

Konsistente Projektentwicklung mit schnell wechselnden Programmierteams ohne Vorwissen

Studiengang: BSc in Informatik | Vertiefung: Mobile Computing

Betreuer: Dr. Reto E. Koenig

Experte: Dr. Joachim Wolfgang Kaltz

«Ist die Umsetzung auch ineffizient, ist die Umsetzung dafür effizient.» – der Trade-off bei Umstellung einer monolithischen Architektur auf Microservices.

1

Ausgangslage

Die Höhere Fachschule für Technik Mittelland (HFTM) nimmt seit einigen Jahren mit dem Team «Solidus» am internationalen RoboCup in der «Logistics League» mit Erfolg teil. In der «Logistics League» geht es darum, mit selber entwickelten Roboter (genormte Plattform) in einer Produktionsanlage (Smart Factory) die Zwischenschritte – also die Logistik – zu automatisieren. Der Roboter bekommt beispielsweise den Auftrag aus dem Lager zur Anlage 1 Material zu bringen, oder von Anlage 2 die produzierten Teile abzuholen und ins Lager oder zu einer weiteren Produktionsanlage zu bringen. Die Roboter-Software ist mittlerweile relativ ausgereift für den aktuellen Cup. Das Team besteht jedes Jahr aus neuen Studenten die keinen vertieften Hintergrund in der Softwareentwicklung haben. Sie übernehmen die bestehende Code-Base vom vorherigen Team und müssen diese an die neuen Gegebenheiten anpassen und erweitern. Die historisch gewachsene, monolithische Architektur der Software wirkt diesen Anforderungen aber entgegen.

Ziel

Umsetzung einer Softwarearchitektur und Entwicklungsumgebung, welche die Anforderungen an kurze Einarbeitungszeit und Flexibilität erfüllt. Fordert die geringere Effizienz der Umsetzung zwar mehr Ressourcen auf der Zielplattform, steigert sie dafür aber die Effizienz bei der Entwicklung.

Umsetzung

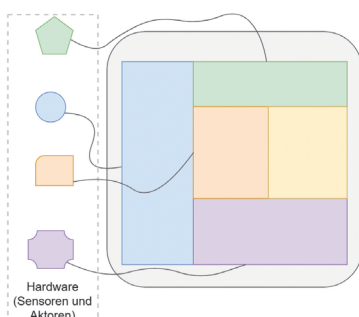
In einer zeitgemässen Tool-Chain für die Entwicklung und einer Architektur basierend auf Microservices wurde die Anbindung eines Sensors (LiDAR) realisiert. Die einzelnen Schritte wurden so dokumentiert, dass sie in Schulungsunterlagen Verwendung finden können. So kann sichergestellt werden, dass zukünftige Entwicklungen voneinander abgekapselt in eigenständigen Microservices stattfinden können und somit keine direkte Abhängigkeit untereinander aufweisen. Die Kopplung und Datenkonversion der einzelnen Services wird mit Hilfe von domänenspezifischen Verbindern (Servants) realisiert.

Resultat und Ausblick

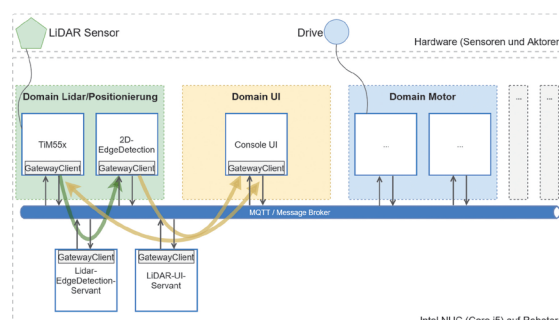
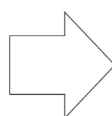
Die exemplarische Entwicklung aus der Umsetzung zeigt, wie man künftig an der HFTM die Optimierung und Weiterentwicklung der Software angehen könnte. Das resultierende Dokument zu dieser Arbeit wurde so erfasst, dass es den zukünftigen HFTM-Studenten als Leitfaden dienen kann. Die Ausgangslage und die Umsetzung erlauben einen fließenden Übergang in die neue Architektur.



Marco Kilchhofer



Alte Architektur (Schaubild)



Neue Architektur