



BACK-END

Desenvolvimento de Software para Internet



MODELAGEM DE DADOS

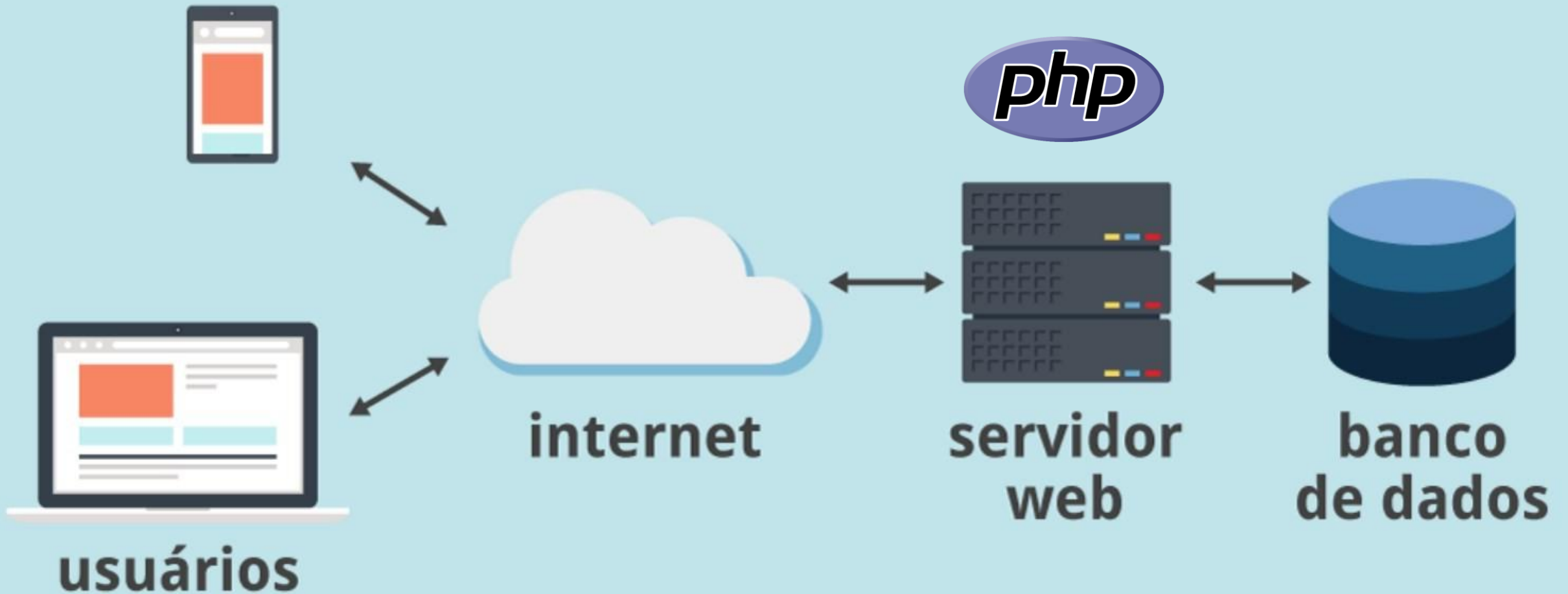


Criar e manter soluções utilizando banco de dados **estruturados e não estruturados (NoSQL)**, softwares gerenciadores de banco de dados



Crédito: [Flaticon](#)

Para que servem os bancos de dados ...

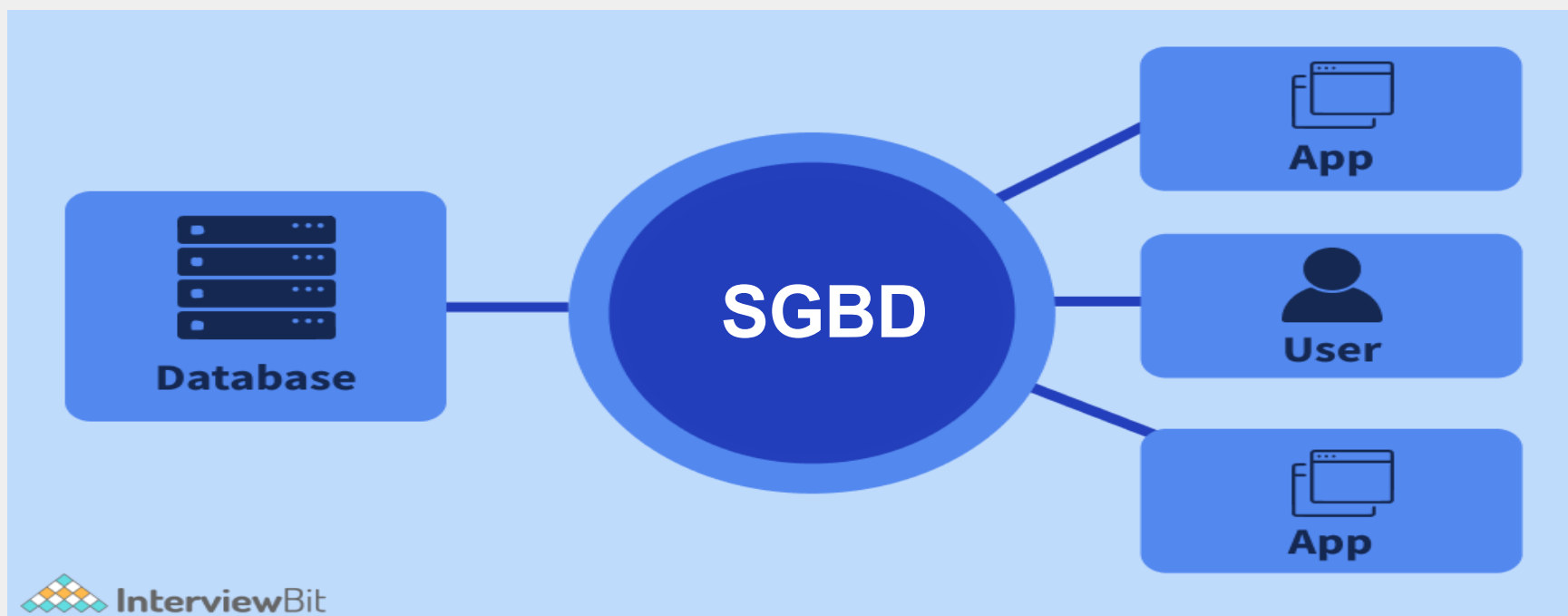


Crédito: [Tudo sobre hospedagem de sites](#)



Sistemas Gerenciadores de Bancos de Dados

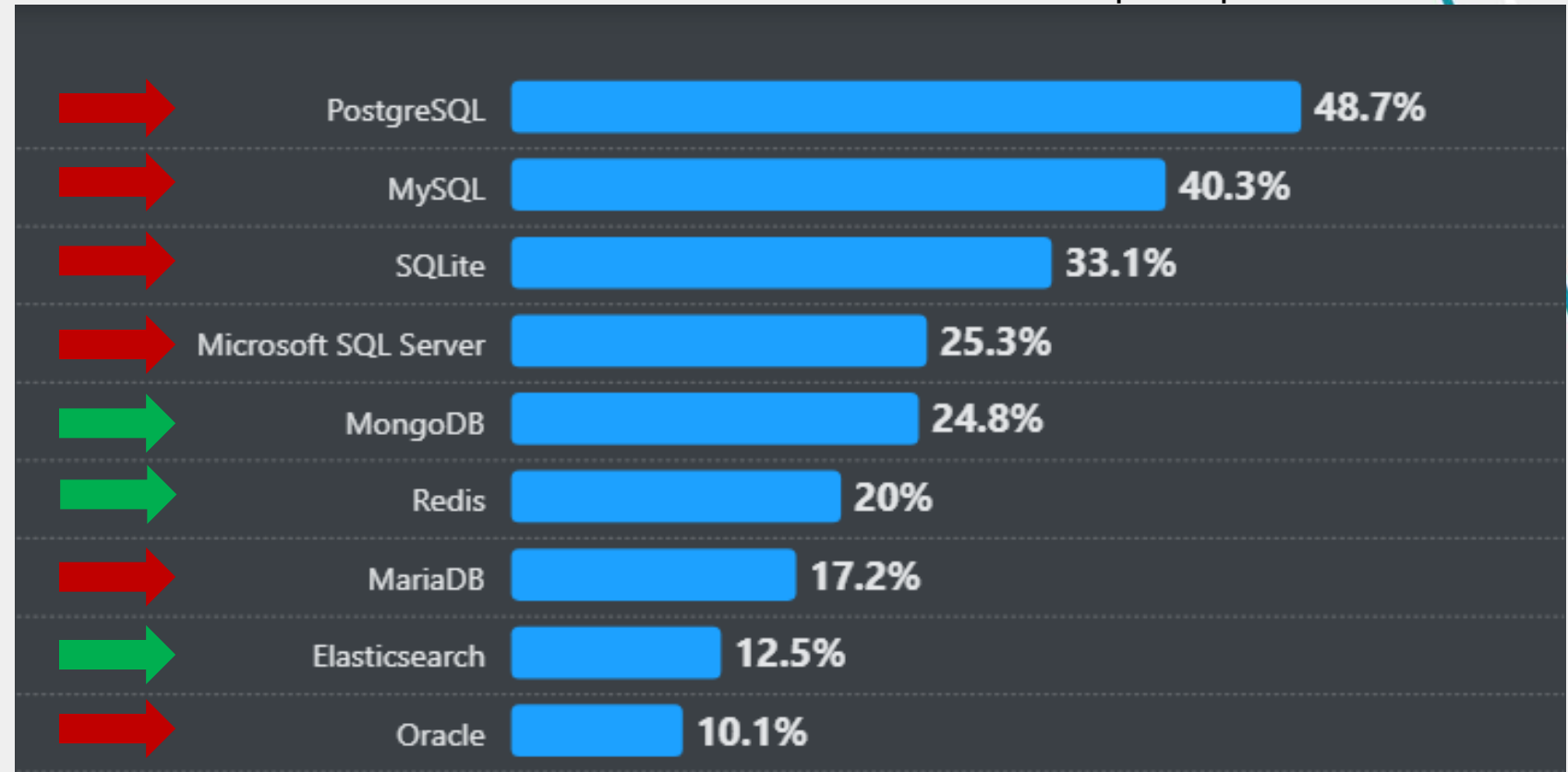
Softwares que retiram da aplicação cliente a responsabilidade de gerenciar o acesso, manipulação e organização dos dados.



PostgreSQL é o favorito dos desenvolvedores em novos projetos



Coleta de dados: 19/05/2024 a 20/06/2024 com 65.437 participantes no total

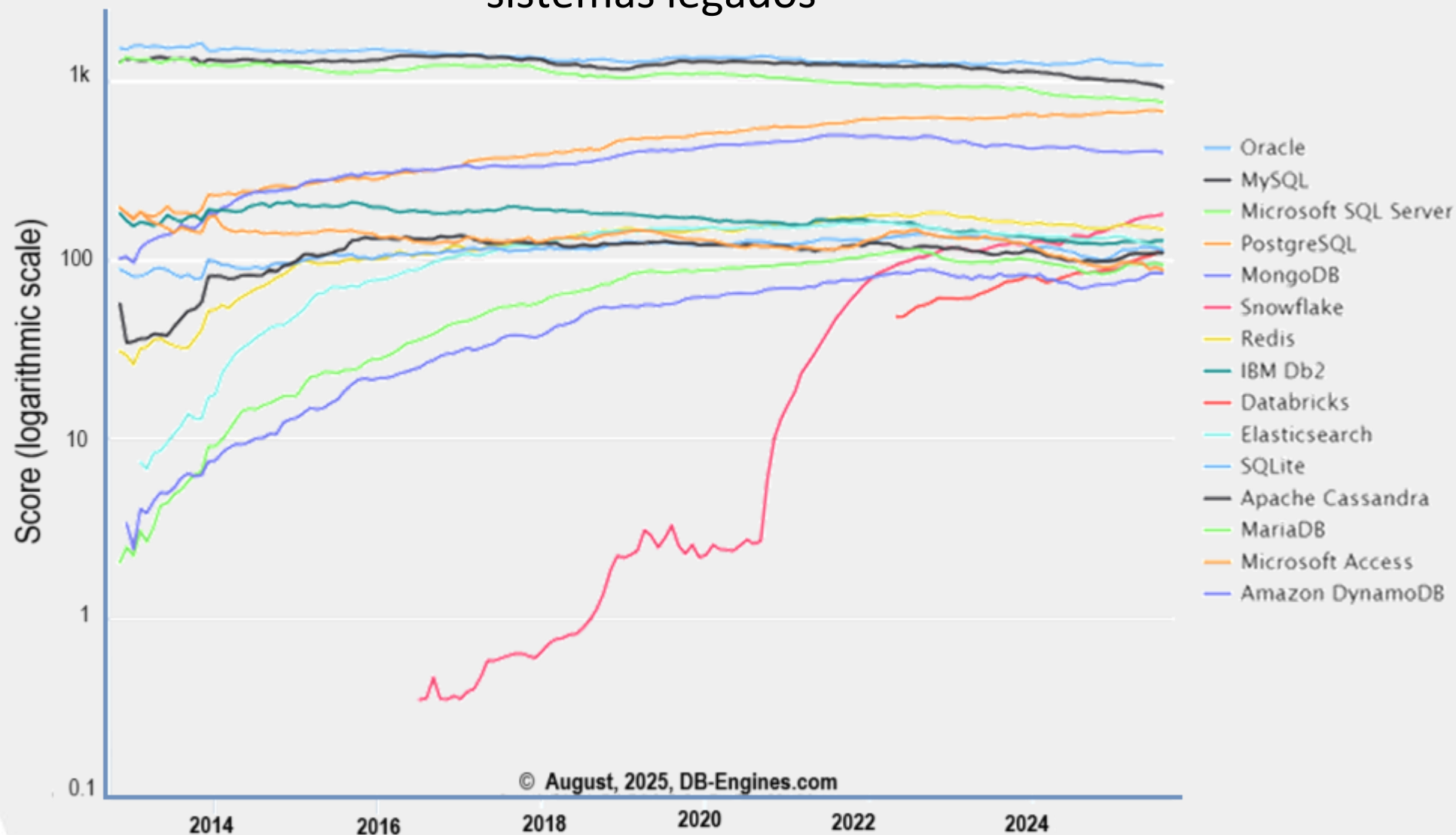


Estruturados (SQL)
Relacionais

Semi estruturados ou não
estruturados (NoSQL)

Fonte: <https://survey.stackoverflow.co/2024/technology#most-popular-technologies-database>

Oracle segue dominante em grandes empresas e sistemas legados

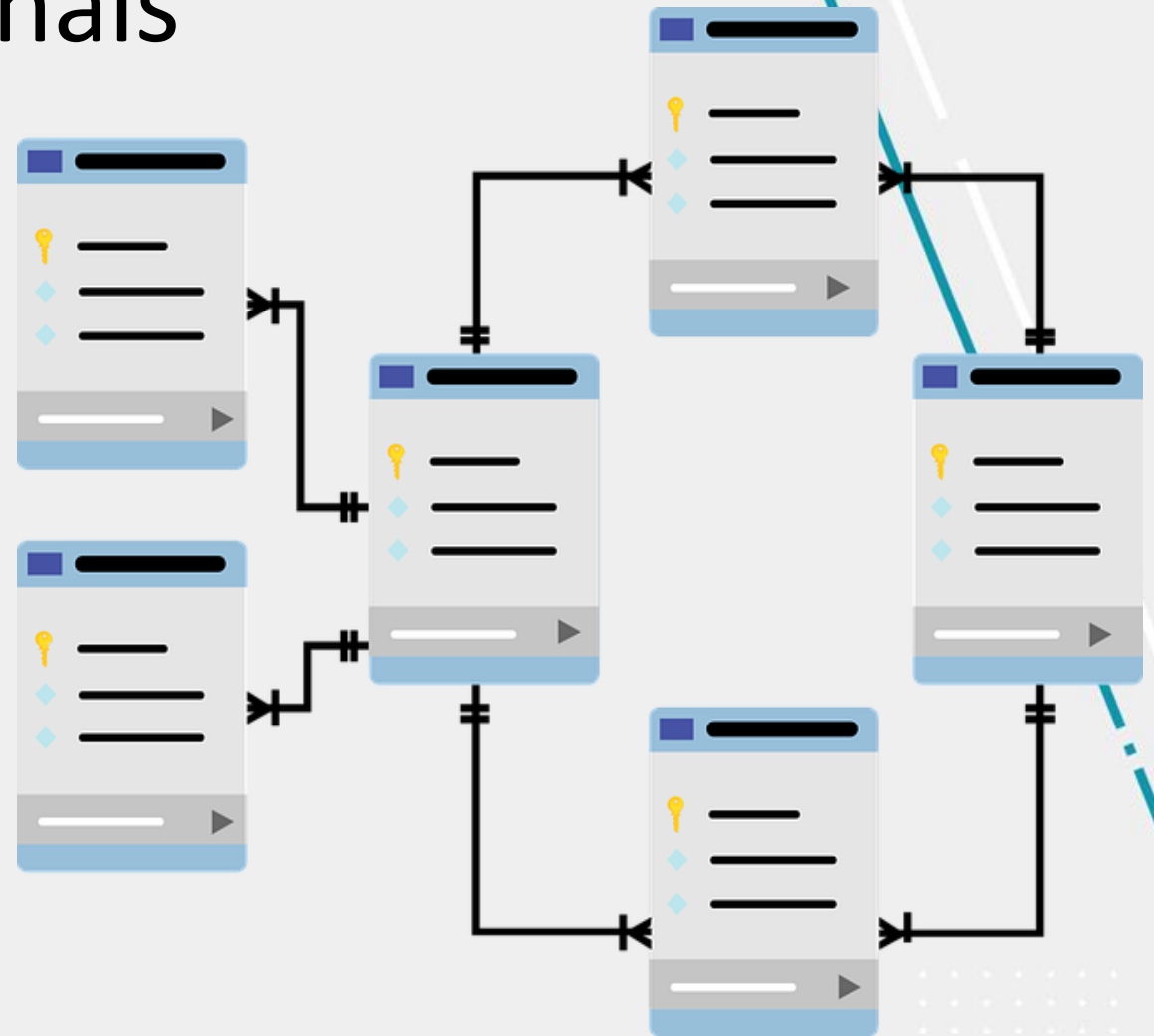


Fonte: https://db-engines.com/en/ranking_trend



Bancos de dados relacionais

- Dados estruturados
- Baseia-se na teoria dos conjuntos e álgebra relacional
- Esquema bem definido
- Estrutura fundamental é a relação (tabela)
- Garantem as propriedades **ACID**



Crédito: [Towards Data Science](#)

Propriedades ACID



A

Atomicidade

Garantia de que a transação será feita totalmente ou não será feita

C

Consistência

Proteção da integridade dos dados. Se uma transação não for completada, ela retorna ao estado inicial

I

Isolamento

Transações são feitas de forma isolada: uma não atrapalha a outra de acontecer

D

Durabilidade

Preservação dos dados após as operações terem sido realizadas

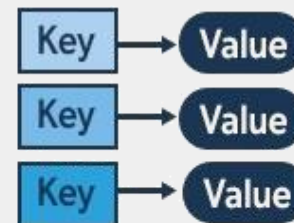


Bancos de dados NoSQL

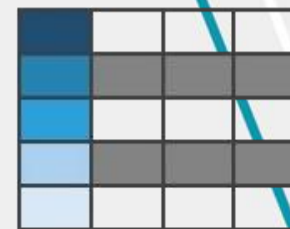


- Não usam SQL (No-SQL ou Not-only-SQL)
- Amplamente difundidos no contexto de BigData
- Geralmente são open-sources
- Adotam formas diferentes de organização (não usam tabelas)

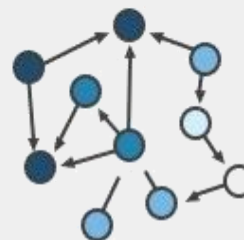
Chave-valor



Colunar



Grafo



Documento



Crédito: [GeeksforGeeks](https://www.geeksforgeeks.org/)
(traduzido pela professora)





Relacional *versus* NoSQL

NoSQL



Key/value store



Document database



Column family store

SQL



Relational table storage



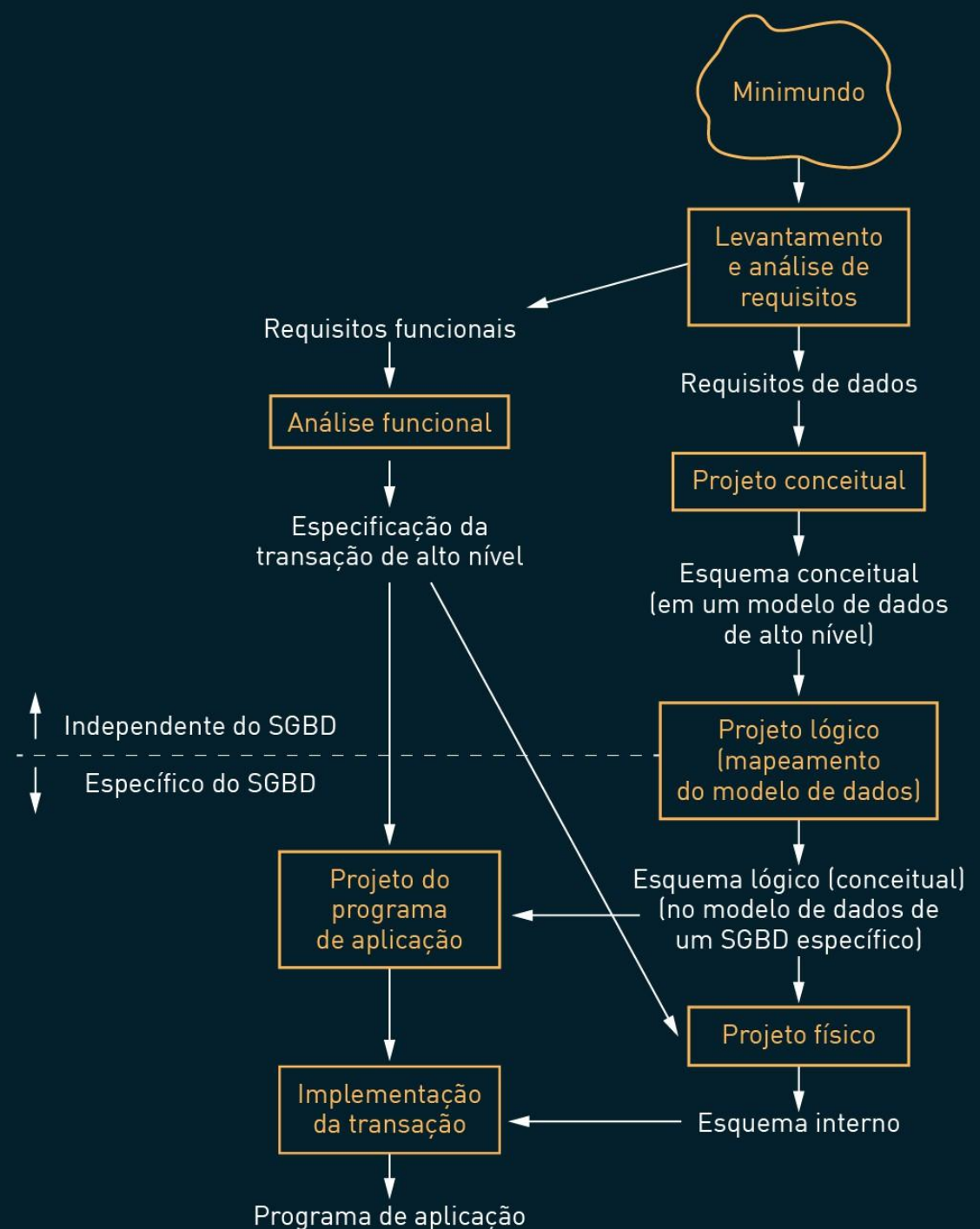
Relationships use joins

NoSQL vem para suprir uma lacuna e não para substituir

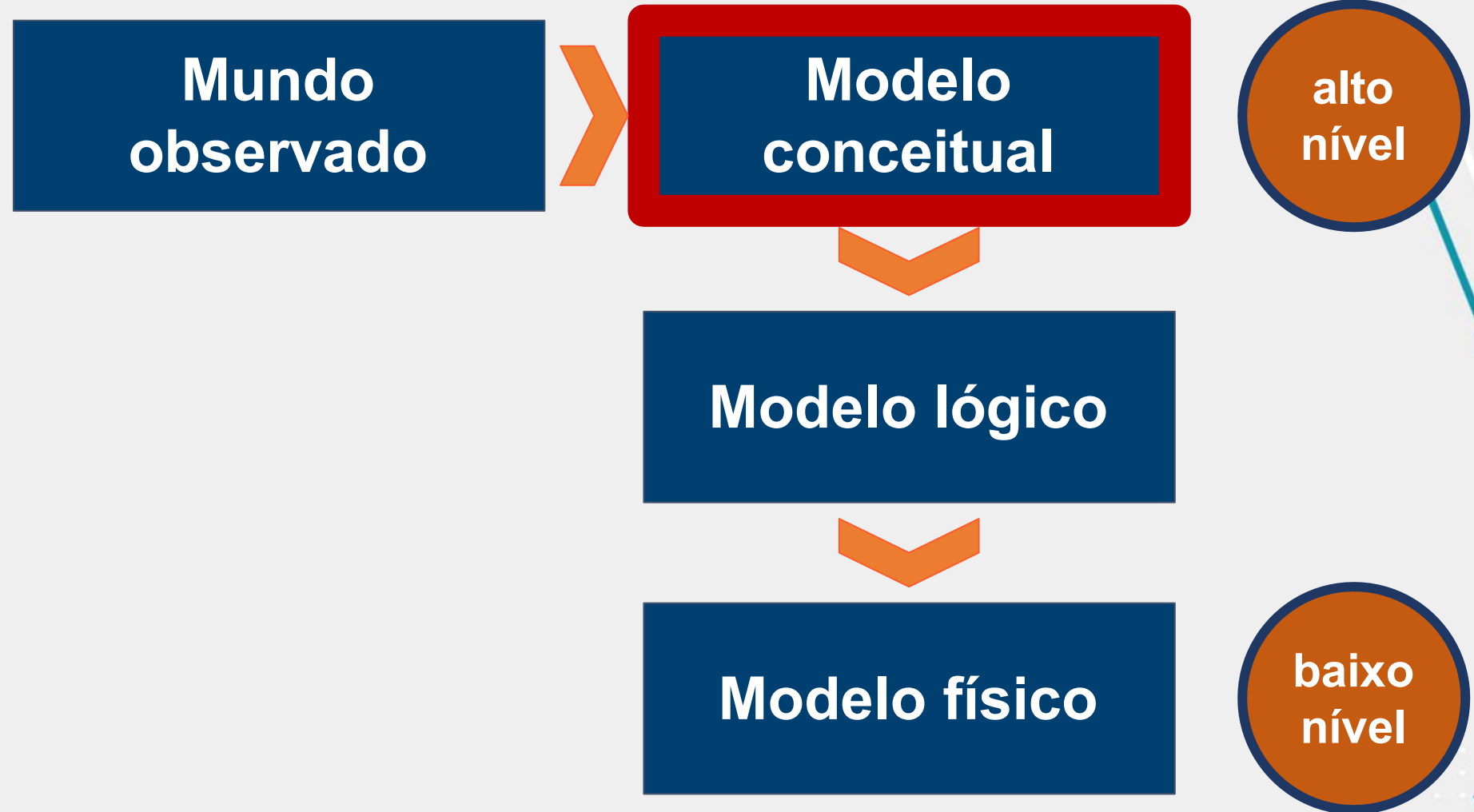


Fluxo de desenvolvimento de um Sistema de Informação

Crédito: AVA da disciplina



Modelagem de dados





Modelo conceitual

Modelo de Entidade Relacionamento (MER): Peter Chen 1976

Modelo baseado na percepção do mundo real, que consiste em um conjunto de objetos básicos chamados entidades e nos relacionamentos entre esses objetos.

DER é seu diagrama



Facilita o projeto de banco de dados, possibilitando a especificação de uma estrutura lógica geral





Diagrama entidade-relacionamento

Descreve a estrutura lógica geral de um banco de dados.

- **Retângulos:** representam entidade
- **Elipses:** representam atributos
- **Losangos:** representam relacionamento
- **Linhas:** ligam atributos a entidades e entidades a relacionamentos





Exemplo

Uma pessoa vai à biblioteca e quer pegar um livro emprestado. Isso deve ser registrado em um sistema de informação.

- Pessoa tem cpf, nome, idade, endereço
- Livro tem isbn, título, autores

Pessoa e Livro são

ENTIDADES

Nome, cpf, isbn, título

são **ATRIBUTOS**

Pegar o livro emprestado

é o **RELACIONAMENTO**





Elementos do Diagrama: Entidade

- Pode representar tanto objetos concretos (pessoa, carro), quanto abstratos (departamento, venda) **(tabelas)**
- Representada por um retângulo com o nome da entidade

Pessoa

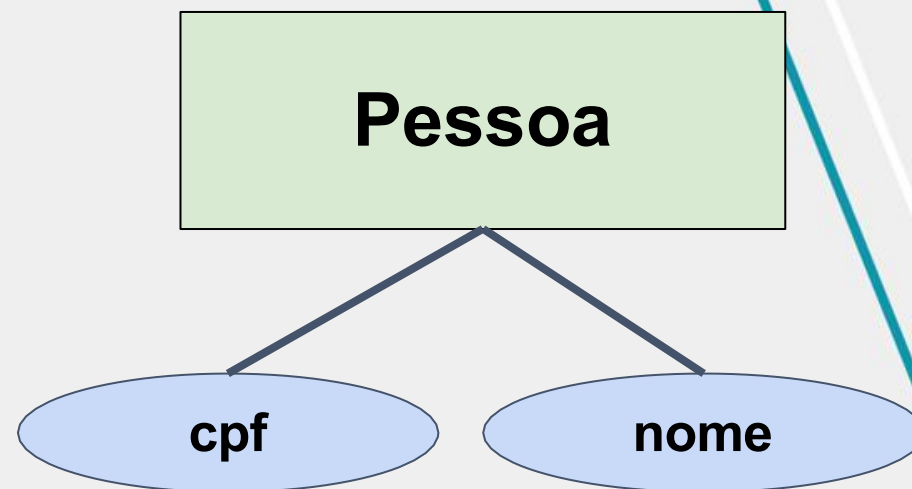
Livro





Elementos do Diagrama: Atributos

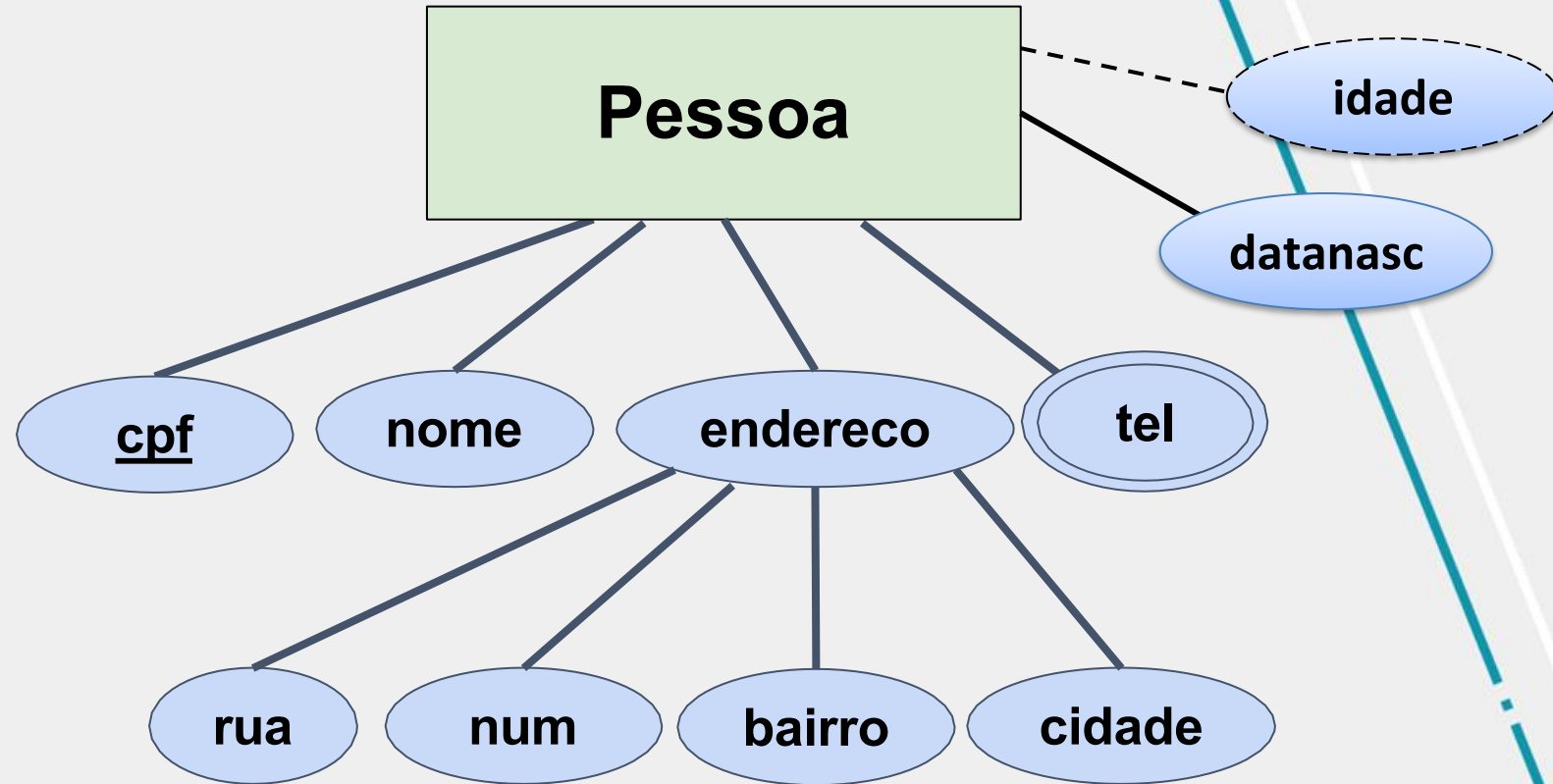
- Descrevem uma determinada Entidade por meio de suas características
- Possuem valor próprio, o qual constituirá o conteúdo efetivo do banco de dados (**campos**)
- Representados por elipses





Classificação de atributos

- Simples ou compostos - divididos em duas ou mais partes (endereço)
- Monovalorados ou multivalorados - possuem mais de um valor (tel)
- Determinantes (chave): identificam unicamente cada entidade (cpf)
- Derivados: calculado a partir de outros (idade)



Tipos: int, real, string, etc
Domínio: valores permitidos





Conceito de instância

Representa o conteúdo do banco de dados

cpf	nome	endereço	telefone
097676-3	João da Silva	Rua X, casa 2	2223-2323
453434-1	Maria Neves	Rua Z, casa 1	4555-4444
334565-1	Ricardo Almeida	Rua M, casa 3	3434-2223
243340-0	Tássio Tavares	Rua X, casa 10	2343-2333
232321-1	Aline Gouvea	Rua Z, casa 4	8822-2323

instância

instância

instância

instância

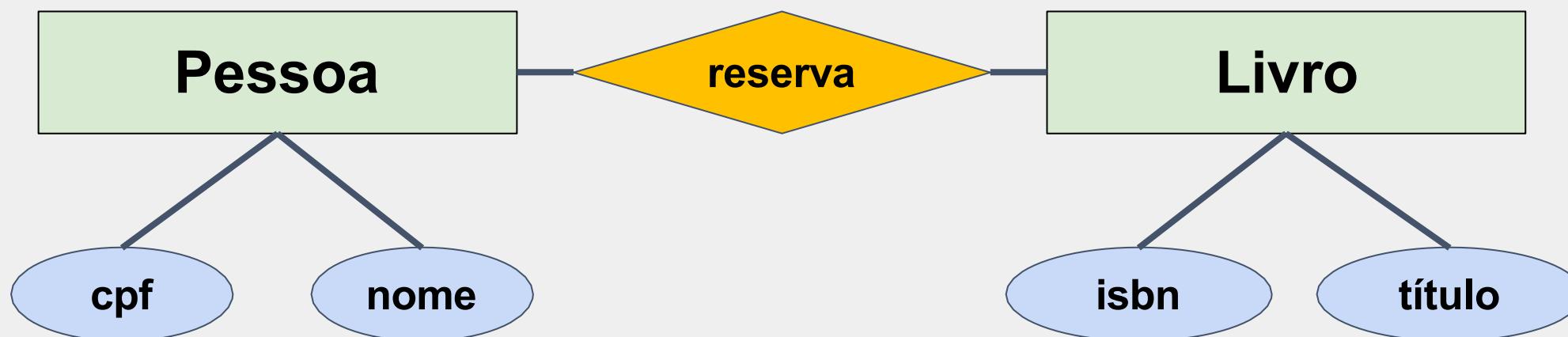
instância



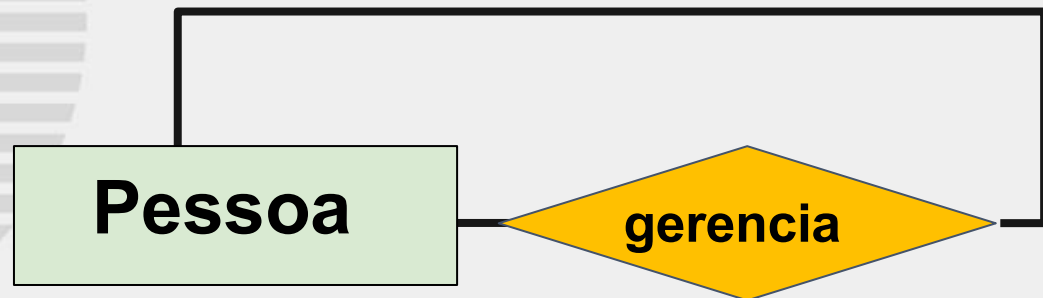


Elementos do Diagrama: Relacionamentos

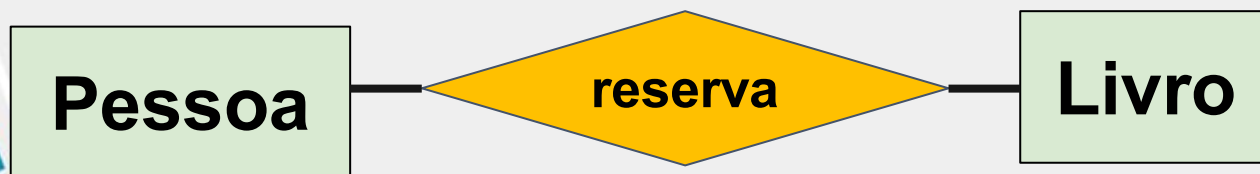
- Conjunto de associações entre as entidades;
- Ocorre quando as entidades envolvidas na associação possuem um ou mais atributos em comum



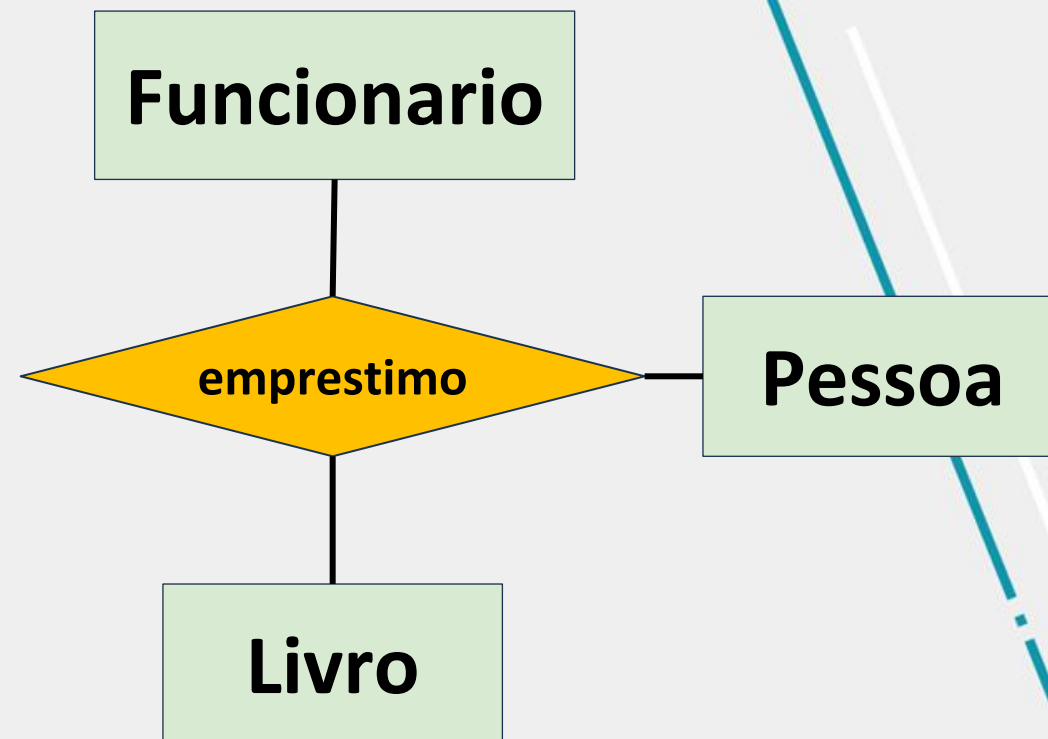
Classificação de relacionamentos



Exemplo unária



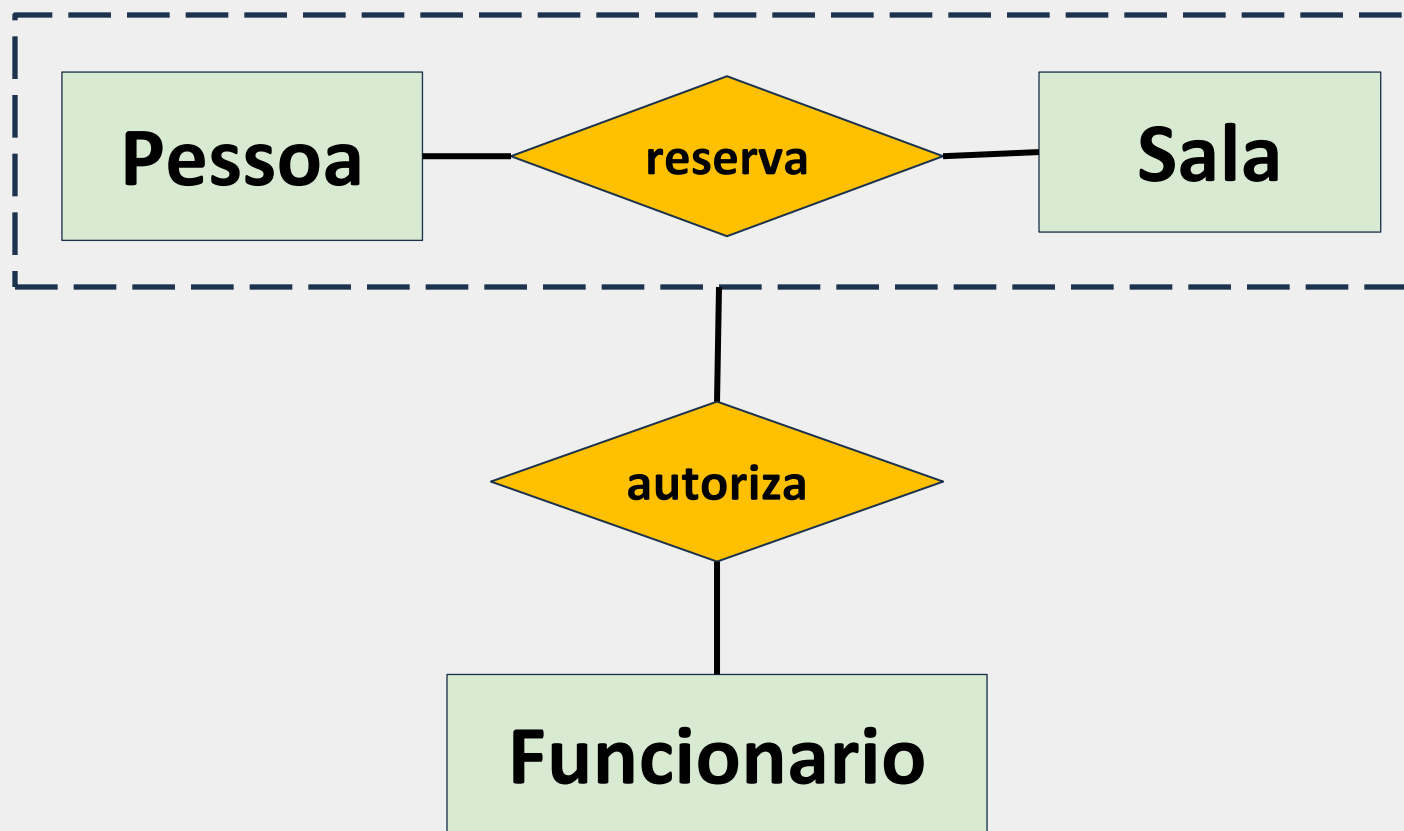
Exemplo binária



Exemplo ternária



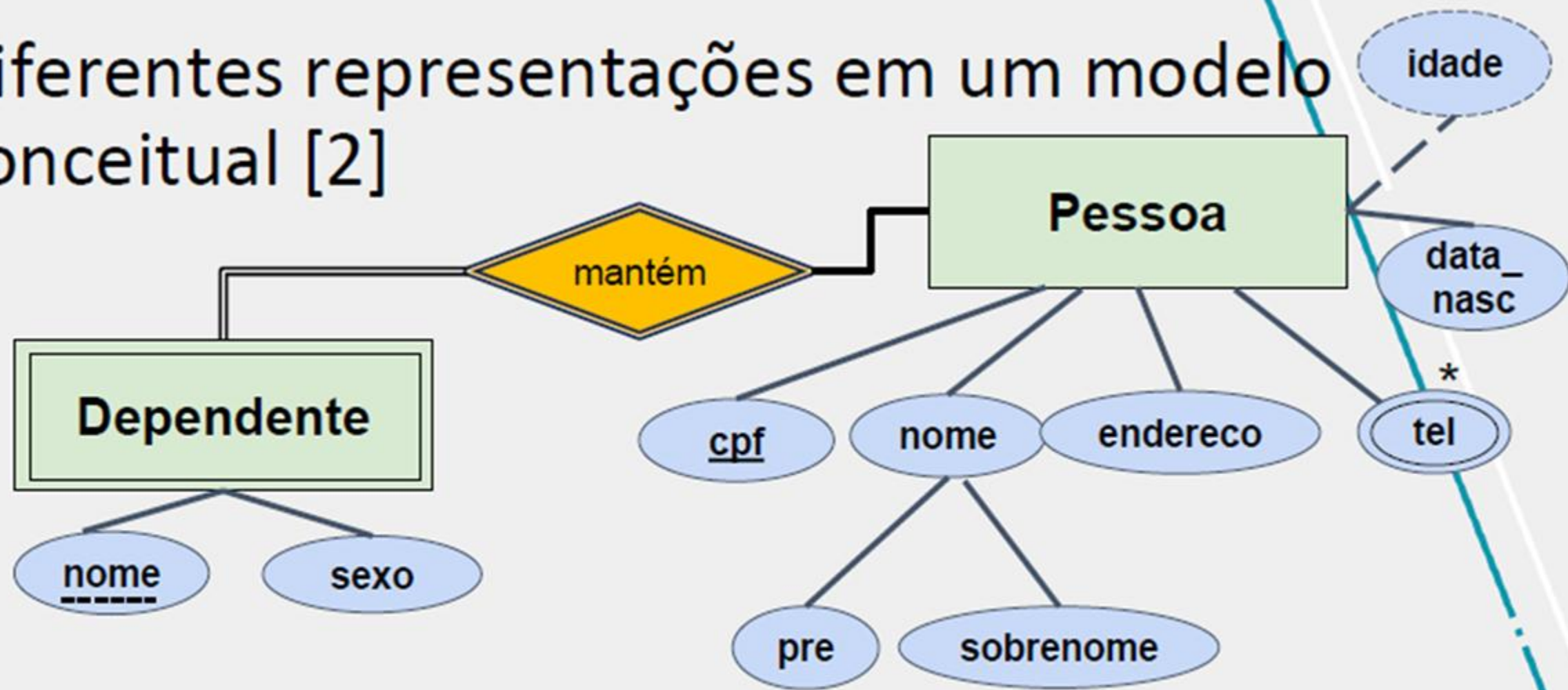
Diferentes representações em um modelo conceitual [1]



Agregação



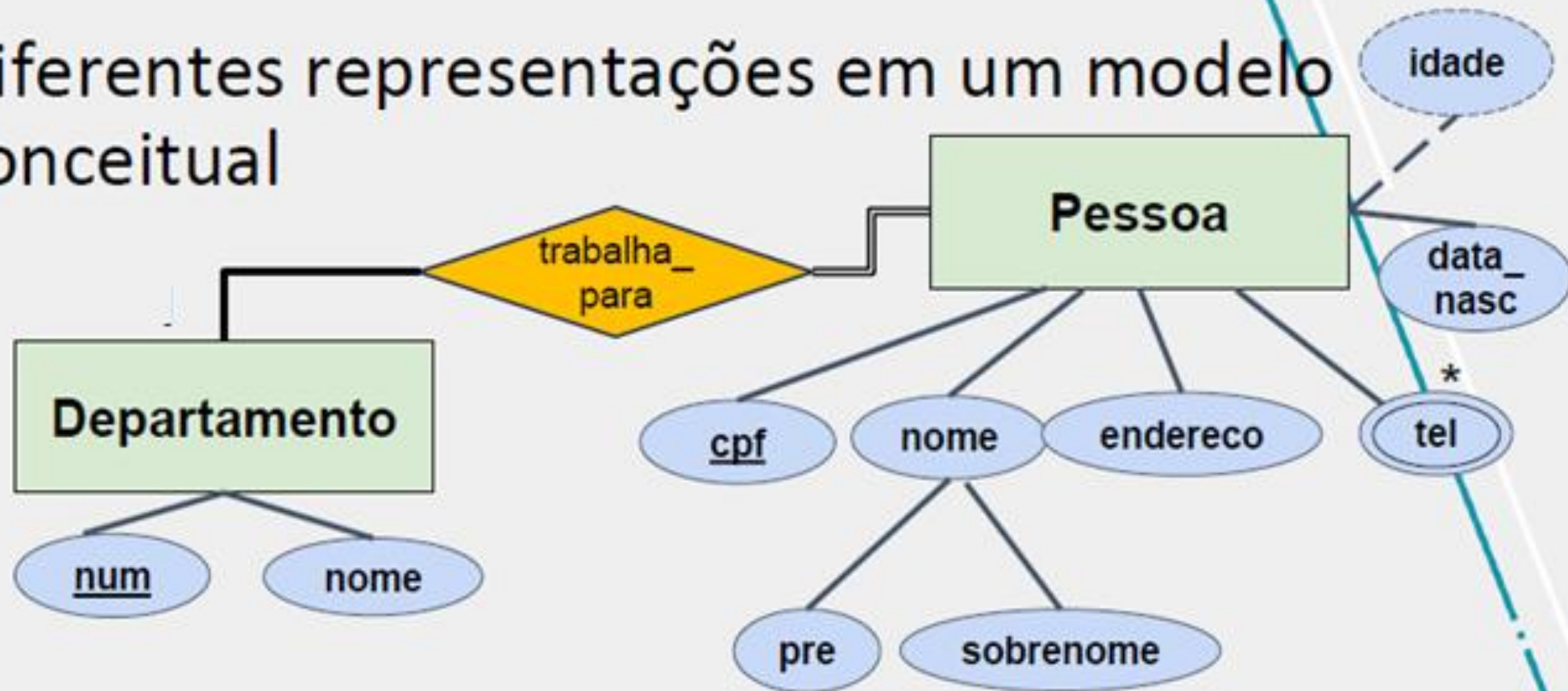
Diferentes representações em um modelo conceitual [2]



Entidade fraca: não possui atributo chave e é identificada por meio de um relacionamento total com pelo menos um tipo Entidade Forte.



Diferentes representações em um modelo conceitual



Restrição de participação: especifica se uma entidade individual precisa, necessariamente, estar associada a outra entidade individual via relacionamento para existir. Se é obrigatória: a restrição é total, caso contrário é parcial





Elementos do Diagrama: Cardinalidade

- Determina os limites mínimos e máximos de instâncias de uma entidade em um relacionamento

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





Exemplo: Uso de uma biblioteca



Crédito: [Unisuam](#)

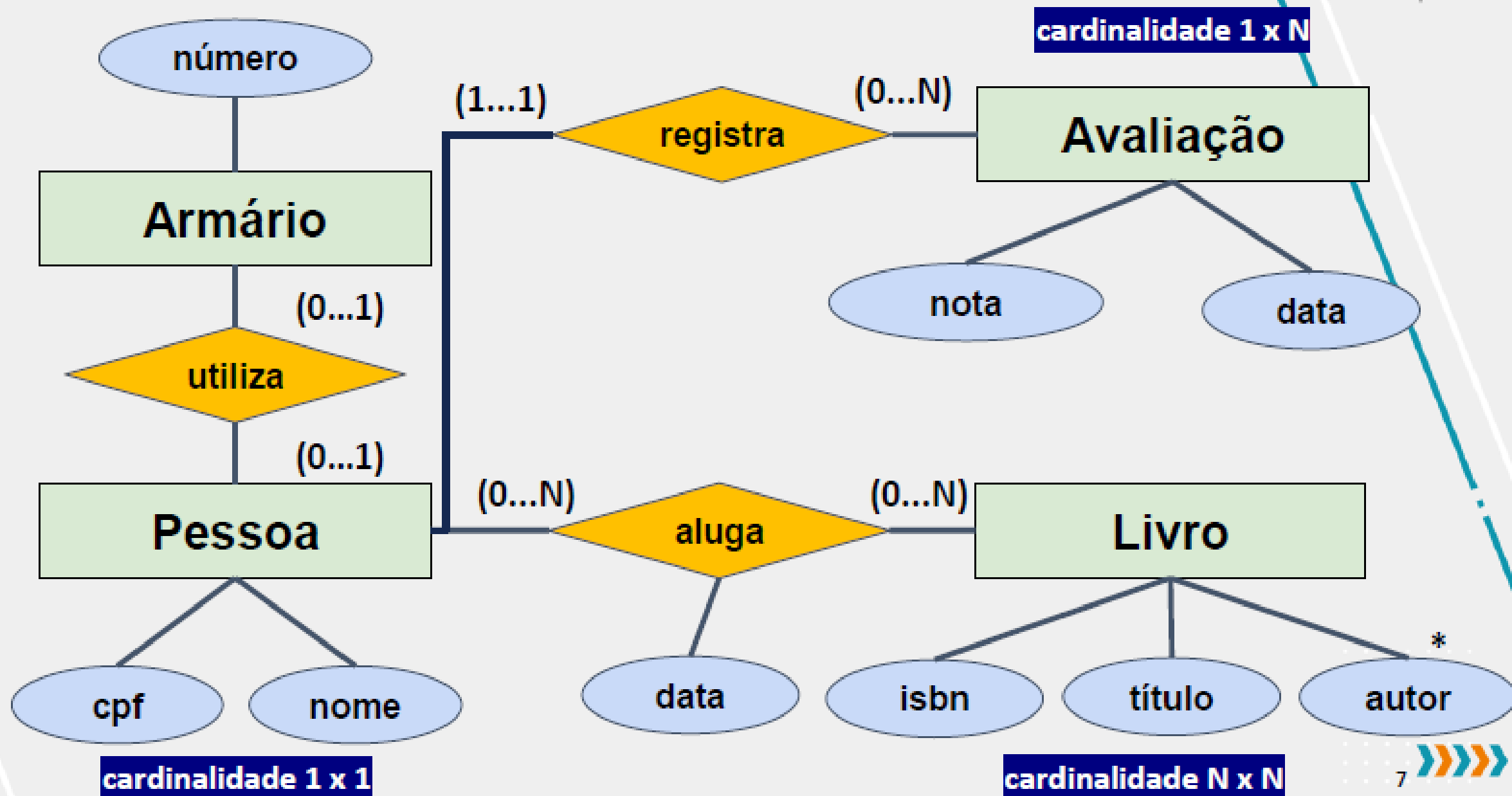




Você foi contratado para criar um sistema para a biblioteca da Unisuam. Neste sistema, a principal funcionalidade é registrar as pessoas que poderão alugar livros. Pessoa tem cpf, nome, idade, endereço. Livro tem isbn, título, autores. O sistema também permite armazenar informações em relação ao número do armário que foi atribuído a determinada pessoa, caso ela queira ter um armário fixo na biblioteca. Um armário só pode ser usado por uma pessoa e vice-versa. Por fim, é necessário registrar avaliações feitas pelas pessoas que usam a biblioteca. Essas avaliações podem ser feita em cada uso. O sistema deve armazenar o texto da avaliação e a data em que foi feita.



Exemplo 1: uso de uma biblioteca





Dicas para elaborar diagrama E-R

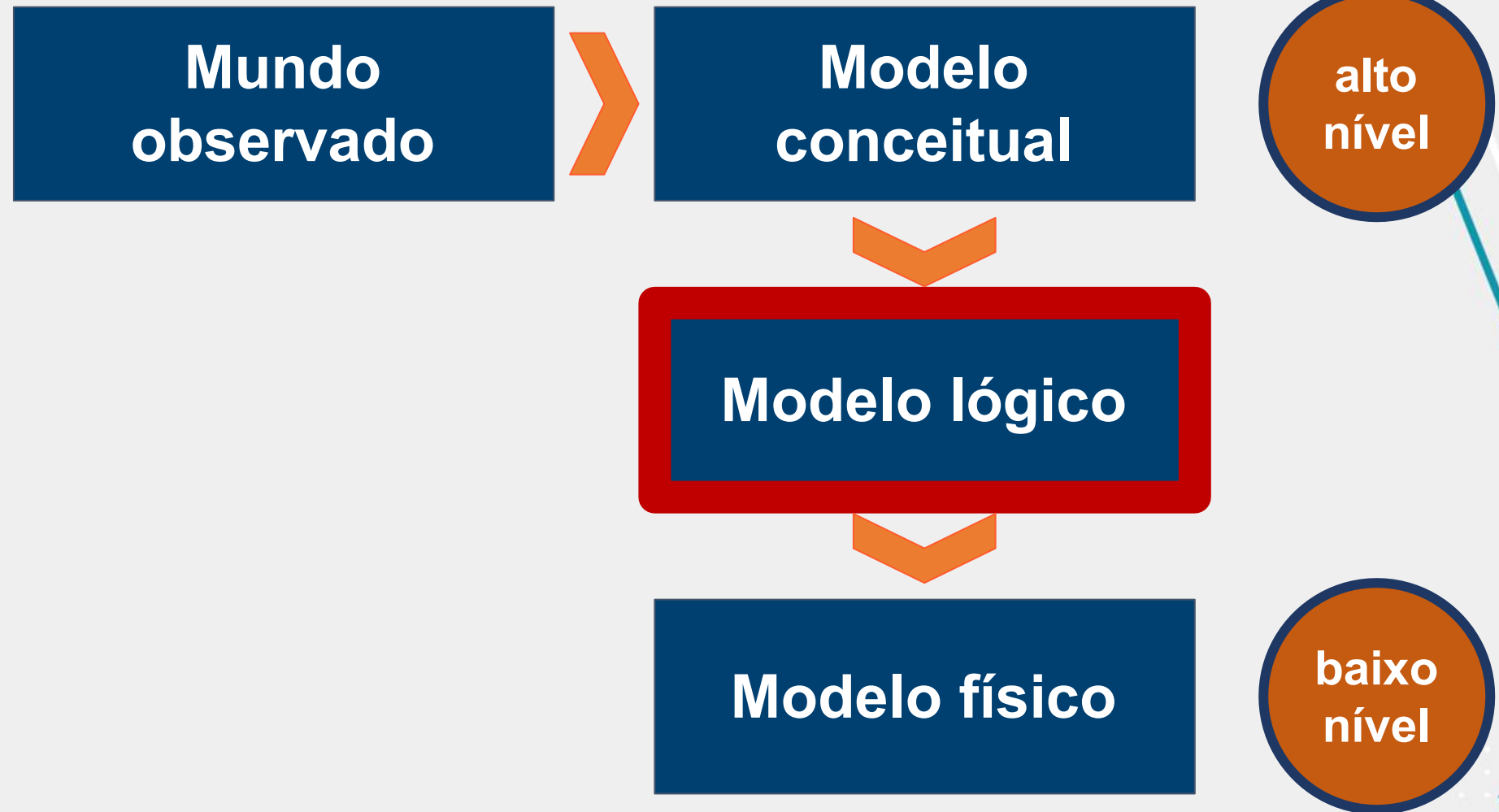
- **Substantivos** indicam entidades
- **Verbos** geralmente estão relacionados a relacionamentos
- **Adjetivos** (qualidade) é uma forte indicação de um atributo



Exercícios



Modelagem de dados





Modelo lógico

- Deve ser construído após a conclusão do modelo conceitual
- É criado com base em um tipo de banco de dados: como SQL Server, Oracle, MySQL



Modelo lógico: mapeamento [1]



★ 1:1 - União de tabelas

pessoa
PK id_pessoa: inteiro
nome: varchar(30)
cpf: varchar (15)
num_armario: inteiro





Modelo lógico: mapeamento [2]



- ★ 1:1 - União de tabelas
- ★ 1:N - Lado N recebe a FK

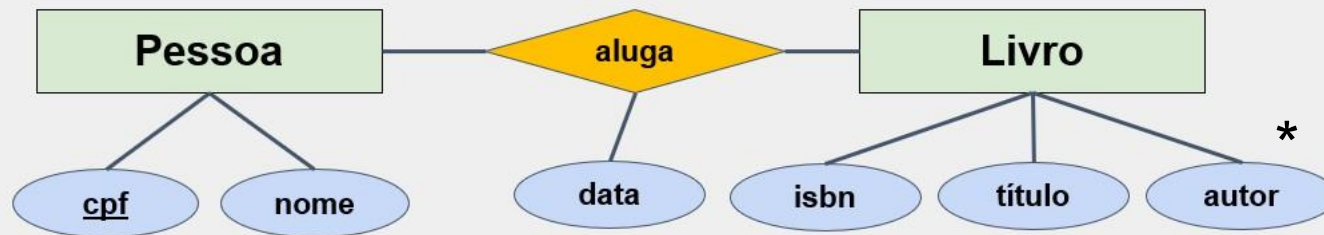
**FK: Foreign Key
(chave estrangeira)**

pessoa
PK id_pessoa: inteiro
nome: varchar(30)
cpf: varchar (15)
num_armario: inteiro

avaliacao
PK id_avaliacao: inteiro
FK id_pessoa: inteiro
nota: inteiro
data: date



Modelo lógico: mapeamento [3]



- ★ 1:1 - União de tabelas
- ★ 1:N - Lado N recebe a FK
- ★ N:N – Nova tabela

PK composta

pessoa
PK id_pessoa: inteiro
nome: varchar(30)
cpf: varchar (15)
num_armario: inteiro

pessoa_livro
FK id_pessoa: inteiro
FK id_livro: inteiro
data: date

livro
PK id_livro: inteiro
isbn: varchar (20)
título: inteiro
autor: varchar(50)



Atributos multivalorados são estruturados em tabelas (neste caso, relação N:N)

autor
PK id_autor: inteiro
nome: varchar (30)



pessoa
PK id_pessoa: inteiro
nome: varchar(30)
cpf: varchar (15)
num_armario: inteiro

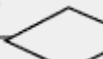
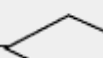
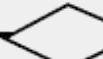
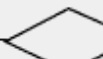
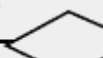
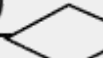
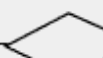
pessoa_livro
FK id_pessoa: inteiro
FK id_livro: inteiro
data: date

livro
PK id_livro: inteiro
isbn: varchar (20)
titulo: inteiro
autor: varchar(50)

livro_autor
FK id_livro: inteiro
PK id_autor: inteiro



Tipos de Relacionamentos (Regras)

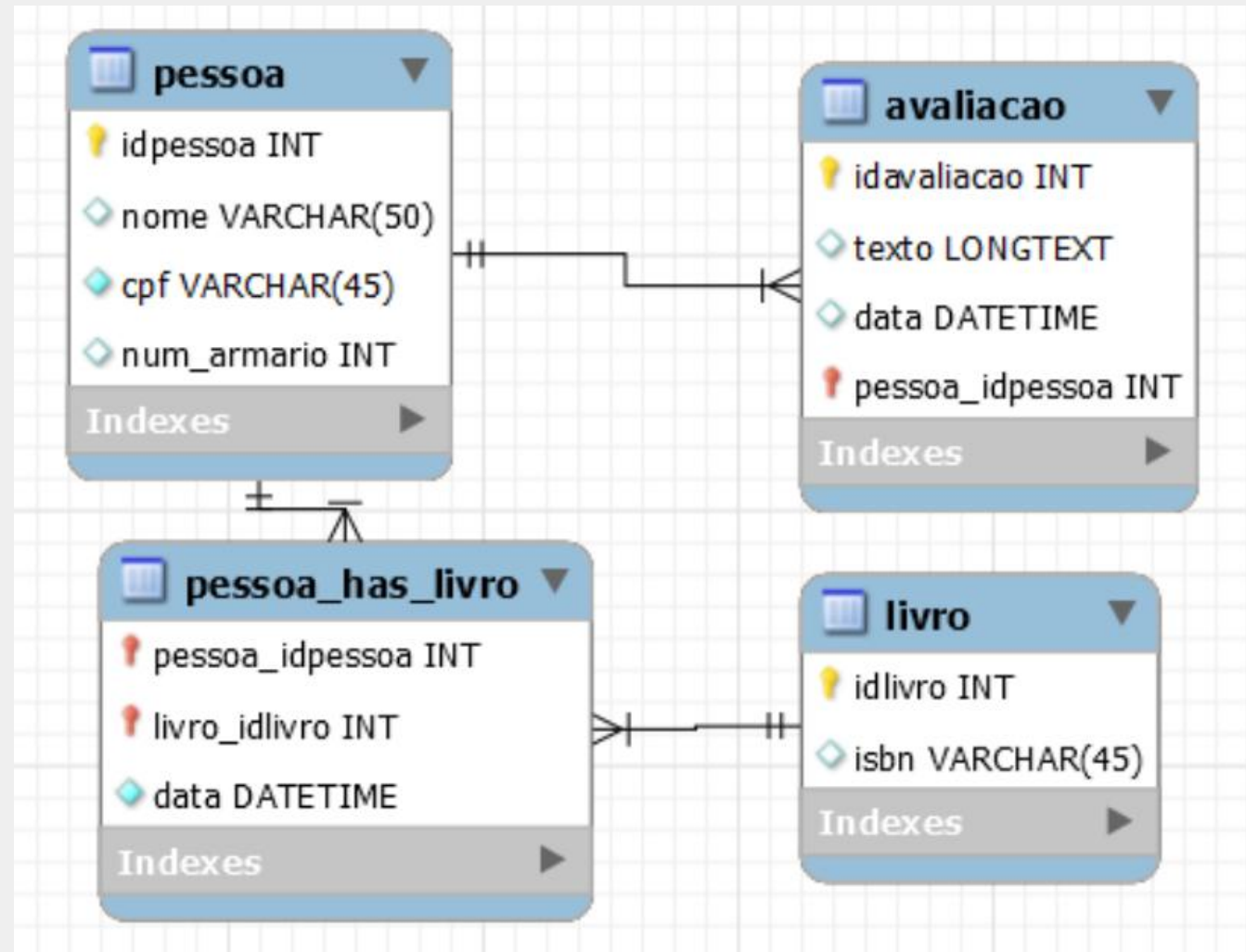
Tipo de relacionamento	Regra de implementação		
	Tabela própria	Adição coluna	Fusão tabelas
Relacionamentos 1:1			
<u>(0,1)</u>  <u>(0,1)</u>	±	✓	×
<u>(0,1)</u>  <u>(1,1)</u>	×	±	✓
<u>(1,1)</u>  <u>(1,1)</u>	×	±	✓
Relacionamentos 1:n			
<u>(0,1)</u>  <u>(0,n)</u>	±	✓	×
<u>(0,1)</u>  <u>(1,n)</u>	±	✓	×
<u>(1,1)</u>  <u>(0,n)</u>	×	✓	×
<u>(1,1)</u>  <u>(1,n)</u>	×	✓	×
Relacionamentos n:n			
<u>(0,n)</u>  <u>(0,n)</u>	✓	×	×
<u>(0,n)</u>  <u>(1,n)</u>	✓	×	×
<u>(1,n)</u>  <u>(1,n)</u>	✓	×	×

✓ Alternativa preferida ± Pode ser usada × Não usar

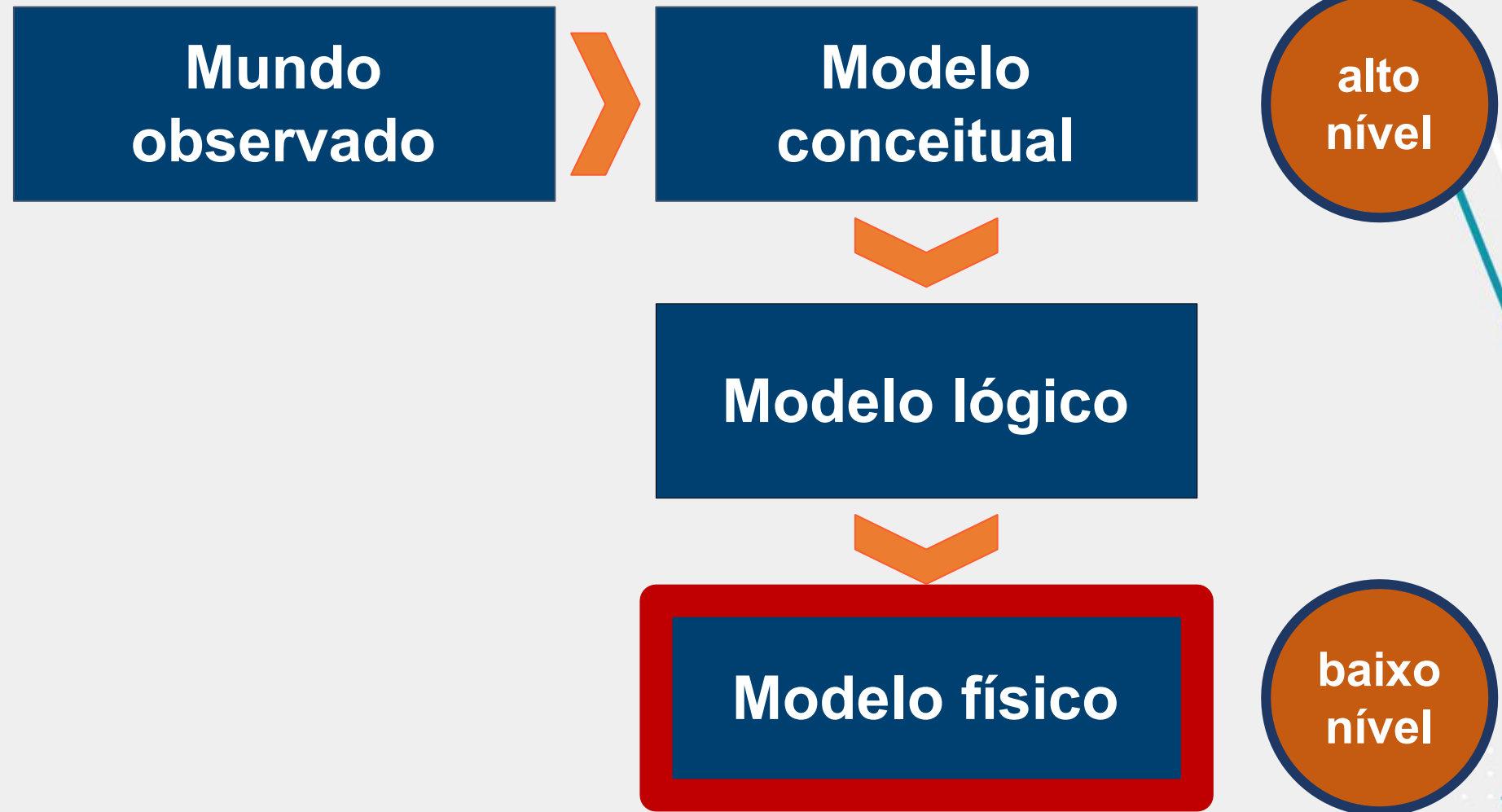




Modelo lógico no MySQL Workbench



Modelagem de dados



Modelo físico

id_pessoa	nome	cpf
1	João	44444455-55
2	Maria	22233333-77

pessoa

id_livro	isbn	titulo
1	NAT02	UX para iniciantes
2	NY02	Harry Potter e a Câmara Secreta

livro

id	id_pes	id_li	data
1	1	2	29/08/2023
2	2	1	30/08/2023

pessoa_livro





Ferramentas que ajudam na modelagem

- **Mysql Workbench**
- Star UML
- Microsoft Visio
- Astah
- erwin Data Modeler
- Lucidchart
- ERD Plus
- GenMyModel
- Visual Paradigm

Ferramentas CASE: Computer-Aided Software Engineering



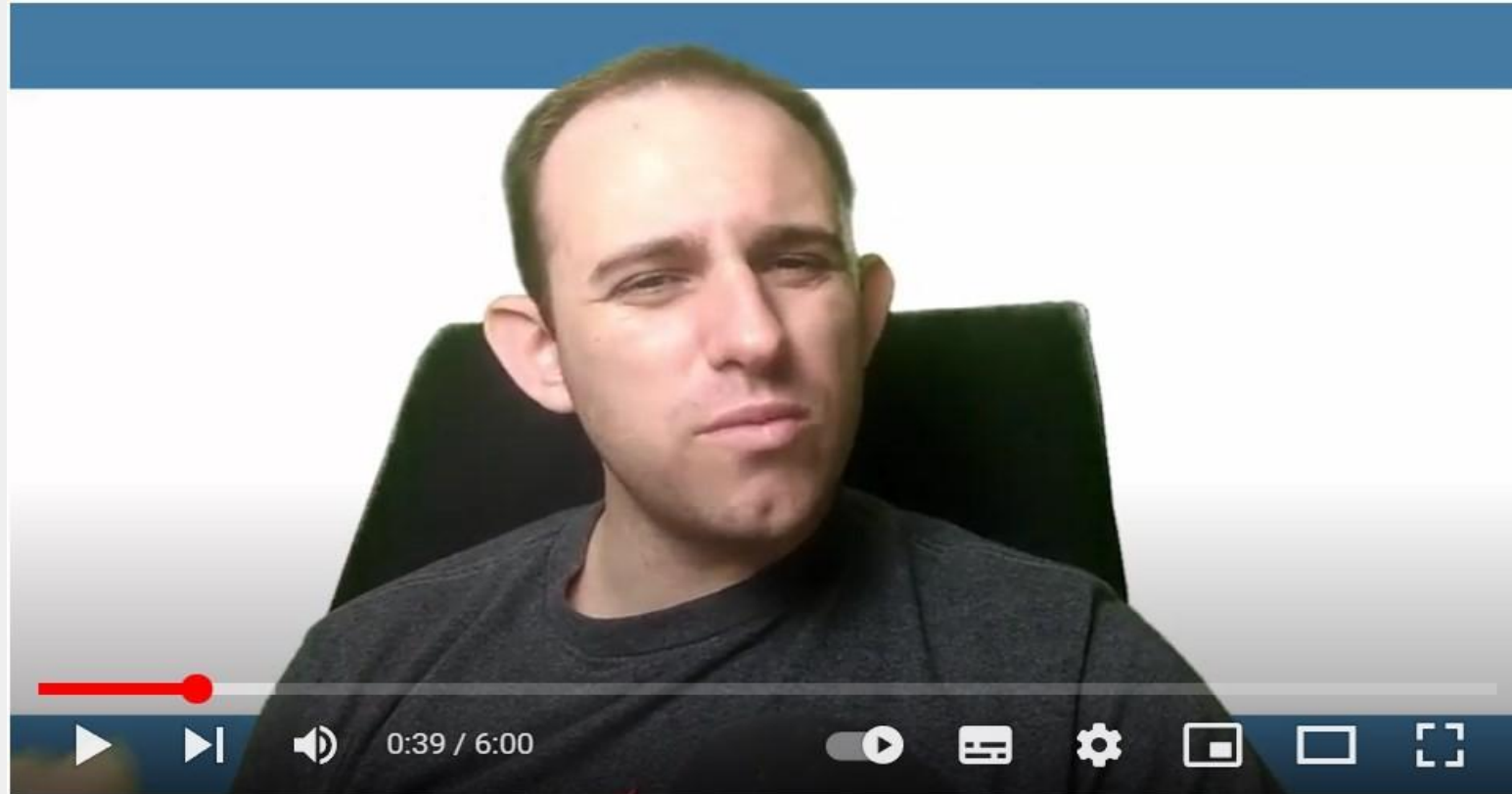
DEVER DE CASA

Construam o modelo lógico do exercício usando o MySQL Workbench





Como modelar no MySQL Workbench



[YouTube: Canal do Professor Binho](#)



Mais referências sobre o tema

- Boson Treinamentos (YouTube). Curso de Modelagem de Dados. Disponível em:
https://www.youtube.com/playlist?list=PLucm8g_ezqNoNHU8tjVeHmRGBFnjDIIXD
- Miriam TechCode (YouTube). Modelagem de dados - modelo conceitual, lógico e físico. Disponível em:
<https://www.youtube.com/watch?v=8CkMX2qXgdY>





Bancos de dados relacionais *versus* NoSQL

EXTRA



Bancos de Dados Relacionais

PK

id	nome	telefone
1	João	4444-4455
2	Maria	2223-3333

Cliente

PK

id	nome	preço
1	Moto	R\$ 20.000
2	Carro	R\$ 80.000

Produto

PK

id	id_cli	id_prod	data
1	1	2	29/03/2023
2	2	1	30/03/2023

Venda

FK



Limites de uso: bancos de dados relacionais em grandes conjuntos de dados

- ★ Capacidade de armazenamento: Peta bytes
- ★ Perda de performance e dificuldade de processar relacionamentos em grandes conjuntos de dados
- ★ Dificuldades em esquemas complexos

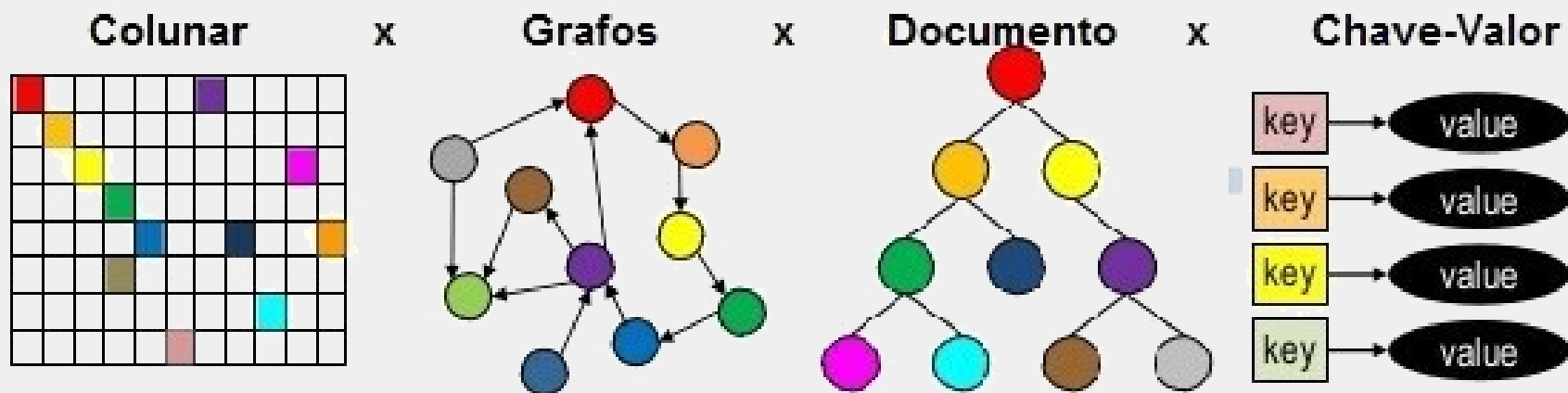
Medida	Simbologia	Equivalencia
byte	b	8 bits
kilobyte	Kb	1024 bytes
megabyte	MB	1024 KB
gigabyte	GB	1024 MB
terabyte	TB	1024 GB
Petabyte	PB	1024 TB
Exabyte	EB	1024 PB
Zetabyte	ZB	1024 EB
Yottabyte	YB	1024 ZB
Brontobyte	BB	1024 YB
Geopbyte	GB	1024 BB



Características: banco de dados NoSQL [1 de 2]



- ★ Não usam SQL (No-SQL ou Not-only-SQL)
- ★ Geralmente são open-sources
- ★ Formas diferentes de organização (não usam tabelas)



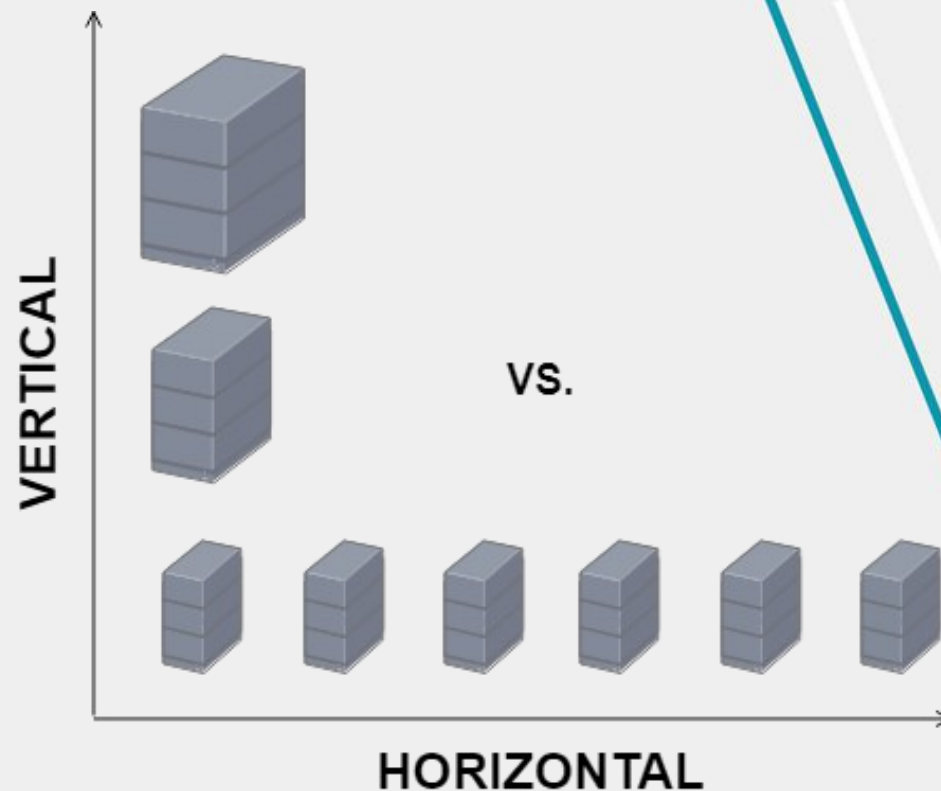
Copyright Kelly-McCreary & Associates, LLC



Características: banco de dados NoSQL [2 de 2]



- ★ Persistência poliglota: usam dois formatos de dados ao mesmo tempo
- ★ Clusterização: executar banco em várias máquinas ao mesmo tempo
- ★ Escala horizontal: possibilidade de aumentar a quantidade de computadores) etc.



Crédito: [Eduardo Lara](#)

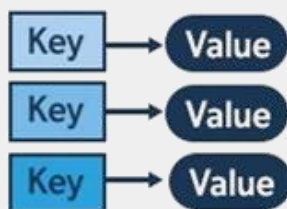




Modelos NoSQL: Chave-Valor

- ★ Banco tem sua formação a partir de estruturas de chaves, que são associadas a um único valor
- ★ Estrutura flexível e própria para armazenamento de BigData
- ★ Favorece a escalabilidade em detrimento da consistência

Key-Value



```
Nome_1010: José
Endereco_1010: Rua Y 9
Cidade_1010: São Paulo, Ribeirão Preto
Funcao_1010: DBA
```





Modelos NoSQL: Chave-Valor (+ usados)

Rank			DBMS	Database Model	Score		
Mar 2023	Feb 2023	Mar 2022			Mar 2023	Feb 2023	Mar 2022
1.	1.	1.	Redis +	Key-value, Multi-model	172.45	-1.39	-4.31
2.	2.	2.	Amazon DynamoDB +	Multi-model	80.77	+1.08	-1.03
3.	3.	3.	Microsoft Azure Cosmos DB +	Multi-model	36.10	-0.40	-4.79
4.	4.	4.	Memcached	Key-value	22.61	-0.56	-3.06
5.	5.	↑ 6.	Hazelcast	Key-value, Multi-model	8.63	-0.48	-1.43
6.	6.	↓ 5.	etcd	Key-value	8.60	+0.09	-3.23
7.	7.	↑ 10.	Aerospike +	Multi-model	6.54	-0.02	+0.31
8.	8.	↑ 9.	Ehcache	Key-value	6.03	-0.01	-0.52
9.	9.	↑ 14.	Google Cloud Bigtable	Multi-model	5.44	-0.47	+1.10
10.	10.	↓ 8.	Ignite	Multi-model	5.29	-0.21	-1.58

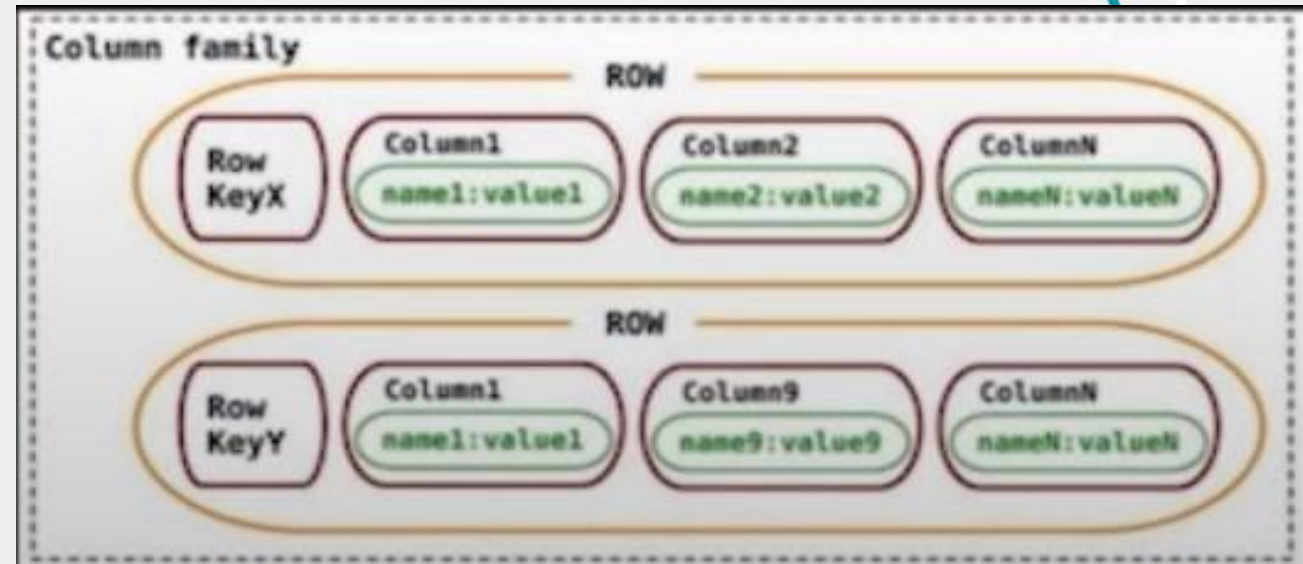


Modelos NoSQL: Colunas



- ★ Armazenamento em linhas particulares de tabelas
- ★ É o oposto do sistema relacional
- ★ É similar a uma tabela, onde cada linha pode ter uma ou mais colunas. Número de colunas pode variar

Column-Family



Crédito das figuras: [Geeks for Geeks](#) e [UNIVESP](#)





Modelos NoSQL: Colunas (+ usados)

Rank			DBMS	Database Model	Score		
Mar 2023	Feb 2023	Mar 2022			Mar 2023	Feb 2023	Mar 2022
1.	1.	1.	Cassandra +	Wide column	113.79	-2.43	-8.35
2.	2.	2.	HBase	Wide column	37.62	-0.79	-7.01
3.	3.	3.	Microsoft Azure Cosmos DB +	Multi-model i	36.10	-0.40	-4.79
4.	4.	4.	Datastax Enterprise +	Wide column, Multi-model i	7.33	-0.87	-2.56
5.	↑ 8.	↑ 7.	ScyllaDB +	Wide column, Multi-model i	5.76	+0.13	+1.77
6.	↓ 5.	↓ 5.	Microsoft Azure Table Storage	Wide column	5.73	-0.19	+0.09
7.	↓ 6.	↓ 6.	Google Cloud Bigtable	Multi-model i	5.44	-0.47	+1.10
8.	↓ 7.	8.	Accumulo	Wide column	5.39	-0.36	+1.45
9.	9.	9.	HPE Ezmeral Data Fabric	Multi-model i	1.17	-0.12	+0.35
10.	10.	10.	Amazon Keyspaces	Wide column	0.85	-0.06	+0.32

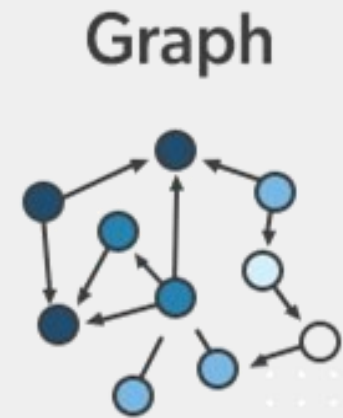
Crédito: [DB-Engines](#)





Modelos NoSQL: Grafos

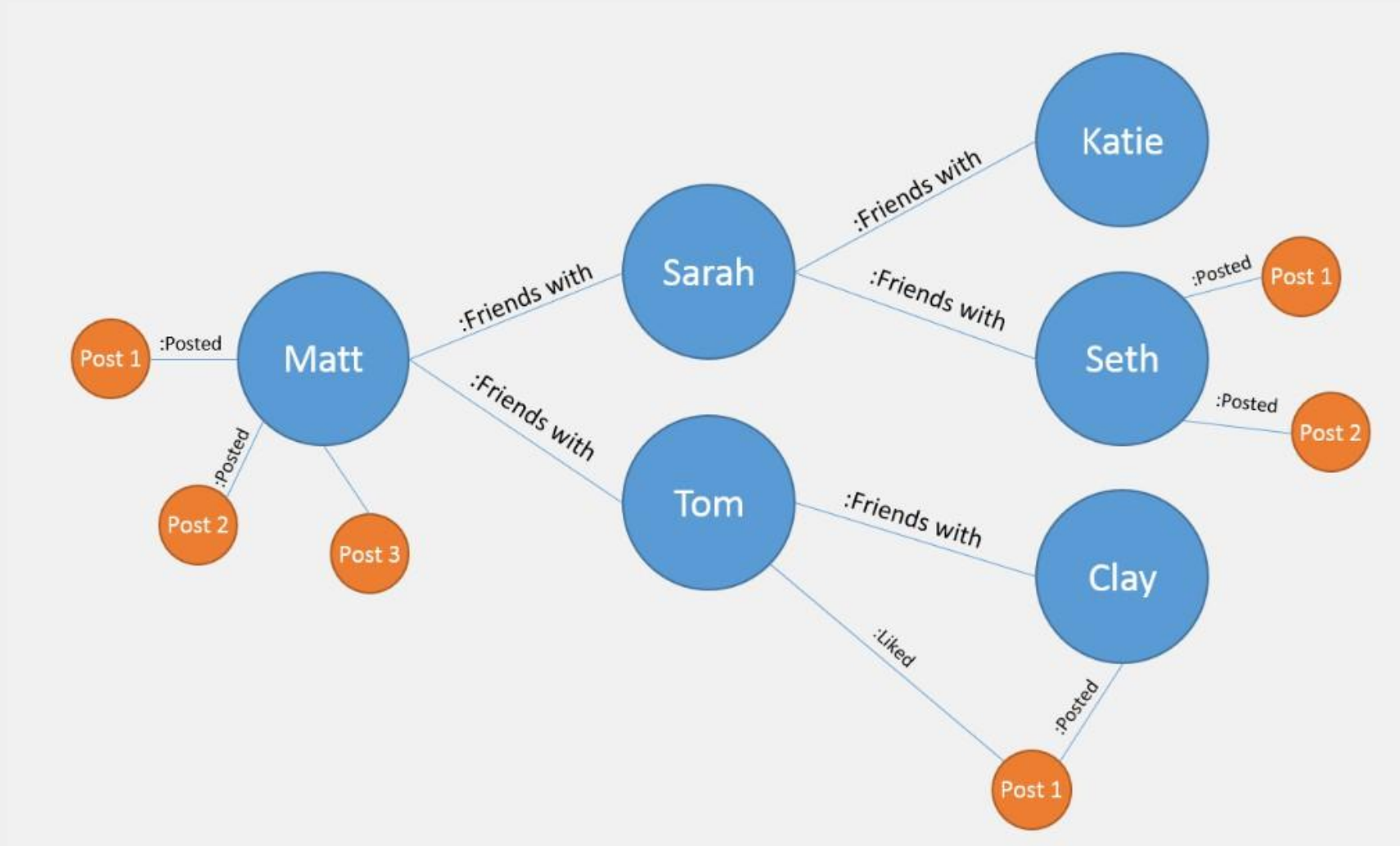
- ★ Modelo matemático formado por vértices (pontos) e arestas (ligações entre os pontos)
- ★ Semelhante ao modelo relacional, com estrutura de relacionamento por meio de atributos
- ★ Muito usado em redes sociais, bioinformática



Crédito da figura: [Geeks for Geeks](https://www.geeksforgeeks.org/graph-database/)



Modelos NoSQL: Grafos - exemplo



Crédito da figura: [Luiz Tools](#)



Modelos NoSQL: Grafos (+ usados)



Rank			DBMS	Database Model	Score		
Mar 2023	Feb 2023	Mar 2022			Mar 2023	Feb 2023	Mar 2022
1.	1.	1.	Neo4j +	Graph	53.51	-1.92	-6.16
2.	2.	2.	Microsoft Azure Cosmos DB +	Multi-model i	36.10	-0.40	-4.79
3.	3.	↑ 4.	Virtuoso +	Multi-model i	6.39	+0.29	+0.82
4.	4.	↓ 3.	ArangoDB +	Multi-model i	5.04	-0.26	-0.57
5.	5.	5.	OrientDB	Multi-model i	4.30	-0.24	-0.63
6.	↑ 7.	↑ 7.	Amazon Neptune	Multi-model i	2.60	-0.13	-0.09
7.	↓ 6.	↑ 8.	JanusGraph	Graph	2.56	-0.24	+0.09
8.	8.	↓ 6.	GraphDB +	Multi-model i	2.32	-0.12	-0.52
9.	9.	9.	TigerGraph	Graph	1.96	-0.21	-0.22
10.	↑ 12.	↑ 15.	NebulaGraph +	Graph	1.83	+0.04	+0.70

Crédito: [DB-Engines](#)



Modelos NoSQL: Documentos



- ★ Dados são compreendidos como documentos
- ★ Estrutura flexível, não precisa de colunas pré-montadas
- ★ Eficiente para tratar dados não estruturados, já que cada documento pode ter um formato diferente;
- ★ Cada documento é um arquivo JSON

Document



Modelos NoSQL: Documentos - ex.



```
{
  "book": [
    {
      "id": "444",
      "language": "C",
      "edition": "First",
      "author": "Dennis Ritchie "
    },
    {
      "id": "555",
      "language": "C++",
      "edition": "second",
      "author": " Bjarne Stroustrup "
    }
  ]
}
```

Json

```
_id: ObjectId("5eb3d668b31de5d588f4292a")
> address: Object
  borough: "Brooklyn"
  cuisine: "American"
  grades: Array
    0: Object
      date: 2014-06-10T00:00:00.000+00:00
      grade: "A"
      score: 5
    1: Object
    2: Object
    3: Object
  name: "Riviera Caterer"
  restaurant_id: "40356018"
```

```
_id: ObjectId("5eb3d668b31de5d588f4292b")
> address: Object
  borough: "Brooklyn"
  cuisine: "Delicatessen"
  grades: Array
    0: Object
    1: Object
    2: Object
    3: Object
    4: Object
    5: Object
  name: "Wilken'S Fine Food"
  restaurant_id: "40356483"
```

BD - Documento





Modelos NoSQL: Documentos (+ usados)

Rank			DBMS	Database Model	Score		
Mar 2023	Feb 2023	Mar 2022			Mar 2023	Feb 2023	Mar 2022
1.	1.	1.	MongoDB +	Document, Multi-model i	458.78	+6.02	-26.88
2.	2.	2.	Amazon DynamoDB +	Multi-model i	80.77	+1.08	-1.03
3.	3.		Databricks	Multi-model i	60.86	+0.52	
4.	4.	↓ 3.	Microsoft Azure Cosmos DB +	Multi-model i	36.10	-0.40	-4.79
5.	5.	↓ 4.	Couchbase +	Document, Multi-model i	23.36	-1.50	-6.09
6.	6.	↓ 5.	Firebase Realtime Database	Document	18.78	+0.29	-0.80
7.	7.	↓ 6.	CouchDB	Document, Multi-model i	14.46	+0.01	-3.02
8.	8.	↑ 9.	Google Cloud Firestore	Document	11.36	-0.15	+2.21
9.	9.	↓ 7.	MarkLogic	Multi-model i	8.86	+0.02	-1.04
10.	10.	↓ 8.	Realm	Document	8.53	+0.28	-1.30

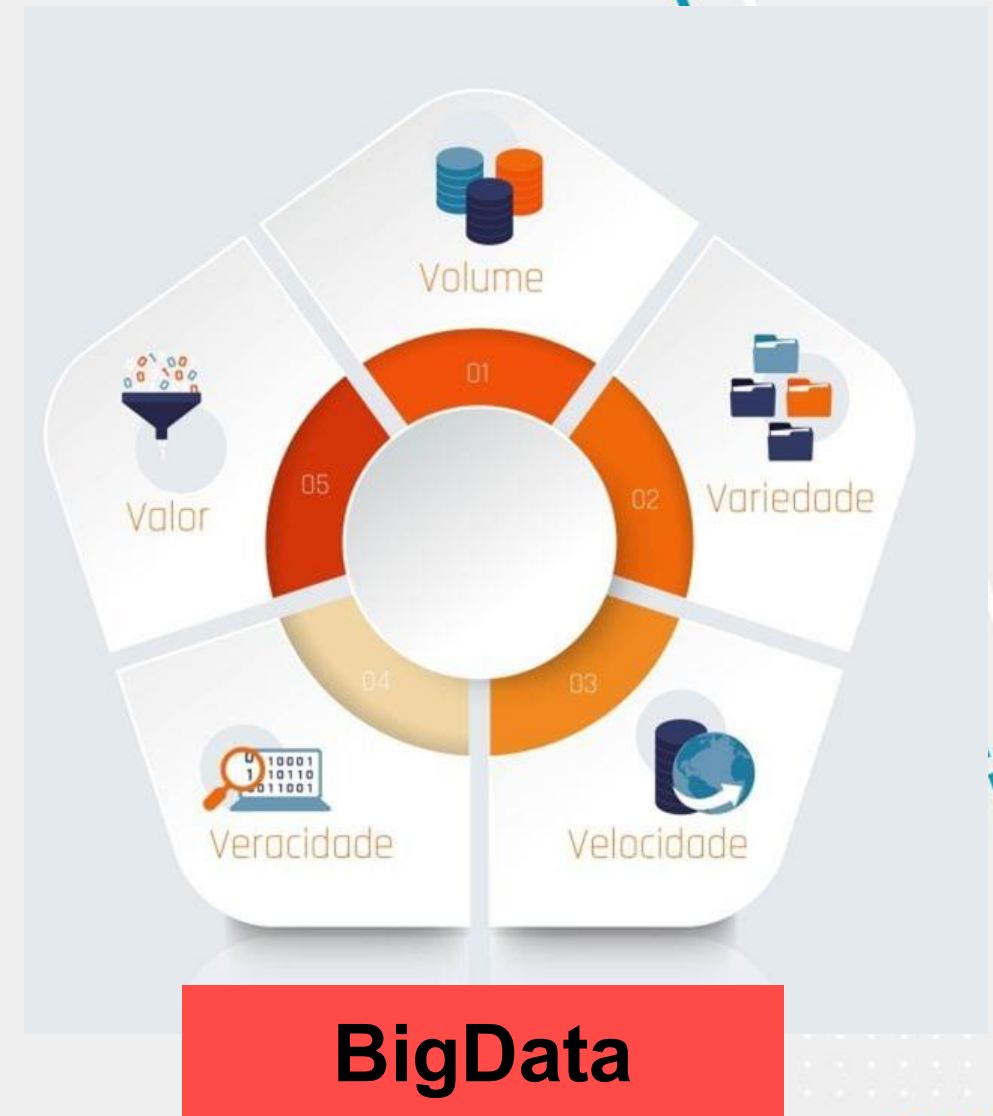
Crédito: [DB-Engines](#)





Por que utilizar NoSQL?

- ★ Dificuldade de modelar bancos de dados complexos, com inúmeras ligações entre as tabelas;
- ★ Necessidade de implementar bancos de dados com estruturas flexíveis;
- ★ Necessidade de escalonamento horizontal (volumes grandes)
- ★ Disponibilidade



Crédito da figura: [Blog Neoway](#)





DÚVIDAS?



Créditos:

Profª: Carolina Sacramento
Carolina.sacramento@fiocruz.br

