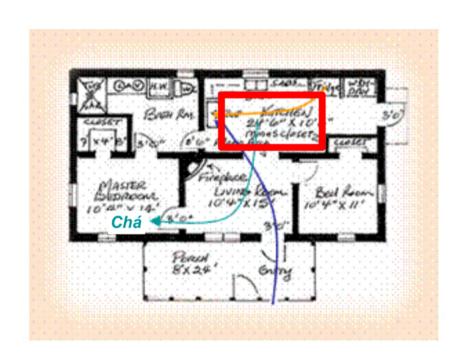


Engenharia de Requisitos Módulo 03 - Requisitos de **Modelagem** Parceria: 1 TATA



Todo sistema de software possui as seguintes características essenciais:

- Possui um limite, ou seja, área que separa o que faz parte de seu escopo do que não faz.
- Capta inputs de agentes externos e gera resultados.
- Possui processos que colaboram entre si para gerar os resultados.
- Os processos permitem criar, modificar, excluir ou consultar dados.
- O sistema pode também usar repositórios de dados para armazenar dados que possam ser utilizados em outros sistemas (em uma migração de plataforma, por exemplo).

Análise e Design de Sistemas Estruturados

A seguir, descreveremos os elementos usados na **Metodologia de Análise e Design de Sistemas Estruturados** (*Structured Systems Analysis and Design Methodology* –SSADM).

Diagrama de Fluxo de Dados (*Data Flow Diagram* – DFD):

Um DFD é usado para processos de modelagem e suas interações.

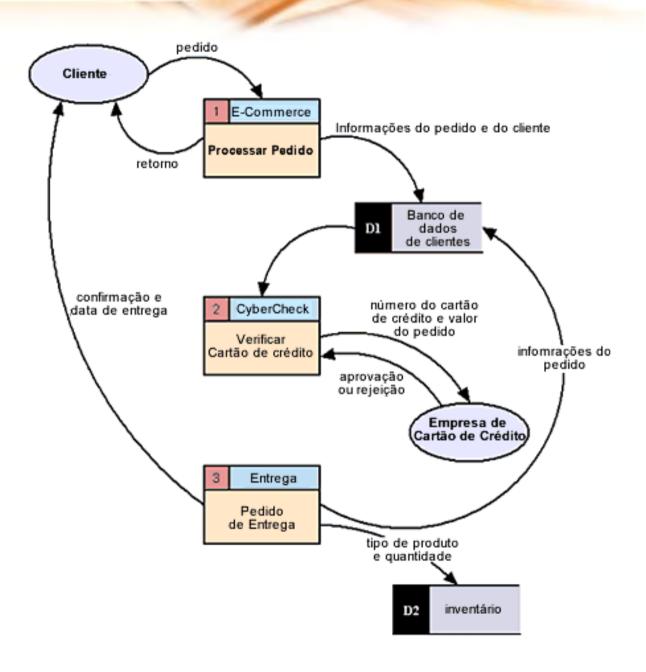
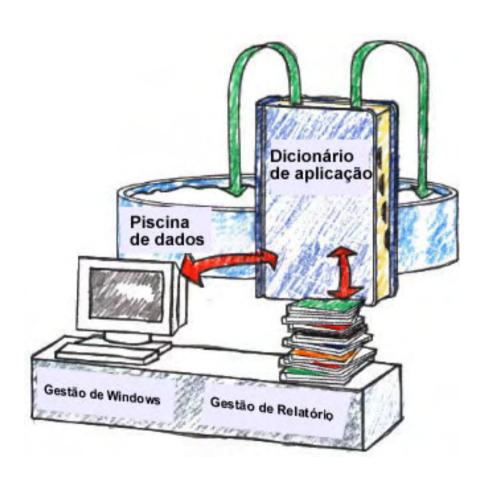




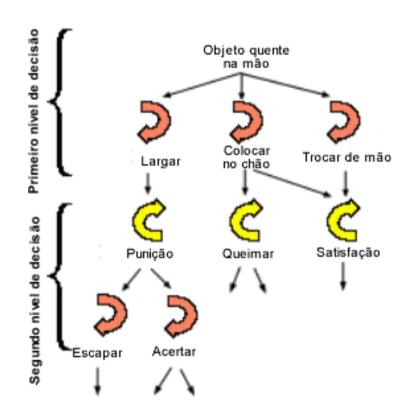
Diagrama de Relacionamento de Entidade (*Entity Relationship Diagram* – ERD):

Um ERD é utilizado para modelagem dos dados e seus relacionamentos.



Dicionário de Dados:

Contem uma lista organizada da estrutura dos dados.



Tabelas de Decisão e Árvores de Decisão:

São utilizadas para modelar decisões complexas.



Inglês estruturado:

Para especificar processos estruturados (algoritmos), utilizase termos em Inglês.

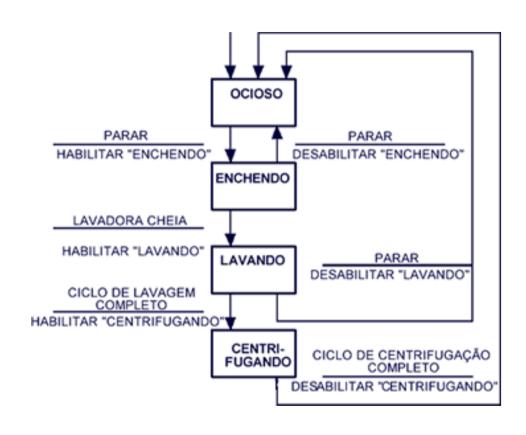
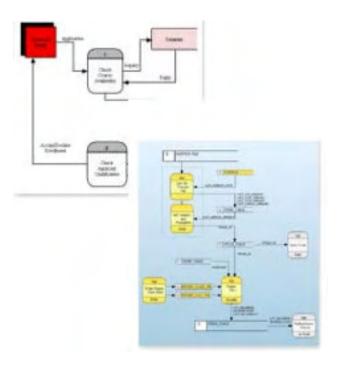


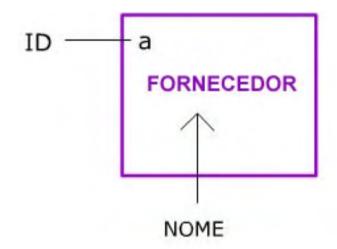
Diagrama de Transição de Estado:

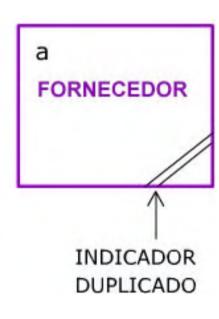
É utilizado para modelar alterações do estado do sistema.



O DFD foca sobre a circulação dos dados através do sistema e suas transformações. Ele é dividido em níveis.

- Nível 0: também conhecido por diagrama de contexto, define o escopo do sistema. É composto por agentes externos, limites de sistema e os dados circulam entre os agentes externos e o sistema.
- Nível 1: é uma expansão do nível 0, na qual todos os principais processos, repositórios de dados e o fluxo de dados entre eles é exibido.
- Nível 2, nível 3 etc.: exibe detalhes de processos individuais.

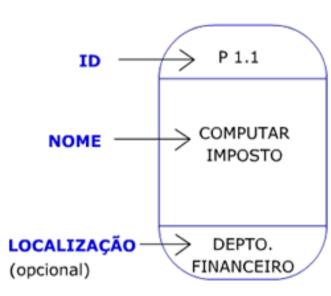




Notações DFD

Agentes externos:

- São externos ao sistema, mas interagem com ele.
- Devem ser desenhados no nível 0, mas não precisam ser desenhados a partir do nível 2.
- Devem receber nomes significativos.
- As duplicidades devem ser identificadas.



PROCESSO ID:

UMA HIERARQUIA NUMÉRICA por exemplo, 1, 1.1, 1.1.1 1.2 1.2.1

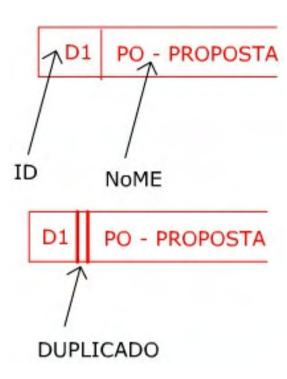
NOME:

- VERBOS FORTES (por exemplo, COMPUTAR, CRIAR) E SOMENTE OBJETOS, SEM ASSUNTO.
- EVITAR VERBOS FRACOS (por exemplo, MANTER, PROCESSAR)

LOCALIZAÇÃO (opcional): LOCAL DE REALIZAÇÃO (por exemplo, DEPTO DE COMPRA, DEPTO FINANCEIRO)

Processo:

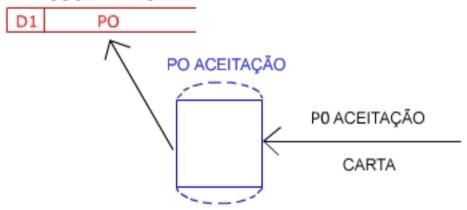
- As atividades de processamento de informações devem ser indicadas em todos os níveis.
- No nível 0, é exibido um único processo, representando o sistema.
- Em níveis subsequentes, o número de processos deve ser limitado a 7 ± 2.
- Duplicações não são permitidas.



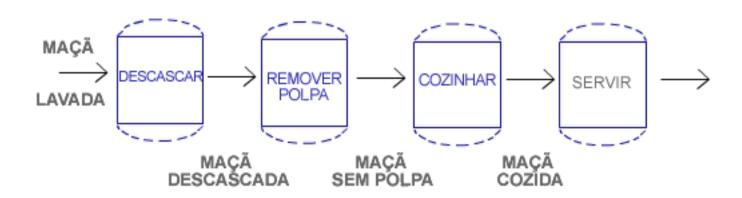
Repositórios de dados:

- São áreas usadas para armazenar informações e não são visualizadas no Nível 0.
- Todos os repositórios de dados devem ser exibidos no Nível 1.
- As duplicações precisam ser indicadas.

ACESSO A DADO



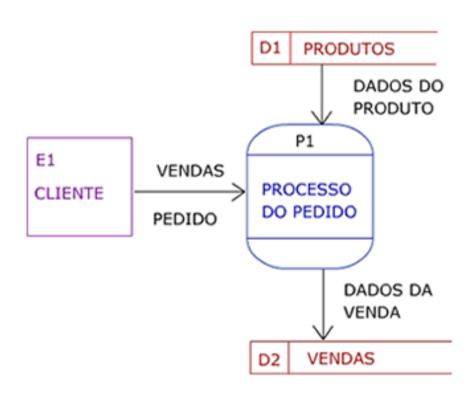
NOME ADICIONADO AO FLUXO DE DADOS



Fluxos de Dados:

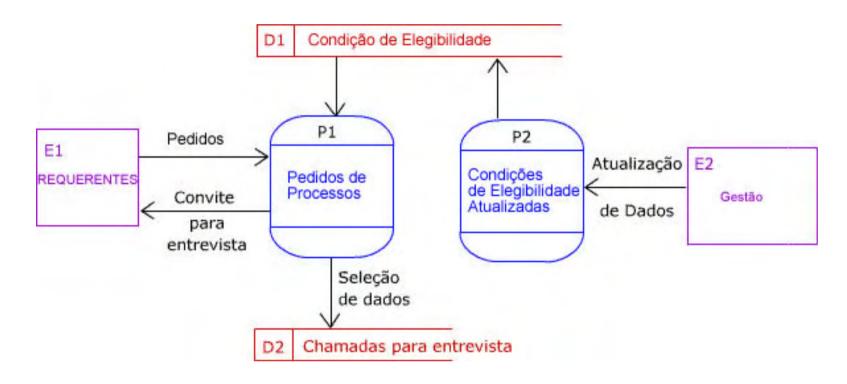
Indicam fluxo das informações.

Precisam ser exibidos em todos os níveis e devem ter nomes significativos.



Exemplos:

 O pedido do cliente emite uma ordem de venda. O sistema checa a disponibilidade do produto e atualiza as informações da venda.



 A empresa recebe um pedido de contratação. Checa as condições de qualificação. Convida os candidatos qualificados para uma entrevista. Mantém uma lista desses candidatos. Atualiza as condições de qualificação de acordo com o desejado pela administração.

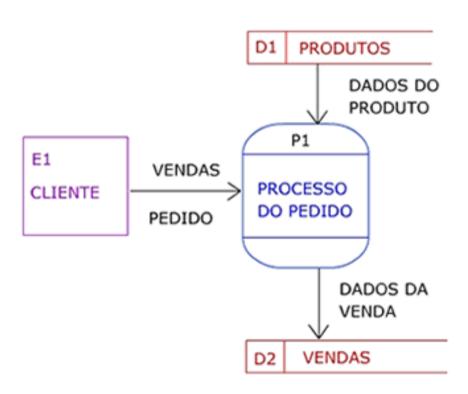


Iniciando um DFD

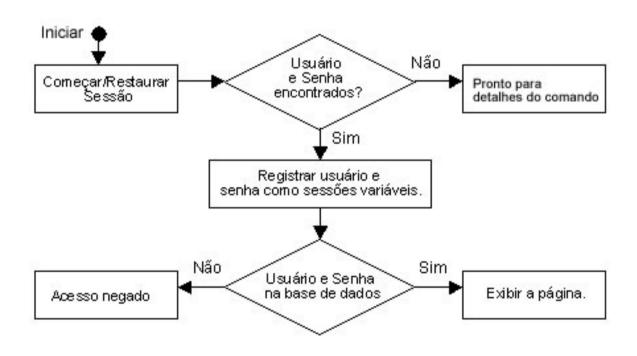
- Identifique as entradas ou eventos que d\u00e3o origem ao sistema, bem como seus resultados ou respostas.
- Identifique as fontes correspondentes e os destinos (agentes externos).
- Elabore um diagrama de contexto (nível 0) que exiba os limites do sistema, os agentes externos e os fluxos de dados que conectam o sistema aos agentes externos.
- Elabore o diagrama de nível 1 que exiba todos os agentes externos, todos os processos principais, todos os repositórios de dados e todos os fluxos de dados conectando os vários compontentes. Os componentes devem ser colocados com base na precedência lógica ao invés de precedência temporal. Evite cruzamentos de fluxos de dados.
- Refina o diagrama de nível 1.
- Expanda os processos individuais conforme necessário.



Lembre-se...



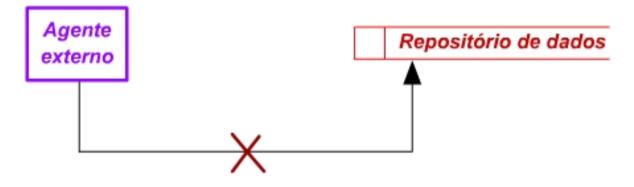
 Nomeie todo agente externo, processo, repositório de dados e fluxo de dados.

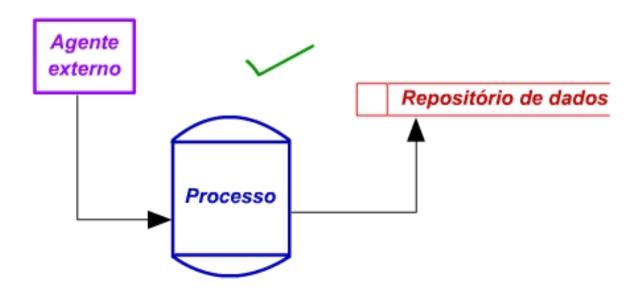


- Não mostre como as coisas se iniciam e terminam.
- Não mostre *loops* e decisões.

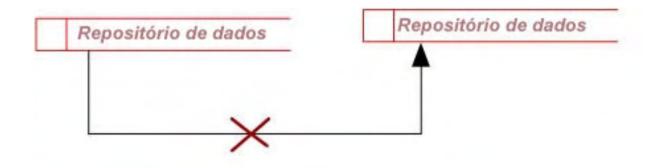


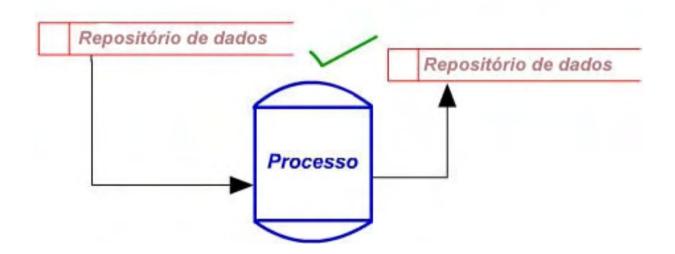
 Não mostre fluxos de dados entre agentes externos. Eles estão fora do escopo do sistema.



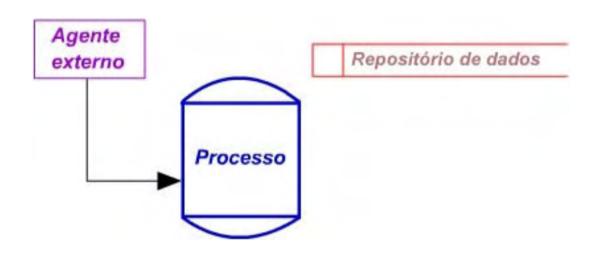


 Não exiba o fluxo de dados entre um agente externo e repositório de dados. Pode haver um processo entre eles.

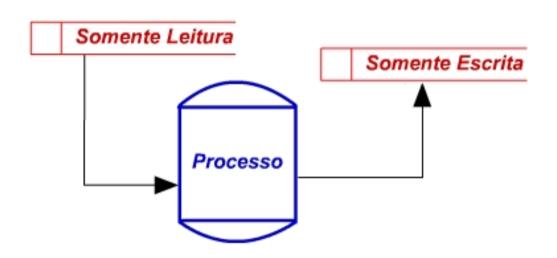




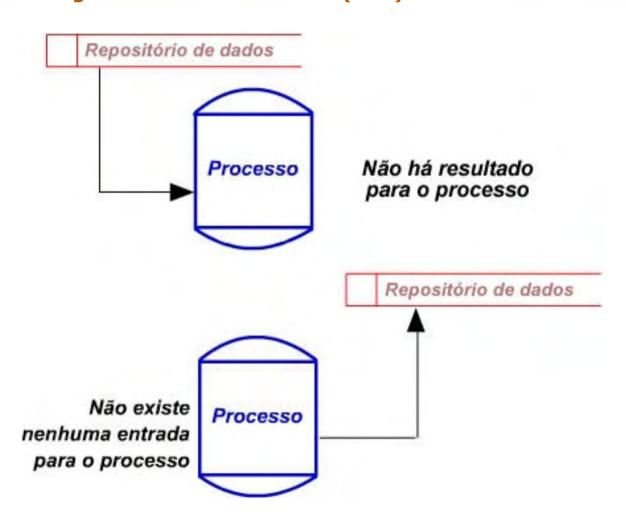
 Não mostre fluxo de dados entre dois repositórios de dados. Pode haver um processo entre eles.



 Não deve haver nenhum agente externo, processo ou repositório de dados desconectado.



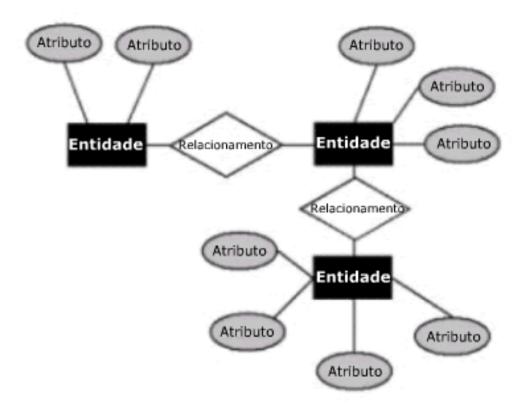
• Tome cuidado com repositórios de dados somente para leitura ou somente para escrita (gravação).



 Tome cuidado com processos que utilizam entradas sem gerar nenhum resultado, bem como processos que geram resultados espontaneamente sem utilizar nenhuma entrada.



- Assegure que o fluxo de dados em um processo corresponda ao fluxo de dados na visualização expandida desse processo. Igualmente para o fluxo de dados fora dele.
- Assegure que o fluxo de dados fora de um repositório de dados corresponda aos dados que foram armazenados nele anteriormente.



O ERD complementa o DFD. Enquanto o DFD foca em processos e no fluxo de dados entre eles, o ERD foca em dados e nos relacionamentos entre eles.

O ERD ajuda a organizar os dados de maneira disciplinada, garantindo a precisão, adaptabilidade e estabilidade deles. É uma ferramenta efetiva para se comunicar com a administração sênior (quais são os dados necessários para executar os negócios?), administradores de dados (como gerenciar e controlar dados), projetistas de bancos de dados (como organizar os dados de maneira eficiente e remover redundâncias).

O ERD consiste de três componentes:

- Entidade
- Atributos
- Relacionamentos

| Reunião | Venda | Cliente | Client

Entidade

Representa uma coleção de objetos ou coisas reais cujos membros individuais ou ocorrências possuem as seguintes características:

- De algum modo, cada um pode ser identificado com exclusividade.
- Cada um desempenha um papel necessário no sistema que está sendo desenvolvido.
- Cada um pode ser descrito por um ou mais atributos.

Entidades geralmente correspondem a pessoas, objetos, localizações, eventos etc. Por exemplo, funcionários, vendedores, fornecedores, materiais, armazéns, data de entrega etc.

A seguir, estão os cinco tipos de entidades.

- Entidade Fundamental
- Entidade Subordinada
- Entidade Associativa
- Entidade de Generalização
- Entidade de Agregação

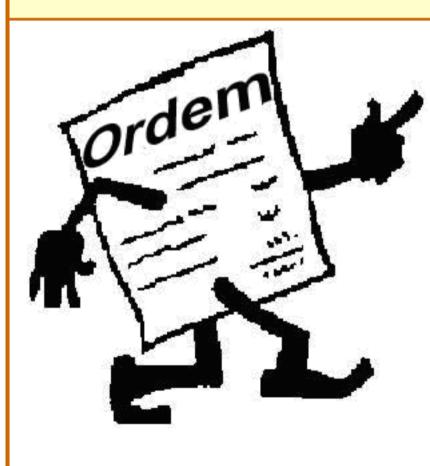
Clique em cada tipo de entidade para obter mais informações sobre ela.

Entidade Fundamental



Entidade Subordinada

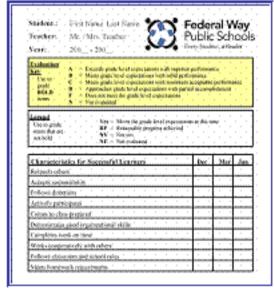
Entidade Subordinada:



Depende de outra entidade para sua existência. Por exemplo, em um sistema de gerenciamento de estoque, ordem de compra pode ser uma entidade e ela dependerá dos materiais sendo adquiridos. De maneira semelhante, as faturas dependerão das ordens de compra.

• Entidade Associativa

Entidade Associativa:



Depende de duas ou mais entidades para sua existência. Por exemplo, notas de um aluno dependerão do aluno e do curso.

• Entidade de Generalização

Entidade de Generalização:



Ela encapsula características comuns de muitas entidades subordinadas. Por exemplo, um carro de quatro rodas é um tipo de veículo. Um caminhão é um tipo de carro de quatro rodas.

• Entidade de Agregação

Entidade de Agregação:



Ela consiste de uma agregação entidades. de outras exemplo, um carro consiste de motor, chassi. caixa de engrenagens etc. Um veículo também pode ser considerado entidade de como uma agregação, porque um veículo pode ser considerado como uma agregação de muitas peças.

Empregado

Número do Empregado Sobrenome Nome dado Data de aniversário Número de telefone Departamento

Atributo das entidades

Atributos:

Expressam as propriedades das entidades.

Toda entidade terá muitos atributos, mas apenas um subconjunto, que é relevante para o sistema sob estudo, será escolhido.

Por exemplo, uma entidade funcionário terá atributos profissionais como nome, designação, salário etc. e também atributos físicos como peso, altura etc. Mas apenas um conjunto será escolhido dependendo do contexto.

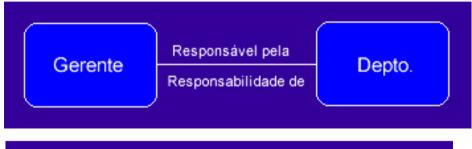
Empregado
EmplID (PK)
EmplID (PK)
Deptid (FK)

Atributos com chave de entidades

Os atributos são classificados como chaves de entidade e descritores de entidade.

As chaves de entidade são usadas unicamente para identificar as ocorrências de entidades. Os atributos que possuem valores únicos são chamados de chaves candidatas e um deles é chamado de chave principal.

Os domínios dos atributos devem ser predefinidos. Por exemplo, se 'nome' for um atributo de uma entidade, seu domínio é o conjunto de caracteres alfabéticos com tamanho predefinido.



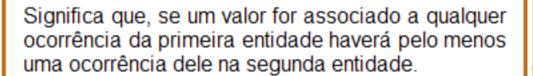


Relacionamentos:

Descrevem a associação entre entidades e podem ser obrigatórios ou opcionais:

- Relacionamento Obrigatório
- Relacionamento Opcional

Obrigatório



Por exemplo, para que possamos visualizar o nome do cargo de um funcionário tendo por base o código do cargo, é obrigatório criar um relacionamento entre a tabela funcionários e a tabela de cargos.

Opcional



Significa que poderá haver ocorrências na primeira entidade que não estejam associadas a nenhuma ocorrência da segunda entidade.

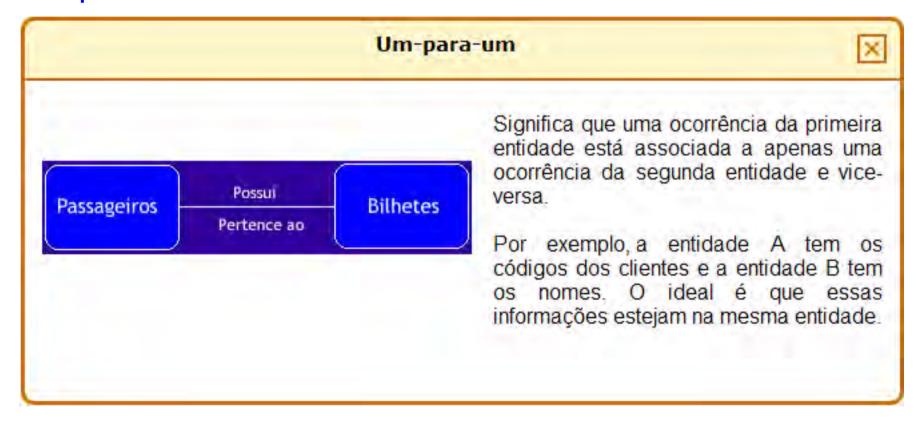
Por exemplo, o relacionamento entre a tabela de funcionários e a de cônjuges deve ser opcional, pois pode haver funcionários solteiros.

Os relacionamentos podem ser de três tipos:

- Um-para-um
- Um-para-muitos
- Muitos-para-muitos

Os relacionamentos podem ser de três tipos:

• Um-para-um



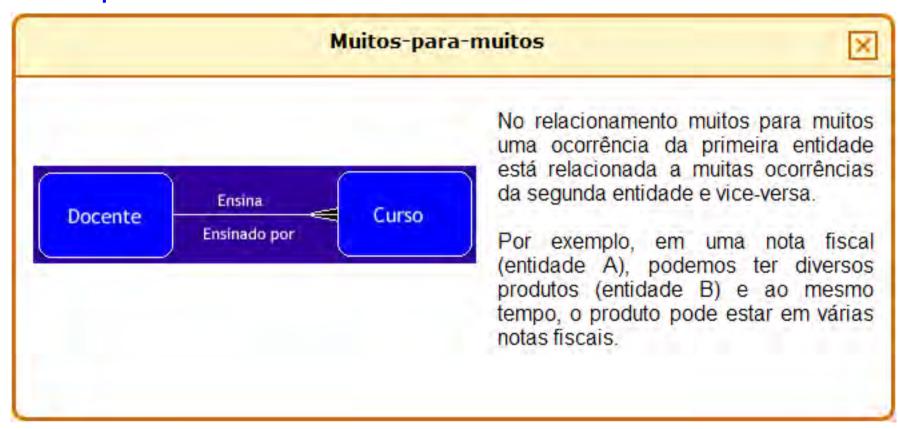
Os relacionamentos podem ser de três tipos:

• Um-para-muitos



Os relacionamentos podem ser de três tipos:

• Muitos-para-muitos



Componente	Representação			
Entidade ou tipo de objeto	Ordem de compra			
Relacionamento	Tem	Tem / Formas		
Cardinalidade	1 N			
Opcionalidade	0 1 (opcional) (obrigatório)	-0-+		
	PETER CHEN	BACHMAN		

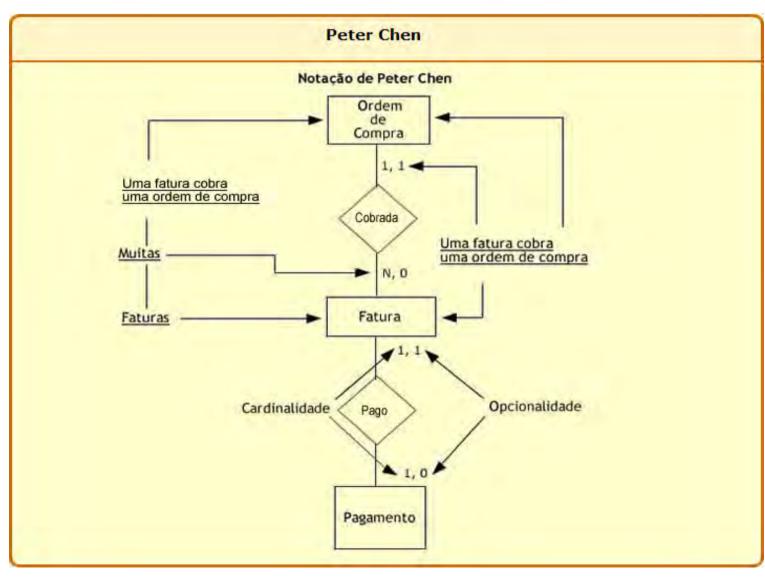
Notação ERD

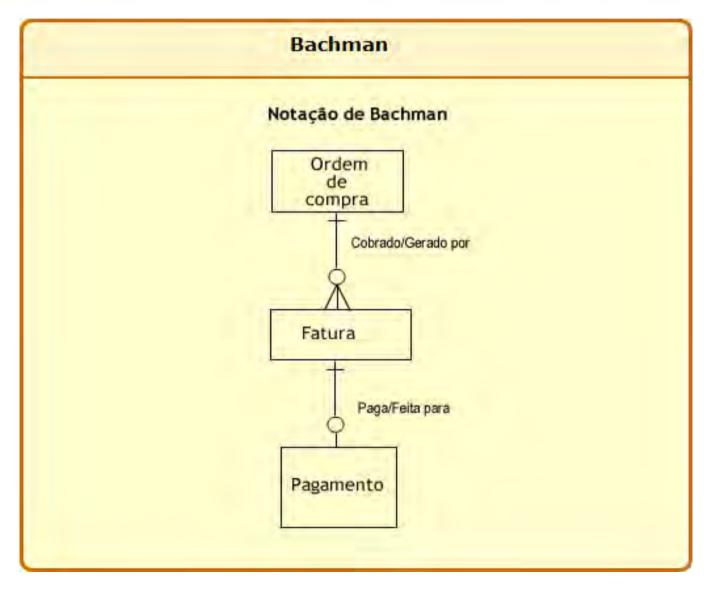
Existem dois tipos de notações:

- Notação de Peter Chen
- Notação de Bachman

Peter Chen e Bachman são os nomes dos inventores da notação.

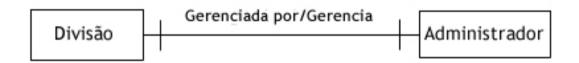
Clique em cada um dos tipos de notação para ver um exemplo.





Abaixo seguem alguns exemplos de diagramas ERD usando notação de Bachman.

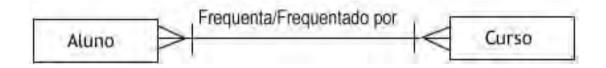
1. Em uma empresa, cada divisão é gerenciada somente por um administrador e cada administrador gerencia apenas uma divisão.



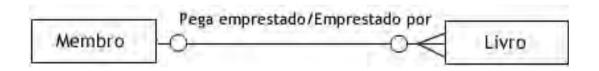
2. Entre as indústrias automobilísticas, cada uma fabrica vários carros, mas determinado carro é fabricado em apenas uma delas.



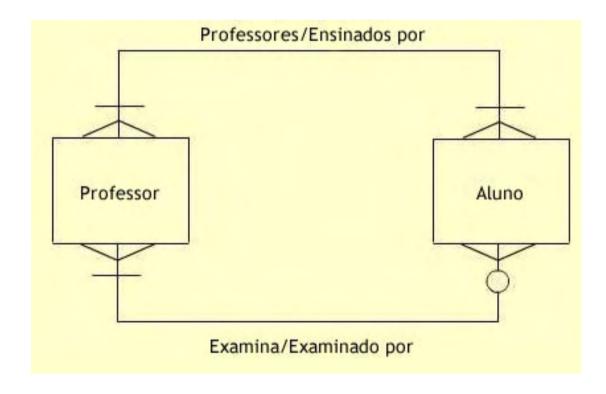
3. Em uma escola, todos os alunos têm muitos cursos e todos os cursos são frequentados por muitos alunos.



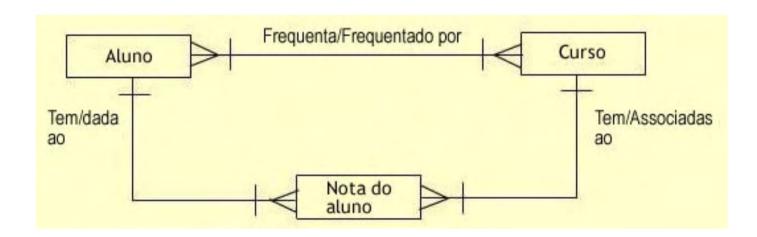
4. Em uma biblioteca, um sócio pode tomar emprestado muitos livros e pode haver livros que não são emprestados por nenhum sócio.



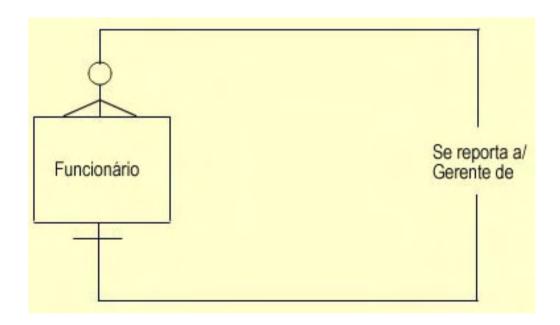
5. Um professor ensina muitos alunos e um aluno é ensinado por muitos professores. Um professor aplica avaliação para muitos alunos e um aluno é avaliado por muitos professores.



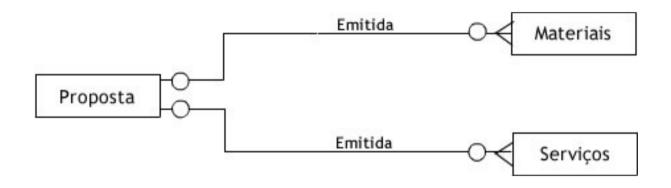
6. Uma extensão do exemplo 3 é de que as notas dos alunos dependem tanto do aluno quanto do curso. Desta forma, este caso é uma entidade associativa.



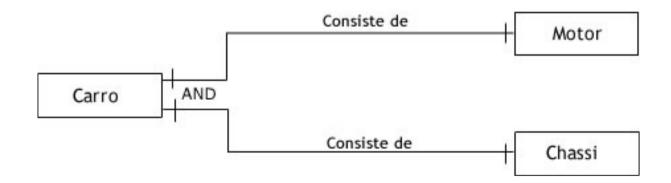
7. Um funcionário pode desempenhar o papel do administrador. Dessa forma, um funcionário se reporta a outro funcionário.

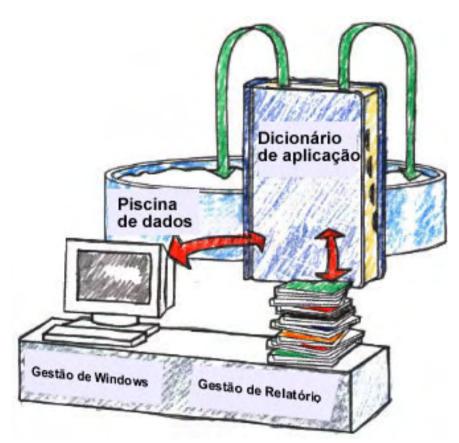


8. Uma proposta é emitida para materiais ou serviços, mas não para ambos.



9. Um carro consiste de um motor e um chassi.





Contém:

- Uma lista organizada de elementos de dados
- Estrutura dos dados
- Fluxos de dados
- Repositório de dados
- Mini especificações de processos primitivos do sistema

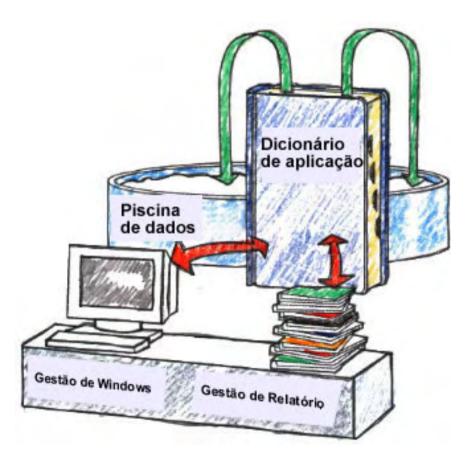
E quaisquer outros detalhes que fornecerão informações úteis sobre o sistema.



Elemento de dados é a parte do dado que não pode ser dividida no contexto atual do sistema. Exemplos:

- Ordem_de_compra_nº
- Nome_do_funcionário
- Taxa_de_juros etc.

Cada elemento de dados é um membro de um domínio. A entrada de dicionário de um elemento de dados deve especificar o domínio.



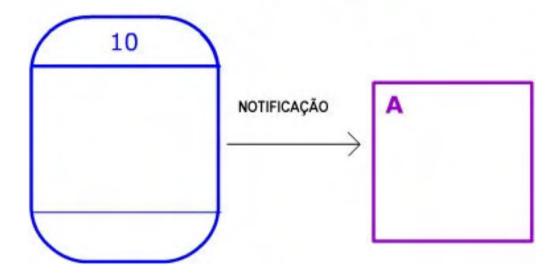
Estrutura de dados é composta de elementos de dados ou outras estruturas de dados. Exemplos:

- Detalhes_do_cliente, que pode ser composto pelo Nome_do_Cliente e Endereço_do_Cliente.
- Endereço_do_Cliente, por sua vez, é uma estrutura.
- Fatura, que pode ser composta de número_da_fatura, detalhes_do_cliente, endereço_de_entrega e detalhes_da_fatura.

Módulo 03

3.4 Dicionário de Dados

Exemplo: 10 - UM FLUXO DE DADOS: NOTIFICAÇÃO



Fluxo de dados é composto de estruturas de dados e/ou elementos de dados. As definições de estruturas dependentes de dados/elementos de dados precedem a definição do fluxo de dados.

Ao se definir o fluxo de dados, os pontos de conexão devem ser mencionados.

Também é útil se incluir o volume de fluxo/freqüência e as taxas de crescimento.

Repositório de dados, como o fluxo de dados, é composto de uma combinação de estruturas de dados e/ou elementos de dados. A descrição é similar aos fluxos de dados.

Os elementos de notação usados no dicionário de dados são os seguintes:

- [nome_do_cônjuge]: indica que nome_do_cônjuge é opcional.
- {nome_do_dependente, relacionamento} * (0 a 15): indica que a estrutura de dados pode ser repetida de 0 a 15 vezes.
- {descrição_da_despesa, nome_da_empresa, cobrança} * (1 a N): indica que a estrutura de dados pode ser repetida de 1 a N onde N não está estabelecido.
- número_de_identidade_do_eleitor/número_da_conta_do_cliente: indica que os dois elementos estarão presentes.

Mini-especificações informam as formas nas quais os fluxos de dados que entram no processo primitivo são transformados em fluxos de dados que deixam o processo.

Apenas o esboço amplo é dado, não os passos detalhados. Elas precisam existir para todo processo primitivo. Inglês estruturado é usado para informar as mini-especificações.

Entrada Pacote de frame Saída Frame recebido

Usuário de Miniespecificação

12 Checar Erros Recebidos

Mini-especificações são elaboradas para transformações de dados no nível de hierarquia mais baixo. O formato das mini-especificações pode ser formato textual livre ou linguagem de programação.

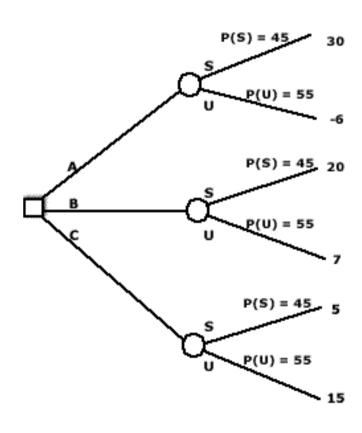




ERD

Uma vez criados o DFD, ERD e o Dicionário de dados, eles devem ser comparados entre si. O DFD e o ERD podem ser criados paralela e independentemente.

- Todo repositório de dados no DFD deve corresponder a pelo menos uma entidade no ERD.
- Devem haver processos no DFD que criam, modificam e deletam ocorrências das entidades no ERD.
- Para todo relacionamento no ERD deve ser utilizado por um processo no DFD.
- Para cada descrição no dicionário de dados, devem haver elementos correspondentes no DFD e no ERD.



Uma árvore de decisão representa decisões complexas sob a forma de uma árvore. Embora seja visualmente atraente, ela pode fugir do controle quando o número e complexidade das decisões aumentam.

Veja um exemplo na próxima página.



Regras para cobrança de energia elétrica são como as que aparecem abaixo:

- Se o medidor estiver "OK", calcule a base de consumo (ou seja, medição).
- Se o medidor aparecer "BAIXO", então verifique se a casa está ocupada.
- Se a casa estiver ocupada, calcule a base de consumo sazonal ou, caso contrário, calcule a base de consumo.
- Se o medidor estiver danificado, calcule com base no uso máximo de energia elétrica possível.

A árvore de decisão correspondente é visualizada na próxima página.



VALORES BINÁRIOS DA TABELA DE DECISÃO

Cliente Domésticos	S	S	N	N
Consumo < 300 unidades por mês	S	N	S	N
Taxa Minima	S	N	N	N
Taxa Especial	N	S	N	N
Dobro da taxa mínima	N	N	S	N
Dobro da taxa especial	N	N	N	S

MULTI-VALORES DA TABELA DE DECISÃO

Cliente	D	D	N	N
Consumo	≥ 300	<300	≥300	<300
Taxa	S	М	25	2M

Existem dois tipos de tabelas de decisão, com valor binário (sim ou não) e valor múltiplo. Veja um exemplo:

Calculo do consumo de energia elétrica com base na classificação do cliente

- Se o cliente usa energia elétrica para fins domésticos e o consumo for inferior a 300 unidades por mês, então cobrar com a taxa mínima.
- Se o cliente doméstico consumir mais de 300 unidades por mês, cobrar a taxa especial.
- Clientes não domésticos são cobrados duplamente em relação aos usuários domésticos (taxas mínimas e especiais são duplas).

VALORES BINÁRIOS DA TABELA DE DECISÃO (Três linhas e duas colunas são adicionadas para lidar com a classe extra de clientes)

Acadêmico	N	N	N	N	S	S
Cliente Doméstico	S	S	N	N	N	N
Consumo < 300 unidades por mês	s	N	s	N	S	N
Taxa Minima	s	N	N	N	N	N
Taxa Especial	N	s	N.	N	N	N
Taxa Minima Dupla	N	N	s	N	N	N
Taxa Especial Dupla	N	N	N	s	N	N
Taxa Concessional	N	N	N	N	S	N
Taxa Concessional Dupla	N	N	N	N	N	S

MULTI-VALORES DA TABELA DE DECISÃO (Apenas duas colunas são adicionados para lidar com a classe extra de clientes)

Cliente	Domestico	Domestico	Não-Doméstico	Não-Doméstico	Acadêmico	Acadêmico
Consumo	≥ 300	<300	≥300	<300	≥300	<300
Taxa	Especial	Minimo	Especial Dupla	Minima Dupla	Concessional Dupla	Concessional

Como as árvores de decisão, as tabelas de decisão de valor binário podem ficar maiores se o número de regras aumentar.

Tabelas de decisão de valor múltiplo possuem uma margem. No exemplo ao lado, adicionamos uma nova classe de clientes, chamada Acadêmico, com as seguintes regras:

 Se o consumo for inferior a 300 unidades por mês, cobrar com a taxa concessional. Caso contrário, cobre duas vezes a taxa concessional.

3.6 Português Estruturado

Para especificar os processos (mini-especificações), usa-se o Inglês estruturado.

Ele consiste de:

- Seqüências de instruções (declarações de ação)
- Decisões (se houver)
- loops (repetir até)
- Caso
- Grupos de instruções

3.6 Português Estruturado

No caso de 'Cobrança', um arquivo mestre é atualizado com a cobrança (ou seja, o numero da conta do consumidor e data da cobrança). Um arquivo de controle também precisa ser atualizado para o 'valor total da cobrança'. Um tratamento similar precisa ser dado a 'Pagamento'.

Inglês normal, livre

Se transação é 'COBRANÇA' então

- Atualize cobrança no arquivo mestre de Contas
- Atualize o valor total da cobrança no arquivo de Controle

Se transação é 'PAGAMENTO' então

- Atualize recebimento no arquivo mestre de Contas
- Atualize o valor total de recebimento no arquivo de Controle

Inglês Estruturado

3.6 Português Estruturado

Se a leitura anterior e a nova leitura forem iguais então:

• Faça uma 'checagem de status'

Se o status for 'parado' então:

• Calcule as cobranças com base no consumo médio

Ou

Outro exemplo de Inglês estruturado.

• Compute a cobrança com base no consumo real

Checagem de status

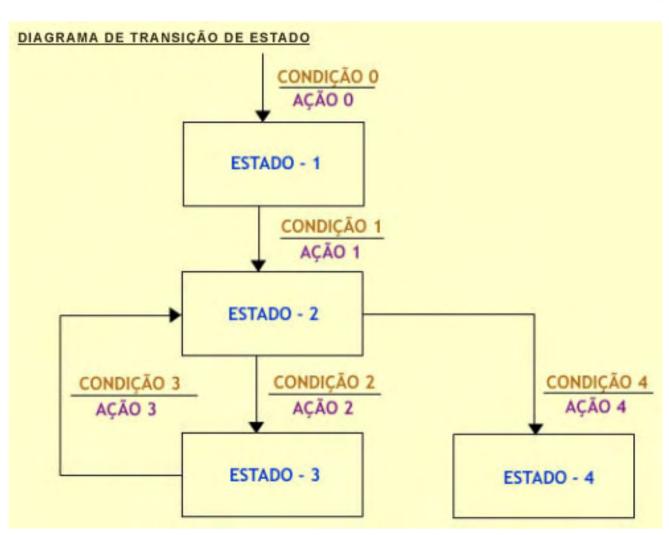
Se o medidor não registrar nenhuma alteração após a troca de qualquer dispositivo elétrico então o

Status do medidor é 'parado'

Ou

Status do medidor é 'OK'

3.7 Diagrama de Transição de Estado



O diagrama de transição de estado é outro diagrama útil.

Ele pode ser usado para modelar as alterações de estado do sistema. Um sistema está em um estado e permanecerá nele até que uma condição ou uma ação o forcem a mudar.

Parabéns!

Você chegou ao final do Módulo 3!

Agora você está apto(a) a continuar este curso.

Siga para o Módulo 4 e tenha um excelente aprendizado!