

## O que é um banco de dados relacional (RDBMS)?

Um banco de dados relacional é um tipo de banco de dados que armazena e fornece acesso a pontos de dados relacionados entre si. [Bancos de dados](#) relacionais são baseados no modelo relacional, uma maneira intuitiva e direta de representar dados em tabelas. Em um banco de dados relacional, cada linha na tabela é um registro com uma ID exclusiva chamada chave. As colunas da tabela contêm atributos dos dados e cada registro geralmente tem um valor para cada atributo, facilitando o estabelecimento das relações entre os pontos de dados.

### Um exemplo de banco de dados relacional

Este é um exemplo simples de duas tabelas que uma pequena empresa pode usar para processar pedidos de seus produtos. A primeira tabela é de informações do cliente, de forma que cada registro inclui o nome, o endereço, as informações de envio e faturamento, o número de telefone e outras informações de contato de um cliente. Cada bit de informação (cada atributo) está em sua própria coluna, e o banco de dados atribui uma ID única (uma chave) a cada linha. Na segunda tabela - uma tabela de pedidos do cliente - cada registro inclui o ID do cliente que fez o pedido, o produto solicitado, a quantidade, o tamanho e a cor selecionados e assim por diante - mas não o nome ou as informações de contato.

Essas duas tabelas têm apenas uma coisa em comum: a coluna ID (a chave). Mas por causa dessa coluna comum, o banco de dados relacional pode criar uma relação entre as duas tabelas. Assim, quando o aplicativo de processamento de pedidos da empresa envia um pedido ao banco de dados, o banco de dados pode ir à tabela de pedidos do cliente, obter as informações corretas sobre o pedido do produto e usar o ID do cliente dessa tabela para procurar as informações de faturamento e envio do cliente na tabela de informações do cliente. O warehouse poderá obter o produto correto, o cliente receberá a entrega do pedido oportunamente e a empresa receberá o pagamento.

### Como os bancos de dados relacionais são estruturados

O modelo relacional significa que as estruturas de dados lógicas: tabelas de dados, exibições e índices - são separadas das estruturas de armazenamento físico. Essa separação significa que os administradores de banco de dados podem gerenciar o armazenamento de dados físicos sem afetar o acesso a esses dados como uma estrutura lógica. Por exemplo, a renomeação de um arquivo de banco de dados não renomeia as tabelas armazenadas nele.

A distinção entre lógico e físico também se aplica às operações do banco de dados, que são ações claramente definidas que permitem aos aplicativos [manipular os dados](#) e as estruturas do banco de dados. As operações lógicas permitem que um aplicativo especifique o conteúdo necessário e as operações físicas determinam como esses dados devem ser acessados e, em seguida, executa a tarefa.

Para garantir que os dados sejam sempre precisos e acessíveis, os bancos de dados relacionais seguem determinadas regras de integridade. Por exemplo, uma regra de integridade pode especificar que linhas duplicadas não são permitidas em uma tabela para eliminar o potencial de informações errôneas que entram no banco de dados.

## **O modelo relacional**

Nos primeiros anos de bancos de dados, todos os aplicativos armazenavam dados em sua própria estrutura única. Quando os desenvolvedores queriam criar aplicativos para usar esses dados, precisavam conhecer muito a estrutura de dados específica para encontrar os dados de que precisavam. Essas estruturas de dados eram ineficientes, difíceis de manter e difíceis de otimizar para oferecer um bom desempenho de aplicativos. O modelo de banco de dados relacional foi projetado para resolver o problema de várias estruturas de dados arbitrárias.

O modelo relacional forneceu uma maneira padrão de representar e consultar dados que poderiam ser usados por qualquer aplicativo. Desde o início, os desenvolvedores reconheceram que a principal força do modelo de banco de dados relacional estava no uso de tabelas, que era uma maneira intuitiva, eficiente e flexível de armazenar e acessar informações estruturadas.

Com o tempo, outra força do modelo relacional surgiu quando os desenvolvedores começaram a usar a linguagem de consulta estruturada (SQL) para criar e consultar dados em um banco de dados. Por muitos anos, a SQL tem sido amplamente utilizada como a linguagem para consultas de banco de dados. Com base na álgebra relacional, a SQL fornece uma linguagem matemática internamente consistente que facilita a melhoria do desempenho de todas as consultas ao banco de dados. Em comparação, outras abordagens devem definir consultas individuais.

### **Benefícios do sistema de gerenciamento de bancos de dados relacionais**

O modelo relacional simples, mas eficiente, é usado por organizações de todos os tipos e tamanhos para uma ampla variedade de necessidades de informações. Os bancos de dados relacionais são usados para rastrear inventários, processar transações de comércio eletrônico, gerenciar grandes quantidades de informações essenciais sobre o cliente e muito mais. Um banco de dados relacional pode ser considerado para qualquer necessidade de informações na qual os pontos de dados se relacionam entre si e devem ser gerenciados de maneira segura e consistente, com base em regras. Bancos de dados relacionais existem desde os anos de 1970. Atualmente, as vantagens do modelo relacional continuam a torná-lo o modelo mais amplamente aceito para bancos de dados.

### **Modelo relacional e consistência de dados**

O modelo relacional é o melhor em manter a consistência de dados entre aplicativos e cópias de banco de dados (chamadas de instâncias). Por exemplo, quando um cliente deposita dinheiro em um caixa eletrônico e analisa o saldo da conta em um celular, o cliente espera ver esse depósito refletido imediatamente em um saldo atualizado da conta. Os bancos de dados relacionais se destacam nesse tipo de consistência de dados, garantindo que várias instâncias de um banco de dados tenham os mesmos dados o tempo todo.

É difícil para outros tipos de bancos de dados manter esse nível de consistência oportuna com grandes quantidades de dados. Alguns bancos de dados recentes, como o NoSQL, podem fornecer somente consistência “eventual.” Sob este princípio, quando o banco de dados é dimensionado ou quando vários usuários acessam os mesmos dados ao mesmo tempo, os dados precisam de algum tempo para "serem atualizados". A consistência eventual é aceitável para alguns usos, como manter listagens em um catálogo de produtos, mas para operações comerciais essenciais, como transações de carrinho de compras, o banco de dados relacional ainda é o padrão ouro.

## Compromisso e atomicidade

Os bancos de dados relacionais lidam com regras e políticas de negócios em um nível muito granular, com políticas rígidas sobre compromisso (isto é, fazer uma alteração no banco de dados permanente). Por exemplo, considere um banco de dados de inventário que rastreie três peças que são sempre usadas juntas. Quando uma peça é retirada do inventário, as outras duas também devem ser retiradas. Se uma das três peças não estiver disponível, nenhuma das peças deve ser retirada - todas as três peças devem estar disponíveis antes que o banco de dados confirme. Um banco de dados relacional não confirmará uma peça até que saiba que pode confirmar todas as três. Essa capacidade de compromisso multifacetada é chamada atomicidade. A atomicidade é a chave para manter os dados precisos no banco de dados e garantir que eles estejam em conformidade com regras, regulamentos e políticas da empresa.

## Propriedades ACID e RDBMS

Quatro propriedades cruciais definem transações de banco de dados relacional: atomicidade, consistência, isolamento e durabilidade; em geral, denominadas ACID.

- A **atomicidade** define todos os elementos que compõem uma transação completa do banco de dados.
- A **consistência** define as regras para manter os pontos de dados em um estado correto após uma transação.
- O **isolamento** mantém o efeito de uma transação invisível para outras pessoas até ser confirmada, para evitar confusão.
- A **durabilidade** garante que as alterações de dados se tornem permanentes quando a transação for confirmada.
- 

## Procedimentos armazenados e bancos de dados relacionais

O acesso a dados envolve muitas ações repetitivas. Por exemplo, uma consulta simples para obter informações de uma tabela de dados pode precisar ser repetida centenas ou milhares de vezes para produzir o resultado desejado. Essas funções de acesso a dados requerem algum tipo de código para acessar o banco de dados. Os desenvolvedores de aplicativos não querem criar um novo código para essas funções a cada aplicativo novo. Felizmente, os bancos de dados relacionais permitem procedimentos armazenados, que são blocos de código que podem ser acessados com uma simples chamada de aplicativo. Por exemplo, um único procedimento armazenado pode fornecer identificação de registro consistente para usuários de vários aplicativos. Os procedimentos armazenados também podem ajudar os desenvolvedores a garantir que determinadas funções de dados no aplicativo sejam implementadas de uma maneira específica.

## Bloqueio e simultaneidade de banco de dados

Os conflitos podem surgir em um banco de dados quando vários usuários ou aplicativos tentam alterar os mesmos dados ao mesmo tempo. Técnicas de bloqueio e simultaneidade reduzem o potencial de conflitos, mantendo a integridade dos dados.

O bloqueio impede que outros usuários e aplicativos acessem dados enquanto estão sendo atualizados. Em alguns bancos de dados, o bloqueio se aplica à tabela inteira, o que cria um impacto

negativo no desempenho do aplicativo. Outros bancos de dados, como bancos de dados relacionais da Oracle, aplicam bloqueios no nível de registro, deixando os outros registros dentro da tabela disponíveis, ajudando a garantir um melhor desempenho do aplicativo.

A simultaneidade gerencia a atividade quando vários usuários ou aplicativos fazem consultas ao mesmo tempo no mesmo banco de dados. Esse recurso fornece o acesso correto a usuários e aplicativos de acordo com as políticas definidas para o controle de dados.

### **O que procurar ao selecionar um banco de dados relacional**

O software usado para armazenar, gerenciar, consultar e recuperar dados armazenados em um banco de dados relacional é chamado de sistema de gerenciamento de bancos de dados relacionais (RDBMS). O RDBMS fornece uma interface entre usuários e aplicativos e o banco de dados, além de funções administrativas para gerenciar armazenamento, acesso e desempenho de dados.

Vários fatores podem orientar sua decisão ao escolher entre tipos de banco de dados e produtos de banco de dados relacional. O RDBMS escolhido dependerá das suas necessidades de negócios.

Pergunte a você mesmo as seguintes questões:

- Quais são os nossos requisitos de precisão de dados? O armazenamento e a precisão de dados dependem da lógica de negócios? Nossos dados têm requisitos rigorosos de precisão (por exemplo, dados financeiros e relatórios do governo)?
- Precisamos de escalabilidade? Qual é a escala dos dados a serem gerenciados e qual é o crescimento previsto? O modelo de banco de dados precisará oferecer suporte a cópias de banco de dados espelhadas (como instâncias separadas) para escalabilidade? Em caso afirmativo, é possível manter a consistência de dados nessas instâncias?
- O quanto a simultaneidade é importante? Vários usuários e aplicativos precisarão de acesso simultâneo a dados? O software de banco de dados oferece suporte à simultaneidade ao mesmo tempo que protege os dados?
- Quais são as nossas necessidades de desempenho e confiabilidade? Precisamos de um produto de alto desempenho e alta confiabilidade? Quais são os requisitos para o desempenho de resposta à consulta? Quais são os compromissos do fornecedor com os acordos de nível de serviço (SLAs) ou tempo de inatividade não planejado?

### **O banco de dados relacional do futuro: o banco de dados autônomo**

Ao longo dos anos, os bancos de dados relacionais ficaram melhores, mais rápidos, mais fortes e mais fáceis de se trabalhar. Mas também se tornaram mais complexos e administrar o banco de dados tem sido um job de tempo integral. Em vez de usarem seus conhecimentos para se concentrar no desenvolvimento de aplicativos inovadores que agreguem valor ao negócio, os desenvolvedores passaram a maior parte do tempo na atividade de gerenciamento necessária para otimizar o desempenho do banco de dados.

Atualmente, a [tecnologia autônoma](#) está aproveitando os pontos fortes do modelo relacional, da tecnologia de [banco de dados em nuvem](#) e de [machine learning](#) para fornecer um novo tipo de banco de dados relacional. O banco de dados independente (também conhecido como banco de dados autônomo) mantém a eficiência e as vantagens do modelo relacional, mas usa inteligência artificial (IA), o machine learning e a automação para monitorar e melhorar o desempenho das consultas e as tarefas de gerenciamento. Por exemplo, para melhorar o desempenho da consulta, o banco de dados independente pode criar hipóteses e testar índices para tornar as consultas mais

rápidas e enviar as melhores para a produção, tudo por conta própria. O banco de dados independente faz essas melhorias continuamente, sem a necessidade de envolvimento humano. A tecnologia [Autonomous](#) libera os desenvolvedores das tarefas comuns do gerenciamento do banco de dados. Por exemplo, eles não precisam mais determinar os requisitos de infraestrutura com antecedência. Em vez disso, com um banco de dados independente, eles podem adicionar armazenamento e processar recursos conforme necessário para oferecer suporte ao crescimento do banco de dados. Com apenas algumas etapas, os desenvolvedores podem facilmente criar um banco de dados relacional autônomo, acelerando o tempo para o desenvolvimento de aplicativos.