ETH zürich

ETH Library

Data stream processing in complex applications

Doctoral Thesis

Author(s):
Sheykh Esmaili, Kyumars

Publication date:
2011

Permanent link:
https://doi.org/https://doi.org/10.3929/ethz-a-006689762

Rights / license:
In Copyright - Non-Commercial Use Permitted

This page was generated automatically upon download from the ETH Zurich Research Collection.
For more information, please consult the Terms of use.

Diss. ETH No. 19907

Data Stream Processing in Complex Applications

A dissertation submitted to the
ETH ZURICH

for the degree of
Doctor of Sciences

presented by
KYUMARS SHEYKH ESMAILI
Master of Science in Software Engineering, Sharif University of Technology
born 26 January 1981
citizen of Iran

accepted on the recommendation of
Prof. Dr. Donald Kossmann, examiner
Prof. Dr. Peter M. Fischer, co-examiner
Prof. Dr. Renée J. Miller, co-examiner
Prof. Dr. Bernhard Seeger, co-examiner

2011

Abstract

With a massive increase of available data sources, their high data rates and requirements for fast response, we have witnessed the emergence of data stream processing as a new data management paradigm. Many Stream Processing Engines (SPEs) have been built in recent years and several business applications have started to successfully use data stream management systems.

However, as industry gears toward using data stream processing in increasingly complex applications, existing approaches and systems are no longer sufficient. In this thesis, we identify, generalize and solve the challenges arising from such complex applications, based on a real-world use case, compliance monitoring in Service Oriented Architectures (SOA). In particular, we make the following contributions toward making data stream processing a viable solution for complex streaming applications:

1. *Establishing Schema for Data Streams and Exploring its Applications.* Static metadata (aka Schema) plays an important role in design and validation of data. We identify the most relevant static metadata aspects for data streams, and formally specify them in a model named Stream Schema. We also present a mechanism for validation of data streams against Stream Schema and analyze its runtime and space complexity. Experimental evaluations confirm the usefulness of semantic optimization techniques based on Stream Schema. Additionally, we show that applicability of Stream Schema extends beyond optimization gains. It broadens the static analysis of stream queries and changes the way that stream applications are modeled.

2. *Fine-grained Provenance Management on Data Streams.* Precise evidence of origin and processing of data (aka provenance) is a crucial part of many complex, data-centric applications. We investigate the requirements of fine-grained provenance on data streams using real-life use cases and determine the challenges, in particular caused by the volatility of data. To solve this problem, we analyze existing provenance computation methods, establish the formal foundations using an algebra, and show the details of the implementation. Numerous optimizations to reduce

the overhead of provenance computation and to enable on-demand computation of provenance make it possible to use fine-grained provenance in environments with high performance requirements. We show this by an extensive set of experiments.

3. *A Framework to Model Continuous Query Lifecycle and Query Modification Model.* Precise descriptions of lifecycle operations of a data stream query are necessary to provide exact semantics not only for "regular operations", but also for "boundary conditions", in particular when modifying a query to fit the new requirements. We establish a punctuation-based framework and a methodology that can formally express arbitrary lifecycle operations on continuous queries. It is based on input/output mapping functions and a limited set of punctuations such as Start and Stop control elements. Using this framework and methodology, we derive all possible variations of query modification, each offering a distinct combination of correctness guarantees. We also show how this framework and model can be integrated into state-of-the-art SPEs with fairly minor effort.

In order to avoid SPE-specificity, all solutions proposed in this thesis have been built upon abstractions and formalizations. Furthermore, all of them have been implemented and experimentally validated, in some cases, using more than one stream processing platforms

Zusammenfassung

Die massiv steigende Anzahl an Datenquellen, deren hohe Datenraten, sowie die Anforderung an schnelle Reaktionszeiten haben zur Etablierung der Datenstromverarbeitung (*Stream Processing*) als eigenständiges Datenverwaltungsparadigma geführt. Im Verlauf der letzten Jahre wurden zahlreiche Systeme zur Datenstromverarbeitung (*Stream Processing Engine*, SPE) entwickelt, welche bereits erfolgreich in vielen Bereichen der Industrie eingesetzt werden.

Im Zuge immer komplexerer Anwendungen sind existierende Ansätze und Systeme jedoch nicht länger adäquat. Diese Doktorarbeit identifiziert, generalisiert und löst die neu entstandenen Herausforderungen anhand eines Beispiels aus der Praxis, der Überwachung von Richtlinien (*Compliance Monitoring*) in service-orientierten Architekturen (SOA). Dabei werden insbesondere die folgenden Beiträge zur besseren Anwendbarkeit von SPEs in komplexen Anwendungen erbracht:

1. *Schema für Datenströme und seine Anwendungen.* Statische Metadaten (Schema) spielen eine zentrale Rolle in der Modellierung und Validerung von Daten. Die vorliegende Arbeit identifiziert die relevanten Aspekte statischer Metadaten bezüglich Datenströmen, und formalisiert diese in Form eines Modell namens *Stream Schema*. Für dieses Modell wird ein Validierungsalgorithmus beschrieben und bezüglich Speicher- und Zeitkomplexität untersucht, welcher die Korrektheit eines Datenstroms hinsichtlich eines benutzerdefinierten Schemas überprüft. Eine analytische und experimentelle Evaluierung zeigt den Nutzen solcher Schema bei der semantischen Optimierung von Datenstromanfragen. Weitere Anwendungsgebiete umfassen die statische Korrektheitsanalyse von Datenstromanfragen sowie neue Entwurfsansätze für Datenstromanwendungen.

2. *Feingranulare Provenance(Abstammung/Herkunft) auf Datenströmen.* Ein genauer Nachweis der Herkunft und Verarbeitungskette von Daten (*Provenance*) ist wichtiger Bestandteil vieler komplexer Anwendungen. Die vorliegende Arbeit untersucht die Anforderungen feingranularer Provenance im Kontext von Datenstromanwendun-

gen anhand konkreter Szenarien, und identifiziert die besonderen Herausforderungen dabei, insbesondere die Flüchtigkeit von Daten. Es werden existierende Ansätze zur Berechnung von Provenance betrachtet, die formalen Grundlagen von Stream Provenance auf Basis einer speziellen Algebra gelegt, sowie eine Implementierung von Stream Provenance inklusive zahlreicher Optimierungen beschrieben. Wie anhand zahlreicher Experimente nachgewiesen wird, ermöglichen diese Optimierungen den Einsatz feingranularer Provenance in Anwendungen mit hohen Leistungsanforderungen.

3. *Rahmenmodell zur Beschreibung von Lebenszyklus und Modifikation von Datenstromanfragen.* Präzise Modelle von Lebenszyklusoperationen einer Datenstromanfrage sind notwendig um exakte Semantik nicht nur im "regulären Betrieb" sondern auch in den "Übergangsbereichen" solcher Anfragen zu erhalten, insbesondere bei der Modifikation einer laufenden Anfrage. Diese Arbeit präsentiert ein Rahmenmodell auf Basis von Markierungen (*Punctuations*), mithilfe dessen beliebige Lebenszyklusoperationen formal ausgedrückt werden können. Dieses Rahmenmodell beruht auf Abbildungsfunktionen zwischen Eingaben und Ausgaben sowie einigen wenigen grundlegenden Markierungen wie Start und Stop. Das Rahmenmodell wird in der Folge für verschiedene Lebenszyklusoperationen angewendet, insbesondere für die Modifikation von laufenden Anfragen. Dabei werden alle Varianten der Anfragenmodifikation vollständig abgeleitet, welche jeweils bestimmte Korrektheitskriterien hinsichtlich des Eingangs- und Ausgangsstroms garantieren beziehungsweise nicht garantieren. Eine vollständige Implementierung zeigt, wie dieses Rahmenmodell mit geringem Aufwand in existierende SPEs integriert werden kann.

Um die beschriebenen Verfahren weitestgehend anwendbar zu machen, wurde für die gesamte Arbeit eine einheitliche Formalisierung entwickelt. Alle Verfahren bauen auf dieser Formalisierung auf, und wurden für diverse existierende SPEs – fallweise für mehrere SPEs – umgesetzt und evaluiert.