



MapReduce-System

NVS Projekt 2

Alexander Grill **5CHIEF**

28. Februar 2021

Informatik
HTBLUvA Wr.Neustadt
Österreich

Inhaltsverzeichnis

1	Einführung	2
1.1	Vorwort	2
1.2	Motivation	3
2	Aufgabenstellung	3
2.1	Erläuterung der Aufgabenstellung	3
2.2	Idee	3
2.3	Themenbereiche	3
3	Grundlagen	4
3.1	Was ist ein MapReduce System?	4
3.2	Map und Reduce	4
3.3	Ablauf	5
4	Umsetzung	6
4.1	Aufbau	6
4.2	Klassendesign	6
4.3	Source Code Dokumentation	6
4.4	Verwendete Bibliotheken	6
5	Anwendungsfälle	6
6	Schlusswort	6

1 Einführung

In diesem Kapitel werden die Gründe der Umsetzung und das Thema, worum es in dieser Arbeit geht, genau erläutert. Es wird auch darauf eingegangen, welche Thematiken das Projekt umfassen soll und wie sich die Benotung auseinander setzt.

1.1 Vorwort

Der Virus "COVID-19" war im Jahr 2020 für die gesamte Bevölkerung auf der Erde eine riesengroße Herausforderung. Die Situation änderte sich am Beginn im darauffolgendem Jahr 2021 nicht, deshalb beschloss die Bundesregierung weitere Maßnahmen, Ausgangsbeschränkungen, Grenzkontrollen, FFP2-Maskenpflicht und weitere Regeln die, die Bevölkerung einzuhalten hat. Alle Schüler in Österreich müssen die Schule blockweise besuchen. Nur mit einem davor verpflichtenden Schnelltest und FFP2-Maske dürfen sie die Schule betreten. Da die Projektarbeiten im ersten Semester dementsprechend gut ausgefallen sind, beschloss Herr Professor Kolousek, dass Schüler in den fünften Klassen im Fach NVS statt der Praktischen Arbeit und dem Theorie Test eine Projektarbeit über den Semesterstoff machen müssen. Folgendessen muss die Dokumentation, über das gewählte Thema, dementsprechend einen weiteren Umfang umfassen als beim ersten Projekt im ersten Semester. Im praktischen Teil geht es in diesem Projekt, darum ein Map-Reduce System in Kombination mit Server-Client Kommunikation zu implementieren. Im theoretischen Teil, werden die Grundlagen die für die Umsetzung relevant sind erklärt. Zusätzlich beinhaltet dies auch sämtliche Abschnitte wie, die Source Code Dokumentation, Abläufe, Erklärung bezgl. MapReduce-System, Aufbau und Anwendungsfälle.

Die zu erreichende Note hängt prinzipiell von der

- Beispielkategorie
- Art der Kommunikation
- Funktion, Umfang und Tiefe der Implementierung
- Fehlerbehandlung
- Ausgaben, Einhaltung der Coding Conventions, Kommentare
- Repository: Commits, Issues
- Ausarbeitung
- Einhaltung der Richtlinien

1.2 Motivation

In diesem NVS Projekt geht es darum, ein MapReduce-System mit der Programmiersprache C++ unter Linux mittels g++ Compiler umzusetzen. Im kurzem zusammengefasst soll eine große Menge an komplexen, unstrukturierten und eine Art von aufwendigen Daten verarbeitet werden. Das heißt es sollen Daten, die planlos abgespeichert sind, zusammengefasst werden, sodass diese wieder ihren Nutzen oder Sinn erbringen. Diese können dann für weitere Verarbeitungsschritte oder Datenanalysen verwendet werden. Die Daten werden am Beginn in kleiner Pakete aufgeteilt und diese werden identifiziert mit einem eindeutigen Schlüssel. In der nächste Phase werden die einzelnen Pakete parallel von unterschiedlichen, getrennten und unabhängigen Prozesse zusammengefasst. Danach werden die gruppierten Daten wieder einen Schlüssel zugeordnet, sodass diese wieder zusammengefasst und minimiert werden. Die Kommunikation zwischen den einzelnen Knoten, soll in dieser Arbeit mittel Server-Client Kommunikation passiern. Der Server hört auch einem Port ab, ob sich ein Client damit verbinen möchte, wenn eine Verbindung aufgebaut werden kann, soll die Splittung der Daten passier, daraufhin soll das Resultat weiter an dem Server gegeben werden, bis alle Daten vereint auf dem Master Server liegen. Die bearbeitetn Daten werden schlussendlich in einem JSON-File abgespeichert.

2 Aufgabenstellung

Diese Kapitel umfasst die genaue Erläuterung der Aufgabenstellung, als auch die Themenbereiche die dieser Projektarbeit. Darüber hinaus wird auch über die Idee der Umsetzung geschrieben.

2.1 Erläuterung der Aufgabenstellung

2.2 Idee

2.3 Themenbereiche

3 Grundlagen

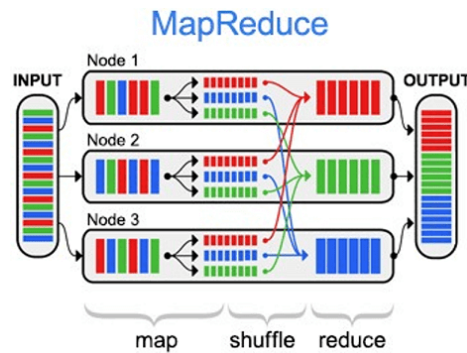
3.1 Was ist ein MapReduce System?

Das Verfahren wurde 2004 von Google entwickelt für die Indexierung von Webseiten. Das Framework wird bei Datenbanken eingesetzt und dient zur Verarbeitung von großen, komplexen, unstrukturierte Datenmengen. Dieses Verfahren findet Anwendung für BigData und Datawarehouse, weil in solchen Fällen große Datenmengen in kürzester Zeit mittels Software verarbeitet, analysiert, aggregiert als auch komprimiert werden. Map Reduce parallelisiert die Bearbeitung, durch die Verteilung auf mehrere gleichzeitig auszuführende Tasks. Der Grund, warum dieses Framework solche Datenmengen verarbeiten kann ist, weil die Aufgaben auf mehreren Rechnern aufgeteilt werden. Jeder einzelne Rechner startet Prozesse, die Parallel die Daten verarbeitet und auswertet. Ein einzelner Rechner stößt schnell an seine Grenzen, deshalb ist die Verarbeitung von Daten, mittels mehreren Knoten sehr effizient und bietet eine bessere Performance. Das Verfahren wurde in vielen verschiedenen Verfahren eingesetzt wie zum Beispiel für die Indizierung von Webseiten, nach einer Suchanfrage mit beliebigen Zeichenketten, ebenso im Umfeld von Google News wird MapReduce verwendet. Andere große Internetfirmen wie Yahoo, die ebenfalls das Verfahren für die Indexierung von Webseiten verwenden, als auch Facebook verwendet das System, um Spam Messages zu minimierung und die Ads zu optimieren. Wohingegen Amazon das Verfahren für das Clustering der Produkete verwendet

3.2 Map und Reduce

Die beiden Grundfunktionen des Verfahrens sind Map und Reduce. Sie sorgen für die Aufteilung der Aufgaben in kleinere parallelisierten Arbeitspakete und führen am Ende die Ergebnisse zusammen. Bei großen relationalen Datenbanken und komplexen Queries lassen sich typische Problem, bezüglich Verarbeitung von großen Datenmengen beseitigen. Die Map Funktion, verteilt die Aufgaben an unterschiedlichen Knoten eines Clusters. Die Reduce Funktion sortiert die verfassten Ergebnisse und fügt sie am Ende wieder zusammen. Die Funktionen Map und Reduce werden vom User bereitgestellt, weil diese schließlich zu den bereitgestellten Daten passen müssen.

3.3 Ablauf



Das Verfahren verläuft durch folgende Schritte:

- Split
 - Die bereitgestellten Daten werden aufgeteilt. Jeder Datensatz darin ist identifiziert durch einen Schlüssel-Wert. Diese Datenmenge wird nun in kleinere Datenmengen aufgeteilt und vom Master an die verfügbaren Knoten verteilt.
- Map
 - Nun wendet jeder Knoten auf die Daten die Map-Funktion an, die schließlich Key/Value Paare zurückgibt. Diese Ergebnisse werden zwischengespeichert.
- Shuffle
 - Bei diesem Schritt geht es darum den reduce-Knoten, die entsprechenden Daten zuzuteilen. Diese Zuteilung entspricht einer Fragmentierung. Hierbei wird den Knoten, die reduce ausführen, ein Key zugeteilt. Diese Knoten holen sich dann die bereits durch Map entstandenen Datensätze mit diesem Key und wenden reduce an.
- Reduce
 - Grundsätzlich ist die Aufgabe dieser Funktion, die Key/Value Paare anhand des Schlüssels zusammenzufassen und dabei die Summe der einzelnen Values zu bilden. Demnach ist die Ausgabe der Reduce-Funktion wieder ein Key/Value Paar mit dem gleichen Aufbau wie vor der Verarbeitung. Dies ermöglicht es, dass reduce mehrere Male angewendet werden kann, bis schließlich alle Daten gesammelt wurden.

4 Umsetzung

4.1 Aufbau

4.2 Klassendesign

4.3 Source Code Dokumentation

4.4 Verwendete Bibliotheken

5 Anwendungsfälle

6 Schlusswort