

# Modelagem de Banco de dados com ER – Entidade Relacionamento

*Trilha de Banco de Dados Relacional*

**Juliana Mascarenhas**

Tech Education Specialist DIO / Criadora de conteúdo no  
@Simplificandoredes e @SimplificandoProgramação

Mestre em modelagem computacional | Cientista de dados

**@in/juliana-mascarenhas-ds/**

# Objetivo Geral

Objetivo deste módulo é apresentar o mundo da modelagem de dados voltado para um sistema de banco de dados.

# Percurso

**Etapa 1**

Introdução à Modelagem e Projeto de BD

**Etapa 2**

Explorando o Modelo ER

**Etapa 3**

Esquema Conceitual Através de Diagramas

**Etapa 4**

Modelagem de dados complexa com ER

**Etapa 5**

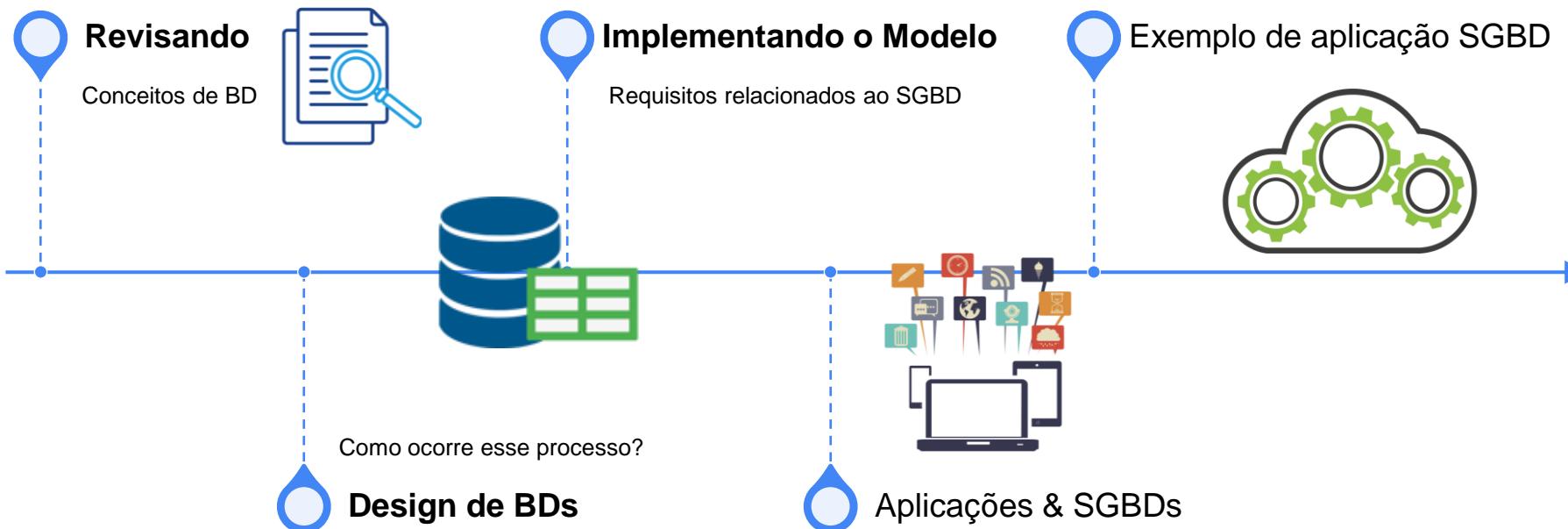
Construindo seu Primeiro Projeto de BD

## Etapa 1

# Introdução à Modelagem & Projeto de BD

//Modelagem de Banco de Dados com ER -

# Objetivo Geral



# Discursando sobre as características de Banco de Dados



# Relembrando

- Gerência de dados
- Isolamento data/program
- Integridade e Consistência
- Segurança
- Views
- Recuperação
- Recovery
- Concorrência

# Mundo Fechado

Preposição

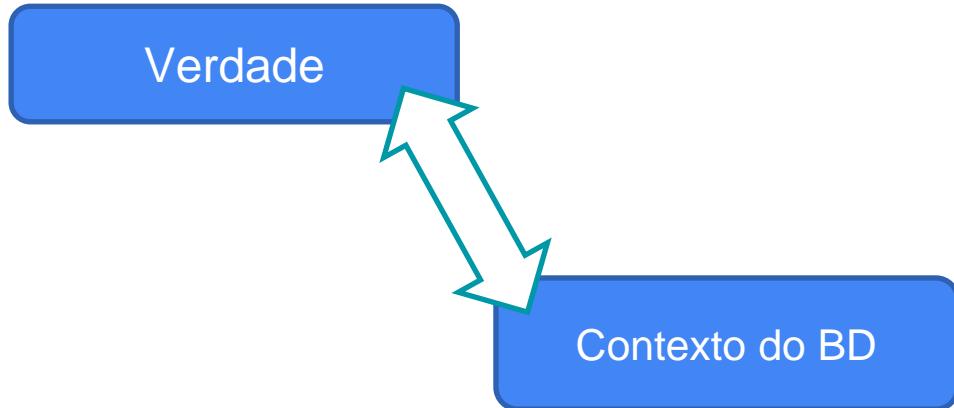
Lógica formal do sistema que  
vai além de BD

Predicado

Close World Assumption



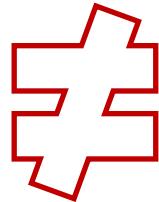
# Mundo Fechado



Close World Assumption

# Mundo Fechado

Open World Assumption



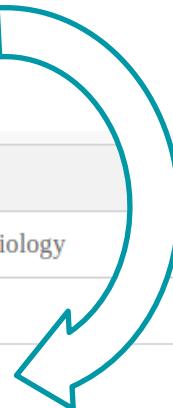
Close World Assumption



# Mundo Fechado

Grau menor que PhD?

Alumnus	Degree obtained
Delani	PhD in Molecular Biology
Anna	PhD in Ecology
Peter	MSc in Informatics
Dalila	PhD in Genetics



Completude

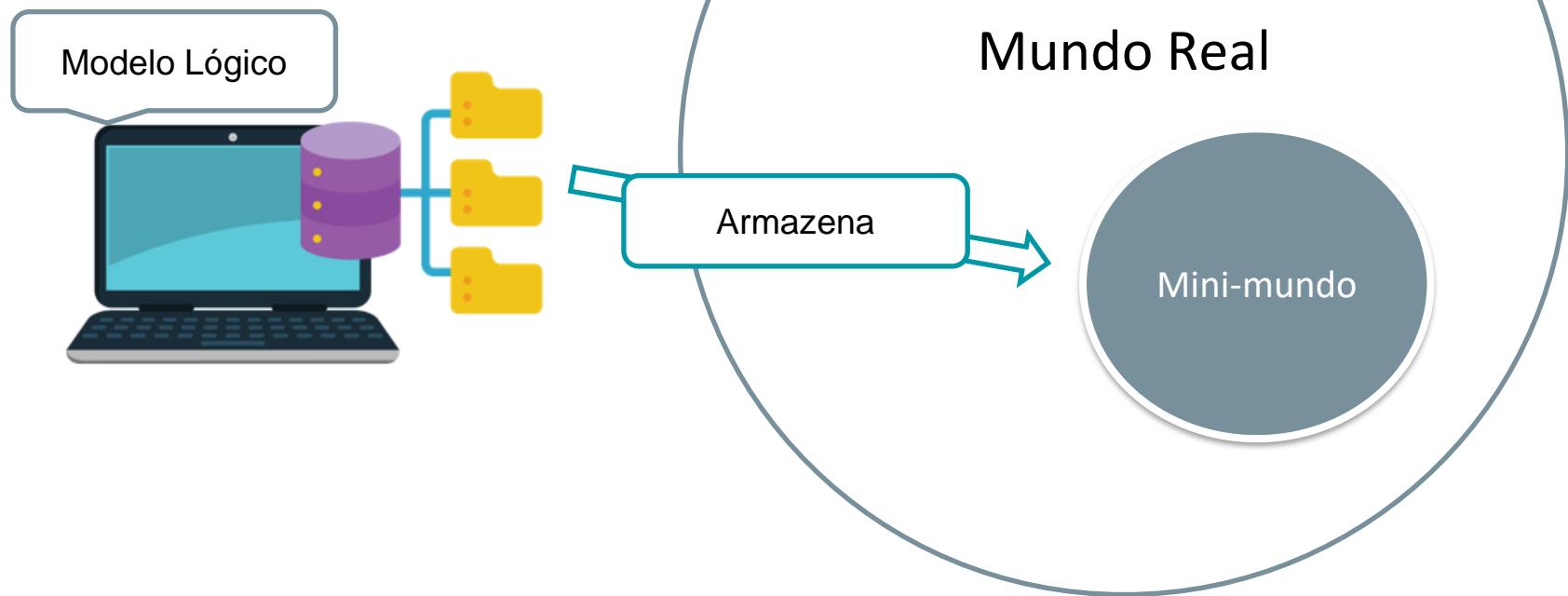


# Mini-mundo

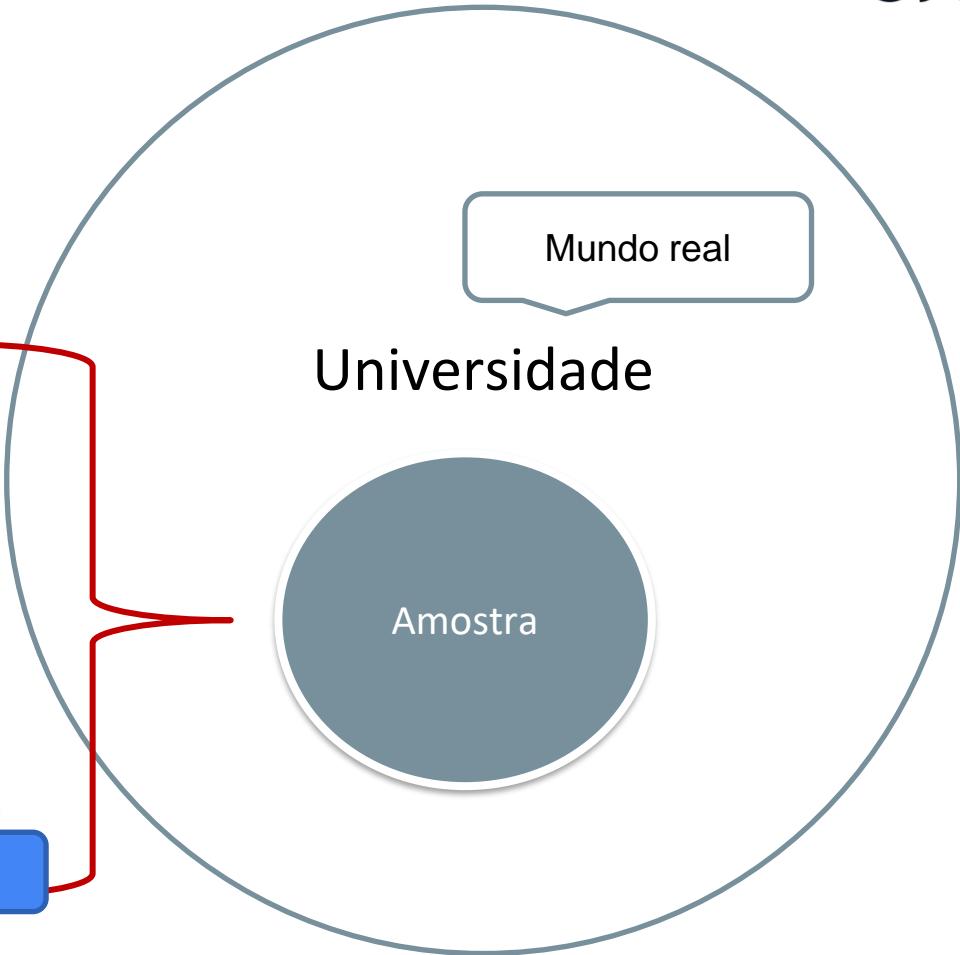


Universe of Discourse

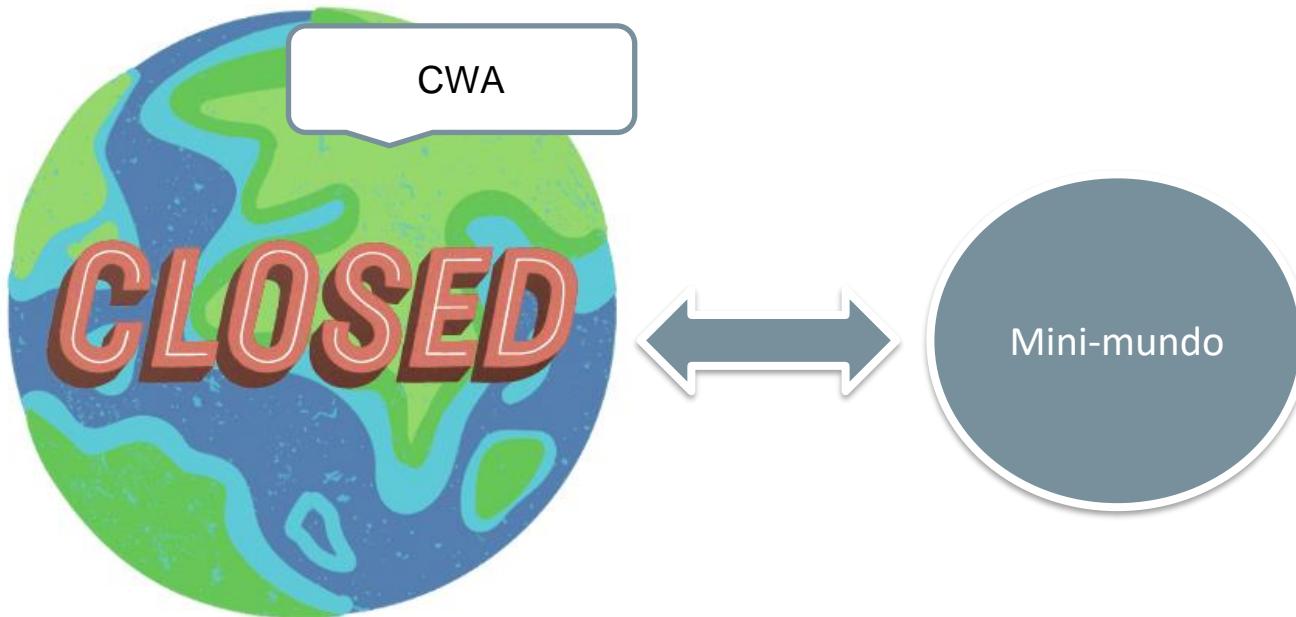
# Mini-mundo



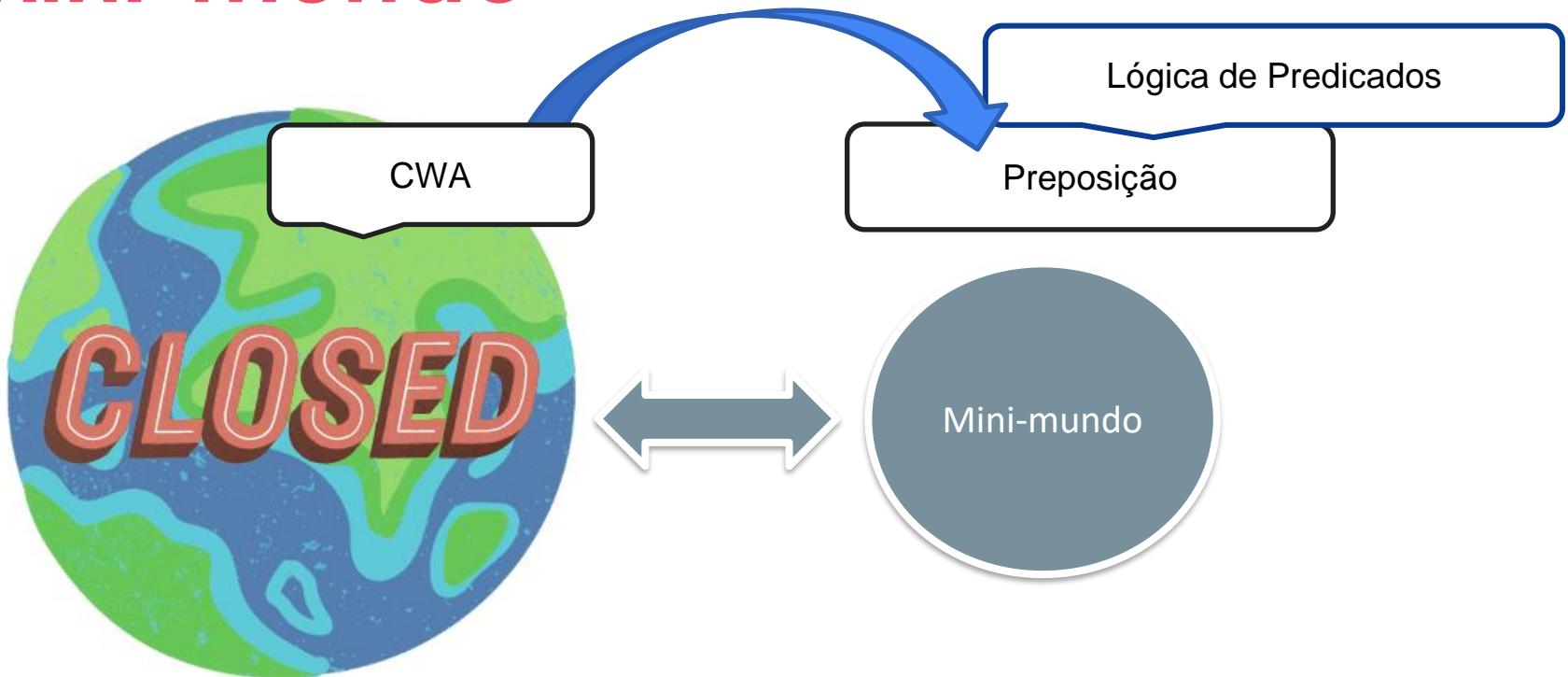
# Mini-mundo



# Mini-mundo

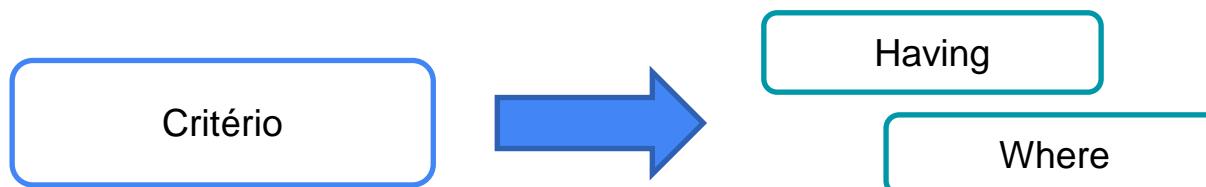


# Mini-mundo



# Álgebra Relacional

O **predicado** é a parte da oração que contém o verbo e que traz informações sobre o sujeito.



# Álgebra Relacional

Teoria dos Conjuntos

Linguagem formal para consulta/extração de dados

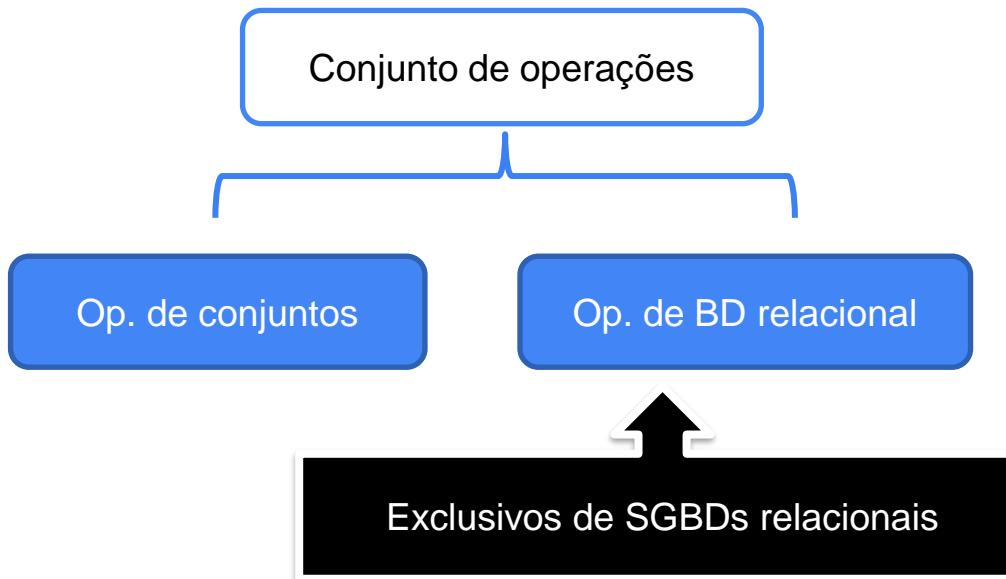
# Álgebra Relacional

Teoria dos Conjuntos

Linguagem formal para consulta/extração de dados



# Álgebra Relacional



Selection	$\sigma$
Projection	$\pi$
Renaming	$\rho$
Union	$\cup$
Intersection	$\cap$
Difference	$-$
Cartesian product	$\times$
Join	$\bowtie$
Logical AND	$\wedge$
Logical OR	$\vee$
Logical NOT	$\sim$

# Álgebra Relacional

ANY

MAX



AVG

COUNT



Conjunto de Funções

+

SUM

MIN

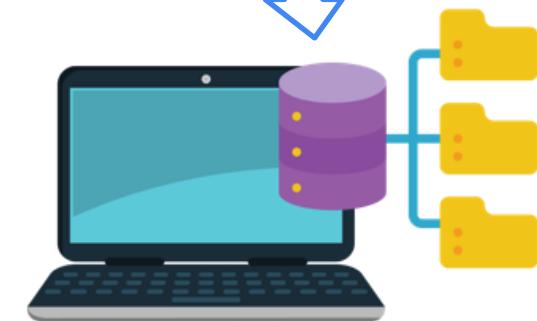
# Alg. Relacional

+

-

÷

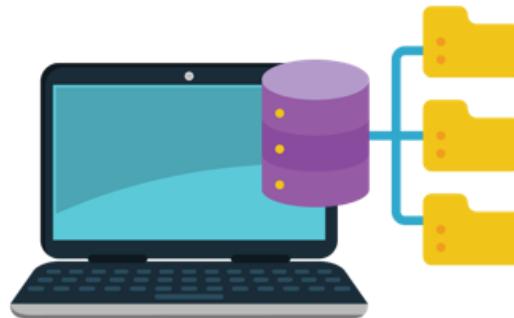
Álgebra Relacional



Resultado da ação

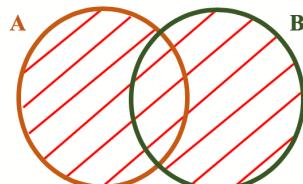
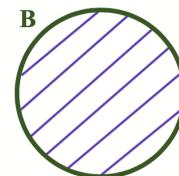
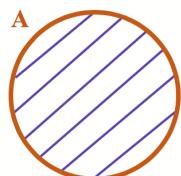
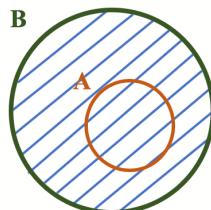
# Alg. Relacional

Lembra da dependência  
aplicação/dados?



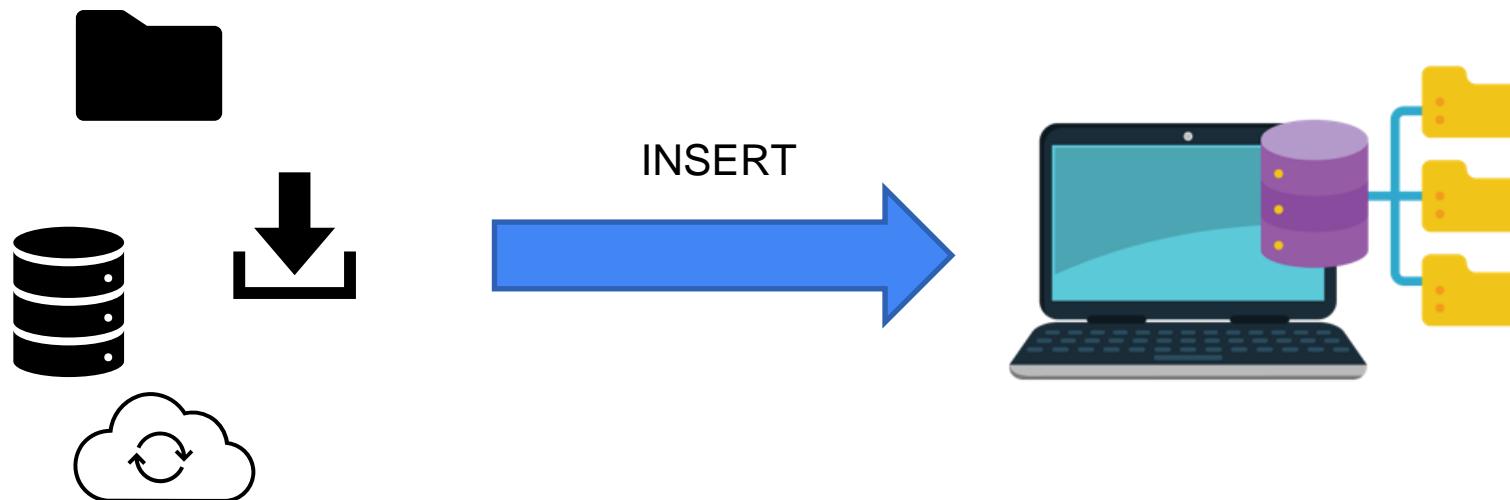
# Alg. Relacional

Lembra da ~~dependência~~  
aplicação/dados?

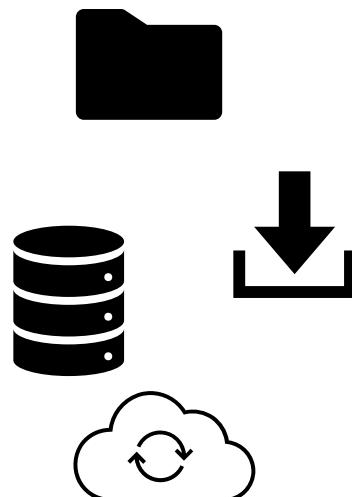


Consultas complexas

# Tradeoff



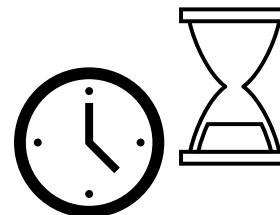
# Tradeoff



Disponibilidade

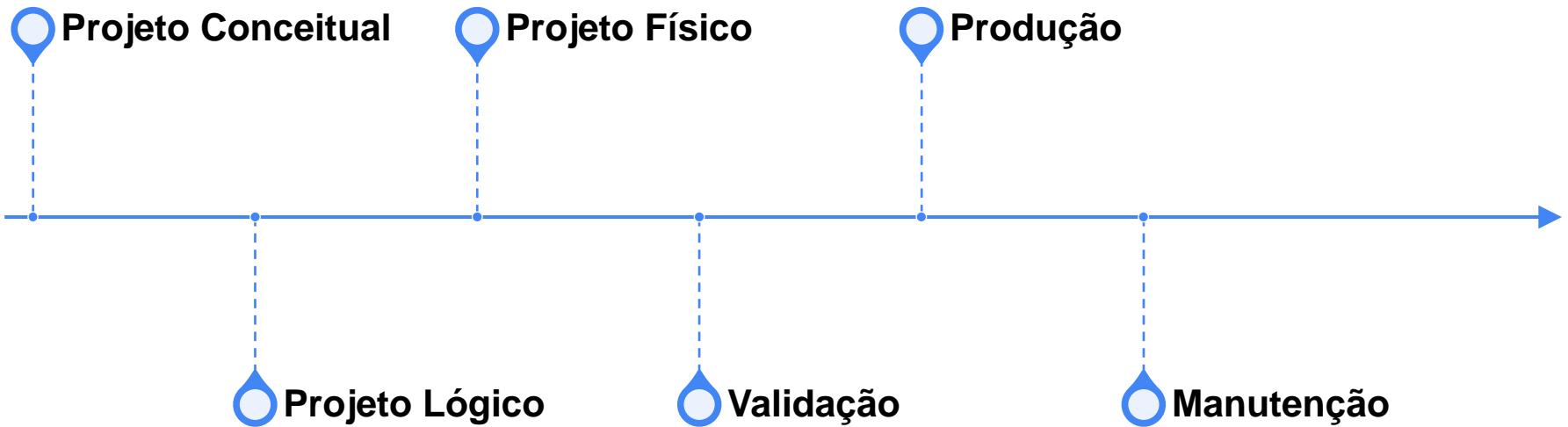


Clock do sistema



Tempo de gravação

# Processo



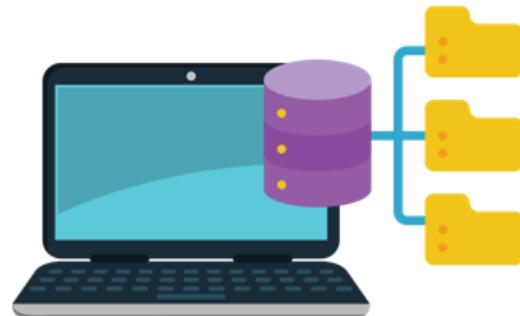
# Falando sobre modelagem ...



# Modelagem de dados

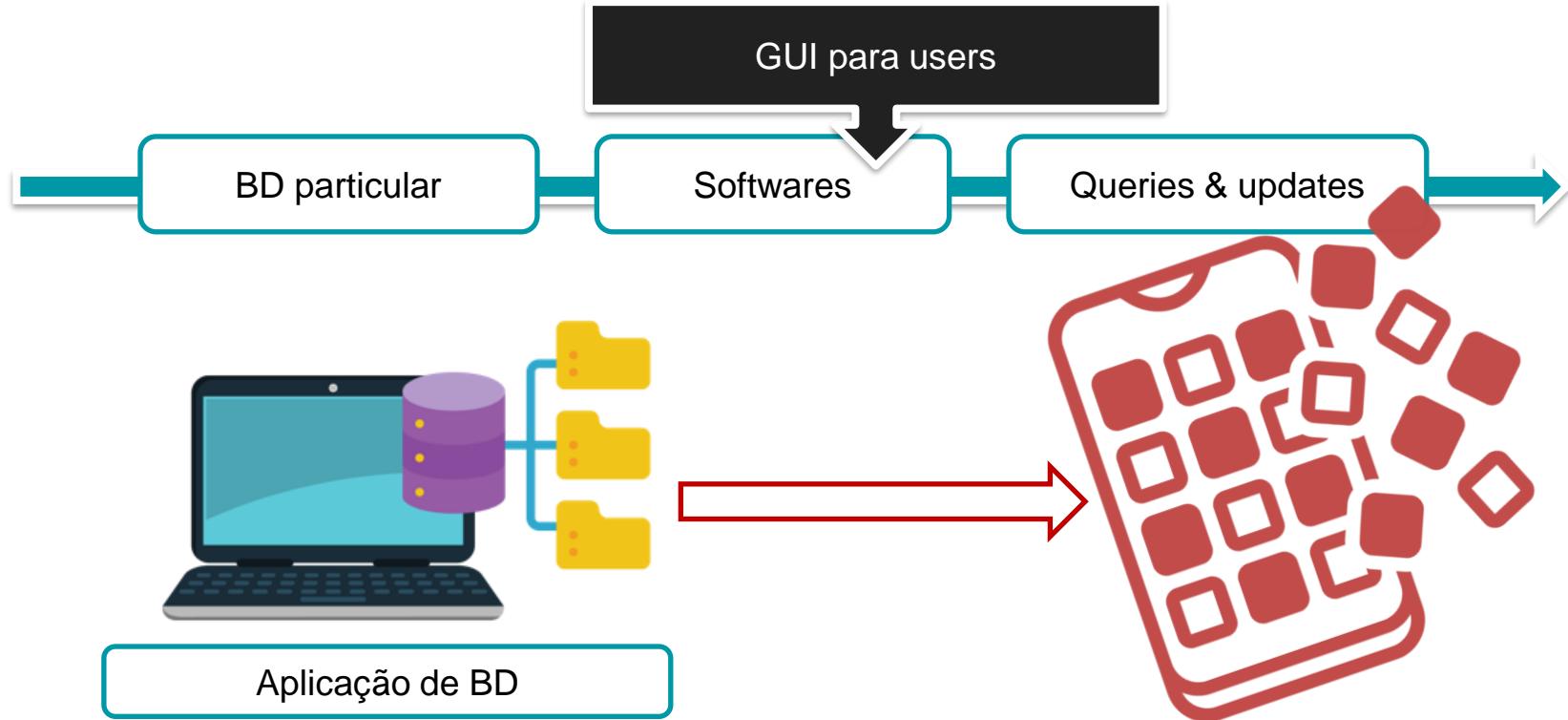


# Modelagem de dados

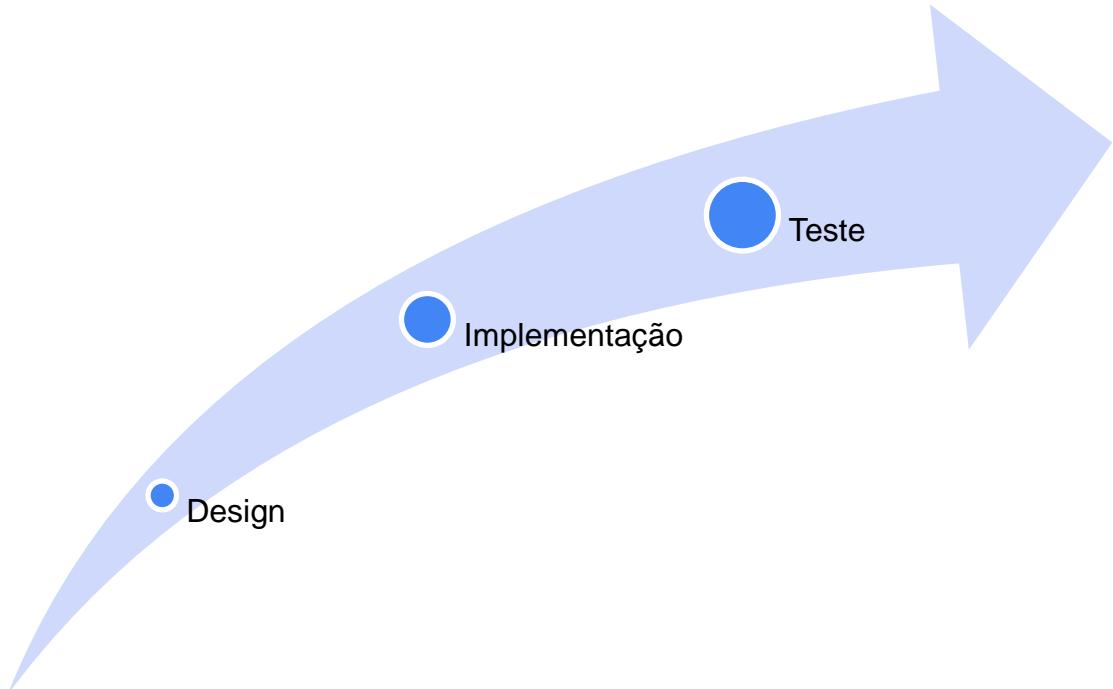


Aplicação de BD

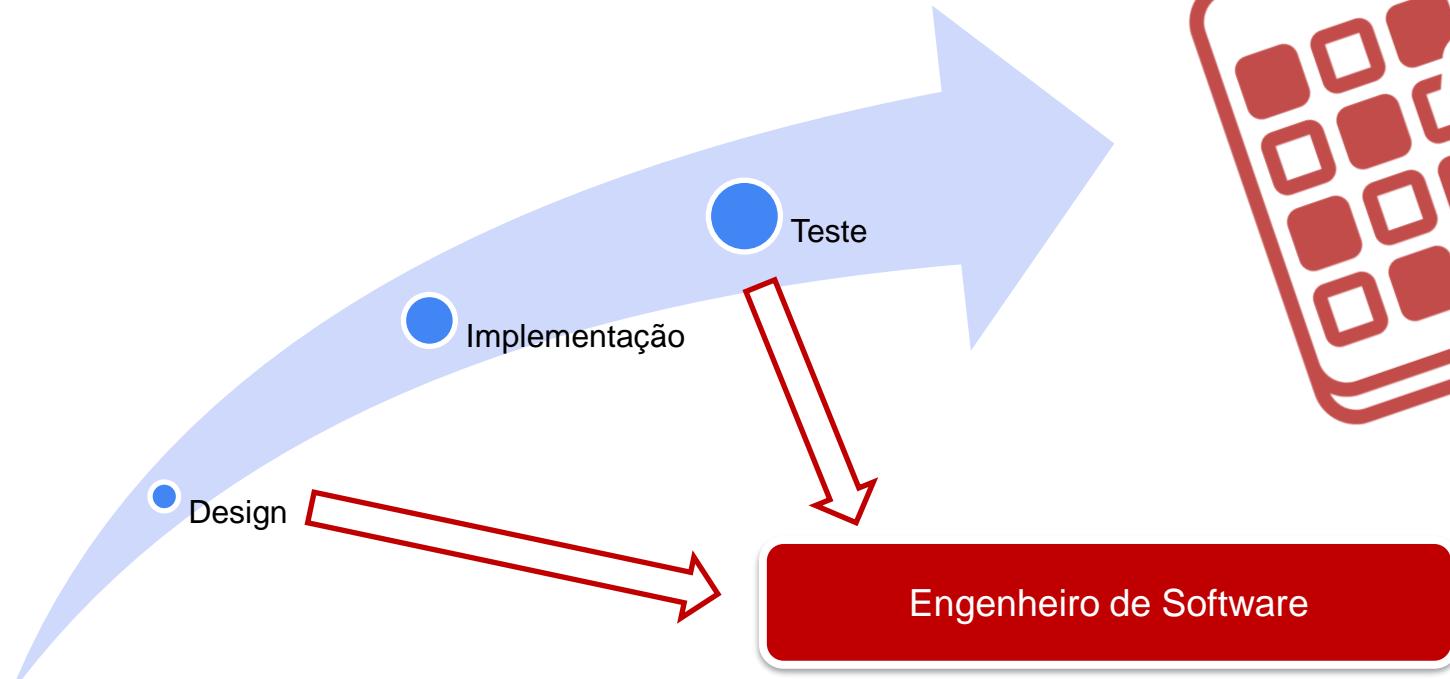
# Modelagem de dados



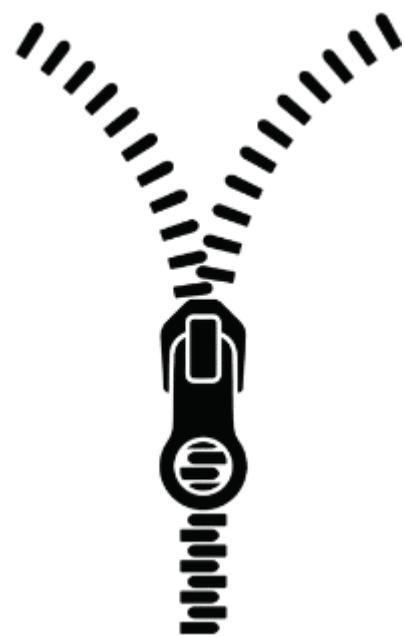
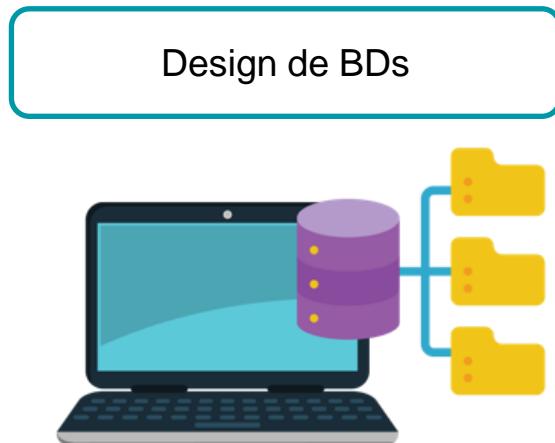
# Modelagem de dados



# Modelagem de dados



# Desenvolvimento

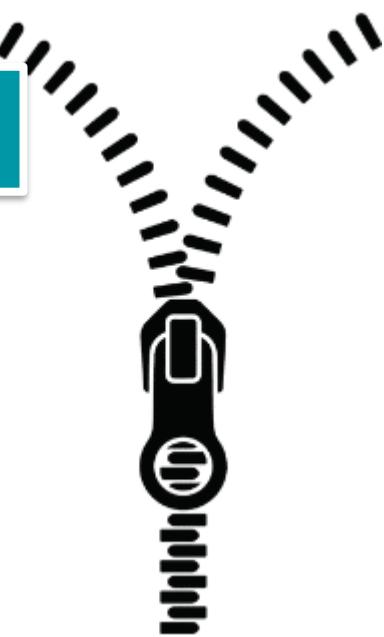


Engenharia de software

# Desenvolvimento

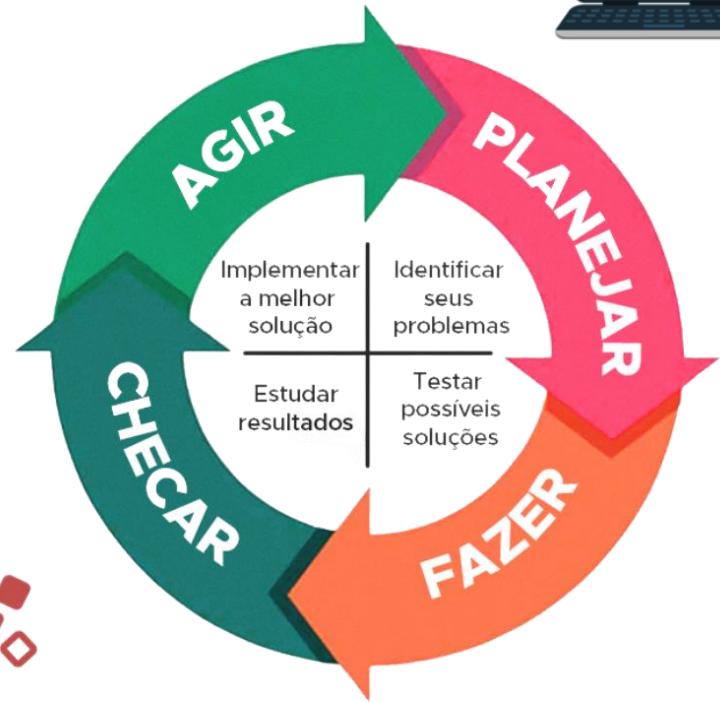
Modelo Entidade Relacionamento

Design de BDs



Engenharia de software

# Desenvolvimento



# Projeto: Como "nasce" um Banco de Dados?



# Desenvolvimento



Como implementar um BD?

# Desenvolvimento



Como implementar um BD?

- Entender o contexto e requisitos
- Perfil ...

O que eu quero representar?

Processo evolutivo ou gradual

# Desenvolvimento



Implementação

Arquitetura

Modelo

Funcionalidades

O que eu quero representar?

# Cenários



Colaboradores

E-commerce

Banco (Financeiro)

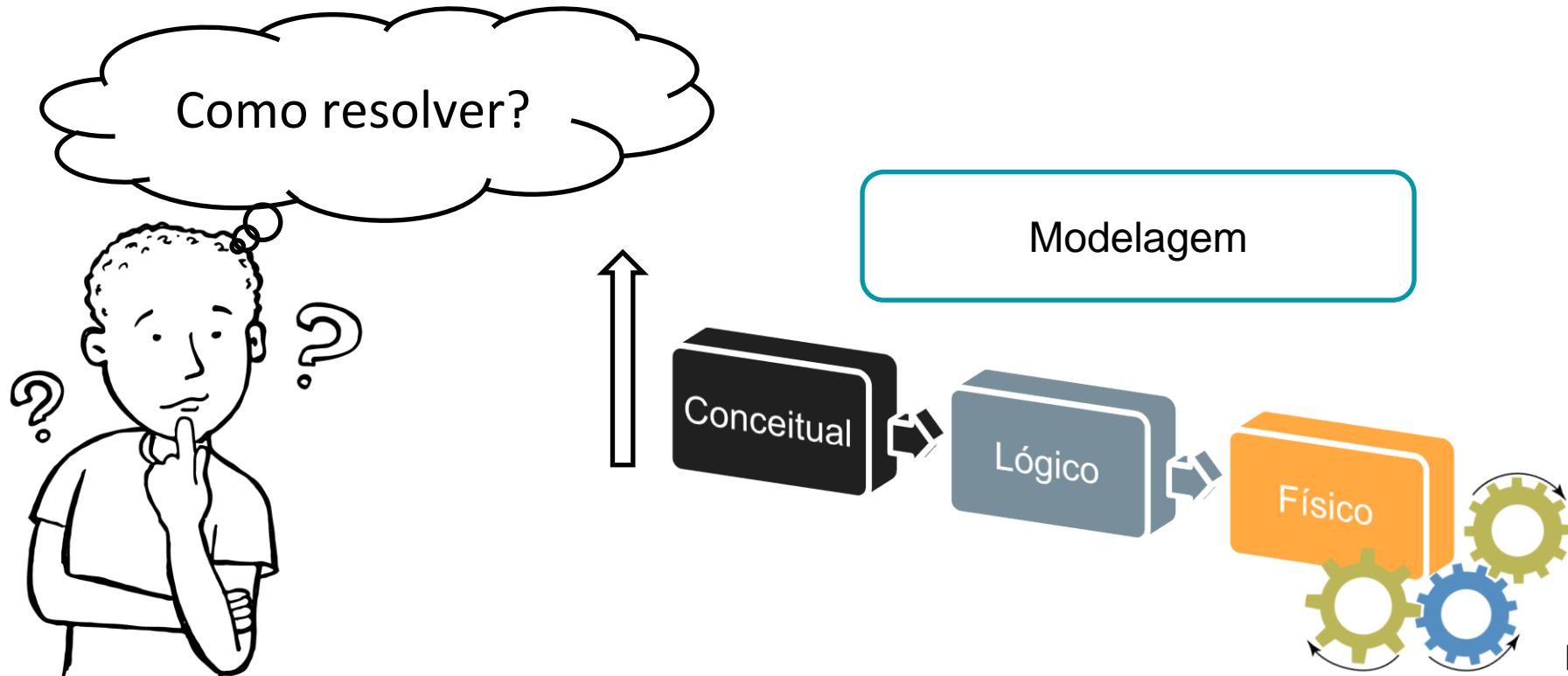
Universidade

Farmácia

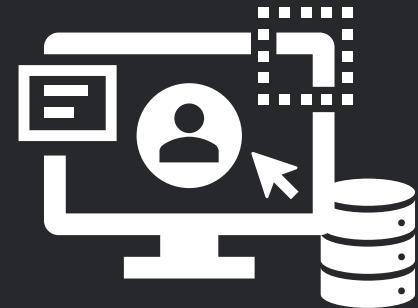
Produção

Bibliotecas

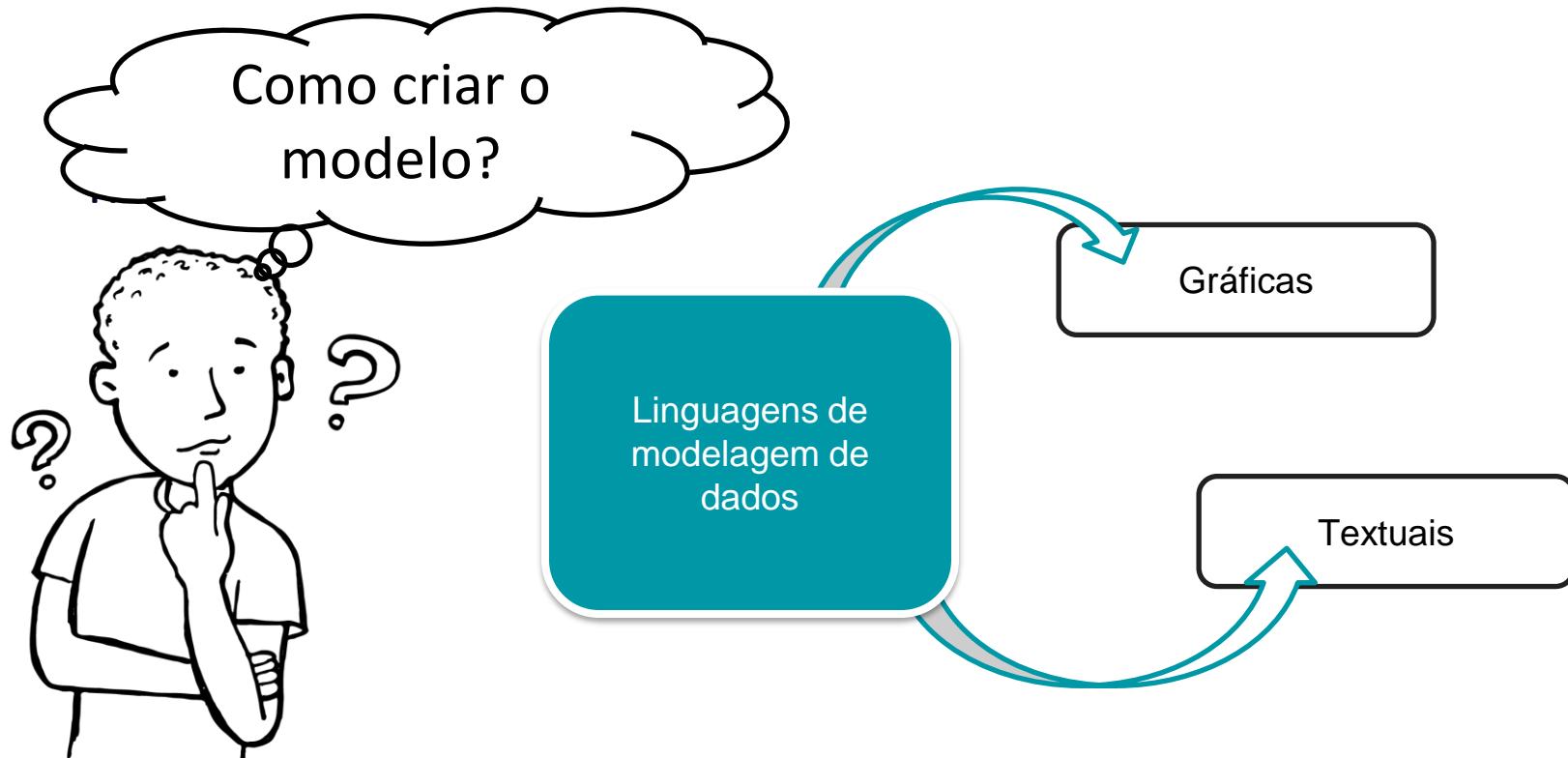
# Desenvolvimento



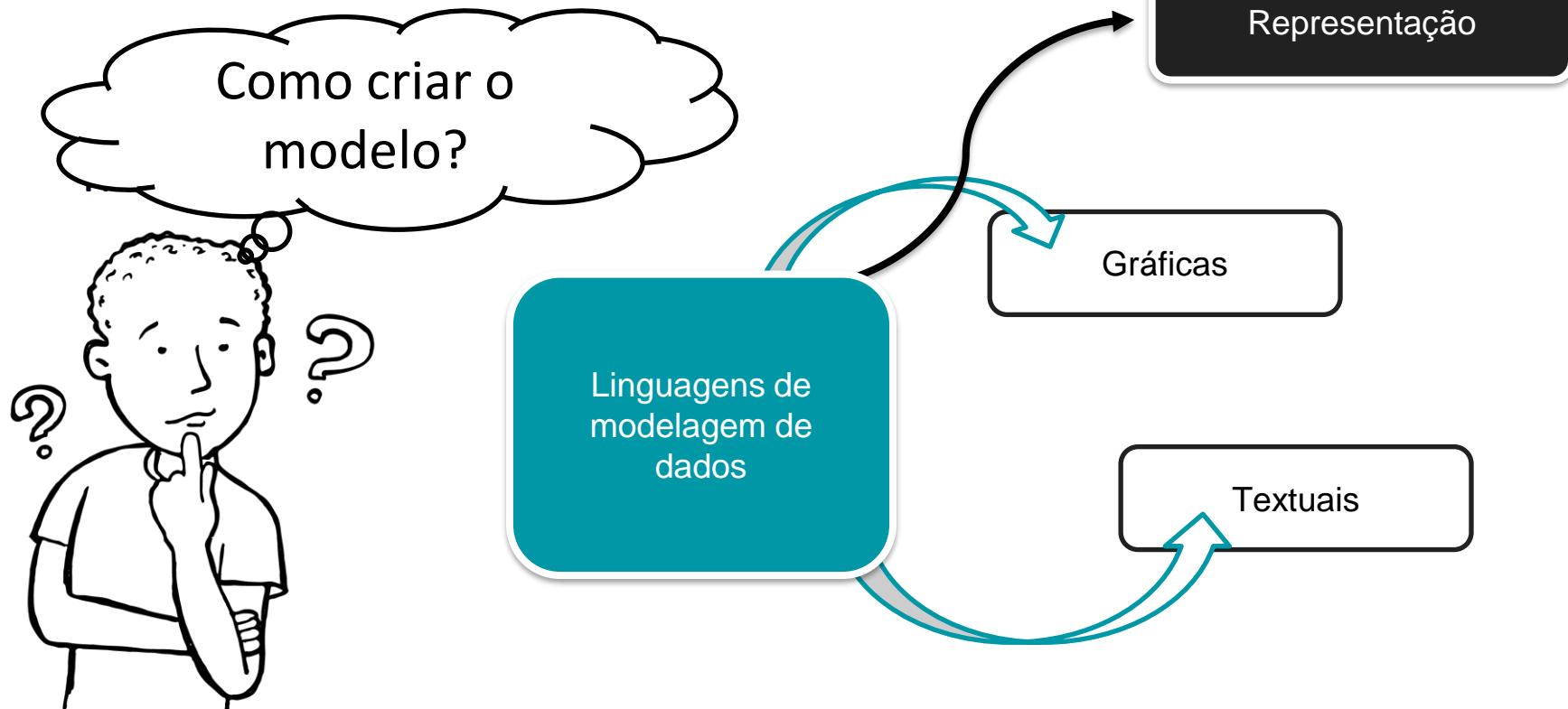
# Design de BDs – Projeto conceitual



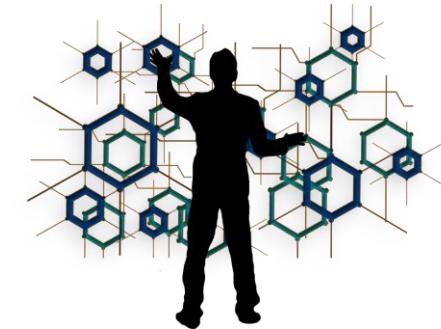
# Projeto Conceitual



# Projeto Conceitual



# Projeto Conceitual



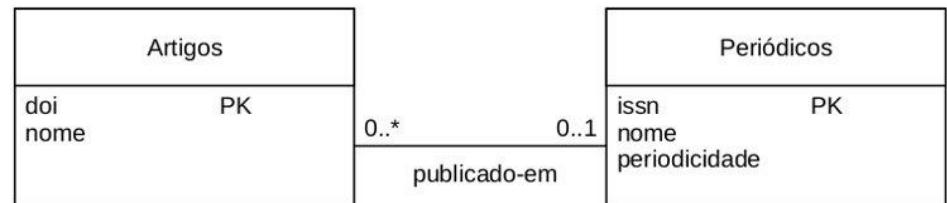
Requisitos, perguntas a serem respondidas, visões ...

Coleta de Dados

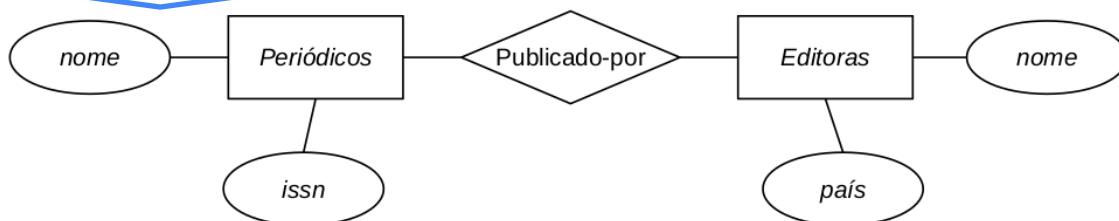
Análise

# Projeto Conceitual

UML – Diagrama de classes



Modelo Entidade Relacionamento

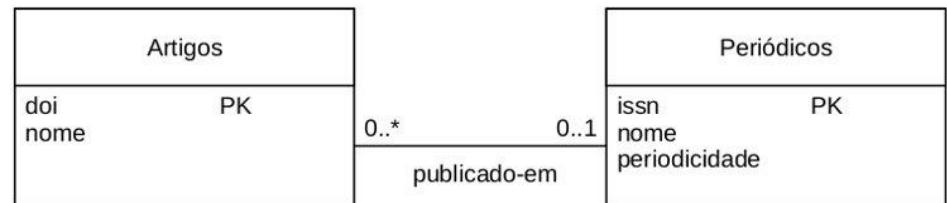


Fonte: livro de referência - Navathe

Esquema Conceitual

# Projeto Conceitual

UML – Diagrama de classes



Modelo Entidade Relacionamento

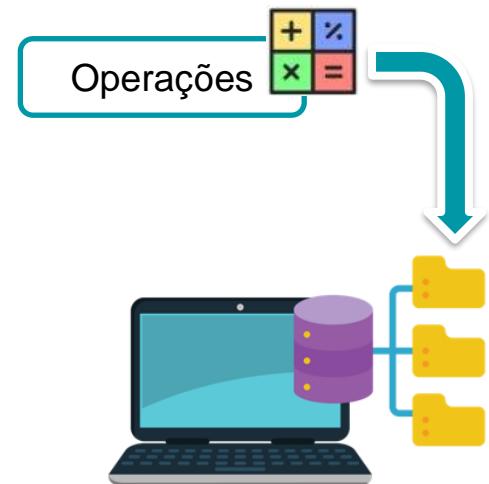


DER

Fonte: livro de referência - Navathe

Esquema Conceitual

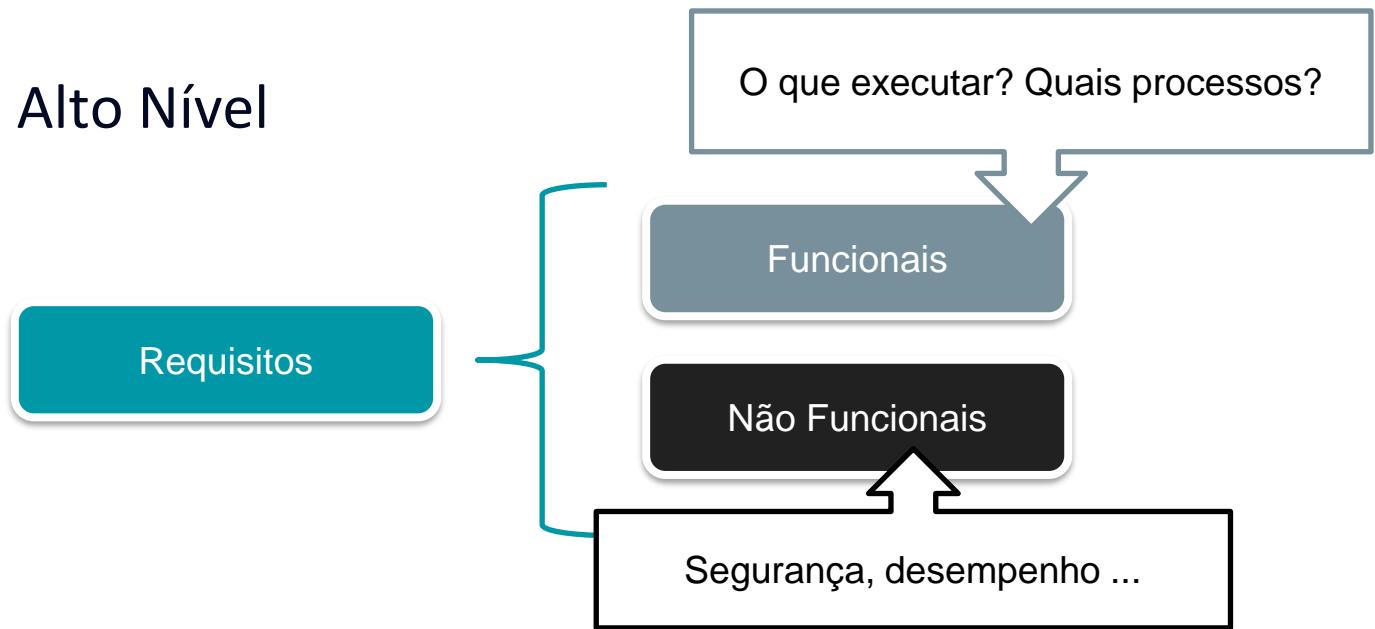
# Projeto Conceitual



Foco do engenheiro de software

# Projeto Conceitual

Modelo de Alto Nível



# Projeto Conceitual

Modelo de Alto Nível

Requisitos



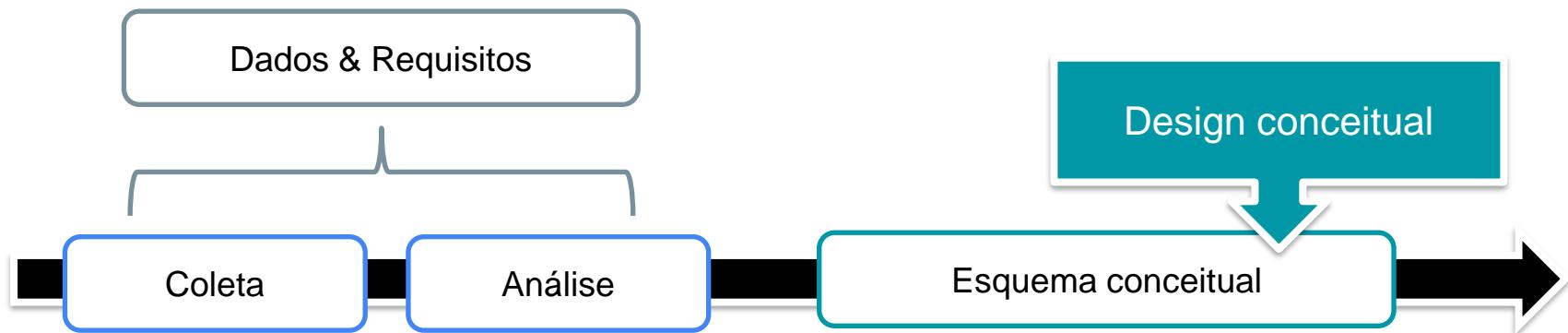
Funcionais

Não Funcionais

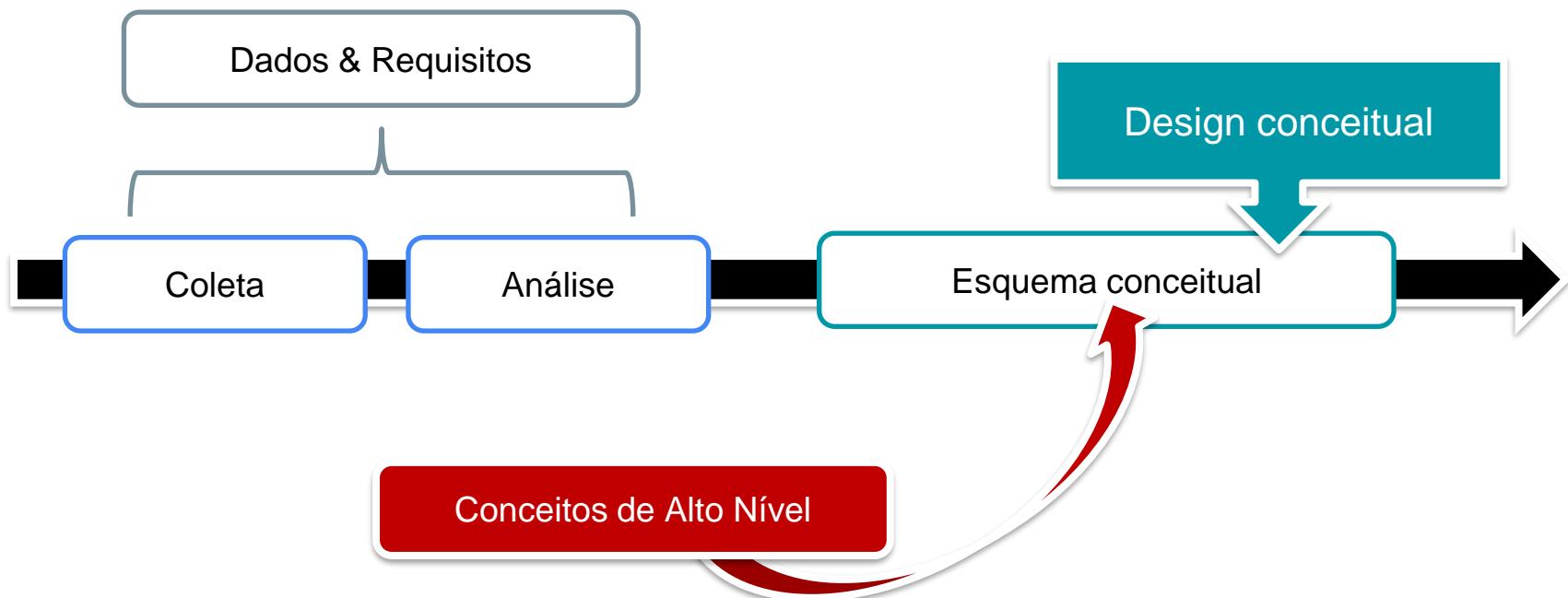
Como será armazenado?

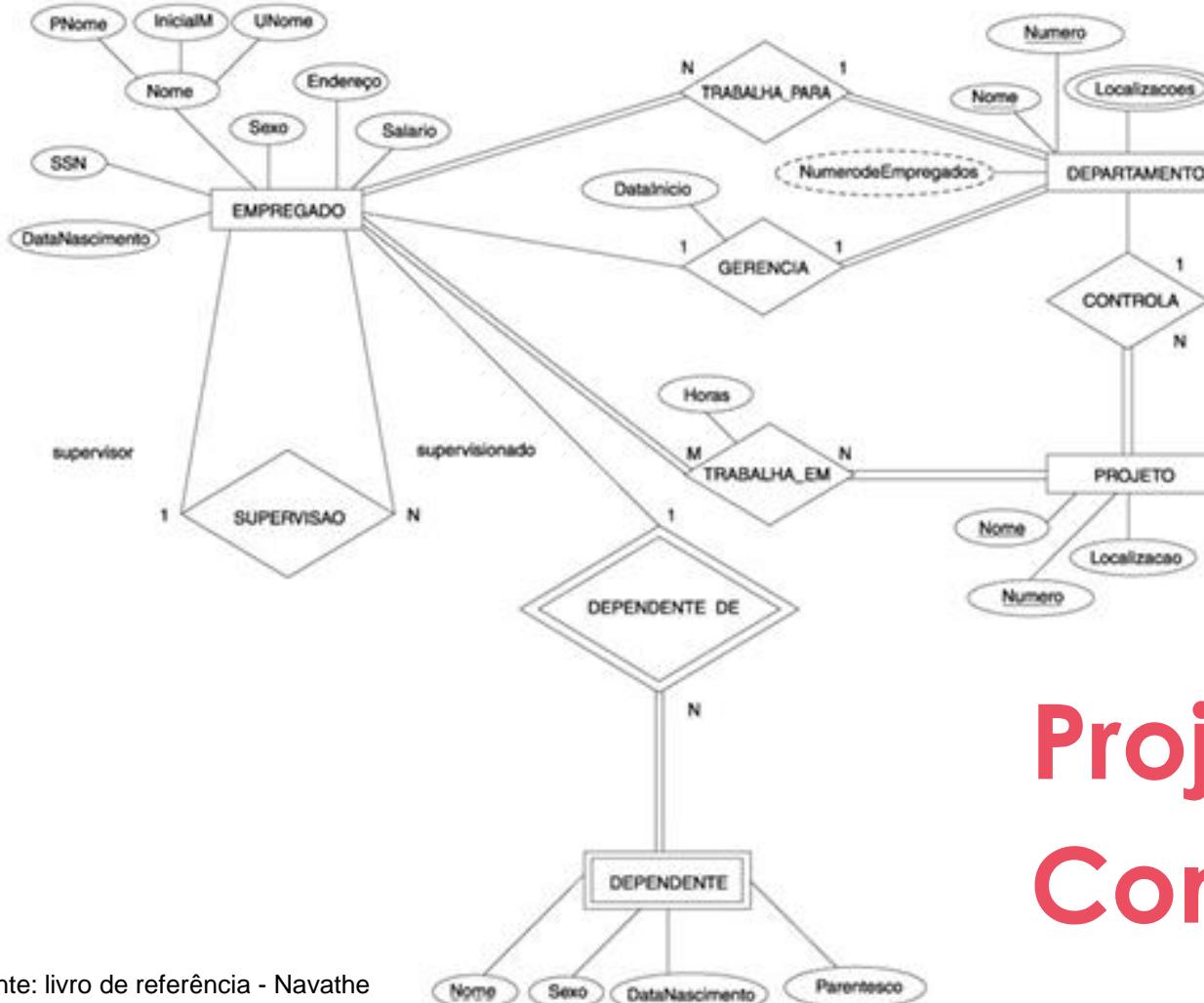
Independente do modelo de Banco de Dados

# Projeto Conceitual



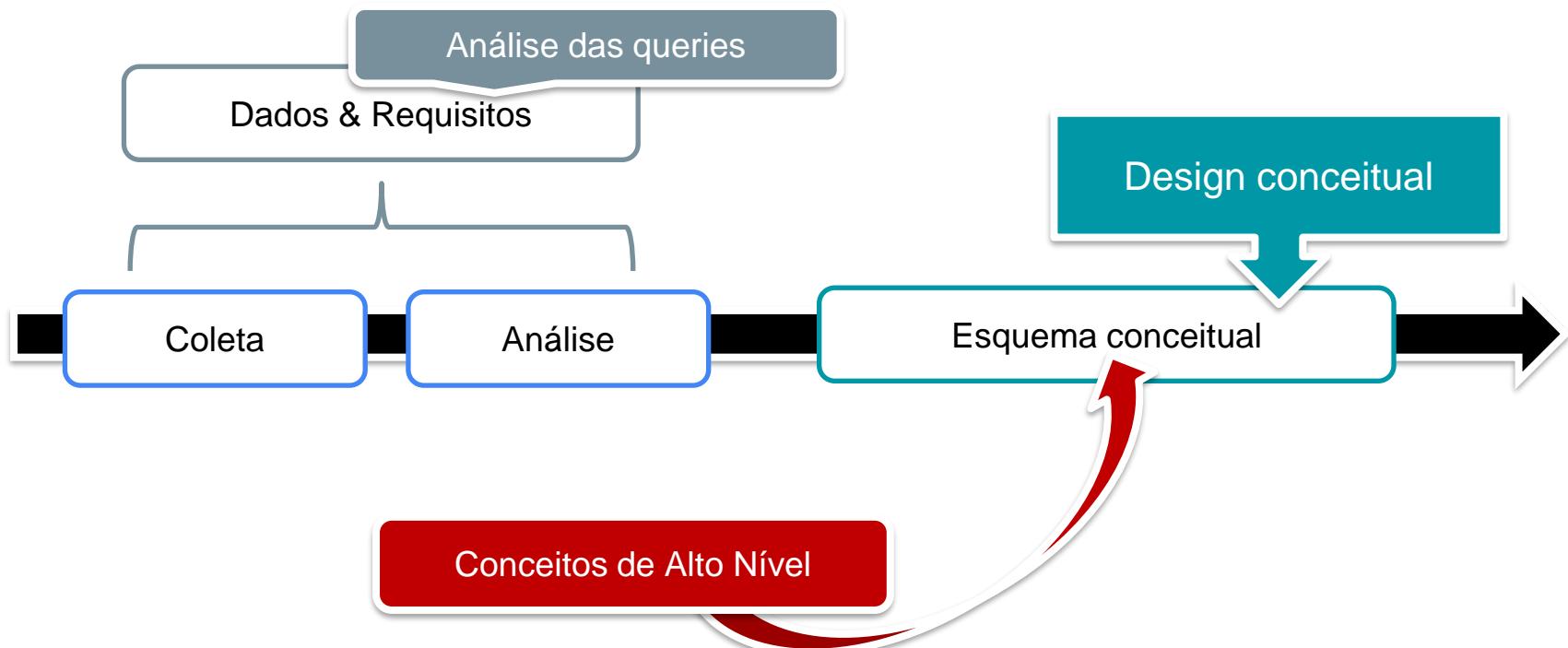
# Projeto Conceitual





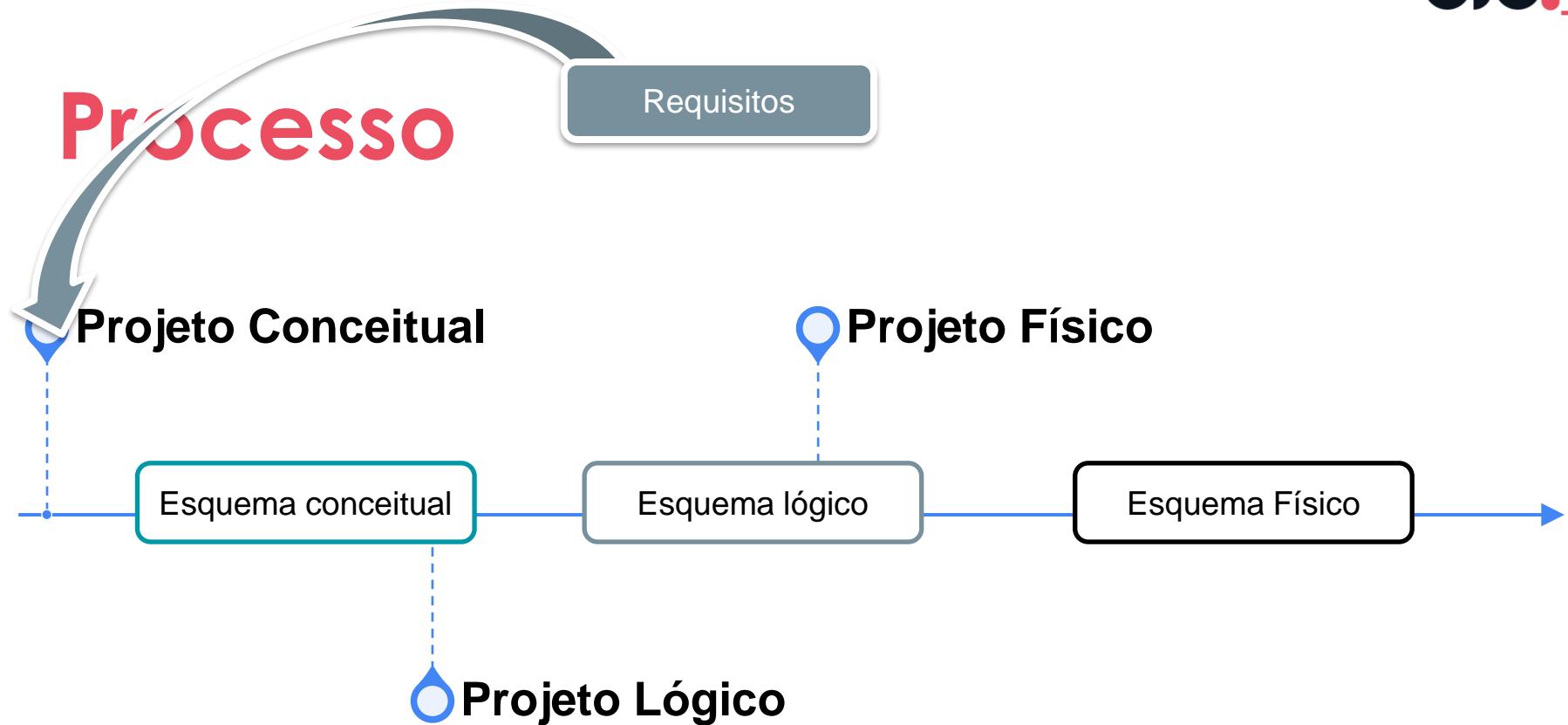
# Projeto Conceitual

# Projeto Conceitual



# Processo





# Implementação – Projeto Lógico & Físico



# Projeto Lógico

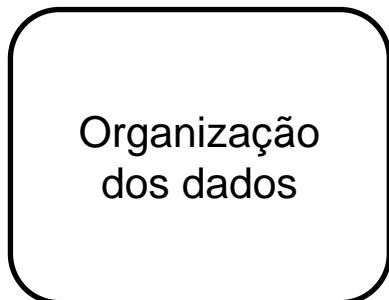
Descrição do modelo conceitual



Depende do modelo de Banco de Dados

# Projeto Lógico

Descrição do modelo conceitual



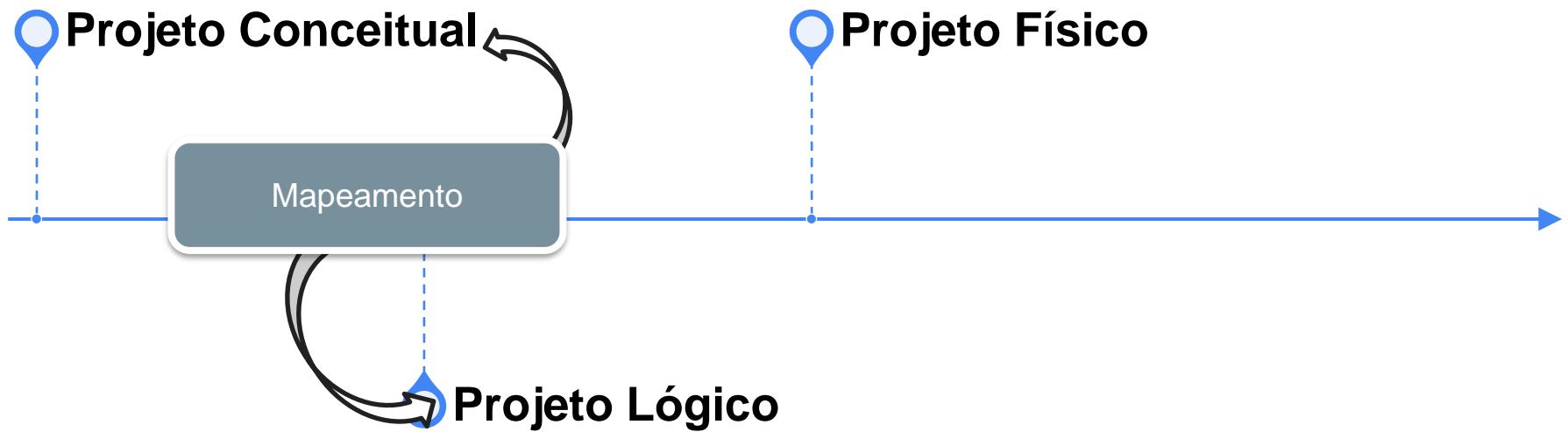
Tabelas



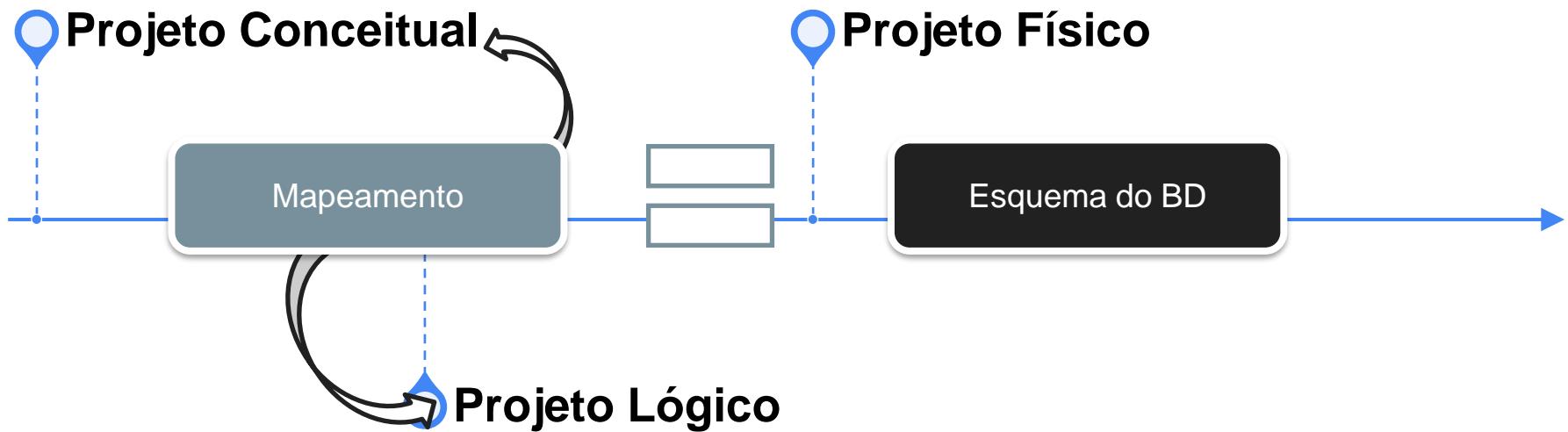
Estrutura

Depende do modelo de Banco de Dados

# Projeto Lógico



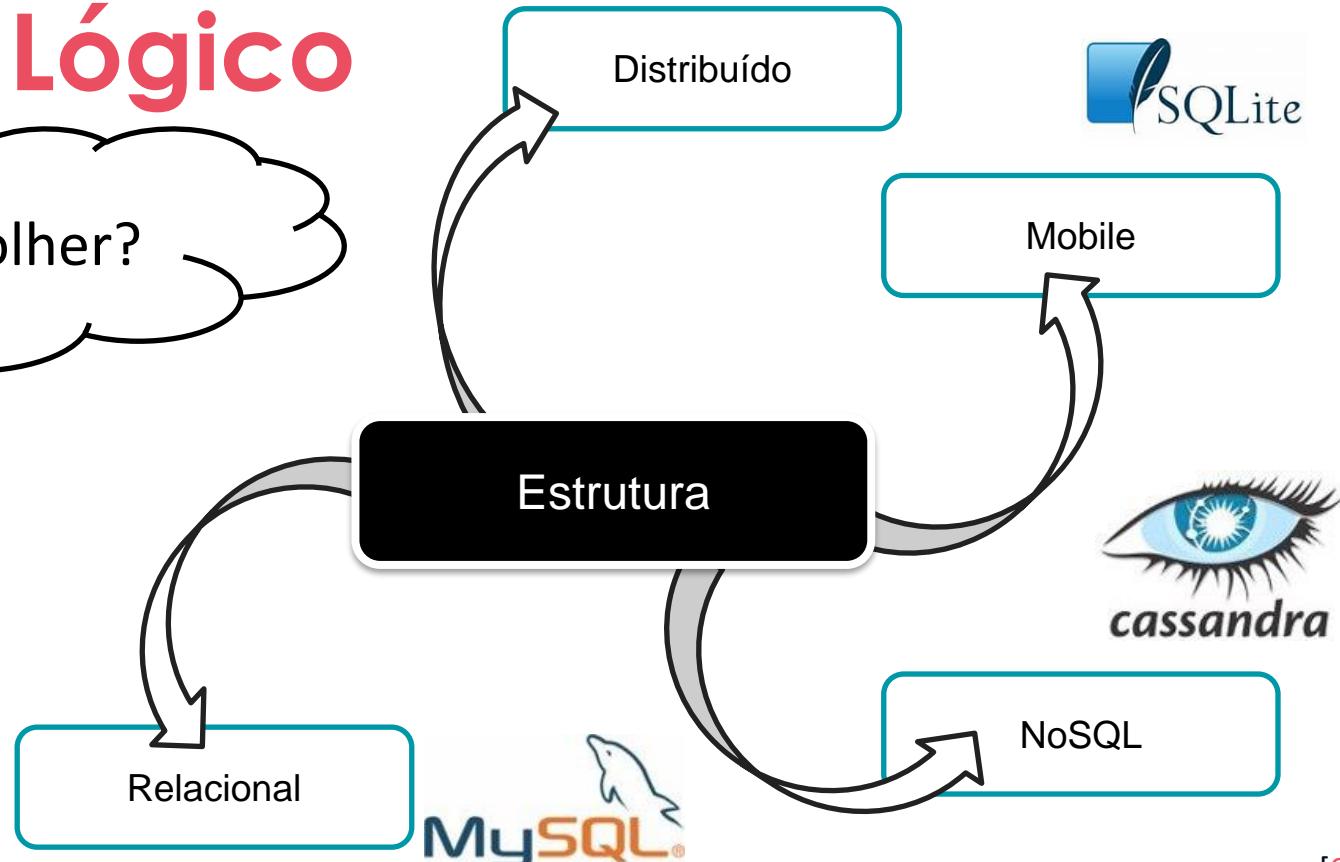
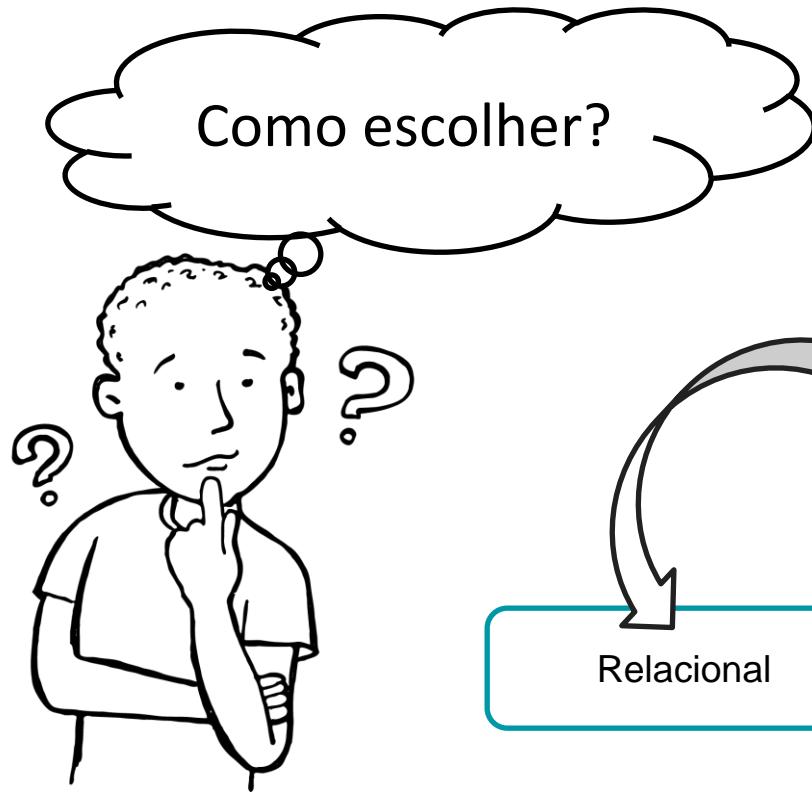
# Projeto Lógico



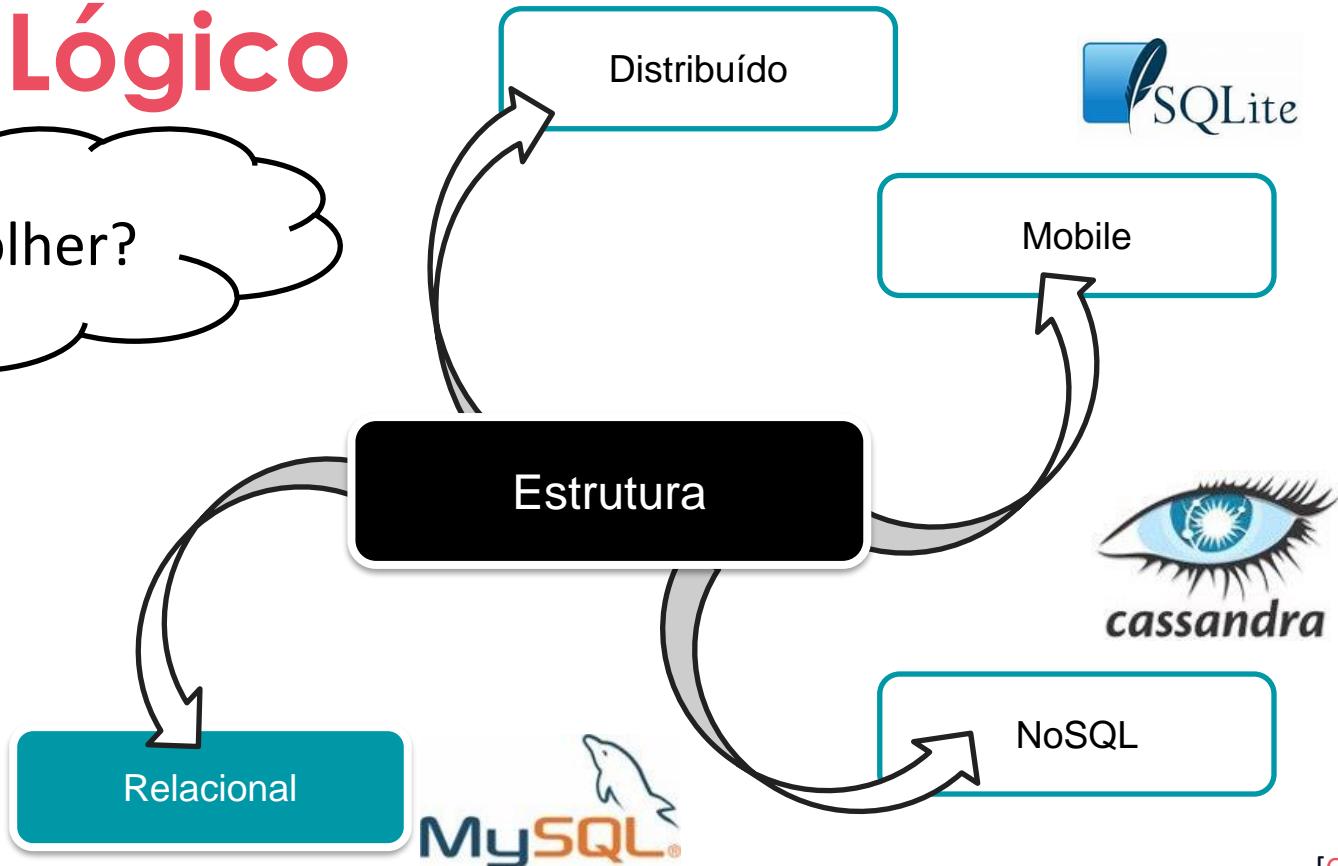
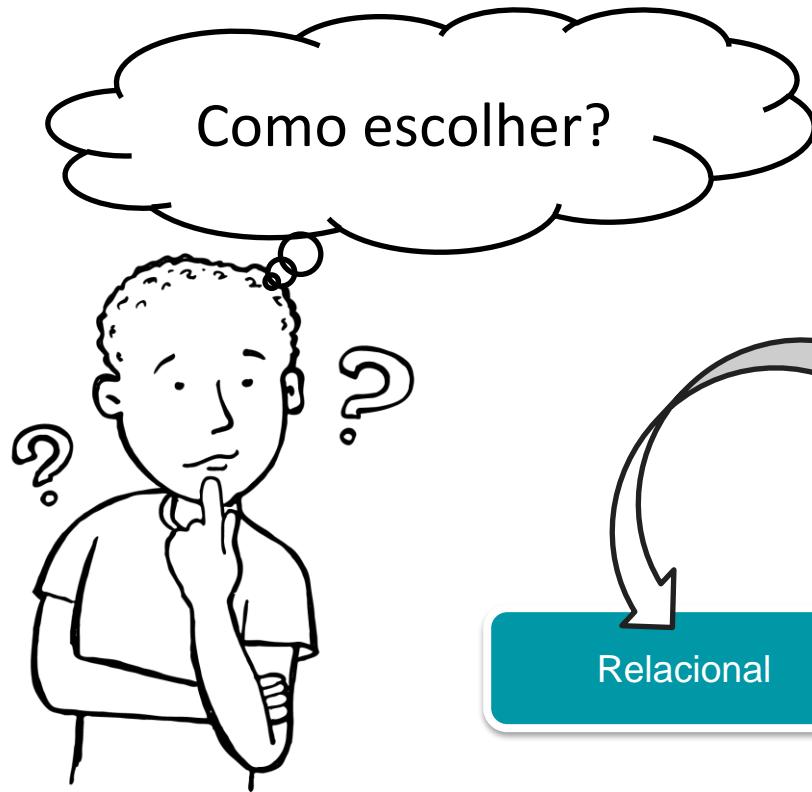
# Projeto Lógico



# Projeto Lógico



# Projeto Lógico



# Projeto Lógico

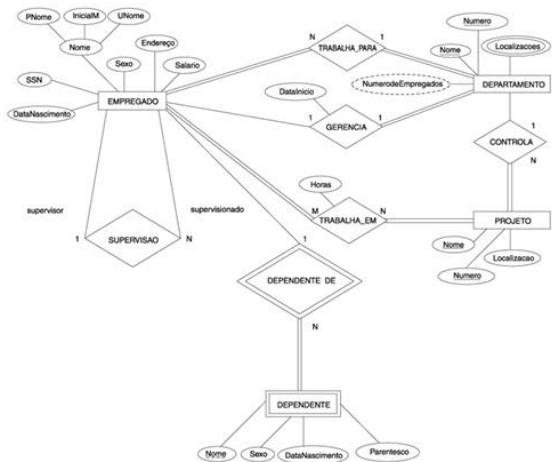


Criação do Esquema Lógico

Instalação e Configuração do SGBD

Criação do Esquema do BD

# Projeto Lógico



Influência

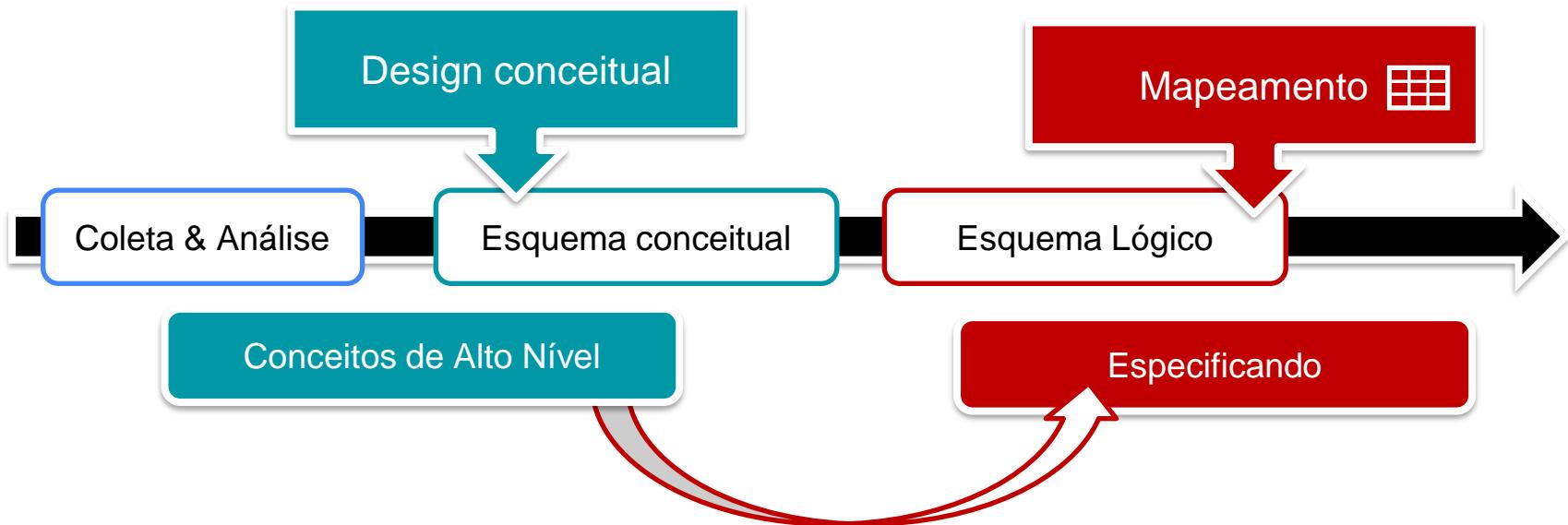


Características do  
modelo de SGBD

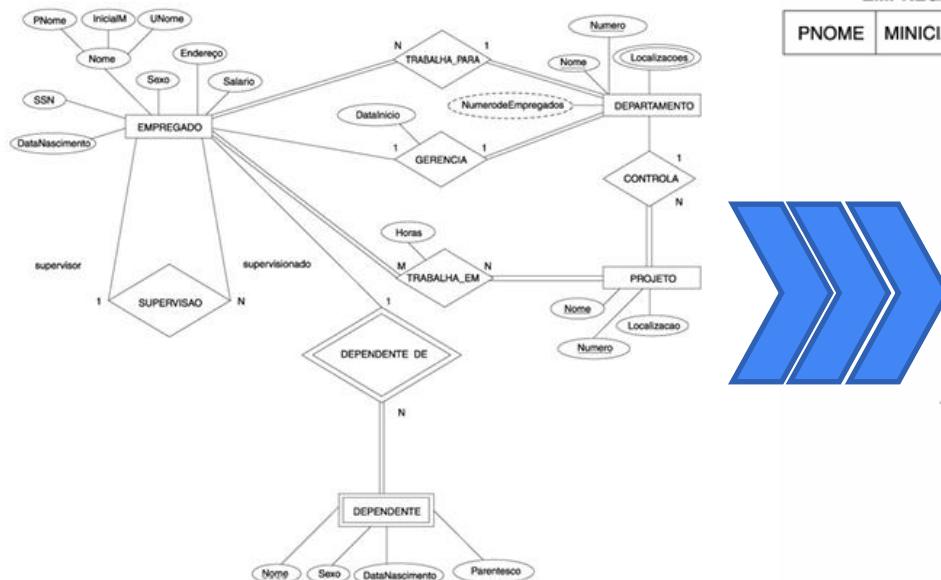


Depende do modelo de Banco de Dados

# Projeto Lógico



# Projeto Lógico



<b>EMPREGADO</b>									
PNAME	MINICIAL	UNOME	SSN	DATANASC	ENDERECO	SEXO	SALARIO	SUPERSSN	DNO

<b>DEPARTAMENTO</b>			
DNAME	DNUMBER	GERSSN	GERDATAINICIO

<b>DEPTO_LOCALIZACOES</b>	
DNUMBER	DLOCALIZACAO

<b>PROJETO</b>			
PJNAME	PNUMBER	PLOCALIZACAO	DNUM

<b>TRABALHA_EM</b>		
ESSN	PNO	HORAS

<b>DEPENDENTE</b>				
ESSN	NOME_DEPENDENTE	SEXO	DATANASC	PARENTESCO

Fonte: livro de referência - Navathe

Mapeamento ER para Modelo Relacional

**EMPREGADO**

PNAME	MINICIAL	UNOME	<u>SSN</u>	DATANASC	ENDERECO	SEXO	SALARIO	SUPERSSN	DNO
-------	----------	-------	------------	----------	----------	------	---------	----------	-----

**DEPARTAMENTO**

<u>DNAME</u>	DNUMBER	<u>GERSSN</u>	GERDATAINICIO
--------------	---------	---------------	---------------

**DEPTO\_LOCALIZACOES**

DNUMBER	DLOCALIZACAO
---------	--------------

**PROJETO**

<u>PJNAME</u>	PNUMBER	PLOCALIZACAO	DNUM
---------------	---------	--------------	------

**TRABALHA\_EM**

<u>ESSN</u>	PNO	HORAS
-------------	-----	-------

**DEPENDENTE**

<u>ESSN</u>	NOME_DEPENDENTE	SEXO	DATANASC	PARENTESCO
-------------	-----------------	------	----------	------------

# Projeto Lógico

**EMPREGADO**

PNAME	MINICIAL	UNOME	SSN	DATANASC	ENDERECO	SEXO	SALARIO	SUPERSSN	DNO
-------	----------	-------	-----	----------	----------	------	---------	----------	-----

**DEPARTAMENTO**

DNAME	DNUMERO	GERSSN	GERDATAINICIO
-------	---------	--------	---------------

**DEPTO\_LOCALIZACOES**

DNUMERO	DLOCALIZACAO
---------	--------------

**PROJETO**

PJNAME	PNUMERO	PLOCALIZACAO	DNUM
--------	---------	--------------	------

**TRABALHA\_EM**

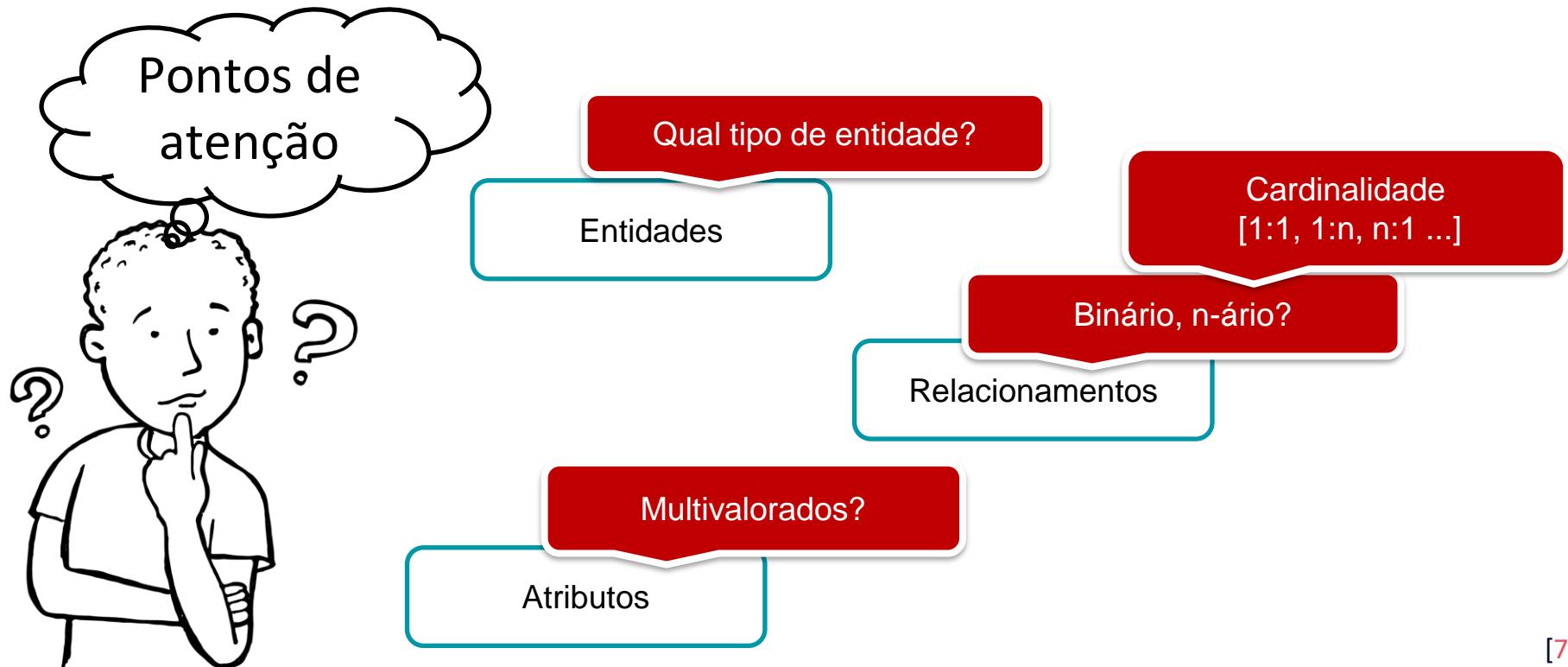
ESSN	PNO	HORAS
------	-----	-------

**DEPENDENTE**

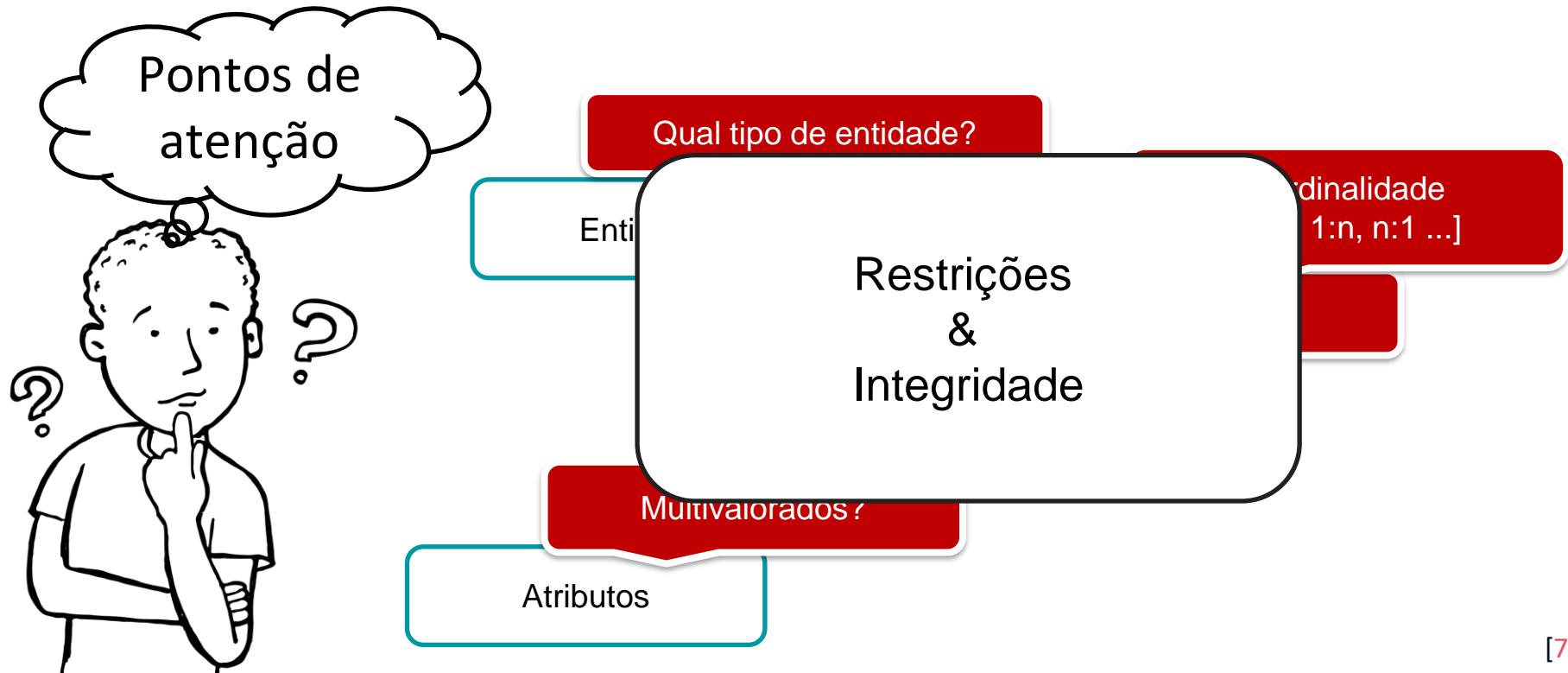
ESSN	NOME_DEPENDENTE	SEXO	DATANASC	PARENTESCO
------	-----------------	------	----------	------------

# Projeto Lógico

# Mapeamento ER x Relacional



# Mapeamento ER x Relacional



# Projeto Físico

Descrição do modelo conceitual

Como será implementado?

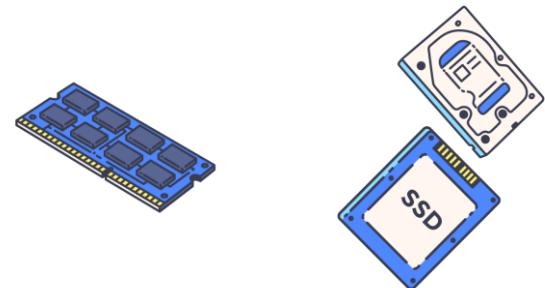
Independente do modelo de Banco de Dados

# Projeto Físico

Como será implementado?

## Parâmetros Físicos

- Estrutura e Índices
- Organização e Caminhos de Arquivos
- Segurança, performance ...



Diretamente ligado ao SGBD

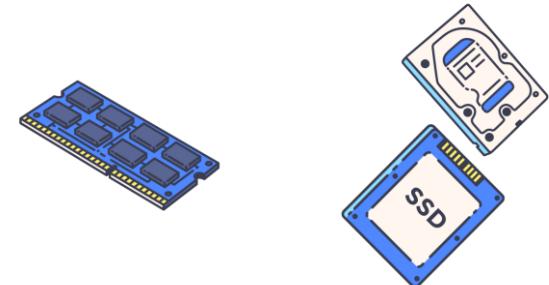
# Projeto Físico

## Parâmetros Físicos

- Estrutura e Índices
- Organização e Caminhos de Arquivos
- Segurança, performance ...

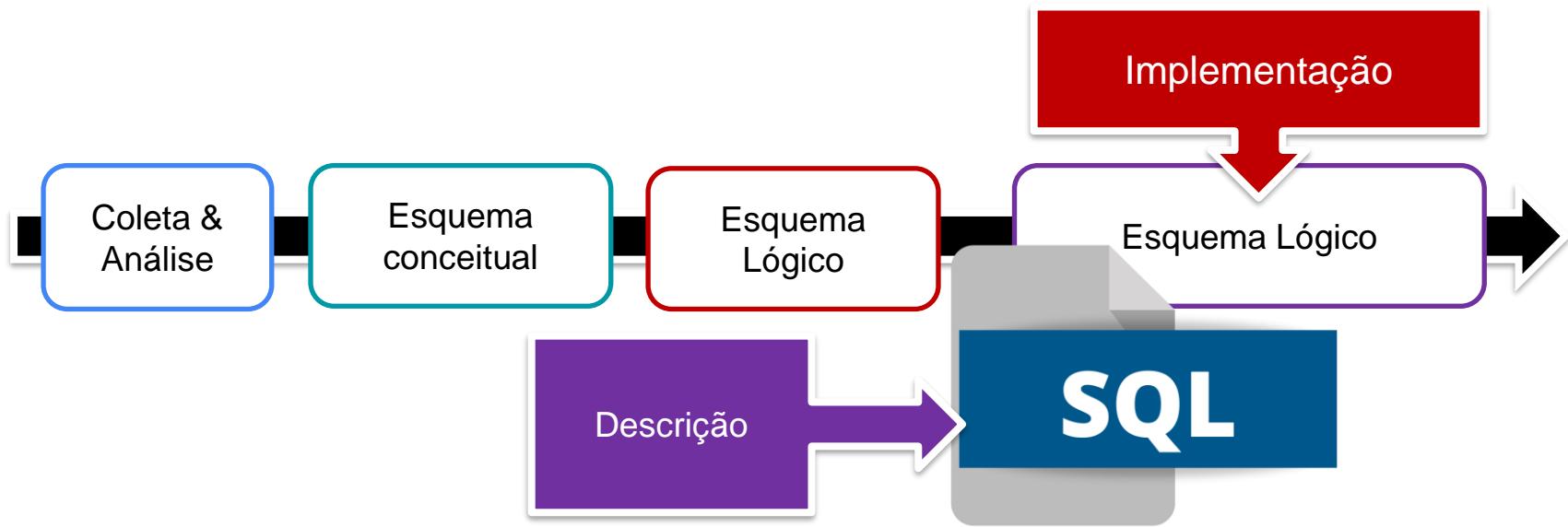
Alocação de memória

Particionamento de tabelas

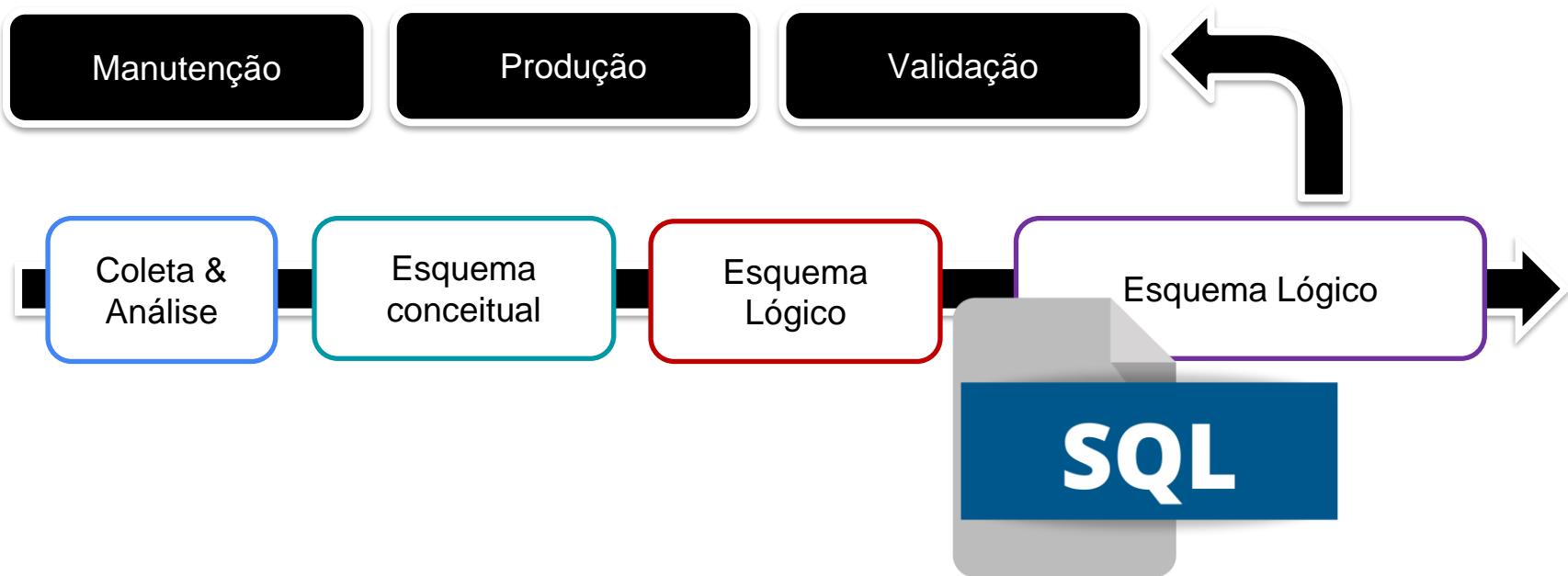


Diretamente ligado ao SGBD

# Projeto Lógico



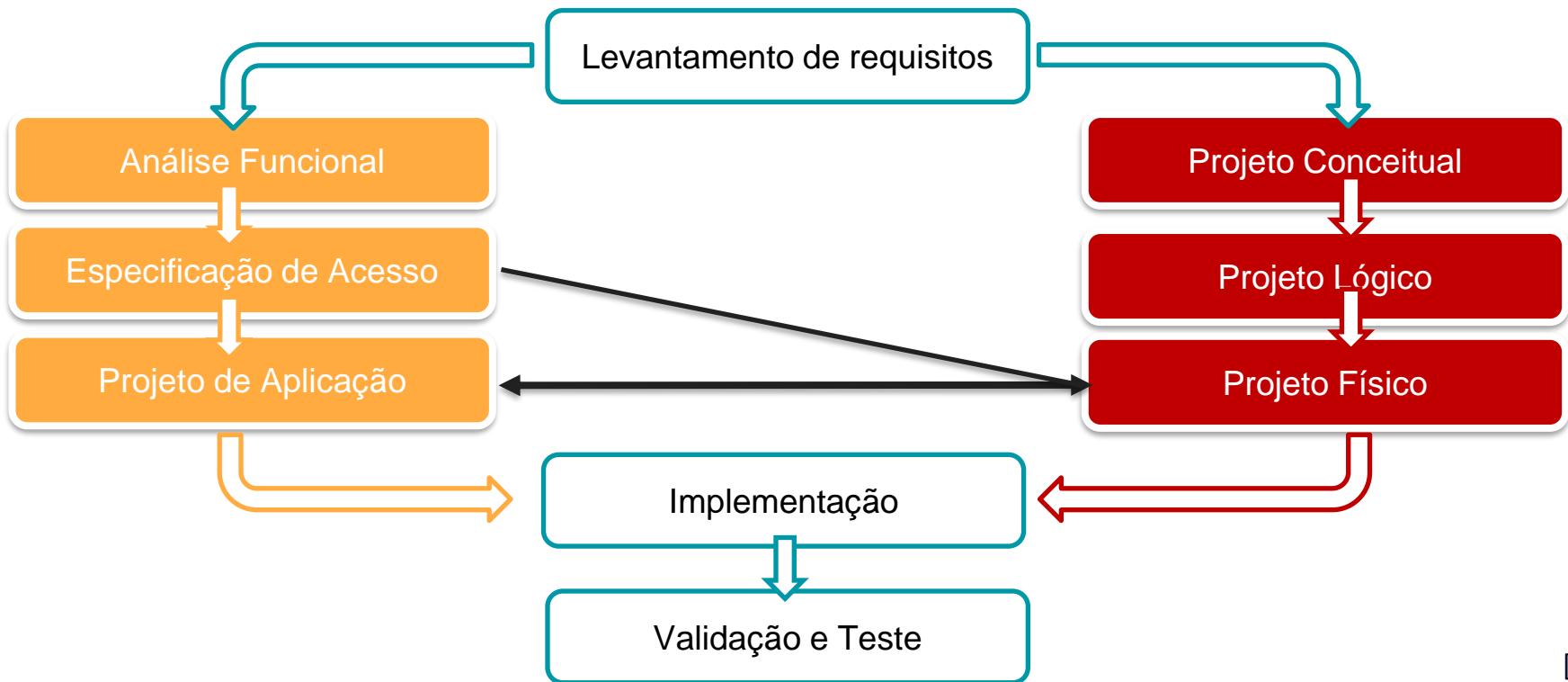
# Projeto Lógico



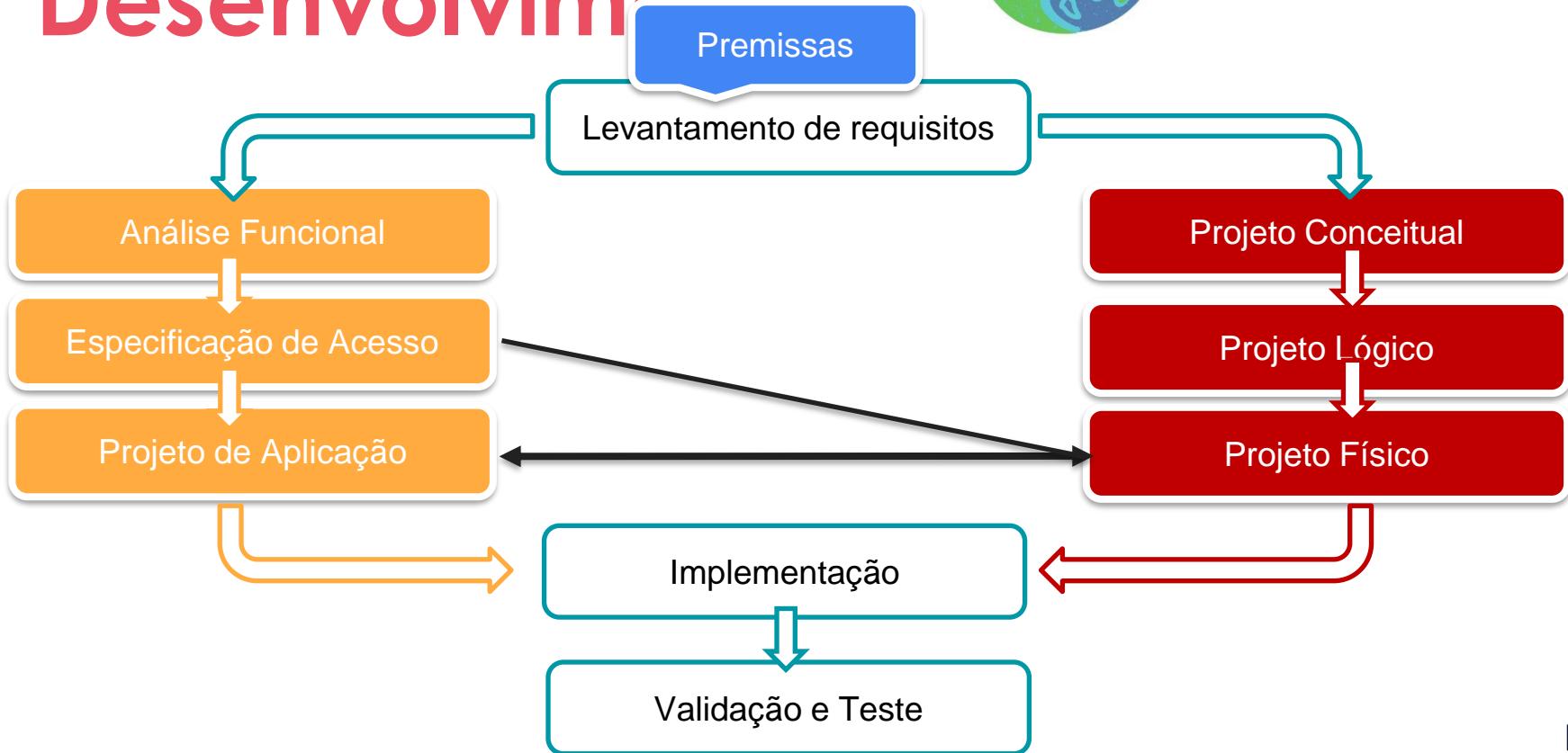
# Fases de desenvolvimento de BDs e Aplicações



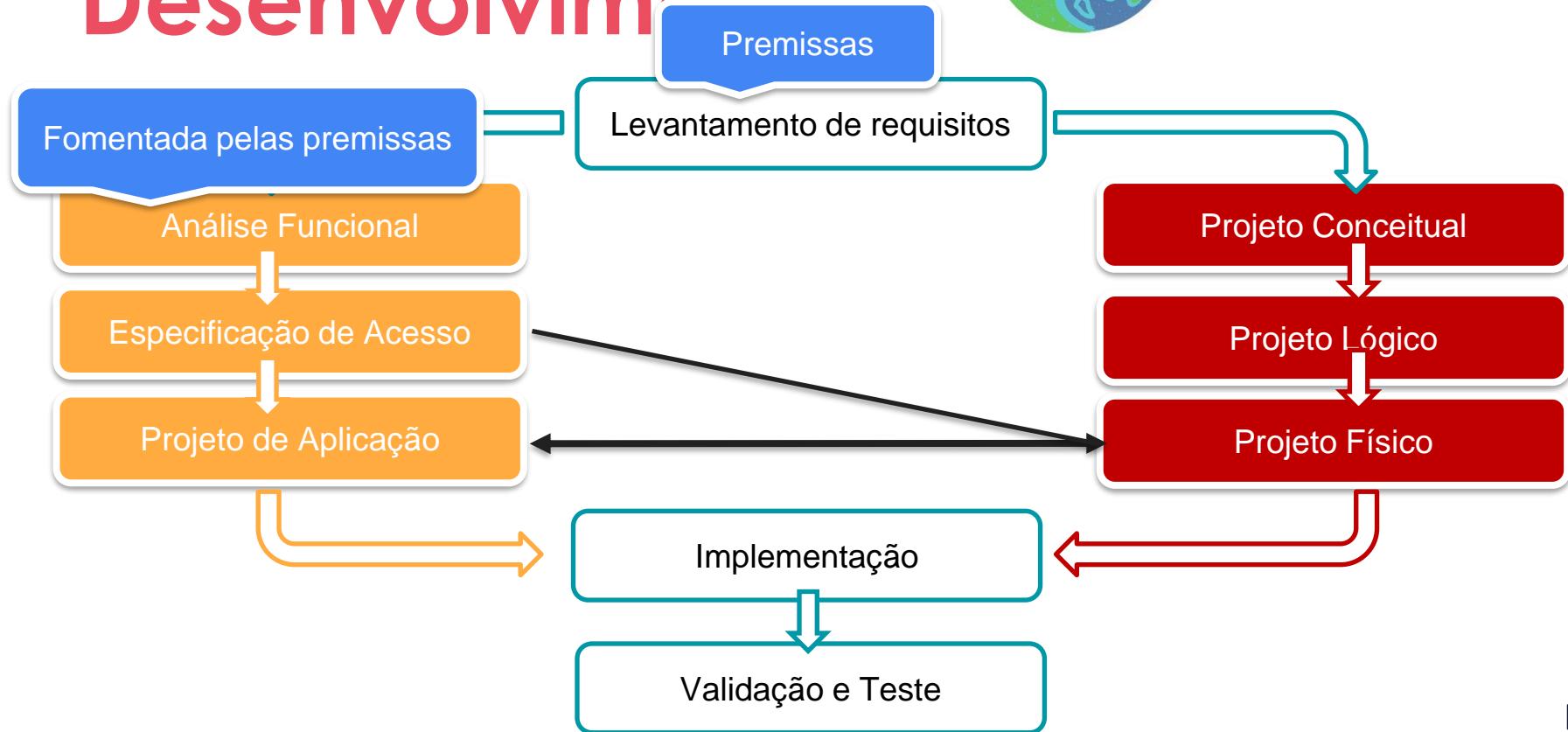
# Desenvolvimento



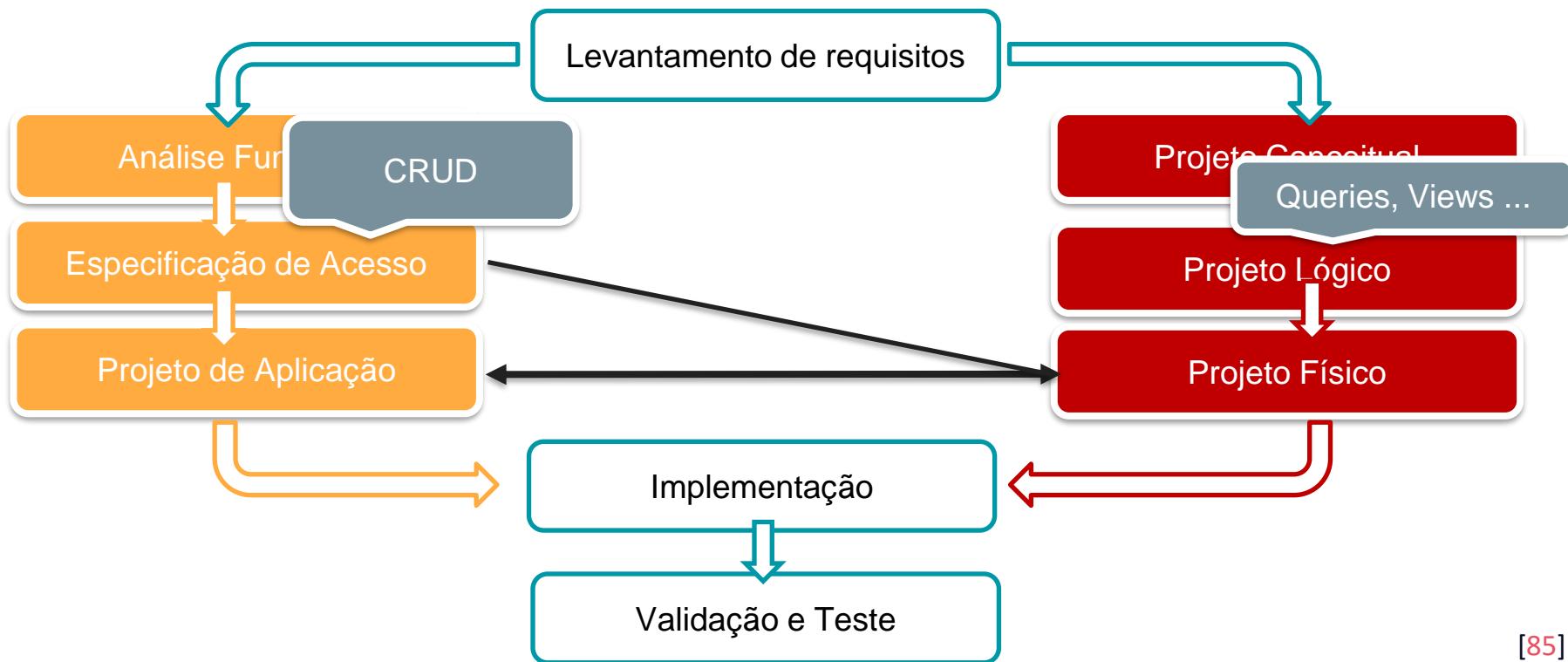
# Desenvolvimento



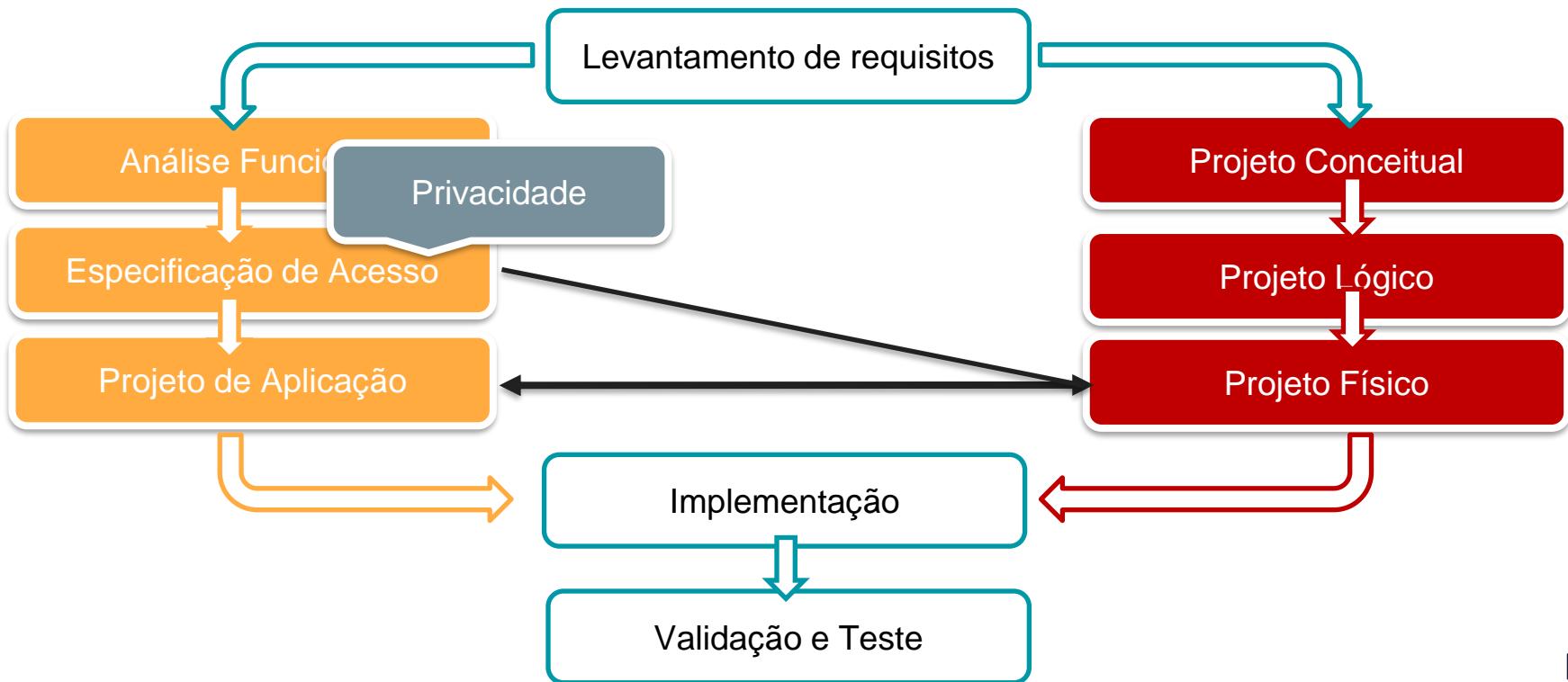
# Desenvolvimento



# Desenvolvimento

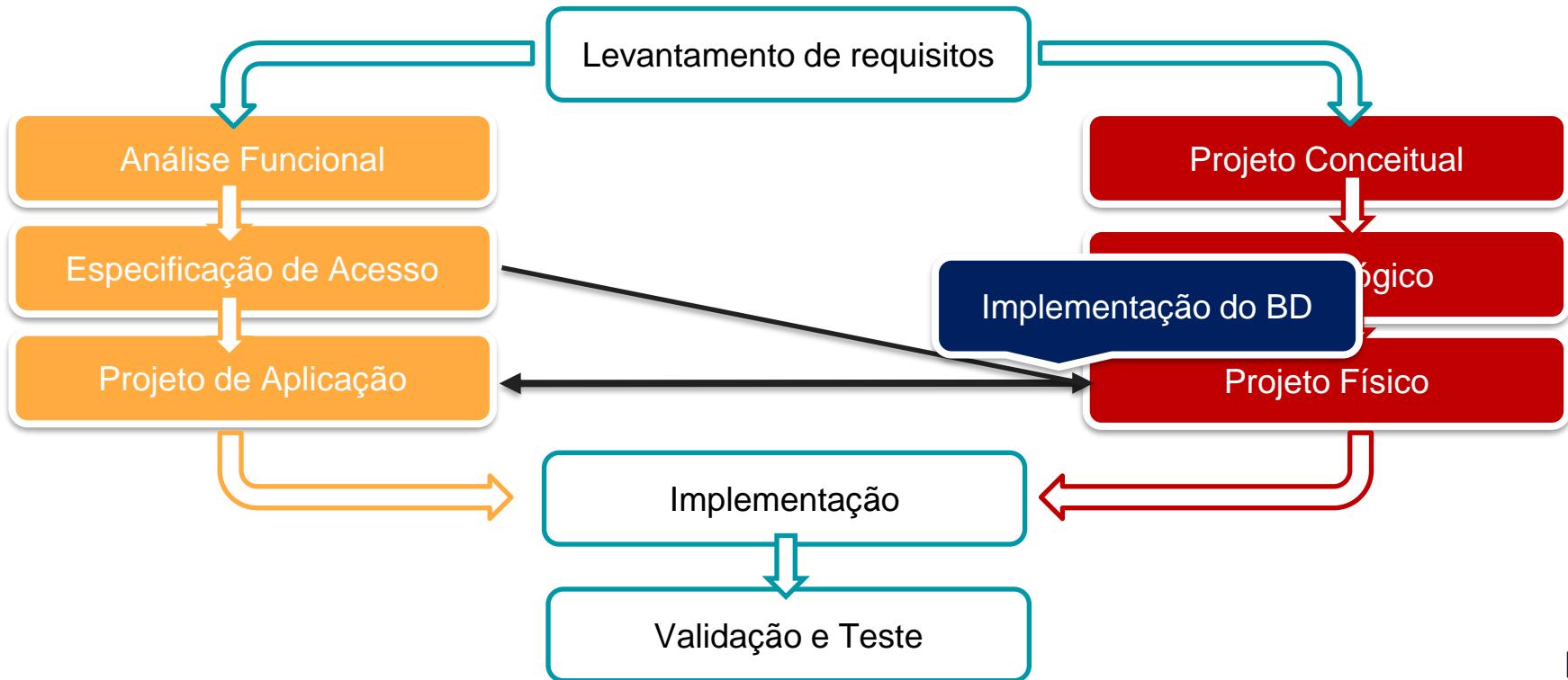


# Desenvolvimento

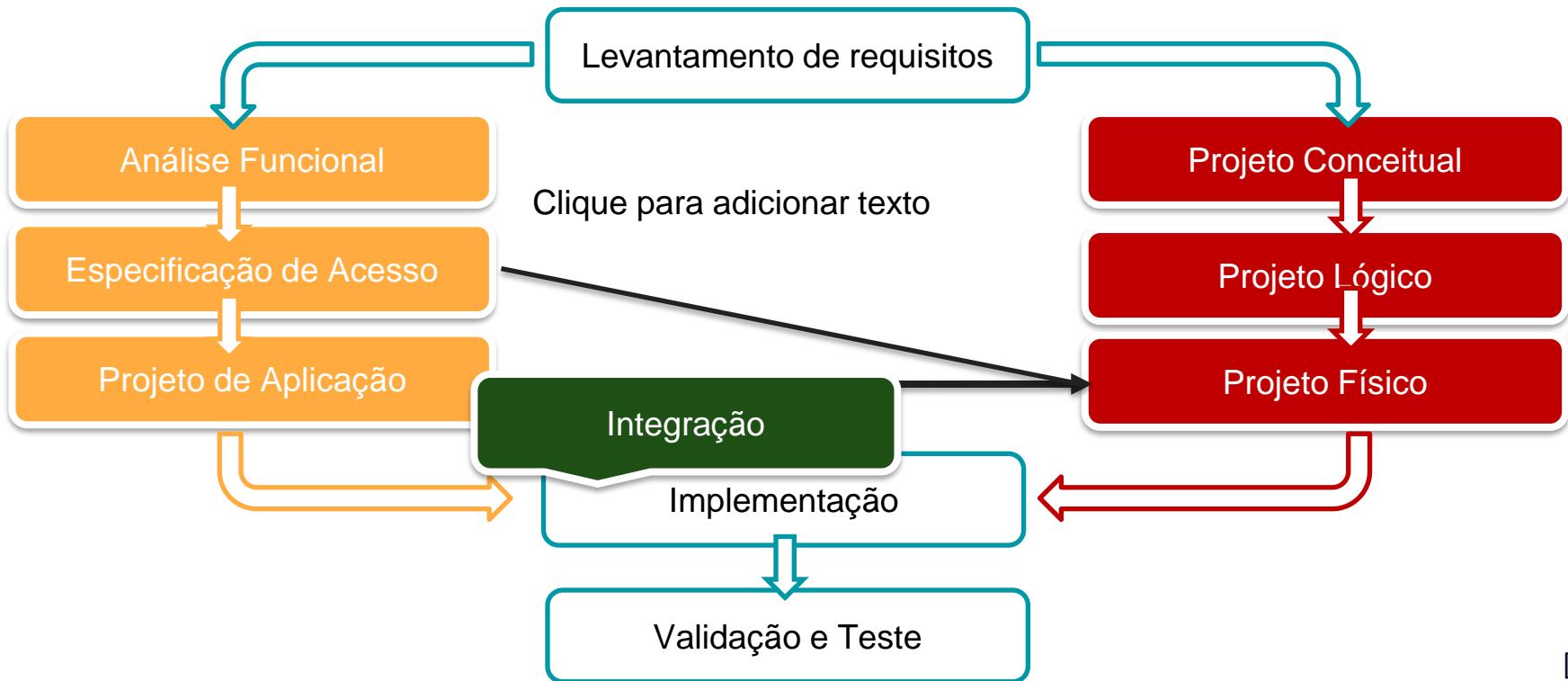


# Desenvolvimento

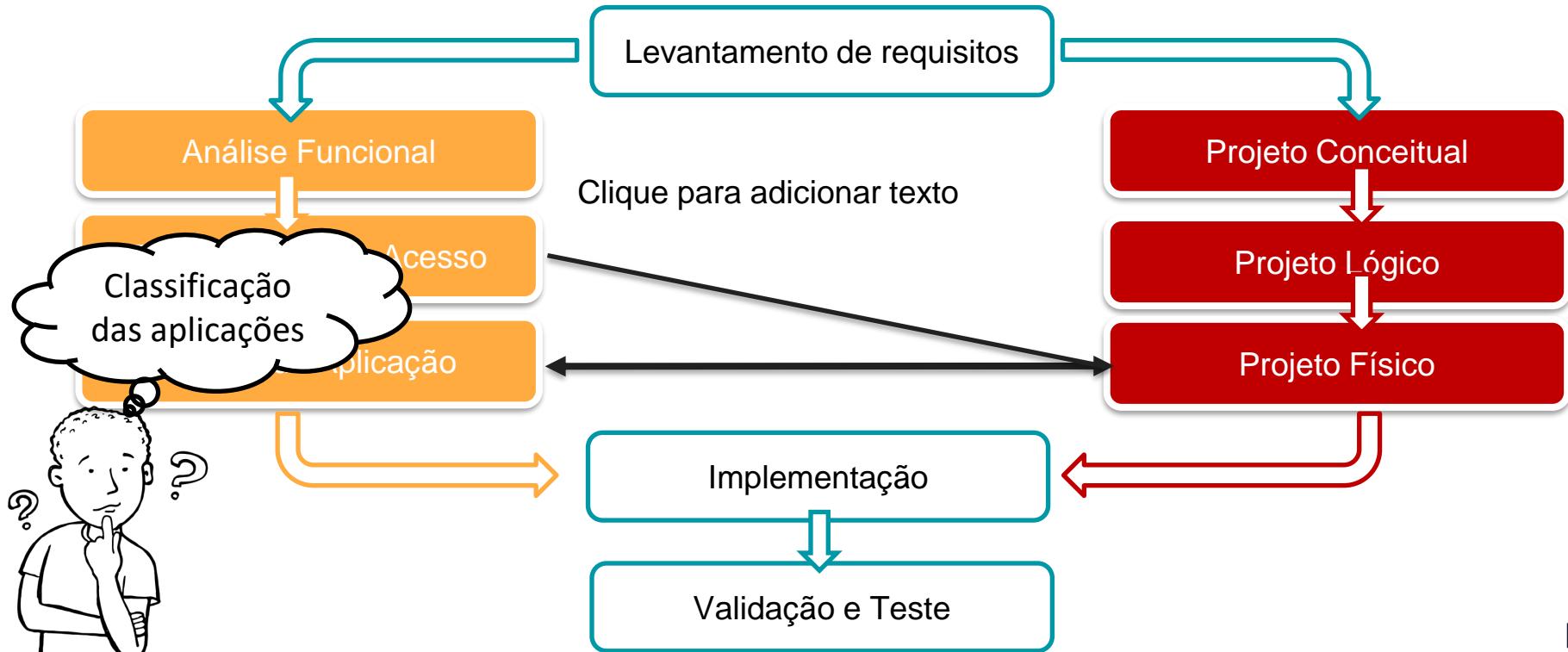
Conhece a aplicação



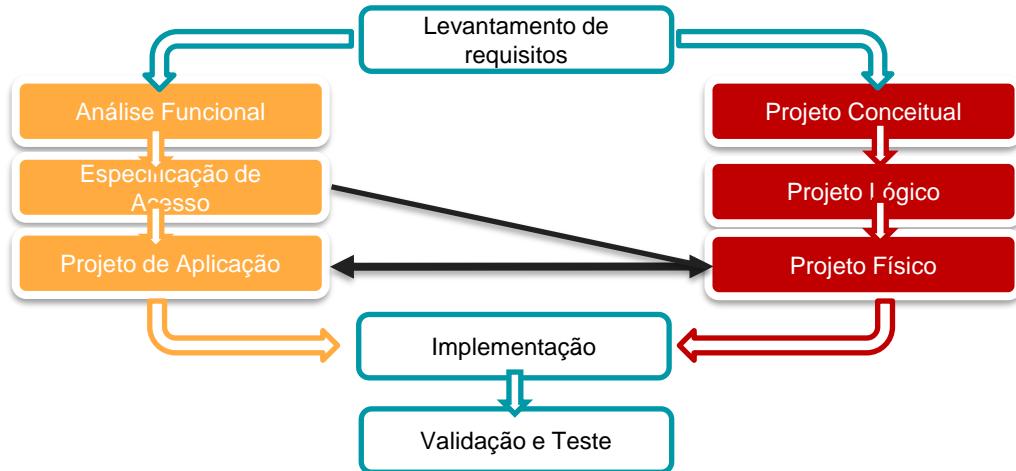
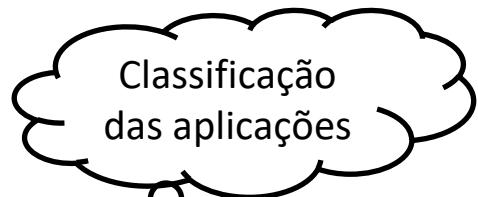
# Desenvolvimento



# Desenvolvimento



# Aplicações

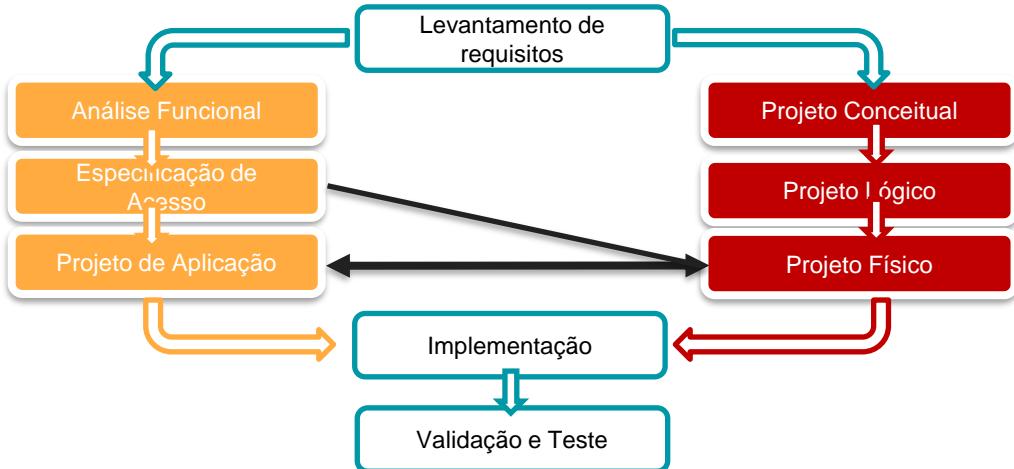


Uso dos dados

OLTP

OLAP

# Aplicações



# Classificação das aplicações



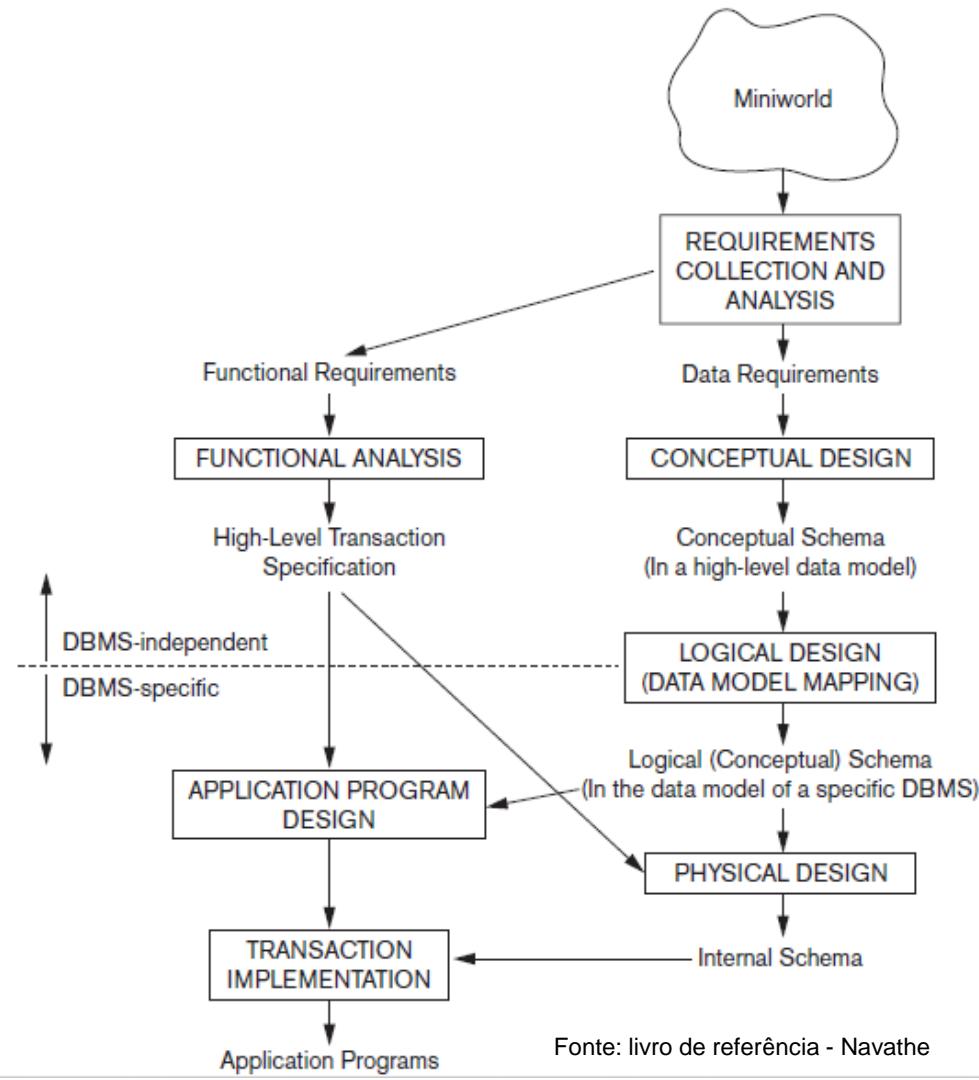
HTAP

## Uso dos dados

OLTP

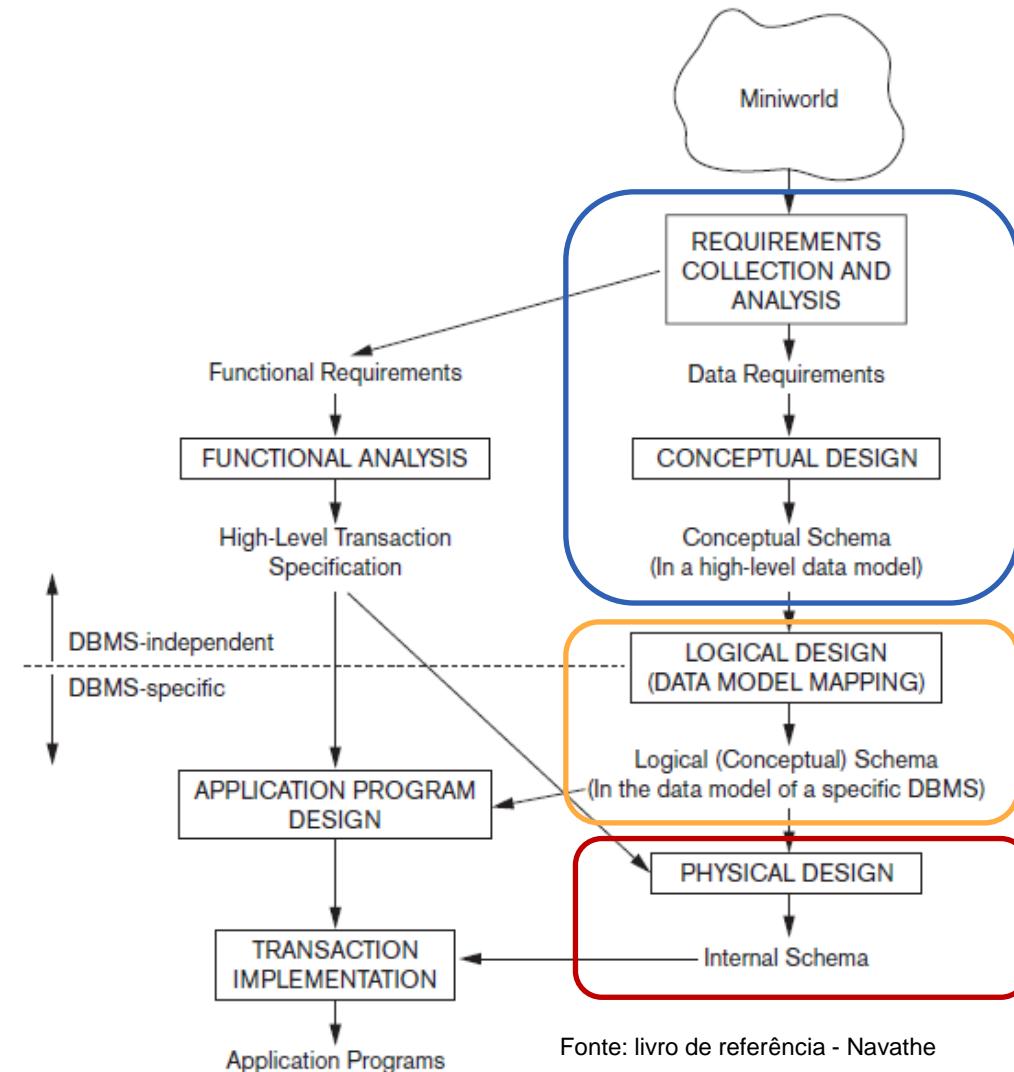
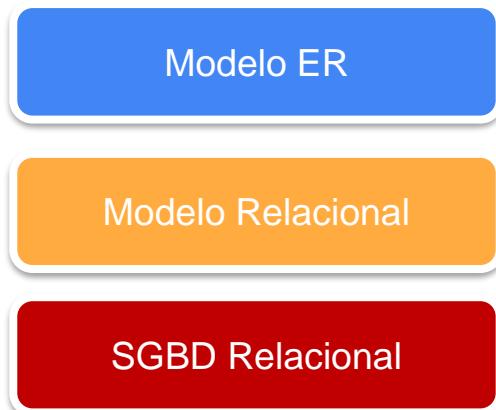
OLAP

# Resumindo...



Fonte: livro de referência - Navathe

# Resumindo...



# Aplicação - Modelagem dos dados



# Exemplo

Company

Objetivo: rastreamento

Empregados

Projetos

Departamentos

# Exemplo

Objetivo: rastreamento



Company

Empregados

Projetos

Departamentos

Levantamento de  
dados e requisitos

# Exemplo

Organização:

Departamentos

- Nome, nº, gerente (empregado)

Rastrear?

- Data de início do gerente & locais

Company

Descrição do  
mini-mundo

# Exemplo

Organização:

Departamentos

- Nome, nº, gerente (empregado)

Rastrear?

- Data de início do gerente & locais

Company

Descrição do  
mini-mundo

# Exemplo

Qual a função?

Departamentos

- Projetos – 1 ou +

Características

- Nome, n°, localização

Company

Descrição do  
mini-mundo

# Exemplo

Qual a função?

- Projetos – 1 ou +

Características

- Nome, nº, localização

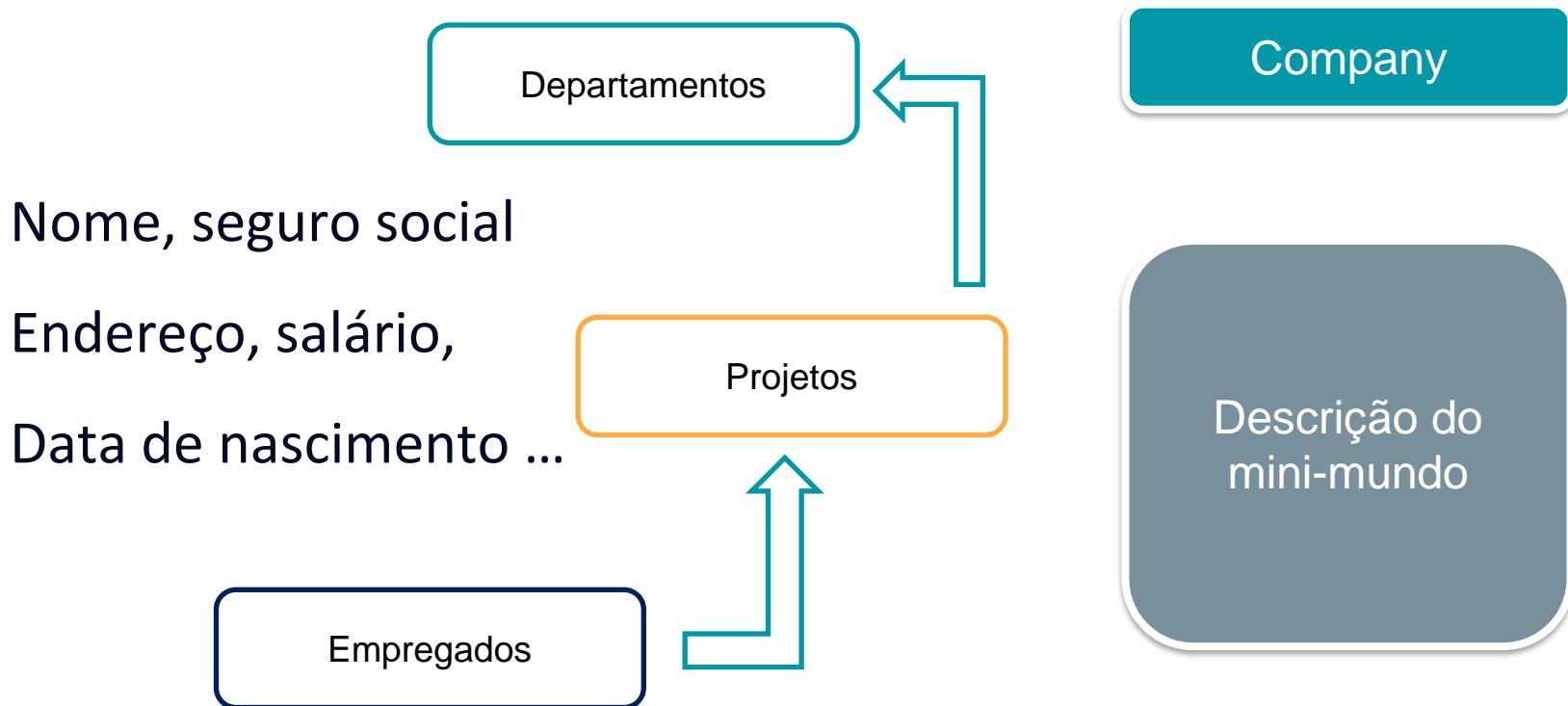
Departamentos

Projetos

Company

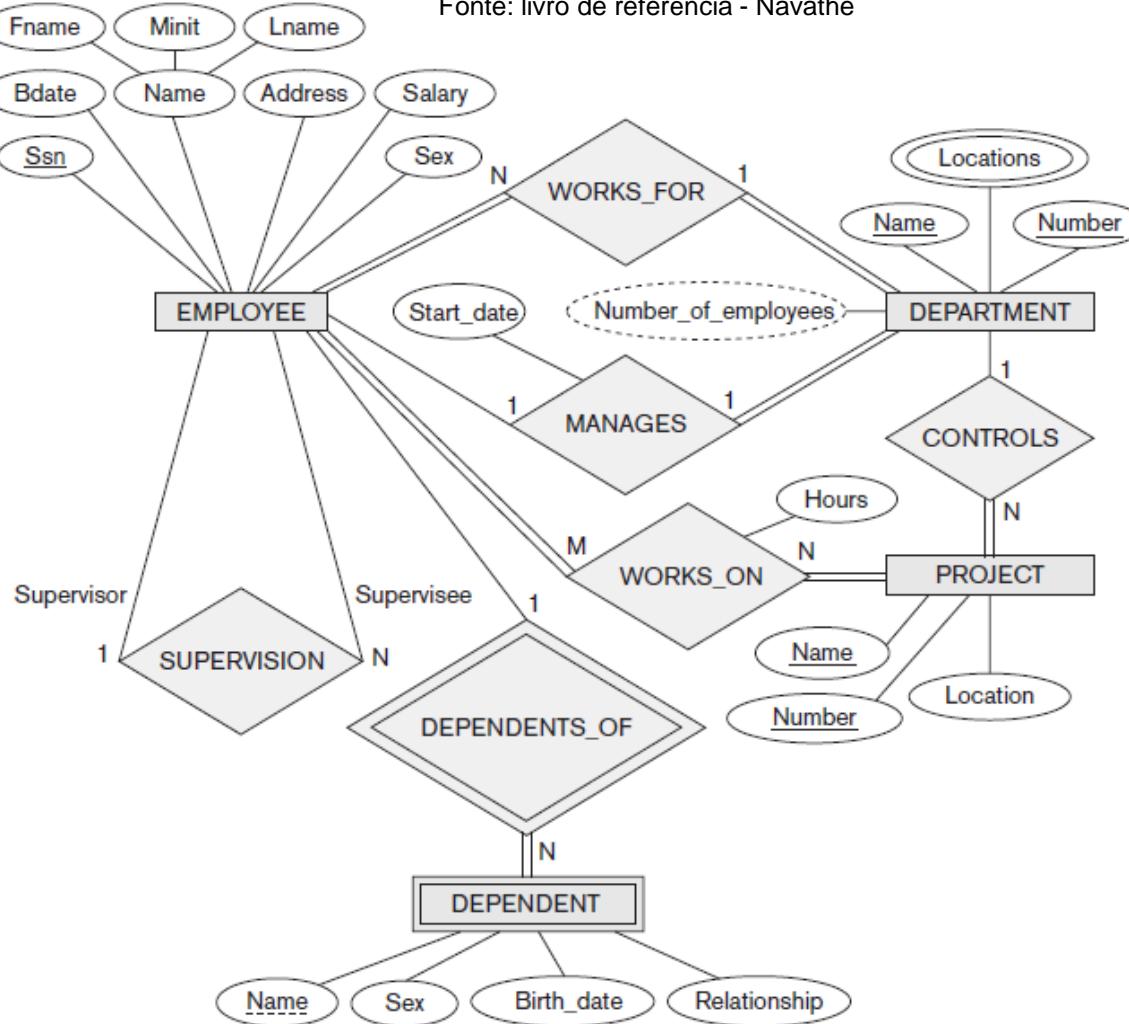
Descrição do  
mini-mundo

# Exemplo



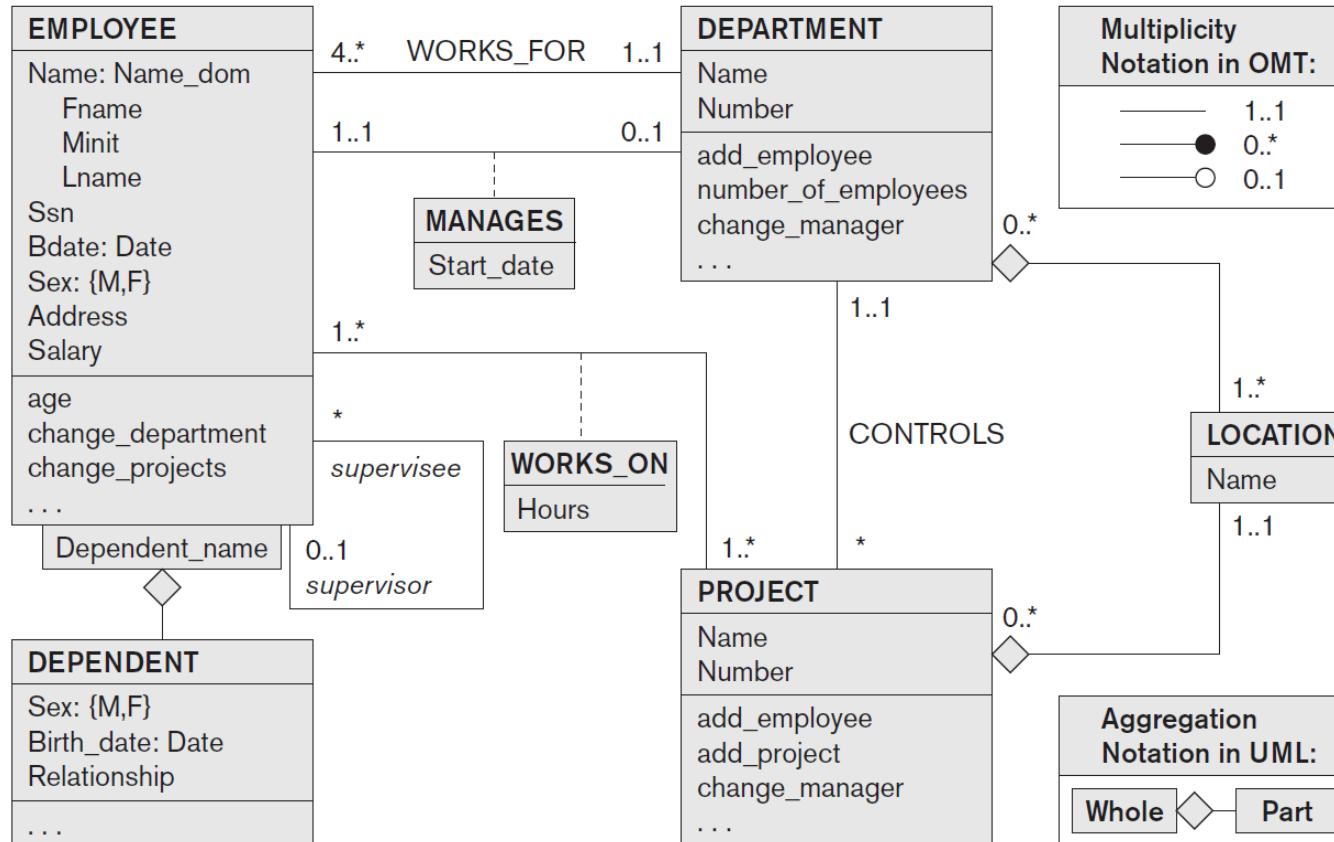
# Exemplo





DER

Company



Company

**EMPLOYEE**

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary	Super_ssn	Dno
-------	-------	-------	------------	-------	---------	-----	--------	-----------	-----

**DEPARTMENT**

Dname	<u>Dnumber</u>	Mgr_ssn	Mgr_start_date
-------	----------------	---------	----------------

**DEPT\_LOCATIONS**

<u>Dnumber</u>	<u>Dlocation</u>
----------------	------------------

**PROJECT**

Pname	<u>Pnumber</u>	Plocation	Dnum
-------	----------------	-----------	------

**WORKS\_ON**

<u>Essn</u>	<u>Pno</u>	Hours
-------------	------------	-------

**DEPENDENT**

<u>Essn</u>	Dependent_name	Sex	Bdate	Relationship
-------------	----------------	-----	-------	--------------

Company

**Figure 5.5**

Schema diagram for the COMPANY relational database schema.

# Company

## DEPARTMENT

Dname	<u>Dnumber</u>	Mgr_ssn	Mgr_start_date
Research	5	333445555	1988-05-22
Administration	4	987654321	1995-01-01
Headquarters	1	888665555	1981-06-19

## EMPLOYEE

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

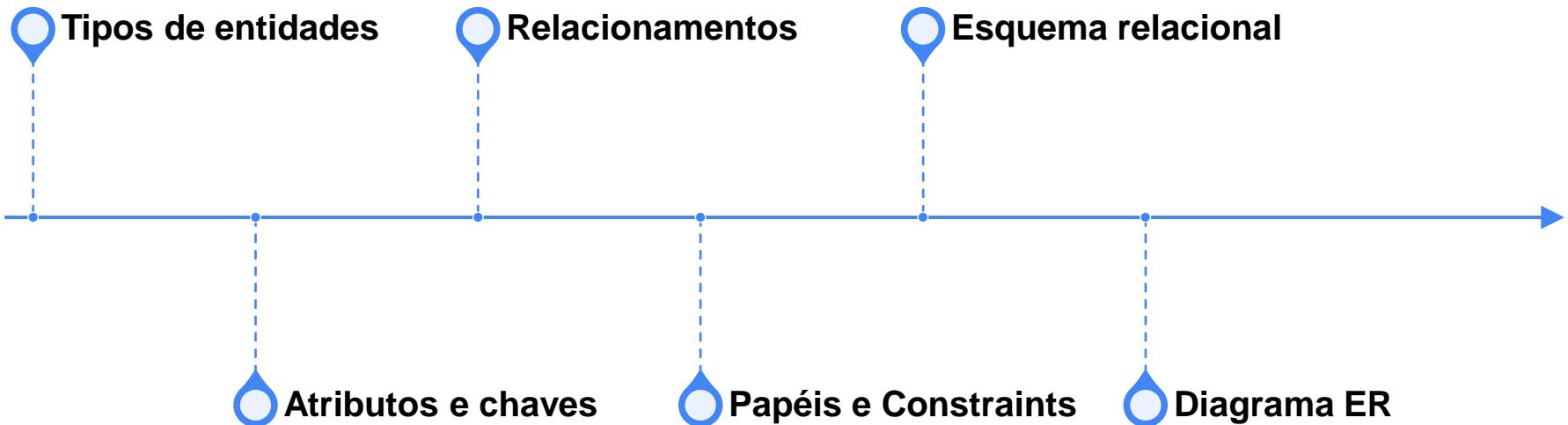


## Etapa 2

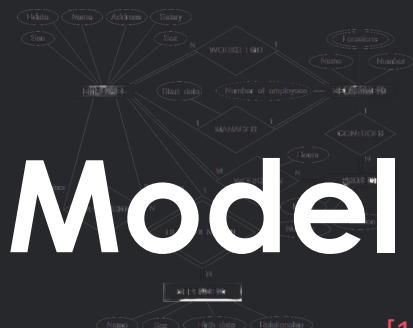
# Explorando o Modelo Entidade Relacionamento

//Modelagem de Banco de Dados com ER -

# Processo



# Tipos de Entidades, Chaves e Atributos

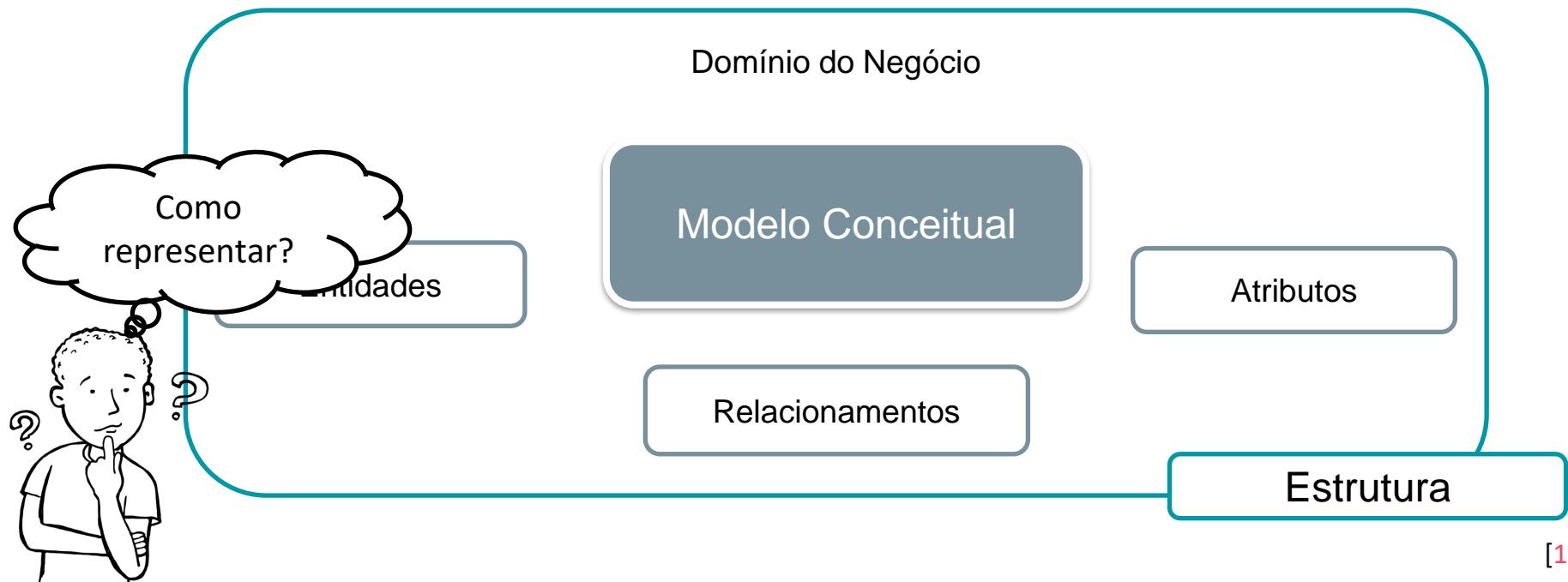


# ER Model

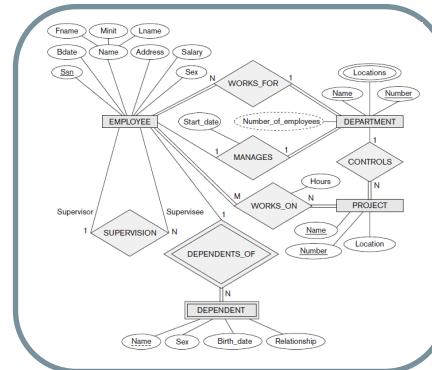
# Modelo ER



# Modelo ER



# Modelo ER



Domínio do Negócio

DER

Entidades

Modelo Conceitual

Atributos

Relacionamentos

Estrutura



# Diagrama ER

## Diagramas

- Raras modificações
- Facilidade de manipular
- Esquema ER do banco de dados

Foco no  
esquema

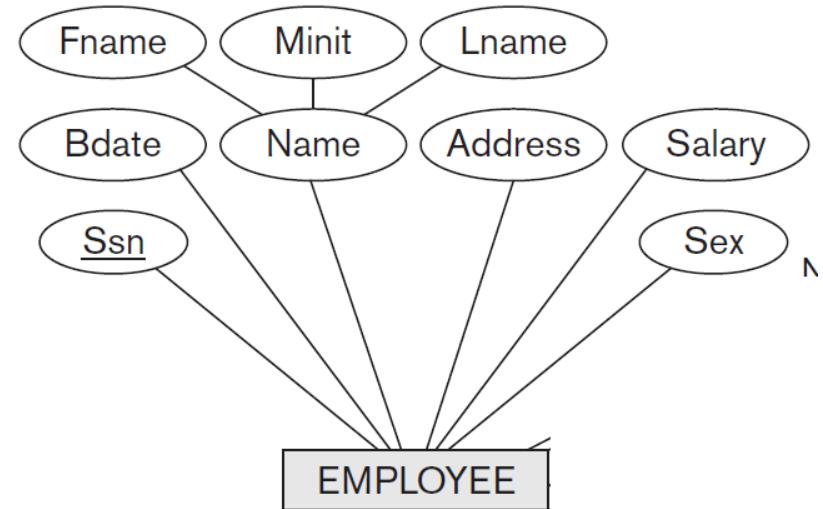
# Entidades

- Componente básico
- Existencia independiente
- Atributos



# Entidades

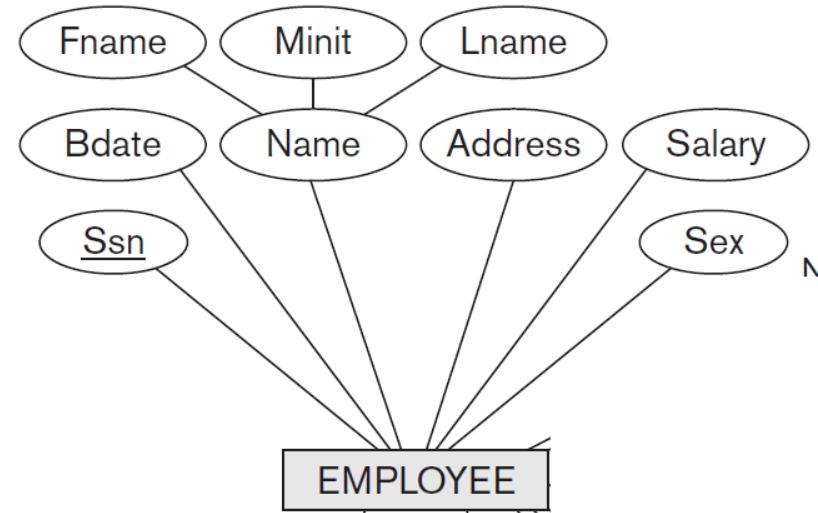
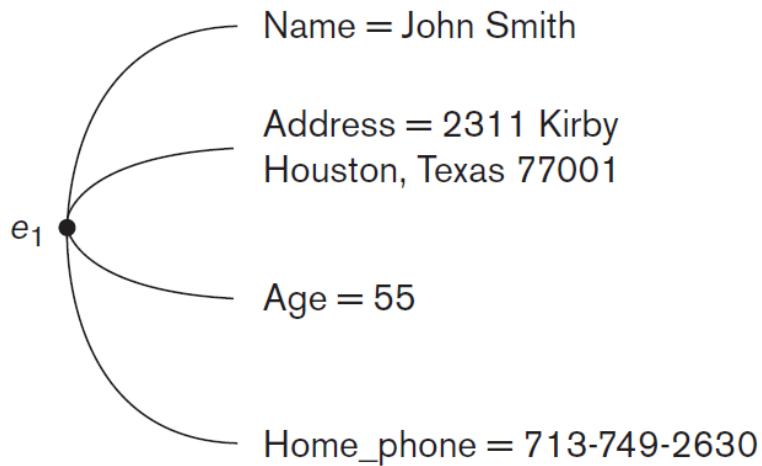
- Componente básico
- Existencia independiente
- Atributos



Entidades

Objetos

# Entidades



Entidades

Objetos

# Atributos

- Características/Descrição das entidade
- Atributos relacionados as intâncias

Atômicos x Composto

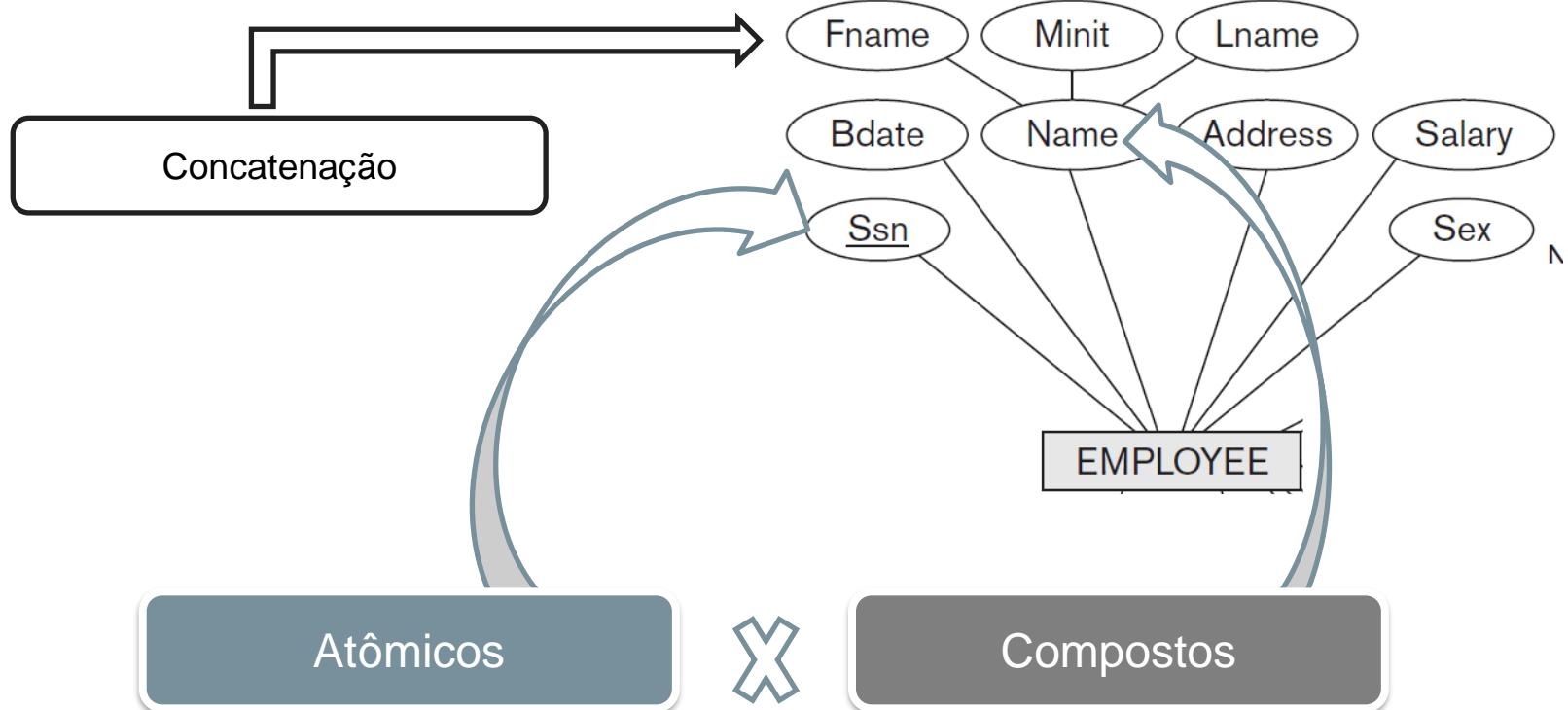
Atributos multivalorados

Armazenados e derivados

Atributos Nulos

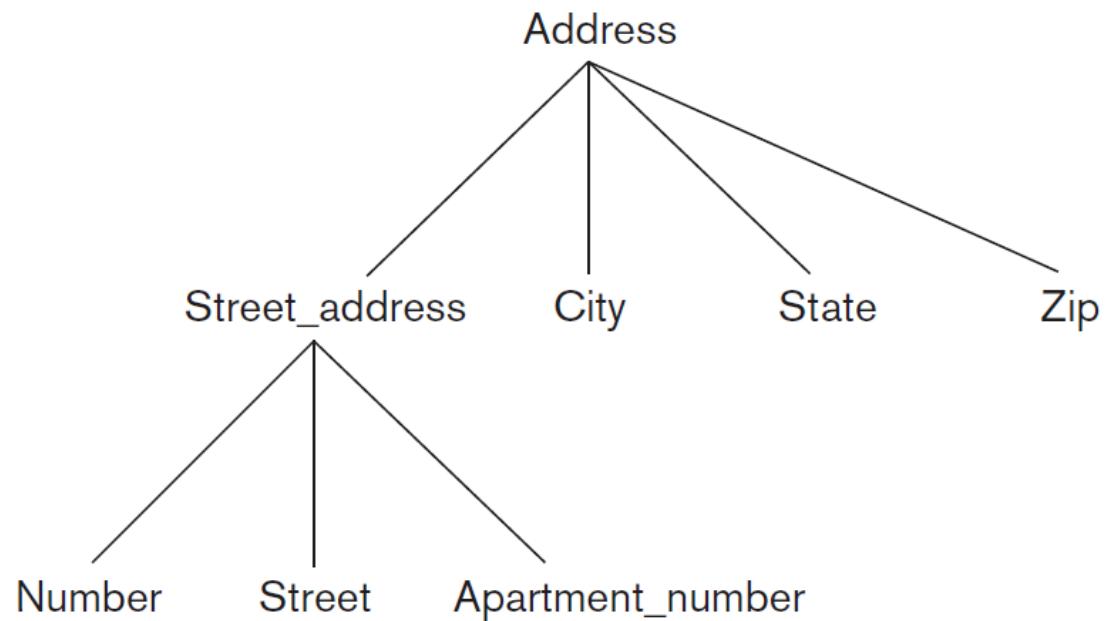
Atributos Complexos

# Atributos



# Atributo Composto

Av. Sete de Setembro  
371 – Araraquara, SP,  
148000-390

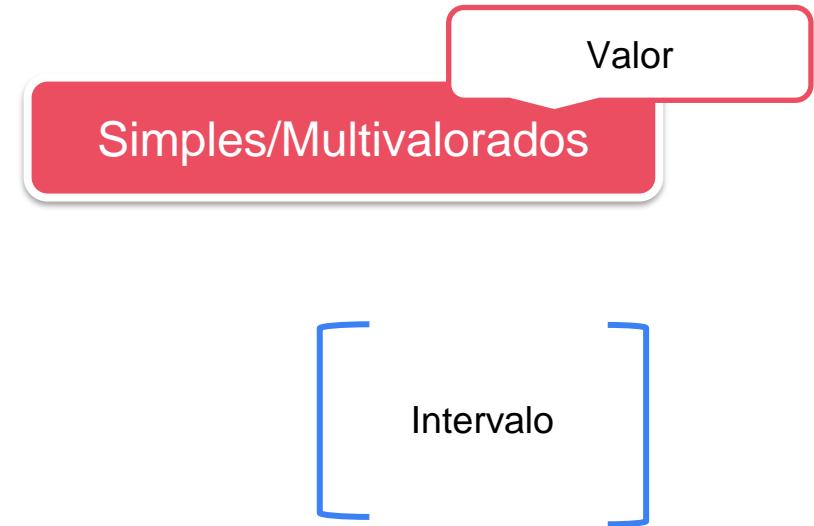
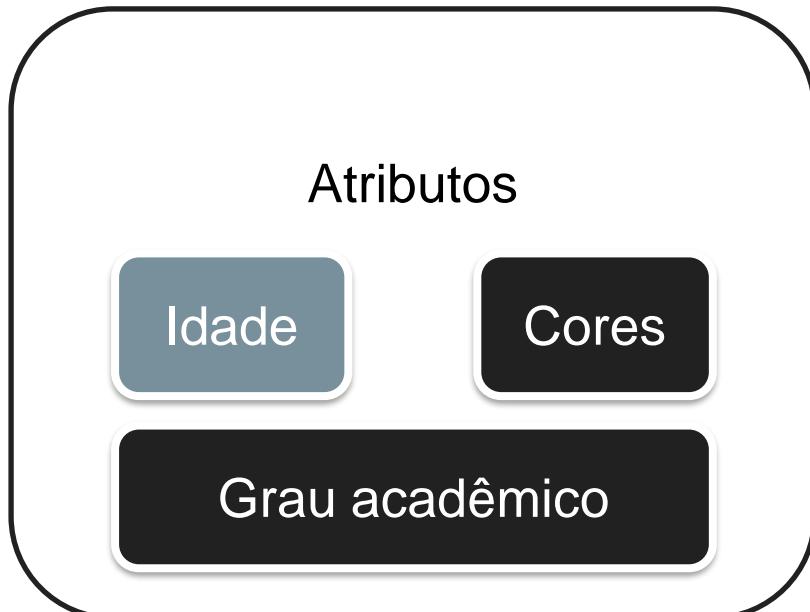


# Atributo Multivalorados

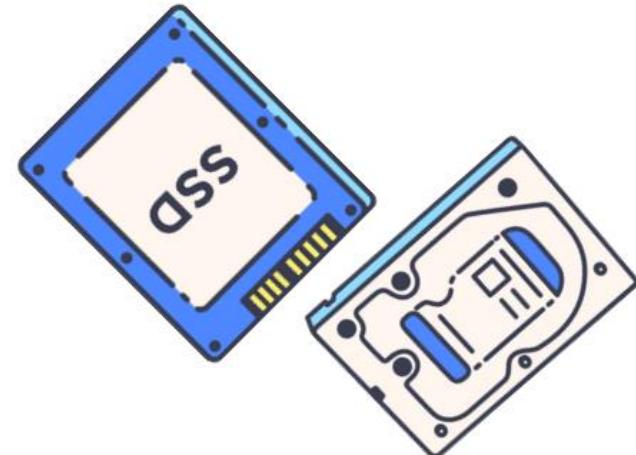


Entidade – possui 1 ou mais valores

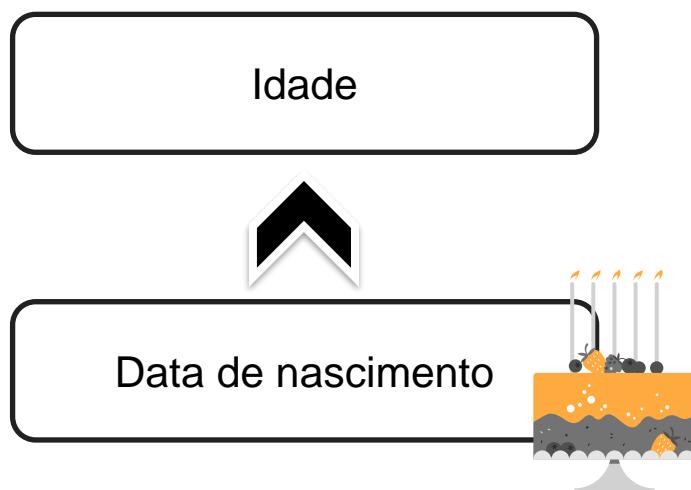
# Atributo Multivalorados



# Atributo Stored/Derivados



# Atributo Stored/Derivados



# Atributo Nulos



# Atributo Nulos



Grau (opcional)

Acadêmico

Rua A n° 23 apt 102

Apartamento

Rua B n° 490

Casa

&

# Atributo Nulos



UNKNOWN



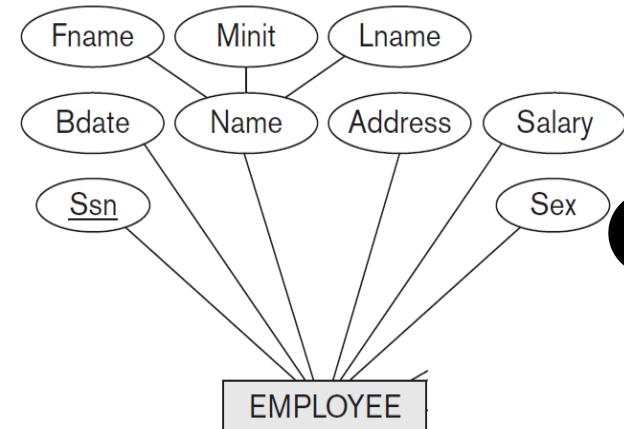
ITEM FALTANTE



# Atributo Complexos

Multivalorados

Compostos



1

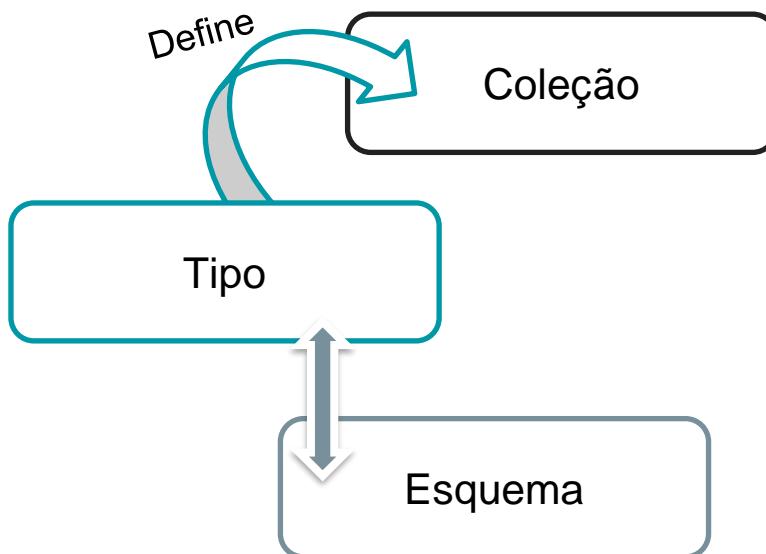
Grupo

{Address\_phone( {Phone(Area\_code,Phone\_number)},Address(Street\_address  
(Number,Street,Apartment\_number),City,State,Zip ) )}

Fonte: livro de referência - Navathe

2

# Entidades



Entity Type Name:

EMPLOYEE

Name, Age, Salary

$e_1$ •
(John Smith, 55, 80k)
$e_2$ •
(Fred Brown, 40, 30K)
$e_3$ •
(Judy Clark, 25, 20K)
⋮

Entity Set:  
(Extension)

COMPANY

Name, Headquarters, President

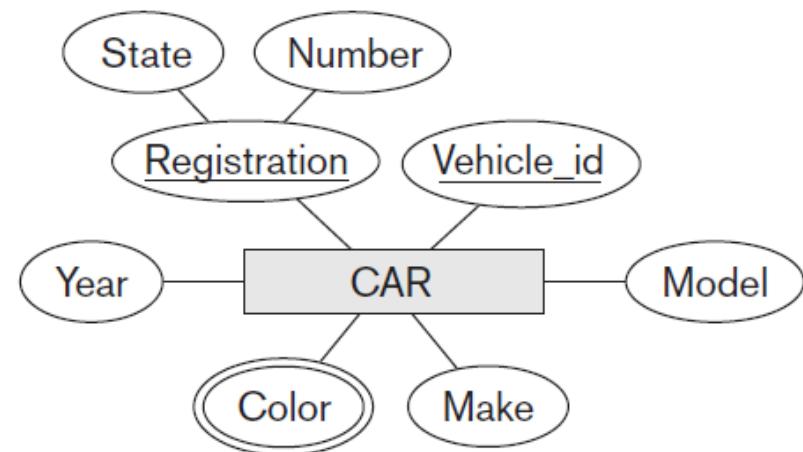
$c_1$ •
(Sunco Oil, Houston, John Smith)
$c_2$ •
(Fast Computer, Dallas, Bob King)
⋮

Fonte: livro de referência - Navathe

# Entidades

## Elementos DER

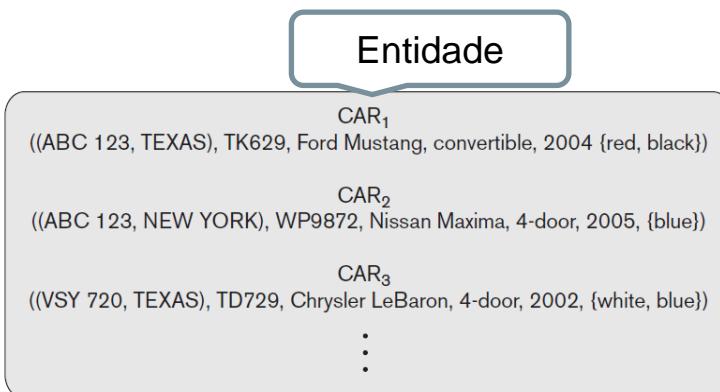
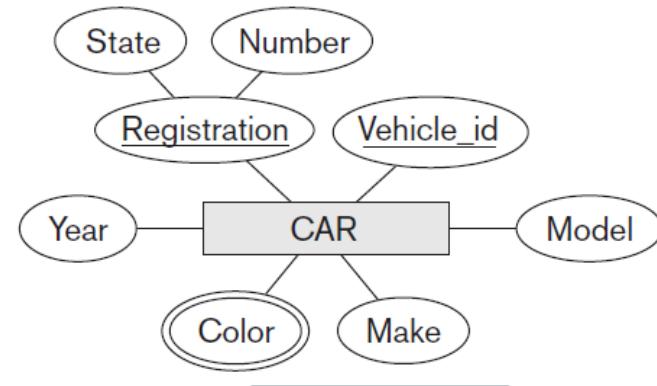
- Entidades
- Atributos
- Chaves



# Entidades

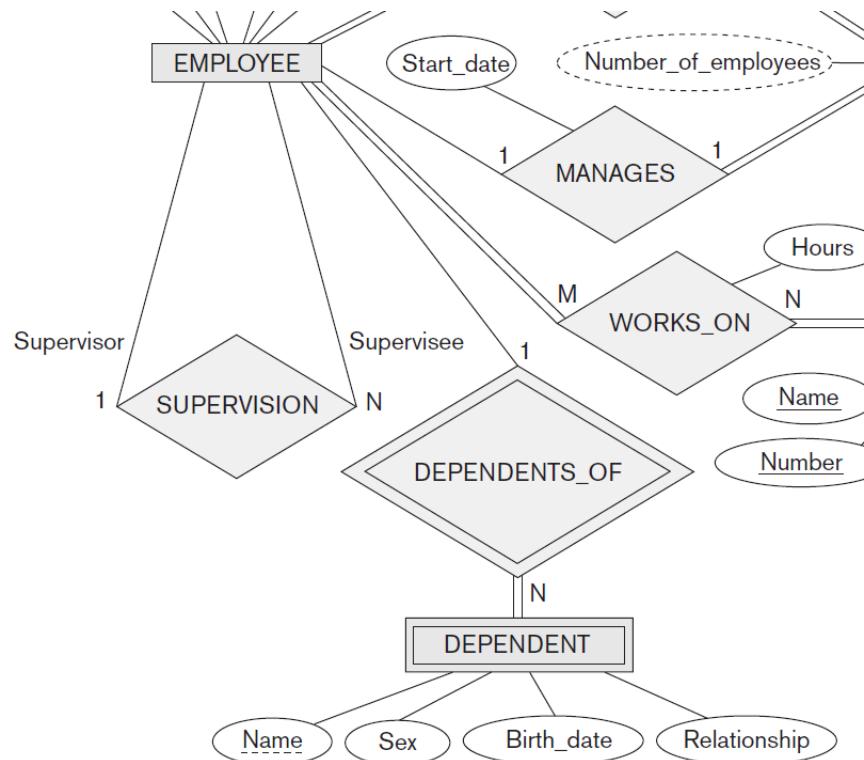
## Elementos DER

- Entidades
- Atributos
- Chaves



# Entidade Fraca

- Chave não obrigatória
- Depêndencia
- Exclusão cascata

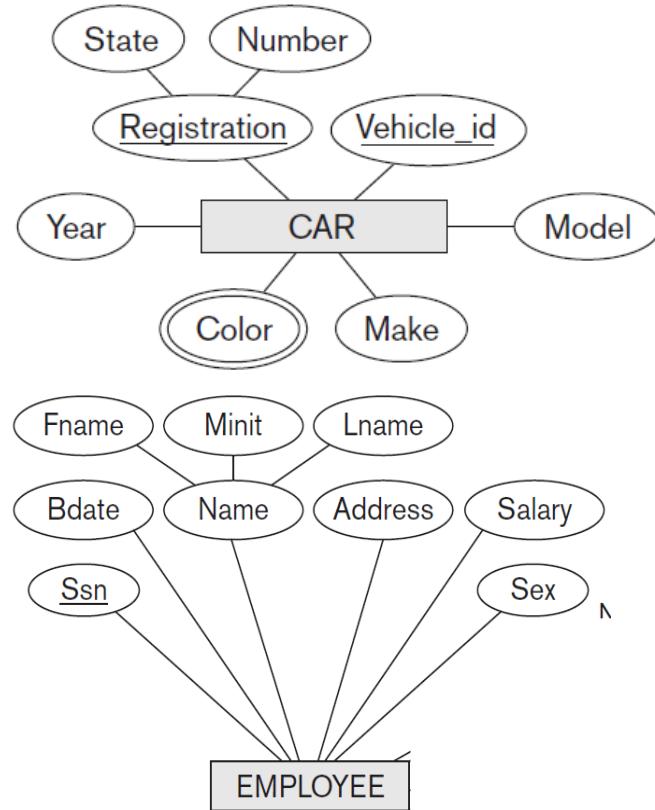


Fonte: livro de referência - Navathe

# Chaves

## Elementos DER

- Entidades
  - Atributos
  - Chaves
- Constraint

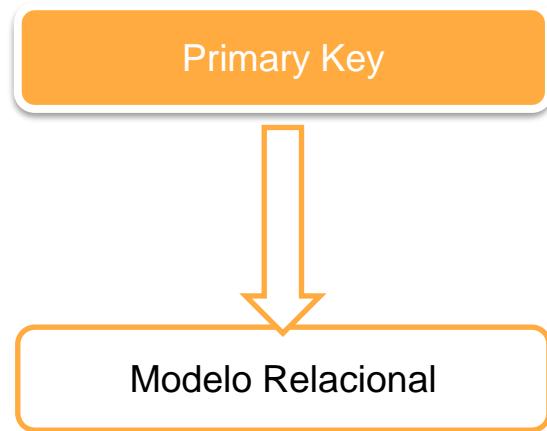
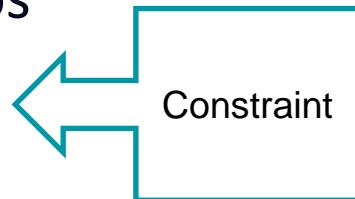


Fonte: livro de referência - Navathe

# Chaves

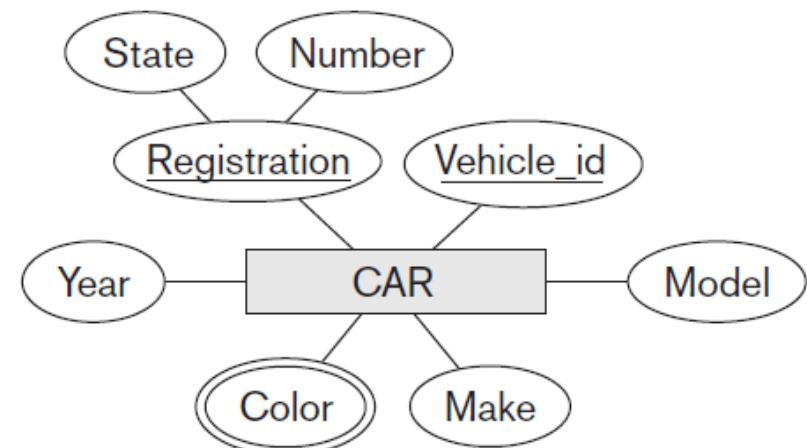
## Elementos DER

- Entidades
- Atributos
- Chaves



# Domínio dos atributos

- Range de Valores
- Tipo de dados
- ~~ER x UML~~

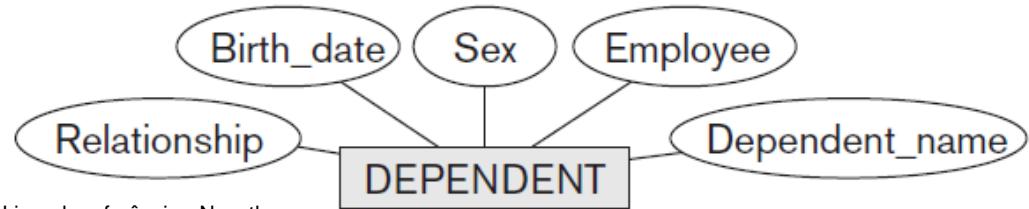
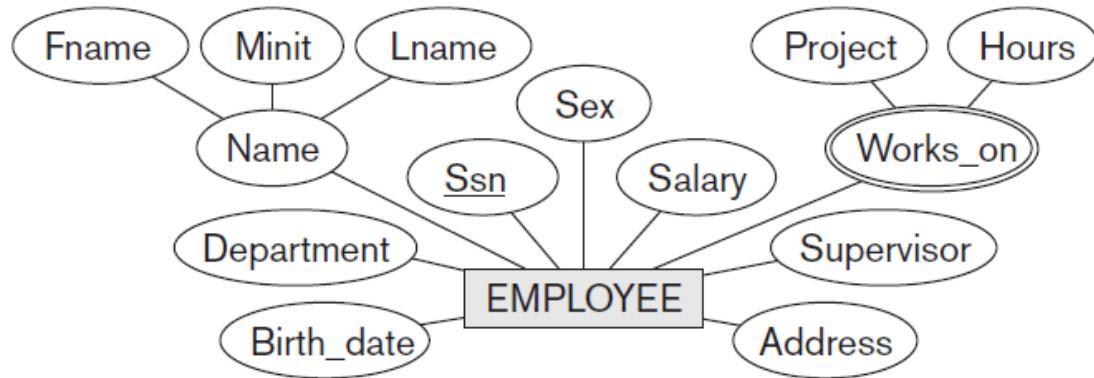


Fonte: livro de referência - Navathe

# Exemplo

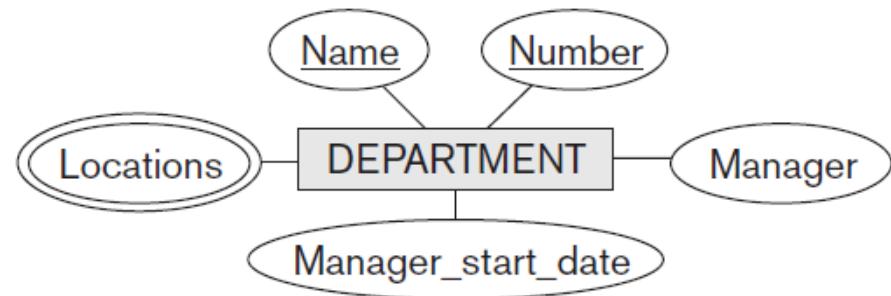
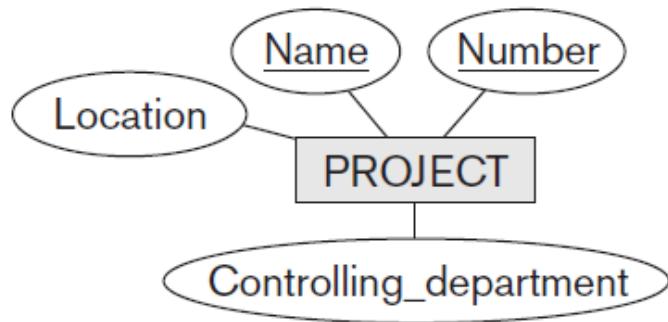


# Exemplo - Company



Fonte: Livro de referência - Navathe

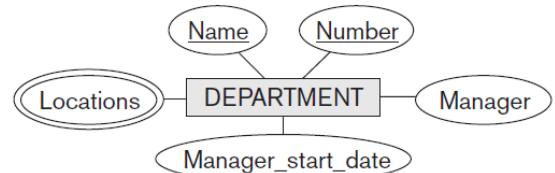
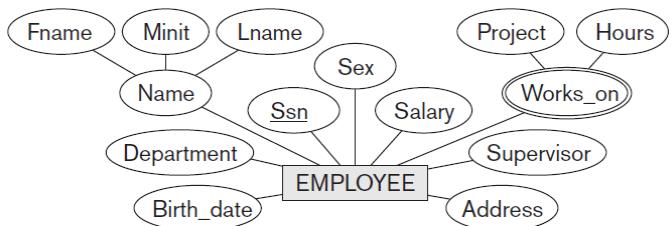
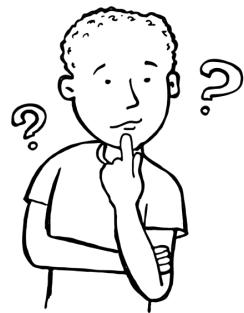
# Exemplo - Company



# Exemplo - Company

Outras características:

- Projetos x empregados
- Número de horas por projeto

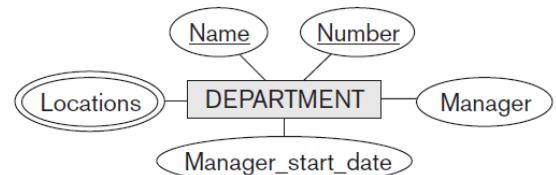
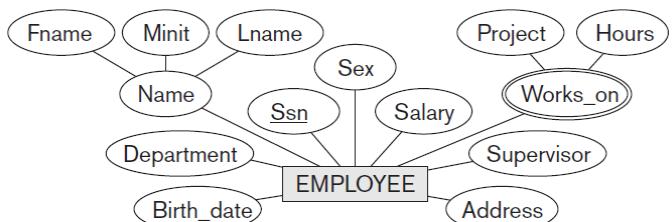


# Exemplo - Company

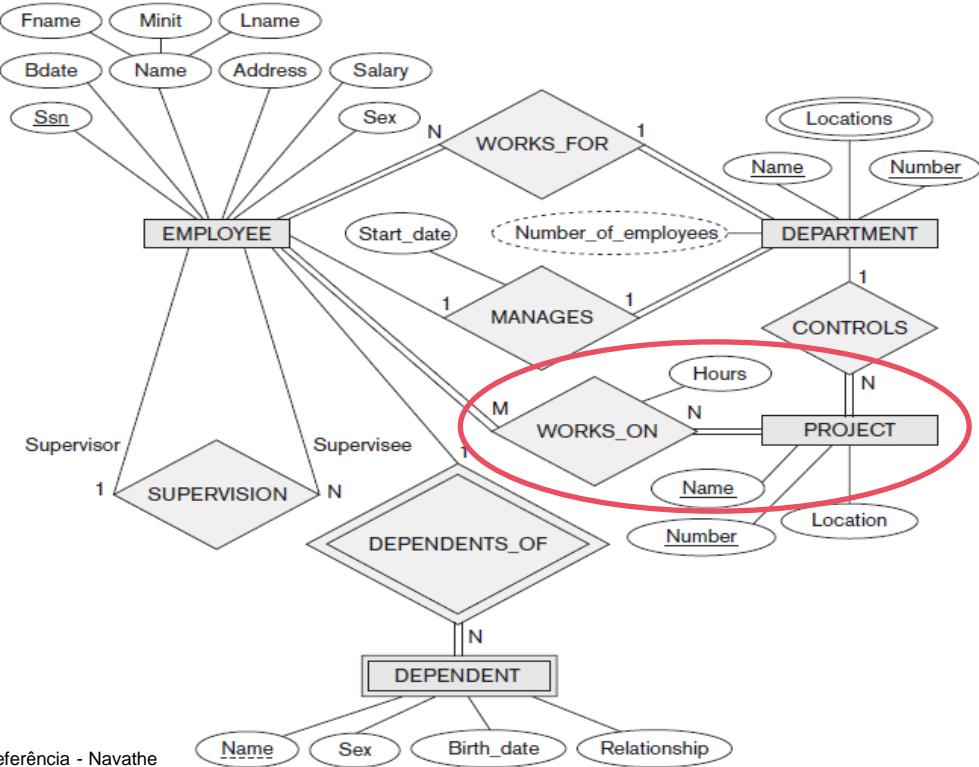
Outras características:

- Projetos x empresa
- Número de horas por projeto

Atributo composto

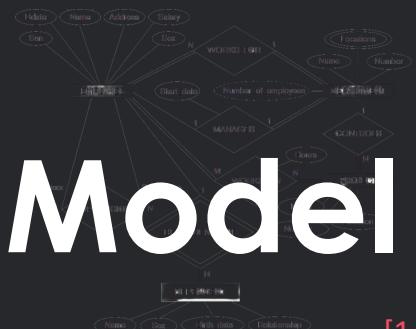


# Indo além ...



Fonte: Livro de referência - Navathe

# Relacionamentos, Papéis e Constraints estruturais

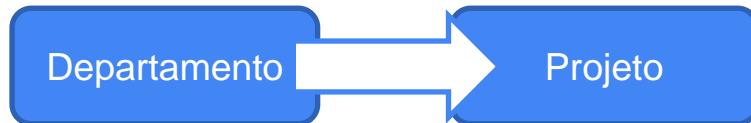
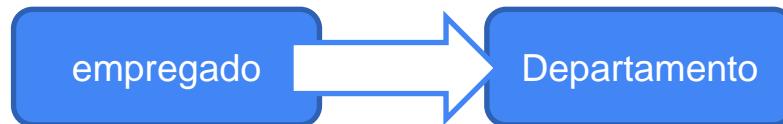


# ER Model

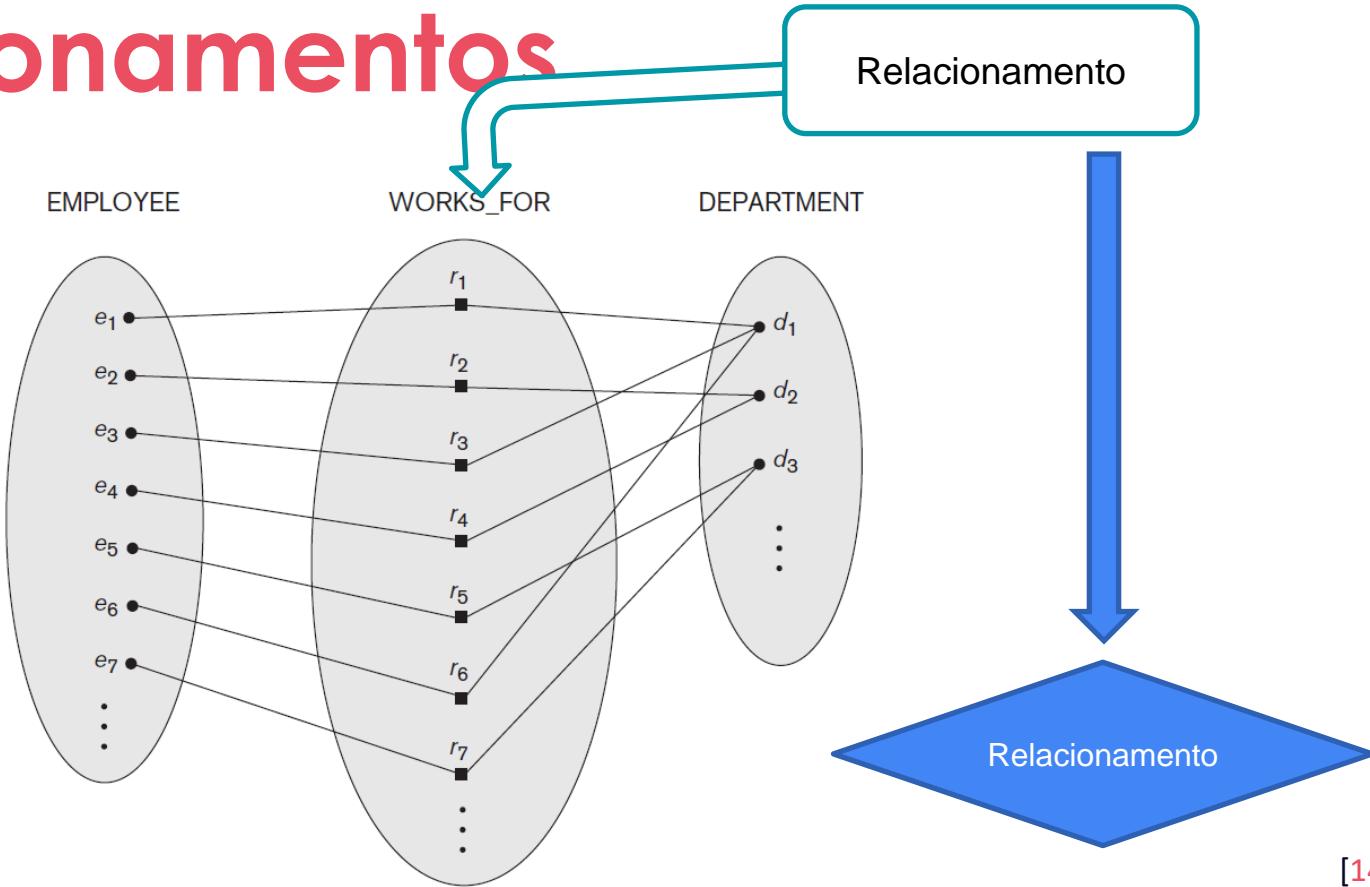
# Relacionamentos

## Relações implícitas

- Gerente do departamento
- Controle do projeto
- Supervisor



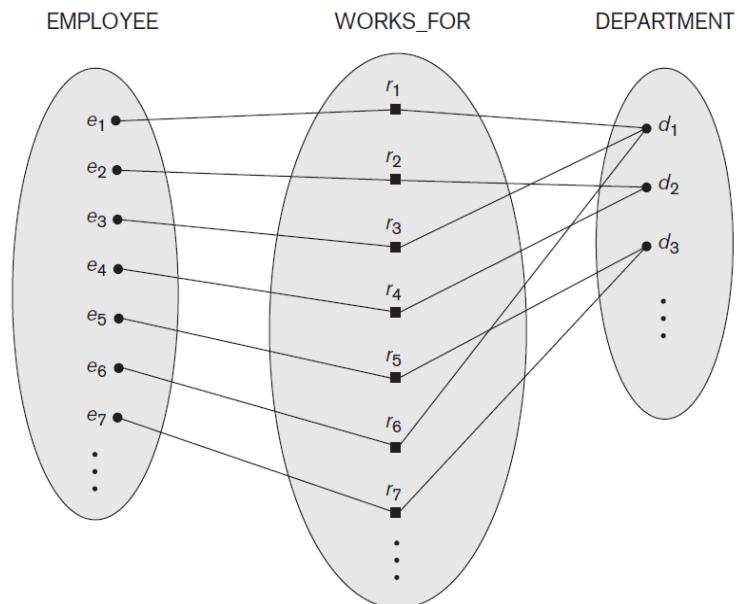
# Relacionamentos



# Relacionamentos

## Classificado

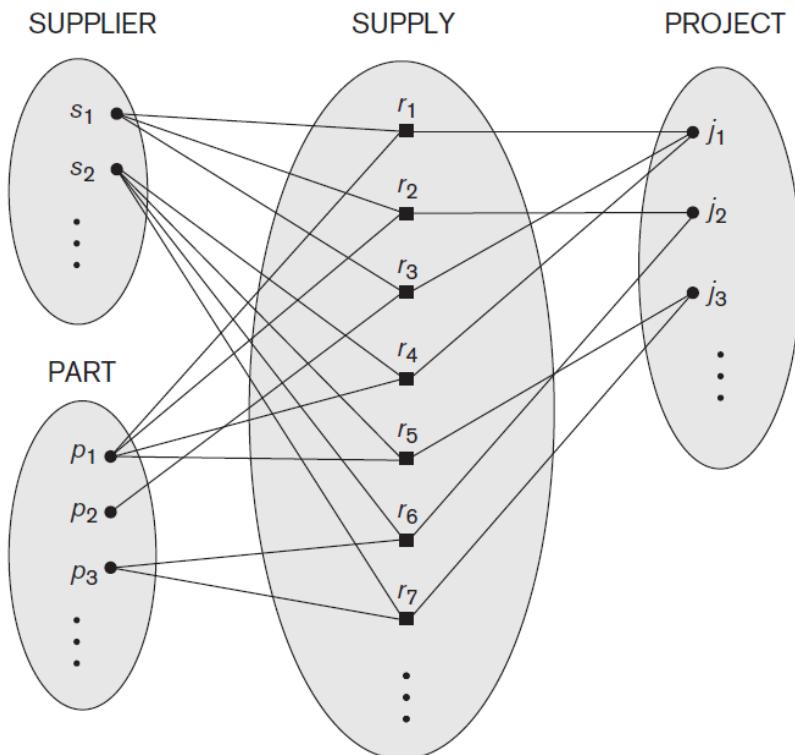
- Grau
- Auto-relacional ou não
- Cardinalidade



# Relacionamentos

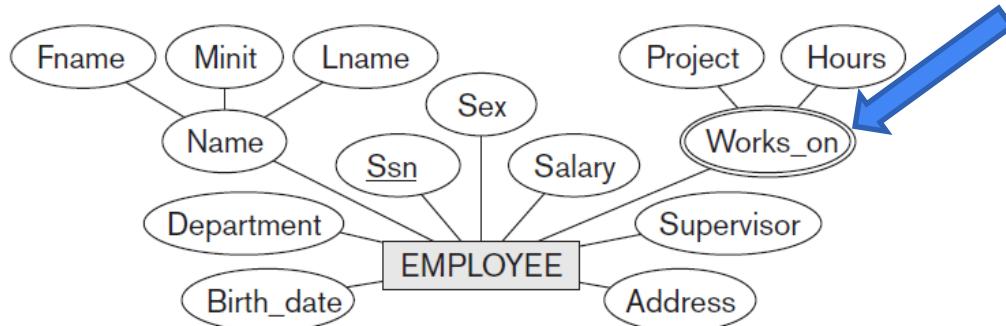
## Classificado

- Grau - Ternário ➔
- Auto-relacional ou não
- Cardinalidade

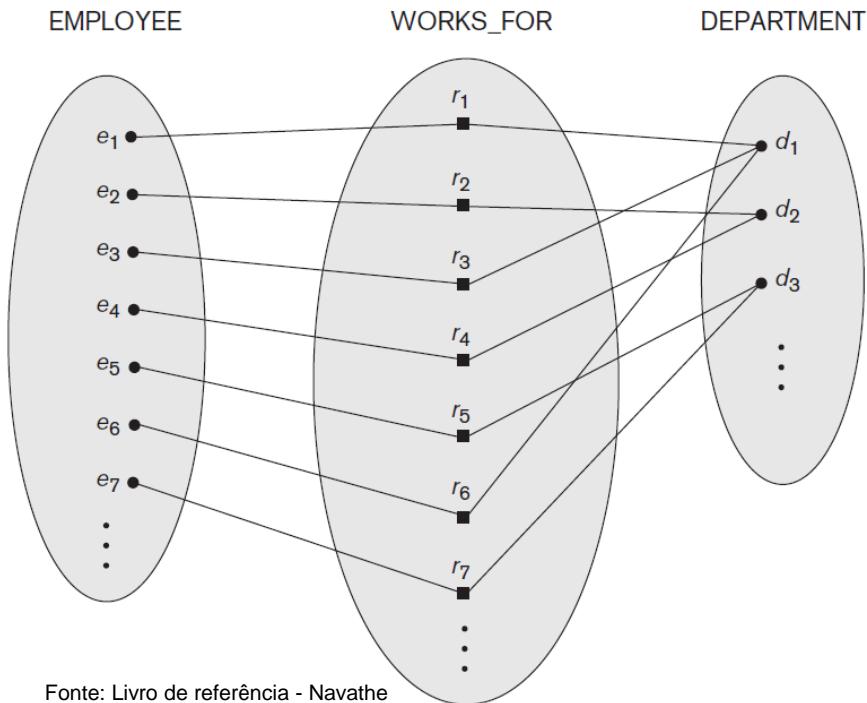


# Relacionamentos

## Relacionamentos como atributos



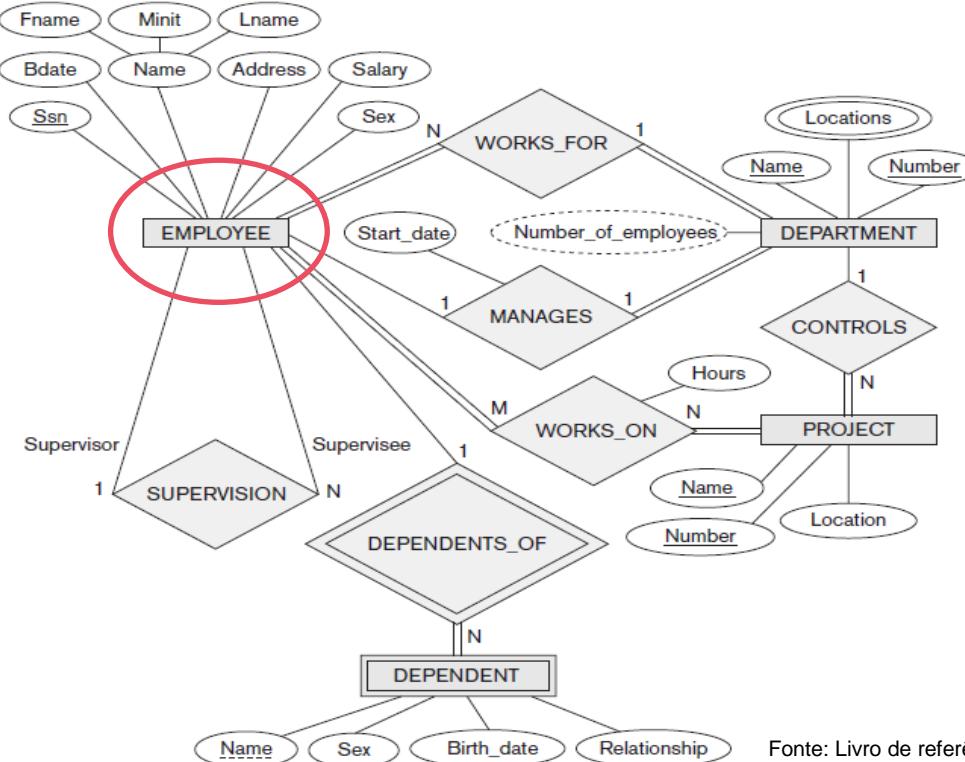
# Papel | Atribuição



Papéis de cada entidade

Significado

# Papel | Atribuição

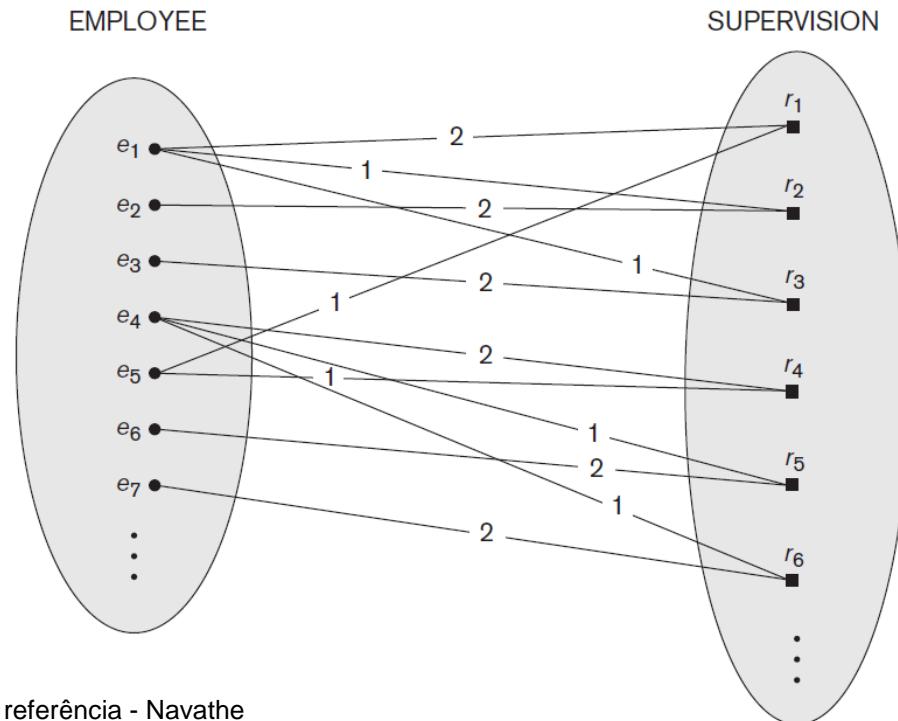
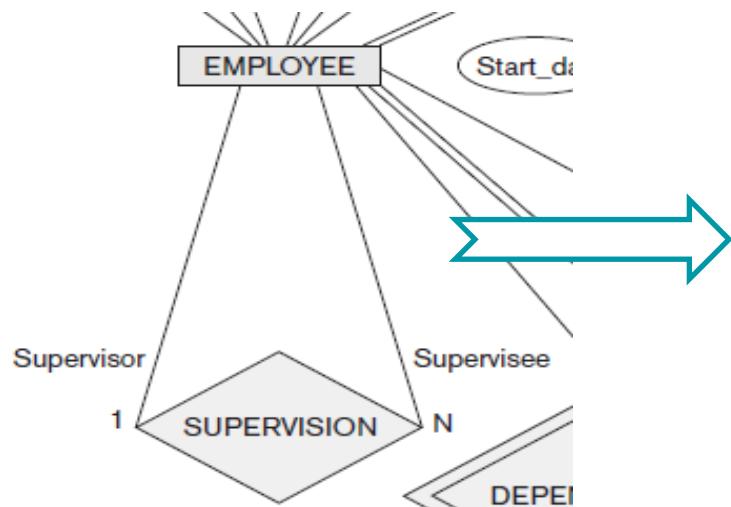


Papéis de cada entidade

Fonte: Livro de referência - Navathe

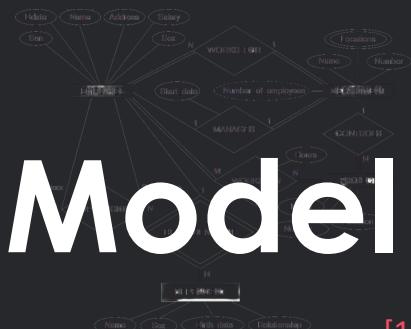
[148]

# Auto-relacionamento



Fonte: Livro de referência - Navathe

# Constraints de relacionamentos binários



# ER Mode

# Constraints - Cardinalidade

1:1

N:1

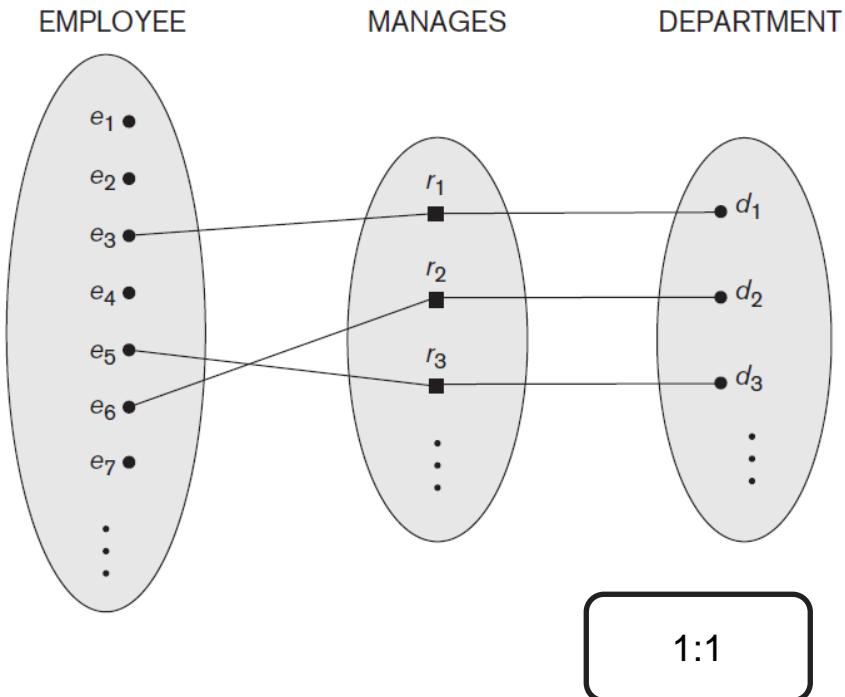
1:N

Cardinalidade corresponde ao nº máximo de instância que participam de um determinado relacionamento.

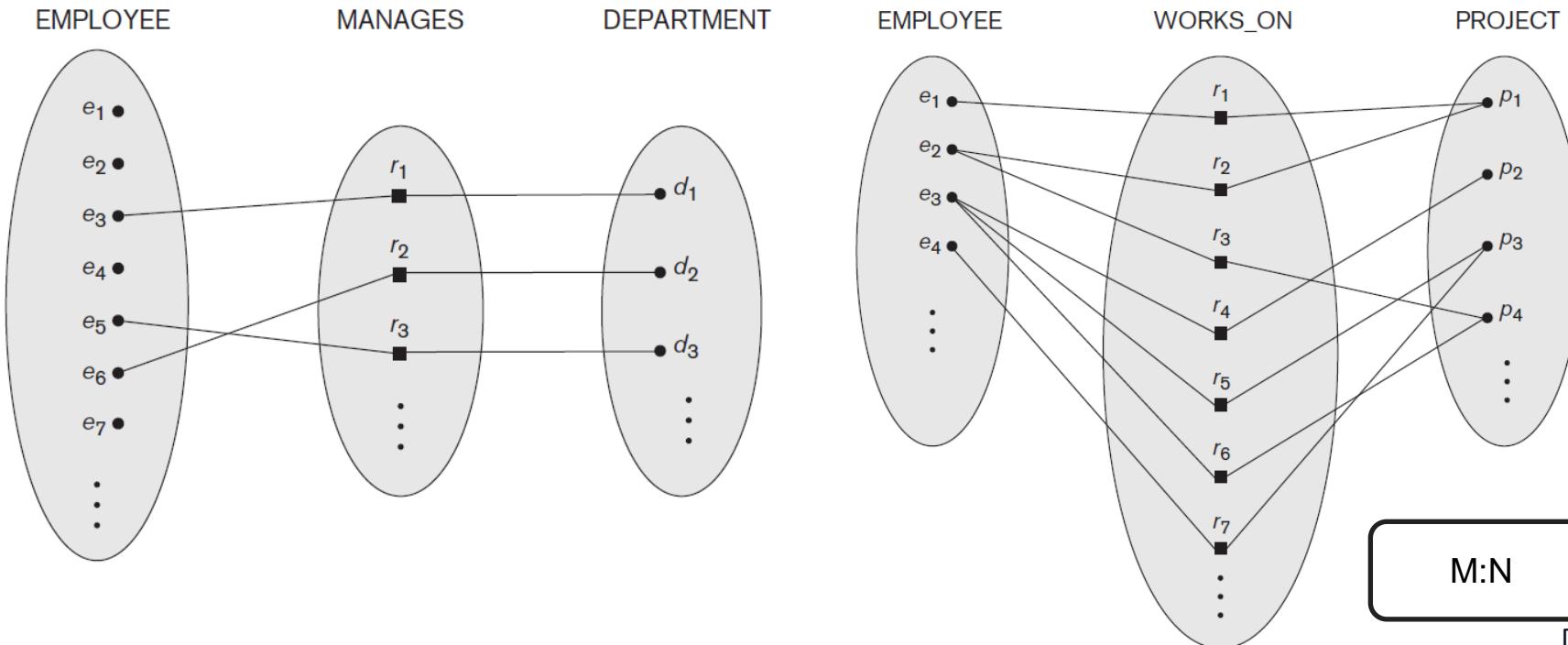
1:3

M:N

# Constraints - Cardinalidade

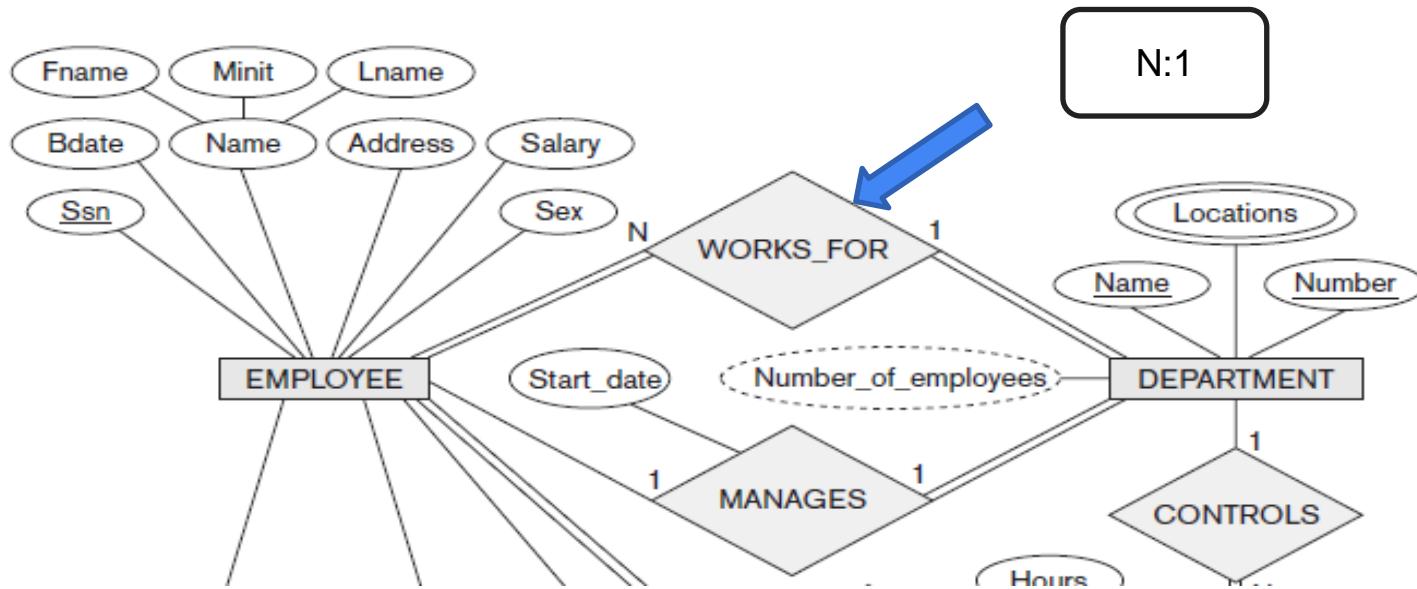


# Constraints - Cardinalidade



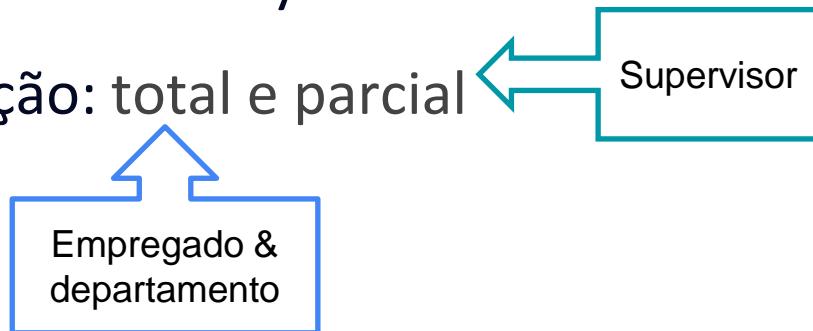
M:N

# Constraints - Cardinalidade

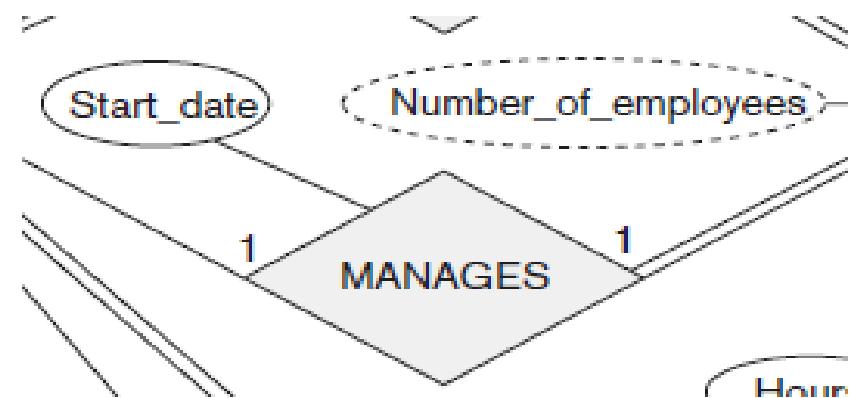
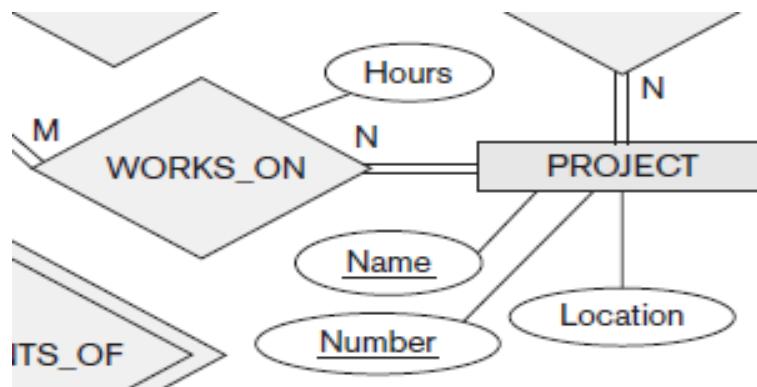


# Constraints - Participação

- Relação de dependência
- Minimum Cardinality Constraint
- Classificação: total e parcial

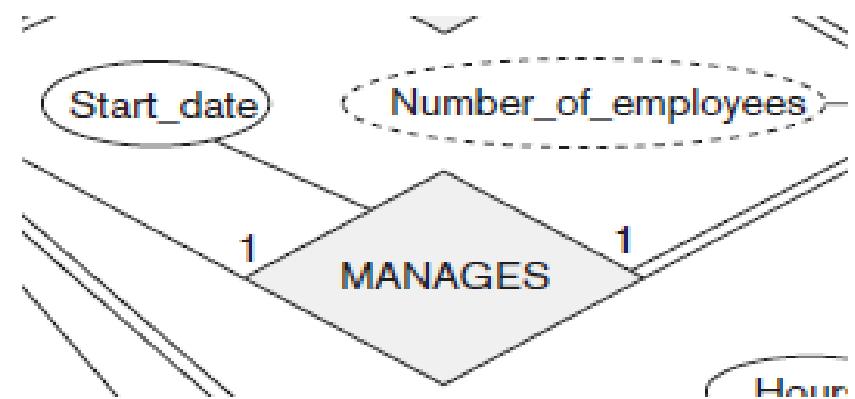
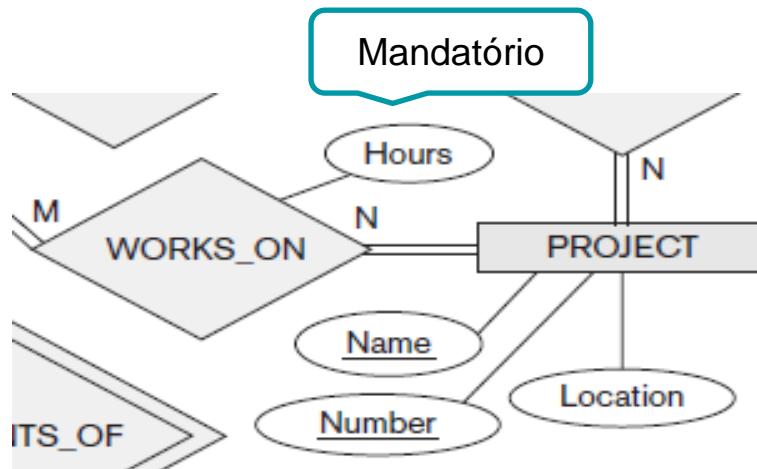


# Atributos de relacionamento



Decisão subjetiva de Design

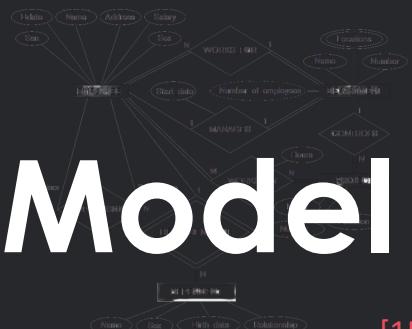
# Atributos de relacionamento



Decisão subjetiva de Design

# Tipos de Entidades

## Fracas



# ER Model

# Entidade Fraca

## Tipos de entidades

- Fracas
- Fortes



Owner Entity Type

Dependem de outra

Relacionamento  
ID

# Entidade Fraca

Total Participation Constraint

## Tipos de entidades

- Fracas
- Fortes



Owner Entity Type

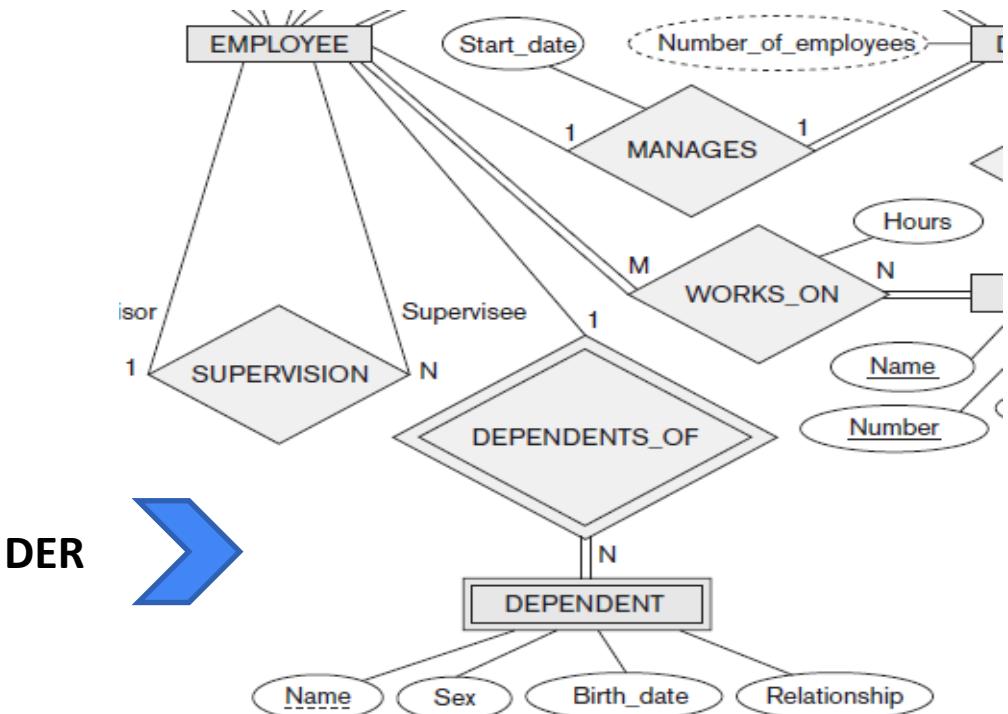
Dependem de outra

Relacionamento  
ID

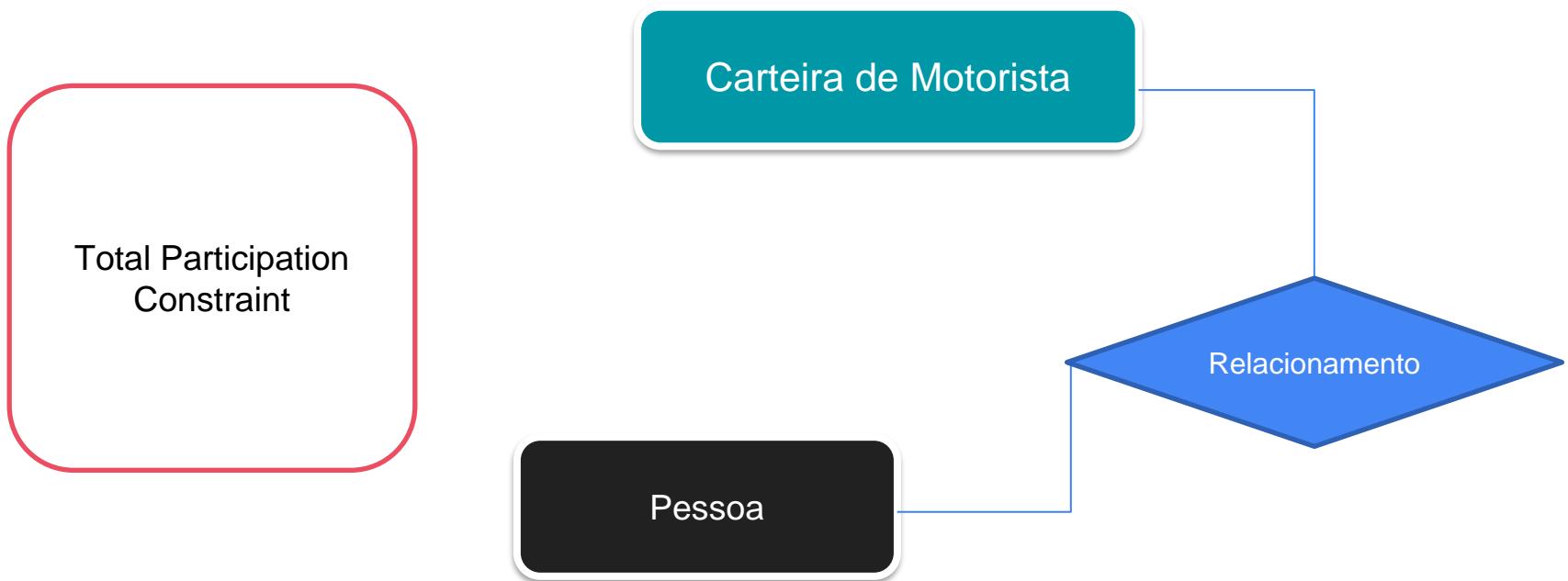
# Entidade Fraca

# Tipos de entidades

- Fracas
  - Fortes

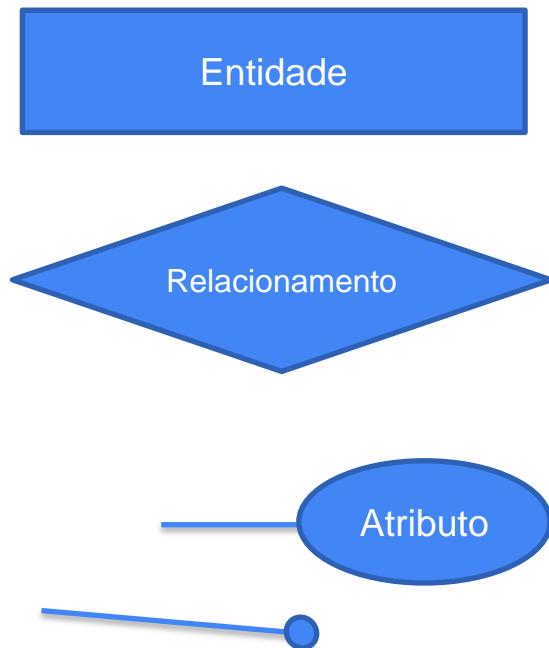


# Entidade Fraca



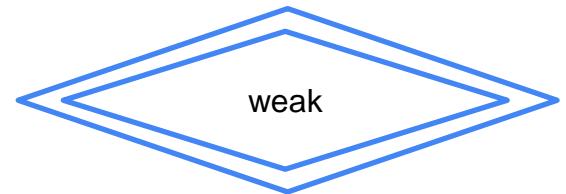
# Notações

- **Entidades:** classes/objetos
- **Relacionamentos:** agregações
- **Atributos:** propriedades elementares



# Notações

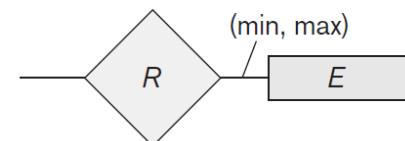
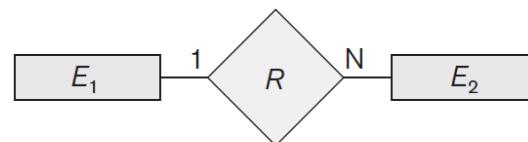
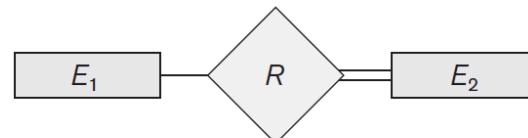
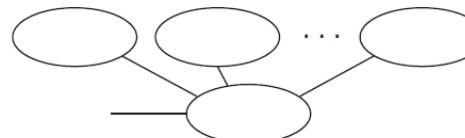
- **Weak Entidades, relacionamentos**
- **Chave Parcial:** Linha pontilhada
- Chave principal: linha continua
- **Dependência de existência:** ||



# Notações



- Atributo derivado
- Atributo composto
- Participação total
- Cardinalidade
- Constraint estrutural



# Construtores do Esquema

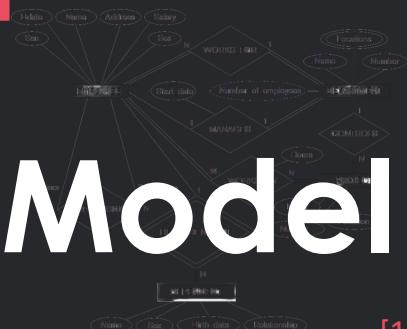
## Convenção - Nomeando

- Entidade e Relacionamento
- Atributo
- função

Significado

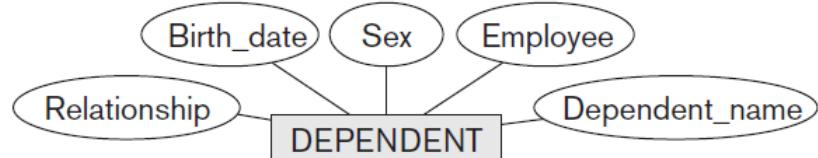
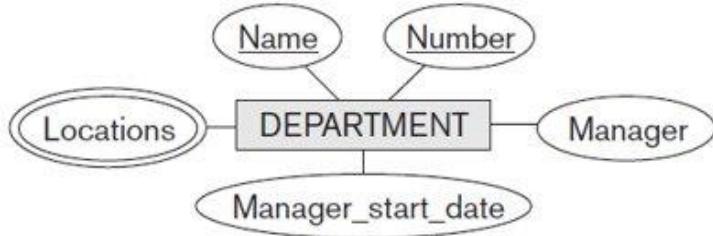
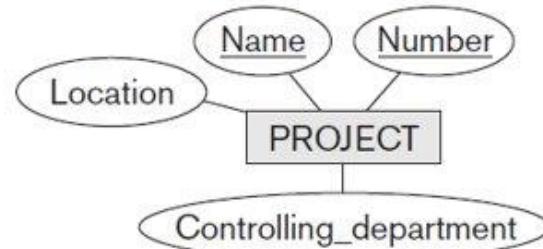
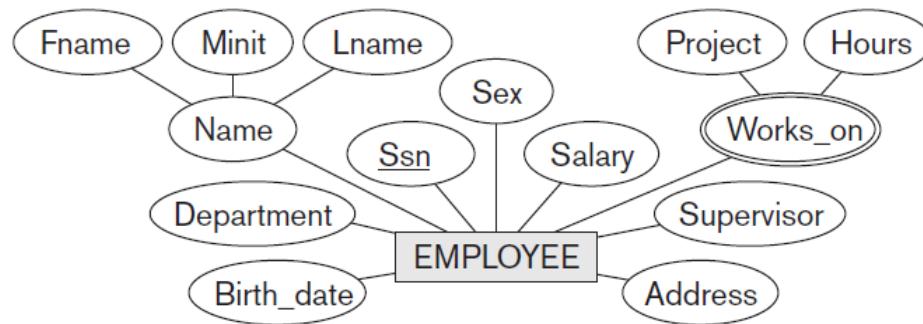
Trivial?

# Refinando o Design ER para COMPANY



# ER Mode

# Exemplo - Company



# Refinando - Company

## Relacionamentos

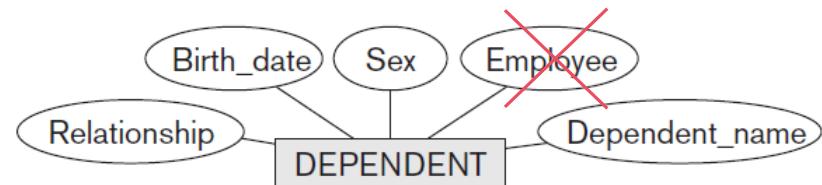
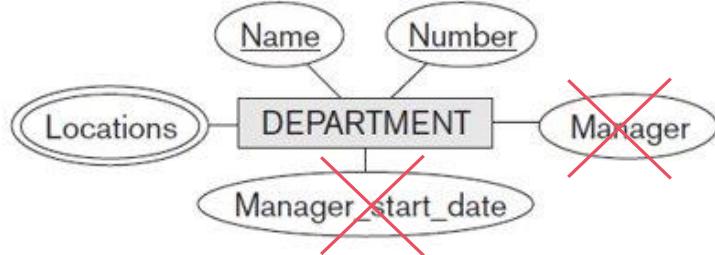
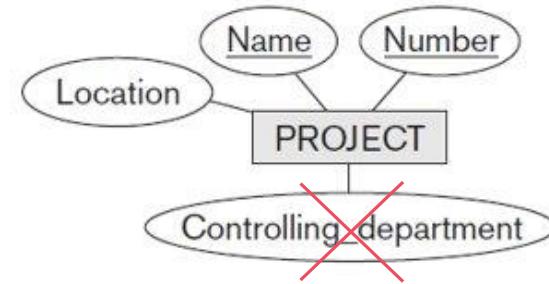
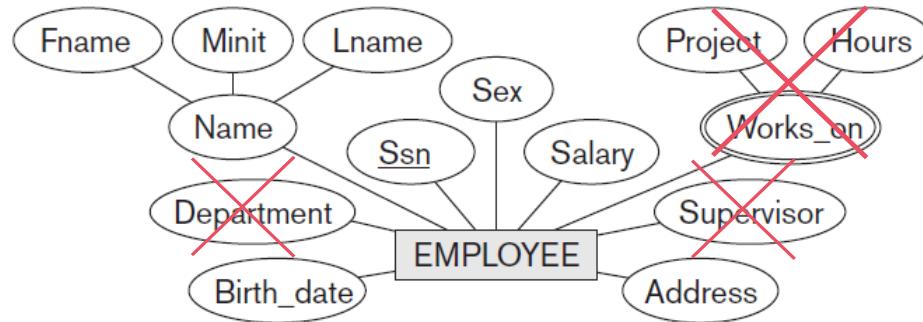
- Empregado x Departamento: **Gerencia**
- Empregado x Departamento: **Trabalha para**
- Departamento x Projeto: **Controla**

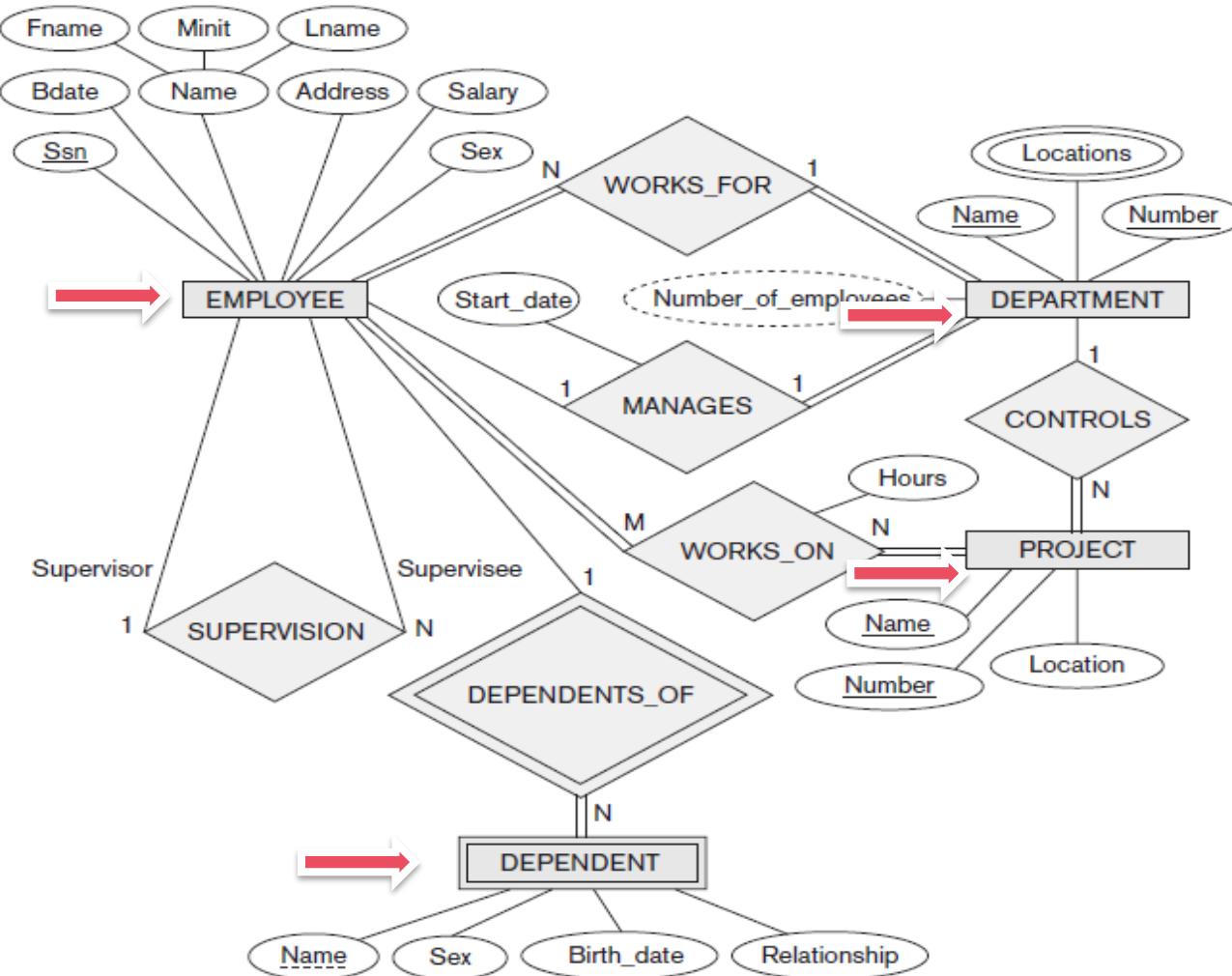
# Refinando - Company

## Relacionamentos

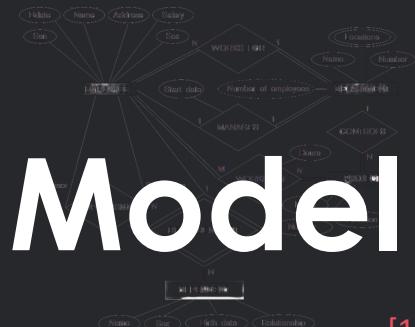
- Empregado x Empregado: **Supervisiona**
- Empregado x Projeto: **Trabalha em**
- Empregado x Dependente: **Dependentes de**

# Exemplo - Company





# Alternativas - UML



# ER Model

# Outra Notação

- Linguagem para desenvolvimento de software
- Trabalha com visões: interpretada e construída

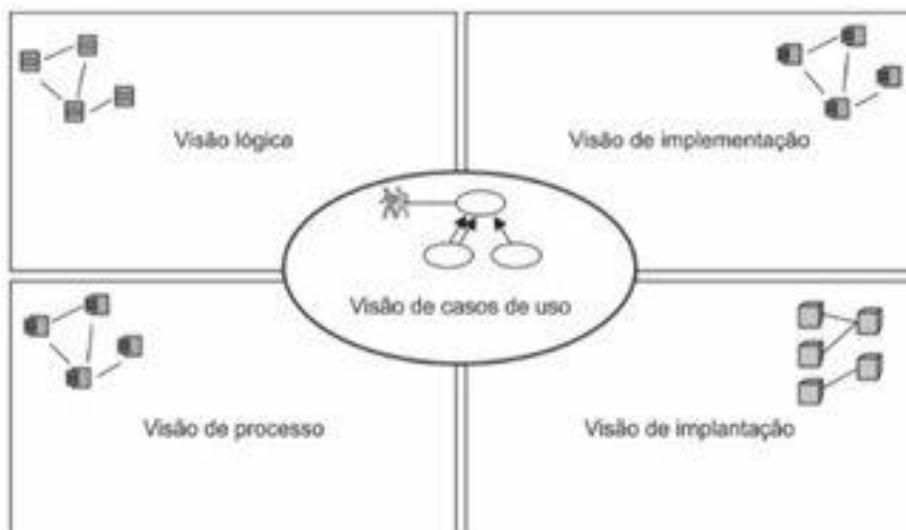


Compreensão  
facilitada

Liberdade para o  
desenvolvedor

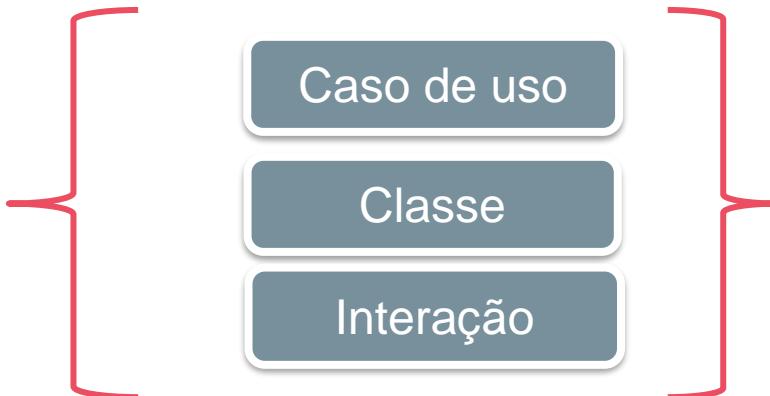
Orientação à  
objeto

# Outra Notação



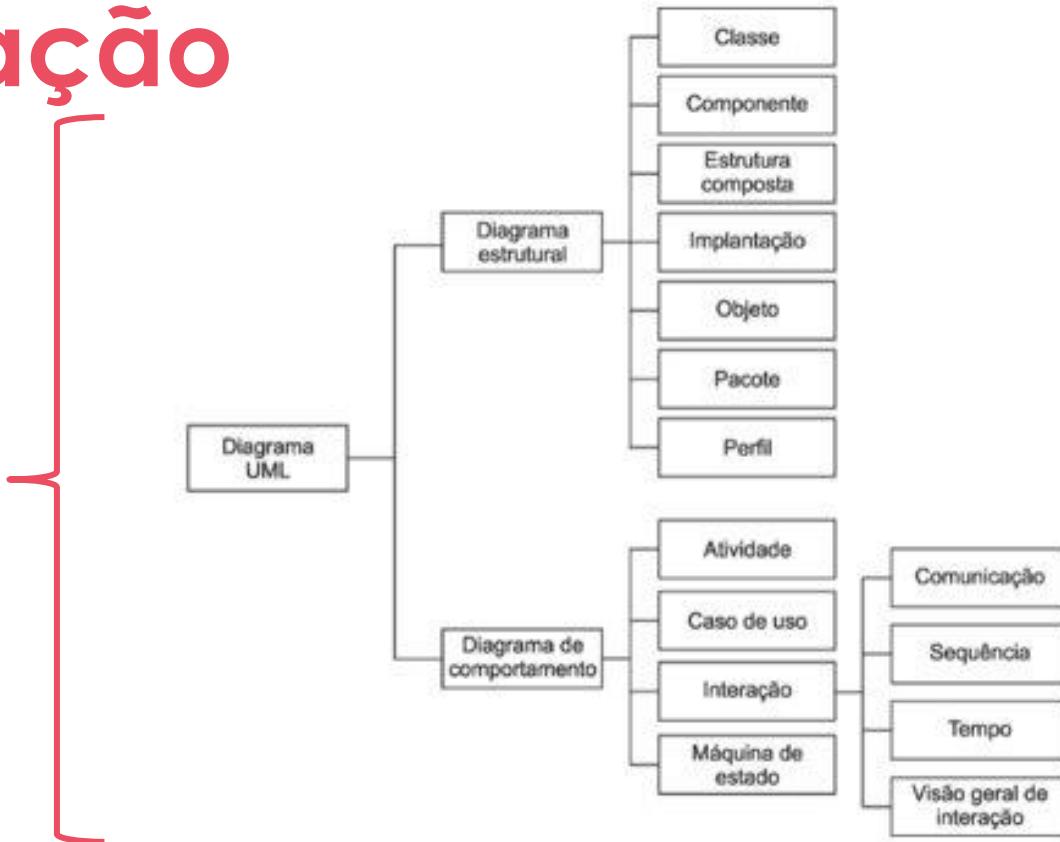
# Outra Notação

Principais



# Outra Notação

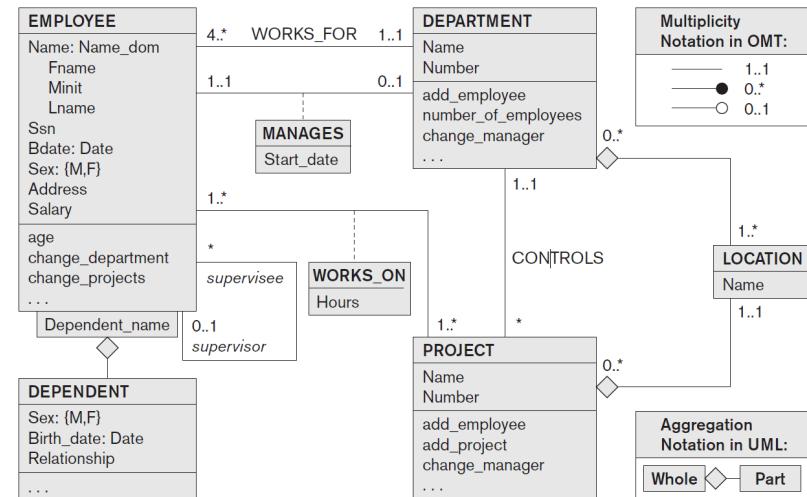
Classificação



# Outra Notação



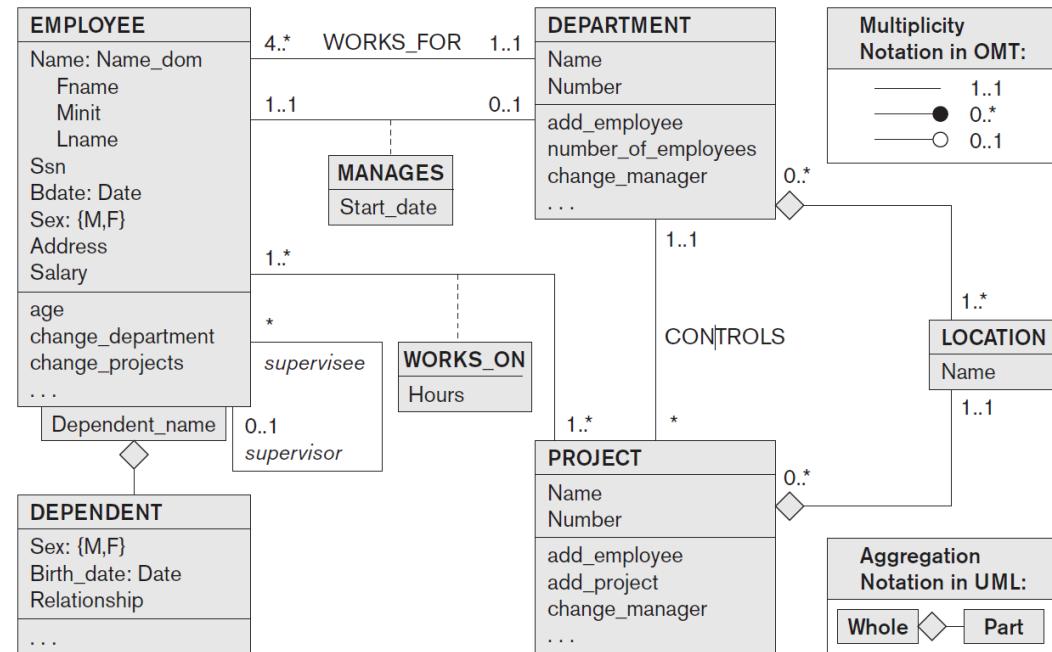
- Desenvolvimento de software
- Paradigma OO
- **Foco:** Diagrama de Classes



# Outra Notação



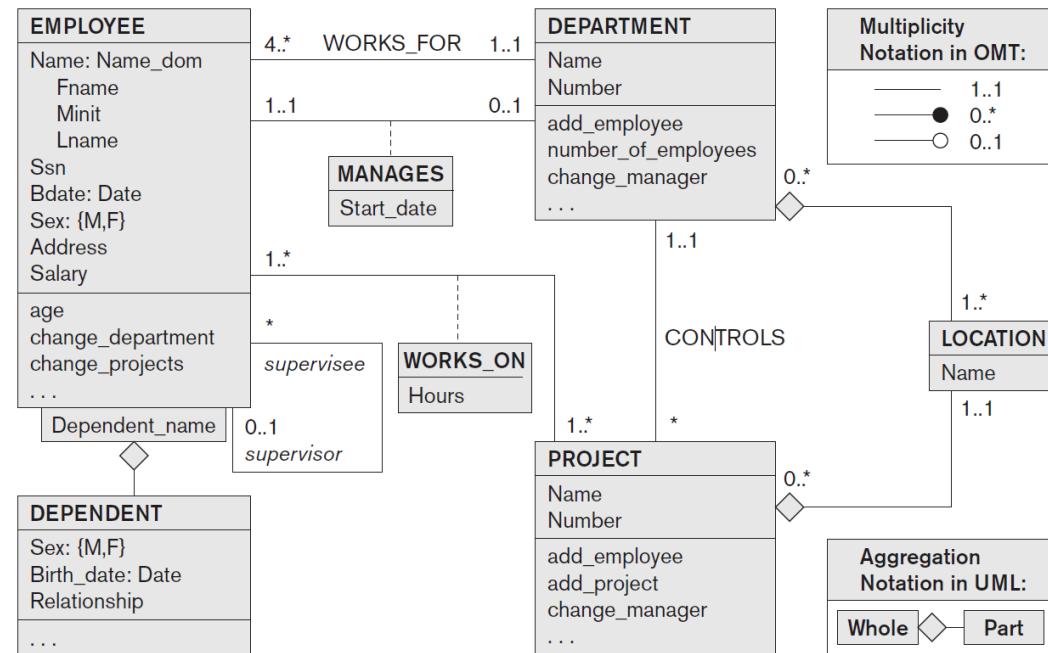
- Classe (entidade)
- Atributos
- Operações
- Associações



# Outra Notação



- Atributo de Link
- Associação binária
- Associação reflexiva
- Multiplicidade \*



# Outra Notação

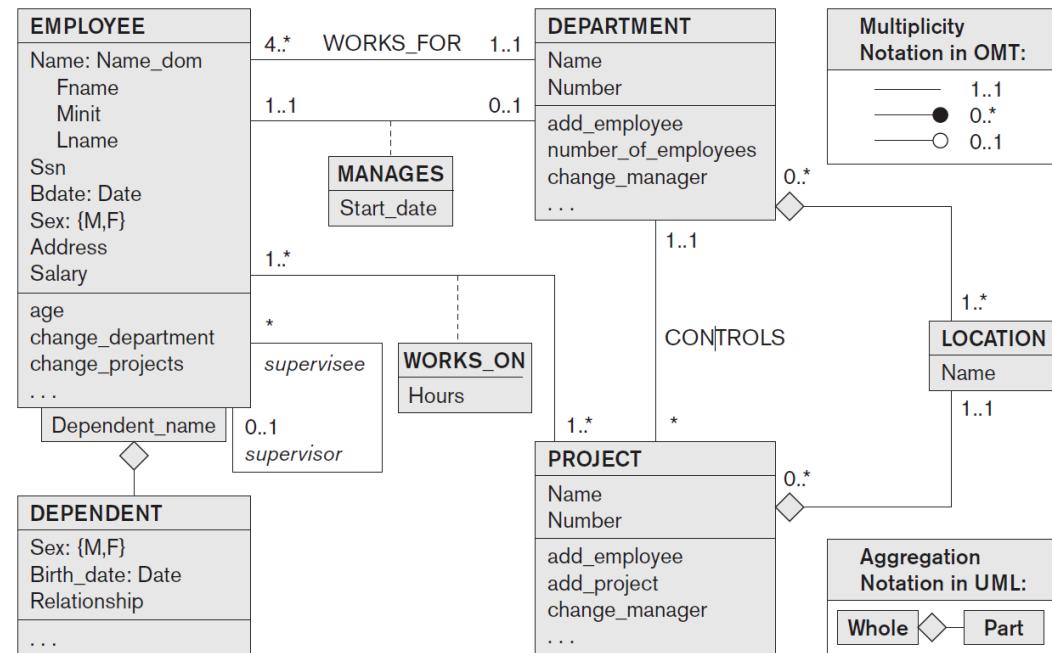


## Relacionamentos

- Associação e
- Agregação



Direcional

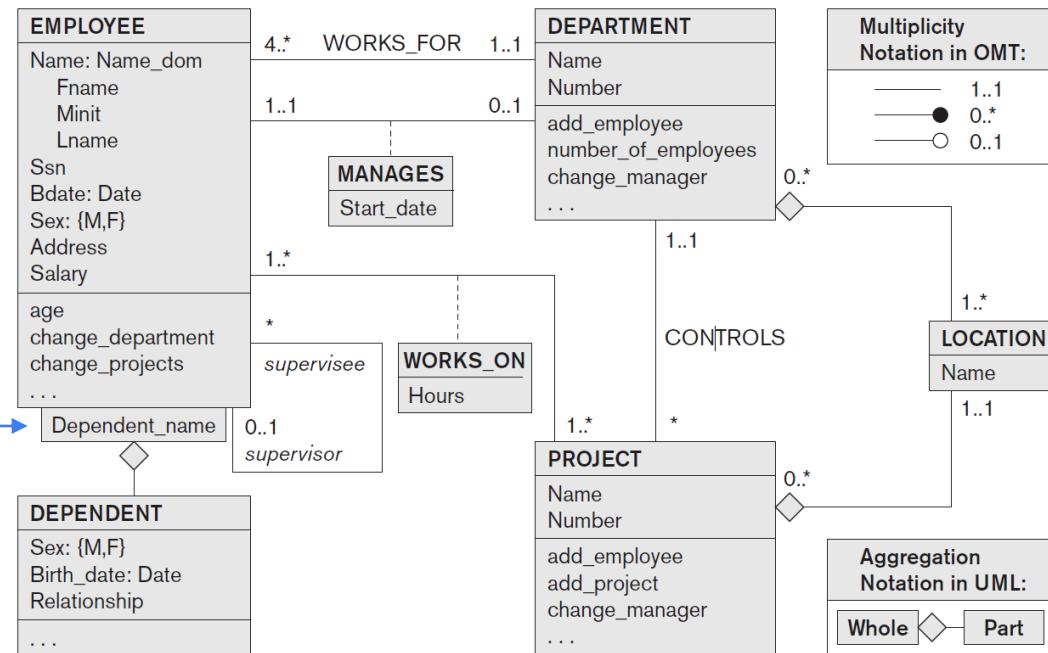


# Outra Notação

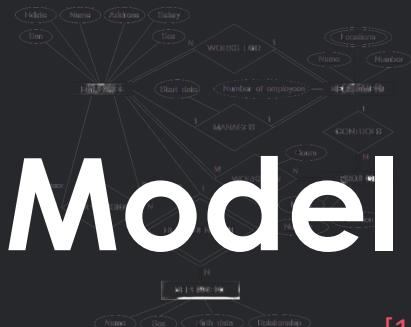


Weak

- qualified association
- Discriminator



# Relacionamentos de alto grau



# ER Mode

# Relacionamento N-ário

- Ternário, ..., N-ário
- Perspectiva diferente do binário

Obs:

- Instâncias refletem a visão

(s,j,p)

Trade-off na tomada  
de decisão

# Relacionamento N-ário

Cenário

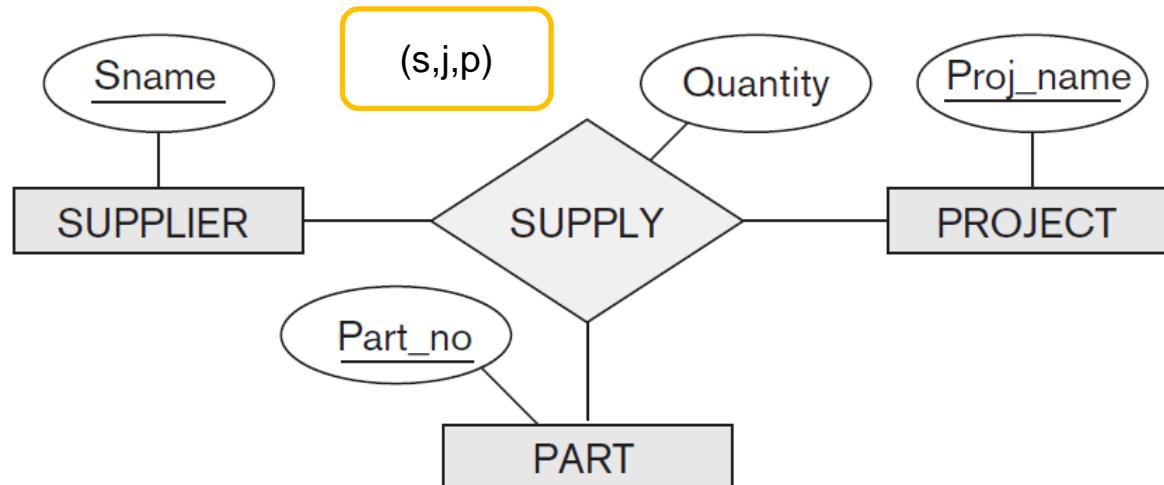
Fornecedor

Projeto

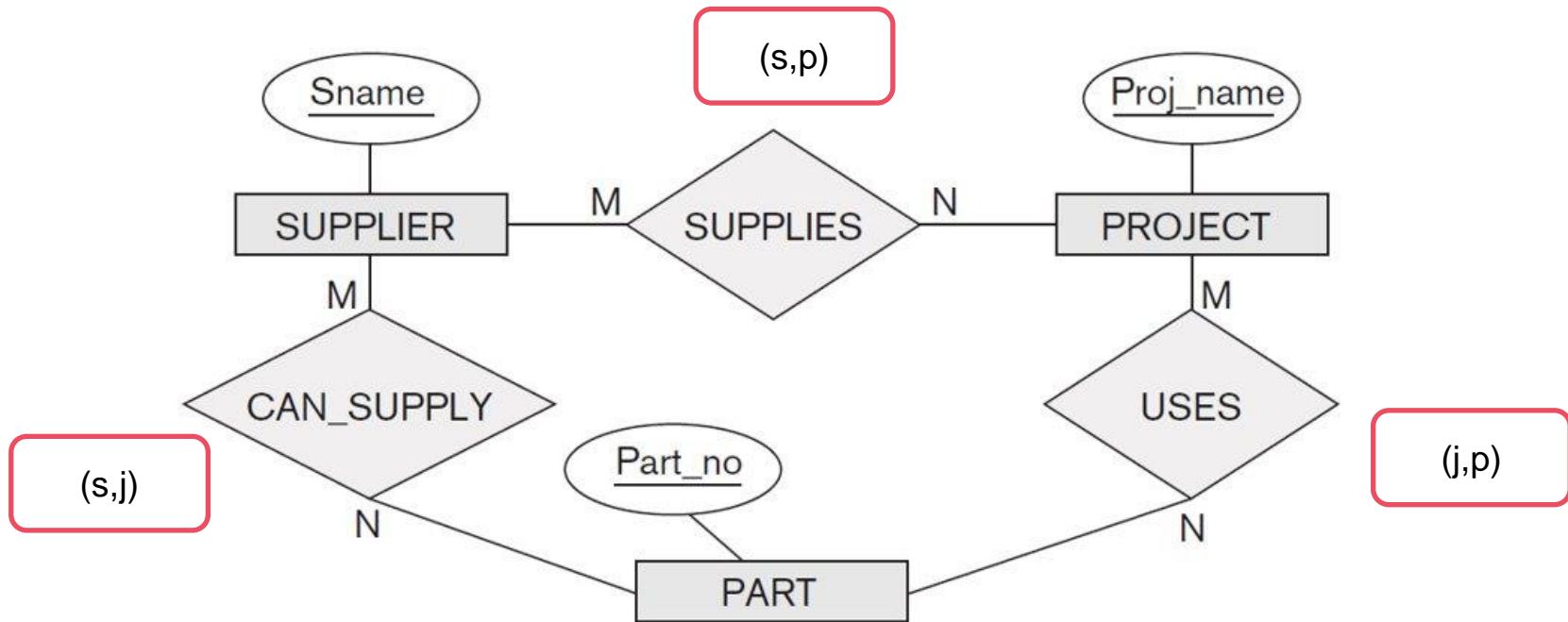
Produto

Estabelecimento (loja)

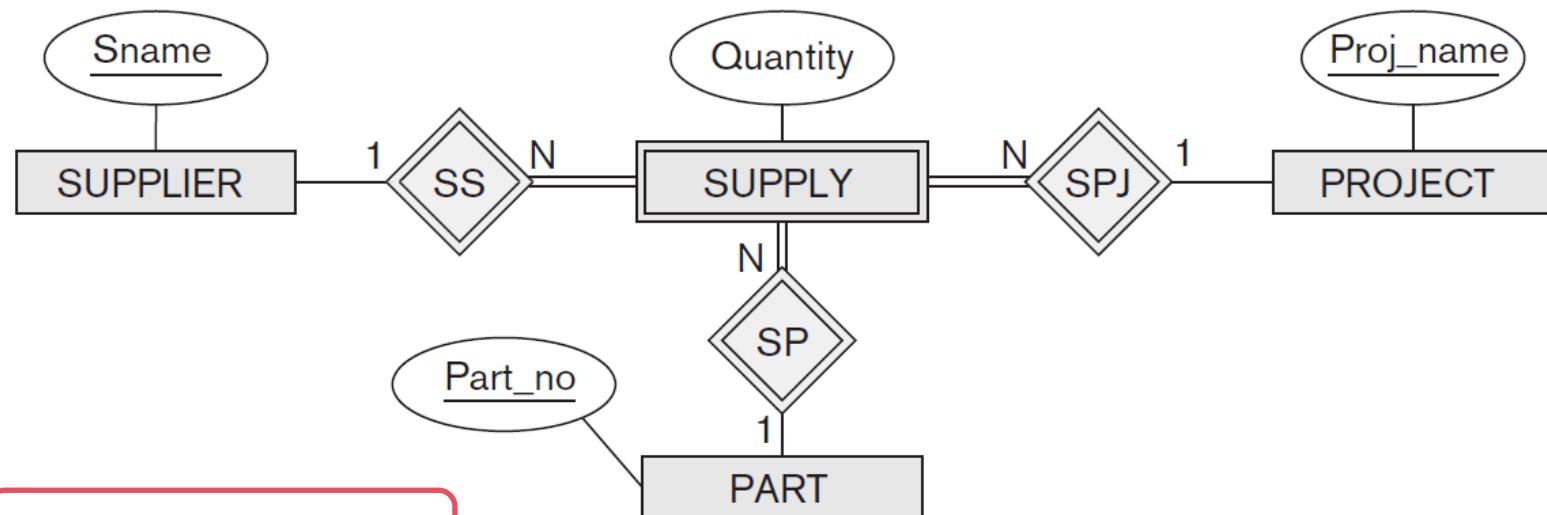
# Relacionamento N-ário



# Relacionamento N-ário



# Relacionamento N-ário



Weak & Depêndencia

# Relacionamento N-ário

Cenário

Consultas médicas

Médico

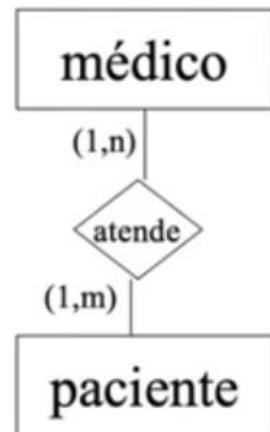
Paciente

Exames

# Relacionamento N-ário

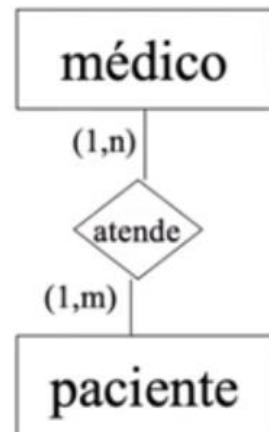
**Exame**

Com médico ou  
com paciente?



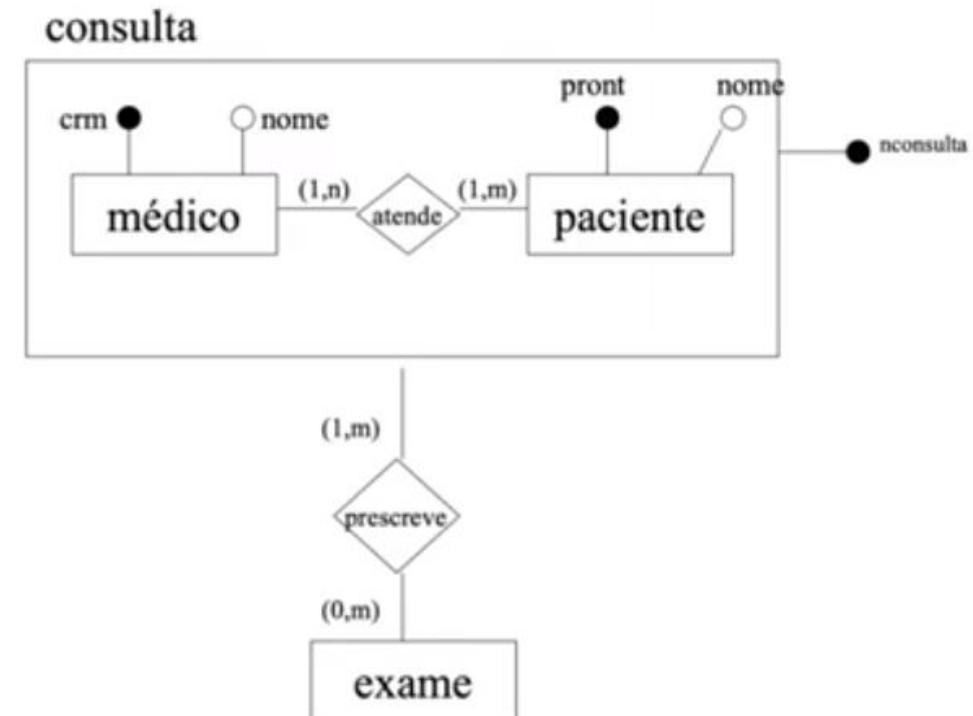
# Relacionamento N-ário

**Exame**  
Com médico ou  
com paciente?

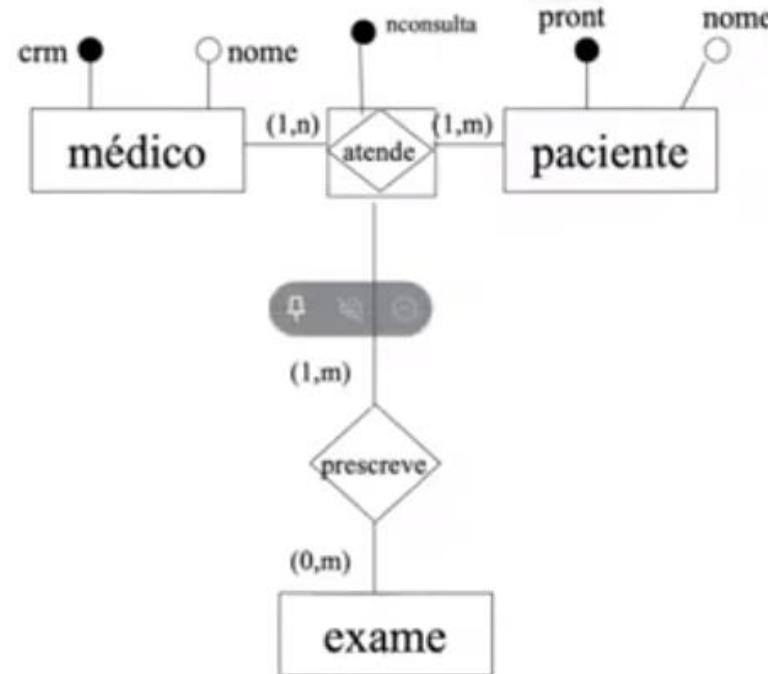


# Relacionamento N-ário

Agregação

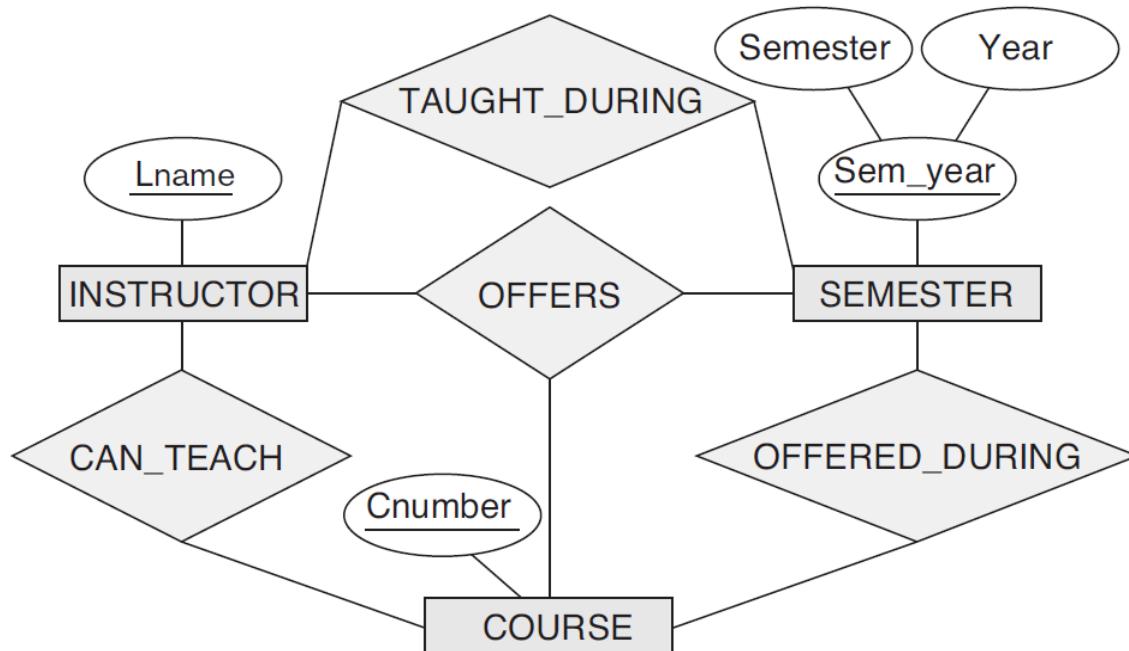


# Relacionamento N-ário

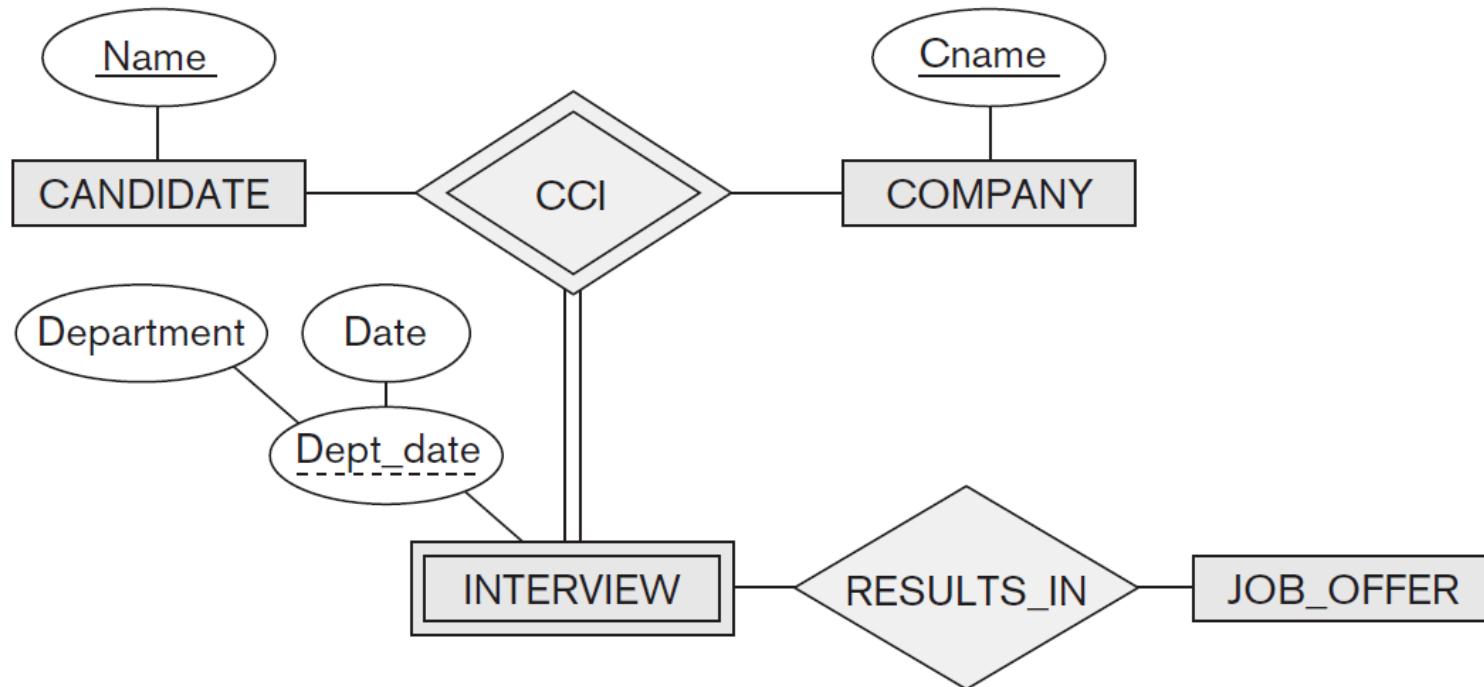


Ternário?

# Relacionamento N-ário



# Relacionamento N-ário

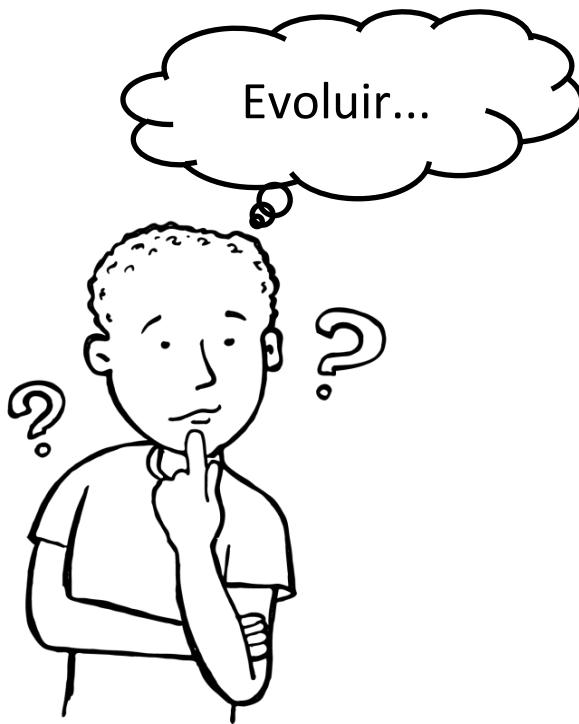


## Etapa 3

# Modelagem de dados Complexa com EER – Enhanced Entity-Relationship

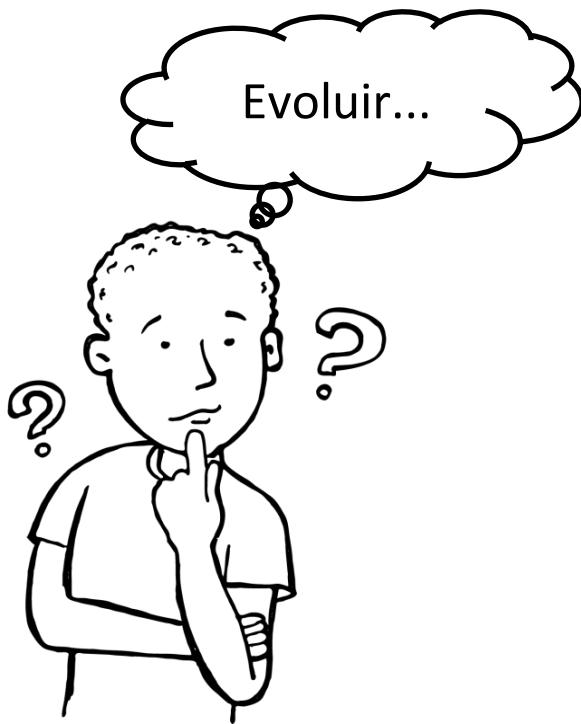
// Design e Projeto de Banco de dados

# Modelo Enhanced ER



- Modelo ER – 1970

# Modelo Enhanced ER



- Modelo ER – 1970
- EER: Novos conceitos semânticos
- Desenvolvidos fora da área de BDs
- Diagrama ERR

Poo



# Classes superclasses e Herança

# Modelo Enhanced ER

Superclasses

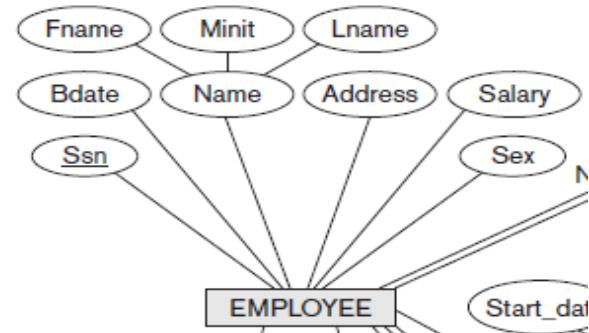
Herança

Subclasses

- Generalização e Especialização
- Categoria ou Union Type
- Herança: Atributo e relacionamento

# Modelo Enhanced ER

- Tipos de entidades
- Conjuntos de entidades



Subclasse & Subtipo

# Modelo Enhanced ER

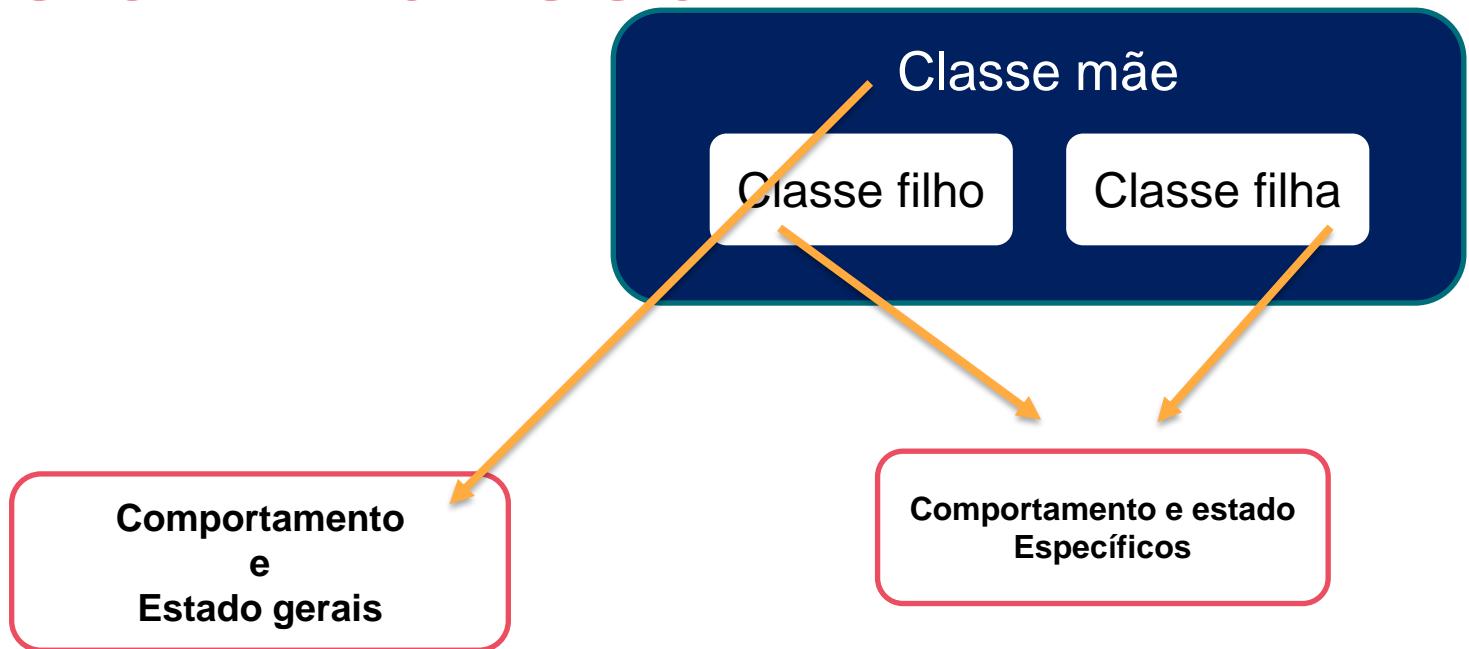
- Tipos de entidades
- Conjuntos de entidades



# Modelo Enhanced ER



# Modelo Enhanced ER



Subclasse & Subtipo

# Modelo Enhanced ER

Subclasse

- Depende do contexto do BD
- Objeto distinto no BD



# Modelo Enhanced ER

## Herança

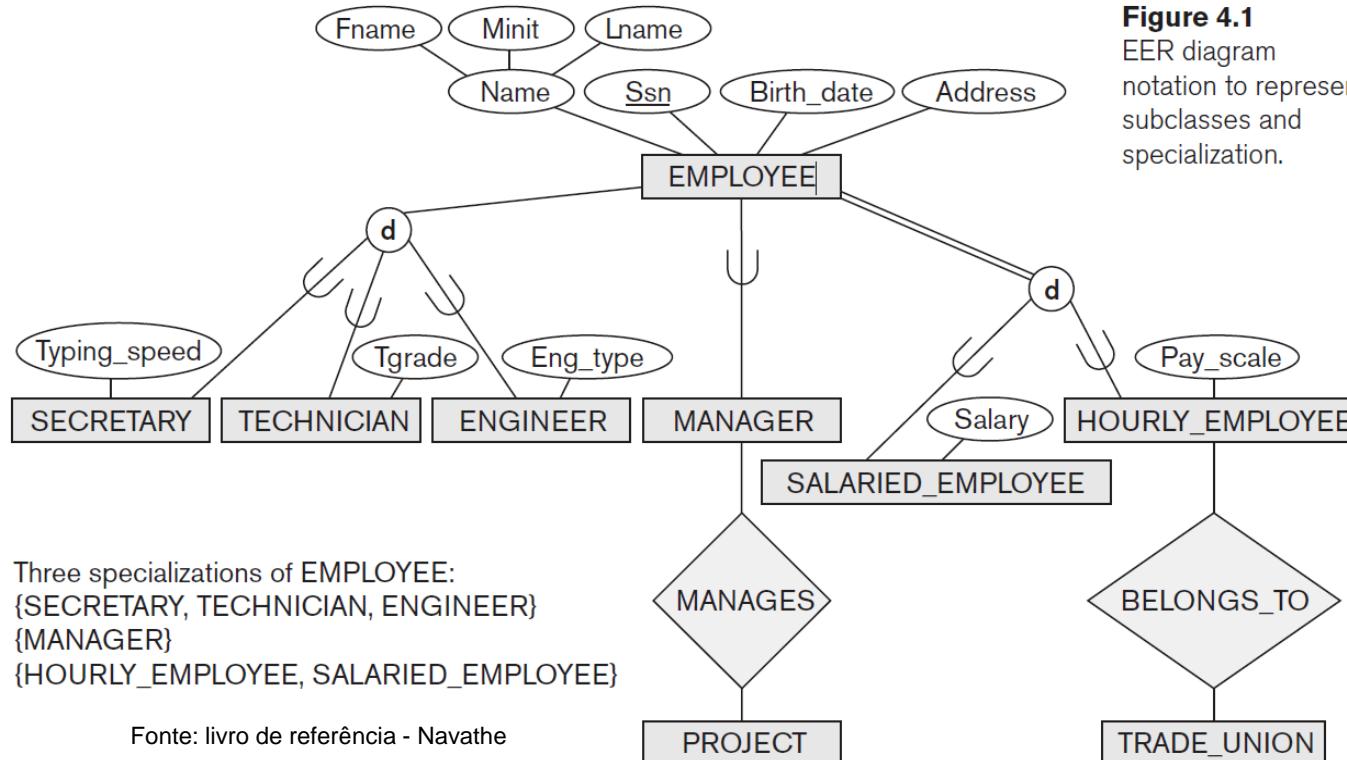
"Princípio próprio à programação orientada a objetos (POO) que permite criar uma nova classe a partir de uma já existente."

Reutilização de código

Agregar atributos e métodos

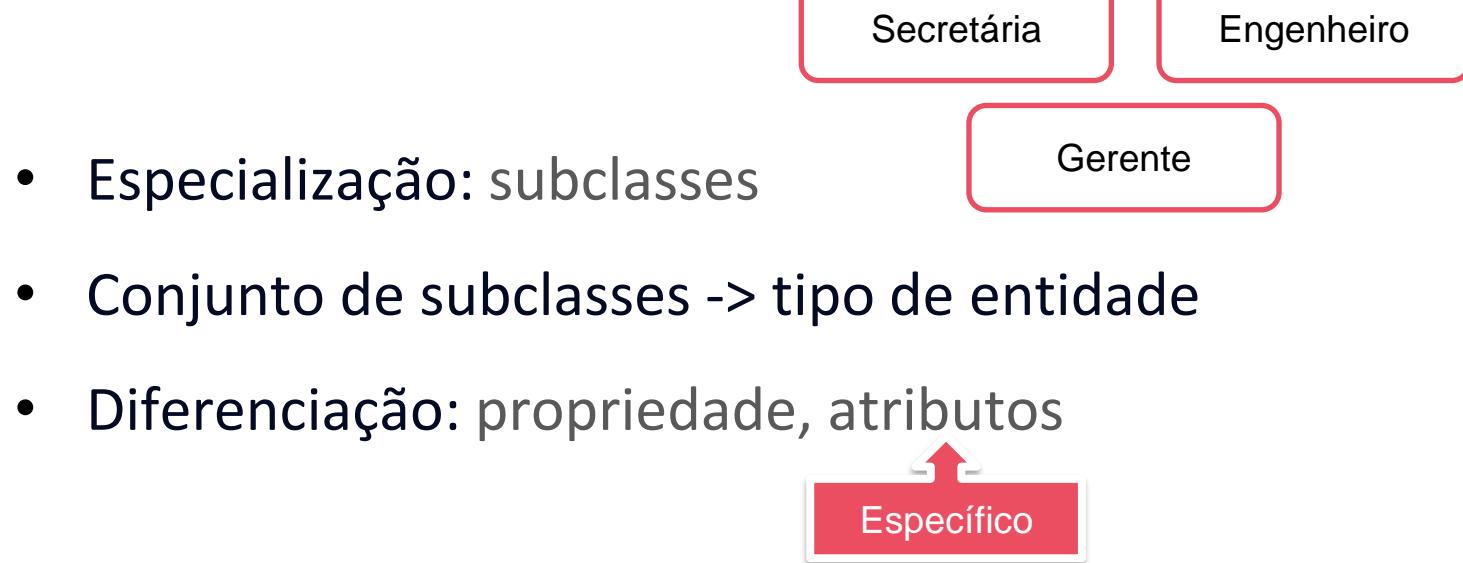
Especialização de classes

# Modelo Enhanced ER



# Especialização e Generalização

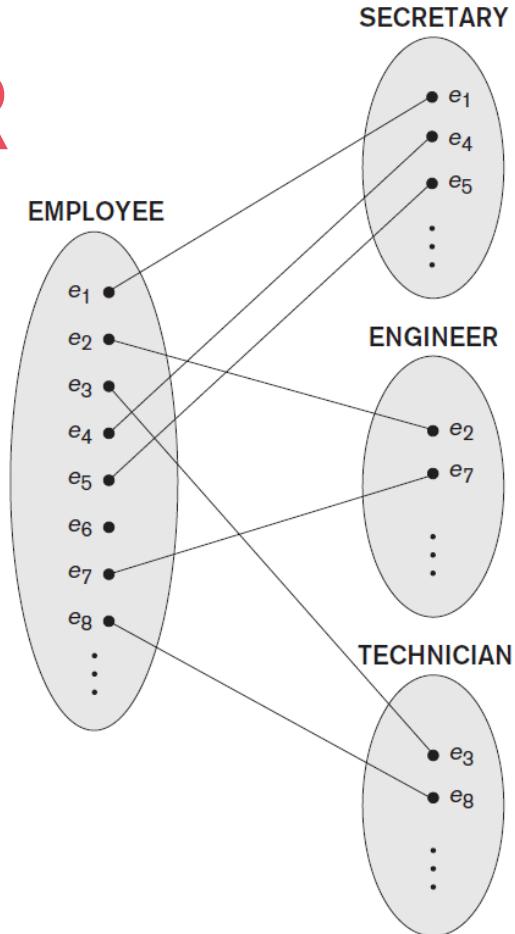
# Modelo Enhanced ER



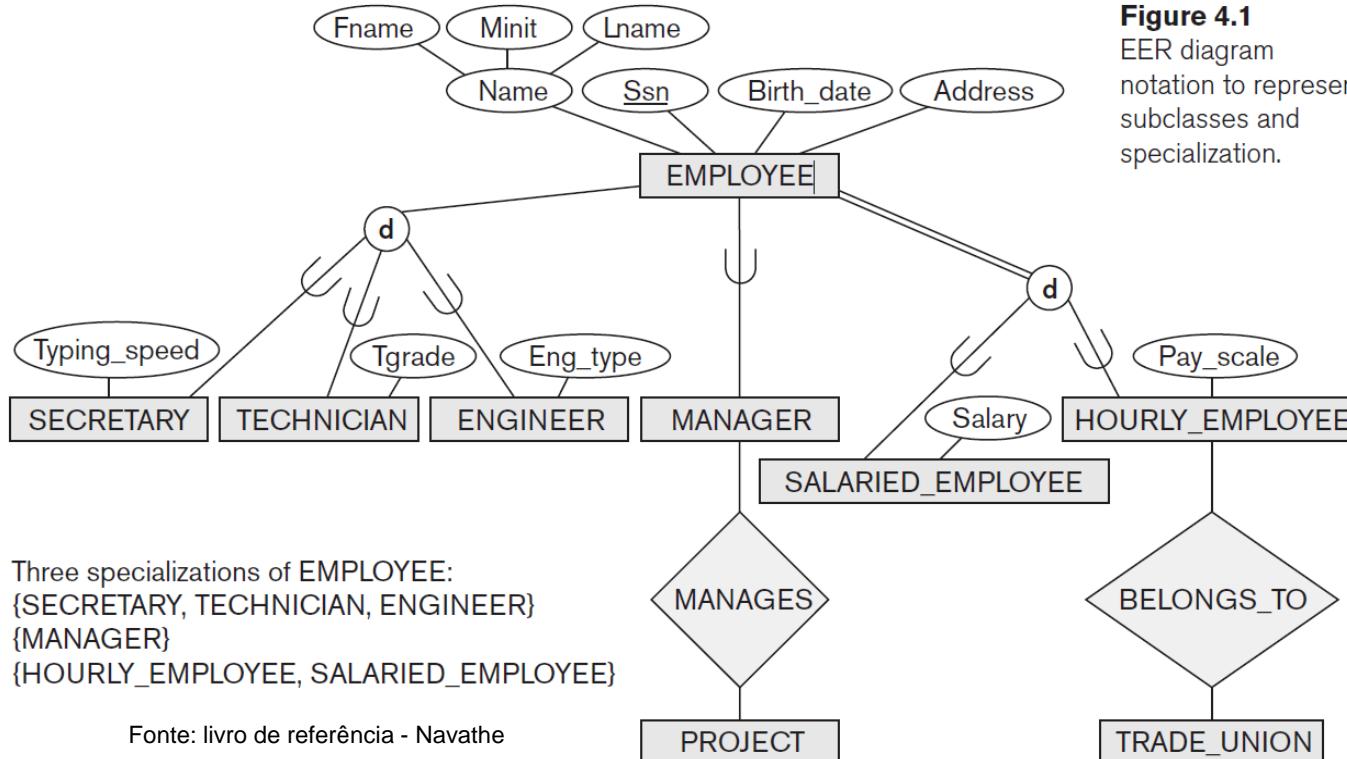
# Modelo Enhanced ER

- Instâncias das superclasse e subclasses
- Mesma representação de mundo real
- Papel especializado

[1:1]

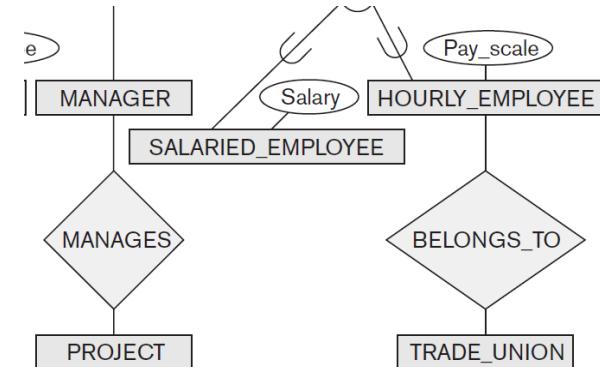


# Modelo Enhanced ER



# Modelo Enhanced ER

- Atributos aplicados a algumas entidades
- Relacionamentos específicos com subclasses



Fonte: livro de referência - Navathe

Motivação

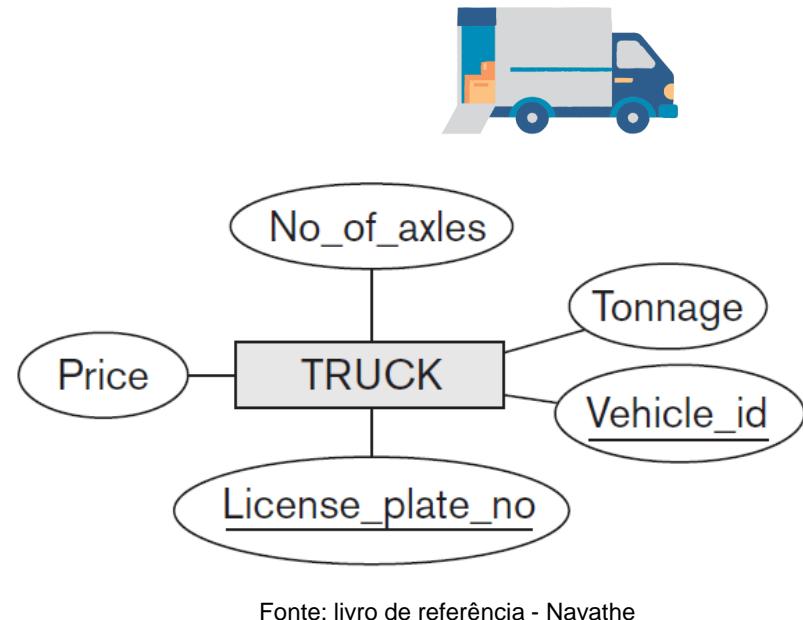
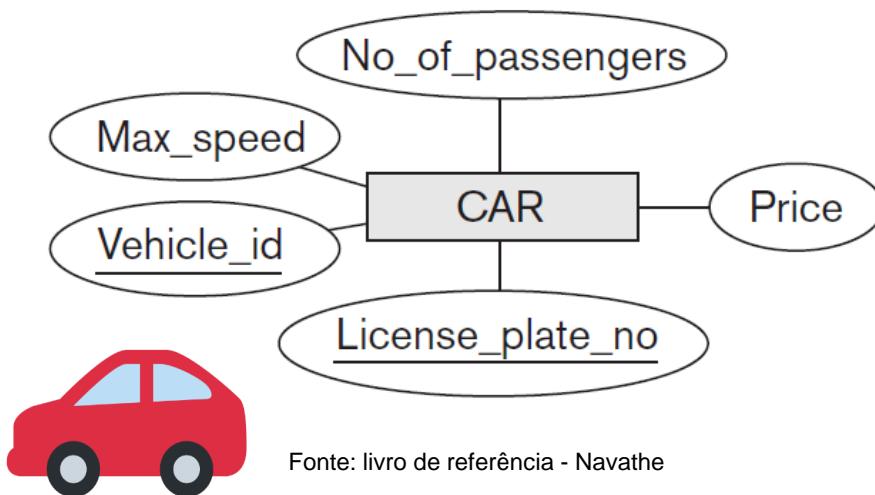
# Modelo Enhanced ER

## Generalização

- Inverso da especialização
- Propriedades comuns as entidades

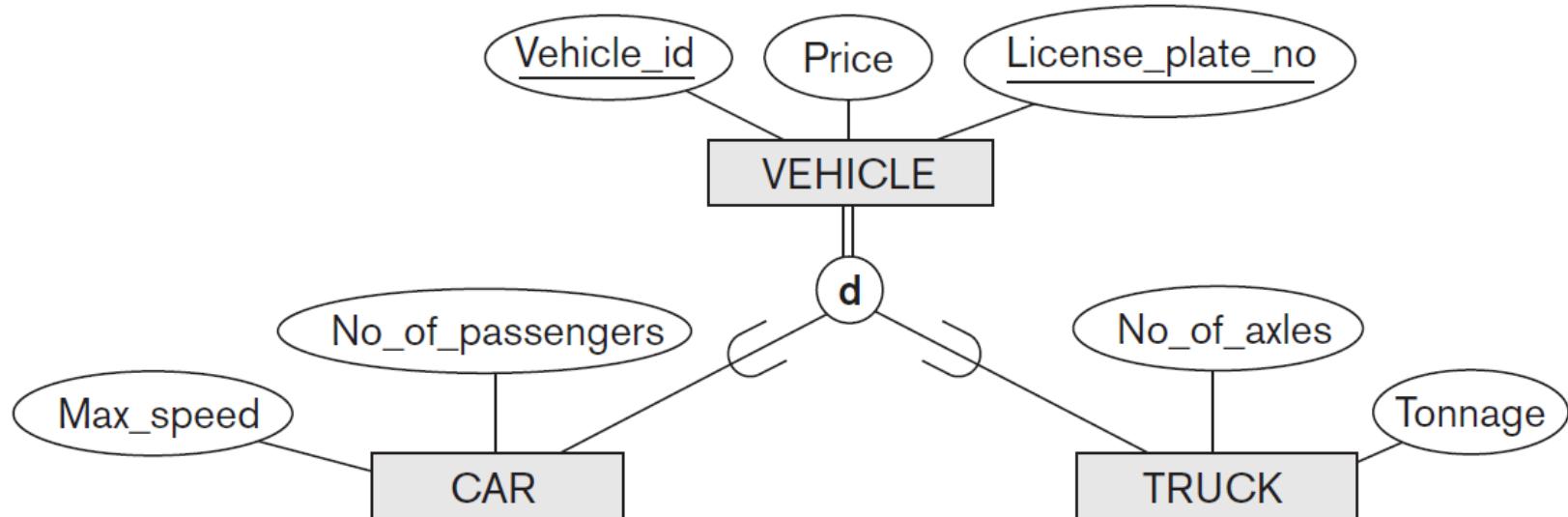


# Modelo Enhanced ER



Generalização

# Modelo Enhanced ER

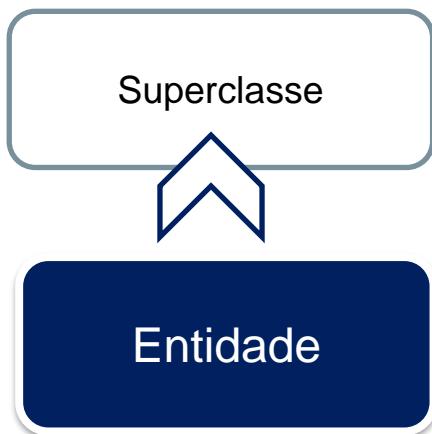


Fonte: livro de referência - Navathe

Generalização

# Refinamento

Generalização



Especialização



# Constraints de Generalização & Especialização

# Constraints – Modelo EER

Constraint:

- Predicated-defined (condição)
- Attribute-defined Specialization
- User-defined

# Constraints – Modelo EER

Constraint:

- **Predicated-defined (condição)**
- Attribute-defined Specialization
- User-defined

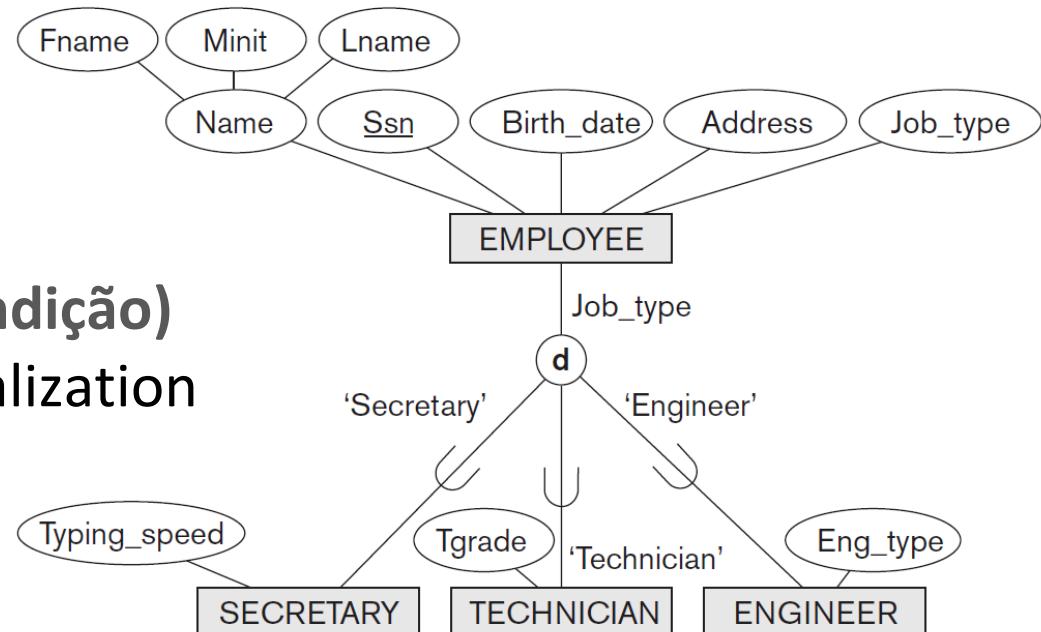
Secretária

(Job\_Type = 'Secretária')

# Constraints – Modelo EER

Constraint:

- **Predicated-defined (condição)**
- Attribute-defined Specialization
- User-defined



# Constraints – Modelo EER

Constraint:

- Predicated-defined (condição)
- **Attribute-defined Specialization**
- User-defined

Especialização definida por atributo

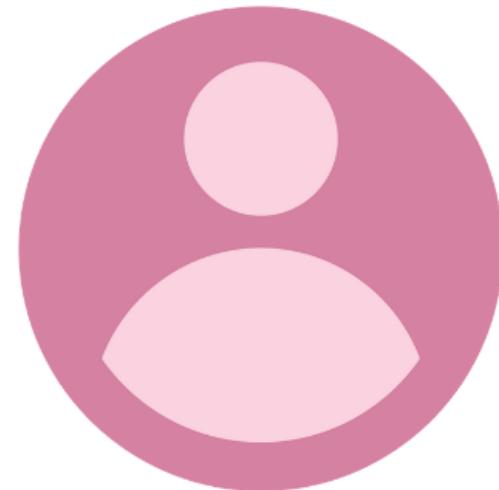


Atributo definidor da especialização

# Constraints – Modelo EER

Constraint:

- Predicated-defined (condição)
- Attribute-defined Specialization
- **User-defined**



# Constraints – Modelo EER

Constraint:

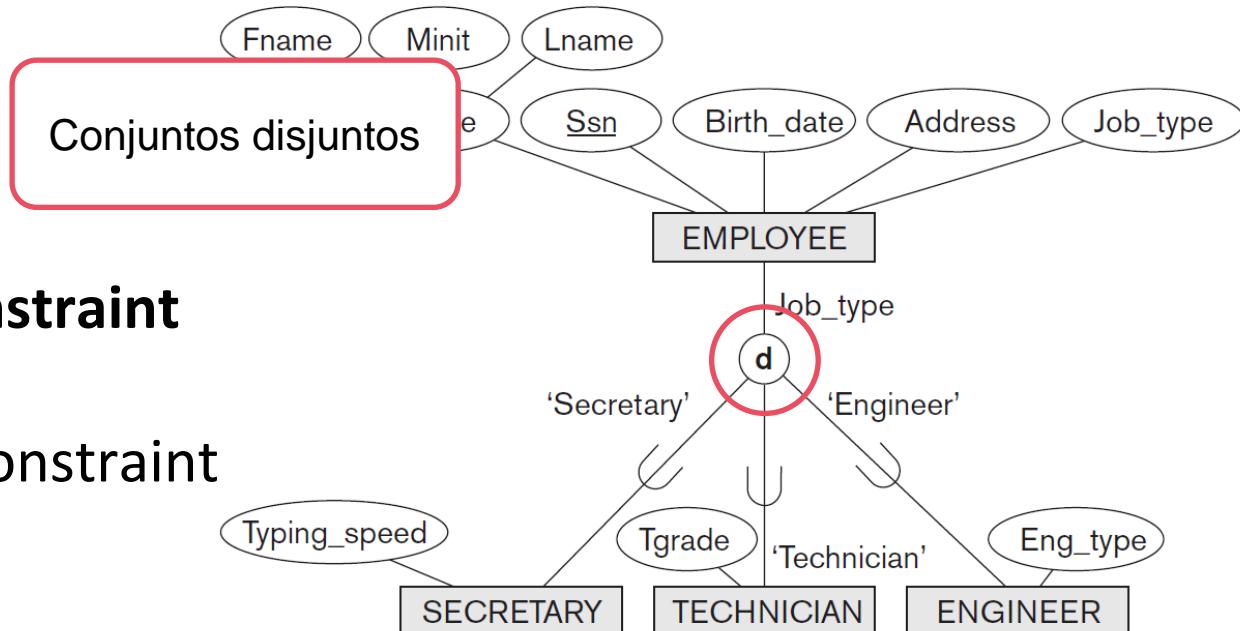
- Disjointness constraint
- Overlapping
- Completeness constraint

ESPECIALIZAÇÃO

# Constraints – Modelo EER

Constraint:

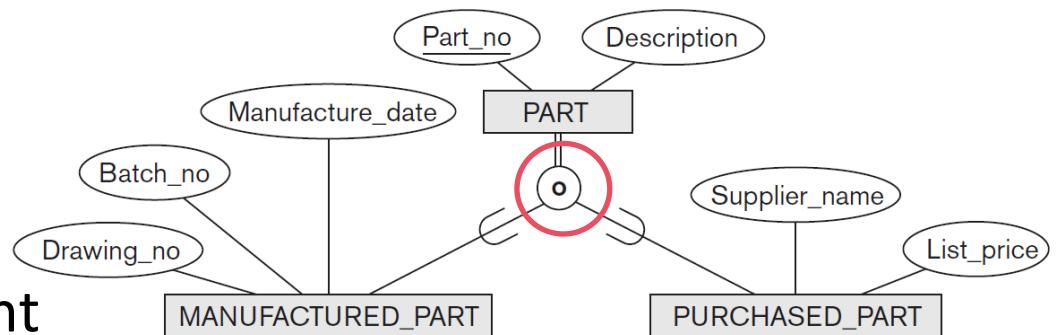
- **Disjointness constraint**
- Overlapping
- Completeness constraint



# Constraints – Modelo EER

Constraint:

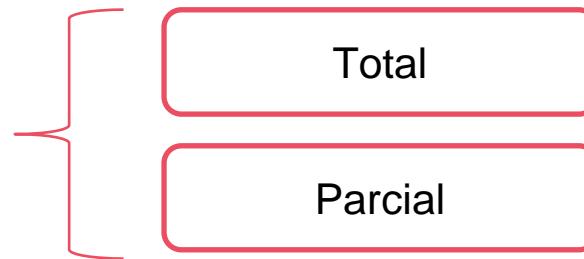
- Disjointness constraint
- Overlapping
- Completeness constraint



# Constraints – Modelo EER

Constraint:

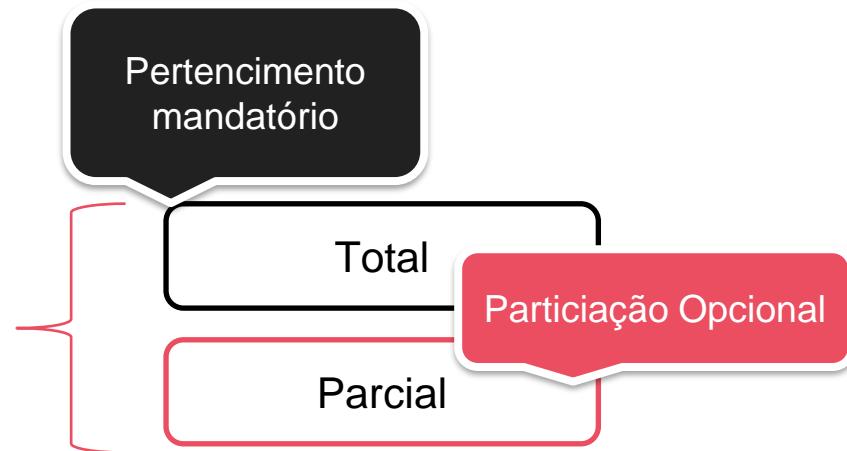
- Disjointness constraint
- Overlapping
- **Completeness constraint**



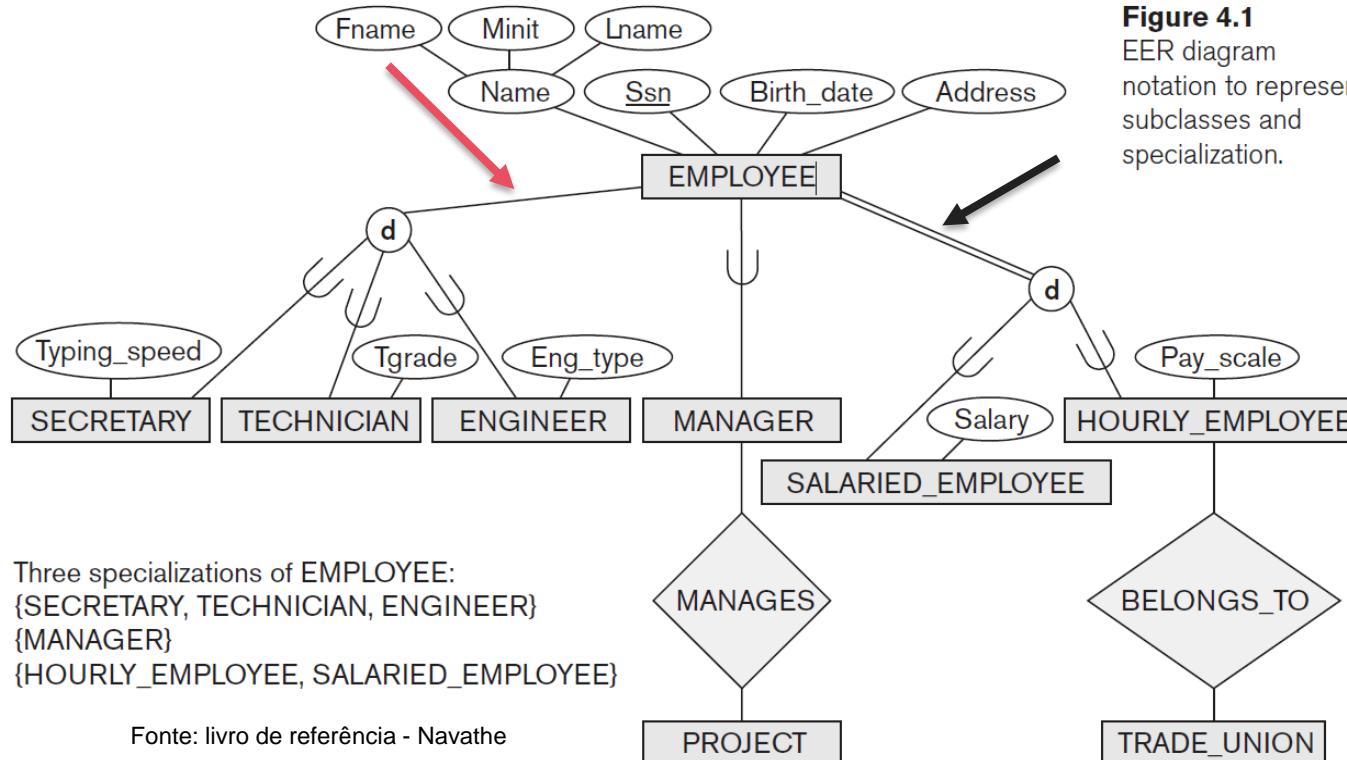
# Constraints – Modelo EER

Constraint:

- Disjointness constraint
- Overlapping
- **Completeness constraint**



# Modelo Enhanced ER



**Figure 4.1**  
EER diagram  
notation to represent  
subclasses and  
specialization.

Three specializations of EMPLOYEE:  
{SECRETARY, TECHNICIAN, ENGINEER}  
{MANAGER}  
{HOURLY EMPLOYEE, SALARIED EMPLOYEE}

Fonte: livro de referência - Navathé

[228]

# Constraints – Regras

Deletar

- Superclasse -> subclasses

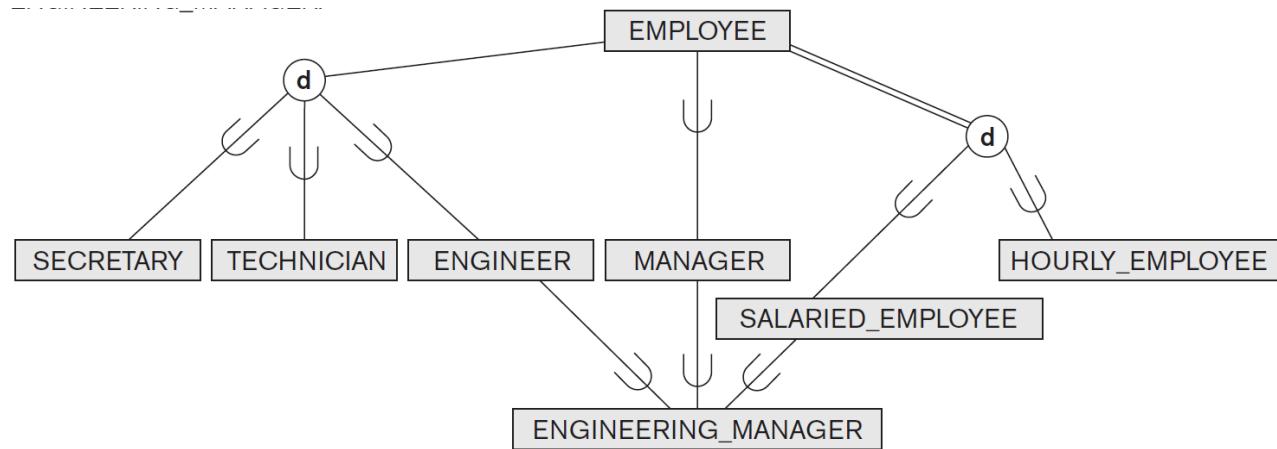
Inserção

- Entidade em superclasse -> predicated-defined
- Entidade em superclasse com total -> subclasse macth

# Hierárquia & rede de especialização

# Especialização

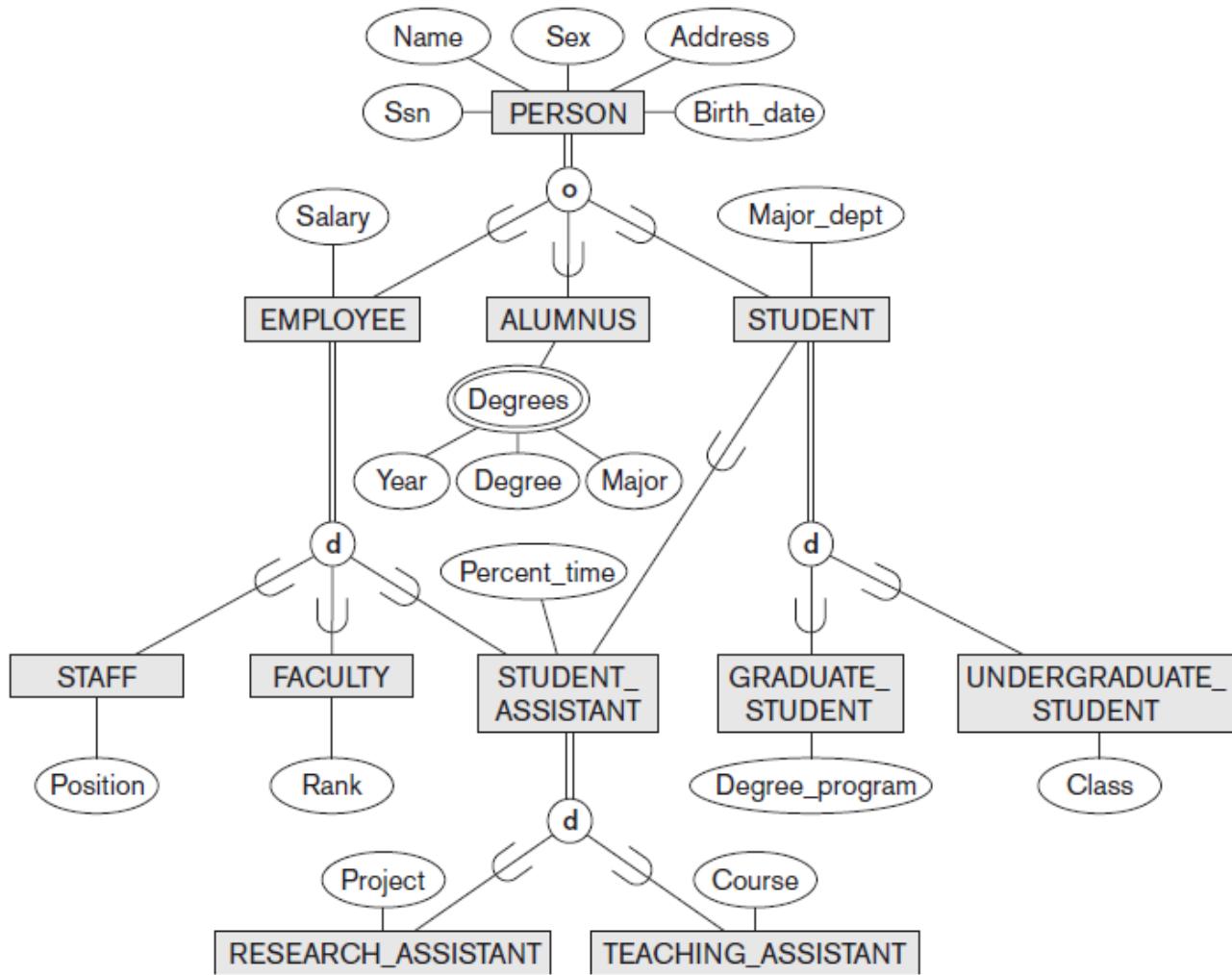
- Hierárquia de especialização
- Rede de especialização

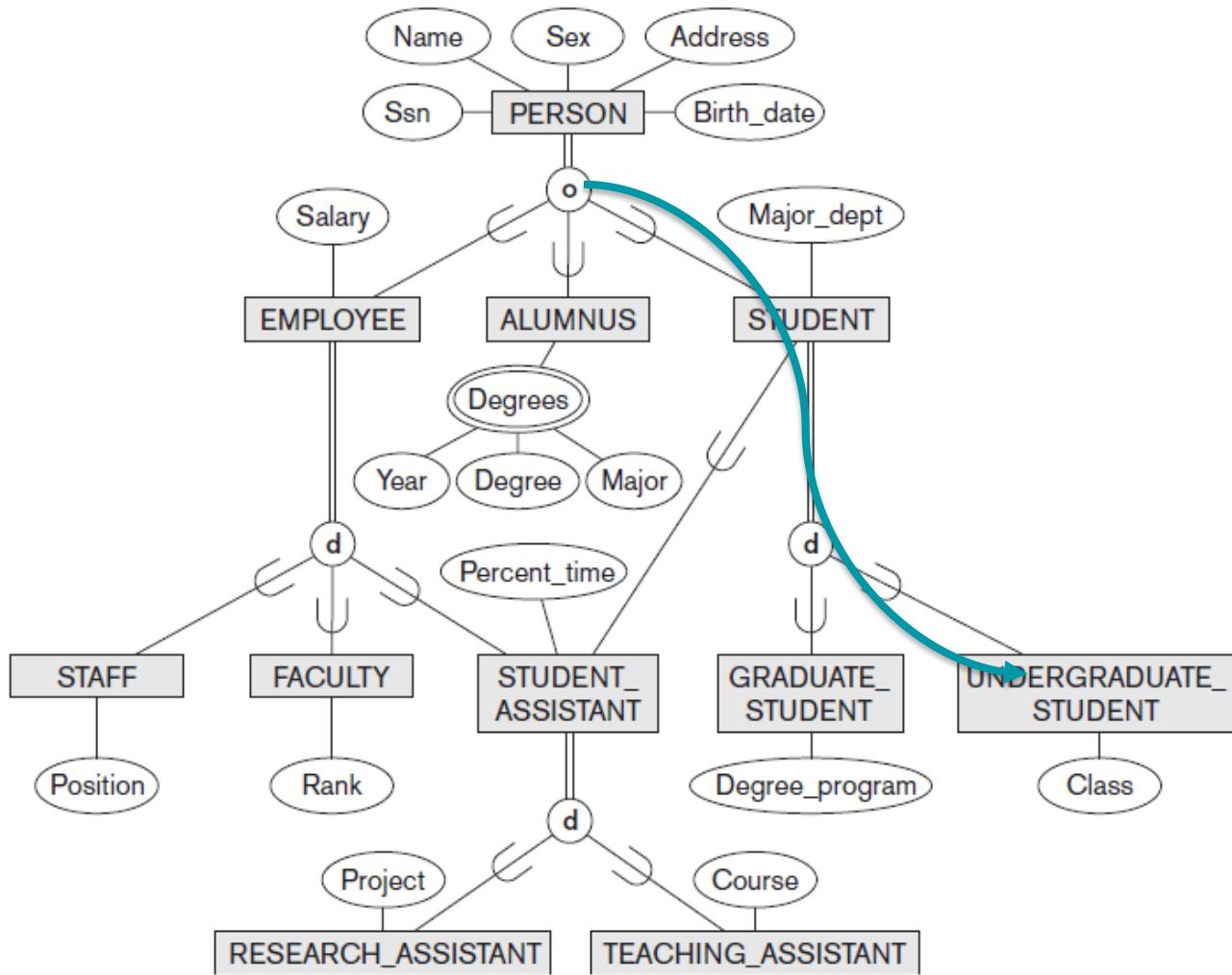


# Exemplo

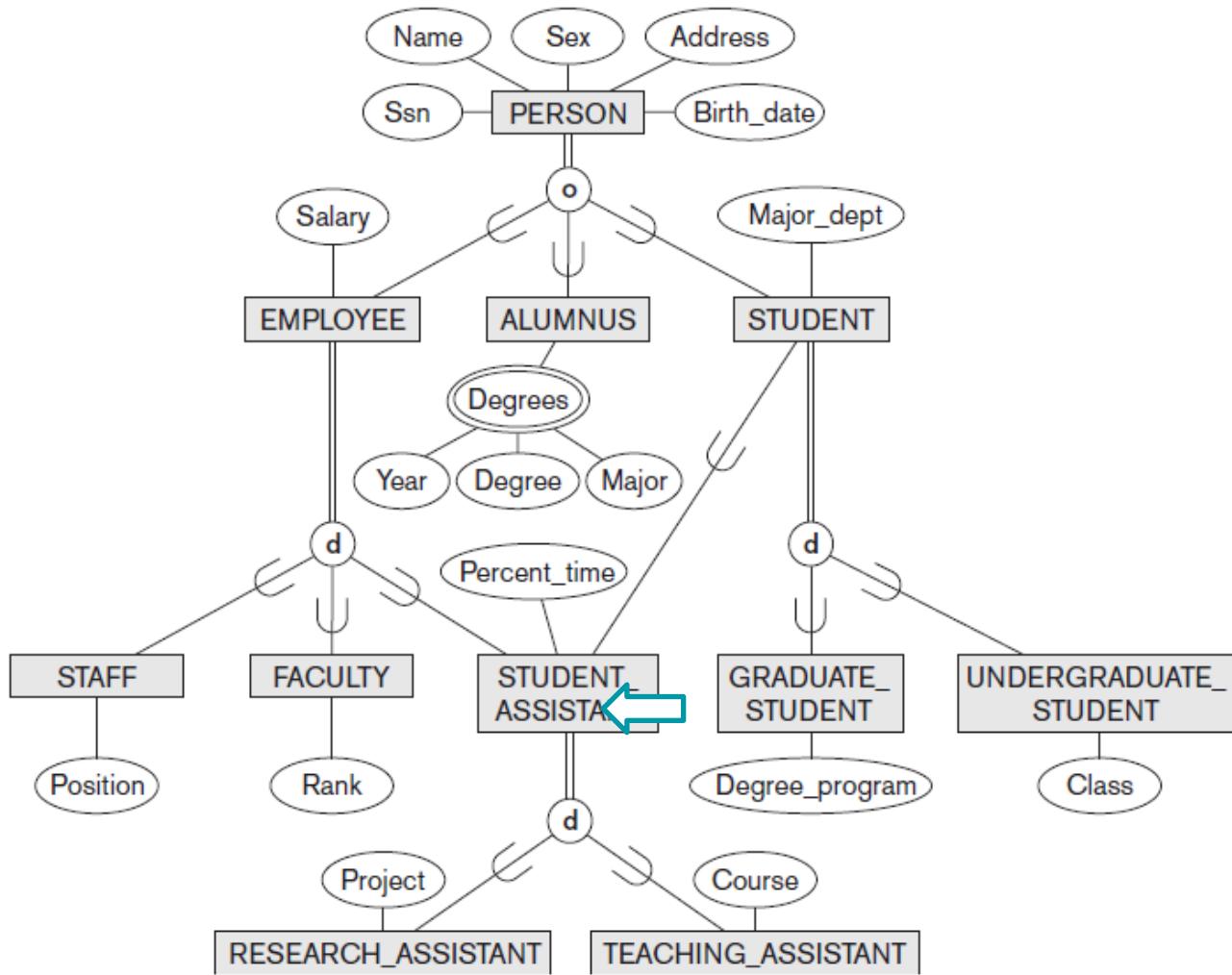
## Requisitos

- Rastreamento das 3 entidades & overlapping de entidades
- Tipos definidos de empregados
- Informação dos aluminis – grau acadêmico
- Estudantes graduados ou não





Herança



Herança múltipla

Subclasse  
compartilhada

# Modelagem de union types usando categorias

# Union Type

- Coleção de objetos de diferentes tipos de entidades
- Representação: subclasse



# Union Type

Subclasse:

- Papel dono do veículo

Dona do carro



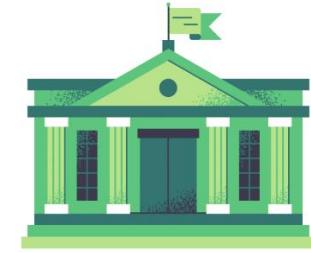
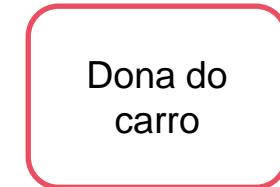
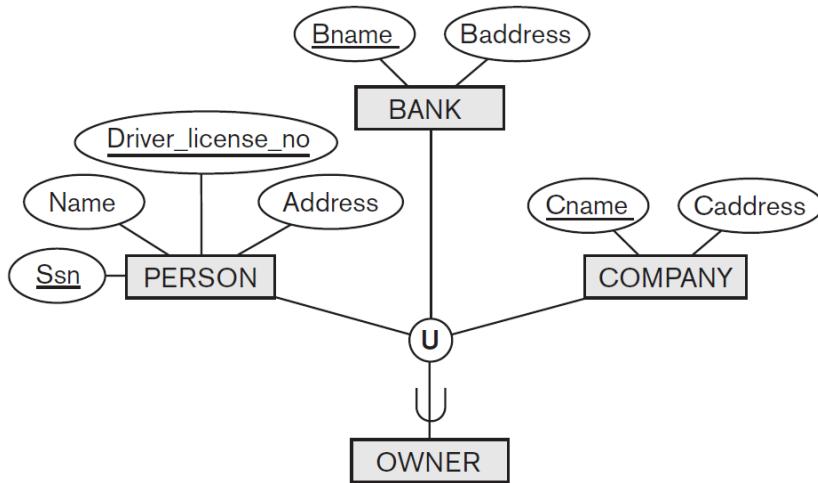
Fiancamento



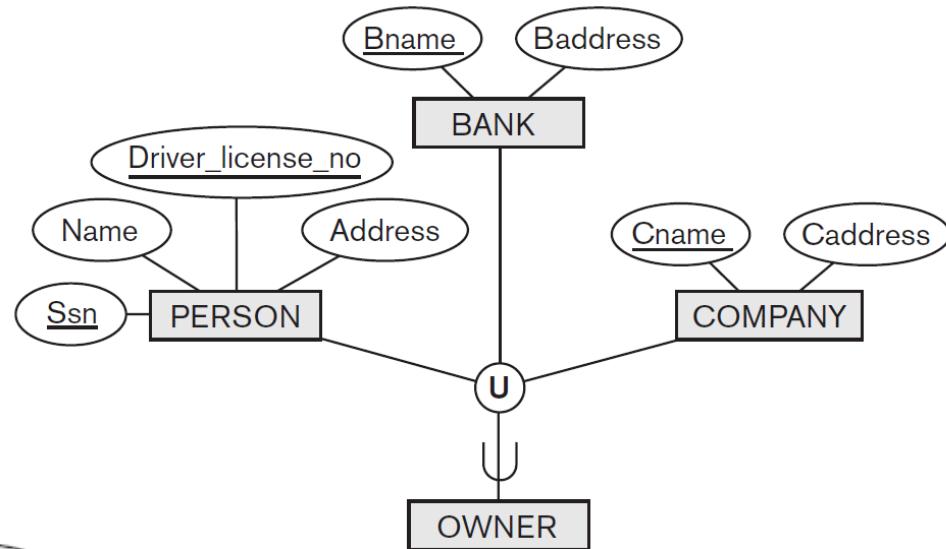
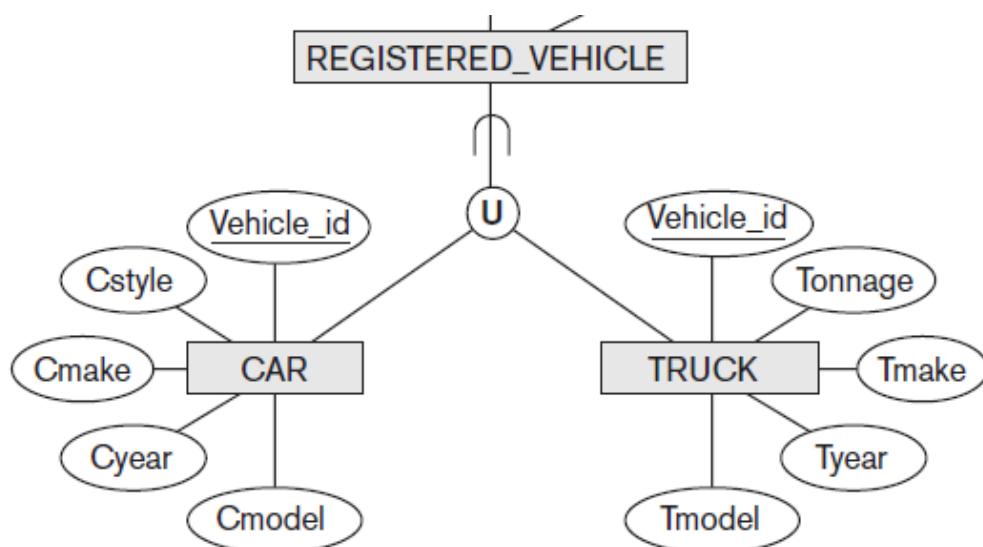
# Union Type

Subclasse:

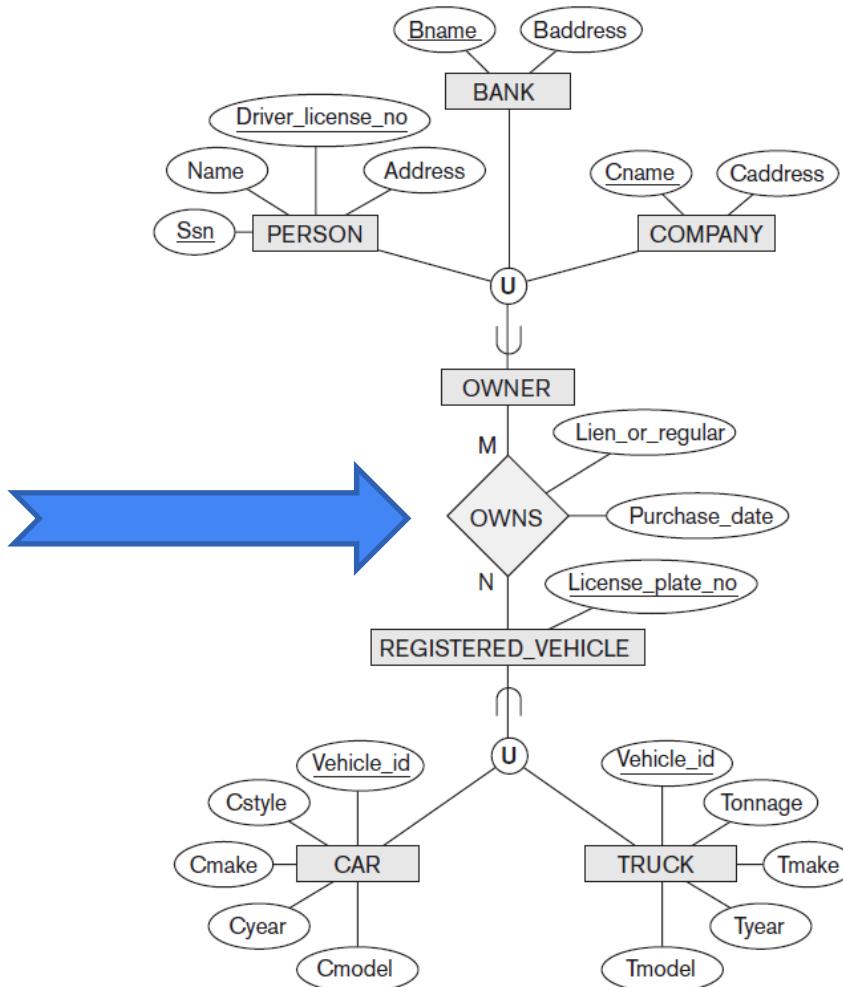
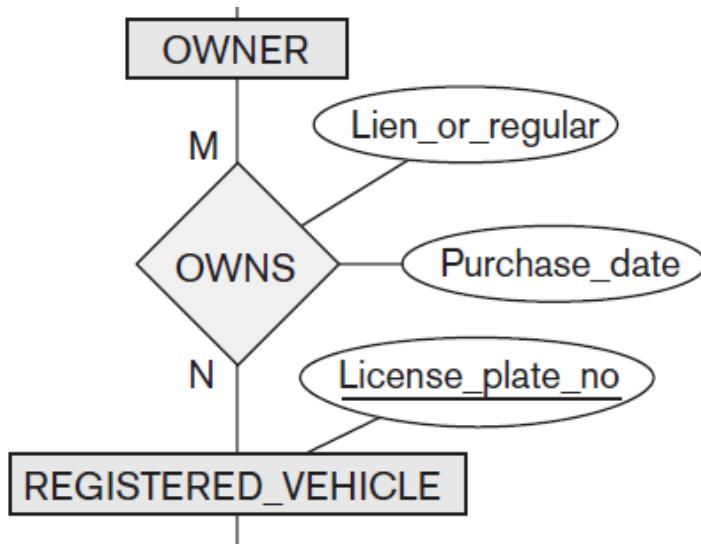
- Papel dono do veículo



# Union Type



# Union Type



# Union Type

Subclasse:

- Papel dono do veículo

Union Type



- Parcial (II)
- Total (I)

Dona do carro



Financiamento



# Union Type



Generalização/Especialização



Union Type/Categorização



# Esquema EER & Definições formais

# Exemplificação

- Aplicação do Modelo Enhancer ER
- Descrição do esquema
- Definições formais
- Reafirmando conceitos



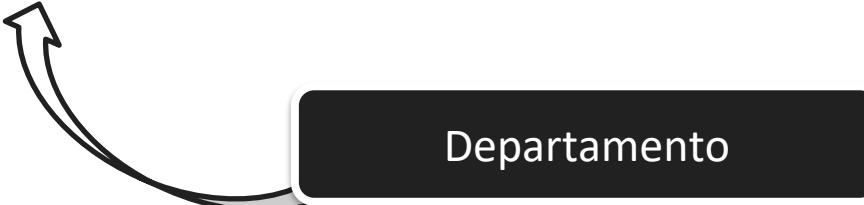
# Exemplificação

Rastrear:

- Estudantes, graduados, registros, trabalhos
- Oferta de cursos

Pessoas:

- Faculdade - rank(assitente, associado, pesquisador ...)
- Estudante

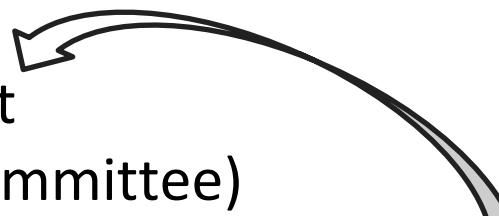


Departamento

# Exemplificação

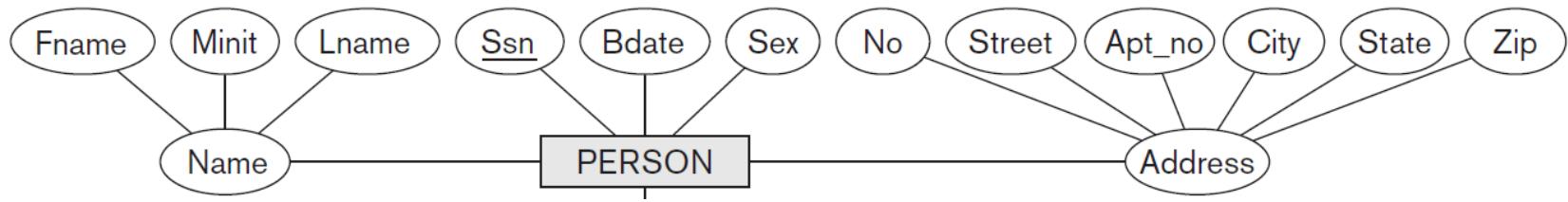
Estudante:

- Class -> grau acadêmico
- Relacionados ao seu minor e major
- Trabalho final deve ser registrado
- Class = 5 ou 6 para ser Grad\_Student
- Entrega a tese para o orientador (committee)



Constraint

# Exemplificação



Pessoas

# Exemplificação

Departamento:

- Uma pessoa acadêmica pertence a um departamento
- Pessoa coordenador – chairperson
- Atributos: nome, telefone, escritório ...



Constraint

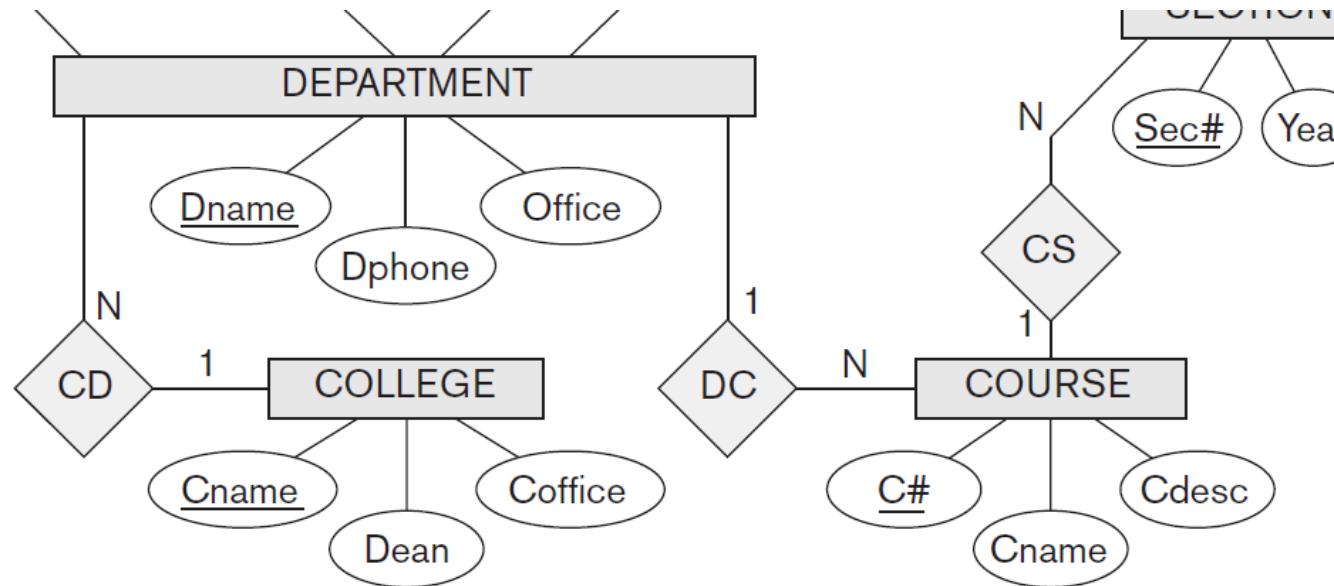
# Exemplificação

Curso:

- Atributos: nome, número, descrição
- Seções do mesmo curso são ofertadas aos alunos
- As seções são ofertadas a cada ano e possui um instrutor

# Exemplificação

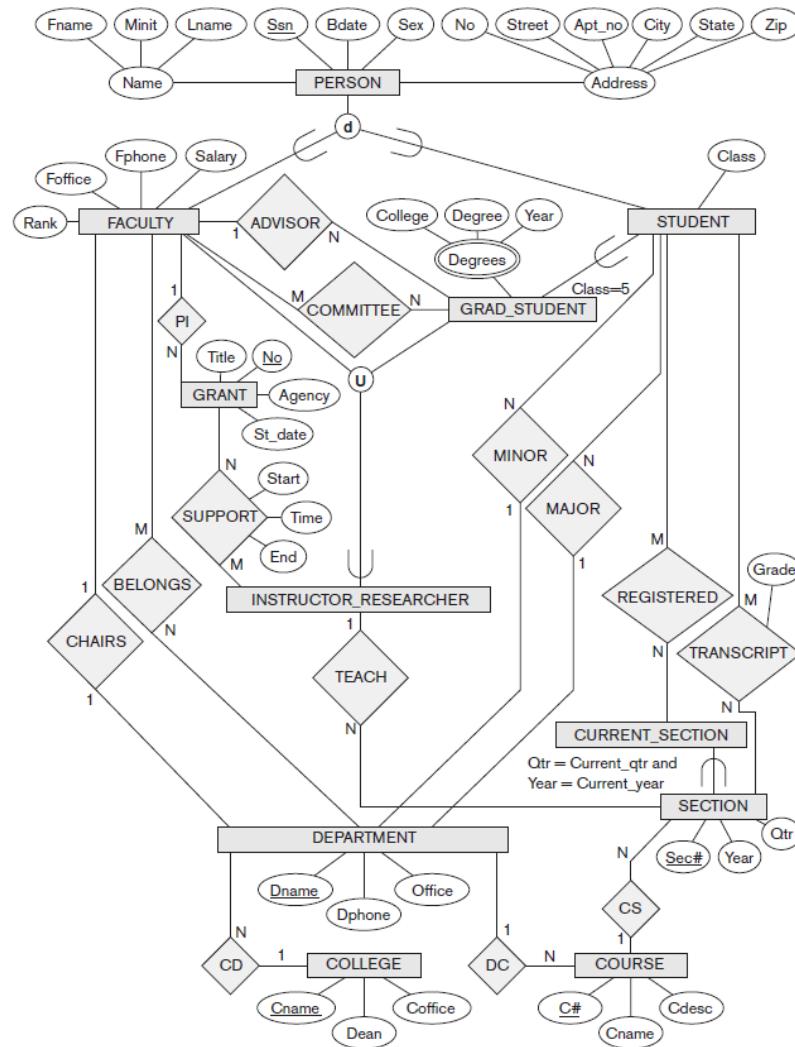
Departamento



# Exemplificação

Instrutor pesquisador:

- Agregado: Faculty (acadêmico) e Grad\_Student (Mestrando ou doutorando)
- Um instrutor pesquisador está associado a uma seção de um curso



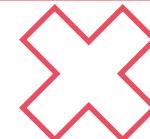
# Universidade

# Decisão sobre design de BD & UML

# Design



Generalização/Especialização



Union Type/Categorização

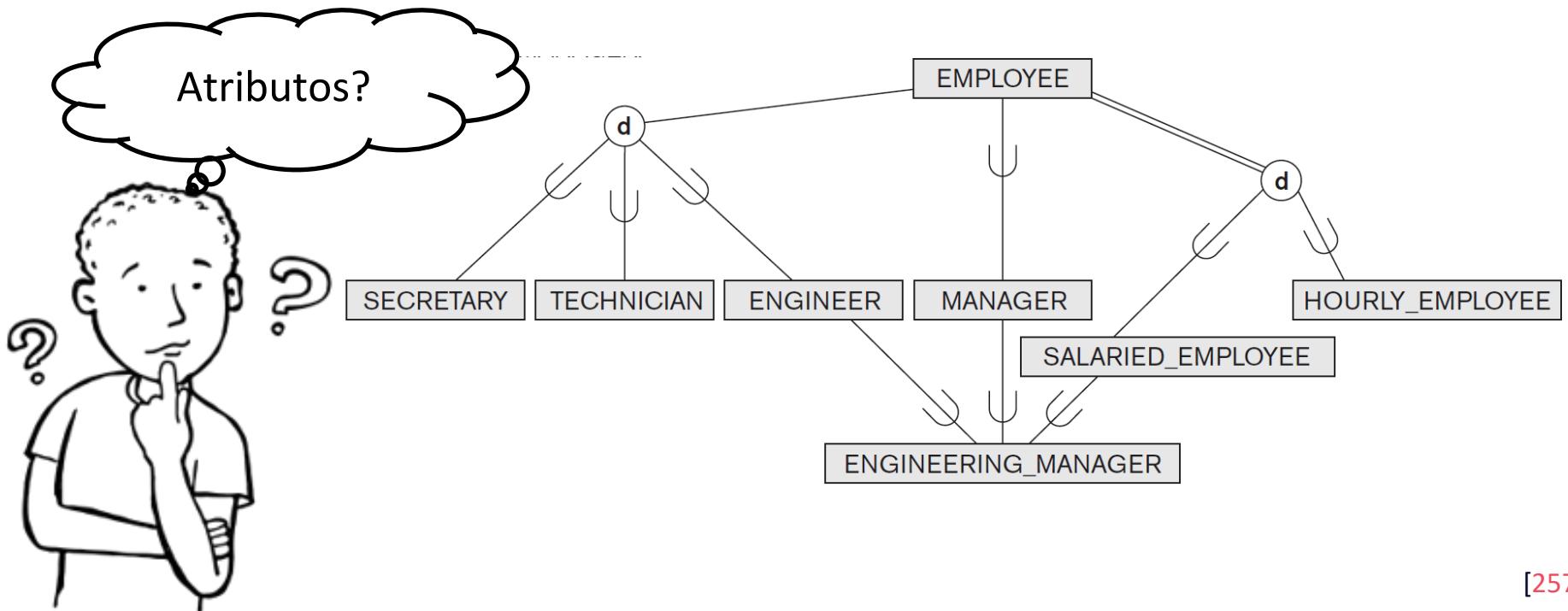


# Decisões de Design

## Guideline

- Sub/Superclasses: Acurácia x Desordem
- Merge de subclasse em uma superclasse: s/  
relacionamentos e poucos atributos
- Union Types: evitados por padrão
- Escolhas guiadas pelo mini-mundo

# Decisões de Design

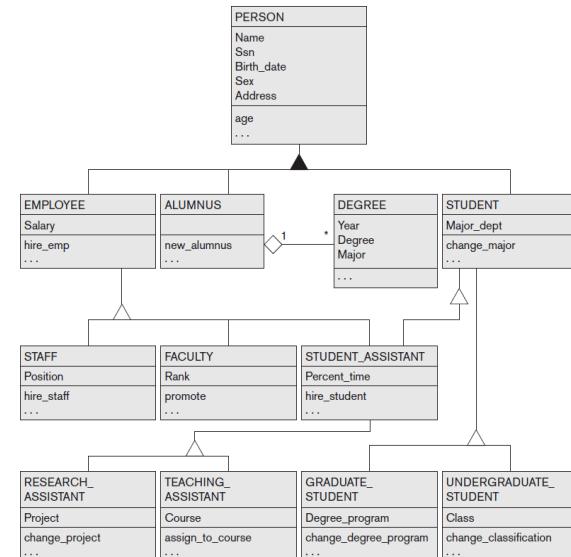


# Representando por UML

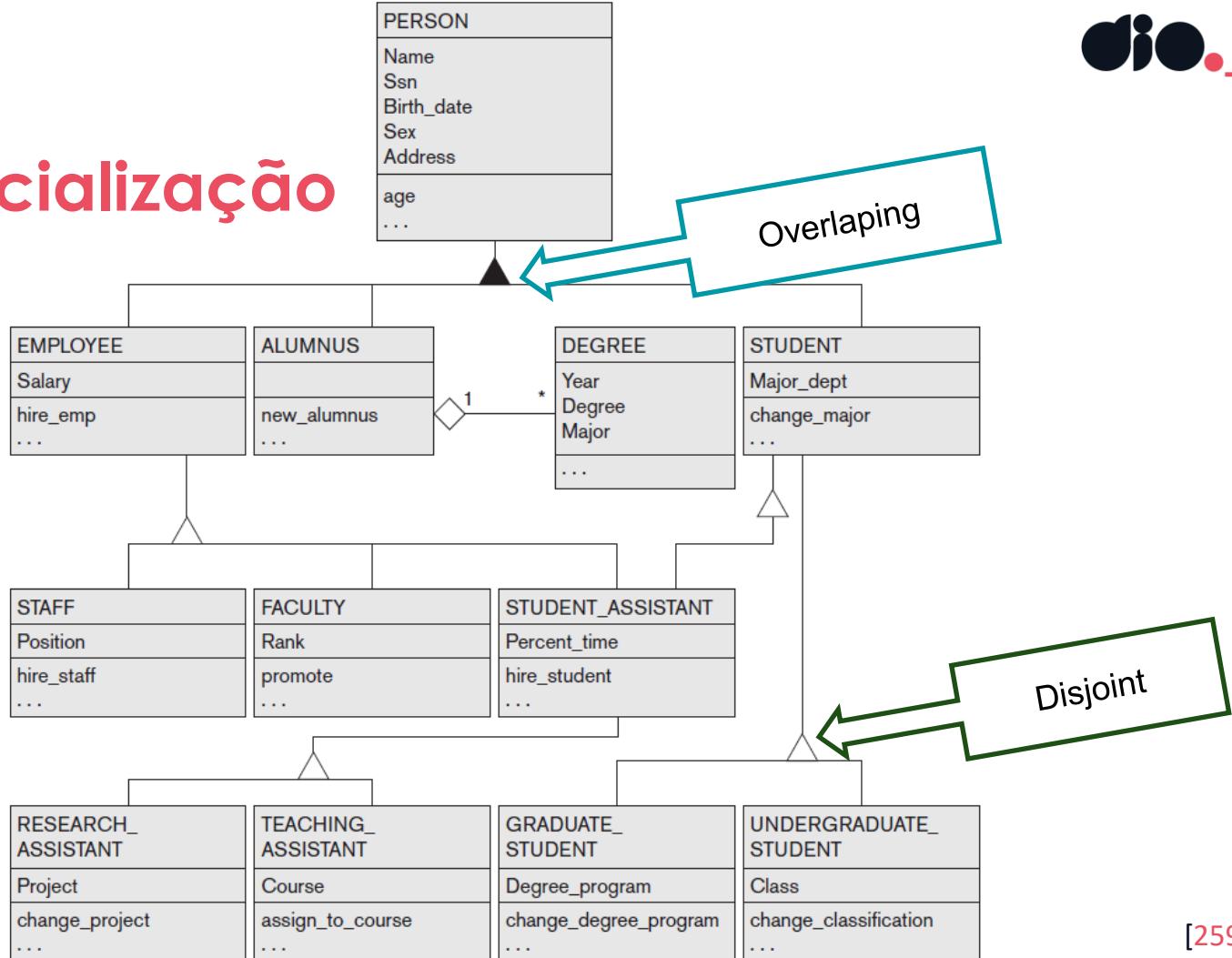
- Foco na modelagem de dados
- Diagrama de classes
- Classes concretas

## Nomenclatura

- Classe base e classes (nós) folha

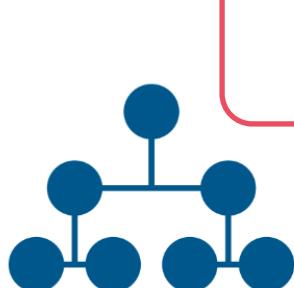


# UML - Especialização

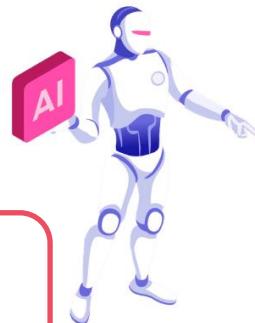


# Terminologias alternativas

# Terminologias



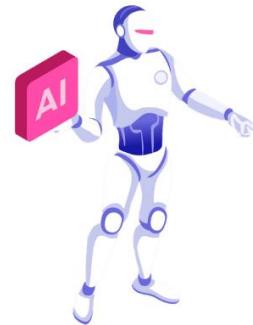
Knowledge representation (KR)



Ontologia

Domínio

# Terminologias

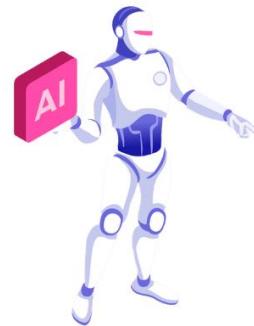
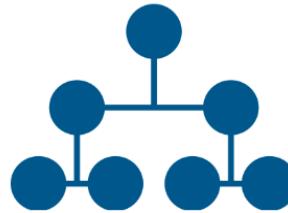


Domínio

## Similaridades

- Processo de abstração para identificação de componentes
- Prove recursos para modelagem de dados

# Terminologias



Domínio

KR

- Modelagem semântica - mais abrangente: regras, conhecimento espacial e temporal
- Reasoning mechanisms
- Mistura de instância e esquema

# Terminologias

## Conceitos abstratos

- Classificação e Instanciação
- Identificação
- Especialização e Generalização
- Agregação e Associação

Modelo de dados  
semântico

Propriedade de classes

Objetos de exceção

Meta-class

Conceitos abstratos

- Classificação e Instanciação
- Identificação
- Especialização e Generalização
- Agregação e Associação

Atribuição de  
objetos/entidades a  
classes/tipo de entidade

# Terminologias

## Conceitos abstratos

- Classificação e Instanciação
- Identificação
- Especialização e Generalização
- Agregação e Associação



Processo abstrato de identificação única

# Terminologias

## Conceitos abstratos

- Classificação e Instanciação
- Identificação
- Especialização e Generalização
- Agregação e Associação

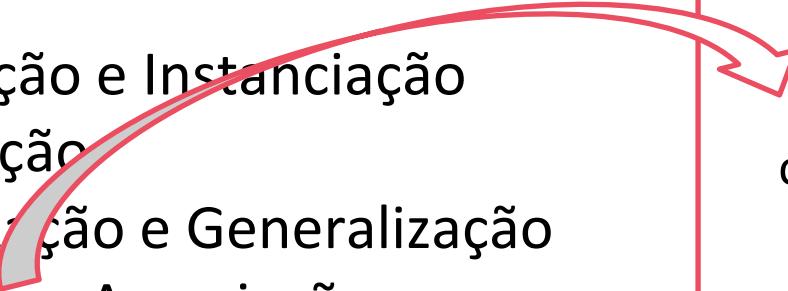


Refinamento utilizando  
sub e superclasses

# Terminologias

## Conceitos abstratos

- Classificação e Instanciação
- Identificação
- Especialização e Generalização
- Agregação e Associação



Abstração para  
composição de objetos  
complexos

# Ontologia

Alimenta o BD com informações sobre o domínio, suas propriedades e relações

Semantic Web



Close World Assumption



Open World Assumption

# Ontologia

Descrição:

- Thesaurus – relacionamentos x conceitos
- Taxonomy – relacionamento de conceitos com estruturas
- Esquema detalhado – descrição de conceitos
- Lógica teórica – lógica matemática para definição de conceitos

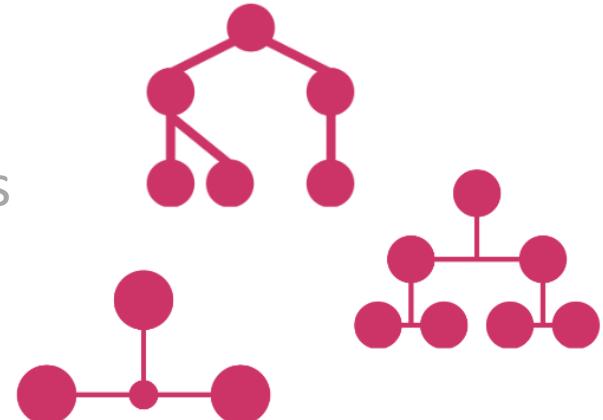
## Etapa 4

# Hands on: Construindo seu Primeiro Projeto de Banco da Dados

// Design e Projeto de Banco de dados

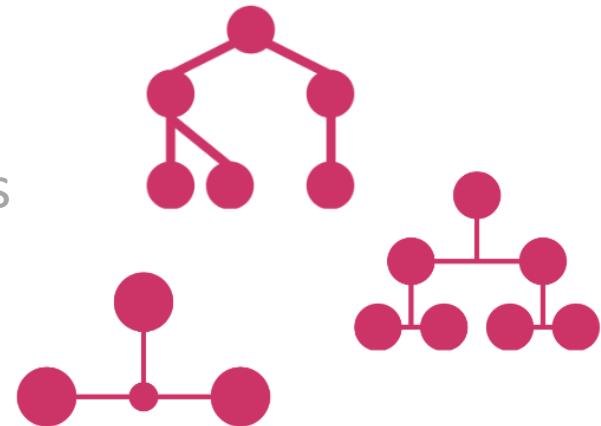
# Projeto de Banco de Dados

- **Contexto:** Levantamento de Requisitos
- **Projeto Conceitual:** Modelo ER
- **Projeto Lógico:** Modelo Relacional



# Projeto de Banco de Dados

- **Contexto:** Levantamento de Requisitos
- **Projeto Conceitual:** Modelo ER
- **Projeto Lógico:** Modelo Relacional

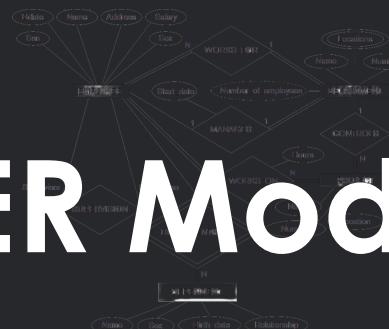


ORDEM DE SERVIÇO

UNIVERSIDADE

ECOMMERCE

# Modelando – Ordem de Serviço



# ER Model

# Projeto de Banco de Dados

## Contexto:

- Dentro de uma empresa os clientes demandam ao helpdesk algumas ações.
- Essas ações são convertidas em ordem de serviço



Ordem de serviço

# Projeto de Banco de Dados

## Contexto:

- Os clientes realizam um pedido
- O pedido é convertido em ordem de serviço caso possa ser realizado
- O técnico executa a ordem de serviço. Após sua finalização a mesma é arquivada



Ordem de serviço

# Projeto de Banco de Dados

## Entidades:

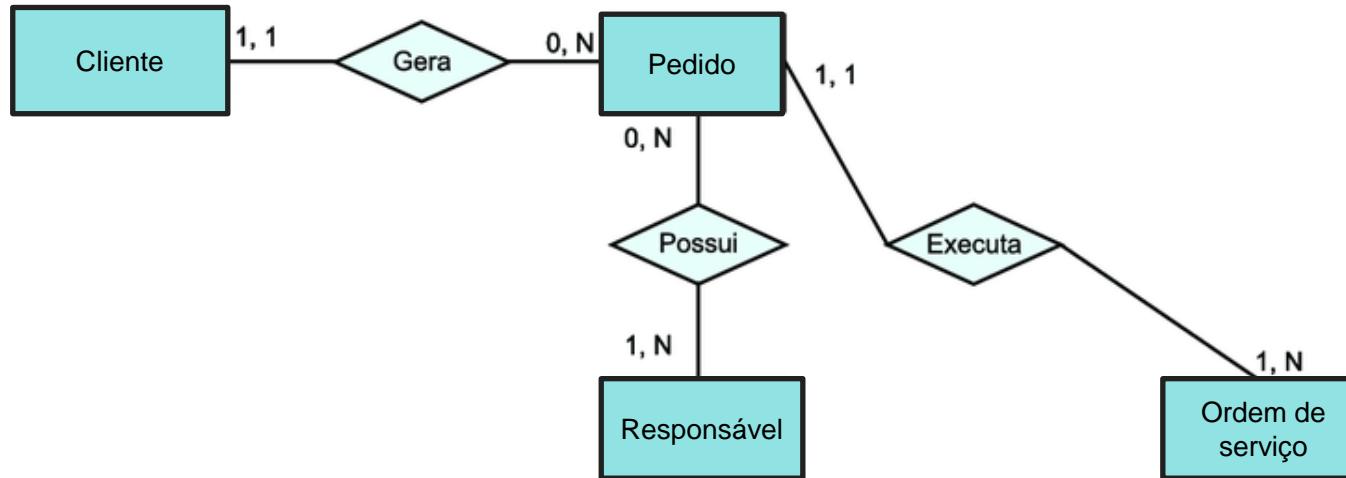
- Cliente, Responsável, Pedido e Ordem de Serviço

Ordem de serviço

## Relacionamentos

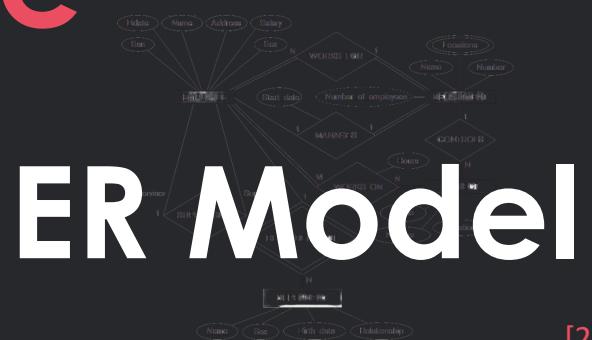
- Solicita, Analisa, Executa, Arquiva

# Projeto de Banco de Dados



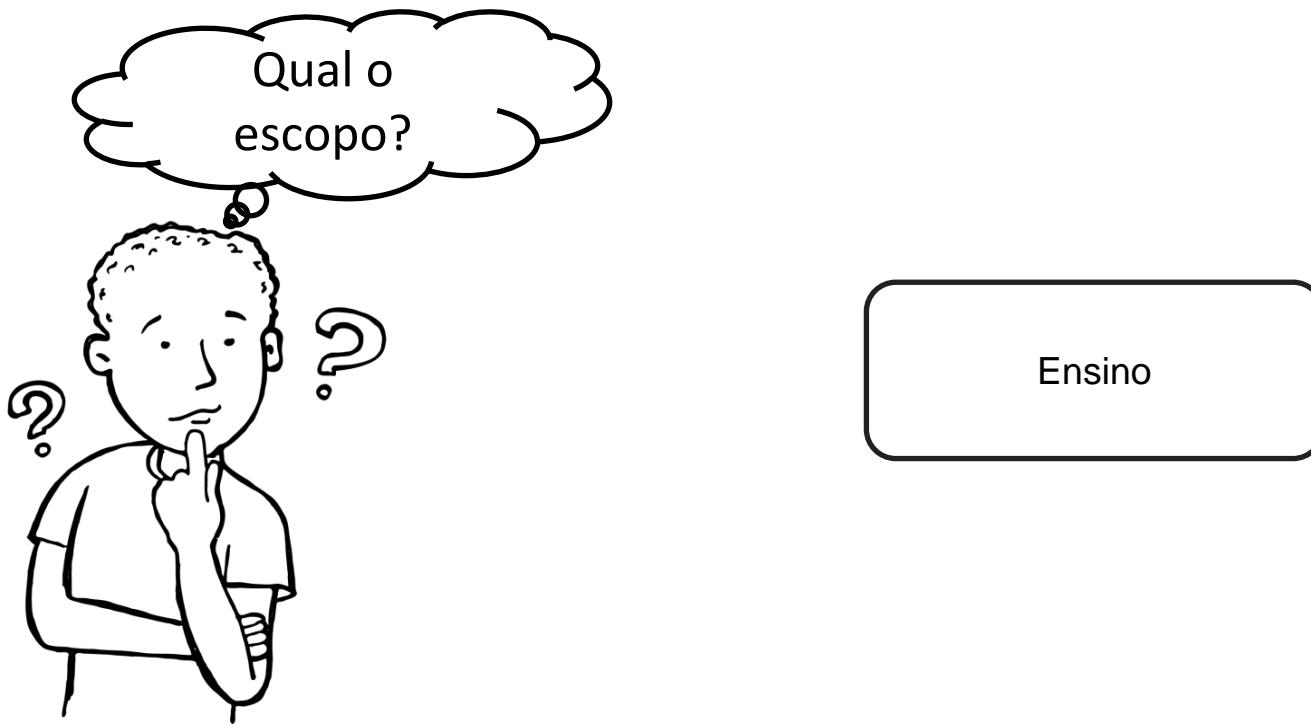
Ordem de serviço

# Modelando – Escopo de Universidade



## ER Model

# Projeto de Universidade



# Projeto de Universidade



Professor

Curso

Coordenação

Ensino

Disciplina

Aluno

# Projeto de Universidade

## Narrativa - Alunos

- A universidade possui diversos alunos que podem estar matriculados em mais de um curso (graduação).
- Os alunos podem fazer cursos extras fornecidos externa e internamente (universidade) para contar como horas complementares

Levantamento de Requisitos

# Projeto de Universidade

## Narrativa - Alunos

- Não há restrição quanto ao número de matérias puxadas se não houver sobreposição de horário.
- Os alunos são submetidos a duas provas por semestre para cada disciplina. Eventuais trabalhos devem ser tratados pelo professor para compor a nota da prova.

Levantamento de Requisitos

# Projeto de Universidade

## Narrativa - Disciplinas

- Cada disciplina é fornecida por um professor. Restrição: apenas por este professor.
- Algumas disciplinas possuem pré-requisitos. Um mesmo pré-requisito pode ser associado a mais de uma disciplina.

Levantamento de Requisitos

# Projeto de Universidade

## Narrativa - Disciplinas

- As disciplinas podem ser comuns a cursos distintos. Ex: Cálculo 1 para computação e engenharia
- O ciclo de vida da disciplina é semestral

Levantamento de Requisitos

# Projeto de Universidade

## Narrativa - Professores

- Os professores que ministram as disciplinas estão associados as coordenações de seus respectivos cursos. Ex: Computação, Física, Engenharia ...

Levantamento de Requisitos

# Projeto de Universidade

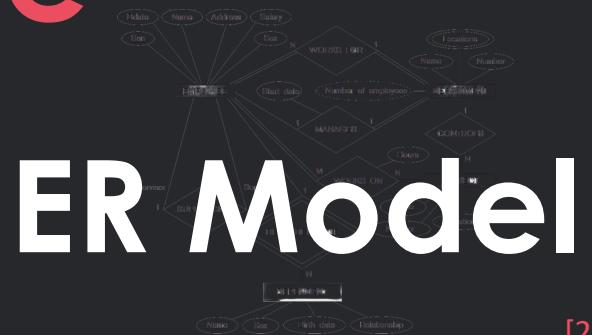
Perguntas:

- Quais informações de aluno e professor guardar?
- Qual média para aprovação?
- Haverá restrição, ou diferentes visões ?
- ...

Levantamento de Requisitos

# Modelando – Escopo

## de E-commerce



# ER Model

# Projeto de E-commerce



Venda de produtos

# Projeto de E-commerce



Qual o  
escopo?

Produto

Estoque

Venda de produtos

Cliente

Fornecedor

Pedido

# Projeto de E-commerce

## Narrativa - Produto

- Os produtos são vendidos por uma única plataforma online.  
Contudo, estes podem ter vendedores distintos (terceiros)
- Cada produto possui um fornecedor
- Um ou mais produtos podem compor um pedido

Levantamento de Requisitos

# Projeto de E-commerce

## Narrativa - Cliente

- O cliente pode se cadastrar no site com seu CPF ou CNPJ
- O Endereço do cliente irá determinar o valor do frete
- Um cliente pode comprar mais de um pedido. Este tem um período de carência para devolução do produto

Levantamento de Requisitos

# Projeto de E-commerce

## Narrativa – Pedido

- Os pedidos são criados por clientes e possuem informações de compra, endereço e status da entrega
- Um produto ou mais compõem o pedido
- O pedido pode ser cancelado

Levantamento de Requisitos

# Projeto de E-commerce

## Narrativa – Fornecedor & estoque

- Vamos pensar juntos ...

Levantamento de Requisitos

# Desafio 1: Replique e melhore!



# Projeto de Universidade

## Refinando

- Cliente PJ e PF – Uma conta pode ser PJ ou PF, mas não pode ter as duas informações
- Pagamento – Pode ter cadastrado mais de uma forma de pagamento
- Entrega – Possui status e código de rastreio

Levantamento de Requisitos

# Desafio 2: Modelando do zero



# Oficina - Narrativa

Levantamento de Requisitos

- Sistema de controle e gerenciamento de execução de ordens de serviço em uma oficina mecânica
- Clientes levam veículos à oficina mecânica para serem consertados ou para passarem por revisões periódicas
- Cada veículo é designado a uma equipe de mecânicos que identifica os serviços a serem executados e preenche uma OS com data de entrega.

# Oficina - Narrativa

Levantamento de Requisitos

- A partir da OS, calcula-se o valor de cada serviço, consultando-se uma tabela de referência de mão-de-obra
- O valor de cada peça também irá compor a OS
- O cliente autoriza a execução dos serviços
- A mesma equipe avalia e executa os serviços

# Oficina - Narrativa

Levantamento de Requisitos

- Os mecânicos possuem código, nome, endereço e especialidade
- Cada OS possui: nº, data de emissão, um valor, status e uma data para conclusão dos trabalhos.

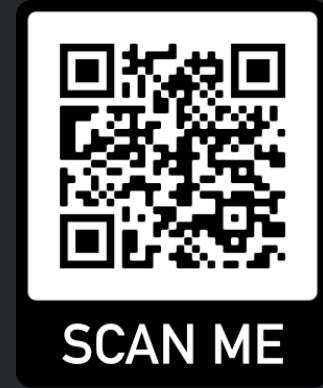
# Oficina - Narrativa

Levantamento de Requisitos

- Uma OS pode ser composta por vários serviços e um mesmo serviço pode estar contido em mais de uma OS.
- Uma OS pode ter vários tipos de peça e uma peça pode estar presente em mais de uma OS

# Dúvidas?

- > Fórum/Artigos
- > Comunidade Online (Discord)



# Links Úteis

## Github

- <https://github.com/julianazanelatto>

## Referências

- <https://dev.mysql.com/doc/>
- <https://www.postgresql.org/docs/>
- <https://www.w3schools.com/sql/>

# Para saber mais

## Livros de Referência

- ELMASRI, Ramez; B. NAVATHE, Shamkant. **Sistema de Banco de Dados**. 7. ed. São Paulo: Pearson, 2018.

## Livros complementares

- RODZVILLA, John. A Review of “**Learning SQL**” Beaulieu, Alan. Sebastopol, CA: O'Reilly, 2009, 320 pp., \$39.99, ISBN 978-0-596-52083-0. 2010.
- TEOREY, Toby J. **Database modeling and design**. Morgan Kaufmann, 1999.
- NIELD, Thomas. **Getting Started with SQL: A Hands-On Approach for Beginners**. " O'Reilly Media, Inc.", 2016.

# Para saber mais

## Links

- [https://www.cs.uct.ac.za/mit\\_notes/database/htmls/chp07.html](https://www.cs.uct.ac.za/mit_notes/database/htmls/chp07.html)