# System of Systems Capability to Requirements Engineering

**Jo Ann Lane**
Department of Astronautical Engineering
University of Southern California
Los Angeles, CA, USA
jolane at usc.edu

**Abstract -** *Given an existing set of interconnected, independent systems, often referred to as a system of systems (SoS), one of the key activities according to the United States Department of Defense* Systems Engineering Guide for Systems of Systems *is "translating SoS capability objectives into high-level SoS requirements". Capability engineering starts with understanding the desired capability and identifying various options for achieving that capability, technically assessing the various alternatives, then further evaluating the most viable alternatives in terms of capability performance, cost, and schedule. This paper provides additional guidance for translating capability objectives into requirements; defines SoS engineering methods, processes, and tools that might support this activity; and illustrates how the methods, processes, and tools would be used and integrated to support SoS engineering using an example SoS.*

**Keywords:** System of systems engineering, capability engineering, requirements engineering.

## 1 Introduction

Given an existing set of interconnected, independent systems, often referred to as a system of systems (SoS), one of the key activities according to the United States Department of Defense (DoD) *Systems Engineering Guide for Systems of Systems* [1] is *Translating SoS Capability Objectives into High-Level SoS Requirements*. According to the DoD SoS guidebook:

*"When a formal SoS is first identified, the systems engineering team is called upon to understand and articulate the technical-level expectations for the SoS. SoS objectives are typically couched in terms of needed capabilities, and the systems engineer is responsible for working with the SoS manager and users to translate these into high-level requirements that can provide the foundation for the technical planning to evolve the capability over time. To accomplish this, the SoS SE team needs to understand the nature and the dynamics of the SoS both to appreciate the context for SoS expectations and to anticipate areas of the SoS that are most likely to vary in implementation and change over time. The SoS systems engineer has a continuous active role in this ongoing process of translating capability needs into technical requirements and identifying new needs as the situation changes and the SoS evolves."* [1]

As illustrated in Figure 1, capability engineering starts with understanding the desired capability and identifying various options for achieving that capability. Initial capability engineering is typically done by assessing available resources and assets to identify existing functions from which the new capability can be composed [2], followed by a gap analysis for each alternative identified. Finally, each alternative is further evaluated in terms of capability performance, cost, and schedule resulting in information that can be used to support the trade decision.

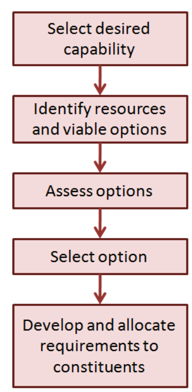


Figure 1. Overview of "Translating Capabilities into Requirements"

The rest of this paper provides additional guidance for translating capability objectives into requirements (C2R); defines SoS engineering (SoSE) methods, processes, and tools (MPTs) that might support this activity; and illustrates how the SoSE MPTs would be used and integrated to support SoS engineering. While many of the techniques and methods described here are not new, they are used in ways tailored to support SoS and SoSE analyses and integrated together through a process to support C2R engineering in a more rigorous, repeatable manner, resulting in meaningful information about alternatives that can be used to support a final decision on how the

This material is based upon work supported by the U.S. Department of Defense through the Systems Engineering Research Center (SERC) under Contract H98230-08-D-0171 and documented in technical reports SERC-TR-17 and USC-CSSE-2012-505. SERC is a federally funded University Affiliated Research Center managed by Stevens Institute of Technology.

capability will be implemented. The MPTs described here are illustrated using a notional example, the Regional Area Crisis Response SoS (RACRS), that is further described in [3]. This example has been developed to support research using actual systems in the public domain that are employed to respond to regional crisis situations.

# 2 C2R methods, processes, and tools

The C2R process shown above is elaborated in Figure 2 to identify techniques and methods that can be used to support the engineering activities associated with each step. The following sections describe each of the tools and how they are used in the C2R engineering process.

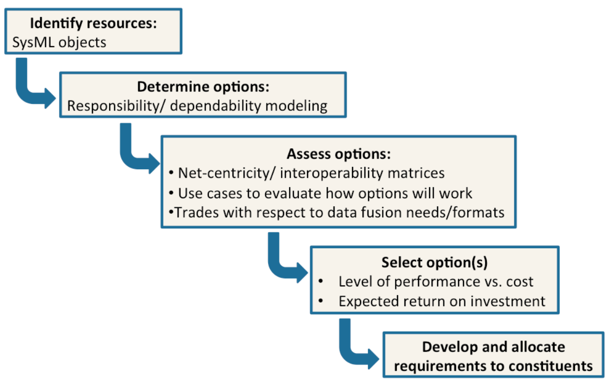


Figure 2. Methodologies and tools to support C2R engineering activities

## 2.1 System Modeling Language object models

Several system object models are used to understand the SoS and its constituent systems as well as to identify and understand single system functions that can be used to develop new capabilities and to assess and define the various options for implementing a new capability. Note that System Modeling Language (SysML) is used in the examples presented in this paper, but the other modeling approaches such as Unified Modeling Language (UML), Department of Defense Architecture Framework (DoDAF), and Ministry of Defence Architecture Framework (MODAF) have similar constructs that can be used.

These models begin with "black box" models to understand the SoS and its constituents at a high level and evolve to "white box" models that capture the internal information about the constituent systems needed to understand options for various SoS capabilities. The SoS SysML models described in [3] that support translating capabilities into requirements include:

- *Context diagrams:* Identifies systems of interest in the SoS from selected system viewpoints as well as who and what will interact and what types of information will be shared.
- *Use case diagrams:* Describe how the SoS capabilities of interest work. These diagrams can be used to model the "as is" SoS capabilities as well as alternative "to be" options for the new capability.
- *Sequence diagrams:* Illustrate the sequence of requests and responses that flow within the SoS environment for capabilities of interest.
- *Constituent system entities:* Describes the key functions and single system capabilities that can be performed by the single system.
- *Interface entities:* Describes the each interface in the SoS and the information that goes over the interface.
- *Input/Output (I/O) entities:* Describes the details of each data element type that goes over each interface.

## 2.2 Responsibility/dependability modeling

Responsibility modeling [4] captures information about constituents that can be used to identify socio-technical threats to the dependability of constituents in a coalition of systems or SoS. For each responsibility/capability of interest and resource/constituent system within the SoS, available resource agents that can support the capability are identified. In the second part of this modeling, the dependability of the each agent is assessed through a risk analysis process.

## 2.3 Interoperability matrices

The level of interoperability between the various constituents in an SoS are captured in an $N^2$ diagram where all of the constituent systems are listed both across the top and down the left side of the matrix. Each of the other boxes in the matrix indicates the level of interoperability between each of the two systems associated with that row/column. The method used to specify the level of interoperability in the $N^2$ diagram is based on the Levels of Information System Interoperability (LISI) model [5].

## 2.4 Data fusion analyses

For capabilities requiring data fusion, [6] provides guidance for trades must be performed with respect to level of fusion, functional aggregation, producer-consumer reconciliation, incongruent or inconsistent metadata management, concept of operations with respect to the fusion(s), fusion lifecycle, network topology, and information assurance.

## 2.5 Level of performance, cost and ROI

Once viable alternatives have been identified and evaluated with respect to feasibility and risks, the final steps are to a) further evaluate the improvement that is expected from the new or improved capability, typically in terms of SoS key performance parameters (KPPs), b) the estimated cost to provide the new or improved capability, and c) the expected return on investment (ROI) for the new or improved capability:

*KPP Improvement:* KPPs are often used to quantify or describe the current capability as well as the expected improvement. The KPPs used in this assessment are those that characterize the capability of interest and can include attributes such as required manpower, response time, speed, reliability, accuracy, safety, security, maintainability, and flexibility.

*Estimated cost:* To evaluate the costs of implementing the various alternatives, parametric cost models can be used. For example, the Constructive Systems Engineering Cost Model (COSYSMO) for SoS [7], can be used to evaluate the relative systems engineering costs of each alternative. To develop a complete cost estimate, additional cost models in the University of Southern California (USC) Center for Systems and Software Engineering (CSSE) Constructive Cost Model (COCOMO) family [8] can be used to estimate the costs of Commercial-Off-the-Shelf (COTS) integration and software development. The final cost estimate must also include the costs of hardware procurement and manufacturing that might be required for the alternative of interest.

*Expected ROI:* The ROI assessment uses both the KPP improvement assessments and expected costs to determine the actual expected value of the new or improved capability. The expected value might be business value, better utilization of scarce resources, improved mission outcomes, or lives saved, depending on the capability.

# 3 RACRS analysis using C2R MPTs

The motivation for RACRS, as described in [3], is based upon recent catastrophes that have happened in recent years within the United States: hurricane Katrina, devastating fires in California, powerful earthquakes in the western United States, and tornadoes in the Midwest United States. Early responders to these incidents found that communications between the different local agencies was difficult at best and often not integrated. When state and federal agencies became involved, the communications problems escalated. As a result, efforts have been initiated to establish a way to better integrate the needed agencies in response to a given incident. The goal is that the agencies will generally operate on their own outside of the SoS, then quickly be able to dynamically reconfigure and join the regional SoS as needed, typically in response to an incident.

For the purposes of this example, the current desire is to enhance the RACRS to provide the following improved capabilities:

- Improve number of fire-fighting resources available to fight major fires in the region. (Currently RACRS is limited to local civil responders augmented with available civil responders from other areas as well as low-risk inmates from local prisons/jails.)
- Further reduce the time and number of official crisis management personnel resources required to evacuate a specified area/region. This capability includes the ability to quickly determine areas that require evacuation and appropriate evacuation routes as well as the ability to evacuate large numbers of people that do not have transportation (e.g., assisted living residences, hospitals, jails).
- Protect evacuated areas from looters.

At the same time, the RACRS stakeholders want to:

- Minimize local government expense (city, county)
- Minimize risk to human life (crisis responders and local population)
- Minimize workload on skilled personnel responsible for responding to crisis.

The following identifies potential resources that might be used to provide improved capabilities:

1. Local: fire fighters, police, and sheriff personnel
2. Volunteer civilians
3. Military personnel at local bases
4. Low-risk inmates incarcerated in local jails
5. Unmanned aerial vehicles (which require people to remotely operate)
6. Unmanned ground vehicles (which require people to remotely operate)
7. TV/radio station announcers
8. Satellite and local road camera images showing crisis area (e.g., fire) and traffic status
9. Buses for transporting people
10. New system: Reverse-911 system that calls homes/residents of given area and tells them when, how, and where to evacuate to via pre-recorded messages.

The following illustrates how the above MPTs might be employed to develop a set of requirements to fulfill the desired capability improvements.

## 3.1 Identify resources

The constituents of interest for this version of the SoS are those described in [3]:

- *Satellite imaging system*: Provides images of interest to requestor
- *Fire department*: Manages the fire response units
- *Police department/sheriff's department*: Provides safety and crime-fighting support that includes evacuation support and protection from looters

- *Handheld devices*: Provides connectivity to crisis responders on the ground via voice and video
- *Reverse-911:* Automatically sends voice messages to people that reside or work in areas that need to be immediately evacuated.
- *Regional area planning and land use data*: Includes building plans, building locations, and maps for utilities (electricity, water, sewer) for regional areas of interest
- *Unmanned aerial vehicles (UAVs)*: Used for surveillance, lightweight fire retardant drops, and can also be armed to start needed backfires
- *Unmanned ground vehicle (UGV)*: Provides a) on-the-ground video feeds in situations where it is too dangerous for personnel and b) clearing of brush/small trees to create fire breaks
- *Aerial water tanker*: Canadian asset shared among multiple U.S. regional areas to drop water on hot spots
- *News helicopter*: Used to capture video feeds for news programs—includes news events as well as traffic flows and may also be used to monitor for signs of looting
- *Command and control center (CCC)*: Central site to monitor and help coordinate activities support decision makers

The constituent systems that interoperate with the CCC for the RACRS fire-fighting scenario are illustrated in the CCC context diagram shown in Figure 3. This is the "black box" view of the CCC and is used to understand at the top level the constituent systems related to fire-fighting from the CCC point of view.

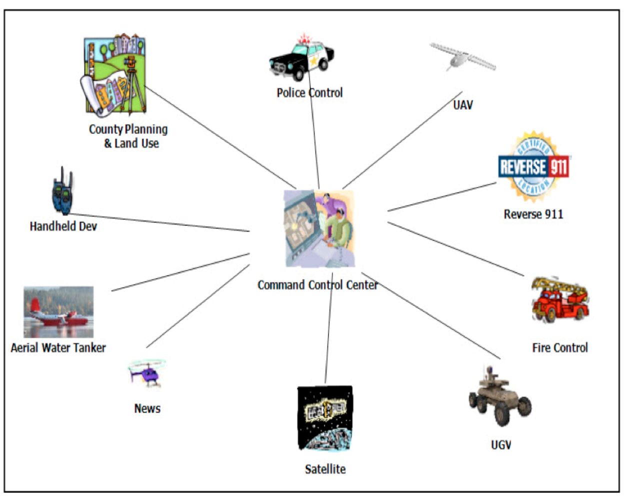


Figure 3. CCC context diagram [3]

## 3.2 Identification of capability alternatives

The initial analysis evaluates the feasibility of utilizing military resources in the region to support firefighting and then focuses on improving the manpower requirements needed to support the evacuation process. Potential alternatives to consider for the improved evacuation capability are:

1. Use current local patrols (police/sheriffs) (e.g., personnel employing loudspeakers, roving patrols, roadblocks)
2. Use civilian (volunteer) patrols (e.g., personnel employing loudspeakers, roving patrols, roadblocks)
3. Use unmanned vehicles (combination of ground and aerial that can warn of potential harm/record suspicious activities), satellites, and traffic monitoring cameras to identify and monitor evacuation routes
4. Install new reverse-911 system that can be used to automatically notify residents in a given area to evacuate.

## 3.3 Analysis of alternatives

The various MPTs are illustrated below and show how they can be used to support the analysis of capability alternatives.

### 3.3.1 Responsibility/dependability analysis

To start the analysis of alternatives, a responsibility matrix is developed that lists the various capabilities in the left-hand column and the potential resources across the top, as illustrated in Table 1. Next, the various resources are evaluated with respect to their dependability to support each responsibility. For those resources that may not be fully dependable, risks are defined and documented in a dependability risk table. The following describes the various risk attributes recommended in [4].

**Risk:** an identifier
**Target:** the specific resource
**Hazard:** a selection from a defined set of keywords – a candidate list in [4] is:
- Early – performs before required
- Late – performs after required
- Never/unavailable – never performs
- Incapable – attempts to perform, but not capable of completing
- Insufficient – performs, but at an insufficient level
- Impaired – performs incorrectly
- Changes – responsibilities permanently change

**Condition:** describes the condition that might occur as a result of the hazard
**Consequences:** impact of condition resulting from hazard
**Severity:** level of impact resulting from hazard

Table 2 identifies some example risks associated with the resources identified in Table 1. Note that this table is not comprehensive, but used to illustrate the MPT.

### 3.3.2 Interoperability Assessment

This MPT assesses the ability of the relatively dependable systems to interoperate with each other.

Table 1. RACRS Evaluate Area Responsibility Matrix

| Responsibility | Resources | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | Fire truck | Sheriff car | Water tanker | UAV | Reverse 911 system | Ambulance | Bus | Manual |
| Fight fire | Local, regional, military | Local | Canadian company | Military | | | | Local, regional, military, volunteer, low-risk inmates |
| Evacuate homes and businesses | Local, volunteer | | | | Local CCC personnel | | | |
| Evacuate assisted living homes | Local | Local | | | Local CCC personnel | | Local transit, charter | Assisted living staff, volunteers |
| Evacuate hospitals | | | | | | Public, private | | Hospital staff, volunteers |
| Prevent looters | | Local | | Military | | | | |

Table 2. Dependability Risk Matrix

| Risk | Target | Hazard | Condition | Consequence | Severity |
|---|---|---|---|---|---|
| 1 | Regional fire trucks/fighters | *Late* or *never* due to fires in own region | Reduced fire-fighting capability | More extensive fire damage | Medium to high, depending on other available resources |
| 2 | Canadian water tanker | *Late* or *never* due to other commitments | Reduced fire-fighting capability | More extensive fire damage, longer to put fires out | Medium to high, depending on other available resources |
| 3 | Local fire trucks | *Unavailable* due to reallocation to other fire | Reduced fire-fighting capability | More extensive fire damage, longer to put fires out | Medium to high, depending on other available resources |
| 4 | Reverse 911 system | *Insufficient* | Not all residents notified to evacuate | Residents at risk for being trapped/affected by crisis e.g., fire, hazardous material | Low to high, depending on type of crisis requiring evacuation |
| 5 | Low-risk inmates | *Various*—may be unskilled, may escape custody | Fire-fighting capability is less than that of experts | Additional resources required to train and monitor inmates | Low severity with respect to crisis, but medium severity with respect to costs related to training and monitoring |

The first step to understanding and managing interoperability within an SoS is understanding the information that flows across each interface and the format of the data elements that are part of that information. SysML interface and input/output (I/O) entities can be used to capture this information for key RACRS interfaces.

The next step is assessing the interoperability of the various constituent systems. For this assessment, an $N^2$ matrix is developed where the systems are listed both down the left column and across the top. Then each pair of systems is evaluated for interoperability using the LISI model [4]. The level of interoperability can be specified for each of four LISI PAID attributes: Procedures, Applications, Infrastructure, and Data. The levels of interoperability that can be specified for each attribute using in the LISI model are isolated, connected, functional, domain, and enterprise [5]. Table 3 shows a Data interoperability matrix for the RACRS firefighting systems. Note that the cells on the diagonal are shaded. This reflects the fact that every system should be fully interoperable with itself (if not, then these cells should contain an assessment value). Also note that in many cases, system interoperability is bi-directional and in these cases, one only need assess the systems interoperability above or below the diagonal (but not both). If interoperability is not bi-directional, then the full matrix should be completed.

### 3.3.3 Use cases

For those systems that are evaluated as reasonably dependable and interoperable, use cases are developed to show how the systems will interact to perform the various desired missions. A top-level use case diagram for the key RACRS mission/support missions might illustrate how the different resources might interoperate with the CCC to support evacuations, fire suppression, and retrieval of topo map information. The use case might be further refined and analyzed by developing sequence diagrams that further illustrate interactions and data sharing between the various constituent systems for a given mission type.

Table 3. Firefighting data interoperability matrix

| Fire-fighting constituents | Local | Regional | Military | Canadian | Volunteer | Low-risk inmates |
|---|---|---|---|---|---|---|
| Local | | | | | | |
| Regional | Functional | | | | | |
| Military | Isolated | | | | | |
| Canadian | Connected | Connected | Isolated | | | |
| Volunteer | Connected via handheld devices | Isolated | Isolated | Isolated | | |
| Low-risk inmates | Connected via handheld devices | Isolated | Isolated | Isolated | Connected via handheld devices | |

#### 3.3.4 Level of performance, cost, and ROI assessment

The final selection of a capability approach is determined by assessing the level of performance for viable alternatives, the level of risk associated with each alternative, the costs associated with the alternative (which would typically include procurement, development, integration, installation, and training), and the expected return on the investment associated with the new or improved capability. In the RACRS example, the Evacuate Area capability analysis determined that the procurement of a Reverse-911 system would greatly facilitate evacuations in the region, was relatively easy to install and use in the CCC, and the return on investment in terms of lives potentially saved during a given crisis was worth the estimated costs. In addition, this system can be utilized together with information from satellites and road cameras to inform people about the best routes for evacuation.

### 3.4 Identifying and Implementing SoS Requirements for Selected Alternative

Once the decision is made to procure a regional Reverse 911 system, an appropriate set of requirements is developed for each of the constituent systems that will interact with the Reverse 911 system. The systems that will be affected by the addition of the Reverse 911 system are the CCC who will operate the Reverse 911 system and monitor evacuations using the available video feeds and the firefighters and law enforcement systems that can initiate and terminate evacuations. Requirements will be allocated to and implemented by these constituent systems to ensure interoperability of the Reverse 911 system.

## 4 Conclusions

The research described in this section shows that existing MPTs can easily be re-purposed and used together to support SoS capability to requirements engineering, resulting in a fairly rigorous technical analysis of capability options and the costs required to implement each. The next steps are to continue to refine these MPTs through the analysis of more complex capabilities and to further investigate and refine the data fusion MPT to support more complex capabilities such as situational awareness that rely on data from multiple sources and sensors.

## References


[1] Department of Defense. 2008. Systems engineering guide for system of systems, version 1.0.

[2] United States Air Force (USAF) Scientific Advisory Board (SAB), *Report on system-of-systems engineering for Air Force capability development*; Public Release SAB-TR-05-04, 2005.

[3] J. Lane and T. Bohn (2013); Using SysML Modeling To Understand and Evolve Systems of Systems, *Systems Engineering Journal*, Vol. 16, No. 1.

[4] D. Greenwood and I. Sommerville. Responsibility modeling for identifying sociotechnical threats to the dependability of coalitions of systems. *Proc. of the 2011 6th International Conference on Systems of Systems Engineering*, 2011.

[5] Department of Defense, C4ISR Architecture Working Group Final Report - Levels of Information System Interoperability (LISI). Washington DC: OSD(ASD(C3I)) C4ISR AWG, 1998.

[6] M. Solano, SoSE architecture principles for net-centric multi-int fusion systems. *Proc. of the 2011 6th International Conference on Systems of Systems Engineering*, 2011.

[7] J. Lane, Cost model extensions to support systems engineering cost estimation for complex systems and systems of systems. *Proc. Conference on Systems Engineering Research (CSER 09)*, 2009.

[8] B. Boehm, R. Valerdi, J. Lane, and W. Brown, "COCOMO Suite Methodology and Evolution", CrossTalk Journal, Vol. 18, No. 4, pp. 20-25. 2005.