



SCHOOL OF COMPUTATION,
INFORMATION AND TECHNOLOGY —
INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Bachelor's Thesis in Informatics

Implementing an Efficient Shuffle Operator for Streaming Database Systems

Ladner, Jonas





SCHOOL OF COMPUTATION,
INFORMATION AND TECHNOLOGY —
INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Bachelor's Thesis in Informatics

**Implementing an Efficient Shuffle Operator
for Streaming Database Systems**

Titel der Abschlussarbeit

Author:	Ladner, Jonas
Examiner:	Neumann, Thomas
Supervisor:	Rieger, Maximilian
Submission Date:	17.02.2025



I confirm that this bachelor's thesis is my own work and I have documented all sources and material used.

Munich, 17.02.2025

Ladner, Jonas

Acknowledgments

Abstract

Contents

Acknowledgments	iv
Abstract	v
1 Introduction	1
1.1 Motivation	1
1.2 Streaming Database Systems	1
1.3 Shuffle Operator	1
1.4 Problem Statement	1
2 Related work	2
2.1 Radix Partitioning	2
2.2 Software Managed Buffers	2
3 Implementations	3
3.1 Time Complexity Analysis	3
3.2 Radix-related Partitioning	3
3.2.1 Radix Partitioning	3
3.2.2 Hybrid Partitioning	3
3.3 Local Pages and Merge-based Partitioning	3
3.4 On-Demand Partitioning	3
3.5 Software Managed Buffers	3
3.5.1 Lock-based Software Managed Buffers	3
3.5.2 Lock-free Software Managed Buffers	3
3.6 Collaborative Morsel Processing	3
3.6.1 Collaborative Morsel Processing with Software Managed Buffers	3
3.6.2 Collaborative Morsel Processing with Processing Units	3
4 Evaluation	4
4.1 Experimental Setup	4
4.1.1 Hardware	4
4.2 Tuple Generation	4

Contents

4.3	Shuffle Benchmark	4
4.3.1	Memory Consumption	4
4.3.2	Performance	4
5	Conclusion	5
	Abbreviations	6
	List of Figures	7
	List of Tables	8

1 Introduction

1.1 Motivation

1.2 Streaming Database Systems

1.3 Shuffle Operator

1.4 Problem Statement

2 Related work

2.1 Radix Partitioning

2.2 Software Managed Buffers

3 Implementations

3.1 Time Complexity Analysis

3.2 Radix-related Partitioning

3.2.1 Radix Partitioning

3.2.2 Hybrid Partitioning

3.3 Local Pages and Merge-based Partitioning

3.4 On-Demand Partitioning

3.5 Software Managed Buffers

3.5.1 Lock-based Software Managed Buffers

3.5.2 Lock-free Software Managed Buffers

3.6 Collaborative Morsel Processing

3.6.1 Collaborative Morsel Processing with Software Managed Buffers

3.6.2 Collaborative Morsel Processing with Processing Units

4 Evaluation

4.1 Experimental Setup

4.1.1 Hardware

4.2 Tuple Generation

4.3 Shuffle Benchmark

4.3.1 Memory Consumption

4.3.2 Performance

5 Conclusion

Abbreviations

List of Figures

List of Tables