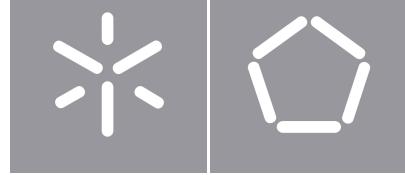


**University of Minho**  
School of Engineering

Diogo Alexandre Correia Marques

**Realistic Benchmarking of Data Deduplication  
and Compression Systems**





**University of Minho**  
School of Engineering

Diogo Alexandre Correia Marques

**Realistic Benchmarking of Data Deduplication  
and Compression Systems**

Master's Dissertation in Informatics Engineering

Dissertation supervised by  
**João Tiago Medeiros Paulo**

## Abstract

Data deduplication is a technique for identifying and removing duplicate content in storage systems, thereby contributing to better use of available space and consequent cost reduction. In fact, modern systems combine compression techniques to extract greater data density and store only what is strictly necessary.

By supporting various data manipulation techniques, the evaluation of these systems becomes increasingly complex, given that workloads need to meet a series of criteria that validate deduplication and compression simultaneously, without forgetting that the common characteristics between systems must remain the focus of the evaluation, in particular the spatial and temporal locality of accesses.

However, the benchmarks available to the community ([fio](#), vdbench) only allow partial manipulation of entropy and deduplication levels. Furthermore, trace simulation is too simplistic and becomes impractical in modern systems because they are too fast to complete the trace and there is no trivial way to extend it and preserve its characteristics.

Furthermore, in order to extract maximum performance, some systems only provide low-level [APIs](#), such as [SPDK](#), which makes it even more complicated to execute workloads, as the aforementioned benchmarks do not directly support such protocols for communication with the disk.

That said, this dissertation aims to develop a benchmark for storage systems, capable of supporting various [I/O](#) interfaces, as well as generating realistic workloads that allow the collection of relevant metrics for system evaluation, thus enabling the identification of performance bottlenecks and impacts associated with the characteristics of the system itself.

**Keywords** storage system, I/O interface, deduplication, compression, realistic workload.

## Resumo

A deduplicação de dados corresponde a uma técnica para identificar e remover conteúdos duplicados em sistemas de armazenamento, contribuindo assim para uma melhor utilização do espaço disponível e consequente redução de custos. De facto, os sistemas modernos combinam técnicas de compressão para extrair maior densidade dos dados e armazenar o estritamente necessário.

Ao suportarem várias técnicas de manipulação de dados, a avaliação destes sistemas torna-se cada vez mais complexa, dado que as workloads necessitam de responder a uma série de critérios que validem a deduplicação e compressão em simultâneo, sem esquecer que as características comuns entre sistemas devem permanecer no alvo da avaliação, em particular a localidade espacial e temporal dos acessos.

No entanto, os benchmarks disponíveis à comunidade ([fio](#), vdbench) apenas permitem uma manipulação parcial dos níveis de entropia e deduplicação. Ademais, a simulação de traces é demasiado simplista e torna-se impraticável em sistemas modernos por estes serem demasiado rápidos a concluir o trace e não existir uma forma trivial de o estender e preservar as suas características.

Além disso, no sentido de extrair o máximo de performance, alguns sistemas disponibilizam unicamente **APIs** de baixo nível, tal como **SPDK**, o que torna ainda mais complicada a execução de workloads, pois os benchmarks anteriormente referidos não suportam diretamente tais protocolos para comunicação com o disco.

Posto isto, esta dissertação tem por objetivo desenvolver um benchmark para sistemas de armazenamento, sendo este capaz de suportar diversas interfaces de **I/O**, bem como a geração de workloads realistas que permitam a recolha de métricas relevantes para a avaliação do sistema, permitindo assim a identificação de gargalos de desempenho e impactos associados às características do sistema em si.

**Palavras-chave** sistema de armazenamento, interface de I/O, deduplicação, compressão, workload realista.

# **Contents**

1	Introdução .....	2
1.1	Definição do Problema e Desafios .....	3
1.2	Objetivos e Contribuições .....	3
1.3	Estrutura do Documento .....	3

## **List of Figures**

## **List of Tables**

## **Acronyms**

**I/O** Input/Output.

**ZFS** Zettabyte File System.

**FIO** Flexible I/O Tester.

**API** Applications Programming Interface.

**SPDK** Storage Performance Development Kit.

**vbdev** virtual block device.

# **Capítulo 1**

## **Introdução**

Com o aumento das aplicações de inteligência artificial, a necessidade de processar e armazenar grandes quantidades de dados tornou-se cada vez mais relevante, consequentemente os sistemas de armazenamento evoluíram no sentido de oferecer uma maior eficiencia de acessos e densidade dos dados.

Sistemas de armazenamento modernos, como o **ZFS**, disponibilizam uma série de recursos que procuram melhorar a performance das aplicações. Em particular, destaca-se a deduplicação - geralmente abreviada para dedup - que procura reduzir o espaço de armazenamento utilizado ao não reescrever dados que já existam. Por outro lado, a compressão também exerce um papel relevante neste sistema, permitindo aumentar a entropia dos dados ao eliminar aqueles que de algum modo podem ser obtidos através de uma amostra menor.

Neste caso em particular, estamos interessados num sistema orientado ao bloco, portanto a técnica de deduplicação tem como unidade básica um bloco de bytes, geralmente 4096 bytes, mas este valor pode variar conforme o sistema em questão. Da mesma forma, as técnicas de compressão são aplicadas ao nível do bloco (intrabloco), no entanto é possível que alguns sistemas realizem uma análise de entropia entre blocos (interbloco) para obter mais densidade.

Por outro lado, e com o objetivo de ultrapassar as limitações impostas pela stack de **I/O** do kernel, nomeadamente as recorrentes mudanças de contexto, interrupções e cópias entre user e kernel space que tornam evidentes o gargalo de desempenho, surgiram várias **APIs** que visam resolver esses mesmos problemas e geralmente funcionam sobre runtimes assincronos.

Tendo em cosideração a diversidade de técnicas que podemos encontrar num sistema de armazenamento e interfaces de **I/O** existentes para interagir com o disco, torna-se difícil avaliar um qualquer sistema nos seus pontos específicos de funcionamento, isto porque nem todos os sistemas foram desenvolvidos para servir o mesmo fim, devendo assim procurar ajustar o benchmark e workload àquilo que o sistema oferece, pois somente assim será alcançada uma análise justa e correta do seu funcionamento.

## **1.1 Definição do Problema e Desafios**

## **1.2 Objetivos e Contribuições**

## **1.3 Estrutura do Documento**



