

Anwendungssicherheit stellt immer ein umfangreiches und vielschichtiges Thema dar, kommt nun die Komplexität einer Cloud-Infrastruktur hinzu, so wird es noch komplizierter. Im Rahmen dieses Beitrags werde ich Sicherheit im Hinblick auf die Integrität der Daten behandeln, und zwar nicht im Zusammenhang mit der traditionellen Speicherung in einer relationalen Datenbank (RDB). Mir geht es hier um eine neue Art der strukturellen Daten-Integrität, die sich aus dem Wachstum der Cloud heraus entwickelt hat: noSQL.



Was ist noSQL?

Um noSQL hat sich in letzter Zeit ein richtiger Hype aufgebaut, so haben Sie vielleicht schon den Witz gehört. Nach seinem Abschluss sitzt ein IT-Student bei einem erfahrenen Berater, um mit dessen Hilfe sein Resumee für die Bewerbung zu verfassen. Der Berater fragt den Bewerber: „Haben Sie Erfahrung mit relationalen Datenbanken und mit SQL?“. Dieser antwortet, „eigentlich nicht“. „Also“, meint daraufhin der Experte, „dann schreiben Sie noSQL in Ihre Bewerbung.“ In der Praxis bedeutet noSQL keineswegs „kein SQL“, sondern „not only SQL“. Relationale Datenbanken sind keineswegs auf dem Rückzug – sie ermöglichen Funktionen, die nicht zum noSQL-Umfang gehören. Im Zuge des raschen Anwachsens der Cloud- Architekturen in den Unternehmen hat sich jedoch herausgestellt, dass RDBs gewisse Anforderungen einfach nicht erfüllen können. Heute existieren Tausende erfolgreicher WebSites, die noSQL einsetzen, um eine Vielzahl von Anforderungen an die Daten-Integrität zu erfüllen. Als Erste haben Google, Amazon und Facebook noSQL-Datenbanken entwickelt, um Ihren Cloud- Anforderungen zu genügen. Google setzt sie ein, um alle Informationen, die sich im Web befinden, zu indizieren. Amazon verwaltet damit das ganze Volumen an Produkten und Warenkörben und Facebook hat das Social Graph damit digitalisiert.

Das waren natürlich nicht die einzigen Unternehmen, die nicht-relationale Lösungen zur Wahrung der Datenintegrität entwickelt haben und viele dieser Unternehmen haben diese mit Hilfe von Open Source Produkten implementiert. Diese Unternehmen haben begriffen, dass Daten das Öl der heutigen Zeit darstellen. Im kommenden Jahr werden mehr Daten generiert werden, als je zuvor. Diejenigen, die diese Datenflut sammeln und auswerten, können sehr großen Nutzen daraus ziehen.

Also, was GENAU ist denn noSQL?

Manche verstehen darunter eine beliebige Datenintegritäts-Engine, die keine RDB-Architektur einsetzt, aber unter diese Definition würden auch Produkte wie ISAM und Lotus Notes fallen. Meine eigene Definition ist, dass noSQL eine Methode zur integren Speicherung von Daten bezeichnet, die

keine relationale Algebra, wie Select, Join und Vorhersagen einsetzt und für den Einsatz in einer Cluster-Umgebung optimiert wurde, sowohl über ein dynamisches als auch über eine Schema-freie Datenspeicherungsstruktur verfügt und im Hinblick auf die Anforderungen von Web-Anwendungen entwickelt wurde.

Diese Beschreibung trifft auf Dutzende von Produkten zu, die zumeist aus dem Open Source Bereich stammen. Jedes Produkt hat seine eigenen Stärken – eine grobe Unterteilung in vier Kategorien ist jedoch möglich: Key/Value, Dokumentenspeicher, Spalten- orientiert und Graph.

NoSQL: Wann und wofür?

Wie bei jedem neuen Tool, das man noch nie benutzt hat, weiß man zunächst nicht so recht, wofür man es einsetzen sollte. Zunächst einmal möchte ich betonen, dass Sie weiterhin RDB verwenden sollten, wenn Ihre Anwendung ein hohes Maß an Transaktionsintegrität erfordert, Funktionen aus dem Bereich relationale Algebra benötigt oder wenn Sie anspruchsvolle Auswertungsfunktionen einsetzen wollen. NoSQL dagegen sollten Sie in Betracht ziehen, wenn Ihre Anwendung eine der folgenden Anforderungen erfüllen soll:



- Ihre Anwendung muss horizontal skalierbar sein.
- Sie managen komplexe Graph-ähnliche Bezüge (wie soziale Daten).
- Ihre Anwendung darf keinen rigiden Schemata unterworfen sein, da die Anforderungen an die Daten zu dynamisch sind.
- Zu Ihren Daten gehören komplexe Dokumente, Bilder und sonstige Content Management Systeme.
- Sie haben komplexe Webanwendungsdaten, die nur vorübergehend für eine Session benötigt werden, wie zum Beispiel Einkaufswagen.
- Sie verwenden einen Agile Entwicklungsprozess, der durch ein Schema-freies Design an Geschwindigkeit gewinnt.

Zusammenfassend kann man sagen:

Ziehen Sie noSQL in Betracht, wenn Sie ein System entwickeln, dessen Funktionen konventionelle relationale Datenbanken nicht bedienen kann.

CAP-Theorem

Wie erwähnt, ist noSQL nicht ganz sicher. Doch muss man andererseits fragen, ob Ihre Anwendungen, die relationale Datenbanken einsetzen, ganz sicher sind. Nein. Es gibt immer überall Schwachpunkte, doch sind diese bei relationalen Datenbanken überall bekannt. NoSQL ist dagegen

eine neue Technologie, sodass sich viele Provider auf Skalierbarkeit und Performance konzentrieren, statt auf Sicherheit.

This content is available for purchase. Please select from available options.

- [7 Euro/Monat NEWSabo digital - sofort zugreifen.](#)
- [13,5 Euro/Monat NEWSabo plus inklusive 5x Login & Print-Ausgabe - sofort zugreifen.](#)

[Login & Purchase](#)