**DESCRIÇÃO DO MAPEAMENTO**

Henry Bernardo Kochenborger de Avila

Marcos Samuel Winkel Landi

Algumas entidades/relacionamentos foram mapeados diretamente para tabelas no modelo relacional. Ou seja, suas chaves primárias são iguais a sua representação no diagrama conceitual.

Entretanto, algumas das modelagens realizadas obrigaram um mapeamento não tão direto - ou então com múltiplas abordagens.

* **DEPARTAMENTO:** seu mapeamento foi um tanto quanto simples, pois essa entidade se encaixa perfeitamente com a descrição em tabelas como sugerida pelo modelo relacional. Assim sendo, atribuir aos atributos do modelo conceitual a chaves no modelo lógico é totalmente equivalente. Dessa maneira, cada atributo fora mapeado seguindo o mapeamento proposto pelos slides da disciplina:
  + **Código:** foi mapeado para a chave primária ‘codDep’ da tabela.
  + **Nome:** foi mapeado para a chave não-nula ‘nome’ da tabela.
* **CURSO:** assim como o departamento, a descrição no modelo relacional foi intuitiva e se deu por:
  + **Código:** foi mapeado para a chave primária ‘codCurso’ da tabela.
  + **Nome:** foi mapeado para a chave não-nula ‘nome’ da tabela.
* **HABILITAÇÃO:** essa entidade não tem um mapeamento tão direto pois é uma entidade fraca no modelo conceitual. Assim sendo, foi necessário atribuir a chave estrangeira que se refere a um curso como um dos componentes da chave primária. Além disso, essa entidade possui uma chave composta e, por este motivo, seus componentes foram atribuídos a chaves independentes no modelo relacional.
  + **Código:** foi mapeado para um dos componentes da chave primária ‘codHab’ da tabela.
  + **Relacionamento de entidade fraca:** foi mapeado para o outro componente da chave primária ‘codCurso’ da tabela - essa chave que referencia a tabela ‘Curso’.
  + **Nome:** foi mapeado diretamente para uma chave não-nula ‘nome’ da tabela.
  + **Créditos:** por ser um atributo composto, apenas seus componentes foram mapeados para a tabela, sendo eles:
    - Obrigatórios: foi mapeado para o atributo não-nulo ‘creditosObrigatorios’ da tabela.
    - Eletivos: foi mapeado para o atributo não-nulo ‘creditosEletivos’ da tabela.
    - Complementares: foi mapeado para o atributo não-nulo ‘creditosComplementares’ da tabela.
* **PESSOA:** seu mapeamento foi feito utilizando a terceira alternativa proposta pela disciplina que é a implementação por relações distintas. Dessa maneira, como essa generalização é total e exclusiva, então essa alternativa implica na criação de apenas duas tabelas:
  + **EDUCADOR:** seus atributos foram mapeados diretamente, assim como os atributos da sua *super-entidade*.
    - **Código:** foi mapeado para a chave primária ‘idEdu’.
    - **CPF:** foi mapeado para a chave alternativa não-primária ‘CPF’.
    - **Nome**: foi mapeado para a chave não-nula ‘nome’.
  + **ALUNO:** seu mapeamento foi trivial para seus atributos e análogos ao mapeamento do educador. Entretanto, o aluno também possui o relacionamento ‘Inscrição’ com as bolsas de graduação e o relacionamento ‘Estudo’ com as habilitações e, por este motivo, foi necessário criar essas ligações por chaves estrangeiras - levando em consideração o mapeamento recomendado pela disciplina presente nos slides de mapeamento.
    - **Número:** foi mapeado para a chave primária ‘numCartao’.
    - **CPF:** foi mapeado para a chave alternativa não-primária ‘CPF’.
    - **Nome:** foi mapeado para a chave não-nula ‘nome’.
    - **Relacionamento ‘Estudo’:** foi mapeado como uma chave estrangeira composta e opcional por (codHab, codCurso).
    - **Relacionamento ‘Inscrição’:** foi mapeado para a chave alternativa, estrangeira e opcional ‘codBolsa’ que referencia a tabela ‘Bolsa’.
* **DISCIPLINA:** seu mapeamento foi análogo ao mapeamento da entidade ‘Aluno’. Dessa forma, o relacionamento ‘Oferecimento’ foi mapeado para uma chave estrangeira nesta tabela. Assim sendo, seus atributos e relacionamentos foram alocados da seguinte maneira:
  + **Código:** foi mapeado para a chave primária ‘codDisc’.
  + **Nome:** foi mapeado para a chave não-nula ‘nome’.
  + **Créditos:** foi mapeado para a chave não-nula ‘creditos’.
  + **Ativo:** foi mapeado para a chave não-nula (inicializada em ‘True’) ‘ativo’.
  + **Relacionamento ‘Oferecimento’:** foi mapeado para a chave estrangeira ‘codDep’.
* **PRÉDIO:** seu mapeamento foi trivial e muito semelhante ao mapeamento da entidade ‘Departamento’, com a diferença sendo a existência de um atributo composto. Portanto, seus atributos foram mapeados assim:
  + **Número:** foi mapeado para a chave primária ‘numPredio’.
  + **Local:** foi mapeado como duas chaves não-nulas que formam uma chave alternativa composta pelo par, respectivamente:
    - Lagitude: chave não-nula dada por ‘lagitude’.
    - Longitude: chave não-nula dada por ‘longitude’.
* **SALA:** seu mapeamento foi feito considerando seus atributos e seu relacionamento com a entidade prédio. Dessa forma, como cada sala pertence a apenas um prédio, suas chaves foram disponibilizadas da seguinte forma:
  + **Número:** foi mapeado para o primeiro componente da chave primária dado por ‘numSala’.
  + **Relacionamento ‘Pertencimento’:** foi mapeado para a chave estrangeira, e segundo componente da chave primária desta tabela, ‘numPredio’.
* **TURMA:** seu mapeamento depende do fato de essa entidade ser dependente da entidade ‘Disciplina’. Ademais, os relacionamento entre essas duas entidades (‘Atribuição’) e ‘Localidade’ foram acoplados nesta tabela - assim como indicado pelos slides de mapeamento. Assim sendo, seus atributos e relacionamentos ficaram da seguinte forma:
  + **Código:** foi mapeado para o primeiro componente da chave primária ‘codTurma’.
  + **Vagas:** foi mapeado para a chave não-nula ‘vagas’.
  + **Horário:** fora simplificado utilizando a tipagem de string. Dessa maneira, seu mapeamento ficou para uma chave não-nula ‘horario’.
  + **Relacionamento ‘Localidade’:** foi mapeado para as chaves estrangeiras que referem-se a sala (numSala, numPredio).
  + **Relacionamento ‘Atribuição’:** foi mapeado com a chave estrangeira ‘codDisc’.
* **BOLSA:** seu mapeamento foi o mais complexo pois esta entidade se relaciona com pelo menos outras três entidades. Assim sendo, como todos os relacionamentos possuíam o mesmo estilo de mapeamento segundo as recomendações dos slides (0-n, 1-1), então seus mapeamentos foram feitos por atributos nesta tabela.  
   Além disso, essa entidade pertence ao nível mais alto de uma das hierarquias contidas no modelo conceitual. Para esse mapeamento, a abordagem escolhida fora a segunda alternativa proposta pelos slides da disciplina. Dessa forma, algumas garantias de integridade com o comando check se fizeram necessárias.  
   Portanto, o seguinte mapeamento foi feito:
  + **Código:** foi mapeado para a chave primária ‘codBolsa’.
  + **Benefício:** por ser um valor monetário, o tipo decimal é o que mais se adequa para representá-lo. Ademais, seu mapeamento foi feito para uma chave opcional chamada ‘beneficio’.
  + **Carga horária:** seu mapeamento foi feito para a chave não-nula ‘cargaHoraria’.
  + **Créditos:** foi mapeado para a chave não-nula ‘creditos’.
  + **Relacionamento ‘Concessão’:** foi mapeado com a ligação pela chave primária da tabela departamento, nesta tabela sendo chave estrangeira não-nula chamada de ‘codDep’.
  + **Relacionamento ‘Orientação’:** foi mapeado como uma chave estrangeira não-nula nesta tabela chamada de ‘eduResponsavel’ que referencia a tabela ‘Educador’.
  + **Hierarquia:** como a bolsa de monitoria possui um relacionamento próprio, então é preciso garanti-lo apenas quando o tipo da bolsa for do tipo ‘monitoria’. Assim sendo, o mapeamento da hierarquia necessitou de um check para não haver o relacionamento ‘Monitoramento’ quando a bolsa for de iniciação científica:
    - **Tipo:** foi criada uma chave não-nula que identifica o tipo de bolsa de graduação, sendo que os possíveis valores que ela pode assumir são ‘ic’ ou ‘monitoria’.
    - **Relacionamento ‘Monitoramento’:** este relacionamento foi mapeado nesta tabela pois fica mais fácil garantir que apenas as bolsas de monitoria terão ligação com a tabela ‘Turma’. Assim sendo, foi necessário criar a chave estrangeira opcional e composta que refere-se a tabela ‘Turma’ com a adição das chaves ‘turmaMonitoriaCod’ e ‘turmaMonitoriaDisc’.
* **ENTRADA DE CURRÍCULO:** por ser uma entidade fraca e dependente de duas outras entidades, os relacionamentos entre esta entidade e essas outras duas foram mapeados para dentro desta entidade. Dessa forma, os atributos e os relacionamentos ficaram da seguinte maneira:
  + **Créditos necessários:** foi mapeado para uma chave opcional chamada ‘requisitoCreditos’.
  + **Obrigatoriedade:** foi mapeado para uma chave não-nula que pode assumir um dos três possíveis valores:
    - 1: ‘obrigatoria’.
    - 2: ‘eletiva’.
    - 3: ‘opcional’.
  + **Etapa:** foi mapeado para uma chave opcional chamada ‘etapa’ e foi necessário criar uma uma verificação de integridade que garante que esta chave não seja nula apenas quando a entrada for obrigatória.
  + **Relacionamento ‘Programação’:** foi mapeado para a primeira componente da chave primária chamada ‘codDisc’ que referencia a tabela ‘Disciplina’.
  + **Relacionamento ‘Currículo’:** foi mapeado para a segunda e terceira componentes da chave primária chamadas ‘codHab’ e ‘codCurso’, respectivamente.
* **RELACIONAMENTO PRÉ-REQUISITO:** por ser um relacionamento ternário, seu mapeamento foi feito por meio de uma tabela própria para o relacionamento. Assim sendo, ele foi mapeado da seguinte forma:
  + **Entidade ‘Disciplina’:** foi criada uma componente da chave primária chamada ‘codDiscRequisito’ que se refere a tabela ‘Disciplina’.
  + **Entidade ‘Entrada de Currículo’:** foram criadas três outras componentes da chave primária chamadas ‘codDisc’, ‘codHab’ e ‘codCurso’.
  + **Entidade ‘Habilitação’:** foi criada a última componente da chave primária chamada ‘codCurso’.

Todas as restrições de integridade, realizadas nas tabelas, ‘CASCADE’ possuem essencialmente o mesmo motivo: a existência da entidade não faz sentido sem a chave estrangeira, como, por exemplo, com a chave estrangeira que referencia prédios na tabela sala, onde a não existência do prédio implica na não existência da sala. Já, nos casos de ‘SET NULL’, são situações plausíveis sem esse dado, como quando uma instância da tabela ‘Turma’ possui sua sala apagada - em que faz sentido deixá-lo como ‘NULL’ por um tempo até que outra sala seja alocada para esta turma.