

CCF 221 - Banco de Dados Portfólio I

Artur Papa - 3886

December 2021

Sumário

1	Conceitos gerais	3
1.1	Banco de Dados	3
1.2	Minimundo	3
1.3	SGBD	3
1.4	Data Warehouse	3
1.5	Catálogo	3
1.6	Metadados	4
1.7	Arquitetura cliente/servidor	4
1.8	Arquitetura de três níveis	4
2	Modelo ER	4
2.1	Entidade	4
2.2	Relacionamento	4
2.3	Atributo	5
2.4	Diagrama ER	5
2.5	Relacionamento ternário	5
2.6	Entidade Fraca	6
2.7	Entidade Forte	7
2.8	Verificando o esquema	7
2.9	Estratégias de modelagem	7
2.10	Resumo da notação ER	7
3	Modelo ER alternativo	8
4	Modelo ER estendido	8
4.1	Superclasse e subclasse	8
4.2	Generalização	9
4.3	Especialização	10
4.4	Hierarquia	10
4.5	Grade de especialização	10
4.6	Herança múltipla	10
4.7	União ou categoria	10

1 Conceitos gerais

1.1 Banco de Dados

É uma coleção organizada de informações estruturadas geralmente armazenadas eletronicamente em um sistema de computador, são modelados em linhas e colunas em uma série de tabelas para tornar o processamento de dados mais eficientes. Normalmente são representados por um cilindro, como se fossem uma pilha.

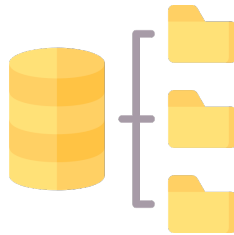


Figura 1: Pilha representando um banco de dados

1.2 Minimundo

Sempre que referimos a algum banco de dados, estamos localizados em algum contexto específico, assim, este universo é chamado de minimundo, onde tratamos de todos os dados relacionados a tal assunto.

1.3 SGBD

Um Sistema de Gerenciamento de Banco de Dados (Data Base Management System) é como o próprio nome diz, um sistema de apoio à gestão de dados, sendo permitido através de softwares realizar uma série de funções como criar relações entre tabelas, eliminar e copiar arquivos, alterar a estrutura de campos, efetuar consultas em tabelas, criar e gerir usuários e importar e exportar dados. Além disso, temos tipos específicos de SGBD's que servem para situações diferentes de tratamento de dados, como os SGBD's **relacionais**, **não relacionais**, **hierárquico**, **de rede** e **orientado a objetos**.

1.4 Data Warehouse

Se trata de um repositório central de informações que podem ser analisadas para tomar decisões mais adequadas, em que os dados fluem de sistemas transacionais, bancos de dados relacionais e de outras fontes para o data warehouse. Geralmente são utilizados para testes com dados previamente criados e com o processamento analítico online.

1.5 Catálogo

É o lugar onde todos os vários esquemas e todos os mapeamentos correspondentes são mantidos, ou seja, o catálogo contém informações detalhadas (metadados) sobre os vários objetos de interesse do próprio sistema.

1.6 Metadados

Tem a função de facilitar o entendimento dos relacionamentos e evidenciar a utilidade das informações dos dados. Assim, temos como exemplo número de telefones, endereços de email e os nomes das pessoas que usam serviços.

1.7 Arquitetura cliente/servidor

Para a arquitetura básica de um SGBD relacional nós temos dois módulos a serem discutidos, primeiro o módulo servidor em que seriam armazenados as relações do banco de dados, junto com os outros arquivos, além do módulo cliente em que este seria a resposta ao serviço feito no primeiro módulo.

1.8 Arquitetura de três níveis

Para a arquitetura de três níveis nós teremos o modelo conceitual centralizado, sendo este influenciado pelas visões externas e embaixo o esquema interno, podendo ser ilustrado da seguinte forma:

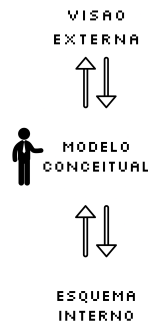


Figura 2: Arquitetura de três níveis

2 Modelo ER

Primeiramente, para introduzir o Modelo Entidade-Relacionamento é de extrema necessidade falar sobre os principais conceitos básicos:

2.1 Entidade

Representação de uma entidade, sujeito ou objeto dentro do minimundo escolhido pelo projetista do banco de dados, como uma empresa, pessoa ou até veículos dependendo da situação. São representados por um retângulo nos diagramas ER.

2.2 Relacionamento

Ligado a duas entidades, representa a relação entre dois sujeitos, são representados no diagrama com um losango, geralmente definidos por um verbo que se adequa a cada entidade, por exemplo, se temos um mecânico que conserta um veículo, no diagrama escreveríamos o verbo "consertar", sendo que um mecânico conserta um automóvel e tal automóvel é consertado por um mecânico. Além disso, temos que nos atentar para a cardinalidade entre duas entidades, caso a relação entre as duas seja parcial, colocamos apenas uma linha ligada ao losango, caso seja total, duas linhas são desenhadas na ilustração do relacionamento.

2.3 Atributo

Geralmente são representados por um círculo e são características particulares de uma entidade, tendo em vista que podem ter um valor único, multivalorado, derivado ou composto.

2.4 Diagrama ER

Por ter sido citado acima, temos a seguir um exemplo de um diagrama ER de uma oficina mecânica:

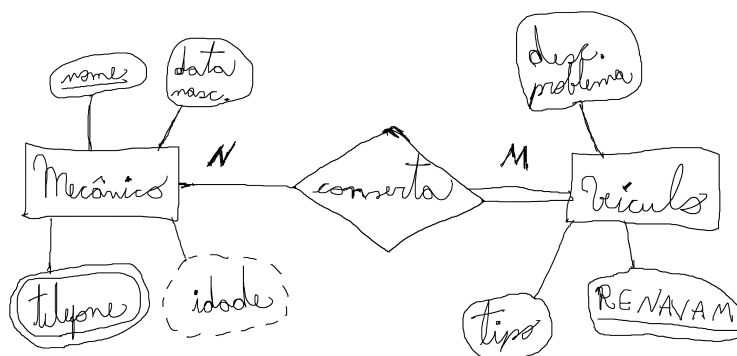


Figura 3: Diagrama ER de uma oficina

Como pode se notar temos duas entidades no diagrama, *mecânico* e *veículo*, ligados a eles temos seus respectivos atributos, analisando a entidade *mecânico* vemos que ele tem um atributo chave(nome), multivalorado(telefone) e derivado(idade), já para o *veículo* nós temos um atributo chave(RENAVAM) e dois simples. Além disso, nota-se o relacionamento entre duas entidades, sendo que o *veículo* possui relação total, já que este será sempre consertado por pelo menos um *mecânico*, por outro lado o funcionário possui relação parcial, visto que nem sempre este será responsável por consertar um *veículo*, e, por fim, vemos que a cardinalidade será de **N** para **M**, tendo em conta que um *veículo* é consertado por pelo menos um ou mais *mecânicos* e um *mecânico* fica responsável por consertar um ou mais *veículos* ou nenhum.

2.5 Relacionamento ternário

Além dos relacionamentos binários, ainda existem os ternários, aqueles em que três entidades estão ligadas há apenas um relacionamento, entretanto é preferível transformar esse tipo de relação em pequenos relacionamentos binários.

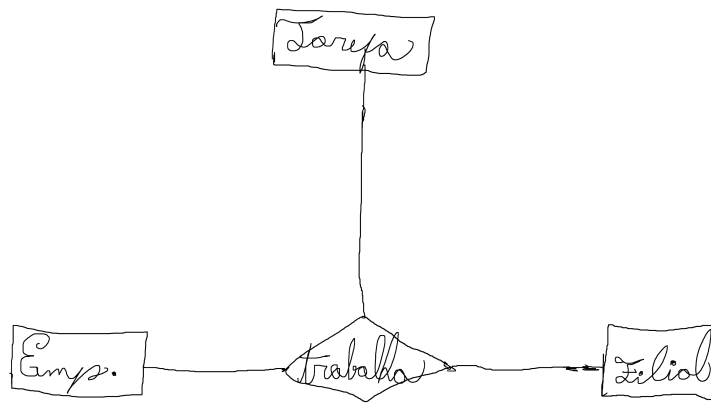


Figura 4: Exemplo relacionamento ternário

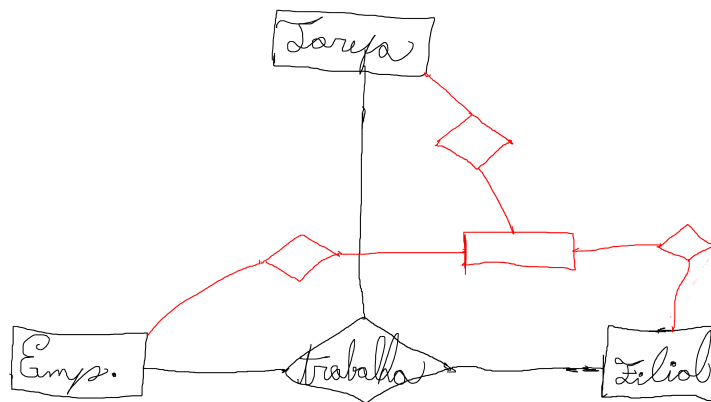


Figura 5: Relacionamento ternário mudado para relacionamentos binários

2.6 Entidade Fraca

Entidade que não tem chave própria, em que as instâncias são identificadas através do relacionamento com entidades de outro tipo, chamado de dono ou identificador junto com uma chave parcial, como por exemplo um dependente de uma empresa

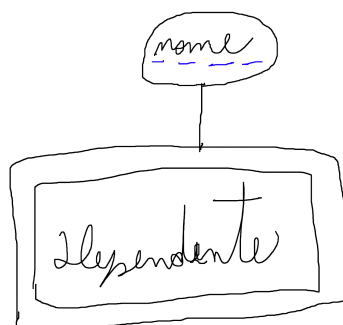


Figura 6: Entidade fraca

2.7 Entidade Forte

Enquanto a entidade fraca não possui uma chave própria, temos que a entidade forte segue um caminho contrário, haja vista que esta possui uma chave única, sendo um atributo independente para determinado sujeito ou objeto. Outrossim, podemos ver no exemplo da oficina que a chave única do mecânico seria o seu nome e a do veículo seria o seu RENAVAM.

2.8 Verificando o esquema

Para saber se o diagrama projetado está correto há alguns passos para serem revisados, como saber se o esquema está completo, se está livre de redundância, se está refletindo o aspecto temporal, se as entidades estão isoladas para que não ocorra nenhum erro de interpretação futuramente e, por fim, verificar se não há entidades sem atributos, caso houver, refazer a análise dos dados.

2.9 Estratégias de modelagem

Primeiramente temos a estratégia **TOP-DOWN** que parte dos conceitos mais abstratos para os mais detalhados e é feita em três passos. Primeiro é feita uma modelagem superficial para enumerar as entidades, criar os relacionamentos e determinar os atributos e identificadores, em seguida faz-se uma modelagem detalhada para determinar as cardinalidades e definir as restrições que não são representadas no DER e por último é feita a validação do modelo.

Do mesmo modo, temos a estratégia **INSIDE-OUT** que parte dos conceitos considerados mais importantes e adiciona gradativamente os conceitos periféricos. Assim como na primeira estratégia esta segue três passos relevantes para sua implementação onde primeiro se procura a entidade mais importante, a partir dela procuram-se os relacionamentos, atributos, por fim, busca-se uma nova entidade importante e repete os passos.

2.10 Resumo da notação ER

Por fim, segue um resumo dos principais símbolos do diagrama entidade relacionamento:

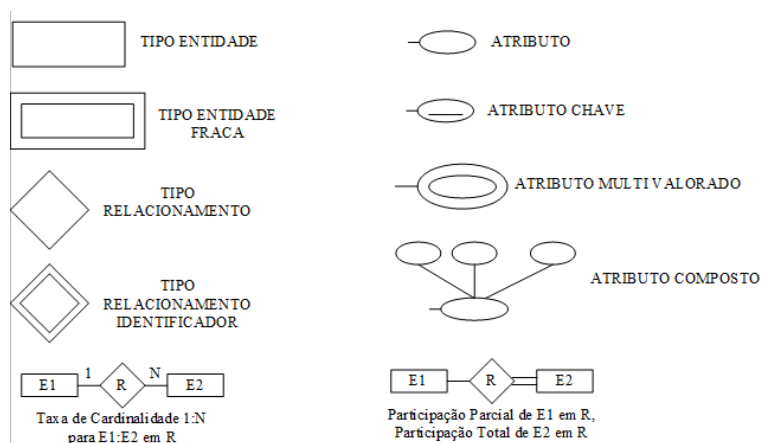


Figura 7: Resumo da notação ER

3 Modelo ER alternativo

Similarmente, vale ressaltar que há uma notação alternativa para o modelo ER em que iremos especificar restrições estruturais em relacionamentos. Dessa maneira, será associado um par de números naturais (min, max) a cada participação de uma entidade **E** num tipo de relacionamento **R**, onde $0 \leq \min \leq \max$ e $\max \geq 1$. Neste método, $\min = 0$ implica participação parcial, enquanto $\min > 0$ implica participação total.

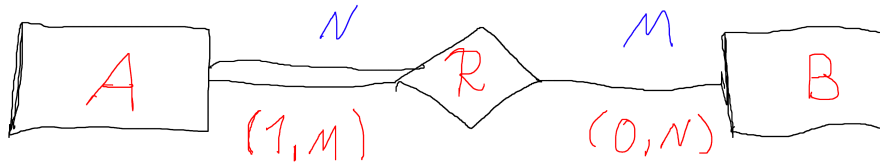


Figura 8: Primeiro exemplo para min-max

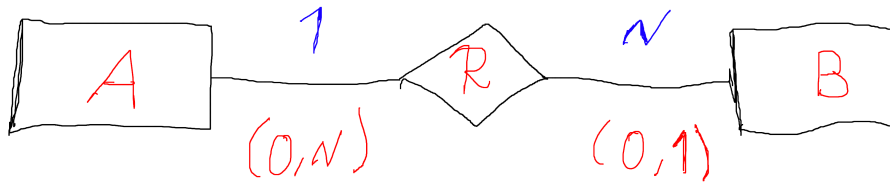


Figura 9: Segundo exemplo para min-max

Explicando a relação min-max nos dois exemplos, pode-se notar a diferença para as duas relações, na Figura 8 o valor mínimo é um, visto que a relação é total e seria necessário pelo menos um agente da entidade A para satisfazer a relação, já o valor de máximo é M, tendo em vista a cardinalidade de B, por outro lado, vemos em B que o mínimo é zero, considerando a relação parcial e o máximo é um baseado na cardinalidade de A.

De igual modo, analisando a figura 9, vemos que o mínimo será (0,N) e (0,1) considerando a relação parcial entre si e o valor da cardinalidade entre as duas entidades. Vale ressaltar que quando se tem uma relação total o mínimo não precisa ser necessariamente um, podendo variar de acordo com os requisitos do projeto

4 Modelo ER estendido

De tal maneira, vale ressaltar que para o modelo ER estendido veremos os conceitos de superclasse, subclasse e o construtor união.

4.1 Superclasse e subclasse

Em primeiro lugar, é válido dizer que uma superclasse se trata de uma entidade que será responsável por um determinado conjunto de instâncias, enquanto que a subclasse estará contida dentro desta instância herdando todos os atributos advindos da superclasse.

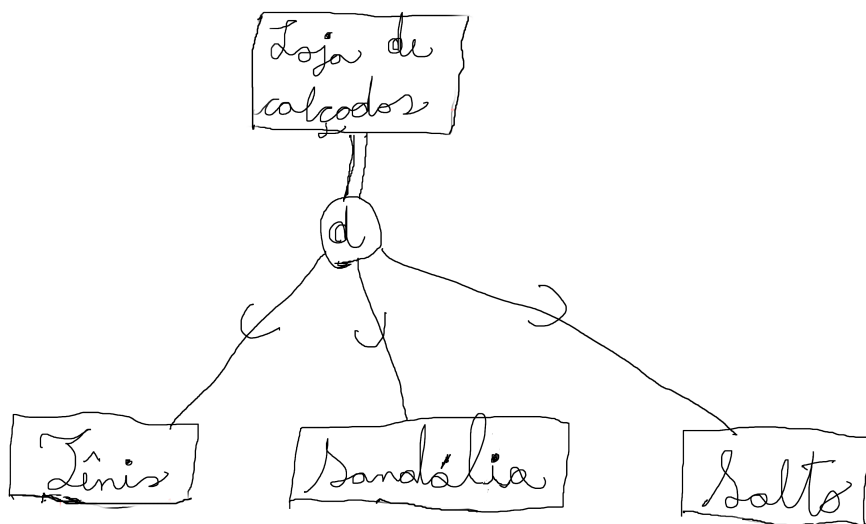


Figura 10: Exemplo de superclasse e subclasse

Temos no exemplo acima uma loja de calçados, observa-se que a entidade *loja* é a superclasse, enquanto que as entidades *tênis*, *sandália* e *salto* são sua subclasse e estão contidas na *loja*.

4.2 Generalização

Processo em que é suprimido as várias diferenças entre vários tipos de entidades, as características comuns são identificadas e generalizadas em uma única superclasse, da qual as entidades originais são subclasses.

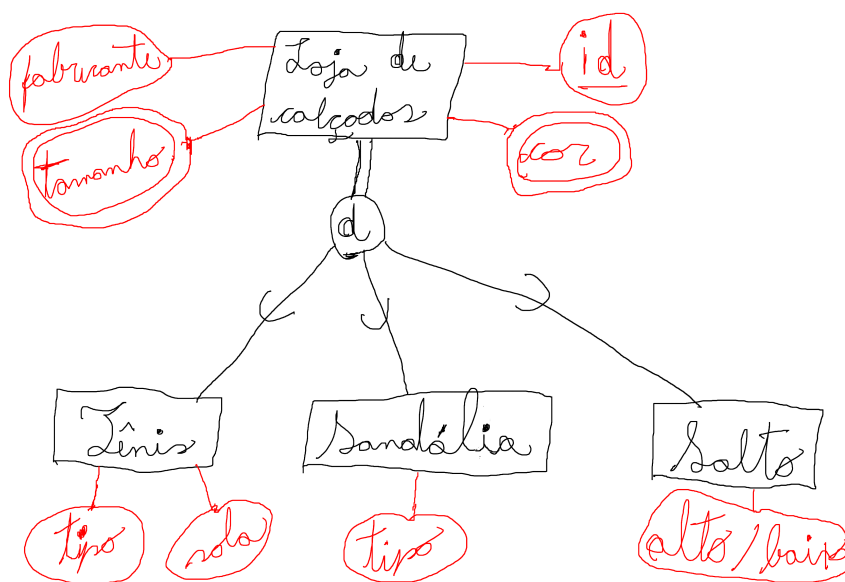


Figura 11: Exemplo de generalização

Usando o mesmo exemplo, percebe-se que a superclasse possui quatro atributos - *fabricante*, *tamanho*, *id*, *cor* - sendo assim, a sua subclasse também possui todos estes atributos.

4.3 Especialização

Temos dois tipos de especialização, a total em que toda instância da entidade na superclasse deve ser membro de alguma subclasse (representada por linha dupla) e parcial, sendo que esta permite que uma instância da entidade não pertença a nenhuma das subclasses (representada por linha simples). Além disso, nós temos quatro possíveis restrições na especialização: disjunção total e parcial e sobreposição total e parcial.

4.4 Hierarquia

Possui a restrição que toda subclasse participa como uma subclasse em somente um relacionamento classe/subclasse.

4.5 Grade de especialização

Pode ser uma subclasse em mais de um relacionamento classe/subclasse. Além disso, em uma hierarquia ou grade de especialização, uma subclasse herda os atributos não somente de sua superclasse direta mas também de todas as suas superclasses predecessoras ao longo de todo o caminho até a raiz da grade.

4.6 Herança múltipla

Acontece quando uma subclasse compartilhada herda atributos e relacionamentos de múltiplas classes.

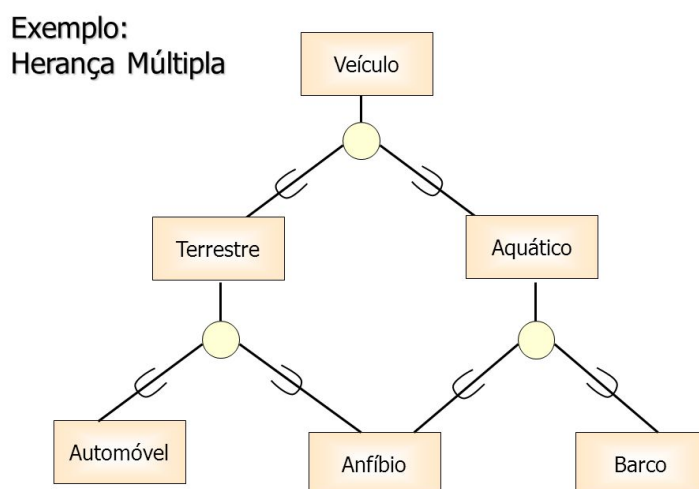


Figura 12: Exemplo herança múltipla

4.7 União ou categoria

Em último lugar, temos o conceito de união, representada por um círculo com a letra **u**, a subclasse de tipo união é útil quando se tem a necessidade de criar uma classe que contenha entidades de vários tipos diferentes.

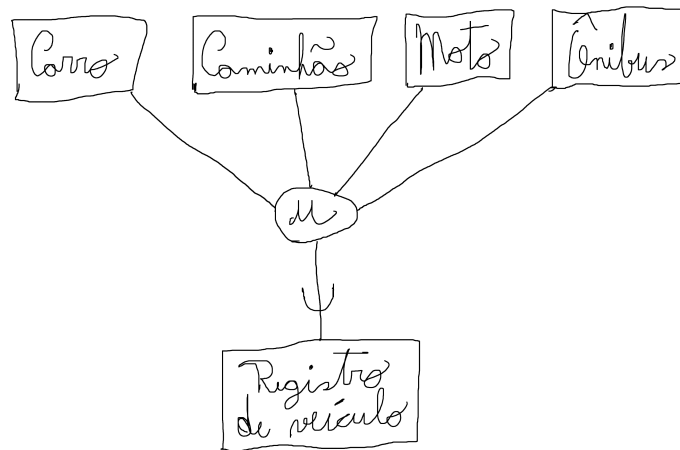


Figura 13: Exemplo de união

Percebe-se acima um exemplo de uma empresa que faz registro de veículos e que todos os tipos de automóveis estão em união com esta empresa. Entretanto, apesar da categoria ser um conceito importante em banco de dados, é preferível se utilizar da disjunção sempre que possível, desta maneira, mudando o exemplo anterior teríamos o seguinte diagrama:

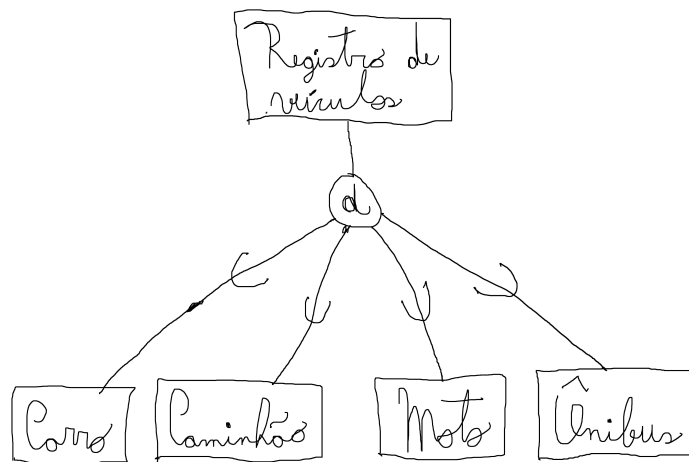


Figura 14: Transformação de uma união para uma disjunção

Posto isso, vemos no exemplo como um diagrama de união foi facilmente transformado em uma superclasse com suas demais subclasses se tornando um diagrama de disjunção parcial.