# Facilitating the Transition to Model-Based Acquisition

**Marlin Ballard**
**Georgia Institute of Technology**
**270 Ferst Drive NW**
**Atlanta, GA 30332-0150**
**Marlin.B.Ballard@gatech.edu**

**Adam Baker**
**Georgia Institute of Technology**
**270 Ferst Drive NW**
**Atlanta, GA 30332-0150**
**a-baker@gatech.edu**

**Russell Peak**
**Georgia Institute of Technology**
**270 Ferst Drive NW**
**Atlanta, GA 30332-0150**
**Russell.Peak@gatech.edu**

**Selcuk Cimtalay**
**Georgia Institute of Technology**
**270 Ferst Drive NW**
**Atlanta, GA 30332-0150**
**cimtalay@gatech.edu**

**Mark Blackburn**
**Stevens Institute of Technology**
**Castle Point on Hudson**
**Hoboken, NJ 07030**
**mark.blackburn@stevens.edu**

**Dimitri Mavris**
**Georgia Institute of Technology**
**270 Ferst Drive NW**
**Atlanta, GA 30332-0150**
**Dimitri.Mavris@aerospace.gatech.edu**

*Abstract*—**One major benefit offered by MBSE is the ability to formalize interactions between subsystems in the design process. This formalization eases the transfer of information between parties. The process of government acquisition is likewise characterized by information transfer: diverse requirements must be altered and tracked between the requesting, responding, and evaluating parties. Thus, it is a natural extension of MBSE is to apply it to the acquisition process.**

**This paper demonstrates a set of tools and patterns developed during a surrogate simulation of an MBSE-enabled Request for Proposal between NAVAIR and a responding contractor. In particular, the tools presented were developed from the NAVAIR Systems Model viewpoint. This paper covers four tools developed in this surrogate pilot. The first analyzes the problem of requirement generation. While standards such as the OMG SysML are being adopted by MBSE practitioners, the model literacy of all stakeholders is unlikely and may never be fully guaranteed. Document generation tools, such as OpenMBEE have been developed for SysML software, which enable presentation of descriptive information about the model. This paper demonstrates modeling patterns and a tool that translates information from native-model form into a text-based format. The second and third tools presented assist in the acquirer's source selection process. Making use of the patterns which generate the text requirements above, Evaluation and Estimation Models are presented, which can act directly on contractors' responses. The Evaluation Model assists the verification process by ensuring numerical requirements are satisfied. The Estimation Model compares the contractors' claimed values with historically expected values, to assist directing the source selection experts' focus of examination. The fourth tool presented offers a method of extracting historical traceability for model elements. This aids the acquisition process by enabling digital signoff at any stage of the acquisition process. These four tools were applied in the surrogate acquisition process for a notional UAV, and a description of this case study is presented.**

978-1-7821-2734-7/20/$31.00 ©2020 IEEE

## TABLE OF CONTENTS

## 1. INTRODUCTION

The US government acquisition process presents a unique challenge characterized by large amounts of requirements and data that must be tracked and transferred between multiple parties. This information must be maintained throughout the many stages of the acquisition process – spanning from initial contractor and design selection up to final manufacturing. This presents an ideal application for MBSE as a solution to guarantee the integrity of data and formalize the transfer of information between all stakeholders. This paper outlines key techniques and best practices to enable the transition to a model-based acquisition process and exploit the full advantages offered by this transition.

These techniques were developed as a part of a project which simulated the typical NAVAIR acquisition process as it might function after the adoption of a proposed model-based approach. The first stage of this project involved simulating the Request for Proposal (RFP) process between NAVAIR and potential contractors, for the acquisition of a hypothetical search-and-rescue unmanned aerial vehicle (UAV) named Skyzer. The authors were principally concerned with the development of the system model from the view of the acquirer (which is NAVAIR in this surrogate pilot). As such, the contents of this paper directly target techniques useful for the contractor selection process, but many of the modeling patterns outlined are applicable to all stages of the acquisition process.

Extracting information from models for review by stakeholders was deemed a crucial enabling technology for the transition to an MBSE environment. Models of complex systems can grow large and labyrinthine even when following best modeling practices. Compounding this problem is the likelihood that the envisioned model-based acquisition process will incorporate models from various groups that potentially practice different MBSE methodologies. This paper outlines several methods for extracting relevant information from these disparate models and compiling it into a single document. This not only enables a more efficient review of key model information but also allows stakeholders with no modeling background to stay abreast with developments.

*Paper Organization*

Section 2 of this paper offers a brief introduction to the architecture used in the Skyzer acquisition project. Sections 3-6 each describe one of the tools developed to assist the system model component of the architecture. Section 7 offers concluding remarks.

## 2. MODEL-BASED ACQUISITION PROCESS

The structure of the model-based acquisition process for the Skyzer surrogate pilot project was chosen to reflect various stages of information safeguarding in the acquisition process. To demonstrate MBSE's ability to enable traceability, information was segmented into several models, which individually contain different access rules (Figure 1). The architecture of the models developed for the surrogate acquisition program is discussed in detail in [1]. One fundamental result of this separation is the creation of a federated authoritative source of truth, where differing agents can maintain control of their own knowledge as the design of the UAV system develops. This concept is developed in detail in [2]. The remainder of this section offers a brief description of the components most relevant to the System

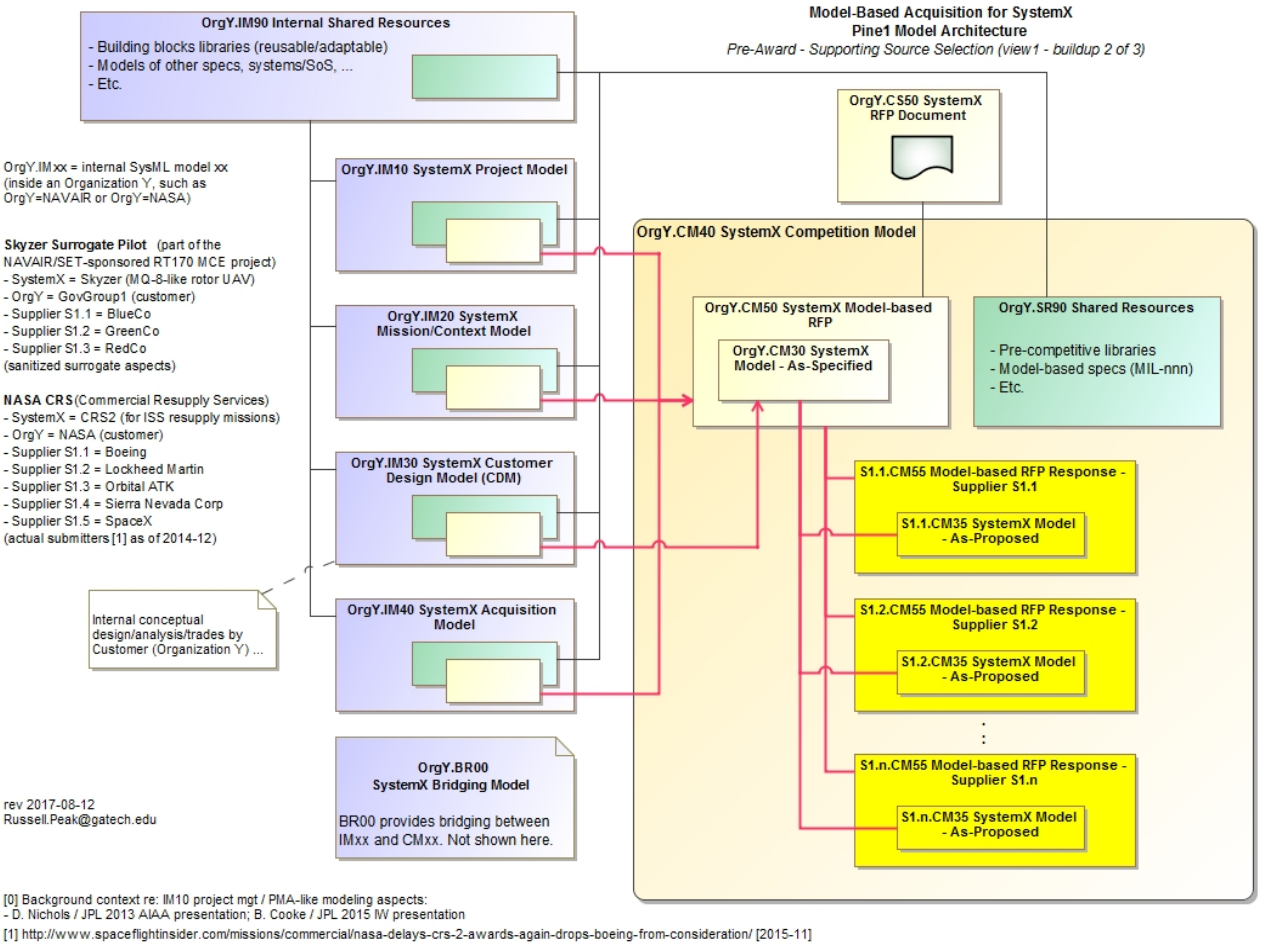


**Figure 1: Skyzer Surrogate Model-Based Acquisition Process (Model Architecture View)**

Model component. To streamline deployment across the surrogate acquisition process, the individual organizations involved have each maintained a coordinated installation of MagicDraw software components as the SysML authoring tool which interfaces to a single Teamwork Cloud repository installation [2].

*Mission Model*

The Mission Model is an acquirer-internal model which details the operational environment of the system-of-interest. This describes the black-box behavior of the system-of-interest as it interacts with other systems in the environment. In the case of the Skyzer UAV system, the mission model defined interactions between the UAV system, its operators, and naval vessels. The Mission Model also houses the performance requirements and capability requirements for the UAV system, represented as SysML requirement elements. The Mission Model itself – or at least significant portions of the Mission Model – could be potentially utilized in other acquisition programs for other systems, for UAV mission elements as well as non-UAV mission elements. Thus it is important to manage the mission model separately from the Skyzer system model, noting that the Skyzer system is just one component at the mission level. The acquirer will typically hold back perhaps most of the Mission Model specifics, for example, for internal sensitive information reasons and/or for other reasons internal to the acquirer.

*System Model*

The System Model is a representation if the system to be acquired (in this case, the Skyzer UAV). At the RFP phase of the acquisition process, knowledge of the system design is typically sparse. This is reflected in the System Model, which represents the internal structure of the system-of- interest only to an extent that it is required to specify the requirements from the Mission Model – at least the System Model portions that are released for the RFP. The System Model imports the Mission Model to expose its requirements using the SysML Project Usage mechanism. A subset of the System Model is provided to responding contractors as Government Furnished Information (GFI) with the expectation that a contractor response extends the System Model in a model-based fashion. Similar to the Mission Model case, the acquirer will typically hold back some System Model specifics. For example, specific information may be withheld so as not to overly influence the proposers' solutions, for internal sensitive information reasons, and/or for other reasons internal to the acquirer.

*RFP Document*

The resulting Request for Proposal (RFP) model has an associated document view in this architecture, which is a web-based document displayed using the OpenMBEE View Editor[1] software. It exposes views that are subsets of multiple models in the architecture, including the Mission Model, System Model, and Statement of Work Model. Each component model that contributes to the text specification includes a section for how to read the model-based information provided.

## 3. MODEL-BACKED TEXT REQUIREMENT GENERATION

One often-cited challenge for the adoption of MBSE-based methodologies is the expectation of upfront training required for engineers to understand system modeling as a practice. This remains a factor for consideration in the Skyzer acquisition research project. While an end state vision would include providing a system model as a first-class acquisition item in an RFP, this research provides a more explicitly transitional approach. As described in the architecture section, the system model is provided to contractors as Government Furnished Information (GFI) supplementing the formal RFP document. The RFP document itself takes the form of a standard text document, readable by those without any specific MBSE training. This improves readability for both parties: (1) contractors are not required to understand a particular methodology to interpret the needs of the request, and (2) the requestor is able to perform internal reviews on the contents of the request with internal domain experts who understand the problem the RFP solves, but who may not necessarily understand all the nuances of the SysML model itself.

The requirements in the Skyzer RFP document use standard "shall statement" language. Nevertheless, those text-based requirements are backed by information in the acquisition system model. This is accomplished through a model-to-text requirements transformation tool. This tool interrogates the system model in order to extract the needed information to fill text templates representing several types of text-based requirements. The model-to-text requirement transformation tool employs the OpenMBEE view and viewpoint structure to enable users to expose the appropriate model elements structurally. When the tool is executed, the viewpoint method invokes a script that traverses the model using the MagicDraw OpenAPI methods to obtain the appropriate information. That information is then used to fill string templates for the various types of requirements necessary. In the case of Performance Requirements, features identified as Key Performance Parameters (KPPs) are represented as SysML value properties in the acquisition system model. These value properties are referred to in the RFP statement of work, with the expectation that proposers' response models will extend and fill the features with values. The defaultValue property and unit for each of those KPP value properties are used to fill the appropriate string template to form each associated text-based requirement found in the RFP document.

A set of requirements from a prior traditional RFP for a Vertical Takeoff UAV was analyzed to identify several patterns of requirements which should be represented in the Skyzer RFP document. This analysis identified a pattern

---

[1] www.openmbee.org

whereby certain Functional Requirements could be represented by a state machine, seen in Figure 2. This state machine was made available in the GFI portion of the acquisition system model, with instructions on how responses would demonstrate the appropriate behavior. To reproduce the corresponding text-based requirements in the main RFP document, the model-to-text based requirement tool was extended to consume state machine elements exposed to it and generate corresponding text-based requirements. The structure of the exposed SysML State Machine element was inspected to identify states, state transitions, and substate machines' states, which comprised the requirement templates identified in the prior vehicle RFP.

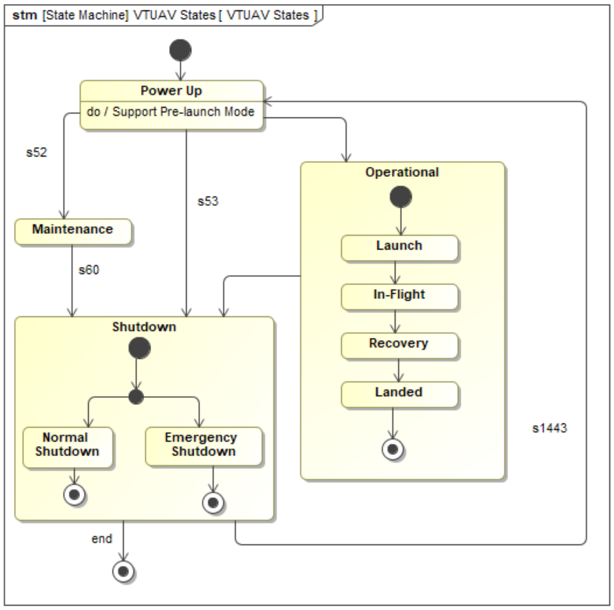


**Figure 2. SysML State Machine Used to Generate Text-Based Requirements (TBRs)**

Samples for the requirement text produced by exposing the state machine in Figure 2 can be found in Table 1. Note that only the top-level state machine element needs to be exposed. The system name is obtained by querying which system component uses the state machine as a classifier behavior. The state machine can also contain information beyond what is exposed as text-based requirements, e.g. the state transition behavior of the "Operational" and "Shutdown" composite states. Defining this type of pattern enables system modelers to produce a detailed specification model which can be automatically transformed into text-based requirements for non–MBSE trained stakeholders. The Skyzer project is applying portions of the TBR/MBR research from [3] in its model-to-text generation direction..

## 4. EVALUATION MODEL

The primary goal of the Evaluation Model is to ensure the visibility and verification of key performance parameters (KPPs) for each model-based contractor response to an RFP. Government acquisition processes are generally characterized by extensive lists of requirements that must be tracked and by which proposed solutions are evaluated. The generalized system model provided to contractors will include all KPPs and other characteristics that correspond to these requirements. As the system model developed by the contractor extends the generalized systems model, the response automatically includes these values via SysML inheritance, and will be evaluated based on them. Requirements can come in many forms. This Evaluation Model focuses primarily on numerical performance requirements that can be compared directly to minimum baseline or target values. However, the general form of the Evaluation Model can be leveraged to evaluate categorical or functional requirements on a pass/fail basis as well.

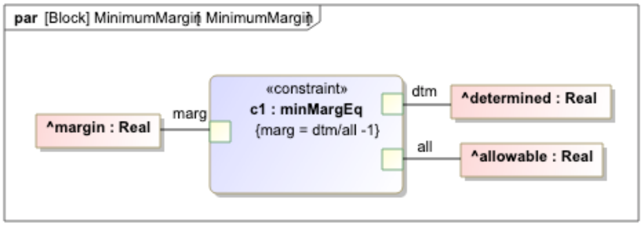


**Figure 3. Margin Block Design Pattern**

The Evaluation Model uses a construct of Margin Blocks to evaluate how well a contractor response satisfies each of the KPP requirements. The Margin Block design pattern is shown in Figure 3 as a SysML parametric diagram. A Margin Block instance specification is created for each KPP

**Table 1. Sample Text-Based Requirements and Corresponding State Machine Facet**

| Facet | Example from Figure 2 | Requirement Text |
|---|---|---|
| State Structure | [Power Up] | "The VTUAV system shall have a Power Up state." |
| State Transition | [s60] | "The VTUAV system shall allow transition from the Maintenance state to the Shutdown state." |
| Composite State Structure | [Recovery] in [Operational] | "The VTUAV system shall have a Recovery mode in the Operational state." |

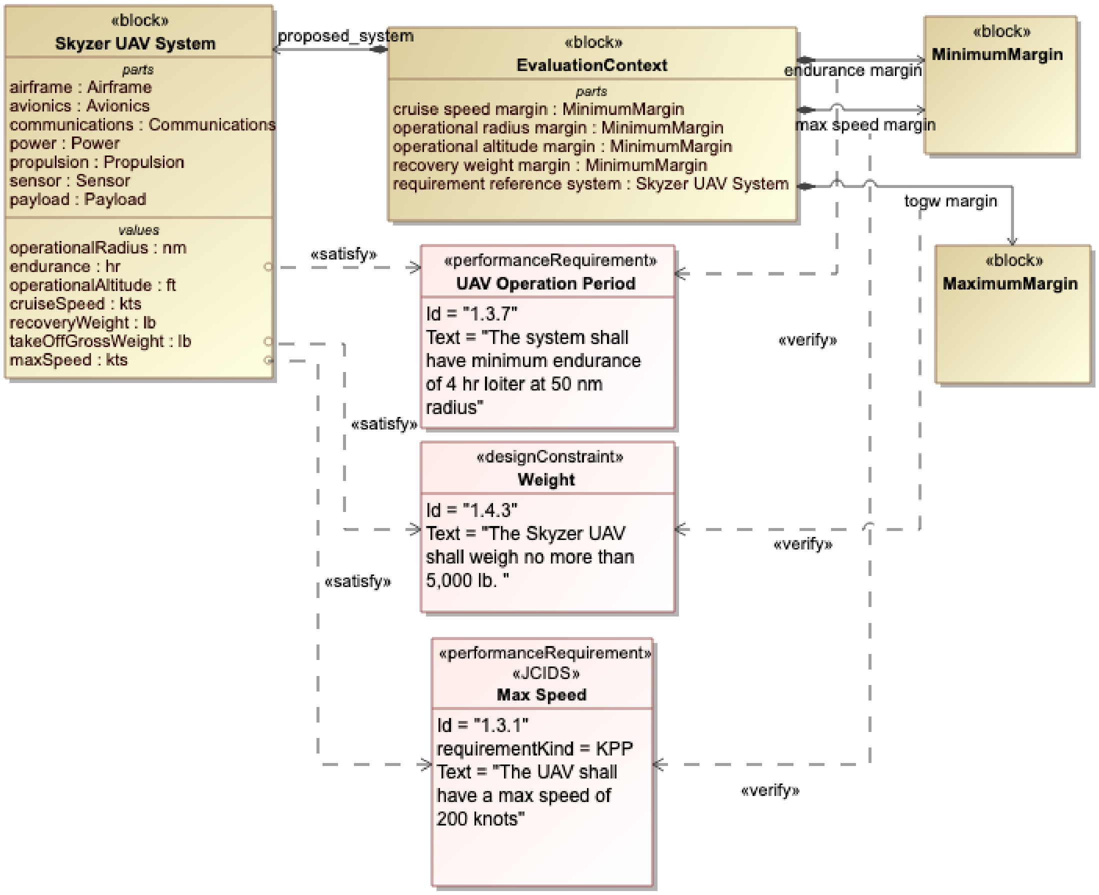
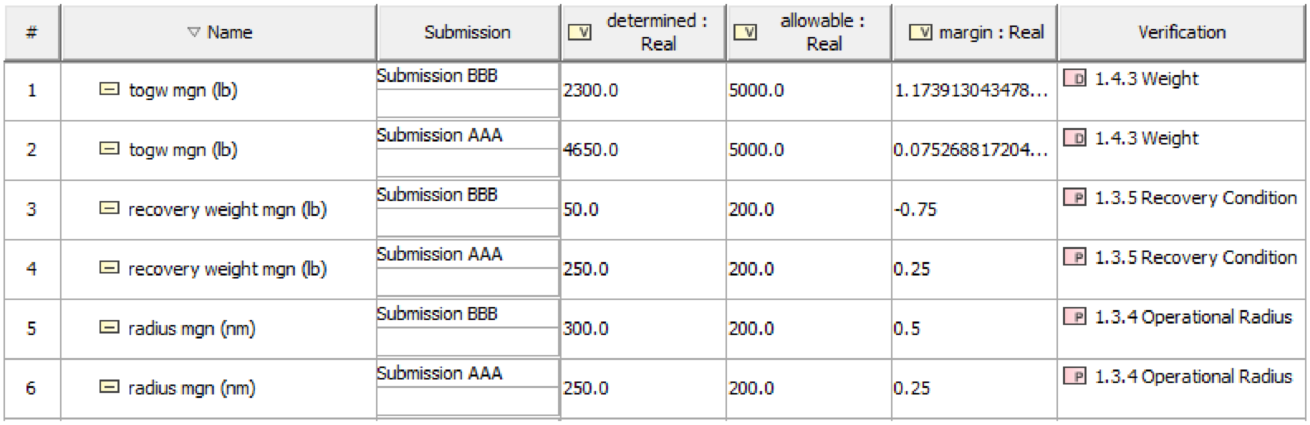


**Figure 4. Relating the Evaluation Model structure to the KPP-based Performance Requirements**

representing the baseline expectation from the government, and has slots populated with the appropriate value properties for that KPP. Then, a contractor-provided instance that specializes the System Model block is slotted in as a system-under-evaluation, shown in Figure 4. The Margin Block instances then calculate, for each respective KPP requirement, whether the provided model exceeds the baseline and by what percentage. KPP requirements can often include various levels of targets, such as a minimum baseline and a more aggressive target goal. The Evaluation Model can evaluate the same KPP against various benchmark requirement levels to further capture how successful a given contractor response is. The model is executed using Cameo Simulation Toolkit, which results in a table of all KPP margins. The setup of the Evaluation Model, and later the Estimation Model, means that it is trivial to add additional margins to track, as long as they exist as value properties in the generalized System Model.

The net result of the Evaluation Model is an easily readable summary of how well all contractor responses measure up to each other and to KPP requirements. This summary, and the patterns required to create it, serves to ensure the KPPs are present and easily accessible in all the contractor responses



| # | Name | Submission | determined : Real | allowable : Real | margin : Real | Verification |
|---|---|---|---|---|---|---|
| 1 | togw mgn (lb) | Submission BBB | 2300.0 | 5000.0 | 1.173913043478... | 1.4.3 Weight |
| 2 | togw mgn (lb) | Submission AAA | 4650.0 | 5000.0 | 0.075268817204... | 1.4.3 Weight |
| 3 | recovery weight mgn (lb) | Submission BBB | 50.0 | 200.0 | -0.75 | 1.3.5 Recovery Condition |
| 4 | recovery weight mgn (lb) | Submission AAA | 250.0 | 200.0 | 0.25 | 1.3.5 Recovery Condition |
| 5 | radius mgn (nm) | Submission BBB | 300.0 | 200.0 | 0.5 | 1.3.4 Operational Radius |
| 6 | radius mgn (nm) | Submission AAA | 250.0 | 200.0 | 0.25 | 1.3.4 Operational Radius |

**Figure 5. Output Table Showing Comparison of Two Submissions Using Evaluation Model**

in a consistent manner. This not only provides the acquisition party a cursory evaluation of how successful a given contractor response is, but in combination with the Estimation Model (Section 5), it can highlight potentially overambitious performance claims in these responses. A sample output table, as would be generated from the tool, is shown in Figure 5, where each margin block instance is referred to by the abbreviated "mgn" name, and the corresponding block architecture is given in Figure 4.

## 5. ESTIMATION MODEL

One of the key outcomes when transitioning to a model-based approach is an increase in transparency between the contractor and acquisition parties. The contractor supplied system models will include unique tools and methods for determining the final performance values. These methods, or documentation of the methods in cases of proprietary processes, are included in the contractor model for review by the acquisition party. It is advantageous for contractors to use generous estimates for the performance levels of their design, and thereby an important responsibility of the acquisition team is to determine how realistic a given response is and how valid the methods are that are contained within that response. The Estimation Model works in tandem with the Evaluation Model to highlight designs that have claimed performance levels that significantly exceed a reasonable value, and function as a warning to the acquisition party that certain calculations should be scrutinized further. This estimate for what constitutes a reasonable value must be dynamically calculated based on characteristics of the contractor provided design. This is accomplished through the use of "surrogate models," mathematical models that each estimate a performance value and are tailored to a specific acquisition project.

The term "surrogate model" here should not be confused with the term "surrogate pilot" as used in Section 2 for the overall Skyzer project. The surrogate model technique highlighted in this section is a well-established technique [4-6] that is useful for trade studies and other purposes (regardless of whether SysML is present or not). Surrogate models can be constructed by analyzing historical designs, extracting relevant design variables that drive a performance variable, and then constructing a relationship between the design variables and the performance variable. This surrogate model derivation can be performed using various regression techniques[7].

The Estimate Model consists of an Estimate UAV block that extends the generalized UAV system model provided to contractors. Constraint properties then are added to this Estimate UAV that each take the form of a surrogate model for estimating a specific UAV performance value. The result is a generic equation that can estimate the performance of a given contractor response using only the contractor supplied design values. Each surrogate model for a respective KPP estimate is incorporated as a constraint block in the Estimation Model. Similar to the Evaluation Model, the Estimation Model slots in the contractor-supplied instance and exclusively uses design values originally present in the general System Block furnished by the acquisition party as inputs. The resulting instance of the Estimation Model can be used directly as an input for the Evaluation Model, which adds these surrogate estimates as evaluation criteria for the contractor responses. A sample SysML parametric diagram shows the connections of the value properties to surrogate model parameters in Figure 6.

These estimates can be used as an additional benchmark to evaluate the validity of a contractor response. However, it is not necessarily problematic if a contractor exceeds these estimates. In fact, the more technologically advanced designs supplied by contractors will likely exceed a historically-based benchmark. Having this simple benchmark is already a useful tool for the acquisition party to judge contractor responses by, but there are several methods to further increase the applicability of these surrogates.

One method is to incorporate an inflation factor that directly increases the performance value estimate by some nominal percentage to more accurately represent modern expectations. Alternatively, a more detailed approach can be used by predetermining specific technological inflation factors that each correspond to the use of a specific morphological choice or technological inclusion. These technology inflation factors can either operate directly on the performance estimate, or they can instead modify the coefficient for a specific design variable input, and thereby increase the impact of this design variable on the final

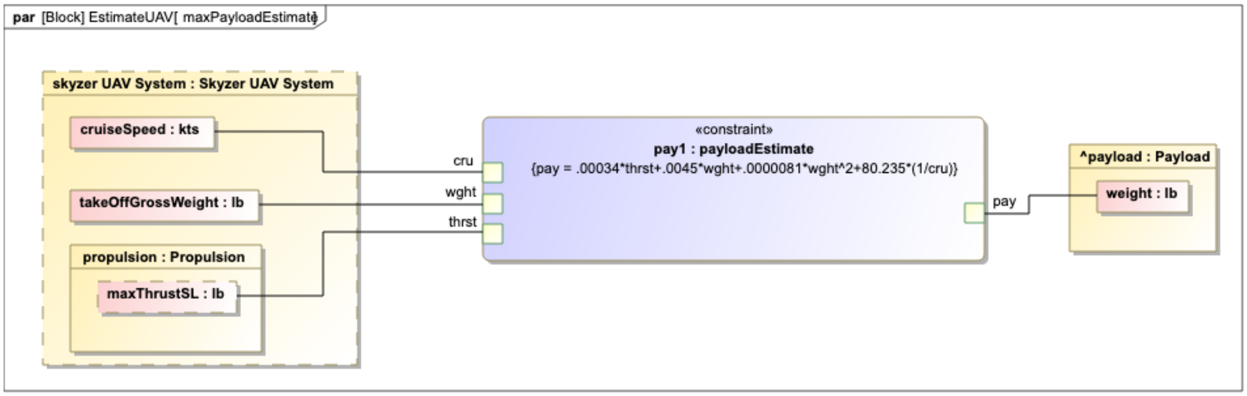


**Figure 6. Connecting Value Properties of Model to Estimating Surrogate**

performance value estimate. In either case, the inflation factors are incorporated as value properties in the Estimation Model and are appropriately incorporated into the surrogate constraint block. The acquisition team can create multiple instances of a constraint in this way, and these are modified on an instance-by-instance basis. Several values can be used for the inflation factors to create estimates at various levels of confidence. The historically derived baseline estimate has inflation factors set to one and is the conservative benchmark, while the other estimates with modified inflation factors – corresponding to various levels of uncertainty – would function as a scale of feasibility for contractor responses. If contractor supplied performance levels are on the upper end of this scale, they likely warrant scrutiny to determine if the underlying assumptions and calculations that yielded these results are valid.

The Estimation Model is not meant to be an infallible rule; it is another tool to help guide the acquisition team and expedite the RFP response evaluation process. One limitation on this approach is that the design variables required as inputs for the Estimation Model must be included in the generic system model supplied to contractors, so they must be applicable to all possible designs. In most acquisition processes this should not be a problem, as the proposed solutions should share enough in common with each other and with historical designs that finding universal design variables is possible. There may be rare situations where large morphological differences in a contractor design causes previously universal design variables to be inapplicable. This may void a specific Estimation Model value, but it merely puts the onus solely back on the acquisition team to individually verify this contractor value. Contractor designs that vary greatly from historically established solutions come with a larger degree of uncertainty and risk, and as such deserve more scrutiny in turn.

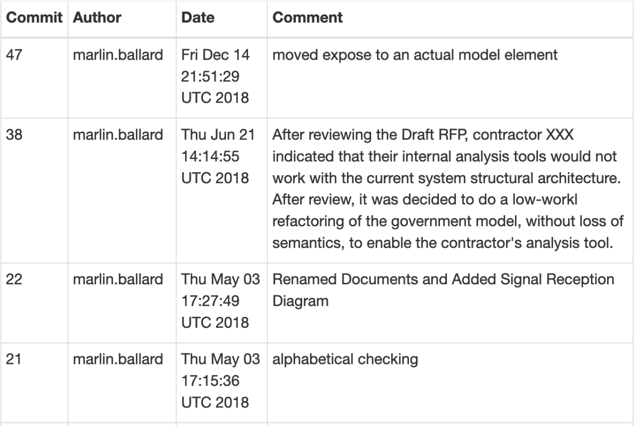
## 6. MODEL HISTORY FOR DIGITAL SIGNOFF

Providing a consistent line of decision traceability through a project is one principal advantage sought by adopting MBSE methodologies. This is similarly true in acquisition processes, where contractors may need to provide data items required by Contract Data Requirement Lists (CDRLs) to support their responses. Validation is needed to ensure that the data items received in an RFP response provide the correct data in the appropriate format. When validating manually, acquiring reviewers sign off on data items which meet the contract requirements. The MBSE acquisition methodology implements this signoff as an explicit model-based artifact itself within the authoritative source of truth model [8].

The authoritative source of truth is in an active living model, and it will likely continue to change as acquirer and responder proceed through the acquisition process. The digital signoff process described can be improved by adding model management techniques. In particular, signoff at various levels may be tied to particular commits to the model in its repository system. In the NAVAIR ecosystem, that database system could be either the Teamwork Cloud or OpenMBEE Model Management System repositories. Both of these systems offer mechanisms to track element history, but neither one is in a model-based format that can be persisted within the model environment directly.

The element history tracking capability provided by OpenMBEE is only visible as a read-only display in the ViewEditor side panel. It also lacks the linkage to the Teamwork Cloud commit history, which is the primary version control mechanism in the architecture. Similarly, the native element history capability provided by Teamwork Cloud produces the appropriate commit differences, but the results are only made available in a read-only MagicDraw window. Additionally, the Teamwork Cloud capability can take several minutes or more to construct the element history for any element, which is not scalable to the requirements of the project. The element history tool developed in this work offers an alternative to these approaches. The tool can be interacted with to produce an explicit model-based sign-off artifact, and thus persist a hierarchy of traceability throughout the acquisition process.

The tool operates as a UserScript-based viewpoint executed by OpenMBEE. As input, the corresponding view takes in any model elements which are connected via an «expose» relationship. When executed, the viewpoint method queries the target element for its database identifiers (as well as its corresponding project's database identifiers) using the MagicDraw OpenAPI. The element history capability makes requests to the Teamwork Cloud REST API interface to build a commit graph of the project. Then, the target element is compared across successive commits to identify those commits which had changes in its specification. The tool creates the corresponding model elements as well as a tabular representation of the commits, including name, date, and commit message, as seen in the Figure 7 example.



| Commit | Author | Date | Comment |
|---|---|---|---|
| 47 | marlin.ballard | Fri Dec 14 21:51:29 UTC 2018 | moved expose to an actual model element |
| 38 | marlin.ballard | Thu Jun 21 14:14:55 UTC 2018 | After reviewing the Draft RFP, contractor XXX indicated that their internal analysis tools would not work with the current system structural architecture. After review, it was decided to do a low-workl refactoring of the government model, without loss of semantics, to enable the contractor's analysis tool. |
| 22 | marlin.ballard | Thu May 03 17:27:49 UTC 2018 | Renamed Documents and Added Signal Reception Diagram |
| 21 | marlin.ballard | Thu May 03 17:15:36 UTC 2018 | alphabetical checking |

**Figure 7. Tabular Output of History Tracker**

This tabular representation may be combined with the digital signoff artifact by the use of instance specifications representing the commit information. Then, any individual commit of an element may be assigned the appropriate digital signoff stereotype and utilize the status enumeration and track the signoff personnel. In summary, the model element histories and digital sign-offs become model-based elements in their own right, which enables enhanced automation and reporting.

## 7. CONCLUSION

The simulation of the initial steps of the Skyzer surrogate acquisition program yielded a number of insights into the transition to a model-based acquisition process along with various tools to enable this transition. Some of this gained knowledge applies exclusively to the RFP-issuing and contractor selection steps, while many aspects are applicable throughout the acquisition process.

The structure of first creating a generalized system model allows for flexibility in the initial design selection for any acquisition process and allows for continuous specialization of the system as the project continues. Another persistently useful technique is the ability to generate text and text-based documents directly from native model elements. We can safely assume near-term that at least some of the stakeholders involved during acquisition are model-illiterate (possibly even in decision-making roles). This makes extracting information from the model and converting it to natural language text doubly important, since it both (1) keeps these stakeholders informed and (2) helps streamline the review of large quantities of data contained in models from a variety of sources. The Evaluation and Estimation Models proposed in this paper apply this concept specifically to the RFP response process where they function as tools for the acquisition team to quickly browse performance summaries contained in each contractor response. Finally, the ability to use digital signoffs on model changes ensures full visibility and accountability during the decision-making process. The transition to model-based acquisition will become increasingly viable if the benefits far outweigh the costs, and the methods in this paper all serve to exploit the full advantages of a model-based approach while minimizing the associated drawbacks.

Although the currently proposed tools will help enable a successful implementation of a model-based approach to the initial RFP stages of an acquisition process, the full benefits of MBSE will not be experienced until it can be applied from beginning to end. One next step in this transition to a fully model-based acquisition process is to simulate subsequent acquisition phases using these proposed tools. Additional methods and tools will be needed to handle the unique challenges that will likely arise within these ensuing phases. See our lab websites[2,3] for the latest ongoing research in these areas.

## ACKNOWLEDGEMENTS

This material is based upon work supported in part by the U.S. Department of Defense through the Systems Engineering Research Center (SERC) under Contract HQ0034-13-D-004 (Task Order 089, RT 195). SERC is a federally funded University Affiliated Research Center managed by Stevens Institute of Technology. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Department of Defense.

The authors would also like to thank the rest of the members of the SERC research team. Additionally, we would like to thank the NAVAIR sponsors for their support of this work.

## REFERENCES

[1] Cohen, D., “SE Transformation.” NASA JPL MBSE Symposium, Presentation. 2019 Symposium, 2019.

[2] Blackburn, M., R., Bone, M. A., Dzielski, J., Grogan, P., Giffin, R., Hathaway, R., Henry, D., Hoffenson, S., Kruse, B., Peak, R., Edwards, S., Baker, A., Ballard, M., Austin, M., Coelho, M., and Petnga, L., “Transforming Systems Engineering through Model-Centric Engineering, Final Technical Report SERC-2018-TR-103”, RT-170 (NAVAIR), February 28, 2018.

[3] Ballard, M., Peak, R., Cimtalay, S., and Mavris, D., “Bidirectional Text-to-Model Element Requirement Transformation”, IEEE 41st Aerospace Conference 2020, IEEE Aerospace, Mar. 2020.

[4] Ender, T., Luerck, R., Weaver, B., Miceli, P., Blair, W.D., West, P., and Mavris, D.N., “Systems-of-systems analysis of ballistic missile defense architecture effectiveness through surrogate modeling and simulation”, IEEE Systems Journal, Vol. 4 (2), pp. 156-166, June 2010.

[5] Ghosh, S., Jacobs, R. B., and Mavris, D. N., “Multi-Source Surrogate Modeling with Bayesian Hierarchical Regression”, AIAA Science and Technology Forum and Exposition (SciTech) 2015, Kissimmee, FL, Jan 5-9, 2015.

[6] Bernardo, J. E., Besson, C., Pfaender, H., Schutte, J., Mavris, D., “A Multi-Stage Surrogate Modeling Approach to Examine Vehicle-Level Technology Impacts at the Airport-Level”, AIAA Science and Technology Forum and Exposition (SciTech) 2015, Kissimmee, FL, Jan 5-9, 2015.

[7] Wang, L., Beeson, D., Wiggs, G., and Rayasam, M., “A Comparison of Meta-modeling Methods Using Practical Industry Requirements.” 47th AIAA/ASDME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, AIAA, May 2006.

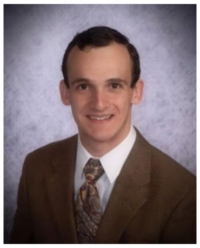
[8] Kruse, B., Blackburn, M., “Collaborating with OpenMBEE as an Authoritative Source of Truth Environment.” 17th Annual Conference on Systems Engineering Research (CSER), April 2019.

---

[2] https://www.asdl.gatech.edu

[3] https://www.msl.gatech.edu

## BIOGRAPHY



***Marlin Ballard*** *is a PhD candidate at the Georgia Institute of Technology's Aerospace Systems Design Laboratory. He received his M.S. in Aerospace Engineering from Georgia Tech in 2019. He received a B.S. of Aerospace Engineering and a B.S. in Computer Science at the University of Maryland in 2015. At Georgia Tech, Marlin has three years of research experience in SysML-related MBSE. He has additionally held three Systems Engineering internships at the NASA Jet Propulsion Laboratory for the development of a collaborative MBSE framework. Marlin is an OMG Certified Systems Modeling Professional Model Builder – Intermediate.*

***Adam Baker*** *is a PhD candidate at the Aerospace Systems Engineering Laboratory at the Georgia Institute of Technology. He formerly received his B.S. in Mechanical Engineering from Lehigh University. His research topics have spanned a variety of systems engineering topics including Model-Based Systems Engineering, Digital Thread Architecture, and the incorporation of structural analysis into the early stages of the design process.*

***Russell Peak, PhD*** *is a Senior Researcher at Georgia Tech in the Aerospace Systems Design Lab where he is MBSE Branch Chief. Dr. Peak originated the multi-representation architecture (MRA)—a collection of patterns for CAD-CAE interoperability—and composable objects (COBs)—a non-causal object-oriented knowledge representation. This work provided a conceptual foundation for executable SysML parametrics. After six years in industry at Bell Labs and Hitachi, he joined the research faculty at Georgia Tech. He is an active INCOSE member where he co-leads the MBSE Challenge Team for MBX Ecosystems. He represents Georgia Tech on the OMG SysML task force, and he is a Content Developer for the OMG Certified Systems Modeling Professional (OCSMP) program. Since August 2008 he has led a SysML/MBSE training program that has conducted numerous short courses for working professionals.*

***Selcuk Cimtalay, PhD*** *is a Senior Research Engineer at the Aerospace Systems Engineering Laboratory at the Georgia Institute of Technology. He formerly received his PhD. in Mechanical Engineering from GT. Model-Based Systems Engineering, He has several years of teaching, research, and industry experience. Dr. Cimtalay has conducted research on various projects on model-based systems engineering (MBSE).*

***Mark R. Blackburn, Ph.D.*** *is a Senior Research Scientist with Stevens Institute of Technology and principal at KnowledgeBytes. Dr. Blackburn's research focuses on methods, models, tools and the use of semantic technologies and ontologies for cross-domain model integration of complex and cyber physical systems, He is the Principal Investigator (PI) on several System Engineering Research Center (SERC) research tasks for both NAVAIR and US. Army ARDEC on Systems Engineering Transformation through Model-Centric Engineering. He was PI on a FAA NextGen project and has received research funding from the National Science Foundation. He develops and teaches a course on Systems Engineering of Cyber Physical Systems. He is a member of the SERC Research Council.*

***Dimitri Mavris*** *earned his B.S. (1984), M.S. (1985), and Ph.D. (1988) in Aerospace Engineering from Georgia Tech. He is the Boeing Chaired Professor of Advanced Aerospace Systems Analysis in Georgia Tech's School of Aerospace Engineering, Regents Professor, and Director of its Aerospace Systems Design Laboratory (ASDL). He is an S.P. Langley NIA Distinguished Professor, AIAA Fellow, Fellow of the Royal Aeronautical Society, and a member of the ICAS Executive Committee, the AIAA Institute Development Committee, and the US Air Force Scientific Advisory Board. He is also the Director of the AIAA Technical, Aircraft and Atmospheric Systems Group, and co-chair of the Committee on Aviation Environmental Protection's review board of independent experts.*

*For the past 25 years, Prof. Mavris and ASDL have specialized in the integration of multi-disciplinary physics-based modeling and simulation tools. ASDL's signature methods streamline the process of integrating parametric simulation toolsets and enable huge runtime improvements that facilitate large scale design space exploration and optimization under uncertainty. Recent research focuses on combining these methods with advances in computing to enable large-scale virtual experimentation for complex systems design.*