34th Annual INCOSE international symposium

hybrid event

Dublin, Ireland
July 2 - 6, 2024

Model-Based Systems Engineering (MBSE) Methodology for Integrating Autonomy into a System of Systems Using the Unified Architecture Framework (UAF)

Mohammadreza Torkjazi
Ph.D. Candidate – Systems Engineering and Operations Research
mtorkjaz@gmu.edu

Dr. Ali K. Raz
Assistant Professor – Systems Engineering and Operations Research
Associate Director – Intelligent Systems and Integration – C5I Center
araz@gmu.edu

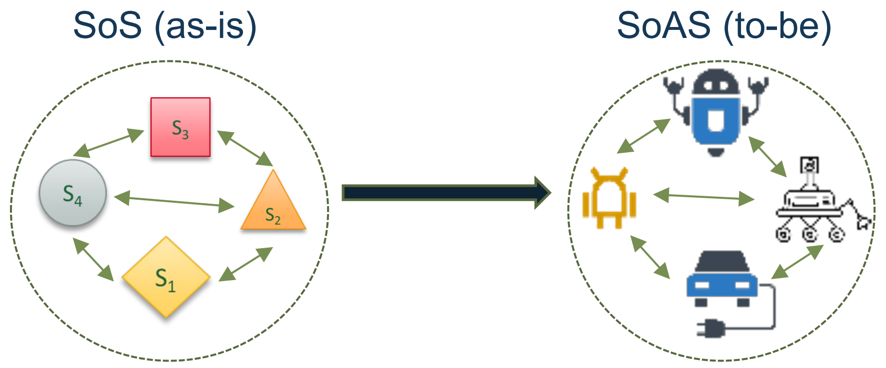
GEORGE MASON UNIVERSITY

RESEARCH OVERVIEW

- **Motivation**
  - Autonomy in engineered systems reduces human workload and involvement in hazardous missions.
  - Integrating autonomous systems into an existing System of Systems (SoS) evolves it into a System of Autonomous Systems (SoAS)
  - Autonomy comes in many levels (LoAs), each associated with uncertainty and risks that makes the SoAS integration and evaluation challenging

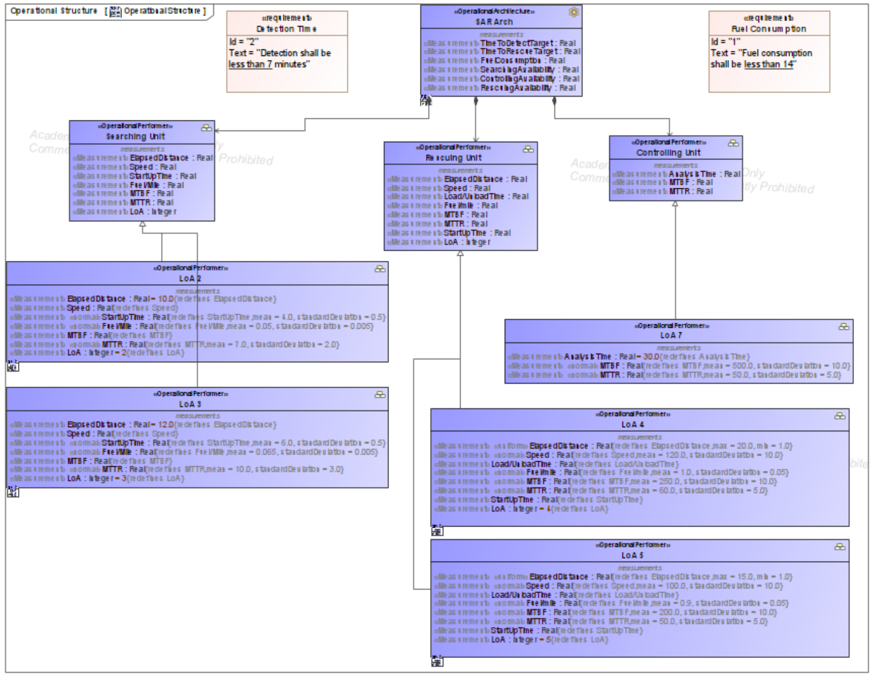
- **Contribution**

  Leverage the Unified Architecture Framework and Object Oriented Systems Engineering Method (OOSEM) and develop an MBSE Method to
  - conduct a comprehensive analysis within the MBSE environment to identify the LoA impacts, and
  - facilitate the integration of autonomy with existing SoSs



Constituent Systems are:
- Engineered systems
- Have well-Defined Behavior, V&V, …

Constituent Systems are:
- Autonomous engineered Systems with AI/ML

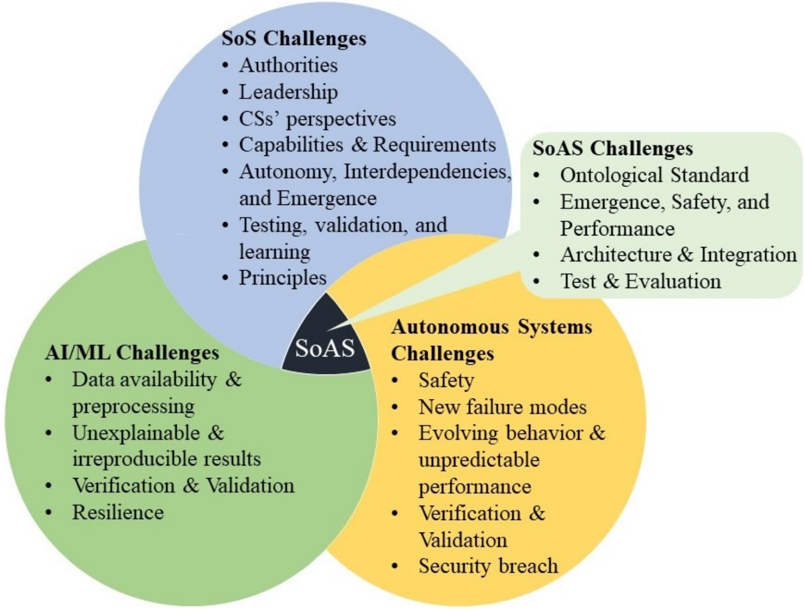


MBSE architecture with LoA

LEVELS OF AUTONOMY (LOA)

- Traditional definition of *autonomy* in SoS:
  - Managerial and Operational autonomy: Constituent systems operate and are managed independently.
- Definition of *autonomy* in AI and autonomous systems:
  - The ability of a system to sense, perceive, analyze, communicate, plan, make decisions, and act/execute, to achieve its goals as assigned independent of human intervention.
- LoA refers to a set of autonomous capabilities provided by a system, depending on its AI technology.

Levels	Definition
Hands-on	*Supported by driver assistance abilities such as lane-keep assist, auto cruise control, parking support*
Hands-off	*Supported by an autopilot requiring constant attention*
Eyes-off	*still Supported by a human under any emergency situations with communication by speech, gesture control, or via a touchscreen*
Mind-off	*No human intervention*
Fully Autonomous	*No human intervention, no steering wheel, no pedals, no breaks, even no windshield*

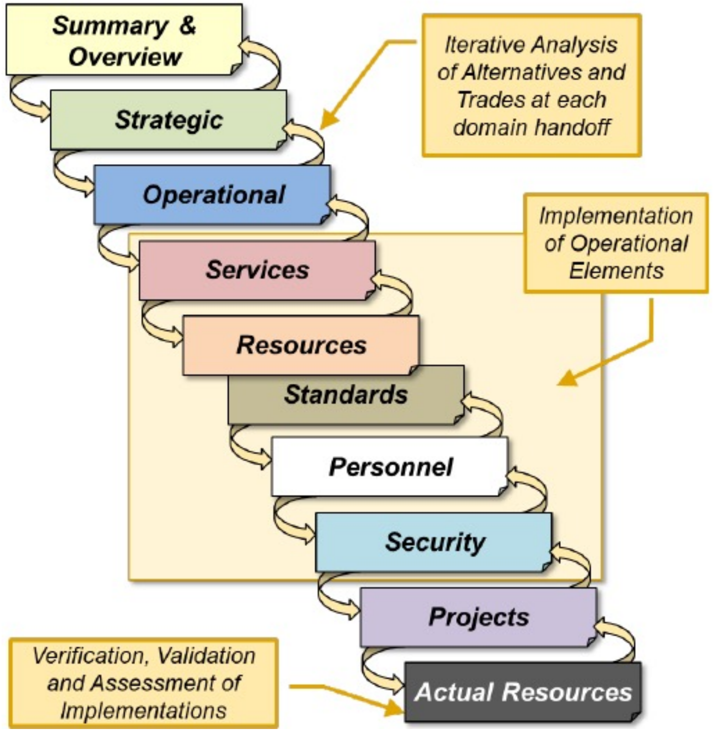


The concept of LoA is missing in SoS, as defined in AI/ML literature, but it is crucial to be considered in SoAS

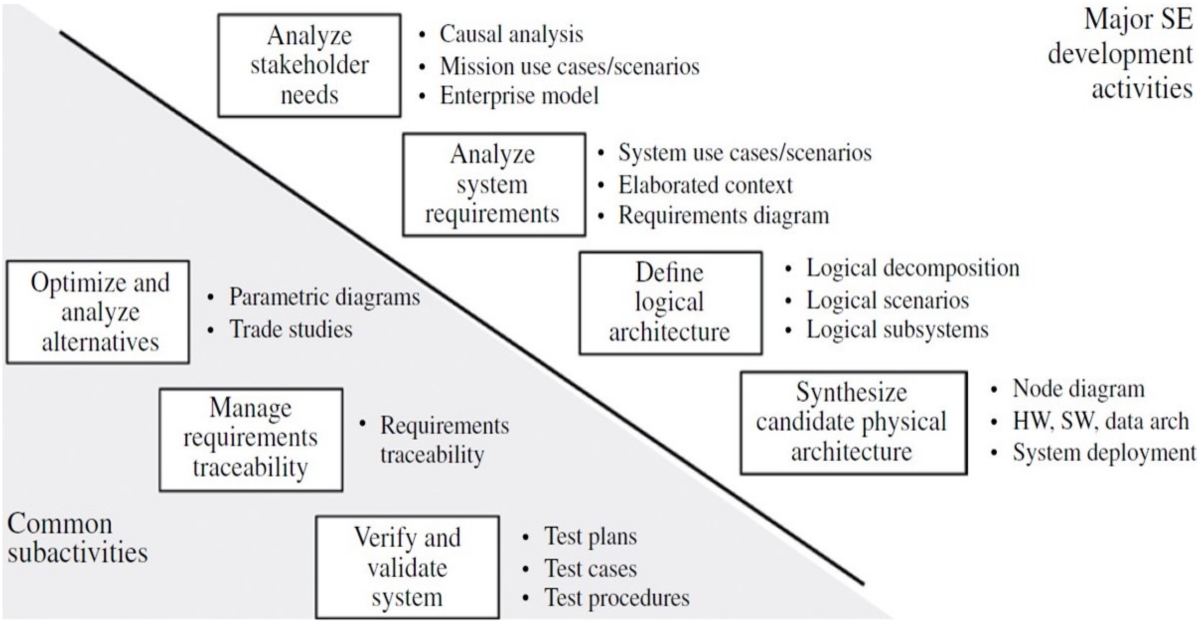
PROBLEM MOTIVATION and SOLUTION APPROACH

- Autonomous engineered systems needs to be integrated into existing SoSs to improve mission capabilities
- Varying LoAs in systems adds to the SoS complexities and lead to issues in terms of interoperability (e.g., incompatibility of interface designs), regulatory policies (e.g., maximum allowed LoA within SoS), etc.
- There is a need to analyze the impacts of varying LoAs on the SoS current operations, and then, identify suitable LoAs with a Systems Engineering methodology

Unified Architecture Framework



OOSEM

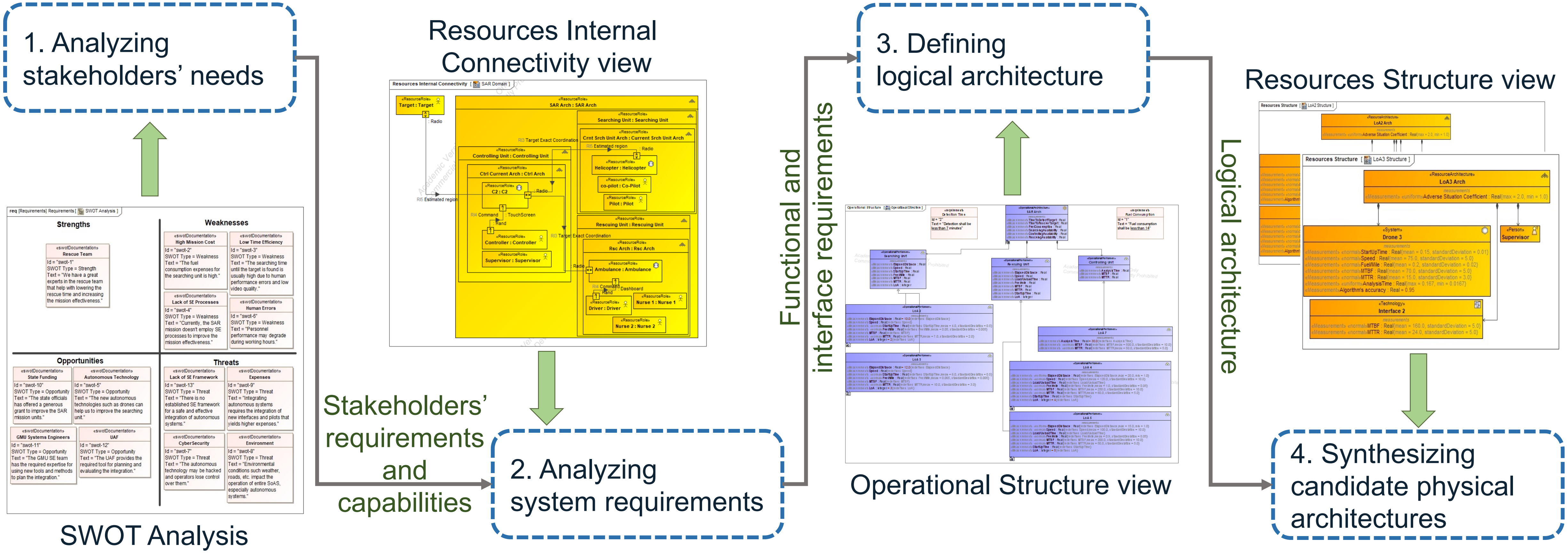


Offers various viewpoints for SoASs' levels of abstraction but does not offer an architecting methodology.

Top-down methodology that integrates the object-oriented concept with MBSE and each step needs to be updated with LoA consideration

THE PROPOSED MBSE METHOD

- **Objective:** Facilitate the integration of varying LoAs into an existing SoS.



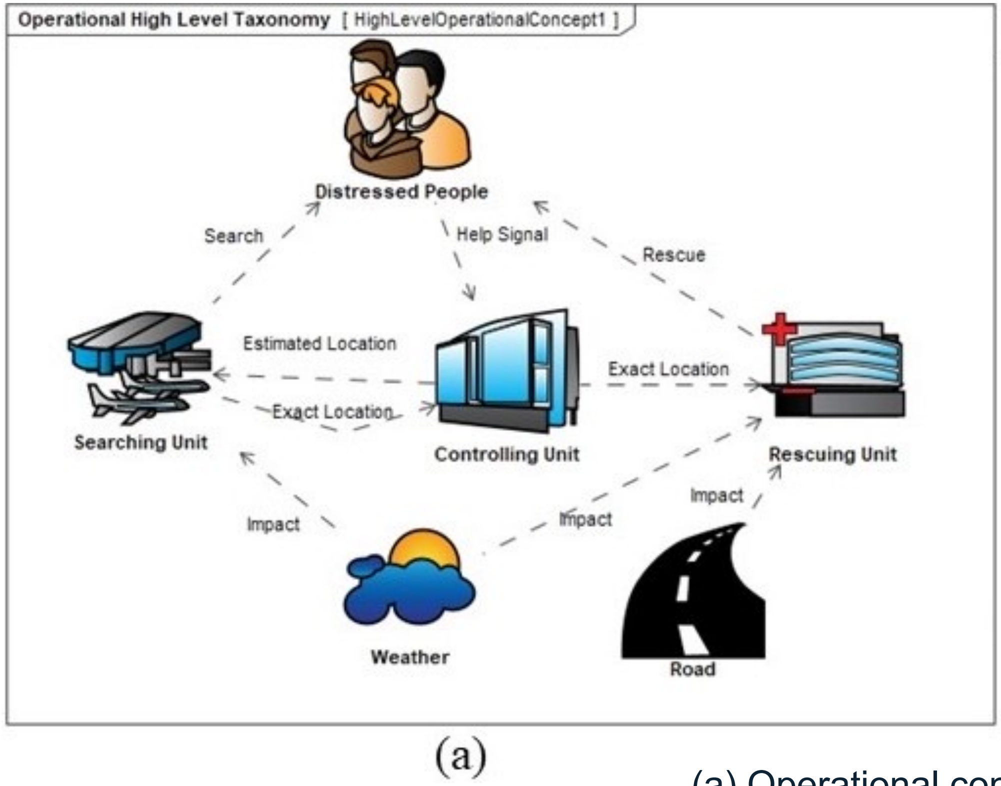
The proposed method provides step-by-step guidance on how to begin the initial analysis, how to model the SoAS architecture, what UAF views to build, and what outputs to deliver in each step

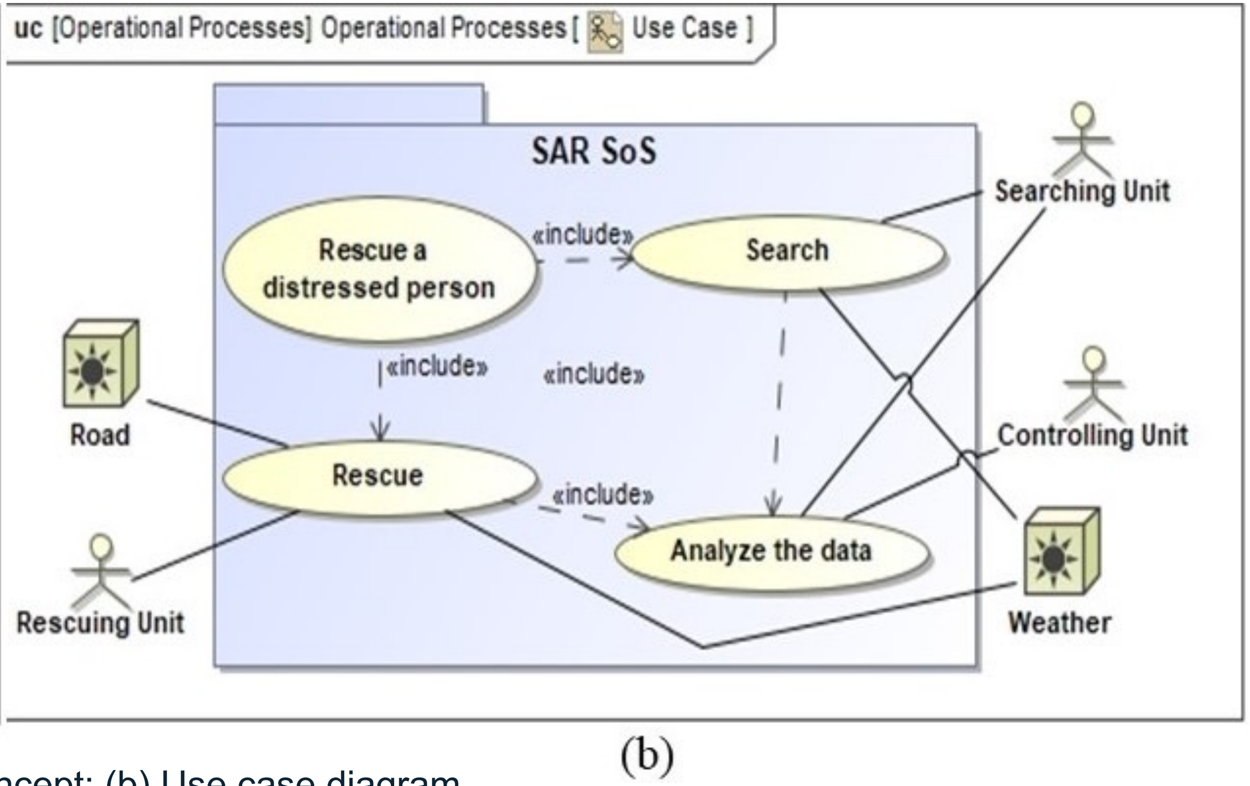
SUB-STEPS, DIAGRAMS, AND OUTPUTS IN THE PROPOSED METHOD

Step	Rationale	Sub-Steps		Diagram	Output
1. Analyze Stakeholders' Needs	Identifying the potential replaceable modules and CSs and Understanding mission and stakeholders' requirements	1.A	1.A.1	Operational High Level Taxonomy, SysML use case,	Current SoS operations
			1.A.2	Resources Structure	Current SoS architecture
			1.A.3	Resources Process Flow	Current SoS activities
		1.B		Profile, Generic table, Requirement	SWOT artifacts
		1.C		-	Replaceable modules & CSs
		1.D.a.	1.D.a.1 1.D.a.2 1.D.a.3 1.D.a.4	Strategic Motivation	SoAS capabilities
		1.D.b		Risks, Security Structure	Autonomy risks and mitigations
		1.E		Strategic Taxonomy	MOEs
		1.F		Requirement table	High-level requirements
2. Analyze System Requirements	Understanding the requirements for the replaceable modules and CSs	2.A		Resources Internal Connectivity, Resources states	Functional and interface requirements
		2.B			
		2.C		Requirement table	
3. Define Logical Architecture	Defining the logical architecture of replaceable modules and CSs	3.A		Operational Structure	Current SoS structure
		3.B			MOPs
		3.C		Operational Structure, Operational Internal Connectivity	Logical architecture of replaceable module
		3.D		Operational Process Flow	
4. Synthesize Candidate Physical Architectures	Developing alternative physical architectures for the replaceable modules and CSs	4.A		Implementation matrix	Candidate resources with varying LoAs
		4.B		Resources Structure, Resources Process Flow	Physical architectures with varying LoAs
		4.C			TPMs for different LoAs
		4.D		Operational Structure	SoAS Architecture

CASE STUDY: SEARCH AND RESCUE (SAR) SOS

- Assume that the current SoS operations result in low-efficiency rates of fuel and the stakeholders desire to investigate improvement alternatives for the systems.
- One approach is using new autonomous systems available in the market that consume less fuel.

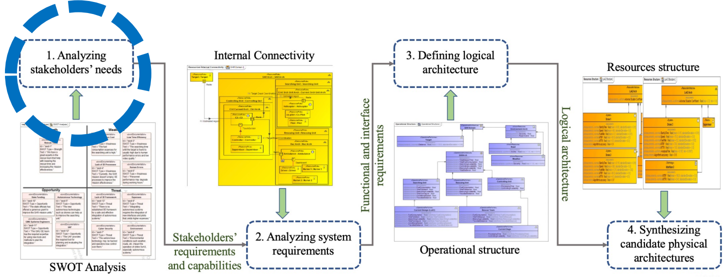




(a) Operational concept; (b) Use case diagram

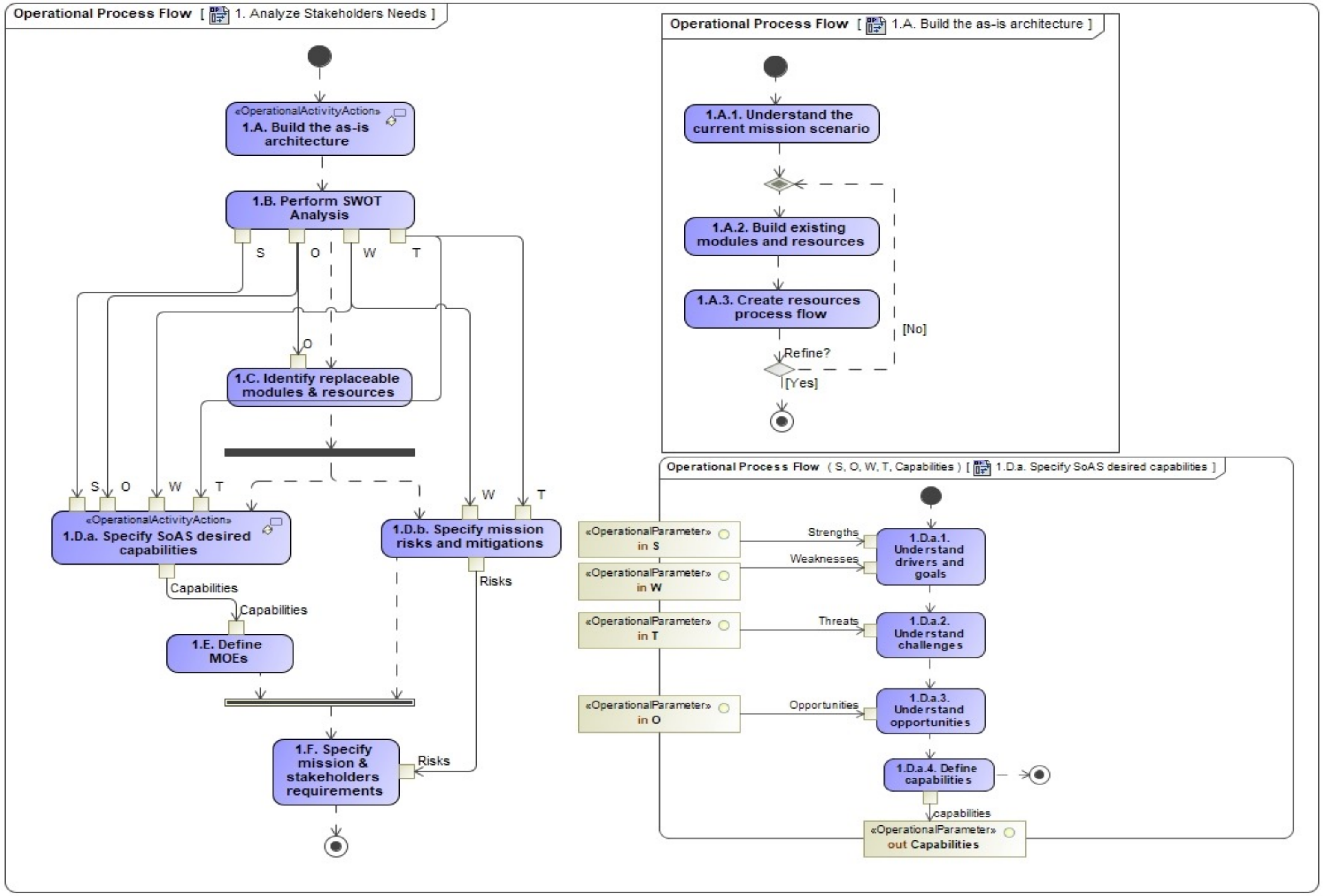
The objective is to identify the potential systems to be replaced and investigate whether integrating varying LoAs improves the Measures Of Effectiveness (MoEs).

STEP 1: ANALYZING STAKEHOLDERS' NEEDS

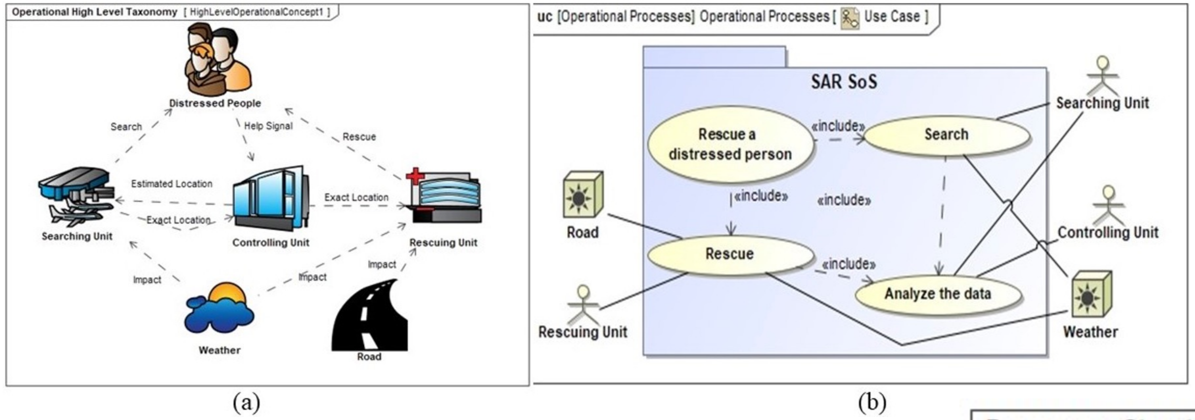


- **Purpose:**
  - Understanding the SoS "as-is" architecture
  - Identifying the desired stakeholders' capabilities
  - Identifying the potential resources to be replaced with autonomous technology

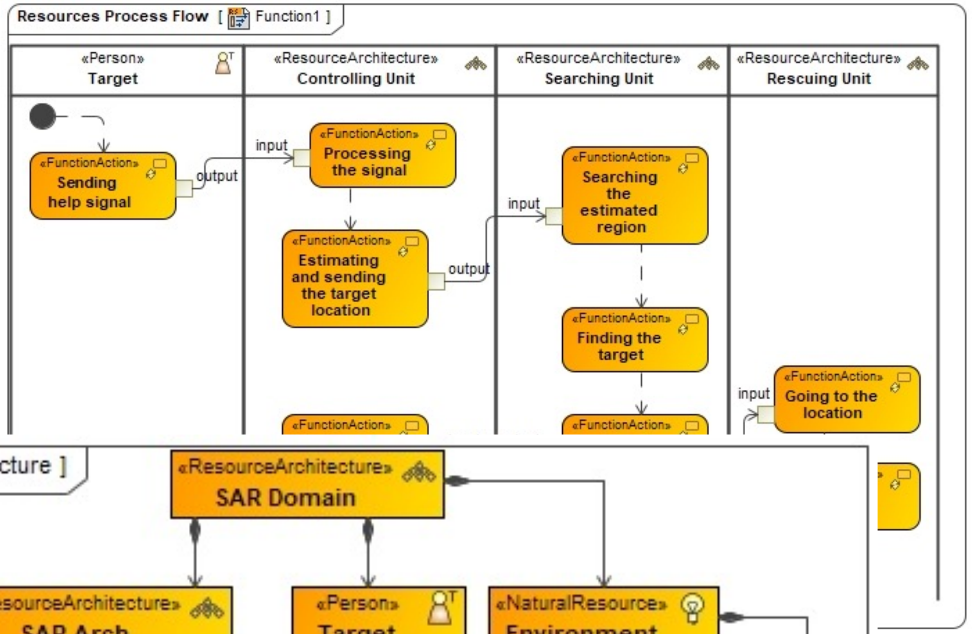
- **Diagrams used:**
  - 1.A.1. Operational High Level Taxonomy, SysML Use Case
  - 1.A.2. Resources Structure
  - 1.A.3. Resources Process Flow
  - 1.B. Profile, Generic Table, Requirement
  - 1.D.a. Strategic Motivation
  - 1.D.b. Risks, Security Structure
  - 1.E. Strategic Taxonomy
  - 1.F. Requirement table



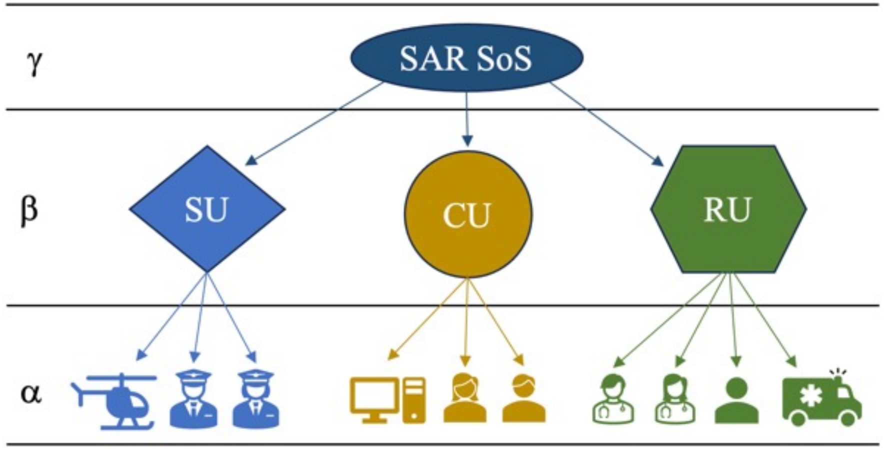
CASE STUDY: STEP 1.A



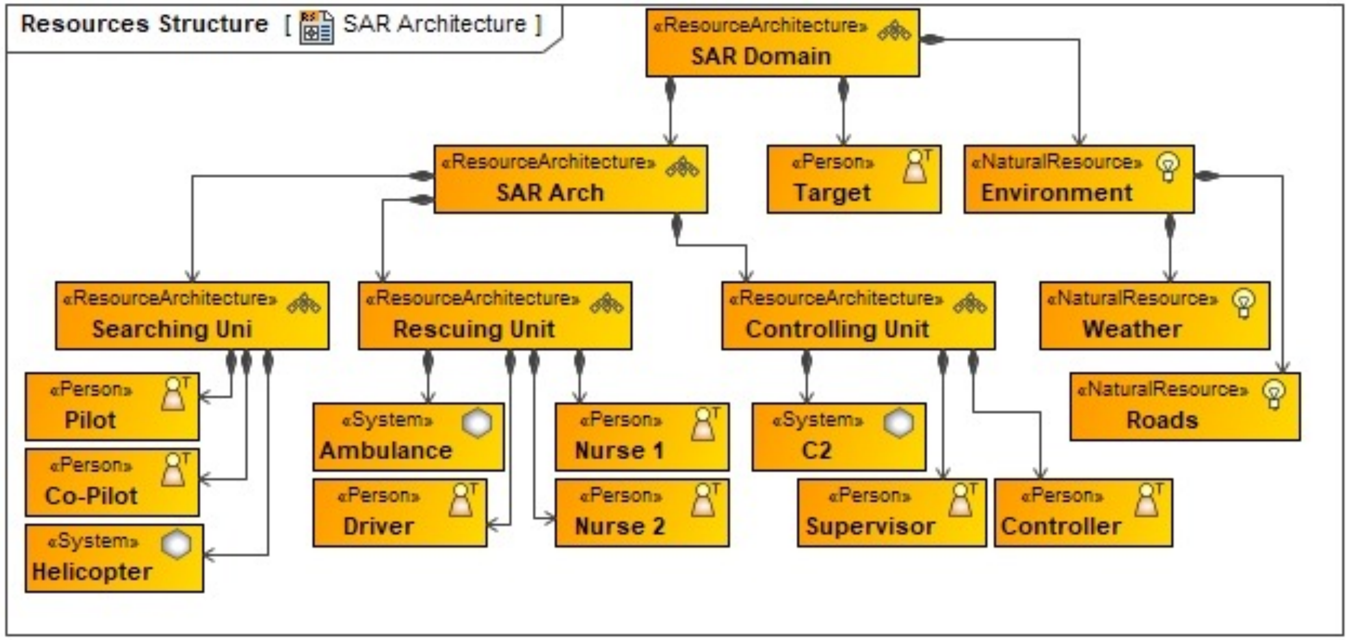
(a) Operational concept; (b) Use case diagram



Resources Process Flow



Levels of the SoS based on the lexicon

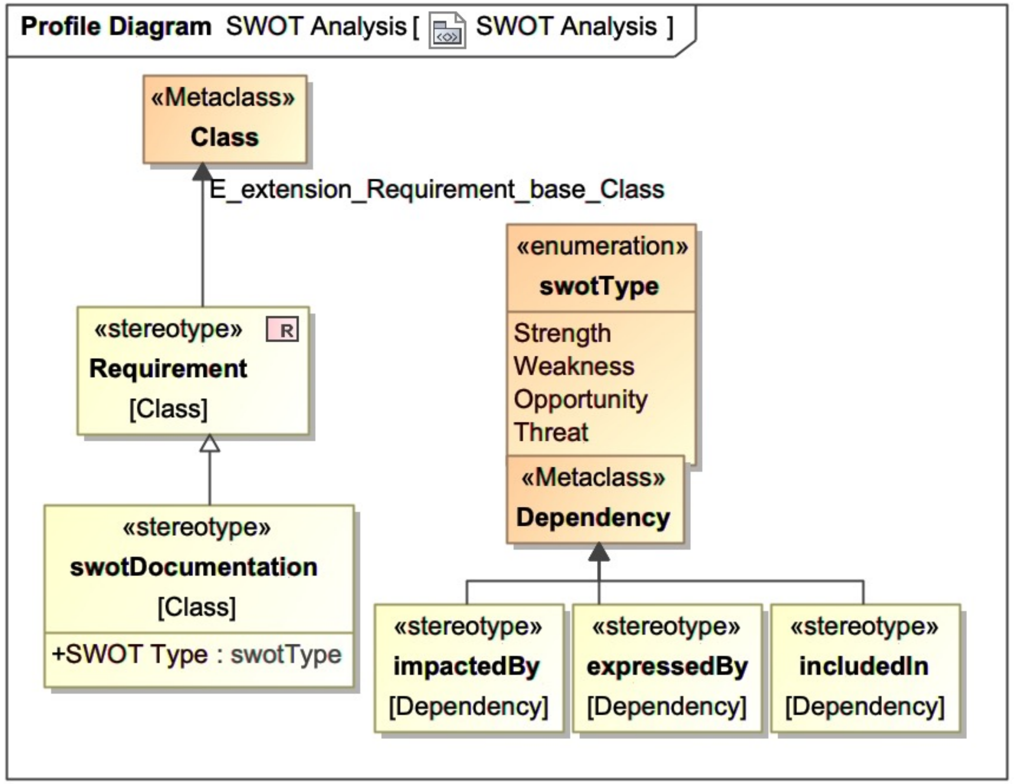


Resources Structure

Output:
SoS "as-is" architecture and activities

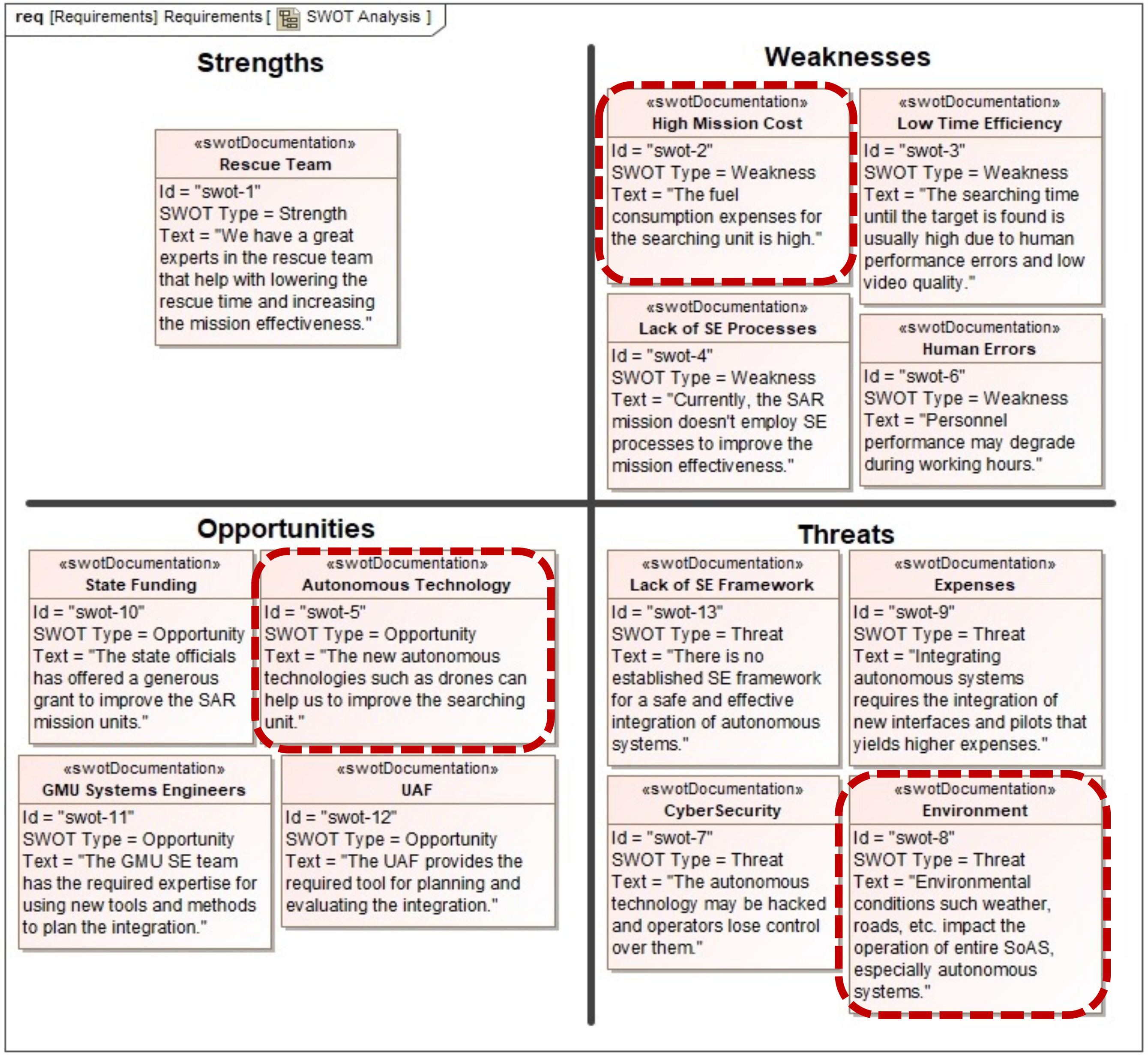
CASE STUDY: STEP 1.B (SWOT Analysis) and 1.C

- Extended the UAF Profile to provide the MBSE SWOT analysis



The UAF Profile Extension for including the MBSE SWOT analysis

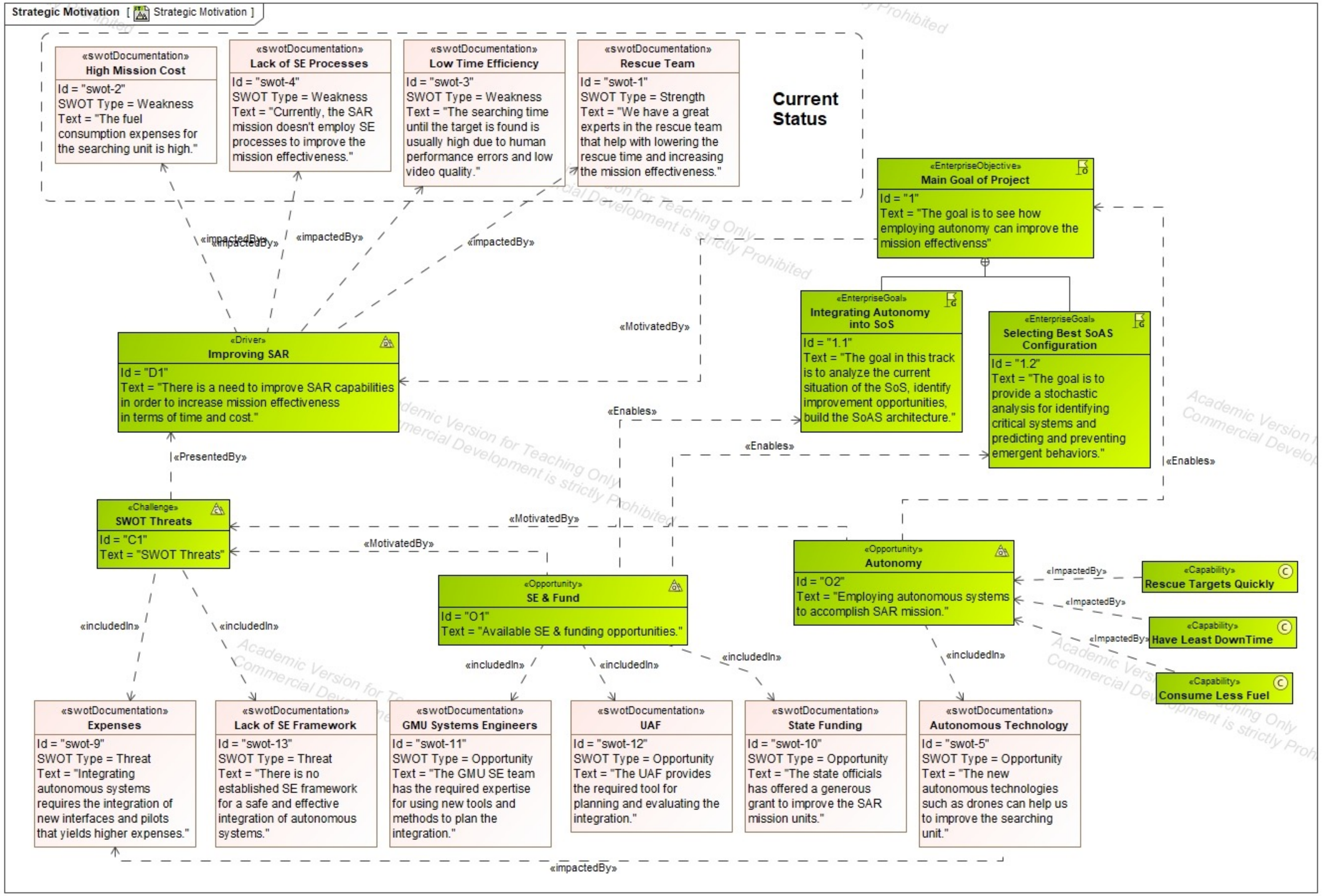
Output:
SWOT artifacts



CASE STUDY: STEP 1.D.a

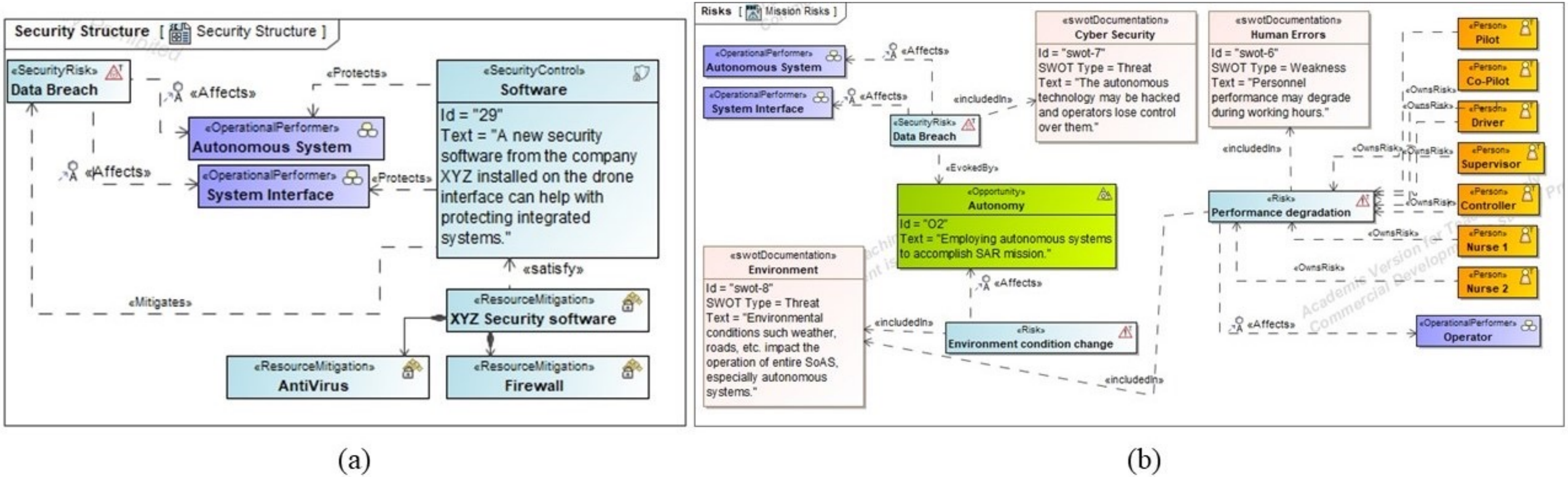
- SWOT helped with identifying the helicopter and the Searching Unit as the potential replaceable system and module respectively.
- SWOT helped uncover two more capabilities that are impacted by autonomy integration: time and availability. (Only the fuel capability was the initial intention of stakeholders)

Output:
SoAS capabilities and replaceable system and module



CASE STUDY: STEP 1.D.b

- Analyzing risks of autonomy integration, the potential and current resources associated with these risks, the resources that can handle these risks, and the affected opportunities.



(a) Analyzing the security risk of data breach; (b) Understanding various risk factors impacting the future SoAS

Output:
Autonomy risks and mitigations

CASE STUDY: STEP 1.E and 1.F

- Strategic Taxonomy that summarizes identified capabilities and their corresponding MOEs.
- Requirement table that shows the identified mission and stakeholders’ requirements.

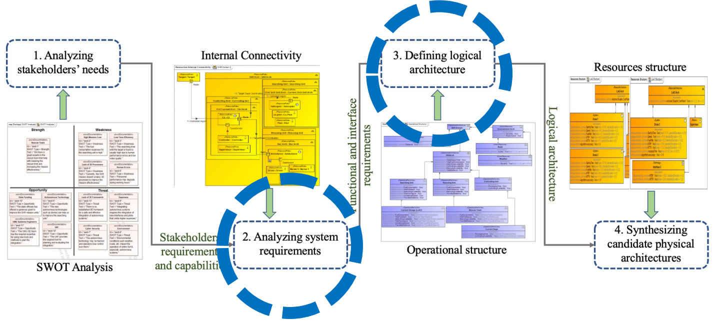
Strategic Taxonomy [Strategic Taxonomy]

Requirement Table [Requirements]

#	Name	Text
1	1 Mission Time	Total mission time shall be less than 1.5 hour
2	2 Security breach	The autonomous systems and their interface shall be secure to cyber attacks
3	3 Fuel Consumption	TotalFuelConsumption shall be less than 8

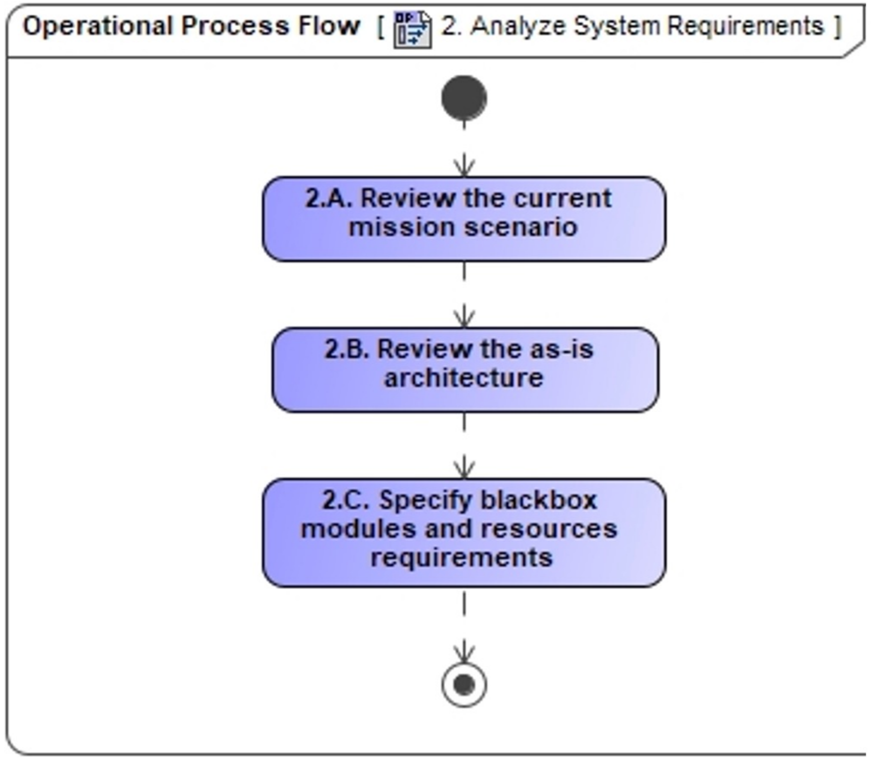
Both the MOEs and identified risks will be used in specifying the mission and stakeholders’ requirements to guide the architecting phase

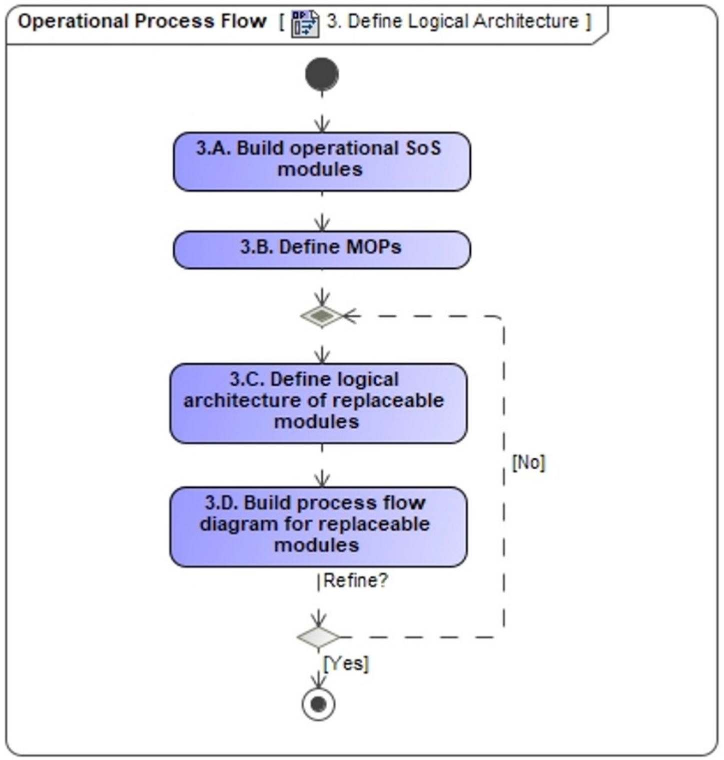
STEP 2: ANALYZING SYSTEM REQUIREMENTS
STEP 3: DEFINING LOGICAL ARCHITECTURE



- **Purpose:**
  - **Step 2:** Understanding the functional and interface requirements for the replaceable modules and systems
  - **Step 3:** Defining the logical architecture of replaceable modules and CSs
- **Diagrams used:**
  - 2.A. and 2.B. Resources Internal Connectivity, Resources states
  - 2.C. Requirement table

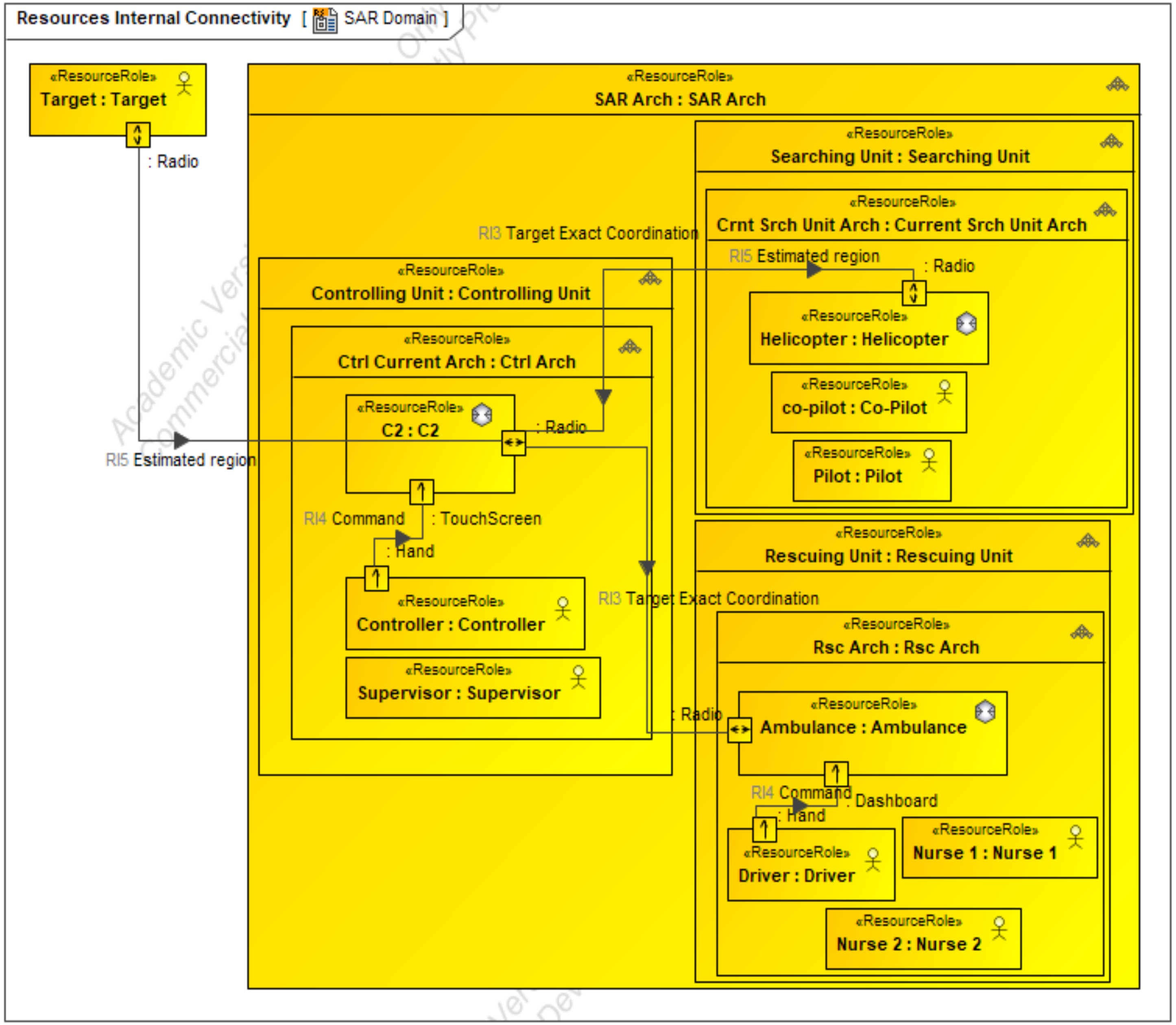
  - 3.A. and 3.B. Operational Structure
  - 3.C. Operational Structure, Operational Internal Connectivity
  - 3.D. Operational Process Flow





CASE STUDY: STEP 2

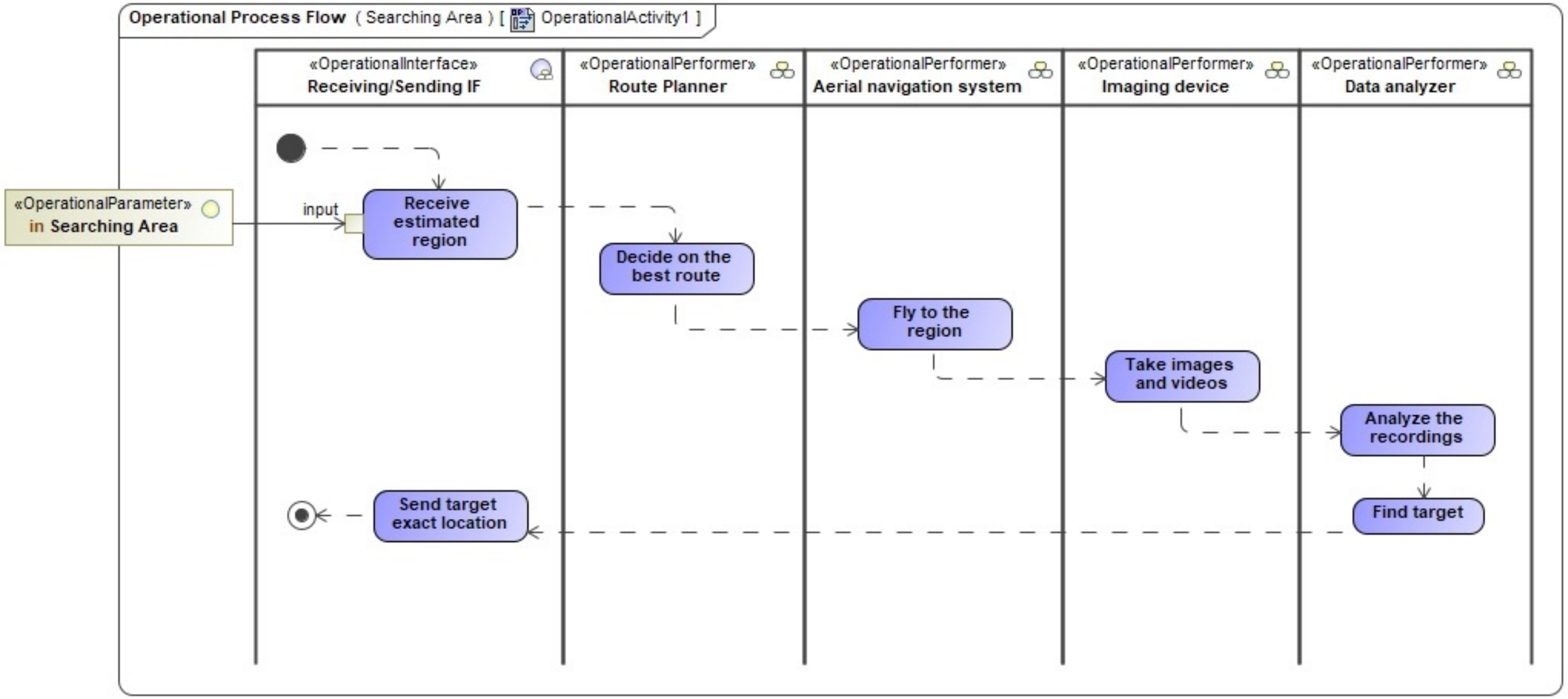
- It shows the interfaces and the type of data exchanged between the SU module and systems in other modules.
- One takeaway:
    - The new design of SU must be able to handle the type of input/output data provided by the Radio interface between SU and other systems.



Output:
Updated Requirement table by adding the identified functional and interface requirements

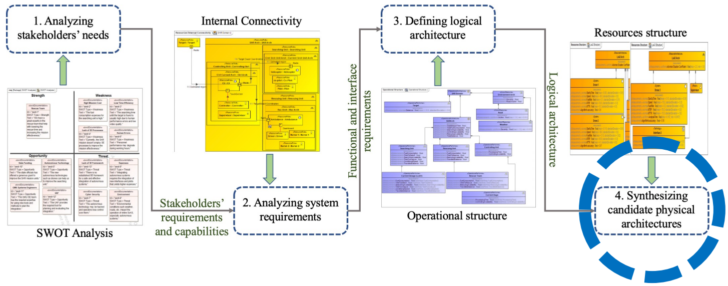
CASE STUDY: STEP 3

- The logical architecture of the Searching Unit gives insight into what entities are needed for executing the required functions of this module.
- The functions are obtained in Step 2 by identifying functional requirements.

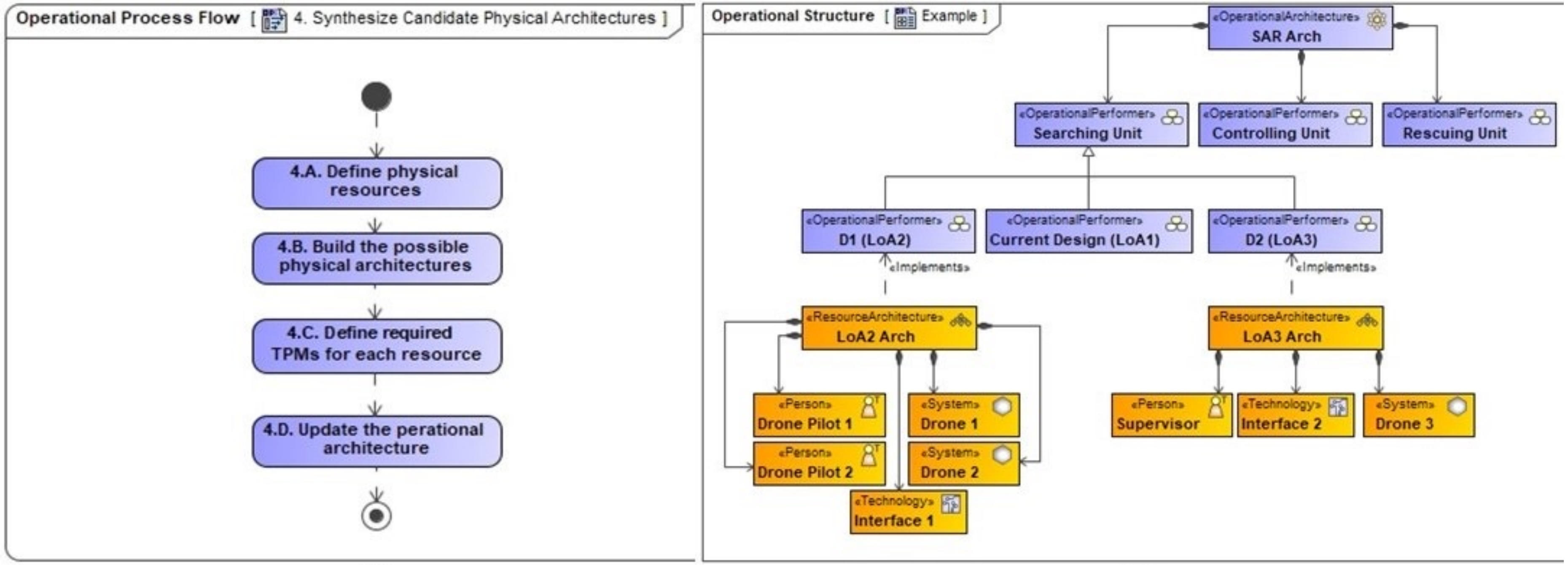


Output:
Logical architecture of the replaceable module

STEP 4: SYNTHESIZING CANDIDATE PHYSICAL ARCHITECTURES



- **Purpose:**
  - Developing alternative physical architectures for the replaceable module and systems
- **Diagrams used:**
  - 4.A. Implementation matrix
  - 4.B. and 4.C. Resources Structure, Resources Process Flow
  - 4.D. Operational Structure



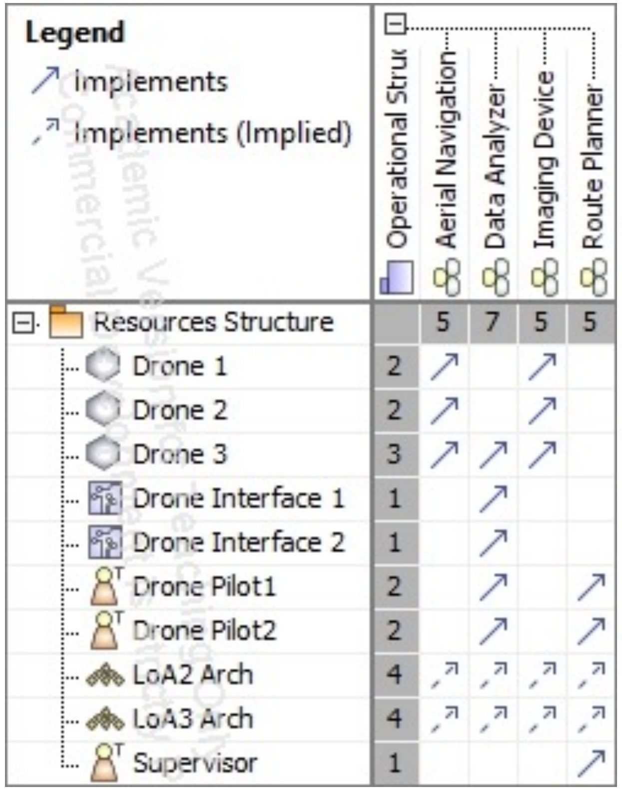
(a) (b)

(a) Sub-steps for synthesizing candidate physical architectures;
(b) An example of incorporating alternative physical architectures

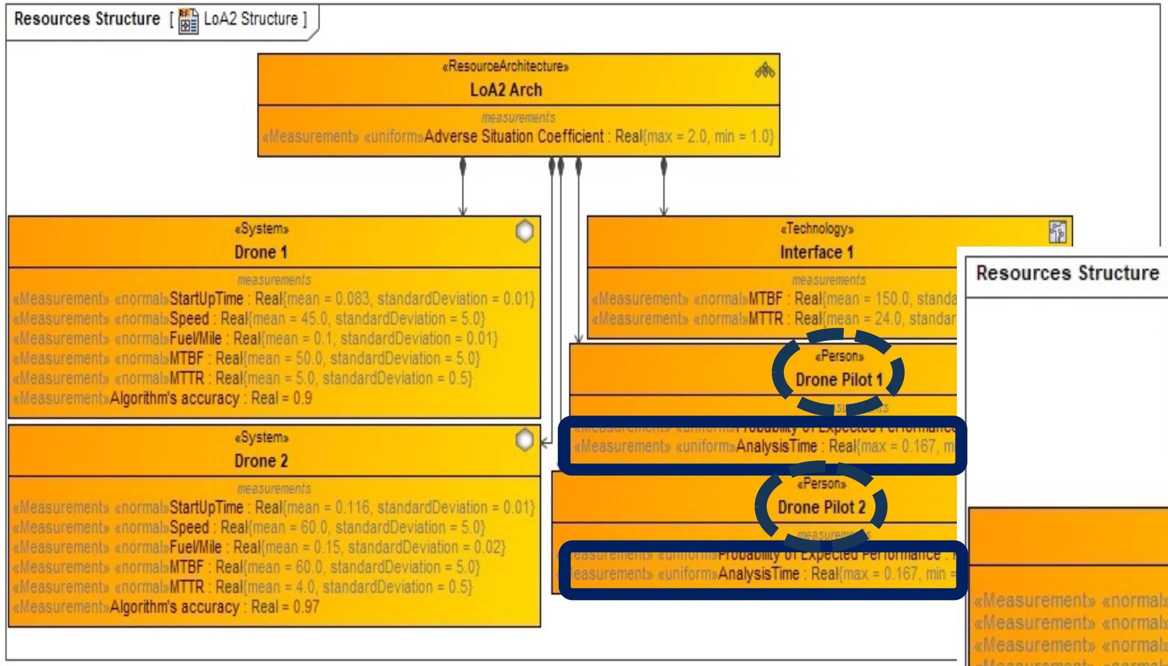
Step 4 defines various physical resources along with their TPMs that can implement the developed logical architecture in Step 3

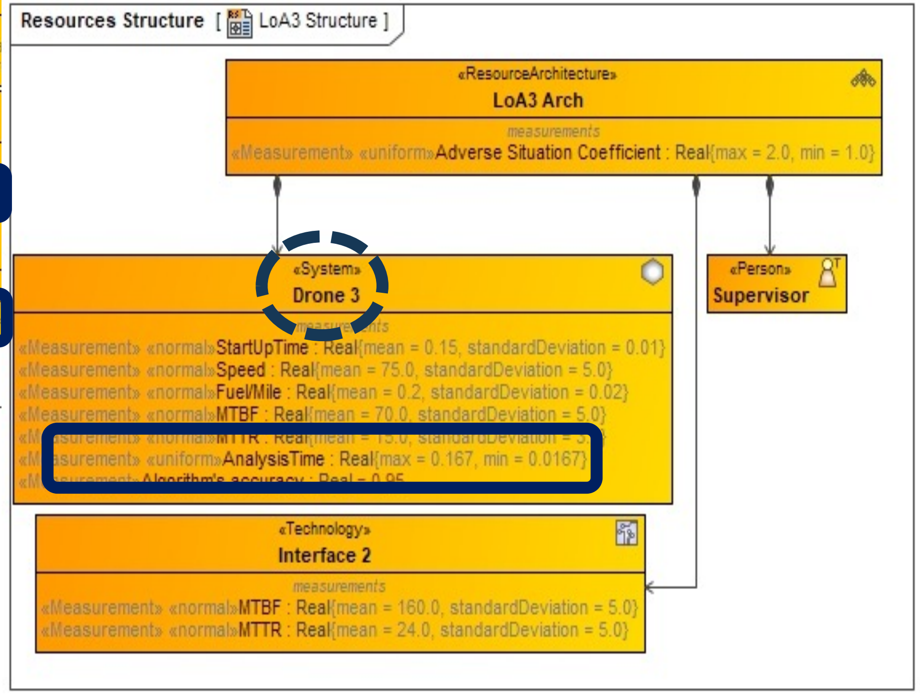
CASE STUDY: STEP 4.A, 4.B, AND 4.C

- Different LoAs were defined based on the number of logical entities an alternative physical architecture can implement.
- TPMs were defined as random variables to better demonstrate real performances.



Identifying possible resources that are able to implement logical entities and their corresponding functions

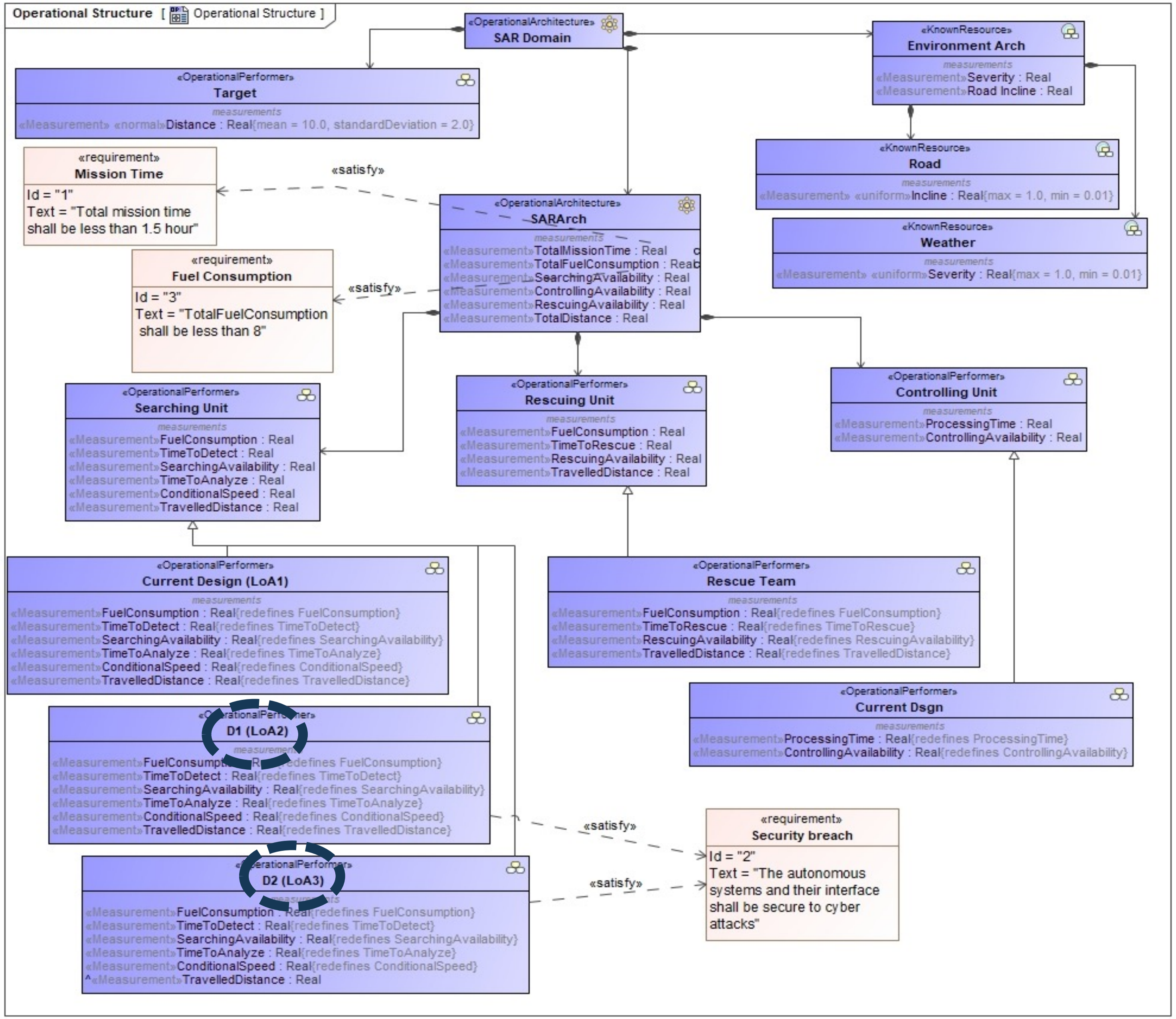




Two Possible alternative physical architectures

CASE STUDY: STEP 4.D

- The operational architecture to include the modules associated with the alternative physical architectures and their corresponding MOPs.
- The resulting SoAS architecture facilitates the future analysis that needs executable UAF models to conduct trade studies.



Output:
Multiple SoAS Architectures to be evaluated

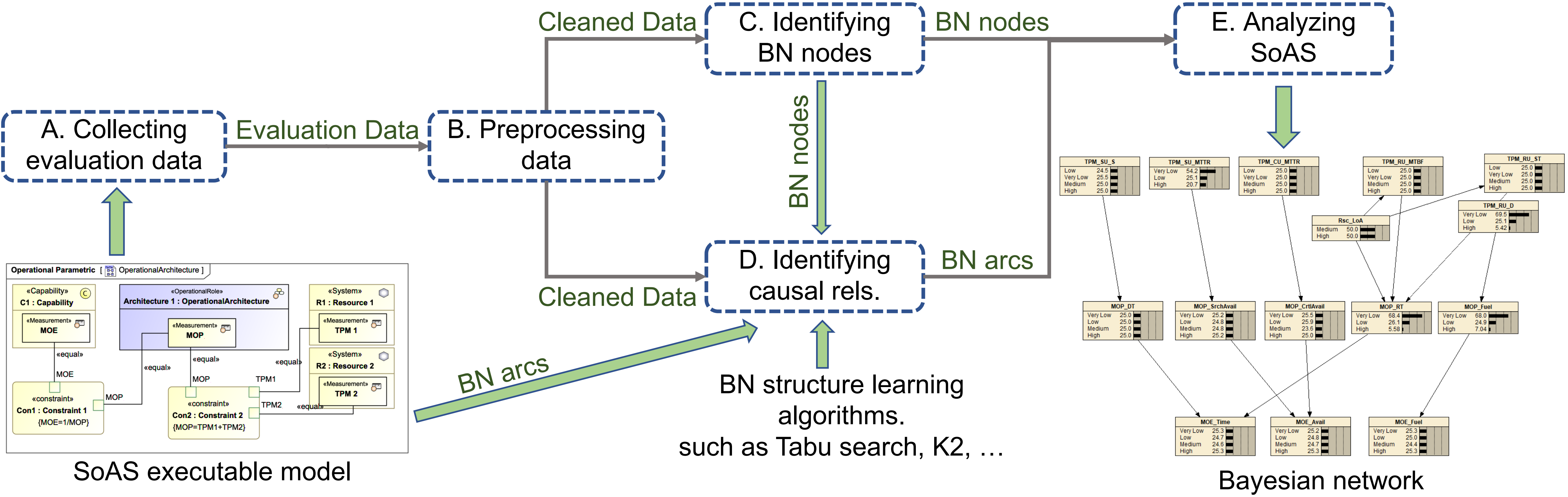
CONCLUSIONS & NEXT STEPS

- **Conclusions:**
  - LoAs in systems exacerbate architecting challenges of SoAS in terms of organizational (i.e., policies) and technical (i.e., interoperability) aspects.
    - The UAF can handle the added complexities in SoAS integration by offering new views and viewpoints that cover different SoAS architecture levels (e.g., *Security* viewpoint)
  - An MBSE architecting methodology for Autonomy Integration is proposed by employing OOSEM within the UAF
    - OOSEM ensures the compliance of the methodology with Systems Engineering standards, while the UAF assists with the modeling phase.
    - The proposed method establishes step-by-step guidance on how to begin the initial analysis, how to model the SoAS architecture, what UAF views to build, and what outputs to deliver in each step
    - The proposed method produces multiple SoAS architectures within a single MBSE environment composed of varying LoAs.

- **Next Steps:**
  - Developing executable UAF architectures.
  - Performing a trade study analysis to quantify the impact and aid in decision-making on the suitable LoAs to be integrated.

THE PROPOSED TRADE STUDY METHOD

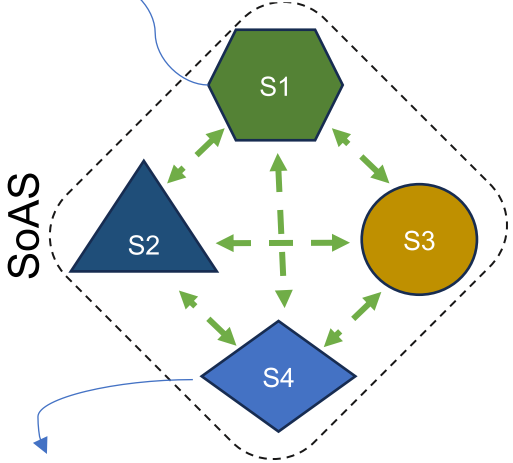
Objective: Choosing the suitable SoAS configuration while considering LoA and the resulting emergent behaviors.



The proposed method integrates MBSE architecture with Bayesian Networks and further improves the analysis by using Machine Learning and optimization algorithms

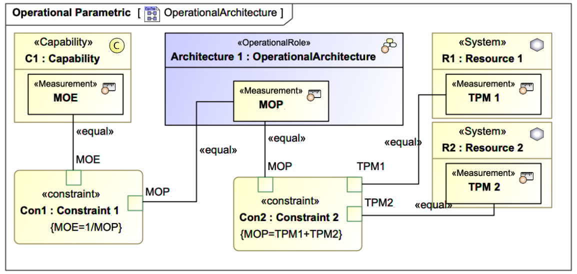
THE PROPOSED TRADE STUDY METHOD

LoA	Example	TPM
1	Legacy System (e.g., Ambulance)	Fuel/Mile

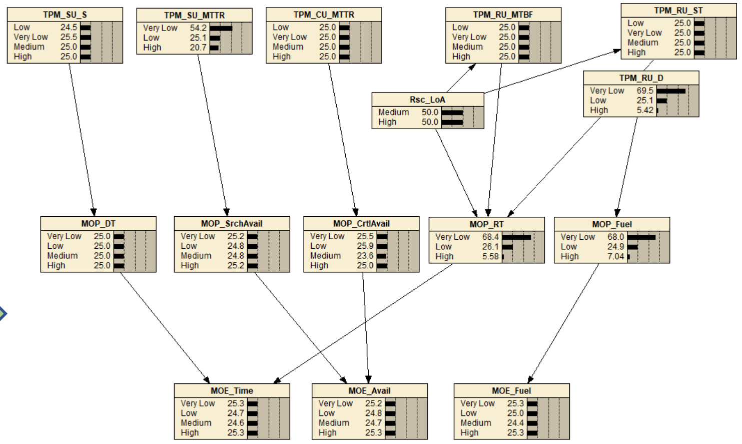


MOP	SoAS fuel consumption
MOE	SoAS fuel efficiency rate

SoAS executable model



Bayesian network



LoA	Example	TPM	Autonomous Capabilities
1	Helicopter (Legacy System)	Fuel/Mile	N/A
2	Autonomous Drone Type 1	AI/ML Performance accuracy	Navigation
3	Autonomous Drone Type 2	AI/ML Performance accuracy	Navigation + Image/video recognition

Predictive Analysis
Impact of TPMs on undesirable emergent behaviors in MOEs

Prescriptive Analysis
Root causes (TPMs) of an undesirable emergent behavior in MOEs
Preventive strategies

Due to LoA, SoAS trade-off analysis is a decision-making problem under uncertainty while the analyst must also take into account possible emergent behaviors

THANK YOU!

Mohammadreza Torkjazi
Ph.D. Candidate – Systems Engineering and Operations Research
mtorkjaz@gmu.edu

Dr. Ali K. Raz
Assistant Professor – Systems Engineering and Operations Research
Associate Director – Intelligent Systems and Integration – C5I Center
araz@gmu.edu

GEORGE MASON UNIVERSITY

34th Annual INCOSE
international symposium

hybrid event

Dublin, Ireland
July 2 - 6, 2024

www.incose.org/symp2024
#INCOSEIS