Scaling Model-Based System Engineering Practices for System of Systems Applications: Software Methods

October 2017

Janna Kamenetsky jannak@mitre.org
Laura Antul lantul@mitre.org
Dr. Aleksandra Markina-Khusid amk@mitre.org
Matt Cotter mjcotter@mitre.org
Dr. Judith Dahmann jdahmann@mitre.org

NDIA 20th Annual Systems Engineering Conference
http://www.ndia.org/events/2017/10/23/20th-systems-engineering-conference
Technical Approach: Inheritable Architectures

- **Kill Chain Architecture**
  - Abstract Killchain
    - Navy
    - Air Force
    - ...

- **Service Oriented Architecture (SOA)**
  - Abstract SOA
    - Avionics Service Bus
    - ...

Enables Model Re-use corresponding to different architecture patterns
Base Model Architecture

- **Base/Derivative Model Framework**
  - Base Model captures key functional SoS architecture
  - Derivative model represents domain-specific behavior

- **This approach helps:**
  - Accelerate domain model development via Base Model reuse
  - Rapidly evaluate different options utilizing predefined stereotypes and analysis engines
  - Iterative design to continuously refine common SoS functions
Base Model: High Level Structure
Base Model: Inheritance Structure

Inheritable and reusable Statecharts

Statechart for Device
BASE Model: Inheritable Types

**BASE**
- Operations (i.e. functions)
  - `processSignals()`
- Attributes (i.e. metrics)
  - `MaxRange`

**DERIVATIVE**
(e.g. CDMaST)

- Towed Array Sonar Sensor
- Bow Sonar Sensor
- ISAR

Statechart
Base Model CSV Importer

Base Model

10 Node Scenario

100 Node Scenario

MBE Utility to reduce development effort associated with modeling large SoS complex networks
CSV Importer Utility

Conceptualize SoS Architecture

Run CSV Importer Utility to automatically generate model/JMS Pub/Sub Architecture

Add Connectivity Framework
Base Model GUI

- A MATLAB GUI has been built to simplify the process of populating a connectivity matrix
- The tool outputs a CSV file that can then be imported into the architecture model
Q2 Metrics – Experiments

▪ Qualitative
  – **Experiment 1**: Give the base model to MITRE employees to use on their projects as they see fit. Collect feedback.
    ▪ Likes, dislikes, pain points, time savings estimates, description of use case, experience level
    ▪ Time Cost: 30 min interview

▪ Quantitative
  – **Experiment 2**: Give MITRE employees a sample coms network and have them create it by hand and by using the CSV importer
    ▪ Networks of different sizes
    ▪ Measure time to complete exercise
    ▪ Time Cost: Approx. 45 min per data point

  – **Experiment 3**: Randomized control trial with ~20 new interns
    ▪ Group A: Create reference model from scratch
    ▪ Group B: Create reference model using base model
Metrics – Experiment 1 Results

**Project 1:**
- 3 reviewers
- Not adopted

**Feedback:**
- “…This base model would be a great reference, e.g., utilizing the package structure framework used, with the inheritable architectures and the focus on reuse.”
- “…We expect to draw ideas from it as we build our own model.”
- “We intend to focus more on activity diagrams than state charts.”
- “Our project is not in the context of the Air Force, so we would have to change the block and activity names.”
- “Overall it is not a good fit for [our project].”

**Project 2:**
- 1 reviewer
- Adopted

**Feedback:**
- **Qualitative**
  Base Model state charts look too “in-depth”, “specific”, need to take a closer look to see if they will work for my use case. But if they work, “that would be awesome”, it will save tons of time.
- **Pseudo - Quantitative**
  Estimated time savings of 40 hours on work completed so far.
- **Update**
  Base Model has proven a good fit for project and has been used extensively.
Metrics – Experiment 2 Results

The Scenario
This is a hypothetical Air Force kill-chain scenario consisting of 1 ground control station (AOC), 1 air command and control (C2), 4 Fighter Jets, 4 Unmanned Aircraft Systems (UASs), and 1 Tanker.

- AOC needs to be able to communicate with C2, since C2 alerts AOC when there is a threat and then gets its orders from the ground.
- C2 also needs to be able to communicate with all fighters and the Tanker during the mission.
- Also, all fighters and UASs need to be able to communicate with the Tanker, since they’ll occasionally need to refuel during flight.
- Every fighter needs to be able to communicate with every other fighter, and
- every UAS needs to be able to communicate with every other UAS.
- Moreover, every fighter should be able to communicate with every UAS, and vice versa.

You may assume all communication channels are bi-directional (any communication matrix you set up should be symmetric with respect to rows and columns).

The Time to model 11-node scenario with and without CSV tool:

- **Mean Time savings:** 39%
- **Standard Dev:** 12%

© 2016 The MITRE Corporation. All rights reserved. For internal MITRE use
Metrics – Experiment 2 Results

Time to model 29-node scenario with and without CSV+GUI tool

- **Time savings**
  - Mean: 63%
  - Standard Dev: 14%

- **Average mistakes**
  - Without tool: 9.2
  - With tool: 0.8