

UNIVERSIDADE DE BRASÍLIA
Faculdade do Gama

Sistemas de Banco de Dados 2

Tecnologias de Banco de Dados (TI-BD)

Bancos de Dados Orientado a Objeto e Objeto-Relacional

Pedro de Miranda Haick, 180129147

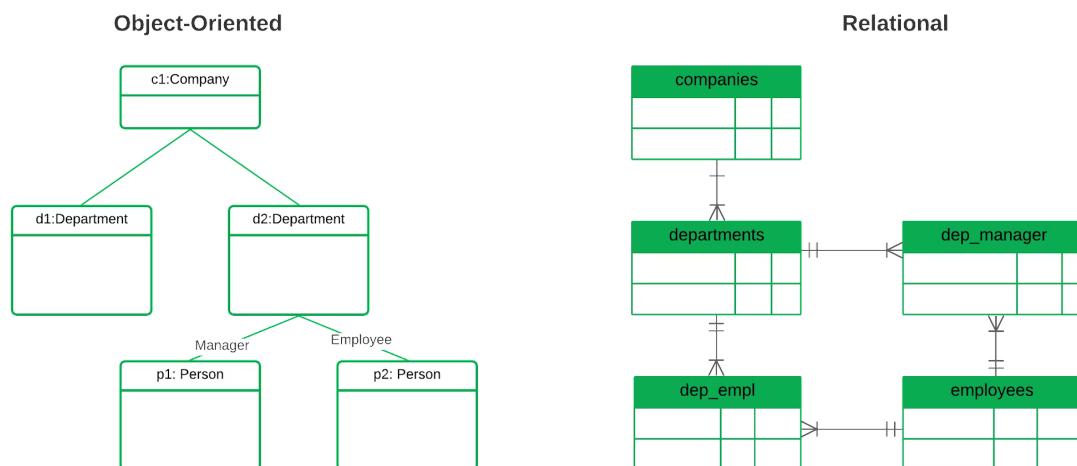
Brasília, DF

2023

Definição da Tecnologia Pesquisada

Bancos de dados orientados a objetos (BDOOs) são um tipo de sistema gerenciador de banco de dados (SGBD) que utiliza conceitos de programação orientada a objetos para armazenar e consultar dados. Em um BDOO, os dados são armazenados como objetos, que podem conter atributos e métodos que definem seu comportamento. Isso permite com que seja feita a modelagem e manipulação de estruturas de dados complexas. Os BDOOs também possuem suporte a encapsulamento, herança e polimorfismo, permitindo um gerenciamento flexível e eficiente dos dados. Em relação aos bancos de dados relacionais, os BDOOs permitem uma representação dos dados mais natural e intuitiva, o que é útil em aplicações que precisam lidar com tipos de dados mais complexos. Os BDOOs também otimizam a performance com a redução de complexidade de operações de consulta e com a otimização de padrões de acesso aos dados.

Bancos de dados objeto-relacionais (BDORs) são um tipo de sistema gerenciador de banco de dados (SGDB) que combina as melhores funcionalidades de um banco de dados relacional e de um banco de dados orientado a objetos. Os BDORs possuem suporte aos conceitos de programação orientada a objetos, como encapsulamento, herança e polimorfismo, além de possuir suporte ao modelo relacional de consulta e armazenamento de dados. Em um BDOR, os dados são representados como objetos, com seus atributos e métodos, mas são armazenados em tabelas, como no modelo relacional. Os BDORs se fazem úteis em aplicações que precisam gerenciar tipos de dados complexos que não são facilmente representados no modelo relacional. Também possuem suporte à linguagem SQL utilizada em bancos de dados relacionais, assim como OQL, utilizada em bancos de dados orientados a objetos.



Representação gráfica de um BDOO e de um BDR.

OBJECT ORIENTED DATABASE VERSUS OBJECT RELATIONAL DATABASE

OBJECT ORIENTED DATABASE	OBJECT RELATIONAL DATABASE
A database that represents information in the form of objects as used in Object Oriented Programming	A database that depends on the Relational model and Object Oriented Database Model
Depends on OOP	Depends on relational model and object oriented database model
Less improved	More improved Visit www.PEDIAA.com

Imagem comparativa entre BDOOs e BDORs.

Objetivo(s) principal(is) da Tecnologia Pesquisada

Tanto os bancos de dados orientados a objetos quanto os bancos de dados objeto-relacionais foram criados com o objetivo de suprir uma limitação dos bancos de dados relacionais tradicionais. Entre essas limitações estão a dificuldade de lidar com tipos mais complexos de dados e a integração com linguagens de programação orientadas a objetos, muito comuns em muitas aplicações atuais.

A necessidade de criação desses SGBDs é descrita por Navathe no livro *Sistemas de Bancos de Dados*:

Sistemas e modelos de dados tradicionais, como relacionais, de rede e hierárquicos, têm tido muito sucesso no desenvolvimento das tecnologias de banco de dados exigidas para muitas aplicações de banco de dados de negócios tradicionais. Porém, eles têm certas deficiências quando aplicações de banco de dados mais complexas precisam ser projetadas e implementadas — por exemplo, bancos de dados para projeto de engenharia e manufatura (CAD/CAM e CIM), experimentos científicos, telecomunicações, sistemas de informações geográficas e multimídia. Essas aplicações mais recentes possuem requisitos e características que diferem daqueles das aplicações de negócios tradicionais, como estruturas mais complexas para objetos armazenados; a necessidade de novos tipos de dados para armazenar imagens, vídeos ou itens de texto grandes; transações de maior duração; e a necessidade de definir operações fora do padrão específicas da aplicação. Os bancos de dados de objeto foram propostos para atender a algumas das necessidades dessas aplicações mais complexas. Um recurso chave dos bancos de dados de objeto é o poder que eles dão ao projetista para especificar tanto a estrutura dos objetos complexos quanto as operações que podem ser aplicadas a esses objetos. Outro motivo para a criação de bancos de dados orientados a objeto é o grande aumento no uso de linguagens de programação orientadas a objeto para o desenvolvimento de aplicações

de software. Os bancos de dados são componentes fundamentais em muitos sistemas de software, e os bancos de dados tradicionais às vezes são difíceis de usar com aplicações de software que são desenvolvidas em uma linguagem de programação orientada a objeto, como C++ ou Java. Os bancos de dados de objeto são projetados de modo que possam ser integrados diretamente — ou transparentemente — ao software desenvolvido usando linguagens de programação orientadas a objeto. Vendedores de SGBD relacional (SGBDR) também reconheceram a necessidade de incorporar recursos que foram propostos para bancos de dados de objeto, e versões mais novas de sistemas relacionais incorporaram muitos desses recursos. Isso levou a sistemas de banco de dados que são caracterizados como objeto-relacional ou SGBDORs. (NAVATHE, 2011, p. 236)

Vantagens da Tecnologia Pesquisada

A principal vantagem de um banco de dados orientado a objetos é prover uma forma mais natural de organização dos dados armazenados quando comparado a um banco relacional. O armazenamento dos dados em tabelas e dos relacionamentos em chaves estrangeiras pode tornar difícil a representação de estruturas de dados complexos em um banco relacional.

Já em um banco de dados orientado a objetos, os dados são armazenados como objetos que contêm outros objetos e também tipos primitivos. Dessa forma, a representação dessas estruturas de dados e o armazenamento e consulta se torna mais apropriado para alguns tipos de aplicações.

Como citado no artigo “What Is an Object-Oriented Database?”, disponível no site do MongoDB:

Com todas as suas associações completas a outros objetos, e porque objetos de dados complexos persistem em um BDOO, a vantagem mais significativa dos BDOO sobre os SGBDR é a habilidade de consultar através desses relacionamentos muito rapidamente.

- Não há “joins” lentos como em um SGBDR. Em vez disso, se tem consultas rápidas com dados complexos.
- Como a estrutura do banco de dados é tão próxima dos objetos

na linguagem de programação, o código é mais simples e mais leve.

(Tradução própria. Original disponível na seção de referências.)

Outra vantagem citada é a proximidade da estrutura do banco de dados com a das linguagens de programação orientadas a objetos, muito utilizadas hoje em dia, o que pode facilitar o processo de modelagem dos dados.

O encapsulamento dos dados é uma outra vantagem dos bancos de dados orientados a objetos. Os objetos podem ocultar seus dados internos e expor apenas os métodos necessários para a interação com outros objetos. Com isso, se tem mais confiança na segurança e confiabilidade do banco de dados, além de permitir uma manutenção mais fácil ao longo do tempo.

Desvantagens da Tecnologia Pesquisada

A principal desvantagem dos BDOOs é a complexidade. Como são baseados em conceitos de programação orientada a objetos, é preciso ter uma boa noção desses conceitos para que se possa modelar e manter um banco de dados orientado a objetos.

Outra desvantagem é a falta de padronização entre BDOOs. Não há uma linguagem padronizada para consulta e manipulação de dados em BDOOs, o que torna difícil lidar com diferentes BDOOs ou migrar dados entre eles. A falta de padronização também impede a integração com outras ferramentas que fariam proveito de uma linguagem padronizada, como é a linguagem SQL.

Por fim, outra desvantagem é a disponibilidade limitada de pessoas com especialidade em BDOOs. Ao contrário de bancos de dados relacionais, que possuem grandes comunidades de desenvolvedores e especialistas e são altamente usados, BDOOs não são tão comuns, o que torna mais difícil de encontrar alguém com especialidade no desenvolvimento e manutenção desse tipo de banco de dados.

Como é citado no artigo “Definition and Overview of ODBMS”, disponível no site Geeks for Geeks:

Uma das principais desvantagens é que eles podem ser mais complexos e difíceis de usar comparados aos bancos de dados relacionais tradicionais. Além disso, BDOOs podem não ser tão amplamente usados

e suportados com bancos de dados relacionais tradicionais, o que pode dificultar a busca por especialidade e suporte. Por fim, algumas aplicações podem não precisar das funcionalidades avançadas providas por um BDOO, e podem ser melhores atendidas por uma solução de banco de dados mais simples.

(Tradução própria. Original disponível na seção de referências)

Exemplo(s) de uso interessante(s) em empresas, organizações, projetos ou instituições dessa tecnologia de Banco de Dados pesquisada

A British Airways utiliza o banco de dados orientado a objetos Versant para seu Sistema de Gerenciamento de Receitas de Origem e Destino. O Sistema de Gerenciamento de Receitas de Origem e Destino é usado para maximizar a receita das redes e alianças mundiais de rotas da British Airways.

A Siemens utiliza a tecnologia GemStone Smalltalk como cache de objetos e backbone para seu sistema CONDIS, que gerencia as infra-estruturas de rede e sistema em tempo real utilizando a tecnologia de objetos. A Siemens é um revendedor de valor agregado do cache de objetos do GemStone Smalltalk. O CONDIS entrou em produção pela primeira vez em 1999 e a Siemens concluiu desde então pelo menos quinze projetos para clientes como a Swiss Railways, City of London, Telecom Argentina e as operadoras austríacas de redes de transporte de energia.

A Starwood Hotels and Resorts Worldwide, Inc. usa o banco de dados orientado a objetos ObjectStore como parte integrante de seu sistema de reservas online. O ObjectStore acessa dinamicamente bancos de dados corporativos existentes e armazena dados como objetos prontos para uso na memória para uso em tempo real.

Exemplos de bancos que utilizam as tecnologias estudadas são: PostgreSQL (Objeto-relacional) e ZOBD (Orientado a Objetos).

Bibliografias Pesquisadas

ELMASRI, R. e NAVATHE, S. B. Sistemas de Banco de Dados, 6a. ed., Pearson, 2011.

WHAT IS AN OBJECT-ORIENTED DATABASE?, disponível em

<https://www.mongodb.com/databases/what-is-an-object-oriented-database>

DEFINITION AND OVERVIEW OF ODBMS, disponível em

<https://www.geeksforgeeks.org/definition-and-overview-of-odbms>