

Plattform zur Verarbeitung schneller Daten

Studiengang: MAS Information Technology

Architekturen sind zunehmend geprägt von Anforderungen an Effizienz, Skalierbarkeit, Fehlertoleranz und Time-to-Market. Die Master Thesis liefert Grundlagen in Form einer Architekturstudie und eines Proof of Concept (PoC) welche der Firma Intersys ermöglichen, zukünftig Projekte im Big Data, IoT und Cloud Umfeld kosteneffizient und in hoher Qualität durchführen zu können.

1

Umfeld

Die Firma Intersys AG ist als Anbieter von Individualsoftware und Produkten stark an Trends und neuen Technologien interessiert, um ihre Kunden optimal zu unterstützen und neue Geschäftsfelder zu erschliessen. Der Markt entwickelt sich dabei vermehrt in Richtung Cloud und Kunden fordern Antworten auf Themen wie Big Data, IoT und DevOps. Die Master Thesis unterstützt die Bestrebungen der Firma im Wissensaufbau, um modulare, flexible und cloudbasierte Applikationsplattformen aufbauen zu können.

Ziel

Ziel der Thesis ist es, eine Architektur zu entwickeln, welche ein breites Einsatzspektrum in den Gebieten Big Data, IoT, DevOps und Cloud abdecken kann, so dass zukünftige Projekte kosteneffizient und in hoher Qualität durchgeführt werden können. Die verschiedenen Technologien, welche die Firma in der Vergangenheit bei Grossprojekten in der Telekommunikationsbranche mit hochverfügbaren, hochperformanten und verteilten System eingesetzt hat, sollen dabei in die Architektur mit einbezogen werden. Die Architektur muss flexibel ausgelegt und skalierbar sein. Die Modularisierung und der Einsatz von Standard Technologien soll dabei helfen, Ressourcen flexibel einzusetzen und den Betrieb der ganzen Plattform sowohl in der Cloud als auch On-Premises zu vereinfachen. Mit einem Proof of Concept (PoC) sollen Teile der Architektur mittels technischen Durchstichen validiert werden.

Ergebnis

Um eine möglichst generische Architektur entwickeln zu können, wurden in einem ersten Schritt die nicht funktionalen Anforderungen aus dem Studium einer bestehenden IoT Plattform abgeleitet. Die dort gewonnenen Erkenntnisse und die projektspezifischen Anforderungen führten zum Schluss, dass eine Applikationsplattform mindestens die Bausteine Ressourcenmanagement, Orchestrierung, Microservices, Fast Data und Big Data enthalten muss, um allen Bedürfnissen gerecht zu werden. Mittels Evaluation und Prototyping wurde anschliessend ein Technologie-Stack zusammengestellt:

- **Ressourcenmanagement:** Abstrahiert Ressourcen wie CPU, RAM oder Festplattenspeicher so, dass der Betrieb auf echter Hardware, virtualisiert oder in der Cloud mittels Infrastructure as a Service (IaaS) ermöglicht werden kann. Diese Aufgaben übernehmen die Komponenten DC/OS zusammen mit Apache Mesos.
- **Orchestrierung:** Hat die Aufgabe, Applikationen in Containern, wie zum Beispiel Docker, automatisiert zu skalieren und deren Verfügbarkeit zu gewährleisten und bietet Schnittstellen um schnelle Release-Zyklen mittels Continuous Delivery zu ermöglichen. Dies wird mit der Komponente Marathon von Mesosphere gewährleistet.
- **Microservices:** Ermöglichen durch den modularen Aufbau, polyglott implementierte und entkoppelte Systeme zu bauen. Durch die Verwendung von Docker können diese einfach skaliert werden.
- **Fast Data:** Systeme müssen in der Lage sein, möglichst zeitnah auf Ereignisse reagieren zu können, dabei kommen Komponenten wie Spark Streaming und Akka zum Einsatz.
- **Big Data:** Komponenten verarbeiten sehr grosse Datenmengen mehrheitlich als Batch, dies wird mit Spark und HDFS als verteiltes Dateisystem ermöglicht.

Zusätzlich zu den oben erwähnten Technologien kommen NoSQL Datenbanken wie Cassandra zum Einsatz, Kafka als Messaging-System und NiFi für das Datenfluss-Management. Für den Zugriff auf die Daten werden REST sowie WebSocket APIs bereitgestellt, welche mit Single-Page-Applikationen verbunden werden können.

Mit dem Proof of Concept wurde an einem Beispiel aus dem eHealth Umfeld bewiesen, dass die verschiedenen Technologien zusammen harmonisieren und für zukünftige Projekte eingesetzt werden können.

Ausblick

Ein erster Cluster bestehend aus DC/OS und Apache Mesos wurde bereits intern installiert und wird im nächsten Schritt für die Produktentwicklung eingesetzt.



Simon Weber