

SENAI SUÍÇO BRASILEIRO

NORMALIZAÇÃO

ANNA GABRIELA MONTEIRO DA SILVA

DOUGLAS HENRIQUE DE LIMA GOMES

GABRIELA LIMA DA LUZ

GEOVANA MOREIRA SANTOS

São Paulo - SP

2023

SUMÁRIO

INTRODUÇÃO.....	2
NORMALIZAÇÃO DE DADOS EM BANCOS DE DADOS RELACIONAIS.....	3
PRIMEIRA FORMA NORMAL (1FN)	3
SEGUNDA FORMA NORMAL (2FN):	5
TERCEIRA FORMA NORMAL (3FN):.....	6
APLICAÇÕES E DESAFIOS.....	8
CASOS DE USO E APLICAÇÕES	8
DESAFIOS	9
TÉCNICAS DE DESNORMALIZAÇÃO	9
CONSIDERAÇÕES FINAIS	11
REFERÊNCIAS	12

INTRODUÇÃO

A normalização é focada na prevenção de problemas com repetição e atualização de dados, assim como o cuidado com a integridade dos dados. Tal procedimento é feito a partir da identificação de uma anomalia em uma relação, decompondo-as em relações melhor estruturadas, o que resultou em um processo flexível e menos custoso para o armazenamento e processamento de dados. Por meio dela que bancos com muita movimentação garantem sua integridade após remoção, inserção e alteração dos dados.

Dentre os benefícios da normalização, pode-se citar: a redução da redundância do espaço de armazenamento e evitando inconsistências; maior integridade, garantindo dados consistentes e corretos; facilidade de manutenção, pois alterações são aplicadas em menos lugares; Maior flexibilidade para lidar com alterações nos requisitos e escalar conforme a demanda; e maior eficiência nas consultas, uma vez que as informações estão melhor organizadas.

NORMALIZAÇÃO DE DADOS EM BANCOS DE DADOS RELACIONAIS

Normalização é o processo de organização de dados em um banco de dados. Isso inclui a criação de tabelas e o estabelecimento de relações entre essas tabelas de acordo com as regras projetadas para proteger os dados e tornar o banco de dados mais flexível, eliminando a redundância e a dependência inconsistente. Há conjuntos de regras para determinar com qual forma normal o banco é compatível. O processo de normalização conta com 6 formas:

- 1º Forma Normal
- 2º Forma Normal
- 3º Forma Normal
- FNBC (Forma normal de Boyce e Codd)
- 4º Forma Normal
- 5º Forma Normal

Neste trabalho será abordado apenas 3 formas normais.

PRIMEIRA FORMA NORMAL (1FN)

Neste nível, as tabelas devem possuir valores atômicos, ou seja, cada célula da tabela deve conter apenas um valor simples, não sendo permitido valores compostos ou repetidos. Podemos dizer que uma tabela se encontra na Primeira Forma Normal se:

- Possui chave primária;
- Não possui grupos repetitivos;
- Todos os seus atributos são atômicos, ou seja, não precisa ser decomposto.

Como exemplo (informações fictícias) tem-se uma tabela com informações sobre duas pessoas:

Código	Nome	Localização	Telefone
1	José	Curitiba	(44) 91234-5678
		Paraná	(44) 94834-8348
2	Arturo	Recife	(81) 91234-8765
		Pernambuco	(81) 91328-7811

A princípio, a chave primária é o atributo de código. Entre os atributos estão os valores associados a chave, então é preciso fragmentar estes valores, desta forma é necessário criar uma nova tabela.

Código	Nome	Cidade	Estado
1	José	Curitiba	Paraná
2	Arturo	Recife	Pernambuco

Agora são criados os atributos cidade e estado, porque estas informações estavam em um único atributo, sendo que é mais útil ter essas informações separadas, para filtrar os dados por exemplo.

Código	Telefone
1	(44) 91234-5678

1	(44) 94834-8348
2	(81) 91234-8765
2	(81) 91328-7811

Esta nova tabela foi criada para poder relacionar telefones com o atributo código, que na tabela principal é a chave primária, sendo definida como chave estrangeira. Todos os dados foram definidos individualmente, ainda assim relacionados.

SEGUNDA FORMA NORMAL (2FN):

Neste nível, a tabela deve estar na 1FN e, além disso, todos os atributos não chave devem depender completamente da chave primária, evitando dependências parciais. Para chegar à segunda forma normal deve-se verificar se a chave primária dessa tabela é composta ou simples. Se for simples, já se encontra na segunda forma normal. Se for composta, deve-se verificar se todos os atributos da relação dependem de todos os atributos que compõem a chave primária. Este exemplo é similar a tabela anterior:

Código	Nome	Código Voo	Origem	Destino
1	José	101	Santiago	São Paulo
2	Arturo	102	Bogotá	Buenos Aires

Os campos de origem e destino não têm relação direta com o campo de código, mas têm uma relação direta com o código de voo, já que são informações relacionadas a uma viagem aérea, por exemplo. Assim, pode-se mover essas informações a uma nova tabela sem que os dados percam as relações originais.

Código Voo	Origem	Destino
101	Santiago	São Paulo
102	Bogotá	Buenos Aires

Por isso, a tabela original elimina os dados que não são necessários nela, porém eles seguem relacionados em uma tabela secundária.

Código	Nome	Código Voo
1	José	101
2	Arturo	102

TERCEIRA FORMA NORMAL (3FN):

Neste nível, a tabela deve estar na 2FN e, além disso, os atributos não chave devem ser dependentes somente da chave primária, eliminando dependências transitivas. Para chegar a terceira forma normal verifique os campos que não são chave primária. Se algum desses campos não chave possuir dependência com outro campo não chave, então essa tabela não se encontra na terceira forma normal. Agora um exemplo referente a modelos de carros:

Placa	Modelo	Ano	Código Fabricação	Nome Fábrica
123X	Modelo 1	2016	1	A
456Y	Modelo 2	2012	2	B
789Z	Modelo 3	2012	3	C

Existe uma dependência entre o atributo 'nome de fábrica' e 'ano' com o 'código de fábrica', entretanto, estes atributos não dependem da chave principal da tabela que é 'placa'.

Código Fabricação Nome Fábrica Ano

1	A	2016
2	B	2012
3	C	2012

Neste caso, é necessário criar uma nova tabela para relacionar o nome da fábrica com seu código, e também remover as relações de dependência entre atributos que não são chaves da tabela original.

Placa Modelo Código Fabricação

123X	Modelo 1	1
------	----------	---

456Y	Modelo 2	2
789Z	Modelo 3	3

NORMALIZAÇÃO VERTICAL

A normalização vertical envolve a decomposição de uma tabela em duas ou mais tabelas para reduzir a redundância de dados. Isso é feito movendo atributos relacionados a um conjunto separado de tabelas. Por exemplo, se você tem uma tabela de funcionários que inclui tanto informações pessoais quanto de salário, pode verticalmente dividir essas informações em duas tabelas separadas: uma para os detalhes pessoais e outra para os detalhes salariais. Isso reduz a repetição de dados, pois os dados pessoais não precisam ser repetidos para cada entrada de salário.

NORMALIZAÇÃO HORIZONTAL

A normalização horizontal está relacionada com a separação de dados em várias tabelas para reduzir a duplicação de linhas. Ela é realizada através da criação de novas tabelas a partir de conjuntos de dados repetidos. Por exemplo, se você tem uma tabela de pedidos de clientes com informações do cliente repetidas para cada pedido, você pode criar uma tabela separada apenas para os detalhes do cliente e referenciar essa tabela na tabela de pedidos. Isso evita a duplicação de dados do cliente em cada registro de pedido.

Ambos os tipos de normalização têm como objetivo principal melhorar a integridade, reduzir a redundância e minimizar as anomalias de atualização e exclusão nos dados. No entanto, é importante encontrar um equilíbrio, pois a normalização excessiva pode levar a um aumento na complexidade das consultas e dificultar a recuperação eficiente dos dados.

A escolha de como normalizar os dados depende das necessidades específicas do sistema, do tipo de consultas que serão executadas com mais frequência e das operações que serão realizadas sobre esses dados.

A normalização vertical e horizontal são abordagens distintas para organizar e estruturar dados em um banco de dados relacional. Aqui estão algumas diferenças e situações de aplicação para cada uma:

Normalização Vertical:

Diferenças:

- **Decomposição dos Dados:** Separa os atributos de uma tabela em diferentes tabelas.
- **Redução da Redundância Vertical:** Evita a duplicação de colunas.
- **Criação de Tabelas Relacionadas:** Relaciona as novas tabelas através de chaves estrangeiras.

Situações de Aplicação:

- **Redução de Redundância de Colunas:** Quando uma tabela tem muitos atributos que são usados apenas para um subconjunto dos registros.
- **Estruturação para Consultas Específicas:** Quando certos atributos são necessários em operações específicas, mas não em todas.

Normalização Horizontal:

Diferenças:

- **Divisão de Linhas Repetidas:** Separa linhas repetidas em tabelas distintas.
- **Redução da Redundância Horizontal:** Evita a duplicação de linhas de dados.
- **Relacionamento entre Tabelas:** Utiliza chaves primárias e estrangeiras para manter a integridade referencial.

Situações de Aplicação:

- Evitar Duplicação de Registros: Quando há repetição de informações em várias linhas, por exemplo, dados do cliente sendo repetidos em múltiplos registros de pedidos.
- Melhoria da Eficiência de Armazenamento: Em casos onde a redução do espaço ocupado pela duplicação de informações é crítica.

Comparação:

- Redundância de Dados: A normalização vertical reduz a redundância dentro de uma mesma linha (colunas), enquanto a horizontal lida com a redundância entre diferentes linhas (registros).
- Estrutura Resultante: A vertical resulta em mais tabelas com menos colunas, enquanto a horizontal resulta em mais tabelas com menos linhas.
- Impacto nas Consultas: A vertical pode simplificar algumas consultas ao segmentar os dados por funcionalidade, mas pode exigir mais joins em consultas complexas. A horizontal pode simplificar joins, mas pode exigir consultas mais complexas para reunir informações.
- Desempenho: Depende da estrutura dos dados e do uso do banco de dados. Em geral, a normalização vertical pode ser mais eficiente para atualizações e inserções, enquanto a horizontal pode ser melhor para leituras eficientes.

A escolha entre normalização vertical e horizontal depende das necessidades específicas do sistema, das consultas frequentes realizadas e das operações realizadas sobre os dados. Geralmente, uma combinação bem planejada de ambas pode oferecer um equilíbrio entre redução de redundância e desempenho em consultas.

APLICAÇÕES E DESAFIOS

CASOS DE USO E APLICAÇÕES

Os princípios de normalização se aplicam a vários casos de uso e aplicativos do mundo real. Aqui estão alguns exemplos práticos de sua aplicação:

- **Aplicações de comércio eletrônico:** Em uma aplicação de comércio eletrônico, várias entidades como clientes, pedidos, produtos e fabricantes podem estar envolvidas. A normalização permite o armazenamento eficiente de dados relacionados sem redundância, garantindo a integridade dos dados em diversas tabelas, como pedidos, itens de pedidos e tabelas de inventário de produtos.
- **Sistemas de gerenciamento de recursos humanos (HRMS):** os aplicativos HRMS normalmente gerenciam registros de funcionários, informações de folha de pagamento, dados departamentais e muito mais. A normalização ajuda a manter a precisão dos dados, evitando a duplicação de informações dos funcionários e garantindo que cada atributo seja armazenado em seu devido lugar.
- **Sistemas de gerenciamento clínico:** Os sistemas de gerenciamento clínico lidam com registros de pacientes, consultas, detalhes da equipe médica e outros dados relacionados. A normalização adequada permite fácil manutenção dos registros dos pacientes e auxilia na recuperação precisa de dados relacionados a consultas, prescrições e exames laboratoriais.
- **Aplicativos de redes sociais:** os aplicativos de redes sociais envolvem relacionamentos complexos entre usuários, postagens, comentários e diversas formas de conteúdo gerado pelo usuário. A normalização é crucial para garantir a consistência dos dados, permitindo o gerenciamento eficiente das conexões, conteúdos e interações dos usuários na plataforma.

Compreender a normalização em bancos de dados relacionais ajudará a melhorar a modelagem de dados e a projetar aplicativos mais eficientes, consistentes e confiáveis. Equilibrar os níveis de normalização e o desempenho é fundamental para construir um esquema de banco de dados eficaz e adaptado ao seu caso de uso específico.

DESAFIOS

Embora a normalização de banco de dados forneça uma maneira estruturada e organizada de armazenar dados, ela traz seu próprio conjunto de compensações.

Uma das principais compensações a considerar é o equilíbrio entre desempenho e integridade dos dados.

À medida que o nível de normalização aumenta, as tabelas numa base de dados relacional tornam-se mais fragmentadas e requerem mais junções para aceder aos dados relacionados. Isso pode levar ao aumento da complexidade e afetar o desempenho das consultas, especialmente ao lidar com conjuntos de dados complexos ou grandes. Para lidar com esta problemática é necessário aplicar técnicas de desnormalização.

Por outro lado, bancos de dados normalizados oferecem maior integridade de dados. A redução da redundância através da divisão de dados em várias tabelas evita anomalias e inconsistências associadas às operações de atualização, inserção e exclusão. Como resultado, manter a consistência e a precisão dos dados torna-se mais fácil.

Encontrar o nível ideal de normalização pode exigir encontrar um equilíbrio entre desempenho e integridade de dados com base em requisitos específicos do aplicativo. A análise cuidadosa dos padrões de consulta, uso de dados e cargas esperadas pode ajudar a determinar o nível de normalização correto para o esquema do seu banco de dados.

TÉCNICAS DE DESNORMALIZAÇÃO

A desnormalização é um processo que envolve a introdução intencional de algum nível de redundância no banco de dados para melhorar o desempenho da consulta, ao mesmo tempo em que atinge um equilíbrio entre integridade e eficiência dos dados. É útil quando as compensações de desempenho da normalização se tornam aparentes e há necessidade de otimizar ainda mais a estrutura do banco de dados. Algumas técnicas comuns de desnormalização incluem:

- **Adicionar campos calculados:** armazene valores calculados ou agregados em uma tabela para evitar cálculos complexos ou junções durante a execução da consulta, agilizando a recuperação de dados.
- **Mesclando tabelas:** combine tabelas relacionadas quando o número de junções de consulta afetar negativamente o desempenho. Isso reduz a complexidade de acesso aos dados relacionados.
- **Replicação de dados ou colunas:** duplique dados em diversas tabelas para reduzir o número de junções necessárias para determinadas consultas. Isso

pode ajudar a melhorar o desempenho da consulta às custas de alguma redundância e possíveis problemas de consistência de dados.

- **Usando indexação:** crie índices em colunas comumente usadas para acelerar a execução da consulta. Embora não seja estritamente uma técnica de desnormalização, a indexação pode ajudar a aliviar alguns problemas de desempenho associados a esquemas altamente normalizados.

É importante analisar cuidadosamente o impacto das técnicas de desnormalização na integridade dos dados e pesar os benefícios em relação aos riscos potenciais. A desnormalização deve ser usada criteriosamente, pois pode introduzir complexidade adicional, aumentar os requisitos de armazenamento e afetar a consistência dos dados.

CONSIDERAÇÕES FINAIS

A normalização de dados é um processo importante para garantir a integridade e a consistência dos dados em um banco de dados relacional. Ao dividir os dados em tabelas relacionadas, a normalização ajuda a evitar anomalias de atualização, inserção e exclusão. No entanto, a normalização também pode levar a um aumento da complexidade das consultas, o que pode afetar o desempenho do sistema.

É importante encontrar um equilíbrio entre a normalização e o desempenho do sistema. Em geral, um esquema de banco de dados altamente normalizado será mais seguro e consistente, mas também pode ser menos eficiente em termos de desempenho. Um esquema de banco de dados menos normalizado pode ser mais eficiente em termos de desempenho, mas pode ser mais propenso a anomalias de dados.

Encontrar o equilíbrio certo entre normalização e eficiência operacional requer levar em consideração o contexto específico do sistema: analisando os padrões de consulta, quais são mais frequentemente executadas no sistema e quais dados são necessários para essas consultas; considerando o uso de dados, quais são mais atualizados, inseridos ou excluídos e pesquisando as cargas esperadas, qual o volume de dados esperado no sistema. Com base nessa análise, é possível identificar as tabelas ou atributos que podem ser desnormalizados para melhorar o desempenho sem comprometer a integridade dos dados.

A normalização de dados desempenha um papel crucial no desenvolvimento de sistemas eficientes e precisos, no entanto, é importante encontrar um equilíbrio entre a normalização e o desempenho do sistema. Ao analisar os requisitos específicos do aplicativo, é possível identificar as tabelas ou atributos que podem ser desnormalizados para melhorar o desempenho sem comprometer a integridade dos dados.

Algumas direções para pesquisas futuras relacionadas à normalização de dados incluem:

- Desenvolvimento de técnicas de normalização que possam melhorar o desempenho das consultas sem sacrificar a integridade dos dados.
- Desenvolvimento de técnicas de desnormalização que possam reduzir o impacto da redundância nos dados.
- Estudo da relação entre normalização e desempenho em diferentes cenários de aplicação.

Essas pesquisas podem ajudar a desenvolver novas técnicas de normalização que possam atender melhor às necessidades dos aplicativos modernos e a desenvolver técnicas mais eficientes para equilibrar a normalização com a eficiência operacional.

REFERÊNCIAS

Por **Diego Machado**. “**Normalização em Bancos de Dados**”, Medium. Acesso em: 30 de novembro de 2023. Disponível em: <https://medium.com/@diegobmachado/normaliza%C3%A7%C3%A3o-em-banco-de-dados-5647cdf84a12>

Por **Brenda Souza, Luis Puig**. “**Normalização em Banco de Dados - Estrutura**”, Alura. Acesso em: 30 de novembro de 2023. Disponível em: <https://www.alura.com.br/artigos/normalizacao-banco-de-dados-estrutura>

“**Normalizando um banco de dados por meio das 3 principais formas**”, SpaceProgrammer. Acesso em: 30 de novembro de 2023. Disponível em: <https://spaceprogrammer.com/bd/normalizando-um-banco-de-dados-por-meio-das-3-principais-formas/#:~:text=Normaliza%C3%A7%C3%A3o%20%C3%A9%20uma%20ferramenta%20usada,inser%C3%A7%C3%A3o%20e%20altera%C3%A7%C3%A3o%20dos%20dados.>

Por **Mauro**. “**Normalização de Bancos de Dados: Explicação e Benefícios**”, BlogDoSQL. Acesso em: 30 de novembro de 2023. Disponível em: <https://blogdosql.com.br/normalizacao-de-bancos-de-dados-explicacao-e-beneficios/#:~:text=Benef%C3%ADcios%20da%20Normaliza%C3%A7%C3%A3o&text=Redu%C3%A7%C3%A3o%20de%20Redund%C3%A2ncia%3A%20A%20normaliza%C3%A7%C3%A3o,dados%20estejam%20consistentes%20e%20corretos.>

Por **Lu Chen, Simonx Xu, Venusmi, Dario Woitasen, Casar com Qiu**. “**Descrição Das Noções Básicas De Normalização Do Banco De Dados**”, Microsoft 365. Acesso em: 02 de dezembro de 2023. Disponível em: <https://learn.microsoft.com/pt-br/office/troubleshoot/access/database-normalization-description>.

Por **Lya laurent**. “**Normalização Em Bancos De Dados Relacionais: Um Mergulho Profundo**”, AppMaster. Acesso em: 05 de dezembro de 2023. Disponível em: <https://appmaster.io/pt/blog/normalizacao-em-bancos-de-dados-relacionais>