



**FernUniversität Hagen**

– Faculty of Mathematics and Computer  
Science –

**Dynamically allocation of Datasets to  
Datastores in a Polystore Environment**

Presented by

**Daniel Langhann**

Matriculation number: 3788687

Supervision : Prof. Dr. Ute Störl



## DECLARATION

---

I declare that I have written the seminar paper independently and without any unauthorized assistance from third parties.

In doing so, I have only used the sources and resources specified and have identified the passages taken from these sources, either verbatim or in meaning, as such.

The assurance of independent work also applies to any drawings, sketches, or graphic representations contained therein.

The paper has not been submitted in the same or similar form to the same or any other examination authority nor has it been published.

By submitting the electronic version of the final paper, I acknowledge that it may be checked for plagiarism using a plagiarism detection service and will be stored exclusively for examination purposes.

*Hagen, 01. Feb 2025*

---

Daniel Langhann

## ABSTRACT

---

In der vorliegenden Masterarbeit wird das Thema Dynamische Allokation von Datasets zu Datastores behandelt. Die Motivation für die Arbeit ist es, einen Vorschlag zu unterbreiten, wie in einem Umfeld aus heterogenen Datasets, für die mehr als ein Datenbank-System ideal ist, eine dynamische Zuweisung und Neuzuweisung von Datasets zu Datastores zu realisieren. In der Vergangenheit wurde bereits viel an dem Themenbereich Polystores geforscht. Polystores beinhalten mindestens zwei verschiedenartige Datastores also letztendlich Datenbank-Systeme um heterogene Datasets, zum Beispiel transaktionsbasierte Daten auf der einen Seite und aggregierte Daten für die Analyse von zum Beispiel sogenannten Key Performance Indikatoren (KPI) auf der anderen Seite optimal verarbeiten zu können. Eine bislang ungelöste Herausforderung in einem polystoren Umfeld ist, die Reaktion auf sich verändernde Workloads in der Applikation bzw. im Gesamtsystem. Wie reagiert man bezogen auf das oben genannte Beispiel, wenn in dem transaktionsbasierten Bereich des Gesamtsystems vermehrt analytische Abfragen entstehen, also GET Requests mit langen und sehr langen Laufzeiten. Dann wäre es wünschenswert, dass diese Abfragen zukünftig über den Datastore, der sich auf analytische Abfragen fokussiert, umgeleitet werden.

Genau hier setzt die vorliegende Arbeit an. Es soll ein Vorschlag unterbreitet werden, wie man in einem polystoren Systemumfeld, dynamisch auf sich verändernde Workloads reagieren kann. Der erste Schwerpunkt beschäftigt sich damit, bezogen auf gegebene Anforderungen an ein Gesamtsystem eine initiale Zuweisung von Datasets zu Datastores vorzuschlagen. Es wird ein Prototyp entwickelt, der algorithmisch eine entsprechende Zuweisung ermittelt. Der zweite Schwerpunkt fokussiert sich auf die permanente und dynamische Analyse von Workloads um bei sich verändernden Parametern bezogen auf den initialen Zustand mit einer Re-Allokation der Datasets zu reagieren bzw. einen angepassten Vorschlag zu unterbreiten. Für diesen zweiten Schwerpunkt wird ebenfalls ein Prototyp entwickelt, der als Input die Analysedaten des Workloads erhält, und beim Erreichen bestimmter Schwellwerte einen Vorschlag für eine Neuzuweisung unterbreitet. Für die konkrete Ausführung der angepassten Zuweisungen wird ein theoretischer Vorschlag gemacht und diskutiert.

Die Arbeit schließt wiederum mit einer Zusammenfassung der Ergebnisse bezogen auf die Ausführung und Testung der Prototypen. Darüber hinaus sollen Vorschläge unterbereitet werden, an welchen Themen noch gearbeitet werden sollte, um tatsächlich ein komplett autonom arbeitendes Gesamtsystem auf der Basis von Polystores zu realisieren.

## CONTENTS

---

1	Einleitung	1
1.1	Motivation . . . . .	1
1.2	Ziel der Arbeit . . . . .	1
1.3	Gliederung . . . . .	1
2	Terminologische Grundlagen	2
3	Initial Zuordnung von Datasets zu Datastores	3
4	Dynamische Allokation und Re-Allokation von Datasets zu Datastores	4
5	Datenmodell und Datenbankseitige Umsetzung von Datastore Re-Allokationen	5
6	Schlussbetrachtung	6
	Bibliography	7

## ABKÜRZUNGSVERZEICHNIS

---

API Application Programming Interface

UML Unified Modeling Language

## EINLEITUNG

---

Lorem ipsum at nusquam appellantur his, labitur bonorum pri no [Due13]. His no decore nemore graecis. In eos meis nominavi, liber soluta vim cu. Sea commune suavitate interpretaris eu, vix eu libris efficiantur.

### 1.1 MOTIVATION

Illo principalmente su nos. Non message *occidental* angloromanic da. Debitas effortio simplicate sia se, auxiliar summarios da que, se avantiate publicationes via. Pan in terra summarios, capital interlingua se que. Al via multo esser specimen, campo responder que da. Le usate medical addresses pro, europa origine sanctificate nos se.

Cras , leo ac adipiscing adipiscing, erat justo vulputate arcu, non sollicitudin ipsum dolor eget lectus. Nulla sed mi non ipsum varius consequat sit amet nec ipsum. Donec ac elit id nibh pretium pulvinar non ut ipsum. Integer congue iaculis augue ac porttitor. Suspendisse sed enim ac eros hendrerit adipiscing. Integer elit libero, lacinia vitae pharetra a, ullamcorper vitae metus. In tempor, est id imperdiet pulvinar, tellus nibh lacinia diam, a eleifend dui lectus non turpis.

*Note: The content of this chapter is just some dummy text. It is not a real language.*

### 1.2 ZIEL DER ARBEIT

Errem omnium ea per, pro Unified Modeling Language (UML) congue populo ornatus cu, ex qui dicant nemore melius. No pri diam iriure euismod. Graecis eleifend appellantur quo id. Id corpora inimicus nam, facer nonummy ne pro, kasd repudiandae ei mei. Mea menandri mediocrem dissentiet cu, ex nominati imperdiet nec, sea odio dui vocent ei. Tempor everti appareat cu ius, ridens audiam an qui, aliquid admodum conceptam ne qui. Vis ea melius nostrum, mel alienum ac elit id nibh pretium pulvina euripidis eu.

### 1.3 GLIEDERUNG

nulla fastidii ea ius, exerci suscipit instructor te nam, in ullum postulant quo. Congue quaestio philosophia his at, sea odio autem vulputate ex. Cu usu mucius iisque voluptua. Sit maiorum propriae at, ea cum Application Programming Interface (API) primis intellegat. Hinc cotidieque reprehendunt eu nec. Autem timeam deleniti usu id, in nec nibh altera.





## INITIAL ZUORDNUNG VON DATASETS ZU DATASTORES

---

## DYNAMISCHE ALLOKATION UND RE-ALLOKATION VON DATASETS ZU DATASTORES

---

DATENMODELL UND DATENBANKSEITIGE  
UMSETZUNG VON DATASTORE RE-ALLOKATIONEN

---

## SCHLUSSBETRACHTUNG

---

## BIBLIOGRAPHY

---

- [Due13] Gunter Dueck. *Dueck's Trilogie 2.1: Omnisophie – Supramanie – Topothese*. Berlin, Germany: Springer, 2013.