

INFORMATIK 2020 – Bericht der Session zum Thema Software Engineering

Steffen Becker ¹, Hubert B. Keller ²

Abstract: Dieser Artikel enthält einen kurzen Bericht über die drei Vorträge in der Session „Software Engineering“ der virtuellen INFORMATIK 2020 Tagung. Innerhalb der Session gab es drei Fachvorträge, die über ein Videokonferenzsystem vorgetragen wurden. Die Vorträge fanden interaktiv statt, Fragen konnten jederzeit in den Vortrag hereingegeben werden.

Keywords: Bericht; INFORMATIK 2020; Software Engineering; virtuelle Konferenz

1 Einleitung

Die Session „Software Engineering“ fand am Dienstag, 29.09.2020, von 12:45 - 14:15 Uhr mit 3 Vorträgen statt. Chairs der Sitzung waren Hubert B. Keller, Institute for Automation and Applied Informatics (IAI), Karlsruhe Institute of Technology (KIT), sowie Steffen Becker, Institute of Software Engineering, University of Stuttgart.

Die drei Vorträge waren

- „Safe and Secure Software Engineering - the SPARK Approach“ von Yannick Moy, AdaCore, Paris
- „Agile Organisation und Führung innerhalb der Softwareentwicklung: Wie geht das zusammen?“ von Andreas Störzbach, Trumpf, Deutschland und
- „Wie Cloud-native-Ansätze Architektur(arbeit) verändert“ von Thomas Franz, adesso.

2 „Safe and Secure Software Engineering - the SPARK Approach“, Yannick Moy, AdaCore, Paris

Yannick Moy ist Senior Software Engineer bei AdaCore und verantwortet die Entwicklung von SPARK. Als Co-Direktor leitet er das Labor ProofInUse. Er hat an der Entwicklung

¹ Softwarequalität und -architektur (SQA), Institut für Software Engineering, Universität Stuttgart, Universitätsstr. 38, 70569 Stuttgart, steffen.becker@iste.uni-stuttgart.de,  <https://orcid.org/0000-0002-4532-1460>

² Fachgebiet Advanced Automation Technologies (A2T) und Arbeitsgruppe Reliability, Safety & Security of Software and Systems (RS4), Institute for Automation and Applied Informatics (IAI), Karlsruhe Institute of Technology (KIT), Hermann-von-Helmholtz-Platz 1, 76344 Eggenstein-Leopoldshafen, hubert.keller@kit.edu,  <https://orcid.org/0000-0001-9916-5283>

der Software-Quellcode-Analysatoren CodePeer, Frama-C und PolySpace Verifier C ++ mitgewirkt. Bei AdaCore arbeitet er an den Software-Quellcode-Analysatoren CodePeer und SPARK zur Erkennung von Fehlern, zur Überprüfung von Sicherheitseigenschaften und zu formalen Spezifikation. Daneben hat er zahlreiche Publikationen über formale Methoden und die SPARK Technologie veröffentlicht.

Yannick Moy ging zuerst auf den Zusammenhang zwischen Software ,Qualität und Zuverlässigkeit ein. Je geringer die Wahrscheinlichkeit von Fehler in Software ist, um so höher ist die zu erwartende Zuverlässigkeit. Eigentlich ist schon Software mit nur einem Fehler unzuverlässig. Ziel ist es also so weit wie möglich in Richtung „zero-defect“ Software zu kommen. SPARK ist ein formaler Spezifikationsansatz aus den Bereichen Avonic, Militär und Schienensysteme, der mittlerweile seinen Weg auch in den Medizinbereich, in Automotive Software und insbesondere in den Security Bereich gefunden hat. SPARK ermöglicht über 5 Ebenen Security Anforderungen abzubilden und abzusichern. Neben der Spezifikationssprache existieren entsprechende Tools für den praktischen Einsatz. Yannick Moy zeigte wie durch formale Spezifikationen Seiteneffekte durch globale Variablen verhindert oder Vor- und Nachbedingungen angegeben werden können sowie welche Toolketten dazu existieren. SPARK baut auf der Sprache Ada mit den Kerneigenschaften der strengen Typisierung etc. auf und erweitert diese um formale Konstrukte, z.B. um Daten- und Flussabhängigkeiten zu analysieren, den Nachweis der Vermeidung von Laufzeitfehlern wie die Verletzung des Priority Ceiling Protocols usw. sicher zu stellen etc. Je nach Umfang der formalen Spezifikation werden 5 Software Assurance Levels erfüllt. Die Anwendung im industriellen Umfeld zeigte er an mehreren Beispielen mit der Angabe des Aufwandes diese Technologie in Projekten zu implementieren. Als Ergebnis des Aufwandes nannte er eine deutlich geringere Zahl an Software Fehlern, einen geringeren Aufwand beim Testen und letztlich eine enorme Einsparung im gesamten Projekt. SPARK 2014 erlaubt dabei ausführbare formale Spezifikationen und geht dabei weit über MISRA C hinaus. Für den Einsatz von SPARK sind viele Dokumentationen verfügbar (siehe www.adacore.com).

3 „Agile Organisation und Führung innerhalb der Softwareentwicklung: Wie geht das zusammen?“, Andreas Störzbach, Trumpf, Deutschland

Andreas Störzbach von der Trumpf GmbH erläuterte in seinem Vortrag das Spannungsfeld zwischen Softwareentwicklung für langlebige, qualitativ hochwertige Produkte (in Fall von Trumpf meist komplexe Werkzeugmaschinen) und agiler Softwareentwicklung. Insbesondere die Spannungen, die durch eine klassische Organisation und Führung eines führenden Maschinenbauunternehmens mit der flexiblen und agilen Struktur bei der modernen Softwareentwicklung entstehen, standen im Zentrum des Vortrags.

Dabei hat Herr Störzbach nicht nur das Problem gut verständlich präsentiert, sondern den anwesenden Zuhörern auch detaillierte Einblicke in die Lösungsansätze bei Trumpf im allgemeinen und seinem Team im Besonderen gegeben.

Das Interesse der Zuhörer galt dabei insbesondere den Fragen der Unternehmenskultur. Die Frage, wie es ein Unternehmen schafft, die agile Struktur und Denkweise seinen Mitarbeitern zu vermitteln und vorzuleben, stand im Kern vieler Fragen der Zuschauer. Im Detail waren Themen zur klassischen Verteilung der Organisationsaufgaben (z.B. Matrixorganisation vs. Agile Projektteams) von großem Interesse. Wir hatten den Eindruck, dass die Zuhörer insbesondere von den Erfahrungen, die Herr Störzbach zu schildern hatte, viel profitiert haben.

4 „Wie Cloud-native-Ansätze Architektur(arbeit) verändert“, Thomas Franz, adesso, Deutschland

Thomas Franz von der adesso SE hat aus seiner Praxis heraus dargestellt, wie sich die Arbeit der Architekten durch Cloud-native Ansätze verändert hat. Dazu hat er verglichen, wie früher klassische Software gebaut und entworfen wurden, gegenüber der Vorgehensweise bei Anwendungen, die für die Cloud und ihre Eigenheiten optimiert werden.

Insbesondere die Aufteilung in kleinere Services, die unabhängig entwickelt, skaliert und betrieben werden können, führt hier doch zu deutlichen Veränderungen. Ein spezieller Aspekt dabei für den Architekten ist der, dass nun auch die Infrastrukturaspekte mit bedacht werden müssen. Dabei kommt zum Tragen, dass Architekten Infrastruktur wie Code behandeln können und damit effizient Zugriff auf große Ressourcenpools haben. Der Betrieb erfolgt in Cloud-Umgebungen und daher muss die Anwendung nicht mehr beim Kunden installiert und betrieben werden.

Aus dem Publikum kamen denn auch neben einigen technischen Detailfragen die Frage auf, ob nicht auch das Geschäftsmodell von adesso sich dahingegen wandelt, dass sie direkt für den Kunden Lösungen anbieten und sogar betreiben können, was in Einzelfällen wohl auch schon der Realität entspricht.

5 Zusammenfassung

Die Session hat ausgehend von dem formalen Ansatz mit SPARK über die Problematik einer agilen Organisation der Softwareentwicklung über die neuen Ansätze des Cloud-basierten Service Angebots gezeigt, dass die Konzepte und Methoden der Softwareentwicklung nicht stehen bleiben. Diese Entwicklungen integriert zusammen zu bringen, um Software kostengünstig in komplexen Organisationen unter Nutzung von, auch Cloud-basierten, Services den Anforderungen von Security, Safety und hoher Zuverlässigkeit zu entwickeln, wird eine weitere Herausforderung der Zukunft sein.

Steffen Becker und Hubert B. Keller