

Parte

4

REGRAS DO MODELO CONCEITUAL

As regras do modelo conceitual visam contextualizar a utilização de recursos da Modelagem Entidade-Relacionamento ora utilizada no Modelo Conceitual. Em função do contexto é importante aplicar cada tipo de estrutura de acordo com a regra de negócio para não incorrer em inconsistências quando da utilização de uma instância do banco de dados.

4.1 MODELO CONCEITUAL COMO MODELO DE ORGANIZAÇÃO

O modelo conceitual surgiu das idéias fundamentais do projeto de banco de dados: a de que através da identificação das entidades que terão informações representadas no banco de dados é possível identificar os arquivos que comporão o banco de dados.

Na prática, convencionou-se iniciar o processo de construção de um novo banco de dados com a construção de um modelo dos objetos da organização que será atendida pelo banco de dados, ao invés de partir diretamente para o projeto do banco de dados.

Esta forma de proceder permite envolver o usuário na especificação do banco de dados. Sabe-se da prática da engenharia de software que o envolvimento do usuário na especificação do software aumenta a qualidade do software produzido. A idéia é que o usuário é aquele que melhor conhece a organização e, portanto, aquele que melhor conhece os requisitos que o software deve preencher.

Modelos conceituais são modelos que descrevem a organização e, portanto, são mais simples de compreender por usuários leigos em Informática que modelos que envolvem detalhes de implementação.

4.2 DIFERENTES MODELOS PODEM SER EQUIVALENTES

Na prática, muitas vezes observam-se analistas em acirradas discussões a fim de decidir como um determinado objeto da realidade modelada deve aparecer no modelo. Às vezes, tais discussões são absolutamente supérfluas, pois os diferentes Modelos Entidade-

Relacionamento em qualquer das opções defendidas pelos diferentes analistas geram o mesmo banco de dados. Há um conceito de equivalência entre Modelos Entidade-Relacionamento. De maneira informal, dois modelos são equivalentes quando expressam o mesmo problema, ou seja, quando modelam a mesma realidade.

Para definir o conceito de equivalência de forma mais precisa, é necessário considerar o Banco de Dados que é projetado a partir do Modelo Entidade-Relacionamento. Para fins de projeto de Banco de Dados, dois Modelos Entidade-Relacionamento são equivalentes, quando ambos geram o mesmo esquema de Banco de Dados. Assim, para analisar se dois modelos são equivalentes, é necessário considerar um conjunto de regras de tradução de Modelos Entidade-Relacionamento para modelos lógicos de banco de dados. Quando falamos “o mesmo modelo de Banco de Dados Relacional” estamos falando de bancos de dados que, abstraindo de diferenças de nomes de estruturas (tabelas, atributos, ...) tenham a mesma estrutura. Um caso é o da equivalência entre um modelo que representa um conceito através de um relacionamento n:n e outro modelo que representa o mesmo conceito através de uma entidade. Um exemplo é o relacionamento CONSULTA que foi transformado em uma entidade. Os dois modelos são equivalentes, pois expressam o mesmo e geram o mesmo banco de dados.

a) A consulta como relacionamento n:n



Figura 39 - Relacionamento n:n

b) A consulta como entidade

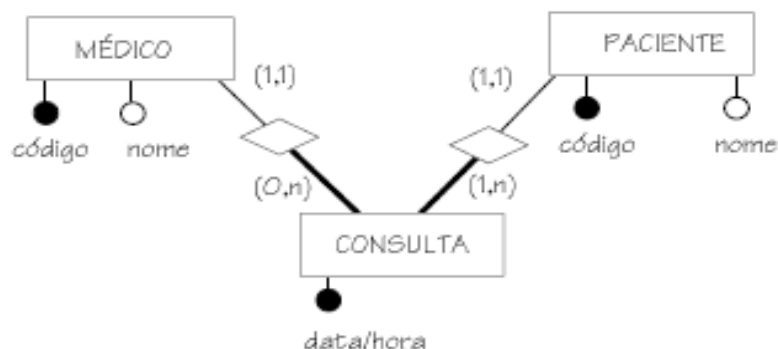


Figura 40 - Relacionamento 1:n

4.3 ATRIBUTO VERSUS ENTIDADE RELACIONADA

Uma questão que às vezes surge na modelagem de um sistema é entre modelar um objeto como sendo um atributo de uma entidade ou como sendo uma entidade autônoma relacionada a essa entidade.

Exemplificando, no caso de uma indústria de automóvel, como devemos registrar a cor de cada automóvel que sai da linha de produção? Caso considerarmos que cada automóvel possui uma única cor predominante, pode-se pensar em modelar a cor como um atributo da entidade AUTOMÓVEL ou modelar a cor como uma entidade autônoma, que está relacionada à entidade AUTOMÓVEL.

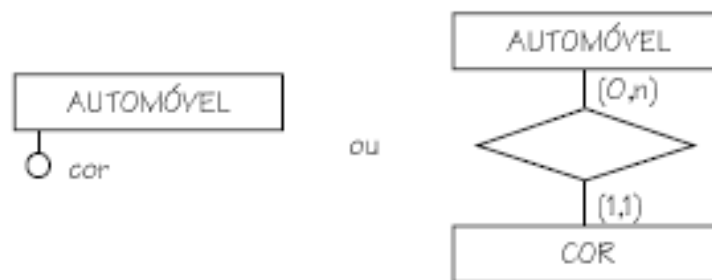


Figura 41 - Atributo X Entidade Relacionada

Alguns critérios para esta decisão são:

- Caso o objeto cuja modelagem está em discussão esteja vinculado a outros objetos (atributos, relacionamentos, entidades genéricas ou especializadas), o objeto deve ser modelado como entidade, já que um atributo não pode ter atributos, nem estar relacionado a outras entidades, nem ser generalizado ou especializado. Caso contrário, o objeto pode ser modelado como atributo. Assim, no caso do exemplo das cores dos automóveis, poder-se-ia optar por modelar a cor como uma entidade, caso se tivesse que registrar no banco de dados os possíveis fabricantes da tinta da referida cor (entidades relacionadas à cor), ou caso se quisesse registrar as datas de início e fim (atributos) do uso de uma determinada cor. Caso não houvesse nenhum objeto relacionado à cor do automóvel, poder-se-ia modelá-la como atributo da entidade automóvel.

- Quando o conjunto de valores de um determinado objeto é fixo durante toda a vida do sistema ele pode ser modelado como atributo, visto que o domínio de valores de um atributo é imutável. Quando existem transações no sistema que alteram o conjunto de valores do objeto, o mesmo não deve ser modelado como atributo. Assim, retomando o exemplo das cores dos automóveis, caso existissem transações de criação/eliminação de cores, seria preferível a modelagem de cor como entidade relacionada à entidade automóvel.

4.4 ATRIBUTO VERSUS GENERALIZAÇÃO/ESPECIALIZAÇÃO

Outro conflito de modelagem para o qual há algumas regras de cunho prático é entre modelar um determinado objeto (por exemplo, a categoria funcional de cada empregado de

uma empresa) como atributo (categoria funcional como atributo da entidade EMPREGADO) ou através de uma especialização (cada categoria funcional corresponde a uma especialização de entidade empregado).

Uma especialização deve ser usada quando se sabe que as classes especializadas de entidades possuem propriedades (atributos, relacionamentos, generalizações, especializações) particulares. Assim, no caso do exemplo acima, faz sentido especializar a entidade empregado de acordo com a categoria funcional, no caso de as classes particulares possuírem atributos ou relacionamentos próprios.

Ainda no exemplo dos empregados, o sexo do empregado é mais bem modelado como atributo de empregado, caso não existam propriedades particulares de homens e mulheres a modelar na realidade considerada.

a) Modelando categoria funcional como atributo:

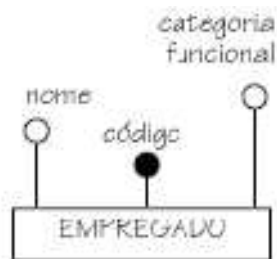


Figura 42 - Modelagem com o atributo

b) Modelando categoria funcional como especialização:

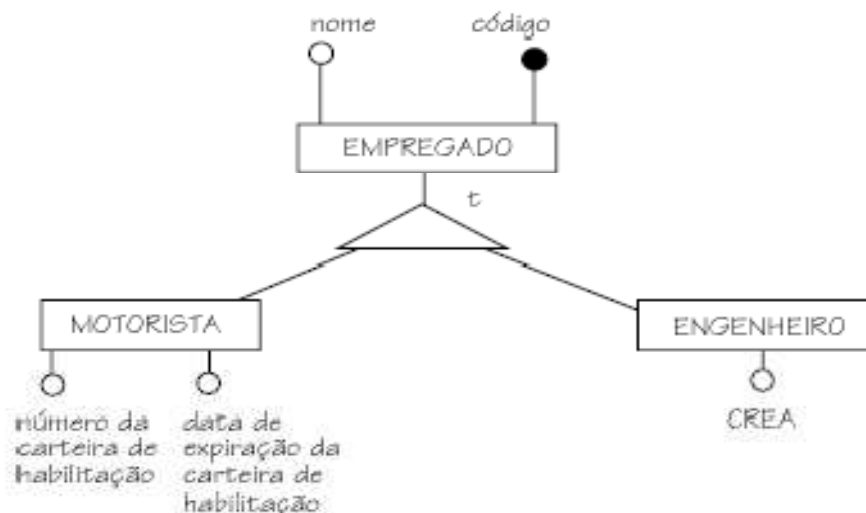


Figura 43 - Modelagem como Generalização/ Especialização

4.5 VERIFICAÇÃO DO MODELO

Uma vez confeccionado, um modelo ER deve ser validado e verificado. A verificação é o controle de qualidade que procura garantir que o modelo usado para a construção do banco de dados gerará um bom produto. Um modelo, para ser considerado bom, deve preencher uma série de requisitos, como ser completo, ser correto e não conter redundâncias.

4.5.1 O Modelo deve ser Correto

Um modelo está correto quando não contém erros de modelagem, isto é, quando os conceitos de Modelagem Entidade-Relacionamento são corretamente empregados para modelar a realidade em questão. Podem-se distinguir entre dois tipos de erros:

- Erros sintáticos
- Erros semânticos

Os **erros sintáticos** ocorrem quando o modelo não respeita as regras de construção de um Modelo Entidade-Relacionamento. Por exemplo: associar atributos a atributos; associar relacionamentos a atributos; associar relacionamentos através de outros relacionamentos ou de especializar relacionamentos ou atributos.

Os **erros semânticos** ocorrem quando o modelo, apesar de obedecer às regras de construção de Modelos Entidade-Relacionamento (estar sintaticamente correto) reflete a realidade de forma inconsistente. Por exemplo: Estabelecer associações incorretas, como associar a uma entidade um atributo que na realidade pertence à outra entidade. Isso acontece quando em um modelo com entidades CLIENTE e FILIAL, associa-se a CLIENTE o nome da FILIAL.

4.5.2 O Modelo deve ser Completo

Um modelo completo deve fixar todas as propriedades desejáveis do banco de dados. Isso obviamente somente pode ser verificado por alguém que conhece profundamente o sistema a ser implementado. Uma boa forma de verificar se o modelo é completo é verificar se todos os dados que devem ser obtidos do banco de dados estão presentes e se todas as transações de modificação do banco de dados podem ser executadas sobre o modelo.

Este requisito é aparentemente conflitante com a falta de poder de expressão de Modelos Entidade-Relacionamento. Quando dizemos que um modelo deve ser completo, estamos exigindo que todas as propriedades expressáveis no modelo conceitual apareçam no modelo.

4.5.3 Modelo Deve ser Livre de Redundâncias

Um modelo deve ser mínimo, isto é não deve conter conceitos redundantes. Um tipo de redundância que pode aparecer é a de relacionamentos redundantes. Relacionamentos

redundantes são relacionamentos que são resultado da combinação de outros relacionamentos entre as mesmas entidades.

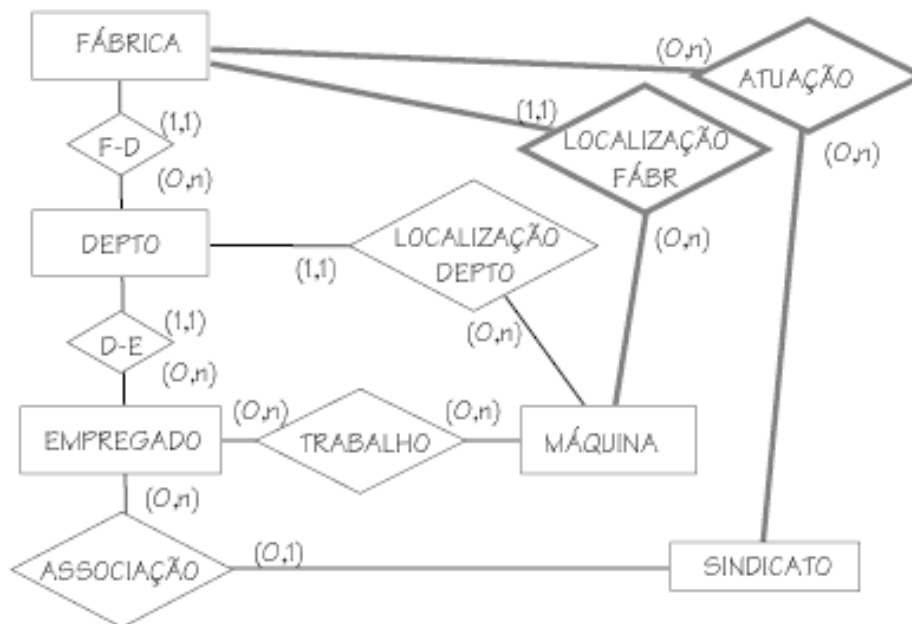


Figura 44 - Modelo Redundante

4.5.4 Modelo Deve Refletir o Aspecto Temporal

Usualmente, ao iniciar a modelagem Entidade-Relacionamento, a preocupação é obter um modelo que descreva os estados válidos e corretos do banco de dados. O primeiro modelo tende a refletir um estado momentâneo do banco de dados. Entretanto, é necessário lembrar que assim como informações são incluídas no banco de dados, elas também podem ter que ser eliminadas do banco de dados.

Um banco de dados não pode crescer indefinidamente. Informações ultrapassadas ou desnecessárias podem ser eliminadas. Portanto, é necessário considerar o aspecto temporal na modelagem de dados. Não há regras gerais como proceder neste caso, mas é possível identificar alguns padrões que se repetem freqüentemente na prática.

- a) Manter ou não um histórico de salários?



Figura 45 - Manutenção de histórico de salário

b) Manter ou não o histórico de alocação de mesas?

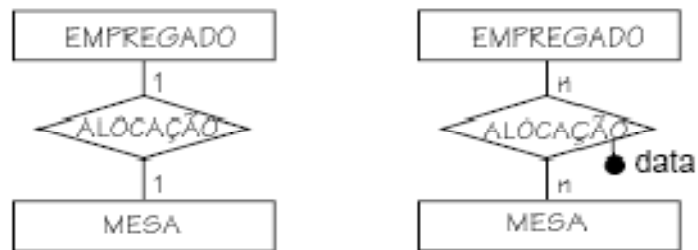


Figura 46 - Manutenção de histórico de empregado-mesa

c) Manter ou não o histórico de lotação do empregado?

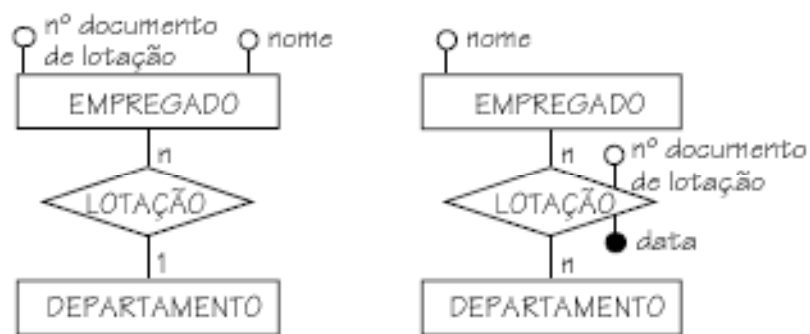


Figura 47 - Manutenção de histórico de empregado-departamento

d) Manter somente a inscrição atual ou o histórico de todas as inscrições nos cursos?



Figura 48 - Manutenção de histórico de participante