

---

## ANÁLISE DE DESEMPENHO DE DIFERENTES SISTEMAS DE GERENCIAMENTO DE BANCO DE DADOS EM DIFERENTES AMBIENTES

Thainá Ferreira SILVA<sup>1</sup>; Hiran Nonato M. FERREIRA<sup>2</sup>

### RESUMO

Os Sistemas de Gerenciamento de Banco de Dados (SGBDs) são responsáveis por organizar e manipular os dados que utilizamos diariamente. Diante disso, são necessários estudos que indiquem melhores formas de escolher esses recursos para as aplicações a serem desenvolvidas. Este trabalho apresenta um estudo sobre o desempenho entre dois SGBDs em diferentes sistemas operacionais, Windows e Linux. Os resultados são satisfatórios e apresentam qual é a melhor solução em determinadas condições.

### INTRODUÇÃO

Os Sistemas que auxiliam na organização e atualização dos dados que utilizamos diariamente são chamados de Sistemas de Gerenciamento de Banco de Dados. Segundo Elsmar e Navathe (2011), Sistema de Gerenciamento de Banco de Dados (SGBD - *Database Management System*) é um sistema de software de uso geral que facilita o processo de definição, construção, manipulação e compartilhamento de banco de dados entre diversos usuários e aplicações. Os

---

<sup>1</sup> Instituto Federal de Educação, Ciência e Tecnologia do Sul de Minas Gerais – Câmpus Passos. Passos/MG - E-mail: thainafferreira1908@gmail.com

<sup>2</sup> Instituto Federal de Educação, Ciência e Tecnologia do Sul de Minas Gerais – Câmpus Passos. Passos/MG. E-mail: hiran.ferreira@ifsuldeminas.edu.br

SGBDs são estruturas que proporcionam uma organização dos dados utilizados nas diversas aplicações que realizamos diariamente. Quando acessamos um determinado software ou sites, estamos fazendo o uso de uma base de dados as quais são conhecidas também como banco de dados, essas bases de dados interagem com os aplicativos para armazenar e organizar as informações sobre os diferentes contextos de utilização.

Atualmente, na construção de aplicações para auxiliar os usuários nas tarefas diárias são utilizados diversos modelos de dados para o armazenamento de informações. Com a grande demanda de dados que se encontram no mundo atual, é cada vez mais necessário que os usuários recebam o resultado de suas requisições o mais rápido possível. Sendo assim, há a necessidade de que se utilize um SGBD que responda o mais breve possível às operações enviadas pelas aplicações.

O modelo de dados relacional é a principal arquitetura utilizada nos sistemas de gerenciamento de banco de dados desde os anos 70. Esse modelo foi apresentado por Edgar F. Codd (Codd, 1970) em 1970, pesquisador da IBM Research, e baseia-se na lógica dos predicados e na teoria dos conjuntos para organização e armazenamento dos dados, utilizando ainda, conceitos de álgebra relacional (Alves, 2011). Este conceito surgiu durante um período no qual às empresas perceberam que estava difícil empregar um número grande de pessoas para fazer trabalhos como armazenar e indexar arquivos. Por este motivo, valia a pena os esforços e investimentos para se pesquisar um meio mais barato e assim ter uma solução mecânica eficiente (Pires et. al 2006).

Neste sentido, este artigo apresenta uma análise de performance de diferentes Sistemas de Gerenciamento de Banco de Dados utilizando o modelo de dados relacional em diferentes Ambientes. Para isso, foi verificado entre os SGBDs de código livre MySQL e PostgreSQL o qual responde mais rapidamente a operações de consultas de dados. Estes testes foram realizados no ambiente Windows e Linux.

Espera-se que esse trabalho auxilie desenvolvedores na escolha de um SGBD que melhor atenda os seus requisitos dependendo, do tipo de operação que irá utilizar, a quantidade de informações que está manipulado e o sistema operacional que será utilizado.

Este artigo está organizado da seguinte forma: Na Seção 2 são apresentados os Materiais e Métodos utilizados para o desenvolvimento da pesquisa. Na Seção 2

são apresentados os resultados e feita uma discussão sobre eles, e por fim, na Seção 5 são apresentadas as conclusões.

## MATERIAL E MÉTODOS

A pesquisa apresentada por esse artigo foi desenvolvida sobre uma abordagem experimental, onde é necessário a execução e aplicação de testes em ambientes reais para validar e verificar a proposta pesquisa.

Para garantir a autenticidade dos dados, todos os testes foram executados em um mesmo *hardware*, alterando apenas as configurações de *software* e tamanho da amostra a ser testada. Os testes foram realizados nos sistemas operacionais Windows e Linux Mint<sup>3</sup>. A justificativa para escolha desses sistemas é pelo fato de serem ambientes bem utilizados pela comunidade, tanto científica quanto usuários finais.

Para realização dos testes, inicialmente foi realizado um estudo para verificar a existência de bases de dados já construídas que seguiu um modelo conciso e simples. Foi possível encontrar na literatura um modelo de dados de alto nível proposto pela Oracle<sup>4</sup> que possui 6 (seis) tabelas de dados com seus relacionamentos e já contendo milhares de registros. A Figura 1 apresenta o Modelo Entidade Relacionamento da base de dados *Employees*, utilizada para os testes.

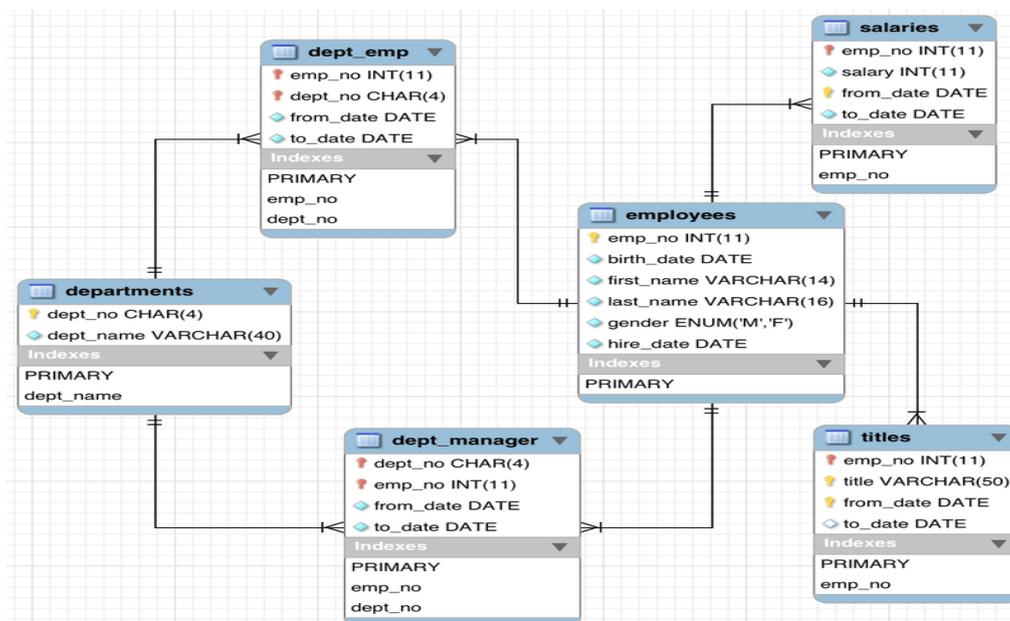


Figura 1: Modelo Entidade Relacionamento do Banco de Dados Employees

<sup>3</sup> <http://www.linuxmint.com/>

<sup>4</sup> <http://www.oracle.com/>

Como a intenção era testar a performance na execução de consultas, foi utilizado dois Bancos de Dados seguindo o mesmo modelo, porém, no primeiro o banco foi implementado utilizando características de integridade referencial dos dados, já no segundo, nenhuma restrição de integridade referencial foi implementada.

Para aumentar o processamento, e por sua vez, garantir um maior rigor nos dados, foram desenvolvidas consultas que utilizassem todas as tabelas disponíveis no banco. Uma consulta que fosse possível relacionar todos os dados disponível nas 6 (seis) tabelas. Uma das consultas utilizava junções (*JOINS*) para relacionar os dados entre as tabelas, isso garantia que os dados seriam processados sem realizar o produto cartesiano nas consultas. Por outro lado, foi desenvolvida uma segunda consulta onde todos os relacionamentos foram implementados a partir de restrições na cláusula *WHERE*, fazendo com os resultados fosse obtidos a partir de operações com produtos cartesianos.

SQL1: Consulta que envolve todas as relações no banco de dados, utilizando conceitos de Junção de dados (*JOIN*).

```
SELECT e.first_name, e.last_name, d.dept_name, s.salary,
       s.from_date, s.to_date, t.title, t.from_date, t.to_date
FROM   employees.e
       inner join dept_emp on e.emp_no=de.emp_no
       inner join departments d on d.dept_no=de.dept_no
       inner join salaries s on e.emp_no=s.emp_no
       inner join titles t on e.emp_no=t.emp_no;
```

SQL2: Consulta que envolve todas as relações no banco de dados, não utilizando conceitos de Junção de dados (*JOIN*) e realizando as junções com a cláusula *WHERE*.

```
SELECT e.first_name, e.last_name, d.dept_name, s.salary,
       s.from_date, s.to_date, t.title, t.from_date, t.to_date
FROM   employees e, dept_emp de, departments d, salaries s, titles titles
WHERE  e.emp_no=de.emp_no and
       de.dept_no=d.dept_no and
       s.emp_no=e.emp_no and
       t.emp_no=e.emp_no;
```

Também com a intenção de normatizar a análise dos dados, as consultas foram divididas em três execuções: a primeira limitada a 1500000, a segunda a 2000000 e a terceira a 2500000.

## **RESULTADOS E DISCUSSÃO**

Buscando uma maior distribuição dos dados a serem analisados, foram construídas duas consultas distribuídas em 3 limites: 1500000, 2000000 e 2500000. Para todas as execuções foram analisadas a média, desvio padrão e variância. Uma análise completa dos resultados pode ser observada abaixo:

Trabalhando com limites de 1500000 tuplas, os dados quando executados no Linux possuíram um melhor desempenho, dentre os SGBDs MySQL e PostgreSQL, o MySQL retornou a resposta da execução em um menor tempo, dentre os bancos de dados analisados, o Employees (consultas sem relacionamentos) possuiu um melhor desempenho considerando a média e o desvio padrão do intervalo de tempo das execuções.

Considerando limites de 2000000 de tuplas, os dados quando executados no Linux possuíram um melhor desempenho, dentre os SGBDs MySQL e PostgreSQL, o MySQL retornou a resposta da execução em um menor tempo, dentre os bancos de dados analisados, o Employees possuiu um melhor desempenho considerando a média, o desvio padrão e também a variância.

Por fim, com o limite de 2500000 tuplas, os dados quando executados no Linux possuíram um melhor desempenho, dentre os SGBDs Mysql e PostgreSql, o Mysql retornou a resposta da execução em um menor tempo, dentre os bancos de dados analisados, o Employees possuiu um melhor desempenho considerando a média do intervalo de tempo das execuções, considerando a variância e o desvio padrão desses dados o Employees2\_sql2 possuiu um melhor desempenho.

Nos limites de 1500000, 2000000 as execuções no SGBD MySQL utilizando o Sistema Operacional Linux apresentou um melhor desempenho com o banco de dados Employees, no entanto, o banco de dados Employees2 apresentou um melhor desempenho com o limite de execução de 2500000.

## CONCLUSÕES

É possível perceber que ao longo dos anos diversas pesquisas têm focado na melhoria na resposta em relação ao tempo dos bancos de dados. O tempo das consultas às bases de dados tem melhorado a cada dia. A partir disso, com esse estudo, foi possível apresentar qual SGBD apresenta melhor resultado em diversos ambientes. Isso facilita e auxilia o desenvolvedor a escolher uma melhor solução, quando se trata de armazenamento de dados.

Dessa forma, a partir da análise da média, desvio padrão e variância dos dados, pôde-se concluir:

- Em todos os limites as execuções no (SO) Linux apresentaram melhor desempenho.
- Na maioria dos casos o Banco de Dados Employees apresentou melhor resultado que o Employees2.
- Analisando a média dos valores, em todos os casos o MySQL apresentou um melhor desempenho.

## REFERÊNCIAS

- Alves, W. P. (2011). Banco de Dados Teoria e Desenvolvimento. Erica, 1th edition.
- Codd, E. F. (1970). A relational model of data for large shared data banks. Commun. ACM, 13(6):377–387.
- Elmasri, R; Navathe, S. B. (2011). Sistemas de Banco de Dados. Pearson Education, 6th edition.
- Pires, C. E. S., Nascimento, R. O., and Salgado, A. C. (2006). Comparativo de desempenho entre bancos de dados de código aberto. ´ Escola Regional de Banco de Dados, pages 21–26.