D. thesis, NPS, June 2016
- CPT Dennis Edwards (USArmy), "Exploring the integration of COSYSMO with a model based system engineering methodology in early trade space analytics and decisions", M.S. thesis, NPS, June 2016
- Peak, R.S. and Lane, J.A., "SysML Building Blocks for Cost Modeling: Towards Model-Based Affordability Analysis", INCOSE International Workshop (IW14), Torrance, California, 2014

Completed Student Theses

- Maj. Zhongwang Chua (Singapore AF), “Application of Executable Architecture in Early Concept Evaluation using DoDAF”, M.S. Thesis, AFIT, September 2016
- Maj. Ryan Pospisal (DTRA/A9, Kirtland AFB), “Application of Executable Architectures in Early Concept Evaluation”, M.S. Thesis, AFIT, December 2015
- Monica Farah-Stapleton, “Resource Analysis Based On System Architecture Behavior”, Ph.D. thesis, NPS, September 2016
- CPT Dennis Edwards (USArmy), “Exploring the integration of COSYSMO with a model based system engineering methodology in early trade space analytics and decisions”, M.S. thesis, NPS, June 2016

Backups

Monterey Phoenix Overview

- Monterey Phoenix (MP) is approach to formal software and system specification based on behavior models
- A view on the architecture model as a high level description of possible behaviors of subsystems and interactions between subsystems
- The emphasis on specifying the interaction between the system and its environment
- The behavior composition operations support architecture reuse and refinement toward design and implementation models
- Executable architecture models provide for system architecture testing and verification with tools
- See http://wiki.nps.edu/display/MP

MBSE Environment Tradeoffs

- Do we want an all inclusive SE tool?
  - Some tools take this approach, but often provide substandard M&S, design and cost estimation environments
  - Some force the SE modelers into the realm of the design engineers – questions arise as to whether this makes best use of valuable resources
- Other approaches involve establishing traceability between the models in their respective environments
  - Architecture tools for early definition and establishing blueprints for development
  - M&S tools that can easily capture stochastic variables/events, rich dynamic interactions, and can perform Monte-Carlo analysis
  - Design tools capable of establishing feasibility, lower level performance analysis
  - Cost estimation tools that provide Cost Estimation Relationships (CERs) to support architecture decisions