

```
...</g>
<path class="highcharts-crosshair highcharts-crosshair-thin undefined"
data-z-index="2" visibility="hidden" d="M 778.5 45 L 778.5 494"></path>
<g class="highcharts-series-group" data-z-index="3"></g>
<g class="highcharts-exporting-group" data-z-index="3"></g>
<text x="10" text-anchor="start" class="highcharts-title" data-z-index="4"
y="24"></text>
<text x="325" text-anchor="middle" class="highcharts-subtitle" data-z-
index="4" y="24"></text>
<text x="10" text-anchor="start" class="highcharts-caption" data-z-index=
"4" y="631"></text>
<g class="highcharts-axis-labels highcharts-xaxis-labels" data-z-index="7">
</g>
<g class="highcharts-axis-labels highcharts-xaxis-labels highcharts-navigat
or-xaxis" data-z-index="7"></g>
<g class="highcharts-axis-labels highcharts-yaxis-labels" data-z-index="7">
</g>
<g class="highcharts-axis-labels highcharts-yaxis-labels highcharts-navigat
or-yaxis" data-z-index="7"></g>
<g class="highcharts-range-selector-group" data-z-index="7" transform="tran
slate(0,0)"></g>
<g class="highcharts-legend" data-z-index="7" transform="translate(40,596)"
></g>
<g class="highcharts-navigator" data-z-index="8" visibility="visible"></g>
<g class="highcharts-tooltip highcharts-tooltip-0 highchar
"></g>
```

```
...transform-origin: 0px 0px;
)
Inherited from .highcharts-
container (
position: relative;
overflow: hidden;
width: 100%;
height: 100%;
text-align: left;
line-height: normal;
z-index: 0;
-webkit-tap-highlight-color:
transparent;
font-family: "Lucida Grande", "Lucida
Sans", Arial, Helvetica, sans-serif;
font-size: 12px;
user-select: none;
)
Inherited from body.A1000
body.A1000 (
--control-border-color: #000000;
--control-background-color: #FFFFFF;
--control-caret-color: #000000;
--control-dragover-background-color:
#D3D3D3;
--theme-light-color: #FFFFFF;
--text-color: #000000;
--info-icon-color: #000000;
--text-color-alt: #000000;
--text-color-error: #FF0000;
--border-color: #000000;
--bg-color: #FFFFFF;
--color-primary-alt: #000000;
--color-primary-color: #000000;
```

# Cognitive Data Science

PROF. DR. FERNANDO T. FERNANDES



# Vimos

---

- Bancos de dados – introdução
- Modelo Relacional

# Agenda

---

- Modelos de Dados
  - Modelo Conceitual
  - Modelo Lógico
  - Modelo Físico
  - Diagrama de Entidade e Relacionamento

# Modelos de Dados

---

- Descrevem a estrutura (*schema*) de um banco de dados.
- Utiliza conceitos de **abstração de dados** que omite detalhes da organização e armazenamento dos dados.

# *Database Schema*

---

- **Schema:** É a descrição de um banco de dados. Exibe alguns aspectos como nomes e restrições.
- Os dados em um banco de dados em um determinado momento são chamados de *snapshots* ou **estado do banco de dados**.
  - Ao criar um banco de dados, seu estado inicial é vazio.
  - A cada alteração, o estado é alterado.
  - O SGBD é responsável para garantir que o estado do banco de dados seja válido, satisfazendo as regras e a estrutura do *schema*.

# Database Schema

---

## STUDENT

Name	Student_number	Class	Major
------	----------------	-------	-------

## COURSE

Course_name	Course_number	Credit_hours	Department
-------------	---------------	--------------	------------

## PREREQUISITE

Course_number	Prerequisite_number
---------------	---------------------

## SECTION

Section_identifier	Course_number	Semester	Year	Instructor
--------------------	---------------	----------	------	------------

## GRADE\_REPORT

Student_number	Section_identifier	Grade
----------------	--------------------	-------

Exemplo de um *database schema*.

Somente a estrutura do banco de dados é especificada.

Fonte: Elmasri e Navathe (2018)

# Modelos de Dados - Categorias

---

- **Modelo Conceitual** – Concepção dos dados por meio da percepção que os usuários fazem dos dados
- **Modelo Lógico** - Detalha como os dados serão armazenados, incluindo relacionamentos e tipos de dados.
- **Modelo Físico** – Implementação do modelo lógico (normalmente com linguagem SQL) – Ex:Create table... (Veremos na aula seguinte)

# Modelo Conceitual

---

- Descreve como os usuários percebem os dados, por meio de:
  - **Entidades:** Representa objetos da vida real
    - Ex: Funcionário, Empresa, Aluno, etc.
  - **Atributos:** Propriedades de uma entidade
    - Ex: nome e salário do Funcionário
  - **Relacionamentos:** Relação entre duas ou mais entidades
    - Ex: Um funcionário *trabalha em* uma empresa



# Modelo de Entidade e Relacionamento (MER)

- Modelo conceitual que descreve uma base de dados
- Diagrama de Entidade e Relacionamento (DER)
  - Representação gráfica do MER.
  - Diagrama que possui uma **notação** utilizada para **representar** a estrutura

<https://dspace.mit.edu/bitstream/handle/1721.1/47432/entityrelationshx00chen.pdf>

## The Entity-Relationship Model—Toward a Unified View of Data

PETER PIN-SHAN CHEN  
Massachusetts Institute of Technology

A data model, called the entity-relationship model, is proposed. This model incorporates some of the important semantic information about the real world. A special diagrammatic technique is introduced as a tool for database design. An example of database design and description using the model and the diagrammatic technique is given. Some implications for data integrity, information retrieval, and data manipulation are discussed.

The entity-relationship model can be used as a basis for unification of different views of data: the network model, the relational model, and the entity set model. Semantic ambiguities in these models are analyzed. Possible ways to derive their views of data from the entity-relationship model are presented.

Key Words and Phrases: database design, logical view of data, semantics of data, data models, entity-relationship model, relational model, Data Base Task Group, network model, entity set model, data definition and manipulation, data integrity and consistency  
CR Categories: 3.50, 3.70, 4.33, 4.34

### 1. INTRODUCTION

The logical view of data has been an important issue in recent years. Three major data models have been proposed: the network model [2, 3, 7], the relational model [8], and the entity set model [25]. These models have their own strengths and weaknesses. The network model provides a more natural view of data by separating entities and relationships (to a certain extent), but its capability to achieve data independence has been challenged [8]. The relational model is based on relational theory and can achieve a high degree of data independence, but it may lose some important semantic information about the real world [12, 15, 23]. The entity set model, which is based on set theory, also achieves a high degree of data independence, but its viewing of values such as "3" or "red" may not be natural to some people [25].

This paper presents the entity-relationship model, which has most of the advantages of the above three models. The entity-relationship model adopts the more natural view that the real world consists of entities and relationships. It

Copyright © 1976, Association for Computing Machinery, Inc. General permission to republish, but not for profit, all or part of this material is granted provided that ACM's copyright notice is given and that reference is made to the publication, to its date of issue, and to the fact that reprinting privileges were granted by permission of the Association for Computing Machinery. A version of this paper was presented at the International Conference on Very Large Data Bases, Framingham, Mass., Sept. 22-24, 1975.  
Author's address: Center for Information System Research, Alfred P. Sloan School of Management, Massachusetts Institute of Technology, Cambridge, MA 02139.

ACM Transactions on Database Systems, Vol. 1, No. 1, March 1976, Pages 9-36.

# Ferramentas CASE para diagramação

---

- brmodelo
- StarUML
- DBDesigner (já vem embutido no MySQL Workbench)
- Astah
- Erwin
- etc.



# Diagrama de Entidade e Relacionamento

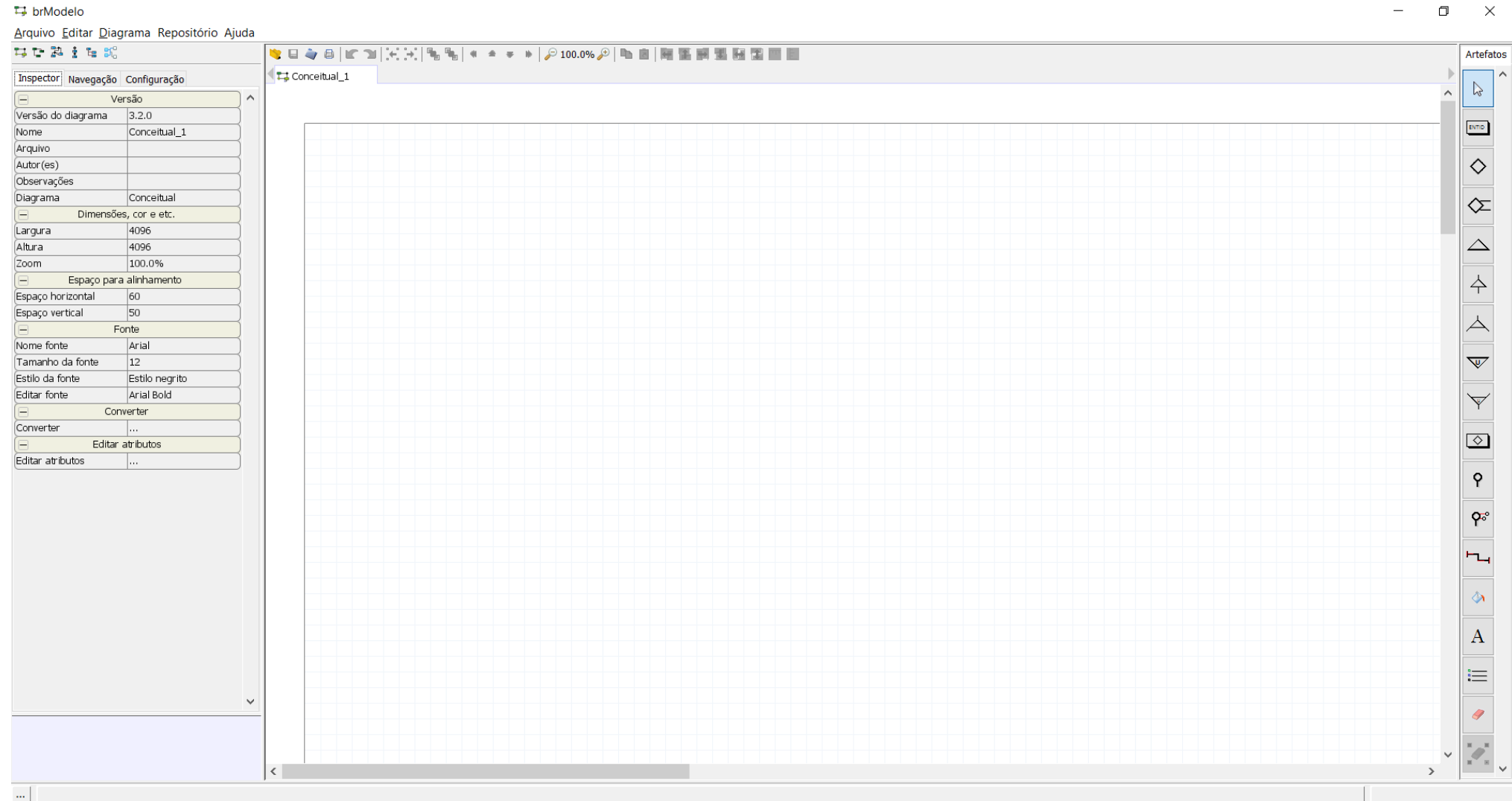
Exemplo prático: Do requisito ao modelo

# Ferramenta utilizada

---

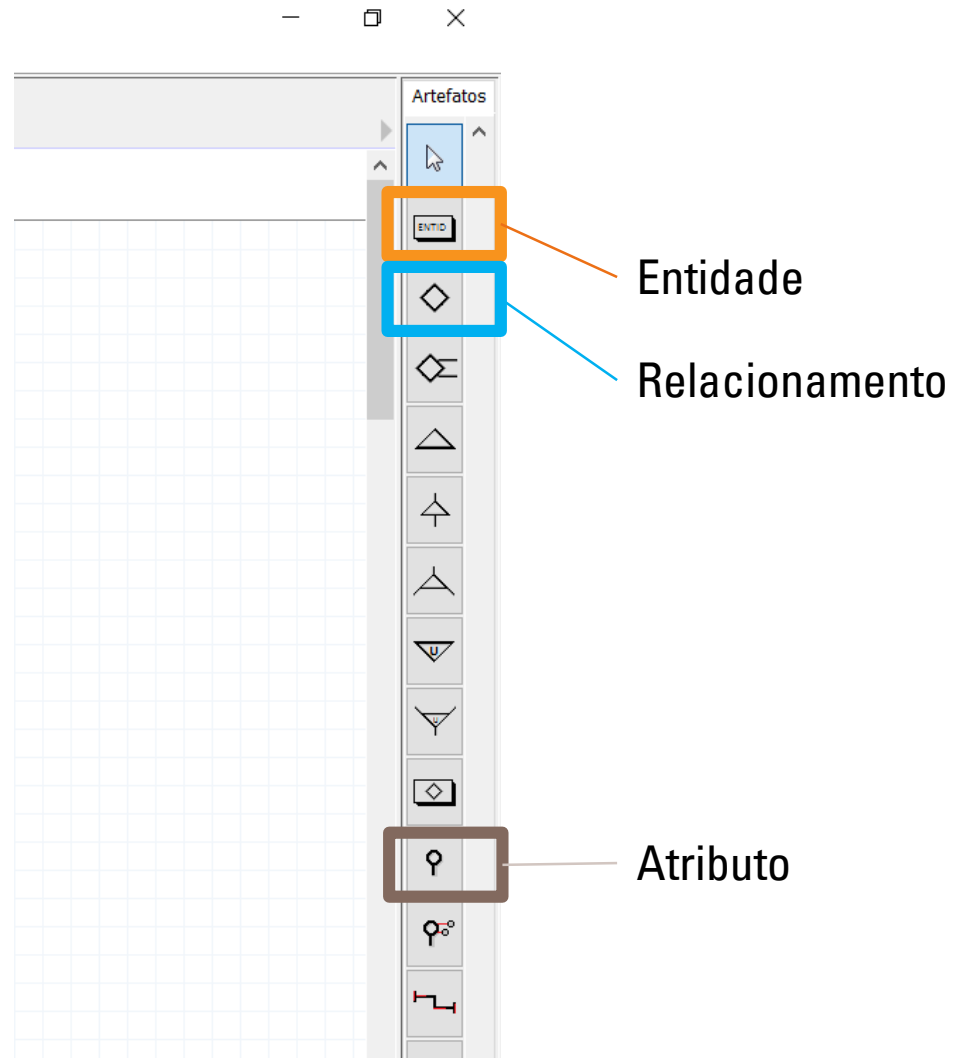
- **BRmodelo** — Ferramenta CASE (*Computer-Aided Software Engineering*) **gratuita** desenvolvida por universidades brasileiras para diagramação de modelos conceituais e lógicos
  - Disponível em: <http://www.sis4.com/brModelo/>
- Disponível na aula de hoje.
  - Descompacte o arquivo zip e adicione o atalho brmodelo.jar (pasta dist) à sua área de trabalho

# Ferramenta utilizada – Tela inicial

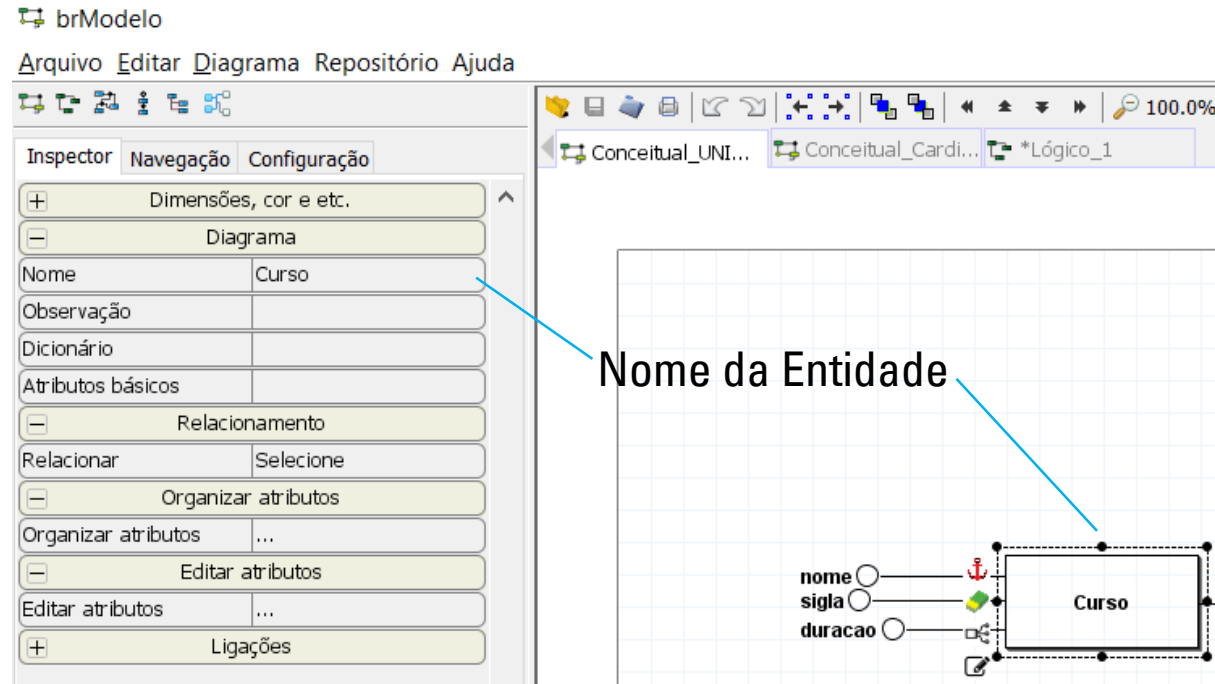


# Ferramenta utilizada – Artefatos

---



# Ferramenta utilizada – Propriedades da Entidade



# Descrição da empresa ACME

---

- A empresa ACME possui diversos departamentos. Cada departamento tem um nome, um número único e um chefe. Cada funcionário tem uma data associada de início na chefia. Um departamento pode funcionar em várias regiões.
- Cada departamento controle um número de projetos, contendo nome, número de identificação única e uma única localização.



# Base de dados da empresa ACME - Requisitos

---

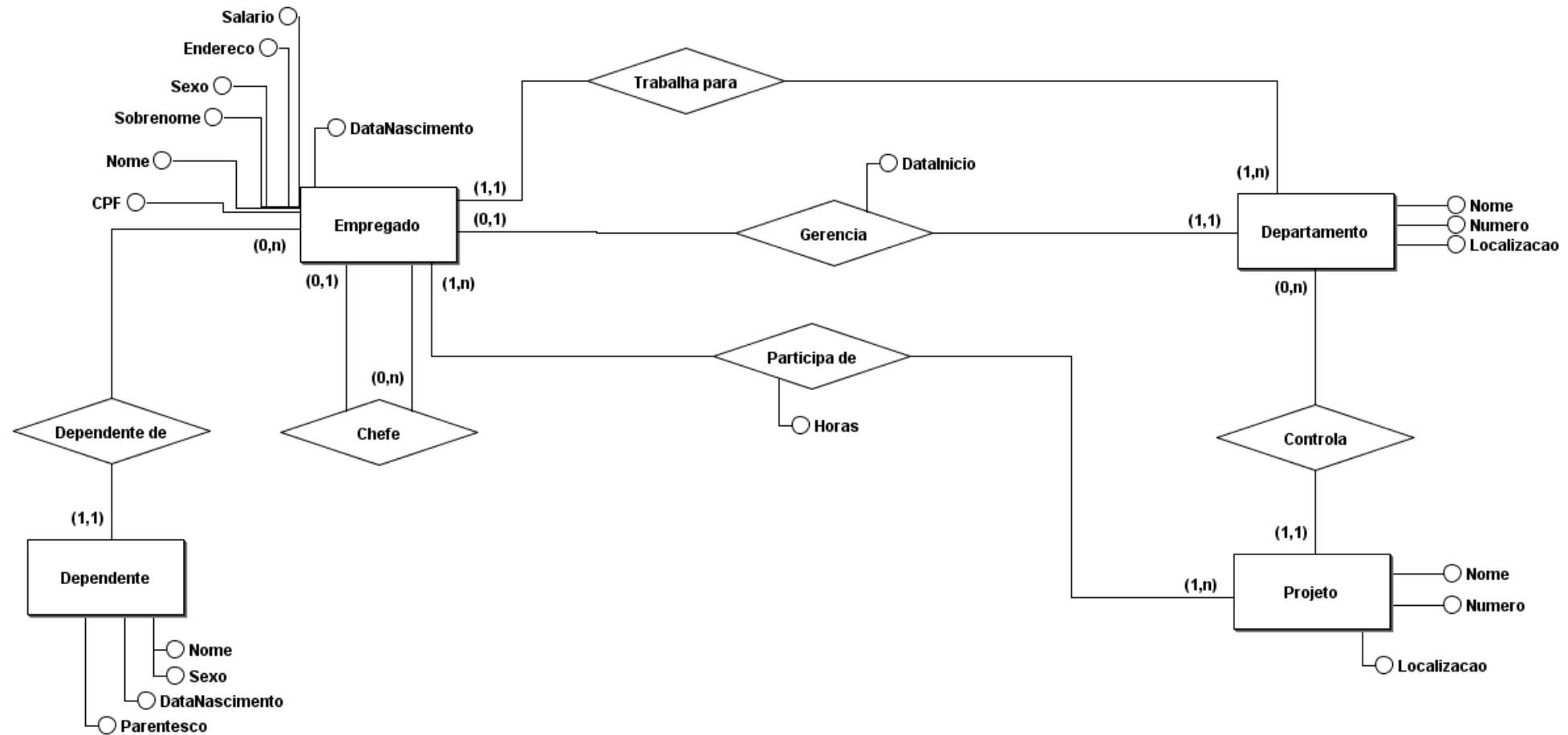
- Para cada funcionário deverão ser armazenados:
  - Nome, CPF, endereço, Salário, sexo (gênero) e data de nascimento
- Cada funcionário trabalha em um único departamento mas pode participar de vários projetos. Devem ser contabilizadas as horas de cada funcionário por projeto.
- Os dependentes dos funcionários devem ser registrados, assim como seus vínculos (ex: filho, filha, etc.)

# Base de dados da empresa ACME - Requisitos

---

- Quais entidades vocês conseguem identificar?
- Quais relacionamentos vocês identificam?
- Quais atributos ?

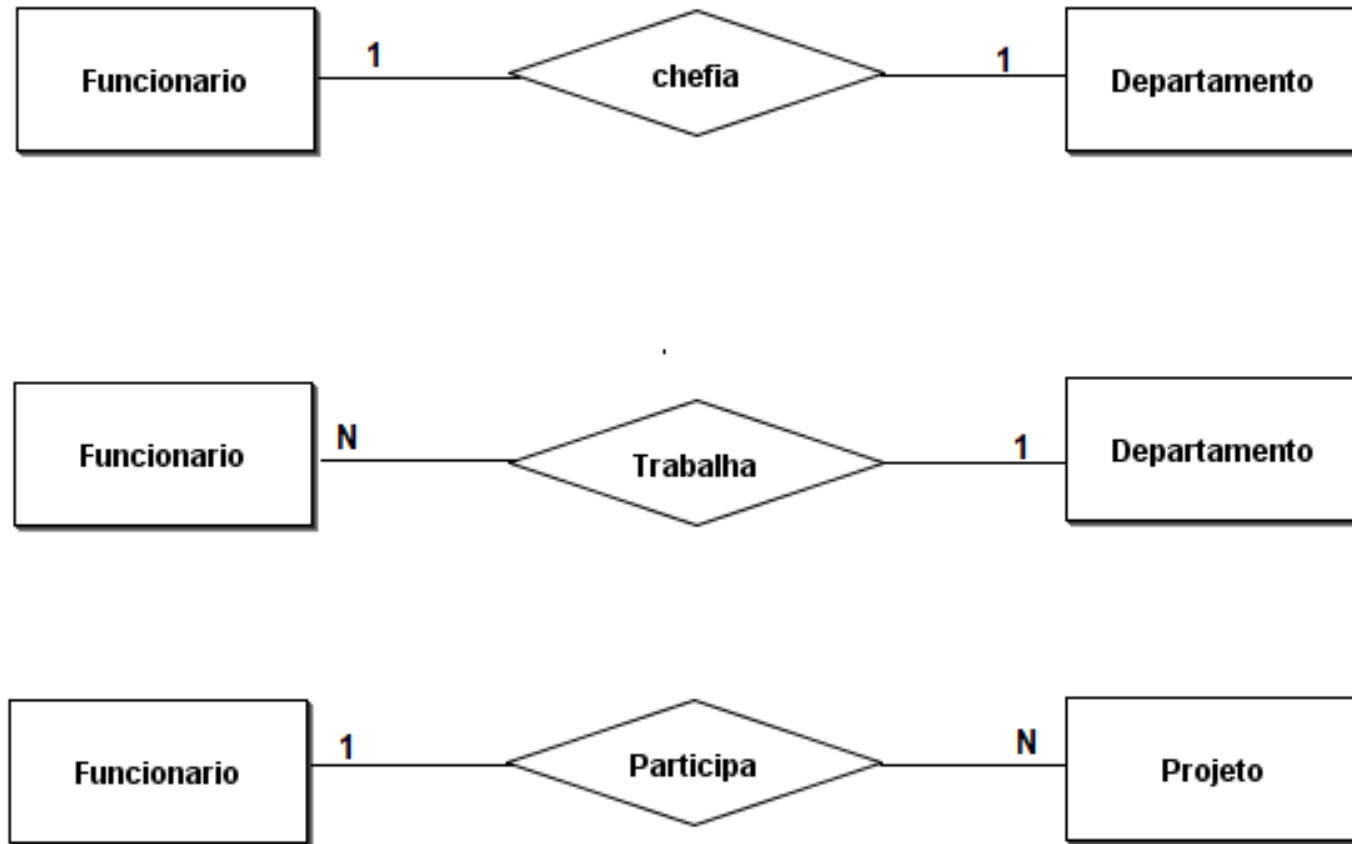
# Base de dados da empresa ACME - DER



Fonte: Adaptado de Elsrá e Navathe (2018)

# Cardinalidade – Define os relacionamento entre entidades

---



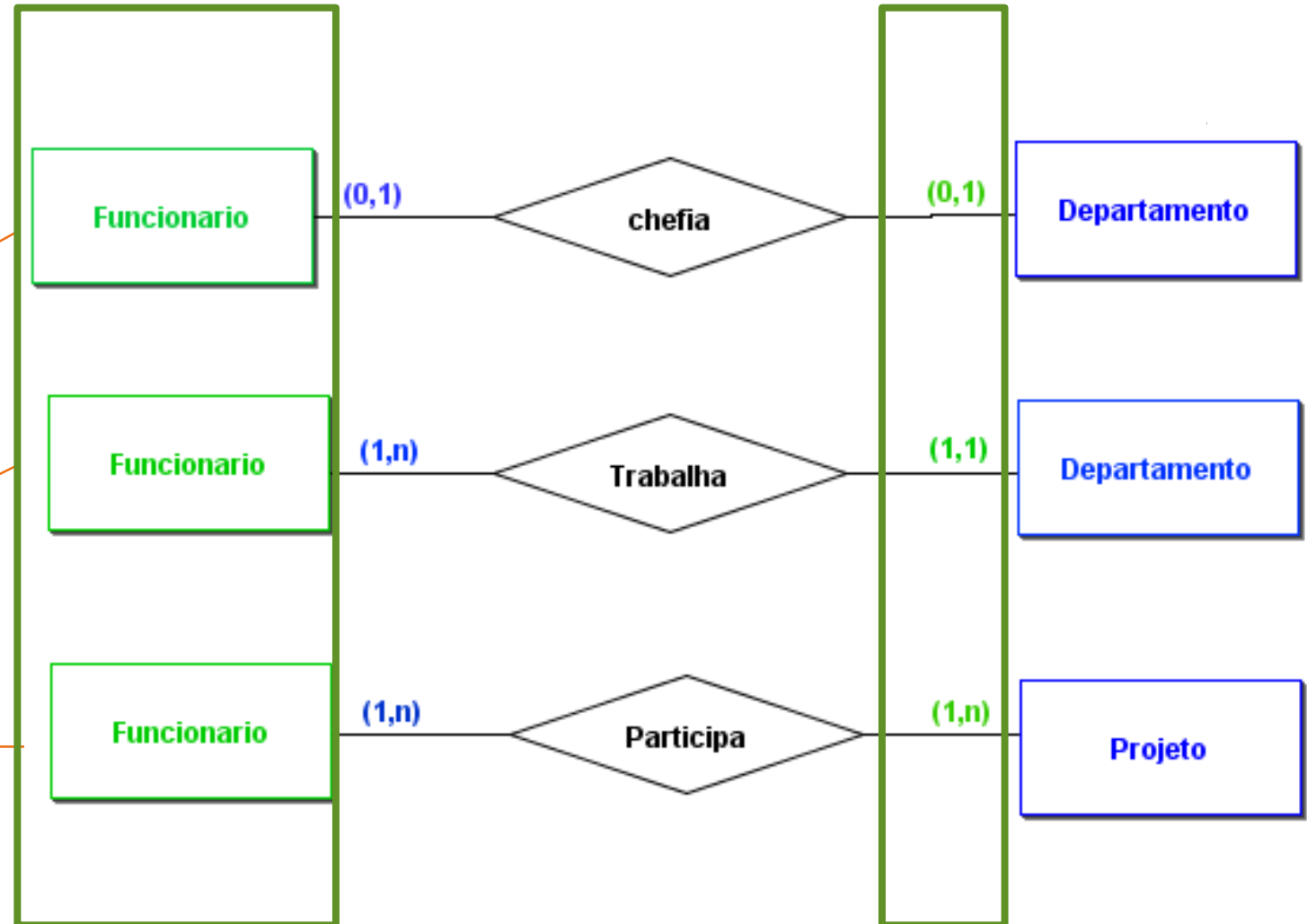
# Cardinalidade – Notação alternativa – mais usada

Refere-se à **participação** (min, max) de cada entidade **E** em um relacionamento **R**

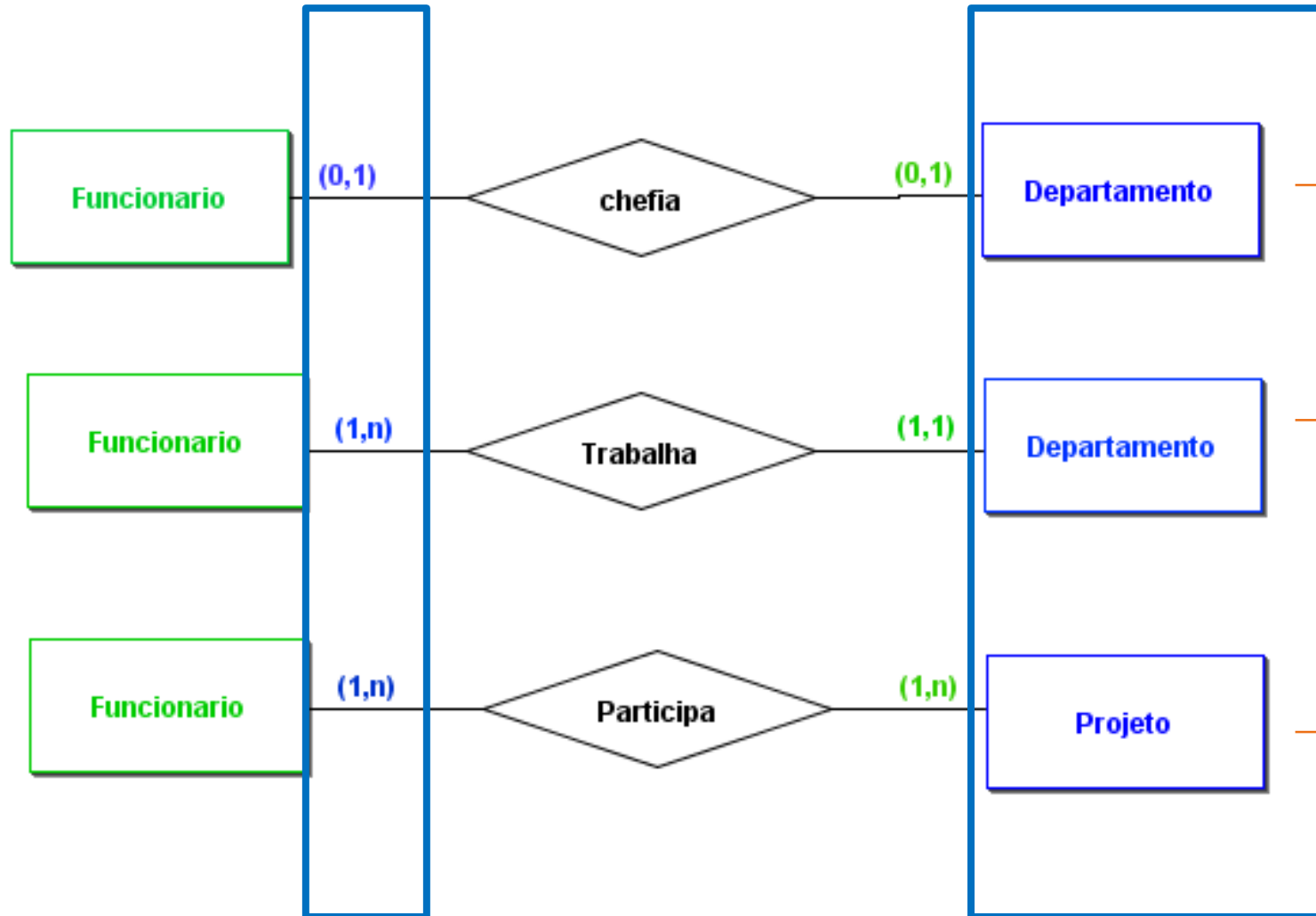
Um funcionário pode ser chefe de nenhum (0) ou 1 departamento no máximo

Um funcionário pode trabalhar em um departamento no mínimo e no máximo.

Um funcionário pode participar em um ou mais (n) projetos.



# Cardinalidade – Notação alternativa – mais usada



Um departamento pode ser chefiado por nenhum funcionário (min) ou por 1 funcionário (max)

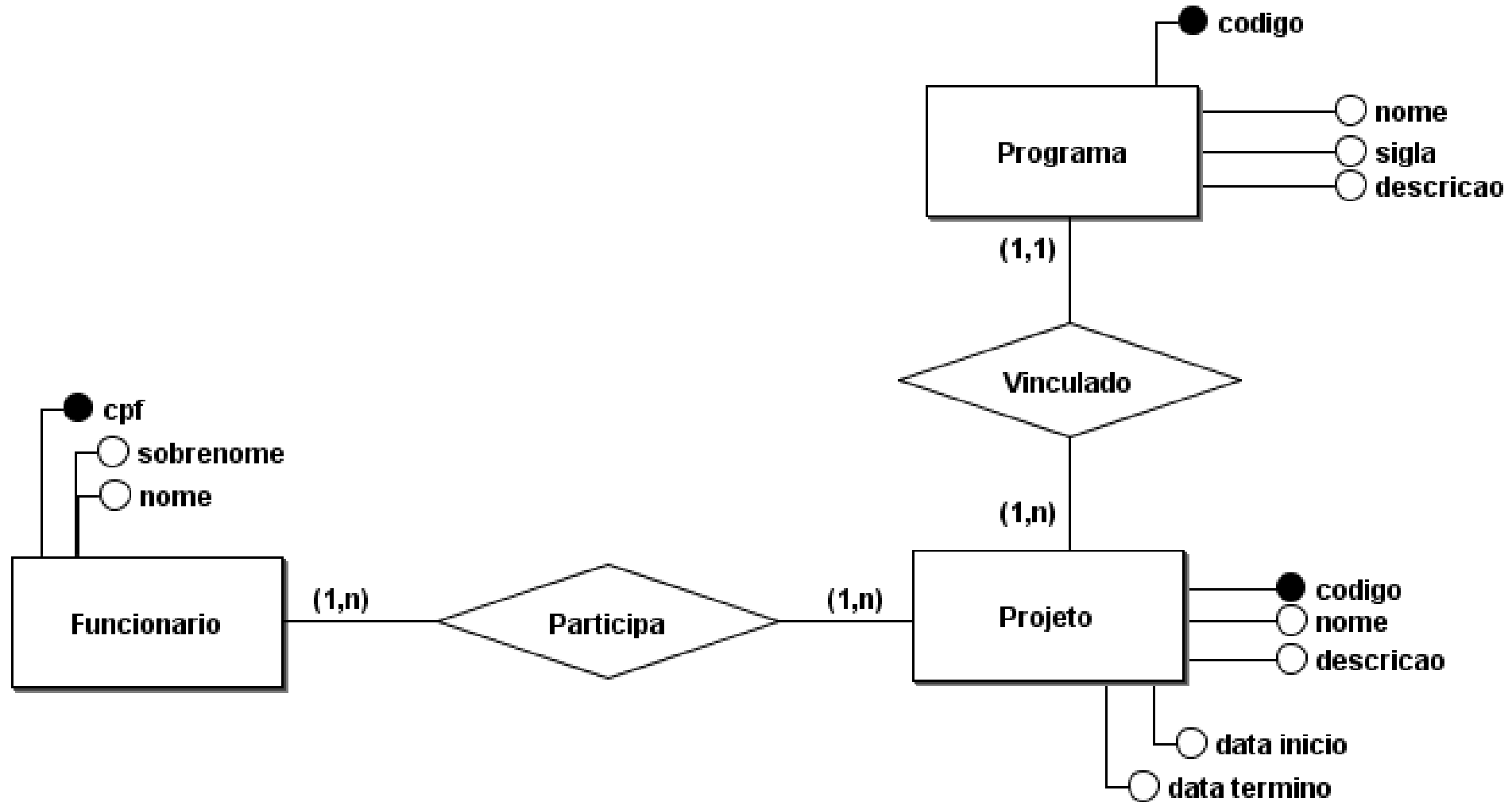
Em um departamento podem trabalhar 1 (min) ou mais (n) funcionários

Um projeto pode ter 1 ou mais funcionários

# SGBD – Diagrama de Entidade e Relacionamento

- ❑ Desenvolva um DER usando o Brmodelo com os seguintes requisitos:
  - ❑ Um departamento resolve registrar os projetos em que seus funcionários participam. Para isso pediu a você modelar a base de dados com os seguintes requisitos:
    - ❑ Um funcionário pode participar de 1 ou mais projetos
    - ❑ Cada projeto pode ter um 1 ou mais funcionários
    - ❑ Cada projeto está vinculado a um único programa
    - ❑ Cada programa pode conter um ou mais projetos.
    - ❑ Para cada funcionário deverão ser armazenados nome, sobrenome e cpf
    - ❑ Para cada projeto deverá constar código, nome, descrição, data de início e data de término
    - ❑ Cada programa deverá ter um código, nome, sigla e descrição.

# SGBD – Diagrama de Entidade e Relacionamento



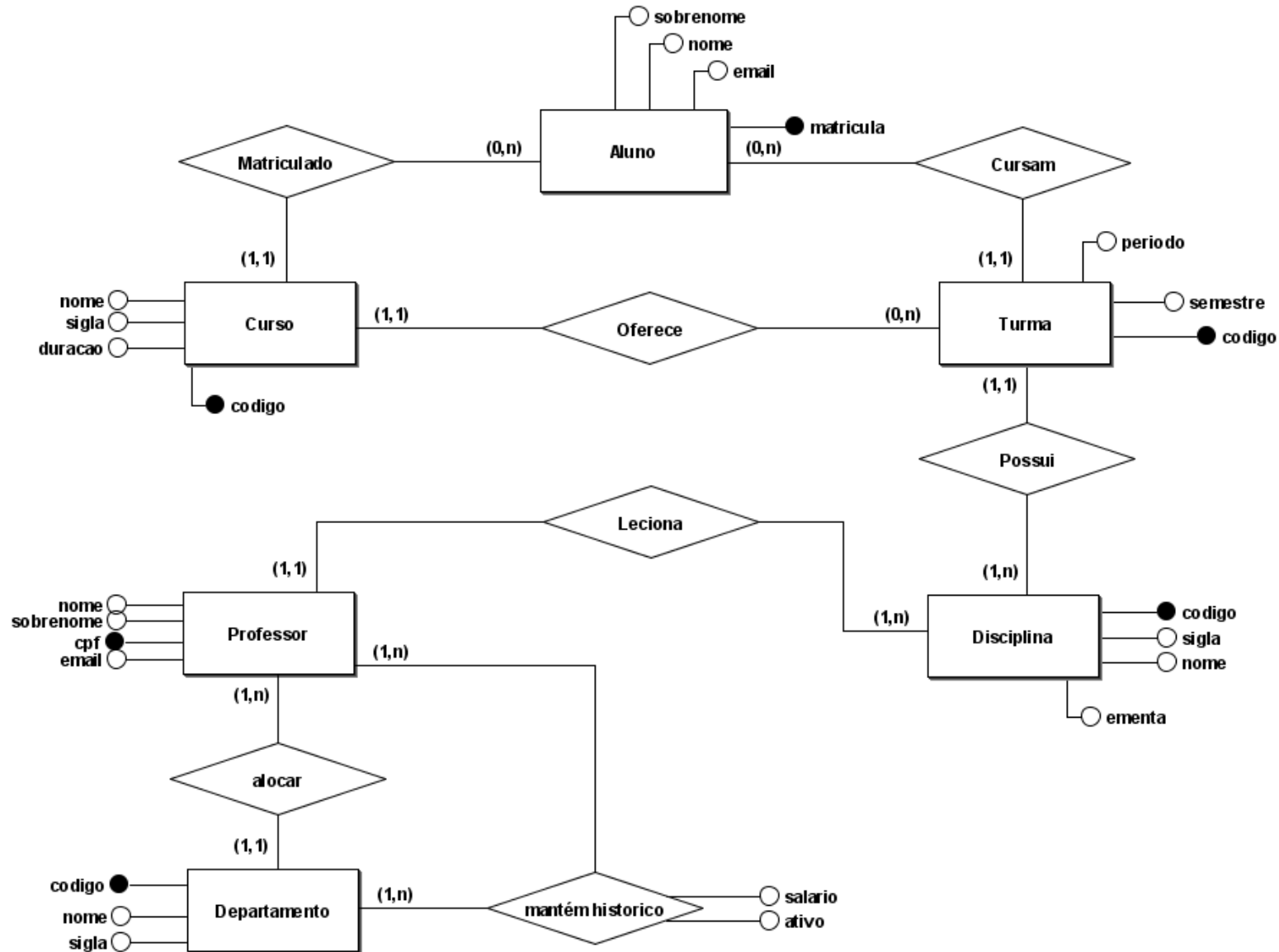


# SGBD – Diagrama de Entidade e Relacionamento

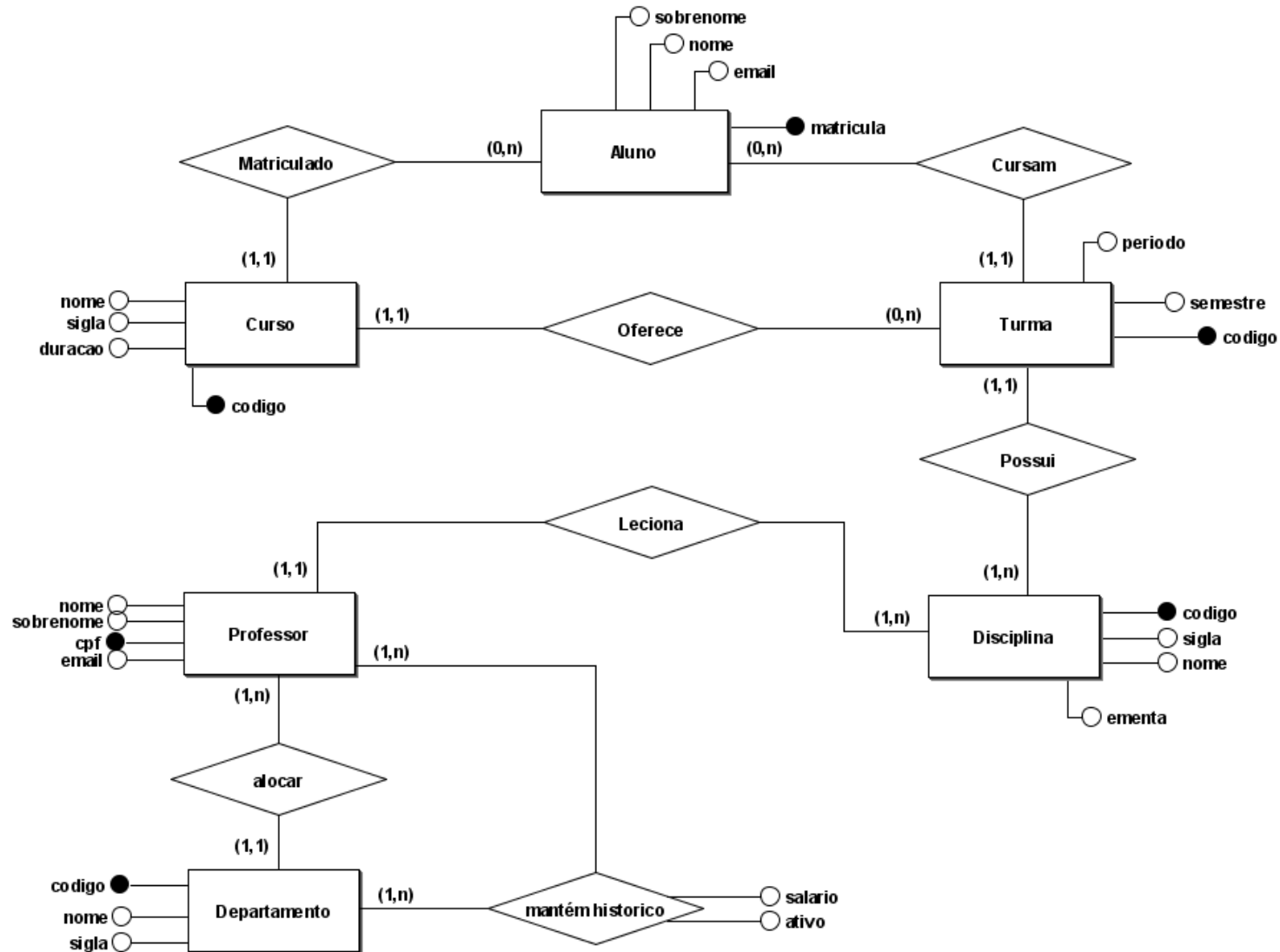
❑ Desenvolva um DER usando o Brmodelo com os seguintes requisitos:

- ❑ A universidade UNIVUS oferece diferentes cursos. Cada curso pode disponibilizar até 3 turmas por período (manhã, tarde e noite). Cada turma possui um código e pertence a um único curso e possui várias disciplinas. Cada professor leciona em uma ou mais disciplinas. Os alunos estão matriculados em um único curso e cursam em apenas uma turma.
- ❑ Uma disciplina é lecionada por apenas um professor. Um professor pode lecionar várias disciplinas. Um professor é vinculado a um departamento. Cada departamento possui um código, nome e sigla, podendo alocar vários professores. O professor pode mudar de departamento, havendo um controle de histórico de alterações de departamento, salários e salário atual do professor.
- ❑ A UNIVUS deseja armazenar o nome, sobrenome, cpf, matricula, email dos alunos, assim como o semestre das turmas que estão cursando. Para cada professor deverá ser registrado seu nome, sobrenome, email e cpf. A disciplina deverá ter um código, sigla, nome e ementa. O curso deverá ter um código, nome, sigla e duração (em anos).

# SGBD – Diagrama de Entidade e Relacionamento



# SGBD – Diagrama de Entidade e Relacionamento



Melhorias ?

Histórico de Disciplinas

Histórico de Professores em Disciplinas

Pré-Requisitos em disciplinas

Etc.

# Obrigado!

Contato: [proffernando.fernandes@fiap.com.br](mailto:proffernando.fernandes@fiap.com.br)