

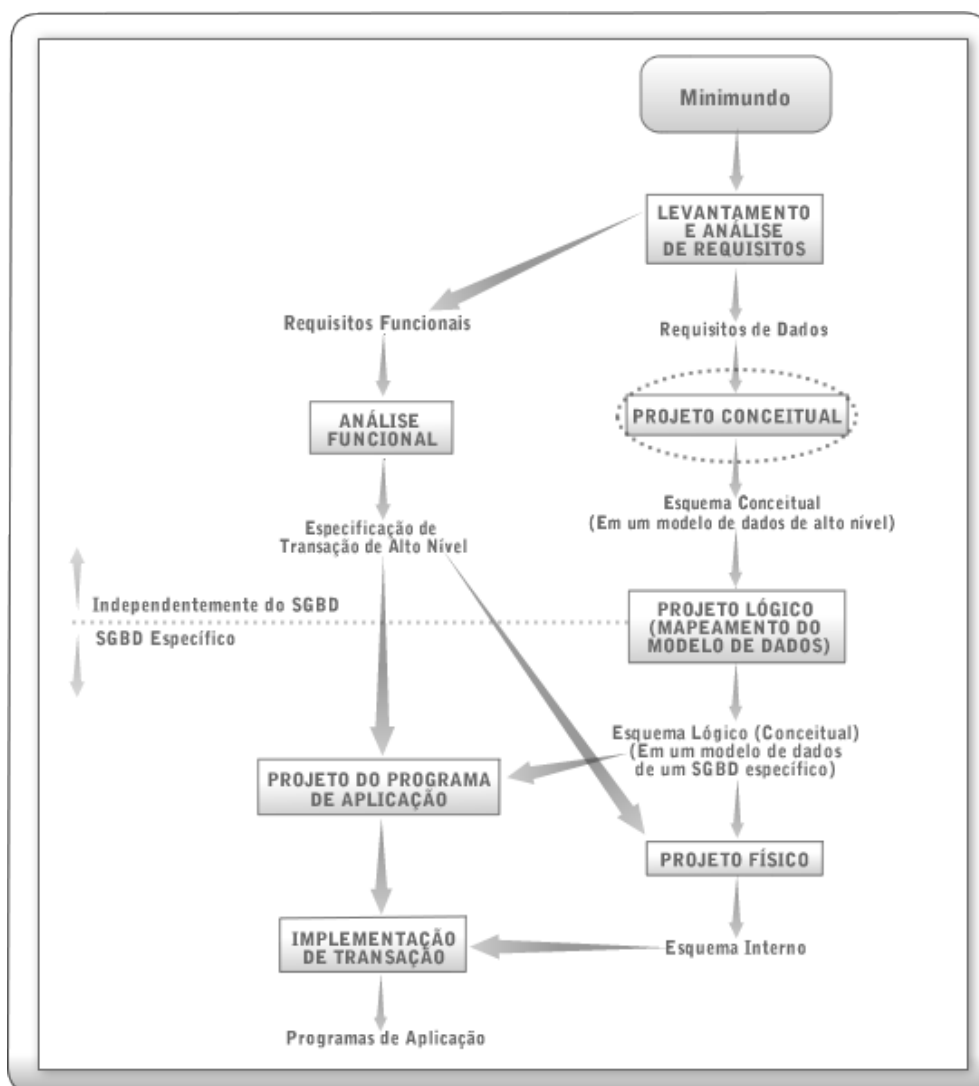
## MODELO CONCEITUAL

Agora que já estudamos o que é um Banco de Dados e outros fatores importantes para este estudo, vamos pensar nos bancos de dados que daqui a algumas aulas já serão implementados em um Sistema de Gerenciamento de Banco de Dados - SGBD. Nesta unidade, começaremos o projeto conceitual, é uma fase que vem após o levantamento de requisitos do sistema, para isso, é necessário que, ao começar o projeto de banco de dados, já exista o levantamento de tudo que será necessário no sistema.

### 5 . Modelo conceitual

A modelagem conceitual é uma fase muito importante no planejamento de uma aplicação de um banco de dados bem-sucedido, este modelo descreve a estrutura de um banco de dados de forma independente de um SGBD particular. É a primeira fase do projeto de um novo Banco de Dados, em que o modelo conceitual é obtido através de um Diagrama de Entidade-Relacionamento(ER).

No livro Sistema de Banco de Dados , os autores mostram uma descrição simplificada do processo de projeto de banco de dados visualizada na figura abaixo.



ELMASRI, Rames; NAVATHE, Shamkant B. Sistema de Banco de Dados. Rio de Janeiro, 4ed.: LTC, 2000.

Na figura, a fase do projeto conceitual vem após o Levantamento e análise de requisitos, pois será a partir dos dados coletados que poderemos elaborar o modelo conceitual usado nesta fase.

O objetivo desta primeira fase do projeto é elaborar uma descrição de quais dados serão armazenados no banco de dados independentemente de como serão implementados no banco de dados. A fase em que definimos como o banco de dados será implementado no SGBD vem após a modelagem conceitual e chamamos de projeto lógico que é obtido a partir da transformação do Diagrama Entidade-Relacionamento em um modelo lógico, que implementa, a nível de SGBD relacional, os dados representados abstratamente no modelo ER.

## 5.1 Modelo Entidade-Relacionamento

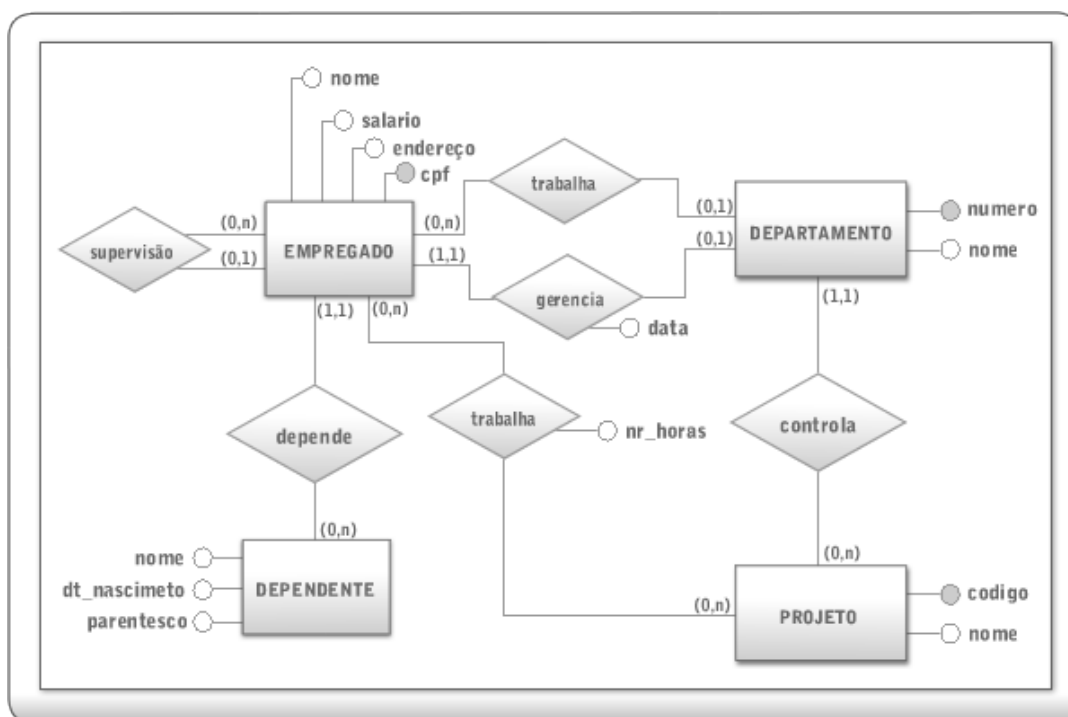
O Modelo ER foi definido por Peter Chen em 1976, sendo representado através de um diagrama de entidade-relacionamento (DER). Vamos estudar alguns conceitos centrais da abordagem entidade-relacionamento e em seguida poderemos elaborar os diagramas.

A seguir apresentamos os requisitos que foram adquiridos a partir da fase de levantamento e análise de requisitos. A partir dos requisitos será apresentado um exemplo de modelo ER, para que a partir deste exemplo possamos ir conhecendo conceitos para elaboração de um modelo ER.

### Os requisitos foram elaborados a partir de uma empresa fictícia chamada “Sorriso”

- A empresa está organizada em departamentos. Cada departamento tem um nome e um número único e um empregado que gerencia o departamento. Temos a data em que o empregado começou a gerenciar o departamento.
- Um departamento controla um numero qualquer de projetos, cada qual com um único nome, um único número e uma única localização.
- Armazenamos o nome de cada empregado, o número do cpf, endereço, salário. Um empregado está alocado a um departamento, mas pode trabalhar em diversos projetos que não são controlados, necessariamente, pelo mesmo departamento. Controlamos o número de horas semanais que um empregado trabalha em cada projeto. Também controlamos o supervisor direto de cada empregado.
- Queremos ter o controle dos dependentes de cada empregado para fins de seguro. Guardamos o primeiro nome, data de nascimento de cada dependente e o parentesco dele com o empregado.

A figura, a seguir, é um exemplo de DER.



Para elaborar os diagramas de ER, precisamos em um primeiro momento conhecer alguns conceitos básicos desta abordagem. Os dados no modelo ER são descritos a partir de **entidades, relacionamentos e atributos**.

### ENTIDADES:

Para Heuser entidade é um conjunto de objetos da realidade modelada sobre as quais se deseja manter informações no banco de dados.

A entidade é considerada o objeto básico representado no modelo ER. Pode ser um objeto do mundo real, por exemplo, uma pessoa ou um objeto com uma existência conceitual como, por exemplo, um curso.

A entidade é representada através de um retângulo contendo o nome da entidade. Cada entidade representa um conjunto de informações sobre o objeto representado, porém quando precisamos nos referir a apenas um objeto, denominamos de ocorrência de entidade.

Exemplo:



No exemplo acima o **DEPARTAMENTO** representa o conjunto de todos os departamentos que se deseja manter informações.

Seguindo o exemplo de levantamento de requisitos e do modelo ER mostrado anteriormente, podemos identificar várias entidades como:

- DEPARTAMENTO
- PROJETO
- EMPREGADO
- DEPENDENTE

### ATRIBUTO:

Cada entidade é representada por Atributos, que são características das entidades representadas. Por exemplo, a entidade **DEPARTAMENTO**, tem como atributos, o nome e o número.

Na prática, os atributos são representados textualmente para não sobrecarregar os diagramas, mas na figura abaixo, podemos ver como os atributos podem ser representados



A figura acima mostra que cada ocorrência de departamento pode ser representada por um número e nome. Por exemplo, vamos armazenar no nosso banco de dados um departamento que terá como número '01' e nome 'Financeiro'.

No livro Sistema de Banco de Dados os autores identificam vários tipos de atributos:

**Atributos compostos x Atributos simples:** Os atributos compostos podem ser divididos em subpartes menores, que representam a maioria dos atributos básicos com significados independentes. Por exemplo, o atributo endereço da entidade empregado por ser dividido em EnderecoRua, Cidade, Estado, CEP. Os atributos que não são divisíveis são chamados de atributos simples.

**Atributos Monovalorados x Multivalorados:** A maioria dos atributos tem um valor único para uma dada entidade; esses atributos são chamados monovalorados. Por exemplo, Idade é um atributo monovalorado de uma pessoa. Em alguns casos, um atributo pode ter um conjunto de valores para a mesma entidade - por exemplo, um atributo Cor para um carro ou um atributo Titulação para uma pessoa. Os carros com uma cor têm um valor único, enquanto aqueles com dois tons contêm dois valores para a cor. Da mesma forma, uma pessoa pode não ter um título acadêmico, outra pessoa pode ter um, e uma terceira pessoa, dois ou mais títulos, portanto pessoas diferentes podem ter números de valores diferentes para o atributo Titulação. Esses atributos são chamados multivalorados.

**Atributos Armazenados x Derivados:** Em alguns casos, dois (ou mais) valores de atributos estão relacionados - por exemplo, os atributos Idade e DataNascimento de uma pessoa. Para uma entidade pessoa em particular, o valor de Idade pode ser determinado pela data corrente(hoje) e o valor de DataNascimento da pessoa. Portanto, o atributo Idade é chamado atributo derivado e é dito derivado do atributo DataNascimento, que por sua vez, é chamado atributo armazenado. Alguns atributos podem ser derivados de entidades relacionadas; por exemplo, um atributo NumerodeEmpregados, de uma entidade departamento, pode ser derivado da contagem do número de empregados relacionados nesse departamento.

**Valores nulls (nulos):** Em alguns casos, determinada a entidade pode não ter um valor aplicável a um atributo. Por exemplo, o atributo Apartamento de um endereço se aplica apenas a endereços que estão em edifícios de apartamentos, e não a outros tipos de residências, como as casas. Por analogia, um atributo titulação só se aplica a pessoas com titulação acadêmica. Para estas situações é criada um valor especial chamado null (nulo).

**Atributos Complexos:** Observa-se que os atributos compostos e multivalorados podem ser aninhados de uma maneira arbitrária. Podemos representar essa organização agrupando os componentes de um atributo composto entre parênteses (), separando os componentes por meio de vírgulas e mostrando os atributos multivalorados entre chaves {}. Esses atributos são chamados de atributos complexos.

Considerando o exemplo da empresa e com as entidades já definidas podemos especificar os atributos de cada uma delas.

DEPARTAMENTO: nome e numero

PROJETO: codigo e nome

EMPREGADO: nome, cpf, endereço e salario

DEPENDENTE: nome, dt\_nascimento e parentesco

### Atributo identificador da entidade

Uma restrição importante das entidades de um tipo de entidade é a chave ou restrição de unicidade em atributos. Um tipo de entidade tem geralmente, um atributo cujos valores são distintos para cada uma das entidades do conjunto de entidade. Cada entidade deve possuir pelo menos um atributo identificador, este é representado pelo círculo preenchido.



A chave é um tipo de restrição de integridade que representa regras de negócio que visam garantir que não possam existir duas ou mais entidades com os mesmos valores de identificadores.

Significa então que o atributo que identifica a entidade serve para que as ocorrências de entidade tenham uma forma de serem distinguidas uma das outras.

A entidade **EMPREGADO** pode ter como atributos:

- cpf
- endereço
- salário
- nome

EMPREGADO			
CPF	nome	Endereço	salário
12345566	Maria	Av. Santos Camargo	860
34567789	Maria	Av. Julio Matheus	980

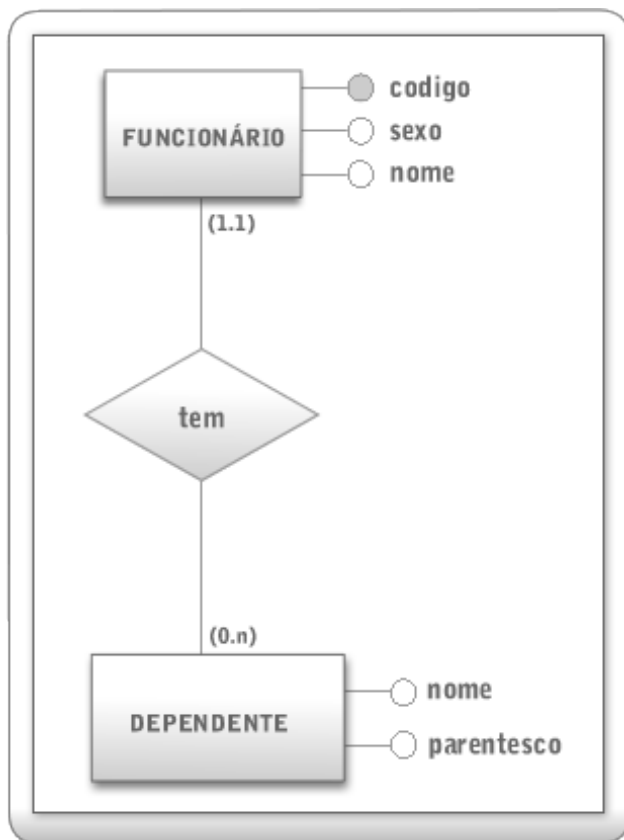
A entidade Empregado guardará informações sobre o cpf, nome, endereço e salário de várias ocorrências de entidade. A partir do exemplo, podemos notar que podem existir pessoas que tenham o mesmo nome.

**O atributo identificador** tem a finalidade de especificar que o atributo que estiver como identificador será único, ou seja, não poderá repetir em nenhuma ocorrência de entidade. Como atributo identificador no exemplo acima, temos o CPF que não poderá se repetir

Os atributos identificadores também podem ser compostos, quando, por exemplo, as ocorrências de entidades necessitem de atributos identificadores diferentes.

### Relacionamento identificador da entidade

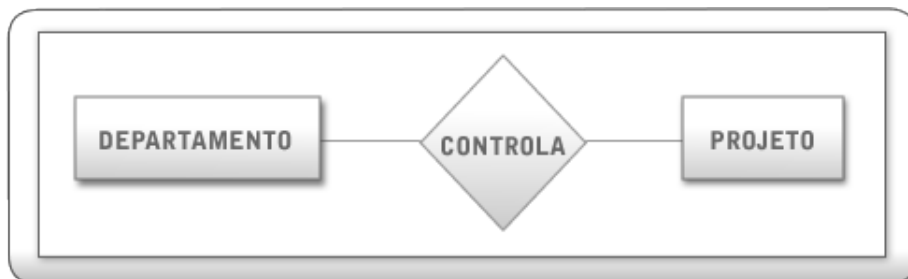
Existem casos em que o identificador de uma entidade é composto de atributo e por relacionamentos em que a entidade participa.



Cada dependente é identificado pelo funcionário em que está relacionado e pelo nome e parentesco. Para identificar um relacionamento identificador usamos uma linha mais densa, conforme a figura acima. Alguns autores definem como a entidade **DEPENDENTE** sendo uma entidade fraca, pois esta depende da entidade **FUNCIONÁRIO** para existir.

### RELACIONAMENTO

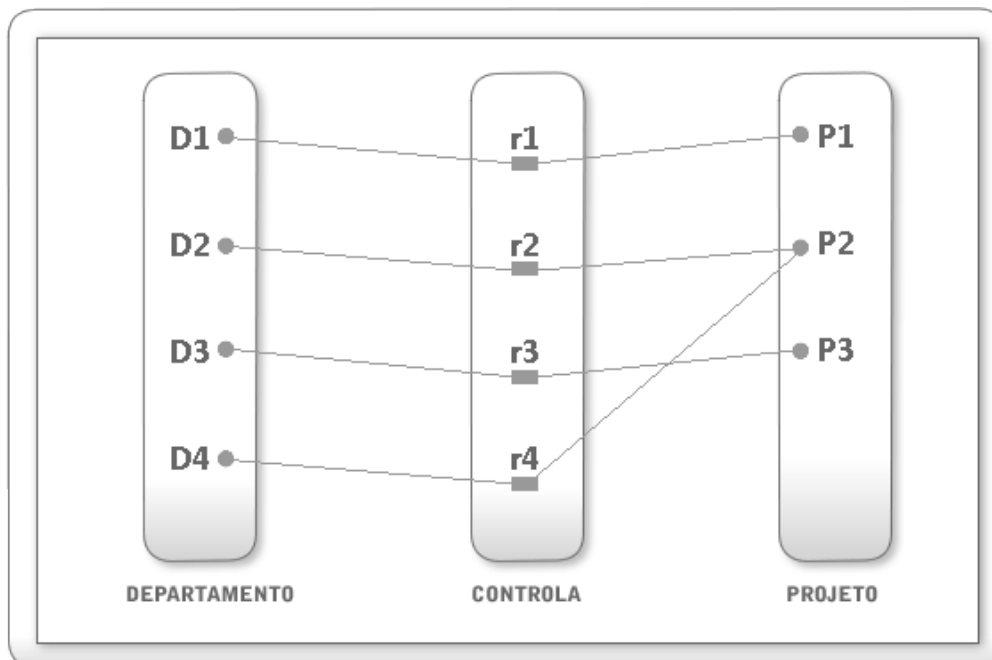
O relacionamento acontece entre as entidades quando existe algum tipo de associação entre estas. Um relacionamento é representado através de um losango ligado por linhas as entidades.



No exemplo acima, temos a entidade **Departamento** e a entidade **projeto**, que estão relacionadas, pois os projetos são controlados por um departamento.

Assim como nós referenciamos as ocorrências de entidades, podemos fazer o mesmo nos relacionamentos. Com isso, quando quisermos nos referir a associações particulares dentro de um conjunto, vamos nos referir a ocorrências de relacionamentos.

Um tipo de relacionamento define um conjunto de associações – ou um conjunto de relacionamentos – entre as entidades.



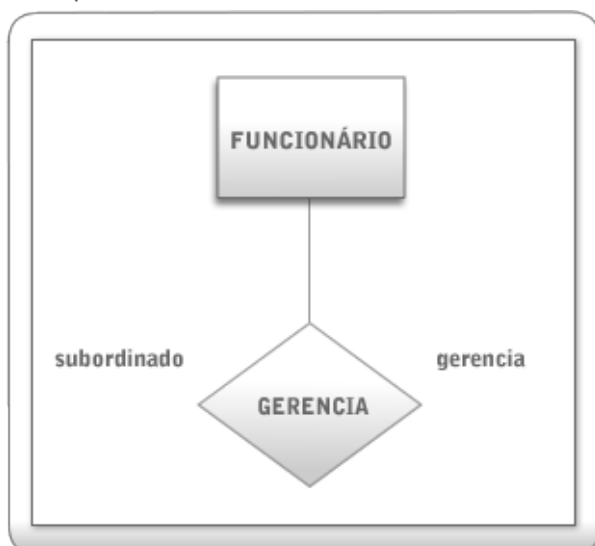
Exemplo:

Consideramos um relacionamento **Controla**, entre dois tipos de entidade, **Departamento e Projeto**, que associa cada controla a uma entidade de Departamento e Projeto. Pela figura acima, identificamos que um departamento pode controlar um projeto assim como pode controlar mais de um projeto

### Auto-Relacionamento

Existe ainda um conceito em Relacionamento que é do auto-relacionamento. Neste tipo de relacionamento a relação ocorre com apenas uma entidade, ou seja, entre ocorrências da mesma entidade. Neste caso surge o conceito de papel da entidade, em que definimos qual é o papel da entidade em um relacionamento.

Exemplo:

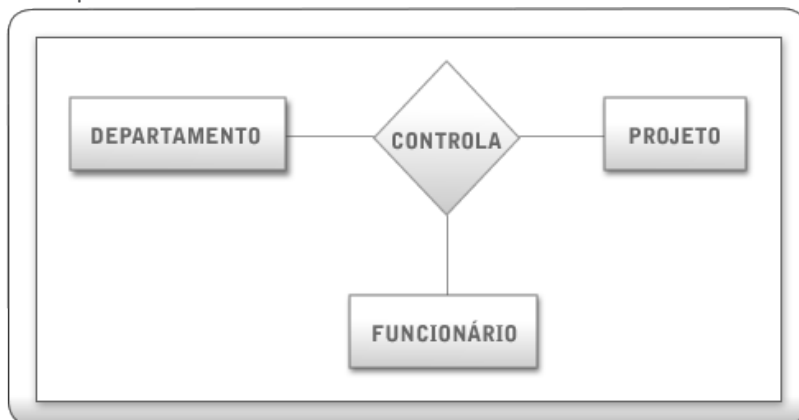


No exemplo acima, a relação que existe entre duas ocorrências de entidades podem ter papéis diferentes. O funcionário poder ser gerenciado por um **funcionário gerente**, assim como um funcionário pode gerenciar os **funcionários subordinados**.

### Relacionamento Binários e Ternários (ou de grau superior)

Os relacionamentos vistos até o momento são identificados como Binários, por se tratar da relação entre duas entidades. Quando temos relação entre três entidades, chamamos de relacionamentos ternários.

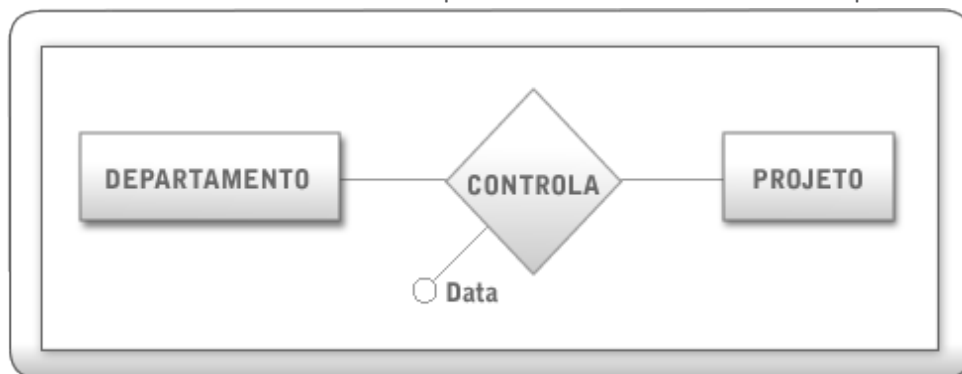
Exemplo:



No exemplo acima, temos uma relação de controle, onde cada ocorrência da relação de controle associa três ocorrências de entidades: um projeto a ser controlado, um departamento onde será feito um controle e um funcionário.

### Identificando Relacionamentos.

Os relacionamentos também podem possuir atributos. Que serão características específicas que ocorrem durante o relacionamento não fazendo parte de nenhuma entidade em específico.



No exemplo acima, podemos armazenar a data em que o departamento começou a controlar o projeto.

### Cardinalidade de Relacionamentos

Cardinalidade de Relacionamento é um conceito utilizado no modelo ER que melhora o conhecimento sobre as políticas e regras dos negócios, consistindo de números colocados ao lado do relacionamento.

Um exemplo para as cardinalidades seria a de um departamento que controla um projeto. As dúvidas que podem aparecer nesta relação seriam:



- Quantos projetos um departamento pode controlar?
- O projeto deve ser controlado por apenas um departamento ou pode ser por mais de um?

Para projetarmos o nosso modelo de dados é essencial que conheçamos este tipo de questão e que estas sejam refletidas no nosso modelo.

Podemos acrescentar algumas dimensões nos nossos relacionamentos para tornar mais claras algumas questões:

DEPARTAMENTO controla “um ou mais projetos”

PROJETO é controlado por apenas um DEPARTAMENTO

Este modelo, agora, indica que um DEPARTAMENTO pode possuir muitos projetos. Porém cada PROJETO só poderá ser controlado por apenas um DEPARTAMENTO.

Para identificar estas características colocamos ao lado do relacionamento dados que representem as mesmas. Chamamos estes dados de **cardinalidade do relacionamento**.

A cardinalidade define, portanto, o número de ocorrências de uma entidade que pode estar associada em um relacionamento, sendo útil para extrair as regras de consistência e integridade dos dados. Assim como define também se um relacionamento é obrigatório ou opcional.

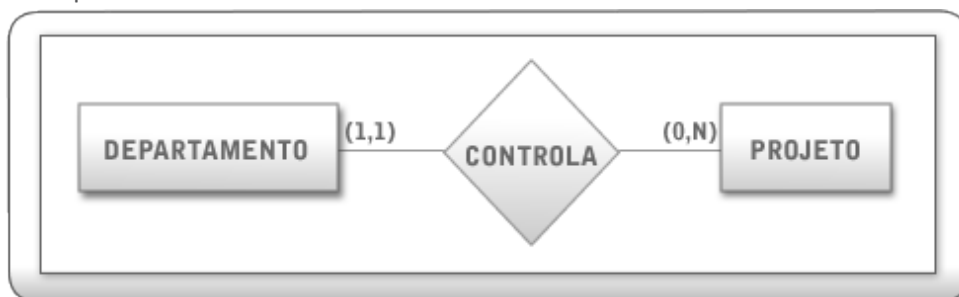
Com isso, há duas cardinalidades a considerar: Cardinalidade Máxima e Cardinalidade Mínima.

**Cardinalidade Máxima:** Representa o número de ocorrências que podem existir em um relacionamento sendo representado por “1” ou “N”, ou seja, um único relacionamento (1) ou muitos relacionamentos (N)

**Cardinalidade Mínima:** Representa a obrigatoriedade ou não do relacionamento sendo representado por “0” ou “1”, ou seja, o relacionamento é opcional (0) ou o relacionamento é obrigatório (1).

## CARDINALIDADE (MÍNIMA MÁXIMA)

Exemplo:



No exemplo, podemos ler da seguinte forma:

**DEPARTAMENTO (1,1) CONTROLA (0,N) PROJETO**

Um DEPARTAMENTO **pode** controlar nenhum (0) PROJETO. (cardinalidade mínima)

Um DEPARTAMENTO **pode** controlar muitos(N) PROJETOS. (cardinalidade máxima)

**OBS:**

O "0" da cardinalidade mínima representa que é opcional o departamento controlar ou não um projeto e o "N" da cardinalidade máxima representa que um departamento pode controlar vários projetos.

Um PROJETO **deve** ser controlado por um(1) DEPARTAMENTO. (cardinalidade mínima)

Um PROJETO **deve ser** controlado por apenas um(1) DEPARTAMENTO. (cardinalidade máxima)

**OBS:**

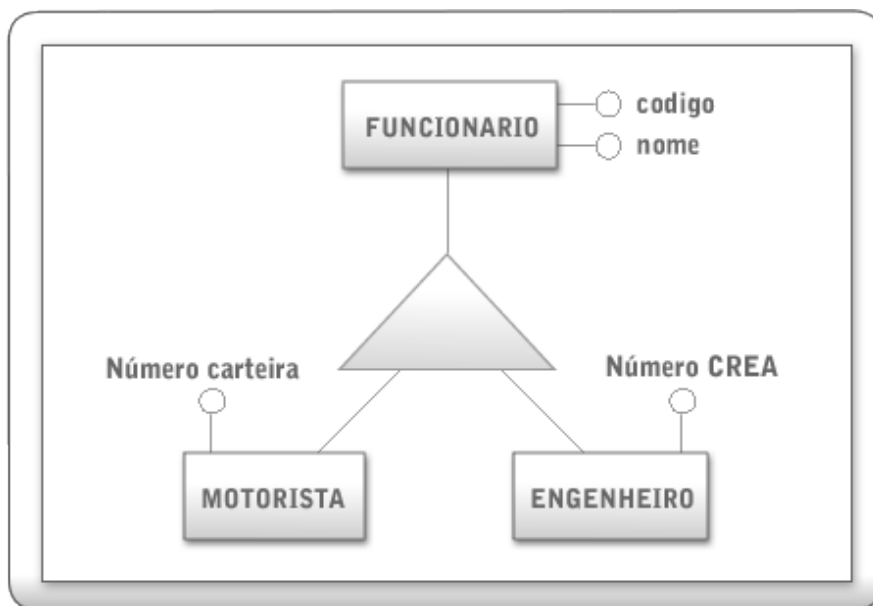
O "1" da cardinalidade mínima representa que é obrigatório o projeto ser controlado por um departamento e o "1" da cardinalidade máxima representa que o projeto só pode ser controlado por um departamento.

**ESPECIALIZAÇÃO e GENERALIZAÇÃO**

Além dos atributos outras características podemos associar as entidades.

**Especialização:**

É um processo de transformar uma entidade que podemos chamar de superclasse em outras subclasses. Por exemplo, as subclasses podem ser {motorista E engenheiro} e a superclasse seria FUNCIONÁRIO. É uma especialização que distingue o FUNCIONÁRIO pelo tipo de trabalho



A figura acima é um exemplo do diagrama para uma especialização. Um relacionamento de superclasse com subclasse é freqüentemente chamado de IS-A (É UM), ou seja, **O MOTORISTA é um FUNCIONÁRIO** e o **ENGENHEIRO é um FUNCIONÁRIO**.

Em relação aos atributos, nos teremos os atributos na superclasse e na subclasse. Os atributos na superclasse serão herdados na subclasse. Por exemplo, os atributos da entidade motorista são: código, nome, número carteira.

Quais seriam os principais motivos de fazer a especialização? Para Navathe, existem duas razões para se incluir as especializações em um modelo de dados.

A primeira é que certos atributos podem ser usados em algumas, mas não em todas as entidades da superclasse. Uma subclasse é definida de forma a agrupar as entidades para as quais estes atributos se aplicam. Os membros de uma subclasse podem ainda, compartilhar a maioria de seus atributos com outros membros da superclasse. A segunda razão para usar as subclasses é que apenas as entidades que sejam membros de alguma subclasse possam participar de algum tipo de relacionamento.

Em resumo, o processo de especialização nos permite fazer o seguinte:

- Definir um conjunto de subclasses de um tipo de entidade.
- Estabelecer atributos específicos adicionais para cada subclasse.
- Estabelecer tipos de relacionamentos adicionais específicos entre cada subclasse e outros tipos de entidade, ou outras subclasses.

### Generalização:

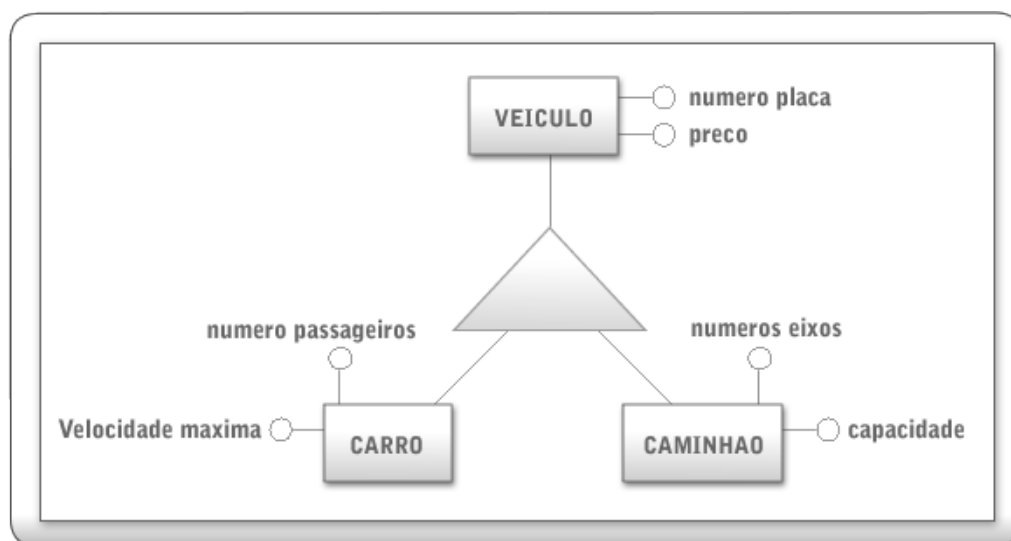
É um processo contrário ao da especialização, que ocorre quando temos mais de uma entidade com atributos comuns entre elas e então neste caso podemos criar uma superclasse com os atributos em comum, utilizando assim o conceito da generalização.

Exemplo:

Entidade Carro: número passageiros, velocidade máxima, preço, número placa.

Entidade Caminhão: preço, número placa, numero eixos, capacidade.

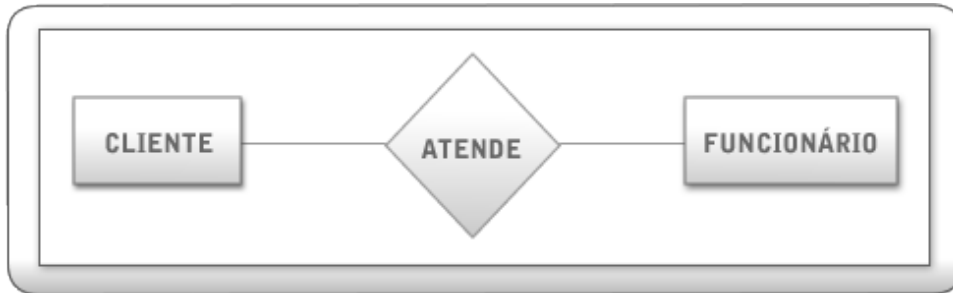
Com a Generalização, podemos criar uma entidade VEICULO que terá como atributos, os em comum das entidades carro e caminhão.



## ENTIDADE ASSOCIATIVA

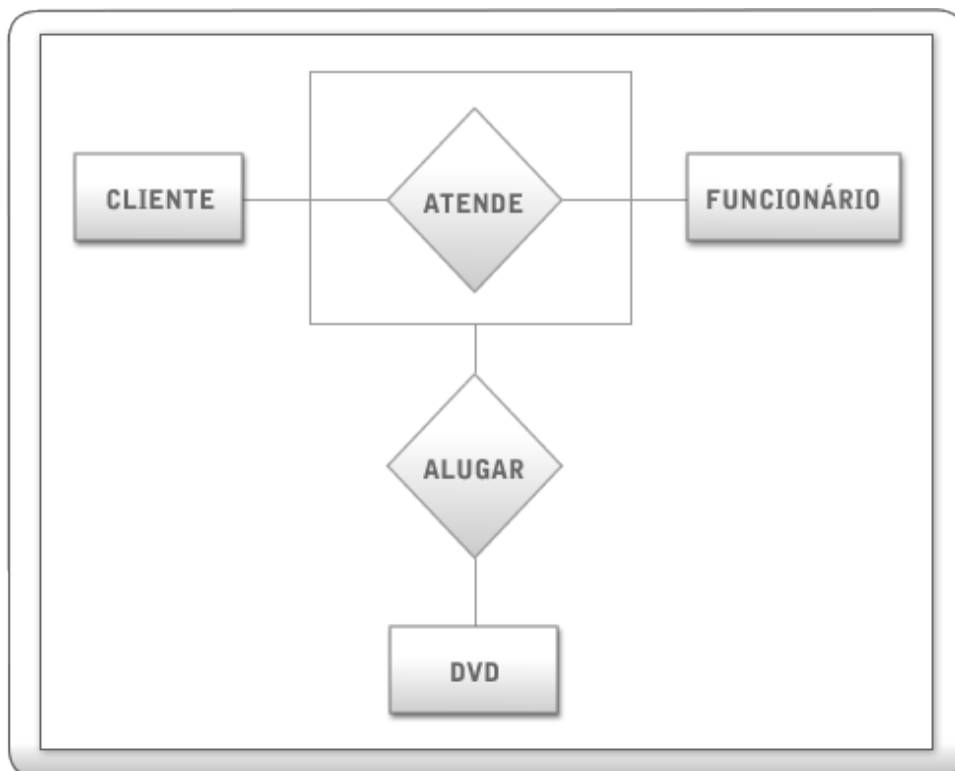
Uma entidade associativa é utilizada quando entre o relacionamento de duas entidades verificamos a necessidade de relacionar com outra relação. Na modelagem er não foi prevista a possibilidade de associar dois relacionamentos entre si, porem na prática quando isso é necessário na modelagem devemos então transformar o relacionamento em uma entidade associativa.

Exemplo:



No exemplo acima, o FUNCIONÁRIO tem uma relação com o cliente de ATENDER. Mas vamos aumentar nosso exemplo, imaginando que o cliente FUNCIONÁRIO ALUGUE Fitas de DVD para o cliente a partir do atendimento. Para isso teremos que criar uma entidade FITAS. Mas o problema é em que entidade vamos relacionar as fitas, se relacionarmos com cliente ficará faltando a informação de que funcionário locou a fita para o cliente e se relacionarmos com o funcionário fica faltando a informação de que cliente locou a fita. Para resolver este problema devemos relacionar as Fitas à relação ATENDE.

Para isso, podemos transformar o relacionamento ATENDE em uma entidade e esta entidade relacionar com FITA.



## CONSTRUÇÃO DE MODELO ER

Segundo Heuser, o Modelo ER tem algumas propriedades que são relevantes para se modelar.

### UM MODELO ER DEVE SER FORMAL

Um modelo ER é um modelo formal, preciso, não ambíguo. Isso significa que diferentes leitores de um mesmo modelo ER devem entender exatamente o mesmo. Tanto é assim, que um modelo ER pode ser usado como entrada em uma ferramenta CASE na geração de um banco de dados relacional. Por isso, é de fundamental importância que todos os envolvidos na confecção e uso de diagramas estejam treinados na sua perfeita compreensão.

### ABORDAGEM ER TEM PODER DE EXPRESSÃO LIMITADO

Em um modelo ER, são apresentadas apenas algumas propriedades de um banco de dados. Em realidade, a linguagem nos modelos ER é uma linguagem muito pouco poderosa e muitas propriedades desejáveis do banco de dados necessitam ser anotados adicionalmente ao DER.

### DIFERENTES MODELOS PODEM SER EQUIVALENTES

Na prática, muitas vezes, observa-se analistas em acirradas discussões a fim de decidir como um determinado objeto da realidade modelada deve aparecer no modelo. Às vezes, tais discussões são absolutamente supérfluas, pois os diferentes modelos ER, em qualquer das opções defendidas pelos diferentes analistas, geram o mesmo banco de dados

### Construção do modelo ER:

O modelo ER lista e define a estrutura requerida para construir um modelo de dados, mas não há um processo padrão para fazê-lo. O importante é saber que o modelo ER não é construído de uma única vez, é um processo incremental, onde o modelo é construído em pequenos passos e pequenas transformações até chegar ao modelo completo.

Existem várias estratégias de modelagens sugeridas pelos autores, mas podemos dentro destas estratégias fazer uma combinação, o que é normalmente aplicado.

Antes de definir qual a estratégia de modelagem devemos identificar todas as descrições e conhecimentos possíveis sobre o processo o que normalmente é coletado na fase de análise de requisitos do sistema.

## Sugestões de estratégias de modelagem:

HEUSER:

Estratégia “ TOP-DOWN”

Nesta estratégia, é sugerido que os conceitos possam ser refinados em conceitos mais detalhados. Partindo assim dos conceitos de entidades genéricas após os seus atributos e na seqüência os relacionamentos entre as entidades após os atributos de relacionamentos e por fim as especializações de entidades.

1. Modelo Superficial: Nesta primeira etapa, é construído um DER pouco detalhado (faltando domínios dos atributos e cardinalidades mínimas de relacionamentos) na seguinte etapa:
  - a) enumeração das entidades;
  - b) identificação dos relacionamentos e hierarquias de generalização/especialização entre as entidades. Para cada relacionamento identificam-se as cardinalidades máximas.
  - c) determinação de atributos de entidades e relacionamentos
  - d) determinação dos identificadores de entidades e relacionamentos
  - e) o banco de dados é verificado quanto ao aspecto temporal
2. modelagem detalhada
  - a) adicionam-se os domínios dos atributos;
  - b) definem-se as cardinalidades mínimas dos relacionamentos
  - c) definem-se as demais restrições de integridade que não podem ser representadas pelo DER
3. Validação do modelo
  - a) Procuram-se construções redundantes ou deriváveis a partir de outras no modelo
  - b) Valida-se o modelo com o usuário.

### Dicas:

*Sites com passos para construir um modelo de dados*

<http://www.sinfic.pt/SinficNewsletter/sinfic/Newsletter41/Dossier2.html>