



**RECIMA21 - REVISTA CIENTÍFICA MULTIDISCIPLINAR**  
**ISSN 2675-6218**

**COMPARATIVO DE DESEMPENHO EM CONSULTAS SQL COM DIFERENTES TIPOS DE  
 ARMAZENAMENTO UTILIZANDO BANCO DE DADOS RELACIONAL**

**COMPARATIVE PERFORMANCE IN SQL QUERIES WITH DIFFERENT TYPES OF STORAGE  
 USING RELATIONAL DATABASE**

**RENDIMIENTO COMPARATIVO EN CONSULTAS SQL CON DIFERENTES TIPOS DE  
 ALMACENAMIENTO UTILIZANDO BASE DE DATOS RELACIONALES**

Vinicius Fernandes Bacili<sup>1</sup>, João Henrique Gião Borges<sup>2</sup>, Fabiana Florian<sup>2</sup>

e4114455

<https://doi.org/10.47820/recima21.v4i11.4455>

PUBLICADO: 11/2023

**RESUMO**

O objetivo deste trabalho é realizar um comparativo do desempenho de consultas SQL (*Structured Query Language*) em Banco de Dados Relacional utilizando diferentes tecnologias de armazenamento HDD, SDD e NVMe. Os bancos de dados tradicionais são construídos na premissa que os dados vão ficar armazenados em discos rígidos (HDD). Com o avanço da tecnologia, surgem novos meios para o armazenamento desses dados como os discos sólidos (SSD) e as memórias não voláteis (NVMe). Para verificar o desempenho dessas consultas, foram criados bancos de dados idênticos para as três tecnologias e medido seu desempenho através das consultas realizadas para o retorno dos dados cadastrados nesses bancos de dados.

**PALAVRAS-CHAVE:** Banco de dados. Comparativo. Consulta SQL. HDD. SSD. NVMe.

**ABSTRACT**

*The objective of this work is to compare the performance of SQL (Structured Query Language) queries in a Relational Database using different HDD, SDD and NVMe storage Technologies. Traditional databases are built on the premise that data will be stored on hard drives (HDD). With the advancement of technology, new means of storing this data emerge, such as solid disks (SSD) and non-volatile memories (NVMe). To verify the performance of these queries, identical databases were created for the three technologies and their performance was measured through queries carried out to return the data registered in these databases.*

**KEYWORDS:** Database. Comparative. SQL Query. HDD. SSD. NVMe.

**RESUMEN**

*El objetivo de este trabajo es comparar el rendimiento de consultas SQL (Structured Query Language) en una Base de Datos Relacional utilizando diferentes tecnologías de almacenamiento HDD, SDD y NVMe. Las bases de datos tradicionales se basan en la premisa de que los datos se almacenarán en discos duros (HDD). Con el avance de la tecnología surgen nuevos medios para almacenar estos datos, como los discos sólidos (SSD) y las memorias no volátiles (NVMe). Para verificar el desempeño de estas consultas se crearon bases de datos idénticas para las tres tecnologías y se midió su desempeño a través de consultas realizadas para devolver los datos registrados en estas bases de datos.*

**PALABRAS CLAVE:** Base de datos. Comparativo. Consulta SQL. HDD. SSD. NVMe.

<sup>1</sup> Graduando do Curso de Engenharia de Computação da Universidade de Araraquara- UNIARA. Araraquara-SP.

<sup>2</sup> Orientador (a). Docente Curso de Engenharia de Computação da Universidade de Araraquara- UNIARA. Araraquara-SP.



## RECIMA21 - REVISTA CIENTÍFICA MULTIDISCIPLINAR ISSN 2675-6218

COMPARATIVO DE DESEMPENHO EM CONSULTAS SQL COM DIFERENTES TIPOS DE  
ARMAZENAMENTO UTILIZANDO BANCO DE DADOS RELACIONAL  
Vinicius Fernandes Bacili, João Henrique Gião Borges, Fabiana Florian

### 1 INTRODUÇÃO

No modelo tradicional, na estrutura de armazenamento dos Bancos de Dados, a predominância vem dos HDD (*Hard Disk Drive*), suportando grandes volumes de dados com confiabilidade, porém, com o passar do tempo, o desempenho de tal tecnologia vem se tornando cada vez mais lento, seja pelo tempo da tecnologia ou pela quantidade massiva de dados armazenados.

Com o surgimento do SSD (*Solid State Drive*) tem-se um novo marco para o armazenamento,

[...] talvez o aspecto mais impressionante tenha sido o aumento massivo na velocidade e confiabilidade da acessível tecnologia de um SSD. A ausência de partes móveis comparada a um disco rígido foi um divisor de águas no quesito consumo de energia, confiabilidade a longo prazo e silêncio. Tudo isso notavelmente melhora o desempenho de qualquer dispositivo que possua um drive de estado sólido (SSD) ao invés de um disco giratório (Kingston, 2022).

Sendo os dados uma parte vital para qualquer sistema em uma empresa, tem ocorrido avanços e pesquisas para novos tipos de armazenamentos, um desses avanços é a tecnologia NVMe (*Non-Volatile Memory Express*), enquanto HDD e SSDs se utilizam da tecnologia SATA o NVMe foi desenvolvido especificamente para SSDs, criando-se, assim, um novo protocolo para o armazenamento, leitura e escrita dos mais variados tipos de dados.

Visando encontrar uma significativa diferença no uso de tecnologias de armazenamento mais antigas HDD e SSDs para a NVMe, foi realizada a pesquisa bibliográfica, complementada pela utilização de um computador com as seguintes configurações:

- Processador AMD Ryzen 9 5900X;
- Memória Asgard 32GB – 2x16 - 3200MHz;
- Placa de vídeo AMD RX580 – 8GB;
- Armazenamentos: HDD 500GB, SSD 1TB, NVMe 256GB.

Os programas definidos para a realização dos testes foram:

- Banco de Dados no Oracle 21c;
- SQL Developer.

O Quadro 1 apresenta as informações das tecnologias utilizadas

NOME	VERSÃO	LOCALIZAÇÃO	DESCRIÇÃO DE USO
Oracle Database 21c		<a href="https://www.oracle.com/br/database/technologies/oracle-database-software-downloads.html">https://www.oracle.com/br/database/technologies/oracle-database-software-downloads.html</a>	Gerenciador do banco de dados.



## RECIMA21 - REVISTA CIENTÍFICA MULTIDISCIPLINAR

ISSN 2675-6218

COMPARATIVO DE DESEMPENHO EM CONSULTAS SQL COM DIFERENTES TIPOS DE ARMAZENAMENTO UTILIZANDO BANCO DE DADOS RELACIONAL  
Vinicius Fernandes Bacili, João Henrique Gião Borges, Fabiana Florian

SQL Developer		<a href="https://www.oracle.com/database/sqldeveloper/technologies/download/">https://www.oracle.com/database/sqldeveloper/technologies/download/</a>	Ferramenta utilizada para fazer o desenvolvimento, acompanhamento e coleta, para a mensuração dos dados.
---------------	--	---	--

Quadro 1: Ferramentas e tecnologias utilizadas (autoria própria), 2023

A partir das configurações do computador, dos programas definidos e das tecnologias utilizadas, o comparativo do desempenho das consultas foi realizado com mais de sessenta casos de teste, onde foram obtidos resultados variados para cada tecnologia utilizada nesta pesquisa.

## 2 REVISÃO BIBLIOGRÁFICA

Esta seção apresenta os conceitos básicos e outros estudos relacionados ao tema proposto neste trabalho. A revisão da literatura subdividiu-se nos itens: (2.1) banco de dados, (2.2) Gerenciador de banco de dados, (2.3) linguagem SQL, (2.4) disco rígido, (2.5) SSD e (2.6) NVMe.

### 2.1 Banco de dados

Banco de dados são uma parte essencial na vida da tecnologia moderna, armazenando dados de maneira eficiente e prática, tendo um papel crítico para o armazenamento de dados de sistemas de diversas áreas em que sistemas para o controle de dados são utilizados, incluindo engenharia, medicina, educação e comércio.

Elmasri e Navathe (2011) citam que um banco de dados é uma coleção de dados relacionados, com isso podemos afirmar que uma coleção de dados são vários dados separados por itens que por sua vez são fatos conhecidos que podem ser registrados e possuem significado implícito.

Date (2003) afirma que um banco de dados é uma coleção de dados persistentes, usada pelos sistemas de aplicação de uma determinada empresa. Nesse contexto, o termo “empresa” é utilizado como uma definição genérica para qualquer organização comercial, científica, técnica ou autônoma.

ORACLE (2022) descreve que um banco de dados é uma coleção organizada de informações - ou dados - estruturadas, normalmente armazenadas eletronicamente em um sistema de computador.

Existem vários tipos de bancos de dados, cada qual com sua especificação, e compete ao usuário escolher a melhor maneira de utilizar seus dados, para essa pesquisa utilizaremos o banco de dados relacional, onde seus dados são repartidos em colunas e tabelas definidas pelo usuário do banco de dados, ORACLE (2022). Os itens em um banco de dados relacional são organizados como um conjunto de tabelas com colunas e linhas. A tecnologia de banco de dados relacional fornece a maneira mais eficiente e flexível de acessar informações estruturadas.



## RECIMA21 - REVISTA CIENTÍFICA MULTIDISCIPLINAR ISSN 2675-6218

COMPARATIVO DE DESEMPENHO EM CONSULTAS SQL COM DIFERENTES TIPOS DE  
ARMAZENAMENTO UTILIZANDO BANCO DE DADOS RELACIONAL  
Vinicius Fernandes Bacili, João Henrique Gião Borges, Fabiana Florian

### 2.2 Gerenciador de banco de dados

Utilizando-se de um banco de dados, surge a necessidade de se obter um Sistema de Gerenciamento de Banco de Dados (SGBD) que auxilia de forma simples e efetiva a organização, segurança e acesso aos dados salvos no banco de dados. O SGBD atua como um comunicador entre o usuário e o banco de dados, é o SGBD que recebe e processa todas as requisições feitas diretamente pelo usuário ou por meio de integrações.

Segundo a ORACLE (2022), um SGBD também facilita a supervisão e o controle de bancos de dados, permitindo uma variedade de operações administrativas, como monitoramento de desempenho, ajuste e backup e recuperação.

### 2.3 Linguagem SQL

Structured Query Language (SQL), é uma linguagem de programação que foi criada especificamente para a manipulação de dados dentro dos SGBD. ORACLE (2022) SQL é uma linguagem de programação usada por quase todos os bancos de dados relacionais para consultar, manipular e definir dados e fornecer controle de acesso. E dentro da linguagem SQL temos as seguintes divisões de comandos.

DML – *Data Manipulation Language*: comandos usados para manipulação, alteração e atualização de dados.

DDL – *Data Definition Language*: comandos usados para o gerenciamento e estruturação de tabelas

DCL – *Data Control Language*: comandos usados para controle de permissões ao banco de dados.

DTL – *Data Transaction Language*: comandos usados para salvar alterações feitas.

### 2.4 Disco rígido

O HDD é um disco mecânico usado para o armazenamento de dados, por ser uma tecnologia antiga, seu baixo custo e opções de tamanho para o armazenamento de dados faz com que ele ainda seja amplamente utilizado. Segundo DEVMEDIA (2016), foram os primeiros dispositivos de armazenamento em larga escala empregados. Esse tipo de *hardware* de armazenamento é composto por discos magnéticos (ou discos rígidos) que armazenam as informações, onde a leitura e gravação dos dados são efetuadas por uma agulha mecânica.



## RECIMA21 - REVISTA CIENTÍFICA MULTIDISCIPLINAR ISSN 2675-6218

COMPARATIVO DE DESEMPENHO EM CONSULTAS SQL COM DIFERENTES TIPOS DE  
ARMAZENAMENTO UTILIZANDO BANCO DE DADOS RELACIONAL  
Vinicius Fernandes Bacili, João Henrique Gião Borges, Fabiana Florian

A Figura 1 apresenta as diferenças entre as estruturas dos três tipos de tecnologias.



Figura 1: Comparativo entre HDD x SSD x NVMe (autoria própria), 2022

### 2.5 SSD

O SSD é um dispositivo de armazenamento com acesso direto, assim dispensando qualquer componente mecânico o que faz que ele seja mais seguro, rápido e duradouro que seu antecessor. Segundo Devmedia (2016), há um destaque para o custo ainda elevado da tecnologia SSD, sendo mais caro que seu antecessor, porém, vem se mostrando cada vez mais acessível com o tempo e com o surgimento de novas tecnologias de armazenamento, com essa afirmação da INTEL (2021) Os SSDs são mais caros que os HDDs por quantidade de armazenamento (em *gigabytes* (GB) e *terabytes* (TB)), mas a diferença está diminuindo à medida que os preços dos SSDs caem em um ritmo mais rápido do que os preços dos HDD ano após ano.

### 2.6 NVMe

NVMe (*Non-Volatile Memory Express*) é uma *interface* e *driver* de comunicação de alto desempenho para SSDs que se utilizam de memória não volátil, embora não tão utilizada como seus antecessores por ainda possuir um alto custo comparado com eles, o sistema NVMe inicia um novo marco para a tecnologia de armazenamento de dados, Kingston (2022) O protocolo NVMe utiliza caminhos de dados paralelos, de baixa latência para a mídia central, como arquiteturas de processador de alto desempenho. Por sua vez, a DELL NVMe representa a próxima etapa evolucionário para mídia de armazenamento *flash* e mídia de memória de classe de armazenamento de última geração.



## RECIMA21 - REVISTA CIENTÍFICA MULTIDISCIPLINAR ISSN 2675-6218

COMPARATIVO DE DESEMPENHO EM CONSULTAS SQL COM DIFERENTES TIPOS DE  
ARMAZENAMENTO UTILIZANDO BANCO DE DADOS RELACIONAL  
Vinicius Fernandes Bacili, João Henrique Gião Borges, Fabiana Florian

### 3 DESENVOLVIMENTO

De acordo com o objetivo proposto, foi necessário criar um banco de dados modelo e o replicá-lo para os demais componentes, tendo três bancos de dados idênticos.

Foi realizada uma instalação limpa do Oracle 21c Enterprise se utilizando do assistente de instalação da Oracle, com a instalação realizada é necessário ser feita a criação de três bancos de dados, sendo eles BD\_HDD, BD\_SSD e BD\_NVME. Onde foram criados respectivamente no HDD, SSD e NVMe. Foram criadas três tabelas para conter os dados que serão armazenados, sendo elas TB\_HDD, TB\_SSD e TB\_NVME. Cada tabela foi criada em seu respectivo tipo de armazenamento, conforme a Figura 2.

```
CREATE TABLESPACE DB_NVME
DATAFILE 'C:\BD\TABLESPACE\TBS_NVME.DBS'
SIZE 1000M AUTOEXTEND ON NEXT 50M
DEFAULT NO INMEMORY ONLINE;

CREATE TABLESPACE DB_SSD
DATAFILE 'D:\BD\TABLESPACE\BD_SSD.DBS'
SIZE 1000M AUTOEXTEND ON NEXT 50M
DEFAULT NO INMEMORY ONLINE;

CREATE TABLESPACE DB_HDD
DATAFILE 'E:\BD\TABLESPACE\BD_HDD.DBS'
SIZE 1000M AUTOEXTEND ON NEXT 50M
DEFAULT NO INMEMORY ONLINE;
```

Figura 2: Criação dos *tablespaces* (autoria própria, 2023)

Para cada *tablespace* foi reproduzido a mesma tabela, utilizando-se de nomes fictícios para as colunas da tabela que se baseia em um sistema de controle de produtos de uma loja, tendo isso em vista a tabela foi criada com as seguintes colunas *id\_nvme*, *id\_produto*, *id\_cliente*, *id\_nota*, *numero1*, *numero2*, *numero3*, *numero4*, *numero5*, *numero6*, *numero7*, *texto1*, *texto2*, *texto3*, *texto4*, *texto5*, *texto6*, *texto7*, *data1* e *data2*, tais colunas foram criadas respectivamente, sendo utilizados os tipos de *integer*, *number*, *varchar2* e *date* e ao final do script, vemos o local de referência onde a tabela vai ser armazenada sendo o DB\_NVME. Para o id da tabela foi utilizada uma chave primária com um número auto incremental controlado pelo próprio bando de dados, criando assim um índice para a tabela.

A Figura 3 mostra como foi gerada a criação da TB\_NVME, onde o mesmo *script* foi utilizado para criar as demais tabelas, sendo somente alterado o nome da tabela, seu id e o *tablespace* de onde ela ficara armazenada com isso para as demais tabelas temos seus nomes TB\_SSD e



## RECIMA21 - REVISTA CIENTÍFICA MULTIDISCIPLINAR ISSN 2675-6218

COMPARATIVO DE DESEMPENHO EM CONSULTAS SQL COM DIFERENTES TIPOS DE  
ARMAZENAMENTO UTILIZANDO BANCO DE DADOS RELACIONAL  
Vinicius Fernandes Bacili, João Henrique Gião Borges, Fabiana Florian

TB\_HDD, seus ids `id_ssd` e `id_hdd` e o local de referência para ser armazenado ao final do *script* `DB_SSD` e `DB_HDD`.

```
CREATE TABLE TB_NVME (
  ID_NVME INTEGER GENERATED BY DEFAULT ON NULL AS IDENTITY
, ID_NOTA INTEGER, ID_CLIENTE INTEGER, ID_PRODUTO INTEGER, NUMERO1
NUMBER(38,2), NUMERO2 NUMBER(38,2), NUMERO3 NUMBER(38,2), NUMERO4
NUMBER(38,2), NUMERO5 NUMBER(38,2), NUMERO6 NUMBER(38,2), NUMERO7
NUMBER(38,2), TEXTO1 VARCHAR2(1), TEXTO2 VARCHAR2(2), TEXTO3 VARCHAR2(3),
TEXTO4 VARCHAR2(4), TEXTO5 VARCHAR2(5), TEXTO6 VARCHAR2(6), TEXTO7 VARCHAR2(7),
DATA1 DATE, DATA2 DATE, CONSTRAINT PK_TB_NVME PRIMARY KEY(ID_NVME)
) TABLESPACE DB_NVME;
```

Figura 3: Criação da tabela TB\_NVME (autoria própria, 2023).

Com as tabelas criadas foi necessário gerar e inserir os registros, onde também são utilizados de dados fictícios e aleatórios gerados pelo próprio Oracle. Para o teste ser o mais fiel possível, foi desenvolvido o *script* da Figura 4, que por sua vez vai gerar e inserir 30.000.000 de linhas em cada tabela. Para a coluna `id_nvme` é atribuído um valor sequencial gerado automaticamente pelo banco de dados, a coluna `id_nota` recebeu um valor sequencial de 1 a 30.000.000 conforme a execução do *script*, a coluna `id_cliente` recebeu um valor aleatório de 1 a 20.000, a coluna `id_produto` recebeu números aleatórios de 20.000 a 40.000, as colunas 'numero' receberam valores aleatórios de 1 a 200.000 com duas casas decimais, as colunas 'texto' receberam textos aleatórios e as colunas de datas receberam datas aleatórias entre os anos de 2010 e 2020.

```
DECLARE
ID_NOTA NUMBER;ID_CLIENTE NUMBER; ID_PRODUTO NUMBER;N1 NUMBER; N2 NUMBER;
N3 NUMBER; N4 NUMBER; N5 NUMBER; N6 NUMBER; N7 NUMBER; T1 VARCHAR2(1); T2
VARCHAR2(2); T3 VARCHAR2(3); T4 VARCHAR2(4); T5 VARCHAR2(5); T6 VARCHAR2(6); T7
VARCHAR2(7); DATA1 DATE; DATA2 DATE;
BEGIN
  FOR I IN 1..30000000
  LOOP
    SELECT
      /* Gera números aleatórios entre 1 e 20.000 para código do cliente */
      ROUND(DBMS_RANDOM.VALUE (1, 20000)),
      /* Gera números aleatórios entre 20.000 e 40.000 para código do produto */
      ROUND(DBMS_RANDOM.VALUE (20000, 40000)),
      /* Gera números aleatórios para as colunas numéricas */
```



## RECIMA21 - REVISTA CIENTÍFICA MULTIDISCIPLINAR ISSN 2675-6218

COMPARATIVO DE DESEMPENHO EM CONSULTAS SQL COM DIFERENTES TIPOS DE  
ARMAZENAMENTO UTILIZANDO BANCO DE DADOS RELACIONAL  
Vinicius Fernandes Bacili, João Henrique Gião Borges, Fabiana Florian

```

ROUND(DBMS_RANDOM.VALUE (1, 200000), 2), ROUND(DBMS_RANDOM.VALUE (1,
200000), 2), ROUND(DBMS_RANDOM.VALUE (1, 200000), 2), ROUND(DBMS_RANDOM.VALUE
(1, 200000), 2), ROUND(DBMS_RANDOM.VALUE (1, 200000), 2),
ROUND(DBMS_RANDOM.VALUE (1, 200000), 2), ROUND(DBMS_RANDOM.VALUE (1, 200000),
2),
/* Gera os caracteres aleatórios para os campos texto */
DBMS_RANDOM.STRING('U', 1), DBMS_RANDOM.STRING('U', 2),
BMS_RANDOM.STRING('U', 3), DBMS_RANDOM.STRING('U', 4), DBMS_RANDOM.STRING('U',
5),DBMS_RANDOM.STRING('U', 6), DBMS_RANDOM.STRING('U', 7),
/*Gera as datas aleatórias para os campos data */
TO_DATE(TRUNC(DBMS_RANDOM.VALUE(2455196,2458846)), 'J'),
TO_DATE(TRUNC(DBMS_RANDOM.VALUE(2455196,2458846)), 'J')
/*Guarda os valores dentro das variáveis que vão ser usadas na inserção */
INTO ID_CLIENTE, ID_PRODUTO, N1, N2, N3, N4, N5, N6, N7, T1, T2, T3, T4, T5, T6, T7,
DATA1, DATA2 FROM DUAL;
/* Insere uma linha na tabela em cada loop */
INSERT INTO TB_NVME (ID_NVME, ID_NOTA, ID_CLIENTE, ID_PRODUTO, NUMERO1,
NUMERO2, NUMERO3, NUMERO4, NUMERO5, NUMERO6, NUMERO7, TEXTO1, TEXTO2,
TEXTO3, TEXTO4, TEXTO5, TEXTO6, TEXTO7, DATA1, DATA2
) VALUES(
NULL, ID_NOTA, ID_CLIENTE, ID_PRODUTO, N1, N2, N3, N4, N5, N6, N7, T1, T2, T3, T4, T5, T6,
T7, DATA1, DATA2);
IF (MOD(I,1000)=0)THEN /* Commit a cada 1000 linhas */ COMMIT; END IF;
END LOOP;
END;

```

Figura 4: Geração e inserção dos dados nas tabelas (autoria própria, 2023)

Com os dados criados nas tabelas, foi necessário criar o *script*<sup>1</sup> contido na Figura 5, dentro dele possuímos um meio para fazer a consulta dos dados dentro de cada uma das tabelas de forma separada, este *script* vai criar uma função que vamos chamar de 'EXEC\_TB\_NVME' que deverá receber por parâmetro os valores 'ID\_EXEC', 'LINHA\_INICIO' e 'LINHA\_FIM', que são respectivamente o número da execução, a linha inicial e final que vão abranger a consulta dentro da tabela, para facilitar a chamada da função, os parâmetros dois e três serão multiplicados por 1.000.000, portanto, quando a função for chamada, os parâmetros que devem ser passados ficaram assim (1, 1, 2), executando uma consulta no banco de dados na tabela TB\_NVME retornando os valores contidos entre as linhas 1.000.000 e 2.000.000. No *script* da Figura 5 é mostrada a criação da

<sup>1</sup> *Script*: Contém instruções SQL, comandos condicionais, variáveis, comentários, palavras-chave e outros elementos textuais importantes para lidar com dados e representar a lógica de algum algoritmo.





## RECIMA21 - REVISTA CIENTÍFICA MULTIDISCIPLINAR ISSN 2675-6218

COMPARATIVO DE DESEMPENHO EM CONSULTAS SQL COM DIFERENTES TIPOS DE  
ARMAZENAMENTO UTILIZANDO BANCO DE DADOS RELACIONAL  
Vinicius Fernandes Bacili, João Henrique Gião Borges, Fabiana Florian

função EXEC\_TB\_NVME, para as demais tabelas utilizamos o mesmo *script* como base somente alterando os nomes para 'EXEC\_TB\_SSD' e 'EXEC\_TB\_HDD' e alterado a tabela de onde será feita a consulta.

```
CREATE OR REPLACE FUNCTION EXEC_TB_NVME
(
  ID_EXEC      IN NUMBER, LINHA_INICIO IN NUMBER, LINHA_FIM   IN NUMBER ) RETURN
VARCHAR2 AS N1  NUMBER; VINI NUMBER; VFIM NUMBER;
BEGIN
  V_INI := LINHA_INICIO * 1000000;
  V_FIM := LINHA_FIM * 1000000;
  N1 := 0;
  FOR I IN(
    SELECT ID_NVME, ID_NOTA, ID_CLIENTE, ID_PRODUTO, NUMERO1, NUMERO2, NUMERO3,
    NUMERO4, NUMERO5, NUMERO5, NUMERO7, TEXTO1, TEXTO2, TEXTO3, TEXTO4, TEXTO5,
    TEXTO6, TEXTO7, DATA1, DATA2
      FROM TB_NVME
     WHERE ID_NVME BETWEEN V_INI AND V_FIM)
  LOOP
    N1 := I.NUMERO1;
  END LOOP;
  RETURN 'EXEC: '||ID_EXEC||' - INICIO: '||V_INI||' - FIM: '||V_FIM;
END;
```

Figura 5: Função para a execução da consulta nas tabelas (autoria própria, 2023)

Para executar as funções contidas na Figura 5, foi criada a Figura 6, que contém um trecho de código que executa a chamada para função vinte vezes, fazendo vinte consultas ao banco de dados, passando valores diferentes aos parâmetros da função para se evitar a memória cache<sup>2</sup> do banco de dados, esse *script* também mede o tempo de cada consulta realizada pelo *script* 4, salvando esses dados em um arquivo de texto. O *script* da Figura 6 mostra como é realizado a chamada do *script* da Figura 5 e o salvamento dos dados retornados na consulta em um arquivo de texto que vamos chamar de 'TB\_NVME\_1KK.sql' tendo seu nome alterado para 'TB\_SSD\_1KK.sql' e 'TB\_HDD\_1KK.sql' para realizar a consulta em suas devidas tabelas. Cada *script* vai retornar vinte consultas com 1.000.000 de linhas cada. Foram criados *scripts* que retornam 1.000.000, 3.000.000, 9.000.000 e 15.000.000 de linhas para cada uma das três tabelas.

<sup>2</sup> Memória cache armazena dados para agilizar execução de aplicações. Retornando assim os mesmos dados mais rapidamente, se consultados com duplicidade.



## RECIMA21 - REVISTA CIENTÍFICA MULTIDISCIPLINAR ISSN 2675-6218

COMPARATIVO DE DESEMPENHO EM CONSULTAS SQL COM DIFERENTES TIPOS DE  
ARMAZENAMENTO UTILIZANDO BANCO DE DADOS RELACIONAL  
Vinicius Fernandes Bacili, João Henrique Gião Borges, Fabiana Florian

```

set timing on
column dt new_val X
select to_char(sysdate,'yyyymmddhh24mmss') dt from dual;
spool c:\BD\resultado_1kk_NVME&X.txt
select EXEC_TB_NVME(1, 1, 2)    from dual;
select EXEC_TB_NVME(2, 2, 3)    from dual;
select EXEC_TB_NVME(3, 3, 4)    from dual;
select EXEC_TB_NVME(4, 4, 5)    from dual;
select EXEC_TB_NVME(5, 5, 6)    from dual;
select EXEC_TB_NVME(6, 6, 7)    from dual;
select EXEC_TB_NVME(7, 7, 8)    from dual;
select EXEC_TB_NVME(8, 8, 9)    from dual;
select EXEC_TB_NVME(9, 9, 10)   from dual;
select EXEC_TB_NVME(10, 10, 11) from dual;
select EXEC_TB_NVME(11, 11, 12) from dual;
select EXEC_TB_NVME(12, 12, 13) from dual;
select EXEC_TB_NVME(13, 13, 14) from dual;
select EXEC_TB_NVME(14, 14, 15) from dual;
select EXEC_TB_NVME(15, 15, 16) from dual;
select EXEC_TB_NVME(16, 16, 17) from dual;
select EXEC_TB_NVME(17, 17, 18) from dual;
select EXEC_TB_NVME(18, 18, 19) from dual;
select EXEC_TB_NVME(19, 19, 20) from dual;
select EXEC_TB_NVME(20, 20, 21) from dual;
spool off

```

Figura 6: *Script* que executa a chamada para a função EXEC\_TB\_NVME

Para aferição dos resultados, foram analisadas as medidas do tempo de execução individual de cada chamada de função, que foi registrado pelo PL/SQL e gravado em arquivo de texto, o registro de tempo no PL/SQL é ativado pelo comando “*set timing on*” que é mostrado no *script*.

#### 4 RESULTADOS

Os resultados estão organizados com base no número de linhas consultadas, em ordem crescente. Iniciou-se a consulta de um milhão de linhas e concluímos com a consulta de quinze milhões de linhas. Nos gráficos subsequentes, cada tecnologia analisada foi identificada da seguinte maneira: disco rígido (HDD), (SSD) e (NVMe).

Para cada quantidade de linhas, apresentamos dois gráficos. Um deles exibe os tempos de execução de cada consulta, enquanto o outro mostra o tempo médio de execução em cada tecnologia. Quanto menor o tempo registrado melhor o desempenho.

**RECIMA21 - Ciências Exatas e da Terra, Sociais, da Saúde, Humanas e Engenharia/Tecnologia**



## RECIMA21 - REVISTA CIENTÍFICA MULTIDISCIPLINAR ISSN 2675-6218

COMPARATIVO DE DESEMPENHO EM CONSULTAS SQL COM DIFERENTES TIPOS DE ARMAZENAMENTO UTILIZANDO BANCO DE DADOS RELACIONAL  
Vinicius Fernandes Bacili, João Henrique Gião Borges, Fabiana Florian

Na Figura 7 são exibidos os tempos individuais de cada consulta com 1.000.000 de linhas.

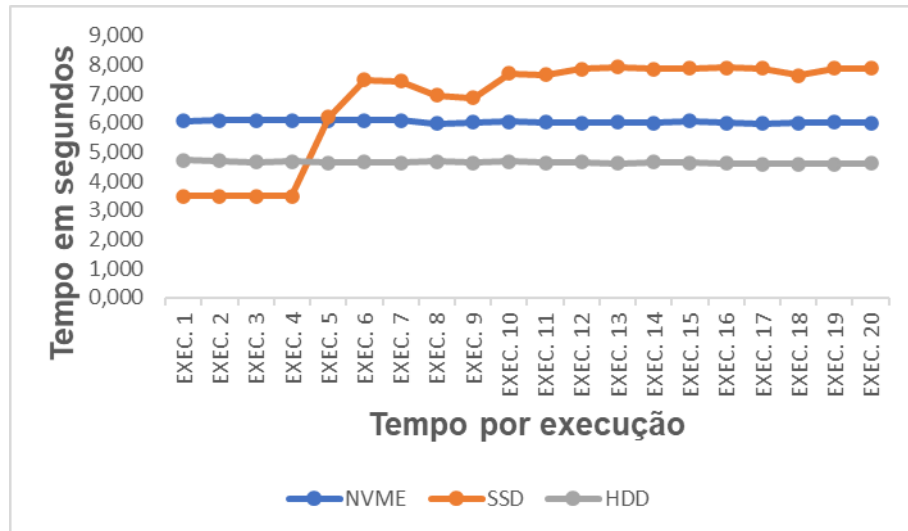


Figura 7: Resultado de cada consulta com 1.000.000 de linhas (autoria própria, 2023)

Na Figura 8 são exibidos os tempos médios de execução da consulta com 1.000.000 de linhas obtidos por meio da função “set timing on” da Figura 6.

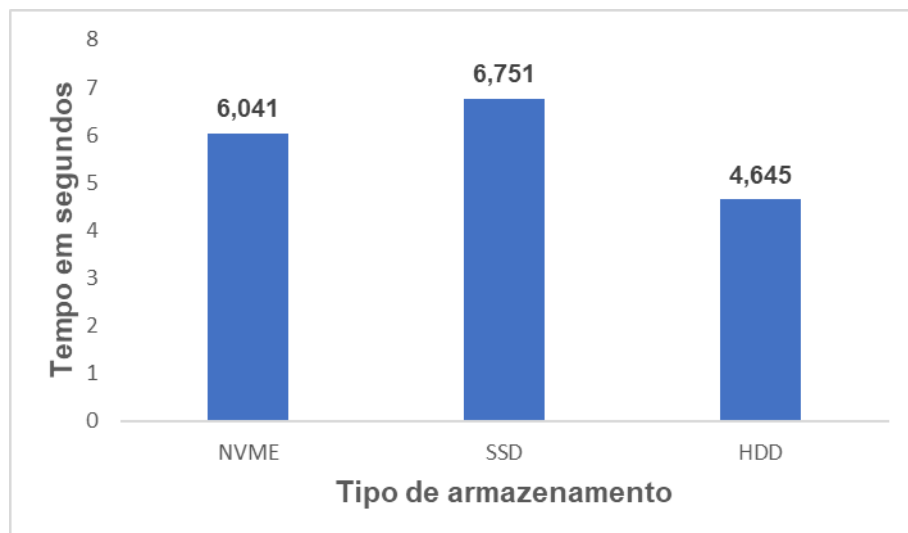


Figura 8: Tempo médio da consulta com 1.000.000 de linhas (autoria própria, 2023)

Nos testes com 1.000.000 de linhas a consulta no SSD teve muitas variações de tempo e a maior média dos tempos sendo a menos eficiente no quesito de velocidade, a consulta no NVMe não teve quase nenhuma variação nos seus tempos de medida e apresentou resultados 11% mais rápidos que o SSD, já a consulta em HDD praticamente não teve variações e seus resultados foram 32% melhores que o modelo do SSD.

Na Figura 9 são exibidos os tempos individuais de execução de cada consulta com 3.000.000 de linhas.



## RECIMA21 - REVISTA CIENTÍFICA MULTIDISCIPLINAR ISSN 2675-6218

COMPARATIVO DE DESEMPENHO EM CONSULTAS SQL COM DIFERENTES TIPOS DE  
ARMAZENAMENTO UTILIZANDO BANCO DE DADOS RELACIONAL  
Vinicius Fernandes Bacili, João Henrique Gião Borges, Fabiana Florian

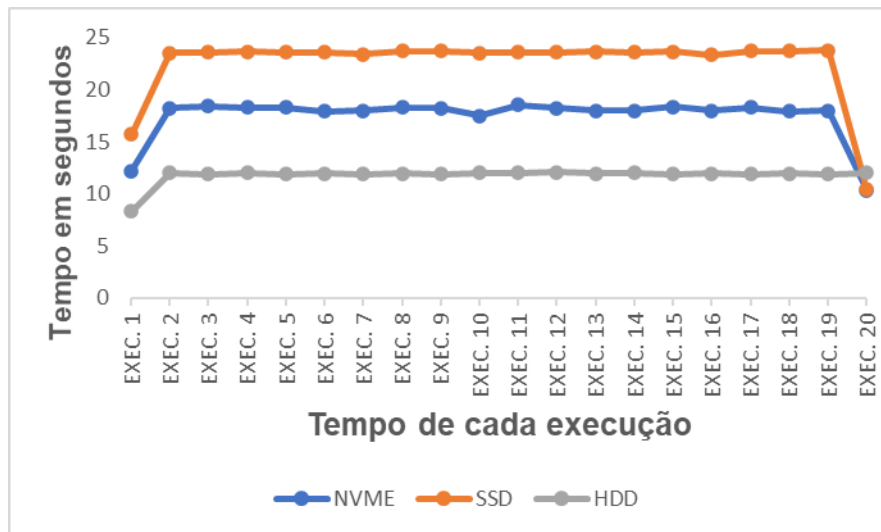


Figura 9 – Resultado de cada consulta com 3.000.000 de linhas (autoria própria, 2023)

Na Figura 10 são exibidos os tempos médios de execução da consulta com 3.000.000 de linhas.

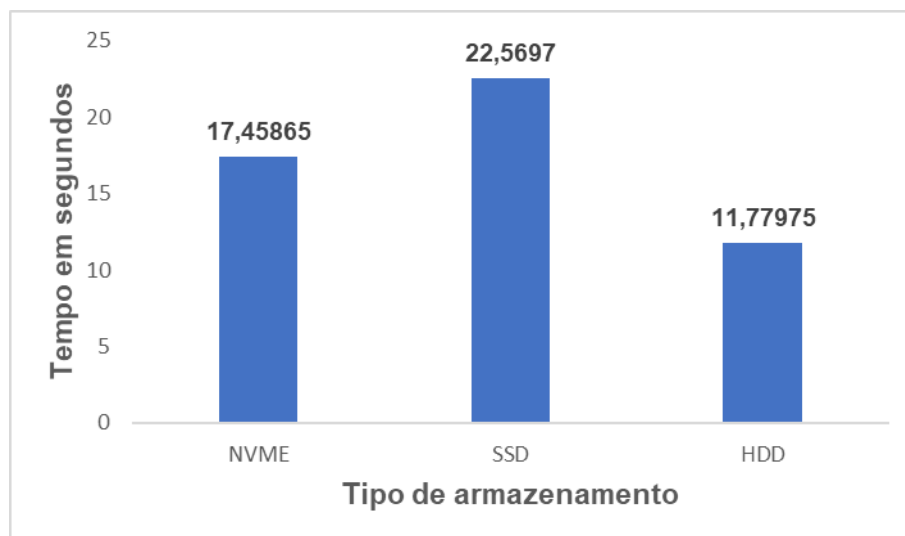


Figura 10 – Tempo médio da consulta com 3.000.000 de linhas (autoria própria, 2023)

Nos testes com 3.000.000 de linhas, todas as tecnologias apresentaram uma grande variedade na primeira e última execução, mas mantiveram poucas variações. Durante as outras execuções, a consulta no SSD novamente foi a menos ágil, a consulta no NVMe apresentou na média resultados 23% melhores que o SSD, já a consulta no HDD novamente foi a que teve maior eficiência em velocidade apresentando resultados 48% melhores que o SSD.

Na Figura 11 são exibidos os tempos individuais de execução de cada consulta com 9.000.000 de linhas.



## RECIMA21 - REVISTA CIENTÍFICA MULTIDISCIPLINAR ISSN 2675-6218

COMPARATIVO DE DESEMPENHO EM CONSULTAS SQL COM DIFERENTES TIPOS DE ARMAZENAMENTO UTILIZANDO BANCO DE DADOS RELACIONAL  
Vinicius Fernandes Bacili, João Henrique Gião Borges, Fabiana Florian

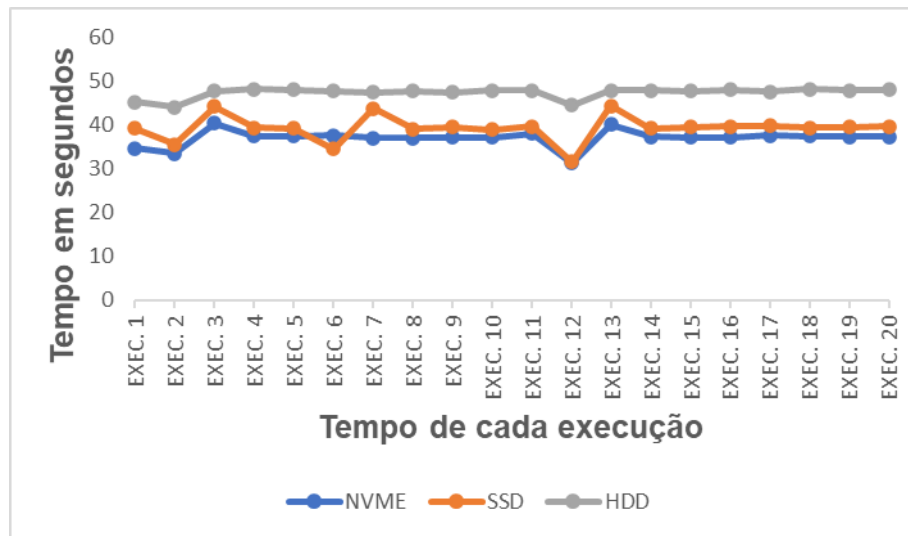


Figura 11 - Resultado de cada consulta com 9.000.000 de linhas (autoria própria, 2023)

Na Figura 12 são exibidos os tempos médios de execução da consulta com 9.000.000 de linhas.

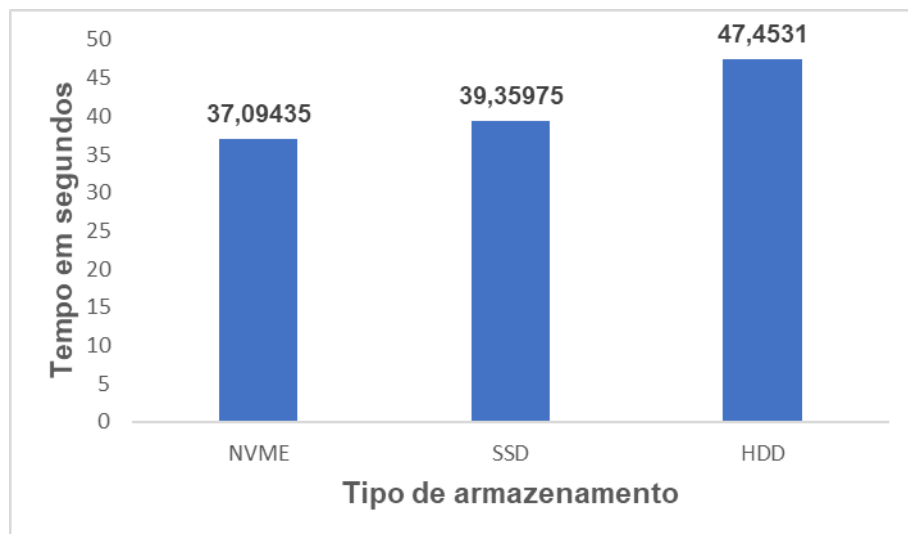


Figura 12 – Tempo médio da consulta com 9.000.000 de linhas (autoria própria, 2023)

Nos testes com 9.000.000 linhas, o HDD se tornou a tecnologia com o pior desempenho, tendo suas execuções com baixa variação, mas com tempos maiores com relação às outras, a consulta no SSD teve algumas variações em suas consultas, e na média das execuções apresentou resultados 18% mais rápidos que o HDD, já a consulta no NVMe se tornou a tecnologia mais eficiente com uma baixa variação entre suas execuções e obteve resultados 22% melhores que o HDD.

Na Figura 13 são exibidos os tempos individuais de execução de cada consulta com 15.000.000 de linhas.



## RECIMA21 - REVISTA CIENTÍFICA MULTIDISCIPLINAR ISSN 2675-6218

COMPARATIVO DE DESEMPENHO EM CONSULTAS SQL COM DIFERENTES TIPOS DE ARMAZENAMENTO UTILIZANDO BANCO DE DADOS RELACIONAL  
Vinicius Fernandes Bacili, João Henrique Gião Borges, Fabiana Florian

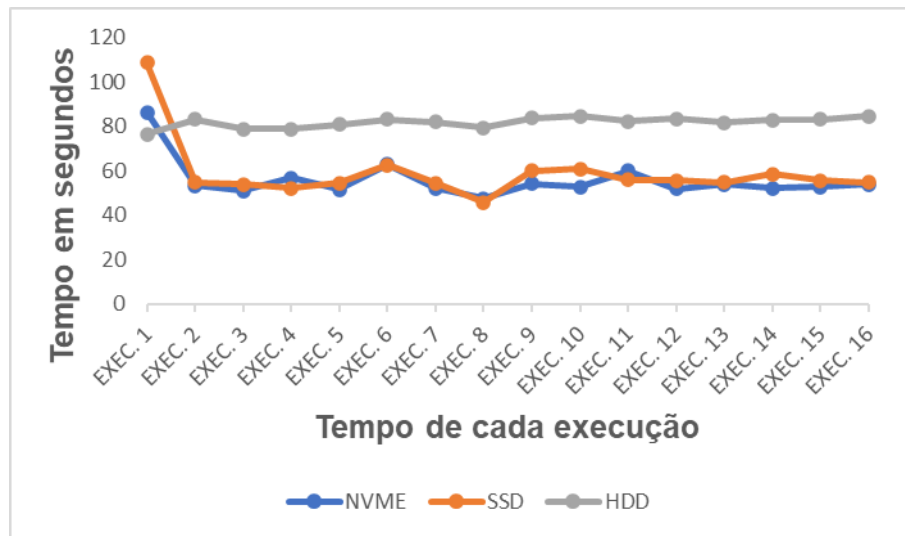


Figura 13 – Resultado de cada consulta com 15.000.000 de linhas (autoria própria, 2023)

Na figura 14 são apresentados os tempos médios de execução da consulta com 15.000.000 de linhas.

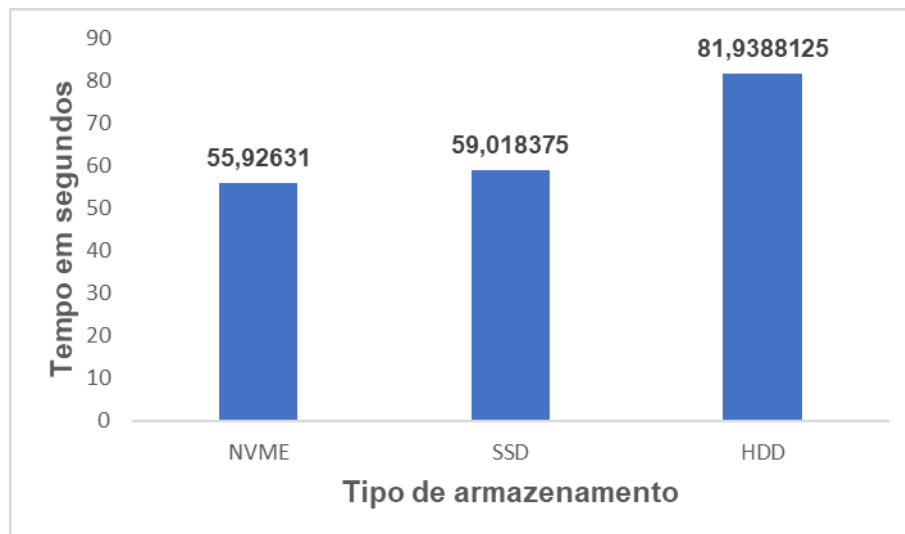


Figura 14 – Tempo médio da consulta com 15.000.000 de linhas (autoria própria, 2023)

Nos testes com 15.000.000 de linhas, a consulta em HDD não apresentou melhorias em relação ao último teste e novamente foi a mais lenta, as consultas no SSD tiveram variações de tempo significativas em suas execuções, e na média das execuções apresentou resultados 28% melhores que o HDD, já as consultas no NVMe não tiveram tantas variações nas execuções de suas consultas e seu desempenho foi o melhor das três tecnologias, deste teste sendo 32% mais rápido que o HDD e 6% mais rápido que o SSD.

Conforme os testes foram aumentando, as linhas consultadas em cada execução, observamos que o HDD que era a melhor tecnologia nos dois primeiros testes foi perdendo desempenho e tendo suas variações de tempo por consulta maiores, fazendo seu tempo médio ficar superior que das outras tecnologias, enquanto o SSD e o NVMe mesmo começando com uma média



## RECIMA21 - REVISTA CIENTÍFICA MULTIDISCIPLINAR ISSN 2675-6218

COMPARATIVO DE DESEMPENHO EM CONSULTAS SQL COM DIFERENTES TIPOS DE  
ARMAZENAMENTO UTILIZANDO BANCO DE DADOS RELACIONAL  
Vinicius Fernandes Bacili, João Henrique Gião Borges, Fabiana Florian

maior e tendo poucas variações em seus tempos, podemos ver que durante os quatro testes, tanto o SSD quanto o NVMe sempre mantiveram uma consistência em seus tempos, tendo pouca variação nos tempos de execução, fazendo assim que seus tempos médios não ficassem elevados. E com isso conseguimos observar que SSD e o NVMe, por serem tecnologias mais novas, mantiveram um padrão confiável no aumento de seus tempos a cada teste, enquanto o HDD por ser uma tecnologia mais antiga não pode oferecer a mesma confiabilidade e não conseguiu manter os tempos abaixo do SSD e NVMe depois do segundo teste. Para não deixar dúvidas, todos os testes foram repetidos várias vezes, também tanto o banco de dados como computador foram reiniciados e testados novamente, mas os resultados foram os mesmos, não obtendo nenhuma mudança dos testes anteriores.

### 5 CONCLUSÃO

Com várias tecnologias de armazenamento de informações disponíveis no mercado, optar pela que melhor atenda às necessidades de cada sistema é um desafio dos profissionais da área de banco de dados. A partir dessa afirmação, notou-se a relevância de estudos comparativos sobre deste tema, perante isso, este trabalho teve como objetivo realizar um comparativo de tempo e desempenho em três tecnologias de armazenamento HDD, SDD e NVMe se utilizando de um banco relacional.

Foi necessário desenvolver e criar o cenário para a aplicação do experimento. Tendo sido aplicadas quatro tipos diferentes de consultas, mais de vinte vezes nas três tecnologias, onde foi possível identificar nas duas primeiras que foram os menores montantes de informações por execução que a tecnologia mais antiga HDD se saiu melhor, contudo, conforme as execuções foram requerendo mais dados, vemos que as tecnologias mais novas SSD e NVMe se saíram melhor do que seu antecessor.

Os resultados encontrados, em parte, conferem com os já existentes em outras pesquisas. Entretanto, não há como indicar a melhor tecnologia de armazenamento, já que para isso, precisamos levar em consideração como e para que fim tal tecnologia vai ser utilizada.

Por fim, reitero aplicar este experimento em outros quantitativos de informações com programas configurados de maneiras distintas à pesquisa atual, e também com novas tecnologias para confrontar os resultados e enriquecer essa importante área de estudos.

### REFERÊNCIAS

DATE, Christopher J. **Introdução a sistemas de banco de dados**. Tradução: Daniel Vieira. 8. ed. Rio de Janeiro: Elsevier Editora Ltda, 2003.

DELL. **NVMe**: a próxima revolução no armazenamento de dados. [S. l.]: Dell, s. d. Disponível em: <https://www.dell.com/pt-br/dt/learn/data-storage/nvme.htm> .



## RECIMA21 - REVISTA CIENTÍFICA MULTIDISCIPLINAR ISSN 2675-6218

COMPARATIVO DE DESEMPENHO EM CONSULTAS SQL COM DIFERENTES TIPOS DE  
ARMAZENAMENTO UTILIZANDO BANCO DE DADOS RELACIONAL  
Vinicius Fernandes Bacili, João Henrique Gião Borges, Fabiana Florian

DEVMEDIA. **Testando o desempenho do HD com banco de dados.** [S. l.]: Devmedia, 2016.  
<https://www.devmedia.com.br/testando-o-desempenho-do-hd-com-banco-de-dados/37429>, 2016.

ELMASRI, Ramez; NAVATHE, Shamkant B. **Sistemas de banco de dados.** 6. ed. São Paulo: Editora Pearson, 2011.

INTEL. **Choosing the right storage:** HDDs vs SSDs. [S. l.]: Intel, 2021. Disponível em:  
<https://www.intel.com.br/content/dam/www/central-libraries/us/en/documents/choosing-hdd-vs-ssd-tech-brief.pdf>.

KINGSTON. **Technology.** [S. l.]: Kingston, 2022. Disponível em:  
<https://www.kingston.com/br/blog/servers-and-data-centers/ssds-changing-data-storage-development-ebook>.

MENDONÇA, Gabriel Tristão do Couto. **Boas práticas de tuning de banco de dados para otimização de um banco de dados relacional.** 2021. TCC (graduação) - Pontifícia Universidade Católica de Goiás - Escola de Ciências Exatas e da Computação, Goiânia, 2021.

ORACLE. 2022. Disponível em: <https://www.oracle.com/br/database/what-is-database/>.

XU, Q.; SIYAMWALA, H.; GHOSH, M.; SURI, T.; AWASTHI, M.; GUZ, Z.; SHAYESTEH, A.; Balakrishnan, V. Performance analysis of nvme ssds and their implication on real world databases. *In: Proceedings of the 8th ACM International Systems and Storage Conference*, SYSTOR 2015, Haifa, Israel, May 26-28, 2015.