



# Methoden zur Beschreibung und Einordnung heterogener Hardware Infrastrukturen für Scientific Workflows

Robert Balink

robert.balink@campus.tu-berlin.de

16.Dezember 2021

Bachelorarbeit

Distributed and Operating Systems

Technische Universität Berlin

Erstprüfer: Prof. Dr. habil. Odej Kao

Betreuer: Jonathan Bader

Zweitprüfer: Prof. Dr. Dr. h.c. Sahin Albayrak



## Eidestattliche Erklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und eigenhändig sowie ohne unerlaubte fremde Hilfe und ausschließlich unter Verwendung der aufgeführten Quellen und Hilfsmittel angefertigt habe.

Die selbständige und eigenständige Anfertigung versichert an Eides statt:

A handwritten signature in black ink, appearing to read "R. Bahk". The signature is written in a cursive, slightly slanted style.

Berlin, den 19.10.2021



# Anerkennung



# **Zusammenfassung**

Die Abschlussarbeit behandelt

# Inhaltsverzeichnis

Eidestattliche Erklärung . . . . .	I
Anerkennung . . . . .	III
Zusammenfassung . . . . .	V
Abbildungsverzeichnis . . . . .	IX
Tabellenverzeichnis . . . . .	X
Listenverzeichnis . . . . .	XI
<b>1 Einführung</b>	<b>1</b>
1.1 Problematik . . . . .	1
1.2 Ziel . . . . .	2
1.3 Beiträge . . . . .	2
1.4 Struktur . . . . .	2
<b>2 Hintergrund</b>	<b>5</b>
2.1 Scientific Workflows . . . . .	5



2.2	Computer Architektur . . . . .	6
2.2.1	Aufbau . . . . .	6
2.2.2	Prozessor . . . . .	7
2.2.3	Arbeitsspeicher . . . . .	9
2.2.4	Sekundärspeicher . . . . .	10
2.2.5	Weitere Ein- und Ausgabegeräte . . . . .	11
2.2.6	Zusammenspiel der Komponenten für die Performance von Scientific Workflows . . . . .	12
2.3	Metriken . . . . .	12
2.3.1	Anforderungen an eine Metrik . . . . .	13
2.3.2	Historische Metriken . . . . .	15
2.3.3	Industrie Benchmarks . . . . .	16
2.3.4	open-source Benchmarks . . . . .	17
<b>3</b>	<b>Methodik</b>	<b>19</b>
3.1	Anforderungen an das Programm . . . . .	19
3.2	Online-Datenbanken . . . . .	20
3.2.1	cpu-world . . . . .	21
3.2.2	userBenchmark . . . . .	21
3.2.3	openBenchmarking . . . . .	21

3.3 Externe Datenbank . . . . .	21
3.4 Berechnung des Scores . . . . .	21
<b>4 Auswertung</b>	<b>23</b>
4.1 Validierung der Ergebnisse . . . . .	23
4.2 Abweichungen . . . . .	23
4.3 Klassifizierung der Ergebnisse . . . . .	23
<b>5 Stand der Forschung</b>	<b>25</b>
<b>6 Diskussion</b>	<b>27</b>
<b>7 Fazit</b>	<b>29</b>
<b>A Abkürzungsverzeichnis</b>	<b>31</b>
<b>B Lexikon</b>	<b>33</b>
<b>C Listings/Anhang</b>	<b>35</b>
C.1 Configuration for Node A . . . . .	35
<b>Literaturverzeichnis</b>	<b>37</b>

# Abbildungsverzeichnis

# Tabellenverzeichnis

2.1	Flynn'sche Klassifikation in Anlehnung [29] . . . . .	8
2.2	Anforderungen an eine Metrik nach [17],[12],[16] und [14] . . . . .	14

# Listenverzeichnis

C.1 Configuration for Node A . . . . .	35
--	----



# Kapitel 1

## Einführung

Die Abschlussarbeit behandelt die unterschiedliche Methoden zur Beschreibung und Einordnung von heterogenen Hardware Infrastrukturen für Scientific Workflows.

### 1.1 Problematik

Für das Ausführen wissenschaftlicher Workflow Management Systeme, wie beispielsweise nextflow.io laufen verschiedene Arbeitspakete auf unterschiedlichen Computern. Die Arbeitspakete können allerdings zeitintensiv sein, da sie viele Ressourcen benötigen. Ohne das Wissen, welche Computereinheit leistungstärker ist und diese dementsprechend nicht priorisieren zu können, gehen viele Ressourcen verloren. Für den Anwender ist es daher von Vorteil zu wissen, welcher Computer von Anfang an leistungsfähiger ist. Durch dieses Wissen spart der Anwender viel Zeit und kann effektiver die Arbeitspakete aufteilen.

## 1.2 Ziel

Das Ziel dieser Abschlussarbeit ist es, die Performance-relevanten und kritischen Bereiche des Computers ausfindig zu machen, die gängigen wissenschaftlichen Performance Metriken zu ermitteln und daraufhin ein eigenes Kommandozeilen Tool zu schreiben, welches die Systemdaten aus dem Computer ausliest, analysiert und die Performance des Computers für Scientific Workflows einschätzt in Relation zu einem Referenz Computer. Am Ende sollen die berechneten Werte validiert und die Computersysteme klassifiziert werden.

## 1.3 Beiträge

Für die Problematik wurde ein Kommandozeilen Tool in Java entwickelt, welches auf Linux Computern ausführbar ist und ein Performance Ergebnis liefert. Das Tool nimmt dabei die Systemdaten aus dem Computer, speichert diese in einer externen Datenbank und nimmt aus anderen Benchmark-Datenbanken die passenden Werte und stellt diese gegenüber, sodass ein Performance Wert errechnet werden kann. Außerdem werden die heterogenen Hardware Infrastrukturen welche in der externen Datenbank gespeichert sind nach unterschiedlichen Kriterien eingeordnet.

## 1.4 Struktur

Im folgenden wird in Kapitel 2 die inhaltliche Aufarbeitung stattfinden. Dazu gehört die Einführung in das Thema Scientific Workflows 2.1, sowie die inhaltliche Auseinandersetzung der Computer Architektur 2.2 und ihren einzelnen Komponenten sowie dem Zusammenspiel dieser mit der Performance für Scientific Workflows 2.2.6. Daraufhin folgt die Betrachtung der Metriken 2.3 und wie diese im historisch-wissenschaftlichen Kontext genutzt und bewertet werden. In Kapitel 3 wird die Entwicklung des Kommandozeilen Tools dargestellt.



In 3.4 wird erläutert, wie sich die finale Bewertung zusammensetzt und welche Gewichtung dabei stattfindet. In dem Kapitel Auswertung 4 werden die berechneten Ergebnisse aus dem Kommandozeilen Tool gegenüber gestellt mit den tatsächlichen Ergebnissen durch eine direkte Gegenüberstellung der Laufzeit. Es erfolgt dann eine Klassifizierung 4.3 der Leistungsfähigkeit der unterschiedlichen Hardware Infrastrukturen für Scientific Workflows. In Kapitel 5 wird der Stand der Forschung dargestellt und es folgt in Kapitel 6 eine Diskussion über die Ergebnisse des Tools. Die Abschlussarbeit wird abgeschlossen mit dem Fazit 7, welches die Ergebnisse kurz zusammenfasst und einen Ausblick gibt über weitere potenzielle Themenfelder in dem Gebiet.



# Kapitel 2

## Hintergrund

### 2.1 Scientific Workflows

Scientific Workflows werden genutzt um rechenintensive Simulationen, Datenanalyse oder auch Visualisierungen im wissenschaftlichen Kontext zu durchzuführen [6] [19]. Ein Workflow ist ein automatisierter Prozesse [19]. Die einzelnen Arbeitsschritte innerhalb eines Workflows werden in sogenannten Tasks organisiert und orchestriert [20]. Die Verarbeitung der Daten unterscheidet sich hierbei in den einzelnen Schritten. Typische Operationen sind beispielsweise das Einlesen von Inputdaten, Datenverarbeitung, Datenanalyse, aggregierte Outputdaten [19]. Scientific Workflow Management Systeme (SWMS) sind Software Programme in welchen sich wiederum Scientific Workflows erstellen, bearbeiten und ausführen lassen. Das Kepler-Projekt, welches 2002 ins Leben gerufen wurde, gilt als eines der ersten SWMS. Es ist außerdem eines der am meist zitiertesten Scientific Workflows im wissenschaftlichen Kontext und spielt dementsprechend auch von besonderer Bedeutung für die Entwicklung der Scientific Workflow Community [1]. Durch das wachsende Interesse und der steigenden Popularität der Scientific Workflows in wissenschaftlichen Kreisen sind in den letzten Jahren auch mehr SWMS entstanden. Prominente Tools sind hierbei unter anderem Snakemake,

Pegasus, Galaxy und Nextflow. Dadurch, dass der Zugriff zu mehr Daten im Allgemeinen wächst, steigt die Daten-Intensität von einzelnen Workflows. Populäre Anwendungsgebiete für Scientific Workflows sind beispielsweise die Bioinformatik [1].

## 2.2 Computer Architektur

In den kommenden Unterkapiteln erfolgt eine Einführung in die relevanten Komponenten eines Computers. Hierbei wird jeweils auf die Funktionalität und der dazugehörigen Leistungsfähigkeit für den Computer eingegangen. In Kapitel 2.2.6 wird zusammengefasst, inwiefern die Performance der Komponenten relevant sind für Scientific Workflows.

### 2.2.1 Aufbau

Ein moderner Personal Computer (PC) besteht aus dem Zusammenspiel von Prozessor, Arbeitsspeicher, sowie einiger Ein- und Ausgabegeräte. Die Komponenten sind untereinander mit einem sogenannten Bus-System verbunden. Dieser Aufbau entspricht der Von-Neumann-Architektur, welche aus den 1940er Jahren stammt und vom ungarisch-amerikanischen Mathematiker John von Neumann et al. entwickelt wurde. Die Von-Neumann Architektur ist bis heute in den meisten Computersystemen vorhanden [9] [7]. Das Besondere an der Von-Neumann Architektur ist, dass sowohl Daten als auch Programme in dem Speicherwerk, heutzutage Arbeitsspeicher genannt zusammengelegt werden. Das Konkurrenzkonzept der Harvard-Architektur besitzt eine physikalische Trennung zwischen Programm Instruktionen und Daten. Dies hat einen Kostenaufwand an Hardware zur Folge. Trotz positiver Effekte, wie beispielsweise der höheren Geschwindigkeit durch parallele Verarbeitung konnte sich die Harvard-Architektur nicht durchsetzen [7]. Üblicherweise sind die einzelnen Komponenten eines PCs auf der Hauptplatine befestigt.

In den folgenden Unterkapiteln wird näher auf die einzelnen Bestandteile des Computers

eingegangen. Allen voran dem Prozessor 2.2.2 und dem Arbeitsspeicher 2.2.3. Unter dem Sammelbegriff Ein- und Ausgabegeräte 2.2.5 versteht man hierbei alle Komponenten, die nicht unmittelbar zum Kern der Von-Neumann-Architektur gehören, dazu gehört auch der Sekundärspeicher 2.2.4.

## 2.2.2 Prozessor

Das Herzstück des Computers bildet der Prozessor, auch CPU (Central processing unit) genannt. Der Prozessor kümmert sich um die Ausführung von Programmen, die er aus dem Arbeitsspeicher erhält. Die CPU besteht aus vier Funktionseinheiten. Dem Steuerwerk, dem Register, dem Rechenwerk und den Bussen. Das Steuerwerk ist für die Koordination des gesamten Prozessors verantwortlich. Er kontrolliert hierbei die Befehle und Operationen, die das Rechenwerk als nächstes ausführen soll. Bei dem Register handelt es sich um Datenregister, welches Adressen aber auch Daten speichern kann. Sie zeichnen sich vor allem damit aus, dass auf diese Speicherbereiche schnell zugegriffen werden kann [29]. Das Rechenwerk, (Arithmetic Logic Unit, ALU) nutzt die vom Steuerwerk geprüften Operationen und Befehle, um diese dann auf den Daten des Registers auszuführen. Bei den Operationen und Befehlen die verarbeitet werden handelt es sich dabei um arithmetische und logische Funktionen [29]. Alle Komponenten sind untereinander über ein Leitungssystem, auch Bus genannt verbunden. Die Arbeitsweise des Prozessors geht dem Abrufen-Dekodieren-Ausführen-Zyklus nach [29].

Durch die stetige Entwicklung von Computerchips und der damit verbundenen Erhöhung von Transistoren auf diesen steigt die Leistungsfähigkeit von Prozessoren weiter an [29]. Fast 5 Jahrzehnte galt der Grundsatz von Gordon Moore, dass sich ungefähr alle 18 Monate die Anzahl der Transistoren pro Flächeneinheit (Integrationsdichte genannt) in einem Computerchip verdoppeln würden. Dieser Grundsatz ist als das Mooresches Gesetz bekannt. Doch auch dieses soll demnächst an seine physikalischen Grenzen kommen, da man sich im Bereich

nm befindet [\[31\]](#).

Heutzutage bestehen übliche PCs aus mehr als nur einem einzelnen Prozessor. Dabei basieren die sogenannten Mehrprozessorsysteme weiterhin auf der Von-Neumann-Architektur. In der Klassifikation von Michael J. Flynn [2.1](#) aus dem Jahr 1966 wird der klassische Von-Neumann-Rechner als SISD (Single Instruction, Single Data) bezeichnet, da dieser nur einen Kern besitzt [\[29\]](#). Die meisten PCs besitzen zwischen 2 und 8 Kerne und sind somit Mehrprozessorsysteme.

Befehlsströme	Datenströme	Name	Beispiel
1	1	SISD	Von-Neumann-Rechner, Einkernprozessor
1	Mehrere	SIMD	Vektor-Supercomputer, Array-Prozessor
Mehrere	1	MISD	nicht bekannt
Mehrere	Mehrere	MIMD	Multiprozessor, Multicomputer

Table 2.1: Flynn'sche Klassifikation in Anlehnung [\[29\]](#)

Während man in einer Computereinheit einen Multiprozessor haben kann, so geht auch die horizontale Skalierung zu sogenannten Multicomputersystemen [\[29\]](#).

Von CPU-Herstellern wird oft die hohe Taktfrequenz beworben. Der numerische Wert lässt Endkunden suggerieren, dass es sich um eine leistungsstarke CPU handelt. Dies ist allerdings ein Trugschluss. Eine 350 Megahertz (MHz) CPU ist nicht zwangsläufig leistungstärker als eine CPU mit 300 MHz. Die Zahl ignoriert nämlich die tatsächliche Berechnung die bei jedem Zyklus durchgeht [\[17\]](#) Bei alltäglicher Nutzung eines Computers beträgt die Auslastung in den wenigsten Fällen 100 Prozent, vor allem beim Hochfahren des Computers wird die CPU unter Stress gesetzt [\[27\]](#).

### 2.2.3 Arbeitsspeicher

Während sich der Prozessor um die Bearbeitung des Programms kümmert, befindet sich das geöffnete Programm auf dem Speicher des Arbeitsspeichers. Es bietet die Fläche für das zu bearbeitende Programm. Der Arbeitsspeicher fungiert als eine Art Zwischenspeicher, der für die Ausführung vom Programm benötigt wird, damit diese ausgeführt werden kann. Im Gegensatz zu einer Festplatte existieren die Daten nur zur Laufzeit und würden, wenn diese nicht im Hintergrund auf einer Festplatte hinterlegt wären alle wieder gelöscht, sobald der Computer ausgeschaltet wird. Im allgemeinen Wortgebrauch wird der Arbeitsspeicher auch als Hauptspeicher oder Memory bezeichnet. Der englische Begriff !Random Access Memory! (kurz RAM) beschreibt den Arbeitsspeicher ziemlich treffend als !Direktzugriffsspeicher!, da dieser nämlich Zugriff auf jeden möglichen Speicherplatz des Computers hat, diesen direkt adressieren kann und im Austausch mit dem Prozessor steht [11]. Der Arbeitsspeicher ist aufgebaut als eine Anordnung von vielen Zellen welche Bits in sich halten, (0 oder 1) [11].

Im Gegensatz zu Prozessoren, welche sich in der Leistung und damit der Geschwindigkeit steigern, bekommen Arbeitsspeicher immer mehr Kapazität [29]. Technisch wäre es zwar möglich den Arbeitsspeicher auch deutlich schneller zu machen, allerdings wäre das redundant, da die Verbindung über den Bus zwischen dem Arbeitsspeicher und dem Prozessor zu langsam ist um. Für die Optimierung der Zugriffsgeschwindigkeit nutzt man daher den sogenannten Cache (bedeutung), einen Speicher der direkt in oder neben dem Prozessor eingebaut ist [29]. Der Cache ist ein kleiner und schneller Speicher, der die am häufig genutzten (Wörter?) aufbewahrt um eine schnellere Zugriffszeit für die CPU ermöglicht [29]. Während bei dem Prozessor die Geschwindigkeit untereinander verglichen werden kann, ist bei dem Arbeitsspeicher vor allem wichtig, dass die Kapazität der Programmanforderung entspricht. Sollte dies nämlich nicht der Fall sein und das Programm benötigt mehr Speicher als der Arbeitsspeicher vorweisen kann, wird die Festplatte benötigt, was dazu führt, dass der gesamte Prozess schlagartig verlangsamt wird [27] Giakamozis (1999) [11] vergleicht den Arbeitsspeicher mit dem Kurzzeitgedächtnis eines Menschen, während der Sekundarspeicher

das Langzeitgedächtnis repräsentiert. Sollte etwas nicht im Kurzzeitgedächtnis sein, müsste Von daher ist es beim Arbeitsspeicher wichtig, genug Kapazität bereitzustellen. Ansonsten wird der RAM zum Flaschenhals des. Der leistungstärkste Prozessor bringt einem keinen Mehrwert, wenn der Arbeitsspeicher nicht genug Platz anbietet.

DDR Wenn der Arbeitsspeicher voll ausgelastet ist, wird die Festplatte genutzt, was den gesamten Prozess extrem verlangsamt [27]. Je langsamer der Speicher  $\rightarrow$  desto mehr Zyklen muss die CPU warten [29]

## 2.2.4 Sekundärspeicher

Als Sekundärspeicher werden alle weiteren Speichermedien zusammengefasst, die im Gegensatz zum Arbeitsspeicher nicht in unmittelbarer Verbindung zum Prozessor stehen. Der Sekundärspeicher kann im Gegensatz zum RAM die Daten persistent speichern. Der Sekundärspeicher steht in Verbindung mit dem Arbeitsspeicher und kann zum Interagieren mit diesem Daten Lesen und Schreiben. In der von-Neumann-Architektur gehören Sekundärspeicher zu den Ein- und Ausgabegeräten. Ein typisch eingebauter Sekundärspeicher in einem PC ist die Festplatte, auch Hard Drive Disk (kurz HDD) genannt, sowie die Solid State Drive (kurz SSD). Weitere Sekundärspeicher wären beispielsweise CDs, DVDs, Blu-rays, USB-Sticks oder auch Disketten. Im Gegensatz zum Arbeitsspeicher ist die Kapazität der eingebauten Sekundärspeichern enorm, dafür allerdings die Zugriffszeit deutlich langsamer. Hauptaugenmerk liegt hier vor allem bei den Festplatten und SSDs. Eine Festplatte ist ein magnetisches Speichermedium bestehend aus mehreren Runden Aluminiumscheiben mit magnetisierbarer Beschichtung [29]. SSDs sind im Gegensatz zu Festplatten Solid State Drive's sind deshalb auch deutlich schneller als HDDs Zugriffszeiten Daher sind SSDs im Jahr 2021 auch vierfach so teuer wie HDDs für die gleiche Kapazität zugriffszeit vergrößert sich Auch in den nächsten Jahren wird sie die Kapazität von Festplatten und SSDs vergrößern. Wesentliche Erkenntnis für die Sekundärspeicher ist, dass für den gesamten Prozess die Zugriffsgeschwindigkeit



von wichtiger Bedeutung ist, da diese ansonsten zum Flaschenhals werden kann. Neben der Zugriffsgeschwindigkeit muss selbstverständlich auch dafür gesorgt werden, dass auf dem Medium genügend Speicher vorhanden ist, um die Daten persistieren zu können.

## 2.2.5 Weitere Ein- und Ausgabegeräte

Da unter dem Sammelbegriff Ein- und Ausgabegeräte im Grunde alle Geräte inbegriffen sind, die in der von-Neumann-Architektur nicht in unmittelbarer Verbindung zum Prozessor stehen gibt es hier viele Beispiele für Ein und Ausgabegeräte. Beispielhafte Eingabegeräte, die wohl die meisten Computer besitzen sind hierbei: Maus, Tastatur oder auch Mikrofon. Ausgabegeräte wären beispielsweise Monitor, Drucker oder Lautsprecher. Ein weiteres Ausgabegerät, welches sich typischerweise in Computern befindet und für alle Anwender wichtig ist, ist die Grafikkarte. Ohne die Grafikkarte, auch Videokarte oder Graphics processing unit (kurz GPU) genannt, würde man kein Bild auf dem Monitor angezeigt bekommen. Es ist die Schnittstelle über die Menschen mit dem Computer interagieren können. Wie bei einer CPU befindet sich in einer Grafikkarte ein Mikroprozessor. Während GPUs vor über einem Jahrzehnt hauptsächlich nur in der Lage waren 3D-Rendering-Prozesse zu verarbeiten, verfügen Grafikkarten Prozessoren mittlerweile über enorme arithmetische Kapazitäten, die sogar die CPU übertreffen [23] [5]. Sie sind optimiert für das parallele Berechnen von repetitiven arithmetischen Aufgaben, wie das Darstellen von Pixeln in einem Computerspiel. Die Grafikkarte muss nämlich ununterbrochen jede Visualisierung in einem Computerspiel berechnen. In den letzten Jahren kamen vor allem zwei weitere Trends hinzu, für die sich die Funktionsweise der GPU als vorteilhaft erwiesen hat und das, obwohl diese Aufgaben keine grafischen sind. Das ist auf der einen Seite das sogenannte Minen für Blockchains, wie die Kryptowährung Bitcoin und auf der anderen Seite die Berechnung für künstliche neuronaler Netze von Machine Learning Prozessen [26] [30]. Die Kombination aus Parallelisierung und der starken arithmetischen Kapazität des Mikroprozessors der Grafikkarte macht sie daher sehr beliebt für Prozesse die enorm viele Aufgaben berechnen muss. Gewöhnliche Office-

Computer oder Laptops benötigen dahingegen nicht die zusätzliche Grafikleistung und haben daher einen integrierten Grafikprozessor auf der Hauptplatine oder in der CPU direkt integriert. Dieser reicht aus, wenn keine grafisch intensiven Arbeiten verrichtet werden.

### **2.2.6 Zusammenspiel der Komponenten für die Performance von Scientific Workflows**

Bei der Betrachtung der Performance der Komponenten im Hinblick auf Scientific Workflows ist zu beobachten, dass diese abhängig ist von der Art des Scientific Workflows. Ein wesentlicher Faktor ist hierbei die Leistungsfähigkeit des Prozessors. Durch die rasante Entwicklung gibt es auch Tendenzen, dass die Grafikkarte aufgrund ihrer Vorteile wie in Kapitel [2.2.5](#) beschrieben bei Scientific Workflows hinzugezogen wird. Dieser Trend wird in dieser Arbeit allerdings nicht näher betrachtet, da es auf den genutzten Plattformen noch nicht vorhanden und genutzt wird. Sowohl der Arbeitsspeicher als auch die Sekundärspeicher spielen eine wichtige Rolle, da es bei Daten-intensiven Workflows auch um die Zugriffszeit geht. Während sich Prozessoren in ihrer Geschwindigkeit überbieten können, müssen sowohl Arbeitsspeicher als auch Sekundärspeicher genug Kapazitäten bereitstellen. Andernfalls werden sie mögliche Flaschenhälse für den gesamten Prozess.

## **2.3 Metriken**

Metriken werden in der Informatik genutzt, wenn man komplexe Zusammenhänge von Software oder auch Hardware reduziert auf wenige Kennziffern bewerten und interpretieren möchte. Man kann eine Vergleichbarkeit herstellen und mit den gewonnenen Informationen Entscheidungen über die Software oder Hardware treffen. Während eine Metrik eine oder mehrere Kennziffern von einem getesteten Prozess liefert, ist eine Benchmark ein Vergleichsmaßstab, welche die Kennziffer in Kontext setzt mit anderen ausgewerteten Ergeb-

nissen. Ein Benchmark-Tool hingegen ist meist ein Software-Programm, welches die Metrik auf dem Computer ausführt und in dem Ergebnis in Vergleich setzt mit beispielsweise dem Vergleichswert oder dem Durchschnitt. Es gibt für viele Anwendungsfälle, beispielsweise 3D-Visualisierung, Machine Learning oder Videobearbeitungsprogramme auch passende Benchmark-Tools. Oftmals konkurrieren diese untereinander. Im Falle von Scientific Workflows und den dazugehörigen Scientific Workflow Management Systemen fehlt allerdings im wissenschaftlichen Diskurs ein solches Benchmark-Tool, wie auch Deelman et al. anmerken [6]. Daher wird in den folgenden Unterkapiteln darauf eingegangen, nach welchen Kernaspekten eine Metrik entsteht und welche Kennzahlen und Metriken sich als sinnvoll für Scientific Workflows erweisen.

### **2.3.1 Anforderungen an eine Metrik**

Für die Analyse und Bewertung vorhandener Metriken, sowie Erstellung einer eigenen müssen die Bewertungskriterien für einen Hardware Performance Test ermittelt werden. Dadurch, dass in der Vergangenheit Metriken in diesem Forschungsgebiet oftmals missverständlich interpretiert oder auch falsch angewendet worden sind, bietet es sich an, hierbei die Voraussetzungen und Anforderungen für eine aussagekräftige Metrik zu eruieren [17]. Das Ziel sollte sein, mit Hilfe einer Benchmark ein Standardwerkzeug zur Vergleichbarkeit zu schaffen [14]. Schon im Jahr 1988 galt es als kontrovers die Performance eines Computers auf eine einzige Zahl zu reduzieren, da man sich im wissenschaftlichen Diskurs einig war, dass die Performance multidimensional ist und sich besser durch mehrere Zahlen repräsentieren lässt [28]. Eine Benchmark und auch die zugrundeliegende Metrik sollte nach Kounev et al. fair und repräsentativ sein [15]. In den vergangenen 15 Jahren haben mehrere Researcher in dem Themengebiet der Performance Evaluation und Bewertung unterschiedliche Anforderungen definiert. David J. Lilja startete 2005 mit der Erstellung der Anforderungen 2.2. Im Laufe der Zeit wurde die Liste ergänzt von Karl Huppler, William Kramer und Jóakim v. Kistowski et al.

Eigenschaft	Erklärung	Researcher
1. linear	Es muss eine lineare Beziehung zwischen der wahren und der berechneten Performance geben.	Lilja, Kramer
2. zuverlässig	Wenn Ergebnis A besser als Ergebnis B ist, dann muss A in Wirklichkeit auch besser sein als B.	Lilja
3. wiederholbar	Wenn man den Test wiederholt, muss das gleiche Ergebnis kommen.	Lilja, Huppler
4. einfach	Der Test kann von anderen genutzt werden und ist für den Nutzer verständlich.	Lilja, Huppler
5. konsistent	Die Definition der Metrik ist über alle (zu testenden) Systemen und Konfigurationen gleich	Lilja
6. unabhängig	Der Test wird nicht durch äußerliche Faktoren beeinflusst	Lilja, Huppler
7. relevant	Der Test muss relevante Informationen für den Nutzer bieten, ansonsten liefert der Prozess keinen Mehrwert	Huppler
8. verifizierbar	Das Vertrauen, dass das Ergebnis wahr ist. Wird geschaffen über Peer Review im wissenschaftlichen Kontext	Huppler, Kistowski et al.
9. wirtschaftlich	Die Geldgeber/Das Unternehmen kann es sich leisten den Test zu finanzieren.	Huppler

Table 2.2: Anforderungen an eine Metrik nach [17],[12],[16] und [14]

Nach Huppler müssen nicht alle Eigenschaften erfüllt werden, da dies in der Realität auch schwer umsetzbar sei. Ein Kompriss dagegen ist es, wenn stattdessen ein paar Punkte eine höhere Gewichtung haben [12]. Im der weiteren Betrachtung von vorhandenen Metriken und Benchmarks wird der Punkt 9 nicht weiter beachtet, da sich diese Eigenschaft für die Entwicklung einer Metrik bzw. Benchmark innerhalb eines Unternehmens dreht. Kramer hat in seinen Ausführungen ebenfalls betont, dass auch das Gewichten von Kennzahlen in Abhängigkeit vom Einfluss für die Anwendung von Bedeutung ist [16].

## 2.3.2 Historische Metriken

Eine historisch oft genutzte und bis heute immer noch beliebte Maßeinheit für die Berechnung der Computer Performance ist Millionen Instruktionen pro Sekunde (MIPS).

Sowohl MIPS als auch MFLOPS wurden oft in den 1980er Jahren genutzt [15].

Vorteil ist, dass sie einfach und vor allem intuitiv zu verstehen sind. Problematisch ist allerdings, dass es im allgemeinen eine eher schwache Metrik ist, die nicht viel aussagt. [25]

Um den schwächen entgegenzuwirken, ist es sinnvoll MIPS/MFLOPS durch das arithmetische Mittel / harmonische Mittel zu nutzen und diese mit geeigneten Gewichten zu bewerten [13]. Beispiel: [21] Mehr inhalt: [3] und [16]. Was Citron und Mashey sagen gecovered auch von [8]. Gewichtet geometric mean besser als arithmetisch und nur geometrisch [8].

MLFOPS - harmonisches Mittel [13].

SpeedUp - [13] recherche !

MHz Ein schlechter Indikator, eine höhere MHz Zahl, sagt nichts über die Leistung aus. [10]

Wie in dem Kapitel 2.2.2 ausgeführt, bietet sich die Taktfrequenz der CPU als ungeeignetes Kriterium einer Metrik an, da ein höherer Megahertz-Wert keine sinnvolle Aussage trifft über die Performance [17].

- Clockrate der CPU: Problem: Ignoriert die tatsächliche Berechnung bei jedem Zyklus. Hohe Clockrate heißt nicht zwangsläufig schnellere Performance. - MIPS (Millionen Instruktionen pro Sekunde): Problem: Jede Instruktion gilt als eine Einheit –; Prozessoren können einzigen Befehl unterschiedlich durchführen. Sprich: Doppelte MIPS Rate heißt nicht doppelte Performance. - MFLOPS (Millionen Gleitkommaoperationen pro Sekunde): Problem: Programm ohne Gleitkommaoperationen werden als 0 gewertet, auch wenn sinnvolle Operationen erledigt werden. Wie genau zählt man die Gleitkommaoperationen?

MIPS

MFLOPS

Problem: Zahlen geben nur wieder was getan wurde, aber nicht wie viel Fortschritt(Arbeit) geleistet wurde [10]

GigaFlops

CPU Performance Equation

Grafikkarte Wie in Kapitel 2.2.5 erwähnt, spielen Grafikkarten für den Anwendungsfall (dieses Projekt) eine untergeordnete Rolle. Aber durch die Attraktivität von AI Applicationen gibt es auch hier Benchmarks und Metriken die das Zusammenspiel von GPU und CPU priorisieren –; Beispiel [5]

BENCHMARKS FÜR SSD UND RAM UND IO !

### 2.3.3 Industrie Benchmarks

Die beiden Non-Profit-Organisationen Standard Performance Evaluation Corporation (SPEC) und Transaction Processing Performance Council (TPC) erstellen seit Jahrzehnten Benchmarks die in der wirtschaft als Industriestandard gelten. Während sich SPEC vor allem bekannt ist für ihre Benchmark Tests im Bereich Prozessoren, Grafikkarten und High-Performance

Computing, ist die TPC eher spezialisiert auf die Performance von Datenbankmanagementsystemen und Transaktionen [24] [18]. Beide Organisationen nehmen es sich als Ziel unabhängig zu agieren und . Auch im wissenschaftlichen Diskurs werden beide Organisationen, beispielsweise von [14] als fair in ihrer Bewertung betrachtet.

Die TPC-C Benchmark von TPC galt seit ihrer Einführung 1992 als eine der wichtigsten Benchmark Tests [12]. Der Test simuliert Echtzeit-Transaktionsverarbeitungs Prozesse (OLTP) für Datenbankmanagementsysteme. Während die TPC kaum relevante Benchmarks für Scientific Workflow besitzt, veranstaltet diese jährlich eigene Konferenzen in denen aktuelle Trends und Entwicklungen innerhalb der Benchmark Community präsentiert werden. In Kapitel 5 wird näher darauf eingegangen.

Die aktuelle CPU Benchmark von SPEC wurde 2017 unter dem Namen SPEC CPU2017 veröffentlicht und ist der Nachfolger der bis dato im wissenschaftlichen Kontext einflussreichsten Benchmark SPEC CPU2006. Sie wurde in über 7000 Research Artikeln zitiert [18] [2]. Auch für Scientific Workflows ist die Benchmark interessant. SPEC Prozessor interessant –j, welche Berechnung  $\text{CPU2017 SPECspeed} = \text{time based metric}$  [2].

Industrie Standard - Kounev 2020 System Benchmark Kapitel 9 - 10 ! Kapitel 10 [15]

TPC Informationen [22].

### 2.3.4 open-source Benchmarks

Ein wichtiger Vertreter von open-source Benchmarks ist die Phoronix Test Suite, die es seit 2008 gibt und über 100 unterschiedliche Benchmark Tests anbietet. Auf der Seite openbenchmarking.org sind die Tests für alle freizugänglich.

... Trotz des breiten Angebots an Benchmark Tests gibt es keinen speziellen Benchmark Tests für Scientific Workflows.





# Kapitel 3

## Methodik

Das Kommandozeilen Tool welches für diese Abschlussarbeit entwickelt worden ist, ist unter folgendem Link auf github zu finden:

### 3.1 Anforderungen an das Programm

Das Kommandozeilen Tool wurde in der Programmiersprache Java geschrieben. Das Tool soll auf Linux Systemen funktionieren und dabei alle relevanten Hardware-Informationen des Computers auslesen, diese dann auf einer externen Datenbank speichern. Gleichzeitig soll anhand der Hardware-Informationen weitere relevante Benchmark Werte aus Hardware Datenbanken ausgelesen und ebenfalls gespeichert werden. Danach soll ein Ergebnis berechnet werden, wie schnell der Computer im Vergleich zu einem Referenz Computer performed. Es soll außerdem möglich sein, aus der Datenbank eine Gruppierung der vorhandenen Hardware Daten zu erstellen. Für das Auslesen der Hardware Information wurden die Standard Kommandozeilen Befehle `lscpu` und `xyz` genutzt. Als externe Datenbank zum Speichern der relevanten Hardware Informationen wurde PostgreSQL gewählt. Die Datenbank läuft extern auf einer virtuellen Maschine, welche von der Google Cloud Platform bereit gestellt wurde.

Somit ist es möglich das Programm überall ausführen können um die Daten zu speichern. Als Hilfstool zum bearbeiten der Datenbank wurde DBeaver genutzt. Im Internet existieren eine Vielzahl von Online-Datenbanken für Hardware Informationen. Allerdings sind die meisten spezialisiert auf die Performance von Computerspielen und daher auch sehr Grafikkarten lastig. Hinzu kommt, dass keine einzige gefundene Online-Datenbank eine Programmierschnittstelle anbietet. Daher wurde die HtmlUnit Bibliothek für Java benutzt zum sogenannten scraping von Webseiten. Dies bedeutet, dass auslesen des gesamten Html Textes von Webseiten. Mithilfe des Document Object Model (kurz DOM) ist es möglich, direkt Elemente aus dem Html Text, wie beispielsweise Tabellen direkt anzusprechen und zu verarbeiten.

Es wurden zur Testen über 10 Computer genutzt. Zwar bietet Google Cloud Platform auch virtuelle Maschinen an, das Problem bei diesen ist allerdings, dass durch die virtualisierung keine standardmäßigen Hardware Komponenten genutzt werden, sondern für virtualisierung prozesse angepasste Komponenten die von den Hardware Herstellern extra entwickelt worden sind und dementsprechend auch in kaum einer Online-Datenbank zu finden sind. Hierfür wurden dennoch interne Mikrobenchmarking Tests gemacht.

<http://cpudb.stanford.edu/processors> : wissenschaftliche Datenbank für CPUs, allerdings SPEC Daten von 2006, sprich viele CPUs Benchmarks gar nicht vorhanden.

## **3.2 Online-Datenbanken**

In den folgenden Unterkapiteln werden die genutzten Online-Datenbanken näher erläutert und analysiert auf welcher Grundlage der Performance Wert berechnet wurde.

### **3.2.1   cpu-world**

### **3.2.2   userBenchmark**

### **3.2.3   openBenchmarking**

## **3.3   Externe Datenbank**

Für die Speicherung der wesentlichen Hardware Informationen, sowie der gefundenen Performance Werte von anderen Online-Datenbanken, sowie dem Ergebnis des eigenen Mikro-Benchmarking Ergebnisses wurde eine PostgreSQL Datenbank entwickelt. Die Datenbank ist auch unabhängig von dem Kommandozeilen Tool zugreifbar.

## **3.4   Berechnung des Scores**



# Kapitel 4

## Auswertung

### 4.1 Validierung der Ergebnisse

### 4.2 Abweichungen

### 4.3 Klassifizierung der Ergebnisse

InputOutput, network, rechenintensive, speicherintensiv [6] datenintensive Workflows / read  
write operationen / input output data netzwerk sagt [4]



# Kapitel 5

## Stand der Forschung

Wie schon in dem Kapitel [2.3.1](#) beschrieben, gibt es bis dato keine gängige Metrik die im wissenschaftlichen Diskurs angewendet wird. Vor allem (insert alte Metrik MFLOPS) wird noch genutzt wobei die Nachteile hier deutlich überwiegen. Oftmals werden in wissenschaftlichen Arbeiten wiederum eigene Metriken oder Benchmarks kriiert, so dass es oft gar nicht dazu kommen kann das sich Metriken etablieren. (evntl auch in den Fazit/Diskussionsteil)

Dadurch das Computerspiele, aber auch das Mining für Kryptowährungen sehr populär geworden ist, gibt es unzählige Benchmark Werte für die Leistung der Komponenten von Computerspielen, vor allem spielt dort die Grafikkarte nochmal einen höheren Nutzen, also für (produktion / entwicklung).

Es gibt kaum wissenschaftliche Hardware Online Datenbanken. Leider gibt es dazu keine API Schnittstellen bei den restlichen Online Datenbanken, sodass die Extraktion relevanter Daten deutlich aufwendiger war.

Eine spannende Thematik in dem Bereich Benchmarking liefert aber das Thema Performance Prediction durch Machine Learning Ansätze : beschrieben –

Seit 2009 findet jedes Jahr die Technology Conference on Performance Evaluation and Bench-

marking (TCPTC) von der Non-Profit-Organisation Transaction Processing Performance Council (TCP) statt. Neben eigenen Benchmark Ergebnissen, auf die in Kapitel [2.3.3](#) eingegangen wurde, wird in den Konferenzen auf relevante Methodiken und neuen Ideen zum Thema Performance Evaluation und Benchmarking eingegangen. Viele der genutzten Quellen stammen aus Aufzeichnungen dieser Konferenzen.



# Kapitel 6

## Diskussion



# Kapitel 7

## Fazit



# Appendix A

## Abkürzungsverzeichnis

**ALU** Arithmetic logic unit (Rechenwerk)

**CD** Compact disc

**CPU** Central processing unit (Prozessor)

**DVD** Digital versatile disc

**GPU** Graphical processing unit (Grafikkarte)

**MHz** Megahertz

**MFLOPS** Floating point operations per second (Gleitkommaoperationen pro Sekunde)

**MIMD** Multiple Instruction, Multiple Data

**MIPS** Million instructions per second (Millionen Instruktionen pro Sekunde)

**MISD** Multiple Instruction, Single Data

**PC** Personal Computer

**SSD** Solid-state drive

**SPEC** Standard Performance Evaluation Corporation

**SIMD** Single Instruction, Multiple Data

**SISD** Single Instruction, Single Data

**TPC** Transaction Processing Performance Council

**TPCTC** Technology Conference on Performance Evaluation and Benchmarking

**USB** Universal serial bus

# Appendix B

## Lexikon

**Bus** Ein Bus in der Datenverarbeitung ist ein Leitungssystem zur Datenübertragung zwischen unterschiedlichen Komponenten [\[29\]](#).

**Mooresches Gesetz** Das Mooresche Gesetz von Gordon Moore aus dem Jahr 1965 sagt aus, dass sich ungefähr alle 18 Monate die Anzahl der Transistoren pro Flächeneinheit in einem Computerchip verdoppeln würden.

**von-Neumann-Rechner** Ein von-Neumann-Rechner ist ein Einkernprozessor, der auf der Von-Neumann-Architektur basiert.





# Appendix C

## Listings/Anhang

This is the appendix for code, that does not need to be provided directly inside the thesis.

### C.1 Configuration for Node A

Listing C.1: Configuration for Node A

```
{
  "nodeID" : "nodeA",
  "publicKey" : "<public key>",
  "encryptionAlgorithm" : "RSA",
  "machines" : [ "192.168.0.132", "192.168.0.165" ],
  "publisherPort" : 8000,
  "messagePort" : 8010,
  "restPort" : 8080,
  "location" : "52.515249, 13.326476",
  "description" : "Raspberry Pi Cluster in EN 004"
}
```



# Literaturverzeichnis

- [1] Malcolm Atkinson et al. *Scientific workflows: Past, present and future*. 2017.
- [2] James Bucek, Klaus-Dieter Lange, and Jóakim v. Kistowski. “SPEC CPU2017: Next-generation compute benchmark”. In: *Companion of the 2018 ACM/SPEC International Conference on Performance Engineering*. 2018, pp. 41–42.
- [3] Daniel Citron, Adham Hurani, and Alaa Gnadrey. “The harmonic or geometric mean: does it really matter?” In: *ACM SIGARCH Computer Architecture News* 34.4 (2006), pp. 18–25.
- [4] Christopher S Daley et al. “Performance characterization of scientific workflows for the optimal use of burst buffers”. In: *Future Generation Computer Systems* 110 (2020), pp. 468–480.
- [5] Anthony Danalis et al. “The scalable heterogeneous computing (SHOC) benchmark suite”. In: *Proceedings of the 3rd Workshop on General-Purpose Computation on Graphics Processing Units*. 2010, pp. 63–74.
- [6] Ewa Deelman et al. “The future of scientific workflows”. In: *The International Journal of High Performance Computing Applications* 32.1 (2018), pp. 159–175.
- [7] Rudolf Eigenmann and David J Lilja. “Von neumann computers”. In: *Wiley Encyclopedia of Electrical and Electronics Engineering* 23 (1998), pp. 387–400.

- [8] Christopher Elford, Dippy Aggarwal, and Shreyas Shekhar. “Revisiting Issues in Benchmark Metric Selection”. In: *Technology Conference on Performance Evaluation and Benchmarking*. Springer. 2020, pp. 35–47.
- [9] Paul Fortier and Howard Michel. *Computer systems performance evaluation and prediction*. Elsevier, 2003.
- [10] John L Gustafson and Quinn O Snell. “HINT: A new way to measure computer performance”. In: *Proceedings of the Twenty-Eighth Annual Hawaii International Conference on System Sciences*. Vol. 2. IEEE. 1995, pp. 392–401.
- [12] Karl Huppler. “The art of building a good benchmark”. In: *Technology Conference on Performance Evaluation and Benchmarking*. Springer. 2009, pp. 18–30.
- [13] Lizy Kurian John. “More on finding a single number to indicate overall performance of a benchmark suite”. In: *ACM SIGARCH Computer Architecture News* 32.1 (2004), pp. 3–8.
- [14] Jóakim v. Kistowski et al. “How to build a benchmark”. In: *Proceedings of the 6th ACM/SPEC International Conference on Performance Engineering*. 2015, pp. 333–336.
- [15] Samuel Kounev, Klaus-Dieter Lange, and Jóakim von Kistowski. *Systems Benchmarking*. Springer, 2020.
- [16] William Kramer. “How to measure useful, sustained performance”. In: *State of the Practice Reports*. 2011, pp. 1–18.
- [17] David J Lilja. *Measuring computer performance: a practitioner’s guide*. Cambridge university press, 2005.
- [18] Ankur Limaye and Tosiron Adegbiya. “A workload characterization of the SPEC CPU2017 benchmark suite”. In: *2018 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS)*. IEEE. 2018, pp. 149–158.

- [19] Ji Liu et al. “A survey of data-intensive scientific workflow management”. In: *Journal of Grid Computing* 13.4 (2015), pp. 457–493.
- [21] John R Mashey. “War of the benchmark means: time for a truce”. In: *ACM SIGARCH Computer Architecture News* 32.4 (2004), pp. 1–14.
- [22] Raghunath Nambiar and Meikel Poess. “Industry standards for the analytics era: Tpc roadmap”. In: *Technology Conference on Performance Evaluation and Benchmarking*. Springer. 2017, pp. 1–8.
- [23] John D Owens et al. “GPU computing”. In: *Proceedings of the IEEE* 96.5 (2008), pp. 879–899.
- [24] Sean M Pieper, JoAnn M Paul, and Michael J Schulte. “A new era of performance evaluation”. In: *Computer* 40.9 (2007), pp. 23–30.
- [25] Salonik Resch and Ulya R Karpuzcu. “Benchmarking quantum computers and the impact of quantum noise”. In: *arXiv preprint arXiv:1912.00546* (2019).
- [26] Daniel Schlegel. “Deep machine learning on Gpu”. In: *University of Heidelber-Ziti* 12 (2015).
- [27] Rajesh Singh. “An approach to enhance performance of Computer”. In: *International Journal of Science & Engineering Development Research* 1.2 (2016), pp. 298–302.
- [28] James E. Smith. “Characterizing computer performance with a single number”. In: *Communications of the ACM* 31.10 (1988), pp. 1202–1206.
- [29] Andrew S. Tanenbaum. *Computerarchitektur: Strukturen, Konzepte, Grundlagen*. 5. Auflage. Pearson Studium, 2006.
- [30] Michael Bedford Taylor. “The evolution of bitcoin hardware”. In: *Computer* 50.9 (2017), pp. 58–66.
- [31] Thomas N Theis and H-S Philip Wong. “The end of moore’s law: A new beginning for information technology”. In: *Computing in Science & Engineering* 19.2 (2017), pp. 41–50.

# Literaturverzeichnis (Online)

- [11] Peter Haugen et al., eds. *A Basic Overview of Commonly Encountered types of Random Access Memory (RAM)*. 2000. URL: <https://www.rose-hulman.edu/Class/ee/yoder/ece332/Papers/RAM%20Technologies.pdf> (visited on 03/10/2021).
- [20] Bertram Ludäscher, Shawn Bowers, and Timothy McPhilips, eds. *Scientific Workflows*. 2009. URL: [https://link.springer.com/referenceworkentry/10.1007%2F978-0-387-39940-9\\_1471](https://link.springer.com/referenceworkentry/10.1007%2F978-0-387-39940-9_1471) (visited on 12/10/2021).