

Modellbasierte Softwareentwicklung mit SCADE für das sichere Übertragungssystem Vital 21

Paul Linder / Falk Sbiegay

Die Informatik Consulting Systems AG (ICS AG) hat die Software für das sichere Übertragungssystem Vital 21 der Thales Rail Signalling Solutions GmbH mithilfe eines modernen modellbasierten Entwicklungsansatzes unter Verwendung des Softwareentwicklungswerkzeugs SCADE der Fa. Esterel Technologies SA entwickelt. Vital 21 ist das erste durch das Eisenbahnbundesamt zugelassene SIL 4 System, dessen Software modellbasiert mit SCADE entwickelt wurde. Der gewählte Entwicklungsansatz zeichnet sich durch hohe Softwarequalität und geringe Fehlerfindungsrate im Systemtest und Feldtest aus, wodurch kostenintensive Entwicklungsiterationen eingespart werden konnten.

1. Einführung

Die normenkonforme Softwareentwicklung für Systeme der Sicherheitsanforderungsstufe SIL 4 gemäß der Norm EN 50128 erfordert besonderes Augenmerk auf die Sicherheit der zu entwickelnden Produkte. Daneben sind aber auch Gesichtspunkte der Systemperformance sowie eines effizienten und beständigen Softwareentwicklungsvorgehens zu beachten. Um diesem Spannungsfeld bei gleichzeitiger Verbesserung der wirtschaftlichen Bedingungen Rechnung zu tragen, wurde von der ICS AG ein moderner modellbasierter Entwicklungsansatz unter

Verwendung des Softwareentwicklungswerkzeugs SCADE zur Erstellung der Software für das sichere Übertragungssystem Vital 21 angewendet.

Der vorliegende Artikel stellt zunächst das sichere Übertragungssystem Vital 21 und deren Software vor. Anschließend wird eine kurze Einführung in SCADE gegeben. In Abschnitt 4 wird auf die Gestaltung des Softwareentwicklungsprozesses unter Verwendung von SCADE eingegangen. Abschnitt 5 stellt praktische Erfahrungen mit SCADE dar. Der Beitrag schließt mit einem Fazit über den Einsatz von SCADE bei der Softwareentwicklung für Vital 21.

2. Das sichere Übertragungssystem

Vital 21 und dessen Software

Vital 21 ist ein neues Übertragungssystem der Thales Rail Signalling Solutions GmbH, welches die sichere Fernübertragung von bis zu 32 Relaiskontakten über Kupferleitungen erlaubt [1]. Die beiden Gegenstellen eines Vital 21-Systems sind modular aus bis zu vier Doppelrechnersystemen auf Basis von Mikrocontrollern aufgebaut (vgl. Abbildung 1). An den Relais-Kontakten können Impulse von wenigen 100 ms Dauer anliegen, welche unter Einhaltung von Verzerrungstoleranzen sicher zu übertragen sind. Für das System gilt die Sicherheitsanforderungsstufe SIL 4.

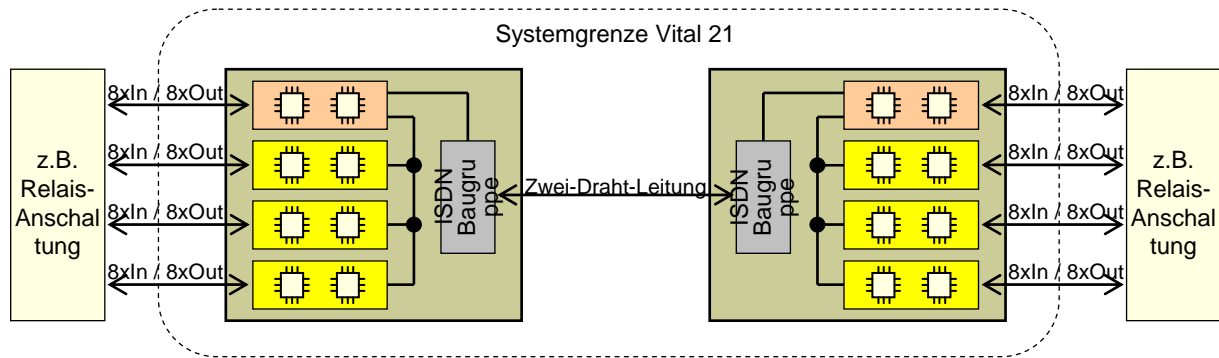


Abbildung 1: Prinzipieller Aufbau VITAL 21

Die Steuerungssoftware für VITAL 21 muss nachfolgende Aufgaben realisieren.

- Sichere Datenübertragung zwischen den Gegenstellen gemäß der Norm EN 50159-1
- Sichere Datenübertragung zwischen den Doppelrechnersystemen einer Gegenstelle gemäß der Norm EN 50159-1
- Sichere Erfassung und Ausgabe von Relaisignalen
- Zyklische Durchführung von Onlinetests und Prozesssicherung im Fehlerfall
- Bereitstellung einer Diagnoseschnittstelle zum Auslesen von Fehlern

Die Softwareentwicklung muss den Vorgaben der Norm EN 50128 an Software für Systeme der Sicherheitsanforderungsstufe SIL 4 genügen.

3. Modellbasierte Softwareentwicklung mit SCADE

3.1 Das Softwarewerkzeug SCADE

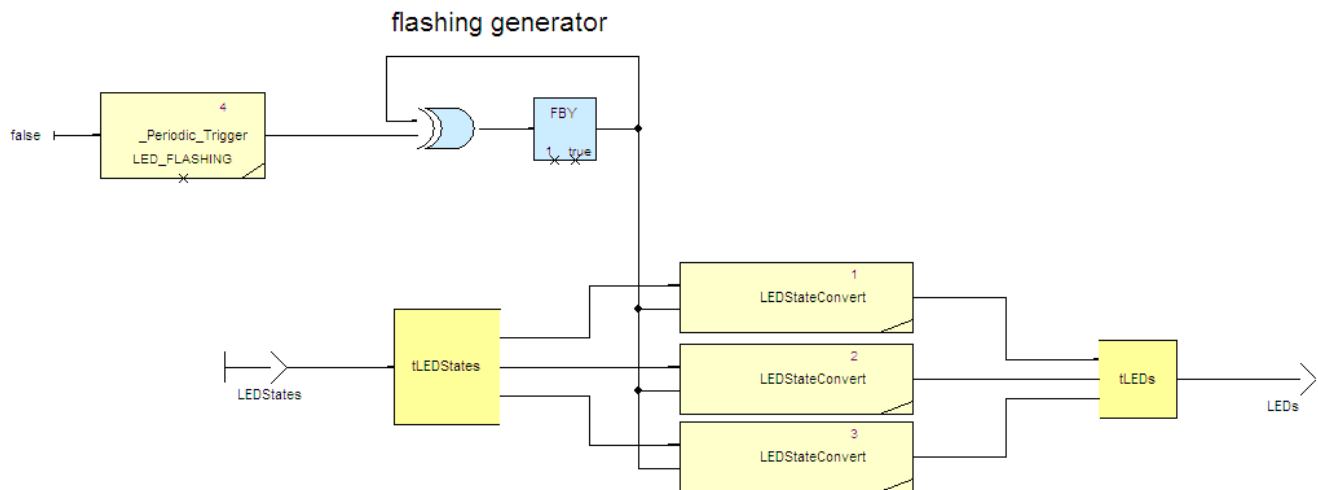
SCADE [1] ist ein modellbasiertes Softwareentwicklungswerkzeug der Fa. Esterel Technologies SA, welches aus dem europäischen Luffahrtumfeld stammt und zur Entwicklung sicherheitskritischer Software

eingesetzt wird. Typisches Einsatzgebiet von SCADE ist die Entwicklung von sicherheitskritischer eingebetteter Steuerungs- und Regelungssoftware wie beispielsweise für Flugsteuerungen.

Die SCADE Suite umfasst neben der Modellierungs- und Simulationsumgebung auch einen nach DO178B, EN50128 und IEC61508 zertifizierten Codegenerator, Werkzeuge zur Modellanalyse hinsichtlich struktureller Testabdeckung und formaler Verifikation, Schnittstellen zu diversen Modellierungswerkzeugen wie MathWorks MATLAB/Simulink oder Telelogic Rhapsody sowie ein Requirements Management Gateway mit Anbindungsmöglichkeit an Requirements Engineering- und Testverwaltungswerkzeuge

3.2 Modellierung und Simulation mit SCADE

Die Modellierung mit SCADE beruht auf den formalen synchronen Sprachen Esterel und LUSTRE [3]. SCADE-Modelle weisen somit ein deterministisches zeitdiskretes Verhalten auf, d. h. sämtliche Berechnungen erfolgen taktgesteuert, wie man es von digitalen Schaltungen her kennt.



Anstelle einer textbasierten Modellierung in LUSTRE erfolgt die Modellierung mit SCADE grafisch mithilfe von Datenflussdiagrammen (vgl. Abbildung 2) und Zustandsmodellen. Dabei kann auf vorgefertigte und selbstdefinierte Modellbibliotheken zurückgegriffen werden sowie importierte Daten bei Bedarf eingebunden werden. Die Modelle weisen streng typisierte Datenflüsse auf und lassen sich hierarchisch strukturieren. SCADE ermöglicht hierdurch eine komfortable Modellierung mit bekannten grafischen Darstellungsmitteln, ohne Kompromisse bei der Eindeutigkeit und Konsistenz der Modelle einzugehen.

Abbildung 2: Beispielhaftes Datenflussdiagramm in SCADE

Zur Modellprüfung lässt sich die Modellkonsistenz statisch analysieren, ausgewählte Modelleigenschaften formal verifizieren sowie das Modell in der Simulation testen. Die Überprüfung der Testabdeckung von Modelltests erfolgt dabei mit der Überdeckungsmetrik MTC (Model Test Coverage). MTC überprüft anhand von Überdeckungsmaßen beispielsweise der Decision Coverage (DC)

oder der Zustandsüberdeckung, welche Funktionsblöcke bzw. Operatoren bei der Simulation ausgeführt werden. Des Weiteren lassen sich mit dem SCADE Requirements Management Gateway Modellelemente mit Anforderungen aus DOORS oder Entwicklungsdokumenten verknüpfen und anschließend die Anforderungsüberdeckung der Modelle analysieren.

3.3 Zertifizierte Codegenerierung

Einer der wesentlichen Vorteile der modellbasierten Softwareentwicklung mit SCADE liegt in der zertifizierten C-Code-Generierung. Der bei VITAL 21 eingesetzte Codegenerator KCG 5.1.1 ist dabei für die höchsten Sicherheitseinstufungen sowohl im Luftfahrtbereich (DO-178B Level A) als auch im Bahnumfeld (EN 50128 SIL 4) zertifiziert. Dies macht den Einsatz von SCADE im Umfeld sicherheitsrelevanter Anwendungen attraktiv, da der aus den SCADE-Modellen generierte C-Code nicht gegenüber dem Modell verifiziert werden muss und die Modulfunktionalität auf Modellebene in der Simulation geprüft werden kann.

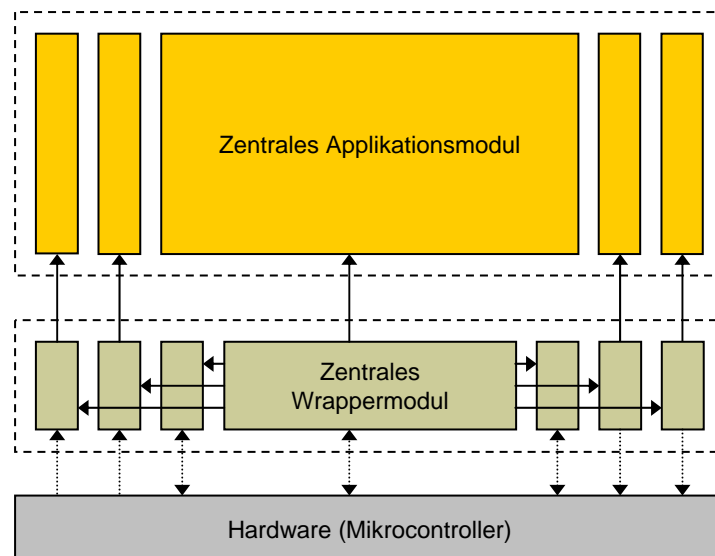
4. Gestaltung des Entwicklungsprozesses

Zur Softwareentwicklung für Vital 21 wurde bei der ICS AG der Ansatz gewählt, eine maximale Applikationserstellung mit SCADE mit einem möglichst schlanken, in C programmierten „Wrapper“ zur Hardwareanbindung zu kombinieren (vgl. Abbildung 3). Der Funktionsumfang der Applikationsmodule machte dabei ca. 80% des Gesamtprogramms aus, der Funktionsumfang der Wrapper entsprechend ca. 20%.

Der Entwicklungsprozess teilt sich dabei nach dem Softwarearchitekturentwurf in eine konventionelle Modulentwicklung des Wrappers und eine modellbasierte Modulentwicklung der Applikation auf Softwareanforderungsspezifikation, Softwarearchitekturentwurf, Integrationstest und Systemtest werden weiterhin konventionell durchgeführt. Dem Softwarearchitekturentwurf fällt die zentrale Aufgabe zu, die Schnittstellen

Modellbasiert mit SCADE entwickelte Applikationsmodule

Konventionell entwickelte Wrapper-Module



und das Zusammenspiel zwischen den modellbasierten Applikationsmodulen und den hardwarenahen, konventionell entwickelten Wrappermodulen zu entwerfen.

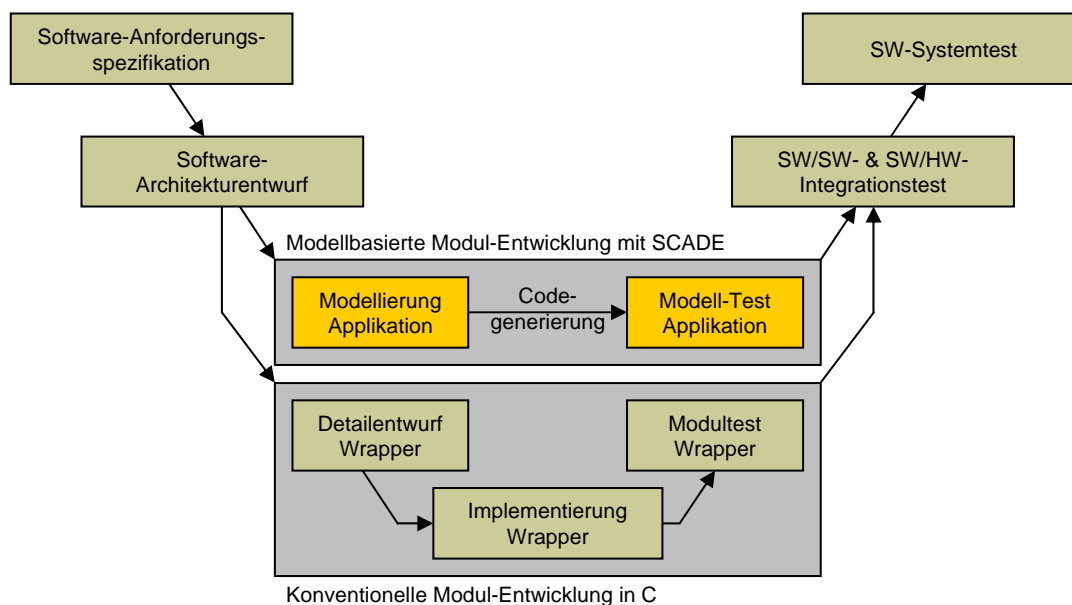
Der konzipierte Entwicklungsprozess hat aufgrund seiner Ähnlichkeit zum konventionellen

Abbildung 3: Prinzipielle Architektur der Software für Vital 21

Als Entwicklungsvorgehen wurde in Anlehnung an [3] das in Abbildung 4 dargestellte hybride V-Modell konzipiert.

V-Modell den Vorteil einer einfachen und transparenten Einbindung in bewährte CENELEC SIL 4 Entwicklungsprozesse. Dies ist insbesondere im Kontext von Validierung, Begutachtung und Zulassung der Software von großer Bedeutung.

Abbildung 4: Softwareentwicklungsprozess für Vital 21



5. Praktische Erfahrungen

5.1 Softwareentwicklung

Durch den Einsatz der zertifizierten Codegenerierung konnte bei der Entwicklung der Applikationsmodule eine Effizienzsteigerung von ca. 50% im Vergleich zur konventionellen Entwicklung der Wrappermodule erzielt werden. Dabei muss jedoch beachtet werden, dass die Modulentwicklung lediglich einen begrenzten Anteil des Gesamtentwicklungsaufwandes eines SIL 4 Projekts ausmacht.

Ein weiterer wichtiger Aspekt ist der Sachverhalt, dass sich bei der modellbasierten Entwicklung Aufwendungen zu früheren Zeitpunkten hin verschieben. So ist die mit Hinblick auf die Codegenerierung erforderliche präzise und umfassende Modellierung der Applikationsmodule in SCADA entsprechend aufwendiger als das Erstellen von Detailentwurfsdokumenten. Betrachtet man jedoch die gesamte Sequenz aus Entwurf, Implementierung und Modultest,

ergibt sich die genannte Effizienzsteigerung. Schließlich ist die hohe Qualität der mit SCADA entwickelten Applikationsmodule zu betonen, welche sich sehr positiv auf den Gesamtprojektverlauf auswirkte.

5.2 Software-Verifikation und Validierung

Die klaren Schnittstellen zwischen modellbasierter und konventioneller Softwareentwicklung (vgl. Abbildung 4) ermöglichten eine einfache und transparente Einbindung modellbasierter Techniken zur Überprüfung der Applikationsmodule in die Verifikations- und Validierungsstrategie. Anforderungsprüfung, Architekturprüfung, Wrapper-Modultest, Integrationstest und Systemtest erfolgten weiterhin mit bewährten konventionellen Techniken. Der Modultest der Applikationsmodule wurde auf Modellebene in der SCADA-Simulationsumgebung unter

Verwendung der modellbasierten Überdeckungsmetrik MTC durchgeführt. Da aufgrund der Zertifizierung der Codegenerierung der generierte Code nicht gegenüber den Modellquellen geprüft werden muss, kann der Modelltest den Modultest auf Codeebene vollständig ersetzen. Bei der Spezifikation der Modelltests ist allerdings zu beachten, dass im Gegensatz zu konventionellen Modultests die Modelltestspezifikation von der Software-Architektur abgeleitet werden muss.

Ergänzend zum Modelltest der Applikationsmodule wurde die Konsistenz der Modelle werkzeuggestützt analysiert sowie die Einhaltung gegenüber aufgestellten Modellierungsregeln manuell geprüft. Die Modellierungsregeln fordern beispielsweise eine ausreichende Kommentierung der Modelle oder untersagen die Verwendung bestimmter Modellelemente, die bei der Zertifizierung des Codegenerators ausgeschlossen wurden.

5.3 Projektabwicklung

Die Einbindung des modellbasierten Entwicklungsansatzes in bewährte Prozesslandschaften ermöglicht es, Verifikations-, Validierungs-, RAMS- und Qualitätsmanagementprozesse weitgehend unverändert zu übernehmen, so dass Umstellungs- und Einarbeitungsaufwendungen gering gehalten werden konnten.

Stärker noch als die Effizienzsteigerung bei der Entwicklung des Modells war die hohe Softwarequalität der mit SCADE entwickelten

Applikation von besonderer Bedeutung für die Projektabwicklung. Zum Einen wurde hierdurch der Verifikations- und Validierungsaufwand für die Applikation reduziert. Noch wichtiger war jedoch, dass kaum Fehler im Systemtest und keine Rückläufer aus dem Feldtest auftraten, wodurch kostenintensive Entwicklungsiterationen und Bugfixes eingespart werden konnten.

6. Fazit

Die Entwicklung der Software für Vital 21 unter Verwendung von SCADE konnte die gestellten Erwartungen in hohem Maß erfüllen. Zusammenfassend sind dabei folgende Aspekte hervorzuheben:

- *Qualität:* Die geringe Fehlerfindungsrate im Rahmen der Integrations-, System- und Feldtests zeugt von der hohen Softwarequalität, die mit SCADE entwickelte Applikationen erreichen können.
- *Effizienz:* Aufgrund der hohen Softwarequalität konnten kosten- und zeitintensive Entwicklungsiterationen und Bugfixes eingespart werden. Hierdurch konnte der Gesamt-Systementwicklungszyklus beschleunigt werden.
- *Wartbarkeit:* SCADE Modelle weisen eine im Vergleich zu herkömmlicher Software erhöhte Transparenz und Lesbarkeit auf und sind einfach zu ändern. Dadurch wird die Wartbarkeit der Software verbessert.

Schließlich ist noch die positive Resonanz der Projektmitarbeiter auf SCADE zu erwähnen, wodurch Einführungsaufwendungen schnell kompensiert werden konnten.

Literatur

[1] Rahn, W.° : Vital 21: Ein System zur sicheren Übertragung digitaler Information. SIGNAL + DRAHT (101) 05/2009.

[2] Esterel Technologies SA: *SCADE Suite Reference and User Manuals, Version 5.1.1.* Esterel Technologies SA, 2006.

[3] Esterel Technologies SA: *Efficient Development of Safe Railway Applications Software with EN 50128 Objectives using SCADE Suite.* Esterel Technologies SA, 2006.

Summary

The Informatik Consulting Systems AG (ICS AG) has developed the software for the Safe Transmission Unit Vital 21 of the Thales Rail Signalling Solutions GmbH applying a modern model-based software development approach using the software tool SCADE. Vital 21 is the first SIL 4 system with approval of the Eisenbahnbundesamt (federal railway authority), whose software has been developed with SCADE. Benefits of the developed approach chosen have been a high software quality and few errors found during system test and field test, hereby reducing cost-intensive development iterations.