

# Banco de Dados I

Introdução à Banco de Dados - Parte 2



# Arquitetura dos sistemas de banco de dados

A arquitetura dos SGBDs têm evoluído desde os primeiros sistemas monolíticos, nos quais todo o software SGBD era um sistema altamente integrado, até os mais modernos, que têm um projeto modular, com arquitetura de sistema cliente/servidor.



# Arquitetura dos sistemas de banco de dados

Essa evolução espelha as tendências na computação, em que grandes computadores mainframes centralizados estão sendo substituídos por centenas de estações de trabalho distribuídas e computadores pessoais, conectados por redes de comunicações a vários tipos de máquinas servidoras Web, servidores de banco de dados, servidores de arquivos, servidores de aplicações, e assim por diante.



# Arquitetura dos sistemas de banco de dados

Em uma arquitetura básica de SGBD cliente/servidor a funcionalidade do sistema é distribuída entre dois módulos.

- **Módulo cliente:** normalmente é projetado para executar em uma estação de trabalho ou computador pessoal;
  - programas de aplicação e interfaces com o usuário que acessam o banco de dados executam no módulo cliente.
  - Esse módulo se encarrega da interação do usuário e oferece interfaces amigáveis, como formulários ou interfaces gráficas do usuário baseadas em menu.

# Arquitetura dos sistemas de banco de dados

Quando é necessário acessar o SGBD, o programa estabelece uma **conexão com o SGBD** (que está no lado servidor); quando a conexão é criada, o programa cliente pode se comunicar com o SGBD.

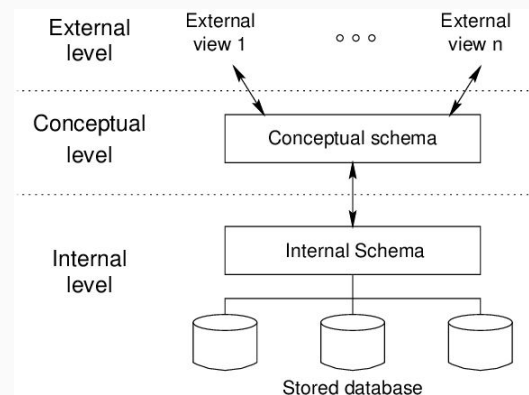


O padrão de **Conectividade de Banco de Dados Aberta** (*ODBC Open Database Connectivity*) é uma interface de programação de aplicativos (API) que permite que aplicativos acessem e interajam com diferentes bancos de dados de maneira uniforme e independente do sistema de gerenciamento de banco de dados (SGBD).

# Arquitetura dos sistemas de banco de dados

## Arquitetura ANSI/SPARC

Desenvolvido pelo Instituto Nacional Americano de Padrões (ANSI) e pelo Comitê de Planejamento e Requisitos de Padrões (SPARC) com o objetivo de criar um padrão conceitual que descrevesse os **componentes** e **funcionalidades de sistemas de gerenciamento de banco de dados (SGBD)** de maneira clara e precisa.

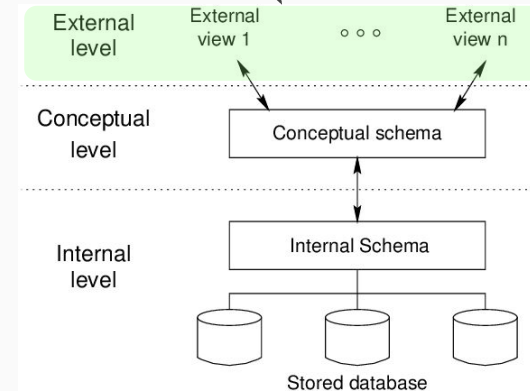


# Arquitetura dos sistemas de banco de dados

## Arquitetura ANSI/SPARC

Nível mais alto, lidando em como usuários finais interagem com o banco de dados.

- descreve as diferentes visões dos dados para diferentes grupos de usuários
- cada grupo pode ter sua própria visão personalizada dos dados



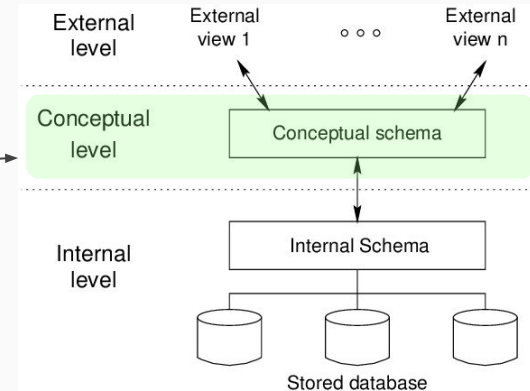
**Nível externo**

# Arquitetura dos sistemas de banco de dados

## Arquitetura ANSI/SPARC

Descreve a estrutura lógica global do banco de dados.

- fornece uma visão abstrata e independente de detalhes de implementação,
- permite que administradores de BD definam a estrutura do BD e as restrições de integridade.



**Nível  
conceitual**

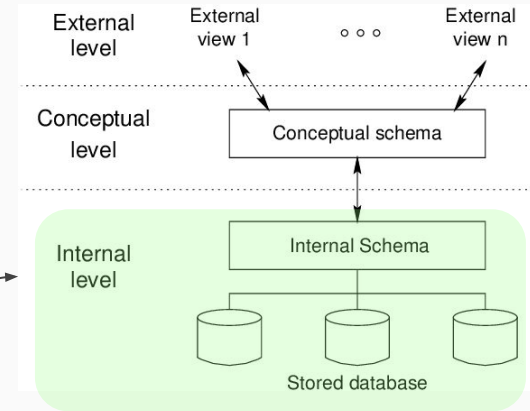


# Arquitetura dos sistemas de banco de dados

## Arquitetura ANSI/SPARC

Nível que lida com a representação física dos dados em um banco de dados.

- engloba detalhes de como os dados são armazenados em dispositivos de armazenamento, e como operações de acesso aos dados são otimizadas.



**Nível interno**

# Arquitetura dos sistemas de banco de dados

**Módulo Servidor:** responsável pelo armazenamento de dados, acesso, pesquisa e outras funções.

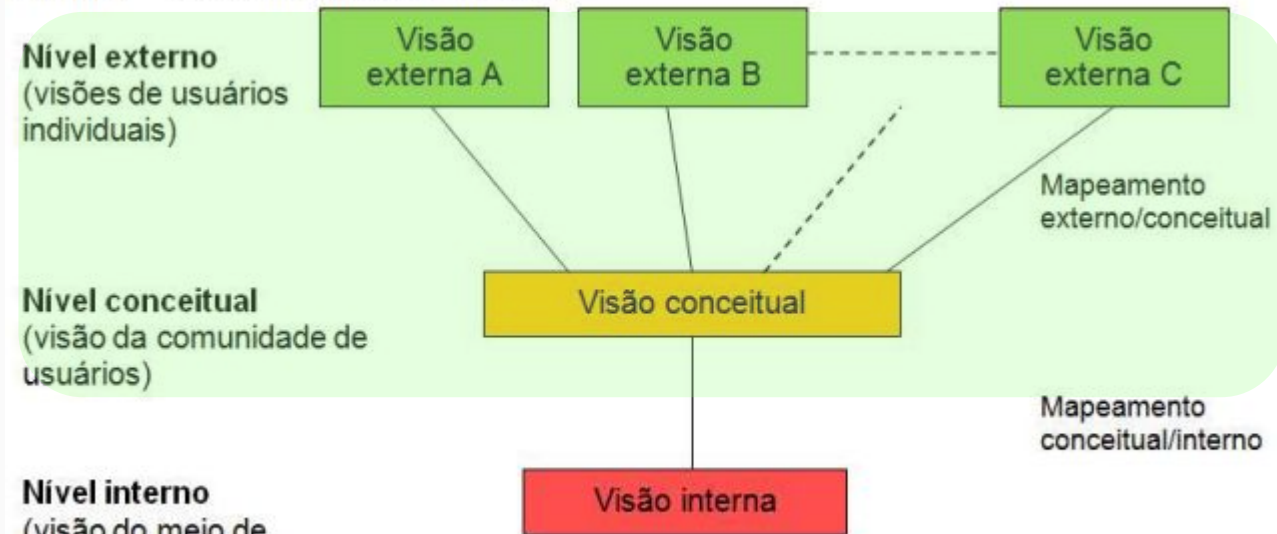
Figura 3 – Arquitetura cliente/servidor lógica em duas camadas



Fonte: Elmasri & Navathe (2011)

# Arquitetura dos sistemas de banco de dados

Figura 4 – Os três níveis da arquitetura

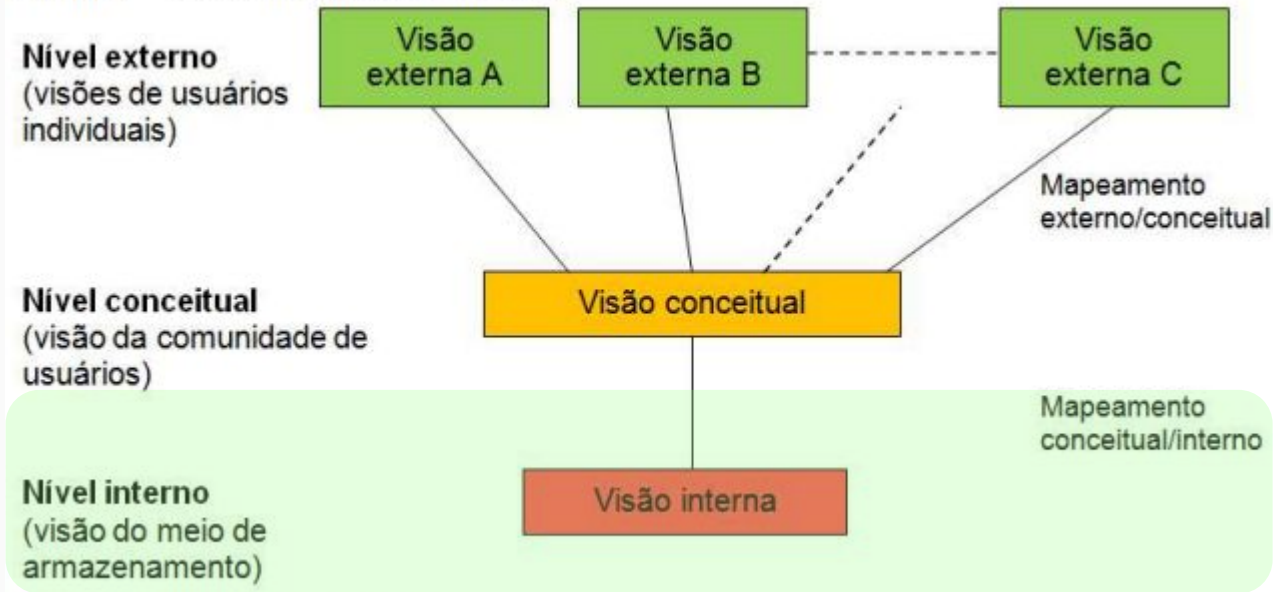


Fonte: adaptado de Date (2004)

construções voltadas para o usuário, como registros e campos

# Arquitetura dos sistemas de banco de dados

Figura 4 – Os três níveis da arquitetura



Fonte: adaptado de Date (2004)

definido em termos de construções voltadas para a máquina, como bits e bytes

# Arquitetura dos sistemas de banco de dados

Os três esquemas são apenas descrições dos dados; os dados armazenados que realmente existem estão **apenas no nível físico**.

Em um SGBD baseado na arquitetura ANSI/SPARC, grupos de usuários têm seus próprios esquemas externos. O SGBD traduz solicitações nos esquemas externos em solicitações no conceitual e, em seguida, em solicitações no esquema interno para processamento no banco de dados armazenado.

**Mapeamento:** ao solicitar a recuperação de dados, os resultados do BD armazenado são formatados para à visão externa do usuário. Mapeamentos facilitam a transformação de solicitações entre os níveis .

# Modelo de Dados Relacional

## Modelo Relacional

Um modelo relacional de banco de dados é um modelo de dados que organiza as informações em tabelas bidimensionais chamadas "relações".

Cada relação se assemelha a uma **tabela de dados** em que cada linha na tabela representa uma coleção de valores de dados relacionados.

Os nomes das tabelas e das colunas auxiliam na interpretação do significado dos valores contidos em cada linha.

Diagram illustrating the relational model structure:

- Atributos** (Attributes) point to the column headers: Cpf, Nome, Sexo, Telefone, Endereco, Data\_nascimento, and Renda.
- Nome da relação** (Relation Name) points to the table name: **CLIENTE**.
- Tuplas** (Tuples) point to the rows of data.

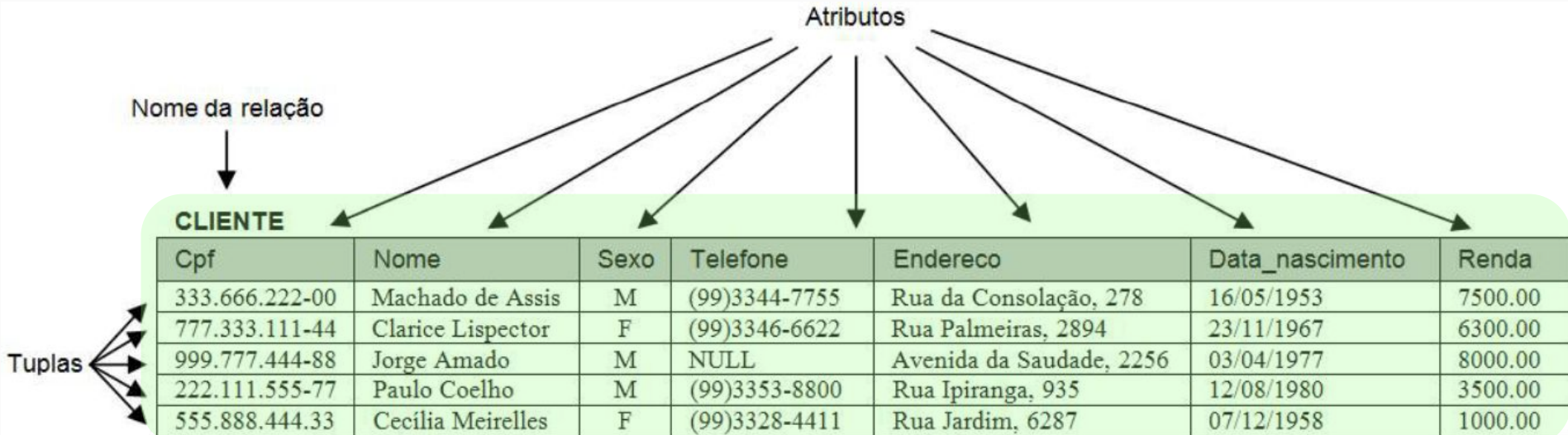
Cpf	Nome	Sexo	Telefone	Endereco	Data_nascimento	Renda
333.666.222-00	Machado de Assis	M	(99)3344-7755	Rua da Consolação, 278	16/05/1953	7500.00
777.333.111-44	Clarice Lispector	F	(99)3346-6622	Rua Palmeiras, 2894	23/11/1967	6300.00
999.777.444-88	Jorge Amado	M	NULL	Avenida da Saudade, 2256	03/04/1977	8000.00
222.111.555-77	Paulo Coelho	M	(99)3353-8800	Rua Ipiranga, 935	12/08/1980	3500.00
555.888.444.33	Cecilia Meirelles	F	(99)3328-4411	Rua Jardim, 6287	07/12/1958	1000.00

# Modelo de Dados Relacional

## Terminologia formal do modelo relacional:

- **Relação** (*tabela*): também chamada de "relação" contém informações sobre um tipo específico de entidade ou conceito, como clientes, produtos ou pedidos.

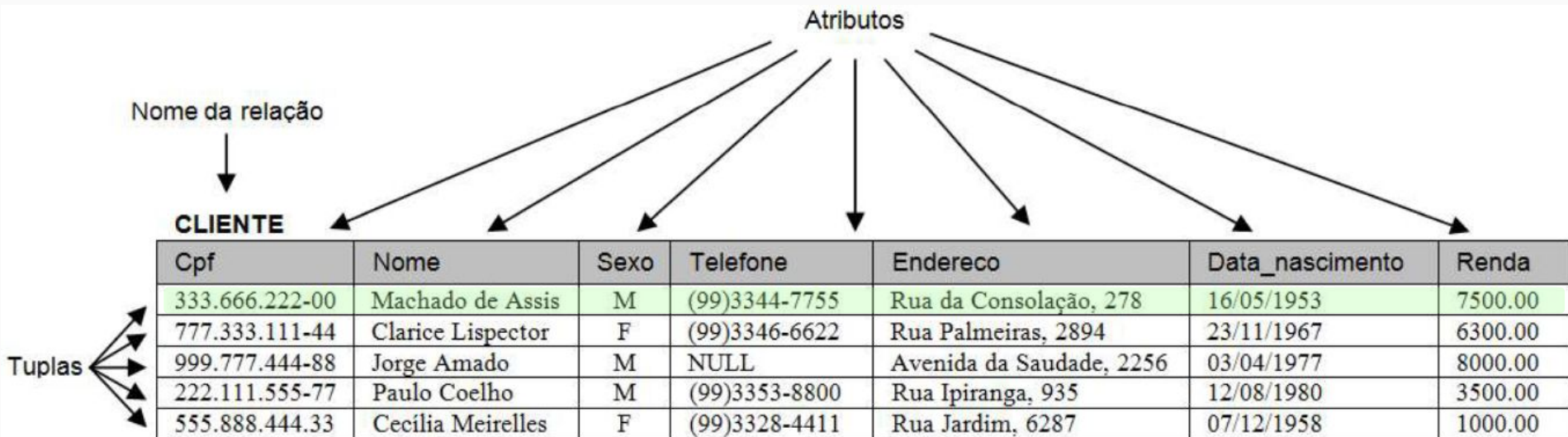
Um cabeçalho da coluna é chamado de **atributo** e a *tabela* é chamada de **relação**.



# Modelo de Dados Relacional

## Terminologia formal do modelo relacional:

- **Tupla (linha):** representa uma entrada de dados individual ou uma instância de uma entidade (tabela). Cada linha é identificada por uma chave primária exclusiva, que a distingue das outras.

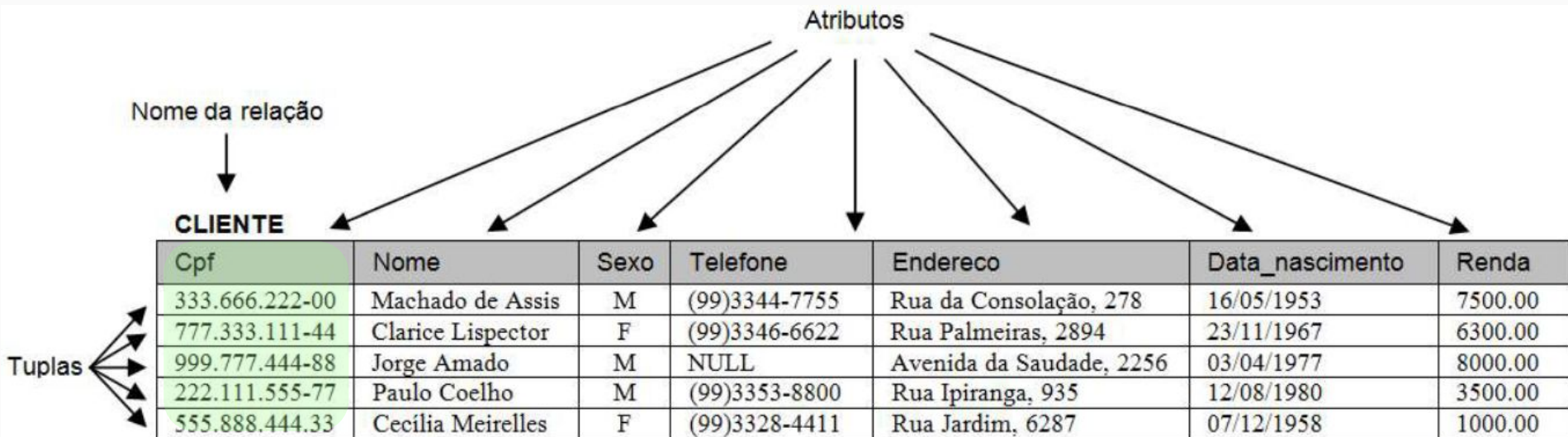




# Modelo de Dados Relacional

## Terminologia formal do modelo relacional:

- **Atributo** (*coluna*): As colunas em uma tabela representam os atributos ou características dos dados. Cada coluna tem um nome que descreve o tipo de informação que ela contém.



# Modelo de Dados Relacional

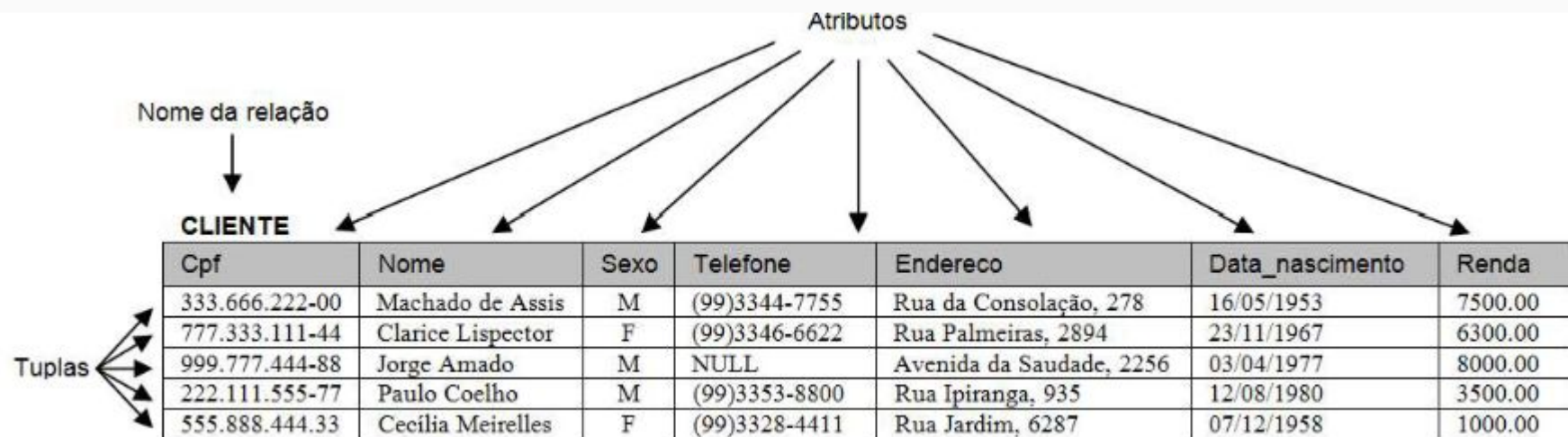
**Esquema Relacional:** estrutura que descreve a maneira como os dados são armazenados e relacionados em um sistema de gerenciamento de banco de dados relacional.

Um **esquema relacional**  $R$  é indicado por  $R(A_1, A_2, A_n)$ , composto de um **nome de relação**  $R$  e uma **lista de atributos**,  $A_1, A_2, A_n$ .

Cada atributo  $A_i$  pode ser composto por um **conjunto de valores** (domínio, representado pela letra  $D$ ) no esquema de relação  $R$ .

O **grau** (ou **aridade**) de uma relação é o número de atributos  $n$  desse esquema de relação (ELMASRI & NAVATHE, 2011).

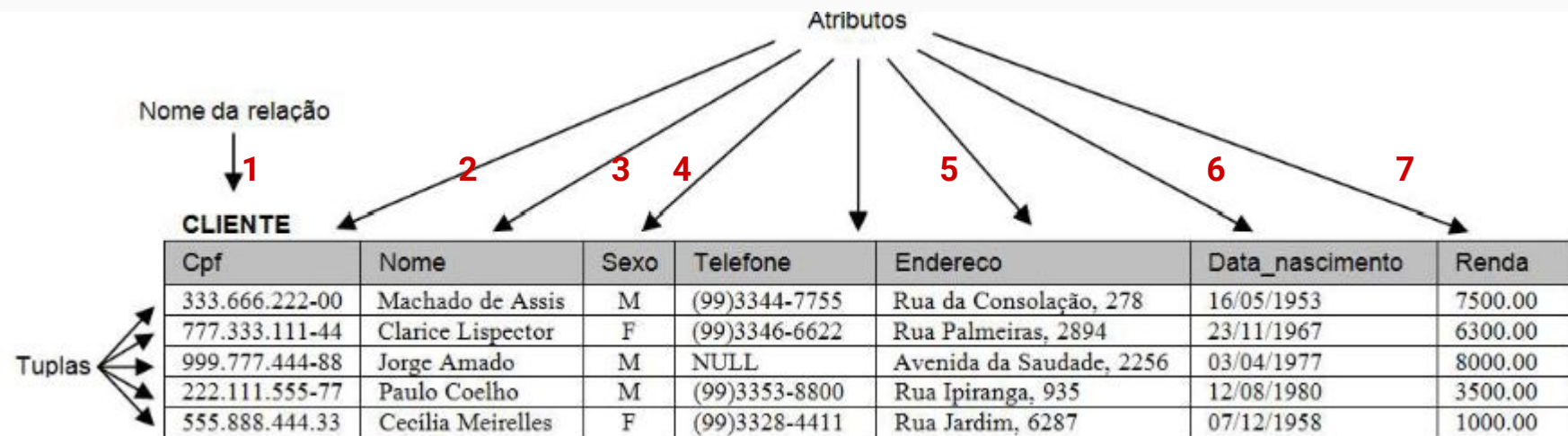
Qual o grau da relação **CLIENTE**?



# Modelo de Dados Relacional

Qual o grau da relação **CLIENTE**?

Relação de **grau sete** que armazena informações sobre cliente, contendo **sete atributos**.



# Restrições em Modelo Relacional

Dentro de um banco de dados relacional, é comum encontrar diversas **relações**, com **tuplas** que frequentemente mantêm relações entre si de diferentes formas.

O estado global do banco de dados representa a união de todos os estados presentes em suas diversas relações em um ponto específico no tempo. Em geral, várias **restrições**, também conhecidas como "**constraints**", são aplicadas aos valores reais presentes no banco de dados.

**Restrições de Integridade:** **especificam regras para garantir a consistência e a qualidade dos dados no banco de dados.**

# Restrições em Modelo Relacional

Restrições em BDs em três categorias principais:

- **Restrições implícitas:** Restrições que são inerentes no modelo de dados.
  - Exemplo: Chave Primária (Primary Key) e Chave Estrangeira (Foreign Key)
- **Restrições explícitas:** Restrições que podem ser expressas diretamente nos esquemas do modelo de dados, em geral especificando-as na DDL.
  - Exemplo: Restrição de Valor Padrão
- **Restrições baseadas na aplicação:** Restrições que não podem ser expressas diretamente nos esquemas do modelo de dados, e, portanto, devem ser expressas e impostas pelos programas de aplicação.
  - Exemplo: Validador de campos, controle de acesso

# Restrições de domínio

**Restrição de domínio:** regra que define os valores aceitáveis que uma coluna específica pode conter, especificando limites, tipos de dados e outras características que os valores nessa coluna devem obedecer para manter a integridade e a consistência dos dados no banco de dados.

Tipos de dados associados ao domínio incluem:

- números inteiros: short integer, integer e long integer;
- números reais: float e double
- caracteres;
- booleanos;
- cadeia de caracteres de tamanho fixo;
- cadeia de caracteres de tamanho variável
- data, hora, moeda ou outros tipos.

## Restrições de chave

No contexto do modelo relacional formal, uma **relação** é representada como um **conjunto de tuplas**, onde todos os elementos de um conjunto devem ser únicos.

**Tuplas** presentes em uma **relação** também devem ser únicas, **não compartilhando a mesma combinação de valores para todos os seus atributos**.

Geralmente, há subconjuntos adicionais de atributos em um esquema de relação R que possuem a propriedade de que duas tuplas não podem apresentar a mesma combinação de valores para esses atributos.

$$t1[SCh] \neq t2[SCh]$$



# Restrições de chave

Qualquer conjunto de atributos SCh desse tipo é chamado de **superchave** do esquema de relação  $R$ .

**Superchave:** **conjunto de um ou mais atributos** (colunas) que pode ser usado para identificar **exclusivamente uma tupla** (linha) em uma **tabela**.

- É uma combinação de atributos que garante a unicidade de cada linha em uma tabela.
- Cada relação tem pelo menos uma superchave padrão.

# Restrições de chave

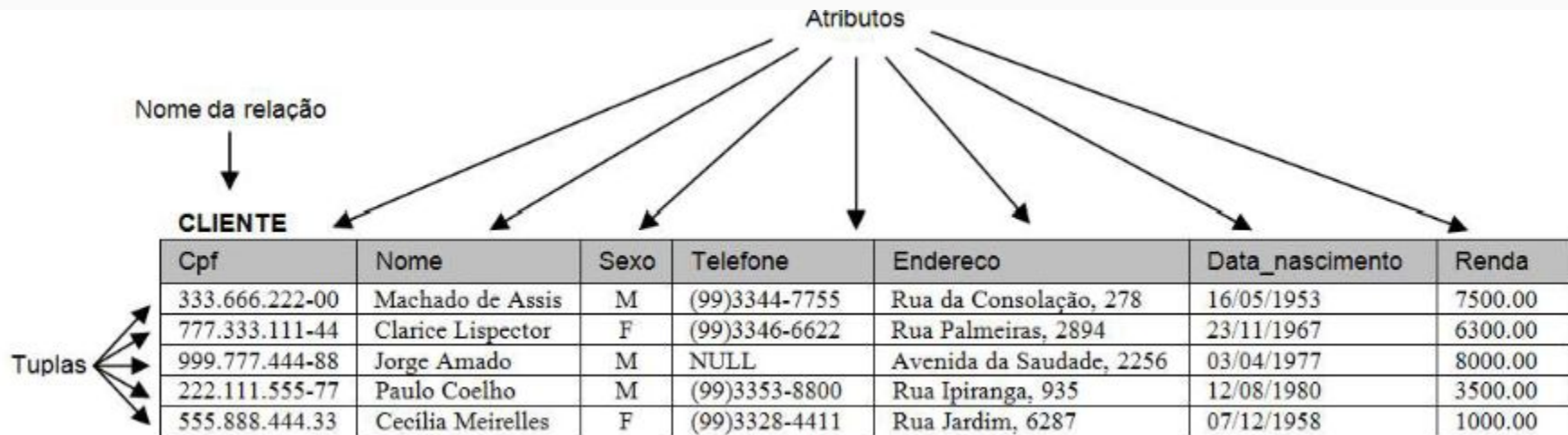
Uma superchave pode ter **atributos redundantes**, de modo que um conceito mais útil é o de **uma chave**, que não tem redundância.

Uma **chave** ( $Ch$ ) de um **esquema de relação**  $R$  é uma **superchave de**  $R$  com a propriedade adicional de que a remoção de qualquer atributo  $A$  de  $Ch$  deixa um conjunto de atributos  $Ch'$  que não é mais uma **superchave de**  $R$ . Logo, uma chave satisfaz duas propriedades:

1. Duas tuplas distintas em qualquer estado da relação não podem ter valores idênticos para (todos) os atributos na chave.
2. Ela é uma superchave mínima ou seja, uma superchave da qual não podemos remover nenhum atributo e ainda mantermos uma restrição de exclusividade na condição 1.

# Restrições de chave

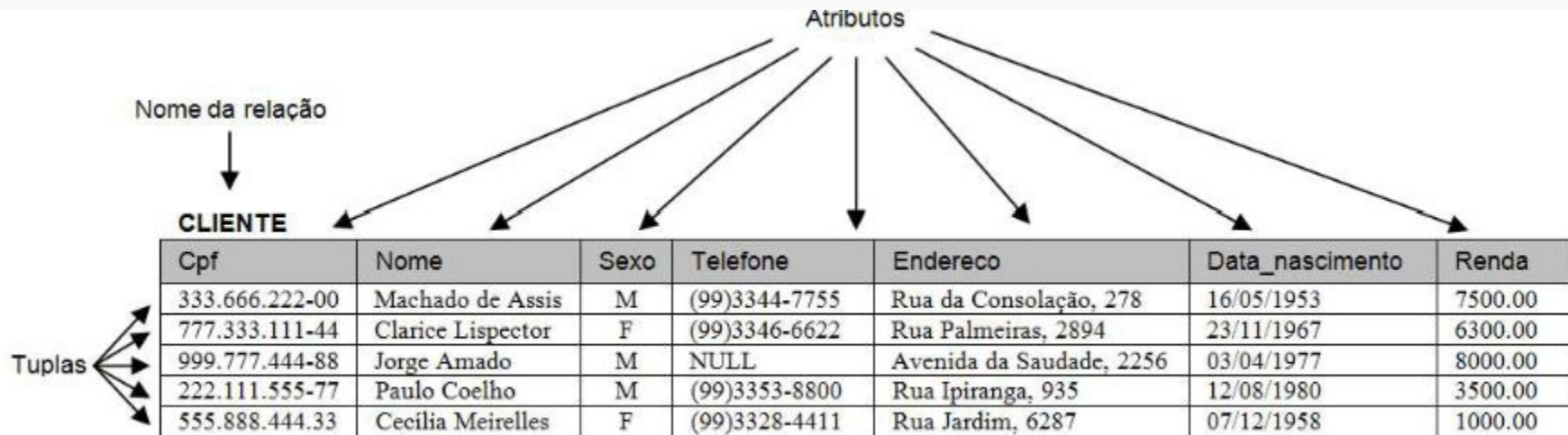
Embora a primeira propriedade se aplique a chaves e superchaves, a segunda propriedade é exigida apenas para chaves. Assim, **uma chave também é uma superchave, mas não o contrário.**



O conjunto de atributos {Nome, Renda} é uma superchave?

# Restrições de chave

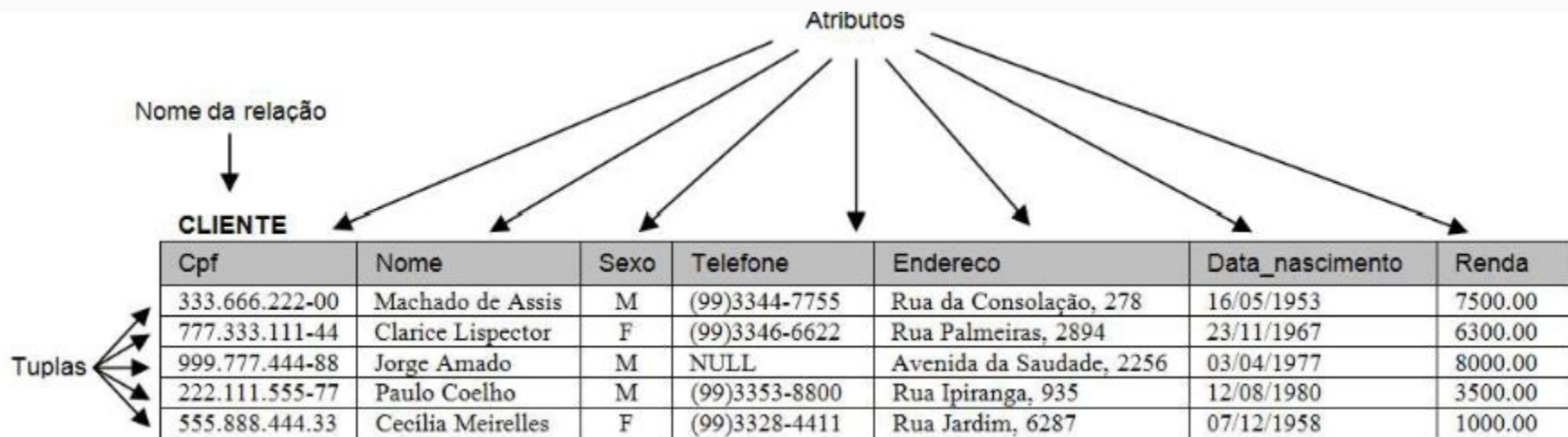
Embora a primeira propriedade se aplique a chaves e superchaves, a segunda propriedade é exigida apenas para chaves. Assim, **uma chave também é uma superchave, mas não o contrário.**



O conjunto de atributos {Cpf, Endereco} é uma superchave?

# Restrições de chave

Embora a primeira propriedade se aplique a chaves e superchaves, a segunda propriedade é exigida apenas para chaves. Assim, **uma chave também é uma superchave, mas não o contrário.**



Qual seria a chave presente na relação CLIENTE?

# Restrições de chave

Um esquema de relação **pode ter mais de uma chave** onde cada uma das chaves é chamada de **chave candidata**.

Exemplo: Relação ALUNO

Quais são as chaves candidatas?

**ALUNO**

<u>Rga</u>	Nome	Cpf	Sexo	Telefone	Endereco	Data_nascimento
2889435	Fernando Braga	222.555.111-99	M	(99)3344-7755	Rua Aquidauana, 325	09/08/1996
8034201	Julia Benson	666.444.000-33	F	(99)3346-6622	Rua Major Capilé, 2350	14/12/1990
6323849	Roberto Passos	999.888.777-66	M	NULL	Avenida da Amizade, 560	29/07/1986
1702136	Paulo Lima	111.555.333-00	M	(99)3353-8800	Rua Ipiranga, 1925	10/10/1992
5466972	Alice Nogueira	444.666.222-33	F	(99)3328-4411	Rua das Camélias, 890	12/04/1985

## Restrições de chave

Quando um esquema de relação tem várias chaves candidatas, a escolha de uma para se tornar a chave primária é um tanto arbitrária; porém normalmente é melhor escolher uma chave primária com um único atributo ou um pequeno número de atributos.

As outras chaves candidatas são designadas como **chaves únicas (unique keys)**, e não são sublinhadas.

# Restrições sobre valores null

**Restrição sobre valores NULL:** especifica se valores NULL são permitidos ou não para atributos.

Considerando ainda a relação ALUNO, se cada tupla de ALUNO precisa ter um valor válido, diferente de NULL, para o atributo Nome, então Nome de ALUNO é restrito a ser NOT NULL.

**ALUNO**

<u>Rga</u>	Nome	Cpf	Sexo	Telefone	Endereco	Data_nascimento
2889435	Fernando Braga	222.555.111-99	M	(99)3344-7755	Rua Aquidauana, 325	09/08/1996
8034201	Julia Benson	666.444.000-33	F	(99)3346-6622	Rua Major Capilé, 2350	14/12/1990
6323849	Roberto Passos	999.888.777-66	M	NULL	Avenida da Amizade, 560	29/07/1986
1702136	Paulo Lima	111.555.333-00	M	(99)3353-8800	Rua Ipiranga, 1925	10/10/1992
5466972	Alice Nogueira	444.666.222-33	F	(99)3328-4411	Rua das Camélias, 890	12/04/1985



# Restrições de integridade

O esquema pode incluir restrições de integridade que especificam regras para garantir a consistência e a qualidade dos dados no banco de dados.

**Restrição de integridade:** nenhum valor de chave primária pode ser NULL. Por exemplo, o valor da chave primária é usado para identificar tuplas individuais em uma relação, portanto, valores NULL para a chave primária implica a não identificação de algumas tuplas.

## ALUNO

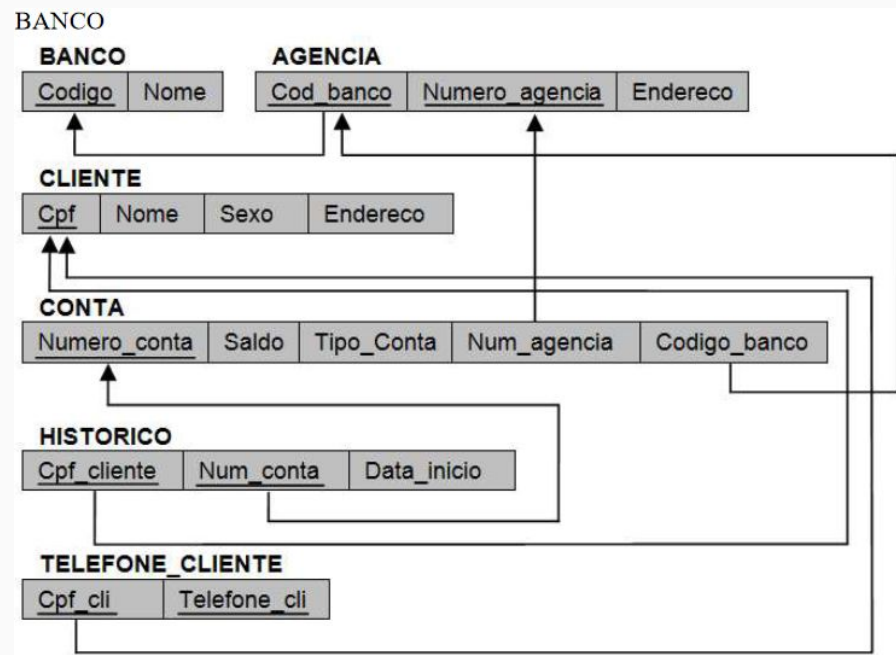
<u>Rga</u>	Nome	Cpf	Sexo	Telefone	Endereco	Data_nascimento
2889435	Fernando Braga	222.555.111-99	M	(99)3344-7755	Rua Aquidauana, 325	09/08/1996
8034201	Julia Benson	666.444.000-33	F	(99)3346-6622	Rua Major Capilé, 2350	14/12/1990
6323849	Roberto Passos	999.888.777-66	M	NULL	Avenida da Amizade, 560	29/07/1986
1702136	Paulo Lima	111.555.333-00	M	(99)3353-8800	Rua Ipiranga, 1925	10/10/1992
5466972	Alice Nogueira	444.666.222-33	F	(99)3328-4411	Rua das Camélias, 890	12/04/1985

# Restrições de integridade referencial

**Restrição de integridade referencial:** especificada entre duas relações e usada para manter a consistência entre tuplas nas duas relações.

Uma **tupla** em uma **relação** que referencia outra **relação** precisa se referir a uma tupla existente nessa relação.

Exemplo: o atributo Cod\_banco de AGENCIA fornece o código do banco que a agência pertence; logo, seu valor em cada tupla de AGENCIA precisa combinar com o valor de Codigo de alguma tupla na relação BANCO.



## Restrições de integridade referencial

Para definir a integridade referencial de maneira mais formal, primeiro estabelecemos o conceito de uma **chave estrangeira** (*foreign key*).

As condições para uma chave estrangeira especificam a restrição de integridade referencial entre os dois esquemas de relação R1 e R2.

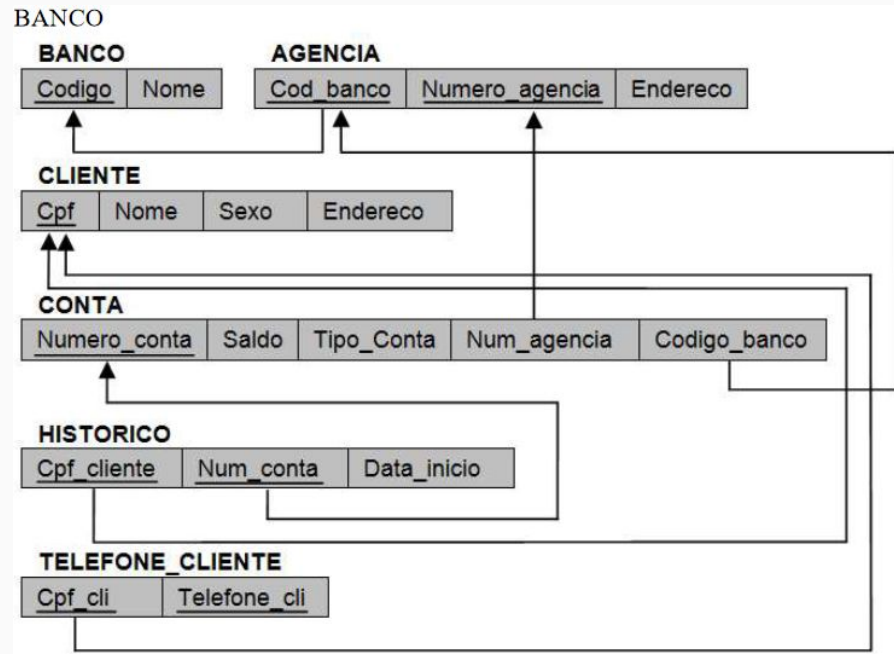
Um conjunto de atributos *ChE* no esquema de relação R1 é uma chave estrangeira de R1 que referencia a relação R2. Assim, os atributos em ChE têm o mesmo domínio (ou domínios) que os atributos de chave primária ChP de R2; diz-se que os atributos ChE referenciam ou referem-se à relação R2.

## Restrições de integridade referencial

Nessa definição, R1 é chamada de relação que referencia e R2 é a relação referenciada. Se essas condições se mantiverem, diz-se que é mantida uma restrição de integridade referencial de R1 para R2. Em um banco de dados de muitas relações, normalmente existem muitas restrições de integridade referencial.

# Restrições de integridade referencial

Podemos exibir em forma de diagrama as restrições de integridade referencial, desenhando um arco direcionado de cada chave estrangeira para a relação que ela referencia. Para ficar mais claro, a ponta da seta pode apontar para a chave primária da relação referenciada.



# Restrições de integridade referencial

Todas as restrições de integridade deverão ser especificadas no esquema de banco de dados relacional (ou seja, definidas como parte de sua definição) se quisermos impor essas restrições sobre os estados do banco de dados. Logo, a DDL inclui meios para especificar os diversos tipos de restrições de modo que o SGBD possa impô-las automaticamente. A maioria dos SGBDs relacionais admite restrições de chave, integridade de entidade e integridade referencial. Essas restrições são especificadas como uma parte da definição de dados na DDL.

