

Model-based Development in Theory: Why should I use it?

Model-based development is an established practice within automotive software development. As part of this, an executable model serves as the core asset in the development of an embedded control system.

This model underpins the entire development process, acting as an early prototype for requirements elicitation, algorithmic design, software implementation, testing, and verification. Code is automatically generated from the model, which is used to build the software that is finally integrated onto the target system.

Boosting efficiency and quality during development

A number of trends are putting a growing level of pressure on developers and engineers in the automotive and automation industries, including:

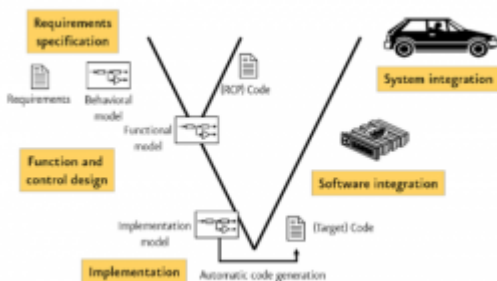
- Shorter development cycles
- Increased software complexity
- A large number of safety requirements and standards
- Growing demand for product customization

These challenges lead to workloads ballooning and compromise efficiency. A model-based approach is designed to streamline development processes and can increase system productivity by 40 percent compared to document-centric approaches (source).

By enshrining functionality in a central development artifact (the model) that can be understood and used across different teams, model-based development facilitates interdisciplinary communication and collaboration and helps prevent misinterpretation of requirements. Beyond this, it makes available a single platform and simulation environment that can be used across many teams, whether distributed or cross-disciplinary, to optimize system design.

This strengthens productivity while driving down development time and cost – the need to manually create code is bypassed, and the risk of manually coded errors is eliminated. In addition, quality assurance is prioritized in the initial stages of the process via early validation of system requirements.

Zeroing in on functionality for better quality assurance



A model-based approach essentially frontloads the development process, placing the lion's share of work in early development stages and keeping system functionality in focus.

It also helps to improve quality assurance measures:

The model is used to simulate and test system functionality, enabling early validation and error detection. Requirements traceability is also enhanced, thanks to a stronger link between software design and

requirements.

On the whole, this leads to a significant improvement in software quality and fewer software errors.

Key benefits

- Increase in productivity
- Decrease in development time and cost
- Better inter-team collaboration
- Code is automatically generated
- Early validation of functionality
- Early error detection
- Stronger link between software design and requirements

Model-based development in practice: The right tools for the right approach

MES has developed a suite of tools to enhance model-based development processes: the MES toolchain. Together, they provide comprehensive support during software development, from design to implementation and verification.



MES Model Examiner[®] conducts automated checks to verify compliance with modeling guidelines for MATLAB Simulink[®]/Stateflow[®], Embedded Coder[®], TargetLink[®], and ASCET models.



MES M-XRAY[®]'s fast and precise structure and complexity analysis gives you complete transparency of your Simulink[®], Stateflow[®], Embedded Coder[®], and TargetLink[®] models.



MES Test Manager[®] perfectly implements requirements-based testing in model-based development.



MES Quality Commander[®] evaluates the quality and product-readiness of your software and delivers key decision-making data throughout the product development lifecycle.

Model-based development in practice: Our customer success stories

Take a look at how some of the world's top car manufacturers, suppliers, and OEMs are using model-based development to their advantage.



Discover how Mercedes-Benz Research & Development North America and MES developed automatically testable compliance rules for modeling with dSPACE TargetLink (PDF).



Find out how Valeo Siemens uses model-based development and the MES Quality Tools to ensure early, systematic, and comprehensive quality assurance in the control software of its electric drive systems.



Gain insight into Renault's model-based development workflow and how they use MXAM to speed up development and improve software quality.