RT-176: Verification and Validation (V&V) of System Behavior Specifications

Sponsor: DASD(SE)

By
Dr. Kristin Giammarco

10th Annual SERC Sponsor Research Review
November 8, 2018
FHI 360 CONFERENCE CENTER
1825 Connecticut Avenue NW, 8th Floor
Washington, DC 20009

www.sercuarc.org

Presentation Agenda

- Research Motivation & Objectives
- Technical Accomplishments
- Future Work

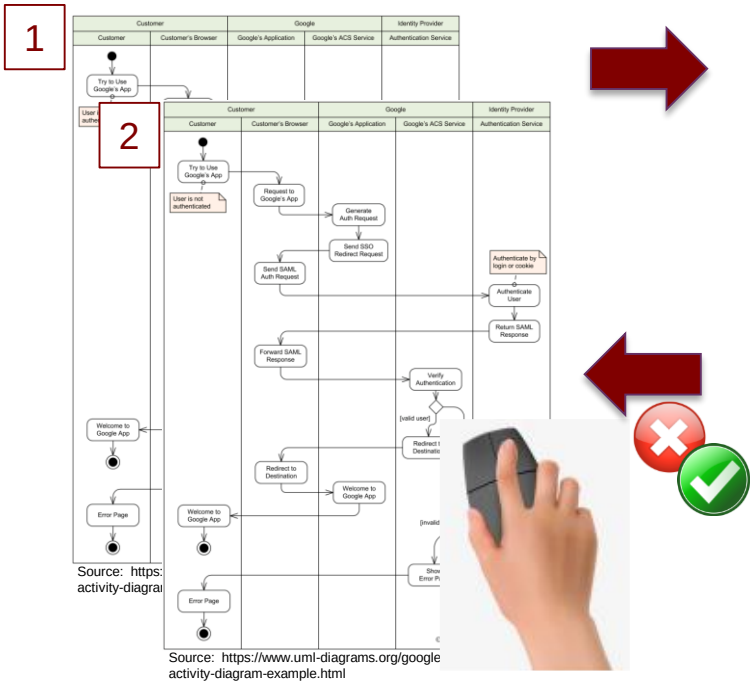
Motivation

- Model-based methods tools and approaches on their own do not guarantee success
- The model may adhere to notational specifications while the design itself may be incomplete, ambiguous, inefficient, or contain unwanted system behaviors
- This research developed methods and tools to steer and shape *behavioral design*
  - to meet requirements (verification)
  - to meet expectations (validation)

Motivation

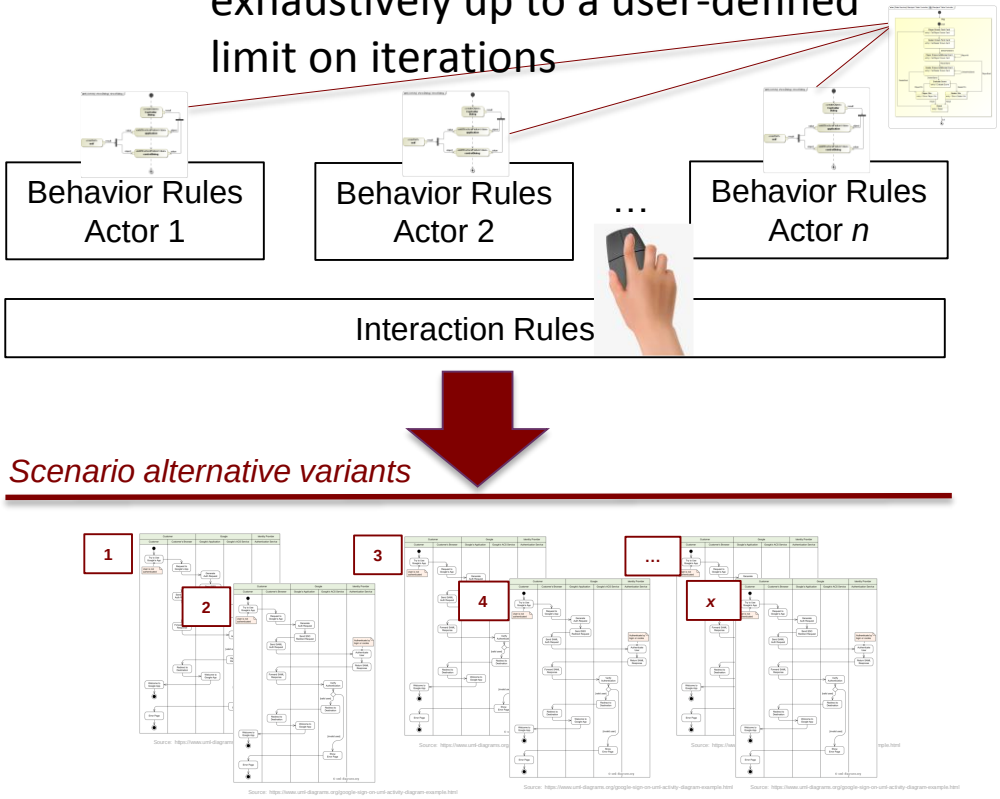
Prevailing Problem:

- Incompleteness
  - Only a subset of possible behaviors are included with actors and interactions drawn on the same diagram



MP Value Proposition:

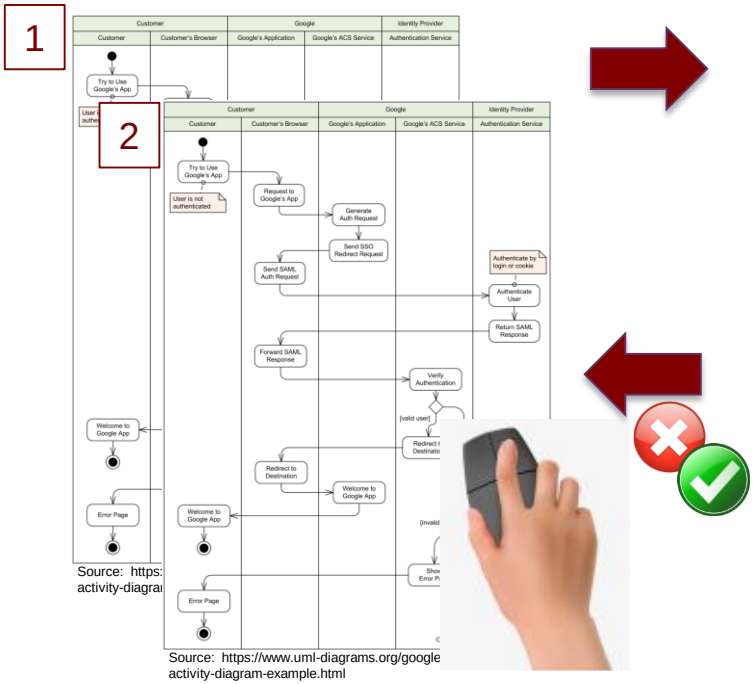
- Scope-completeness
  - Generates full set of possible event traces (use case extensions) exhaustively up to a user-defined limit on iterations



Motivation

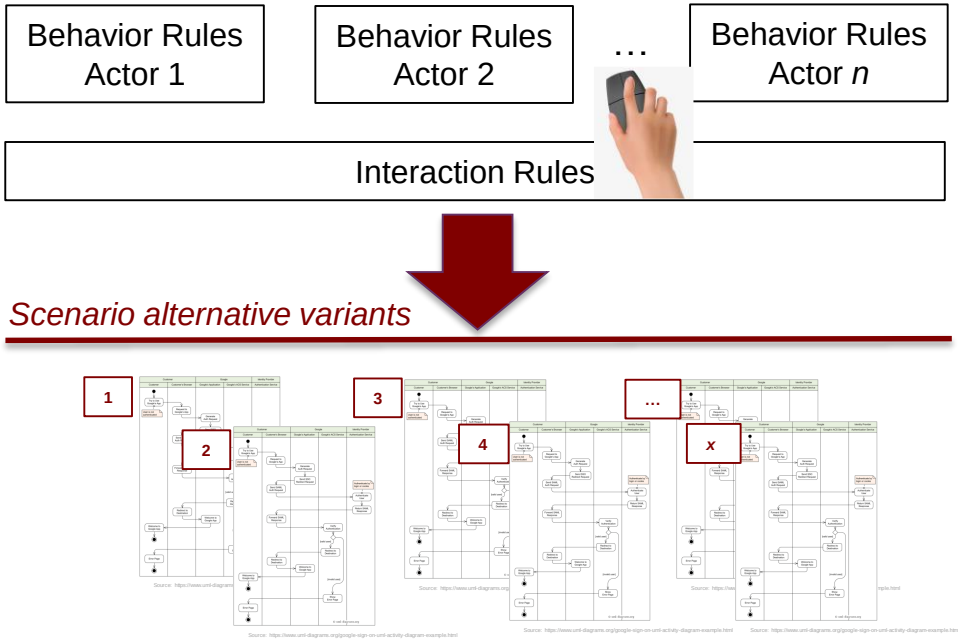
Prevailing Problem:

- Ambiguity
  - Behavior models that describe general activities but are unclear about who is doing each activity, or are otherwise unclear about activities performed



MP Value Proposition:

- Separation of concerns
  - Behaviors are separated by actor, and interactions between actors are separately layered on as constraints
  - Modeling in MP enables discussion and clarification of the behavior logic



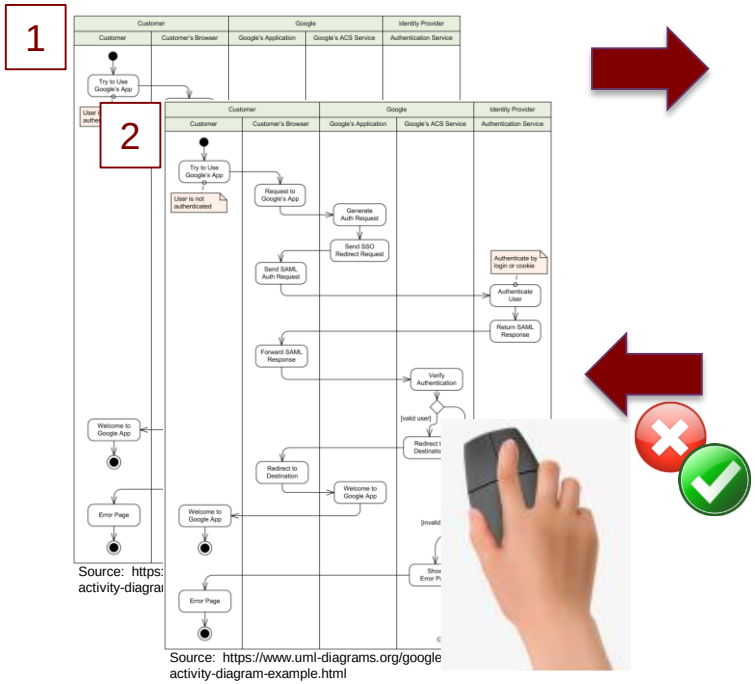
Motivation

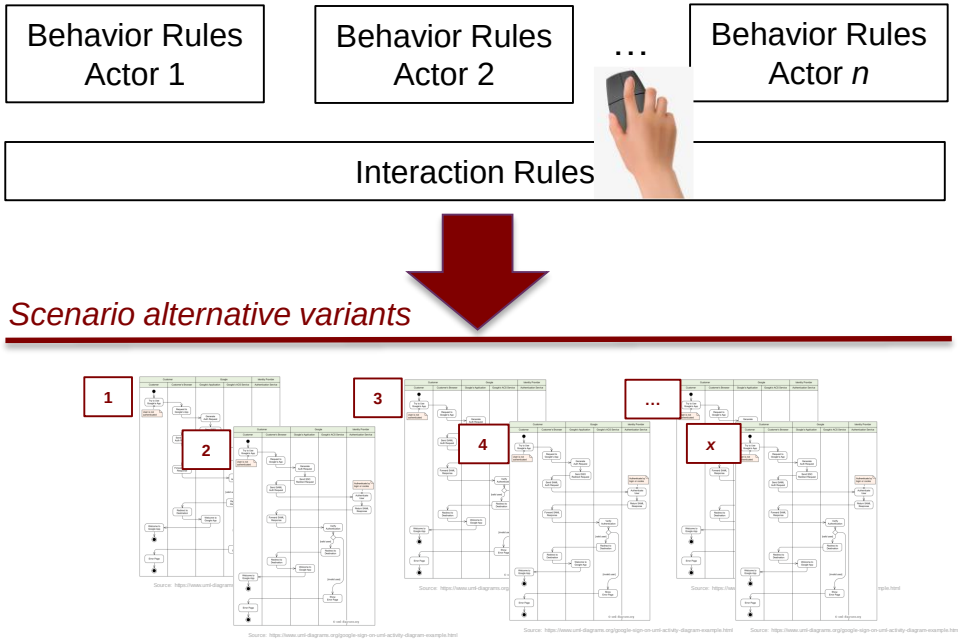
Prevailing Problem:

- Inefficiency
  - When people continue to do work that an automated computing device could do faster and with fewer errors

MP Value Proposition:

- Efficient task allocation
  - Humans focus on using their experience, creativity, and pattern detection skills to inspect and evaluate, and use automated tools to compute, generate, and search





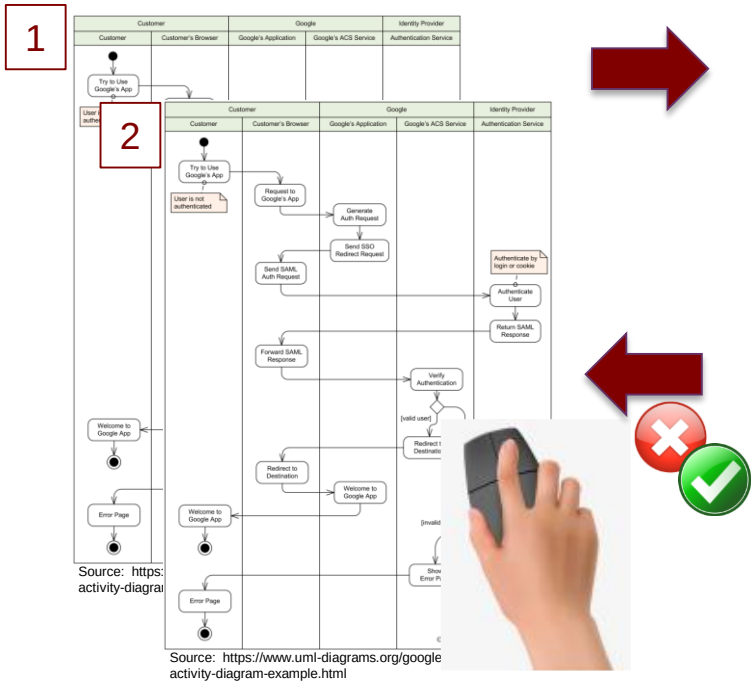
Motivation

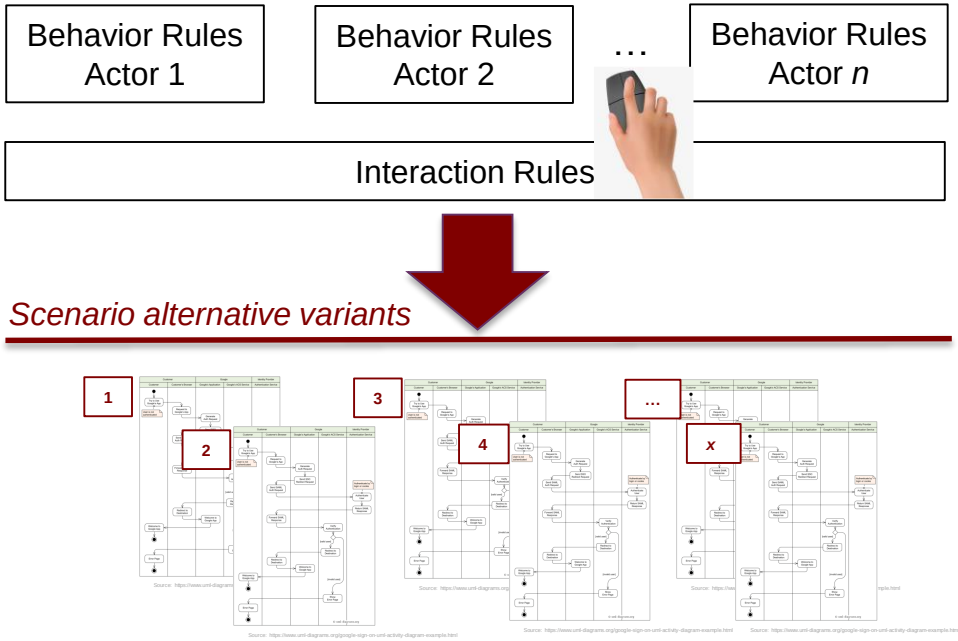
Prevailing Problem:

- Unwanted behaviors
  - Built systems that may meet requirements, but also permit extra undesired behaviors

MP Value Proposition:

- Behavior pruning
  - Enforces the necessary model structure for exposing and purging unwanted behaviors in the design before they emerge in the actual system





Research Objectives

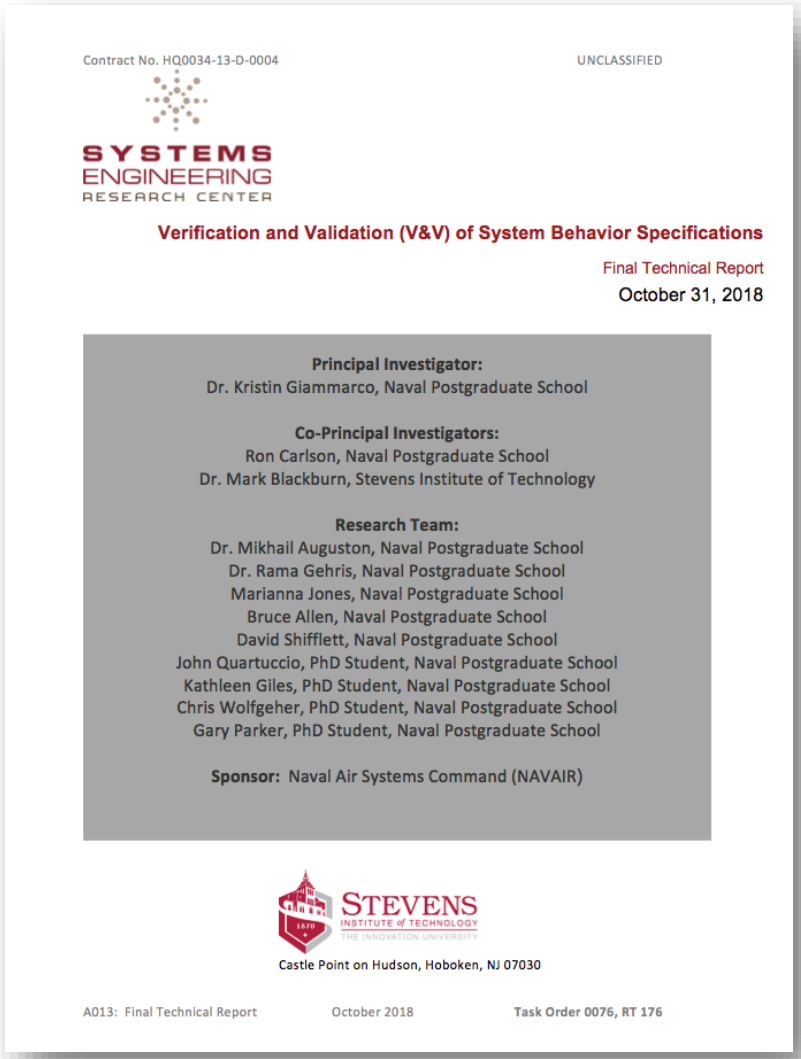
- *Demonstrate use of the UAV behavior models for early V&V analysis of requirements*
  - —using MP to expose positive and negative system behaviors permitted by the design

- *Formalize patterns of common design flaws or other model properties*
  - —Catalog of anti-patterns catalog

Presentation Agenda

- Research Motivation & Objectives
- Technical Accomplishments
- Future Work

Organization of the Technical Report


Contract No. HQ0034-13-D-0004 UNCLASSIFIED

SYSTEMS ENGINEERING RESEARCH CENTER

Verification and Validation (V&V) of System Behavior Specifications

Final Technical Report

October 31, 2018

Principal Investigator:
Dr. Kristin Giammarco, Naval Postgraduate School

Co-Principal Investigators:
Ron Carlson, Naval Postgraduate School
Dr. Mark Blackburn, Stevens Institute of Technology

Research Team:
Dr. Mikhail Auguston, Naval Postgraduate School
Dr. Rama Gehris, Naval Postgraduate School
Marianna Jones, Naval Postgraduate School
Bruce Allen, Naval Postgraduate School
David Shifflett, Naval Postgraduate School
John Quartuccio, PhD Student, Naval Postgraduate School
Kathleen Giles, PhD Student, Naval Postgraduate School
Chris Wolfgeher, PhD Student, Naval Postgraduate School
Gary Parker, PhD Student, Naval Postgraduate School

Sponsor: Naval Air Systems Command (NAVAIR)

STEVENS INSTITUTE of TECHNOLOGY THE INNOVATION UNIVERSITY

Castle Point on Hudson, Hoboken, NJ 07030

A013: Final Technical Report October 2018 Task Order 0076, RT 176

Appendix A: List of Publications and Invited Talks

Appendix B: References Cited

Appendix C: Collaborator Courses that Integrate or Contribute Research Results

Appendix D: Monterey Phoenix Overview

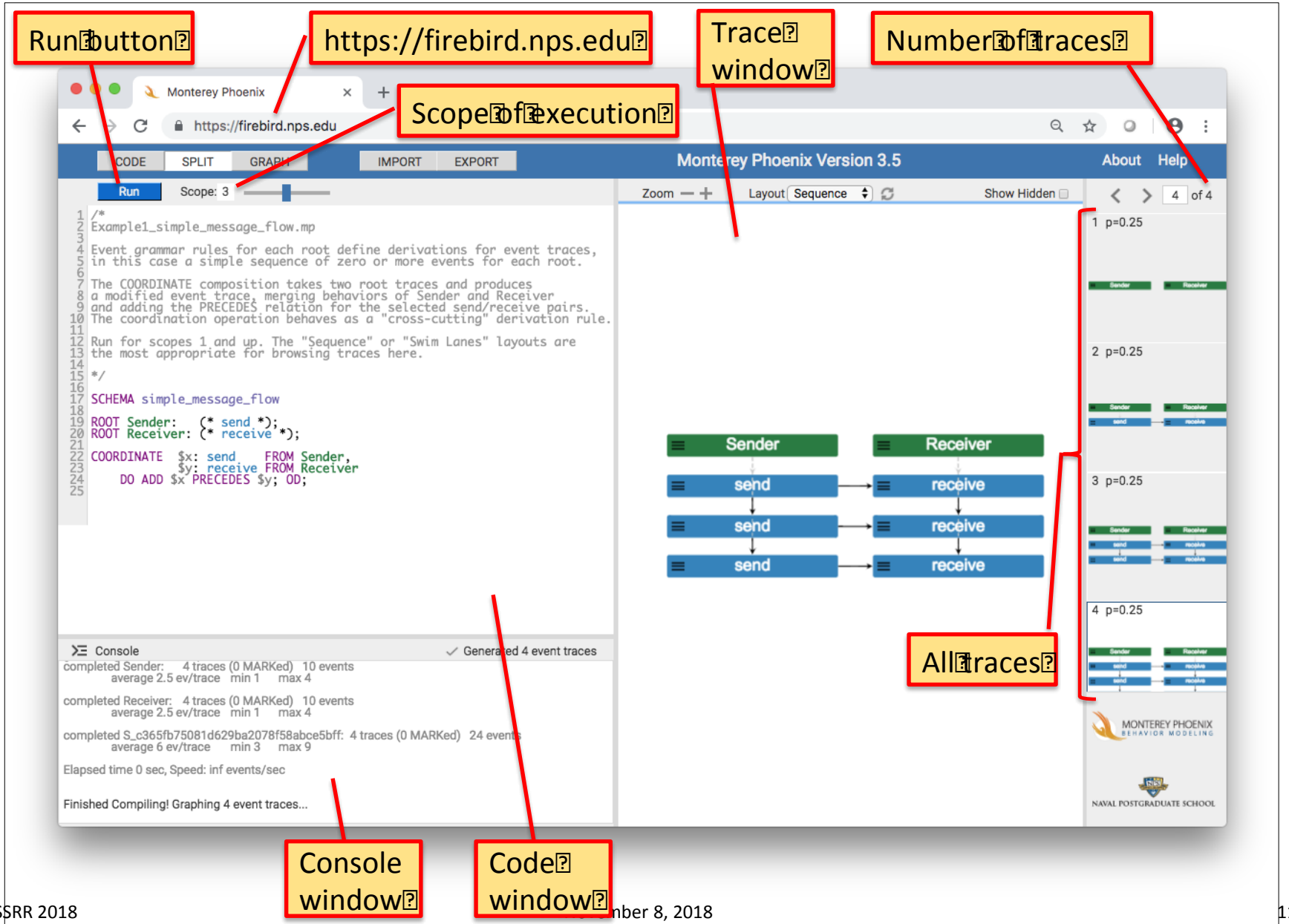
Appendix F: Instructions for Downloading MP Models

Appendix G: Model Based V&V (MCSE MPT) Demonstration

Appendix E: Catalog of Reusable Architecture Patterns

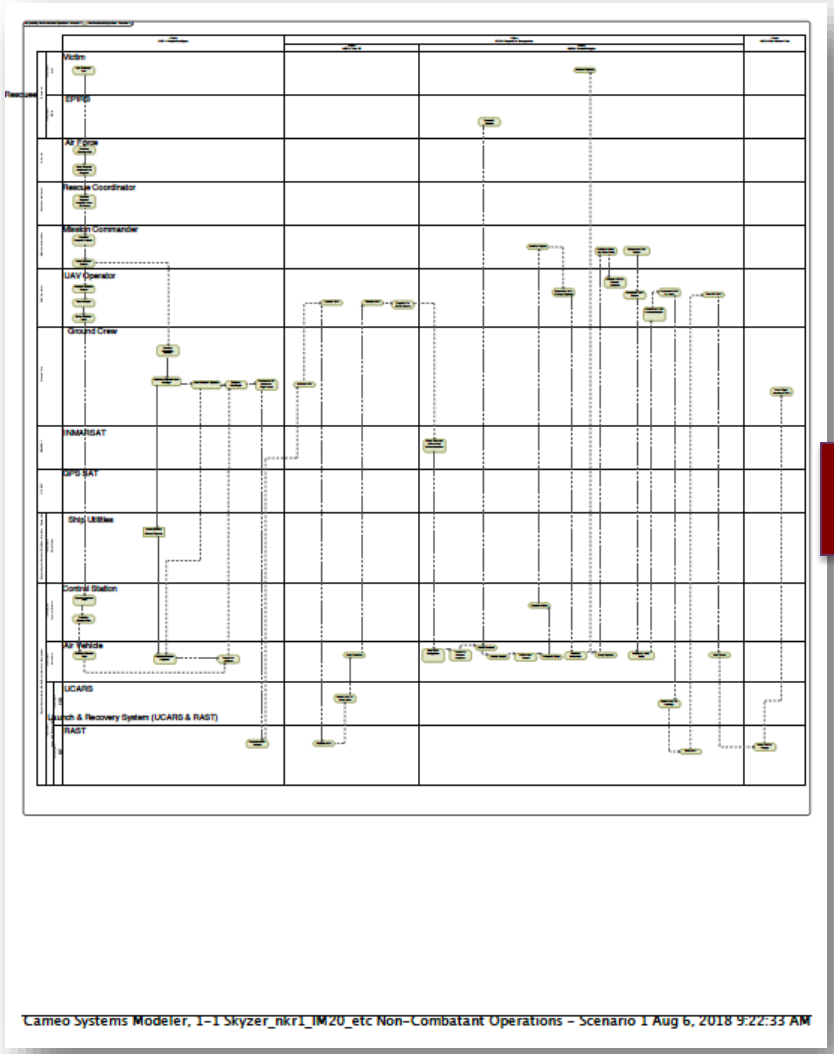
https://sercuarc.org/project/?id=35&project=Verification+and+Validation+%28V%26V%29+of+System+Behavior+Specifications

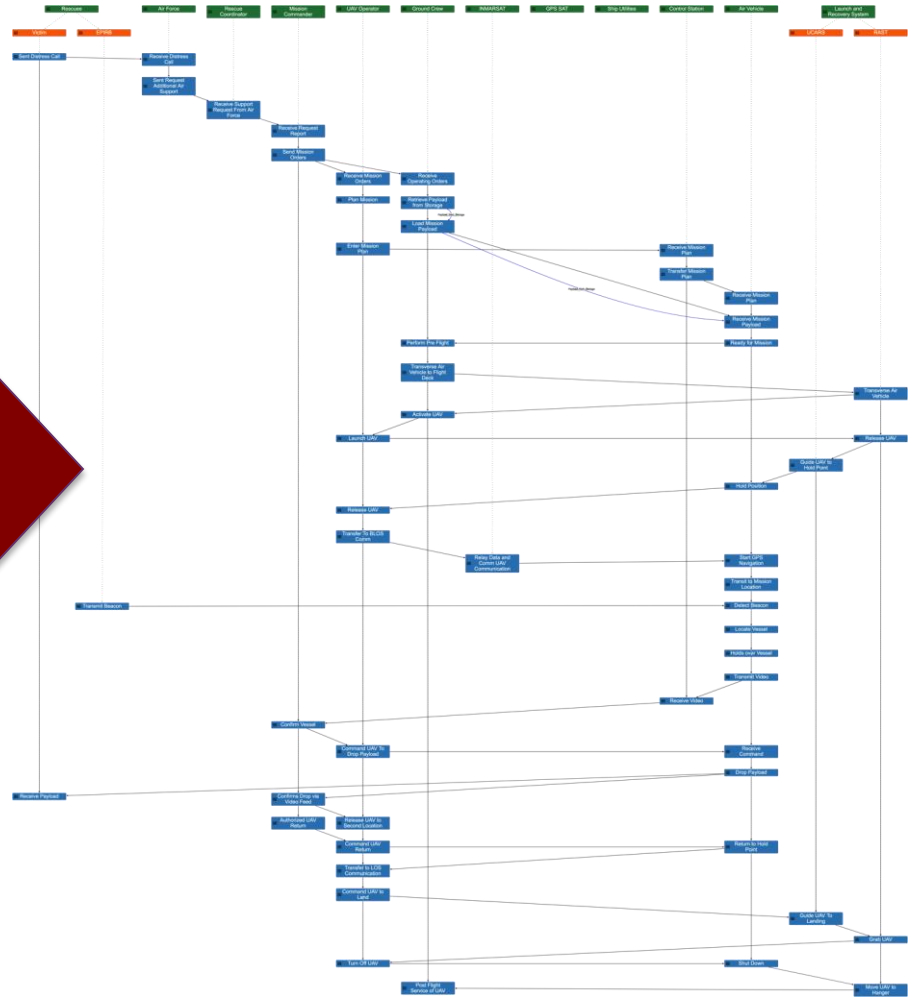
MP-Firebird Layout



Skyzer MP Modeling (Equivalency Demonstration)

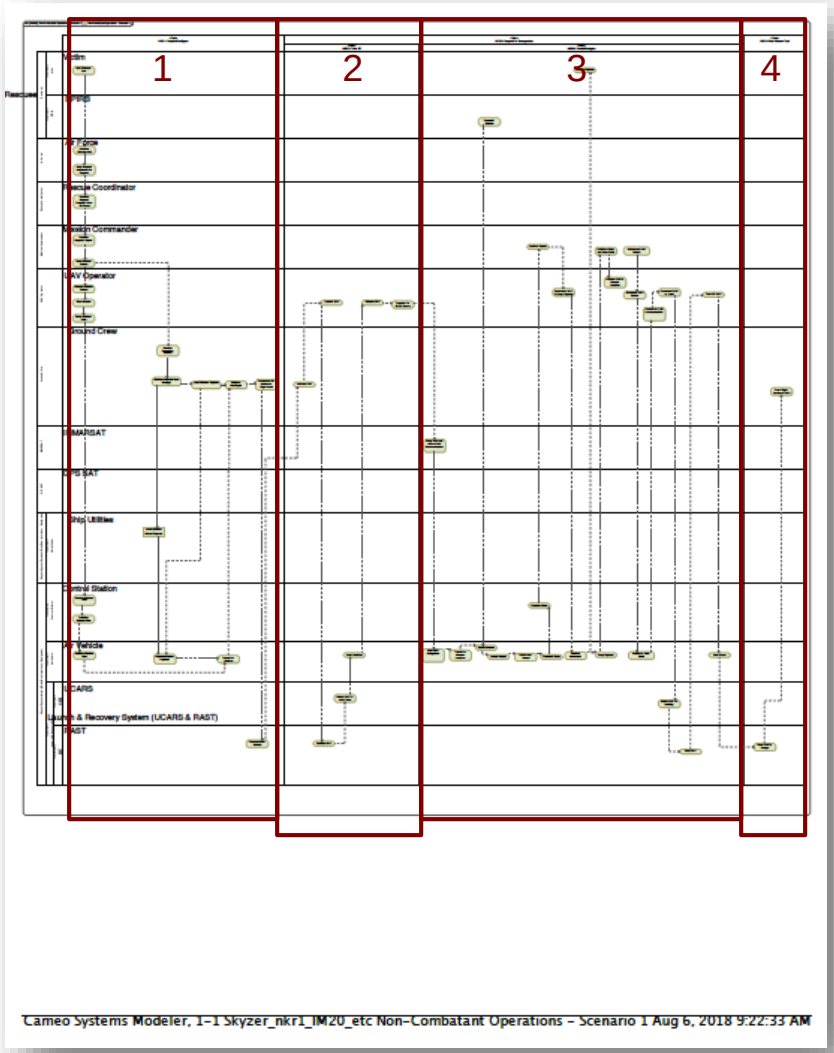
Non-Combat Operations Scenario 1

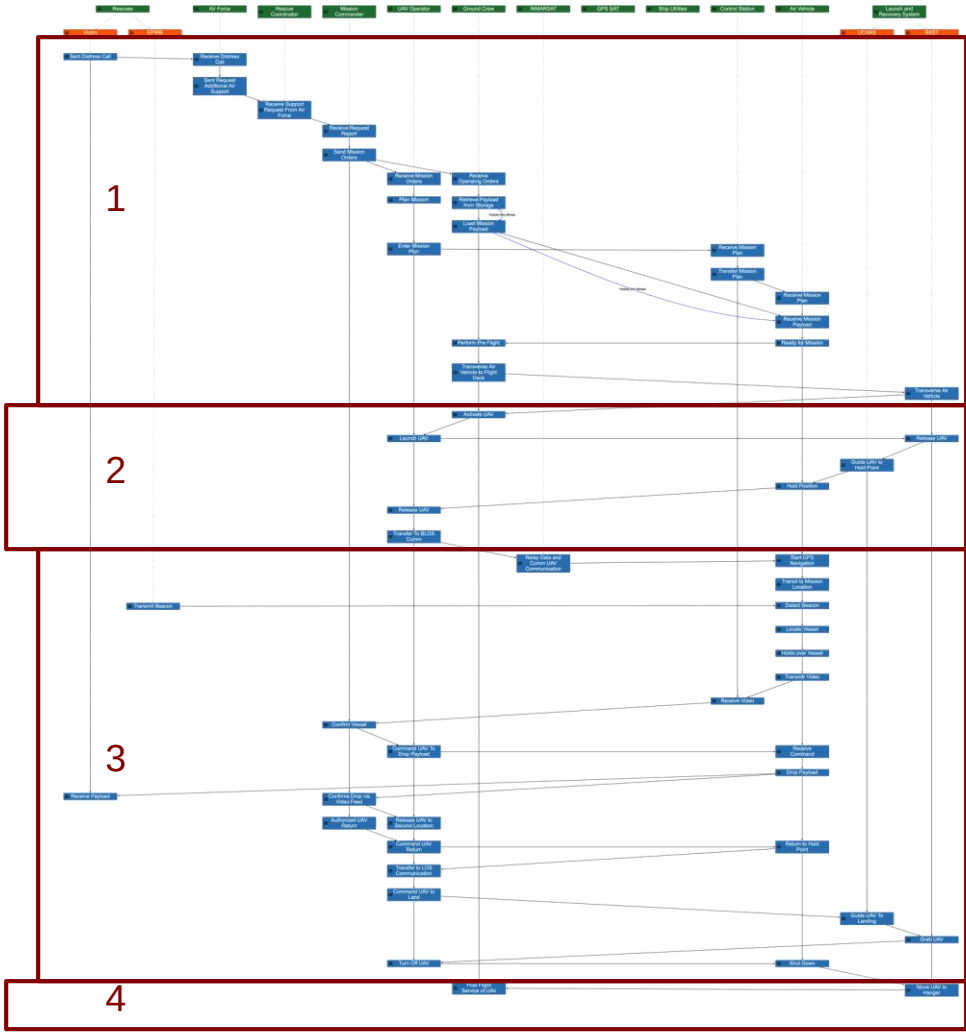




Skyzer MP Modeling (Model Segmentation)

Non-Combat Operations Scenario 1

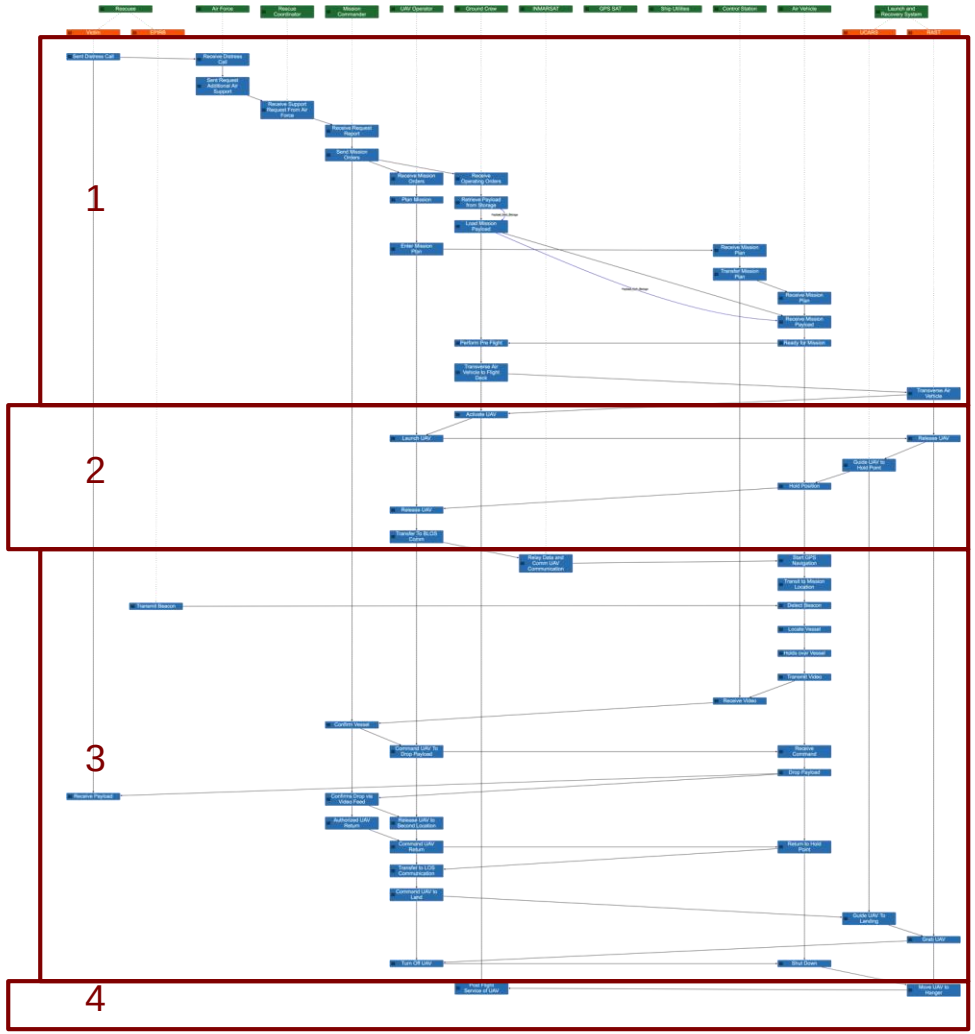




Skyzer MP Modeling (Model Elaboration)

1 scenario

Non-Combat Operations Scenario 1



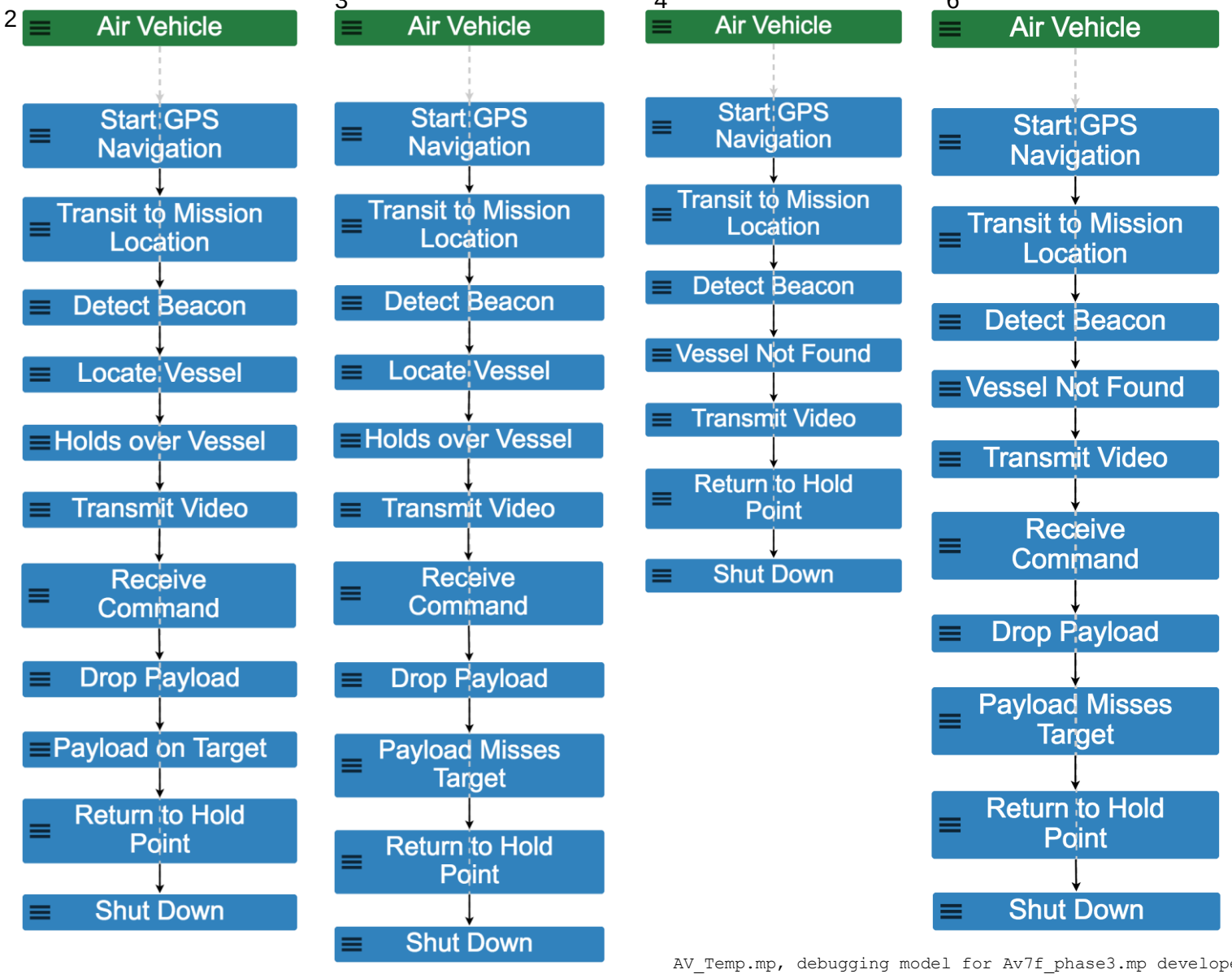
Phase 1 – Prepare/Configure → Phase 1 scenarios

Phase 2 – Take Off → Phase 2 scenarios

Phase 3 – Transit/Navigate → Phase 3 scenarios

Phase 4 – Post Mission Task → Phase 4 scenarios

Phase 3 Alternative Emergent Behaviors



Far left: Baseline scenario; vessel located and payload on target.

Middle left: Vessel located but payload missed target.

Middle right: AV needs to return before vessel is located.

Far right: Vessel not found but AV drops payload.

AV_Temp.mp, debugging model for Av7f_phase3.mp developed by D. Shifflett 8/21/2018

Requirements Discovery

- What should happen if the payload just misses the target (trace 3)?
  - —Could the payload still be retrieved by target vessel? What would help?
- What should happen if the AV has to return before locating/reaching the vessel (trace 4)?
  - —Could the payload be dropped at max range with a means for vessel retrieval?
- What should happen if the AV drops the payload prematurely, enroute to the vessel (trace 6)?
  - —Though unintended by the modeler, does trace 6 contain an idea for handling out of range vessels or AVs experiencing a return to base condition?

All of these operational "what ifs" were exposed through MP modeling of the provided baseline scenario.
MP modeling of SysML behavior diagrams can help to expose requirements that may otherwise not be considered until later in the lifecycle.

Architecture Model Anti-Patterns (Examples in Four Languages)

No.	DM2/UPDM	UPIA	SDL	LML
H.1.1	Activities with no child and no parent	Operational tasks with no child and no parent	Functions with no child and no parent	Actions with no child and no parent
H.2.4	Requirements with more than one parent	Requirements with more than one parent	Requirements with more than one parent	Requirements with more than one parent
H.5.1	Performers having itself as a child	Capability roles having itself as a child	Components having itself as a child	Assets having itself as a child
FPA.1.1	Activities that are not performed by any performer	Operational tasks that are not performed by any capability role	Functions that are not performed by any component	Actions that are not performed by any asset
FI.3.1	Activities that do not produce or consume any resources	Operational tasks that do not produce or consume any information elements	Functions that do not produce or consume any items	Actions that do not generate or receive any input/outputs
PI.6.1	Performers that exchange some resource, but are not connected to any common connectors	Capability Roles that exchange some information element, but are not connected to any common needlines	Components that exchange some item, but are not connected to any common links	Assets that exchange some Input/output, but are not connected by any common conduits
T.2.1	Activities that do not trace to any requirement	Operational Tasks that do not trace to any requirement	Functions that are not based on any requirement	Actions that do not satisfy/verify/trace to any requirement
S.5.1	Performers that interact with each other through exchange of resources, but are not subject to a common standard	Capability roles that interact with each other through exchange of information elements, but are not subject to a common standard	Components that interact with each other through exchange of items, but are not specified by a common standard-labeled requirement	Assets that interact with each other through exchange of input/outputs, but satisfy no common standardizing requirement

Technical report contains a total of 46 anti-patterns

Presentation Agenda

- Research Motivation & Objectives
- Technical Accomplishments
- Future Work

Future Work

- Further test the Monterey Phoenix approach on MBSE pilot projects
- Formalize the types and definitions of emergent behavior for use in risk analysis
- Train model developers how to verify and validate SysML models from other tools using MP
- Generate SysML sequence, activity, and state transition views from MP models
- Develop a graphical gateway to MP (enable code generation from diagrams)

Questions?

RT-176 Interim Report and Models:
https://sercuarc.org/project/?id=35&project=Verification+and+Validation+%28V%26V%29+of+System+Behavior+Specifications

Monterey Phoenix and Related Work:
https://wiki.nps.edu/display/mp
https://firebird.nps.edu

MONTEREY PHOENIX
BEHAVIOR MODELING

Kristin Giammarco: kmgiamma (at) nps.edu