Evaluierung der Qualität von Open Source Stream Processing Frameworks

Simon Kern

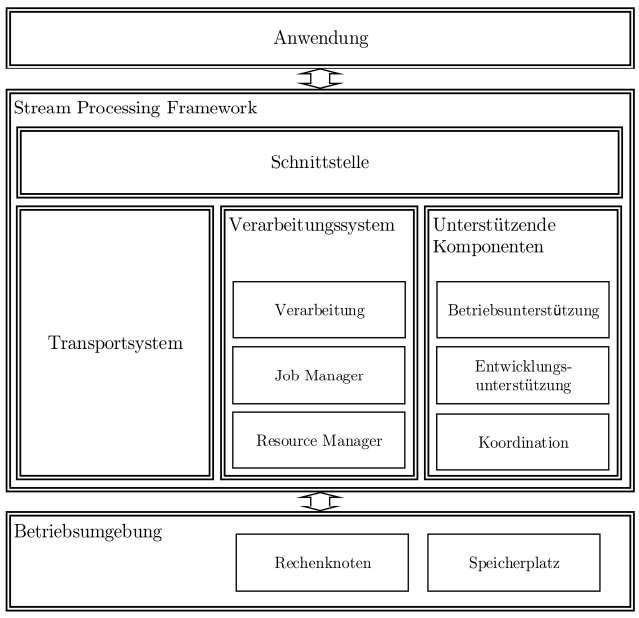

Abbildung 1 • Referenzarchitektur von Stream Processing Frameworks

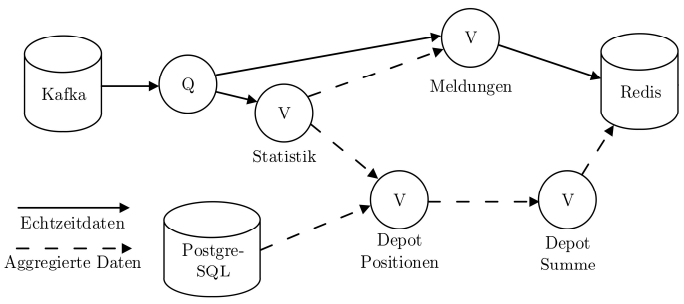

Abbildung 2 • Architektur der Beispielanwendung

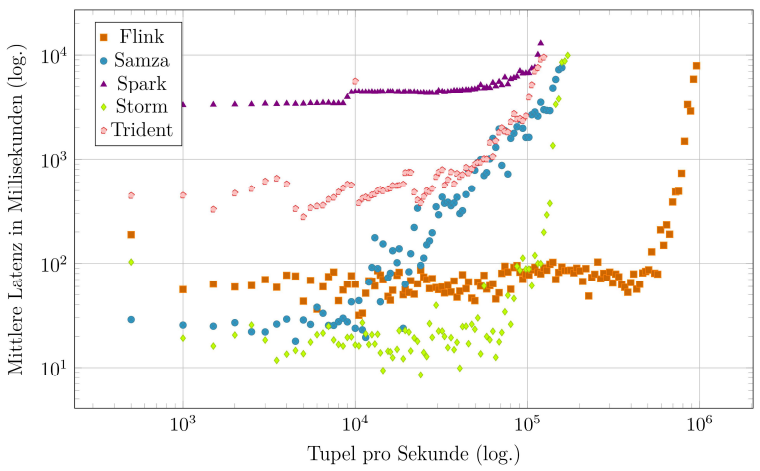

Abbildung 3 • Mittlere Verarbeitungszeit der Tupel in Abhängigkeit der Geschwindigkeit der ankommenden Daten aller evaluierten Frameworks im Vergleich

Motivation

Big Data ist im Bereich der Informationstechnologie aktuell eines der großen Schlagworte. Technologisch bezieht sich der Begriff meist auf die Stapelverarbeitung von gesammelten Daten. Im Gegensatz dazu verarbeiten Stream Processing Frameworks Datenströme direkt.

Während bei den Batch-Systemen Hadoop als etabliert gilt, hat sich im Bereich der Stream Processing Frameworks noch kein Produkt als allgemeine Basis herausgebildet. Es existiert eine Vielzahl von proprietären und quelloffenen Stream Processing Frameworks, die sich im Hinblick auf verschiedene Kriterien unterscheiden. Allen Frameworks ist gemein, dass sie das Entwickeln und den Betrieb von Anwendungen erleichtern, indem sie essentielle Funktionen, wie beispielsweise die Verteilung der Anwendung über mehrere Knoten, übernehmen. Es existieren aber auch Unterschiede zwischen den Frameworks: etwa bei den Programmiermodellen, der Fehlertoleranz oder den Integrationsmöglichkeiten.

Da sich noch kein Framework als Quasi-Standard etabliert hat und einige Unterschiede zwischen den Frameworks bestehen, muss bei Neuentwicklungen von Anwendungen im Bereich Stream Processing geprüft werden, welches Framework für die Entwicklung und den Betrieb am besten geeignet ist. Eventuell lassen sich je nach Anwendungsfall schon anhand der Funktionen Frameworks vom Vergleich ausschließen. Andere Kriterien lassen sich schwer nur anhand der Dokumentation bewerten, etwa die Benutzerfreundlichkeit der Programmierschnittstelle oder des Betriebs. Es existieren zwar verschiedene Vergleiche, allerdings nutzen diese Arbeiten hauptsächlich die Dokumentationen der Frameworks als Informationsquelle.

Ziel dieser Arbeit ist es, aktuelle Stream Processing Frameworks im Hinblick auf deren Qualität zu evaluieren. Hier liegt der Fokus des Vergleichs auf der praktischen Einsatzfähigkeit der Frameworks. Es werden ein Kriterienkatalog, eine Beispielanwendung und geeignete Methoden erarbeitet, anhand derer aktuelle Stream Processing Frameworks verglichen werden. Die Ergebnisse sollen die Entscheidungsfindung bei der Auswahl von Frameworks für Neuentwicklungen unterstützen und als Leitfaden für die Evaluierung von künftigen Frameworks dienen.

Vorgehensweise

Diese Arbeit stellt eine Referenzarchitektur vor, anhand derer die Komponenten eines Stream Processing Frameworks zur Verbesserung der Vergleichbarkeit zugeordnet werden können. Sie ist in Abbildung 1 dargestellt.

Mithilfe des Evaluierungsprozesses nach ISO 25000 und insbesondere des Softwarequalitätsmodells nach ISO 25010 werden Kriterien zur Evaluierung von Stream Processing Frameworks entwickelt. Das Qualitätsmodell liefert einen Überblick, welche Charakteristiken zur Feststellung der Softwarequalität zu berücksichtigen sind.

Bei der Informationsgewinnung kommen verschiedene Methoden zum Einsatz: vorhandene Quellen, wie die Dokumentation der Frameworks, wissenschaftliche Arbeiten und sonstige relevante Literatur. Entscheidende Faktoren, wie die Benutzerfreundlichkeit und die Praxistauglichkeit, lassen sich dadurch nur unzureichend bewerten. Daher werden zur Erhöhung der praktischen Relevanz die Methoden Cognitive Walkthrough und Cognitive Dimensions zur Feststellung der Benutzerfreundlichkeit genutzt.

Der Cognitive Walkthrough ist eine aufgabenorientierte Usability-Inspektionsmethode. Der Evaluator versetzt sich in die Lage des Nutzers und löst mithilfe der ausgewählten Frameworks typische Aufgaben. Als Rahmen der Aufgaben dient die Beobachtung von Wertpapieren, die an der New York Stock Exchange gelistet sind. Die Architektur der Beispielanwendung ist in Abbildung 2 dargestellt.

Cognitive Dimensions sind Kriterien zur Analyse und Bewertung der Benutzerfreundlichkeit einer API. Dazu gehören etwa das Abstraktionslevel, die Konsistenz oder die Lesbarkeit der Schnittstelle.

Beide vorgestellten Methoden sind eher den subjektiven Vorgehensweisen zuzuordnen. Sie erfassen weiche Kriterien, die vom Evaluator interpretiert werden müssen. Objektivere eingesetzte Methoden sind Durchsatz- und Latenz-Messungen und der Test der Fehlertoleranz beim Ausfall eines Hosts. Abbildung 3 zeigt den mittleren Durchsatz aller Frameworks im Vergleich.

Ergebnisse

Anhand einer vorgelagerten Literatur- und Internetrecherche wurden die Frameworks Apache Flink, Apache Samza, Apache Spark und Apache Storm für die Evaluierung ausgewählt. Sie werden anhand der entwickelten Kriterien verglichen. Die Ergebnisse werden für jedes Framework im Folgenden kurz zusammengefasst.

Apache Flink weist eine gute und mächtige API und eine hohe Leistungsfähigkeit auf, allerdings hat Flink noch einen geringen Reifegrad.

Apache Samza hat eine übersichtliche API und eine gute Dokumentation. Die mäßige Leistungsfähigkeit, kleine Community und mangelnde externe Unterstützung schränken die praktische Relevanz allerdings ein.

Apache Spark hat eine umfangreiche API, eine große Community und viel externe Unterstützung, außerdem ist der Reifegrad hoch. Prinzipbedingt ist das Framework aufgrund seiner hohen Latenz für sehr zeitkritische Anwendungen ungeeignet.

Apache Storm hat eine sehr geringe Latenz und einen hohen Durchsatz. Es hat eine große Community und wird stetig erweitert. Die API und die Dokumentation sind nur von mäßiger Qualität, außerdem fehlt die Unterstützung für einige Operationen. Storm Trident hat im Gegensatz zu Flink und Spark eine deutlich schlichtere API.

Der Vergleich hat gezeigt, dass keines der evaluierten Frameworks alle Kriterien vollständig abdeckt. Vielmehr müssen die Ergebnisse dieser Arbeit nach den konkreten Anforderungen einer Neuentwicklung gewichtet werden, um eine Auswahlentscheidung zu treffen.

h_da
HOCHSCHULE DARMSTADT
UNIVERSITY OF APPLIED SCIENCES
fbi
FACHBEREICH INFORMATIK

Kontakt
Simon Kern
Fachbereich Informatik
simon.kern@stud.h-da.de

Projektpartner
Infomotion GmbH, Frankfurt am Main

Referentin
Prof. Dr. Uta Störl
Fachbereich Informatik

Koreferentin
Prof. Dr. Inge Schestag
Fachbereich Informatik