

LU10a - Grundlagen der Datenhaltung



Die Datenhaltung in verteilten Systemen umfasst verschiedene Konzepte und Technologien um Daten effizient über mehrere Rechner zu speichern und zu verwalten.

In verteilten Systemen werden Daten über mehrere Knoten oder Rechner verteilt gespeichert, um die Last zu verteilen und die Skalierbarkeit zu verbessern. Verteilte Datenbanken ermöglichen es, Daten über verschiedene Standorte hinweg zu speichern, was Flexibilität und Verfügbarkeit verbessert.

Siehe auch [dev-insider - Was ist eine verteilte Datenbank](#)

Replikation und Partitionierung

Eine häufige Technik in verteilten Systemen ist die Datenreplikation, bei der Kopien der Daten auf mehreren Knoten gespeichert werden. Dies verbessert die Ausfallsicherheit und die Verfügbarkeit, da Daten von anderen Knoten abgerufen werden können, wenn ein Knoten ausfällt. Es gibt verschiedene Replikationsstrategien wie Master-Slave-Replikation, Master-Master-Replikation usw.

Datenpartitionierung ist die Aufteilung großer Datenmengen in kleinere Teile, die auf verschiedenen Knoten gespeichert werden. Dies ermöglicht eine bessere Skalierbarkeit, da Anfragen auf mehrere Knoten verteilt werden können, was die Last gleichmässig verteilt und die Antwortzeiten verbessert.

Konsistenz und Integrität

Die Konsistenz der Daten stellt in verteilten Systemen eine grosse Herausforderung dar. Besonders bei der Replikation der Daten müssen Änderungen an einem Knoten mit Sicherheit in allen anderen Knoten nachvollzogen werden. Dabei müssen zeitliche Verzögerungen, widersprüchliche Änderungen (zwei Benutzer ändern den gleichen Datensatz fast zeitgleich auf verschiedenen Knoten) und Ausfälle einzelner Knoten berücksichtigt werden.

Transaktionen spielen eine wichtige Rolle in der Datenhaltung in verteilten Systemen. Mechanismen zum Transaktionsmanagement wie ACID (Atomicity, Consistency, Isolation, Durability) werden verwendet, um die Integrität der Daten sicherzustellen, auch wenn Transaktionen über mehrere Knoten verteilt sind.

Fehlererkennung und -behebung

In verteilten Systemen können Knoten ausfallen oder das Netzwerk kann unzuverlässig sein. Daher werden Mechanismen zur Fehlererkennung und -behebung wie Replikation, Fehlererkennungsprotokolle und automatische Wiederherstellung implementiert, um die Robustheit

des Systems sicherzustellen.

Verteilte Dateisysteme

Neben verteilten Datenbanken werden auch verteilte Dateisysteme verwendet, um grosse Dateien über mehrere Knoten zu speichern. Beispiele hierfür sind [Hadoop Distributed File System \(HDFS\)](#) und [Google File System \(GFS\)](#).

M321-LU10



Marcel Suter

From:

<https://wiki.bzz.ch/> - **BZZ - Modulwiki**

Permanent link:

<https://wiki.bzz.ch/modul/m321/learningunits/lu10/grundlagen>

Last update: **2024/03/28 14:07**

