Massachusetts Institute of Technology

Model-Based Systems Engineering: Documentation and Analysis

Week 3: Critiquing an MBSE Approach

Project

Name

Tomas Mawyin

Massachusetts Institute of Technology

Week 3 Project

Overview

In Week 3, you will choose, download, and critique one of the projects we provided.
Please note that the project provided combines weeks 1 and 2 project submissions – good practices– from a previous course so you will not be conducting a critique of a full model.

Each document contains real cases from different industries, so you can select the one you are more interested in.

Project Objectives

- To practice creating quality critiques

You should presume that the rationale for your critique is to: "**Evaluate whether a project should adopt the MBSE approach or stay with the status-quo.**"

REQUIRED STEPS

Step 1: Choose one of the good practices provided

Step 2: Download and read the project

Step 3: Critique the project using the template from the document you downloaded

Step 4: Add your name and surname to the project, and save your file with your name and surname too, i.e., [**W3Critique_Name_Surname**]

Step 4: Evaluate the two critiques from your peers that the platform will automatically assign you

Massachusetts Institute of Technology

Step 1: Choose System and Define Scope

What is your system? What is in scope for the MBSE effort for your chosen system?

Vehicle Traffic Assist (VTA) is a software and hardware package for consumer passenger vehicles to manage both keeping the vehicle in it's lane as well as managing the distance to the car ahead during traffic jams. The design of the vehicle itself is outside the scope, as are other vehicles on the road, and infrastructure elements. VTA is envisioned to be an SAE Level 3 [1] feature, such that the driver is always expected to oversee the vehicle operation and takeover when requested by the system.

Initial MBSE effort for VTA will include capturing system functional requirements, system architecture, feature level use cases, and information on how the use cases are realized. Subsequent iterations of the VTA model will include architecture decomposition from function to form elements, identification of interfaces, requirements decomposition and their allocation to elements and interfaces.

[1] Levels of Driving Automation

Step 2: Define MBSE Approach

How would the approach to this project change with MBSE?

The approach to building VTA will change dramatically with MBSE-driven development, as compared to a traditional approach to building this kind of vehicle functionality. Specifically, the team will have designed with the rule that no requirements are allowed outside of the model, and all requirements must be testable against the model. Further, the team will incorporate traffic simulations directly with system models, so that the system parameters can be calibrated against simulation data automatically.

Step 3: Define MBSE Purpose

What are the purposes of your MBSE effort? Describe the financial and non-financial benefits you expect.

The purpose of employing MBSE to help build VTA is strictly to support integration of the system with the vehicle. By forcing all team members to work through a single model, we will mature our understanding of the internal interface between parts of the system. If this MBSE effort results in some increase in productivity, it will be an added benefit (but not one that the company will invest against).

Step 4: List Major Tenets of MBSE

Describe how will you model the system. Briefly describe your approach to each of the major tenets of MBSE.

Central model or federation of models: The VTA project will employ a central model that could be accessed (r&w) by multiple teams. Changing the model will involve a review process involving technical experts and/or peers.

Model Views: To start, the model views will be identical to the SysML diagrams, until the team is able to gather feedback on SysML diagrams from internal and external stakeholders. We may customize these views for stakeholders (VP of Engineering, Product Manager, External Vehicle Partner) in the future.

All system blocks will be named with capitalized first letters to promote easier identification as a pattern. A numbering convention will be followed for all requirements.

Model repository or library: The model repository will be stored on the team's shared drive - it will not be permitted to branch the repository. Users may check out code, but it must be re-integrated with the central model.

Model Checking: The model will enable unit consistency checks and logic check for normal and degraded (including diagnostic) functionality. Performance checks and calibration shall be excluded from the scope of model checking.

Ontology: Function, component, and interface requirements should show clear allocation to architectural elements. Taxonomy as defined in SAE's J3016 shall be employed when modeling the system.

MBSE Methodology: The team will not use a proprietary MBSE Methodology, but rather will follow the company's existing Agile process for the first attempt

Step 5: Identify the Most Important Qualities of Great Models

Reflect on the qualities of great models – what are the top three you are concerned about for your MBSE approach?

The most important quality for the model is "Internally Consistent", as referenced above in the Purpose of MBSE - our objective is to make it obvious where we have interface deficiencies internally and fix them. A secondary quality we are looking for is "Well Formed for Optimization", as this VTA function is going to require significant training with simulation data.
Finally, we would like the model to be reusable. The VTA system must be usable on multiple vehicle models.

Step 6: Identify Systems Engineering Tasks

Systems engineering has a variety of different tasks depending upon its role in the organization such as interface management, change management, facilitate information transfer. Which of these or other tasks in your view are applicable to your chosen MBSE strategy?

One of the key tasks for VTA is to encode the system requirements in a model, so that we can trace where requirements impact the different pieces of the system. VTA will also be used to formally specify the interfaces between system components, to the extent that we will not use the existing company "Interface Control Document" on this program.

Step 1: Develop Five Queries for Your System

As you know by now, models provide a good deal more than just a collection of attractive diagrams. Models are often stored in repositories with a defined data structure. Like databases, model repositories make it possible to query the model for specific information, e.g. an impact analysis when changing a requirement. SysML doesn't define a query language, and most modeling tools allow the user to write a script to query the model. You can write queries like "are all actions allocated to parts?", "are all requirements satisfied?" and so on.

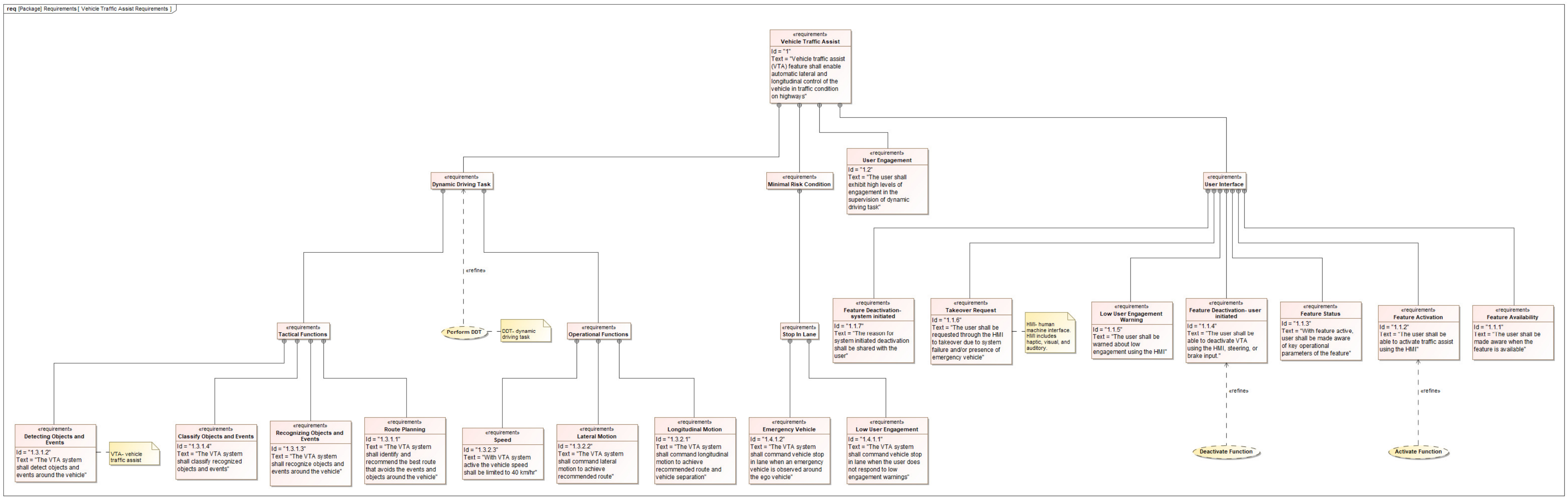
If you had a full data model available for your system, what would be five of the most important queries you would write to inform your system engineering functions?

Query	Rationale
What is the requirement traceability?	*Ensuring traceability of the requirements across all functions and abstraction levels. Feature definition requirements at the top shall be traceable to component (hardware and software) requirements at the lowest level. This query should provide similar current DOORS functionality*
Have the functions been allocated to components?	*There needs to be a clear decomposition of function and form, along with allocation of function to one or more components.*
Are all the requirements covered by testcases?	*Requirements need to be tested during unit testing, verification, and validation stage. The testcases are part of the testing plan and based on the component, element, and functional requirements. Each requirement should then map to one or more testcases.*
Do the use cases cover the complete operational concept (OpsCon)?	*A use case completeness check needs to be performed. The existing use cases need to be checked against the existing/agreed OpsCon to confirm coverage.*
How do the requirements perform against various requirement quality metrics?	*A quality assessment of all the requirements will be based on their ability to express the information clearly and precisely in textual format. This check could be based on the guidelines offered in the INCOSE guide for writing requirements.*

Massachusetts Institute of Technology

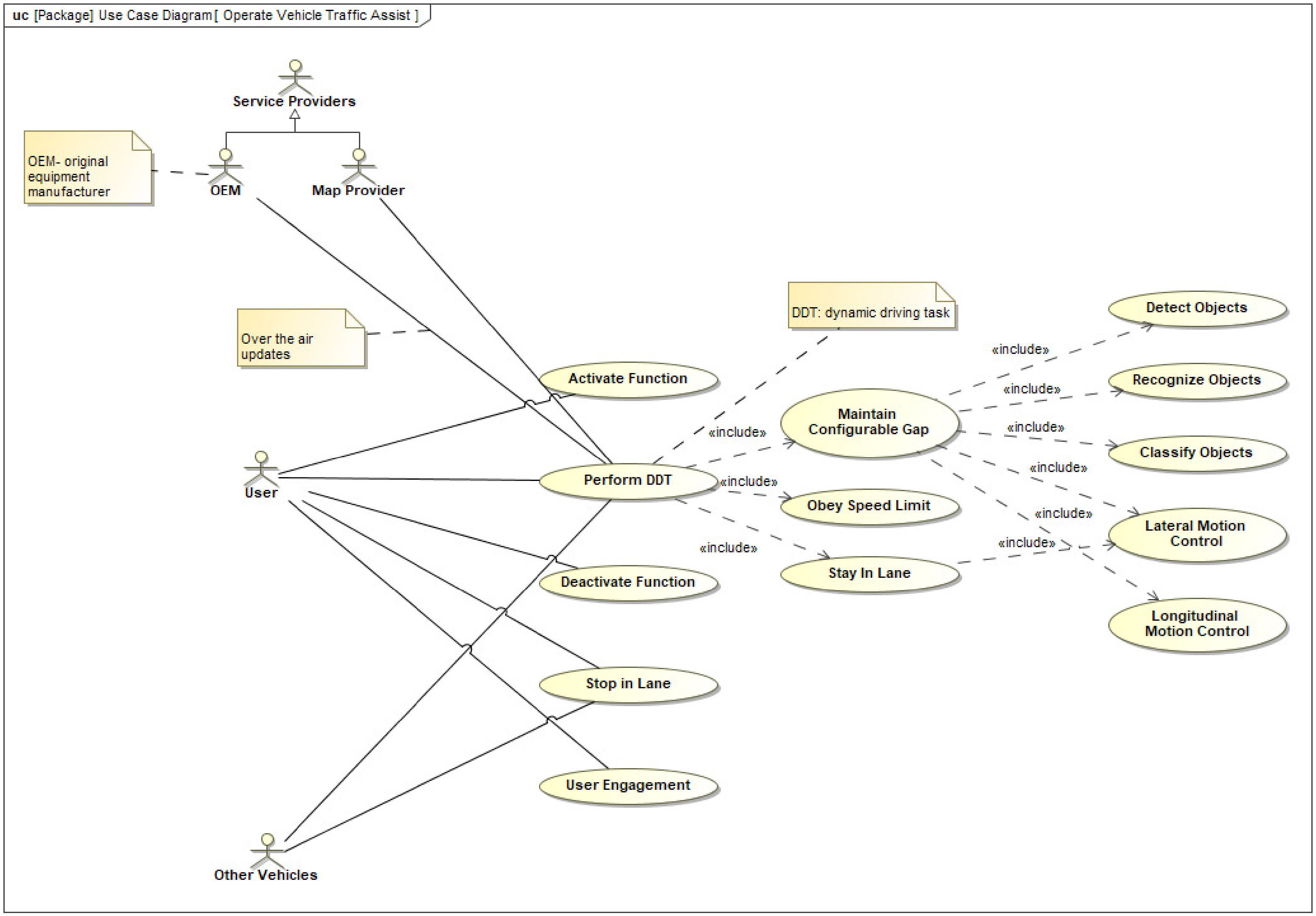
Step 2: Develop a Requirements Diagram

For the system you chose in Week 1, please create a requirements diagram below. Include at least <u>five requirements</u>.



Step 3: Develop a Use Case Diagram

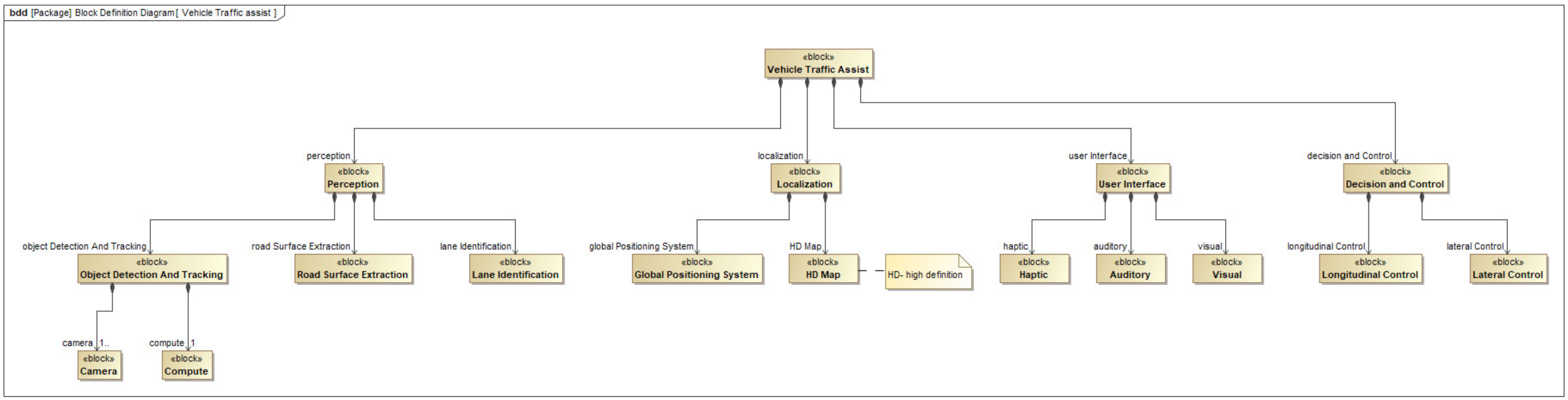
For the system you chose in Week 1, please create a use case diagram. Please feel free to leverage the format below or create your own.



Massachusetts Institute of Technology

Step 4: Develop a Behavior or Structure Diagram

For the system you chose in Week 1, please create a behavior or structure diagram. You do not need to use an MBSE modeling software; we suggest using simple shapes available in PowerPoint to represent the blocks, and arrows for direction.



Week 3 Project: Critiquing an MBSE Approach

Overview

Critique this project MBSE approach covering:

1. Scope and Purpose (limit 300 words)
2. Strengths and Weakness (limit 300 words)
3. Qualities of Great Models (limit 300 words)
4. Conclusions and Recommendations (limit 300 words)

You should presume that the rationale of the critique is "**Evaluate whether a project should adopt the MBSE approach or stay with the status-quo.**"

Please note that this project combines weeks 1 and 2 project submissions – good practices– from a previous course so you will not be conducting a critique of a full model.

REQUIRED STEPS

Step 1: Critique this project following the instructions.

Step 2. Save and upload the project in the platform.

Step 3. Peer review the two critiques from your peers that the platform will automatically assign you.

Massachusetts Institute of Technology

Critiquing an MBSE Approach

1. Remark on whether the model is meeting the intended scope and purpose *[Limit 300 words]*

This project is using MBSE to build a Vehicle Traffic Assist (VTA) model. The scope of the model is clear in terms of the main function of the model but there are some components of the scope missing. The analysis does a good job in giving the reader an idea of what elements are excluded on the model, for instance, the vehicle itself or other vehicles on the road. However, there is missing information about the organizations involved, and what level of hardware/software will be included.

There is a good amount of detail in terms of other dimensions. For example, there is a good level of detail on what information the model will contain to inform the users. This is well related to the purpose of the MBSE approach which is to help in the integration of the VTA system with the vehicle. The reader can see the benefits of using an MBSE approach on such a complex system, but there are no details on how the main purpose of the model will be implemented or under what conditions. For instance, there is no real representation on how the VTA system will integrate and work with the vehicle.

After reviewing the diagrams on this model, it’s clear that the developers spend a lot of time on the internal requirements and structure of the system, as well as the main use cases, but the model still lack the details on the integration to the vehicle as stated in the purpose. On another note, the diagrams are detail enough to show some of the additional benefits described in the purpose. There is a good amount of information that can be used to query the model in a clear and effective way.

Massachusetts Institute of Technology

Critiquing an MBSE Approach

2. List the model's strengths and weaknesses. Your findings should relate back to specific instances of the model with screenshots or callbacks.

Strengths
(Limit 150 words)

- There is a really good organization and definition on how the model is set up. Most diagrams indicate their main benefits and the relationship to the overall system.
- Sequence diagrams show a good amount of information on how the system needs to behave to complete the function of assisting during traffic.
- There is a clear list of requirements that are classified in a proper manner. By implementing requirements this way it is easy for engineers to query the system effectively.
- Uses cases show a good amount of detail for the main function and the interaction with other vehicles and services.

Weakness
(Limit 150 words)

- The scope of the project mentions hardware and software implementation but there seems to be more focus on software functions.
- There is no indications of failure modes or any mitigating actions in case of a fault. For example, what would happen if the perception subsystem fails to recognize other vehicles?
- There is no allocation of requirements to any of the components shown in the structural diagrams, making traceability difficult.
- There is no method shown to verified and validate requirements.
- The approach mentions the use of data for simulations but there are no parametric diagrams to achieve this.

Critiquing an MBSE Approach

3. Evaluate the model against three qualities of great models, as well as supplementary behaviors or qualities you believe are relevant to MBSE. *[Limit 300 words]*

- **Linked to Decision Support:** Even though this is an initial model and is still missing some information, the model shows value for the decision making process. The users of the model would be able to understand if the system is meeting the expected function or which parts are not performing as required.
- **Understandable and Well-Organized:** The model is very well organized. It starts by providing an overview of the different diagrams and how they are connected to form the system. All the diagrams are clear to understand and it would be easy to aggregate new diagrams (for example, parametric or state machines) or add/update components to the existing diagrams. Some additional work can be put into the modularization of the different components.
- **Elegant:** The model is at its infancy and therefore it is difficult to tell if the model is build with an economy of description. Once the model is build with representative data it would be ideal to keep in mind this point in mind.

- **Deployment:** Another aspect I would emphasize is the issue of deployment. Since the VTA feature is to be used on commercial vehicles, this could be an important factor to be represented if the system needs to interact with different brands of vehicles. Also, this is an important factor to consider if there is a supply chain involved in the development of this system.

Critiquing an MBSE Approach

4. Offer your conclusions and recommendations evaluating whether a project should adopt the MBSE approach or stay with the status-quo. *[Limit 300 words]*

There seems to be a lot of effort put in the initial creation of the model. As stated before, some additional work needs to be done to complete other areas of the model, this will include the hardware components, allocation of requirement to the hardware and software blocks, failure mode detection and corrective actions, as well as some emphasis in the distribution and deployment of the system. That being said, it is clear that the MBSE approach is showing great benefits and will allow the system developers to analyze, optimize, and verify their system with great results.

I would recommend the use of the MBSE approach for the remainder of this project. I would also recommend, if possible, to assign a developer to ensure the modularity of the system is included, that way this project could serve as a template for future projects.