

Mercedes-Benz R&D North America: Erhöhte Sicherheit und Effizienz im Softwareentwicklungsprozess bei Mercedes-Benz

Wie Mercedes-Benz Research & Development North America (MBRDNA) die Modellmetriken von MES* einsetzt, um den modellbasierten Softwareentwicklungsprozess zu verbessern.

Als Forschungs- und Entwicklungszweig eines der renommiertesten Automobilhersteller der Welt ist das Entwicklungsteam von MBRDNA immer bestrebt die effizientesten und effektivsten Methoden und Prozesse bei der Entwicklung sicherer Software für ihre richtungsweisenden Automobilfunktionen, wie den E-Drive-Controller, einzusetzen. Es ist jedoch eine Herausforderung, einem bestehenden Softwareprogramm Funktionalität hinzuzufügen und gleichzeitig das Programm verständlich, wartungsfreundlich und testbar zu halten. Um dieses Ziel zu erreichen, sind auch beim Einsatz von modellbasierter Softwareentwicklung fortschrittliche Tools und Prozesse erforderlich.

Kontinuierliche Verbesserung der Steuerungssoftware: Der Bedarf an Refactoring

Plug-In-Hybrid-Fahrzeugfamilie (Quelle: MBRDNA)

Plug-In-Hybrid-Fahrzeugfamilie (Quelle: MBRDNA)

MBRDNA suchte insbesondere nach Möglichkeiten, die Leistungsfähigkeit der Steuerungssoftware für ihre E-Drive-Komponenten zu verbessern. Das Team mit Sitz in Redford, MI (USA), hat die modellbasierte Entwicklung bereits als Schlüsselkonzept für die Entwicklung eingebetteter Steuerungssoftware definiert. Dabei dient das Softwaremodell als grundlegendes Artefakt im Entwicklungsprozess - die funktionale Integrität und die Abdeckung der beabsichtigten Funktionalität werden in erster Linie auf der Ebene des Softwaremodells verifiziert. Die bestehenden und in der Produktion verwendeten Softwaremodelle werden im Rahmen eines kontinuierlichen Verbesserungsprozesses periodisch erweitert, um immer mehr Funktionalität einzubinden. Das bei MBRDNA in Frage kommende E-Drive-Softwaremodell dient als Grundlage für die Steuereinheit eines elektrischen Antriebssystems. Es steuert Funktionen wie Drehmoment und Zugkraft des elektrischen Antriebs. Diese Controller-Software wird auch für 48V-Systeme, reine E-Drive-Systeme und Brennstoffzellenprogramme verwendet.

Die MBRDNA-Strategie bestand darin, zunächst die Grundfunktionalität der Software zu implementieren. Als komplexere Anforderungen und zusätzliche Projektbeteiligte wie neue Entwickler:innen, Tester:innen und Applikateur:innen, hinzukamen, erkannte MBRDNA die Notwendigkeit, die "Komplexität zu verwalten" und das komplexer gewordene Softwaremodell vergleichsweise leicht verständlich zu halten und die direkte Kommunikation im Team zu fördern. Das Softwaremodell war erheblich angewachsen, und dies wurde zum Hauptgrund für die Überarbeitung des Basismodells. Vor allem das Testteam für die Funktionalität der Softwaremodelle musste oft zusätzliche Anstrengungen zum Testen aufbringen.

Entwicklung eines systematischen Ansatzes für das Refactoring von Softwaremodellen

MBRDNA definierte zunächst eine Reihe von Zielvorgaben für das Refactoring von Softwaremodellen. **Das Hauptziel war die Verbesserung der Testbarkeit und Verständlichkeit des Modells durch die Reduzierung der Komplexität.** MBRDNA beabsichtigte sowohl die Komplexität auf Subsystemebene als auch die Komplexität auf der Ebene des gesamten integrierten Softwaremodells zu reduzieren. MBRDNA erfüllt die Sicherheitsstandards der ISO 26262, so dass für sie entscheidend ist, ihre Einheiten und Modelle gut zu strukturieren und nicht mit übermäßig komplexen Abschnitten zu versehen, da diese beim Testen und bei der allgemeinen Verwendung häufig Probleme verursachen. Übermäßig komplexe Abschnitte müssen in weniger komplexe und kleinere Einheiten aufgeteilt werden - idealerweise mit einer dedizierten Funktionalität pro Subsystem. Ein weiteres Ziel ist die Identifizierung potenzieller Bibliothekselemente, da dadurch die Gesamtkomplexität des Modells weiter reduziert wird. Der Ansatz von MBRDNA besteht darin, die Automatisierung auf das höchste Niveau zu heben. Im Einklang mit diesem Ziel hat MBRDNA nach einem Werkzeug zur Verbesserung der Modellarchitektur gesucht, um die Ergebnisse des Refactorings zu unterstützen und zu überprüfen. Langfristig sollte geprüft werden, ob ein solches Tool dazu beitragen kann, die kontinuierliche Kontrolle und Optimierung der Modellarchitektur als Teil des definierten Softwaredesignprozesses zu unterstützen.

Das richtige Tool für diese Aufgabe finden

Zur Unterstützung des Modell-Refactoring-Projekts wurden die Modellmetriken von MES* ausgewählt. Diese berechnen die lokale Komplexität von Softwaremodellen für jedes Subsystem und liefern auch die globale Modellkomplexität. Diese Funktion weist Benutzer:innen auch auf so genannte Hotspots des Modells hin. Hotspots sind Subsysteme des Modells, die überarbeitet und refaktorisiert werden sollten. Um die Wirkung eben dieser Funktion weiter zu optimieren, nutzte MBRDNA deren Flexibilität und passte sie an ihre spezifische Umgebung an. Insbesondere fügte MBRDNA spezifische Funktionsblöcke hinzu und bestimmte benutzerdefinierte Skalierungsfaktoren in der Konfiguration der Modellmetriken von MES*. Aufgrund dieser einfachen Konfigurierbarkeit konnte mithilfe von MES eine breite Palette von Metriken geliefert werden, die problematische Teile der MBRDNA-Modelle genau identifizieren. Sowohl qualitative als auch quantitative Analyse sind enthalten. Die quantitative Analyse umfasst die globale Gesamtkomplexität des Softwaremoduls. Diese Informationen werden im "Re-Architecting-Prozess" und für die weitere Modularisierung von Software benötigt, bei der die globale Komplexität genutzt wird, um zu bestimmen, welches überdimensionale Softwaremodul in mehrere Softwaremodule aufgeteilt werden soll. Die qualitative Analyse lieferte MBRDNA-Informationen über fehleranfällige Hotspots innerhalb eines bestimmten Softwaremoduls. Dabei interessiert sich MBRDNA insbesondere für die lokale Subsystemkomplexität und die Frage, wie die Komplexität auf intelligente Weise begrenzt werden kann.

Angewandte Remodellierungstechniken

Auf der Grundlage der vom MES-Tool gelieferten Metriken entwickelte MBRDNA eine Reihe von Techniken zum Refactoring der identifizierten Softwaremodellteile und wandte diese an:

- Systematische Aufteilung übermäßig komplexer Subsysteme in eine Reihe weniger komplexer, funktionell kohärenter Subsysteme.
- Verwendung der Clone Group Detection von MXAM/MXRAY* zur Ermittlung redundanter Elemente/Blöcke im Modell (z. B. Blöcke für Polynome), die wiederholt verwendet werden. Diese redundanten Elemente werden dann zur MBRDNA-spezifischen Blockbibliothek oder zu einem referenzierten Modell hinzugefügt. Die Grundidee ist hier, redundante Modellkomponenten zu vermeiden, die einzeln geprüft und getestet werden müssen.

Positive Ergebnisse mit den Modellmetriken von MES*

Humphrey Achiri, Senior Developer bei MBRDNA
Humphrey Achiri,
Senior Developer bei
MBRDNA

Die Analyse durch die Modelletriken von MXAM/MXRAY* zeigt fehleranfällige Hotspots innerhalb eines Softwaremoduls mit hoher Genauigkeit an. Laut Humphrey Achiri, Senior Developer bei MBRDNA, **"hat dies die allgemeine Lesbarkeit, Testbarkeit und Wartbarkeit unserer Softwaremodule erheblich verbessert"**. Darüber hinaus wiesen die Applikateur:innen von Mercedes-Benz darauf hin, dass "die **Navigation** durch die fertige Software während der Fahrzeugerprobung **viel einfacher geworden** ist, bei der die Zeit auf den Teststrecken eine Einschränkung darstellt". Für die Testingenieur:innen von Mercedes-Benz ist es auch bequemer geworden, MIL- und SIL-Testfälle (**d.h. einfacheres anforderungsbasiertes Testen**) zu erstellen, da die einzelnen Anforderungen besser auf die tatsächliche Umsetzung in bestimmten Teilsystemen abgestimmt wurden. Außerdem ist die **Dokumentation** der Software, dank der weniger komplexen einzelnen Subsysteme, **einfacher geworden**.

MBRDNA konnte das Hauptziel, die Verbesserung der Lesbarkeit, Testbarkeit und Wartbarkeit seiner Softwaremodule für die neue Generation von Hybrid- und vollelektrischen Fahrzeugen, erreichen. Auch die Dokumentation der Software wurde durch die Reduzierung der komplexen einzelnen Subsysteme vereinfacht. Zu den numerischen Ergebnissen, die während des Refactorings erzielt wurden, gehören:

- Die globale Komplexität, ein Maß für die Verständlichkeit und Lesbarkeit des gesamten Softwaremodells, wurde **um etwa 10% reduziert**. Dies bedeutet eine erhebliche Reduzierung des Aufwands, der für das Verständnis, die Wartung, das Testen und die Implementierung des refaktorierten Softwaremodells erforderlich ist.
- **Eine erhebliche Reduzierung der lokalen Komplexitätswerte wurde erreicht**, und bei einigen Softwaremodulen wurde die Anzahl der komplexen Subsysteme auf Null reduziert.

Ergebnisse, die bei den Softfacts erzielt wurden:

- Anmerkung des Requirements-Engineering-Teams: **"Wow - diese Softwaremodelle sind jetzt viel einfacher zu verstehen und zu bearbeiten."** Mit der zusätzlichen Bedeutung des Modells für die 48V-Systeme von MBRDNA, reine E-Drive-Systeme und Brennstoffzellenprogramme, hat sich die Beherrschung der Komplexität auch für das Variantenmanagement als vorteilhaft erwiesen.

Außerdem hat sich der automatisch generierte Code nicht geändert, da die im Modell beschriebene Funktionalität nicht verändert wurde.

Angesichts der positiven, spürbaren Ergebnisse, der Prozessverbesserungen und des positiven internen Feedbacks ist der Aufwand, den MBRDNA für die Einrichtung der MES-Tools und die Verbesserung des Softwaremodells betrieben hat, sehr gut investiert worden. Das Feedback und die tatsächlichen konkreten Ergebnisse des Refactoring-Projekts sind sehr gut.

Ausblick auf das Komplexitätsmanagement bei MBRDNA

Alexander Dolpp, Director E-Drive-Software bei MBRDNA
Alexander Dolpp,
Director E-Drive-Software
bei MBRDNA

In Zukunft werden die Modellmetriken von MXAM* die kontinuierliche Modelloptimierung und das Refactoring bei MBRDNA unterstützen. Die Prozessrichtlinien für die Entwicklung und das Refactoring großer Softwaremodelle für Controller-Funktionen bei MBRDNA werden durch die Verwendung der MES-Tools eine Reihe von Prüfungen und Metriken als teamweite Best Practice zur Überwachung der

Modellarchitektur über den gesamten Entwicklungsprozess hinweg vorsehen. MBRDNA plant außerdem, das Tool in seinen hochautomatisierten Continuous-Integration-Build-Prozess zu integrieren, der auf der Jenkins-Technologie basiert. Mit weiteren Modellmetriken, wie z.B. die Inkohärenz- und Schnittstellen-Effizienzmetriken, "kann der Prozess zur kontinuierlichen Modellverbesserung noch weiter rationalisiert werden", resümiert Alexander Dolpp, Director E-Drive Software bei MBRDNA.

* Modellmetriken waren ursprünglich Bestandteil von MES M-XRAY (MXRAY). Die MXRAY-Funktionen sind inzwischen in den MES Model Examiner (MXAM) integriert und werden hier kontinuierlich weiterentwickelt.