

## **Aula 00**

*Câmara dos Deputados (Analista  
Legislativo - Informática Legislativa)  
Banco de Dados e BI*

Autor:  
**Thiago Rodrigues Cavalcanti**

28 de Fevereiro de 2023

<b>Conceitos de banco de dados.....</b>	<b>2</b>
<b>Apresentação do Professor .....</b>	<b>2</b>
<b>PARE TUDO! E preste atenção!! .....</b>	<b>4</b>
<b>Motivação para o curso .....</b>	<b>5</b>
<b>Objetivos da aula.....</b>	<b>7</b>
<b>Conceitos de Banco de dados.....</b>	<b>7</b>
<i>Conceitos básicos.....</i>	<i>7</i>
<i>Características da abordagem de BD.....</i>	<i>17</i>
<i>Personagem do ecossistema de BD.....</i>	<i>24</i>
<i>Evolução histórica dos SGBDs.....</i>	<i>28</i>
<i>Classificação dos SGBDs .....</i>	<i>42</i>
<b>Modelo de dados e Arquitetura em três esquemas .....</b>	<b>44</b>
<i>Instâncias x esquemas .....</i>	<i>44</i>
<i>Modelo de Dados.....</i>	<i>47</i>
<i>Arquitetura três esquemas .....</i>	<i>53</i>
<i>Ambiente do Sistema de banco de dados .....</i>	<i>62</i>
<b>Resumo.....</b>	<b>67</b>
<b>Mapa Mental .....</b>	<b>70</b>
<b>Questões comentadas .....</b>	<b>71</b>
<i>Exercícios FGV.....</i>	<i>82</i>
<i>Gabarito.....</i>	<i>88</i>
<b>Considerações finais.....</b>	<b>89</b>
<b>Referências .....</b>	<b>89</b>



  
**THIAGO CAVALCANTI**  
PROFESSOR



## CONCEITOS DE BANCO DE DADOS. APRESENTAÇÃO DO PROFESSOR

Olá senhoras e senhores,

Sejam bem-vindos a mais um curso de Tecnologia da Informação (TI)! Hoje apresentamos o mais completo curso no que se refere Banco de Dados para concursos. Gosto sempre de dizer que é um prazer imenso fazer parte desta equipe de professores do Estratégia Concursos e ter a oportunidade de apresentar um pouco do meu conhecimento e experiência em concursos públicos!

Antes de começar de fato o conteúdo teórico desta aula, vou me apresentar de forma rápida. Meu nome é Thiago, sou casado, pernambucano, tenho três filhos, Vinícius (9 anos), Lucas (*in memoriam*) e Júlia (1 ano). Torço pelo Sport Clube do Recife. Sou cristão. Frequento a IPN – Igreja Presbiteriana Nacional. Me formei em Ciência da Computação pela UFPE. Tenho mestrado em engenharia de software na mesma instituição. Atualmente faço doutorado em economia na UnB.

Frequento academia para manter a forma, mas meu hobby mesmo é pedalar! Decidi vender o carro e viver num desafio intermodal de transporte. Ia para o trabalho de *bike* sempre que possível! Ultimamente tenho usado mais Uber do que a magrela, mais isso é um detalhe! A pergunta é: onde eu trabalho? No Banco Central do Brasil!

Fruto de uma trajetória de dois anos de estudos diários. Aposentei as canetas em 2010. Hoje estou de licença do Banco Central para fazer doutorado que começou em março de 2017. Antes de me licenciar eu trabalhava com análise e modelagem de dados.

Minha mais recente experiência com gestão de dados é parte de uma estratégia profissional de alinhar meu trabalho diário como servidor público com minha carreira paralela de professor e consultor de Banco de Dados (BD) e *Business Intelligence* (BI). A ideia é conseguir me especializar cada vez mais no tema desta nova carreira dentro da TI, que o mercado está denominando de **cientista dos dados (Data scientist)**.

Entre neste universo de professor de concurso há alguns anos. Desde 2012, tenho me dedicado especificamente ao conteúdo de BD e BI. Minhas experiências em cursos presenciais aqui em Brasília e em diversas partes do Brasil, bem como na gravação sistemática de aulas on-line me ajudaram a desenvolver um conteúdo exclusivo para os alunos do Estratégia Concursos.

A ideia é desenvolver um material completo, recheado de questões e com diversas dicas para ajudar você no seu objetivo: **ser aprovado e nomeado!**



Agora gostaria humildemente de fazer um pedido, não deixe de seguir meu perfil no [Instagram®](#) (@profthiagocavalcanti), onde eu publico, sistematicamente, questões comentadas e dicas semanais.

Para facilitar sua vida você pode usar o QR code ao lado para acessar meu perfil no Instagram. Se precisar falar comigo por e-mail, mande mensagem para:

[rcthiago@gmail.com](mailto:rcthiago@gmail.com)



Por fim, e talvez a dica mais importante relacionada a redes sociais, gostaria de apresentar a vocês o meu canal no [Telegram®](https://t.me/profthiagocavalcanti) (https://t.me/profthiagocavalcanti) ... neste canal procuro condensar todas as dicas que apresento nas minhas redes sociais. Na minha opinião, é a melhor forma de acompanhar todas as minhas publicações sem precisar ficar procurando nas redes sociais, lá eu ainda tiro dúvidas (no chat do canal) e interajo diretamente com os alunos. Ou seja, é uma forma de otimizar seus estudos!



Agora que você já me conhece! Vamos seguir em frente com o nosso curso!



## PARE TUDO! E PRESTE ATENÇÃO!!

Hoje eu faço parte de uma equipe **SENSACIONAL** de professores! Depois de muita luta conseguimos reunir **um time** de profissionais extremamente **QUALIFICADO** e sobretudo **COMPROMISSADO** em fazer o melhor pelos alunos. Para tal, criamos um conjunto de ações para nos aproximarmos dos alunos, entendermos suas necessidades e evoluirmos nosso material para um patamar ainda mais diferenciado. São 3 as novidades que gostaria de convidá-lo a conhecer:

//estratégia tech



Nosso podcast alternativo ... livre, descontraído e com dicas rápidas que todo **CANETA PRETA** raiz gosta de ouvir. Já temos alguns episódios disponíveis e vários outros serão gravados nas próximas semanas ... acompanhe em:

<http://anchor.fm/estrategia-tech>



Telegram

a new era of messaging

Nosso grupo do Telegram! É um local onde ouvimos os alunos e trocamos ideias. Está crescendo a cada dia. A regra do grupo é: só vale falar sobre concursos. Lá divulgamos nossas aulas ao vivo e falamos sobre os concursos abertos, expectativas de novos concursos, revisões de véspera ...

[http://t.me/estrategia\\_ti](http://t.me/estrategia_ti)

Instagram



Criamos um perfil no Instagram ... e qual o objetivo? Fazer com que os alunos percam tempo nas redes sociais? Claro que não!! Estamos consolidando diversos posts dos professores! São dicas especiais, um patrimônio que deve ser explorado por todos os concurreiros de TI ...

<http://instagram.com/estrategiaconcursosti>



## MOTIVAÇÃO PARA O CURSO

Preparar esse curso é um desafio! Consolidar de forma amigável o conhecimento de banco de dados, análise de informações ou business Intelligence para concursos não é uma tarefa fácil! Calibrar o nível do teórico associado a uma didática eficiente tem sido minha meta nos últimos anos. Separamos o conteúdo de forma a segmentar e impulsionar seu aprendizado. Para que você entre na primeira aula com um pouco mais de segurança, vou aproveitar para fazer uma rápida apresentação sobre o assunto.



Você já ouviu falar sobre **Data Science** ou **ciência dos dados**? É um conceito relativamente recente que agrupa diversas atividades executadas sobre um conjunto de dados, em especial, sobre grandes conjuntos de dados. Para analisar os dados eles precisam estar **armazenados e organizados** de maneira **convenientes** para os cientistas dos dados. Essa base de dados facilita o trabalho e o entendimento do conteúdo armazenado.

Cientistas de dados são uma nova geração de especialistas em análise que têm habilidades técnicas para resolver problemas complexos e a curiosidade de explorar quais são os problemas que precisam ser resolvidos. A solução desses problemas passa por analisar os dados presentes em um banco de dados. Neste curso veremos o passo-a-passo para construção de um banco de dados.

Nossa primeira aula deve inserir você no universo dos bancos de dados. Um banco pode ser visto como uma estrutura que armazena algo, por exemplo, um banco de leite guarda leite materno para que possa ser reutilizado de forma adequada em momentos posteriores. Um banco de dados guarda dados. Esses dados devem ser controlados de forma adequada. É nesse momento que surge um sistema para “cuidar” do acesso consistente aos dados.

Os sistemas de gerenciamento de banco de dados (SGBDs) contribuem para a disponibilidade de um conjunto de informações para diferentes usuários simultaneamente. É preciso decidir quais dados armazenar, estruturar e manter na base de dados. Para controlar esse sistema e todo o desenvolvimento do projeto e da infraestrutura associada ao sistema de banco de dados várias tarefas têm que ser feitas.

Veremos que existem profissionais dedicados a tarefas específicas. Veremos ainda que a construção de um banco de dados, em especial um banco de dados relacional, passa por



algumas etapas bem definidas. Essas etapas criam modelos de dados ou esquemas que permitem um melhor entendimento da estrutura de dados da organização ao tentar abstrair a complexidade presente no armazenamento físico dos dados.

Todos esses conceitos serão vistos em detalhes nas próximas páginas. Ao final, teremos nossa tradicional lista de exercícios. Espero conseguir contribuir para a sua aprovação. Vamos em frente?!

Teremos muito trabalho! Por isso, montamos um **curso teórico em PDF**, baseado nas mais diversas bancas, em especial na **FGV**, apresentando o conteúdo observando as variadas formas de cobrança do mesmo pelas bancas examinadoras.

Teremos ainda videoaulas que apresentam o conteúdo teórico de forma detalhada para todo o conteúdo deste curso. Caso você não esteja visualizando os vídeos, peço que entre em contato comigo, o mais rápido possível, para que eu possa associá-los as respectivas aulas.

Ao final deste curso, nosso objetivo é garantir que você tenha capacidade e conhecimento para ser aprovado.

**Observação importante:** este curso é protegido por direitos autorais (copyright), nos termos da Lei 9.610/98, que altera, atualiza e consolida a legislação sobre direitos autorais e dá outras providências.

Grupos de rateio e pirataria são clandestinos, violam a lei e prejudicam os professores que elaboram os cursos. Valorize o trabalho de nossa equipe adquirindo os cursos honestamente através do site Estratégia Concursos ;-)

**Observação importante II:** todo o conteúdo deste curso encontra-se completo em nossos textos escritos. As videoaulas visam reforçar o aprendizado, especialmente para aqueles que possuem maior facilidade de aprendizado com vídeos e/ou querem ter mais uma opção didática.

**Agora vamos voltar para a nossa aula. Vamos juntos?**



## OBJETIVOS DA AULA

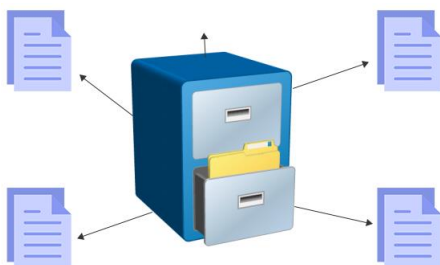
Nossa aula começa com os objetivos. Ao final dela, ao ler o resumo, você deve ter certeza de que fixou os principais conceitos associados aos seguintes objetivos:

- Entender **o que é um banco de dados** e por que sistemas de banco de dados (SBD) e os sistemas de gerenciamento de banco de dados (SGBDs) são úteis.
- Entender a importância da **abstração** na construção de um banco de dados e como essa abstração é desenvolvida por meio de modelos de dados em diferentes níveis.
- Conhecer uma arquitetura genérica para sistemas de banco de dados denominada **ANSI/SPARC**.
- Descrever o ambiente de banco de dados em função dos diversos **componentes**.

Falaremos ainda sobre os **personagens** que se envolvem no funcionamento diário de um banco de dados e traçaremos uma rápida história dos Sistemas de Gerenciamento de Banco de dados.

## CONCEITOS DE BANCO DE DADOS

### CONCEITOS BÁSICOS



Antes de adentrar nos conceitos de banco de dados gostaria de voltar um pouco no tempo. Preciso que você conheça **os sistemas de arquivo**. Você deve ter acesso a um sistema de arquivo do computador ou dispositivo que você está usando para ter acesso a esse conteúdo. Agora imagine a seguinte situação: você trabalha em uma empresa e a lista de clientes está armazenada em um arquivo do Excel.

Você e todos os demais funcionários do setor de vendas (isso mesmo, na nossa história você trabalha no setor de vendas!!) fazem acesso ao mesmo arquivo para incluir e atualizar dados dos clientes. Quando o arquivo começa a ficar sem controle vocês resolvem criar cópias do mesmo para que cada vendedor possa ter **seu próprio cadastro**. Perceba que a solução do problema de acesso simultâneo nos leva a outro problema: **redundância de dados**. Várias cópias **sem um controle centralizado e automatizado** podem ainda ter inconsistência entre os dados armazenados.

Para resolver o problema da **redundância não controlada**, você e seus colegas de trabalho ouvem falar de **uma solução robusta** para o problema de armazenamento de dados. Essa solução passa por manter apenas uma cópia dos dados em um banco de dados e usar um sistema gerenciador para **controlar o acesso**





**concorrente.** Perceba que neste momento a redundância, caso exista, passa a ser controlada por um sistema, de forma automática. Neste sentido, podemos definir os dois tipos de redundância que podem existir entre arquivos ou conjuntos de dados.

ESCLARECENDO!



Há dois tipos de redundância de dados:

**Redundância controlada de dados:** Acontece quando o software tem conhecimento da múltipla representação da informação e garante a sincronização entre as diversas representações.

**Redundância não controlada:** Acontece quando a responsabilidade pela manutenção da sincronia entre as diversas representações de uma informação está com o usuário e não com o software.

No universo tecnológico o que aconteceu em um passado recente teve como motivação a mesma lógica do problema do cadastro de clientes da sua empresa se vendas (Isso mesmo, você ainda está trabalhando lá!). Aplicativos empresariais faziam acesso a arquivos sem um elemento central para controlar o processo, isso criava o caos pois não tínhamos controle da redundância. Eis que surge o **Sistema de Gerenciamento de Banco de Dados (SGBD)** para atuar como um guardião do banco de dados, que substituiu a abordagem de arquivos. Vejamos uma figura para esclarecer esses contextos:

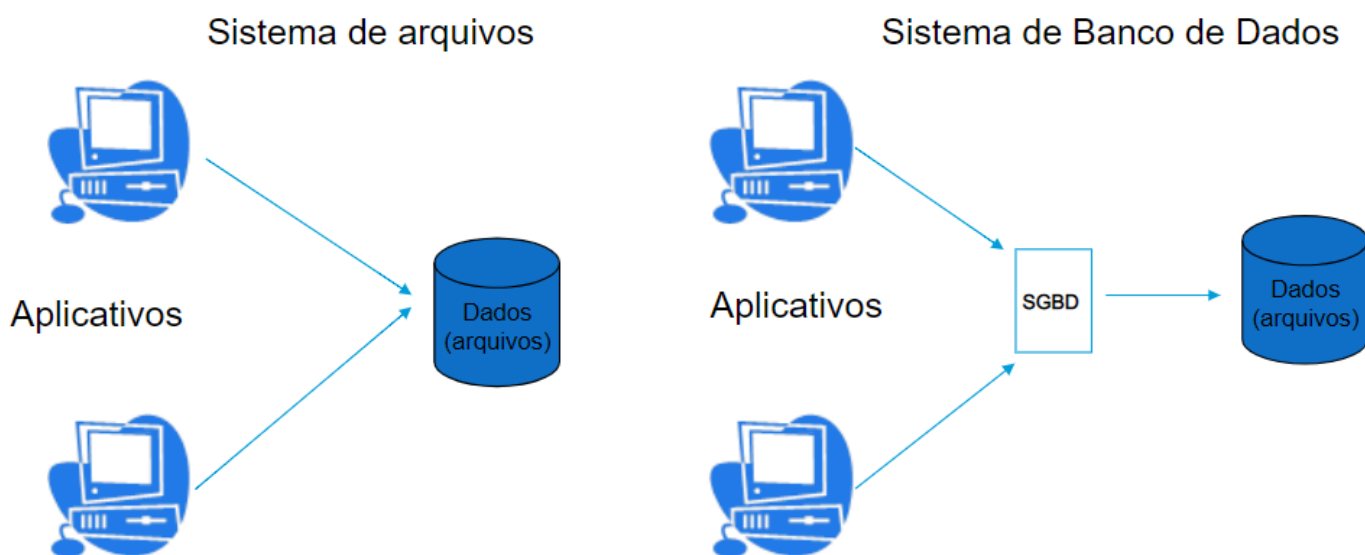


Figura 1 - A figura mostra dois sistemas. O primeiro representa a estrutura de um sistema de arquivos onde os dados são acessados diretamente pelos aplicativos. Na segunda temos um sistema de banco de dados no qual o SGBD aparece entre os dados e os aplicativos.

Perceba que do lado esquerdo não temos o SGBD, já do lado direito existe o elemento. Para entender como e por que esse camarada apareceu aí vamos compreender os conceitos elementares para o entendimento do assunto. Em qualquer ciência, o entendimento completo do seu conteúdo deve se basear nos conceitos fundamentais. Nossa aula começa focada nesses conceitos.



E o primeiro conceito que preciso que você armazene é o de banco de dados, neste instante, você seria capaz de responder a seguinte pergunta:

O que é **banco de dados**?



Você já consegue definir esse termo e suas principais características? Se você ainda não tem o entendimento perfeito do que seria um banco de dados, nosso objetivo agora é construir o conceito. Uma forma tradicional de definir banco de dados é começar entendendo o significado das palavras que compõe o termo: **Banco** e **dados**.

**Banco** tem diversas definições possíveis no dicionário da língua portuguesa. Dentre elas a que melhor se encaixa no nosso contexto é um conjunto organizado e categorizado de objetos, por exemplo, podemos ter um banco de fotografias ou um banco de leite.

**Dados** são fatos conhecidos que podem ser registrados e possuem um significado implícito. Esse conceito, porém, é um pouco amplo e abstrato para nosso intuito. Quando reduzimos o escopo à tecnologia da informação, temos um conceito mais adequado para dado. Ele é a representação física de um evento no tempo e espaço que não agrega fundamento ou significado para quem o sente ou recebe. É, basicamente, um registro!

Imagine que eu fale para você por “32260436, cinco, aprovado, Thiago”. Você vai pensar, o professor está ficando louco! Mas em um banco de dados, esses registros são armazenados e chamados de dados. Para representarem uma informação eles precisam de um contexto associado. Se pensarmos em uma agenda telefônica o número “32260436” pode ser o telefone de Thiago. Aprovado pode ser a sua situação no seu próximo concurso e cinco seria a colocação no referido certame.



Agora que temos o entendimento dos termos vamos partir para a definição do banco de dados.

De forma simples e direta: um **Banco de dados** é uma coleção de dados relacionados. Vejam que essa definição não estabelece a necessidade dos dados serem armazenados em formato digital. Alguns livros trazem o exemplo de uma agenda telefônica de papel como um exemplo bastante didático do conceito de banco de dados.

Esta definição, porém, é considerada muito simplista para alguns autores por não contextualizar o termo. O *Navathe*, por exemplo, cita três propriedades implícitas que contribuem para o entendimento do termo banco de dados (BD). Primeiramente, o BD

representa **algum aspecto do mundo real**, às vezes chamado de **minimundo** ou de **universo de discurso** (*UoD – Universe of Discourse*). As mudanças no minimundo devem ser refletidas no banco de dados.

A **segunda** característica implícita diz que a **coleção de dados é logicamente coerente** com algum significado inerente. Uma variedade aleatória de dados **não** pode ser chamada de banco de dados. Um banco de dados pode armazenar as informações de uma empresa, uma faculdade ou um órgão do setor público. Veja que essas informações estão **dentro de um contexto**, sendo, logicamente coerente.

A **terceira** propriedade afirma que um banco de dados é construído e populado com dados para uma **finalidade específica**. Ele possui um grupo de **usuários** bem definido e algumas aplicações, previamente concebidas, sobre as quais esses usuários interessados fazem acesso aos dados. Não adianta você criar um repositório sem propósito! Ele tem que atender alguma necessidade de acesso ao conjunto dos dados.

Vamos voltar ao nosso exemplo físico da agenda telefônica, você consegue visualizar as propriedades definidas pelo Navathe neste banco de dados. Qual o aspecto do mundo real que ela representa? Os dados estão logicamente relacionados? Possuem uma finalidade específica? Acredito que sim! Tente responder a essas perguntas mentalmente antes de seguir em frente.



## EXEMPLIFICANDO



**Respondendo** ... a agenda telefônica descreve um aspecto do mundo real, por exemplo, a agenda que seu pai armazena os contatos dos amigos e familiares. Esses dados são logicamente relacionados, geralmente, em ordem alfabética, cada letra apresenta os contatos com números de telefone, endereço e

outras informações. Por fim, existe um propósito para a existência da agenda: todos da casa podem ter um acesso rápido aos contatos familiares. Seu Zacarias (painho) possui até hoje uma agenda telefônica ao lado aparelho ... funciona muito bem!

Ok!! As três propriedades definidas pelo Navathe começam a estruturar os conceitos na sua cabeça. Vamos consolidar o conceito de banco de dados apresentando mais algumas definições presentes na literatura:



O banco de dados, por si só, pode ser considerado como o equivalente eletrônico de um armário de arquivamento; ou seja, ele é um repositório ou recipiente para uma **coleção de arquivos de dados computadorizados**. – C J Date

Banco de dados é um conjunto de dados integrados que tem por objetivo **atender a uma comunidade de usuários** - Carlos Heuser.

Banco de dados é um conjunto de dados **estruturados** que são confiáveis, coerentes e compartilhados por usuários que têm necessidades de informações diferentes. - Silberchatz

Acho que você já entendeu o conceito de banco de dados! Na lista acima, você conheceu todas as definições que podem aparecer na sua prova. :)

Vamos agora entender a diferença entre **banco de dados**, **sistemas de gerenciamento de banco de dados (SGBD)** e **sistemas de banco de dados (SBD)**. São três conceitos diferentes para os autores dos livros teóricos sobre o assunto. Para entender essas diferenças peço que você observe a figura a seguir:

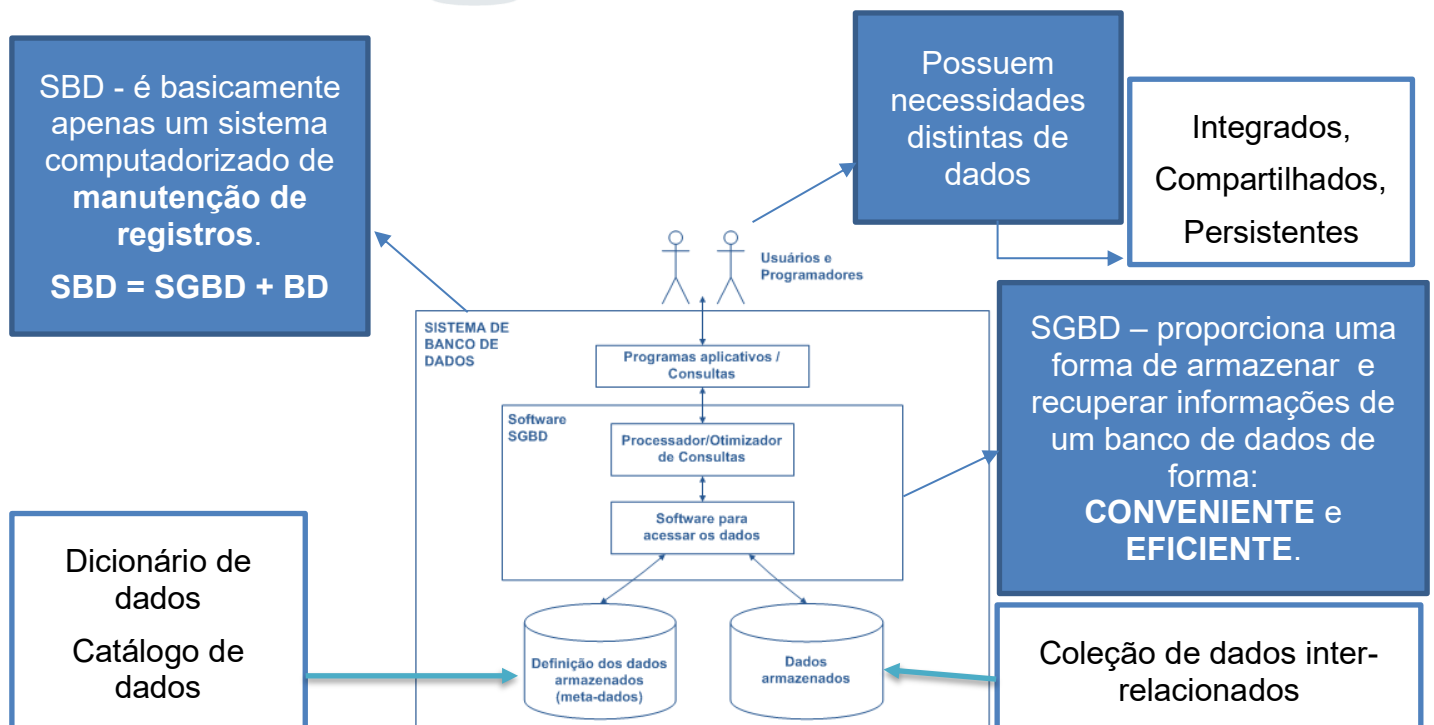


Figura 2 - Visão geral do relacionamento entre banco de dados e SGBD

Seguindo o fluxo de acesso aos dados, podemos observar que os **usuários e programadores** se comunicam com o sistema de banco de dados. Este, por sua vez, faz acesso ao sistema de gerenciamento do banco de dados. O SGBD usa as informações presentes nos bancos de dados, representados pelos cilindros da figura acima, para ter acesso aos dados armazenados.



Um **Sistema de Gerenciamento de Banco de Dados (SGBD)** é um conjunto de programas que permitem armazenar, modificar e extrair informações de um banco de dados. Seu **principal objetivo** é proporcionar um ambiente tanto conveniente quanto eficiente para a **recuperação e armazenamento** das informações do banco de dados.

Contudo, os SGBDs não se restringem apenas a manipulação dos dados. Eles fornecem uma variedade de programas com diferentes funcionalidades.



Figura 3 - Principais funcionalidades de um SGBD.

A figura acima apresenta algumas funcionalidades dos SGBDs. Cada espaço está preenchido com alguma funcionalidade que faz parte do escopo de um software de SGBD. Não se preocupe se você não tiver ideia do que está presente em cada espaço, ao longo desta aula ou do nosso curso, quando você começar a entender melhor o assunto, você perceberá a presença deles dentro do contexto de um SGBD. Vamos em frente!

Outro aspecto interessante sobre banco de dados são as descrições ou definições dos objetos, pense em uma tabela do modelo relacional. Para termos acesso a uma tabela precisamos conhecer sua estrutura, começando pelo seu nome e das suas colunas. Essa tarefa envolve especificar os **tipos, estruturas e restrições** dos dados a serem armazenados.

A definição ou informação descritiva do banco de dados também é armazenada pelo SGBD numa estrutura conhecida como **catálogo** ou **dicionário de dados**, que armazena os chamados de **metadados** dos objetos. Os metadados carregam consigo um significado. Uma coluna de uma tabela pode ser definida por um tipo de dados, inteiro, uma restrição, *not null*.

É possível ainda fazer o **compartilhamento dos dados** entre diversos usuários e programas, possibilitando o acesso ao banco de dados de forma simultânea. Outras funções importantes também são providas como **proteção** do sistema **contra defeitos** de hardware e software, feitos por meio de redundância ou replicação, e **proteção** de segurança **contra acesso** não autorizados ou maliciosos.

Outros aspectos interessantes estão relacionados com o controle de transações, recuperação após falha, otimização de consultas ou do próprio SGBD, auditoria por meio de logs de sistema, enfim, são várias as **funcionalidades providas** pelos softwares presentes em um **SGBD**. Vamos agora definir o próximo conceito: sistema de banco de dados!

O **sistema de banco de dados (SBD)** é considerado a união entre o banco de dados e o sistema de gerenciamento de banco de dados. Em outras palavras, consiste em uma coleção de dados inter-relacionados e de um conjunto de programas para acessá-los. Partindo da figura que apresentamos anteriormente conseguimos construir a seguinte fórmula:

$$\text{SBD} = \text{BD} + \text{SGBD} + (\text{Programa de aplicação/consulta})$$

Antes de continuarmos construindo nosso arcabouço teórico sobre o assunto vamos resolver algumas questões de provas passadas.



Antes de continuarmos construindo nosso arcabouço teórico sobre o assunto vamos resolver algumas questões de provas passadas.

**1. Ano: 2018 Banca: FGV Órgão: MPE-AL Prova: FGV - 2018 - MPE-AL - Técnico do Ministério Público - Geral**

O conjunto de programas responsável pelo gerenciamento de uma base de dados e que, entre outras funções, suporta uma linguagem de consulta, gera relatórios e disponibiliza uma interface para que os seus clientes possam incluir, alterar ou consultar dados, é chamado de

A Banco de Dados Relacional (BDR).

B Dicionário de Dados (DD).

C Modelo Entidade Relacionamento (MER).

D Sistema de Suporte à Decisão (SSD).

E Sistema Gerenciador de Bancos de Dados (SGBD).



**Comentário:** Perceba que o conjunto de programas é o SGBD, presente na alternativa E. O banco de dados relacional é um conjunto de dados interrelacionados estruturados de acordo com o modelo relacional (será visto ainda nesta aula). Dicionários de dados são os metadados ou catálogo de dados que apresenta a descrição dos dados armazenados. Modelo Entidade Relacionamento é um modelo de dados de alto nível usado para descrever quais os dados serão armazenados no banco de dados usando elementos gráficos. Já os sistemas de suporte a decisão procuram tratar os dados para auxiliar na tomada de decisão.

**Gabarito: E.**

## 2. Gestor da Informação (Curitiba)/2019

O principal objetivo de um Sistema Gerenciador de Banco de Dados (SGBD) é:

- a) criar a infraestrutura para a construção de um data warehouse.
- b) armazenar e recuperar os dados de forma conveniente e eficiente.
- c) organizar os dados para suportar operações de OLAP.
- d) possibilitar a armazenagem distribuída dos dados.
- e) facilitar a implementação de tecnologias de armazenagem em nuvem.

**Comentário:** Vejam que, pelas definições acima apresentada um SGBD é definido como um sistema cujo objetivo é armazenar e recuperar dados de forma eficiente. Analisando as demais alternativas, elas não fazem sentido para definirem o principal objetivo do SGBD. Vamos deixar para apresentar os conceitos referente as palavras “data warehouse”, OLAP e computação na nuvem em outro momento, para não atrapalhar seu processo de aprendizado.

Mas professor, eu queria aprender agora ... ok! Então vamos nessa.

**Data Warehouse (DW)** é um repositório de dados utilizado para guardar dados analíticos. **OLAP** é um conjunto de ferramentas e definições que permitem acessar e trabalhar com os dados no DW.

O armazenamento distribuído dos dados é uma possibilidade dentre das funções do SGBD, mas não é seu principal objetivo.

A **computação em nuvem** é o fornecimento de serviços de computação, incluindo servidores, armazenamento, bancos de dados, rede, software, análise e inteligência, pela Internet (“a nuvem”) para oferecer inovações mais rápidas, recursos flexíveis e economias de escala. Você normalmente paga apenas pelos serviços de nuvem que usa, ajudando a reduzir os custos operacionais, a executar sua infraestrutura com mais eficiência e a escalonar conforme as necessidades da sua empresa mudam.

Assim, reforçando a nossa resposta encontra-se na alternativa B.

**Gabarito: B.**

## 3. Ano: 2019 - Prefeitura de Jataí - GO - Analista de Tecnologia da Informação



Com relação aos conceitos e às definições de banco de dados, assinale a alternativa correta.

A Um banco de dados não é formado por um conjunto de arquivos, mas sim por um conjunto de dados com as mesmas características.

B Um banco de dados é um conjunto de dados organizados, com o objetivo de armazenamento persistente dos dados, que possui mecanismos de manipulação e recuperação de informações.

C Um banco de dados é um conjunto integrado de dados não relacionados logicamente.

D A melhor definição para banco de dados é que ele é uma representação estática, visto que os dados não podem sofrer alterações temporais.

E O banco de dados é uma estrutura de compartilhamento parcial, ou seja, os dados existentes em um banco de dados não podem ser compartilhados por várias pessoas; apenas uma pessoa por vez pode ter acesso ao banco de dados.

**Comentário:** Vamos comentar cada uma das alternativas. (A) Todo banco de dados digital, em última instância, é formado por um conjunto de arquivos. (B) Perfeita a definição!! Perceba que ele disse que o banco de dados é o conjunto de dados organizados e que possui mecanismos de manipulação, esses são parte do SGBD que possui outras funcionalidades além da manipulação dos dados, como backup e controle de acesso. (C) O banco de dados é um conjunto **logicamente relacionados**. (D) O banco de dados possui uma representação dinâmica, tanto os dados, quanto os modelos podem ser ajustados para se adaptarem a mudanças no mundo que eles descrevem. (E) Os dados do banco de dados são compartilhados entre diferentes grupos de usuários. No geral, cada grupo possui a visão de apenas parte do banco de dados.

**Gabarito: B**

#### 4. Ano: 2016 Órgão: TCE-SC Prova: Auditor Fiscal de Controle Externo - Informática

Com relação aos bancos de dados relacionais, julgue o próximo item.

O catálogo de um sistema de gerenciamento de banco de dados relacional armazena a descrição da estrutura do banco de dados e contém informações a respeito de cada arquivo, do tipo e formato de armazenamento de cada item de dado e das restrições relativas aos dados.

**Comentário:** Perceba que a definição acima está de acordo com o termo dicionários de dados, catálogo de dados ou metadados presentes em um sistema de banco de dados. Lembre-se que essa separação entre a descrição dos dados e os dados propriamente dito é uma das características relevantes que foram apresentadas na evolução de sistemas de arquivos para a **abordagem de banco de dados**. Sendo assim, podemos afirmar que a questão está correta!

**Gabarito: C.**

#### 5. Ano: 2010 Órgão: Banco da Amazônia Prova: Técnico Científico - Tecnologia da Informação





O dicionário de dados é uma das principais ferramentas para a administração dos dados corporativos. Por meio da engenharia reversa, pode-se armazenar os modelos de dados, as estruturas de dados, seus relacionamentos e toda a documentação necessária para garantir facilidade na localização e manipulação dos dados. Acerca dos papéis do administrador de dados (AD) e dos dicionários de dados, julgue os itens a seguir.

[1] O dicionário de dados é considerado um subconjunto das funções de um catálogo de sistema.

[2] O catálogo do sistema é um repositório com função de armazenar as definições dos esquemas dos bancos de dados.

**Comentário:** É importante lembrar que existe uma hierarquia entre os objetos ou elementos em um dicionário de dados. Um dicionário de dados possui a descrição dos esquemas ou catálogo de sistemas. Cada catálogo deve conter a descrição dos objetos que fazem parte do contexto de um sistema, como tabelas, visões e domínios. Dentro das definições das tabelas temos as descrições dos atributos e restrições de integridades dos dados.

Assim, ao analisar as alternativas acima, podemos inferir que a alternativa [1] encontra-se incorreta, já a afirmação [2] está certa!

**Gabarito: E C.**

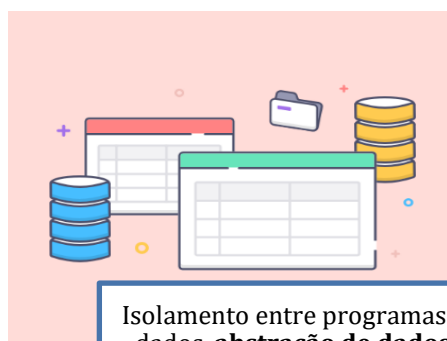


## CARACTERÍSTICAS DA ABORDAGEM DE BD

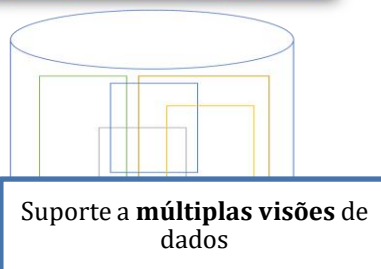
Segundo Navathe, são quatro, as principais características da abordagem de banco de dados que a fazem sobressair em relação às abordagens de processamento de arquivo.



Natureza autodescritiva de um sistema de banco de dados



Isolamento entre programas e dados, **abstração de dados**



Suporte a **múltiplas visões** de dados



**Compartilhamento** de dados e processamento de transação multiusuário.

Esses esforços visam **reduzir a redundância** o que implica em reduzir o desperdício no espaço de armazenamento e os esforços para manter os dados comuns ou duplicados atualizados. Tudo realizado por meio de um único repositório!

Vejam que a lista acima pode ser caracterizada como uma enumeração e, como eu sempre digo, listas fazem parte do rol de questões de prova de concurso. Seja qual for a matéria, sempre gaste um pouco do seu tempo lendo, mais de uma vez, cada uma das listas pertencentes aos assuntos. A verdade é: não importa o grau de relevância dentro do assunto, um examinador preguiçoso sempre está propício a utilizar deste artifício ao elaborar uma questão.

A primeira característica listada pelo Navathe é conhecida por nós como **catálogo do SGBD, dicionário de dados ou metadados**. Esta propriedade permite ao SGBD gravar as definições das suas estruturas e restrições. E, quais são as descrições que podem ser gravadas? Descrição de tabelas, tamanho do campo, tipo dos dados, propriedade de ser nulo ou não, valores default, restrições de integridade, entre outros. Para facilitar sua

visualização pense numa definição de uma tabela em SQL<sup>1</sup>. Veja o exemplo a seguir e observe algumas dessas descrições.

```
CREATE TABLE EMPREGADO (  
    ID_EMPREGADO INT PRIMARY KEY,  
    FK_ID_SUPERVISOR INT,  
    PRIMEIRO_NOME VARCHAR2(100) NOT NULL,  
    ULTIMO_NOME VARCHAR2(100) NOT NULL,  
    EMAIL VARCHAR2(100) NOT NULL,  
    DATA_NASC_FUNC DATE NOT NULL,  
    DATA_INICIO_FUN DATE DEFAULT SYSDATE,  
    DATA_FIM_FUN DATE DEFAULT NULL,  
    CONSTRAINT FK_SUP FOREIGN KEY EMPREGADO (ID_EMPREGADO)  
);
```

A próxima característica é uma decorrência da anterior. A partir do momento em que temos um dicionário de dados, é possível excluir da estrutura dos programas a definição dos dados presentes nos mesmos. Agora isolados, dados e aplicações, criam um conceito chamado **independência de dados do programa**. Este só é possível por conta da **abstração de dados**. A abstração de dados permite a criação de diferentes níveis de modelos. Cada modelo revela a estrutura dos dados de uma forma específica.

O suporte a **múltiplas visões** parte do princípio de que diferentes usuários têm diferentes necessidades sobre os dados. Se pensarmos em SQL, uma VIEW representa um subconjunto de informações referentes a uma ou mais tabelas (ou até a nenhuma tabela). Do ponto de vista mais abstrato, uma visão é a parte do banco de dados ao qual um usuário ou grupo de usuários tem acesso. Porém, existe a possibilidade dessa visão conter um **dado virtual** que é derivado das informações armazenadas. Imagine, por exemplo, a idade calculada a partir da data de nascimento.

Quando falamos de suporte a **múltiplos usuários** queremos, basicamente, permitir que diferentes usuários acessem o banco de dados ao mesmo tempo. Para garantir que isso ocorra é preciso que o SGBD forneça um mecanismo de **controle de concorrência**. As transações efetuadas devem levar o sistema a um estado válido (C), não ter conhecimento umas das outras (I), serem executadas sempre por completo (ou não serem executadas) (A) e, uma vez gravadas na base, devem persistir ao longo do tempo (D).

Você deve estar se perguntando, o que são essas letras ao lado das características das transações citadas anteriormente? Elas se referem às características de uma transação: **atomicidade, consistência, isolamento e durabilidade**. Elas formam uma sigla conhecida como ACID e são bastante conhecidas dentro do modelo de dados relacional/transacional.

---

<sup>1</sup> SQL é uma linguagem declarativa usada para enviar comandos aos sistemas de gerenciamento de banco de dados. Esses comandos permitem a criação de tabelas e a manipulação dos dados armazenados. Na criação da tabela podemos observar a definição dos tipos de dados (VARCHAR2, DATE, NUMBER), bem como das restrições de integridade (NOT NULL, DEFAULT).





Acabamos de tratar das características que o Navathe utiliza para diferenciar sistemas de arquivo dos sistemas de banco de dados. Vamos agora listar as características descritas pelo Date e pelo Silberschatz. Date chama de benefícios da abordagem de banco de dados. Quais sejam:

1. O dado pode ser compartilhado
2. A redundância pode ser reduzida
3. Inconsistências podem ser evitadas
4. Pode-se utilizar o suporte a transações
5. A integridade pode ser mantida
6. A segurança pode ser aperfeiçoada
7. Requisitos conflitantes podem ser balanceados
8. Padrões podem ser utilizados

Já Abraham Silberschatz trata das desvantagens de se utilizar um **sistema de arquivo**:

1. Redundância e inconsistência dos dados
2. Dificuldade de acesso a dados
3. Isolamento dos dados



4. Problemas de integridade
5. Problemas de atomicidade
6. Anomalias de acesso concorrente
7. Problemas de segurança

Lembrem-se, não precisamos decorar todas essas listas, apenas tomar conhecimento da sua existência, pois fazem parte do contexto. Elas procuram sempre expor as características que diferenciam os sistemas de arquivos dos sistemas de banco de dados.

Vejamos mais algumas questões que tratam do assunto.



## 6. FGV - Analista de Tecnologia da Informação (BANESTES)/Suporte e Infraestrutura/2018

As propriedades usualmente requeridas para transações em bancos de dados são identificadas pela sigla ACID.

Essas propriedades são:

- A assincronia, concorrência, integridade, durabilidade;
- B assincronia, consistência, isolamento, distribuição;
- C atomicidade, concorrência, integridade, durabilidade;
- D atomicidade, conformidade, independência, distribuição;
- E atomicidade, consistência, isolamento, durabilidade.

**Comentário:** Seguindo a lista de propriedade ou características de uma transação que abamos de conhecer, podemos marcar a nossa resposta na alternativa E.

**Gabarito: E**

## 7. Ano: 2019 - CREA-TO - Analista de Sistemas

No que diz respeito a banco de dados, julgue o item.

Em um banco de dados, é necessária a abstração dos dados, de tal forma que o usuário não se importe com a forma como eles estão armazenados.

**Comentário:** Quando falamos em abstração dos dados, estamos retirando dos usuários a visão do armazenamento físico e da sua complexidade. A divisão dos discos rígidos em setores e a organização dos arquivos nos SGBDs são abstraídas por meio dos diversos níveis de modelos de dados.



**8. Ano: 2018 Órgão: TCE-PB Cargo: Auditor de Constas Públicas Questão: 97**

A respeito de SGBDs, assinale a opção correta.

A Um SGBD, por definição, não é flexível, dada a dificuldade de mudar a estrutura dos dados quando os requisitos mudam.

B Um SGBD é um software que não prevê as funções de definição, recuperação e alteração de dados, sendo essa tarefa a função básica de um sistema de banco de dados.

C A consistência de dados é o princípio que determina a manutenção de determinado dado em vários arquivos diferentes.

D Conforme o princípio da atomicidade, caso ocorra erro em determinada transação, todo o conjunto a ela relacionado será desfeito até o retorno ao estado inicial, como se a transação nunca tivesse sido executada.

E O controle de concorrência é o princípio que garante e permite a manipulação, no mesmo momento, de um mesmo dado por mais de uma pessoa ou um sistema.

**Comentário:** Vamos analisar as alternativas acima. Elas são relevantes para consolidar nosso conhecimento sobre o assunto. Começando pela alternativa A, o erro aparece quando o examinador afirmar que os SGBDs não são flexíveis. Lembre-se que um SGBD veio solucionar um problema de replicação dos dados em diversos arquivos distintos. Agora, com uma fonte única dos dados, as alterações dos mesmos são centralizadas.

Já a alternativa B nos remonta as funcionalidades do SGBD. O que podemos fazer com tal sistema? É possível, primariamente, armazenar e manipular dados, para tal, é necessário descrever as estruturas das tabelas. Todos os SGBDs possuem estrutura para construção do banco de dados, usando uma linguagem como SQL. Sendo assim, a alternativa B também está incorreta.

As próximas duas alternativas tratam das propriedades das transações, mais especificamente da consistência, que tem por objetivo levar o banco de dados de um estado válido para outro estado consistente, e da atomicidade, que parte do princípio que uma transação é composta por vários comandos de modificação da base de dados e que estes comandos devem ser executados em conjunto completamente ou não serem executados. Desta forma, para garantir a atomicidade, caso uma transação falhe todas as operações já efetivadas precisam ser desfeitas. Desta forma, a alternativa C está incorreta e a alternativa D é a nossa resposta.

Por fim, o controle de concorrência é um mecanismo que permite que apenas um usuário consiga modificar um dado do sistema em um determinado momento. É importante entender que é possível várias pessoas terem acesso aos dados para leitura. Como se todos estivessem lendo um livro. Contudo, na hora de escrever, só existe uma caneta e, se alguma pessoa estiver de posse da caneta, apenas ela poderá fazer alterações na base.



### 9. Ano: 2019 - UFPB - Analista de Tecnologia da Informação

Os bancos de dados estão implícitos na vida da sociedade moderna. Assinale a alternativa que NÃO apresenta uma implicação adicional do uso de banco de dados.

- A Economias de escalas.
- B Disponibilidade de informações atualizadas.
- C Desenvolvimento de novos dispositivos.
- D Flexibilidade.
- E Tempo reduzido para o desenvolvimento de aplicações.

**Comentário:** Se pensarmos em dispositivos como componente de hardware, percebemos que a utilização de um banco de dados não auxilia no desenvolvimento de um novo dispositivo. As demais alternativas apresentam benefícios da utilização da abordagem de banco de dados. A economia de escala refere-se a possibilidade expansão da quantidade de usuários do banco a um custo relativamente baixo. As informações centralizadas com redundância controlada permitem a disponibilização atualizadas das mesmas. Quando você pensa no desenvolvimento de um novo sistema, o fato das informações estarem isoladas das aplicações facilita a vida dos desenvolvedores, reduzindo o tempo para construção das aplicações. Reforçando, temos nossa resposta na alternativa C.

Gabarito: C

### 10. ANO: 2014 ÓRGÃO: TJ-AP PROVA: ANALISTA JUDICIÁRIO - BANCO DE DADOS - DBA

A redundância controlada de dados em um sistema de banco de dados ocorre quando

A um programa está ciente da múltipla representação de uma dada informação e garante o sincronismo entre as diversas representações.

B a responsabilidade pela manutenção do sincronismo entre as múltiplas representações de uma dada informação é compartilhada entre o programa e o usuário.

C os dados mais importantes são duplicados a cada backup do sistema, visando aumentar a garantia da recuperação da informação em caso de problemas.

D a responsabilidade pela manutenção do sincronismo entre as múltiplas representações de uma dada informação é do usuário.

E um programa está ciente da múltipla representação de uma dada informação, mas não garante o sincronismo entre as diversas representações.

**Comentário:** Nesta questão vamos entender o que está sendo cobrado antes de avaliarmos as alternativas. Em primeiro lugar, precisamos lembrar que o SGBD vai



controlar a redundância dentro do banco de dados. Esse controle é feito quando diferentes usuários compartilham a mesma informação.

Suponha uma tabela de endereços de clientes em um banco comercial. Todos os setores do banco podem fazer uso desta informação. A área de investimento pode mandar o extrato das suas aplicações, o setor de cartão de crédito pode enviar sua fatura e o relacionamento com o cliente pode te enviar um cartão de feliz aniversário. A importância deste ponto único de contato é verificada quando a cliente muda de endereço, a atualização da tabela vai ser efetiva para os diferentes usuários da informação.

Agora vamos analisar as alternativas, ao ler cada uma delas, podemos observar que no SGBD um programa deve estar ciente da múltipla representação de uma dada informação e garantir o sincronismo entre as diversas representações. Essa descrição é a presentes na alternativa A que é a nossa resposta.

**Gabarito: A.**

## 11. CETREDE - Fiscal Ambiental (Pref Itaitinga)/2019

Antigamente, os bancos de dados eram manuais. Atualmente com a difusão tecnológica a grande maioria dos bancos de dados são digitais. Eles compõem um conjunto estruturado de dados que obedecem a um modelo de dados e são armazenados em formato digital – em um hardware (geralmente, um servidor da rede – storage). NÃO é vantagem de um banco de dados,

- a) maior volume de dados armazenado.
- b) ocupar maior espaço físico.
- c) maior velocidade de acesso aos dados.
- d) permite o compartilhamento de informações.
- e) persistência dos dados: depois de muitos anos, ainda podem ser acessados.

Comentário: Sabemos que um **banco de dados** é uma **coleção de dados** sobre pessoas, coisas, lugares. Esses dados se relacionam entre si de forma que geram informação, ou seja, algo que tenha sentido para seus usuários. No passado, os bancos de dados eram manuais. Lembrem-se do exemplo da lista telefônica do Seu Zacarias.

Com o passar do tempo, a tecnologia foi avançando e foram surgindo bancos de dados digitais, que armazenam os dados em máquinas. Hoje em dia, a grande maioria das empresas em geral utilizam um sistema que registra de forma digital as suas operações.

Vamos analisar cada uma das alternativas:

**a) ERRADA.** Essa é **uma vantagem** de um banco de dados digital. Ele consegue armazenar um volume muito maior do que os bancos de dados manuais, que utilizavam papéis ou outros meios para armazenar as informações de forma manual.





**b) CORRETA.** Essa **não** é **uma vantagem** dos bancos de dados digitais. Na verdade, eles ocupam um espaço físico menor. Isso ocorre porque a organização dos dados nos arquivos de banco de dados é mais eficiente, e porque a redundância dos dados é reduzida.

**c) ERRADA.** Essa é **uma vantagem** dos bancos de dados digitais. Através de linguagens de consulta, o poder de processamento dos computadores permite acessar os dados de forma muito mais rápida que uma busca realizada manualmente.

**d) ERRADA.** Essa é **outra vantagem** dos bancos de dados. Com a internet, por exemplo, é possível compartilhar informações com outros usuários em qualquer lugar do mundo por meio de bancos de dados distribuídos.

**e) ERRADA.** Mais uma vantagem dos bancos de dados. Um caderno, por exemplo, pode ter suas páginas deterioradas com o tempo ou a tinta da caneta pode apagar. As mídias digitais, entretanto, permitem que a informação esteja acessível por quanto tempo for necessário. Claro que, para isso, devem ser utilizadas técnicas de backup e redundância dos dados.

Desta forma, podemos encontrar a nossa resposta para essa questão na alternativa **B**.

**Gabarito: B.**

## PERSONAGEM DO ECOSISTEMA DE BD

Quando tratamos de grandes organizações, as atividades relacionadas a banco de dados devem ser compartilhadas entre diferentes pessoas. Trataremos agora dos dois principais papéis dentro desse processo: o administrador de banco de dados (DBA) e o administrador de dados (AD).

Só para termos uma ideia, algumas empresas do setor bancário chegam a ter algumas dezenas de ADs dentro da organização. O Bradesco tem por volta de 40 ADs. Vamos então começar falando um pouco sobre esse perfil de trabalhador especializado em banco de dados.

O AD é a pessoa que toma as decisões estratégicas e de normas com relação aos dados da empresa. Os **administradores de dados** também podem ser conhecidos por projetista de dados. Suas tarefas são realizadas principalmente antes do banco de dados ser realmente implementado e/ou populado.

Eles são responsáveis por identificar os dados a serem armazenados e escolher estruturas apropriadas para representar esses dados. Para isso precisam se comunicar com todos os potenciais usuários a fim de entender suas necessidades e criar um projeto que as atenda suas necessidades. Eles definem então visões para cada grupo de usuários. Podemos listar ainda como atribuições do AD:

- Padronizar os nomes dos objetos criados no BD
- Gerenciar e auxiliar na definição das regras de integridade



- Controlar a existência de informações redundantes
- Trabalhar de forma corporativa nos modelos de dados da organização

Falaremos agora do **DBA – Database Administrator**, ou, da pessoa que fornece o **suporte técnico** necessário para implementar as decisões. Assim, o DBA é responsável pelo controle geral do sistema em um nível técnico. Tem como **recurso primário** o banco de dados e como **recursos secundários** o SGBD e os softwares relacionados.

O DBA é o responsável por autorizar o acesso ao banco de dados, coordenar e monitorar seu uso, adquirir recursos de software e hardware conforme a necessidade e por resolver problemas tais como falhas de segurança ou demora no tempo de resposta do sistema. Segundo o Date uma lista de atividades associadas ao DBA contém as seguintes tarefas:

- Definir o esquema conceitual<sup>2</sup> (às vezes conhecido como lógico)
- Definir o esquema interno
- Contatar com os usuários
- Definir restrições de segurança e integridade
- Monitorar o desempenho e responder a requisitos de mudanças.
- Definir normas de descarga e recarga (dumping)<sup>3</sup>

Vejamos algumas questões rápida sobre o assunto:



## 12. Ano: 2019 - CRN - 3ª Região (SP e MS) - Assistente Técnico

No que se refere aos conceitos gerais de banco de dados, assinale a alternativa correta.

A Os dados, independentes ou não, de um banco de dados formam necessariamente uma informação.

B Os projetistas, também conhecidos como administradores do banco de dados (database administrators), são os usuários iniciantes ou paramétricos do sistema.

C Um banco de dados não representa um aspecto do mundo real.

D Uma planilha do Microsoft Excel ou uma lista de contatos (nome, telefone e e-mail) de uma agenda configuram um banco de dados.

<sup>2</sup> Esse esquema conceitual é estabelecido no âmbito da arquitetura em três esquemas. Veremos que ele não tem relação com o modelo de dados conceitual.

<sup>3</sup> Carga e recarga ou dumping é uma outra forma de definir backup de banco de dados.



E A edição ou a alteração dos dados de um banco de dados é muito onerosa ao administrador, o que torna comum a prática de sua total exclusão.

**Comentário:** Vamos comentar cada uma das alternativas. (A) Dados e informações são conceitos distintos. A informação precisa de contexto e significado, já o dado é apenas um fato registro (bruto). (B) Os projetistas são conhecidos como administradores de dados. (C) Todo banco de dados deve descrever alguma coisa do mundo real, podendo ser concreto ou abstrato. (D) Falamos sobre isso no início da nossa aula. A lista telefônica do seu Zacarias é considerada um banco de dados, mesmo não sendo digital. (E) Fazer alterações no banco de dados nem sempre é onerosa, alguns fatores podem permitir uma alteração rápida do registro.

**Gabarito: D**

### 13. Ano: 2018 Órgão: EBSE RH Prova: Analista de Tecnologia da Informação

Com relação a banco de dados, julgue o item seguinte.

Após um banco de dados ser criado, o administrador executa uma série de tarefas para dar permissão de acesso aos usuários que necessitam ler e gravar informações na base de dados. A responsabilidade de gerir os acessos ao banco de dados é do sistema gerenciador de banco de dados (SGBD).

**Comentário:** Observe que neste texto podemos verificar explicitamente o benefício do sistema de gerenciamento de banco de dados. O administrador de banco de dados vai atribuir aos diversos usuários as permissões compatíveis com suas necessidades. Contudo, quem vai gerenciar o acesso, no sentido de autorizar o usuário a acessar o banco de dados e visualizar os dados, é o próprio SGBD. Desta forma, temos uma alternativa correta.

**Gabarito: C.**

### 14. ANO: 2010 ÓRGÃO: TCE-SP PROVA: AGENTE DA FISCALIZAÇÃO FINANCEIRA - PRODUÇÃO E BANCO DE DADOS

Considerando que os dados constituem um dos bens mais valiosos de uma empresa, é necessário que haja um papel que tenha a responsabilidade central pelos dados, principalmente entendendo as necessidades empresariais nos altos níveis da organização. Tal papel é mais adequadamente desempenhado pela

- A administração de banco de dados.
- B administração de dados.
- C alta administração da organização.
- D análise de sistemas.
- E gerência de sistemas.

**Comentário:** Percebam que a questão trata da padronização dos dados corporativos. Entender a organização e sua relação com as informações de forma a gerenciar os



dados de maneira eficiente é reponsabilidade da equipe de **administração de dados**. Hoje em dia um termo muito utilizado para os dados que permeiam vários setores das empresas ou órgãos públicos é o conceito de **dados mestres**. Eles são armazenados em um repositório central e distribuídos aos diversos usuários.

**Gabarito: B.**

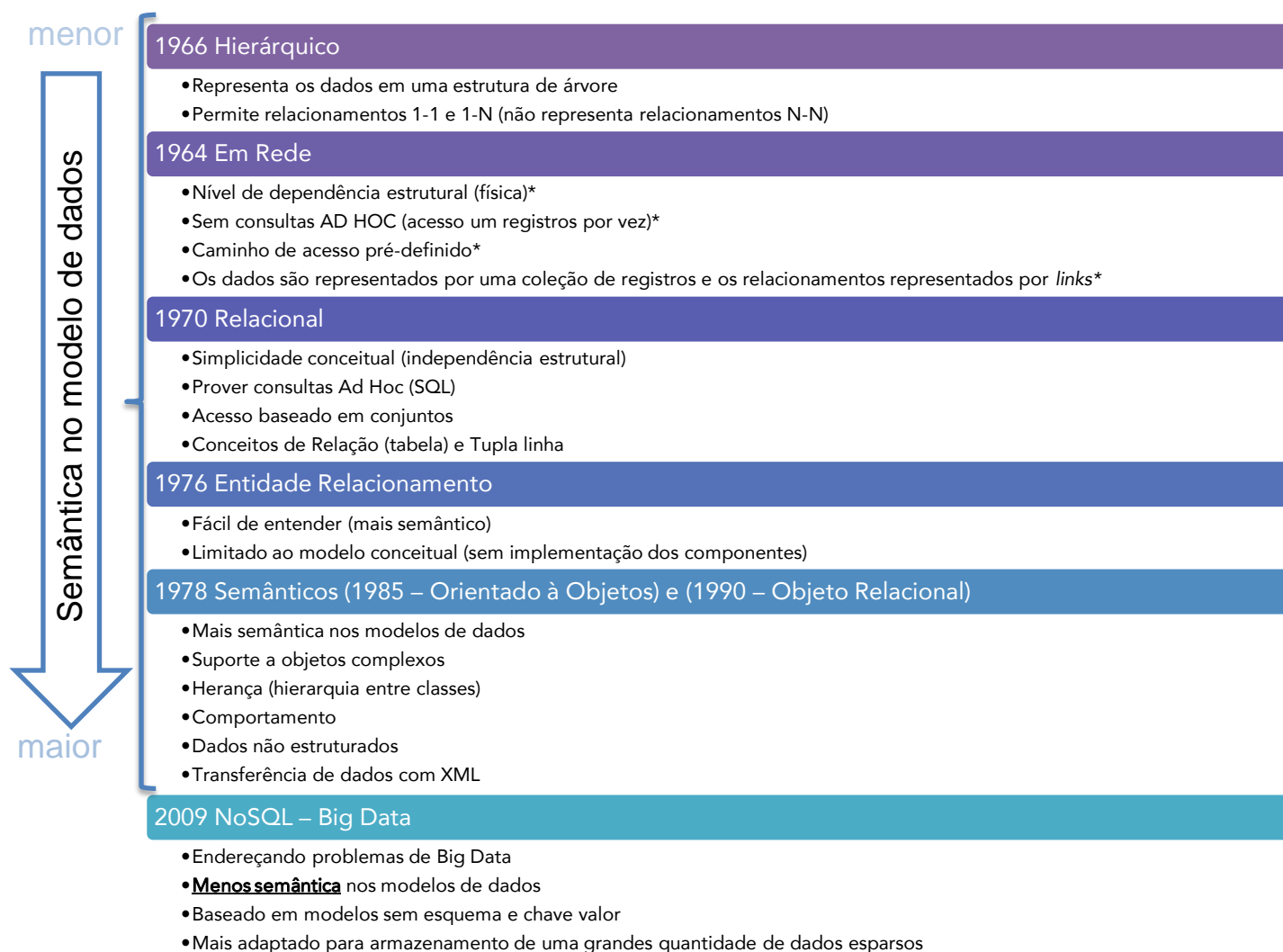
Você precisa entender que são dois os principais papéis presentes no trabalho diário de organização e manutenção de banco de dados. O **administrador de dados** que cuida do contexto estratégico dos dados, sua preocupação está em saber quais os dados serão armazenados em um banco de dados, quem são as pessoas que precisam dos mesmos e ainda, procurar padronizar os nomes dos atributos e outros objetos para que dados redundantes não sejam armazenados.



O outro indivíduo importante neste processo é o **administrador de banco de dados**. Nessa função encontramos todo o suporte técnico necessário para as atividades diárias de um sistema de banco de dados. O DBA distribui as permissões de acesso aos dados, criar os objetos que vão armazenar os dados, faz ajustes para que eles funcionem de forma adequada.

## EVOLUÇÃO HISTÓRICA DOS SGBDs

É interessante conhecer a evolução dos modelos de dados até os bancos de dados NoSQL. Afinal, quando começamos a tratar as informações em sistemas como elas eram armazenadas? A preocupação com o armazenamento dos dados começa na década de 1960. Até então os computadores funcionavam como grandes calculadoras. Veja a figura abaixo para conhecer um pouco mais sobre os modelos de armazenamento de dados.



4

Figura 4 - A figura acima apresenta a evolução histórica dos modelos de dados. Perceba que os aspectos semânticos vão sendo aprimorados até os modelos semântico.

4 \* Também vale para o modelo hierárquico.



Os primeiros sistemas de gerenciamento de banco de dados são implementados no final da década de 1960. Charles Bachmann desenvolveu o primeiro SGBD chamado *Integrated Data Store* (IDS) enquanto trabalhava na Honeywell em 1964. Esse sistema usava o **modelo de rede** onde as relações de dados são representadas como um **grafo bidirecional**. A principal vantagem de um modelo de banco de dados em rede é o **suporte ao relacionamento muitos-para-muitos**. O resultado é um acesso mais rápido aos dados, pesquisa e navegação.

Contudo, o primeiro SGBD que obteve sucesso comercial foi desenvolvido pela IBM chamado *Information Management System* (IMS). Ele usava o **modelo hierárquico** no qual as relações entre os dados são representadas como **uma árvore**. Por incrível que pareça, o IMS ainda está em uso hoje no sistema de reservas SABRE da IBM na American Airlines.

A **estrutura hierárquica** é usada para ordenar fisicamente os registros no armazenamento. É possível acessar os registros navegando para baixo na estrutura de dados usando ponteiros. Vamos tentar a apresentar um exemplo para retirar um pouco da abstração do conceito.

A figura abaixo descreve um banco de dados de Estudante, cada estudante pode se matricular em várias disciplinas ... Neste caso, os estudantes têm um relacionamento de 1-N com disciplina ... Ou seja, se diferentes alunos cursarem a mesma disciplina, elas serão cadastradas várias vezes no banco de dados, uma para cada aluno.

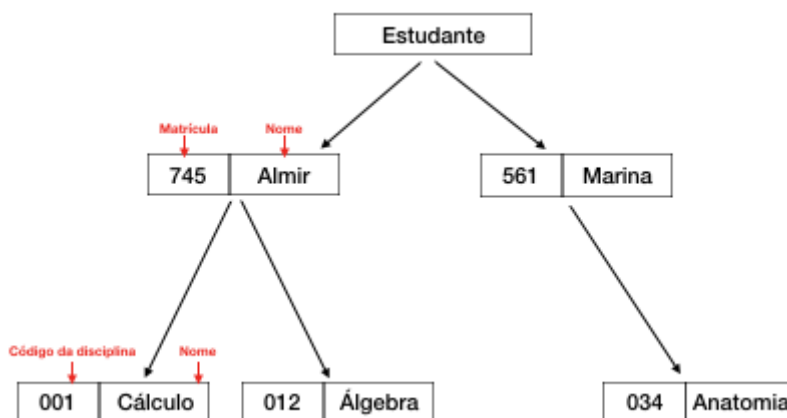


Figura 5 - Modelo de banco de dados hierárquico.

Esses dois modelos, em rede e hierárquico, apresentavam problemas sérios, entre eles:

- O acesso ao banco de dados feito através de operações com o ponteiro de baixo nível.
- Detalhes de armazenamento dependiam do tipo de dados a serem armazenados.
- Para adicionar um campo no banco era necessário reescrever o esquema subjacente de acesso/modificação, em outras palavras o modelo de dados físico.
- Ênfase nos registros a serem processados, não na estrutura global.



- O usuário deveria conhecer a estrutura física da BD, para fim de consulta das informações.

Vejamos algumas questões recentes sobre o assunto:



### 15. Analista (Pref Vila Velha)/Desenvolvimento/2020

Uma das estruturas de bancos de dados é a que tem o formato conhecido por “árvore”. Nessas estruturas, cada registro tem apenas um possuidor. Esse modelo é chamado:

- a) hierárquico.
- b) indexado.
- c) block chain.
- d) relacional.
- e) sequencial.

Comentário: No **modelo hierárquico**, cada registro é definido como um nó numa **estrutura em árvore**, onde cada nó-pai pode ter vários filhos, mas **cada nó-filho só pode ter um pai**. Ou seja, percebemos um relacionamento 1-N.

Esse tipo de banco de dados permite que as referências entre os registros sejam automaticamente construídas por conta das ligações (links) presentes entre eles. Por outro lado, relações complexas são difíceis de se representar no modelo.

Desta forma, temos o gabarito da questão na alternativa A.

**Gabarito: A.**

### 16. Ano: 2019 Órgão: MPC-PA - Analista Ministerial – Tecnologia da Informação

Assinale a opção que apresenta o modelo de dados caracterizado por organizar os dados em uma estrutura do tipo árvore, na qual cada registro tem um único “pai” e é classificado em uma ordem específica.

- A híbrido
- B de rede
- C relacional
- D hierárquico
- E orientado a objetos



Comentário: O modelo hierárquico organiza dados em uma estrutura de árvore, nele cada registro tem um único "pai" ou raiz. Registros "irmãos" são classificados em uma ordem específica. Essa ordem é usada como a ordem física para armazenar o banco de dados. Este modelo é bom para descrever muitas relações do mundo real. Contudo, possui limitações para descrever relacionamentos N-N entre os elementos de dados. Esse modelo foi usado principalmente pelos Sistemas de Gestão de Informações da IBM nos anos 60 e 70, mas são raramente vistos hoje devido a certas ineficiências operacionais.

Curiosidade: O modelo hierárquico foi implementado em um esforço conjunto da IBM e North American Rockwell em 1965. A IBM teve o **Information Management System IMS (DL/1)**, largamente utilizado durante as décadas de 1970 e início da década de 1980

**Gabarito: D**

### 17. Administrador de Rede (CM Piracicaba)/2019

Sobre o modelo hierárquico de bancos de dados, é correto afirmar que

- a) um registro não pode ser pai de mais de um registro filho.
- b) um registro pai pode ter relacionamento com, no máximo, 3 registros filhos.
- c) um registro do tipo raiz só pode ter relacionamento com um único registro filho.
- d) um registro do tipo raiz não participa como registro filho em qualquer relacionamento.
- e) o campo de um registro filho admite apenas tipos de dados inteiros.

Comentário: Vamos comentar cada uma das alternativas. (A) Um registro pode ser pai de vários registros, mas cada registro só pode ter um pai. (B) Mais uma vez, um registro pai pode ter relacionamento com vários filhos, não há limitação. (C) O registro raiz é o pai de todos ... logo, ele terá relacionamento com vários filhos. (D) Essa é a nossa resposta!! (E) Campos podem ter outros valores além de inteiros, por exemplo, valores textuais.

**Gabarito:**

No geral, os primeiros SGBDs eram muito complexos e inflexíveis, o que tornou cada vez mais difícil o trabalho, quando era necessária a adição de novos aplicativos ou a reorganização dos dados. Para resolver esses e outros problemas **Edgar (Ted) Codd**, conhecido com o pai do **modelo relacional**, trabalhando no laboratório da IBM em San Jose propôs no artigo "A Relational Model of Data for Large Shared Data Banks" a definição do **modelo relacional**.





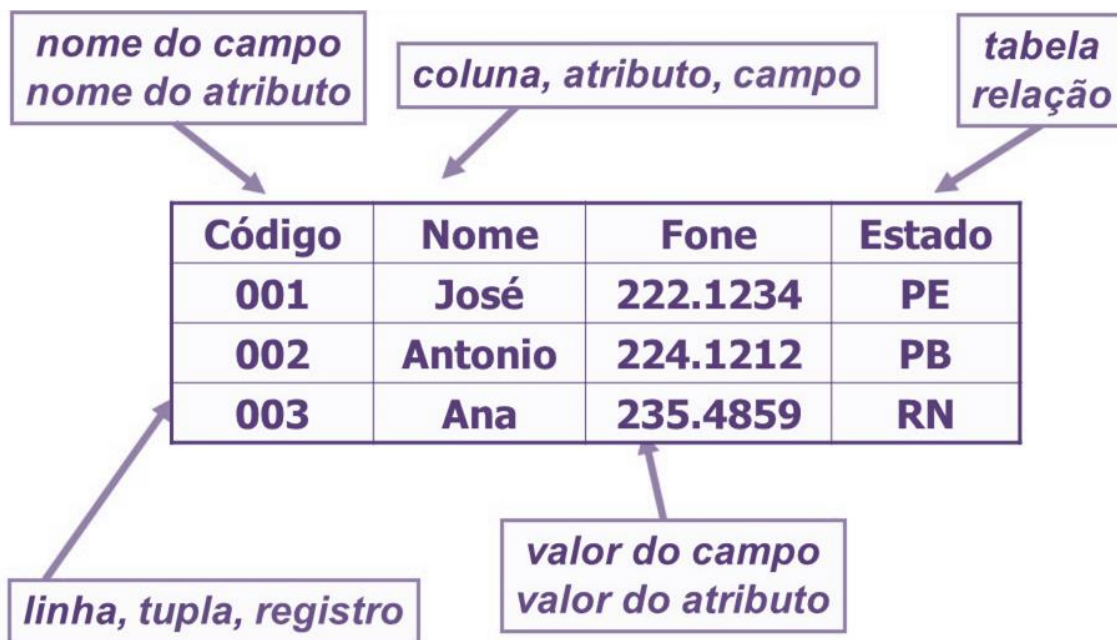


Figura 6 - Elementos básicos do modelo relacional

A ideia de Codd era usar conceitos matemáticos da **teoria dos conjuntos** e da **lógica de primeira ordem** para definir objetos de dados conhecidos como **relações**. As relações podem ser vistas como tabelas compostas por linhas. O modelo chamou as linhas de **tuplas**. As tuplas são constituídas por um conjunto de **atributos** cada um definido dentro de um conjunto de valores possíveis (**domínio**). Perceba que todos os dados são armazenados em uma estrutura predefinida, com tipos de dados e tamanhos bem estabelecidos.

Segundo Codd, o modelo fornece um meio de descrição de dados apresentando apenas a sua estrutura natural - isto é, sem sobreposição de qualquer estrutura adicional para efeitos de representação física dos dados. Perceba a grande sacada de Codd!! Ele construiu um modelo que abstraia a representação física dos dados!! Quando você descreve uma relação, você não se importa com a forma de acesso aos dados quando estes estiverem armazenados.

Essa ideia trouxe uma simplicidade estrutural ao modelo. Assim, ele forneceu uma base para uma linguagem de dados de alto nível que permite obter a **independência máxima entre dados e programas** com a **representação** de um lado e a **estrutura física** da máquina do outro.

Em outras palavras, o modelo relacional teve suas bases estabelecidas na independência de dados e, na forma de acesso aos dados definida por uma linguagem clara e ampla. Em vez de processar um registro de cada vez, um programador pode usar o idioma para **especificar operações** individuais que seriam **realizados** em todo o **conjunto de dados**.

Devido à natureza técnica do artigo e a relativa complicação matemática presente no texto, o significado e proposição do artigo não foram prontamente identificados. Entretanto, Codd levou a IBM a montar um grupo de pesquisa conhecido como **System R** (Sistema R) para tentar avançar sobre o assunto.



O projeto do Sistema R era criar um sistema de banco de dados relacional o qual eventualmente se tornaria um produto. Os primeiros protótipos foram utilizados por muitas organizações, tais como na *Sloan School of Management* (renomada escola de negócios norte-americana). Novas versões foram testadas com empresas de aviação para rastreamento de manufaturas em estoque.

Eventualmente, o Sistema R evoluiu para SQL/DS, o qual posteriormente tornou-se o **DB2**. A **linguagem** criada pelo grupo do Sistema R foi a **Structured Query Language** (SQL) ou linguagem de consulta estruturada. Esta linguagem tornou-se um padrão na indústria para bancos de dados relacionais e, hoje em dia, é um padrão ISO (*International Organization for Standardization*). A linguagem SQL era originalmente conhecida como SEQUEL (*Structured English QUery Language*). Depois teve seu nome modificado para SQL por problemas de patentes.

Vejamos uma questão simples sobre bancos de dados relacionais que já conseguimos resolver com essa pequena introdução ao assunto ...



## 18. FAEPESUL - Assistente (CRC SC)/Suporte em Informática/2019

A definição correta para banco de dados relacionais é:

- a) Um sistema que serve para criar uma relação de confiança, para acesso à internet.
- b) Um sistema que serve para armazenar arquivos dentro do computador.
- c) Um banco de dados que modela os dados de forma que eles sejam percebidos pelo usuário como tabelas.
- d) Um banco de dados onde cada informação é armazenada na forma de objetos.
- e) Um banco de dados que não possui tabelas.

Comentário: As tabelas ou relações dos bancos de dados relacionais são formadas por linhas ou tuplas, que indicam cada registro da tabela, e colunas ou atributos, que identificam os campos da tabela. Cada atributo possui um domínio associado a ele, ou seja, um conjunto de valores que ele pode assumir. Assim, podemos marcar nossa resposta na alternativa C.

**Gabarito: C.**

Em meados da década de 80, tornou-se óbvio que existiam várias **áreas onde bancos de dados relacionais não eram aplicáveis**, por causa dos tipos de dados envolvidos. Estas áreas incluíam medicina, multimídia e física nuclear, todas com necessidades de flexibilidade para definir como os dados seriam representados e acessados.



Este fato levou ao início de pesquisas em **bancos de dados orientados a objetos**, nos quais os usuários poderiam definir seus próprios métodos de acesso aos dados e como estes seriam representados e acessados. Ao mesmo tempo, linguagens de programação orientadas a objetos (*Object Oriented Programming* - POO), tais como C++, começaram a surgir na indústria.

No início de 1990, temos a aparição do primeiro Sistema de Gerenciamento de Banco de Dados Orientado a Objetos (SGBDOO), através da companhia *Objectivity*. Isso permitiu que usuários e organizações criassem bancos de dados para armazenar resultados de pesquisas como o CERN (maior laboratório que trabalha com partículas em pesquisas de física nuclear - europeu) e SLAC (Centro de Aceleração Nuclear - norte-americano), para mapeamento de rede de provedores de telecomunicações e para armazenar registros médicos de pacientes em hospitais, consultórios e laboratórios.

A evolução nos levou aos SGBDs orientados a objetos, mas a praticidade nos trouxe de volta para o modelo **objeto-relacional**, pois a grande maioria das empresas continuou utilizando os bancos de dados relacionais. Contudo esse modelo começou a apresentar outra lista de problemas ou desafios:

1. Dados na ordem de dezenas ou centenas de TB (terabytes) – abordagem de cluster é cara.
2. Poder de crescimento elástico horizontal – controle de transação ACID torna inviável com a elasticidade.
3. Fácil distribuição dos dados e/ou processamento – SGBD paralelos são caros.
4. Tipos de dados variados, complexos e/ou semiestruturados – modelo de dados objeto-relacional não resolve todos os requisitos.

Tivemos então o surgimento de um novo movimento no mercado em busca de uma solução que superasse tais problemas: **o movimento NoSQL**. Este teve sua origem em junho de 2009, para nomear um encontro promovido por **Johan Oskarsson** e **Eric Evans**, que teve como objetivo discutir o surgimento crescente de **soluções open source de armazenamento de dados distribuídos** não relacionais.

Podemos considerar NoSQL uma nova onda de SGBDs, pois propõe algumas alternativas ao modelo relacional, porém com uma grande diferença histórica: o movimento NoSQL não tem como objetivo invalidar ou promover a total substituição do modelo relacional, e sim o fim do modelo relacional como bala de prata, como a única solução correta ou válida.

*Curiosidade: Ao que tudo indica o termo NoSQL foi criado em 1998 por Carlo Strozzi para nomear seu projeto open source, que tinha como objetivo ser uma implementação mais leve de um banco de dados relacional, porém sua principal característica era não expor a interface SQL. Portanto é bem irônico usar o termo NoSQL, criado para nomear um banco de dados relacional, para classificar soluções de armazenamento de dados não relacionais.*



Inclusive, é importante entender que NoSQL não significa “no SQL” (não ao SQL), mas sim “not only SQL” (não só SQL).

Juntamente com NoSQL surge o conceito de Big Data. A definição mais tradicional usa a equação dos cinco Vs. Nela, Big Data = volume + variedade + velocidade + veracidade + valor, de dados. Cada característica desta é associada aos dados armazenados. É possível justificar cada uma delas. Vejamos:

**Volume** porque além dos dados gerados pelos sistemas transacionais, temos a imensidão de dados gerados pelos objetos na Internet das Coisas, como sensores e câmeras, e os dados gerados nas mídias sociais por meio de PCs, smartphones e tablets.

**Variedade** porque estamos tratando tanto de dados textuais estruturados como não estruturados como fotos, vídeos, e-mails e tuites.

**Velocidade** porque os dados são gerados muito rapidamente e os sistemas devem ter capacidade de receberem esses dados. Muitas vezes precisamos responder aos eventos quase que em tempo real, ou seja, estamos falando de geração e tratamento de dados em fluxo massivos.

O ponto de vista da **veracidade** também deve ser considerado, pois não adianta muita coisa lidar com a combinação "volume + velocidade + variedade" se não houver dados confiáveis. É necessário que haja processos que garantam a consistência dos dados.

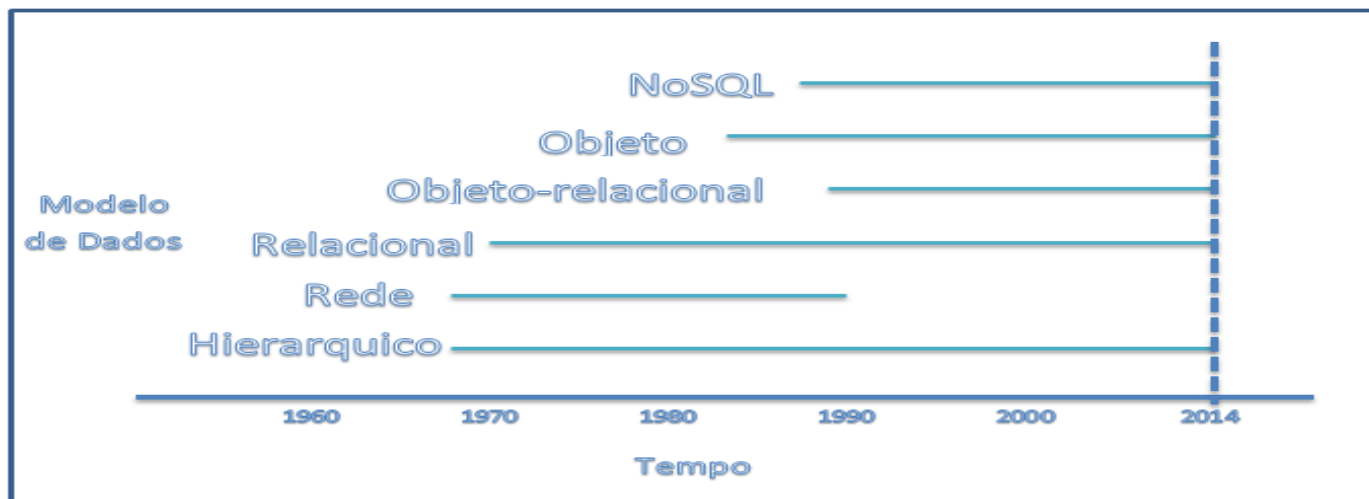
O último V, **valor**, considera que informação é poder, informação é patrimônio. A combinação "volume + velocidade + variedade + veracidade", além de todo e qualquer outro aspecto que caracteriza uma solução de Big Data, se mostrará inviável se o resultado não trouxer benefícios significativos e que compensem o investimento.

O que acontece agora? Diante destas definições, importantes à implementação de SGBDs que suportem a estratégia definida pelo Big Data podemos sugerir diferentes soluções. Bancos do tipo NoSQL são mais flexíveis, sendo inclusive compatíveis com um grupo de premissas que "compete" com as propriedades ACID dos SGBDs tradicionais: **a BASE (BAsically Available, Soft state, Eventually consistency** – Basicamente disponível, estado leve, eventualmente consistente).

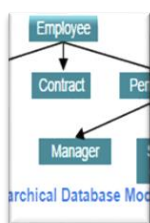
Exemplos de bancos de dado NoSQL são o Cassandra, o MongoDB, o HBase, o CouchDB e o Redis. Mas, quando o assunto é Big Data, apenas um banco de dados do tipo não basta. É necessário também contar com ferramentas que permitam o tratamento correto do volume de dados. Neste ponto, o Hadoop é, de longe, a principal referência.

O **Hadoop** é uma plataforma *open source* desenvolvida especialmente para processamento e análise de grandes volumes de dados, sejam eles estruturados ou não estruturados. Pode-se dizer que o projeto teve início em meados de 2003, quando o Google criou um modelo de programação que distribui o processamento a ser realizado entre vários computadores para ajudar o seu mecanismo de busca a ficar mais rápido e livre das necessidades de servidores poderosos (e caros). Esta tecnologia recebeu o nome de **MapReduce**.





Vamos ficando por aqui, isso é o que nos interessa para o contexto histórico. Apresentamos acima uma figura com uma evolução dos modelos de dados ao longo do tempo e logo em seguida um resumo com as principais características de cada tipo de modelos de dados.



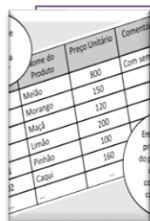
### Modelos de dados hierárquicos

- 1968 - Primeiro modelo de dados a ser reconhecido. Usa estruturas de árvores onde cada registro é considerado uma coleção de campos ou atributos. Exemplo: IMS.



### Modelo de dados em Rede

- 1969 - Eliminou o conceito de hierarquia, permitindo que um mesmo registro estivesse envolvido em várias associações. Ex: IDS.



### Modelo de dados relacional

- Década de 1970 - Sucessor dos modelos de dados legados. Baseia-se no conceito de relações, tuplas e atributos. Tem fundamentação na teoria dos conjuntos.
- Década de 1980 – Os SGBDs começam a ser melhorados devido a grande aceitação do mercado.



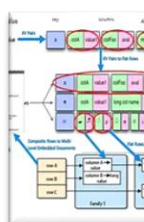
### Modelo Entidade Relacionamento.

- É um modelo conceitual de alto-nível, ou seja, é projetado para ser compreensível aos usuários comuns.
- Formado por um conjunto de objetos chamados de **entidades** e pelo conjunto dos **relacionamentos** entre esses objetos.



### Modelos de dados semântico

- Modelos de dados orientado a objetos – são mais adequados para tratamento de objetos complexos e dinâmicos.. Um objeto estará associado a um estado e um comportamento.
- Modelo de dados objeto relacional



### Modelo de dados NoSQL.

- Década de 90 – Surgem as primeiras alternativas aos modelos relacionais baseados em documentos, chave-valor e colunar.
- Anos 2000 – As bases de dados NoSQL começam a ser reconhecidas devido ao alto poder de performance e escalabilidade

Figura 7 - Diferentes tipos de modelos de dados.

Vamos agora fazer algumas questões sobre o assunto.





HORA DE  
PRATICAR!

## 19. ANO: 2014 ÓRGÃO: TJ-SE PROVA: ANALISTA JUDICIÁRIO – BANCO DE DADOS

Acerca de bancos de dados semiestruturados e bancos de dados NOSQL, julgue os itens subsecutivos.

[86] Bancos de dados NOSQL orientados a documentos são apropriados para o armazenamento de dados semiestruturados.

[87] Para garantir a eficiência das consultas a bancos de dados semiestruturados, é fundamental a adoção de técnica de indexação que leve em consideração, além das informações, as propriedades estruturais dos dados.

[88] Devido à escalabilidade esperada para os bancos de dados NOSQL, a implementação desses bancos utiliza modelos de armazenamento de dados totalmente distintos dos utilizados em sistemas relacionais.

**Comentário:** Vimos que um dos desafios que os banco de dados NoSQL tenta resolver tem relação com os tipos de dados variados, complexos e/ou semiestruturados. Assim podemos considerar a alternativa 86 como **correta**.

A questão 87 envolve alguns conceitos interessantes. Começa falando sobre dados semiestruturados, por exemplo, XML ou JSON. Consultas em bancos de dados semiestruturados consideram tanto a estrutura quanto os valores. Outra questão é a criação de índice sobre um conjunto de dados semiestruturados. Para avaliar se um índice deve ou não ser criado é importante usar as informações sobre a estrutura dos dados e os valores armazenados. Neste caso, considerando a necessidade de um espaço maior para armazenamento e do custo de manutenção, a criação do índice deve melhorar a performance para ser de fato implementado. Logo, a assertiva está **correta**.

A alternativa 88 vai exigir conhecimento sobre os modelos de armazenamento utilizados por bancos de dados NoSQL. Quando tratamos de bases de dados NoSQL podemos classificá-las em quatro diferentes tipos, são eles:

**Chave/valor (Key/Value):** conhecidos como tabelas de *hash* distribuídas. Armazenam objetos indexados por chaves, e facilita a busca por esses objetos a partir de suas chaves.

**Orientados a Documentos:** os documentos dos bancos são coleções de atributos e valores onde um atributo pode ser multivalorado. Em geral, os bancos de dados orientados a documento não possuem esquema, ou seja, os documentos armazenados não precisam possuir uma estrutura em comum. Essa característica faz deles boas opções para o armazenamento de dados semiestruturados.



**Colunar:** Bancos relacionais normalmente guardam os registros das tabelas contiguamente no disco. Ou seja, um registro ou tupla seguida da outra. Por exemplo, caso se queira guardar id, nome e endereço de usuários em um banco de dados relacional, os registros seriam:

Id1, Nome1, Endereço1;

Id2, Nome2, Endereço2.

Essa estrutura torna a escrita muito rápida, pois todos os dados de um registro são colocados no disco com uma única escrita no banco. Também é eficiente caso se queira ler registros inteiros. Mas para situações onde se quer ler algumas poucas colunas de muitos registros (conhecido como família de colunas), essa estrutura relacional (organizada em forma de tabelas) é pouco eficiente, pois muitos blocos do disco terão de ser lidos.

Para esses casos onde se quer otimizar a leitura de dados estruturados verticalmente, bancos de dados de famílias de colunas são mais interessantes, pois eles guardam os dados **contiguamente por coluna**.

O exemplo anterior em um banco de dados dessa categoria ficaria:

Id1, Id2; Nome1, Nome2; Endereço1, Endereço2.

Os bancos de dados de famílias de colunas são mais interessantes para processamento analítico online (OLAP). **BigTable** é uma implementação da Google dessa categoria de bancos de dados.

**Orientado a Grafos:** diferente de outros bancos de dados NoSQL, esse está diretamente relacionado a um modelo de dados estabelecido, o modelo de grafos. A ideia desse modelo é representar os dados e/ou o esquema dos dados como grafos dirigidos, ou como estruturas que generalizem a noção de grafos. O modelo de grafos é aplicável quando “informações sobre a interconectividade ou a topologia dos dados são mais importantes, ou tão importante quanto os dados propriamente ditos”. Possui três componentes básicos: os nós (são os vértices do grafo), os relacionamentos (são as arestas) e as propriedades (ou atributos) dos nós e relacionamentos.

Agora vamos voltar a questão, precisamos responder a seguinte pergunta: nenhum dos modelos acima tem relação com bancos de dados relacionais? Ou ainda, não é possível criar estruturas em SGBDs relacionais que representem esses conceitos?

Embora NoSQL apresente diferentes sistemas de armazenamento que vieram para suprir necessidades em demandas onde os bancos de dados tradicionais (relacionais) são ineficazes. Muitas dessas bases tradicionais apresentam características muito interessantes como alta performance, escalabilidade, replicação, suporte à dados estruturados e sub colunas. Essas características são utilizadas pelas bases de dados NoSQL.

Para finalizar vamos fazer apenas um comentário sobre escalabilidade: A escalabilidade em um banco de dados relacional pode ocorrer de duas formas: horizontal e vertical. A forma horizontal ocorre pela utilização de mais equipamentos e particiona a estrutura de dados de acordo com critérios estabelecidos. A forma vertical ocorre pelo aumento da capacidade do equipamento em que o sistema gerenciador de



banco de dados está instalado. Bases de dados NoSQL têm como um de seus motivadores o baixo custo para realizar uma escalabilidade horizontal, o que torna possível o uso de equipamentos mais acessíveis. Além disso, proporciona um modelo de particionamento nativo (Sharding).

Gabarito: C C E.



## 20. ANO: 2017 ÓRGÃO: TST PROVA: TÉCNICO JUDICIÁRIO – PROGRAMAÇÃO

[57] Considere que um determinado tipo de banco de dados organiza os dados na forma de uma pirâmide, onde o registro principal no topo da pirâmide é chamado registro raiz. Os registros são organizados como pai e filho onde um registro filho sempre tem apenas um registro pai ao qual ele está ligado, como em uma árvore familiar normal. Em contrapartida, um registro pai pode ter mais de um registro filho a ele ligado.

Trata-se do Banco de Dados

- a) hierárquico.
- b) relacional.
- c) orientado a objeto.
- d) objeto-relacional.
- e) de rede.

**Comentário:** Vamos fazer uma rápida revisão sobre os conceitos associados aos modelos de banco de dados orientados a registros:

O **modelo hierárquico** de dados foi o primeiro modelo de dados desenvolvido na década de 1960. O modelo de dados hierárquico foi chamado de IMS (Information Management System) e foi desenvolvido pela IBM e pela Rockwell Company. Ele foi amplamente utilizado durante os anos 1960 e 1970. As entidades e as relações entre entidades foram estruturadas e gerenciadas com a ajuda de uma **estrutura em forma de árvore**. Nesta árvore, existe uma **raiz** que está relacionada aos seus **filhos**. Uma raiz é conhecida como pai. Podemos observar que essa é a nossa resposta. Veja a figura abaixo com a representação de uma estrutura hierárquica.





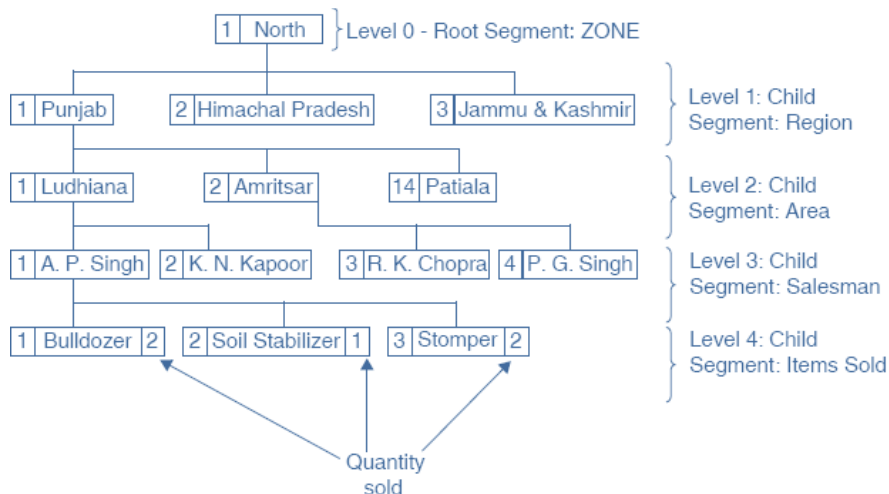


Figura 8 - A figura acima apresenta um exemplo de um modelo de dados hierárquico.

Já o **modelo de dados de rede** representa dados usando o **link** entre os registros. O registro pai é chamado de **Registro Proprietário**, e o registro filho é chamada de **Registro de Membro**. Se os registros Proprietário e Membro estiverem relacionados com o relacionamento muitos-para-muitos, eles estão conectados através do registro do conector que é conhecido como **Set**. Agora vejamos um modelo semelhante ao visto acima, agora com as características do modelo em rede:

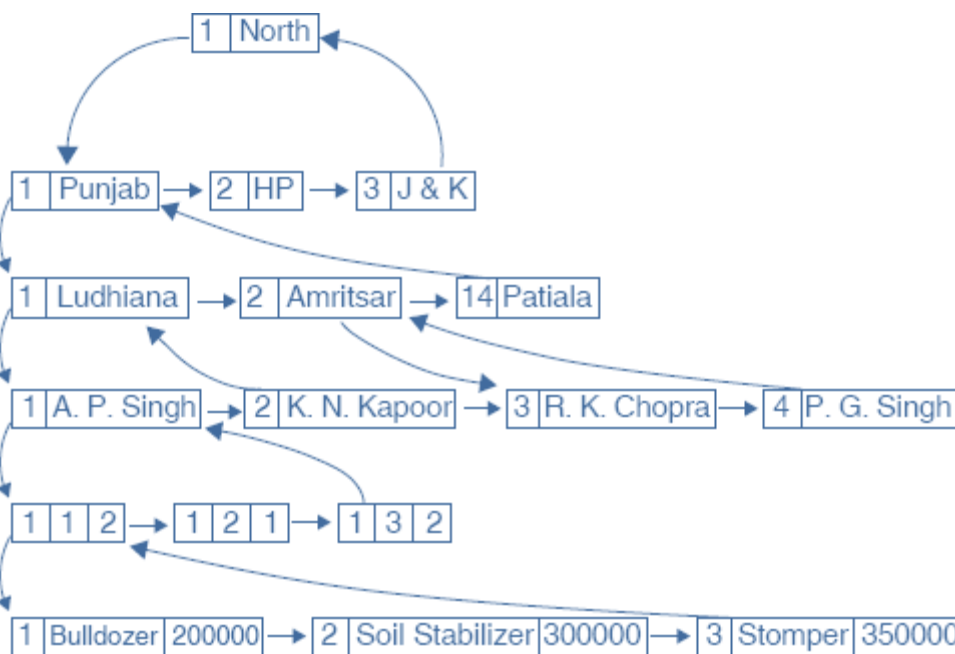


Figura 9 - A figura acima apresenta um modelo de dados em rede.

Temos ainda o conceito de **modelo relacional** foi dado por E. F. Codd, em 1970, em seu documento histórico sobre o modelo de dados relacionais. No modelo relacional, os dados são representados em uma forma tabular denominada, **relação** (tabela), e estão associados a relacionamentos. Portanto, o nome desse modelo é o modelo de dados relacional. Cada **entidade** é convertida em relação e a associação é tratada através de **chaves primárias e estrangeiras**. Cada ocorrência da entidade é conhecida como **tupla** (registro) e a característica de uma entidade é chamada de **atributo** (coluna). É muito fácil representar a relação muitos-para-muitos usando o



modelo de dados relacionais. O modelo relacional é amplamente utilizado em todo o mundo, hoje em dia, para armazenar dados. Vejamos agora um exemplo de um banco de dados relacional de uma livraria on-line.

**BOOK**

ISBN	Book_title	Category	Price	Copyright_date	Year	Page_count	P_ID
001-354-921-1	Ransack	Novel	22	2005	2006	200	P001
001-987-760-9	C++	Textbook	25	2004	2005	800	P001

**PUBLISHER**

P_ID	Pname	Address	State	Phone	Email_ID
P001	Hills Publications	12, Park street, Atlanta	Georgia	71340 19	H_pub@hills.com

**REVIEW**

R_ID	ISBN	Rating
A002	001-987-760-9	6.0
A006	001-354-921-1	7.5
A008	001-987-760-9	7.2

Figura 10 - A figura acima apresenta um exemplo de banco de dados relacional.

**Gabarito: A**



## CLASSIFICAÇÃO DOS SGBDs

Depois de estudar a evolução dos modelos de dados, podemos agora entender as diversas classificações dos SGBDs. A primeira classificação que devemos saber é quanto ao modelo de dados. Desta forma temos os seguintes tipos:



### EXEMPLIFICANDO

TIPO	EXEMPLO	OBSERVAÇÃO
HIERÁRQUICO	IMS (IBM), SYSTEM 2K, TDMS	SISTEMAS DE BANCOS DE DADOS LEGADOS
EM REDE	IDMs, DMS 1100, IMAGE, VAX-SGBD E SUPRA	
RELACIONAL	SQL SERVER, DB2, ORACLE, POSTGRES, MYSQL, SQLITE, MICROSOFT ACCESS	
ORIENTADO OBJETOS	A VERSANT, OBJECT STORE, DB4O, MATISSE.	
ORIENTADO DOCUMENTOS	A MONGODB, COUCHDB, COUCHBASE	NoSQL
COLUNAR	HBASE, ACCUMULO	
CHAVE-VALOR	REDIS, CASSANDRA, DYNAMODB, MEMCACHE	
ORIENTADO GRAFOS	A NEO4J, JANUSGRAPH, DGRAPH, GIRAPH, TIGERGRAPH	

Uma informação importante é que vários desses SGBDs pode se encaixar em mais de um tipo de modelo de dados. Uma lista completa por tipo de banco de dados pode ser vista no site: <https://db-engines.com/en/ranking>. Esse site apresenta uma lista atualizada dos bancos de dados mais utilizados no mercado.

Outro critério usado para classificar SGBDs é o **número de usuários suportados** pelo sistema. Sistemas **monousuário** admitem apenas um usuário de cada vez, e são usados



principalmente com PCs. Sistemas **multiusuário**, que incluem a maioria dos SGBDs, admitem múltiplos usuários simultaneamente.

O terceiro critério é o **número de locais** sobre os quais o banco de dados está **distribuído**. Um SGBD é **centralizado** se os dados estiverem armazenados em um único computador. Um SGBD centralizado pode atender a vários usuários, mas o SGBD e o banco de dados residem integralmente em um único computador. Um SGBD **distribuído** (SGBDD) pode ter o banco de dados real e o software de SGBD **distribuídos por vários locais**, conectados por uma rede de computadores.

Os SGBDDs **homogêneos** usam o mesmo software de SGBD em todos os locais, ao passo que SGBDDs **heterogêneos** podem usar um software de SGBD diferente em cada local.

Um outro critério de classificação é quanto ao custo para aquisição do produto. Temos os SGBDs livre, ou de código aberto, e os SGBDs pagos ou de código fechado.

Resumindo temos que o SGBD pode ser:



<b>MONOUSUÁRIO</b>	<b>OU</b>	<b>MULTIUSUÁRIOS</b>
<b>CENTRALIZADO</b>	<b>OU</b>	<b>DISTRÍBUÍDO</b>
<b>HOMOGÊNEO</b>	<b>OU</b>	<b>HETEROGÊNEO</b>
<b>PAGO</b>	<b>OU</b>	<b>GRATUÍTO</b>

## MODELO DE DADOS E ARQUITETURA EM TRÊS ESQUEMAS

Na parte anterior do nosso curso, tratamos das definições relacionadas aos termos básicos SGBD, BD, SBD e dicionário de dados. Agora vamos tratar de outro grupo de conceitos. Este grupo está relacionado ao entendimento dos diversos elementos do processo de definição de um banco de dados, para definir os diferentes **níveis de modelos de dados** que dão origem aos esquemas. A partir desses esquemas podemos obter instâncias.

OK! Mas o que seriam instâncias e esquemas? É o que veremos a seguir ...

### INSTÂNCIAS X ESQUEMAS

Antes de adentrar no estudo dos modelos de dados queria que você entendesse a diferença entre esquema e instância. O **esquema** é a definição da estrutura do modelo e a **instância** se refere aos dados armazenados em um esquema em um momento do tempo.

Para descrever os modelos utilizamos os elementos fornecidos por cada um deles e construímos esquemas. Conhecido com o projeto geral do banco de dados, o esquema é basicamente a descrição do banco, conhecido também como a **intenção** ou **conotação**. Baseado nesse esquema é possível criar uma instância do BD. Essa coleção de dados armazenados no banco de dados em um determinado instante é o próprio banco de dados, também denominada de **extensão**.

## Esquema



Projeto de banco de dados

Intenção

Conotação

Descrição

## Instância



Fotografia (snapshot)

Extensão

Estado

Ocorrências

Quando trata de uma instância de banco de dados, o Date faz uma distinção interessante sobre os valores que estão armazenados. Ele divide os dados armazenados em campo, registro e arquivo. A figura abaixo nos ajuda a consolidar essa ideia:



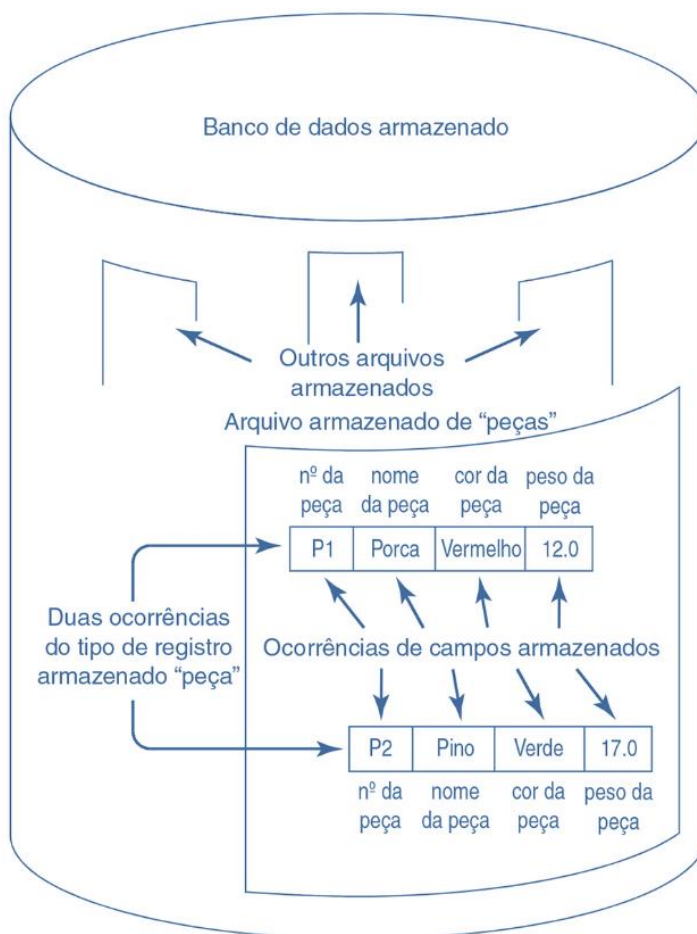


Figura 11 - Campo, registro e arquivos armazenados

Um **campo** é a menor unidade de dados armazenado, veja na figura os diversos atributos de peças, cada um possui um valor específico. Já um **registro** pode ser visto como uma coleção de campos armazenados e relacionados entre si. Cada ocorrência de peça representa um registro. Por fim, um **arquivo** armazenado é o conjunto de todas as ocorrências de um único tipo de registro.

Esse assunto já foi cobrado em provas anteriores ... vejamos:



## 21. FGV - Analista Legislativo (ALESP)/Administração de Banco de Dados e Arquitetura de Dados/2002

Quando falamos em banco de dados é necessário deixar clara a distinção entre esquema de banco de dados e a instância no banco de dados. Esta distinção pode ser descrita da seguinte forma:

- A) Um conceito de um esquema de banco de dados corresponde, em linguagem de programação, à noção de definição das entidades e relacionamentos.
- B) A instância no banco de dados pode ser descrita como uma foto dos dados num determinado momento.
- C) O conceito de instância de relação corresponde, em linguagem de programação, à noção de definição de tipos.
- D) O conteúdo de uma instância de relação não pode mudar com o tempo, quando esta relação for atualizada.

**Comentário:** Vamos comentar cada uma das alternativas:

- a) Entidades e relacionamentos são conceitos associados a modelagem de dados. Em linguagem de programação, em especial nas linguagens orientadas a objetos, o esquema corresponde a classe.
- b) Essa é a nossa resposta. Se você voltar ao esquema apresentado no início do tópico, verá do lado direito a palavra fotografia associada a ideia de instância do banco de dados.
- c) A instância em linguagem de programação está associada a criação de objetos que tem como referência uma classe.
- d) A instância muda toda vez que um dado é inserido, removido ou atualizado. Logo, ela muda com o tempo. O que não muda quando a relação é atualizada é o esquema.

**Gabarito: B.**

## 22. Ano: 2019 Prefeitura de São Roque do Canaã - ES - Técnico em Processamento de Dados

A coleção de informações armazenadas de um banco de dados é chamada de:

- A Parâmetros.
- B Instância.
- C Esquema.
- D Arquitetura.
- E Projeto.

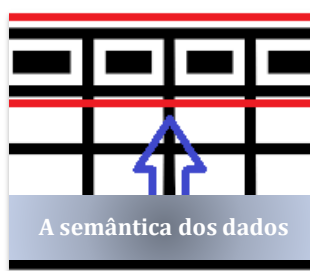
**Comentário:** Perceba que a coleção de informações de um banco de dados em um considerada uma instância, fotografia, extensão, estado ou ocorrências. Assim temos a nossa resposta na alternativa B.

**Gabarito: B**



## MODELO DE DADOS

Um **modelo de dados** fornece um significado necessário para permitir a **abstração dos dados**, ocultando detalhes de armazenamento. Pode ser visto como uma coleção de conceitos que são usados para descrever a estrutura de um banco de dados. Cada modelo deve definir uma coleção de ferramentas conceituais para as seguintes tarefas:



Boa parte dos modelos também dá **suporte a operações**, algumas dessas operações podem representar o aspecto dinâmico ou comportamento de uma aplicação de banco de dados. Com esse conhecimento já podemos resolver uma questão da FGV:

### 23. FGV - Auditor Fiscal da Receita Estadual (SEFAZ RJ)/2011

Para que um sistema de informação possa ser útil e confiável, deve ser fundamentado na modelagem de dados, para posterior análise do processo. A modelagem de dados se baseia nos seguintes elementos:

- A fluxos de dados, atributos e requisitos.
- B fluxos de dados, diagramas e requisitos.
- C classes de dados, métodos e componentes.
- D objetos de dados, diagramas e componentes.
- E objetos de dados, atributos e relacionamentos.

Comentário: A modelagem de dados pede que tenhamos a perfeita noção dos dados que serão armazenados, ou seja dos **objetos de dados**, das características que esses dados possuem **atributos** descritores vão dar mais semântica aos dados armazenados e, por fim, precisamos especificar os **relacionamentos**. Logo, temos a resposta na alternativa E.

**Gabarito: E**

Os **modelos de dados** podem ser divididos em três categorias de acordo com os tipos de conceitos usados para descrever a estrutura do banco de dados. No nível mais alto temos os modelos de dados **conceituais** que apresentam os dados da forma como os **usuários finais** percebem. Em um nível intermediário está a classe de modelos de dados **lógicos ou representacionais** que fornece um entendimento aos envolvidos no processo de





desenvolvimento do BD, mas já introduz informações sobre a forma pela qual os dados são armazenados dentro de um computador. O último nível apresenta os modelos de dados **físicos** ou de baixo nível. Observem na figura a seguir:

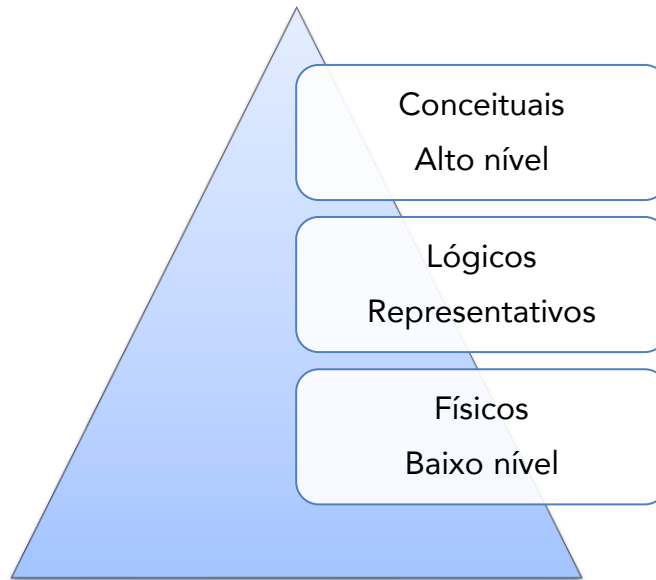


Figura 12 - Categorias de modelos de dados.

Antes de continuarmos gostaria de apresentar para você uma figura que mostra a existência de uma evolução ou um refinamento nos modelos de dados. Esse refinamento reduz o nível de abstração permitindo a implementação da estrutura de dados no disco rígido ou em outro dispositivo físico. O passo a passo do projeto de banco de dados será visto em outro momento neste curso.

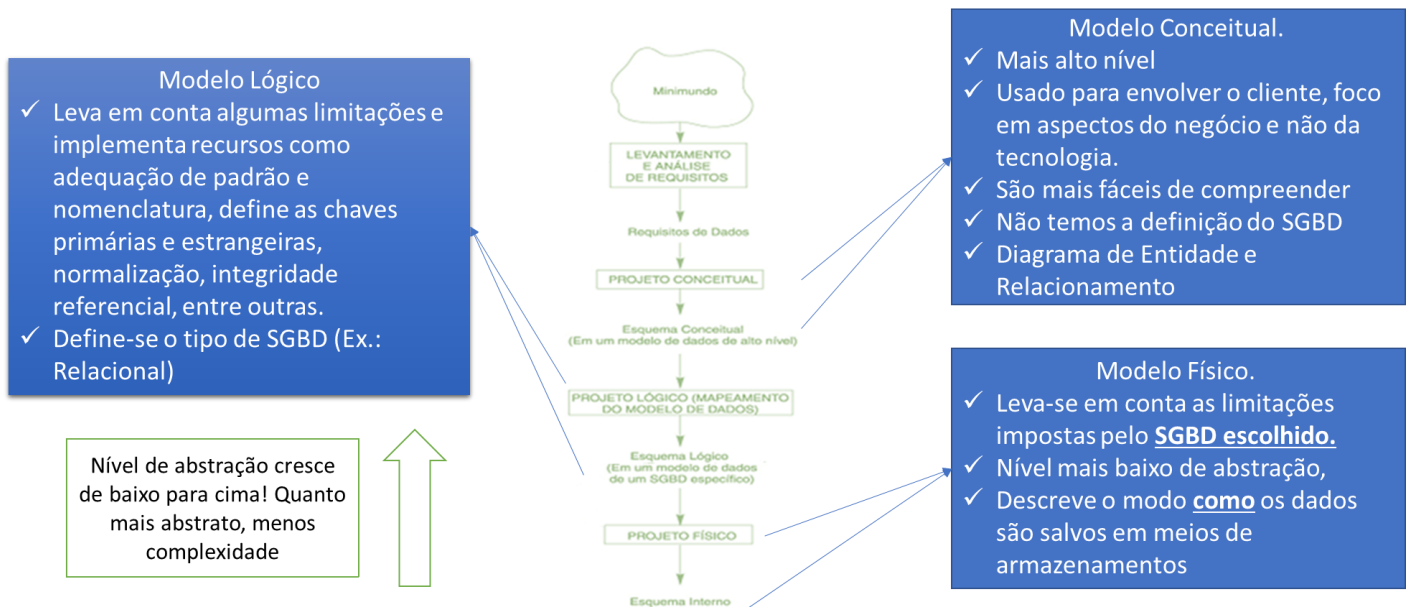


Figura 13 - Resumo sobre os modelos de dados.

Vamos fazer algumas questões da FGV sobre o assunto:

## 24. FGV - Analista (MPE MS)/Informática/Banco de Dados/2013



Um modelo de dados é uma descrição formal das estruturas de dados presentes em um banco de dados. Os dois níveis de abstração dos modelos de dados considerados em um projeto de banco de dados são:

- A modelo conceitual e modelo lógico.
- B modelo lógico e modelo físico.
- C modelo conceitual e modelo físico.
- D modelo de classes e modelo de entidades.
- E modelo de entidades e modelo de relacionamentos.

Comentário: A provável intenção da FGV nesta questão é que você soubesse quais dois modelos possuem maior abstração de dados. Ela considerou que o modelo físico não abstraía informações por definir como os dados serão armazenados. Dessa forma, os modelos onde pode se perceber maior abstração de dados são **o modelo conceitual e o lógico**.

Relembrado o conceito de abstração de dados: “o sistema de banco de dados deve garantir uma visão totalmente abstrata do banco de dados para os diversos tipos de usuários, ou seja, a partir do grau de conhecimento sobre modelagem e da necessidade de entendimento, pode ser apresentado ao usuário uma estrutura mais simples, sem muita complexidade, que descreve apenas quais os dados são armazenados.”

Assim, temos a nossa resposta na alternativa A.

**Gabarito: A**

## 25. FGV - Analista de Sistemas (AL-MT)/Banco de Dados/2013

O modelo de dados abstrato que descreve a estrutura de um banco de dados de forma independente de um SGBD chama-se modelo

- A lógico.
- B conceitual.
- C físico.
- D algorítmico.
- E funcional.

Comentário: Conforme apresentado acima, o modelo de dados conceitual se preocupa apenas em listar e organizar QUAIS os dados serão armazenados. Ele não se preocupa em descrever COMO, logo ele é independente de SGBD.

**Gabarito: B.**

## 26. FGV - Analista (SAD PE)/Controle Interno/Tecnologia da Informação/2009



Modelagem de Dados é uma parte importante do desenho de um sistema de informação, que especifica as estruturas de dados e as regras necessárias para suportar uma área de negócios e que representa um conjunto de requerimentos de informações desse negócio.

A respeito dos modelos existentes, analise as afirmativas a seguir:

I. trata de regras de normalização das estruturas de dados, derivação de relacionamentos e de estruturas de agregação e generalização-especialização, restrição de domínio, de Integridade e de Implementação;

II. trata da análise das características e dos recursos necessários para armazenamento real e manipulação das estruturas de dados.

Os modelos I e II são conhecidos, respectivamente, por:

A conceitual e hierárquico.

B relacional e conceitual.

C hierárquico e lógico.

D físico e relacional.

E lógico e físico.

Comentário: Mais uma questão sobre os níveis de modelos de dados. Deste assunto introdutório de banco de dados, talvez, seja o assunto que a FGV mais gosta. Na descrição acima, temos duas afirmações.

I. Descreve o modelo de dados relacional, que é um modelo **lógico**.

II. Descreve o modelo de dados **físico**, que se preocupa com a organização de arquivos e métodos de acesso. Desta forma, temos a nossa resposta na alternativa E.

**Gabarito: E**

## 27. FGV - Analista Censitário (IBGE)/Geoprocessamento/2017

O projeto de um SGBD para emprego em SIG se divide em várias fases, de modo a prover os dados geográficos de forma eficiente para atender adequadamente às demandas próprias da aplicação.

Nesse contexto, é elaborado o Modelo Entidade-Relacionamento como resultado do(a):

A coleta e análise de requisitos;

B projeto conceitual;

C projeto lógico;

D projeto físico;

E projeto executivo.

Comentário: Vamos aproveitar a questão para detalharmos um pouco do passo a passo do projeto de banco de dados. Para evoluir dentro do processo de definição de um banco de dados, as necessidades dos usuários desempenham um papel



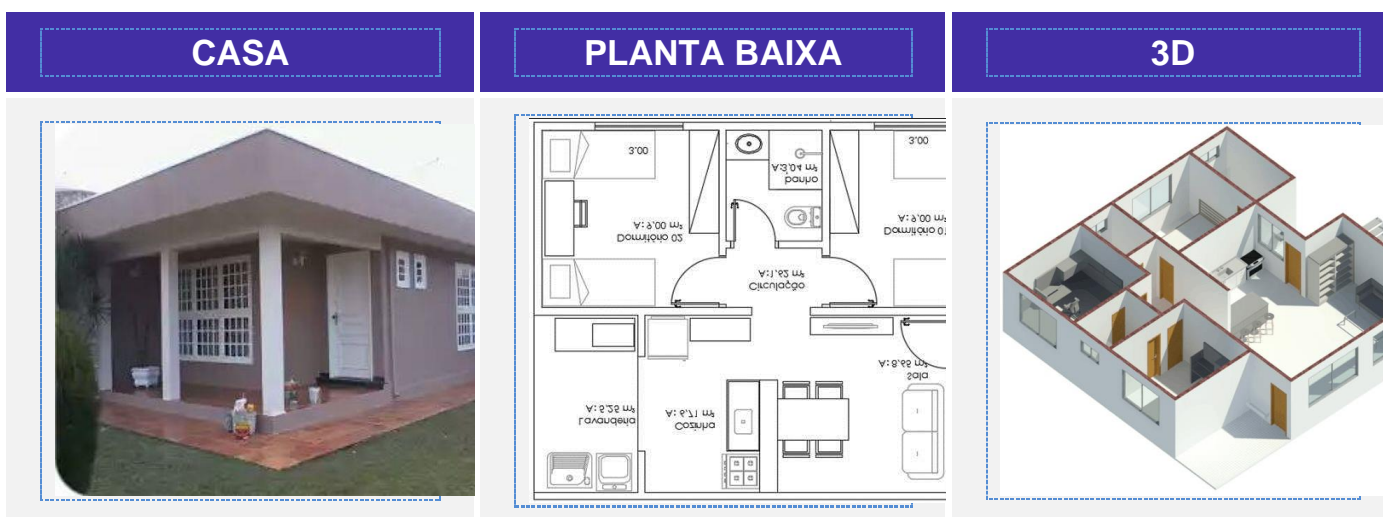
fundamental. Essas necessidades vão **estabelecer os requisitos do nosso esquema de banco de dados**. Visando capturar de forma consistente as funcionalidades esperadas pelos usuários devemos definir um **esquema conceitual de dados**.

Utilizamos um **esquema conceitual** para representar as informações percebidas pelo usuário, retirando do modelo as informações técnicas envolvidas. Desta forma, é possível debater e evoluir o modelo com as pessoas que vão de fato utilizar os dados armazenados. Essas ações levam a construção de um projeto de banco de dados mais robusto e adequado para a organização. O projetista de banco de dados precisa interagir com os usuários da aplicação para entender suas demandas. Essas demandas vão dar origem **um diagrama** que representa as informações de forma simples, em **alto nível**.

Esse diagrama geralmente utiliza **notações gráficas** para representar as **entidades, relacionamentos e atributos** que serão armazenados em um banco de dados. Esse tipo de construção consegue estabelecer uma semântica para os dados. Dentre os exemplos de modelos conceituais podemos citar o modelo entidade-relacionamento. Partindo dessa representação do modelo de dados podemos refinar o diagrama, diminuindo a abstração do modelo.

Uma abstração similar pode ser observada quando pensamos na construção de uma casa ou prédio. Para entender melhor os requisitos do cliente, um arquiteto pode se utilizar de uma planta baixa ou de uma simulação em três dimensões, ambas são abstrações ou representações da casa. Assim, o futuro morador pode decidir sobre a disposição dos móveis em um ambiente, cores e tamanhos.

Perceba que a casa real é uma implementação do modelo abstrato descrito na planta baixa ou no modelo em três dimensões (3D). Essa ideia também vale para bancos de dados, o banco de dados será a implementação de um modelo abstrato que apresenta os requisitos e as definições necessárias. Para chegarmos no nível de implementação precisamos começar no nível conceitual dos modelos de dados, definindo um esquema conceitual.



Perceba que no modelo de dados conceitual queremos definir **quais dados** serão armazenados. Neste momento, **pouco importa como** os dados serão armazenados

fisicamente. Por isso, podemos dizer que os **modelos de dados conceituais são independentes de hardware e software. Eles também não estão associados a um SGBD específico e possuem o mais alto nível de abstração.**

Depois de estabelecermos um modelo conceitual, podemos incorporar mais informações a respeito das restrições e tipos de dados presentes em cada entidade ou relacionamento. Assim, reduziremos o nível de abstração do modelo, definindo um **modelo lógico ou de implementação** para os dados. A relação ou tabela é a estrutura básica que define um modelo lógico, mais especificamente **um modelo lógico relacional.**

Nesta etapa já é possível termos detalhes suficientes para estruturarmos nosso banco de dados. Podemos, portanto, utilizar a linguagem SQL para definição das tabelas. O comando SQL é recebido pelo SGBD que se encarrega de criar a estrutura física para armazenamento dos dados. Ao final da execução dos comandos SQL temos um **esquema físico de banco de dados.**

O fluxo do projeto de banco de dados possui algumas etapas mais importantes que aparecem com frequência em provas de concursos: **projeto (ou modelagem) conceitual, projeto lógico e projeto físico.**

**Gabarito: B**

Após conhecermos a divisão de modelo de dados, vamos mudar nosso foco para outra classificação presente no contexto de banco de dados. Silberschatz apresenta um conceito de níveis de abstração. O nível de abstração **mais baixo** ou **físico** descreve **como** os dados realmente são armazenados. Este nível descreve em detalhes estruturas de dados complexas.

O próximo nível de abstração descreve **que** dados estão armazenados no banco de dados e que relações existem entre eles. O **nível lógico** descreve o banco de dados inteiro em termos de um pequeno número de estruturas relativamente simples. Embora a implementação das estruturas simples no nível lógico possa envolver estruturas complexas em nível físico, o usuário do nível lógico não precisa tomar ciência desta complexidade.

O nível de abstração mais alto descreve apenas parte do banco de dados. Muitos usuários de sistema de banco de dados não precisam de todas as informações armazenadas. Em vez disso, eles precisam apenas de uma parte do banco de dados. O **nível de visão** (*view*) existe para simplificar sua interação com o sistema, que pode fornecer muitas visões para o mesmo banco de dados.

A hierarquia de níveis de abstração de dados pode ser vista na figura abaixo:



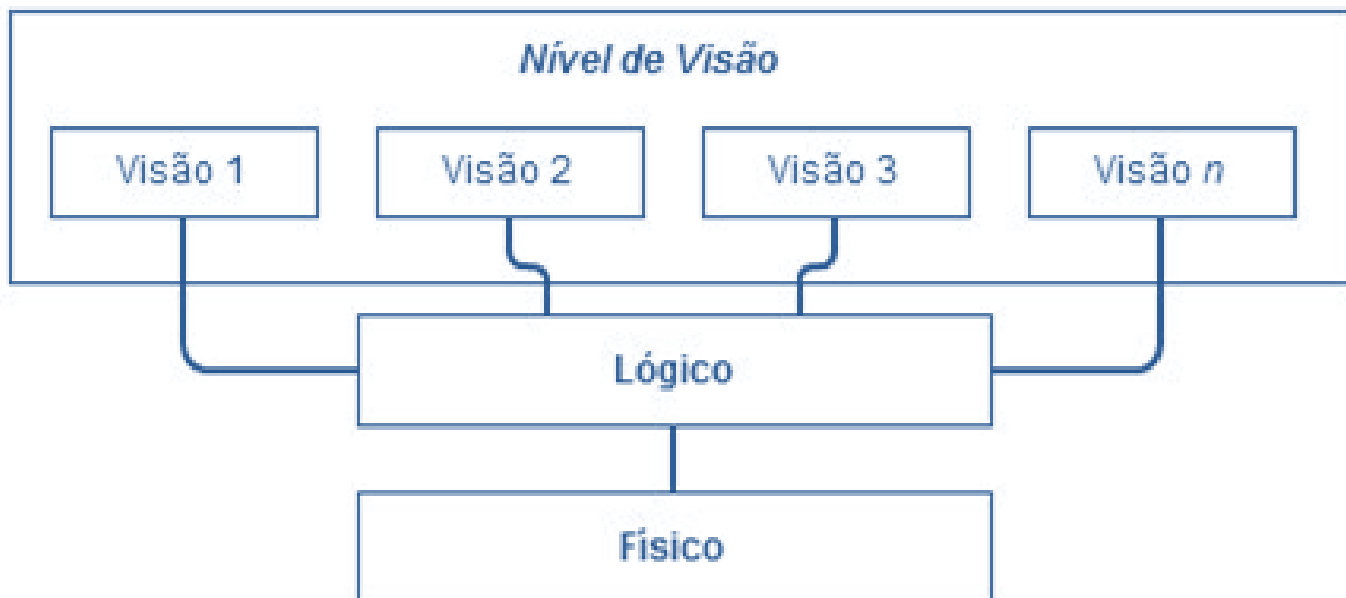


Figura 14 - Os 3 níveis de abstração: Visão, lógico e físico.

## ARQUITETURA TRÊS ESQUEMAS

Agora que já entendemos de modelos de dados, instâncias e esquemas, vamos passar para a arquitetura em três esquemas. O *American National Standards Institute* (ANSI) através do Standards Planning and Requirements Committee (SPARC) estabeleceu um padrão para o desenvolvimento de tecnologias de base de dados, definindo uma arquitetura de três níveis independentes: interno, conceitual e externo.

Essa arquitetura tem por objetivo separar o usuário da aplicação do banco de dados físico. Possuem, logicamente, os esquemas definidos em três níveis distintos. Vamos definir cada um deles e em seguida apresentar uma figura que servirá de base para uma explicação mais detalhada.



● Nível interno - (também conhecido como nível de armazenamento) é o mais próximo do meio de armazenamento físico - ou seja, é aquele que se ocupa do modo como os dados são fisicamente armazenados dentro do sistema.

● Nível conceitual - (também conhecido como nível lógico de comunidade, ou às vezes apenas nível lógico, sem qualificação) é um nível "indireto" entre os outros dois.

● Nível externo ou visão - (também conhecido como nível lógico do usuário) é o mais próximo dos usuários – ou seja, é aquele que se ocupa do modo como os dados são vistos por usuários individuais.

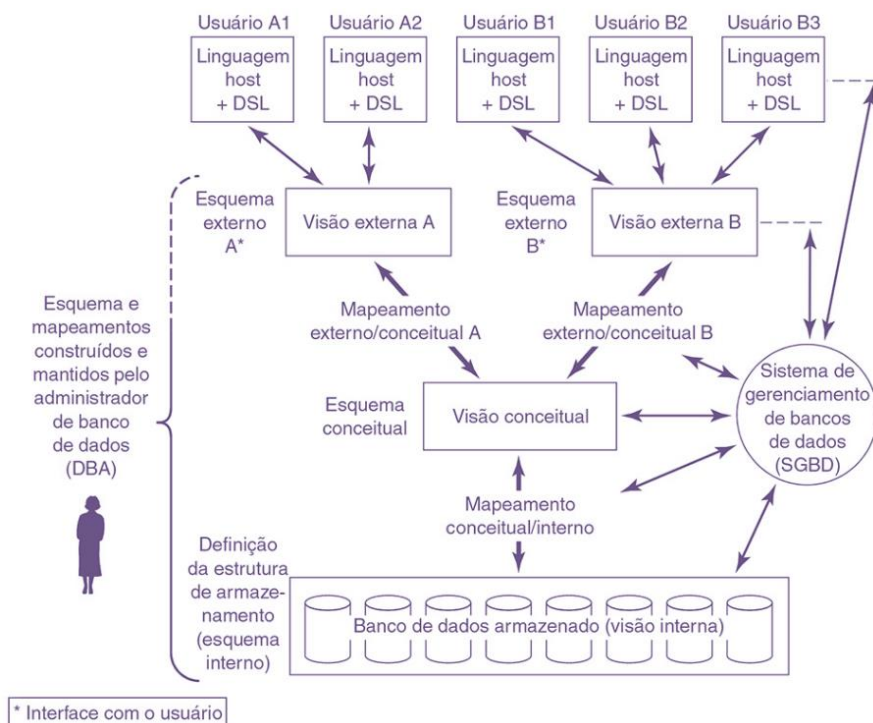


Figura 15 - Arquitetura detalhada do sistema

Olhando para a figura anterior percebemos que diferentes **grupos de usuários** acessam visões externas distintas. A figura mostra dois grupos de usuários A e B acessando suas respectivas visões externa por meio de uma sublinguagem de dados (DSL). Toda DSL é dividida em pelo menos duas sublinguagens: uma **linguagem de definição de dados** (DDL) que dar suporte à criação de objetos no banco de dados; e a linguagem de manipulação de dado (DML) que permite o processamento ou manipulação dos objetos.

O **nível externo** é o nível de usuário individuais. Uma visão externa representa o conteúdo visto por um determinado usuário. Pense que para esse usuário o banco de dados é composto apenas pela parte que ele enxerga (sabe nada inocente! :)). Muitas vezes, um usuário tem acesso a apenas alguns atributos de uma tabela ou arquivo. Essa composição de atributos que não compreende a totalidade das colunas é conhecida como **registro externo**. Cada visão externa é definida como um **esquema externo** e descrita por meio de uma **DDL externa**.

A visão conceitual representa todo o conteúdo do banco de dados também em um nível de abstração razoável quando compara do o nível interno. A **visão conceitual** consiste em várias ocorrências de cada um dos tipos de **registros conceituais**. Um esquema conceitual é usado para descrever cada um dos registros para tal usa uma linguagem de definição conhecida como **DDL conceitual**.

A **visão interna** é uma representação de baixo nível do banco de dados inteiro. Ela é formada por várias ocorrências dos **registros internos**<sup>5</sup>. No nível interno deverá haver referência a representações de campos armazenados, sequências de registros

<sup>5</sup> Registro interno é o termo ANSI/SPARC que representa a construção que temos chamado de registro armazenado



armazenados, índices, esquemas de hashing, ponteiros ou outros detalhes de armazenamento e acesso. Para tal, vamos usar um esquema interno usando uma **DDL interna**.

Veja na tabela abaixo que, embora os termos nível, registros, esquemas e DDL apareçam várias vezes na explicação acima, eles seguem a lógica do "cada um no seu quadrado".

Nível	Registros	Esquemas	DDL
Externo	Externo	Externo	Externo
Conceitual	Conceitual	Conceitual	Conceitual
Interno	Interno	Interno	Interno

Neste momento vamos falar dos mapeamentos **externo/conceitual** e **conceitual/interno** eles são a chave para a **independência de dados** que veremos a seguir. Observe que os três esquemas representam descrições dos dados. Se cada um dos níveis usar sua própria linguagem para descrição e manipulação dos dados, é necessário fazer um mapeamento entre esses níveis.

Uma consulta feita por um usuário no nível externo precisará ser convertida em uma linguagem aceita pelo nível conceitual. A mesma lógica vale para os processos de transformação de requisições e os resultados obtidos entre os níveis conceitual e interno. Segundo o CJ Date:



O **mapeamento conceitual/interno** define a correspondência entre a visão conceitual e interno, ele especifica o modo como os registros e campos conceituais são representados no nível interno.

Um **mapeamento externo/conceitual** define a correspondência entre uma visão externa específica e a visão conceitual.

É possível ainda criar um **mapeamento externo/externo** quando criamos um esquema externo a partir de outro.

Falta falar sobre um último tópico desta seção ... a **independência dos dados** que nada mais é do que a capacidade de alterar o esquema em um nível dos sistemas de banco de dados sem alterar o esquema no nível mais alto ou, em outras palavras a habilidade de modificar a definição de um esquema em um nível sem afetar a definição do esquema em um nível mais alto.

Segundo Navathe é possível definir dois tipos de independência de dados:



PRESTE  
ATENÇÃO!



1. **Independência lógica de dados** - a capacidade de alterar o esquema conceitual sem ter de alterar os esquemas externos ou os aplicativos.

2. **Independência física de dados** - a capacidade de alterar o esquema interno sem ter de alterar o esquema conceitual.

Se a estrutura do banco de dados armazenado for alterada – isto é, se for efetuada uma mudança na definição do banco de dados armazenado – o mapeamento conceitual/interno terá de ser alterado de acordo, a fim de que o esquema conceitual possa permanecer invariável. Em outras palavras, os efeitos dessas mudanças devem ser isolados abaixo do nível conceitual, a fim de preservar a **independência de dados física**.

Apresentamos a seguir uma figura que descreve os níveis da arquitetura em três esquemas. Observem que pela definição de independência de dados é necessário a existência de um nível superior ao esquema alterado. Desta forma só temos o conceito para os níveis conceitual e interno da figura.

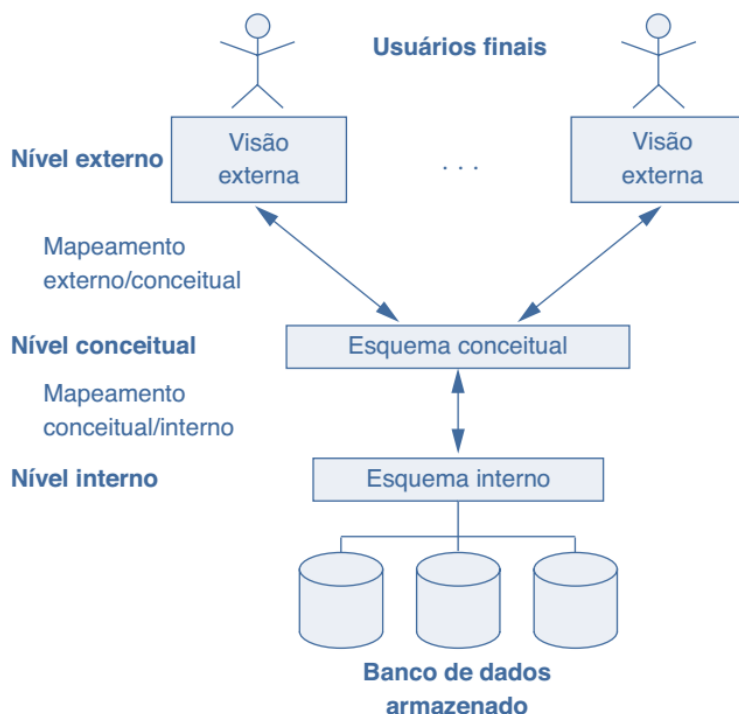


Figura 16 - Níveis da arquitetura em 3 esquemas.

De uma forma simples, cada um dos níveis possui uma função dentro das suas respectivas abstrações. O nível de visão do usuário determina a parte em que o usuário tem acesso. O nível conceitual identifica os dados armazenados e suas relações. Por fim, o nível interno é o nível mais baixo de abstração, define a maneira como os dados estão armazenados. Vejamos mais algumas questões sobre o assunto:

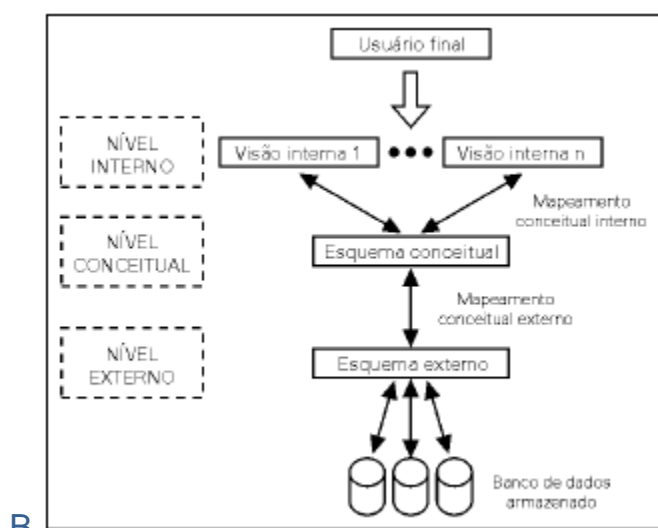
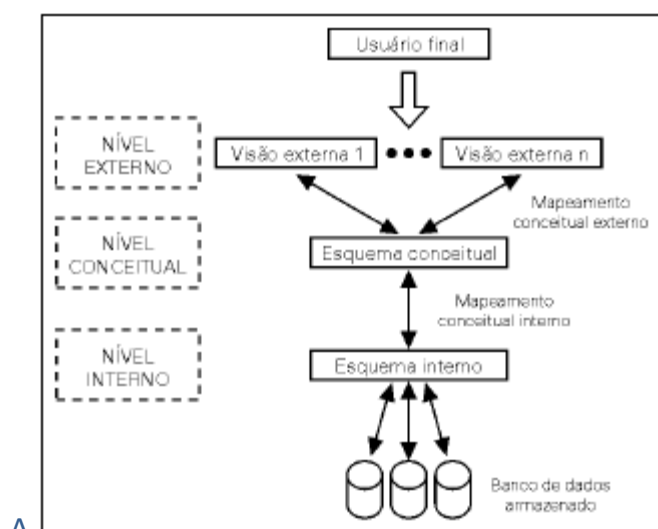


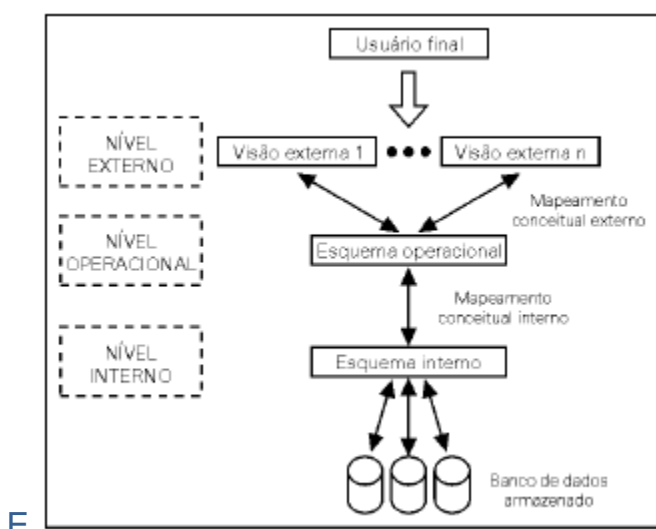
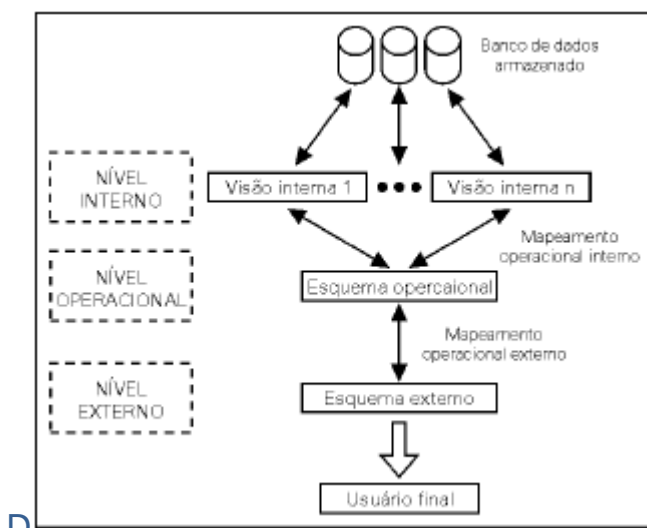
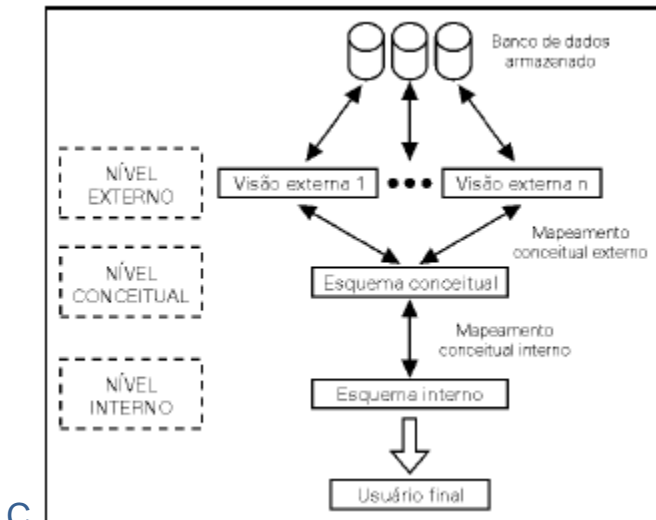


## HORA DE PRATICAR!

### 28. FGV - Fiscal de Rendas (SEFAZ MS)/2006

No que diz respeito aos Sistemas Gerenciadores de Banco de Dados - SGBD, o principal objetivo da arquitetura "três esquemas" é separar as aplicações do usuário do banco de dados físico. Das alternativas apresentadas, a figura que representa essa arquitetura está indicada em:





Comentário: Perceba que a figura que apresenta as descrições corretas para os nomes e os níveis está presente na alternativa A,

Gabarito: A



## 29. SUGEP - Técnico (UFRPE)/Tecnologia da Informação/Sistemas/2019

O padrão de ANSI/SPARC para arquitetura de SGBD define uma arquitetura em três níveis. São eles:

- a) nível interno, nível de usuário e nível físico.
- b) nível interno, nível externo e nível conceitual.
- c) nível externo, nível de tabelas e nível físico.
- d) nível conceitual, nível de usuário e nível de arquivos.
- e) nível de tabelas, nível de arquivos e nível de visão.

Comentário: Acabamos de falar da arquitetura em três esquemas ANSI/SPARC que tem como objetivo separar a aplicação do banco de dados físico. Nessa arquitetura, temos uma divisão dos esquemas em três níveis:

**Interno:** nesse nível está o esquema interno do banco de dados. Esse esquema contém a descrição da **estrutura física**, ou seja, informações detalhadas sobre como os dados são armazenados no hardware, definições das estruturas, índices, caminhos de acesso.

**Conceitual:** nesse nível está o **esquema conceitual**, onde é descrita a estrutura do banco de dados para uma **comunidade de usuários**. Esse esquema não traz detalhes físicos, o foco é descrever **quais dados do banco** são armazenados, como eles se relacionam e as restrições existentes.

**Externo:** nesse nível existem uma série de esquemas externos ou **visões do usuário**. Cada uma dessas visões descreve uma parte do banco que interessa a um determinado usuário (ou **grupo de usuários**), ocultando todo o restante dos dados.

Assim, temos nossa resposta na alternativa B.

**Gabarito: B**

## 30. Ano: 2016 Órgão: TRE-PI Prova: Analista Judiciário - Análise de Sistemas

A respeito das características de um SGBD e das atividades de administração de banco de dados, assinale a opção correta.

- a) Para fins práticos, é necessário distinguir diferentes cardinalidades máximas, que podem ser maiores ou iguais a zero.
- b) A característica autodescritiva de um banco de dados define que o banco de dados contém o próprio dado assim como uma descrição desses dados e suas restrições. Essas descrições e restrições estão armazenadas no catálogo (dicionário) do SGBD.
- c) A independência física de dados consiste na habilidade de modificar o esquema conceitual sem a necessidade de reescrever os programas aplicativos. As modificações no nível conceitual são necessárias quando a estrutura lógica do banco de dados é alterada.



- d) Na linguagem SQL, os comandos DDL GRANT e ROLLBACK permitem a implementação de um controle de acesso discricionário, criando e retirando permissões no banco de dados.
- e) A coleção das informações armazenadas em um banco de dados, em determinado momento, corresponde ao esquema do banco de dados.

**Comentário:** Observe que a descrição presente na alternativa B está perfeitamente adequada. Tente encontrar os erros das demais alternativas. Apenas por curiosidade, na alternativa D, a linguagem SQL possui algumas subdivisões dos comandos. Os comandos GRANT e REVOKE server para implementação do controle de acesso discricionário. O ROLLBACK, juntamente com o COMMIT e SAVEPOINT servem para controle de transações.

Outro ponto importante, é que, na alternativa A, a cardinalidade máxima tem que ser obrigatoriamente maior ou igual a 1. A justificativa para isso será dada na próxima aula.

**Gabarito: B.**

### 31. Ano: 2014 Órgão: TJ-CE Prova: Analista Judiciário - Ciências Computação

Considerando o sistema gerenciador de banco de dados (SGBD), assinale a opção correta acerca de bancos de dados.

- a) Enquanto a DDL (Data Definition Language) é utilizada para definir a estrutura do banco de dados, a SDL (Storage Definition Language) é utilizada para especificar o esquema conceitual e seus mapeamentos com o esquema interno.
- b) A informação armazenada no catálogo do SGBD é denominada metamodelo.
- c) Na independência de dados do programa, propriedade do SGBD, a estrutura dos arquivos de dados é armazenada no catálogo separadamente dos programas de acesso.
- d) Na arquitetura de três esquemas de um banco de dados, o nível conceitual é responsável por descrever de forma detalhada as estruturas de armazenamento físico, incluindo os relacionamentos entre as tabelas.
- e) Na arquitetura de três esquemas, a capacidade de alterar o esquema interno sem ter de alterar o esquema conceitual consiste na independência lógica de dados.

**Comentário:** Nesta questão, tão importante quanto saber que a alternativa C está correta, e esse assunto nós já vimos no início da aula, é entender por que as outras estão erradas. Perceba primeiramente que, se formos teoricamente precisos deveríamos trocar SGBD por SDB. (lembra?!, SGBD + DB = SDB) A questão é que alguns autores falam da abordagem de sistemas de gerenciamento de banco de dados. Assim, essa acaba sendo a menos errada das alternativas.

Na alternativa A, o examinador sugere que SDL, é utilizada para descrever um esquema conceitual. Vejam que SDL está relacionada com armazenamento, e, quando existe de forma separada em uma linguagem, é utilizada para definir os detalhes do nível interno.

Na alternativa B, no lugar de meta modelo, o correto seria metadados.



Na alternativa D, o nível descrito é o nível interno.

Por fim, a alternativa E trata de independência física e não lógica como descrito no texto.

**Gabarito: C.**

**32. Ano: 2016 Órgão: TCE-PA Prova: Auditor de Controle Externo - Área Informática - Analista de Sistema**

Julgue o item subsequente, no que se refere a sistemas de gerenciamento de bancos de dados (SGBD).

Independência lógica de dados refere-se à capacidade de alterar o esquema conceitual sem a necessidade de alterar os esquemas externos ou os programas de aplicação.

**Comentário:** Lembre-se que a independência lógica ou conceitual atual no nível intermediário da arquitetura em 3 esquemas. Ela está relacionada ao fato de que modificações nesta camada da arquitetura não gera necessidade de alteração nos esquemas externos ou programas de aplicação. Desta forma, a alternativa encontra-se correta.

**Gabarito: C.**

**33. Ano: 2016 Órgão: TCE-PA Prova: Auditor de Controle Externo - Área Informática - Administrador de Banco de Dados**

Com relação a sistemas gerenciadores de bancos de dados (SGBD), julgue o próximo item.

No nível conceitual da arquitetura de três camadas de banco de dados, cada esquema externo descreve a parte do banco que interessa a determinado grupo de usuários e oculta desse grupo o restante do banco de dados.

**Comentário:** Perceba que essa questão apresenta uma casca de banana das mais malvadas. Ele mistura o nível conceitual da arquitetura em três esquemas com o nível externo. Sabemos que os esquemas externos estão associados as diferentes visões dos grupos de usuários. Tal fato reflete os interesses distintos de cada grupo quanto ao acesso as informações do banco de dados. Assim, podemos marcar nosso gabarito como errado.

**Gabarito: E.**

**34. ANO: 2010 ÓRGÃO: TCE-SP PROVA: AGENTE DA FISCALIZAÇÃO FINANCEIRA - CONHECIMENTOS BÁSICOS**

As três visões da arquitetura básica de um SGBD, pela ordem, desde a mais próxima do usuário até a mais distante, são:

A externa, conceitual e interna.

B externa, interna e conceitual.

C conceitual, interna e externa.



D conceitual, externa e interna.

E interna, conceitual e externa.

**Comentário.** Ao analisar as alternativas temos que ter em mente duas informações relevantes de enunciados. Primeiro é solicitado as três visões da arquitetura, desta forma podemos concluir que o examinador está se referindo a arquitetura em três esquemas. A outra informação é que a questão pede para você colocar em ordem crescente, da mais próxima do usuário até a mais distante. Com essas duas informações podemos lembrar dos conceitos vistos anteriormente e descrever a seguinte ordem: **externa, conceitual e interna.**

**Gabarito: A.**

## AMBIENTE DO SISTEMA DE BANCO DE DADOS

A figura a seguir apresenta, de forma simplificada, os componentes típicos de um SGBD. Perceba que temos uma divisão em duas partes. A parte superior mostra os usuários e a forma de interação deles com o SGBD. Cada tipo de usuário possui a sua interface de relacionamento.

A parte inferior ilustra os detalhes internos do SGBD, eles são responsáveis pelo armazenamento dos dados e processamento das transações dos usuários. Vejam que o **banco de dados** e o **catálogo do SGBD** estão representados por um cilindro, representando o fato de serem armazenados permanentemente. Os **discos rígidos** representam uma das diversas possibilidades de armazenamento não volátil ou permanente dos dados.



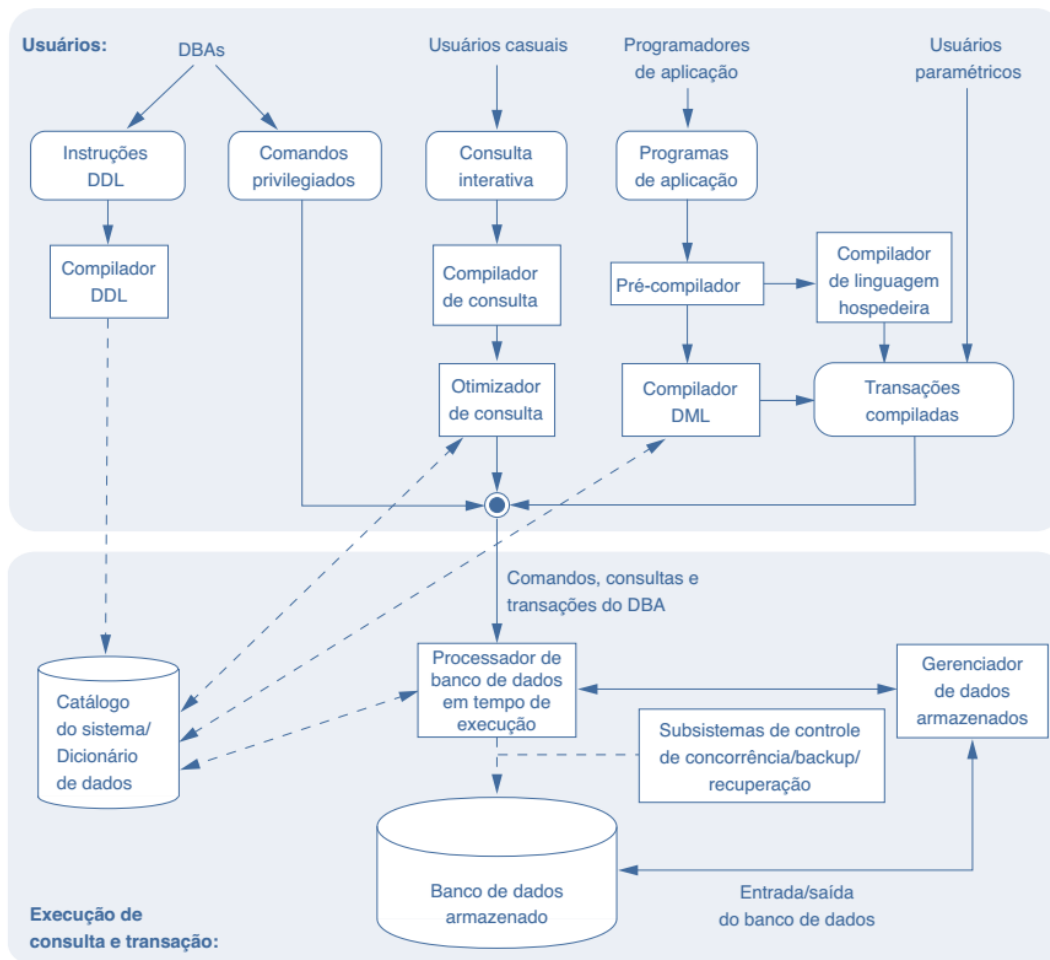


Figura 17 - Ambiente do sistema de banco de dados.

Nas próximas linhas vou descrever um pouco dos componentes da figura acima. Os conceitos podem ser úteis para que você entenda um pouco da “caixa preta” por trás de um SGBD.

O acesso aos discos rígidos é controlado pelo sistema operacional ou pelo próprio SGBD. O controle vai escalonar as operações de leitura e escrita sobre o disco. Quando consideramos a execução de uma transação sobre o banco de dados, o intervalo de tempo gasto na transferência de dados entre a memória e o disco é considerado um gargalo no processamento. Para otimizar o processo alguns SGBDs podem contar com um **módulo de gerenciamento de buffer** que planeja a troca de dados entre a memória principal e o disco.

Outro módulo, o **gerenciador de dados armazenados**, controla o acesso às informações do SGBD que estão armazenadas, seja no catálogo de dados ou no bando de dados. Ele utiliza os serviços básicos do sistema operacional para executar operações de entrada/saída (leitura/escrita) de baixo nível entre o disco e a memória principal.

Observa-se, na parte superior da figura, as interfaces para os diferentes usuários do sistema. De um lado temos **os usuários casuais** que trabalham com interfaces interativas para formular consultas. Logo em seguida, visualizamos os **programadores de aplicação** que usam uma linguagem de programação hospedeira para ter acesso aos dados. Por fim,





temos os **usuários paramétricos** que inserem valores para os parâmetros predefinidos pelas transações.

No parágrafo anterior, falamos apenas das manipulações de dados que podem ser feitas sobre as bases de dados. Essas operações incluem consultas, inserções, deleções e atualizações das informações armazenadas. Contudo, é necessário, antes de manipularmos os dados, construir as estruturas do banco de dados. A linguagem de definição de dados ou *data definition language* é utilizada pelo **DBA** para descrever os objetos presentes na base de dados. Uma tabela do modelo relacional é um exemplo de objeto que podemos criar por meio de uma **instrução DDL** em nossa base de dados.

Quando o DBA digita uma instrução ou comando DDL, essa é enviada ao **compilador DDL**. Um compilador transforma o código fonte (da linguagem de programação) em um código em que o computador entenda. O compilador da DDL processa as definições de esquema especificadas e armazena as descrições de esquema (metadados) no catálogo do SGBD. Esse fluxo pode ser observado no lado esquerdo da figura anterior.

Outra função do DBA é o ajuste fino ou *tuning* do sistema de gerenciamento de banco de dados, bem como a configuração de parâmetros que são feitos por meio dos **comandos privilegiados**. Apenas para exemplificar, um comando presente na maioria dos SGBDs é o *REORG*, serve para reorganizar uma tabela ou índice na estrutura física do banco de dados.

Os **usuários casuais** interagem usando alguma interface de consulta interativa. Essas consultas são analisadas e validadas pela exatidão da sintaxe da consulta, os nomes de arquivos e elementos de dados, e assim por diante, por um **compilador de consulta**. Essa consulta interna está sujeita a melhorias feitas pelo otimizador de consultas, que se preocupa com o rearranjo e a possível reordenação de operações, com a eliminação de redundâncias e uso dos algoritmos e índices corretos durante a execução.

Ele consulta o catálogo do sistema em busca de informações estatísticas e outras informações físicas sobre os dados armazenados, gerando um código executável. Este por sua vez realiza as operações necessárias para a consulta e faz chamadas ao **processador em tempo de execução** (falaremos mais sobre ele logo mais).

Os **programadores de aplicação** escrevem programas em linguagens hospedeiras, como Java e C#, que são submetidas a um **pré-compilador**. Este extrai os comandos DML do programa de aplicação. Para entender melhor como funciona essa divisão vamos partir para um exemplo prático. No exemplo abaixo temos um código Java com um comando SQL. O comentário (//) no código delimita o início da instrução SQL dentro do código Java.



```
public boolean verificarUsuario(String login, String senha){
    String sql = "";
    Connection conn = conectarBD();
    //INSTRUÇÃO SQL
    sql += "select nome from usuarios ";
    sql += "where login = " + "'" + login + "'";
    sql += " and senha = " + "'" + senha + "'";
    try{
        Statement st = conn.createStatement();
        ResultSet rs = st.executeQuery(sql);
        if(rs.next()){
            result = true;
            nome = rs.getString("nome");}
    }catch (Exception e) { }
    return result; }
```

Figura 18 - Exemplo de código em Java que faz acesso a uma tabela de nome usuários. Primeiramente criamos uma conexão com o banco, depois um objeto Statement e executamos a consulta passando o código SQL para o método executeQuery da classe Statement. O resultado é gravado em uma instância da classe ResultSet.

Observem que o comando *select*. Ele está escrito na linguagem SQL considerada uma DML. Esses comandos são enviados ao compilador DML para serem compilados em código objeto com acesso ao banco de dados. O restante do programa é enviado ao compilador da linguagem hospedeira. Os códigos objetos para os comandos DML e o restante do programa são ligados ('linkados') formando uma transação programada ou compilada.

As **transações programadas** são executadas repetidas vezes pelos usuários paramétricos, que apenas fornecem os parâmetros para as transações. No nosso exemplo anterior, as informações necessárias são *login* e senha. Cada execução é considerada uma transação separada. Outro exemplo de transação é o saque no caixa eletrônico, no qual o número da conta e o valor são fornecidos como parâmetros.

Na parte inferior da figura temos o **processador de banco de dados em tempo de execução (PBDTE)**. Ele é responsável por executar os comandos privilegiados, os planos de consulta executáveis e as transações programadas. Para isso são utilizadas informações e dados estatísticos do catálogo do sistema. O PBDTE também trabalha com o **gerenciador de dados armazenados**.

Os **sistemas de controle de concorrência, backup e recuperação** são apresentados como um módulo da figura. Eles são integrados ao processador de banco de dados em tempo de execução para fins de gerenciamento de transações. Você precisa ter em mente que esses controles são necessários para o perfeito funcionamento do SGBD. O backup é utilizado durante a recuperação caso alguma falha aconteça. A concorrência entre transações deve existir dentro de limites bem definidos para evitar que o banco de dados entre em um estado inconsistente.

A figura apresentada não pretende descrever um SGBD específico nem esgotar suas funcionalidades. Nossa ideia foi ilustrar os módulos básicos de um SGBD e estruturar seu raciocínio. Lembre-se que um SGBD é um sistema informatizado. Para executar todas as suas tarefas sua implementação é dividida em diferentes módulos. O SGBD interage ainda com o sistema operacional quando o acesso ao disco rígido é necessário. Vamos fazer uma questão que trata do assunto.





### 35. Ano: 2010 Órgão: BADESC Cargo: Analista de Sistemas

Os objetivos dos compiladores DDL, DML e DCL são, respectivamente:

- A) criar os objetos do banco de dados, manipular (recuperação, inserção, remoção e alteração) de dados nos objetos criados pela DDL e fornecer privilégio de acesso às informações.
- B) fornecer privilégio de acesso às informações, criar os objetos do banco de dados e manipular (recuperação, inserção, remoção e alteração) de dados nos objetos criados pela DDL.
- C) manipular (recuperação, inserção, remoção e alteração) de dados nos objetos criados pela DML, criar os objetos do banco de dados e fornecer privilégio de acesso às informações.
- D) fornecer privilégio de acesso às informações, manipular (recuperação, inserção, remoção e alteração) de dados nos objetos criados pela DDL e criar os objetos do banco de dados.
- E) criar os objetos do banco de dados, fornecer privilégio de acesso às informações e manipular (recuperação, inserção, remoção e alteração) de dados nos objetos criados pela DDL.

**Comentário:** Vimos que o compilador DDL permite que o DBA emita comandos para a criação dos objetos do banco de dados, esses vão fornecer informações sobre as estruturas das tabelas como atributos e restrições. O compilador DML é responsável por transformar o código SQL para manipulação dos dados armazenados.

Por fim, temos o **Data Control Language – DCL**, essa parte da linguagem SQL vai permitir aos administradores de banco de dados a distribuição de privilégios de acesso sobre a base de dados. Com essa informação, podemos marcar tranquilamente a resposta na alternativa A.

**Gabarito: A.**

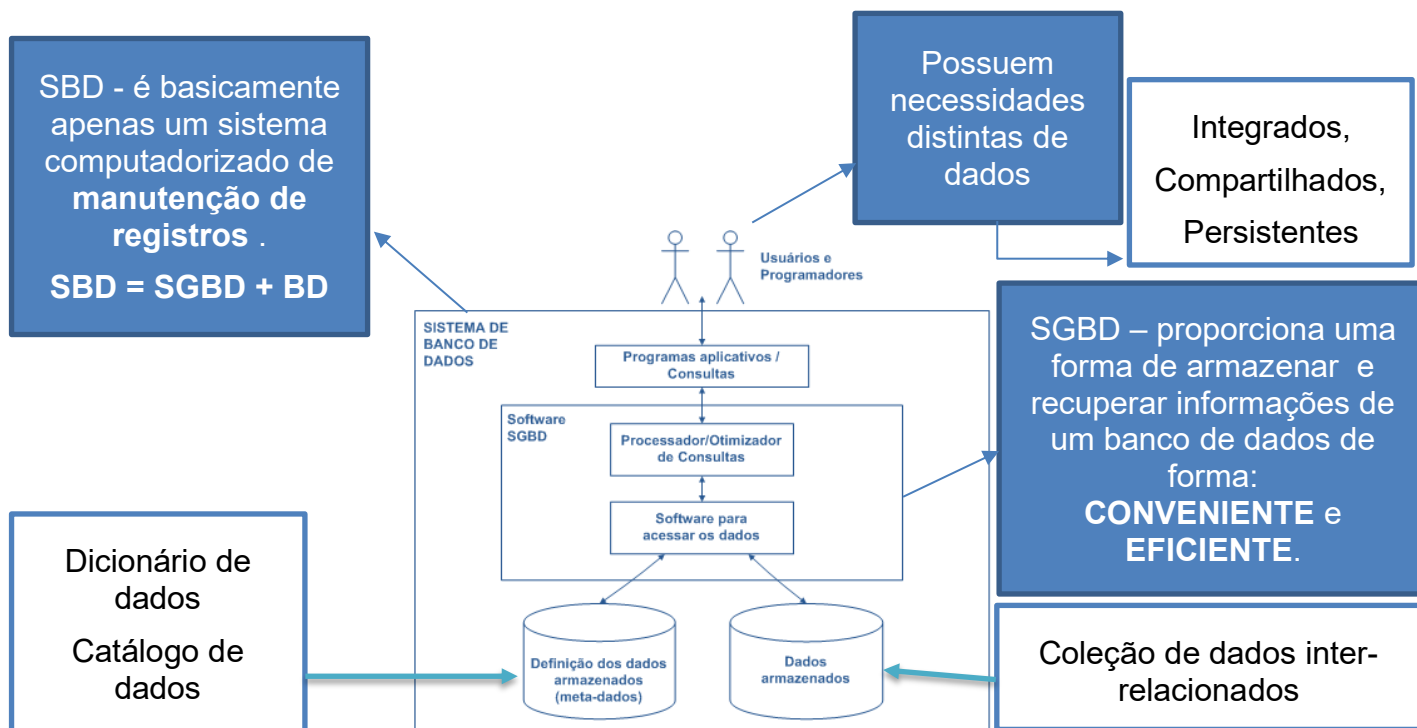
Espero que você tenha entendido nossa proposta de apresentar o ambiente do SGBD. Acabamos aqui o nosso primeiro conjunto de assuntos teóricos envolvidos na introdução dos sistemas de bancos de dados.



## RESUMO

Antes de finalizar de fato a nossa aula vamos revisar alguns tópicos importantes que forma vistos que gostaria que você fixasse. Vamos começar com uma citação ao Silberchatz.

Um **sistema de gerenciamento de banco de dados (SGBD)** é uma coleção de dados inter-relacionados e um conjunto de programas para acessar esses dados. A **coleção de dados**, normalmente conhecida como **banco de dados**, contém informações relevantes para uma empresa. O **principal objetivo** de um SGBD é proporcionar uma forma de armazenar e recuperar informações de um banco de dados de maneira **conveniente e eficiente**.

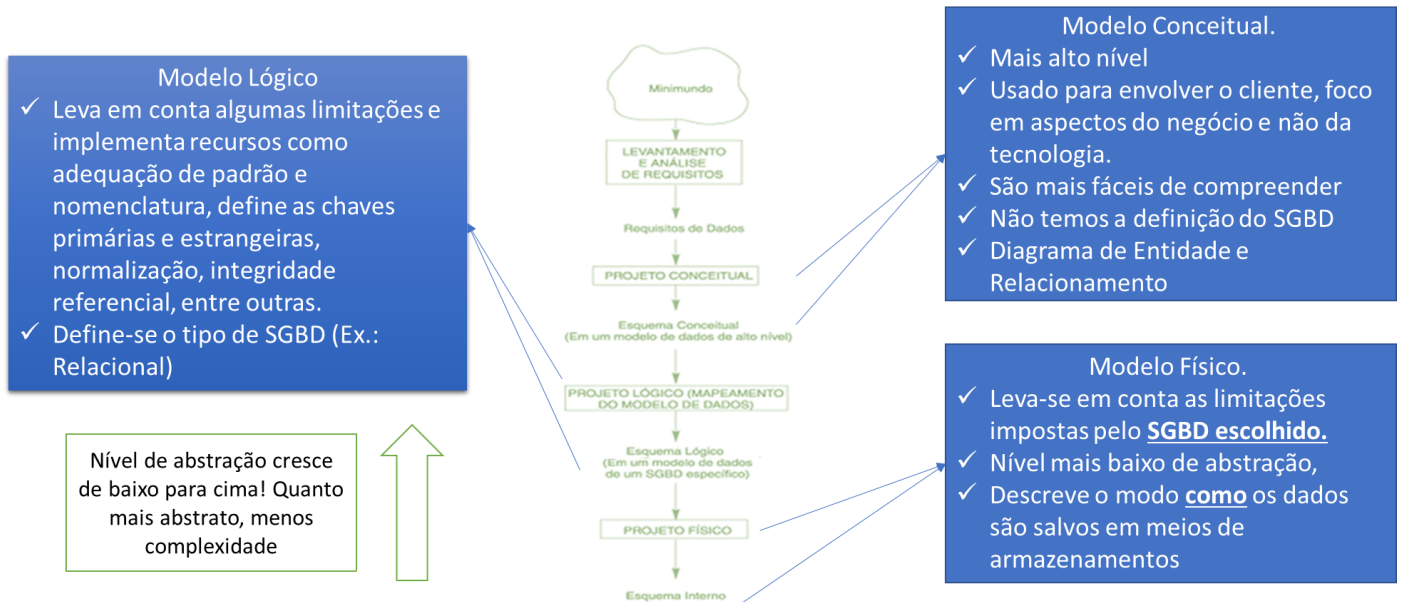


Outro ponto importante são os conceitos hierárquicos de banco de dados vistos ao longo da aula. Reconhecemos que em determinado momento parece que elas estão tratando do mesmo conceito.

Eu gosto sempre de dizer que os conceitos acabam se complementando. Os **modelos de dados** podem ser associados a forma de representação dos dados. Eles variam de acordo com a percepção das pessoas que interagem com o sistema em cada um dos níveis. Sendo assim, temos o **modelo conceitual**, que representa os dados para os usuários de negócio; o **modelo representativo ou lógico**, que estrutura os dados para implementação em um SGBD; e os **modelos físicos**, que descrevem a forma como os dados são armazenados nos dispositivos de armazenamento, por exemplo, disco rígido.

Por outro lado, temos as hierarquias definidas pelos **níveis de abstração** e pela **arquitetura em 3 esquemas**. Uma característica importante em ambas é que no nível mais alto de abstração elas separam o banco de dados de acordo com grupos de usuários. Imagine que cada setor de uma empresa tenha acesso a apenas uma parte dos dados. O **nível de visão** ou a **visão externa** descreve exatamente essa perspectiva.





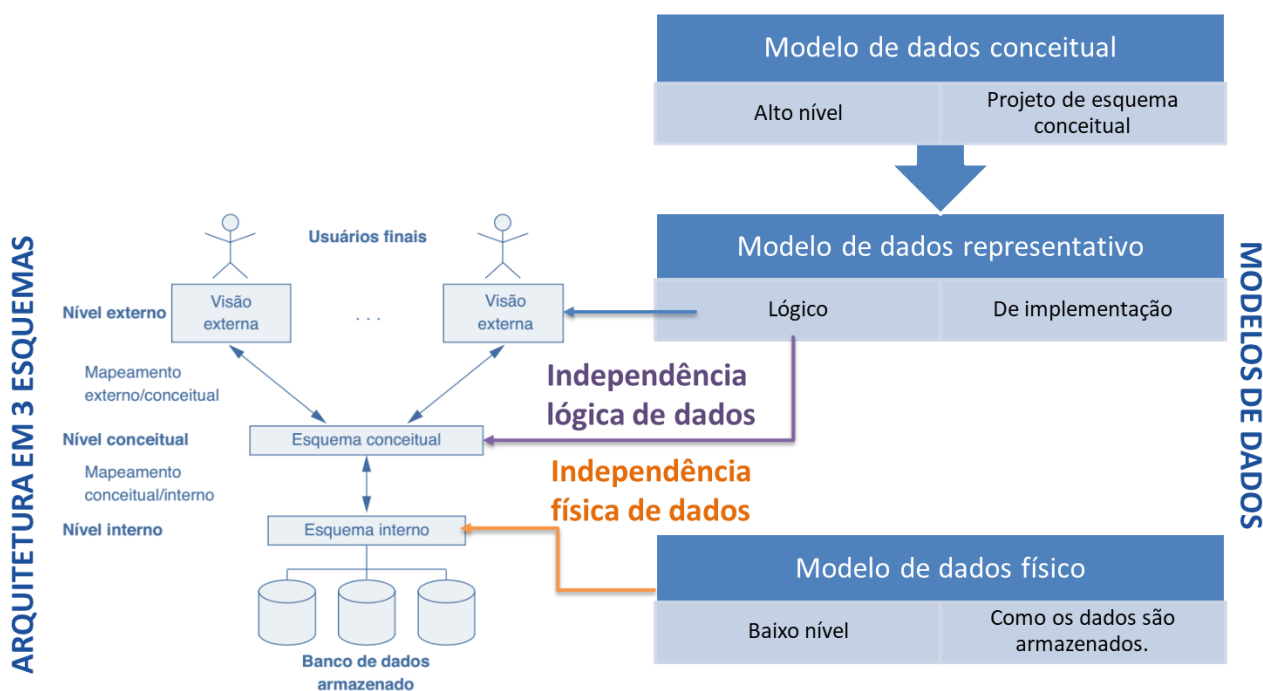
Abaixo deste nível temos o **nível de lógico** na **hierarquia de abstração** e o **nível conceitual** na **arquitetura em três esquemas**. Você deve estar se perguntando como esses níveis se juntam com os modelos de dados? Os modelos de dados conceituais dão origem à modelos de dados lógicos que servem para representar tanto esquemas externos quanto esquemas conceituais da arquitetura em 3 esquemas. É interessante dizer que alguns autores chamam os modelos de dados conceituais de modelos lógicos baseados em objetos e os modelos de dados representativos de modelos lógicos baseados em registros. Por fim, temos o nível físico ou interno. Esse é igual em todas as hierarquias. Este nível é responsável pela forma de organização dos dados no disco e por descrever os métodos de acesso para ele.

Para finalizar temos que lembrar do conceito de independência de dados que trata da capacidade de alterar o esquema em um nível do sistema de banco de dados sem ter que alterar o esquema nos outros níveis. A independência de dados lógicos é a capacidade de alterar o esquema conceitual sem afetar os esquemas externos ou os programas de aplicativos. O esquema conceitual pode ser alterado devido a mudanças nas restrições ou adição de novos itens de dados ou a remoção de itens de dados existentes.

Já a independência de dados físicos é a capacidade de alterar o esquema interno sem afetar o esquema conceitual ou externo. Um esquema interno pode ser alterado por vários motivos, como por exemplo, para criar uma estrutura de acesso adicional, alterar a estrutura de armazenamento etc. A separação do esquema interno do esquema conceitual facilita a independência física dos dados.

Vejamos então as listas de tópicos e a figura com o relacionamento entre elas:

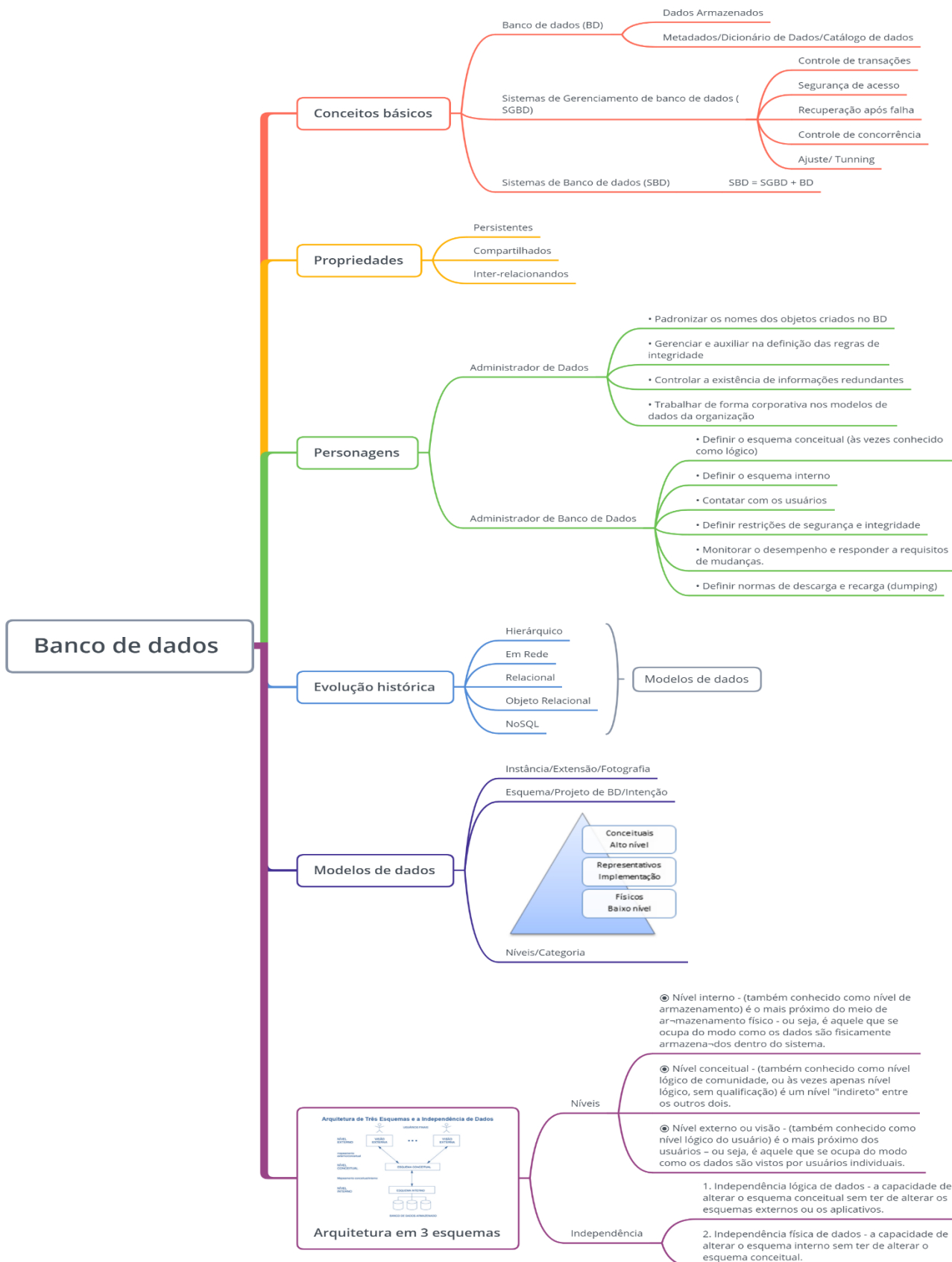
**Modelo de dados:** conceitual (alto nível), representativos (lógico ou de implementação) e físicos  
**Níveis de abstração de dados:** nível de view, nível lógico e nível físico.  
**Arquitetura 3 esquemas:** visão externa, esquema conceitual e esquema interno.  
**Arquitetura 3 esquemas (níveis):** nível externo, nível conceitual e nível interno.  
**Independência de dados:** lógica e física.



A figura mostra como grudar a hierarquia dos modelos de dados com a hierarquia dos níveis da arquitetura em 3 esquemas. Tanto o nível externo quanto o conceitual da arquitetura em 3 esquemas são representados por esquemas descritos por modelos lógicos.



# MAPA MENTAL



## QUESTÕES COMENTADAS

Nas próximas linhas comentaremos algumas questões da FGV. Por ser a banca do seu concurso, sugiro que você preste mais atenção na forma como a banca cobra o assunto. Qualquer dúvida estou às ordens! Forte abraço e bons estudos!



### 1. Ano: 2009 Banca: FGV Órgão: MEC Cargo: Analista de Sistemas

A figura abaixo ilustra a administração de recursos de dados.



O estabelecimento e manutenção do dicionário de dados, o desenvolvimento e aplicação de políticas da propriedade e acesso aos dados e preparação dos planos estratégicos e técnicos para os bancos de dados são atividades a serem implementadas, respectivamente, nas seguintes etapas:

- A) Administração de Banco de Dados, Planejamento de Dados e Administração de Dados
- B) Administração de Banco de Dados, Administração de Dados e Planejamento de Dados
- C) Planejamento de Dados, Administração de Banco de Dados e Administração de Dados
- D) Administração de Dados, Administração de Banco de Dados e Planejamento de Dados
- E) Administração de Dados, Planejamento de Dados e Administração de Banco de Dados

**Comentário:** Essa questão trata do tema de administração de recursos de dados. É uma perspectiva paralela à abordagem usada pelos livros clássicos de banco de dados. A administração de recursos de dados inclui:

**Administração de Bancos de Dados.** É uma importante função de gerenciamento de recursos de dados responsável pelo **uso adequado da tecnologia** de gerenciamento de bancos de dados. A administração de bancos de dados possui mais **responsabilidades**





**operacionais e técnicas** do que outras funções de gerenciamento de recursos de dados. Isto inclui responsabilidade pelo:

- Desenvolvimento e manutenção do dicionário de dados da organização.
- Projeto e monitoração do desempenho dos bancos de dados
- Aplicação de padrões para uso e segurança dos bancos de dados.

**Planejamento de Dados.** O planejamento de dados é uma função de planejamento e análise empresarial que se concentra no gerenciamento de recursos de dados. Ela inclui a responsabilidade pelo:

- Desenvolvimento de uma arquitetura global de dados para os recursos de dados da empresa vinculada à missão e planos estratégicos da empresa e aos objetivos e processos de suas unidades de negócios.
- O planejamento de dados é um componente importante do processo de planejamento estratégico de uma organização. Ele é feito por organizações que assumiram um compromisso formal com o planejamento de longo alcance para o uso e administração estratégicos de seus recursos de dados.

**Administração de Dados.** É outra função da administração de recursos de dados. Ela envolve:

- O estabelecimento e execução de políticas e procedimentos para gerenciamento de dados como um recurso estratégico das empresas.
- A administração da coleta, armazenamento e disseminação de todos os tipos de dados de tal forma que os dados se tornem um recurso padronizado disponível para todos os usuários finais na organização.
- O planejamento e controle de dados no apoio às funções e objetivos estratégicos de uma organização.
- O estabelecimento de uma atividade de planejamento de dados para a organização.
- O desenvolvimento de políticas e definição de padrões para arranjos de concepção, processamento e segurança e seleção do software para gerenciamento de bancos de dados e dicionário de dados.

Após analisarmos as considerações acima, podemos marcar nossa resposta na alternativa B

**Gabarito: B.**



**2. ANO: 2014 BANCA: FGV ÓRGÃO: DPE-RJ PROVA: TÉCNICO SUPERIOR ESPECIALIZADO - ADMINISTRADOR DE DADOS**

A sigla DBA é usualmente empregada para designar um profissional que é responsável



A pela administração, instalação, configuração e monitoramento de instalações de bancos de dados.

B pela concepção, desenvolvimento e implantação de sistemas gerenciadores de bancos de dados.

C pelo processo de coleta e exploração de dados para geração de novos conhecimentos.

D pelo teste e manutenção de sistemas WEB construídos sob a égide de metodologias ágeis.

E pelo treinamento e certificação para utilização de gerenciadores de bancos de dados.

**Comentário:** Vejam que a questão que melhor se encaixa com a definição que vimos nesta aula é a descrita na alternativa A.

**Gabarito: A.**



### 3. Ano: 2016 Banca: FGV Órgão: IBGE Cargo: Analista - Análise e Desenvolvimento de Aplicações

Considere as seguintes características de um projeto de banco de dados.

- I. O modelo de dados é conhecido a priori e é estável;
- II. A integridade dos dados deve ser rigorosamente mantida;
- III. Velocidade e escalabilidade são preponderantes.

Dessas características, o emprego de bancos de dados NoSQL é favorecido somente por:

- A) I;
- B) I e II;
- C) II;
- D) II e III;
- E) III.

**Comentário:** Esse assunto será visto novamente na aula de Big Data, contudo, se lembramos da evolução dos SGBDs explicada no início desta aula podemos responder à questão. A grande característica dos bancos NoSQL é dar suporte à escalabilidade e velocidade no processamento dos dados. Com essa informação observamos que o item III está correto.

As demais alternativas estão erradas. No item I, a falha é dizer que o modelo de dados é conhecido a priori, na realidade ele não é conhecido e também não é estável. Já a alternativa II diz que a integridade deve ser mantida, sabemos que os sistemas NoSQL tem o conceito de *soft state* que admite pequenos momentos de inconsistência parcial em favor da performance.



**Gabarito: E.**



**4. Ano: 2009 Banca: FGV Órgão: MEC Cargo: Analista de Sistemas**

O Diagrama Entidade-Relacionamento, proposto por P. Chen, é uma ferramenta tipicamente utilizada para a elaboração do seguinte modelo de dados:

- A) físico
- B) interno
- C) externo
- D) conceitual
- E) hierárquico

**Comentário:** Sabemos que os modelos de dados são classificados em três categorias: conceitual, lógico e físico. Ou conforme já apresentamos anteriormente:

**Modelo de dados:** conceitual (alto nível), representativos (lógico ou de implementação) e físicos

O modelo entidade-relacionamento de Peter Chen é considerado um modelo conceitual e foi tratado em detalhes no decorrer desta aula. Desta forma, podemos marcar nossa resposta na alternativa D.

**Gabarito: D.**



**5. Ano: 2010 Banca: FGV Órgão: DETRAN-RN Cargo: Administrador de banco de dados**

NÃO corresponde a um aspecto da modelagem de dados:

- A) Observação.
- B) Representação dos objetos.
- C) Verificação de fidelidade e carências.
- D) Validações.
- E) Operação de semijunção.

**Comentário:** A questão nos apresenta alguns aspectos da modelagem de dados que é a primeira etapa de um projeto que envolva banco de dados e tem como seu principal objetivo



o desenvolvimento de um modelo que contenha entidades e relacionamentos. Numa visão mais específica de banco de dados, um modelo de dados consiste de três componentes: uma coleção de tipos de objetos, uma coleção de operadores e uma coleção de regras de integridades.

Nesta situação estamos **avaliando ou observando** o contexto que servirá de referência para a construção do modelo. Precisamos entender quais **os objetos** farão parte do nosso banco de dados. Neste momento verificamos a **fidelidade das informações** que serão obtidas e possíveis **carências**. Por fim, precisamos de regra de integridade mínimas que farão a **validação** do modelo.

Percebam que **não** existe necessidade de avaliarmos as operações de **semijunção** neste momento. Essa operação aparece na teoria de banco de dados distribuídos quando queremos fazer a junção de tabelas dispersas geograficamente, portando, não tem nenhuma relação com modelagem. Sendo, então a **alternativa E** a resposta da questão.

**Gabarito: E.**



#### **6. Ano: 2009 Banca: FGV Órgão: MEC Cargo: Administrador de banco de dados**

Em um Sistema Gerenciador de Banco de Dados - SGBD, os elementos de dados possuem uma estrutura lógica visualizada na forma de tabelas.

Nesse sentido, existem inclusive, pacotes nos SGBDs que podem vincular elementos oriundos de várias tabelas desde que essas compartilhem elementos comuns de dados.

Essa estrutura lógica de dados é conhecida como do tipo:

- A) em rede.
- B) relacional.
- C) em pilha.
- D) hierárquica.
- E) em árvore.

**Comentário:** Vejam que essa é uma questão simples que aborda a definição do modelo relacional. Veremos mais detalhes sobre ele na próxima aula, mas vamos aproveitar a questão para relembrar a definição dos bancos de dados em rede, hierárquico e relacional:

**MODELO HIERÁRQUICO** (HDS – Hierarchical Database System). Surgiu na década de 1960 com a primeira linguagem de banco de dados: a **DL/I** desenvolvida pela IBM e a North American Aviation. Organiza os dados de cima para baixo, como uma árvore. Cada registro é dividido em partes denominadas segmentos. O **banco de dados se assemelha a um organograma** com um segmento raiz e um número qualquer de segmentos subordinados.

**MODELO EM REDE** (NDS - Network Database System). Definido pelo DBTG (DataBase Task Group) do comitê do CODASYL (Conference on Data Systems Language) a partir de



1971. Esse modelo é **uma extensão do modelo hierárquico**. Os registros são organizados no banco de dados por um **conjunto arbitrário de gráficos**.

**MODELO RELACIONAL** (Relational Model). Foi apresentado por Edgard F. Codd (IBM) em seu artigo *A Relational Model of Data for Large Shared Data Banks* (1970). Foi o evento mais importante na história recente da área de banco de dados. O objetivo do modelo é **representar os dados de forma mais simples**, através de um de **conjuntos de tabelas inter-relacionadas**. Este modelo abandona os conceitos anteriores, tornando os bancos de dados mais flexíveis, tanto na forma de representar as relações entre os dados, como na tarefa de modificação de sua estrutura, sem ter que reconstruir todo o banco de dados.

Pela teórica exposta na descrição dos modelos podemos concluir que a resposta é de fato o modelo relacional, presente na alternativa B.

**Gabarito: B.**



## 7. Ano: 2009 Banca: FGV Órgão: MEC Cargo: Administrador de banco de dados

Para os diversos tipos de aplicativos com bancos de dados, a arquitetura cliente/servidor oferece as vantagens relacionadas a seguir, **À EXCEÇÃO DE UMA**. Assinale-a.

- A) Tráfego de rede reduzido, devido à transmissão de dados mais eficiente. Apenas os dados que o aplicativo precisa são transferidos.
- B) Melhor flexibilidade, devido aos servidores de banco de dados em vez de bancos de dados de compartilhamento de arquivos.
- C) Se mais usuários são acrescentados ao sistema, é mais simples escalar as capacidades do sistema para atingir os requerimentos adicionais.
- D) Operações mais confiáveis e robustas, porque várias cópias de um banco de dados interagem com os dados.
- E) Capacidade de "missão crítica", como logs de transações, capacidades de backup sofisticadas, vetores redundantes de disco e ferramentas de recuperação de falhas.

**Comentário:** Na arquitetura Cliente-Servidor, o cliente (front\_end) executa as tarefas do aplicativo, ou seja, fornece a interface do usuário (tela, e processamento de entrada e saída). O servidor (back\_end) executa as consultas no DBMS e retorna os resultados ao cliente. Apesar de ser uma arquitetura bastante popular, são necessárias soluções sofisticadas de software que possibilitem: o tratamento de transações, as confirmações de transações (commits), desfazer transações (rollbacks), linguagens de consultas (stored procedures) e gatilhos (triggers). A principal vantagem desta arquitetura é a divisão do processamento entre dois sistemas, o que **reduz o tráfego de dados na rede**.

Esta arquitetura se tornou muito popular por algumas razões. Primeiro, a facilidade de implementação dada a clara separação das funcionalidades e dos servidores. Segundo, um servidor é inteligentemente utilizado porque as tarefas mais simples são delegadas às



máquinas clientes mais baratas. Terceiro, o usuário pode executar uma interface gráfica que lhe é familiar, ao invés de usar a interface do servidor. Desta maneira, a arquitetura cliente-servidor foi incorporada aos SGBDs comerciais. Diferentes técnicas foram propostas para se implementar essa arquitetura, sendo que a mais adotada pelos Sistemas Gerenciadores de Banco de Dados Relacionais (SGBDRs) comerciais é a inclusão da funcionalidade de **um SGBD centralizado** no lado do servidor.

Veja que a arquitetura cliente servidor não está preocupada com a distribuição dos dados em diferentes servidores. Desta forma, a alternativa D está incorreta, sendo a resposta da questão.

**Gabarito: D.**



### **8. Ano: 2009 Banca: FGV Órgão: MEC Cargo: Desenvolvedor**

Acerca do nível de visões de um banco de dados, analise as afirmativas a seguir:

- I. Fornecem mecanismos de segurança, restringindo o acesso dos usuários.
- II Os usuários vêem um conjunto de aplicações e visualizam os detalhes dos tipos de dados.
- III. O nível de visões é definido para simplificar a interação entre o usuário final e o banco de dados.

Assinale:

- A) se somente a afirmativa I estiver correta.
- B) se somente as afirmativas I e II estiverem corretas.
- C) se somente as afirmativas I e III estiverem corretas.
- D) se somente as afirmativas II e III estiverem corretas.
- E) se todas as afirmativas estiverem corretas.

**Comentário:** Sabemos de o nível externo ou de visão possui esquemas externos ou **visões de usuários**. Cada esquema externo descreve a visão da base de dados de um grupo de usuários da base de dados. Cada visão descreve, tipicamente, a parte da base de dados que um particular grupo de usuários está interessado e esconde deste o restante da base de dados. Um modelo de dados de alto-nível ou um modelo de dados de implementação podem ser usados neste nível.

A visão é considerada também um dispositivo de segurança, evitando que usuários não autorizados tenham acesso a partes do sistema que não façam parte do seu escopo de trabalho. Considerando esse aspecto e a definição vista no parágrafo anterior podemos concluir que a alternativa correta está na letra C.

No nível de visão, aspectos internos de implementação não são exibidos como detalhes de tipos de dados, sendo assim, o item II está incorreto.



**Gabarito: C.**



**9. Ano: 2010 Banca: FGV Órgão: BADESC Cargo: Analista de Sistemas**

A propriedade de uma transação denominada de ACID envolve os seguintes conceitos:

- A) Atômico, Concorrente, Isolamento e Durabilidade.
- B) Atomicidade, Consistência, Isolamento e Durabilidade.
- C) Atomicidade, Concorrente, Inconsistente e Durabilidade.
- D) Atomicidade, Consistência, Integridade e Dependência de dados.
- E) Atomicidade, Consistência, Isolamento e Dependência de dados.

**Comentário:** As propriedades de uma transação são atomicidade, consistência, isolamento e durabilidade. Trataremos esse assunto com maiores detalhes na aula de transações e controle de concorrência.

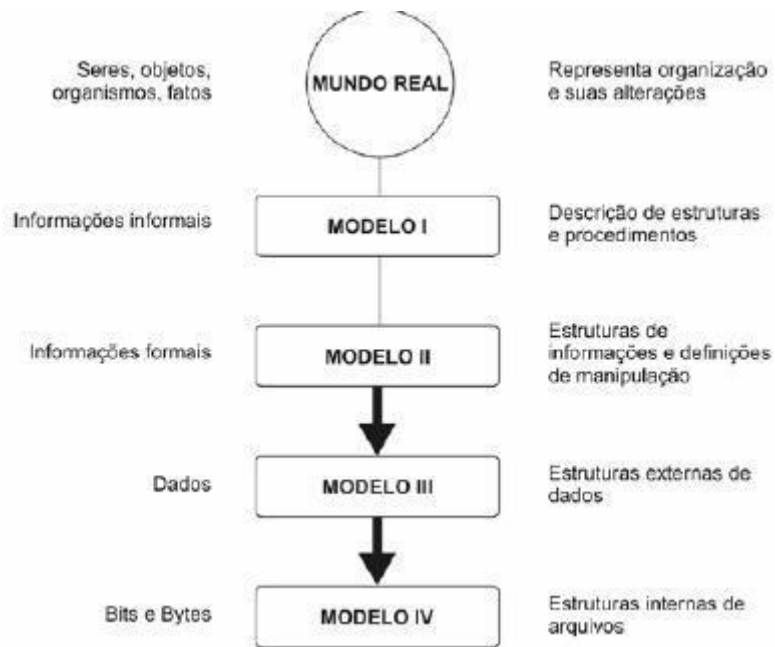
**Gabarito: B.**



**10. Ano: 2010 Banca: FGV Órgão: SEAD-AP Cargo: Auditor da Receita do Estado**

Um Banco de Dados é definido como uma coleção de dados interrelacionados, representando informações sobre um domínio específico. Nesse sentido, observe a figura abaixo, que ilustra os níveis de abstração de um banco de dados.



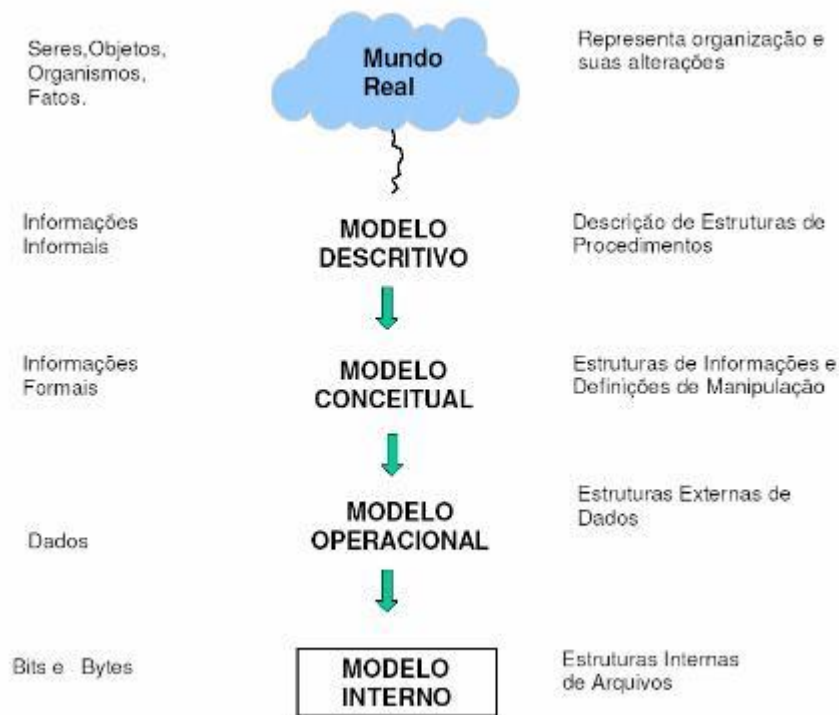


Os modelos III e IV são denominados operacional e interno; os modelos I e II são conhecidos, respectivamente por:

- A) descritivo e conceitual.
- B) tático e descritivo.
- C) estratégico e tático.
- D) gerencial e estratégico.
- E) conceitual e gerencial.

**Comentário:** Essa figura foi retirada do livro do Waldemar Setzer, já foi usada por algumas bancas de concursos, ele difere um pouco as abstrações de banco de dados, vejam na figura abaixo:





Analisando a figura podemos marcar sem medo a gabarito na alternativa A.

**Gabarito: A**



### 11. Ano: 2013 Banca: FGV Órgão: MPE-MS Cargo: Analista - Análise de Sistemas

Com relação ao tema “Sistemas gerenciadores de bancos de dados”, analise as afirmativas a seguir.

I. Cabe ao administrador de banco de dados de uma organização definir e modificar esquemas por meio da execução de instruções de definição de dados, conceder diferentes tipos de autorização para acesso a dados e manter rotinas que têm por objetivo garantir a integridade do sistema.

II. Cabe ao administrador de banco de dados de uma organização escolher, dentre as diversas ferramentas disponíveis no mercado, as mais adequadas e desenvolver aplicações que utilizem o sistema gerenciador de bancos de dados e que ofereçam interfaces amigáveis para os usuários finais.

III. Cabe ao administrador de banco de dados de uma organização acessar o banco de dados e monitorar as tarefas que estão sendo executadas pelos usuários e garantir que o desempenho do sistema não seja prejudicado por elas.

Assinale:



- A) se somente a afirmativa I estiver correta.
- B) se somente a afirmativa II estiver correta.
- C) se somente a afirmativa III estiver correta.
- D) se somente as afirmativas II e III estiverem corretas.
- E) se somente as afirmativas I e III estiverem corretas.

**Comentário:** Percebam que os itens I e III estão de acordo com as tarefas desenvolvidas pelo DBA. O item II tem um erro ao afirmar que o DBA desenvolve sistemas, sabemos que isso não é verdade.

**Gabarito: E.**



## EXERCÍCIOS FGV

Nas próximas linhas comentaremos algumas questões da FGV. Por ser a banca do seu concurso, sugiro que você preste mais atenção na forma como a banca cobra o assunto. Qualquer dúvida estou às ordens! Forte abraço e bons estudos!



### 1. Ano: 2009 Banca: FGV Órgão: MEC Cargo: Analista de Sistemas

A figura abaixo ilustra a administração de recursos de dados.



O estabelecimento e manutenção do dicionário de dados, o desenvolvimento e aplicação de políticas da propriedade e acesso aos dados e preparação dos planos estratégicos e técnicos para os bancos de dados são atividades a serem implementadas, respectivamente, nas seguintes etapas:

- A) Administração de Banco de Dados, Planejamento de Dados e Administração de Dados
- B) Administração de Banco de Dados, Administração de Dados e Planejamento de Dados
- C) Planejamento de Dados, Administração de Banco de Dados e Administração de Dados
- D) Administração de Dados, Administração de Banco de Dados e Planejamento de Dados
- E) Administração de Dados, Planejamento de Dados e Administração de Banco de Dados



### 2. ANO: 2014 BANCA: FGV ÓRGÃO: DPE-RJ PROVA: TÉCNICO SUPERIOR ESPECIALIZADO - ADMINISTRADOR DE DADOS



- A sigla DBA é usualmente empregada para designar um profissional que é responsável
- A pela administração, instalação, configuração e monitoramento de instalações de bancos de dados.
  - B pela concepção, desenvolvimento e implantação de sistemas gerenciadores de bancos de dados.
  - C pelo processo de coleta e exploração de dados para geração de novos conhecimentos.
  - D pelo teste e manutenção de sistemas WEB construídos sob a égide de metodologias ágeis.
  - E pelo treinamento e certificação para utilização de gerenciadores de bancos de dados.



### 3. Ano: 2016 Banca: FGV Órgão: IBGE Cargo: Analista - Análise e Desenvolvimento de Aplicações

Considere as seguintes características de um projeto de banco de dados.

- I. O modelo de dados é conhecido a priori e é estável;
- II. A integridade dos dados deve ser rigorosamente mantida;
- III. Velocidade e escalabilidade são preponderantes.

Dessas características, o emprego de bancos de dados NoSQL é favorecido somente por:

- A) I;
- B) I e II;
- C) II;
- D) II e III;
- E) III.



### 4. Ano: 2009 Banca: FGV Órgão: MEC Cargo: Analista de Sistemas

O Diagrama Entidade-Relacionamento, proposto por P. Chen, é uma ferramenta tipicamente utilizada para a elaboração do seguinte modelo de dados:

- A) físico
- B) interno
- C) externo
- D) conceitual



E) hierárquico



**5. Ano: 2010 Banca: FGV Órgão: DETRAN-RN Cargo: Administrador de banco de dados**

NÃO corresponde a um aspecto da modelagem de dados:

- A) Observação.
- B) Representação dos objetos.
- C) Verificação de fidelidade e carências.
- D) Validações.
- E) Operação de semijunção.



**6. Ano: 2009 Banca: FGV Órgão: MEC Cargo: Administrador de banco de dados**

Em um Sistema Gerenciador de Banco de Dados - SGBD, os elementos de dados possuem uma estrutura lógica visualizada na forma de tabelas.

Nesse sentido, existem inclusive, pacotes nos SGBDs que podem vincular elementos oriundos de várias tabelas desde que essas compartilhem elementos comuns de dados.

Essa estrutura lógica de dados é conhecida como do tipo:

- A) em rede.
- B) relacional.
- C) em pilha.
- D) hierárquica.
- E) em árvore.



**7. Ano: 2009 Banca: FGV Órgão: MEC Cargo: Administrador de banco de dados**

Para os diversos tipos de aplicativos com bancos de dados, a arquitetura cliente/servidor oferece as vantagens relacionadas a seguir, **À EXCEÇÃO DE UMA**. Assinale-a.



- A) Tráfego de rede reduzido, devido à transmissão de dados mais eficiente. Apenas os dados que o aplicativo precisa são transferidos.
- B) Melhor flexibilidade, devido aos servidores de banco de dados em vez de bancos de dados de compartilhamento de arquivos.
- C) Se mais usuários são acrescentados ao sistema, é mais simples escalar as capacidades do sistema para atingir os requerimentos adicionais.
- D) Operações mais confiáveis e robustas, porque várias cópias de um banco de dados interagem com os dados.
- E) Capacidade de "missão crítica", como logs de transações, capacidades de backup sofisticadas, vetores redundantes de disco e ferramentas de recuperação de falhas.



#### **8. Ano: 2009 Banca: FGV Órgão: MEC Cargo: Desenvolvedor**

Acerca do nível de visões de um banco de dados, analise as afirmativas a seguir:

- I. Fornecem mecanismos de segurança, restringindo o acesso dos usuários.
- II Os usuários vêem um conjunto de aplicações e visualizam os detalhes dos tipos de dados.
- III. O nível de visões é definido para simplificar a interação entre o usuário final e o banco de dados.

Assinale:

- A) se somente a afirmativa I estiver correta.
- B) se somente as afirmativas I e II estiverem corretas.
- C) se somente as afirmativas I e III estiverem corretas.
- D) se somente as afirmativas II e III estiverem corretas.
- E) se todas as afirmativas estiverem corretas.



#### **9. Ano: 2010 Banca: FGV Órgão: BADESC Cargo: Analista de Sistemas**

A propriedade de uma transação denominada de ACID envolve os seguintes conceitos:

- A) Atômico, Concorrente, Isolamento e Durabilidade.
- B) Atomicidade, Consistência, Isolamento e Durabilidade.
- C) Atomicidade, Concorrente, Inconsistente e Durabilidade.
- D) Atomicidade, Consistência, Integridade e Dependência de dados.

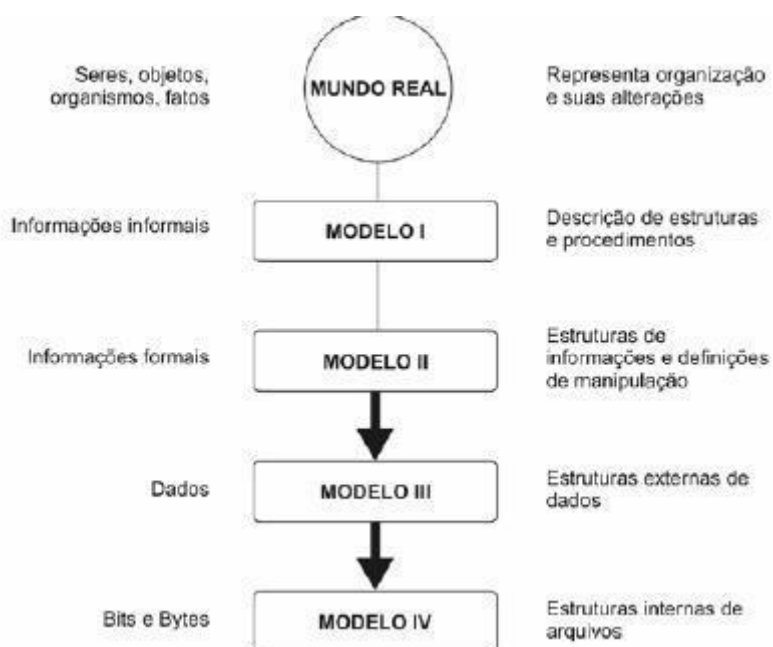


E) Atomicidade, Consistência, Isolamento e Dependência de dados.



**10. Ano: 2010 Banca: FGV Órgão: SEAD-AP Cargo: Auditor da Receita do Estado**

Um Banco de Dados é definido como uma coleção de dados interrelacionados, representando informações sobre um domínio específico. Nesse sentido, observe a figura abaixo, que ilustra os níveis de abstração de um banco de dados.



Os modelos III e IV são denominados operacional e interno; os modelos I e II são conhecidos, respectivamente por:

- A) descritivo e conceitual.
- B) tático e descritivo.
- C) estratégico e tático.
- D) gerencial e estratégico.
- E) conceitual e gerencial.



**11. Ano: 2013 Banca: FGV Órgão: MPE-MS Cargo: Analista - Análise de Sistemas**

Com relação ao tema "Sistemas gerenciadores de bancos de dados", analise as afirmativas a seguir.



I. Cabe ao administrador de banco de dados de uma organização definir e modificar esquemas por meio da execução de instruções de definição de dados, conceder diferentes tipos de autorização para acesso a dados e manter rotinas que têm por objetivo garantir a integridade do sistema.

II. Cabe ao administrador de banco de dados de uma organização escolher, dentre as diversas ferramentas disponíveis no mercado, as mais adequadas e desenvolver aplicações que utilizem o sistema gerenciador de bancos de dados e que ofereçam interfaces amigáveis para os usuários finais.

III. Cabe ao administrador de banco de dados de uma organização acessar o banco de dados e monitorar as tarefas que estão sendo executadas pelos usuários e garantir que o desempenho do sistema não seja prejudicado por elas.

Assinale:

A) se somente a afirmativa I estiver correta.

B) se somente a afirmativa II estiver correta.

C) se somente a afirmativa III estiver correta.

D) se somente as afirmativas II e III estiverem corretas.

E) se somente as afirmativas I e III estiverem corretas.





## GABARITO

1. B
2. A
3. E
4. D
5. E
6. B
7. D
8. C
9. B
10. A
11. E



## CONSIDERAÇÕES FINAIS

Prezados Alunos,

Chegamos, pois, ao final da aula introdutória de Banco de Dados! As videoaulas referentes ao conteúdo visto até aqui já estão gravadas e devidamente disponibilizadas na área do aluno. Espero que gostem!

Forte abraço, bons estudos e até breve! Que Deus abençoe!

Thiago Cavalcanti

## REFERÊNCIAS

Fiz uma lista com alguns livros que são referências do conteúdo apresentado na aula caso você queria se aprofundar um pouco.

1. Elmasri, Ramez. Sistemas de Bancos de Dados. Edição do Kindle.
2. Introdução a sistemas de bancos de dados - By C. J. Date - Elsevier Brasil, 2004 - 865 pages
3. Sistema de Banco de Dados - Abraham Silberschatz, Henry F. Korth, S. Sudarshan - Editora: ELSEVIER BRASIL
4. Heuser, Carlos Alberto. Projeto de banco de dados - V4 - UFRGS. Edição do Kindle.
5. Mannino, Michael V.. Projeto, Desenvolvimento de Aplicações e Administração de Banco de Dados. Edição do Kindle.



THIAGO CAVALCANTI  
PROFESSOR



# ESSA LEI TODO MUNDO CONHECE: PIRATARIA É CRIME.

Mas é sempre bom revisar o porquê e como você pode ser prejudicado com essa prática.



**1** Professor investe seu tempo para elaborar os cursos e o site os coloca à venda.



**2** Pirata divulga ilicitamente (grupos de rateio), utilizando-se do anonimato, nomes falsos ou laranjas (geralmente o pirata se anuncia como formador de "grupos solidários" de rateio que não visam lucro).



**3** Pirata cria alunos fake praticando falsidade ideológica, comprando cursos do site em nome de pessoas aleatórias (usando nome, CPF, endereço e telefone de terceiros sem autorização).



**4** Pirata compra, muitas vezes, clonando cartões de crédito (por vezes o sistema anti-fraude não consegue identificar o golpe a tempo).



**5** Pirata fere os Termos de Uso, adultera as aulas e retira a identificação dos arquivos PDF (justamente porque a atividade é ilegal e ele não quer que seus fakes sejam identificados).



**6** Pirata revende as aulas protegidas por direitos autorais, praticando concorrência desleal e em flagrante desrespeito à Lei de Direitos Autorais (Lei 9.610/98).



**7** Concurseiro(a) desinformado participa de rateio, achando que nada disso está acontecendo e esperando se tornar servidor público para exigir o cumprimento das leis.



**8** O professor que elaborou o curso não ganha nada, o site não recebe nada, e a pessoa que praticou todos os ilícitos anteriores (pirata) fica com o lucro.



Deixando de lado esse mar de sujeira, aproveitamos para agradecer a todos que adquirem os cursos honestamente e permitem que o site continue existindo.