

Modelagem de Dados

Introdução à Conceitos de Modelagem e Acesso a Data Sets em Banco de Dados

Prof. Dr. Pedro Luiz Pizzigatti Corrêa

22 de Setembro de 2020

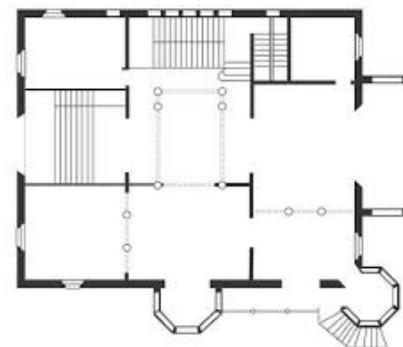
Agenda

- Introduction to data models
- Relational data model – SQL
- Multidimensional data models
- NoSQL data models



Introduction to data models

O conceito de Modelo



Fonte: imagens disponíveis na internet e marcadas para reutilização

O conceito de Modelo

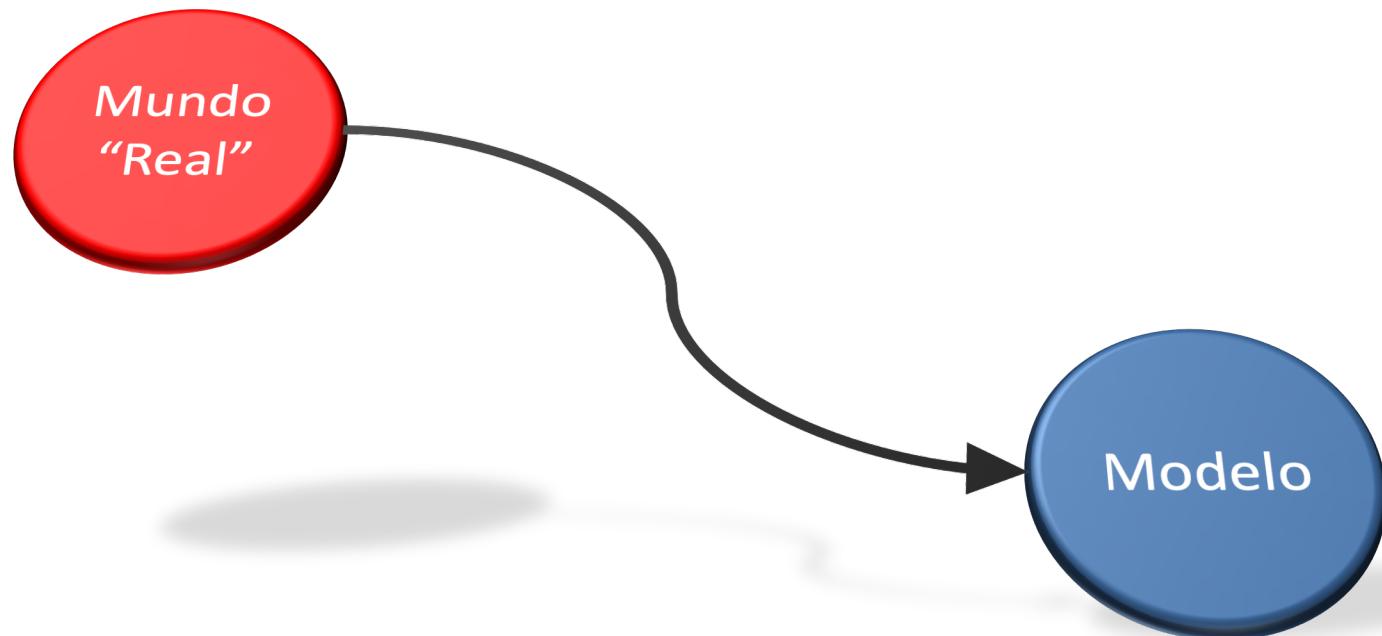
- Modelo:
 - Desenho ou imagem que representa o que se pretende desenhar, pintar ou esculpir;
 - Aquele a quem se procura reproduzir nas ações e nas maneiras;
 - Versão, em pequena escala, de um objeto que se pretende executar em grande escala;
 - Artigo manufaturado, com características específicas;
 - O exemplar de uma determinada população.

Fonte: Dicionários Michaelis Online. Disponível em <http://michaelis.uol.com.br/>. Acesso em: 30 Março 2016.

O conceito de Modelo

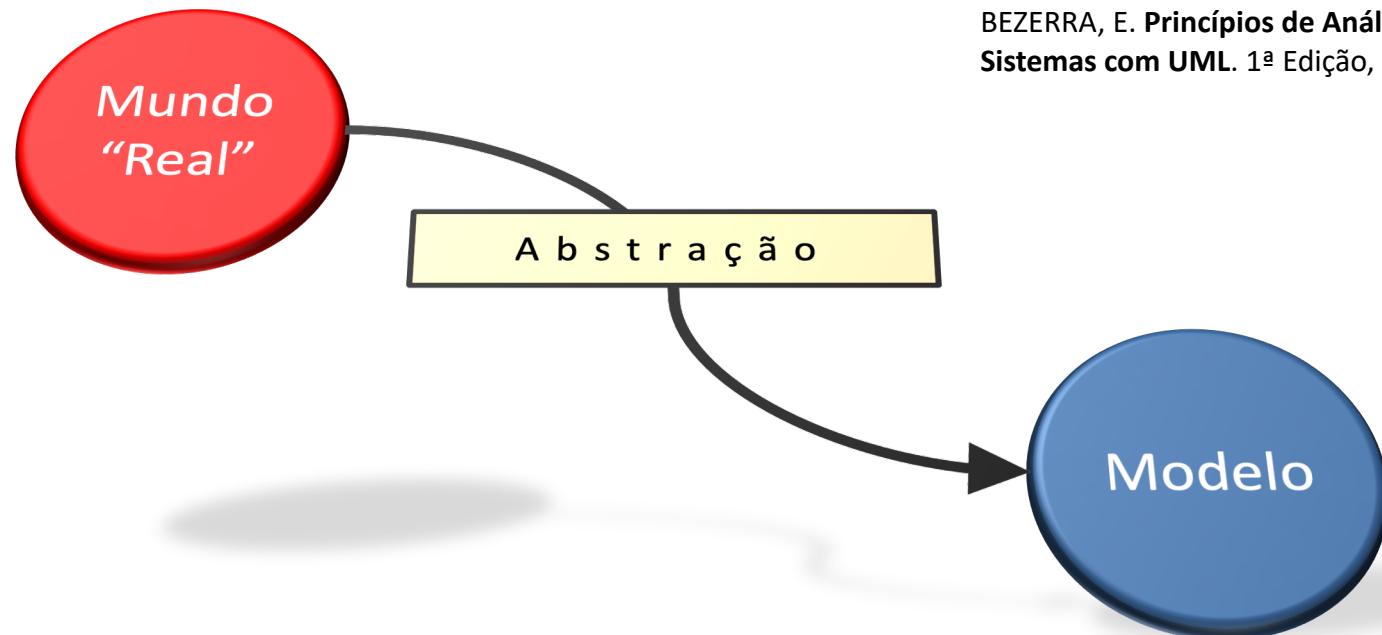
- **Modelo:**
 - É a representação idealizada de uma determinada “realidade”.

BEZERRA, E. **Princípios de Análise e Projeto de Sistemas com UML**. 1^a Edição, Elsevier, 2014.



O conceito de Abstração

- Abstração:
 - Linha de raciocínio usada para suprimir detalhes de um modelo, destacando características que venham facilitar o seu entendimento. A abstração depende do contexto.



BEZERRA, E. Princípios de Análise e Projeto de Sistemas com UML. 1ª Edição, Elsevier, 2014.

O modelo de dados relacional

O modelo de dados relacional

- O modelo relacional:
 - Representa os dados por meio de um conjunto de relações, constituídas de uma série de atributos, os quais definem as propriedades relativas ao elemento que representam.

LAUDON K. C.; LAUDON J. P. **Management Information Systems**, 2007.

O modelo de dados relacional

- O modelo relacional:
 - A Linguagem de Definição de Dados (DDL) é usada para definição do esquema do Banco de Dados. Exemplo:

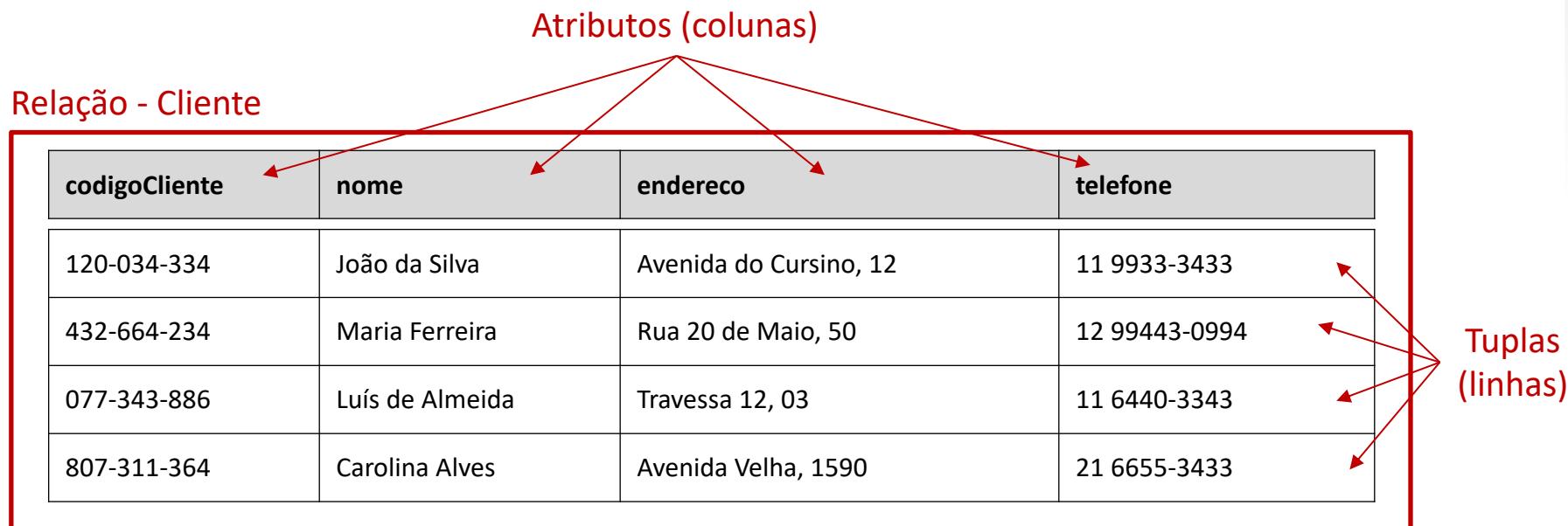
```
create table conta (
    numeroConta  char(10),
    nomeSegmento char(10),
    saldoAtual    decimal(15,2)
    primary key(numeroConta)
)
```

O modelo de dados relacional

- O modelo relacional:
 - O compilador DDL gera um conjunto de tabelas em um Dicionário de Dados;
 - O Dicionário de dados contém metadados descrevendo:
 - O Esquema do banco de dados;
 - A estrutura de armazenamento e os métodos de acesso aos dados;
 - As regras de integridade:
 - Domínio de restrição
 - Integridade referencial;
 - Autorizações para acesso aos dados.

O modelo de dados relacional

- Exemplo de dados representados em uma tabela no Modelo Relacional:



O modelo de dados relacional

O modelo relacional:

- A Linguagem de Acesso ao Banco de Dados:
Structured Query Language (SQL). Exemplo:

Select name from Client

Relação - Cliente

codigoCliente	nome	endereco	telefone
120-034-334	João da Silva	Avenida do Cursino, 12	11 9933-3433
432-664-234	Maria Ferreira	Rua 20 de Maio, 50	12 99443-0994
077-343-886	Luís de Almeida	Travessa 12, 03	11 6440-3343
807-311-364	Carolina Alves	Avenida Velha, 1590	21 6655-3433



nome
João da Silva
Maria Ferreira
Luís de Almeida
Carolina Alves

O modelo Entidade-Relacionamento (E-R)

- O modelo Entidade-Relacionamento:
 - Foi proposto por Peter Chen em um artigo de 1976;
 - Representa um domínio por meio de uma coleção de **Entidades e Relacionamentos**, sendo que:
 - Cada Entidade representa uma “**coisa**” ou um “**objeto**” do mundo **real** capaz de existir por si próprio;
 - A Entidade é **descrita** e se diferencia de outras entidades **por meio de seus atributos**;
 - O Relacionamento é representado pela **associação entre várias Entidades**, que por sua vez, compartilham atributos em comum;

O modelo Entidade-Relacionamento (E-R)

- É importante diferenciar o Modelo Relacional do Modelo Entidade-Relacionamento:
 - Modelo Relacional:
 - Proposto por Edgar Codd;
 - Representa os dados por meio de relações e atributos, que por sua vez, definem as propriedades relativas a um elemento;
 - É o conceito usado para projetar o “modelo de dados”;
 - Modelo Entidade-Relacionamento:
 - Proposto por Peter Chen;
 - Representa o domínio por meio de uma coleção de entidades e relacionamentos;
 - A representação gráfica do domínio é chamada: Diagrama Entidade-Relacionamento(DE-R).

Primitivas

- Entidade:
 - É uma “coisa” ou um “objeto” do mundo real capaz de existir por si próprio;
 - Pode ser tanto um elemento físico quanto um elemento conceitual.

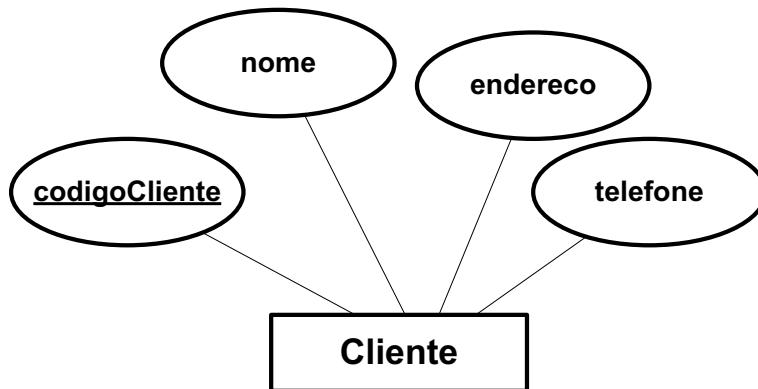
Cliente

Fatura

Publicação

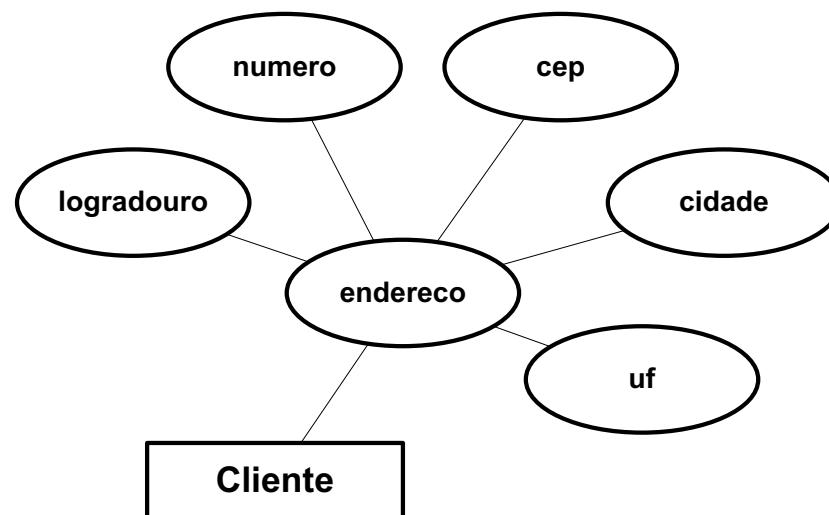
Primitivas

- Atributo:
 - Descreve uma determinada Entidade por meio de suas características;
 - Cada atributo possui um valor próprio, o qual constituirá a maior parte da base de dados.



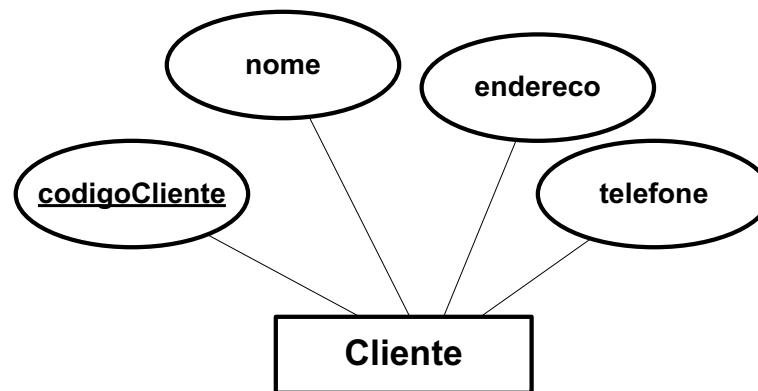
Primitivas

- Atributo Composto:
 - Atributo capaz de ser dividido em duas ou mais partes, cada uma delas com seu próprio significado.



Primitivas

- Conceito de Instância:
 - Representa o conteúdo atual do banco de dados.



Primitivas

- Conceito de Instância:
 - Representa o conteúdo atual do banco de dados.

codigoCliente	nome	endereco	telefone	
120-034-334	João da Silva	Avenida do Cursino, 12	11 9933-3433	← Instância 01
432-664-234	Maria Ferreira	Rua 20 de Maio, 50	12 99443-0994	← Instância 02
077-343-886	Luís de Almeida	Travessa 12, 03	11 6440-3343	← Instância 03
807-311-364	Carolina Alves	Avenida Velha, 1590	21 6655-3433	← Instância 04

Primitivas

- Relacionamento:
 - Representa a associação entre várias Entidades;
 - Ocorre quando as entidades envolvidas na associação possuem um ou mais atributos em comum.

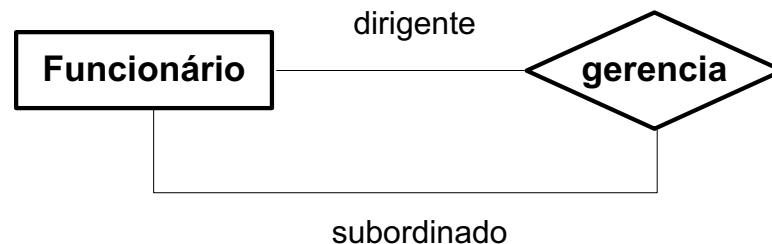


Primitivas

- Graus de Relacionamento:
 - Refere-se ao número de entidades participantes de um relacionamento:
 - Relação Unária
 - Relação Binária
 - Relação Ternária

Primitivas

- Graus de Relacionamento:
 - Relação unária (ou auto-relacionamento): ocorre quando a entidade se relaciona com ela mesma. Exemplo:



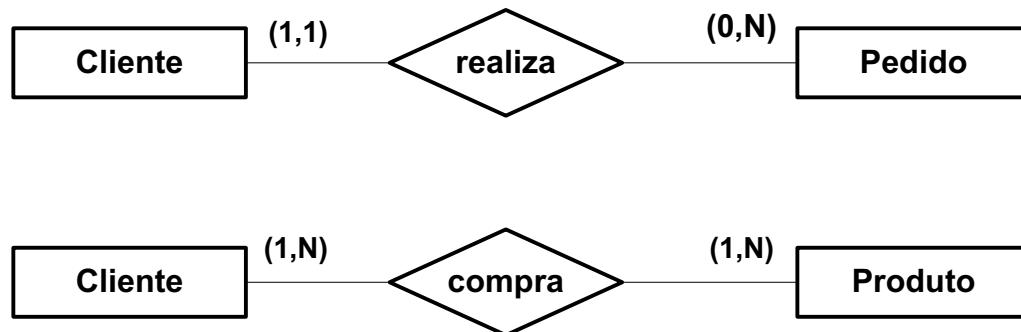
Primitivas

- Graus de Relacionamento:
 - Relação binária: ocorre quando duas entidades diferentes se relacionam. Exemplo:



Primitivas

- Cardinalidade:
 - Determina os limites mínimos e máximos de instâncias de uma Entidade em um Relacionamento;



Notação	Mínimo	Máximo
(1 , 1)	Um	Um
(0 , N)	Zero	Muitos
(1 , N)	Um	Muitos
(0 , 1)	Zero	Um
N	Zero	Muitos
1	Um	Um

Primitivas

- Atributo Chave:
 - Identificam as tuplas de cada Entidade;
 - Podem pertencer à própria Entidade ou podem ter origem em outras Entidades do modelo;
 - Estabelece os conceitos de Chave Primária e Chave Estrangeira;

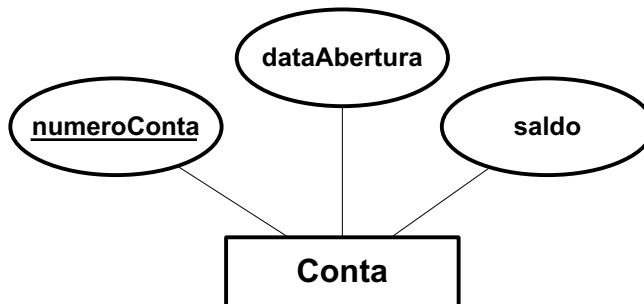
Primitivas

- Chave Primária:
 - Conjunto de um ou mais atributos, cujos valores são únicos para uma determinada Entidade;
 - É usado na identificação das linhas (tuplas) de uma tabela;
 - No modelo E-R, a Chave Primária (PK) é representada por um atributo sublinhado.

O SGBD aplica uma restrição existencial aos valores da Chave Primária, garantindo que aquele valor seja único para todos os registros pertencentes à mesma Entidade.

Primitivas

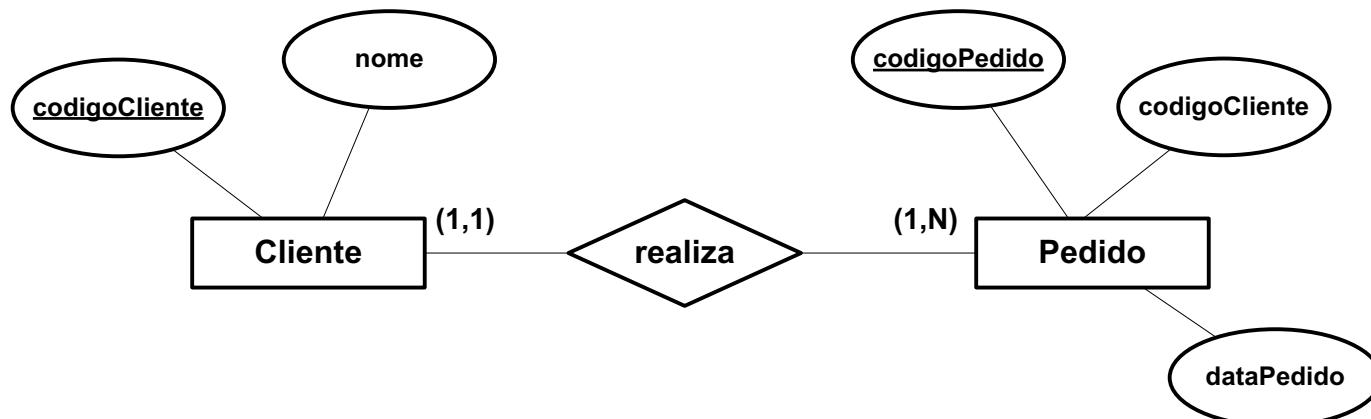
- Chave Primária:



Conta		
<u>numeroConta</u>	dataAbertura	saldo
00328351	31/10/2004	1.500,00
00892346	20/08/2005	1.500,00
00336480	20/08/2005	759,23
00428375	25/07/2013	0,00
00429572	10/12/2013	0,00

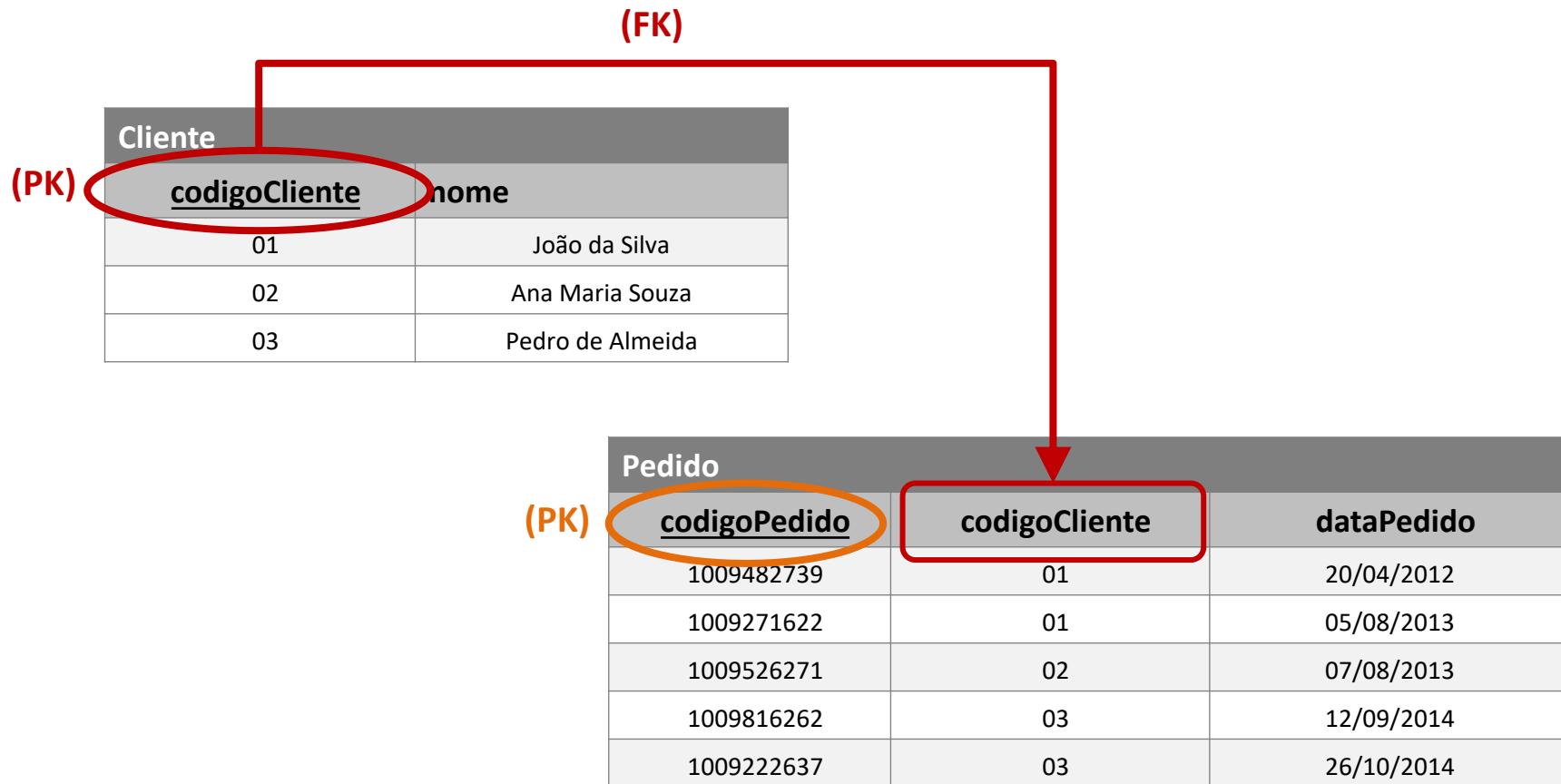
Primitivas

- Chave Estrangeira:
 - A chave estrangeira (FK) é criada quando o valor da chave primária da Entidade A é usada para estabelecer um relacionamento com uma Entidade B.



Primitivas

- Chave Estrangeira

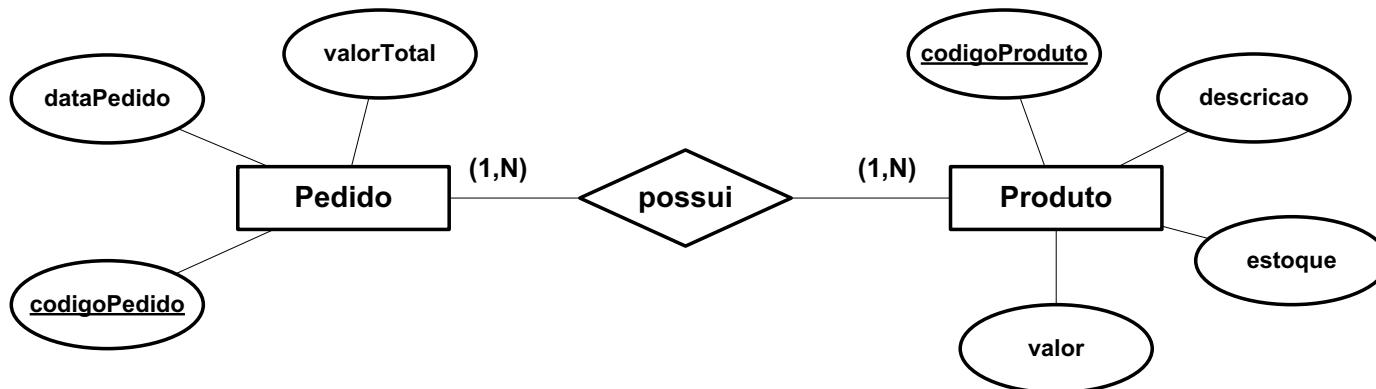


Primitivas

- Chave Composta:
 - É um tipo particular de Chave Primária;
 - Ocorre quando é necessário utilizar o valor de dois ou mais atributos chave para identificar unicamente o registro;
 - Implica na dependência existencial das entidades envolvidas no relacionamento.

Primitivas

- Chave Composta:

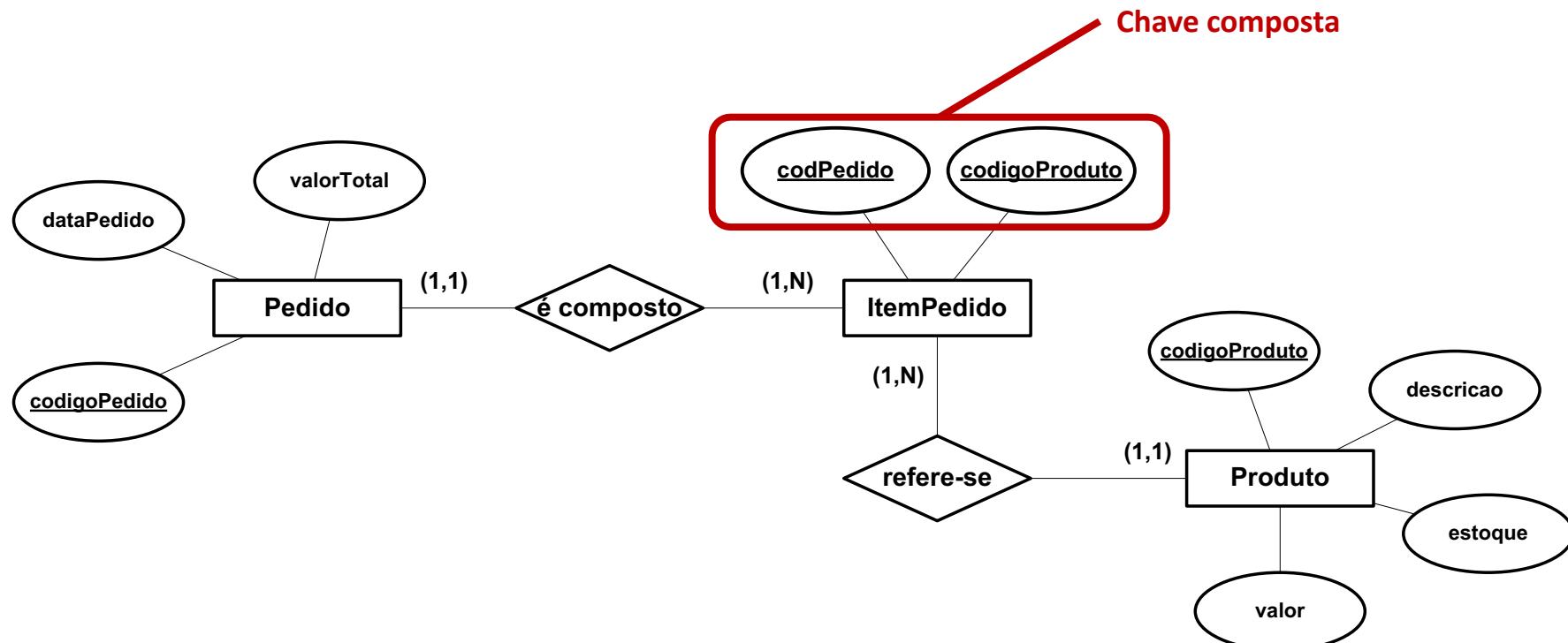


Pedido		
<u>codPedido</u>	<u>data</u>	<u>valor</u>
0001	31/10/2004	R\$ 2.780,00
0002	20/08/2005	R\$ 2.250,00
0003	20/08/2005	R\$ 10.950,00

Produto			
<u>codProduto</u>	<u>descricao</u>	<u>valor</u>	<u>estoque</u>
50001	Camiseta	R\$ 50,00	100 un
50002	Calça	R\$ 89,00	200 un
50003	Bermuda	R\$ 50,00	300 un
50004	Meia	R\$ 15,00	600 un

Primitivas

- Chave Composta:



Primitivas

- Chave Composta

Pedido		
	data	valor
0001	31/10/2004	R\$ 2.780,00
0002	20/08/2005	R\$ 2.250,00
0003	20/08/2005	R\$ 10.950,00

(PK)
Chave composta

Produto				
	codProduto	descricao	valor	estoque
50001	Camiseta	R\$ 50,00	100 un	
50002	Calça	R\$ 89,00	200 un	
50003	Bermuda	R\$ 50,00	300 un	
50004	Meia	R\$ 15,00	600 un	

Item Pedido		
	codPedido	codProduto
	0001	50001
	0001	50003
	0002	50003
	0003	50001
	0003	50002
	0003	50004

Multidimensional data models

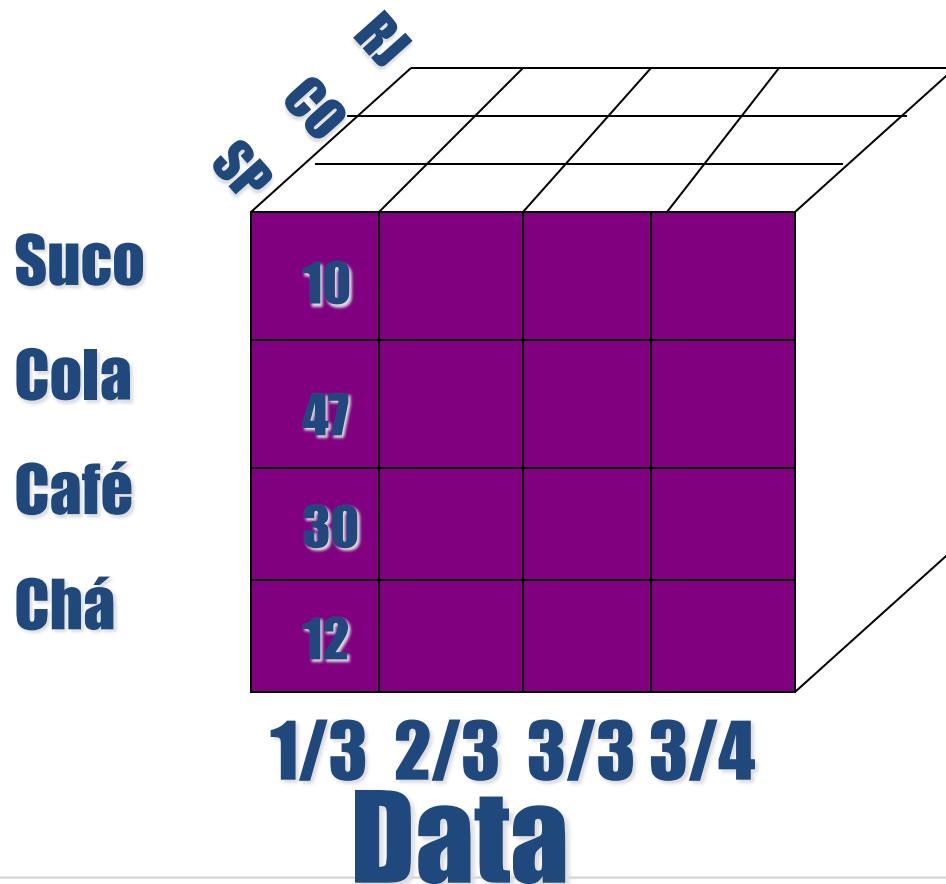
Modelagem Multidimensional

Modelo de Dados Multidimensional:

- BD é um conjunto de fatos (pontos) num espaço multidimensional;
- Um fato tem uma dimensão de medida:
 - quantidade que é analisada. Ex: venda, orçamento
- Um conjunto de dimensões sobre qual o fato é analisado
 - ex: loja, produto, data associada com a venda
- Dimensões formam um sistema de coordenadas
- Cada dimensão tem um conjunto de *atributos*
 - Ex: cidade e bairro da loja.

Modelagem Multidimensional

Dado multidimensional



Volume
de
Vendas
como
uma
função
do
tempo,
cidade e
produto

Operações Multidimensionais

Operações no modelo multidimensional:

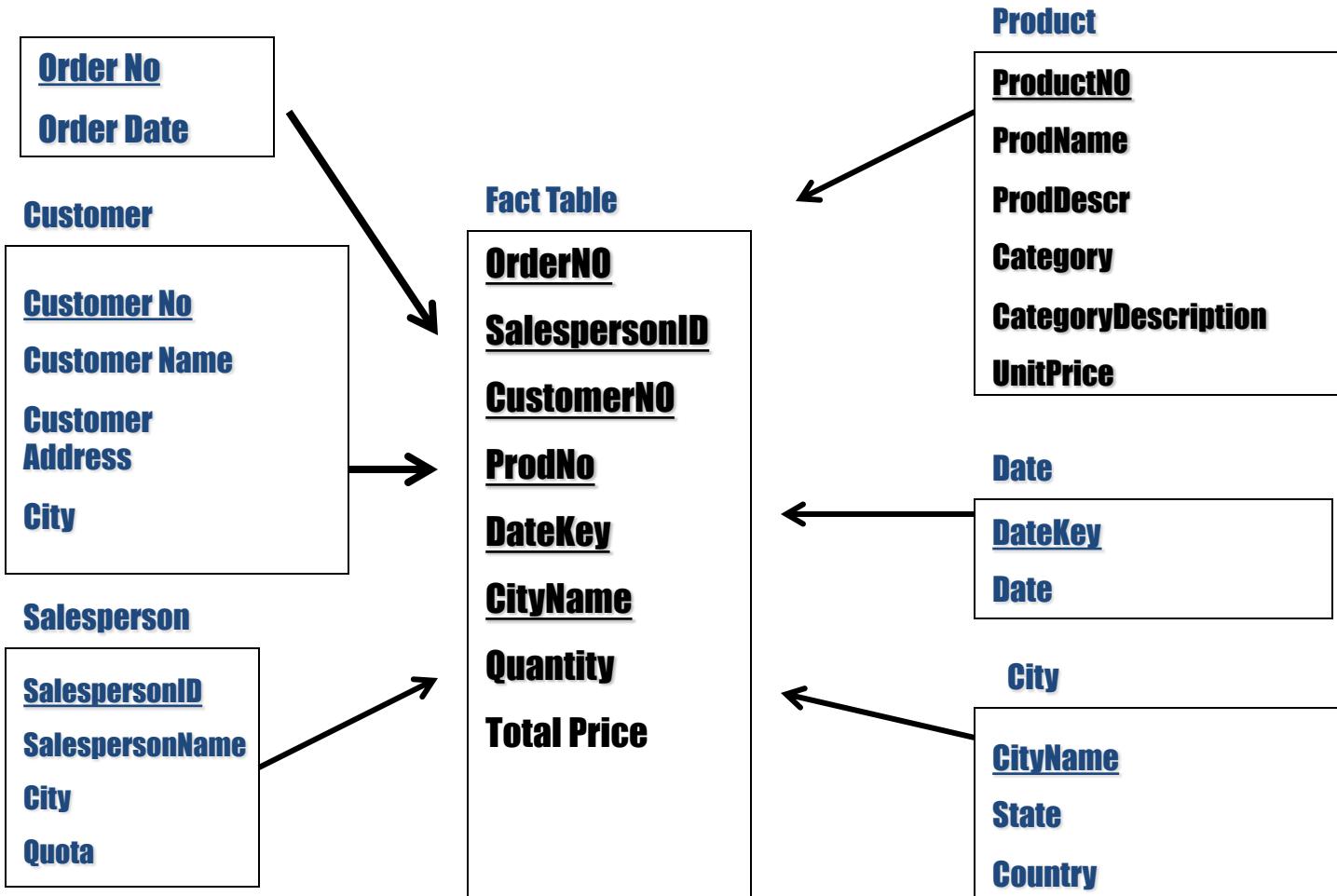
- Agregação (*roll-up*)
 - redução de dimensão: ex: vendas total por cidade;
- Seleção (*slice*) define um subcubo
 - ex:, vendas onde cidade = SP and data = 15/1/04
- Navegação sobre detalhe dos dados (*drill-down*)
- Operações de visualização (e.g., *Pivot* ou rotação)

Esquema de Warehouse

- Técnicas de E-R não se aplicam;
- Projeto reflete a visão multidimensional
 - Esquema Estrela
 - Esquema Snowflake
 - Esquema de constelação de Fatos

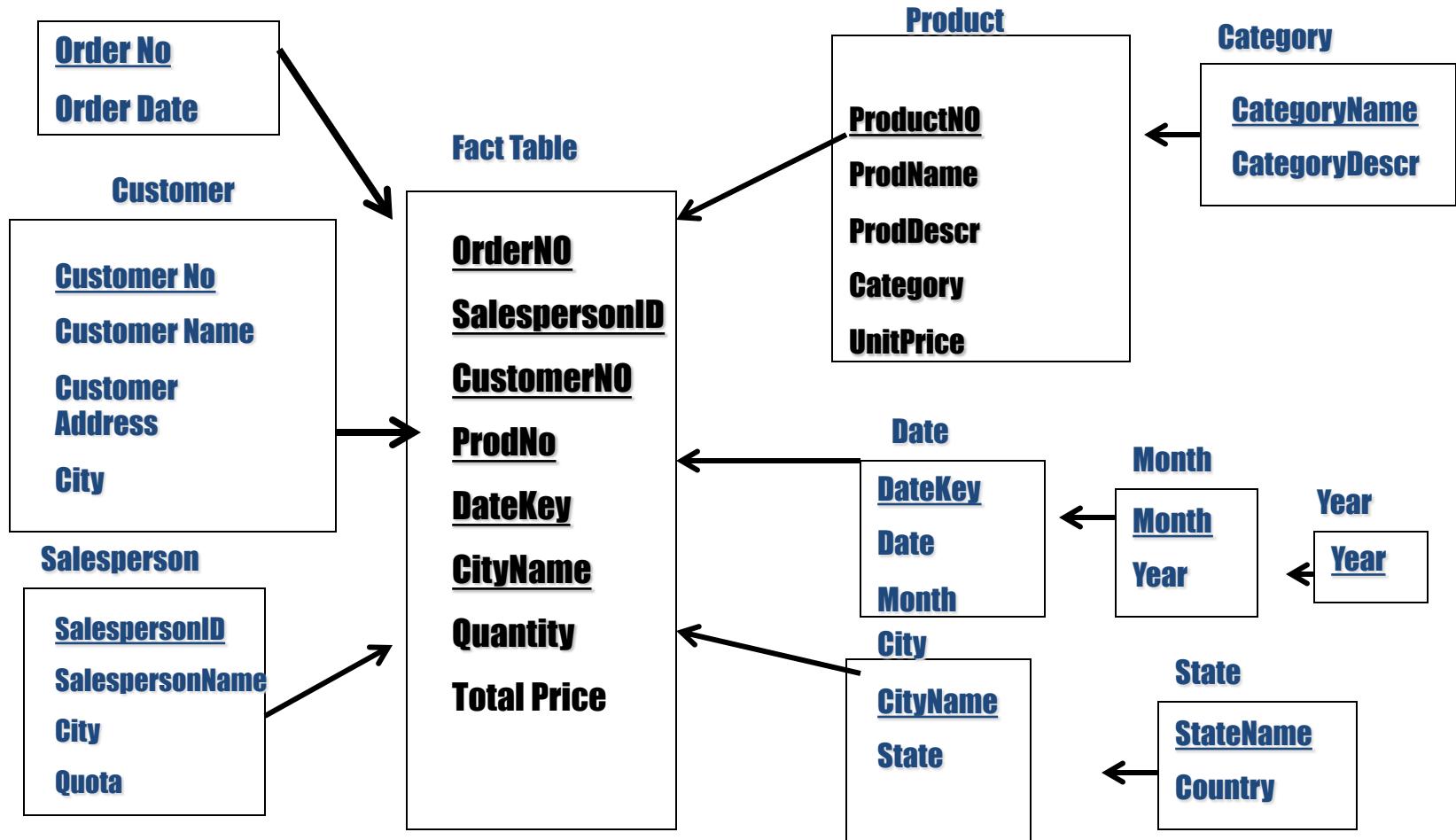
Modelagem Multidimensional

Exemplo de Esquema Estrela:



Modelagem Multidimensional

Exemplo de Esquema Snowflakes



Análise: Navegação

Volume
de
Vendas
como
uma
função
do
tempo,
cidade e
produto

- Agregação (*roll-up*)
 - Seleção (*slice*) define um subcubo
 - Navegação sobre detalhe dos dados (*drill-down*)
 - *Pivot* ou rotação
-
- | | 1/3 | 2/3 | 3/3 | 3/4 |
|------|-----|-----|-----|-----|
| Suco | 10 | | | |
| Cola | | 47 | | |
| Café | | | 30 | |
| Chá | | | | 12 |

Análise: Data Mining

- Objetivo: Encontrar padrões de dados,
- Usado em conjunto com DW e/ou Big Data
- Resultado de mineração, pode encontrar:
 - Regras de associação. Ex: se um cliente compra um console de videogame, ele também compra outro acessório;
 - Padrões sequenciais. Ex: um cliente que compra mais que duas vezes num pequeno período deverá comprar pelo menos uma vez no natal;
 - Árvores de classificação. Ex: clientes podem ser classificados de acordo com o tipo de itens que compram

Uso dessas informações, para por exemplo:

- Planejar a localização de filiais;
- executar promoções de vendas;
- planejamento de marketing sazonal.

Aplicações de Data Mining:

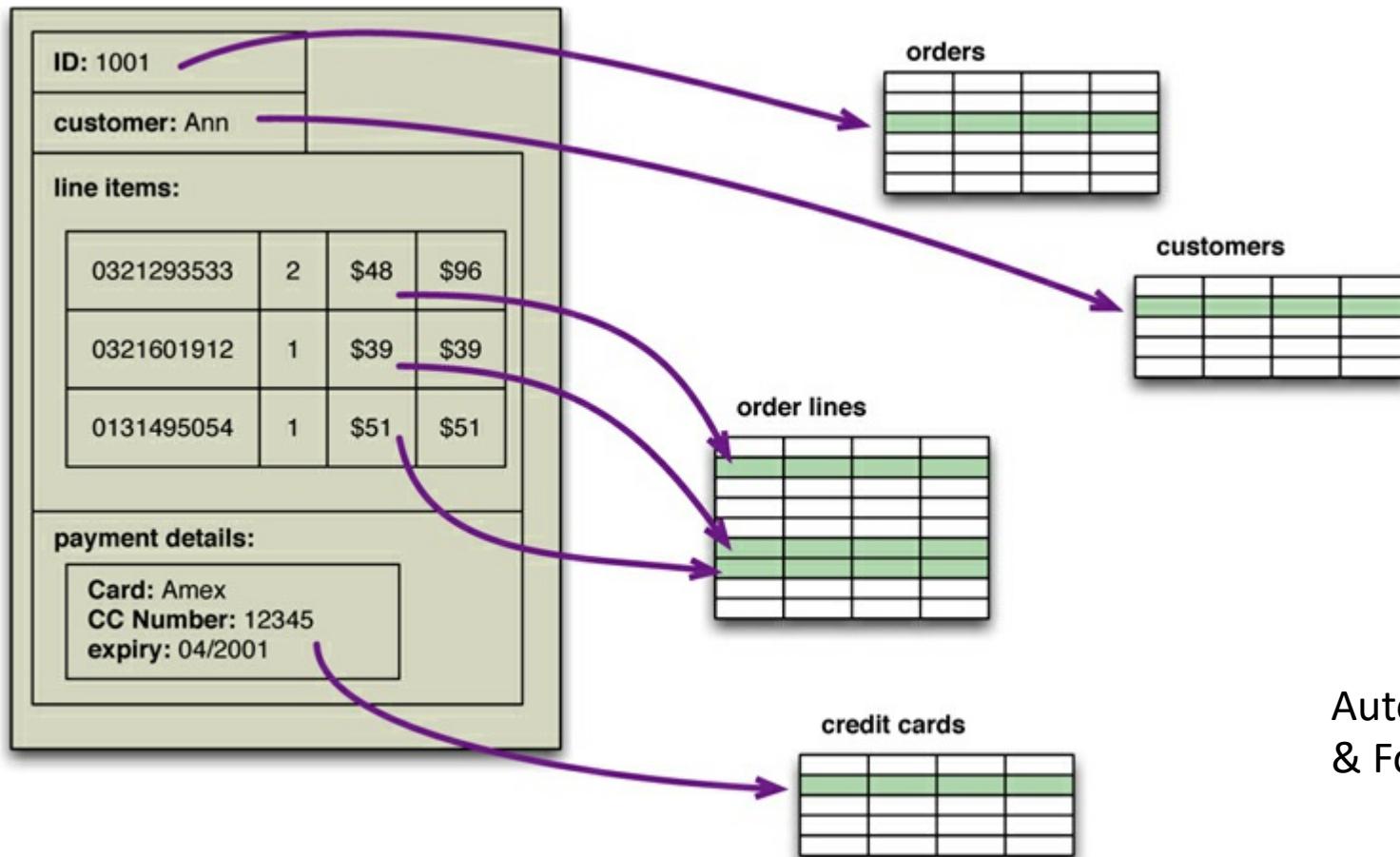
- Marketing: padrão de consumo, determinação de estratégia de marketing, segmentação de clientes, etc.
- Finança: análise de crédito de clientes, detecção de fraudes, etc.
- Produção: automóveis baseado nos requisitos dos clientes, otimização de recursos, etc.
- Saúde: análise da efetividade de um tratamento, análise de efeitos colaterais de drogas, etc.

NoSQL data models

Incompatibilidade dos Modelos de Dados de Memória e os Modelos de Armazenamento

- Necessidade de reorganização dos dados manipulados em memória pelos programas (mais rica) para uma representação relacional (banco de dados)
- **Impedância** de modelos usados pelas aplicações e os modelos de persistência dos dados.

Incompatibilidade do Modelos de Dados de Memória e os Modelos de Armazenamento



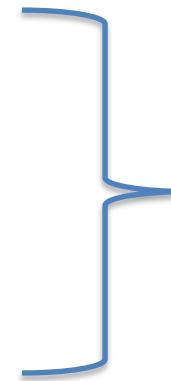
Autor: Sadalage
& Fowler, 2013

Características de Banco de Dados Big Data

- Não utilizam o Modelo Relacional;
- Demanda de Processamento e Armazenamento em *Clusters (mesma estrutura)/Grid (arquiteturas e sabores diferentes)*;
- Apropriados para aplicações WEB;
- Não tem um esquema;
- Também conhecidos como NoSQL.

Categorias de Soluções Big Data

- Chave-valor. Ex: Riak, Redis;
- Documento. Ex: MongoDB;
- Famílias de Colunas. Ex:
Cassandra, HBase
- Grafos. Ex: FlockDB, Neo4J



**Orientação a
Agregados**

Agregado: conjunto de objetos relacionados que desejamos tratar como unidade.

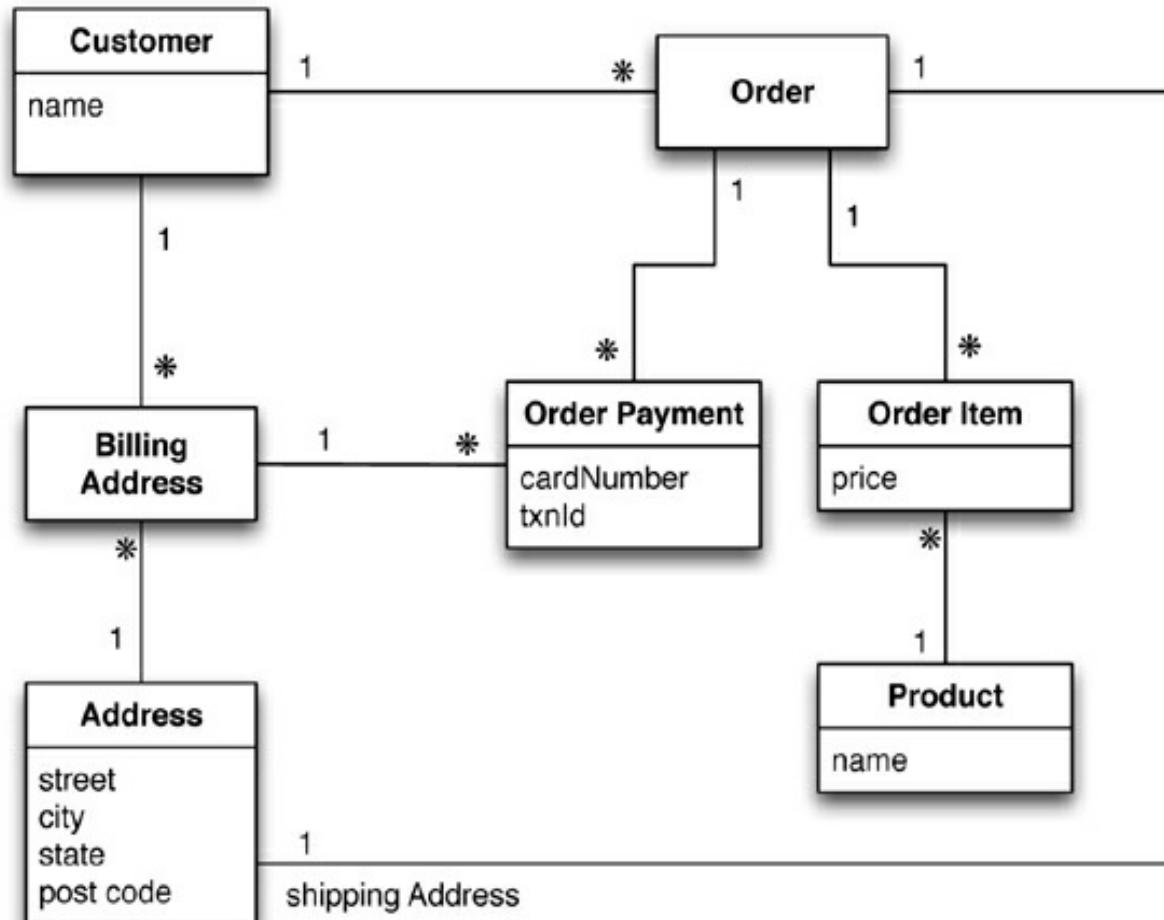
Características dos Agregados

- Unidade de armazenamento de dados e gerenciamento de consistência;
- Facilita a execução de banco de dados num *cluster*, pois constitui uma unidade natural de fragmentação e replicação;
- São mais simples de serem tratados pelas aplicações uma vez que lidam com os dados por meio de uma estrutura agregada.
- Estrutura ACID (**A**tomicidade, **C**onsistência, **I**solamento e **D**urabilidade)

Cenário: website de Comércio Eletrônico

- Aplicação de venda de ítems pela web. Teremos que armazenar dados sobre usuários, catálogo de produtos, pedidos, as remessas, os endereços de envio, os endereços de cobrança e os dados sobre o pagamento.

Diagrama de Classe



Autor: Sadalage
& Fowler, 2013

Modelo Relacional

Customer	
Id	Name
1	Martin

Orders		
Id	CustomerId	ShippingAddressId
99	1	77

Product	
Id	Name
27	NoSQL Distilled

BillingAddress		
Id	CustomerId	AddressId
55	1	77

OrderItem			
Id	OrderId	ProductId	Price
100	99	27	32.45

Address	
Id	City
77	Chicago

OrderPayment				
Id	OrderId	CardNumber	BillingAddressId	txnId
33	99	1000-1000	55	abelif879rft

Autor: Sadalage & Fowler, 2013

Modelo Relacional

Customer	
Id	Name
1	Martin

Orders		
Id	CustomerId	ShippingAddressId
99	1	77

Product	
Id	Name
27	NoSQL Distilled

BillingAddress		
Id	CustomerId	AddressId
55	1	77

OrderItem			
Id	OrderId	ProductId	Price
100	99	27	32.45

Address	
Id	City
77	Chicago

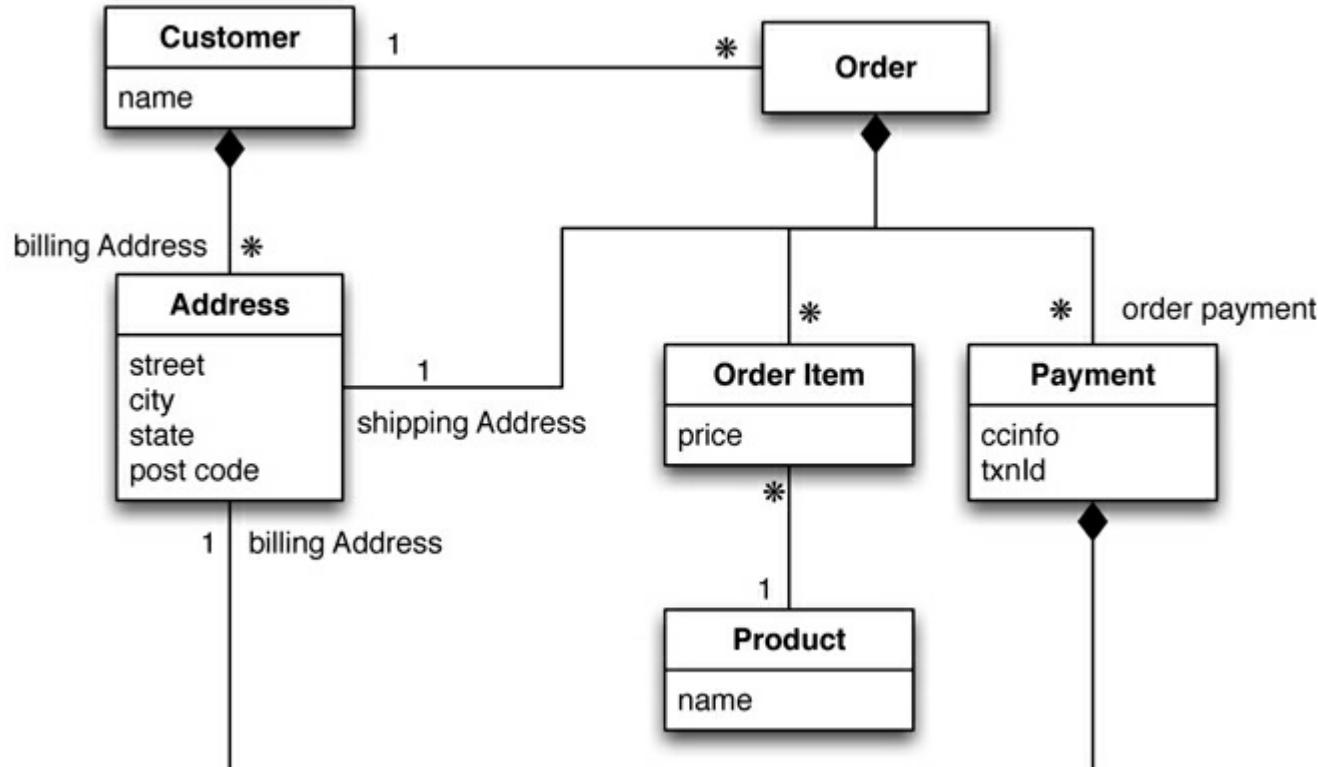
OrderPayment				
Id	OrderId	CardNumber	BillingAddressId	txnId
33	99	1000-1000	55	abelif879rft

Autor: Sadalage & Fowler, 2013

Considerações sobre o Modelo Relacional do Exemplo.

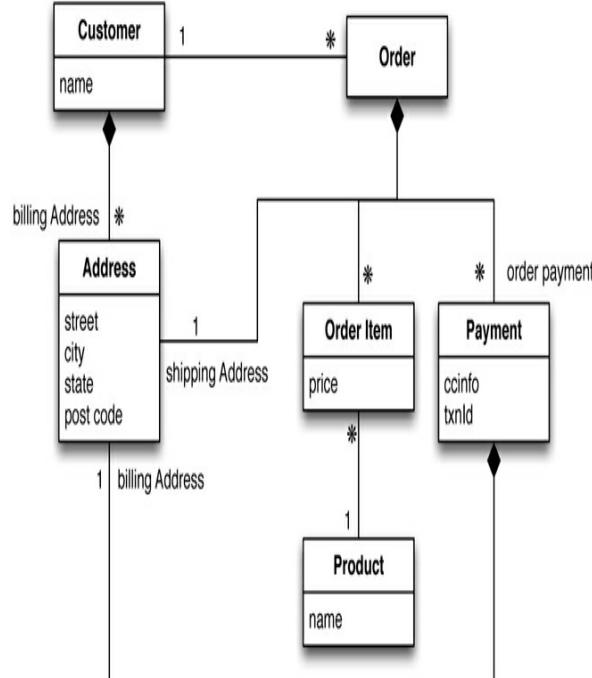
- Tabelas estão Normalizadas (nenhum dado repete-se em múltiplas tabelas);
- Integridade referencial (todas as chaves estrangeiras das tabelas, são identificadores únicos de registros em outras tabelas);

Orientação a agregados



Autor: Sadalage
& Fowler, 2013

Exemplo de Dados no formato JSON



```
// in customers
{
  "id":1,
  "name":"Martin",
  "billingAddress": [ {"city":"Chicago"} ]
}

// in orders
{
  "id":99,
  "customerId":1,
  "orderItems": [
    {
      "productId":27,
      "price": 32.45,
      "productName": "NoSQL Distilled"
    }
  ],
  "shippingAddress": [ {"city":"Chicago"} ]
  "orderPayment": [
    {
      "ccinfo":"1000-1000-1000-1000",
      "txnid":"abelif879rft",
      "billingAddress": {"city": "Chicago"}
    }
  ],
} 
```

Autor: Sadalage
& Fowler, 2013

Considerações sobre o Agregado

- Temos 2 agregados independentes: Cliente e Pedido;
- Losango preto (Símbolo do Diagrama de Classes UML) indica um agregado;
- Um único endereço aparece 3 vezes, ao invés de usarmos IDs. Esse registro é tratado como valor copiado;
- A conexão entre os agregados Cliente e Pedido estabelece um relacionamento de agregados, e não é um agregado. Da mesma forma que acontece entre Item de Produto e Produto. O nome do produto foi incluído em Item de Pedido para minimizar o número de agregados que acessamos durante o acesso aos dados.
- Os limites dos agregados são definidos com base no modelo de acesso dos principais aplicativos.

Observações gerais sobre Agregados

- O Modelo Relacional não diferencia relacionamentos que representam agregações, daqueles que não são agregações. Como consequência restringe o armazenamento distribuído dos dados;
- O modelo de agregados é adequado para uso com Cluster/Grid Computacional;
- Agregados permitem suportar transações com características de ACID (Atômicas, Consistentes, Isoladas e Duráveis) num único por vez.

Bibliografia

- SADALAGE, P.J; FOWLER, M. NoSQL Essencial. Novatec. 2013.
- Fundamentals of Database Systems, Elmasri & Navathe.
- LAUDON K.C. & LAUDON J.P. Management Information Systems, Capítulo 5.

Modelagem de Dados

Introdução à Conceitos de Modelagem e Acesso a Data Sets em Banco de Dados

Prof. Dr. Pedro Luiz Pizzigatti Corrêa

22 de Setembro de 2020