



# BandTec

DIGITAL SCHOOL

## **BD – Banco de Dados**

Aula07 – Tipos de atributos

Relacionamento muitos-para-muitos

- Conceitos:
  - Atributo atômico ou primitivo ou simples vs atributo composto
  - Atributo monovalorado vs multivalorado
  - Atributo derivado
  - Atributo identificador ou determinante

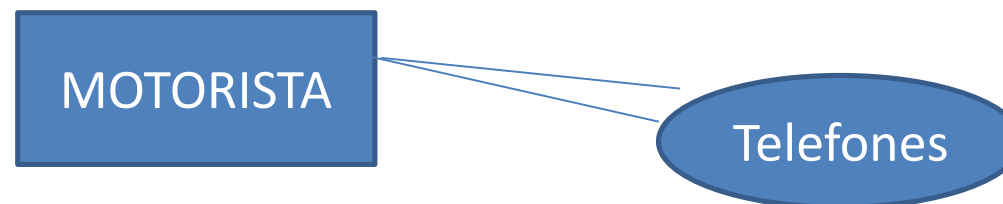
# Atributo atômico versus Atributo composto

- Atributo atômico ou primitivo ou simples:
  - Não pode ser dividido em subpartes menores
  - Exemplo: sexo
- Atributo composto:
  - Pode ser dividido em subpartes menores, com significado independente
  - Exemplo: Endereco – pode ser dividido em Endereço da Rua, Cidade, Estado, CEP



# Atributo monovalorado vs Atributo multivalorado

- Atributo monovalorado:
  - Tem apenas um valor na mesma entidade
  - Exemplo: Nome (do Motorista)
- Atributo multivalorado:
  - Pode ter mais de um valor
  - Exemplo: Telefone (do Motorista) – pode haver o telefone celular e o telefone residencial
  - Representado por um traço duplo ligando a entidade ao atributo:



# Atributo derivado

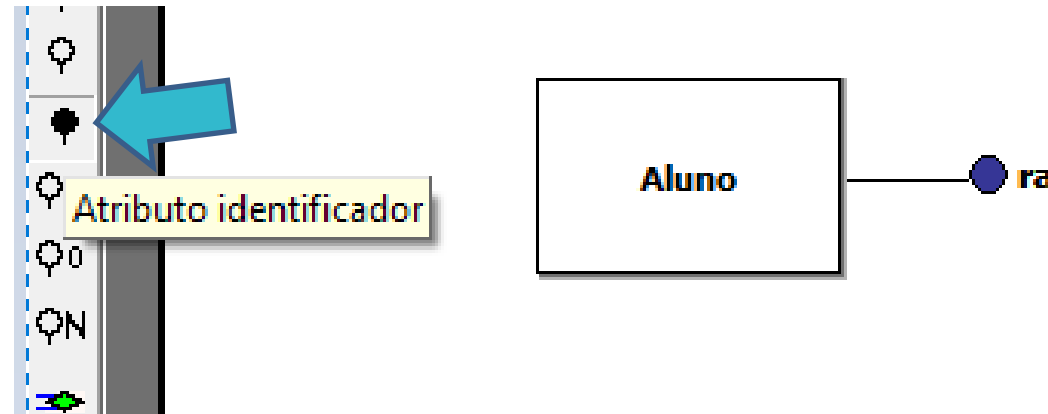
- Atributo cujo valor deve ser obtido após algum processamento de informações do próprio banco de dados:
  - Exemplo: valor\_fatura (no cenário do Rádio Táxi On-Line) – calculado por meio da soma dos valores das corridas realizadas num período, para uma conveniada
  - Outro exemplo: idade

# Atributo identificador ou determinante

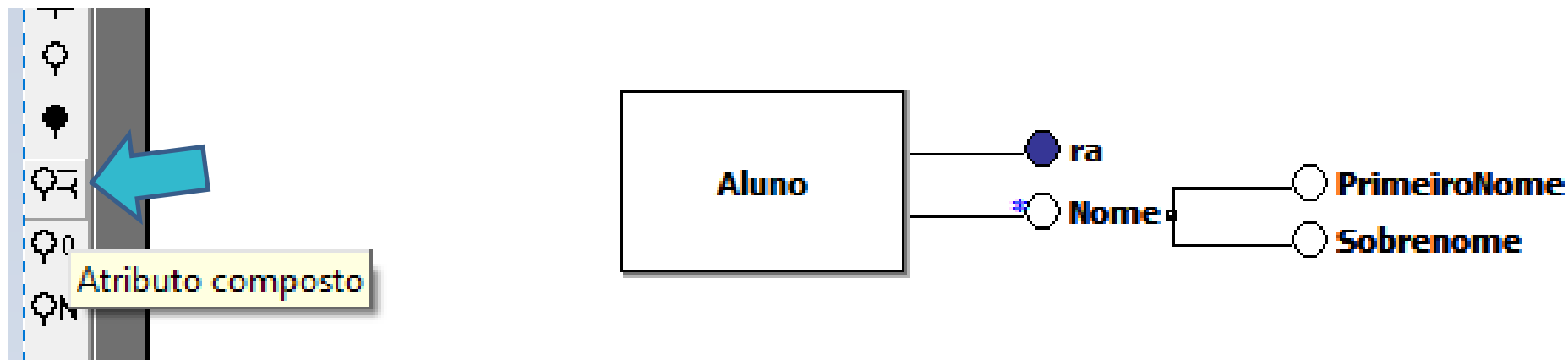
- Atributo cujo valor identifica de forma única uma ocorrência da entidade
- Esse atributo corresponde à chave primária da tabela.
  - Exemplo: RA do aluno, código do produto
- Boas práticas: recomenda-se escolher como atributo identificador alguma informação que o sistema tenha controle, como um código gerado internamente.
- Recomenda-se não utilizar RG, pois o RG é um número gerado pelo estado, podendo haver RGs de mesmo número para pessoas de estados diferentes

# Como fica a representação no brModelo?

Atributo identificador:  
ra (chave primária)



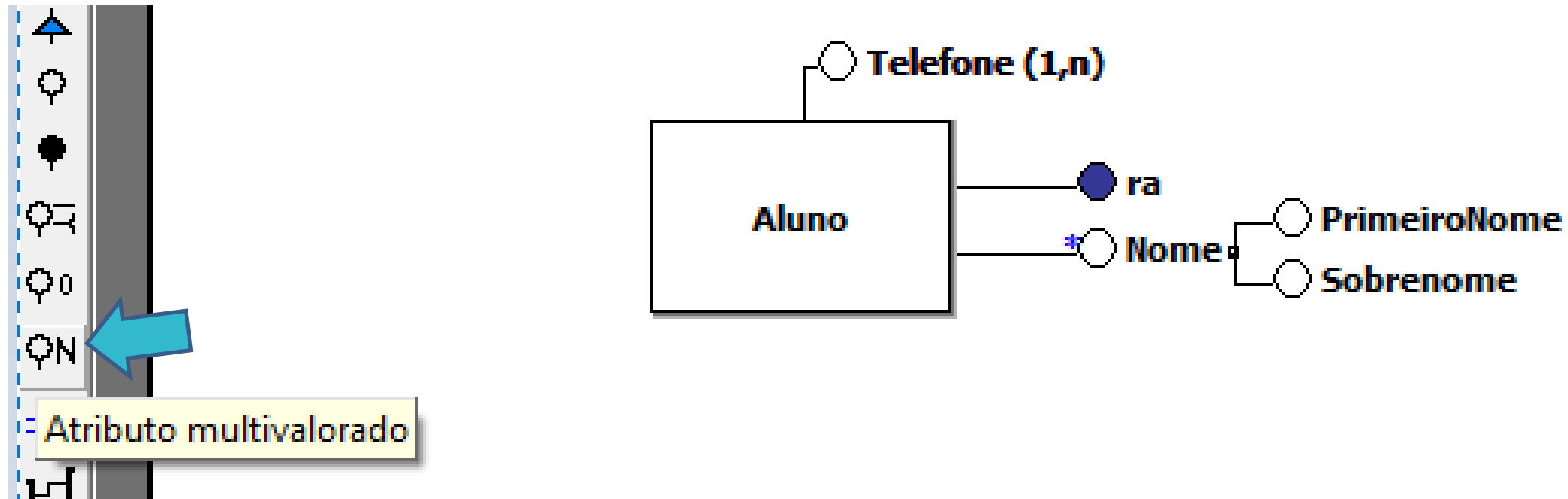
Atributo composto: nome (não necessariamente precisa ser composto)





# Como fica a representação no brModelo?

Atributo multivalorado: telefone



# Como fica a implementação no SQL?

- Atributo identificador
  - Corresponderá a um ou mais campos (colunas) que são PRIMARY KEY da tabela
  - Ex: `ra int primary key`
- Atributo composto
  - Cada parte do atributo corresponderá a um campo (coluna) da tabela
  - Ex: `primeiroNome varchar(40),  
sobrenome varchar(40)`

# Como fica a implementação no SQL?

- Atributo multivalorado
  - Será que podemos inserir mais de um valor dentro de um campo, numa mesma linha?
  - Por exemplo, será que podemos inserir mais de um número de telefone para um mesmo aluno, na coluna telefone?

ra	primeiroNome	sobrenome	telefone
01201200	Lucas	Silva	(11) 5678-2233 (11) 99123-7676
01201201	Pedro	Sousa	(11) 3365-5567 (11) 98765-8109 (11) 99000-6523

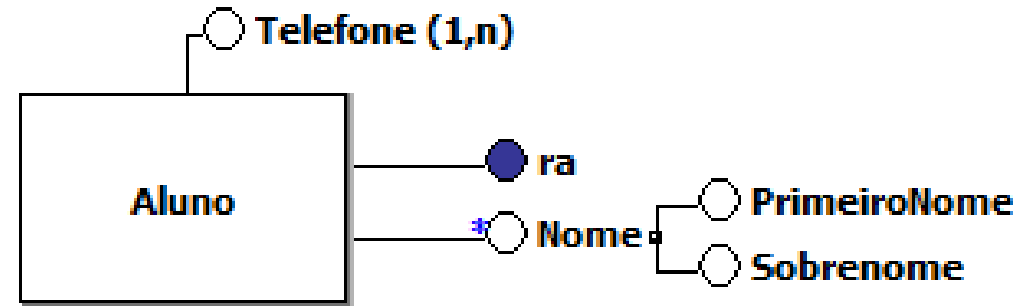
- O que vocês acham? É possível inserirmos dessa forma?

# Como fica a implementação no SQL?

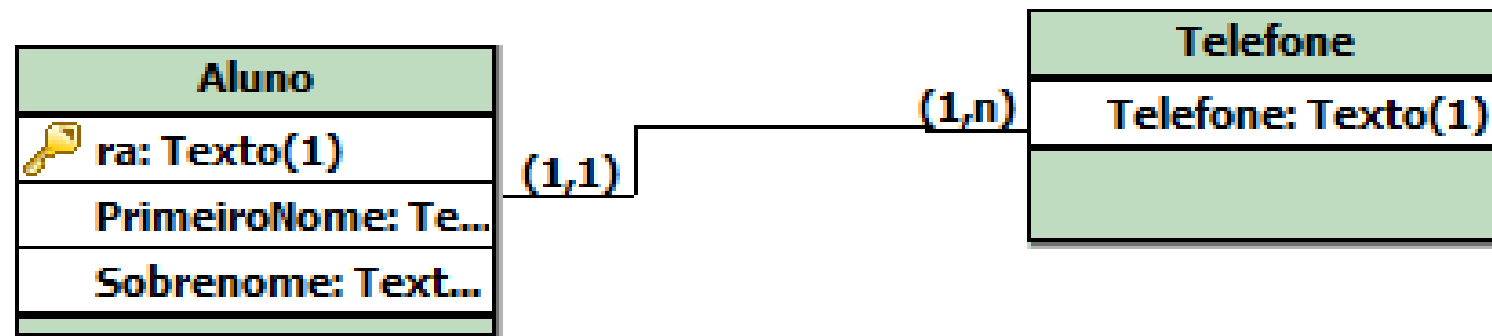
- Atributo multivalorado
  - A resposta é NÃO.
  - Em cada campo de cada linha da tabela, só podemos inserir UM valor!
  - Então como fica? O que vocês acham?
  - Depende da regra de negócio do projeto:
    - Se for até 2 ou 3 telefones no máximo por aluno, podemos criar 2 ou 3 colunas, uma para cada número de telefone
    - Se a quantidade de números de telefones que serão cadastrados por aluno não for definida, podendo ser um, 2, 5, 6, etc... Nesse caso, é melhor ter uma tabela Telefone, que se relaciona com a tabela Aluno, através de uma chave estrangeira.

# Modelo lógico gerado pelo brModelo

- Voltando ao nosso exemplo de modelo conceitual no brModelo:



- Vejam o modelo lógico correspondente, gerado pelo brModelo:



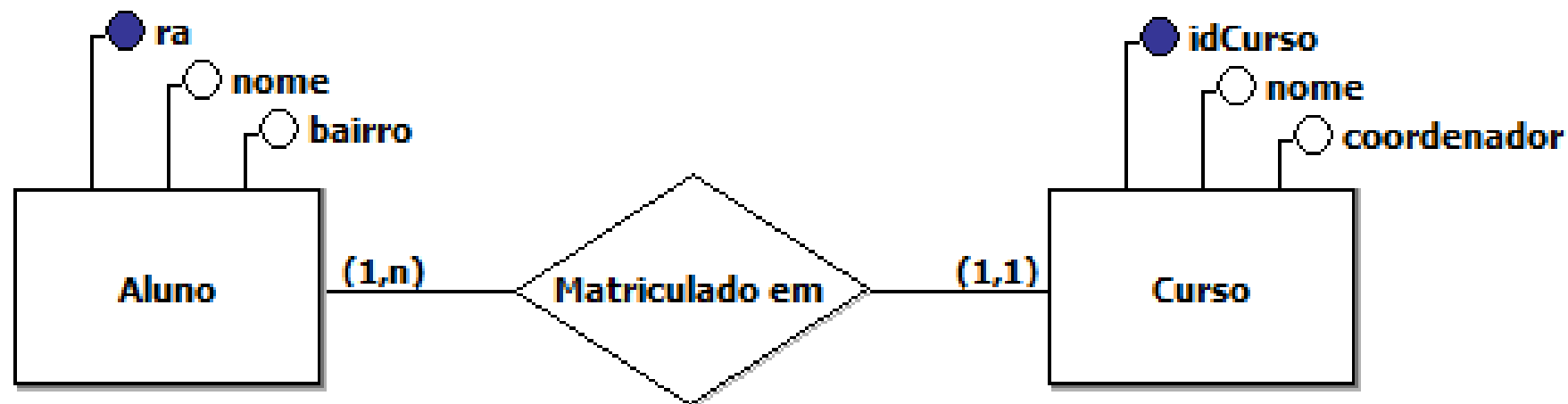
Obs.: Está faltando a chave estrangeira na tabela Telefone

# Relacionamento um-para-muitos

- Na modelagem conceitual que trabalhamos até agora, vimos o **relacionamento um-para-muitos**:
  - De um lado do relacionamento, a cardinalidade é no **máximo UM**
  - Do outro lado, a cardinalidade é no **máximo MUITOS**

- Por exemplo:

- Como se lê a cardinalidade?



- Seria o caso da Bandtec:

- Um Aluno está matriculado em no máximo UM Curso
- Um Curso pode ter no máximo MUITOS Alunos

# Relacionamento um-para-muitos

- Como se implementa no SQL o relacionamento um-para-muitos?
- Alguém sabe a resposta?
- Resposta: através de **uma chave estrangeira**, que fica ONDE?
- Na tabela do lado em que é **no máximo MUITOS**.
- Exemplo:

Tabela Aluno

ra	nome	bairro	fkCurso
50000	Juliana	Consolação	2
50001	Marcos	Saúde	1
50002	Ana	Jabaquara	1

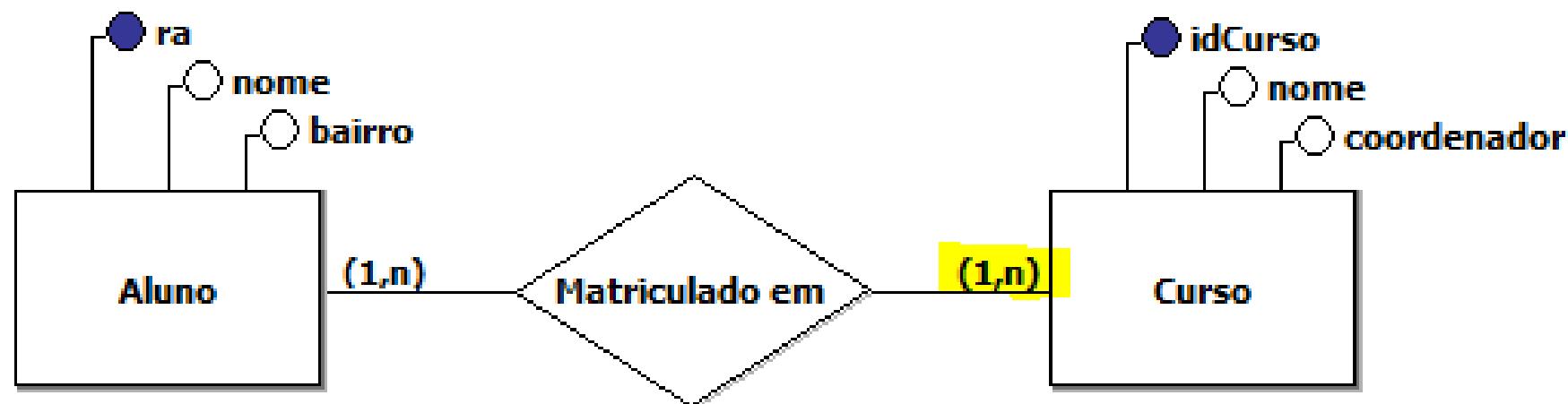


Tabela Curso

idCurso	nome	coordenador
1	ADS	Gerson
2	CCO	Marise

# Relacionamento muitos-para-muitos

- No entanto, e se fosse uma escola de idiomas?
- Nesse caso, cada aluno poderia fazer mais de um curso
- O relacionamento passaria então a ser **muitos-para-muitos**
  - Um Aluno pode fazer no máximo **MUITOS** Cursos
  - Um Curso pode ter no máximo **MUITOS** Alunos
- A modelagem conceitual ficaria assim:





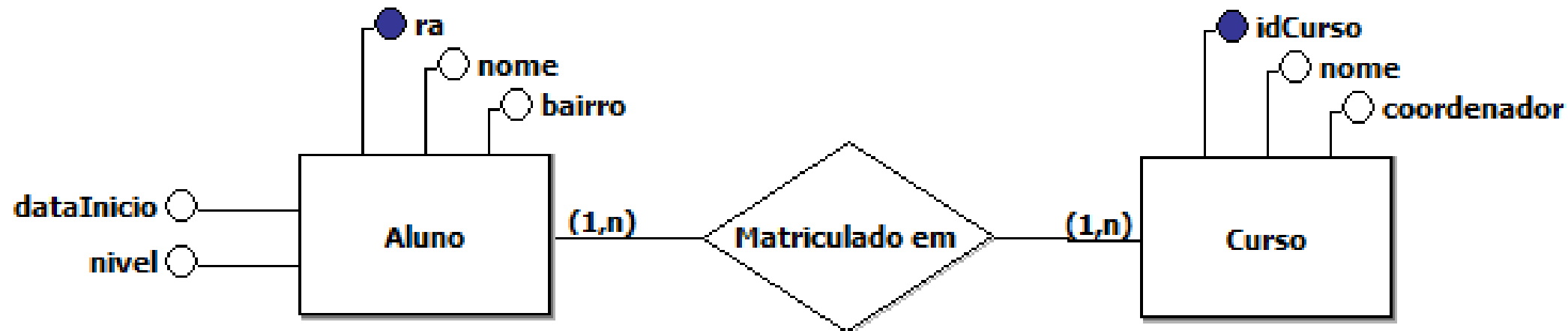
# Relacionamento muitos-para-muitos

- No caso de uma escola que oferece curso de idiomas, por exemplo, podemos ter:
  - O aluno João estuda espanhol desde 01/02/20 – nível básico B1 e estuda inglês desde 01/02/19 – nível intermediário I2.
  - A aluna Maria estuda espanhol desde 01/02/15 – nível avançado A1 e estuda inglês desde 01/02/17 – nível intermediário I3.
- Essas 2 informações:
  - Data de início do curso
  - Nível que está no curso

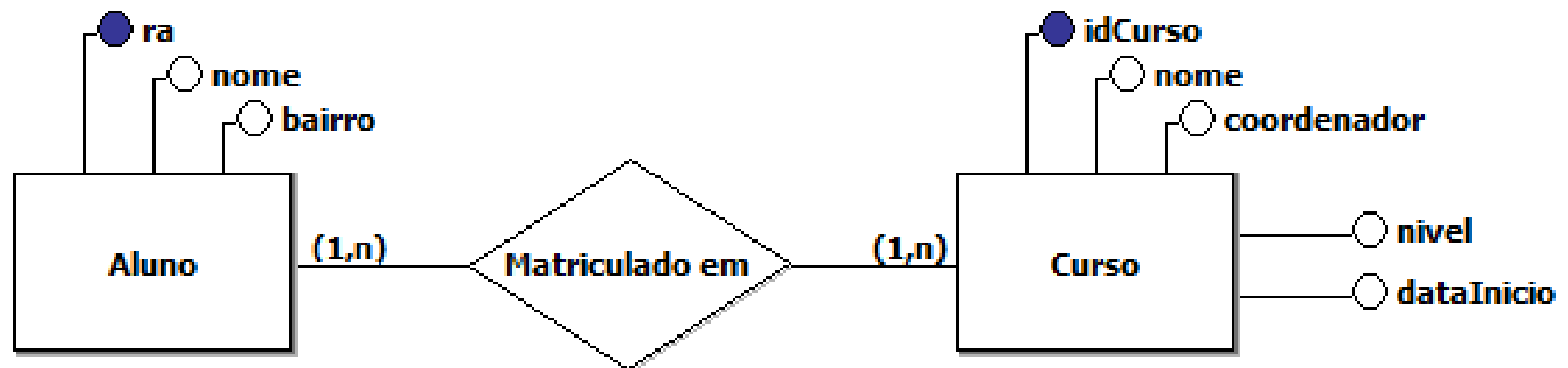
Aonde essas 2 informações devem estar na modelagem conceitual?

# Relacionamento muitos-para-muitos

- dataInicio e nível: são atributos de Aluno?

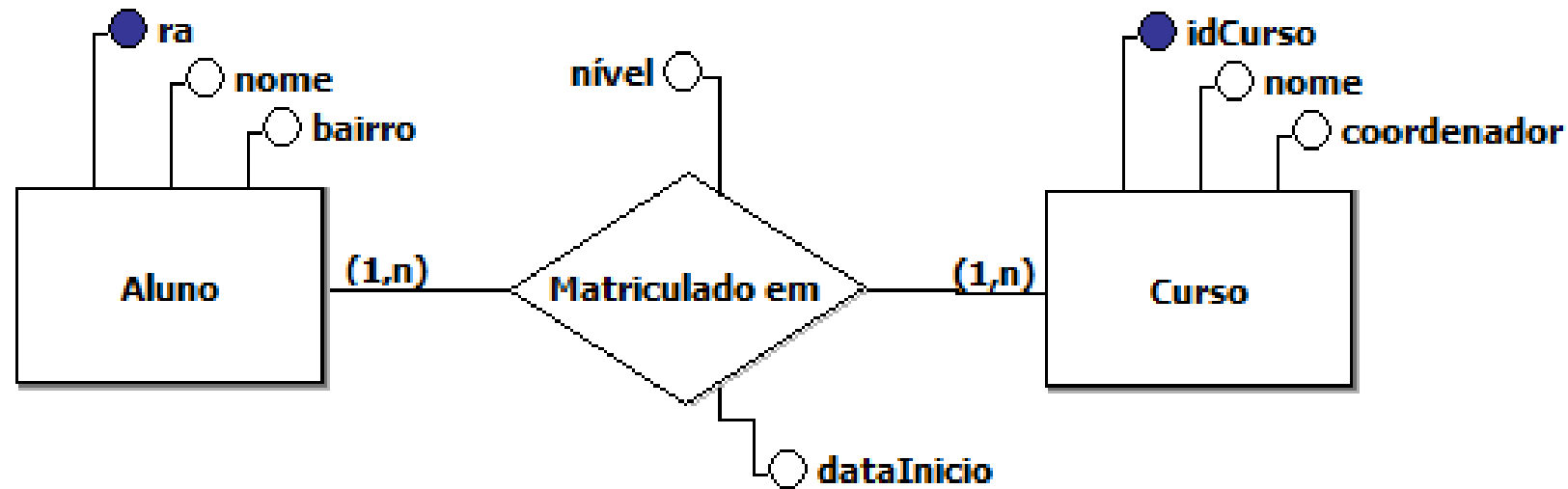


- dataInicio e nível: são atributos de Curso?



# Relacionamento muitos-para-muitos

- dataInício e nível: são atributos do relacionamento



- Quando o relacionamento é muitos-para-muitos, o relacionamento pode ter atributos

# Relacionamento muitos-para-muitos

- Implementação do relacionamento muitos-para-muitos

Tabela Aluno

ra	nome	bairro
1000	Maria	Saúde
1001	João	Consolação

Tabela Curso

idCurso	Nome	coordenador
1	Espanhol	Juan
2	Inglês	Beth

Tabela Associativa

fkRa	fkCurso	dataInicio	nível
1000	1	01/02/15	A1
1000	2	01/02/17	I3
1001	1	01/02/20	B1
1001	2	01/02/19	I2

# Referências Bibliográficas

- PUGA, Sandra; FRANÇA, Edson; GOYA, Milton. **Banco de Dados: Implementação em SQL, PL/SQL e Oracle 11g.** São Paulo: Pearson Education do Brasil, 2013. 329 p.
- ELMASRI, Ramez; NAVATHE, Shamkant B. **Sistemas de banco de dados.** Tradução de Marília Guimarães Pinheiro et al. 4. ed. São Paulo: Pearson Addison Wesley, 2005. 724 p.

**Obrigada!**

**BandTec**  
DIGITAL SCHOOL

Em caso de dúvidas, entre em contato com:  
[celia.taniwaki@bandtec.com.br](mailto:celia.taniwaki@bandtec.com.br)