Effiziente Speicherung von Zeitreihen mit Betriebsdaten aus Software-Systemen zur Analyse von Laufzeitanomalien

Florian Lautenschlager

Effiziente Speicherung von Zeitreihen mit Betriebsdaten aus Software-Systemen zur Analyse von Laufzeitanomalien

Der Technischen Fakultät

der Friedrich-Alexander-Universität Erlangen-Nürnberg

zur

Erlangung des Doktorgrades Dr.-Ing.

vorgelegt von

Florian Lautenschlager

aus München

Als Dissertation genehmigt
von der Technischen Fakultät
der Friedrich-Alexander-Universität Erlangen-Nürnberg

Tag der mündlichen Prüfung:	**07.05.2019**
Vorsitzender des Promotionsorgans:	**Prof. Dr.-Ing. Reinhard Lerch**
Gutachter:	**Prof. Dr. rer. nat. Michael Philippsen** **Prof. Dr.-Ing. Klaus Meyer-Wegener**

Berichte aus der Informatik

Florian Lautenschlager

Effiziente Speicherung von Zeitreihen mit Betriebsdaten aus Software-Systemen zur Analyse von Laufzeitanomalien

D 29 (Diss. Universität Erlangen-Nürnberg)

Shaker Verlag
Düren 2019

Bibliografische Information der Deutschen Nationalbibliothek
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über http://dnb.d-nb.de abrufbar.

Zugl.: Erlangen-Nürnberg, Univ., Diss., 2019

Printed in Germany.

ISBN 978-3-8440-6785-9
ISSN 0945-0807

Shaker Verlag GmbH • Am Langen Graben 15a • 52353 Düren
Telefon: 02421 / 99 0 11 - 0 • Telefax: 02421 / 99 0 11 - 9
Internet: www.shaker.de • E-Mail: info@shaker.de

Eidesstattliche Erklärung

Ich versichere, dass ich die Arbeit ohne fremde Hilfe und ohne Benutzung anderer als der angegebenen Quellen angefertigt habe und dass die Arbeit in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen hat und von dieser als Ganzes oder als Teil einer Prüfungsleistung angenommen wurde. Alle Ausführungen, die wörtlich oder sinngemäß übernommen wurden, sind als solche gekennzeichnet. Die Promotionsprüfung in dem angestrebten Doktorgrad wurde nicht anderweitig endgültig nicht bestanden.

Der Friedrich-Alexander-Universität Erlangen-Nürnberg, vertreten durch den Lehrstuhl Informatik 2 (Programmiersysteme), wird für Zwecke der Forschung und Lehre ein einfaches, kostenloses, zeitlich und örtlich unbeschränktes Nutzungsrecht an den Arbeitsergebnissen der Dissertation einschließlich etwaiger Schutzrechte und Urheberrechte eingeräumt, soweit diese nicht bereits abgetreten wurden. Die Dissertation darf elektronisch gespeichert und zum Zwecke der Zitatkontrolle genutzt werden.

Mir ist bekannt, dass der Doktorgrad erst nach Aushändigung der Urkunde geführt werden darf und die erworbenen Rechte erlöschen, wenn die Pflichtexemplare nicht rechtzeitig eingereicht werden.

Florian Lautenschlager

Abstract

When business-relevant software systems work inefficiently or fail, they can lead to economic damage and a loss of reputation. These software systems must therefore meet high stability requirements. If they do not meet these requirements, they can show conspicuous runtime behavior, such as abnormal resource usage, sporadic errors due to synchronization conflicts, or suspicious security-related actions. The analysis of the runtime behavior is a mean of understanding the behavior of the software system. It requires a structured approach as well as a tool chain.

The tool chain consists of tools for collecting, storing and analyzing operational data. Monitoring tools collect any kind of operational data, such as metrics, events or traces. Analyses use statistics, machine learning, etc. to detect abnormalities in the operational data. But there is a gap between the collection and the analysis, since standard time series databases, when used to store operational data, do not optimize for analyzing the runtime behavior. The reason for this is that they use a restricted data model with primitive types and functions for these types, which means that an unforeseen combination of any operational data in analyses is not possible. They also have insufficient means to process asynchronous time series that arise due to the problematic nature of the data collection, e.g., noise or different collection strategies. Finally, they achieve a high storage efficiency, but could be more efficient if they exploit domain-specific characteristics. In summary, these limitations of standard time series databases lead to restrictions when they are used to analyze the runtime behavior of a software-system.

The aim of this thesis is to avoid the limitations of standard time series databases and thus to enable an efficient analysis of the runtime behavior. The thesis provides two contributions: an approach for synchronizing asynchronous time series and the domain-specific time series database *Chronix*. Asynchronous time-discrete time series are converted into time-continuous time series and then are synchronized so that they can be processed as synchronous time-discrete time series. Chronix is optimized for analyzing the runtime behavior. It has a generic multidimensional data model, a plugin mechanism for time series types and functions, respects the approach for synchronizing time series and has a functional-lossless long-term storage. In the quantitative evaluation, Chronix outperforms established time series databases: Not only does Chronix have a lower memory footprint, but it also saves 20 %–68 % of the hard disk space, is 80 %–92 % faster to access data, and saves 73 %–97 % of the runtimes on database analysis functions. The qualitative evaluation also shows the advantages of Chronix based on three case studies from the industry.

Zusammenfassung

Arbeiten geschäftsrelevante Software-Systeme ineffizient oder fallen aus, können sie zu wirtschaftlichem Schaden und einem Imageverlust führen. Diese Software-Systeme müssen daher hohen Stabilitätsansprüchen genügen. Erreichen sie die Ansprüche nicht, können sie auffälliges Laufzeitverhalten zeigen, wie anormale Ressourcennutzung, sporadische Fehler durch Synchronisationskonflikte oder auffällige sicherheitsrelevante Aktionen. Die Analyse des Laufzeitverhaltens ist ein Mittel, um das Verhalten des Software-Systems zu verstehen. Sie erfordert ein strukturiertes Vorgehen und eine Werkzeugkette.

Die Werkzeugkette besteht aus Werkzeugen zum Aufzeichnen, Speichern und Analysieren: Monitoring-Werkzeuge sammeln beliebige Betriebsdaten, wie Metriken, Ereignisse oder Spuren. In Analysen werden Statistik, Machine Learning etc. zum Erkennen von Auffälligkeiten in den Betriebsdaten genutzt. Dazwischen existiert eine Lücke, da Standard-Zeitreihendatenbanken, wenn sie zum Speichern der Betriebsdaten verwendet werden, diese nicht für die Analyse des Laufzeitverhaltens optimiert speichern. Der Grund ist, dass sie ein eingeschränktes Datenmodell mit primitiven Datentypen und Funktionen nutzen, das eine Kombination beliebiger Betriebsdaten in den Analysen nicht erlaubt. Zudem haben sie unzureichende Mittel, um asynchrone Zeitreihen zu verarbeiten, die wegen der Erhebungsproblematik entstehen (Rauschen, unterschiedliche Erhebungsstrategien etc.). Sie erreichen schließlich eine hohe Effizienz beim Speichern, die aber durch das Ausnutzen domänenspezifischer Eigenschaften noch höher sein könnte. Deshalb können Standard-Zeitreihendatenbanken nur begrenzt für die Analyse des Laufzeitverhaltens von Software-Systemen eingesetzt werden.

Die vorliegende Arbeit hat das Ziel, die Einschränkungen der Standard-Zeitreihendatenbanken zu vermeiden und so eine effiziente Analyse des Laufzeitverhaltens zu ermöglichen. Dazu liefert die Arbeit zwei Beiträge: einen Ansatz zum Synchronisieren asynchroner Zeitreihen und die domänenspezifische Zeitreihendatenbank *Chronix*. Asynchrone zeitdiskrete Zeitreihen werden in zeitkontinuierliche Zeitreihen überführt, danach synchronisiert, sodass sie anschließend als synchrone zeitdiskrete Zeitreihen verarbeitet werden können. Chronix ist für die Analyse des Laufzeitverhaltens optimiert: Es hat ein generisches multidimensionales Datenmodell, einen Erweiterungsmechanismus für Zeitreihentypen und Zeitreihenfunktionen, berücksichtigt den Ansatz zum Synchronisieren von Zeitreihen und hat einen effizienten funktional-verlustfreien Langzeitspeicher. In der quantitativen Evaluation übertrifft Chronix etablierte Zeitreihendatenbanken: Chronix hat nicht nur einen geringeren Arbeitsspeicherbedarf, spart 20 %–68 % des Festplattenspeicherbedarfs, ist 80 %–92 % schneller beim Datenzugriff und spart 73 %–97 % der Laufzeiten bei datenbankseitigen Analysefunktionen. Die qualitative Evaluation zeigt die Vorteile von Chronix anhand von drei Fallstudien aus der Industrie.

Danksagung

Danken möchte ich an dieser Stelle zuerst meinem Doktorvater, Herrn Prof. Dr. Michael Philippsen, der mich auf meinem Weg zur Promotion stets durch konstruktive und zielführende Anregungen fachlich und persönlich begleitet und unterstützt hat.

Mein herzlicher Dank gilt zudem Herrn Prof. Dr. Klaus Meyer-Wegener, der ohne Zögern und mit großem Interesse bereit war, die Rolle des Zweitgutachters zu übernehmen.

Außerdem möchte ich mich bei der QAware bedanken, insbesondere bei Dr. Josef Adersberger und Johannes Weigend sowie der ganzen Geschäftsführung, die mein Promotionsvorhaben stets wohlwollend begleiteten.

Ein großer Dank geht auch an Diejenigen, die mir wichtige Impulse für meine wissenschaftliche Arbeit gaben: Tilman Seifert, Prof. Dr. Johannes Siedersleben, Felix Lautenschlager, Prof. Florian Künzner und Andreas Kumlehn.

Bei Julia Meier und Claudia Diepold bedanke ich mich sehr für die stilistische und rechtschriftliche Überarbeitung dieser Arbeit.

Anna, es ist vollbracht!

Veröffentlichungen

Teile der vorliegenden Arbeit sind bereits in den nachfolgenden Artikeln veröffentlicht. Die Inhalte fließen insbesondere in die Kapitel 3, 5 und 6 ein.

- Lautenschlager, F.; Philippsen, M.; Kumlehn, A.; Adersberger, J.: Chronix: Long Term Storage and Retrieval Technology for Anomaly Detection in Operational Data. In: Proceedings of the USENIX Conference on File and Storage Technologies, Santa Clara, CA., USA, 2017, S. 229–242.
- Lautenschlager, F.; Kumlehn, A.; Adersberger, J.; Philippsen, M.: Fast and efficient operational time series storage: The missing link in dynamic software analysis. In: Proceedings of the Symposium on Software Performance, München, Deutschland, 2015, Softwaretechnik-Trends, Bd. 35, Nr. 3.
- Lautenschlager, F.; Kumlehn, A.; Adersberger, J.; Philippsen, M.: Rahmenwerk zur Ausreißererkennung in Zeitreihen von Software-Laufzeitdaten. In: Tagungsband Software Engineering & Management, Dresden, Deutschland 2015, S. 177–182.

Die Autoren Florian Lautenschlager, Michael Philippsen, Andreas Kumlehn und Josef Adersberger haben an den drei Artikeln folgende Anteile:

Forschungsfrage: Den Rahmen der Artikel bildet das Forschungsprojekt *Design for Diagnosability* [66]. Es ist ein Kooperationsprojekt zwischen der Friedrich-Alexander-Universität Erlangen-Nürnberg und der QAware GmbH. Florian Lautenschlager, Michael Philippsen und Josef Adersberger haben den Förderantrag formuliert. Florian Lautenschlager hat die Forschungsfragen der Artikel herausgearbeitet.

Analyse, Implementierung, Evaluation, Artikel: Florian Lautenschlager hat die Analyse der Forschungsfrage, die Implementierung des Ansatzes sowie die Evaluation durchgeführt und die initialen Fassungen der Artikel formuliert.

Darstellung, Klarheit, Ergänzungen: Michael Philippsen redigierte als kritischer Gutachter die initialen Fassungen. Er lieferte Beiträge zur Darstellung, Klarheit sowie zur Nachvollziehbarkeit der Inhalte und regte ergänzende Auseinandersetzungen an. Andreas Kumlehn und Josef Adersberger trugen als Teil des Projektteams zur inhaltlichen Diskussion bei und initiierten sprachliche und stilistische Korrekturen.

Inhaltsverzeichnis