Transforming System Engineering through Model-Based Systems Engineering (Model-Centric Engineering)

August 15, 2014

Principal Investigators
Dr. Mark Blackburn, Stevens Institute of Technology

Research Team
Dr. Rob Cloutier, Stevens Institute of Technology
Dr. Gary Witus, Wayne State University
Eirik Hole, Stevens Institute of Technology

Sponsor
NAVAIR, DASD (SE)

Task Order 0018, RT 118
Table of Contents

Acknowledgments .......................................................................................................................6

Executive Summary .....................................................................................................................7

1 Introduction ..............................................................................................................................9
  1.1 Objective ............................................................................................................................9
  1.2 Organization of Document ..............................................................................................10

2 In-Process Summary ..............................................................................................................12
  2.1 Model-Centric Engineering and Model Lexicon Status ......................................................12
  2.2 Clarifying the Vision Model Concept ............................................................................12
    2.2.1 What is a model? .........................................................................................................13
    2.2.2 Operational Perspective of Model-Centric Integration ............................................15
  2.3 Task 3 - Vision and “As Is” Model ................................................................................16
    2.3.1 Containing System ....................................................................................................18
    2.3.2 Designing System .....................................................................................................19
  2.4 Task 1: Industry, Government, and Academia Visits .......................................................20
    2.4.1 Discussion Perspectives (Anonymous) ....................................................................21
      2.4.1.1 Mission-Level Simulation Integration with System Simulation and Digital Assets ...21
      2.4.1.2 3D Environments and Visualization .................................................................22
      2.4.1.3 Dynamic Operational Views for Mission and System Simulation and Analysis ...23
      2.4.1.4 1D, 2D, & 3D Model Creation, Simulation, Analysis for Physics-based Design ...24
      2.4.1.5 Platform-based Approaches .............................................................................25
      2.4.1.6 Workflow Automation to Subsume Process .......................................................26
      2.4.1.7 Product Lifecycle Management ........................................................................27
    2.4.2 Decision Framework ....................................................................................................27
    2.4.3 Discussion Perspectives (Publically Known) ............................................................28

2.5 Gaps and Challenges .........................................................................................................29
  2.5.1 Complexity of Software and the Verification Challenge .............................................30
  2.5.2 Lack of Precise Semantics to support Model Integration, Interoperability, and Transformation ..................................................................................................................34

2.6 Discussion Narratives and Measurement Summary .........................................................35

2.7 In-Process Summary and Next Steps ............................................................................36
  2.7.1 Final Technical Report for RT-118 and Next Steps ...................................................36

Part II ........................................................................................................................................39

3 Task 1 - Assessing the State-of-the-Art MBSE ..................................................................40
  3.1 Task 1 - Process .............................................................................................................40
  3.2 Scenario Collection .........................................................................................................41
    3.2.1 Organizational Type ..................................................................................................42
    3.2.2 Organizational Scope ...............................................................................................42
    3.2.3 Factors Definition Example .....................................................................................43
  3.3 Discussion Summaries .....................................................................................................44

3.4 Predictive Model ..............................................................................................................44
  3.4.1 Rationale for Bayesian Networks .............................................................................45
  3.4.2 Data - Likert Scales (Ranked Scales) .......................................................................46

4 Task 2 - Common Model Lexicon .......................................................................................47
Figures

Figure 1. Four Tasks to Assess Technical Feasibility of “Doing Everything with Models” ......................... 7
Figure 2. Two Model Views: Mobile and Relative Size of Planets .......................................................... 14
Figure 3. Model Centric Provides Digital Integration Between Views ...................................................... 15
Figure 4. Dynamic Models and Surrogates to Support Continuous “Virtual V&V” Early in the Lifecycle .... 16
Figure 5. Putting the Vision into Context ................................................................................................. 18
Figure 6. Perspectives on Aspects of the Containing System ..................................................................... 19
Figure 7. Dynamic OV1 with Integrations to Other Models and Digital Assets ........................................ 24
Figure 8. Vehicle System Model ............................................................................................................. 26
Figure 9. DARPA META Program ........................................................................................................... 31
Figure 10. META Program Claims Conventional V&V Techniques do not Scale to Highly Complex Systems .......................................................... 32
Figure 11. The Trend Indicates that Software Increases by an Order of Magnitude Every 10 Years ........ 33
Figure 12. Complexity Results in Diseconomy of Scale In Software often impacting size, scope and cost estimates .......................................................................................................................... 34
Figure 13. Measurement Collection Instrument ....................................................................................... 36
Figure 14. Model Vision at Program of Record Scope and Integrate Risk-based Decision Framework .... 38
Figure 15. Collection Instrument Results .................................................................................................. 41
Figure 16. Spreadsheet Instrument Collection ....................................................................................... 42
Figure 17. Bayesian Network Underlying Collection Instrument ............................................................ 45
Figure 18. Sample Graphic Representation from Ontological Software .................................................. 48
Figure 19. Published Web Page from Data Collection Spreadsheet ......................................................... 51
Figure 20. Model Representation and Lexicon Tree .................................................................................. 51
Figure 21. Partial Graphical Representation of Lexicon ......................................................................... 52
Figure 22. Tabular Representation of Lexicon ......................................................................................... 53
Figure 23. SysML Context of IWC Vision ............................................................................................... 57
Figure 24. I & I Environment .................................................................................................................... 59
Figure 25. Vision Model Interface to Simulation and Test Resources ..................................................... 59
Figure 26. System Level Maturity Representation for Development and Verification ............................. 62
Figure 27. Model Topology Often Mirrors Architecture of System ........................................................... 63
Figure 28. Mission Context for System Capability .................................................................................. 64
Figure 29. Bayesian Model Derived from Airworthiness Factors ............................................................ 67
Tables

Table 1. Discussion Group Summary ................................................................. 21
Table 2. Initial Lexicon Capture Tool................................................................. 49
Table 3. Discussion Instrument Factor Definition............................................. 73
Acknowledgments

We wish to acknowledge the great support of the NAVAIR sponsors and stakeholders, including stakeholders from other industry partners that have been very helpful and open about the challenges and opportunities of this promising approach to transform systems engineering.

We want to specifically thank Dave Cohen who established the vision for this project, and our NAVAIR team, Jaime Guerrero, Eric (Tre’) Johnsen, and Ron Carlson, who has worked closely on a weekly basis in helping to collaboratively research this effort. Tre’ Johnsen has taken on the deputy role now that Larry Smith, and Ernest (Turk) Tavares have left the project, who we also thank for their support and contributions.

We also want to thank all of those stakeholders, including some from industry that will remain anonymous in recognition of our need to comply with proprietary and confidentiality agreements associated with Task 1.

As there are so many stakeholders that supported this effort, we wish to recognize them all. We sincerely apologize if we have missed anyone else that has supported our efforts.

Andrew Devine       Chris Owen      Joseph Tolarski     Richard Price
Art Pyster          Dale Moore       Judith Dahmann      Richard Yates
Aumber Bhatti        Dennis Reed       Leslie Taylor       Scott Lucero
Bill Brickner        Doris Schultz     Lissette Fortuno    Shahram Bavani
Bhawana Sharma       Fatma Dandashi    Michael Gaydar      Stan Rifkin
Brad Kidwell         Howard Owens      Mike Kelley         Stu Young
Brian Nolan          Jae Pfeffer       Paul Montgomery     Todd Standard
Brent Gordon         James Carroll     Philomena Zimmerman Tom Blakely
Bob Levitt           John McKeown
Executive Summary

This is an interim process report for the Systems Engineering Research Center (SERC) research task (RT-118). The RT focuses on a Vision held by NAVAIR’s leadership to **assess the technical feasibility** of creating/leveraging a more holistic Model-Based Systems Engineering (MBSE) approach, which we are now referring to as model-centric engineering. The expected capability of such an approach would enable mission-based analysis and engineering that reduces the typical time by at least 25 percent from what is achieved today for large-scale air vehicle systems. The research need includes the evaluation of emerging system design through computer (i.e., digital) models. The effort extends RT-48 to investigate the technical feasibility of moving to a “complete” model-centric lifecycle and includes four tasks as shown in Figure 1. The key tasks include:

- **Task 1**: Surveying Industry, Government and Academia to understand the state-of-the-art of a holistic approach to MBSE (model-centric engineering, “everything digital”)
- **Task 2**: Develop a common lexicon for MBSE, including model types, levels, uses, representation, visualizations, etc.
- **Task 3**: Model the “Vision,” but also relate it to the “As Is” and Airworthiness processes
- **Task 4**: Integrate a Risk Management framework with the Vision

![Figure 1. Four Tasks to Assess Technical Feasibility of "Doing Everything with Models"](image)

Our NAVAIR sponsor envisioned this research effort would take approximately two years, but due to the ending of the original SERC contract in December of 2013, this first phase (Phase I), under RT-48 had a period of performance of nine months. This report provides the interim process status about these ongoing tasks under the follow-on research task, RT-118, which started 1-April-2014. Since the kickoff...
of RT-118 there has been considerable emphasis on the discussions we are having with industry, government and academia.

We have conducted over 20 discussions with industry, government, and academic organizations, and have a few remaining. We see a movement towards a more widespread adoption of model-centric engineering, however not a radical transformation as desired of the NAVAIR vision. We have seen demonstrations of mission-level simulations that are being integrated with system simulation, digital assets and aircraft products providing new type of services. We have seen demonstrations of 1D, 2D, and 3D modeling and simulations with a wide array of solvers and visualization capabilities. We have been in an immersive Cave Automated Virtual Environment. We have seen the results of platform-based approaches directly focused on speed-to-market, and more, which is discussed in Section 2. However, as we are focusing on the goal of “25 percent reduction in time” for large-scale 5th generation air vehicle systems that must satisfy airworthiness and safety requirements as required by NAVAIR, we’ll discuss some challenge areas that are detailed in Section 2.5 that have been discussed at most meetings:

- Our NAVAIR sponsor often mentions in our discussions with organizations that 90 percent of the functionality of in a 5th generation air vehicle system is in software
  - The growth and complexity of software is creating verification challenges
  - The significant amount of software verification, which are essential to airworthiness and safety often has long durations
  - The aspects of software were not originally high on our list, but in model-centric engineering, software connects almost everything, and while the impact of software was not believed to be an issue in the past, it is likely to be going forward

- It was stated in meetings that there is an “explosion of models,” however
  - There is a lack of cross-domain model interoperability, consistency, and limitations in our ability to transform models with the required semantic precision to provide accurate information for decision making
  - Unvalidated models are used leading to incorrect or invalid results and organizations are not identifying design or integration problems until late in lifecycle

This list is not comprehensive. We need to understand the conditions associated with these challenges and their impact on the overall objective. The report discusses implications and alternatives to the transformation for NAVAIR. We have had follow-ups to our meetings on several different topics, and have more planned that are focused on some of the challenge areas. We have been asked to bring industry together to share their perspectives on challenges, issues, concerns, and enablers for a transformation. We want to explore ideas and concepts to improve the efficiencies and break down the barriers that inhibit speeding the needed capabilities to the NAVAIR and the warfighter.

The document is structured so that the key new efforts under the new RT-118 for the in-process review are described in Section 2. Section 1 provides some updated clarification on the scope given by our NAVAIR sponsor. Sections 3 through 6 provide additional detail to summarize the efforts that are aligned with tasks 1 through 4. Section 7 provides some conclusions and discusses the next steps and information planned for the RT-118 final technical report.
In 2013, the Naval Air Systems Command (NAVAIR) at the Naval Air Station, Patuxent River, Maryland initiated a research task (RT-48) to assess the technical feasibility of creating/leveraging a more holistic Model-Based Systems Engineering (MBSE) approach to support mission-based analysis and engineering in order to achieve a 25 percent reduction in development time from that of the traditional large-scale air vehicle weapon systems. The research need included the evaluation of emerging system design through computer models. The first phase of the effort created a strategy and began collecting and structuring evidence to assess the technical feasibility of moving to a “complete” model-driven lifecycle.

The larger context of the NAVAIR mission seeks a Transformation of Systems Engineering (SE) through MBSE, where MBSE is used in the most general and broad way. A key goal is to leverage virtual designs that integrate with existing systems data and simulations, as well as surrogates at varying levels of refinement and fidelity to support a more continuous approach to mission and systems analysis and design refinement. This broader view of the use of models has moved our team to use the term model-centric engineering, because the concept involves the integration of models with digital assets, from virtual integration through physical implementation.

The Vision of NAVAIR is to establish an environment to evaluate the emerging system design through computer models and demonstrate system compliance to user performance and design integrity requirements, while managing airworthiness risks. It is anticipated that this model-centric approach can streamline or radically transform the decomposition of requirements and their subsequent integrated analysis, which is currently aligned with the Department of Defense (DoD) systems engineering V-model (i.e., the “V”). By providing more tightly coupled and dynamic linkages between the two sides of the traditional “V,” more efficient and focused requirements decomposition would eliminate thousands of pages of documentation delivered via contract data requirements that now substitute for directly invoking, manipulating, and examining the design through computer-based models.

1.1 OBJECTIVE

This transformation initiative for NAVAIR is broad and can be thought about in at least three parts as it relates to our task:

1. The focus of this research task, RT-118, is scoped at the system level, some times characterized as the Program of Record (POR) plus weapons, for an air vehicle system
2. There is another related effort focused at the mission level, which must consider capabilities cutting across platforms of systems
3. There is a third effort focused on transitioning through adoption of model-centric engineering

While our directive is to focus on the technical feasibility of a radical transformation for item #1, our discussions with organizations and working sessions involving other stakeholders often have cross cutting concerns and implications. We do continue to document these various aspects of both enablers and challenges, some of which are included in Section 2, and other are documented in meeting minutes.

Therefore, the overarching and potentially controversially worded research question is:

- Is it technically feasible to use model-centric engineering (“Vision”) in order to achieve at least a 25 percent reduction in the time it takes to deliver an air vehicle weapon system
The emphasis by the sponsors is on the “technical feasibility” of such a Vision. It is acknowledged that there are many possible hurdles beyond technical feasibility (e.g., organizational adoption, training, usability, etc.), but they have in general been reduced in priority for this phase of the effort.

There are many additional research questions such as:

- What are the emerging technologies and capabilities that will enable this Vision?
- How will such a Vision work in the face of complex, human-in-the-loop, autonomous, and adapting system?
- Can such approaches work in the face of safety and airworthiness requirements?
- What are the technical gaps limiting the Vision?
- What are the approaches to deal with risk when making decisions based on virtual models and the associated simulations, surrogates and analyses?

There are four key tasks, which are described in this report, but a key decision made during the kickoff meeting was to attempt to “model the Vision” the rationale being that:

If one can “do everything with models” then we should be able to “model the Vision.”

The plan is to look to Industry (both contract developers of NAVAIR systems and MBSE technology providers), Academia and Government to identify the most advanced holistic state-of-the-art approaches to model-centric engineering and represent those aspects and characteristics in the “modeled Vision.” There are some things that will likely be challenging to model, at least for now (e.g., human cognitive system interactions), and therefore there will be a risk framework integrated with the Vision. This risk framework will also leverage other types of predictive models (e.g., stochastic) and methods to both embed decision-making knowledge and formalize both quantitative and qualitative information to support risk-informed decision-making.

Given this context, we have been further directed to reduce the scope for this effort to focus on the front-end of the lifecycle from pre-milestone A to critical design review (CDR), typically what is considered the front half of the “V” model. However, as is discussed in this report, many of our meeting discussions go well beyond this scope, as we consider the potential impacts that models or digital assets will have on the other phases of the lifecycle.

1.2 Organization of Document

Section 1 provides a statement of the research objectives, overview of the current progress and organization of this report.

Section 2 provides the in-process summary of our efforts and results since the kickoff of RT-118, 3-April-2014.

In order to be comprehensive, we’re including discussions of the four tasks in a manner consistent with the final technical report from RT-48 [10].

Section 3 describes the approach for having discussions with commercial, government, and academic organizations in order to assess the most holistic state-of-the-art use or vision for MBSE. This section provides also a description of a discussion measurement instrument and model that is being used to transform subjective information into a quantitative representation.

Section 4 describes the approach for developing a model lexicon to characterize such things as: model levels, model types, model uses, representations, and other categories related to “models.”
Section 5 discusses the scope and concept of the Vision model and its relationship to the “As Is” artifacts and process that are currently in place for developing NAVAIR air vehicle systems; this also includes the airworthiness process.

Section 6 discusses a framework for risk identification and management that is primarily focused on how airworthiness and safety risk can be integrated in the Vision model, but it will also deal with program execution risks.

Section 7 provides some conclusions with a brief summary of the planned next steps.

There is additional backup information, including the factor definitions associated with the discussion collection instrument, a list of acronyms and abbreviations following the conclusion.
2 IN-PROCESS SUMMARY

This section provides a summary of the in-process information required as an intermediate deliverable for RT-118. Sections 3 through 6 align with the task ordering shown in Figure 1. This section presents the following in-process information:

- What we mean by model-centric engineering and the model lexicon status
- A discussion about the Vision concept and enabling technologies related to theme of a Vision for the transformation (e.g., everything digital)
- Status of and perspectives on the visits to industry, government, and academia to seek out the most advanced holistic uses of model-centric engineering
- Perspectives on what we have heard from the visits as it relates to the “technical feasibility” of a radical transformation to model-centric engineering
- Discussion about some challenges areas
- Discussion collection and measurement instrument
- Summary and next steps

2.1 MODEL-CENTRIC ENGINEERING AND MODEL LEXICON STATUS

We were tasked to create a model lexicon (Task 2), however, modeling terminology can be confusing, and a simple definition is not always adequate as there are many overlapping definitions. Some of the terms are overloaded. While we did give some references and example uses in our lexicon, they don’t necessarily completely convey the broad concepts such as model-centric engineering. Other organizations (e.g., NASA) are working on similar definitions and groupings of terms, and we agreed to work more collaboratively on the lexicon.

Of particular note is that this task was characterized under the term Model Based Systems Engineering (MBSE), and we have repeatedly stated that NAVAIR means MBSE in the broadest way. Others suggested that we meant Model-Based Engineering (MBE) or Model-Based Design (MBD) or both. We think the term MBSE may be viewed too narrowly, and find ourselves working to clarify that we are thinking about the use of models that are integrated with other digital assets that can include integrated models, Software in the Loop (SiL), Hardware in the Loop (HiL), Platforms in the Loop (PiL) and humans in the loop, etc. NASA was one of our first discussion meetings and they used the term model-centric, and we think this term may better fit our perspectives and align with the more complete concept of the NAVAIR Vision (Task 3, see Section 5).

Status Task 2: we have captured over 300 named lexicon items related to the term “model,” including levels, types, uses, representations, standards, etc. The details are described in Section 4; we have recently delivered these model-lexicon artifacts to NAVAIR for them to post internally.

2.2 CLARIFYING THE VISION MODEL CONCEPT

At our RT-48 kickoff meeting, the question was asked “is it technically feasible to model everything?” As a result, we said that we would attempt to model the “Vision.” This effort is part of Task 3, and we have been making progress and illustrating similar examples that are discussed in working sessions. Two things have resulted from our efforts in researching what a “Vision” model might be, and how it might be represented:

1. We have seen only a few example fragments of Vision-like models. Organizations typically model the systems they want to develop and evolve. Often organizations don’t think about
modeling all of the elements of the environment (containing system), the elements and interactions of the designing system, including elements of existing system, subsystem and parts, in order to create an instance of the target system, which would be then stored in a version of what some call a “System Model.”

2. One organization has created at least a start of something that relates to the Vision; they are using the System Modeling Language (SysML). We started an example SysML model to illustrate this concept too. However, in some of our working sessions, we found that not everyone was familiar or comfortable about these perspectives using SysML modeling Views.

Therefore, we’re going to refrain from using those types of views in this section of the report. Instead, we will provide some examples to help with clarifying the concept of what we think should be included in the Vision.

### 2.2.1 WHAT IS A MODEL?

We have also come to realize that we are not sure that people understand what is meant by the term model, as well as having a consistent view on model-centric engineering, especially as it relates to the Vision. We are going to provide some details before moving on to the concepts involved in the Vision.

Modeling, in the broadest sense, is the cost-effective use of something in place of something else for some cognitive purpose. It allows us to use something that is simpler, safer or cheaper than reality instead of reality for some purpose. A model represents reality for the given purpose; the model is an abstraction of reality in the sense that it cannot represent all aspects of reality. This allows us to deal with the world in a simplified manner, avoiding the complexity, danger and irreversibility of reality [46].

George E.P. Box, said: “Essentially, all models are wrong, some models are useful”

One key aspect of models and modeling is abstraction, which supports communication through different views with various levels of details. Details of importance can be emphasized while other details are not described. Most of us have been exposed to models for a long time, for example, a mobile of the solar system as shown in Figure 2 shows the number of planets and might show the relative position of the planets, but it does not accurately show the plant’s size or distance from sun. Different views can provide alternative and relevant perspectives on the planets of the solar system and emphasize the relative size of the planets. To get an accurate perspective of a problem or solution often requires several views with some type of formal description of the relationship between the views. For example, the distance from the sun to each planet needs to be described using consistent units (e.g., miles).
Model-centric capabilities to achieve NAVAIR’s vision will heavily rely on computationally-based models. We need to use the relevant abstractions that help people focus on key details of a complex problem or solution combined with automation to support the simulation and dynamic analysis of both the problem and solution, along with the mechanism for combining the information collected from the various abstractions to construct a system correctly. Some of the key abstractions can be categorized into types, such as:

- **Structure** – 1D, 2D, 3D models, systems, subsystems, components, modules, classes, and interfaces (inputs and outputs)
- **Behavior** (functionality)
- **Timing** (concurrency, interaction)
- **Resources** (environment)
- **Metamodels** (models about models)

Many of these model-centric abstraction concepts have existed and evolved from programming languages, but within a programming language the combination of these views may be lumped or tangled together (e.g., spaghetti code). Most dynamic model capabilities cannot effectively leverage simulation, analysis or generation capabilities if the models are constructed in an ad hoc way.

Modeling methodologies (beyond process steps) are needed to guide the structuring of models to provide a means of systematically separating these views, because certain types of models are constrained to permit only certain types of information. Model-centric automation relies on automated means for analyzing the views, deriving information from one-or-more views, and ultimately pulling sets of views together correctly to produce some type of computationally-based system, simulation or associated analysis artifacts and evidence.

---

2.2.2 OPERATIONAL PERSPECTIVE OF MODEL-CENTRIC INTEGRATION

Therefore, our model-centric view means integrating different model types with simulations, surrogates, systems and components at different levels of abstraction and fidelity. Figure 3 provides an example documented in a case study that was published in 2008 [29]. While this was possible then, it doesn’t go as far as the Vision we believe NAVAIR is seeking. Hidden behind the scenes, there was manually created code to integrate the levels and views. This reflects on the types of software skills that will be required to assemble model-centric simulations for analysis until we improve the integration and interoperability of models (see Section 2.5 for discussion on challenges).

![Figure 3. Model Centric Provides Digital Integration Between Views](image)

Extending the previous example and relating it to a scenario of moving through the lifecycle phases, our team provided another representation that was included in the RT-48 final technical report [10] that extends this concept and relates to the lifecycle phases. This example is also abstract, but reflects on a NAVAIR objective, which is to “cross the virtual V” early in the lifecycle in order to better ensure that the system design meets the SoS mission need, essentially collapsing the timeline. Consider the following scenario using the typical release phase reviews as the time points to represent a notional timeline moving from left to right (e.g., System Requirements Review (SRR), System Functional Review (SFR), Preliminary Design Review (PDR), Critical Design Review (CDR)).

In a model-centric engineering world the models at SRR would reflect on the new high-level aircraft needs/capabilities, as conceptually rendered in Figure 4; there is likely a strong relationship between these new operational capabilities and the mission needs. These models would need to be sufficiently “rich” that we could computationally connect them to other surrogates, such as software components (new/legacy), hardware and physical surrogates (e.g., previous generation aircraft). We ideally want to have some type of dynamic operational capability operating from the very beginning of the effort (all digital). As we transition through the lifecycle phases SFR, CDR, and PDR, we would use a similar process on lower-level models and components that provide increasing levels of fidelity that is moving us closer.
to the actual system as the physical design is created and refined. We begin to focus on detailed functional and timing behavior, with models that predict performance characteristics and begin to clarify the margins; we would continue the transition from the use of surrogates to versions of the implemented design. As we continue to move through to CDR, especially for 5th generation air vehicle systems, we’re going to have much more software than ever before, including software that connects models with the simulations, surrogates, components and live or historical environmental data.

![Diagram showing phase design/ payload maturity](image)

**Figure 4. Dynamic Models and Surrogates to Support Continuous “Virtual V&V” Early in the Lifecycle**

Increasingly there will be much more complexity in the software prior to PDR and CDR, and this creates different concerns for NAVAIR from prior generations of air vehicle systems. Testing will be required to ensure that the continuously refined representation of the system models and implementation meet the timing (temporal interactions) and performance needs.

Challenge: Given this scenario, what is NAVAIR’s role? Does this perspective have any negative impacts on the “technical feasibility,” or is this just some type of operational perspective. We raise this point, because it does have an implication on the “Vision” model, which is additionally clarified in Section 2.3.

### 2.3 Task 3 - Vision and “As Is” Model

Task 3 is to model the “Vision” and the “As Is,” including the Airworthiness process. This is a joint effort with:

- “As Is” process model being led by Ron Carlson, from Naval Postgraduate School (NPS)
- Airworthiness aspects being led by Richard Yates and Rick Price from MITRE
- Vision being led by Mark Blackburn

Details can be found in Section 5, but we want to make some clarifications related to the Vision. We have used the term, and provide some descriptions, but want to be clear about what it means to us, as
we are trying to look to other organizations to see if they are pursing similar types of efforts using model-centric engineering.

We typically hear about a System Model [2], [52], which should represent all the data and information to produce the target system. In this case, it is scoped to the Program of Record (POR) for our task. The information to cover what we believe to be the Vision should include:

- Sufficient information about the **containing system** (information about environment)
  - This information should come from the mission analysis
  - We know that NAVAIR is conducting some similar type of research effort at the mission-level, and we have requested some type of discussion meeting on this subject
  - This is briefly discussed in Section 5, but a high-level perspective is provided in Section 2.3.1

- All the information about the **designing system**
  - Every tool for model creation, storage, simulation, analysis and their interconnections that is used to create, simulation, or produce analysis information related to decisions or dependent information
  - We give an example in Section 5

- All other platform related information that provides some aspects of the interrelated capabilities associated with the POR (**system to be design/evolved**), including revisions, variants, and even trade spaces analysis results from design alternatives not selected

Some of these perspectives are provided in Figure 5 (these related figures come from a briefing given by Jaime Guerrero that is approved for public release; distribution is unlimited). This figure puts into perspective the mission capability threads that have relationships to different PORs for the different platforms, and puts into context some of the aspects of the containing system (i.e., the interrelationship to other PORs in order to support a capability). This image abstractly reflects on the information about the existing assets of previous systems (platforms) that can play a role in model-centric engineering:

- These would be represented in some type of reference architecture
  - All of the existing elements (components) that could be put into a system
  - The relationships (valid connections)
  - Model representation of new elements (components)
- These provide the building blocks for defining and refining a new capability
- Currently, this information is largely defined in documents; it may be partially modeled, and/or held by contractors (data, models, knowledge)

Therefore, in order to realize the Vision, NAVAIR going forward needs total and continuous access to this type of information.
2.3.1 CONTAINING SYSTEM

The Containing System must represent the System-of-System, including environment and resources with sufficient fidelity and semantic precision. In general, a complete high-fidelity representation is not possible, therefore there will be some type of abstraction of the containing system such as reflected in Figure 6. This is one scenario of a capability, and the particulars of the interface parts can include the environment, such as the ship, landing surface, arresting system, weapons, weather, threat types, operators, etc.
Some of the current approaches to modeling used by NAVAIR such as DoDAF models, are static and do not capture enough information. They are semantically imprecise when it comes to representing behavior and the temporal (timing) interaction to be able to assess and predict the needed performance. Our field visits to commercial and industry organizations, discussed in Section 2.4, reflect on modeling and simulations capabilities that are neither well integrated nor interoperable. While the interest and intent exists, the standards do not keep pace the technologies. We heard that there is an inadequate level of cooperation to foster the needed integration and interoperability. We make this point, because it is important to the overarching research question.

2.3.2 Designing System

The Designing System is the entire enterprise, which includes model capture, synthesis, simulation, analyses, testing, configuration control, workflow automation, etc. The idealized goal is to transform all information so that it can be used for simulation, synthesis, analysis and ideally “models for manufacturing,” (e.g., “3D print the system”). This is not realistic for the entire system or all of the parts, at least not today.

We have had a number of discussions with industry suppliers and users of technologies, and there are technologies that support multi-domain and multi-physics-based 1D, 2D, and 3D modeling, analysis and simulations, but we need go beyond. This means that NAVAIR has options as there are competing commercial suppliers, but the emerging capabilities often outpace the standards, and impact integration and interoperability between different vendor capabilities.

Challenge: This can make it harder for NAVAIR, because they cannot impose one set of tools, so do they need to maintain all?

---

2 bigstory.ap.org
One particular challenge that we discuss in the meetings with organizations is software. Jaime Guerrero usually brings up the point in our discussions that: “90 percent of the functionality in 5th generation air vehicle systems (e.g., F-35) is in software.” While the first flight of the F-35 was 15-December-2006, we still do not have a flight certified system. While we have 1D, 2D, and 3D types of physics-based models for simulation, optimization and analysis, we don't have very many models of software where formal analysis and testing can be performed to the degree it is done for physics-based models.

There are reports that software testing is taking a long time (GAO report [27]). While there is use of models, the detailed software behavior is often written manually, which minimizes the ability to formalize analysis, generate the code, and automate test, with the possible exception of Simulink (but not everything is modeled like a control system). This is one of the greatest concerns to the goal of reducing 25 percent of the time. This is an area of investigation for DARPA, and we'll have more discussion about this topic in Section 2.5.

Challenge: Most of these capabilities don’t address the software problem previously mentioned
- More thought needs to be given to this topic if we hope to better be able to assess 5th generation designs at PDR and CDR
- Software has major implications on airworthiness today
- Even the auto industry is now being impacted by safety guidelines associated with software

### 2.4 TASK 1: INDUSTRY, GOVERNMENT, AND ACADEMIA VISITS

This section summarizes some information about the visits with industry, government, and academia. In coordinating the agenda and at the start of each meeting we usually posed the question:

“Tell us about the most advanced holistic uses of MBSE (model-centric engineering) you have seen in use on projects/programs or that you know about”

We send our coordinator package, which explains the overarching goals of the research task, and we often iterate with the organization about an agenda. Our interests ultimately relate to the question posed by our NAVAIR sponsors:

Do we think it’s “technically feasible” for an organization like NAVAIR to have a radical transformation through model-centric engineering (everything digital) and reduce the time by 25 percent for developing and delivering a major 5th generation air vehicle system?

This section summarizes at a high-level what we heard about the responses to the “technical feasibility” question of model-centric engineering, but also about other factors that will help with a transition as opposed to a radical transformation. As of August 7th 2014, we have visited or coordinated with the following organizations as shown in Table 1. Early preliminary meetings were used to refine the factor categories used in our discussion collection and measurement instrument, which is explained in more detail in Section 3. We used the discussion collection measurement instrument to transform our subjective views of the meeting into quantitative values. The discussion meetings summarized in the third column typically have occurred at organizations’ business operations or at NAVAIR.
Some of the discussions with industry and commercial organizations are often governed by some type of Proprietary Information Agreements (PIA) or Non-disclosure Agreements (NDA). In addition some of the provided material is marked in a manner that limits our distribution of the material. Due to the need to sign a PIA/NDA, we are being careful to not disclose the organizations in this report. In addition, because we cannot disclose information about commercial or industry organizations, we are limiting how we discuss the other organizations too.

We have created meeting minutes, which generalize the information we heard during the discussions. NAVAIR wants to share it with our NAVAIR research team, therefore we are including the following notice on meeting minutes, per Jaime Guerrero, Director, SEDIC - AIR-4.1, NAVAIR:

DISTRIBUTION STATEMENT D. Distribution authorized to the Department of Defense and U.S. DoD contractors only. Other requests shall be referred to SEDIC Director, AIR-4.1, Naval Air Systems Command, 22347 Cedar Point Rd., Bldg 2185/Rm 3A54, Patuxent River, MD 20670 - (301) 342-0075.

These meeting minutes are not part of the official RT-118 deliverable, but because this report is an official deliverable and is likely to be publically available, we are not going to include any information about the organizations that we met with, rather we’re going to provide a generalization through the following narrative, and provide our measurement collection results to date.

### 2.4.1 Discussion Perspectives (Anonymous)

This section provides a high-level summary of the wide range of information we discussed in the meetings. Some of the meetings were scoped to particular disciplines and domains that would support the overall NAVAIR objectives, as there are often many areas of expertise required to cover the engineering efforts of an entire system. We are going to present the summaries in a top-down manner starting from a mission-level scenario.

#### 2.4.1.1 Mission-Level Simulation Integration with System Simulation and Digital Assets

One organization demonstrated mission-level simulations that are being integrated with system simulation, digital assets and aircraft products providing new type of services:

- We attended a live (with some artificial data) multi-scenario system-of-systems demonstration that runs on a modeling and simulation (M&S) infrastructure that integrates with other M&S capabilities as well as live products that can be hosted within a cockpit or operate through server and web-based services.
- The scenarios represented commercial version for DoD-equivalent mission analyses
- The M&S infrastructure is used to both analyze new types of services that can be added to their portfolio, but is integrated with other existing systems and can be populated with real or artificial data
- These capabilities are used in a way that improves their own systems and capabilities, but they use these capabilities to solicit inputs from potential customers on new types of products and services
  - A scenario was provided about capabilities that are part of a services platform to support logistical planning through real-time operations and maintenance
  - New digital products such as real-time health management that integrates through web-services connecting pilots in the air with maintenance operations at the next airport
- However, even with the advancements this organization discussed some challenges with developing the integrations as there was not a grand architecture or plan when they first started developing the underlying infrastructure
  - We have heard this several times from other organizations too
  - This is the challenge for both representing simulations of the Containing System, and then integrating them through the Designing System, existing systems and components, and new model-centric designs

Implication:

- Companies are advancing the types of technologies needed by NAVAIR, because they are leveraging new business opportunities out of some of the enabling technologies of yesterday
- The integration to make it a reality is still challenging, as they are trying to leverage existing (legacy) systems that were not necessarily designed to be integrated
- This example not only addressed part of the scope of our research task, but should provide some type of relevant information for the other NAVAIR initiative scoped at the mission-level
  - We also have a discussion with another organization who is moving this way, and there may be information that NAVAIR could leverage in such an effort

2.4.1.2 3D Environments and Visualization

Several organizations demonstrated (or showed results from) some of their 3D and visualization capabilities

- One organization uses two different types of 3D environments for customer engagements, but also for on-going (often daily) design engineering analysis and review sessions in 3D environments
- They do use commercial technologies, but have developed a significant amount of infrastructure on their own
- We heard similar stories by others about the need to develop their own infrastructure
- We also visited the Cave Automated Virtual Environment (CAVE) where we were immersed in a virtual 3D environment that is used for both analysis and design review purposes

Implication:
These capabilities provide a collaborative environment not only for design, but for continuous reviews; this scenario aligns with a concept we continually discuss, which was stated at the kickoff review by Dave Cohen.

If NAVAIR is going to integrate and/or transform the SETR process to leverage these types of capabilities, they will need to define some new types of methodological guidance to align with a model-centric approach.

### 2.4.1.3 Dynamic Operational Views for Mission and System Simulation and Analysis

There are modeling environments to create dynamic Operational Views (e.g., an OV1) to understand and characterize Mission Context for the needed System Capabilities, as shown in Figure 7:

- At the systems-level, where our task is scoped, the industry is making progress.
- In traditional DoD models, we are used to static Operational View (OV1), but the newer capabilities allow for dynamic operational scenarios.
  - People often use these capabilities for analysis.
  - But they can also be leveraged as scenarios for testing.
- In many instances these types of capabilities have integrations with other types of models, simulation and analysis capabilities, similar to Figure 3.
- This example comes from an organization, and while we did not speak with them in any of our discussion meetings, we have had a number of interactions with them through Stevens Institute of Technology on other research tasks.
  - Being consistent with our goal to not single out any organization, we provide an image credit, but will not mention this company directly.
  - Many of the organizations we spoke with use the tool kit, which has an evolving set of libraries that can be used through their platform to support dynamic visual-based analysis.
  - The example discussed in Figure 3 used this tool kit at the OV1 level.
  - NAVAIR mentioned that they use this tool kit too.
  - Note: this is not the only product in this space.

**Implication:**

- Model-centric engineering is moving beyond static DoD views.
- The computational and visualization technologies bring both the behavioral views into perspective, but can increasingly bring the temporal aspects into play.
- This provides much more information to support decision making.
2.4.1.4 1D, 2D, & 3D Model Creation, Simulation, Analysis for Physics-based Design

We have heard from a number of commercial organizations that discuss 1D, 2D and 3D model creation, simulation, analysis, and management capabilities focused primarily on physics-based design, with increasing support for cross-domain analysis:

- Most are focused on physics-based models
- Some have unique capabilities and there is an increasing trend/push to support broader cross-domain analysis through better integration of different types of models
- Some allow for the plug-in of niche-capability libraries and solvers, using a platform-based approach to create more of an ecosystem (i.e., “think apps”)
- Some are customizable to leverage High Performance Computing (HPC)
  - That is, they have been programmed to take advantage of parallel computation
  - While this is typically assumed, it is not the case – we spoke with organizations that stated that organizations may use HPC, but the simulations/analyses are not always programmed in a way to take advantage of the speed derived from parallelism
- There are challenges in model transformation and/or interoperability, and the need for formalized semantics is known
  - We have heard the concerns both from the commercial organizations, who are continually working to address the situation in the tools they sell
  - We heard similar concerns from the industry organizations
  - See Section 2.5.2 for more details
- There are also multiple suppliers that often provide a suite of tools that cover some portion of the lifecycle

---

3 Image credit: AGI
Some of the major players acquire companies to better complete their portfolio of tools; we know that just because a company acquires a product that there is not necessarily seamless integration with the other products.

Implication:

- This is necessary, but not sufficient as many of these capabilities still do not deal with software; see Section 2.5.1 for more details
  - Multiple organizations that rated highly acknowledge the challenge of software that continues to in increase in complexity
- Without better model integration, interoperability, and/or model transformations, how is NAVAIR going to deal with so many types and variants of models

### 2.4.1.5 Platform-based Approaches

Platform-based approaches are used not only by the commercial tool vendors as discussed above, the developer of systems have been improving their approach to use platform-based approach and use virtual integration to help refresh systems and do system upgrades on periodic schedules, which is business critical

- We heard two organizations discuss platform-based approaches that are tactically driven by the periodic cycles demanded for sales roll outs at 12, 18, 24, 30, and 36 month delivery cycles
  - 12, 18 months - they might change feature colors, but every interface is exactly the same, and no electric changes
  - 24 may make some minor changes, and electrical
  - 30 change types of subsystem, component (e.g., Figure 8, based on approach that uses the Modelica and Functional Mockup Interface (FMI) standard)
  - 36 months – complete redesign
- For longer cycles times, they think more of a “W” rather than a “V,” where the first part of the “W” might cover more of the prototyping efforts, and the second part of the “W” is more related to verification of the implementation
- Unlike NAVAIR, they completely decouple R&D from development (even the W). This means that some of the key aspects of what we’re looking at for this research project, from pre-milestone A to Critical Design Review are not part of their typical roll out process
2.4.1.6 Workflow Automation to Subsume Process

We have continued to have conversations about workflow automation; the ultimate idea is that if we could model everything including the process, we could “generate” information to drive a workflow engine that would completely subsume the process. This would get NAVAIR to the place where decisions were completely data-driven, and the process would be autonomous and adaptive, and coordination would replace planning.

- Automated workflows arose from the manufacturing world
- There are workflow engines to drive the enterprise operations, but when we asked the question do you use a model to generate the information to drive your workflow engine, they said:
  - No, but that’s a good idea
  - It seems that most workflow engines are programmed more manually
  - We do think this is possible in the future, and we’ll give an example about someone that seems to have this type of vision – this is one of the meetings that we’ve coordinated, but we have not yet had the discussion meeting
- To a lesser degree, there are other types of products that provide workflow automation support integrations for work such as design optimization
  - We spoke with both commercial companies that provide these capabilities
  - We spoke with industry companies that use these technologies
  - They do help speed up and make the design optimizations more efficient
  - We have another meeting with a commercial tool vendor that has some elements of a workflow engine, and we’ll discuss their perspectives on the future
    - The key reason for this concern/question is that the effort in modeling the “As Is” process is reflecting that it’s potentially too difficult to ever fully create or follow a document-driven process
    - Workflow automation has the potential to subsume the process
Implication:
- In the future, we think that a Vision model would allow workflow engines to completely subsume the process and drive every decision based on real-time data; this could completely subsume the SETR.
- This will be critical as the process cannot adjust quickly enough to adopt new technologies that will be needed to keep pace with NAVAIR’s need for speed to address continuously emerging threats.

2.4.1.7 Product Lifecycle Management

One organization is modeling elements of their Vision concept from a holistic perspective and relating that to the artifacts captured in a System Model. Holistic approaches invariably bring in the need for some type of product lifecycle management (PLM) so that every piece of data/information is captured and traceable from design through manufacturing and possibly training. While this again might be slightly out of scope for the objectives of our research task, we briefly report on this, because it was covered in a number of discussions with other stakeholder from NAVAIR who are directly concerned with the need.

- We have heard two of the large commercial companies discuss their myriad of products, including the Product Lifecycle Management (PLM) systems.
  - We will be hearing about another solution from one of NAVAIR’s lead technology suppliers that has a related product.
- Some companies said that the key reason for moving to PLM is for tracking every single item in a vehicle for warranties and recalls.
  - They said that they could/can not live without it, as they need to know every feature associated with every vehicle.
  - They are now working on how to fit software into this model in a manner similar to hardware components.
- If NAVAIR could characterize every type of data element required within a Total System Model, this would provide the schema for information that should and could be captured in a PLM-type system.
- This would/could be used to support every decision that needs to be made.
  - Every time an artifact was obtained a workflow engine could trigger automation of additional analysis or trigger individuals that a decision was needed based on new data.

Implication:
- The Vision model concept is an enabler for the types of information that would need to be captured/stored by NAVAIR in order to make decisions.
- Information at the finger tips of the SMEs could work towards improving the speed of decision making.
- However, a PLM system is only as useful as the information it contains – and it is not inherently transformational.

2.4.2 Decision Framework

Our collaborator Gary Witus has been working with the Army’s TARDEC, and he arranged a discussion with them. TARDEC described the work they do as an acquisition organization that has challenges similar to NAVAIR. This discussion was focused on publically known information, and therefore, we’ll share some of our perspectives from our visit. TARDEC has developed and is evolving a framework...
called the Integrated System Engineering Framework (ISEF); this capability could provide a decision framework for NAVAIR that can be layered on top of MBSE. While the information from this meeting may not directly address the research question for a radical transformation, the information we received seems valuable, as ISEF is complementary for a transitional model-centric approach.

Briefly, ISEF is a Web-enabled collaborative decision-making framework to support knowledge creation and capture built on a decision-centric method, with data visualizations, that enables continuous data traceability.

- The framework integrates a number of different technologies that support decision-making applicable to different phases of the lifecycle, for example:
  - Requirements – they have their own requirement management capability
  - Feedback mechanism
  - Portfolio management
  - Architecture (through other MBSE tools)
  - Tradespace analysis
  - Risk
  - Road mapping

NAVAIR held a follow-up session 5-August-2014 to discuss possibilities of using ISEF within NAVAIR.

Implication:
- In the Vision, can NAVAIR or does NAVAIR need to have all of the capabilities to produce “all of the data”
- Rather If NAVAIR can formalize the information/data needed for decisions, could they require that type of information to be delivered as digital artifacts in the contract to enable their decision-making
- This might be a useful way for NAVAIR to think about adoption and transformation

2.4.3 DISCUSSION PERSPECTIVES (PUBLICLY KNOWN)

We hosted a Panel Session at CESUN 2014, the 4th International Engineering Systems Symposium, June 11, 2014. Mark Blackburn was the moderator, and he aligned the theme of this panel discussion with our research task. We titled the talk controversially:

“Is it Technically Feasible to Do Everything with Models?”

We’ll briefly summarize the event and information, because it was presented at a conference and is public domain.

- We asked our panelists to answer the same question that we pose to the people on our site visits
  - What’s the most advanced a holistic MBSE (model-centric) approach you’ve seen (or know of)

As this conference has an academic tone, we tried to get panelist that might provide a slightly more academic perspective, pointing to some challenging areas for future research. Briefly, we covered a few points:

- Increasing complexity is a real challenge
One speaker (Stephan vanHorn from Honeywell) represented the Aerospace Vehicle Systems Institute (AVSI), which is a member-based collaboration between aerospace system development stakeholders that aims to advance the state of the art of technologies that enable virtual integration of complex systems

- The System Architecture Virtual Integration (SAVI) team believes a model-based virtual integration process is the single most effective step the industry can take to deal with complexity growth

Some high-level information about the responses from our panelists about question: “Is it Technically Feasible to Do Everything with Models?”

- Axel Mauritz (Airbus Group) – No
  - He confirmed the move towards the concept of a Reference Technology Platform for a platform-based design system
  - Final thoughts:
    - Not practical to do everything in models, for example: hard to represent “ilities”
    - Modeling efficiency - What is worth (value of) to model?
    - Legal question - How to sign a model?
    - Total system representation - How can we model, what we don’t know (system interaction, unintended/ emergent behavior)?

- Chris Paredis (National Science Foundation) – No
  - He provided a good characterization for the need of precise formalism in models in order to address some concerns of semantics and model transformations
  - Emphasized the importance of modeling based on the value (e.g., efficiency, reliability, performance, demand, cost) of the results
  - Challenges:
    - Integration of different views
    - Integration of different formalisms
    - Holistic elicitation and incorporation of knowledge
    - Ontologies

- Stephan vanHorn (Honeywell/SAVI) – “Never say Never”
  - Described a Model Vision from the beginning through to the Supply Chain
  - Work needed:
    - Integration of descriptive models (e.g., SysML) and executable models (e.g., Simulink)
    - Incremental certification using provably correct virtual design – address verification concerns for safety (e.g. Formal Methods and MBD annexes to DO-178C)
    - Sufficient system model fidelity to elicit emergent behavior to test for unknown unknowns

Some of the comments from our panelists provide a good lead in to the next section on some of the gaps and challenges associated with the “technical feasibility” question.

### 2.5 Gaps and Challenges

During our site visits, we asked the organizations to share some of the gaps and critical challenges too, some similar to what our panelist from CESUN mentioned in Section 2.4.2. Some provided inputs
beyond the question of “technical feasibility,” including some other good ideas, like the decision-framework discussed Section 2.4.2. We heard many different types of challenges such as:

- Lack of **affordability** of projects and activities
- Mission **complexity** is growing faster than our ability to manage it
- Not identifying design or **integration** problems until late in lifecycle
  - We emphasize integration, as the concept of cross-domain simulation from models has been pointed out before
  - Complex systems have greater cross-domain dependencies, and many of the modeling and simulation efforts are not doing analysis in terms of the integration of models and their associated simulations
  - In addition, we stress that once integration occurs, it requires more precise semantics: structurally, behaviorally, and temporally; these may be some of the biggest challenges related to the technical feasibility question for the research task
- Having to **hunt** for data or supporting material during mission anomaly resolutions
- **Inability to share** models in a collaborative environment
  - This point again may relate to the underlying semantics of models in specific domains that are not easily shared
- Ineffective use of precious **testing** time and resources
- Too many design **reviews** that focused on the documents vs. the design
  - This too is an interesting point as Dave Cohen is asking for a more continuous approach to review based on the models – will this help people focus more on the design aspects
  - In addition, properly constructed models can enable significant analyses that could eliminate reviews
- Use of **unvalidated models** in simulations leading to incorrect/invalid results
  - We have heard this point several times

We’re focusing on the goal of “25 percent reduction in time” for major air vehicle systems that must satisfy airworthiness and safety requirements required by NAVAIR. Therefore, we’ll emphasize two key challenge areas in this subsection that have been discussed at most meetings, which include:

- Growth and complexity of software and the verification challenges, which are essential to airworthiness and safety
- Cross-domain model interoperability, consistency, and transformability with the required semantic precision to provide accurate information for decision making

### 2.5.1 Complexity of Software and the Verification Challenge

The strict requirement for safety and airworthiness for the NAVAIR air vehicle systems requires comprehensive rigor in verification. As 90 percent of the functionality of in a 5th generation air vehicle system is in software that implies a significant amount of software verification.

NASA presented industry data indicating that verification is 88 percent of the cost to produce DO-178B Level A\(^4\) software, and 75 percent for Level B software [15]. These types of verification requirements are required for many aspects of NAVAIR vehicles, such as the control laws for the F-35. As shown in Figure 9, the DARPA META pre-program solicitation (META) describes how continually increasing complexity impacts the verification costs of software and delivery time [6]. META claims that the fundamental

---

\(^4\) DO-178B/C is the Software Considerations in Airborne Systems and Equipment Certification document dealing with the safety of software used in certain airborne systems. Level A is a characterization for the most safety-critical aspects of the software, and required a more comprehensive amount of software verification.
design, integration, and testing approaches have not changed since the 1960s, as shown in Figure 10. The META program goal is to significantly reduce, by approximately a factor of five, the design, integration, manufacturing, and verification level of effort and time for cyber physical systems. The complexity has increased for integrated circuits, as it has for software-intensive systems, but the developers of integrated circuits have maintained a consistent level of effort for the design, integration and testing efforts, as reflected in Figure 9. The need is to understand key reasons why software-intensive systems production is different from integrated circuits. One fundamental difference is that software behavior requires nonlinear operations and constraints that are implemented on computing hardware where operations are performed and results stored in floating point representations. This makes the automated verification problem more challenging than for integrated circuits, where automated verification and analysis is based primarily on logic or bit-level manipulations. Chip developers used to rely on simulation, much like software development uses debugging and manual testing, but the chip verification would cost more than 50% of the effort and defects that escape to the field could cost $500M. They now rely more on formal methods and tools to support development and verification.

Implications:

- In the past software might not have been a major concern moving through the PDR or CDR decisions, but it may be going forward as we have not heard of many breakthroughs that can significantly reduce the time for software verification of safety-critical systems.

Figure 9. DARPA META Program

---


6 DARPA META program APPROVED FOR PUBLIC RELEASE. DISTRIBUTION UNLIMITED
Figure 10. META Program Claims Conventional V&V Techniques do not Scale to Highly Complex Systems

There may be many differences between hardware and software, and we briefly summarize the points:

- Software behavior often relies on floating point variables with nonlinear relationships and constraints, which is not the case in integrated circuits.
- This requires different mechanisms for analysis and verification than are used in hardware analysis and verification.
- Other than models like Simulink, the detailed software behaviors (functions) are still written mostly by hand, limiting automated analyses.
  - Some discuss the use of automated generation of code.
  - But many are using coding frameworks, which can generate the code structure, but the detailed behavior is written in the code.
  - Newer approaches that rely on domain-specific modeling are being researched through DARPA efforts, but most have not become mainstream [45], [47].

Figure 11 was created in the early 2000s, and now the number Lines of Code (LOC) in the F-35 is about 9,000,000, which is almost an order of magnitude beyond the F-22. We are trying to get data to make the comparison, even though we know there are many types of technology differences from the way software is produced today versus in the 1990s.
The problem is that with software there are diseconomies of scale as reflected in Figure 12:

- In software, the larger the system becomes, the greater the cost of each unit.
- If software exhibited economies of scale, a 100,000-LOC system would be less than 10 times as costly as a 10,000-LOC system, but the opposite is almost always the case.
- Based on data from Boehm, as shown in Figure 12, the effect of the diseconomy of scale isn’t very dramatic when the range is within the 10,000 LOC to 100,000 LOC.
- Some automobiles may have several million lines of code, but they are distributed to small micro controllers.
- No one piece of code, at least today ever gets to be the size or complexity of Mission Systems Software in the F-35.

Implication:

- We need to think about this problem differently, and software does matter to NAVAIR at PDR and CDR, and JSF is a prime example.
- We didn’t get many new answers relative to this item from our industry visits, but we know that the F-35 first flight was 15-December-2006, and the latest GAO report indicates that software testing is still a key problem [27].
  - It is important to note that the mission system software has very unique capabilities (e.g., data fusion), and these types of capabilities are unlike a new Microsoft operating
system that is often beta tested by millions of people prior to release; it is also not safety critical
- Our ability to meet the 25 percent reduction in time is likely to be driven by our ability to verify software, as it seem inevitable that the number of LOC will continue to increase
- We need to understand the conditions limiting the way to do the verification now, and determine if there are alternative (and possibly radically new) approaches

![Figure 12. Complexity Results in Diseconomy of Scale In Software often impacting size, scope and cost estimates](image)

**Source:** Computed using data from the Cocoma II estimation model, assuming nominal diseconomy of scale (Boehm, et al 2000).

### 2.5.2 Lack of Precise Semantics to Support Model Integration, Interoperability, and Transformation

As presented at CESUN by the SAVI representative (see Section 2.4.3), there is an explosion of models. With all of the unique and advanced capabilities, systems engineering needs to manage the integration of all of the disciplines across many domains:

- We may have a “sea” of models, simulators, solvers, etc., but they lack consistent meaning across or between them
- There is lack of precise semantics especially in both behavior of models and timing/interactions of models
  - We have covered this point many times in working session
  - This point was made at the kickoff meeting of RT-48 by Dave Cohen, and has been reported by many others in our visits [1], [39]
- This is limiting the full spectrum of analyses and simulations needed to provide adequate coverage over a system’s capabilities
- Some are looking at how to work and integrate a federation of models and digital assets, but that is not an ideal solution

One of the hosts of a discussion meeting requested that we, the SERC, bring industry together to discuss these types of challenges in order to address this problem.
Implication:

- We have acknowledged that there are aspects of information/data that will be used in decisions where there is uncertainty about information/data.
- Task 4 is the risk framework we intend to include within the Vision model.
- We have an upcoming discussion with an individual that was involved with the developers of the methodology for Quantifying Margins under Uncertainty (QMU), developed at Sandia National Labs, and has also used the DAKOTA framework in support of these types of methods to risk on air vehicle systems.
- ISEF also includes some support to manage risk, and we are having ongoing discussion with TARDEC about ISEF.

### 2.6 Discussion Narratives and Measurement Summary

The objective for our team during the site visits was to facilitate conversations through discussions that draw out insights into leading advances in model-centric engineering. As discussed in Section 3 (Task 1), we developed and refined a guideline for our collective NAVAIR team to hold discussions with Industry, Government and Academia (organizations) to understand the most state-of-the-art and holistic approaches to MBSE (model-centric engineering). We have created a collection instrument (spreadsheet) to collect inputs after each discussion. The instrument provides a constructive approach to conduct a discussion with organizations as well as a way to provide some type of quantitative measure associated with using subjective information to rate the “state-of-the-art” of a holistic approach to the “technical feasibility” of the model-centric engineering. We use that information with a predictive model that uses a qualitative subjective approach that computes a probabilistic value associated with cross-cutting aspects (factors) associated with the technical vision for this task.

As shown in Figure 13, the model produces two probability distributions, one for the Technical Risk State of the Art (max of 10), and another for the Technical State of the Art (max 100). We think these factor values will provide a probabilistic value that is related to the technical feasibility questions, and help in reflecting on the factors that are enablers, as well as help identify where gaps exist that must be addressed through risk identification and management. We have made some adjustments to the factor weightings based on some of the discussions we have had with organizations, and need to make further adjustments.
2.7 IN-PROCESS SUMMARY AND NEXT STEPS

Since the kickoff of phase 2 of this effort under RT-118, we have spent most of our time conducting and documenting the information we received from the 13 discussion meetings, as well as coordinating the remainder of the planned discussions that we would like to conclude by September as reflected in Table 1.

During RT-48, we held periodic (~monthly) working sessions at NAVAIR. Since the kickoff in April we have conducted working sessions in April and May. Due to our travel and meeting discussion, we have held a number of virtual meetings to discuss the “As Is” and Airworthiness efforts (also part of Task 3). The next working session at NAVAIR is scheduled for the week of 18-August along with a plan for up to three discussion meetings.

The working sessions usually cover the status of all four tasks. Recently, a key theme of these meetings was to better understand key aspects related to Task 3, and specifically the set of “As Is” artifacts that are produced by NAVAIR as they move through the process (primarily the Systems Engineering Technical Review [SETR]). The key artifacts are needed for assessing Airworthiness, and how these artifacts and processes would relate to a “Vision” model. The understanding of the “As Is” artifacts should allow us to think about the roles models need to play in the SE transformation through MBSE; that is how the models with analysis, simulation, and synthesis/generation capabilities can replace document-based artifacts and the associated manual processes. Potentially more challenging is to understand how artifacts for Airworthiness (e.g., flight readiness) can be understood as early as Preliminary Design Review (PDR); what are the models and evidence to support these type of safety-related decisions. Additional information is provided below about the working sessions.

2.7.1 FINAL TECHNICAL REPORT FOR RT-118 AND NEXT STEPS

While we have a few discussion meetings to conclude and several follow-up meetings:
We will continue to have follow-on discussions with a number of the organizations that we have visited (Task 1)

We need to discuss the content and representation of the lexicon (Task 2) that has been transferred to NAVAIR

The rest of our efforts relate to continuing the efforts associated with Task 3 and Task 4, as reflected in Figure 14 involving a more joint collaboration with the NAVAIR team and subcontracts such as Naval Postgraduate School (NPS) and MITRE. The following enumerates our ongoing efforts:

1. Developing a model representation of a derived list of artifacts that are currently produced to support NAVAIR System Engineering Technical Review (SETR) process (see Section 5.5 for details)
   - This is led by Ron Carlson from NPS
   - A CORE model was started, but some recent decisions about consolidating the efforts of the Airworthiness process with the “As Is” process representation has moved this modeling effort to an IBM modeling tool; NAVAIR prefers the IBM tools
   - It is important to understand the artifacts that are produced to comply with the “As Is” process, along with the relationship and dependencies among these artifacts
   - In the Vision, the information described in these artifacts (some of which are models today) must be ultimately represented in models (digital form), or be derivable from models
   - The modeling efforts have revealed that there is a knowledge that needs to be transferred from SME to these models; that is, there is a considerable amount of knowledge held by SMEs that are not explicitly in the process or check lists
2. Representation of the “As Is” process, which relates to the DoD 5000.02 and SETR process
   - The analysis of the “As Is” artifacts and process should provide a means to assess the completeness of the Vision, and help people understand how a process would work when transitioning from a document-centric operational model to a model-centric approach
3. Development of a representation of the Airworthiness Process
   - This is led by Richard Yates and Rick Price from MITRE
   - Their involvement in this effort drove to the decision to select Rational System Architect as the tool for modeling the “As Is” and the associated airworthiness aspects
   - This effort will characterize those critical aspects that make the NAVAIR requirements more challenging than for other organizations
   - The types of required Airworthiness evidence (e.g., Flight clearance) must be identified and presented either in some model representation (4) and/or support risk-based decision making, which should be captured in conjunction with the Vision (5)
4. The Vision model representation that subsumes all information that is currently represented in the “As Is” process, if deemed to be necessary, and all of the associated digitized automation that is required to transform the process
   - This is led by the SERC team
   - As discussed in Section 2.3, this should represent the containing system, designing system, and include the platform-based reference elements for creating any air vehicle system instance
   - This is being significantly informed by the discussions (Task 1)
5. The integrated risk framework (see Section 6 for details)
   - This led by the SERC team
   - The discussion with the experts on Quantitative Margins under Uncertainty should be valuable inputs
We are also working with TARDEC to understand how their risk management would play into this effort.

The work on RT-107 also plays a role into how this is represented.

6. Describing an associated process for applying the Vision model; in many instances, when the information is formalized in model, a corresponding model-driven automated workflow is also automated, however, because of the aspects of risk and airworthiness, it is likely that there are some human-driven steps in the process.

This led by the SERC team.

There is a significant amount of effort that would be required to characterize a vast spectrum of modeling methodologies; this is beyond the scope of this research task.

Sections 3 through 6 incorporate information from the RT-48 Final Technical report [11] that is related to tasks 1 through 4. We have included it in this document, because some of the information in this section refers to this information. We also wanted to make it more convenient for our reviewers, rather than requiring them to obtain the RT-48 Final Report. As we proceed through our discussions, this information will be extended and refined to reflect the current state of our efforts.

Section 7 provides a summary and discusses a few key items that need to be discussed moving forward.
Part II

Much of the material in the remainder of the document has been extended from the RT-48 Final Technical Report [11]. Some of this material has been documented in the bi-monthly status or working session meeting minutes. We include it in this in-process review report for your convenience.
Section 2 provided an overview of some of the responses from our meeting with industry, government and academia. Our team developed a guideline for our collective NAVAIR team to hold discussions in an effort to understand the most state-of-the-art and holistic approaches to MBSE (model-centric engineer). The objective for our team members is to facilitate conversations through discussions that draw out insights into leading advances in model-centric engineer. We agreed early on with the sponsor that open-ended discussions, as opposed to surveys, would bring out new and more innovative approaches and strategies. We are particularly interested in demonstrations of actual technical capabilities, and we have seen a few, and we also want to understand the critical gaps and limitations too, which we briefly summarized in Section 2.

We created a collection instrument. The instrument provides a constructive approach to conduct a discussion with organizations as well as a way to provide some type of quantitative measure associated with using subjective information to rate the “state-of-the-art” of a holistic approach to MBSE. We are using a qualitative subjective approach that computes a probabilistic value associated with crosscutting factors associated with the technical Vision for this task.

The collection instrument uses an Excel spreadsheet as the input mechanism to collect factor values about an organization’s use of MBSE as discussed in Section 3.2. Each row in the spreadsheet represents the subjective information associated with one organization. The latest version of the instrument includes one organizational classifier and 22 factors.

Status: Sections 2.4 and 2.5 summarizes our perspectives on the “technical feasibility” and implications to NAVAIR from our 22 discussions. We have the possibility of at least seven more meetings, two of which are confirmed.

### 3.1 Task 1 - Process

After a meeting with an organization, we request our team members who conducted the discussion to:

- Complete one row of the spreadsheet; see Section 3.2 for details on the collection process
- Write a short summary reflecting on the key unique capabilities of the organization

The spreadsheet responses are incorporated in a master list. The value for each factor will be entered in a modeling tool, which quantifies the subjective inputs provided to the tool, as shown Figure 15. The maximum value of the mean of the probability distribution is 100. As reflected in Figure 15, it was decided that because there are some organizations that require confidentiality or proprietary information agreements, we have decided to keep the names of all organizations anonymous. In addition, a narrative will be created for each organization; this narrative will highlight the most key capabilities and challenges, but be generalized to ensure each organization’s anonymity. Additional details about interpreting the results are provided in Section 3.2.
3.2 **SCENARIO COLLECTION**

After each discussion we complete the spreadsheet collection mechanism as shown in Figure 16 by working through the row and uses the pull down menus to select a factor value of Low, Medium, or High (see Section 3.4.2 for details on Ranked factor values). A complete list of factors is provided in a worksheet tab of the spreadsheet collection mechanism titled: Factor Meaning-Definition. Example definitions are provided in Section 3.2.3, with some additional rationale; a complete set of definitions is provided in Discussion Collection Instrument Guide and provided in the back up material of this report.

Team members may want to use one spreadsheet to collect all of the discussions; it is possible and acceptable that after a few meetings with organizations that one or more of the factor values be changed in order to be more globally consistent. The key is not to identify a particular organization, rather the objective is to identify if there are state-of-the-art methods, tools, processes and innovative strategies that are being used to significantly advance the development and deployment of systems through MBSE and related approaches, and to incorporate these concepts in the Vision model (see Section 5).
### 3.2.1 Organizational Type

The general convention used is:

- **Academia** – this should include strictly academic organizations; other organizations performing research should either be Industry, Commercial or Government
- **Industry** – these are organizations that are using MBSE to develop products and systems (e.g., those contractors to NAVAIR that produce air vehicle systems)
- **Commercial** – this is a special case of Industry that relates to MBSE product developers
  - These organizations either develop MBSE tools and services, or may apply them with Industry or Government customers
  - These organizations are in the list, because they may have insights into some of the leading or novel uses of the tools, and they are aware of the need to continually advance their own product and services
- **Government** – this includes military, and other non-military organizations such as Department of Transportation, and the FAA

### 3.2.2 Organizational Scope

One challenge for some of the initial uses of the collection mechanism was to appropriately reflect on the organization scope for which these MBSE usage questions apply. Remembering that the key objective of the survey is to assess the "Technical Feasibility" of "doing everything with model." We recognize that actual adoption can be difficult, and might not make sense on older systems. Therefore it is probably best to hold discussions with individuals in the roles such Chief Engineer, Chief Technical Offer, Program Manager or some MBSE technical experts in the organization. To carry this a step further, it might also be important to keep the "systems" perspective in mind, because some of the concepts discussed may have been applied at the hardware level and possibly in types of software (e.g.,

---

**Figure 16. Spreadsheet Instrument Collection**

<table>
<thead>
<tr>
<th>Organization (consider the use of an anonymous ID)</th>
<th>Candidate Factors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Industry, Academia, Commercial, Government, Other</td>
<td>Magnitude, Proven, Crossing Virtual V, Cross Domain Coverage, Virtual System Representation, Management and Criticality Risks (Relates to Task 4), Attributes of Modeling Maturity, Operational Risks (Relates to Task 4), Indirect support from Models</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Industry 1</th>
<th>Site</th>
<th>Med</th>
<th>Pick</th>
<th>Low</th>
<th>Low</th>
<th>Hi</th>
<th>Med</th>
<th>Low</th>
<th>High</th>
<th>Low</th>
<th>High</th>
<th>Low</th>
<th>Low</th>
<th>Med</th>
<th>High</th>
<th>High</th>
</tr>
</thead>
<tbody>
<tr>
<td>Academia 1</td>
<td>Depa</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td></td>
</tr>
<tr>
<td>Commercial 1</td>
<td>Business Unit</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
</tr>
<tr>
<td>Government 1</td>
<td>Program</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
<td>Pick</td>
</tr>
</tbody>
</table>

Menu for selecting factors value (L, M, H)
the control laws for the F35 are built in Simulink, with auto code generation, and a significant portion of auto test generation), but these types of approaches may only be emerging in use at the systems level. We seek to understand how comprehensive the use, and also need to understand the technical gaps. The technical gaps areas will likely need to have additional focus related to risk management (Task 4).

Finally, this research is not limited to NAVAIR, however when thinking about NAVAIR systems the scope is often quite large and involves large-scale multi-year programs, where the systems are actually built by one or more contractors.

Therefore, we would like to know the organizational scope associated with the MBSE discussion: Program, Project, an entire Business Unit, Platform (e.g., a type of a specific aircraft, tank, automobile), Department, or Site.

3.2.3 Factors Definition Example

The factor categories do not necessarily relate to specific MBSE practices, rather they are higher-level characteristics of the organization’s ability to leverage the use of models and the associated technologies that enable simulations and accelerate the analyses, design, synthesis, V&V and manufacturing processes. For example:

- **Crossing the Virtual V** is a high-level category that has significant weighting in the model, because our sponsor emphasized this as a critical need and the ability to understand the design capabilities through early V&V activities at the system and mission level (as opposed to the subsystem or component level). Therefore, this factor category has three main factor characteristics:
  - **Simulation of Integration**
    - If an organization has simulations of integration or integrated simulations across domains of the system, and especially at the “higher” levels of the “V,” this is a likely indicator that such an organization is likely to have the ability to understand simulations of the system within the context of a mission, and there is a better understanding of the integration impacts, because the simulations are integrated or represent integration, including critical temporal aspects in simulation
    - This includes the integration of surrogates, use of instrumented systems, actual system components, new prototypes, and/or in development
    - Other attributes of this type of simulation, would be human-in-the-loop, as well as multi-level mixed fidelity simulations that provide the right abstractions at the right level
  - **Formal analysis**
    - This means that the analysis is automated, because the models are semantically rich; we are looking for automated analysis, rather than looking at humans performing the analysis
    - Models are increasingly have more semantic richness that enable automated-types of analysis, and models are increasingly being integrated (see factor category Cross Domain Coverage)
  - **Domain specific**
    - These types of systems involve the integration of many disciplines
    - Models need to provide the relevant abstractions that are related to the domain of the engineer performing the work; domain-specific modeling is an emerging type of modeling that often provides the relevant abstractions, with the semantic richness to enable automated analysis, simulation, synthesis (generation) and automated test
• DARPA-sponsored research that demonstrated the capability for continuously evolving Domain Specific Modeling and analyses in 2008 as an emerging capability and theme [21], [45]. In contrast, modeling languages like System Modeling Language (SysML) are general purpose [32] they generally lack the semantic richness needed for formal analysis leveraging for example formal methods of automated V&V [11]; while they may be understood by system engineers, control system engineers would prefer Matlab/Simulink, and other engineers may require other domain-specific models and tools (e.g., computational fluid dynamics, radio-frequency, heat transfer). However, SysML does provide an underlying framework for holding system model information [51], yet the models are not executable even with existing plug-in authoring tools [18] (see Section 5.1 for more details).

3.3 DISCUSSION SUMMARIES

We request small narratives for each organization, where the discussion should highlight those organizational uses of MBSE that represent the greatest advances.

For example:

An industry organization provided a historical perspective on evolutionary adoption of MBSE. Some of the more exceptional characterization related to advances they have made that included: composable designs, common platforms, catalogs of capabilities, and synthetic representations of the environments. Much of their efforts were grown out of SysML-centric approaches. They are actively using model-based metrics, Live-info for simulation, predictive analytics based SysML, interpreting and tracing data consistency across disciplines, data set metric, trusted models for downstream production, PDM consistency and semantic links; these efforts include working the MBSE efforts top-down and bottom-up, starting first with an understanding workflow, and meta-spec concept.

As a second example, we talked with a SME from the Department of Transformation that has been involved in crash testing of automobiles for many years. The focus of our question was not as broad as the factors covering the lifecycle of a NAVAIR program, but more interested in the question: “are automotive companies reducing the number of crash tests through the use of model?” While, we did calculate a score for the responses, some of the factors may not directly relate, and therefore, we are requesting a narrative to help with elevating awareness of specific uses of models that may have relevance to the overall research task. There is a factor in the model that describes Scope Impact, this value is used to characterize the scope of an organization’s use of models.

3.4 PREDICTIVE MODEL

This section is provided for those interested in more details about the mechanism for converting the subjective factors into a quantitative number. The model is created using a Bayesian Network [44] (BN) tool. There are two basic reasons we selected this approach, BNs:

- Provide for the translation of subjective information into quantitative probabilities
- Allows for the use of subjective expert qualitative judgment and captures the casual relationship between subjective factors

The outputs are also probability distributions, which means that they provide some type of range to provide a comparison between different organizations. The specific numbers are not necessarily as important as our ability to compare different organizations and relate the responses back to advanced uses of MBSE and related enabling technologies. While no organization may have all “High” values, this
approach provides a way to look at relative comparison in conjunction with the narratives. Each of the nodes in the BN shown in Figure 17 provides a type of weight called a conditional probability. We have used the team’s judgment to weight the different nodes in a way that would relate to evaluating the key question for this task: is it technically feasible to “do everything with model.” In addition, we will refine the weightings as we proceed through discussions.

Figure 17. Bayesian Network Underlying Collection Instrument

3.4.1 RATIONALE FOR BAYESIAN NETWORKS

A Bayesian network is a representation, which organizes one’s knowledge about a particular situation into a coherent whole [22]. They are increasingly being used in the modeling of uncertain and incomplete knowledge. Bayesian thinking is inherently more intuitive than many other evaluation techniques; it best reflects commonsense thinking about uncertainty that humans have. We frequently use words like “likely,” “rarely,” and “always” to express varying degrees of uncertainty. Subjective probability is our way of assigning numbers (between 0 and 1) to these different degrees of uncertainty, and our probabilities can change as we are presented with new information, or we have new experiences which cause a shift in beliefs or expectations. When this shift occurs, the way our probabilities change are governed by Bayes’ rule.

A Bayesian network, as used in this framework, is a joint probability distribution and as such, any question that can be asked in a probabilistic form can be answered with a stated level of confidence. Some typical questions might be:

- Given a set of effects, what are the causes?
- How can an outcome be controlled, given a set of circumstantial values?
- If we model a causal relationship, what result would an intervention or change bring?

While there are several ways to structure a Bayesian network, we used prior experience to structure the model. The subjective factors in the spreadsheet instrument map directly to the yellow oval nodes of the BN model. The purple rectangles are intermediate nodes and generally relate to factor categories.
The orange rectangles represent the probability outputs of both Technical state of the art (Task 3) and the Technical Risk state of the art (Task 4).

3.4.2 **Data - Likert Scales (Ranked Scales)**

The subjective factors in the model use a Ranked node type, which is a type of Likert Scale. It is important to note that although Likert scales are arbitrary, they can retain a level of reliability for our use. The value assigned to a Likert item has no objective numerical basis, either in terms of measure theory or scale (from which a distance metric can be determined). In this case, the value assigned to a Likert item has been determined by the researcher constructing the Bayesian network, but can be refined as the research progresses. The results have been a balanced representation of strata and detail.

Typically, Likert items tend to be assigned progressive positive integer values. Likert scales usually range from 2 to 10 – with 5 or 7 being the most common. In this model, 3 levels are used, at least for now as it minimizes the number of computational states, which minimizes time for the analysis. The progressive structure of a Likert scale is such that each successive Likert item is treated as indicating a ‘better’ response than the preceding value. Note that the direction of ‘better’ (i.e., Higher) depends on the wording of the factor definition, which is provided in Section 3.2.3.

In terms of good practice, a bias in the computations may result if the suppliers of data for the framework do not agree on the relative values of each factor. However, there are enough factors that a bias in a one or two values will likely not skew the results significantly.
4 TASK 2 - COMMON MODEL LEXICON

The team was tasked at the kickoff meeting to create a common lexicon for things related to modeling in the systems engineering domain, and in fact, in the broader engineering space. An example of this is what is meant by the word "model." Most engineers will agree that a model is a facsimile of reality. Yet, to an industrial engineer, a model may represent a production facility; to a mechanical engineer it may be a finite element model analysis; to a systems engineer it may be an IDEF0 or a SysML representation of the system, subsystem, or some lower level element. None of those perspectives are wrong; they are just different views of some part of the same enterprise.

Some claim that there is no existing model lexicon or taxonomy [3], although there are a number of different types of taxonomies that all fit within the more general context of a model lexicon [19], [51]. The Object Management Group (OMG) in conjunction with INCOSE has established an Ontology Action Team to work on similar efforts [41]. The NDIA Modeling & Simulation Committee is about to approve the Final Report on the Identification of Modeling and Simulation Capabilities by Acquisition Life Cycle Phase [4].

Status: we have captured over 300 named lexicon items related to the term “model,” including levels, types, uses, representations, standards, etc. The details are described in Section 4; we have delivered these model-lexicon artifacts to NAVAIR for them to post internally.

4.1 ONTOLOGY VS. LEXICON

According to Wikipedia, ontologies are the structural frameworks for organizing information and are used in artificial intelligence, the Semantic Web, systems engineering, software engineering, biomedical informatics, library science, enterprise bookmarking, and information architecture as a form of knowledge representation about the world or some part of it [50]. The creation of domain ontologies is also fundamental to the definition and use of an enterprise architecture framework.

A lexicon is a similar concept – it is normally a book or glossary like document, or words (and their definitions) in a language or domain, arranged in alphabetical order. The team decided that a simple glossary would not be sufficient because it does not show the relationships between terms.

In simplistic terms, an ontology becomes a complex network of words, and their relationships to each other. A lexicon is a glossary. Neither was exactly what was needed for this project. Instead a hybrid is needed. The team needs something that provides definitions and simple relationships – not complex, rigid definitions. We chose to use the word Lexicon, though the words could also be represented in a tree-like structure that is common for ontologies.

4.2 TOOL FOR REPRESENTING WORD RELATIONSHIPS

There are tools available for creating ontologies. There actually exists a class of workers that consider themselves Ontologists. These tools come in many different flavors – from open source tools to commercial tools. The common thread is that they create graphical representations as shown in an example in Figure 18. These tools require rather rigorous definitions and relationships to complete. The open source tools are actually very good, and very robust. However, after some evaluation of available open source tools, the team decided that it would be better to create a straightforward spreadsheet of terms (e.g. a Lexicon), and then create a script that could represent that lexicon graphically.
Figure 18. Sample Graphic Representation from Ontological Software

4.3 THE LEXICON

A spreadsheet was first created in Excel. At first, the team was simply capturing the words, their definition, and where it made sense, a key reference or two for that definition. Table 2 shows the implementation of this data gathering spreadsheet. Once the decision was made to create a tool to make this information available graphically, and also on the web, it became apparent that a "relationship" data element was necessary. Therefore, the data collection tool captures:

- Name
- Has Parents [0 or more] separate with ";;" if more than one
- Definition
- Sample Usage
- Also Known As
- Key URL (optional)

The current spreadsheet represents a continuous accumulation of relevant terms, their definitions, and their classification. The initial definitions have been drawn from readily available sources on the Internet (often from Wikipedia where the assumption is that it has been created by a group of people with both knowledge and passion about the subject). In other cases members of the research team have authored a definition based on their understanding of the term in a relevant context. The team is using the spreadsheet feature of GoogleDocs to foster a collaborative effort.
Intuitively, many of the terms in this spreadsheet are ambiguous and their meaning is highly dependent on the context and usage domain. This has been found to be true in reality also as terms are collected from various domains. It is therefore important to emphasize that this is an evolving process.

### 4.4 Sources of Information

There were a number of sources used for this initial Lexicon. Journal papers on MBSE provided a good first cut. Interestingly, an article from the Journal of Object Technology [26] proved to be very useful. Other sources included The Open Group, the Object Management Group, INCOSE, NDIA, and Wikipedia.

### 4.5 Web Presentation

A short script was created that takes the information contained in the data-entry spreadsheet, and publish the results to the web. Figure 2 shows the published page as it looks at the time of this report, (http://www.markblackburn.com/MBSE/model_lexicon.html). This page includes four sections:

- Model Lexicon Overview (Figure 19)
- Model Representation/Lexicon (Figure 21)
  - This is a generated image produced by vizGraph, but with over 300 lexicon items it is difficult to use, although it reflects the interrelationships of the lexicon elements
- Hyperlinked Tree of the Model Lexicon (Figure 20)
  - As an alternative, a collapsible and expandable tree (outline) allows people to understand the hierarchy of model lexicon with hyperlinks to a particular lexicon definition.
- Definitions - A common structure is used for each term (Figure 22)

---

9 The final location of the lexicon may move to another location.
Model Lexicon Overview

Generated by lexicon2html; created by Mark Blackburn, Rob Cloutier, Gary Witz and Erik Hole
Last update: 2014-03-19

This is the first version of the Model Lexicon. This effort was initiated as part of a Systems Engineering Research Center (SERC) task for a NAVAIR project to investigate the possibility of doing everything with models. This lexicon is focused on providing a common language for all to use in the development and evolution of this effort.

This page includes four sections:

  Model Lexicon Overview (this section)

  Model Representation (a graphical representation of the tree - can take a long time to load)

  Hyperlinked Tree of the Model Lexicon (click to go to the definition)

  Definitions - A common structure is used for each term. There is a:

    Name: definition
    Sample Use: TBD
    Parent: hyperlink to parent in table
    Tree: hyperlink list of the tree associated with the term, if tree is expanded
    Sources: other related sources or key references

A Excel modeling template file with a few examples is here.
Model Representation

There is a graphical representation of the lexicon generated by vizGraph.

Click here image.

Note: this is a large image and may take a few seconds to load.

Model Lexicon

Model Lexicon Tree

Model lexicon

Concurrent engineering
+ Definition
+ Model
+ Model acquisition
+ Model levels
+ Model management
+ Model representations
+ Model types
+ Model uses
+ Modeling approach
+ Modeling standards
+ System
The definitions table shown in Figure 22, is a screen image from the website, and includes the following columns:

- **Name**
- **Definition**
- **Parent**
  - This is a hyperlink to the parent in the table
- **Tree**
  - This is a hyperlink back to the collapsible and expandable tree (outline); clicking on this hyperlink takes the focus back to the name in the tree only if the item is expanded in the tree
- **Sample Use**
- **Key Source** (if applicable)
### Model Lexicon Definitions

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition</th>
<th>Parent</th>
<th>Tree</th>
<th>Sample Use</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>2d modeling:</td>
<td>a geometric model of an object as two-dimensional figure, usually on the Euclidean or Cartesian plane</td>
<td>mechanical modeling</td>
<td>mechanical modeling, 2d modeling</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3d solid modeling:</td>
<td>the product of 3D solid modeling</td>
<td>mechanical modeling</td>
<td>mechanical modeling, 3d solid modeling</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3d solid modeling:</td>
<td>is the process of developing a mathematical representation of any three-dimensional surface of object (either inanimate or living) via specialized software</td>
<td>mechanical modeling</td>
<td>mechanical modeling, 3d solid modeling</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| Model driven architecture: | a software design approach for the development of software systems | model technique | model driven architecture |                                | http://www.omg.org/MDA |
| Model levels:            | Level of the system or system of systems; Also discussed in terms of Resolution level. The amount of detail or degree of aggregation employed in the model or simulation | model lexicon       | model lexicon, model levels | Often used in modeling and simulation world to discuss the different types of models |                           |
| Model lexicon:           | the words used in a language or by a person or group of people             | model lexicon         | model lexicon               | This is a lexicon associated with terms derived from models and modeling |                           |
| Model management:        | Approaches to managing models                                              | model lexicon         | model lexicon, model management | Configuration control, PLM, CATIA |                           |

Figure 22. Tabular Representation of Lexicon

4.6 **RECOMMENDATIONS MOVING FORWARD**

1. We expect as the effort continues, team members will continue to collaborate in the definition and classification, causing discussion related to their relevance and “correctness.”
2. Additionally, the intent is that the broader community will contribute examples and sample usages of the terms to improve the understanding and proper use in various contexts.
3. We will therefore provide mechanisms that allow for feedback/annotation from the community and a basic change control process.
4. It might be good to add a “comment” link on each table row on the website that could link directly to the corresponding row in the Google spreadsheet to enable the submission of a new terms and definitions directly into the spreadsheet (or database).
5. A longer-term plan would be to drive the graphical image, and textual listing from a database instead of a spreadsheet.
5 TASK 3 - MODELING THE VISION AND RELATING TO THE “AS IS” AND AIRWORTHINESS PROCESS

Section 2.3 briefly discussed the concept of the Vision model, which is not a representation of a NAVAIR air vehicle system, rather the Vision model must include the required information (data) and embedded knowledge that is normally captured in documents, drawing, specifications, pseudo-formal models, and tests (some refer to it as the “total” system model [52]). This was discussed in terms of the containing system, designing system and ultimately the system instance that is the design. This includes or subsumes every piece of information that relates to the artifacts captured in the “As Is” process, but should also include formalized information such as the inputs and outputs of modeling and simulations, analyses, surrogates, risk information, etc. and include specific versions of each tool, simulation, and analysis engine used to support the necessary evidence required to produce an airworthy system version. Ideally, this should include every piece of information to the Bill of Material (BOM), including models to manufacturing and models to training.

Preliminary discussions with organizations suggest that some individuals and organizations understand the Vision model concept. Some are attempting to develop variants on the concept that are more specific to product development. Some have cross-business/discipline projects established to refine strategies to roll out and support adoption by programs in these different business units. Other efforts are focused more at the software level (using the characterization Model Driven Engineering [MDE]) [30]. One study cited a multi-level, multi-domain instance case that started at the campaign level moving down through the mission, engagement, and engineering levels [1]. There are also organizations that claim to be applying MBSE, yet they have not seen the benefits; we understand that there are often adoption challenges [13], and that is why our sponsor has directed us to focus on the technical feasibility for this phase of the research.

From a high-level perspective, as reflected in Figure 14 Task 3 is a collaborative effort being worked by our SERC team, SMEs from NAVAIR, Naval Post Graduate School (NPS), MITRE, and consultants who have extensive NAVAIR and aircraft system engineering experience. This section provides a summary of our Phase I efforts.

Status: Section 2.7.1 discusses the specific tasks and assignments for developing the “As Is” combined with Airworthiness process, and the Vision models.

The following subsections are presented primarily in chronological order associated with research investigation, working sessions, and task scoping and refinement.

5.1 “AS IS” PROCESS AND AIRWORTHINESS PROCESS

Ron Carlson from NPS is leading the modeling of the “As Is” process. Ron’s model includes a large number of the artifacts that are produced as part of the Systems Engineering Technical Review (SETR) process. What he has been spending his time on is attempting to understand the inherent processes that are not formally described to further refine both the artifacts and overlay a process. One could argue that the definition of the “As Is” process is near complete. Many of the artifacts are just named items with no formalized definition of the artifacts or resources used to produce them. One could also argue that it is time to begin movement to the “To Be” state (towards the Vision).

Richard Yates and Rick Price from MITRE have just recently met with Airworthiness subject matter experts (SMEs), John Jabara, Bob Lehman, Tre, and John McKeown. They are using those discussions to
model the Airworthiness aspects of the “As Is” process as some of the authoritative documents do not provide a lot of detail.

- We heard about the Engineering Data Requirements Agreement Plan (E/DRAP) as a key artifact that is used to go to flight readiness assessment.
- We need to decompose E/DRAP as a metamodel (all of possible artifact classes and their relationships). Normally the E/DRAP is done in terms of allocated baselines that characterize both the operational effectiveness and operational maturity.
  - It is recommended that we work backwards from all E/DRAP-required information classes that are needed for airworthiness decisions, and them characterize the inputs and associated processes required to produce the E/DRAP information.
- We need to understand how would this be done at a more abstract logical level, where the fidelity is lower (e.g., at PDR)?
- Remembering that the Vision model is about system data and information that must be produced to go through the different decision gates, therefore, we need to hear more about the E/DRAP to better understand how it would relate to a Vision model.
- On a related note, it was stated that if the SEMP and SEP define the required criteria for a PDR, then from an artifact perspective, we need to understand what needs to be accomplished to meet the criteria for PDF, and how that would relate to the E/DRAP.

We have had numerous discussions about tools and modeling notations and languages, not only for the “As Is” process but also the Vision model. It was decided in July in a meeting with Jamie, that the “As Is” and Airworthiness process would be modeled in Rational Software Architect.

It is unlikely that engineers are going to use a model to guide the process, rather the model (not just process) needs to map to the workflow that inherently represent the process that guides the work, additionally characterizing the roles of individuals, tools and other resources that are needed to either produce, review, or consume artifacts that would be produced and captured in a “System Model.” This is not a new idea, but what we are interested in doing is modeling the Vision, and using modeling capabilities to analyze it to ensure that there are no issues (inconsistencies, dependencies) in the model, and again ideally generating the information to enable a workflow automation to inherently control the Vision processes in producing a system model for any air vehicle system.

Several discussions made it clear that there is more effort needed to communicate the concept and purpose of a Vision model, the types of information being captured and its relationship to the Containing system, the POR, and the Designing system. In addition, while we are following the planned approach to “model the Vision,” the lack of understanding of modeling languages/methods such as the SysML seems to cause some individuals not familiar with SysML to not understand the approach we’re taking. Again, this approach seems to be similar to one that NASA/JPL is using [2], but we may need to do a little more just-in-time training on modeling notations, if we are to use these models to illustrate where we are, where we’re going, and identify the gaps.

5.2 Vision Objectives

With a number of discussions (22 as of August 2014) behind us, it is a fairly consistent message that many organizations have not defined a Vision model. Instead they are involved in an evolutionary process of model adoption, and many want to better understand the return on investment (ROI). Some organizations do have to address some airworthiness and safety-related requirements and those efforts can lead to longer delivery schedules. Even the automakers are expending more resources in their need to address safety constraint. In addition, some of these organizations are working on a subset of the
problem (e.g., V&V) [32], while others are approaching this from the contractor point-of-view, which is significantly different from that of NAVAIR. NAVAIR is working in the early stages of DoD 5000.02 [24] lifecycle (i.e., Milestone A, B, C), and they ultimately produce requirements and design constraints that are provided to the contractors.

The objective for the Vision should address the questions:

- Can we create models to cover every type of artifact that is required to produce a system and comply with DoD and NAVAIR processes and requirements (e.g., Airworthiness)?
- Can we use model-based simulation, analysis, synthesis and generation to rapidly traverse the “Virtual Vee” and continuously, both horizontally and vertically, leverage evolving digital representations (e.g., models, surrogates) to assess the system design at various levels of fidelity in the context of continuously evolving mission scenarios?
- How does the risk framework fit into the model?
- What would a “Model-Driven” process be? With model-driven processes there are many types of automated workflows engines that automate the processes inherent in producing and relating model-based artifacts (see Section 5.6 for an example).

We initially developed (as a straw man, see Section 5.4) an example model in System Modeling Language (SysML) that represented the Integrated Warfighter Capability (IWC). The example provided a common understanding that the goal of the modeled Vision is going to formally characterize all of the “data,” relationships, automation throughout the entire lifecycle, including for example the relationship to data used by, and produced by modeling and simulation, analyses and other resources, as well as evidence captured within the models to support risk assessment and management (see Section 6).

5.3 MODELLING AND TOOLS FOR THE VISION

Our team has had numerous discussions about modeling representations, languages and tools for the Vision. The examples in Section 5.4 use SysML, which is a standard modeling language. It is general [32], but there are limitations. The basic SysML diagrams in the modeling environments are mostly static. System engineering models defined in SysML are descriptive in nature and do not directly produce analytical results [32], nor are they executable [18]. Different tool vendors provide extensions or their own analytical capabilities that solve SysML parametric diagram [14]. Since the parametric relationships are solved as a system of equations, the analytical model is limited to simple equations. To be able to use more sophisticated engineering analyses, external analysis tools need to be connected to SysML models. Other research efforts are attempting to leverage other standard modeling languages such as Modelica that have a broad range of analytical support through integration between SysML and the Modelica [43]. Modelica is a standardized general-purpose systems modeling language for analyzing the continuous and discrete time dynamics of complex systems in terms of differential algebraic equations. Domain Specific Modeling environments (e.g., Simulink for control systems) often have richer semantics (e.g., structure, behavioral and sometimes temporal) to support dynamic analyses, simulation and some have formal method analysis and automated test generation [11], [45]. Other approaches provide process integration and design optimization framework allowing for many available analysis solvers or custom solvers for all type of analysis with simulation and workflow automation [35].

There are many options available to us; this is not an exhaustive list and the specific modeling language and tool(s) for the Vision model has not yet been decided. Because SysML is general, there are possible mappings to many types of modeling languages (as is true for UML too) [53] as well as support for

---

10 There are a number of useful representations and documentation that are not currently released for public viewing.
programmatic interchange based on the XMI standard. This may rationalize why some organizations are using SysML as an integrating framework, that is, they may not be modeling in SysML, but they are using SysML (and associated tooling) as a mapping or an interchange medium between different modeling languages and environments. While the SysML and UML languages and tools help significantly to formalize the expression, exchange, and graphical representation of system models, they remain ambiguous and in need of extensions to capture the specific semantics of a given engineering domain [49].

Even with the concerns about the understanding of SysML, discussed in Section 2, our team will use modeling notations like SysML in this section of report. However, the perspectives cited in this section reflect on why the Vision must go beyond and use other more semantically rich and precise model representations, as well as supporting semantically consistent model (digital) interchange between different simulation and analysis tools. Our efforts planned for Phase II involve a modest demonstration, and we fully envision that we’ll need to use other modeling approaches that support simulation, analysis, synthesis and generation.

5.4 STRAW MAN

During the kickoff meeting, it was decided that we would attempt to build a model of the Vision. Therefore following good modeling guidelines, we started with a context-level representation that was derived from Integrated Warfighter Capability (IWC) graphic associated with Task 3 shown in Figure 1. The top level IWC is represented using a SysML Block Definition Diagram (BDD) diagram as shown in Figure 23. This provided a way to reflect that the effort involves characterizing all types of information that is necessary to design, verify, validate, produce, acquire and deploy a weapon system. We used other documents describing the Operational Concept Document of Navy Integration and Interoperability (I&I) Environment [39] and created a similar diagram as shown in Figure 24. Regardless of the content and modeling approach (SysML), the mere existence of these examples stimulated significant discussion at the working session and clarified for the team what is meant by modeling the Vision and the concept of capturing all information in “system” model.

![Figure 23. SysML Context of IWC Vision](image)

Conceptually, the Vision model will be a reference architecture (or metamodel) of a multi-level, multi-domain integrated engineering approach to support the IWC. It is not going to describe a specific
instance of a system; instead it will ideally characterize all of the types of information related to the design including characterizations of the supporting environmental resources for simulation and analyses, design parameters and constraints, verification and flight readiness evidence, and associated risk-based signoffs. Ultimately, it should include everything to the Bill of Material (BOM) required to manufacture and produce the system (or in the future the specifications for 3D printing).

It was decided to scope the effort at a Program of Record (POR) (e.g., F18 with weapons, CH-53). Referring to the BDD in Figure 23 and Figure 24, a POR relates to the Integrated Capability Technical Baseline (ICTB) block. The ICTB block is also represented in Figure 23 (the Integrated Warfighter Capability BDD). From the perspective of the I&I Environment BDD, relationships from the ICTB block to the Mission Technical Baseline (essentially where the requirements for the ICTB are derived), and System/Program Technical Baseline blocks are reflected. All of these blocks relate to the I&I Repository. The I&I Repository is part of the Navy’s Integration and Interoperability Integrated Capability Framework vision that includes an enterprise data environment for storing and sharing DoDAF architecture and other supporting I&I data in a common format with common ontologies to support cross-correlation and alignment [39]. These BDDs provide two perspectives on the relationships to the ICTB within the NAVAIR vision, but this is still at a very high level. In order to complete a representation of the Vision it will be necessary to formalize:

- All information as typed data elements, which can be represented as attributes in a model
- Data flows reflecting the data dependencies between blocks
  - BDD diagrams often have associated Internal Block Diagrams (IBD), which show hierarchically lower-level diagrams with the corresponding data flow between the lower-level blocks
  - As another type of example, Figure 25 shows that the Vision must not only be able to characterize the elements of the vehicle system, but should also characterize the elements within the overarching environment that show uses or dependencies to resources such as simulation, test environment, instrumentation and logging
  - Surrogates would also be represented by blocks
- Control flow reflecting both sequential and concurrent flows
  - Activity diagrams in SysML can represent both control flow, and the associated data flows that would be associated with flows within an Internal Block Diagram (IBD)

There are other behavioral views (e.g., sequence and state diagrams) and constraint views (parametrics) that would be necessary to fully characterize the information needed to produce an air vehicle system.
5.5 "As Is" Artifact Analysis

A key guideline for the SE process is the SETR process. An approach that we used in the past when assisting organizations in adopting MBSE is to examine all of the required artifacts that are produced in their current processes, and then examine the ways models and the associated model-based automation (e.g., simulation, analysis, synthesis, generation, test generation, etc.) can be applied to those models to reduce or eliminate “paper-based” documentation and/or modify or eliminate the manual processes. We have heard similar stories from some of the industry organizations in our preliminary discussions about using a combined bottom-up and top-down approach to transform their MBSE efforts. Our NAVAIR team categorized about 330 artifacts and is now realigning its modeling effort. We will report on this in the next bi-monthly status report.
5.6 MODEL-CENTRIC ENGINEERING PERSPECTIVES

Section 2 provides some perspectives on model-centric engineering. This section provides additional information related to discussions or actions from our working sessions that relate to what it means to do model-centric engineering.

5.6.1 MODEL TRANSFORMATION RATHER THAN MODEL EVOLUTION

To reflect on the concept of model transformation rather than model evolution, we provide the following example to describe how model-based automation can completely eliminate manual effort and result in radical transformation of the “As Is” process through an automated workflow. The following provides a scenario for how to think about using models to replace artifacts, and more importantly how model-based automation subsumes manual efforts. This scenario relates to an “As Is” artifact called the “Flight Control Detailed Design Report.” In a model-centric world this type of artifact would:

- Represent “Control Law” in a model
  - Simulink\(^{11}\) and Stateflow are commonly used to model control laws (e.g., F16, F18, F35)
- Automated analysis that exists today, (e.g., it has been applied to F35) would include:
  - Satisfiability: proving that each thread through the model has no contradictions (mathematical consistency)
- Simulation
  - Simulation of Simulink models is often done using Matlab
  - Support high-fidelity simulation using Matlab
  - Support higher fidelity with real-time execution within the surrogate or prototype system implementation or actual hardware through automatic code generation
- Synthesis or generation
  - Code generation from Simulink models can be provided by Mathworks and other commercial products
  - Automatic test generation directly from Simulink models
  - Automatic test driver generation
- The test vectors are transformed into various types of test drivers that are run both against a Matlab simulation and the auto-generated code; if all tests pass (the actual output equals the expected output) in both the simulation and generated code execution environments then there is a strong verification argument that the code satisfies the specification
  - Organizations run the test through both the simulation and code, because organizations have been able to find errors in the code generation (Risk reduction argument for using model-based tools)
- Code coverage tools such as LDRA and VectorCast have been used to show that the tests provide Modified Condition/Decision (MC/DC) coverage
  - Code coverage measurement, which provides quantified risk reduction evidence
- The Mathworks code generation uses a particular algorithm that produces code that is “deadlock” free
  - Eliminates concurrency analysis

These are types of model-based automation that leverage models to “Cross the Virtual V.” While this can be, and is commonly done on low-level high-fidelity models, we are also interested in applying this

\(^{11}\) We are not promoting Simulink, we use it as an example, because it is almost a defacto standard for control system modeling and simulation
type of concept at the upper-levels of the “V” with varying levels of fidelity that provide integration of model and model automation at different levels of the “V.”

This is a positive story as it relates to the use Simulink-based modeling tool chains that can significantly reduce time by both supporting simulation, code generation, analysis and test generation. However, other forms of software modeling have not had this same type of automation, because key information in a modeling framework (e.g., UML Rhapsody) is manually coded, and that cannot be analyzed in the same way that Simulink models. This is a concern as software is growing in complexity and size. This is related to the challenge areas discussed in Section 2.5.1.

5.6.2 CROSSING THE VIRTUAL “V” BY LEVERAGING MODELS, DIGITAL AND PHYSICAL SURROGATES

We have continually discussed the notion of “Crossing the Virtual V” as an important way to assess system design concepts in the context of mission scenarios. However, in discussions with organizations, there are some that believe that the notion of a “V” is a historic manifestation of “the” traditional flow of document-driven work, and we should eliminate the use of the “V” as part of systems engineering dogma as it is counterproductive to embracing approaches that support the continuous integration of digital artifacts. The “V” introduces points for disconnects and failure. What’s more critical in the Vision is continuous integration of various types of digital assets, with varying levels of abstraction and various degrees of fidelity. Section 2.3 provides some discussion on this point using examples to further clarify the notion of physical surrogates, and support the argument that the “V” may not be a good metaphor.

The concept of model-centric engineering relies heavily on digital assets such as physical surrogates, existing system or component re-purposed for new concept exploration and prototyping. Our NAVAIR team created a concept for representing System Level Maturity, as shown in Figure 26. It reflects on the idea that as we are attempting to “Cross the Virtual V” and will rely on physical surrogates, which is commonly done today, both in aerospace and other domains such as auto racing. The actual airframe, shown on the right side of the V matures (top-to-bottom) and the actual aircraft is first flow (e.g., F-35, 15-December-2006) long before many of the complex software intensive systems are developed and integrated, as the aircraft airframe and new materials are being evaluated. Key early capabilities such as software for the control laws to fly the aircraft are often evolved from earlier aircraft systems (e.g., many versions of MATRIXx and/or Simulink models have been evolved for years, and will continue to be evolved for years). Yet, all of these systems are continually refined and as the timeline of system capabilities mature, new capabilities are added to the system. As the subtitle “integrate and build” suggests, continuous integration is the key to this paradigm. Formalized interfaces are required for integration, and the semantics for the interfaces often need to be formalized: 1) structurally, 2) behaviorally, and 3) temporally, in order to use surrogates and simulations. Document-based specifications do not formalize these, however some modeling approaches can, and with semantic formalization, automated verification can be supported directly from the models.
The key issue with the traditional V as shown in Figure 26 is that the V model does imply time progressing chronologically from left-to-right, and while the concept of “Crossing the Virtual V” is notionally easy to understand, and as pointed out by one of our discussion groups, the V model does not work well in a model-based paradigm. Figure 4 presents the same logical progression of the design and verification that is aligned with the traditional lifecycle phases, and progress does correspond with a left-to-right time ordering.

5.6.3 VISION MODEL REFERENCE ARCHITECTURE

We have heard from many different organizations about the use of platform-based designs and reference architectures. As shown in Figure 27, there are many types of major systems elements, and subsystems within an aircraft system. Therefore, we can image that the model of the Vision must need to use a type of reference architecture in the characterization of an integrated set of model types (software, electrical, hydraulic, human interface, etc.) that represent all of the engineering aspects of an aircraft system (i.e., the “total” system model). There must also be ways to characterize different types of elements, for example, a winged aircraft may not have rotors, and a UAV may not have avionics displays.
5.7 **SCOPE TO PROGRAM OF RECORD THROUGH DIGITAL CRITICAL DESIGN REVIEW**

The scope reduced has been reduced to focus on the process up to critical design review (CDR), for the “As Is,” and for the Vision model and risk framework for a POR. It was thought that the technical reviews are good “checkpoints” since they focus on different decisions and levels of engineering content that would need to be represented in the models. Only the PDR and CDR are always required. Other reviews ASR, SRR, SFR, TRR, SVR, PRR may or may not be required on a given program. Ideally, we’re looking for a new concept: Digital design from CDR artifacts (DCDR). We want to investigate a more continuous notion of PDR and CDR (or DCRD) where reviews of models and analysis artifacts occur everyday until all of the required evidence is provided in order to support contracts and signoffs.

5.7.1 **CONTEXT FOR PROGRAM OF RECORD SYSTEM**

The context for the POR starts from environmental aspects at the mission-level as discussed in Section 2.3. For many efforts organizations often start with an OV-1 diagram of the mission-level with systems-of-system (SoS) level interactions. The operational views decompose the mission within context of the situation, and provide different viewpoints that describe the tasks and activities operational elements, and resource flow exchanges required to conduct operations.

Building on lessons learned creating early DoDAF models, analyses have uncovered that interoperating at the lowest (data) levels is insufficient for scenarios, and scenarios require behaviors; missing at the data level. DoDAF does not accommodate other scenario requirements (e.g., conditions assumptions) very well, and is insufficient to fully characterize the dynamic needed for analysis. This is not surprising, as DoDAF has mappings to SysML [53], and we have discussed some of the limitation of SysML in Section 5.1, especially related to dynamics of models.

As a result, study views were created to address a number of challenges at this level. A mission-level SoS analysis begins with formalization through Study Views, as reflected in Figure 28. Study views
provide structure and a common context that acts as a basis for framing and bounding the functional decomposition of DoDAF products. Study views formalize the need and intent, provide a situational context and influencing factors to frame and bound the functions and activities of the mission and scenarios that ultimately lead into corresponding representations of the Mission and System Capabilities (i.e., the capabilities for the POR). These capability representations are further analyzed using modeling and simulation and corresponding analysis capabilities. The outputs of which are then formalized in terms of DoDAF artifacts. This information will form the analysis boundaries for the System Capabilities information that is to be captured in the Vision model.

**Figure 28. Mission Context for System Capability**

### 5.7.2 Case Studies and Existence Cases

We have bounded the scope of this effort. Figure 28 identifies the context level view for the definition of a POR system capability. It does it in the context of a Study View, which is a more abstract view than a DoDAF or SysML characterization. We are still trying to have a discussion (similar to Task 1) with the NAVAIR groups responsible for providing the outputs of the study views into the POR, and across platforms as shown in Figure 5.

We are still in search of a case study. It is desirable to have a case study that can provide technical design, cost, airworthiness and risk data. NAVAIR identified in the final week of this project a possible

---

12 Image source: Thomas Thompson, Enabling Architecture Interoperability Initiative, B210-001D-0051 Unclassified.
candidate, but if that information cannot be provided, we will look to develop our own case study example or transform some existence cases [29] into scenarios.

One possible candidate was developed as part of a Simulation Based Acquisition experiment, where a low-fidelity abstract model of an F-35 slow takeoff vertical landing (STOVL) state machine model was integrated with high-fidelity control laws model [9]. This was a virtual representation of system capabilities, where two different models were used to automatically generate simulation code. The models were analyzed using a formal methods tool (i.e., theorem prover), and test vectors were automatically generated to illustrate both the integration of low and high fidelity models, synthesis, simulation, and model-based verification. No airworthiness requirements were applied to this example.

Another possible candidate was demonstrated by a commercial MBSE tool company that we met with as part of our Task 1 discussions. We have a follow-up discussion planned for 14-August-2014. This model illustrates a UAV capability to support the mission of aerial artifact camera-based examination (e.g., water-based wind turbines), and demonstrates aspects such as: flight control, flight dynamic, propulsion and energy, and mission execution. This model may be supplied under our confidentiality agreement. This model does not address airworthiness, but could be tailored to support this type of required analysis.
6  TASK 4 – INTEGRATED FRAMEWORK FOR RISK IDENTIFICATION AND MANAGEMENT

We want to investigate the use of risk-based models that align with the Vision to “model everything.” This involves how the Vision should include integrated risk identification and management as reflected in Figure 29. While there are many classes of risks to manage, for NAVAIR there are fundamentally two key classes of risk that must be considered:

- Airworthiness and Safety (most critical in Technical Feasibility assessment)
- Program execution (cost, schedule and performance)

There are also two complementary views of model-based acquisition with respect to risk:

- Risks introduced by modeling deficiencies and risks reduced by enhanced use of modeling
- Modeling to predict or assess risks (i.e., modeling for uncertainty quantification in acquisition and in the use of models)

Status: Section 2.7.1 discusses the specific tasks and assignments for developing the “As Is” combined with Airworthiness process, and the Vision models. We envision that our discussion with the organization that has used Quantification of Margins under Uncertainty (QMU), developed at Sandia, will provide additional information to this task.

6.1  PREDICTIVE MODELS FOR RISK

The SERC team has used models in the prediction of risk, and plans to use predictive analytic models to support risk identification and management. More generally we can use models to provide risk quantification for almost all types of decisions that are made by stakeholders (e.g., model-based reviews). As an example, we created a Bayesian model using factors derived from the Airworthiness standard MIL-HDBK-516B [25] as shown in Figure 29. This is conceptually similar to the approach we are using on an FAA NextGen research task for collaborative risk-informed decision-making [6][7][8]. The key characteristics of the approach are they ensure that all factors are considered in the decision-making process, and that all classes of stakeholders are adequately represented in the decision making process. A systematic and comprehensive treatment of all relevant factors provides better risk identification.

We used this model and an example from a true story related to a C130 Weapon Delivery system to illustrate the concept. While this model is notional at this time, this example started a discussion with the team about how stochastic (probabilistic) models can play an important part of the Vision as they formalize many aspects of the human decision making process that will be important at many gates, reviews, and decision points of the Vision concept. Each factor covers a specific aspect of airworthiness, to ensure that all possible uncertainties and risk are considered in the quantification of risk. The risk index is a probability distribution, where for example, the mean can map to quantities in a risk matrix.
Figure 29. Bayesian Model Derived from Airworthiness Factors

6.2 RISK FRAMEWORK APPROACH TO UNCERTAINTY QUANTIFICATION MODELING AND PREDICTION

The SERC team has also been working with NAVSEA to develop a framework and approach to Uncertainty Quantification modeling and prediction. The approach has three main components:

- Identifying the design, test and modeling factors at different system scales
- Analyzing the uncertainty, variability, and error in design implementation, testing, and modeling
- Using experimental design methods to assess the contributions and interactions to system (airworthiness and safety) and program execution risks

The risk modeling and analysis approach also addresses potential errors and uncertainties in the overuse of limited data. Ideally:

- One data set is used to identify critical factors
- A second independent data set is used to develop the models
- A third independent data set is used to calibrate the models
- A fourth independent data set is used to assess the expected error in model results

In practice data sets, surrogate vehicle test data, etc. are limited. Bootstrap methods use repeated resampling of the data and repeating the modeling and analysis process to obtain a statistical estimate of the uncertainty in model-based acquisition given the available data. Further analysis reveals the value – reduction in uncertainty – for additional data.

These types of models capture and embed knowledge associated with expert judgment, historical evidence and rules of thumbs that are used in the decision-making process.
6.3 Scope of the Risk Framework

We worked with our NAVAIR team members to determine the scope for the risk framework. Key to the representation of the models (and Task 3) to support risk identification and management is to characterize the types of evidence that are required for Flight clearance and Flight readiness. It is important to understand how the models are developed and derived in order to understand the risk strategies that must be in place for assessing the evidence for flight clearance.

The process for risk under consideration for this SE transformation covers system development from Milestone A to CDR (at least for now). These questions related to risk also helped to refine the scope for Task 3, and introduced a new term Digital CDR (DCDR), with a heavy emphasis on digitally-derived evidence for airworthiness and safety, but to also include program execution.

In both preliminary discussions with organizations and our NAVAIR team, it is recognized that it is important to quantify “margins” and “sensitivities” as a way to quantify risk.

As an example, one of the organizations (in our preliminary Task 1 discussion) creates new types of advanced material for a system. They cited a particular effort working with advances in new material and processes at the nanoscale. At the component level the margins seemed acceptable. However after composing the components, margins propagated to unacceptable levels in the final integrated form.

Risk implies probabilities of what might go wrong or might not happen (on time or due to the degree expected), and some distribution of magnitude of consequences. This requires “impossible certainty” of the degree of uncertainty and advance knowledge of the likelihood and effects of unidentified events and factors. Therefore, we suggested that a better framework might be to work in terms of design margin. Design margin is more closely related to design. Design margin is how much room there is for a subsystem or component to perform less well than expected or to have greater burdens than expected until it becomes a problem. In some cases, e.g. weight, any increase adds to total weight, so instead of a weight margin, we might want to think in terms of sensitivities (sensitivity in increase in total weight, time, cost, etc. to a percentage increase in the component weight, time, power draw, etc.). This creates a number of questions for this task, for example:

- Can we use models to see how much design margin there is in a system – specifically when we cannot push the system to failure; which types of models and how can we use them to estimate the conditions under which the system begins to exhibit unstable response
  - In control systems analysis this is often taken to be the 3dB point – the frequency of input variation at which the output-to-input ratio is half what it was for low frequency change, or the 90-degree phase-shift point, where the frequency of input variation at which the system response lags by 90 degrees
  - Control systems analysis methods also address the acceleration, velocity and displacement limits at which the system dynamics change
  - Failures are often associated with transitions from linear to highly non-linear regimes; often the structure, interactions and/or dynamics change in these regions (e.g., insulators or isolators fail, etc.) – e.g., the acceleration, velocity and displacement limits at which the system transitions from linear to non-linear response
  - Models that are relevant in the “linear” regime will give erroneous results in the non-linear regime
  - Models that do not represent the dynamics that change the structure of a system (e.g., insulation wearing off causing a short-circuit, structural failure of a linkage, strain transitions from elastic to plastic deformation, etc.) will give erroneous results
Mechanical or electro-mechanical control and isolation systems are good examples, and important for airworthiness. Control systems work within a limited range. Standard control system analysis examines the frequency response and looks for the 3dB frequency, i.e. the frequency at which the transfer function is half of the low-frequency value (the transfer function is just the ratio of output-to-input). Other limits include maximum displacement, velocity and acceleration – when the system hits hard-stops, current limits etc.

Surrogates can be driven with increasing frequency inputs to find the 3dB point without having to experience the failure. The input parameters of virtual models are often “tuned” to match the 3dB point of test data, and then used to extrapolate to find the 3dB point of hypothetical systems. Physically realistic models can be used to estimate the limiting thresholds of stable response, provided the models and inputs are adequately calibrated and validated. Special consideration is needed for basic physical processes with non-linear response in the regime of operation, e.g., friction between moving parts versus friction between stationary parts.

Nested control loop models have been used effectively in system safety modeling and analysis [36]. The outer control loops detect changes in the response behavior of inner control loops, and then adjust the parameters of the inner control loops to bring the inner loops back into the stable regime. The SERC team has used this framework to model driving behavior and system response at the “safety limits.”

In the use of modeling and simulation, there are different types of simulation with different levels of fidelity. A significant challenge is that tools don’t often map well to different levels of abstractions. These are areas to frame risk. There are increasing uses of model transformation from one level or to different disciplines. Model transformation and model consistency between these views becomes a risk issue.

A companion concept is credibility of the estimates of performance, cost, etc. High credibility if it has worked in a surrogate system, less if it is similar to something demonstrated in a surrogate and model extrapolation. It will be important to better understand model extrapolations.

- Less credibility the farther the model extrapolation is extended
- Less credibility going from surrogate system to bench testing, etc.
- Use of multi-scale calibration and validation
- Use of progressive model-based design confirmation in technical reviews
  - Subsystems mature and are integrated at different rates
  - Sometimes early decisions are needed for long-lead time items whose specifications can be confirmed before other aspects of the system (e.g., final control system parameter values)

### 6.4 Risk Framework Captures Knowledge

These types of risk frameworks are actually knowledge models of credibility (not models of performance, but models of uncertainty). Part of the effort on modeling the “As Is” process (Task 3) is to identify and then formalize within the models the information and associated knowledge for evidence-based decisions and evidence-based timing of decisions. Other considerations and opportunities:

- In the “As Is” process, what decisions are artifacts of the process, but not essential to the engineering development?
- Are there lost opportunities by making early concept and design decisions?
- Is there a risk of bad decisions, risks and costs of no or deferred decisions, during planning, or during execution?
• Reconsider the “full system” technical review model. Not all parts of the system are ready for PDR, CDR at the same time. Some are more mature than others. Maybe a granular approach is needed.

The timing of technical reviews and decisions should be made when there is an accumulation of evidence sufficient to make a credible decision. Ideally, this will be inherent in the Vision concept, when the required information and associated analyses are complete, the evidence and timing for decisions should be triggered events in the automated workflow.

6.5 Risk Related Research

SERC research teams are involved in several related research efforts that will be leveraged in the risk framework. We need to explore how the following can be leveraged:

• The High Performance Computing Modernization (HPCM) CREATE program to use high-fidelity models in systems design is establishing a working group on Uncertainty Quantification. SERC partners are collaborating with NAVSEA and the HPCM program.
• The DARPA internet-fabrication (iFab) project sponsored research by a SERC collaborator to develop software to automatically detect and complete gaps in specifications for a “build to” design.
• The US Army TARDEC is developing knowledge models to capture design factors and relationships in system design and development. The resulting decision breakdown structure and process should help distinguish substantive design and engineering decisions versus artifacts of the “As Is” process. SERC partners are coordinating with this effort.
• OSD is sponsoring a SERC project in “risk leading indicators” and “risk estimating relationships,” analyzing the consistency, completeness, and complexity of the system architecture, requirements, task structure, and team organization, and combining those with TRL/IRL levels and Advancement Degree of Difficulty indicators (this project is being conducted in collaboration with TARDEC and an acquisition program).
• The Engineered Resilient Systems (ERS) effort is addressing lost opportunity by making early concept & design decisions, the time and cost to reverse decisions, and tradeoffs between timely but bad decisions versus deferred decisions. SERC partners are collaborating with the NAVSEA ERS and set-based design projects.

6.6 Risk of SE Transformation to MBSE

There is also a concern that the risk of SE transformation to MBSE will fail to provide an efficient, effective and reliable alternative to the current process. There are currently no good ways to assess if a new MBSE approach produces outcomes as good or better than the current process. Regardless, it will be important to establish Measures of Effectiveness (MOEs) or Measures of Performance (MOPs). However, this effort still remains focused on assessing the technical feasibility, and not the adoption of MBSE.
7 CONCLUSION AND NEXT STEPS

This research supports the SERC’s research thrust for SE Transformation, where the goal is to move from engineering approaches for systems designed for optimal performance against a static set of requirements over long procurement cycles to approaches that enhance the productivity of engineers to rapidly develop cost effective, flexible, agile systems that can respond to evolving threats and mission needs.

Key to this effort is an assessment of the “technical feasibility” of a radical transformation enabled by model-centric engineering in order to reduce the time to design and deploy a 5th generation large-scale air vehicle system by 25 percent.

We have conducted over 20 discussions with industry, government, and academic organizations, and have a few remaining. We see a movement towards a more widespread adoption of model-centric engineering, however not a radical transformation as desired of the NAVAIR vision. However, this report sites a number of challenges areas derived from our discussions and other research, for example:

- Our NAVAIR sponsor often mentions in our discussions with organizations that 90 percent of the functionality of in a 5th generation air vehicle system is in software
  - The growth and complexity of software is creating verification challenges
  - The significant amount of software verification, which are essential to airworthiness and safety often has long durations
  - The aspects of software were not originally high on our list, but in model-centric engineering, software connects almost everything, and while the impact of software was not believed to be an issue in the past, it is likely to be going forward
- It was stated in meetings that there is an “explosion of models,” however
  - There is a lack of cross-domain model interoperability, consistency, and limitations in our ability to transform models with the required semantic precision to provide accurate information for decision making
  - Unvalidated models are used leading to incorrect or invalid results and organizations are not identifying design or integration problems until late in lifecycle

We need to understand the conditions associated with these challenges and their impact on the overall objective. We have used a measurement instrument that transforms our subjective beliefs about the factors of model-centric engineering that produces a probabilistic quantity to support the feasibility assessment. The instrument also helps us identify the factors most related to some challenge areas. We will continue to refine the instrument in order to characterize those types of enabling technologies, methods and innovations that result in such a radical transformation.

We have identified a number of existence proofs in example uses of model-centric engineering. We will document these in the Final Technical report. We will develop case studies as we focus on the objective for Tasks 3 and Task 4 in order to develop a modest demonstration. We will continue to develop an approach to map the “As Is” process to the Vision, which should help provide confidence about the completeness of the Vision and ensure that we deal with the critical requirements of safety and airworthiness.

We need to make some decisions about how to represent the Vision model, and how that relates to the mission-level models and analyses characterized as the “containing system” in Section 2.3.1. The representation for the Vision is more of a reference architecture of the possible types of program data that are captured; there will be a need to have dependencies/constraints as different types of data are captured and needed depending for example based on: land-based, sea-based, etc. We will refine the
Vision and link/map to existing documentation, until the attributes associated with the data are formalized in the Vision model. We will use case study data to scope the problem, use and mine historical data from programs of record. This should include how to capture and represent other aspects such as airworthiness constraints, affordability data, earned value data, and risk examples. Many key distinguishing characteristics of the vision will revolve around more integration, interoperability and interaction with the modeling and simulation world. We will represent where and how models are used to produce model artifacts that were typically captured and reviewed as documents.

We are working with our NAVAIR consultants who are characterizing what a “Contractor” could use as model-based requirements and design constraints with respect to this Vision concept. This includes what would be used at a Digital CDR in order to provide the needed information and evidence for the decision gates represented as models or digital artifacts in order to support risk-informed signoff.

We also need to engage in conversations about NAVAIR’s role in a transformed environment to better understand the “designing system,” discussed in Section 2.3.2. As NAVAIR makes this transformation, we should discuss if NAVAIR should focus on a decision framework that pulls or integrates information from contractors and researchers, or consider if they should work towards developing an all-encompassing environment that can allow them to repeat any type of simulation or analysis.

We would like to better understand the inventory of simulation assets used by NAVAIR. We have an upcoming discussion meeting planned for August where we hope to obtain more insights about the information that relates mission analyses to design constraint for a POR (the scope for our task). We are still in need of case study data; we will use case study data to look at the various gates from a historical point of view to understand the data required for decisions represented in the formal of digital evidence. We have one organization that has tentatively offered a demonstration for a UAV that could be used in a modest demonstration.

The specific efforts for Phase II will include:

- Conclude the remainder of the discussions with Government, Industry and Academia, along with refinement of the collection and measurement instruments
- Evolution of the model lexicon, and continuous updates to a website
- Support NPS and MITRE in the development of the “As Is” and mapping to model of the “Vision” with traceability and completeness analysis to the DoD 5000.02
- Characterize a risk management framework that aligns with the “Vision”; this can include examples such as the Airworthiness risk model, risk factor analysis from case study data and/or program execution risk models
- Demonstration thread(s) based on case study or surrogate data; we will either use case study data provided to us, or we will create some type of surrogate case study

Finally, the report discusses implications and alternatives to the transformation for NAVAIR. We have had follow-ups to our meetings on several different topics, and have more planned that are focused on some of the challenge areas. We have been asked to bring industry together to share their perspectives on challenges, issues, concerns, and enablers for a transformation. We want to explore ideas and concepts to improve the efficiencies and break down the barriers that inhibit speeding the needed capabilities to the NAVAIR and the warfighter.
Factor Definitions

The following is the current set (fifth version) of the set of factors associated with the discussion measurement instrument (see Section 2). As the discussions with organizations are held, these factors and the associated categories will be refined.

Table 3. Discussion Instrument Factor Definition

<table>
<thead>
<tr>
<th>Factor Category</th>
<th>Factors</th>
<th>General These factors relate to the degree to which advance MBSE provides a holistic approach to SE</th>
<th>Commentary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Magnitude of applicability over the lifecycle</td>
<td>Organizational Scope</td>
<td>What is the scope of the MBSE usage? Normally, when thinking about NAVAIR systems the scope is quite large and involves large programs. Therefore, what organizational scope does the MBSE usages apply: Program, Project, an entire Business Unit, Platform (e.g., a type of a specific aircraft, tank, automobile), Department, or Site.</td>
<td>Key to all of these questions is that we are looking for &quot;Technical Feasibility of &quot;doing everything with model.&quot; We recognize that actual adoption can be difficult, and might not make sense on older systems. Therefore related to this, it is probably best to ensure that the question perspectives come from a Chief Engineer, Chief Technical Offer, MBSE Organizational Specialist and possibly the Program Manager. To carry this a step further, it might also be important to keep the &quot;systems&quot; perspective in mind, because some of the concepts discussed may have been applied in Hardware and possibly in Software (e.g., the control laws for the F35 are built in Simulink, with auto code generation, and a significant portion of auto test generation), but not completely at the Systems level.</td>
</tr>
<tr>
<td>Scope Impact</td>
<td>How broadly does the answers cover the entire lifecycle (for example, a university research project might be very advanced in terms of analysis or simulation, but it does not cover the entire DoD 5000 lifecycle).</td>
<td>The answer to this question has a lot of weight, because we need to consider answer in context of lifecycle applicable to NAVAIR (and in general DoD Acq. Programs).</td>
<td></td>
</tr>
<tr>
<td>Proven beyond a research concept</td>
<td>Demonstrations</td>
<td>Are the capabilities discussed actually in operations - have they been demonstrated?</td>
<td>We want to understand that things discussed are more than just research concepts.</td>
</tr>
<tr>
<td>Crossing the Virtual V</td>
<td>Integrated Simulation</td>
<td>In order to Cross the Virtual V, there will be many types of modeling and simulation required to support various type of domains within the system. To what degree are the simulations integrated, and better yet do different simulations work off of shared models?</td>
<td>In order to &quot;cross the virtual V&quot; during the early stages of development, it is important to understand if the inputs/outputs from one set of simulations can feed another (e.g., to be able to understand the capability in the mission context)</td>
</tr>
<tr>
<td></td>
<td>Formal Analysis</td>
<td>Are the analyses (e.g., property analysis) formal, meaning that they are performed on models automatically?</td>
<td>Is the analysis fully automated from the models (H) or is there human interpretation required (M or L) or none (L)?</td>
</tr>
<tr>
<td>Domain Specific</td>
<td>Are the different types of models related to the domain? For example, control system engineers often use Simulink/Matlab. Also, most modeling and simulation environments are domain-specific.</td>
<td>Domain-specific modeling languages are an emerging trend; these types of approaches provide intuitive abstractions (often graphical) that are familiar to engineers within the domain. Rather, SysML, while good for systems engineers, it might not be applicable to flight controls, networks, fluid dynamics, etc. In addition, there is not significant support for automated V&amp;V from SysML as the semantics are not very rich.</td>
<td></td>
</tr>
<tr>
<td>----------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>Cross Domain Coverage</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Domain Interoperability</td>
<td>Are the models that are in different, but related domains integrated? Are the models consistent across the domains?</td>
<td>For example, are the models that are used for performance the same models used for integrity/dependability analysis?</td>
<td></td>
</tr>
<tr>
<td>Synthesis/Generation</td>
<td>Can the models be used for synthesis/generation of other related artifacts such as code, simulation, analysis, tests and documentation</td>
<td>We know that one type of modeling is not always appropriate for everything, and that is why there is emergence of Domain-Specific Modeling languages; the key question is: are the models for one use consistent for other users (e.g., performance, integrity).</td>
<td></td>
</tr>
<tr>
<td>Meta-Model/Model Transformations</td>
<td>Are the models used in one domain, or for one purpose, transformable into another domain where the well-defined semantics in one domain is carried through the transformation into the other domain; if so are they known to be consistent?</td>
<td>Example, Formula 1 racing, uses data logging during physical experimentation and then factors results (and logs) back into simulation environment; can we fly some new capability on an existing aircraft and then factor the results from the test flights back into the modeling and simulation environments? This in the future should allow more virtual flight testing (once the approach becomes trusted).</td>
<td></td>
</tr>
<tr>
<td>Surrogate Integration</td>
<td>Are surrogates used to support analysis, and are the results of the surrogates captured so that they can be factored into modeling and simulation in the future?</td>
<td>Are the abstractions from the models still &quot;rich enough&quot; to be representative of the actual mission environment when used in a virtual environment? For example, if we use a 3D immersive environment, can we understand the physical characteristic of the operational system?</td>
<td></td>
</tr>
<tr>
<td>Formal Capability Assessment</td>
<td>How well do the models, simulations and analyses capabilities support the ability to understand the capabilities being developed?</td>
<td>As an example, margin tolerances at the component level can propagate as the system is composed (or assembled). Are these factors understood and controlled?</td>
<td></td>
</tr>
<tr>
<td>Virtual Accuracy/Margin Analysis</td>
<td>Are the modeling, simulation and analysis accurate? How well do they allow the designers to understand the margins?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3D Immersive Environments</td>
<td>What is the degree to which 3D Immersive Environments are used to improve the understanding (and possibly training) of the virtual systems.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| Management Criticality Risks (Relates to Task 4) | Risk Management | Are there proper risk management identification, analysis and mitigations applied based on the use of models? | This should also consider:  
- Adequately deal with critical timelines  
- Integrated operational risk  
- Change management (model-based change management is different than document-based) |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Predictive Analytics</td>
<td>Are there models used to support a quantitative approach to risk management?</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| Attributes of Modeling Maturity | Model-based metrics | Are there model-based metrics (or a comprehensive set of model measurements) and are they used to support the management of programs/projects? | The use of model-based metrics reflects on the modeling maturing of the organization.  
- Dealing with interdependencies and modeling consistency often deals with having a detailed understanding of the semantics across models. Positive results for this answer suggest a very advanced use of models. |
| | Multi-model interdependencies / consistency and semantic precision | If the organization is dealing with many different types of models, are the interdependencies managed and are the models semantically precise enough to manage model consistency? |
| | High Performance Computing (HPC) | Is HPC applied to the modeling, simulation and analysis efforts? | Use of HPC is an indicator of high modeling maturity. |
| Operational Risks (Relates to Task 4) | Procedures | Are the procedures for using the models understood, so that we can trust the model outputs to support other related types of analysis, both in terms of technical as well as risk? | This applies heavily in airworthiness (e.g., Mil Std. 516)  
- This is another indicator of a more advanced organization. As a side effect the use of 3D Immersive systems can be valuable in both collaboration and early training. |
| | Staff and Training | With the advances in the technologies associated with models, are the staff and training in place to support the use of models? |
| Human Factors | How well are human factors supported by the modeling, simulation and analysis capabilities?  
This should consider Usability. |
| Indirect support from Models | Certification | How well do the models-based automation and practices support certifications (if required)? | If not applicable use M - for Medium |
| | Regulation | How well do the models-based automation and practices support regulations (if required)? | If not applicable use M - for Medium |
| | Modeling and Simulation Qualification | How much do we trust our models? |
# Acronyms and Abbreviation

This section provides a list of some of the terms used throughout the paper. The model lexicon should have all of these terms and many others.

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AADL</td>
<td>Architecture Analysis &amp; Design Language</td>
</tr>
<tr>
<td>ACAT</td>
<td>Acquisition Category</td>
</tr>
<tr>
<td>AGM</td>
<td>Acquisition Guidance Model</td>
</tr>
<tr>
<td>AP233</td>
<td>Application Protocol 233</td>
</tr>
<tr>
<td>ATL</td>
<td>ATLAS Transformation Language</td>
</tr>
<tr>
<td>ASR</td>
<td>Alternative System Review</td>
</tr>
<tr>
<td>BDD</td>
<td>SysML Block Definition Diagram</td>
</tr>
<tr>
<td>BNF</td>
<td>Backus Naur Form</td>
</tr>
<tr>
<td>BPML</td>
<td>Business Process Modeling Language</td>
</tr>
<tr>
<td>CAD</td>
<td>Computer-Aided Design</td>
</tr>
<tr>
<td>CASE</td>
<td>Computer-Aided Software Engineering</td>
</tr>
<tr>
<td>CDR</td>
<td>Critical Design Review</td>
</tr>
<tr>
<td>CMM</td>
<td>Capability Maturity Model</td>
</tr>
<tr>
<td>CMMI</td>
<td>Capability Maturity Model Integration</td>
</tr>
<tr>
<td>CORBA</td>
<td>Common Object Requesting Broker Architecture</td>
</tr>
<tr>
<td>CREATE</td>
<td>Computational Research and Engineering for Acquisition Tools and Environments</td>
</tr>
<tr>
<td>CWM</td>
<td>Common Warehouse Metamodel</td>
</tr>
<tr>
<td>dB</td>
<td>Decibel</td>
</tr>
<tr>
<td>DBMS</td>
<td>Database Management System</td>
</tr>
<tr>
<td>DAG</td>
<td>Defense Acquisition Guidebook</td>
</tr>
<tr>
<td>DARPA</td>
<td>Defense Advanced Research Project Agency</td>
</tr>
<tr>
<td>DAU</td>
<td>Defense Acquisition University</td>
</tr>
<tr>
<td>DoD</td>
<td>Department of Defense</td>
</tr>
<tr>
<td>DoDAF</td>
<td>Department of Defense Architectural Framework</td>
</tr>
<tr>
<td>DSL</td>
<td>Domain Specific Languages</td>
</tr>
<tr>
<td>DSM</td>
<td>Domain Specific Modeling</td>
</tr>
<tr>
<td>DSML</td>
<td>Domain Specific Modeling Language</td>
</tr>
<tr>
<td>E/DRAP</td>
<td>Engineering Data Requirements Agreement Plan</td>
</tr>
<tr>
<td>ERS</td>
<td>Engineered Resilient Systems</td>
</tr>
<tr>
<td>HPCM</td>
<td>High Performance Computing Modernization</td>
</tr>
<tr>
<td>IBM</td>
<td>International Business Machines</td>
</tr>
<tr>
<td>IBD</td>
<td>SysML Internal Block Diagram</td>
</tr>
<tr>
<td>ICD</td>
<td>Interface Control Document</td>
</tr>
<tr>
<td>INCOSE</td>
<td>International Council on Systems Engineering</td>
</tr>
<tr>
<td>IPR</td>
<td>Integration Problem Report</td>
</tr>
<tr>
<td>IRL</td>
<td>Integration Readiness Level</td>
</tr>
<tr>
<td>ISO</td>
<td>International Organization for Standardization</td>
</tr>
<tr>
<td>IT</td>
<td>Information Technology</td>
</tr>
<tr>
<td>Linux</td>
<td>An operating system created by Linus Torvalds</td>
</tr>
<tr>
<td>MARTE</td>
<td>Modeling and Analysis of Real Time Embedded systems</td>
</tr>
<tr>
<td>MATRIXx</td>
<td>Product family for model-based control system design produced by National Instruments</td>
</tr>
<tr>
<td>MBT</td>
<td>Model Based Testing</td>
</tr>
<tr>
<td>MDA®</td>
<td>Model Driven Architecture®</td>
</tr>
<tr>
<td>Acronym</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
</tr>
<tr>
<td>MDD™</td>
<td>Model Driven Development</td>
</tr>
<tr>
<td>MDE</td>
<td>Model Driven Engineering</td>
</tr>
<tr>
<td>MDSD</td>
<td>Model Driven Software Development</td>
</tr>
<tr>
<td>MDSE</td>
<td>Model Driven Software Engineering</td>
</tr>
<tr>
<td>MIC</td>
<td>Model Integrated Computing</td>
</tr>
<tr>
<td>MMMM</td>
<td>Modeling Maturity Model</td>
</tr>
<tr>
<td>MoDAF</td>
<td>United Kingdom Ministry of Defence Architectural Framework</td>
</tr>
<tr>
<td>MOE</td>
<td>Measure of Effectiveness</td>
</tr>
<tr>
<td>MOF</td>
<td>Meta Object Facility</td>
</tr>
<tr>
<td>MOP</td>
<td>Measure of Performance</td>
</tr>
<tr>
<td>MVS</td>
<td>Multiple Virtual Storage</td>
</tr>
<tr>
<td>NASA</td>
<td>National Aeronautics and Space Administration</td>
</tr>
<tr>
<td>NAVAIR</td>
<td>U.S. Navy Naval Air Systems Command</td>
</tr>
<tr>
<td>NAVSEA</td>
<td>U.S. Naval Sea Systems Command</td>
</tr>
<tr>
<td>OCL</td>
<td>Object Constraint Language</td>
</tr>
<tr>
<td>OMG</td>
<td>Object Management Group</td>
</tr>
<tr>
<td>OO</td>
<td>Object oriented</td>
</tr>
<tr>
<td>OSD</td>
<td>Office of the Secretary of Defense</td>
</tr>
<tr>
<td>PDM</td>
<td>Product Data Management</td>
</tr>
<tr>
<td>PDR</td>
<td>Preliminary Design Review</td>
</tr>
<tr>
<td>PIM</td>
<td>Platform Independent Model</td>
</tr>
<tr>
<td>PLM</td>
<td>Product Lifecycle Management</td>
</tr>
<tr>
<td>POR</td>
<td>Program of Record</td>
</tr>
<tr>
<td>PRR</td>
<td>Production Readiness Review</td>
</tr>
<tr>
<td>PSM</td>
<td>Platform Specific Model</td>
</tr>
<tr>
<td>RFP</td>
<td>Request for Proposal</td>
</tr>
<tr>
<td>ROI</td>
<td>Return On Investment</td>
</tr>
<tr>
<td>SE</td>
<td>System Engineering</td>
</tr>
<tr>
<td>SERC</td>
<td>Systems Engineering Research Center</td>
</tr>
<tr>
<td>SETR</td>
<td>System Engineering Technical Review</td>
</tr>
<tr>
<td>Simulink/Stateflow</td>
<td>Product family for model-based control system produced by The Mathworks</td>
</tr>
<tr>
<td>SCR</td>
<td>Software Cost Reduction</td>
</tr>
<tr>
<td>SDD</td>
<td>Software Design Document</td>
</tr>
<tr>
<td>SFR</td>
<td>System Functional Review</td>
</tr>
<tr>
<td>SOAP</td>
<td>A protocol for exchanging XML-based messages – originally stood for Simple Object Access Protocol</td>
</tr>
<tr>
<td>Software Factory</td>
<td>Term used by Microsoft</td>
</tr>
<tr>
<td>SRR</td>
<td>System Requirements Review</td>
</tr>
<tr>
<td>SRS</td>
<td>Software Requirement Specification</td>
</tr>
<tr>
<td>STOVL</td>
<td>Short takeoff and vertical landing</td>
</tr>
<tr>
<td>SVR</td>
<td>System Verification Review</td>
</tr>
<tr>
<td>SysML</td>
<td>System Modeling Language</td>
</tr>
<tr>
<td>TARDEC</td>
<td>US Army Tank Automotive Research</td>
</tr>
<tr>
<td>TRL</td>
<td>Technology Readiness Level</td>
</tr>
<tr>
<td>TRR</td>
<td>Test Readiness Review</td>
</tr>
<tr>
<td>UML</td>
<td>Unified Modeling Language</td>
</tr>
<tr>
<td>XMI</td>
<td>XML Metadata Interchange</td>
</tr>
<tr>
<td>XML</td>
<td>eXtensible Markup Language</td>
</tr>
<tr>
<td>US</td>
<td>United States</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Full Form</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------</td>
</tr>
<tr>
<td>XSLT</td>
<td>eXtensible Stylesheet Language family (XSL) Transformation</td>
</tr>
<tr>
<td>xUML</td>
<td>Executable UML</td>
</tr>
<tr>
<td>Unix</td>
<td>An operating system with trademark held by the Open Group</td>
</tr>
<tr>
<td>VHDL</td>
<td>Verilog Hardware Description Language</td>
</tr>
<tr>
<td>VxWorks</td>
<td>Operating system designed for embedded systems and owned by WindRiver</td>
</tr>
</tbody>
</table>
Trademarks

Analysis Server is a registered trademark of Phoenix Integration, Inc.
Astah SysML is Copyright of Change Vision, Inc.
BridgePoint is a registered trademark of Mentor Graphics.
Cameo Simulation Toolkit is a registered trademark of No Magic, Inc.
CORE is a registered trademark of Vitech Corporation.
IBM™ is a trademark of the IBM Corporation
iGrafx is a registered trademark of iGrafx, LCC.
Java™ and J2EE™ are trademark of SUN Microsystems
Java is trademarked by Sun Microsystems, Inc.
LDRA is a registered trademark of Trademark of LDRA Ltd. and Subsidiaries.
Linux is a registered trademark of Linux Mark Institute.
Mathworks, Simulink, and Stateflow are registered trademarks of The Mathworks, Inc.
MagicDraw is a trademark of No Magic, Inc.
MATRIxx is a registered trademark of National Instruments.
Object Management Group (OMG): OMG's Registered Trademarks include: MDA®, Model Driven Architecture®, UML®, CORBA®, CORBA Academy®, XML®
OMG's Trademarks include, CWM™, Model Based Application Development™, MDD™, Model Based Development™, Model Based Management™, Model Based Programming™, Model Driven Application Development™, Model Driven Development™
ModelCenter, is a registered trademark of Phoenix Integration, Inc.
Model Driven Programming™, Model Driven Systems™, OMG Interface Definition Language (IDL)™,
Unified Modeling Language™, <<UML>>™
OMG®, MDA®, UML®, MOF®, XML®, SysML™, BPML™ are registered trademarks or trademarks of the Object Management Group.
ParaMagic is a registered trademark of InterCAX, Inc.
PHX ModelCenter is a registered trademark of Phoenix Integration, Inc.
PowerPoint is a registered trademark of Microsoft, Inc.
Real-time Studio Professional is a registered trademark of ARTISAN Software Tools, Inc.
Rhapsody is a registered trademark of Telelogic/IBM.
Rose XDE is a registered trademark of IBM.
SCADE is copyrighted to Esterel Technologies.
Simulink is a registered trademark of The MathWorks.
Stateflow is a registered trademark of The MathWorks.
Statemate is a registered trademark of Telelogic/IBM.
STK is a registered trademark of Analytical Graphics, Incorporated (AGI), Inc.
UNIX is a registered trademark of The Open Group.
VAPS is registered at eNGENUITY Technologies.
VectorCAST is a registered trademark of Vector Software, Inc.
Visio is a registered trademark of Microsoft, Inc.
VxWorks is a registered trademark of Wind River Systems, Inc.
Windows is a registered trademark of Microsoft Corporation in the United States and other countries.
XML™ is a trademark of W3C
All other trademarks belong to their respective organizations.
References


[21] DARPA, Producible Adaptive Model-based Software (PAMS) technology to the development of safety critical flight control software. PAMS has been developed under the Defense Advanced Research Projects


