

Restrições de Integridade

Uma restrição de integridade referencial no modelo de dados de entidade (EDM) é semelhante a uma restrição de integridade referencial em um banco de dados relacional. Da mesma forma que uma coluna (ou colunas) de uma tabela de banco de dados pode referenciar a chave primária de outra tabela, uma Propriedade (ou Propriedades) de um tipo de entidade pode referenciar a chave de entidade de outro tipo de entidade.

O tipo de entidade referenciado é chamado de extremidade principal da restrição. O tipo de entidade que faz referência à extremidade principal é chamado de extremidade dependente da restrição.

Uma restrição de integridade referencial é definida como parte de uma Associação entre dois tipos de entidade. A definição de uma restrição de integridade referencial especifica as seguintes informações:

- O final principal de restrição. (Um tipo de entidade cuja chave de entidade é referenciada pela o final dependente.)
- A chave de entidade de extremidade principal.
- O final dependente de restrição. (Um tipo de entidade que tem uma propriedade ou um propriedades que referenciem a chave de entidade de extremidade principal.)
- A propriedade ou propriedades referência de extremidade dependente.

O objetivo de restrições de integridade referencial em EDM é garantir que as associações válidos sempre existe. Para obter mais informações, consulte Propriedade Foreign Key.

Exemplo

O diagrama a seguir mostra um modelo conceitual com duas associações: WrittenBy e PublishedBy.

O tipo de entidade de Book tem uma propriedade, PublisherId, que faz referência a chave de entidade do tipo de entidade de Publisher quando você define uma restrição de integridade referencial em associação de PublishedBy .

O Entity Framework ADO.net usa uma DSL (linguagem específica de domínio) chamada CSDL (linguagem de definição de esquema conceitual) para definir modelos conceituais.

CSDL seguir define uma restrição de integridade referencial em associação de PublishedBy mostrada no modelo conceitual anterior.

Mapeamento de Modelo ER para modelo Relacional

Quando estudamos os conceitos introdutórios de banco de dados sempre falando do modelo conceitual entidade relacionamento (ER) e do modelo lógico relacional. Explicamos que durante um projeto de banco de dados a evolução natural leva você a construir os dois modelos a partir dos requisitos do cliente.

Legal, então você começou pelo mini mundo, juntou os requisitos e construiu seu modelo ER! Agora você está com dificuldade de passar esse modelo para o modelo relacional. Seus problema acabaram! Nas próximas linhas vamos explicar o passo a passo da transformação do seu diagrama ER em tabelas.

O livro do Navathe sugere um algoritmo que nos ajuda no processo de transformar um esquema ER e um mapeamento relacional. Vamos utilizar o esquema abaixo para ilustrar o passo-a-passo do mapeamento.

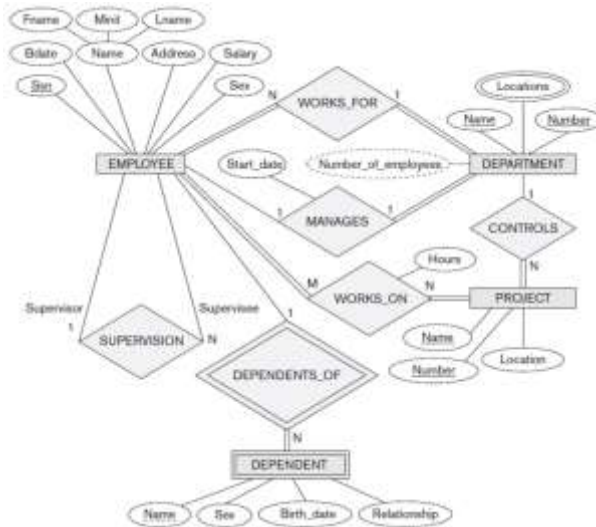


Diagrama Entidade Relacionamento

Passo 01: O mapeamento começa com as entidades regulares do modelo ER. Para cada tipo de entidade forte E presente no diagrama, criamos uma relação R que inclui todos os atributos simples da entidade E. Escolha os atributos chaves da entidade para ser chave primária da relação. Se a chave da entidade for composta, o mesmo conjunto de atributos que formam a chave de E vão fazer parte da chave primária de R.

Vejam no exemplo temos as entidades fortes EMPLOYEE, DEPARTMENT e PROJECT serão transformados em relações. Os atributos SSN, DNUMBER e PNUMBER serão as chaves primárias das respectivas tabelas. Assim temos:

Employee

Fname	Minit	Lname	Ssn	Bdate	Adress	Salary	Sex
-------	-------	-------	-----	-------	--------	--------	-----

Department

Dname	Dnumber
-------	---------

Project

Pname	Pnumber	Plocation
-------	---------	-----------

Passo 02: A segunda etapa se concentra em mapear as entidades fracas. Para cada entidade fraca W no modelo ER com uma entidade pai ou proprietária E, você deve criar uma relação R e incluir todos os atributos simples (ou os componentes simples de atributos compostos) de W como atributos de R.

É necessário ainda incluir como chave estrangeira os atributos que fazem parte da chave primária da entidade proprietária E. Assim a chave primária será formada pela combinação da chave primária da entidade pai mais a chave parcial da entidade fraca, caso exista.

Vamos mais uma vez para o nosso diagrama ER. A entidade fraca DEPENDENT deve dar origem a uma relação. Essa relação terá os atributos da entidade, além da chave estrangeira que vem da entidade pai.

Vejam, então, que o Ssn fará parte da relação. Desta forma podemos construir a relação dependente conforme descrito abaixo:

Dependent

Essn	Dependent_name	Sex	Bdate	Relationship
------	----------------	-----	-------	--------------

Passo 03: Vamos agora começar a mapear os relacionamentos binários 1:1 entre as entidades. Para cada relacionamento binário 1:1 R no esquema ER, identificamos as relações que correspondem as entidades S e T que participam do relacionamento R. A partir deste momento temos três ações possíveis.

A primeira seria usar chave estrangeira para efetivar o relacionamento. Neste caso você escolhe uma das chaves primárias de S ou de T. Vamos supor que escolhemos a chave de S. Você vai usar a chave primária de S e incluir ela como chave estrangeira da relação T.

A melhor opção é escolher a entidade que tenha participação total no relacionamento. Em outras palavras, você deve escolher a entidade que para cada instância existente da entidade ela tenha uma instância no relacionamento e incluir a chave primária da outra entidade nela. Vejam que neste caso evitamos o uso valores nulos.

Um exemplo desta opção é colocar na relação DEPARTAMENT a chave do funcionário que gerencia (MANAGES) o departamento. Vejam que todo departamento deve ter um funcionário como gerente. Agora nossa tabela de DEPARTAMENT aparece com os novos atributos: o id do gerente e a data de início da gerência.

Departament

Dname	Dnumber	Mgr_ssn	Mgr_start_date
-------	---------	---------	----------------

A segunda opção seria fazer uma merge ou fusão das relações. Vejam que se temos um relacionamento 1:1 podemos combinar os atributos das duas relações e do relacionamento em uma única relação. Essa opção é adequada quando a participação das duas entidades no relacionamento é total.

A terceira opção seria por cross-reference ou criação de uma relação para o relacionamento. Essa alternativa basicamente cria uma nova relação com a chave primária das duas relações participantes para prover o relacionamento entre elas. Conhecida como tabela de ligação.

Passo 04: Neste momento vamos nos preocupar com os relacionamentos binários 1:N. Para cada relacionamento binário 1:N, vamos identificar a relação S que representa a participação da entidade no lado N do relacionamento. Em seguida, devemos incluir a chave estrangeira da relação T em S. Os atributos da relação 1:N também devem aparecer na relação S.

Vejamos um exemplo, o relacionamentos WORKS_FOR, CONTROLS e SUPERVISION no diagrama ER acima. Para WORKS_FOR vamos incluir o número do departamento como chave estrangeira na relação EMPLOYEE, vamos chama-lo de DNO.

Da mesma forma vamos incluir SUPER_SSN para implementar o relacionamento SUPERVISION e o número do departamento na tabela de projeto para implementar o relacionamento CONTROLS. Vejam como ficaram as relações EMPLOYEE e PROJECT com essas mudanças.

Employee

Fname	Minit	Lname	Ssn	Bdate	Adress	Salary	Sex	Super_ssn	Dno
-------	-------	-------	-----	-------	--------	--------	-----	-----------	-----

Project

Pname	Pnumber	Plocation	Dnum
-------	---------	-----------	------

Passo 05: O próximo mapeamento deve tomar conta dos relacionamentos M:N. Para cada relacionamento M:N criar uma nova relação para representar o relacionamento. Inclua como chaves estrangeiras da nova relação as chaves primárias das entidades participantes. Inclua ainda os atributos simples do relacionamento na nova tabela.

Observe o relacionamento WORKS_ON no diagrama ER. Ele é mapeado pela criação da entidade WORKS_ON no esquema relacional. A chave primária é formada pelas chaves estrangeiras de projeto e empregado, renomeadas respectivamente para PNO e ESSN, respectivamente. O atributo HOURS é representado na relação.

Works_On

Essn	Pno	Hours
------	-----	-------

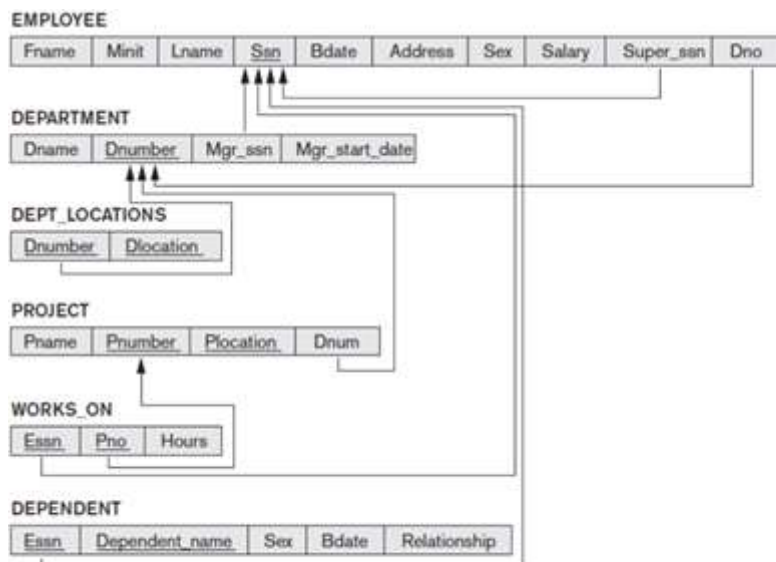
Passo 06: Vamos agora fazer o mapeamento dos atributos multivalorados. Para cada atributo multivalorado A, crie uma tabela R. Essa relação R vai incluir qualquer atributo pertencente ao conjunto A, além da chave primária da entidade K que tem A como atributo multivalorado sendo chave estrangeira em R. A chave primária da relação é uma combinação de A e K.

No nosso modelo temos o exemplo de DEPT_LOCATIONS que deve ser criado. O atributo DLOCATION representa um atributo multivalorado de departamento, que tem como chave primária DNUMBER. A relação DEPT_LOCATIONS fica com a seguinte configuração:

Dept_Locations

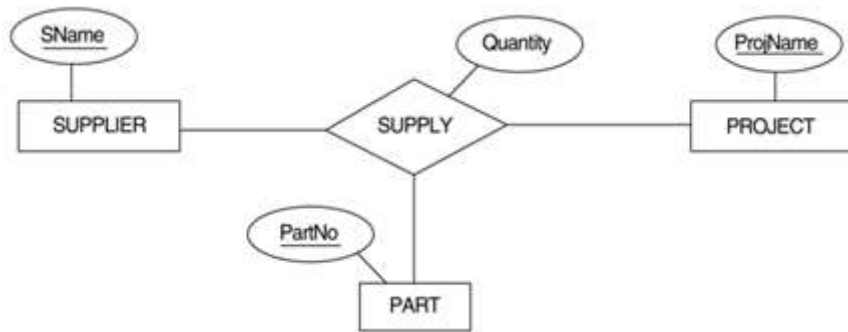
Dnumber	Dlocation
---------	-----------

Antes de passar para o passo 07, vamos apresentar abaixo o esquema relacional completo derivado do diagrama ER apresentado acima. As setas indicam a existência de uma chave estrangeira que garante a integridade referencial.



Modelo relacional

Passo 07: A última etapa é o mapeamento de relacionamentos N-ários. Devemos criar uma nova relação que inclua as chaves primária de todas as entidades participantes do relacionamento quanto $n > 2$. Adicione também os atributos simples. Observe o diagrama abaixo o resultado do mapeamento do relacionamento SUPPLY em uma relação.



Digrama ER de um relacionamento ternário

SUPPLIER			
<u>SNAME</u>	...		
PROJECT			
<u>PROJNAME</u>	...		
PART			
<u>PARTNO</u>	...		
SUPPLY			
<u>SNAME</u>	<u>PROJNAME</u>	<u>PARTNO</u>	QUANTITY

Modelo relacional derivado de um relacionamento ternário.

[illegible]