Configuration and Data Management of the NASA Power and Propulsion Element MBSE Model(s)

Edith L. Parrott
NASA Glenn Research Center
Cleveland, OH 44135
216–433–8749
edith.parrott@nasa.gov

Laura C. Spayd
Vantage Partners, LLC
Brook Park, OH 44142
216–433–3212
laura.c.spayd@nasa.gov

Abstract—Systems engineers worldwide have been working to implement Model-Based Systems Engineering (MBSE) environments, tools, and methodologies. A significant, expected benefit from MBSE is a reduction in the time commitment for all aspects of systems engineering. Although advances in MBSE have occurred, a gap has emerged between the traditional configuration management (CM) processes for documents and those needed in an MBSE environment.

MBSE is a tool to support systems engineers in requirements, design, analysis, verification, and validation activities from concept phase throughout the life cycle of the project. MBSE is not a new process being added to the existing SE processes, but captures the data that systems engineers create into a single environment. Systems engineers working in an MBSE environment have become more efficient and produce a higher resolution of data because elements representing the system of interest are placed within a single construct where requirements, architecture (physical & logical), and concept of operation scenarios are all related to one another.

Historically, systems engineers capture and track system information through the generation of documents, drawings, and other artifacts, which are released, baselined, and governed by a document-centric CM process. The work required to maintain and distribute documents throughout the system life cycle using the traditional CM processes is a time-consuming task for both systems engineers and CM practitioners alike. The MBSE environment requires adaption of traditional CM processes or the creation of entirely new CM approaches in order to 1) manage the MBSE model output in data or documents and 2) perform CM within the actual digital model. Systems engineers and CM personnel from the Power and Propulsion Element (PPE) Project at NASA Glenn Research Center have developed and are continually improving a method for baselining and configuration-managing technical content within the PPE MBSE model(s).

This paper describes the process of defining how to configuration manage not just the artifacts but the requirements, verifications, and functional decomposition of an MBSE model for an evolving project. The paper also presents the results of trial and error testing, required adjustments to CM processes, and lessons learned from developing CM processes for the PPE system-level requirements and generating procurement artifacts. While developed for the MBSE environment, these processes still address standard CM activities such as protecting NASA-sensitive and partner proprietary information, baselining content, processing change requests, and providing configuration status accounting.

TABLE OF CONTENTS

1. INTRODUCTION .......................................................1
2. BACKGROUND ..................................................2
3. ADAPTING CM/DM PROCESSES FOR MBSE ..........3
4. ADAPTING THE MBSE MODEL FOR CONSIDERATION OF CM/DM ........................................6
5. LESSONS LEARNED ...........................................8
6. CONCLUSION AND FORWARD WORK ...............10
ACRONYM LIST ....................................................11
REFERENCES .......................................................11
BIOGRAPHY ..........................................................11

1. INTRODUCTION

NASA has been investigating the capabilities and opportunities that Model-Based Systems Engineering (MBSE) can provide for the past decade. Historically, systems engineers capture and track information through the generation of documents. These documents are released, baselined, and governed by a document-centric configuration management (CM) process.

The true nature of MBSE develops and details the interconnection between the captured information and the artifacts it provides. The complexity of interconnected models means that changes in the model could inadvertently change a project’s baseline information. Therefore, the ability to configuration manage the models used to capture system engineering content is critical. This ability to lock down an MBSE model would then confirm that during MBSE development and maturity, all captured data has not been changed unintentionally and that any artifacts produced will always contain the same baseline content, despite changes or maintenance made to non-baselined areas of a model.

When system engineers think about what is meant by model CM, their first thought might be that this represents model validation. Model validation pertains to making sure that the model(s) are following a consistent set of rules throughout the modeling construct. Configuration management of the model is more than that. Model CM is making sure that the
information within the model does not change from one instant to another unless it is planned, approved, and goes through the proper change management and configuration management processes.

This paper presents the development approach, implementation, and a set of basic lessons learned by the Power and Propulsion Element (PPE) Project at the National Aeronautics and Space Administration (NASA) Glenn Research Center while bringing MBSE of the PPE requirements under configuration and data management (DM).

The PPE, a 50-kW class spacecraft, will provide key functionality for the Gateway Program, including: power transfer to other Gateway modules, orbital maintenance and orbital transfers, attitude control, and communications with Earth, lunar systems, and visiting vehicles. The PPE is acquired through a deviated Broad Agency Announcement (BAA), resulting in the development of the spacecraft through a public-private partnership.

The PPE Systems Engineering and Integration (SEI) team, which includes the MBSE team, is responsible for developing and maintaining MBSE models for different requirements sets for the PPE spacecraft. The requirements are decomposed from the Gateway (Level 2) requirement set that informs the PPE Level 3 requirements. The “PPE Level 3 Requirements” model, in addition to the “NASA Unique Requirements” model are incorporated by the PPE industry partner into their requirements and design (Level 4). Additionally, reference information regarding Safety and Mission Assurance (SMA) and Interoperability Standards (IOS) has been modeled for technical consideration for the PPE partner.

The need to control technical baseline content in the multi-model environment for PPE – and not just in the released requirements documents – led the MBSE and CM/DM teams to define and administer the models under the rigors of CM and DM.

2. BACKGROUND

Traditionally, CM of requirements data has been accomplished in requirements documents released and controlled by the CM team, as governed by a CM plan. These documents were prepared with input, but were separate entities, from the MBSE models used to develop and define the requirements. A requirements document was the defining artifact of the project technical baseline requirements – the “source of truth” for the requirements. Changes were worked in the MBSE model only so far as to create a new output for the document. Change boards approved and baselined the requirements document and any subsequent changes. Change Requests (CRs) were written against a requirements document and presented a clear From/To that was used to obtain approval from the signatories. The users and stakeholders interacted with a requirements document. Modelers interacted with the model.

This focus on the document as the source of truth for project requirements baseline allowed for both a conscious and unconscious decoupling of a document from the MBSE model itself. Changes could be made to the requirements data in the model, but as long as the published document requirements set remained the same, CM was not broken.

In a multi-user environment such as MBSE modeling, shared profiles/attributes may be updated or branches may be worked separately and then merged. This allows changes to creep into the MBSE model (intentionally or not) without perception or acknowledgement that baselined data is potentially or actually being changed. In traditional document-centric CM, the most common practice is that a single book owner controls the document file and content with the CM team holding the master file after baseline approval and release, thus controlling all changes and preventing unintentional changes.

In order to accomplish CM and DM of MBSE models, the PPE CM team had to adapt the products, processes, and mindset about requirements and the products that represent the project requirements. The PPE MBSE team had to learn to work together with the CM team within the constraint of CM “locking” the models from changes. Locking means the content cannot be changed unless it is unlocked by the owner who locked it in the MBSE tool. Working together to change the norms for requirements, the PPE CM team accepted responsibility for working with the MBSE team to learn the basic tenets of MBSE modeling approach and tools. The CM team adjusted processes that migrated the project requirements’ source of truth to be within the MBSE models while still producing a document of baselined content that can be attached to the acquisition mechanism (contract, for example).

The MBSE and CM teams had to work closely with various entities to create and refine the construct for MBSE model CM/DM:

- The SEI leads for PPE provided guidance and approval for the innovations or adaptations made to the CM for models and requirements.
- The project technical authorities and Project Management provided concurrence on the approach to change authority and management.
- NASA Glenn Research Center (GRC) IT security team helped to create an approval process and access capabilities for a secure interface that provides broader access to the requirements models for non-MBSE, PPE, and select Gateway program team members.
3. ADAPTING CM/DM PROCESSES FOR MBSE

Determining how the PPE Project CM/DM processes were adapted for MBSE is best stated by noting how critical tenets of CM were addressed. Configuration Management key responsibilities, as defined in SAE/EIA 649 (current revision C), National Consensus Standard for Configuration Management [1], include the five main functions in the lifecycle of a project:

- Configuration Management and Planning
- Configuration Status Accounting
- Configuration Identification
- Configuration Change Management
- Configuration Verification & Audit

Additionally, SAE/GEIA 859, National Consensus Standard for Data Management [2] serves as a reference basis for data management practices. For the purposes of this paper, note that CM in terms of documents or models means that the technical content is under baseline control and change management. DM is meant to encompass the release and control of technical information (such as MBSE models and documents) that is not under baseline control. This DM content that PPE uses within the model constructs includes data from the PPE partner that PPE does not own technically and therefore, cannot baseline. However, it remains important for technical data that is under DM to remain consistent and it still is reviewed by SEI when updated or incorporated. However, it is not subject to the CR process. DM also encompasses storage of partner data and project records. To simplify, when referencing general activities, the term “CM team” or “CM lead” is meant to also represent any DM activities conducted.

Configuration Management and Planning

Initially, as the PPE project first received Authority to Proceed into formulation, the draft CM/DM plan was based on the standard, document-centric CM process. Document numbering was set for the configuration- and data-managed documents that PPE releases. Change management and the Change Request (CR) process was defined based on standard change control structure and rigid change control approach. The Configuration and Status Accounting (CSA) tool was set up to reflect the traditional CM approach in a project.

As the PPE CM team became more involved in the MBSE process, the CM approach had to answer and adjust for circumstances or situations that had not been presented in the typical processing of a document. CM personnel needed to work with MBSE personnel to define naming conventions for MBSE models and branches, determine the level of control over changes and access to data, and develop processes for auditing changes and communicating status accounting for models. The PPE CM team needed to identify their role in the MBSE environment and learn to navigate the software to sufficiently provide CM and DM of the data as well as plan workload to include the time it takes to observe and work within the MBSE modeling environment for tracking changes.

The PPE CM and MBSE teams also had to determine how to control modifications. For example, PPE released both a Level 3 PPE Requirements document and a NASA Unique Requirements Document under baseline CM control for the Broad Agency Announcement (BAA) for PPE. In addition, at that time, PPE developed several MBSE models that traced content regarding interoperability standards and Safety and Mission Assurance (SMA) reference data. These models were not baselined nor were they used to generate baselined PPE documents. Because they were not baselined, these models were considered under DM control, as they were revision-controlled and had content released through the CM team but not baselined.

In the process of working through how to manage the PPE MBSE models under CM or DM control, in order to control the technical requirements data that was baselined by the project, the decision was made to hold the Level 3 Requirements and the NASA Unique Requirements models under strict CM control. This meant that once the technical data was baselined, the models that contain that data were locked recursively at the top level of the model by CM lead. No changes could be incorporated into the models by anyone (even members of the MBSE team) without the CM lead unlocking the model. Changes were required to follow the CR and board approval process if they affected the technical baseline data.

While they were being developed as reference, the non-baselined models were left open for the team to work. It was later determined that the CM team would also lock the reference models (that did not generate baselined requirements documents) under what was termed DM control. The models were locked by the CM lead at the top level, but changes to these models would not require the same level of change management and rigor.

In this way, MBSE model management mirrors the approach to configuration- versus data-managed documents that the team follows. Documents that are baselined are subject to CRs and board approval to change baseline technical content. Documents that are version controlled but not baselined are processed through the CM team but are not subject to CR or board approval as they do not contain any baseline technical content. All master files are controlled by the CM team upon release.

Configuration Status Accounting

Configuration Status Accounting (CSA) is a key CM responsibility and activity on a project. A CSA tool needs to be developed to provide instant status information on the baselined configuration data items. The status provided by the tool should include change status and related CRs, data
sensitivities, baseline status, current revision, title, and other points of data.

For PPE, the CSA tool includes not only the configuration-managed but also the data-managed content. In the early stages of PPE, the requirements documents were baselined and released with PPE document numbers and entered into the CSA. After MBSE model CM was introduced, PPE CM needed a way to reflect the actual models in the CSA tool alongside the released requirements documents and more importantly, to denote which attributes/properties within the model housed baselined values under CM or values under DM.

Listing the MBSE models in the CSA tool serves more than one purpose. From a CM standpoint, it is appropriate and necessary to list the models as configuration data items and to provide their status (released, in change, etc.) and pertinent information such as title and version. Additionally, the CSA tool records the critical information of which fields/attributes/properties in the model contain baselined content as those attributes are under CM/DM control.

From an MBSE and team standpoint, listing the MBSE models in the CSA tool serves as a visual and key reminder to the team that the models themselves are under CM team control – that changes must be approved and tracked, that the MBSE team does not “own” the content outside of project management and CM purview, and that all team members should use the models as a source of truth.

**Configuration Identification**

When implementing CM on an MBSE model, it becomes especially important to note which fields or attributes in the model contain technical content under CM/DM control. This configuration identification allows the flexibility and use of MBSE agility when appropriate while enforcing a CM lock to changes where required.

For each version or iteration of a model (for CM, when a model is placed under baseline control or a revision is processed; or for DM, when a version is updated), the MBSE team produces a .ZIP file with all of the necessary model files associated with the newly baselined or re-baselined model version. The model’s titles include the version number associated with the server version. The CM team places those .ZIP files in the CSA tool as a project record. At any point, the project could revert to a previous version or identify values and information from that version. This method allows CM to open the models without having to worry about connecting to the server or determining the correct software version, as each .ZIP file version is self-contained and backwards compatible.

**Configuration Change Management**

The biggest challenge in the change management of the MBSE models for CM has been to find a balance in control. After some stops and starts, the team has worked on processes that allow the modelers to complete their work in the model by CM or DM locked model trunks with approved content while allowing branches and “sandbox” areas). The CM team has responsibility to control changes to technical content but not over-control the process. Once models are locked under CM or DM, the CM team controls an entire model but needs to be able to determine and work with MBSE to identify which changes constitute baselined changes that need approval from the project control board vs. non-impact modeling changes, such as model maintenance or server migration.

The PPE change management process initially required a CR for any change to a baselined document. Documents that were never baselined by the control board would be revision controlled (data managed) but did not require a CR or board approval. As the PPE project matured and the MBSE modeling efforts were brought into CM/DM control, the team had to identify which fields in the model contained the baselined content or content to be held under DM and how to adapt the change process. Otherwise, because certain models were considered baselined, any change in those models would appear to require a CR, even if the change did not affect baselined content.

When model changes to attributes or content affect the technical baseline, those changes are reviewed and implemented only with the approval of the project change board and technical authorities through the CR process (after potentially also going through an engineering, risk, or safety review board). From the model, the baseline requirements are posted to a web interface for PPE team members; an artifact generated from the model is also posted for team use. CM-locked MBSE requirements models may require changes that in no way affect the baselined technical data, such as changes to profile information. To communicate these changes to non-baselined technical content on a CR or to unnecessarily revise the requirements document and take them to control board for approval would be a waste of resources (time, budget). The model can be updated without updating the requirements document; they are decoupled.

Changes can be approved and implemented in the model – and published to the team server – in a much more expedient manner than waiting for the document to process through formatting and signatures. Changes can be made visible in the server to the team immediately upon approval and post-merge audit. If the project requirements document needs to be updated – for publishing to a partner, for example – that can follow at a later time. But the model, with change approval, can be updated and the team’s use of the model as a source of truth allows for changes to be reflected in the model much sooner.

After the Level 3 and NASA Unique Requirements content had been baselined, the CM team and MBSE teams worked to identify and note which attributes in the model represented the approved baselined data that was published and distributed beyond the project for the PPE procurement efforts. Those attributes were listed in the CSA as the baselined attributes and are ones that require a CR if they are
changed in the model. As other attributes are populated with data, such as when verifications and interfaces activity is matured and linked in the model, the board will be asked to approve and bring additional attributes under the baseline CM control. The model follows the same level of rigor for change board authority and requires a CR to change any technical content that is baselined. However, identifying the fields that are baselined content allows the team to also note which fields do not contain baselined content — and thus, would not require a CR.

The PPE MBSE and CM teams initially worked on a Preliminary Change Notice (PCN) process for MBSE models. The intent would be to allow for multiple changes to be made in multiple branches and then incorporated into the model at different times. PCNs would be processed individually and then one CR would be processed to update the document when deemed necessary, such as in advance of major milestone reviews. This would avoid pushing through multiple CRs on model updates. The team also worked through an early change process construct that applied the PCN process to non-CR changes, in order to find a way to document changes in the model and alert the team to pending updates.

Upon use of the process, it became apparent that in actual use cases, multiple simultaneous branches would be very unlikely in the PPE construct. Requiring technical approvers to sign PCNs for data-managed models or for changes to non-baselined content for the CM models was overly constricting and over-managing the changes in the model.

Configuration Verification and Audit

Working through various versions of the change process pointed the CM team to verification and audit activities as a way to ensure preservation of the baseline technical content and also to communicate or track changes to data-managed content. The CM and MBSE teams developed an audit process that takes the place of Change Requests in some instances. In instances where a CR is required, the audit process serves to provide more context and verification that a CR was correctly incorporated into the model. The audit approach received approval from PPE Technical Authorities (Chief Engineers and Chief Safety Officer) and SEI leads. Project Management also confirmed this approach.

For PPE, changes to baselined technical content will always require a CR — including attributes changes from “blank” to populated data for data fields that are being added to the baseline technical content or From/To changes in the baselined attributes. However, for changes that do not affect the technical baselined content (for CM models) and for DM models, the CM and MBSE teams work together to identify and incorporate changes into the MBSE models in real time. These changes could include model maintenance-type updates to attributes or metadata, sharing other reference information, or data cleanup.

For example, models contain profile information, which functions like a template used for model structure. Profile information is always changing with attributes and new model stereotypes being added in order for the model to work though the project lifecycle. Some of the changes to profile information are flowed down to the PPE project from a higher level (MBSE profiles implemented at the program level). These profile information changes that do not affect baseline technical information within the model can change without a CR. However, if the profile deletes an attribute that is being used by Baseline data or data-managed data, then it would follow model change/baseline processes.

To process changes in a data-managed model, the MBSE lead provides the list of changes in the completed branch and provides a contextual explanation of the changes. This information serves to advance the CM team understanding of MBSE modeling constructs and allows the CM team member to verify that no technical content is changing. The CM lead audits the merging of the branch (with changes) into the model trunk and MBSE provides a change summary report. While changes are worked in a branch, the PPE team is always able to access the current approved model trunk on the forward facing teamwork server (cloud-like server). While the audit is taking place, the internal version of the model is unavailable for changes during the merge and commitment to the server. As soon as changes are approved and merged into the model, the MBSE lead can then push the approved general model onto the server for team access. The model is available without waiting for a document to then be revised and processed through the CM signoff and release process.

The CM team audits to verify not only that the intended changes are made but also that no unintentional changes have been introduced. The CM team member conducting the audit records the file names (models, From/To files, etc.) of any artifacts used in the merge and to determine completeness and accuracy. The CM lead summarizes the activity and writes the audit report. Once the MBSE lead reviews and provides input, the CM lead signs the report, which is posted as a project record and is accessible to the team members.

To understand how the audit process can be used in lieu of a CR, consider these examples: In MBSE, like all software, there are updates that need to be made to the environment. The NASA PPE Level 3 Requirements model and the NASA Unique Requirements models are both currently held under CM baseline control. During the course of the PPE project, both models have had to migrate servers, which changed their version number; and both models were changed to incorporate MBSE profiles from the Gateway program. Strict interpretation of traditional CM would have required that, because this was a change that affected a configuration-managed model, a CR and change board review was required. However, by paying attention to configuration identification and reviewing the listed attributes that constitute the baselined technical content in the models, CM was able to run a report to confirm that no baselined technical content was
impacted by either proposed change. This allowed for the models to migrate servers and incorporate the Gateway profile without a CR or change board. The CM lead reviewed the proposed changes with the MBSE lead, unlocked the models, supervised the merge, and then re-locked the models. The CM lead documented a CM MBSE audit report that was detailed how the change was made without impact to the baseline.

For DM models, the CM team also processes changes and provides audit reports. For example, the MBSE team needed to change the titles of 169 data-managed requirements to remove specific punctuation (colons) that prevented analysis within the MBSE software from running properly. Without having to involve many levels of the team, CM and MBSE leads were able to process the (editorial) punctuation change, certify that those were the only changes made to the model, and re-lock the model without the time and resources of a PCN, CR, or other change review. The audit report serves as the record defining the reasons behind the version change of the model as well as showing the net effect (169 requirements title changes with no impact to content or intent).

This is not to say that all content changes are incorporated into DM models without the appropriate technical review and approval. Model changes are not made outside of the realm of approval and guidance from SEI and technical teams; however, model maintenance and other non-technical changes are made without taking time from the technical teams when the impact of the change does not affect their data.

**Special Considerations for Export Control, Proprietary, and Sensitive But Unclassified**

Documents that are released with data sensitivity are readily identifiable with export control, pre-decisional, proprietary, or other sensitivity markings; and notations. Users of a requirements document processed through traditional CM methods would have any number of visual clues to designate restrictions on content. Migrating to the MBSE environment for CM of requirements necessitated that PPE handle the notation and consideration of data sensitivity in a different manner.

PPE has implemented or is in the process of implementing the following considerations for data sensitivities:

- In the model, the MBSE lead developed and instituted a READ ME label that contains the standard Export Control, Proprietary, or other markings/wording. This information is pushed at the front of the model in the server environment and appears at the top of screen, much like a header.

- Attributes or other fields are used to label for partner proprietary, sensitive (such as NASA pre-decisional), and export control.

- PPE follows a file naming convention that includes key words regarding data sensitivities in the file name of the model. Development branches can be similarly named.

- MBSE created a template for the requirements content so that when it is exported to a printable document, any data or export sensitivity read me attributes print as the first information on the cover.

- Information posted to the external server for broader team access would receive appropriate markings. Currently, only the publicly-available requirements information is posted for broader use and the proprietary/more sensitive information is held in models that are not shared to the general model.

Data sensitivities also require PPE to protect and limit access to any data that is not publicly-available. In order to access the server that houses the requirements models interface for users, one has to request access and have approval vetted not only through NASA IT security but also through PPE MBSE lead. Security training must be current and two-factor authentication is required. Additionally, on the site, only the “forward facing” MBSE model that is approved and released through the PPE project control board is available. Members of the SEI and MBSE team, technical authorities, and others may access other server areas that house branches and models containing unapproved changes or partner-proprietary values and linkages that are not widely shared.

**4. ADAPTING THE MBSE MODEL FOR CONSIDERATION OF CM/DM**

Utilizing MBSE for requirements management has the advantage of capturing traceability and verification, including traceability to other functional and physical elements used to describe the system, such as functions, activities, and product breakdown in a single versatile and robust environment.

In order to start implementing CM/DM into the modeling environment, the MBSE modelers needed to ensure that they had a solid understanding of what CM is and how to incorporate its practices. MBSE started to look at the process that is used to configuration manage documents. Most projects that use MBSE typically configuration manage the output from the model as a configuration data item and an official record. Typically, the model had not become a distinct configuration data item of the project under CM control. The model was simply a tool used to produce the requirements set for a document; the model existed separately for the MBSE team to manipulate and change. Because of that, the modelers could make multiple changes within the modeling construct and not have to worry if they affected the data that was used to create the official record. However, if the official record needed to be updated, there could be inconsistencies that had been introduced that would also need to be addressed. Additionally, the model could contain information that was not output to the requirements...
document; thus, sensitive information and data “in work” could be housed in the model and in work while a set of baselined data was output and existed in a released requirements document. The standard practices by the MBSE team had to be adapted.

Multiple-Model Architecture

Many MBSE models are self-contained, meaning that all the modeling done for a project is performed within a single model. When considering CM functionality within the modeling environment, the MBSE team determined that a single model modeling construct would not work as effectively or efficiently as multiple models would. As a single model evolves, the CM process would become more difficult as the CM team would have to re-verify baseline model content at every major version of the model. Versioning could entail changes in relationship, documentation, or attributes within the model elements. These changes would be easy to identify when a model is small; but as a model grows in complexity during its life cycle, a CM review could likely miss changes in a larger, all-encompassing model. To alleviate that possibility, the PPE MBSE lead decided to implement multiple shared models to capture the data for PPE.

The multiple model construct also allows data updates to be made on the same change cycle. A project would never put all interface information for that project within a single interface document. Rather, the interfaces would be broken up to minimize changes, only updating affected interface data when necessary. By having the NASA Unique Requirements in a separate model from the SMA or IOS reference requirements, for example, the data is grouped in a way to allow for processing of changes to one data set without having to review the effects or potential changes in another data set, as they are each held in separate, locked models.

Shared Model Content

The PPE requirements and reference models are configured to be shared between one another as read-only. This allows a system engineer to still be able to see all of the data within the model, but grants the CM team control over the model content that has been approved as baselined technical content. This construct also gives flexibility in configuration management or data management of the models so that not all of the models would need to be updated all at once and can be done individually, as needed, much like an individual document would be updated when necessary.

Not only does this multiple-model schema provide flexibility for CM of baselined technical content, but also for handling different types of data, such as proprietary information from the PPE partner. For example, the PPE project is able to maintain two models: a general model able to be shared broadly or publicly, and another model that contains the proprietary data to be kept internal to the project. Depending on how data in a model is shared in MBSE linkages, models can be set up so that the general model never sees the proprietary data model but the proprietary data model reads the general model, as shown below in Figure 1.

![Figure 1. Relationships in Shared Models](image)

This figure portrays the various relationship cases in the shared models and demonstrates how some information passes through to all models and is visible to all users while other information is only shared in a structure that limits broad access.

The use arrow direction depicts a model being brought into another model. The spacecraft (SC) System Model can view shared data from SC Requirement Model, SC Verification Model, Program System Model, and Profile Model since it is read into the SC Requirement Model. However, the SC System Model is unable to view Proprietary Model (the arrow is not bi-directional). However, as shown by the arrow direction all pointing into the Proprietary Model, the Proprietary Model can view the data in all the models, even though SC System Model, in this example, cannot see the Proprietary Model’s data. Since the Proprietary model cannot be viewed with other models, it therefore can contain proprietary/sensitive data that might not be able to be shared broadly. This construct helped the PPE MBSE team to set up the modeling environment with flexibility to answer model accessibility and visibility of sensitive data.

Model Set-Up: Server and Branches Add Capability

Before starting, the MBSE team had to confirm a base understanding of how changes to the models could be implemented with regard to how base content is affected and how the model versioning could be controlled. For version control, the model(s) were placed onto a teamwork server. This server updates the version every time a new model version is recommitted to the server. The teamwork server works well for model versioning. To have complete confidence that changes are not done to baseline technical content until approved through a change board, the team has also implemented the ability of the database server to provide branching capability. This branching allows the modeler the ability to make the necessary updates, to perform analysis, or to add additional data to the model within a branch without affecting the baseline. When the new content receives approval, that branch is then merged back into the model trunk, which contains the baseline technical content under CM control. The PPE CM lead locks the trunk of any baselined model. The development branch has its own
version control, which is separate from the model trunk. Figure 2, below, illustrates how this works.

![Model Trunk Diagram](image)

**Figure 2. Development Branches**

Maintaining an approved trunk and making changes in a branch of the model allows the CM team and the PPE team at large to understand that models within the trunk are the baseline and branches are working versions of unapproved or proposed changes. Branches function much like suggested From/To changes in a document that are presented to the change authority and technical reviewers for consideration. Content changes can be approved as a whole or individually. During review, changes in a branch can be reworked or removed, and are only incorporated into the trunk during the branch merge. Due to versioning, CM and the modeling team have the capability to revert back to an older model version if ever deemed necessary. To prevent erroneous changes to the baseline content, the model trunk is locked by CM personnel after an approved branch is merged and committed. Locking of this data inhibits modelers from making changes to baseline content. Only the model trunk is shared between models, which results in only baseline content being used by the team.

Figure 2 also shows a maintenance version of the model trunk. Basic model maintenance occurs routinely to adapt to changing needs within the MBSE environment. Maintenance of the model includes – but is not limited to – updates to attributes, naming conventions, and model organization; and clean-up of diagrams. Before any model maintenance is performed, a review of the maintenance work required is conducted by the MSBE Lead and the CM lead to assure that the maintenance will not affect the baseline data content. If the maintenance affects baselined content approved by the project control board, which requires the baseline model to be unlocked by CM and baseline content to update, a CR is initiated, following the CM process. To prevent erroneous changes to the baseline content, the baseline model is kept locked by CM personnel.

**MBSE Baseline Action Log**

In order to efficiently capture and understand all the changes that could potentially change baseline content, a “MBSE Baseline Action” Log (Action log for short) was created for the MBSE team’s use. This action log specifically captures work associated with potential changes to the baseline content and major maintenance tasks. The information captured, at a minimum, is listed in Table 1. The Action Log is the MBSE team’s tracking log and is separate from the CR numbering log that is maintained by PPE CM. A model evaluation review is completed before merging a development branch back into the model trunk. This Action log also assists other in understanding all the changes that are completed.

<table>
<thead>
<tr>
<th><strong>Table 1. MBSE Baseline Action Log – Minimum Information</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data Captured</strong></td>
</tr>
<tr>
<td>Database number</td>
</tr>
<tr>
<td>CR Number</td>
</tr>
<tr>
<td>Date received</td>
</tr>
<tr>
<td>Due date</td>
</tr>
<tr>
<td>Model</td>
</tr>
<tr>
<td>Model Branch and version</td>
</tr>
<tr>
<td>Assignee</td>
</tr>
<tr>
<td>Status</td>
</tr>
<tr>
<td>Completion date</td>
</tr>
</tbody>
</table>

**MBSE and CM Process**

To summarize the change process for baselined technical content in MBSE models under CM control: The MBSE modeler requests a CR number from the CM lead whenever a change to the baseline technical content in a model is required. A branch is then created with the CR number in its file name to identify that the branch contains the change content associated with the CR. Branches could have multiple CRs associated with them, but the PPE project has initially kept only one open CR per branch in early implementation of this process. The development branch for the CR allows modelers to make changes to the model without affecting the baseline main model until content is approved. This is shown in Figure 2. By using development branches, other PPE models do not include/incorporate any of the changes being made within the branch until the change is approved and merged back into the main model trunk. Before a development branch is created, the modeler logs it into the “MBSE Baseline Action Log” located in eRoom. Once changes are completed and approved, the CM lead unlocks the baseline model and oversees the merge process. A review of the differences between the branch and the baseline model is performed with the CM team before completing the merging process and committing the new model to the server. Once the model is merged into the model trunk, the CM lead conducts a post-merge audit of the model to confirm accuracy before locking it. After the model evaluation review is completed, the modeler enters the development branch version and the final baseline model version used to complete the CR into the “Action Log”. The CM and MBSE leads complete any audit report documentation to accompany the CR.

**5. Lessons Learned**

When bringing the CM and MBSE processes together, the team started with a steep learning curve on both sides. The
MBSE modeling team needed to understand what information the CM team needed and how their process works. The CM team had to understand the modeling tools and approach, and to fully comprehend what aspects of technical data in a model can change, how CM should document those changes, and what level of approval and documentation is necessary based on the certain criteria.

Previous sections in this paper have detailed how the PPE project MBSE and CM teams have already adopted some of the lessons learned. This section serves to distill and reiterate some of the key areas where processes were adapted and innovated for MBSE.

**Speaking the Same Language & Gaining Fluency**

One of the first lessons learned in bringing the PPE MBSE models under CM and DM was making sure that each party understood what the other was stating. The CM team needed to learn to speak somewhat rudimentary MBSE language – learning about attributes, model profiles, and other features. And the MBSE team needed to better learn configuration management terminology and tenets.

Knowing what each term means and how it influences the technical data helps the CM practitioners to recognize when baseline content or data-managed content may be impacted. For example, learning that diagrams are simply views of locked relationships and that they are not locked or managed themselves helped the CM team to sort through changes to hone in on ones that needed more scrutiny. And knowing more about CM practices helped the MBSE team to recognize that their frustration about being “locked out” of “their” model trunks was actually a good thing – that protecting the models from inadvertent change was CM in action.

**Managing the Overall Configuration in Shared, Reference, and Other Models**

The ability to configuration manage models individually has helped with the CM process. However, because the MBSE construct adopted by the PPE project uses multiple models that share information from reference and other models, the MBSE team had to develop and follow a process for integrating approved changes into the separate, shared models during the audit and merge process. During sessions to merge changes that affect multiple models, the MBSE team ensures that any model opened prior to the merge is updated after the first set of changes is made before proceeding with the next branch merge. Otherwise, a model loaded prior to a branch being merged will contain old (pre-merged) data. Proceeding without an update will show unnecessary From/To changes when comparing the two models, as they would be pulling from different shared data files – one using the new trunk version and one still pulling from the pre-merge trunk.

The MBSE team has also had to make sure that after each merge is done, they make the local copy (.ZIP) file for the CM team before another merge takes place – to capture the correct versions and values to replicate a version.

**Recognizing the Model as a Source of Truth While Maintaining a Document**

Because the CM team members were used to a requirements document being the source of truth, the CM team routinely edited punctuation, spelling, and grammar in the requirements document prior to sign off. With CM being implemented in the model, the implications of the language tweaks to model output became apparent during an audit, because a disconnect was introduced by CM between the document and the model. The CM team made minor wording edits in a rationale statement of a requirement, which was discovered during an audit and was subsequently corrected. However, a lesson was learned in thinking of the model as the source of truth even when a document is still required and created. At another point in time, the CM team provided editorial insight into the requirements in development in the model, cleaning up punctuation, spelling, and grammar within the model prior to approvals. This replaced the standard edit of the document prior to release and allowed the CM team to still assist in that capacity.

In the working environment for PPE, the model has not fully replaced the requirements document that is output from the model. Having to publish a requirements set on a public website for the solicitation made a published document more usable and portable than publishing the model. But PPE team has made strides in referencing the MBSE model as the source of truth for the requirements set and using the related document as simply one artifact that reflects the model.

**Streamlining the Change Process Where Possible**

While developing the change process, the SEI lead challenged CM to rethink the approach to the MBSE models as a special case, which led to a breakthrough. CM had documented a separate change process for documents and one for MBSE models, which included using PCNs and CRs. While similar, there were differences documented in how the changes would be implemented and what was required for each type of change; namely, DM documents not related to an MBSE model would not have been subject to a PCN, Reconsidering the approach to provide change control and planning of MBSE artifacts as any different from other products led the CM team to document one change process for all technical data products. The implementation may differ slightly based on the environment or format of the data, but to get the team thinking of MBSE as under CM and to streamline the process, combining into one process flow was a good lesson learned. CM personnel had to think of the models as any other technical data product under the CM purview – not as a separate entity to be handled in a different manner from the other changes processed.

Additionally, because data-managed documents did not require a PCN or a CR, this challenged the original approach of handling MBSE model changes not affecting baselined
technical content in a different way. By taking a step back and thinking through what could be processed in the same manner and what absolutely had to be documented differently, CM and MBSE personnel were able to develop the audit report process. Whenever the CM team unlocks a model under DM or a CM-model that is not having baseline technical content changed, generating an audit report captures the change changes, verifies that the baseline is not impacted, and creates a record without requiring the technical authorities, management, and others on the change board to be involved.

Planning Time and Implementation of Changes

Both the MBSE lead and the CM lead had to improve planning for changes. Each had responsibilities – for getting CM into the MBSE tools and environment and getting the MBSE team to think in terms of affecting CM/DM – and identified the training required for both teams. Fitting in the training and starts and stops of developing a change process and other implementations within the course of a very busy project schedule was a challenge. In the course of learning new tools and what could be accomplished – having to try and discontinue different variations of approaches – could have been perceived to be a failure. However, all of the efforts were advancing an overall approach and implementation, in increments that are still being built upon.

The PPE CM team has had to plan more strategically around updates to baseline content in the model to allow time in their process for differences between simply processing the document revision (without regard for the model environment) and auditing a branch merge into the model and writing an audit report. The CM team must allow time to audit the models for intentional changes and to verify that no unintended changes have been made. Once the learning curve has leveled out, the audit discussions and details should be easier to document. The team recognizes that having more than one CM team member versed in MBSE tools and processes would help alleviate the burden from the CM lead.

To help streamline the time and effort to audit a branch, the PPE MBSE and CM teams have learned how to run analyses and project comparison reports from a branch to a trunk prior to the session for merging. CM has learned how to best extricate the changed data from the model (fields under configuration or data-managed scrutiny) and compare to the existing values. In short, the team is getting better and quicker at reviewing branches prior to a merge.

Using the Compare feature for a report prior to merging gives the CM team the ability to see what all of the potential changes are in the branch. This report, when run during the merge, can take longer than 20 minutes to generate. MBSE team has learned to prepare the report and have it ready prior to any CM audit sessions, so that the review does not take even longer to conduct. Having the report opens up the communication lines so the CM team can understand the changes and ask the necessary questions to further understand the intended changes and verify that they are made.

Identifying Baselined Fields in Models

As the MBSE models were created, for the BAA, the requirements set was baselined for the Level 3 and NASA Unique Requirements and the requirements for Interoperability Standards and Safety and Mission Assurance reference models were placed under DM. It became clear that the best way to manage the configuration identification of the baselined technical content in the configuration-managed models was to list which fields (properties, applied stereotypes, etc.) contained the technical content that generated the baselined requirements set. Listing the fields that contained values – those that have technical content that CM needed to control – assisted when auditing routine model maintenance or other changes that did not affect those fields.

With requirements models, not all the attribute fields are populated at the start of the project but rather, get filled in later in the project lifecycle and then be brought under the baselined technical data. For CM to readily identify when a branch is affecting the baselined technical content and would require a CR and change board is critical.

For data-managed models, tracking which fields contain data under the DM information set helps for traceability and ingestion of changes during the lifecycle. For example, the Interoperability Standards reference model was populated with data from the original partner proposal. This set was locked by CM personnel as a data-managed model. At various stages in the partner lifecycle (such as internal requirements reviews), once changes are made to the fields containing technical data (vs. profile or model information), a good From/To document can be generated because the original values were captured and locked.

Documenting Processes and Procedures Allows for Consistency and Broader Team Knowledge

Over the course of implementing the CM and MBSE process, the MBSE lead has developed step-by-step procedures for how to merge a branch into a trunk. The CM lead has developed templates for the Audit Report and created data fields in the CSA to better support CM/DM of MBSE artifacts. Processes and guidelines are continually being drafted and refined.

There remains a lot to learn. But the PPE team is getting smarter and more agile in their efforts and the tools and processes to bring MBSE models under CM.

6. CONCLUSION AND FORWARD WORK

As the implementation of configuration and data management of the MBSE models matures, PPE MBSE and CM teams recognize areas of possible improvement, broader application, and additional work.

At the beginning of the process, the knowledge transfer between both groups was about 10% in that each team fully understood their discipline but not entirely the
interrelationship between CM and MBSE. Since the start of this effort, the knowledge transfer and understanding on how to do configuration management using MBSE, is approximately 50%. Continued refining of the process and procedures should slowly increase this over time. Both the MBSE team and the CM team are still learning more about configuration control of MBSE models in the SE environment and how the two can work together and have a stronger product at the end.

The CM lead has identified a need to more completely document the processes and decisions for implementing configuration and data management of MBSE models so that other NASA projects could potentially make use of what has been learned and what could be done differently from the start of a project to maximize implementation. Training is always being identified for the team, to keep them abreast of how CM is being implemented in the MBSE space and for the CM team to become more fluent in MBSE features. Collaboration with other practitioners of CM who have implemented work processes with MBSE – especially for large, complex systems – would be beneficial, much like technical working and users groups that exist for MBSE modelers.

For MBSE, implementing the other MBSE features in a project lifecycle for requirements – such as parent/child relationships, verification and validation, and other linkages – remains open work and considerations to work through as the PPE project matures.

**ACRONYM LIST**

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>BAA</td>
<td>Broad Agency Announcement</td>
</tr>
<tr>
<td>CM</td>
<td>Configuration Management</td>
</tr>
<tr>
<td>CSA</td>
<td>Configuration Status Accounting</td>
</tr>
<tr>
<td>CR</td>
<td>Change Request</td>
</tr>
<tr>
<td>DM</td>
<td>Data Management</td>
</tr>
<tr>
<td>GRC</td>
<td>Glenn Research Center</td>
</tr>
<tr>
<td>IOS</td>
<td>Interoperability Standards</td>
</tr>
<tr>
<td>MBSE</td>
<td>Model-Based System Engineering</td>
</tr>
<tr>
<td>NASA</td>
<td>National Aeronautics and Space Administration</td>
</tr>
<tr>
<td>PPE</td>
<td>Power and Propulsion Element</td>
</tr>
<tr>
<td>PCN</td>
<td>Preliminary Change Notice</td>
</tr>
<tr>
<td>SC</td>
<td>Spacecraft</td>
</tr>
<tr>
<td>SEI</td>
<td>System Engineering and Integration</td>
</tr>
<tr>
<td>SMA</td>
<td>Safety and Mission Assurance</td>
</tr>
</tbody>
</table>

**REFERENCES**
