

Model-Based Systems Engineering: How to Drive Enterprise-Wide Adoption

Executive Overview

Products are getting more complex. The evidence is in the climbing number of electronics, ever-lengthening cables and wires in harnesses, and burgeoning lines of software code in practically every offering in every industry. Product development is becoming more complicated as well. New roles with broader and wider responsibilities join the development effort. And compliance and integration across systems require new processes.

Systems engineering can help. It provides a robust methodology to mitigate the risks associated with growing complexity. It instills structure and procedures to make sure everyone starts, and stays, on the same page.

Savvy organizations are increasingly turning to model-based systems engineering (MBSE) as a single source of truth for multi-domain systems. However, simply arming systems engineers with MBSE tools is not enough. They cannot mitigate increasing complexity if they are working in silos. They need participation from stakeholders across functional departments. To enable organization-wide participation, companies must extend MBSE model access using intuitive and familiar tools. One such approach relies on spreadsheet-based interfaces to the MBSE model. It provides expert and non-expert users with the capabilities and information to perform their jobs and reap the full rewards of MBSE.

The traditional usage of Excel spreadsheets to manage systems engineering processes is not the ideal approach, however.



Why Spreadsheets and Documents Fail

Companies with systems engineering competencies have used documents and spreadsheets to manage these initiatives in the past. While these tools are useful for general purpose applications, they fall short when applied to a systems engineering initiative.

This is because documents and spreadsheets are not designed with the complexity of today's systems engineering initiatives in mind. Documents are often used to track requirements as individual sentences. Those requirements cannot be easily accessed and individually modified, especially by a large number of concurrent users. This leads to inconsistencies and errors. Spreadsheets are also problematic. Access is highly constrained: only one user can open and modify the spreadsheet at a time, causing a version control nightmare.

To make matters worse, these files often don't just contain the requirements, but extend to other parts of the requirements, functional, logical and physical (RFLP) systems engineering approach. In other words, they contain functions, logical architectures, and physical architectures. This is vital systems engineering information that is now difficult to access for many simultaneous users across the full spectrum of a systems engineering initiative.

Documents and spreadsheets are also inadequate when creating relationships or allocations between RFLP entities. This information is inputted and maintained manually, using a text-based format. When dealing with a large quantity of entities, which is often required given the rising complexity and scale of today's systems, the process completely breaks down. This is because there is simply too much information to manage using documents and spreadsheets.

MBSE: The Right Approach

Every systems engineering initiative must facilitate close interaction and collaboration between different

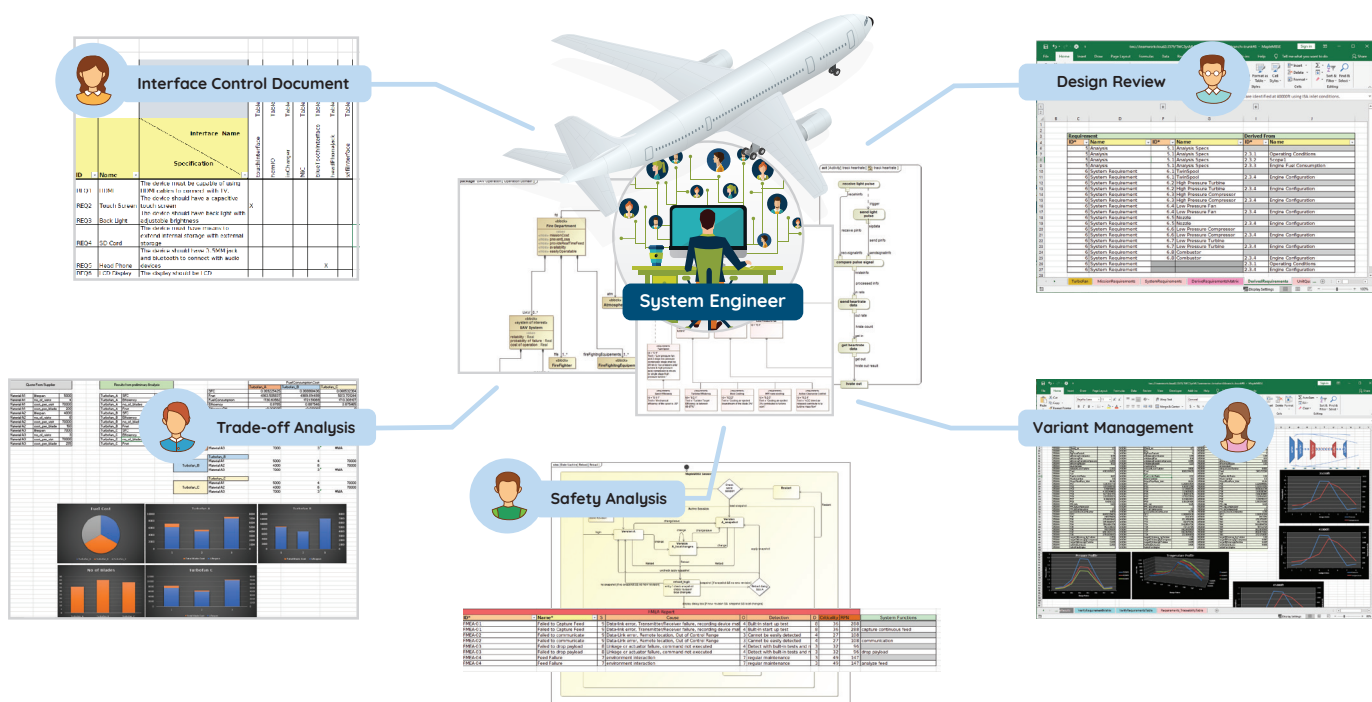
engineering disciplines and other cross-organizational teams. This helps companies increase the efficiency of their processes and enhance product quality. It also ensures that they meet business requirements and project deadlines. This is where MBSE can help. It formalizes the application of systems engineering through the development and use of a unified model that is consistent, and that persists across the systems engineering lifecycle.

MBSE replaces the traditional document-based approach with a model-based methodology. In doing so, the MBSE approach treats each RFLP entity as its own, separate item. This allows each entity to be individually created, modified, tracked, and managed across the systems engineering lifecycle. And it ensures every stakeholder is working with the latest information available. Any changes made to an individual entity propagates across the model, providing a single source of information accessible to all users. Everyone can access the right information, at the right time.

By treating each RFLP entity as a separate entity, an MBSE approach can create relationships between these entities. Each requirement can now be allocated to a specific function. Each function can be allocated to a specific area of the logical architecture, and so on. This creates an automated network of RFLP entities. The organization enables intelligent relationship management that can be scaled up to meet the complex needs of very large development efforts. It's a cohesive solution for any systems engineering initiative.

The Shortcomings of an MBSE Approach

An MBSE initiative should empower systems engineers to use MBSE processes, methods, and tools to perform their work and share it across their organization. Unfortunately, there are several problems with many of today's MBSE tools, which hamper this vital work and cross-organizational collaboration.



An example of collaboration within an MBSE framework. By using tools such as MapleMBSE, different stakeholders can use simplified, task-specific interfaces to contribute to the overall systems model.

Many MBSE tools provide advanced, powerful functionality to empower systems engineers. While these capabilities improve productivity for these experts, they are too difficult for casual participants to use. This reduces participation in the systems engineering process. It puts users off and hampers companies' efforts to realize value from any such initiative.

Too often, systems engineers end up doing their work in isolation. Some engineers may break down high-level requirements into more detailed ones, but others in the company do not close the loop back to the original high-level requirement. Participants may translate physical architectures to create the bill-of-materials inventory via completely disconnected tools, duplicating work and introducing the risk of error. This fragmented approach puts up further communication barriers and blocks organization-wide participation. As a result, organizations

miss the opportunity to realize a single, unambiguous definition of the system as represented by MBSE.

Leadership teams then can optimize their processes, people, and technologies to realize the full benefits of systems engineering and communicate these with the wider business.

Democratizing MBSE with the Right Tools

Despite these issues, MBSE is still the best approach for most systems engineering initiatives. It helps organizations find and apply the right design tradeoffs and decisions to meet the project requirements. It cuts costs, saves time, and expedites innovation. To realize real value from MBSE, organizations must put the right tools in place to enable participation across the company.

To enable access to an MBSE model, a new (but very familiar) interface is the best fit: the spreadsheet. But its application is very different from solely managing systems engineering information, as discussed earlier.

Here, a spreadsheet acts as the user interface to the MBSE model, providing valuable information and insights to non-expert users. Systems engineers create the MBSE model using specialist MBSE tools. Non-experts use the spreadsheet-based interface to access and contribute to the same MBSE model. These expert and non-expert tools work against the same model just through two different interfaces. As a result, changes are seamlessly tracked and managed, providing everyone with the right level of up-to-date information.

This spreadsheet-based approach allows non-expert users to complete their work quickly and effectively, without the expert-level interface requirements of MBSE tools. Systems engineers can continue to use specialized MBSE tools under the hood, allowing them to combine this expert-level functionality with their expert-level knowledge.

Using this approach, companies realize the value of MBSE efforts by allowing systems engineers to use this advanced functionality while the rest of the company participates through a familiar but connected spreadsheet interface. Everyone has the right tools for the job they need to carry out.

Summary and Recommendations

- Model-based systems engineering (MBSE) can address rising complexity, helping organizations manage systems engineering initiatives. Collaboration is key for success. Companies enable collaboration by extending access to MBSE models using intuitive and familiar tools, including Excel spreadsheets.
- Traditional spreadsheets and documents lack the functionality to successfully manage the complexity of today's systems engineering initiatives. File-based requirements cannot be individually modified, causing accessibility and version control issues. Files also contain other RFLP entities, and provide inadequate and siloed access to many simultaneous users across the full spectrum of a systems engineering initiative.
- MBSE formalizes a model-based approach, where each RFLP entity is treated separately to provide a single source of information. This creates relationships between these entities, increasing collaboration and participation with systems engineering initiatives. The result is an automated, intelligent relationship management system.
- However, MBSE tools require expert-level knowledge, placing a high skill burden on non-experts. This can lead to systems engineers working in isolation, putting up further communication and accessibility barriers.
- To realize real value from MBSE, organizations can use a spreadsheet interface to provide non-experts with access to the information they need, while engineers continue to develop the model using specialist MBSE tools.