

# Effiziente String-Verarbeitung in Datenbankanfragen auf hochgradig paralleler Hardware

Masterarbeit

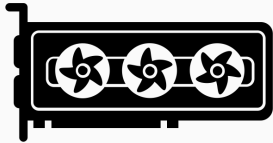
---

Florian Lüdiger

5. Juli 2019

Fakultät Informatik - Lehrstuhl 6 - TU Dortmund

**Ein einfacher String-Vergleich auf der GPU**



## NVIDIA GTX 950

768 CUDA-Cores,  
2GB Speicher,  
6 Maxwell Streaming  
Multiprocessors,  
128 CUDA-Cores pro SMM

## Warp Scheduling

32 Threads = 1 Warp

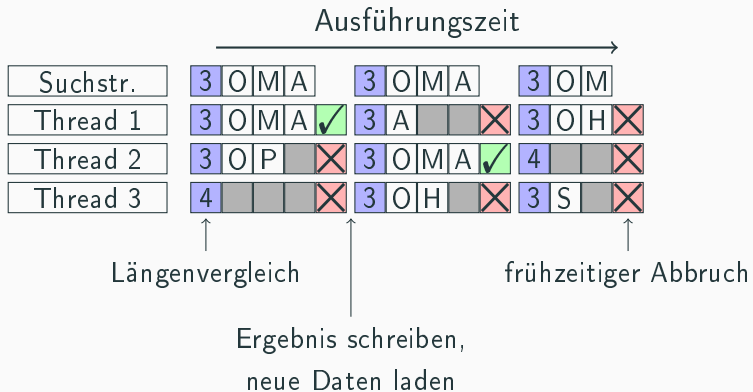
n Warps = 1 Block

m Blocks = Grid

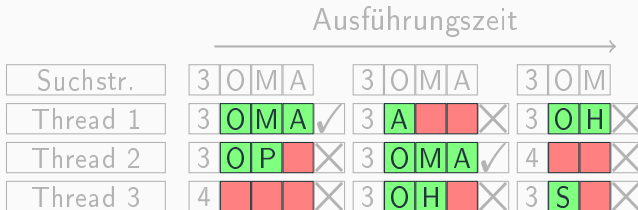
**Single Instruction,  
Multiple Threads**

# String-Vergleich

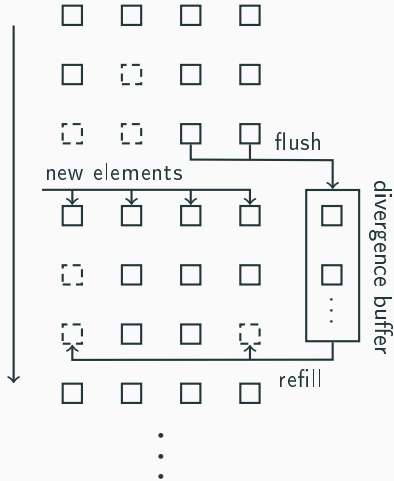
Datensatz: OMA, OPA, OTTO | AHA, OMA, OHM |  
OHA, DBMS, SQL



# Auslastung der GPU

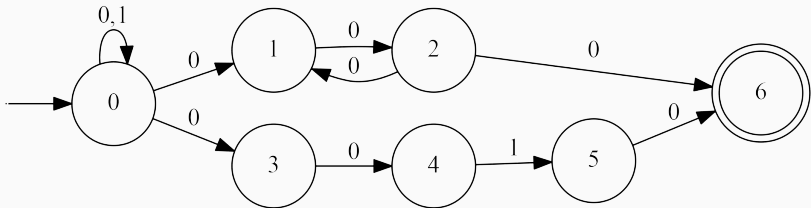


# Verbesserung durch das Lane Refill



# Paralleler Musterabgleich auf der GPU

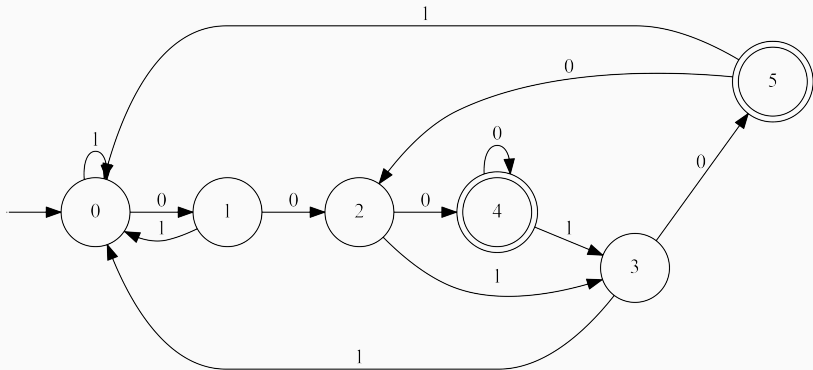
# NFA zum regulären Ausdruck $(0|1)^*((00)^+|001)0$



	0	1	2	3	4	5	6
0	0,1,3	2	1,6	4	-	6	-
1	0	-	-	-	5	-	-



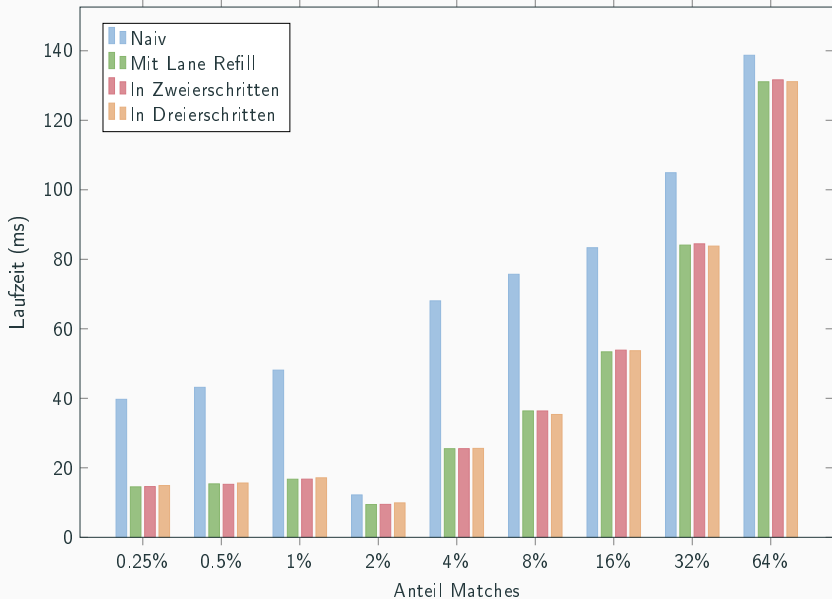
# DFA zum regulären Ausdruck $(0|1)^*((00)^+|001)0$



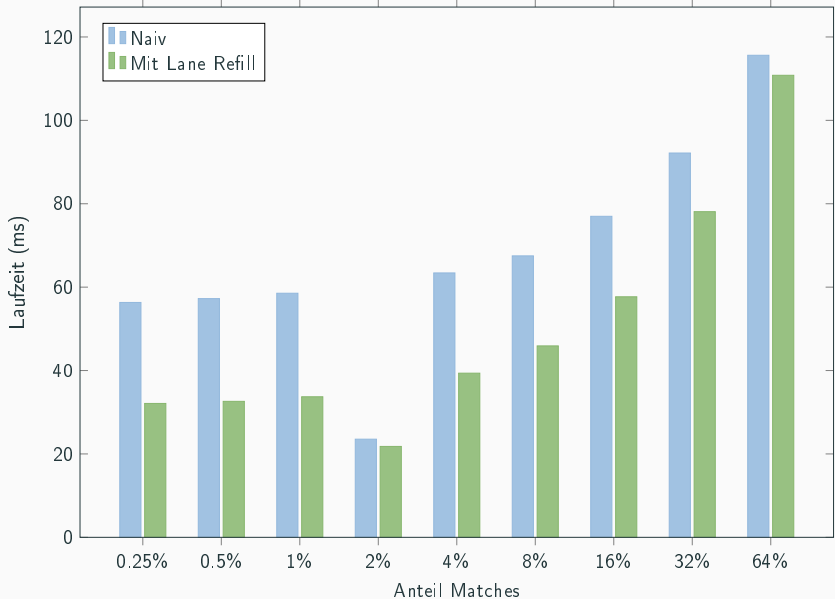
	0	1	2	3	4	5
0	1	2	4	5	3	2
1	0	0	3	0	4	0

## Evaluation des einfachen String-Vergleichs

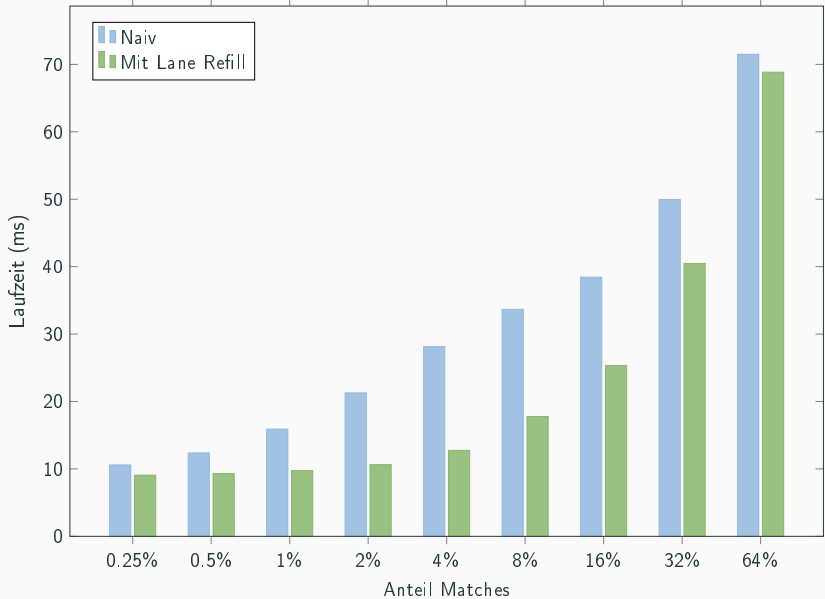
# Gleichheitstest mit verschiedener Verteilung für Type-Daten



# Präfixtest mit verschiedener Verteilung für Type-Daten

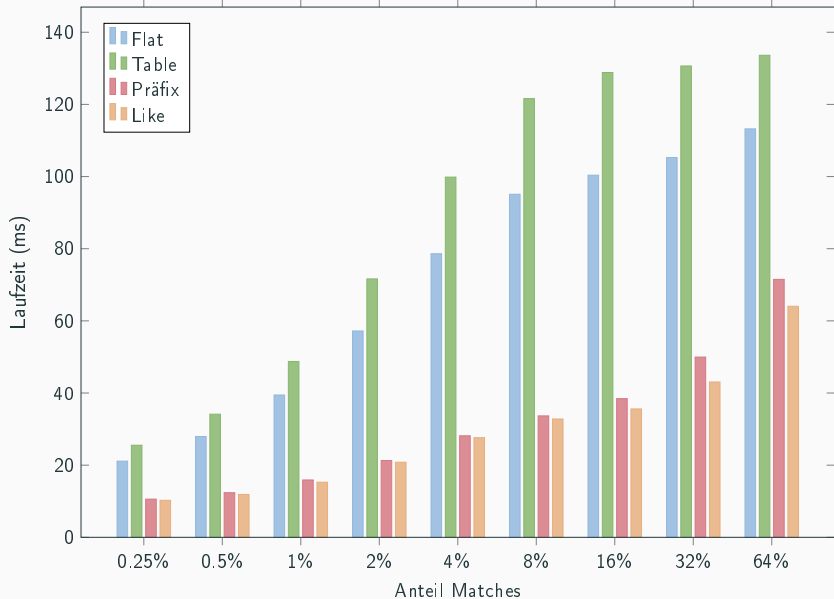


# Präfixtest mit verschiedener Verteilung für DBLP-Daten

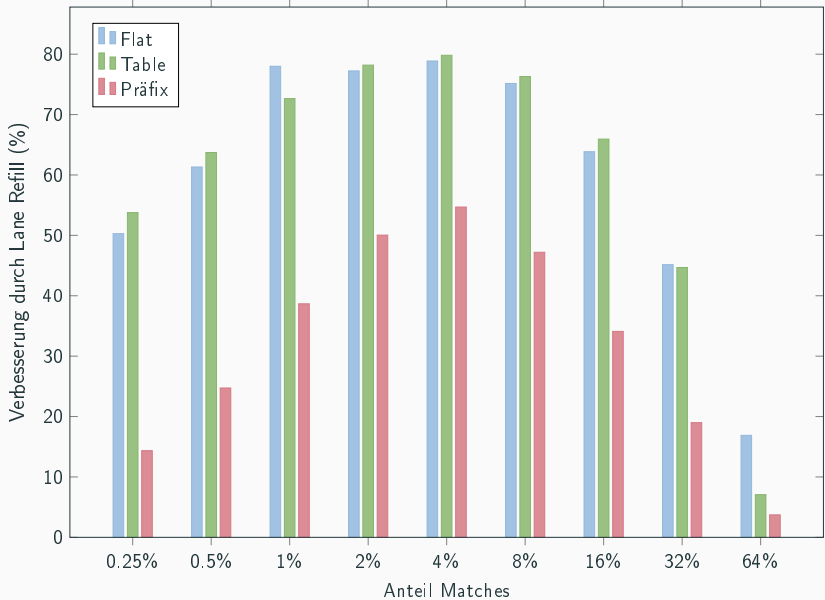


# Evaluation des parallelen Musterabgleichs

# Präfixtest mit Basisalgorithmen für DBLP-Daten

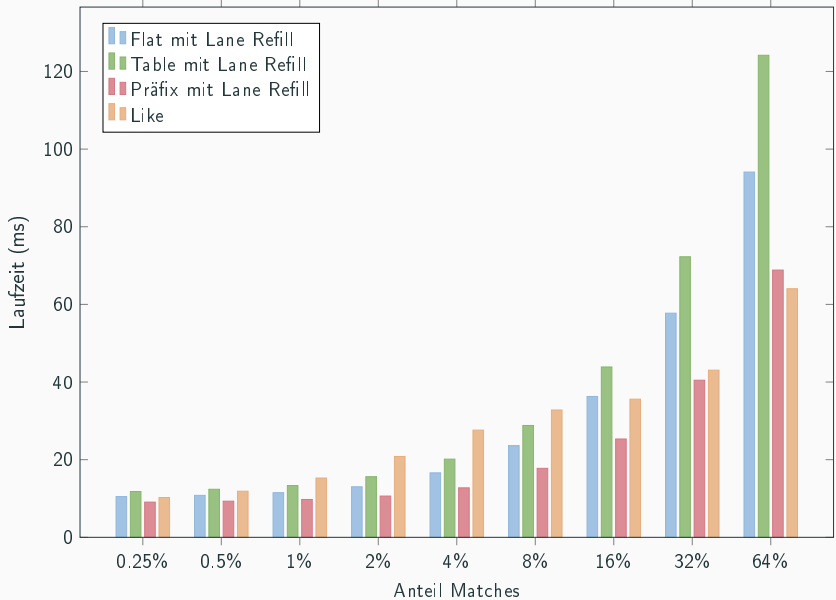


# Verbesserungen durch das Lane Refill für DBLP-Daten

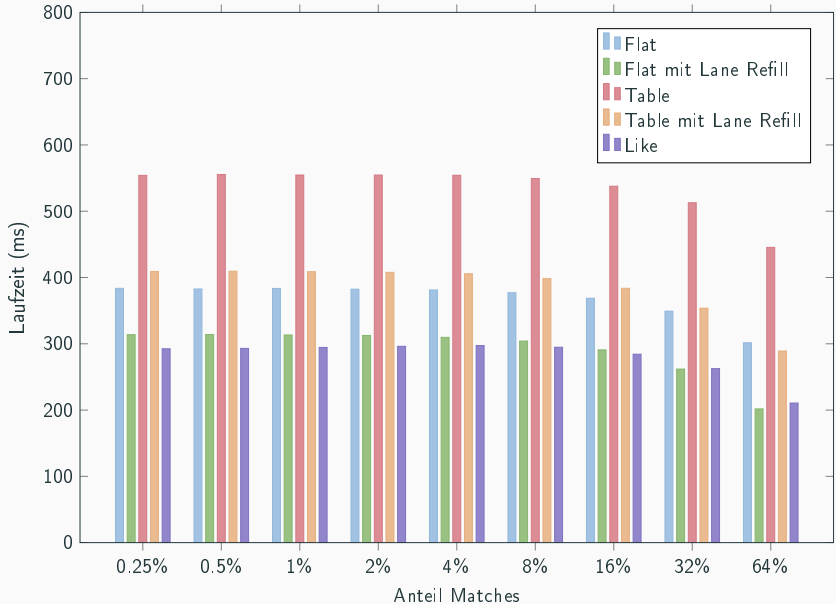




# Optimierte Algorithmen mit DBLP-Daten



# Benchmark, der beliebige Anfangszeichen zulässt



# Unterschiedliche Automatengrößen mit TPC-H-Daten

