Big Data Benchmarking

Tim Horgas

Hochschule für Angewandte Wissenschaften Hamburg, Department Informatik,
Berliner Tor 7, 20099 Hamburg, Deutschland,
tim.horgas@haw-hamburg.de

Zusammenfassung. Diese Arbeit beschäftigt sich mit den Anforderungen von Benchmarks für den Bereich Big Data und beschreibt den aktuellen Stand von existierenden Benchmark-Lösungen. Dazu wird eine kurze Übersicht der aktuellen Big Data Benchmarks ergestellt und darauf aufbauend die Notwendigkeit eines Konzepts für die Auswahlentscheidung von Big Data Benchmarks vorgestellt.

Schlüsselwörter: Big Data, Benchmarking

Inhaltsverzeichnis

1 Einleitung und Motivation

Big Data ist aktuell ein Trendwort in der Informatik und eines der am häufigsten beachteten Themenfelder der Informationswissenschaft. Der Begriff Big Data ist besonders durch Anwendungen von Unternehmen wie zum Beispiel Google und Facebook geprägt, da diese bedingt durch das Internet täglich Datenmengen im zweistelligen Petabyte-Bereich speichern, verarbeiten und analysieren (vgl. [DG08], S. 1). Jedoch nutzen auch Unternehmen wie zum Beispiel Wal-Mart Big Data (vgl. [PZ14], S. 2), um Mehrwert für das Unternehmen zu schaffen. Obwohl Wal-Mart im Vergleich zu Internetunternehmen einer tendenziell traditionellen Branche zuzuordnen ist, besteht auch in diesem Fall zunehmend die Möglichkeit zur Nutzung von Big Data. Diese Möglichkeit zur Generierung von Mehrwehrt und die immer stärkere Verfügbarkeit von Daten, werfen für Unternehmen Fragen nach technischen Lösungen für die Informationsgenerierung aus Daten auf, die die Eigenschaften von Big Data aufweisen.

Weitere aufkommende Trendwörter wie das Internet of Things und Industrie 4.0 sind Themengebiete, die Überschneidungen zum Kontext Big Data aufweisen und für die es nötig wird, die existierenden Softwarelösungen als Anwendungen bereitzustellen und zu integrieren. Eine große Herausforderung ist dabei, die Sotwarelösungen für den Anwendungsfall zu testen und miteinander zu vergleichen, um Unternehmen bei der Auswahlentscheidung zu unterstützen. Diese Vergleiche der Anwendungen in Form von Benchmarks müssen dabei den Anforderungen von Big Data gerecht werden. Außerdem müssen für das Testen und den Vergleich von Anwendungen Benchmarks implemementiert werden, die eine gute Vergleichbarkeit der auszuführenden Operationen und der Ergebnisse aufweisen.

Die Nutzung von Big Data ist mittlerweile in vielen unterschiedlichen Branchen möglich und bedient damit auch grundsätzlich verschiedene Anwendungsfälle. Einer der ersten praktischen Anwendungsfälle ist durch die Mitarbeiter Jeffrey Dean und Sanjay Ghemawat von Google beschrieben, die bereits in 2008 ein Beispiel für die Verarbeitung von vielen Terrabytes an Daten über tausende Maschinen täglich genannt haben (vgl. [DG08], S. 1). Die sich ändernden Anforderungen von Big Data haben zu der Entwicklung von Programmiermodellen wie zum Beispiel MapReduce geführt, die zum Beispiel von Michael Stonebraker et. al durch Benchmarks mit den bereits existierenden Lösungen verglichen wurden (Siehe [SAD+10]). Big Data Benchmarks müssen also damit neben vielen verschiedenen Anwendungsfällen auch unterschiedliche Programmiermodelle vergleichen können. Daher benötigen Big Data Benchmarks im Vergleich zu etablierten Benchmarks (beispielsweise Benchmarks aus dem Standardwerk von Jim Gray ([Gra92])) die Integration von neuen Anforderungen bedingt durch Big Data. In den folgenden Abschnitten werden die Anforderungen von Big Data und deren Veränderungen für das Benchmarking beschrieben. Im Anschluss wird der aktuelle Stand von Big Data Benchmarks geschildert.

2 Big Data

Der Begriff Big Data ist bis heute nicht einheitlich definiert, jedoch ist grundsätzlich die Beschreibung mit den Begriffen Volume, Velocity und Variety (drei V's) die am häufigsten verwendete und weitestgehend akzeptierte Definition. Dazu gibt es Erweiterungen durch Begriffe wie zum Beispiel Veracity (vier V's) und Value (fünf V's), die jeweils weitere Aspekte zur Definition von Big Data hinzufügen.

Die Definition der drei V's wurde ursprünglich in Bezug auf E-Commerce vorgestellt und führt in Bezug auf Daten die drei Dimensionen Volume, Velocity und Variety ein (Siehe [Lan01], S.1). Diese ursprüngliche Definition war damals noch nicht dem Begiff Big Data zugeordnet, jedoch lassen sich die Inhalte auf diesen übertragen. Das erste V mit der Bezeichnung Volume beschreibt Datenmengen, die aufgrund ihrer Größe nicht mehr von traditionellen Datenbanksystemen verarbeitet werden können und in der Regel eine Speicherung in einem Computer-Cluster benötigen (vgl. [Fre14], S.9). Volume bezieht sich heute im Unternehmenskontext auf Datenmengen im Petabyte-Bereich oder mehr. Die spezifischen Probleme für Big Data sind dabei zum einen in der schnellen Speicherung und Verteilung der Daten im Cluster und in der Aufbereitung der Daten für die spätere Informationsextraktion begründet. Mehr Daten bedeuten nicht automatisch mehr Informationen, die Informationen müssen explizit aus den Daten generiert werden (vgl. [ZdP+13], S. 4). Durch die Größe der Datenmengen braucht es neue Anwendungen und Algorithmen, da diese nicht mehr effizient von traditionellen Datenbanken verarbeitet werden können.

Das zweite V mit der Bezeichnung Velocity kann in zwei Komponenten aufgeteilt werden, dazu gehören die Geschwindigkeit der Datenerzeugung und die Geschwindigkeit der Datenverarbeitung. Die Geschwindigkeit der Datenerzeugung ist besonders durch Social-Media-Anwendungen und Sensordaten im Vergleich zu traditionellen Anwendungen (beispielsweise ERP-Systemen) deutlich erhöht und braucht deshalb besondere Methoden zur Speicherung (vgl. [PZ14], S. 2). Dementsprechend müssen auch OLAP (Online Analytical Processing) und Analyse-Tools diesen Bedingungen gerecht werden, um Echtzeitanforderungen für Anwendungen bedienen zu können.

Das dritte V mit der Bezeichnung Variety bezeichnet in der Ursprungsdefinition die starke Verschiedenheit der Daten in Bezug auf Formate, Strukturen und inkonsistente Semantiken (vgl. [Lan01], S. 2). Die Daten kommen aus unterschiedlichen Quellen und sind in der Regel nicht für Datenbanken normalisiert. Beispiele dafür sind Transaktionsdaten im klassischen Börsenhandel und Protokolldaten von Servern (vgl. [Fre14], S. 12). Die bisher beschriebenen drei V's bilden die Grundlage für die Definition von Big Data. Es gibt noch weitere Ansätze zur Definition von Big Data, die als Erweiterungen zu den bereits beschriebenen Komponenten gesehen werden können.

Big Data hat auf Basis der drei V's nach ([SR13], S. 1) vier Herausforderungen und damit auch vier entprechende Anwendungsfälle, die im folgendem aufgezählt und erläutert sind.

1. "big volumes of data, but small analytics"

2. “big analytics on big volumes of data”
3. “big velocity”
4. “big Variety”

Die erste Herausforderung ist “big volumes of data, but small analytics”, womit der Anwendungsfall von SQL-Analysen auf sehr großen Datenmengen gemeint ist. Nach [SAD+10], S. 10) können diese Anforderungen von verteilten DBMS übernommen werden, wobei mit dieser Aussage der Big Data Definition von Volume auf den ersten Blick widersprochen wird. Allerdings muss beachtet werden, dass die Speicherung von sehr großen Datenmengen in traditionellen DBMS mit sehr viel Konfigurations- und Optimierungsaufwand verbunden ist (Siehe [PPR+09], S. 13). Für Big Data geeignete Systeme wie Hadoop haben im Vergleich deutlich weniger Konfigurations- und Optimierungsaufwand, besonders im Fall von Skalierung. Der zweite Herausforderung ist “big analytics on big volumes of data”. Dieser beinhaltet Anwendungsfälle im Bereich von komplexen Analysen wie zum Beispiel Clustering, Regression und Machine Learning. “Big velocity” stellt die dritte Herausforderung und beschreibt den Anwendungsfälle wie zum Beispiel elektronisches Trading, Realtime-Werbung auf Websiten und mobiles Social-Networking. Dabei werden in kurzer Zeit sehr große Datenmengen erzeugt, wobei diese direkt gespeichert und verabeitet werden müssen. Die vierte Herausforderung ist “big Variety”. Damit ist das Sammeln von Daten aus immer größer werdenden Anzahl an Datenquellen gemeint.

Zusätzlich zu den drei V’s wurden durch Mitarbeiter von IBM zwei weitere V’s eingeführt, die zusätzlich zu der Basisdefinition weitere Aspekte einführen. Tendenziell basieren diese Aspekte jedoch auf den drei V’s und sind eher eine Erweiterung als eine Grunddefinition. Das vierte und das fünfte V werden in den folgenden Abschnitten beschrieben. Das vierte V mit der Bezeichnung Veracity umfasst die Qualität und die Vertrauenswürdigkeit der Daten (vgl. [ZdP+13], S. 14). Inhaltlich verschiedene und vorallem fehlerhafte Daten müssen erkannt und bereinigt werden, um zum Beispiel die Richtigkeit von Analysen auf Big Data gewährleisten zu können. Dieser Schritt ist auch im traditionellen ETL-Prozess bereits bekannt, jedoch führen die Eigenschaften der drei V’s auch hier zu neuen Herausforderungen. Bedingt durch das immer größer werdende Datenvolumen und die hohe Geschwindigkeit der Datenerzeugung ist dabei auch eine stärkere Automatisierung der benötigten Prozesse nötig. Außerdem bedarf es noch mehr Aufmerksamkeit auf den ETL-Prozess, da durch Big Data Anwendungen die Anzahl der Datenquellen weiter steigt.

Die beschriebenen vier V’s sollen demnach zu dem entscheidenden fünften V führen, dem Value. Der Value kann aus dem immer größer werdenden Volume, Variety und Velocity mit Berücksichtigung der Veracity generiert werden und ist letztendlich für Unternehmen der entscheidene Faktor zur Investition in Technologien zur Speicherung und Analyse von Daten im Kontext von Big Data. Es muss dabei beachtet werden, dass aus der reinen Speicherung von Daten in den meisten Fällen kein ökonomischer Gewinn erzielt werden kann (vgl. [ZdP+13], S. 4). Erst Analysen können als Produkt verkauft werden und so zur Wertsteigerung eines Unternehmens beitragen. Die zentrale Fragestellung für Unternehmen in

Bezug auf Big Data ist also, inwiefern Big Data Technologien Mehrwert für das Unternehmen generieren können. Die fünf V's können also als Basis für Ansätze zur Zieldefinition von Benchmarking im Kontext Big Data genommen werden. Da der Value komplett auf den restlichen vier V's basiert, könnte sich dieser zum Beispiel in Form einer Kennzahl für eine Abschlussentscheidung zwischen verschiedenen Big Data Technolgien eignen.

3 Benchmarking

Benchmarking ist in der Informatik eine Methode zum Vergleich von unterschiedlichen Computer- oder Softwaresystemen, dieser Vergleich dient letztendlich als Unterstützung zur Auswahl für eines der getesteten Systeme (vgl. [Gra93], S. 1). Beim Benchmarking von traditionellen SQL-Systemen werdem dabei in der Regel Datenbanksysteme verglichen, jedoch kommen besonders durch Trends wie Big Data und die NoSQL-Bewegung vermehrt Frameworks zum Einsatz, die nicht als reine Datenbanksysteme einzuordnen sind. Für die konkrete Gegenüberstellung der Systeme müssen Testdatensätze, Testoperationen und geeignete Metriken ausgewählt werden, um mit diesen dann eine Vergleichbarkeit der Systeme zu ermöglichen. Diese bilden den Benchmark, der häufig auch als Workload bezeichnet wird.

Grundsätzlich soll jeder Benchmark eine konkrete Frage beantworten: *"Welches System soll ich kaufen?"* (Siehe [Gra93], S. 1). Diese Frage kann dann durch Benchmarks mit folgender Anwort beantwortet werden: *"Das System, das die Aufgabe mit dem niedrigsten Cost-Of-Ownership erreicht."* Cost-Of-Ownership sind dabei die Gesamtkosten eines Systems, gemessen über einen bestimmten Zeitraum. Diese beinhalten Programmierkosten, Hardwarekosten, Softwarekosten, laufende Kosten und Projektrisiken (vgl. [Gra93], S. 1). Quantitative Kenngrößen, zu denen die Hardwarekosten, Softwarekosten und die Computerperformance gehören, können miteinander verglichen werden (vgl. [Gra93], S. 1). Bei den nicht quatifizierbaren Kenngrößen wie den Programmierkosten, laufenden Kosten und vorallem den Projektrisiken ist ein Vergleich deutlich schwieriger oder sogar im voraus nicht möglich. Besonders Projektrisiken sind in vielen Fällen nicht vorhersehbar und können sogar dazu führen, dass der Benchmark teurer als das eigentliche Projekt werden kann.

Die quantitative Computerperformance wird in der Regel mit Durchsatzmetriken in Form von (work/second) bestimmt, wie zum Beispiel Transaktionen pro Sekunde (tps). Hierbei ist jedoch zu beachten, dass solche Metriken oft keine Aussage über die konkrete Effektivät der Anwendung machen. Die quantitative Computerperformance kann dann den quantitativ bestimmten Preis gegebübergestellt werden, zum Beispiel Preis/tps. Solche Benchmarks lassen einen groben Vergleich der relativen Systemperformance zu (Siehe [Gra93], S. 1). Die Systemperformance von Datenbanksystemen ist stärker von den internen Algorithmen als von der reinen Hardwarekonfiguration abhängig (Siehe [Gra93], S. 2). Die Effektivität von Algorithmen hängt wiederum stark von den spezifischen Anwendungsfall ab.

Dieser Umstand führt (nach [Gra92], S. 2) zu der Notwendigkeit von Domain-specific Benchmarks, die domänenabhängige Tests von Systemen erlauben und damit auch einen Vergleich von "Äpfeln mit Birnen" umgehen sollen. Der Workload von Domain-specific Benchmarks umfasst typische Anwendungen für die jeweils spezifische Domäne, zum Beispiel Cookie-Analysen in der Domäne E-Commmerce. Die bekanntesten Organisationen für Domain-specific Benchmarks sind SPEC (System Performance Evaluation Cooperative) und TPC (Transaction Processing Performance Council), wobei sich besonders letzteres für Anwendungsfälle im Kontext von Datenbanken spezialisiert hat. Domain-specific Benchmarks sollten vier Kriterien erfüllen (Siehe [Gra93], S. 3f):

- Relevanz
- Portabilität
- Skalierbarkeit
- Einfachheit

Ein Domain-specific Benchmark muss dabei Auskunft über die maximale Performance der zu testenden Systeme geben und eine Preis/Performance-Metrik im Sinne von Cost-Of-Ownership für den spezifischen Anwendungsfall anbieten, um relevant zu sein. Portabilität ist in diesem Fall die Fähigkeit, den Domain-specific Benchmark auf unterschiedlichen Systemen und Architekturen implementieren zu können. Skalierbarkeit beschreibt die Fähigkeit, den Benchmark sowohl auf großen als auch auf kleinen Computer-Systemen ausführen zu können und außerdem die Möglichkeit zur Skalierung des Benchmarks bei einer Skalierung des zu testenden Systems. Außerdem sollte ein skalierbarer Benchmark parallele Systeme unterstützen. Ein Domain-specific Benchmark muss einfach zu verstehen sein, da dieser sonst aufgrund von fehlender Plausibilität schwer zu benutzen wäre.

Zusätzlich zu den oben von ([Gra92]) beschriebenen Kriterien führt TPC-DS (TPC Decision Support Benchmark) als Beispiel für einen aktuellen Standardbenchmark drei weitere Eigenschaften mit ein, die besonders für die Standardisierung notwendig sind (vgl. [Tra15a], S. 7):

- Verfügbarkeit der Systeme für Benutzer
- Marktrelevanz
- "real world"-Anwendungen

Die zu testenden Systeme müssen allgemein für die Benutzer verfügbar sein oder zumindest für einen Benchmark von einem Produkthersteller bereitgestellt werden. Wenn diese Eigenschaft nicht vorhanden ist, lassen sich bereits durchgeführte Benchmarks nicht wiederholen und eignen sich damit nicht für einen Standardbenchmark. Die Marktrelevanz beschreibt die Eigenschaft, dass die zu testenden Technologien bereits von vielen Nutzern in dem produktspezifischen Marktsegment verwendet werden sollten. An dieser Stelle ist es jedoch nicht quantitativ definiert, ab wie vielen Benutzern ein System die benötigte Martkrelevanz hat. Da Standardbenchmarks teilweise sowohl finanziell als auch durch

direkte Mitarbeit von Unternehmen unterstützt werden, ergibt sich die Marktrelevanz oft durch die Anzahl der Unterstützer und den betriebenen Aufwand für den Standardisierungsprozess des Benchmarks. Eine entscheidende zusätzliche Eigenschaft für Benchmarks ist die Verwendung von Echtwelt-Anwendungen beim Benchmark. Damit soll eine Verbindung zum wirtschaftlichen Kontext geschaffen werden und die Vermeidung von Systemen, die nur auf gute Ergebnisse des Benchmarks ausgerichtet sind.

4 Big Data Benchmarking

Der erste Abschnitt (4.1) beschäftigt sich mit den Herausforderungen von Benchmarking im Kontext Big Data und zieht dabei Vergleiche zu den im vorherigen Abschnitt (3) beschriebenen Eigenschaften von Benchmarks. Die darauf folgenden Abschnitte (4.2 – 4.4) stellen Ansätze von Benchmarking für Big Data Anwendungen vor. Der erste beschriebene Ansatz und aktueller Industrie-Standard ist TPCx-HS (Siehe [Tra15b]). Zu den weiteren Ansätzen gehören BigBench und BigDataBench. Es werden dabei die grundsätzlichen Eigenschaften dieser unterschiedlichen Benchmark-Ansätze beschrieben und bewertet.

4.1 Herausforderungen von Benchmarking im Bereich Big Data

Die zentrale Herausforderung von Benchmarking im Kontext von Big Data entsprechend der Definition der fünf V's ist nach den Erkenntnissen aus Abschnitt 2 die Möglichkeit zur Bestimmung des Values eines Big Data Tools, beziehungsweise eine Entscheidungshilfe für die Fragestellung zu bilden, welches System am besten für den jeweiligen Anwendungsfall geeignet ist. Der Kontext Big Data stellt nach ([ABM13], S. 1) fünf neue Anforderungen an Benchmarks, zusätzlich zu den in Abschnitt 3 beschriebenen Anforderungen traditioneller Benchmarks:

- Verarbeitung von Petabytes an Daten
- Unterstützung von komplexen Datentypen (zum Beispiel Dokumente, Graphdaten und Bilddateien)
- Komplexe Anwendungen (zum Beispiel Machine Learning und Graphalgorithmen)
- Fehlertoleranz in Bezug auf Hardwareausfall
- geringe Latenz trotz größerer Datenmengen

Die erste Herausforderung für das Benchmarking im Kontext Big Data ist damit die Beschaffung beziehungsweise die Generierung von Daten, die dem Kontext Big Data zuzuordnen sind. Vergleicht man die obigen Anforderungen mit denen der 5 V's, dann ist ersichtlich, dass Volume und Variety in beiden Definitionen vorhanden sind. Im Fall einer ständigen Datengenerierung kommen auch beide Komponenten von Velocity vor (Geschwindigkeit der Datenerzeugung und Verarbeitung). Das von IBM eingeführte vierte V (Veracity) ist dabei durch Datengenerierung schwerer umzusetzen. Die Datenqualität kann bei

einer Datengenerierung künstlich verändert werden, eine variierende Vertrauenswürdigkeit der Daten ist allerdings schwer zu generieren. Dies kann wenn überhaupt nur mit „echten“ Daten oder mit aus Echtdaten generierten Daten umgesetzt werden. Das fünfte V (Value) ist ebenfalls schwer mit einem Benchmark abzudecken, jedoch bringt der Einsatz von typischen Operationen eines Domain-specific Benchmarks mit Beispieldaten eventuell Erkenntnisse über eine mögliche Verwendung für Big Data Anwendungen. Außerdem kann es sein, dass bestehende Metriken durch anderere Metriken ersetzt werden müssen. Nach ([KTS+15]) kann es sich zum Beispiel lohnen, für die Verarbeitung von Big Data generell andere Computer-Architekturen zu verwenden, da diese eventuell besser für Anwendungen im Kontext Big Data geeignet sind. An dieser Stelle müsste geprüft werden, ob die bereits verwendeten Metriken unter anderen Architekturen bei einem Benchmark noch relevant sind. Zusätzlich müssten Metriken für die Einfachheit der Bedienung, die Einfachheit der Programmierung und die Einfachheit der Operationen geschaffen werden, um tendenziell qualitative Kriterien wie zum Beispiel Programmierkosten für dien Entscheidung über Projekte im Kontext Big Data besser abschätzen zu können.

4.2 TPCx-HS

TPCx-HS ist ein Benchmark zum Vergleich von Big Data Tools, die mit der API des Hadoop File Systems kompatibel sind. Dabei werden Hardware- und Softwareeigenschaften gemessen und die Performance und die Performance im Verhältnis zum Preis durch Metriken bestimmt. Die Daten werden über einen internen Datengenerator erzeugt und sind im Key-Value-Format.

Die primäre Performance-Metrik misst den Datendurchsatz und ist wie folgend aufgebaut (Siehe [Tra15b], S. 15): HSph@SF = $\frac{SF}{T/3600}$. SF ist dabei ein Scale-Faktor, für den Datenmengen aus verschiedenen Größen einsetzbar sind (1TB, 3TB, 10TB, 30TB, 100TB, 300TB, 1000TB, 3000TB und 10000TB). T ist die in Sekunden gemessene Zeit für den Durchlauf. Die zweite Metrik ist eine Preis/Performance-Metrik und wird wie folgend berechnet: \$/HSph@SF = $\frac{P}{HSph@SF}$, wobei P die Total Cost of Ownership darstellt. Die Total Cost of Ownership setzen sich aus den Hard- und Softwarekosten der beteiligten Geräte und des Netzwerkes und die Kosten für alle Produkte, die zum Aufbau der Testumgebung nötig sind (Siehe [Tra15b], S. 17). Desweiteren sind noch Metriken in Bezug auf Energieverbrauch möglich.

TPCx-HS deckt damit die Big Data Anforderung Volume ab. Die zu erzeugenden Datenmengen entsprechen der aufgestellten Definition von Big Data, durch den Benchmark lassen sich Rückschlüsse auf die benötigte Hardware für den Anwendungsfall mit entsprechender Datenmenge ziehen. Der Benchmark ist skalierbar, portabel und einfach. TPCx-HS basiert stark auf Hadoop, es lassen sich keine von Hadoop unabhängigen Datenbanksysteme testen. Hadoop ist im Kontext von Big Data eine der entscheidenden Technologien, weshalb der TPCx-HS Benchmark für IT-Unternehmensarchitekuren grundsätzlich relevant ist. Allerdings lässt die explizite Einschränkung auf Hadoop und dessen kommerzielle Anbieter nur für relativ beschränkte Anwendungsfälle Rückschlüsse

in Bezug auf die Verwendbarkeit zu. Außerdem sind die Anforderungen Velocity und Variety nicht komplett abgedeckt. Es werden weder Datenströme über einen bestimmten Zeitraum gemessen, noch ist eine Variabilität der Daten oder Datenformate gegeben. Fairerweise muss jedoch beachtet werden, dass für Big Data nicht unbedingt alle Anforderungen auf einmal erfüllt sein müssen. Für das „Phänomen“ Big Data reicht auch eine der Anforderungen von den 5 V's, da wie bereits erwähnt, keine einheitliche Definition von Big Data existiert. TPCx-HS stellt insgesamt einen relativ spezialierten Big Data Benchmark dar. Wie bereits erwähnt fehlen einige Faktoren, um den kompletten Kontext Big Data abzudecken.

4.3 BigBench

BigBench ist einer der ersten Versuche zur Entwicklung eines Standardbenchmarks im Kontext Big Data und zielt auf den Vergleich von DBMS und MapReduce Systemen ab, die Lösungen für Big Data darstellen (Siehe [GRH+13], S. 9). BigBench umfasst die Big Data Eigenschaften Volume, Velocity und Variety (Siehe [GRH+13], S. 1). Die Daten werden über einen Scale-Faktor generiert und sind dabei anteilig strukturiert, semistrukturiert und unstrukturiert (vgl. [GRH+13], S. 1). Der Anwendungsfall ist ein Produkthändler und lässt sich damit eher in den klassischen Anwendungsfall eines Produktverkäufers einordnen. Die strukturierten Daten sind weitesgehend aus dem TPC-DS Benchmark übernommen, die semistrukturierten Daten sind im Key-Value-Format und ähnlich zu Apaches Webserver Log-Format (Siehe [GRH+13], S. 2). Die unstrukturierten Daten sind durch unstrukturierten Text in Form von Produktrezensionen umgesetzt, dessen Generierung basiert auf Beispieldaten (vgl. [GRH+13], S. 2).

Der Benchmark besteht aus 30 Queries, mit 10 deklarativen Queries in Form von SQL und 7 prozeduralen Abfragen in Form von MapReduce (Siehe [GRH+13], S. 9). Die restlichen 13 Abfragen sind eine Mischung aus deklarativen und prozeduralen Abfragen in Form von Pig-Queries. Big Data spezifische Anwendungen sind dabei die Weblog- und Produktrezensionssnalysen. Die eigesetzte Metrik ist die Zeit der Ausführung der jeweiligen Queries, allerdings ist zu beachten, dass je nach Abfrageform unterschiedliche Ausführungsdauern in den verschiedenen Phasen (beispielsweise initiales Laden) vorkommen. MapReduce und parallele DBMS eignen sich teilweise für verschiedene Anwendungsfälle (vgl. ([SAD+10], S. 8), deshalb kann ein reiner Zeitvergleich irreführend sein. BigBench ist insgesamt skalierbar und portabel, außerdem deckt der Benchmark die drei V's des ursrpünglichsten Definitionsversuch ab. Es aber gibt keinen Mechanismus für das Testen von Fehlertoleranz in Bezug auf Hardwareausfall. Auch ist keine Metrik zum Vergleich von unterschiedlichen Systemen in Bezug auf den Preis vorhanden.

4.4 BigDataBench

BigDataBench ist aktuell einer der komplettesten Benchmarks mit insgesamt 5 Anwendungsdomänen, 14 Datensets und 33 Workloads (vgl. [WZL+14], S. 6).

Die 5 Anwendungsdomänen sind zum einem Suchmaschinen, Soziale Netzwerke und E-Commerce aus dem Kontext Internet Services (vgl. [WZL+14], S. 4) und Bioinformatik und Multimedia-Datenanalyse (vgl. [Zha15]). Alle 14 Datensets sind auf Basis von „real world" Daten, 8 davon sind dabei skalierbar und jeweils strukturiert, semi-strukturiert oder unstrukturiert ([Zha14], S. 6). Die skalierbaren Datensets werden durch einen Datengenerator an die Eigenschaften der vier V's von Big Data angepasst (vgl. [MLG+14], S. 8). In Tabelle 1 sind die skalierbaren Datensets aufgeführt.

Tabelle 1. Basisdaten für des BigDataBench Datengenerators

Quelle: [LGJ+14], S. 3

Datenset	Beschreibung
Wikipedia Entries	4,300,000 English articles (unstrukturierter Text)
Amazon Movie Reviews	7,911,684 reviews (semi-strukturierter Text)
Google Web Graph	875713 nodes, 5105039 edges (unstrukturierter Graph)
Facebook Social Network	4039 nodes, 88234 edges (unstrukturierter Graph)
E-commence Transaction	Table1: 4 columns, 38658 rows. Table2: 6 columns, 242735 rows (strukturierte Tabellen)
Person Resumes Data	278956 resumes (semi-strukturierte Tabellen)
Genome sequence data	unstrukturierter Text
Assembly of the human genome	unstrukturierter Text

Die insgesamt 33 verschiedenen Workloads sind jeweils einer Anwendungsdomäne zugeteilt. Zum Beispiel sind die Operationen WordCount und PageRank der Anwendungsdomäne SearchEngine zugeordnet, SQL-Queries wie Join und OrderBy sind nur in der Anwendungsdomäne E-Commerce vorhanden (Siehe [LGJ+14], S. 37). Außerdem lassen sich mit BigDataBench komplexere Analysen wie zum Beispiel Kmeans in der Anwendungsdomäne Social Network testen (Siehe [LGJ+14], S. 37).

BigDataBench hat zwei verschiedene Kategorien von Metriken, dazu gehören vom Benutzer bemerkbare Metriken und Architekturmetriken (Siehe [WZL+14], S. 7). Die vom Benutzer bemerkbaren Metriken sind von den ausgeführten Operationen anhängig und damit nicht für einen Vergleich von unterschiedlichen Architekturen geeignet. Die Durchsatzmetrik ist die Anzahl der Requests pro Sekunde (RPS), die Latenz wird über die Anzahl Operationen pro Sekunde (OPS) gemessen (Siehe [WZL+14], S. 7). Außerdem gibt es in dieser Kategorie noch die DPS-Metrik, eine Kennzahl zu Messung der verarbeiteten Daten pro Sekunde (DPS = $\frac{input\ datasize}{total\ processing\ time}$). Die Architekturmetriken sind MIPS und Cache-MPKI.

Wenn man an dieser Stelle die vier Kriterien nach ([Gra92], S. 3) anwendet ist zu beachten, dass in Bezug auf die Relevanz von BigDataBench keine Preis-Metrik vorhanden ist. Im Gegensatz zu BigBench und TPC-HS lassen sich aber Operationen im von Stonebraker als „big analytics on big volumes of da-

ta“ (vgl. [SR13], S. 10) bezeichneten Anwendungsfall testen. BigDataBench ist portabel, skalierbar und einfach. Allerdings ist zum Beispiel die Architekturmetrik MIPS nach ([Gra92], S. 4) weder portabel noch skalierbar, außerdem ist der Hauptkritikpunkt von MIPS die Irrelevanz. Die Irrelevanz bezieht sich auf die Aussage, dass MIPS keine sinnvolle Computerarbeit misst, da ein MIPS-Rating auf demselben Computer sich mit unterschiedlichen Compilern um Faktor drei unterscheiden kann (vgl. [Gra92], S. 4). Hinzu kommt, dass BigDataBench keine Metriken für die Einfachheit der Benutzung, Einfachheit der Programmierung und Einfachheit der Operationen bereitstellt.

5 Ausblick: Entwicklung einer Integrationsplattform

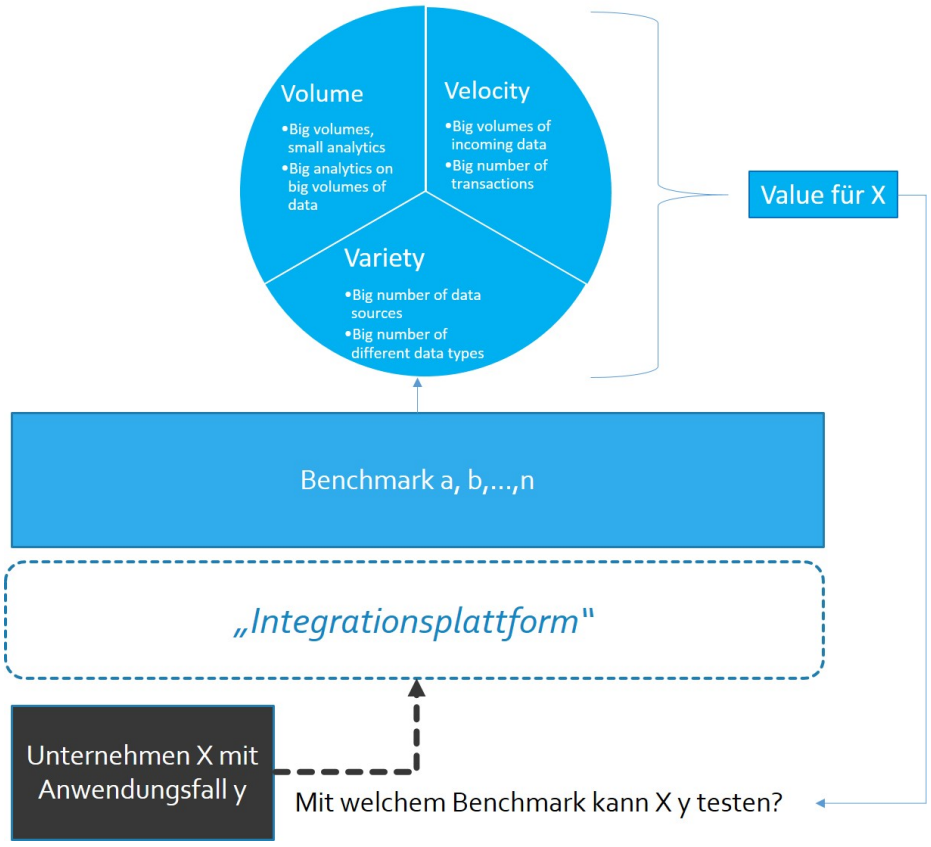

Abb. 1. Integrationsplatform
(Quelle: Eigene Darstellung)

Aktuelle Big Data Benchmarks bedienen jeweils unterschiedliche Anforderungen und haben teilweise verschiedene Ziele. Es werden oft nur einzelne Big Data Eigenschaften durch die Benchmarks abgedeckt, an dieser Stelle müsste durch eine Integrationsplatform eine Übersicht erstellt werden, die eine eindeutige Zuordnung im Kontext Big Data zulässt. Die Integrationsplatform für Big Data Benchmarks soll eine Auswahlmethodik beinhalten, mit deren Hilfe für den jeweiligen vorher definierten Anwendungskontext der richtige Benchmark ausgewählt werden kann. Dem Nutzer der Integrationsplattform werden dabei die unterschiedlichen Benchmarks zur Verwendung angeboten. Dabei soll ein Empfehlungssystem vorhanden sein, mit dem entschieden werden kann, welches der Big Data Werkzeuge für den Anwendungskontext am besten geeignet ist.

Bei einigen Big Data Benchmarks fehlt zum Beispiel eine Preis-Metrik, womit die Einordung in den Kontext Benchmarking und dessen Ziele teilweise ungenau werden. Es ist oft nicht ersichtlich, ob es sich bei einem Big Data Benchmark zum Beispiel um einen Domain-specific Benchmark handelt und ob dann auch die damit einhergehenden Anforderungen korrekt implementiert sind. Insgesamt gibt es bisher keine einfache Auswahlmethodik zur Entscheidung für den richtigen Benchmark. Diese Problematik soll durch die Integrationsplattform behoben werden.

In keinem der Big Data Benchmarks gibt es Metriken für die Einfachheit der Benutzung, die Einfachheit der Programmierung oder die Einfachheit der Operationen. Dies liegt wahrscheinlich daran, dass diese Faktoren alle qualitativ sind und deshalb schwer zu messen sind. Solche Metriken könnten aber einen entscheidenden Einfluss auf die Auswahlentscheidung für ein Big Data Werkzeug haben. Dabei könnten Schätzwerte und eventuell spätere Erfahrungswerte wenigstens einen groben Überblick geben und so entscheidende Faktoren in einer Auswahlmethodik sein. Ein weiteres Ziel der Integrationsplattform könnten also Vorschläge für die Entwicklung von oben genannten Metriken sein.

Die Idee der Integrationsplattform ist zusammengefasst erstens einen aufschlussreichen Überblick über die aktuellen Benchmarks und eine Einordnung in den Kontext Big Data zu geben und zweitens die Integration einer Auswahlmethodik für den am besten geeigneten Benchmark. Im Idealfall sind abschließend, durch die korrekte Auswahl eines Benchmarks für den spezifischen Anwendungsfall anhand der Integrationsplatform, Rückschlüsse auf den Wert einer möglichen Implementierung des Anwendungsfalls möglich.

Literatur

[ABM13] Alexandrov, Alexander ; Brücke, Christoph ; Markl, Volker: Issues in big data testing and benchmarking. In: *Proceedings of the Sixth International Workshop on Testing Database Systems - DBTest '13* (2013), 1. `http://dx.doi.org/10.1145/2479440.2482677`. – DOI 10.1145/2479440.2482677. ISBN 9781450321518

[DG08] Dean, Jeffrey ; Ghemawat, Sanjay: MapReduce: Simplified Data Processing on Large Clusters. In: *Commun. ACM* 51 (2008), Januar,

Nr. 1, 107–113. http://dx.doi.org/10.1145/1327452.1327492. – DOI 10.1145/1327452.1327492. – ISSN 0001–0782

[Fre14] Freiknecht, Jonas: *Big Data in der Praxis : Lösungen mit Hadoop, HBase und Hive ; Daten speichern, aufbereiten, visualisieren.* München : Hanser, 2014. – ISBN 978–3–446–43959–7

[Gra92] Gray, Jim: *Benchmark Handbook: For Database and Transaction Processing Systems.* San Francisco, CA, USA : Morgan Kaufmann Publishers Inc., 1992. – ISBN 1558601597

[Gra93] Gray, Jim (Hrsg.): *The Benchmark Handbook for Database and Transaction Systems (2nd Edition).* Morgan Kaufmann, 1993. – ISBN 1–55860–292–5

[GRH+13] Ghazal, Ahmad ; Rabl, Tilmann ; Hu, Minqing ; Raab, Francois ; Poess, Meikel ; Crolotte, Alain ; Jacobsen, Hans-Arno: BigBench: Towards an Industry Standard Benchmark for Big Data Analytics. In: *Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data.* New York, NY, USA : ACM, 2013 (SIGMOD '13). – ISBN 978–1–4503–2037–5, 1197–1208

[KTS+15] Kos, Anton ; Tomažič, Sašo ; Salom, Jakob ; Trifunovic, Nemanja ; Valero, Mateo ; Milutinovic, Veljko: New benchmarking methodology and programming model for big data processing. In: *International Journal of Distributed Sensor Networks* 2015 (2015). http://dx.doi.org/10.1155/2015/271752. – DOI 10.1155/2015/271752. – ISSN 15501477

[Lan01] Laney, Doug: 3D Data Management: Controlling Data Volume, Velocity, and Variety. In: *Application Delivery Strategies* 949 (2001), Nr. February 2001, S. 4

[LGJ+14] Luo, Chunjie ; Gao, Wanling ; Jia, Zhen ; Han, Rui ; Li, Jingwei ; Lin, Xinlong ; Wang, L: Handbook of BigDataBench (Version 3.1) A Big Data Benchmark Suite. Version: 2014. http://prof.ict.ac.cn/BigDataBench/wp-content/uploads/2014/12/BigDataBench-handbook-6-12-16.pdf. 2014. – Forschungsbericht. – 1–96 S.

[MLG+14] Ming, Zijian ; Luo, Chunjie ; Gao, Wanling ; Han, Rui ; Yang, Qiang ; Wang, Lei ; Zhan, Jianfeng: BDGS: A scalable big data generator suite in big data benchmarking. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 8585 (2014), S. 138–154. http://dx.doi.org/10.1007/978-3-319-10596-3{_}11. – DOI 10.1007/978–3–319–10596–3_11. – ISBN 9783319105956

[PPR+09] Pavlo, Andrew ; Paulson, Erik ; Rasin, Alexander ; Abadi, Daniel J. ; DeWitt, David J. ; Madden, Samuel ; Stonebraker, Michael: A Comparison of Approaches to Large-scale Data Analysis. In: *Proceedings of the 2009 ACM SIGMOD International Conference on Management of Data.* New York, NY, USA : ACM, 2009 (SIGMOD '09). – ISBN 978–1–60558–551–2, 165–178

[PZ14] Philip Chen, C. L. ; Zhang, Chun Y.: Data-intensive applications, challenges, techniques and technologies: A survey on Big Data. In: *Information Sciences* 275 (2014), 314–347. http://dx.doi.org/10.1016/j.ins.2014.01.015. – DOI 10.1016/j.ins.2014.01.015. – ISBN 0020–0255

[SAD+10] Stonebraker, Michael ; Abadi, Daniel ; DeWitt, David J. ; Madden, Sam ; Paulson, Erik ; Pavlo, Andrew ; Rasin, Alexander: MapReduce and Parallel DBMSs: Friends or Foes? In: *Commun. ACM* 53 (2010), Januar, Nr. 1, 64–71. http://dx.doi.org/10.1145/1629175.1629197. – DOI 10.1145/1629175.1629197. – ISSN 0001–0782

[SMD13] Stonebraker, Michael ; Madden, Sam ; Dubey, Pradeep: Intel "Big DataSScience and Technology Center Vision and Execution Plan. In: *SIGMOD Rec.* 42 (2013), Mai, Nr. 1, 44–49. http://dx.doi.org/10.1145/2481528.2481537. – DOI 10.1145/2481528.2481537. – ISSN 0163–5808

[SR13] Stonebraker, Michael ; Robertson, Judy: Big Data Is 'Buzzword du Jourr;' CS Academics 'Have the Best Job'. In: *Association for Computing Machinery. Communications of the ACM* 56 (2013), Nr. 9, 10. http://dx.doi.org/10.1145/2500468.2500471. – DOI 10.1145/2500468.2500471. – ISBN 00010782

[Tra15a] Transaction Processing Performance Council: *TPC Express Benchmark(tm) DS*. http://www.tpc.org/tpc_documents_current_versions/pdf/tpc-ds_v2.1.0.pdf, 11 2015

[Tra15b] Transaction Processing Performance Council: *TPC Express Benchmark(tm) HS*. http://www.tpc.org/tpc_documents_current_versions/pdf/tpcx-hs_v1.3.0.pdf/, 02 2015

[WZL+14] Wang, Lei ; Zhan, Jianfeng ; Luo, Chunjie ; Zhu, Yuqing ; Yang, Qiang ; He, Yongqiang ; Gao, Wanling ; Jia, Zhen ; Shi, Yingjie ; Zhang, Shujie ; Zheng, Chen ; Lu, Gang ; Zhan, K. ; Li, Xiaona ; Qiu, Bizhu: BigDataBench: A big data benchmark suite from internet services. In: *High Performance Computer Architecture (HPCA), 2014 IEEE 20th International Symposium on*, 2014, S. 488–499

[ZdP+13] Zikopoulos, Paul C. ; deRoos, Dirk ; Parasuraman, Krishnan ; Deutsch, Thomas ; Corrigan, David ; Giles, James: *Harness the Power of Big Data*. McGraw-Hill, 2013. – ISBN 978–0–07180818–7

[Zha14] Zhan, Jianfeng: BigDataBench Technical Report / Chinese Academy of Sciences. Beijing, Dezember 2014. – Forschungsbericht. – 21 S.

[Zha15] Zhan, Jianfeng: *BigDataBench: An Open-source Big Data Benchmark Suite*. http://prof.ict.ac.cn/BigDataBench/wp-content/uploads/2014/12/BigDataBench-WBDB2015.pptx. Version: 2015