**Unidade 1 | Fundamentos de Bancos de Dados**

UML (*Unified Modeling Language*): Modelo de Linguagem Unificada

CASEs (do inglês *Computer-Aided Software Engineering*)

(SGBD) Sistema Gerenciador de Banco de Dados

Nessa fase vamos imaginar que precisamos atender 03 clientes distintos: Precisamos indicar o SGBD mais propício para cada cliente;

Pequeno Porte Médio Porte Grande Porte

No **Projeto** precisamos saber quais serão as **aplicações** que utilizaram o banco de dados, é necessário esse dimensionamento para poder indicar o SGBD mais apropriado;

O analista de sistemas deverá propor soluções, levando em consideração o volume de informações que deverá ser armazenado.

Digamos que recebi uma promoção a Analista de sistemas: Terei mais responsabilidades e a chance de obter novos conhecimentos, além de precisar visitar clientes, levantar os requisitos do software e projetar soluções para os clientes da empresa em que você trabalha.

O termo **persistente** faz referência ao **armazenamento** dos dados que só podem ser apagados do banco de dados através de alguma função específica e, claro, por alguém que tenha a permissão de exclui-los do banco de dados.

Assimile

Um **SGBD** é um **software** cuja finalidade de **gerenciar** as informações de um banco de dados (também chamada de base de dados) segundo Navathe e Ramez (2005), e que também devem **organizar**, **acessar** **controlar** e **proteger** as informações contidas no banco de dados.

O SGBD tem por objetivo facilitar a vida do programador ou analista, deixando livre para pensar na modelagem e não ficar pensando em questões técnicas de armazenamento de dados (sendo esta uma das funções do SGBD).

Sistema Gerenciador de Banco de Dados é constituído por um conjunto de dados associados a um conjunto de programas para acesso a esses dados.

Caso o **SGBD** esteja em **locais físicos diferente**s, cada um passa a receber o nome de **nó**, e uma operação realizada no banco de dados pode ser executada em um ou em mais nós.

A **integridade** é a garantia de que a informação armazenada no banco de dados **esteja correta.**

Dois vendedores acessam simultaneamente o registro e vendem a geladeira para seus clientes. Com certeza, um cliente ficaria sem a geladeira, gerando muitos conflitos tanto para o cliente, quanto para a loja. Para este tipo de evento damos o nome de **controle de concorrência,** uma das finalidades essenciais de um **SGBD** e que, segundo Korth, Silberschatz e Sudarshan (2012), **são técnicas** utilizadas para garantir a propriedade de **isolamento de transações** que estão sendo executadas ao mesmo tempo.

O **conjunto de requisitos de um SGBD** recebe o nome de ACID dos termos em inglês A*tomicity,* C*onsistency,* I*solation,* D*urability* ou, respectivamente, **Atomicidade, Consistência, Isolamento e Durabilidade.**

Uma **transação** é uma **consequência** da efetivação de um programa (ou uma rotina) que acessa e possivelmente atualiza vários itens de dados. A transação é o resultado da execução de um programa de usuário escrito em uma linguagem de manipulação de alto nível ou em uma linguagem de programação, como Java, C# ou SQL, entre outras.

Garantia de atomicidade é que o sistema de banco de dados mantenha um registro (em disco) dos antigos valores de quaisquer dados a serem alterados.

A **consistência** preserva as regras impostas no banco de dados. Assim que a transação for finalizada, todos os dados devem estar íntegros.

O **isolamento** é a segurança de que uma transação não interfira no trabalho de outra.

**A durabilidade ou persistência** (como também é conhecida) em um meio de armazenamento confiável e seguro é um dos requisitos mais importantes de um Sistema Gerenciador de Banco de Dados.

**Principais características do uso de um banco de dados,** conforme Navathe e Ramez (2005), são as seguintes:---

-Natureza auto descritiva do SGBD.

1. -Isolamento entre os programas, os dados e a abstração dos dados.
2. -Suporte a diversas visões dos dados inseridos no banco de dados.
3. -Transações para diversos usuários do banco e a possibilidade de compartilhar os dados da base de dados.

Basicamente, os bancos de dados podem ser classificados como:

1. **Banco de dados usando sistemas hierárquicos e de rede.**
2. Os modelos hierárquicos possuíam uma estrutura semelhante à de uma árvore, muito rígida. Caso fosse necessária a adição de uma nova informação (tabela ou campo), o banco de dados em sua totalidade precisaria ser reorganizado ou redefinido.
3. Banco de dados relacional.
4. os bancos de dados relacionais foram originalmente projetados para separar a forma de armazenamento, diferenciando o projeto físico do projeto conceitual do banco de dados. Ofereceram uma flexibilidade maior no desenvolvimento dos sistemas
5. **Banco de dados relacional/objetos**.
6. os bancos de dados relacionais/objetos são uma evolução do banco de dados relacional. Com o aparecimento das linguagens orientadas a objetos, a evolução dos bancos de dados foi naturalmente surgindo.
7. **Banco de dados para Web com XML**.
8. A internet possibilitou que muitos sistemas ficassem on-line, com sistemas de banco de dados distintos trocando informações entre si. O XML (*e*X*tensible* M*arkup* L*anguage* ou linguagem de marcação que permite a integração de dados) possui um formato que, independentemente da plataforma que o sistema utiliza, permite a comunicação entre sistemas de banco de dados diferentes.**XML** (*e*X*tensible* M*arkup* L*anguage* ou linguagem de marcação que permite a integração de dados)
9. **Banco de dados para nuvem.**

A computação em nuvem está revolucionando a forma de armazenamento, de processamento dos dados e principalmente no quesito de infraestrutura, com pagamentos relacionados ao uso dos recursos disponibilizados. Os SGBDs estão migrando para os serviços em nuvens, diminuindo os custos com equipamentos e softwares, principalmente os *e-commerce*.

.

Reflita

Qual é o melhor modelo de banco de dados? A resposta é simples: aquele que melhor atende às necessidades do cliente. É necessário levar em conta fatores como a infraestrutura, os recursos disponíveis e principalmente as reais necessidades de cada cliente.

Podemos citar alguns SGBDs que mais se destacam atualmente:

**Oracle**: Não é freeware.

É utilizado em médias e grandes empresas e foi projetado para sistemas que requerem alto desempenho e segurança

**SQL Server:**

pertence à empresa Microsoft e possui versões gratuitas e pagas, sendo que as pagas são de valores bem inferiores ao seu principal concorrente (visto anteriormente, o Oracle).

É utilizado em diversos segmentos de empresas que precisam de um SGBD estável e seguro (e claro, não tão caro!).

Um dos problemas deste SGBD era sua plataforma que funcionava somente com o sistema operacional Windows. As novas versões permitem que o SGBD funcione no LINUX e em Container Docker (tecnologia que oferece um conjunto de ferramentas empacotadas e isoladas, como se estivem em um contêiner).

**MySQL**:

é *Open Source* ou código aberto e possui licenças GNU/GPL (Licença Pública Geral – *General Public License*) permitindo que qualquer usuário edite o seu código fonte de forma que atenda aos requisitos de uma determinada aplicação que está sendo implementada. Pertence a Oracle

Sua capacidade de processamento de transações é muito grande e pode ser utilizado por grandes empresas. Um fator a ser considerado é o quesito segurança. Por ser código aberto, o cuidado com falhas de segurança deve ser redobrado.

**Postgree:**

é um SGBD muito utilizado por rodar em várias plataformas de desenvolvimento como código aberto (*Open Source*) e de desenvolvimento livre.

É usado em sistemas mais robustos, em que a base de dados é muito grande, em empresas corporativas. Possui funcionalidades de controle de concorrência mais elaboradas e sofisticadas.

Existem muitos outros SGBDs

**Access da Microsoft**

Outro exemplo é o **Firebird**

o DB2 da IBM,

Pesquise mais

<https://singep.org.br/4singep/resultado/245.pdf>.

