



DBA TASKS SQL SERVER

MATERIAL EXCLUSIVO PARA ALUNO UDEMY - DESENVOLVIDO
POR ANDRÉ ROSA - IT FOREST



OBJETIVO

Capacitar o Aluno a instalar,
configurar, monitorar e
gerenciar o SQL Server

Realizar Rotinas de Backup e
Restore

Executar tarefas pró-ativas

Configurar Always on (HA)

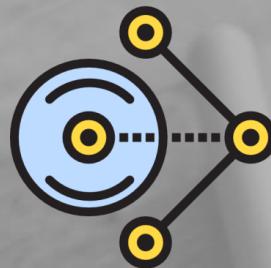
O PAPEL DO DBA

O QUE FAZ UM DBA?



Administra

- Banco de dados
- Usuários
- Segurança
- Servidores



Planeja

- Manutenções
- Migrações
- Testes



Executa

- Backup
- Restore
- Go-live



Identifica

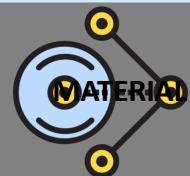
- Riscos
- Problemas



O PAPEL DO DBA

O que Faz um DBA?

- DBA-Administrador de Banco de dados
- **Planeja e executa** as manutenções em bancos de dados de produção. Pesquisa inovações tecnológicas para banco de dados. Garante que **backups** de produção estejam sendo executados. Maximiza e aprimora a **performance** do banco de dados. Garante a **segurança** dos bancos de dados de produção. **Administra** servidores de bancos de dados de produção. **Identifica** riscos de atrasos nos trabalhos. Presta suporte aos usuários, **orienta** analistas e desenvolvedores na **otimização** de **performance** das aplicações referente a área de banco de dados.



EXCLUSIVO PARA ALUNO UDEMY - DESenvolvido
POR ANDRÉ ROSA - IT FOREST



RESPONSABILIDADES

Responsabilidades

Gerência

Instala

Configura

Monitora

Lista de habilidades que um DBA deve possuir:

- ✓ Conhecimentos na linguagem estruturada SQL.
- ✓ Conhecimentos em estrutura de banco de dados, mais precisamente em diagrama entidade-relacionamento.
- ✓ Entendimento básico de Arquitetura de computadores.
- ✓ Um bom entendimento do funcionamento dos sistemas operacionais.
- ✓ Domínio de SGBDs tais como Microsoft SQL Server, PostgreSQL, MySQL, Oracle database, DB2.



EXCLUSIVO PARA ALUNO UDEMY - DESenvolvido
POR ANDRÉ ROSA - IT FOREST



ADMINISTRAÇÃO DE BACO DE DADOS

Tarefas DBA

- **1- Avaliar o hardware do Servidor de Banco de Dados:**

Avaliar e definir o hardware necessário para instalar o BD e comportar os seus dados e acessos.

- **2- Instalar o Software do Banco de Dados:**

Instalar o Software do BD, instalar atualizações e correções de bugs e tudo o que for necessário para manter a estrutura física e lógica do BD.

- **3- Planejar e Implementar o Banco de Dados:**

Definir e criar tabelas, indices e outros objetos de BD.

- **4- Criar e abrir o BD:**

Criar o BD e garantir que ele esteja disponível para os usuários.

- **5- Fazer backup do BD:**

Efetuar backups dos BD's e garantir que eles sejam *recuperáveis*.

- **6- Gerenciar usuários de BD:**

Criar usuários de BD e dar a eles privilégios de acesso aos dados, priorizando sempre a segurança dos dados, ou seja, os usuários devem ter o menor privilégio possível de acesso aos dados para evitar riscos desnecessários e proteger o BD como um todo.

- **7- Recuperar o BD em caso de falhas:**

Definir e implementar estratégias e planos para recuperar o BD em caso de falhas.

- **8- Monitorar e ajustar a performance do BD:**

Monitorar constantemente a performance do BD para identificar gargalos de desempenho e definir/implementar soluções para otimizar o seu desempenho. Manter o tempo de resposta de acesso aos dados de acordo com as expectativas do usuários é o objetivo principal desta tarefa

Autoavaliação

- Concentração.
- Conhecimento técnico.
- Trabalhar sob pressão.
- Confidencialidade

Estudos

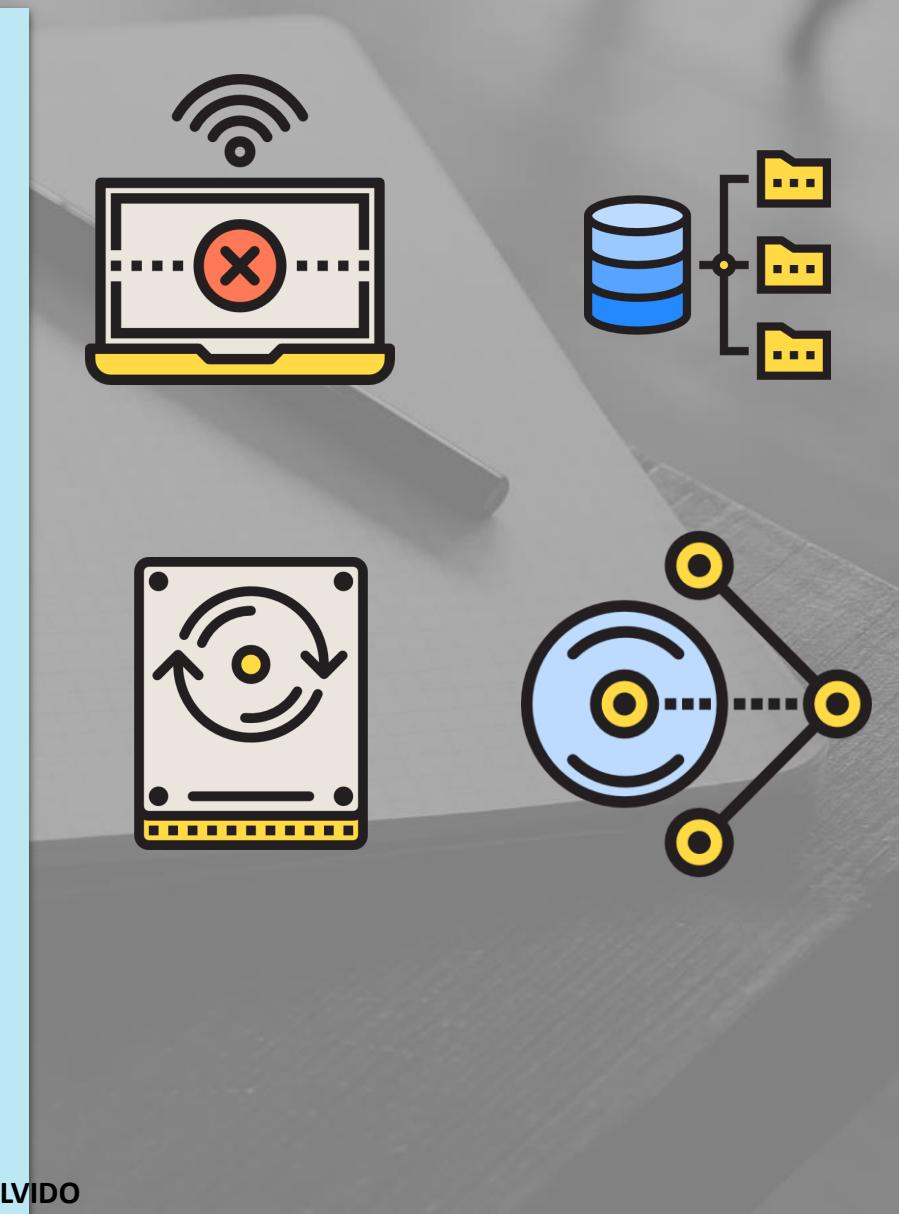
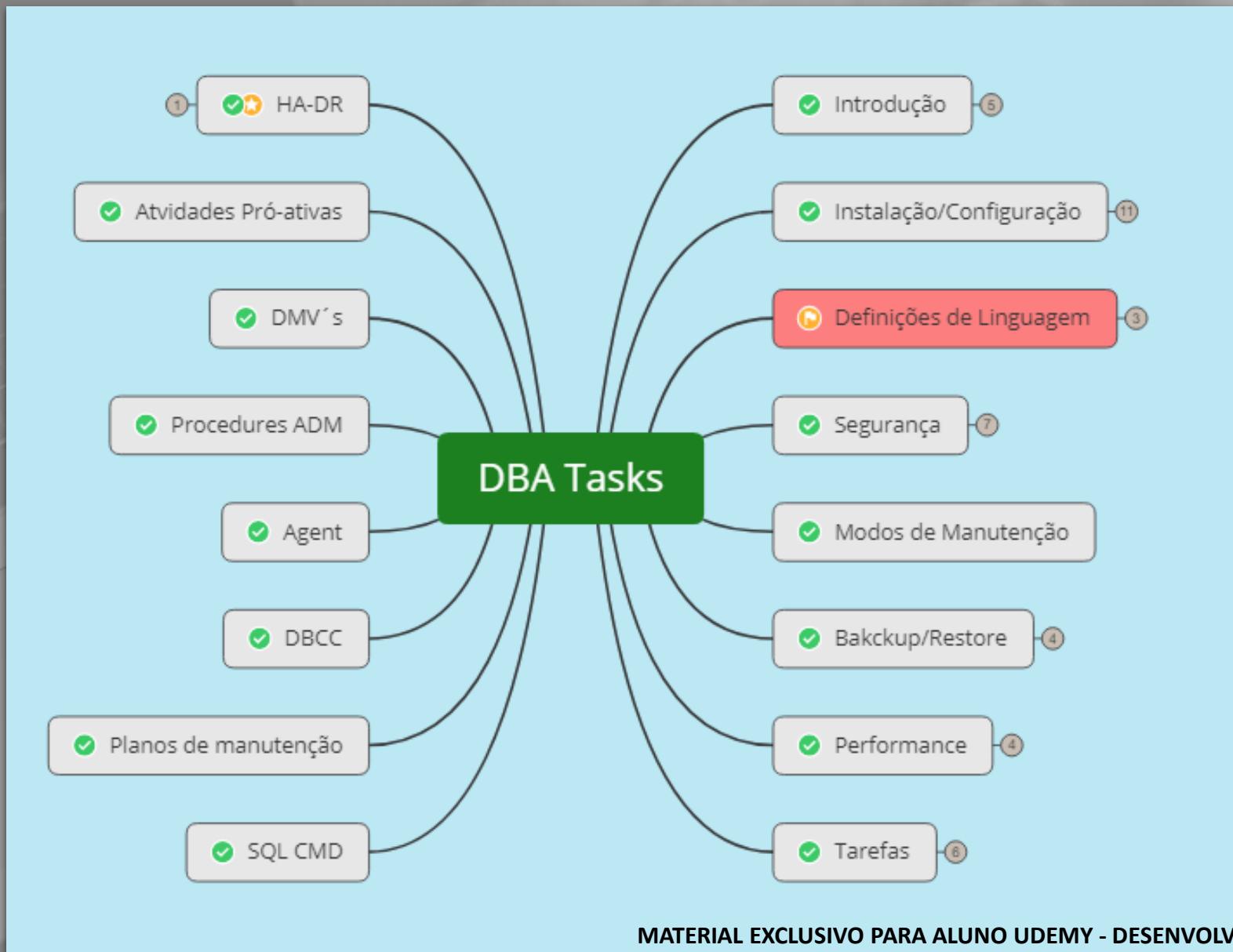
- Modelagem de dados
- Linguagem SQL
- Administração de dados
- Conhecimento continuo



EXCLUSIVO PARA ALUNO UDEMY - DESenvolvido
POR ANDRÉ ROSA - IT FOREST



AINDA SOBRE O DBA



HISTÓRICO DO SQL SERVER

Ano	Versão
1988	SQL Server
1992	SQL Server 4.2
1995	SQL Server 6.0
1996	SQL Server 6.5
1997	SQL Server 6.5 Enterprise
1998	SQL Server 7
2000	SQL Server 2000
2003	SQL Server 2000 64 Bits
2005	SQL Server 2005
2007	SQL Server 2008
2008	SQL Server 2008 R2
2012	SQL Server 2012
2014	SQL Server 2014
2016	SQL Server 2016
2017	SQL Server 2017 **
2019	SQL Server 2019***

A Microsoft anunciou que em 9 de julho de 2019 deixará de dar suporte ao SQL Server 2008 e SQL Server 2008 R2.

Serviço do SQL Server	Principal Função
Database Engine	Armazenamento de dados OLTP
Reporting Services	Relatório de dados
Analysis Services	Análise de dados OLAP
Integration Services	Fluxos de dados
Data Quality Services	Limpeza de dados
Master Data Services	Repositório único de dados
Replicação	Replicação de dados entre servidores

** Disponível em Linux

*** Apache Spark e Hadoop

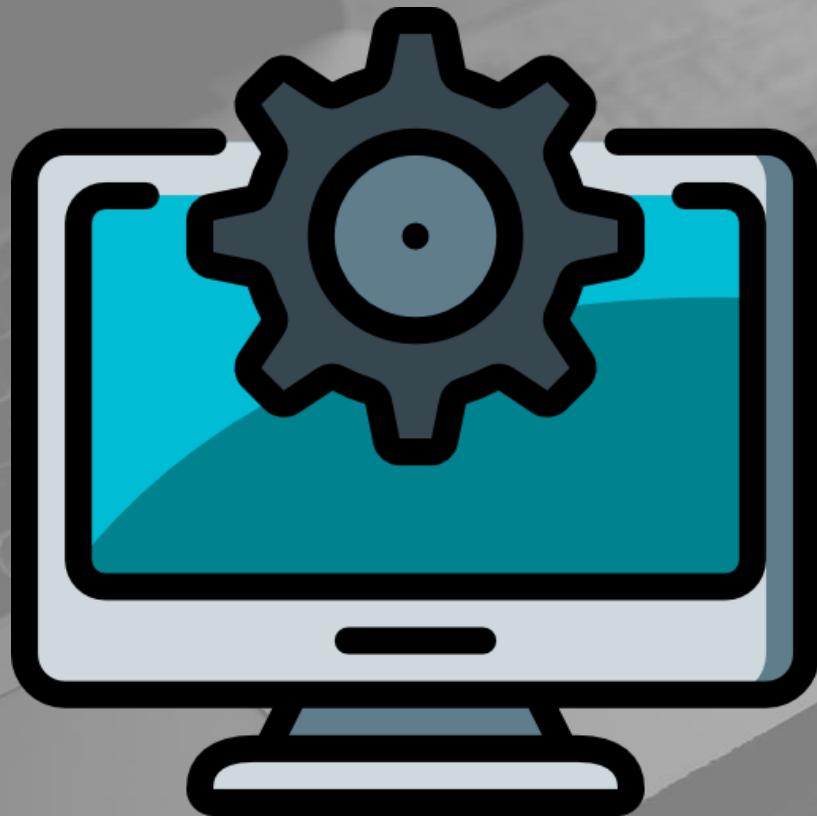
LIMITES DO SQL SERVER

SQL Server	Tamanho
Tamanho do banco de dados	524.272 terabytes
Bancos de dados por instância do SQL Server	32.767
Tamanho de arquivo (dados)	16 terabytes
Tamanho de arquivo (log)	2 terabytes
Instâncias por computador	50 instâncias em um servidor autônomo.
Linhas por tabela	Limitado pelo armazenamento disponível
Tabelas por banco de dados	Limitado pelo número de objetos em um banco de dados
Índices clusterizados por tabela	1
Colunas por instrução SELECT/INSERT	4096

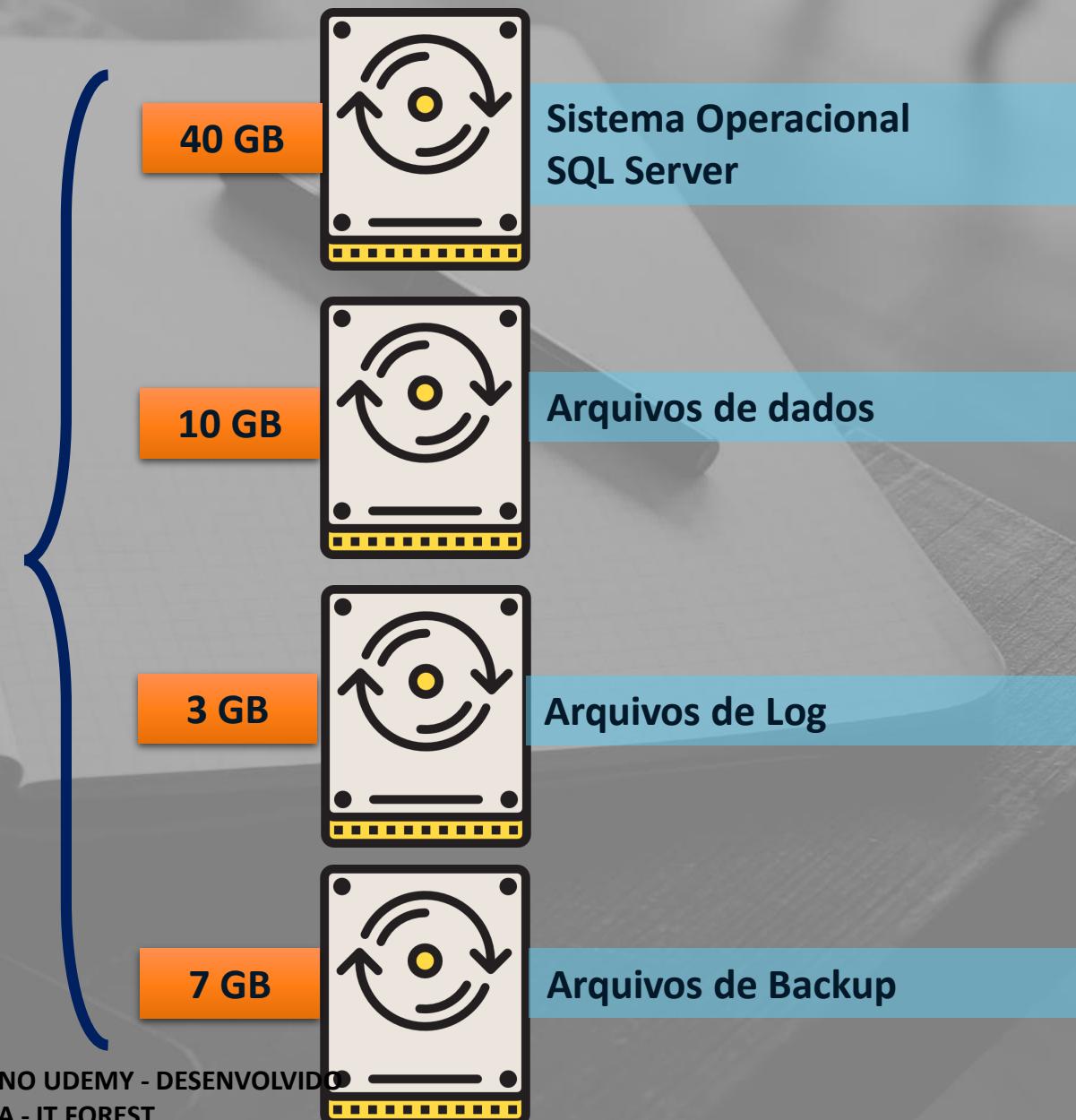
REQUISITOS SQL Server

COMPONENTE	REQUISITO MÍNIMO
Memória	1 GB
Velocidade Processador	x64: 1,4 GHz
Disco Rígido – Instalação	6 GB
Versões do Windows	
Windows Server 2019	Datacenter, Standard, Essentials
Windows Server 2012 R2	Datacenter, Standard, Essentials, Foundation
Windows Server 2016	Datacenter, Standard, Essentials, Foundation
Windows 10	Enterprise, professional, home
Windows 8.1	Enterprise, professional, home

CENÁRIO DO INSTALAÇÃO



- ✓ Windows Server 2016 Datacenter
- ✓ SQL Server 2017 Developer
- ✓ 60 GB – HD Particionados
- ✓ 8 GB de memória(4 GB SQL server/2 GB S.O)
- ✓ 2 processadores



INSTALAÇÃO SQL SERVER

Instalação SQLSERVER 2017 Developer

Passo 1:

Baixar o instalador

Microsoft® SQL Server® 2017 Developer (Gratuita)

<https://www.microsoft.com/pt-br/sql-server/sql-server-downloads>

Passo 2:

Instalação passo a passo.

Passo 3:

Verificação dos Serviços. (evidência do SQLSERVER Online)

INSTALAÇÃO SSMS

Instalação SQL Server Management Studio

Passo 1:

Baixar o instalador

Download SQL Server Management Studio (SSMS)

<https://docs.microsoft.com/pt-pt/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-2017>

Passo 2:

Instalação passo a passo.(Português)

Passo 3:

Verificação dos Serviços. (evidência do SQLSERVER Online)

INSTALAÇÃO 2º INSTANCIA SQL SERVER

Instalação SQLSERVER 2017 Developer

Passo 1:

Usar a mídia de instalação da 1º Instância

Microsoft® SQL Server® 2017 Developer (Gratuita)

<https://www.microsoft.com/pt-br/sql-server/sql-server-downloads>

Passo 2:

Instalação passo a passo.

Passo 3:

Verificação dos Serviços. (evidência do SQLSERVER Online)

OVERVIEW SSMS

The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. The title bar indicates the connection is to `adjacencyList.sql - SQL2017CTP2-1.Northwind (SQL2017CTP2-1\Nimda (55))`. A red box highlights the **Quick Launch (Ctrl+Q)** button in the top right corner of the title bar.

The **Object Explorer** pane on the left shows the database structure for `SQL2017CTP2-1`, including `Databases`, `System Databases`, `Database Snapshots`, `DWDiagnostics (Single User)`, `graphdemo`, and the `Northwind` database. The `Tables` node under `Northwind` is expanded, showing `System Tables`, `FileTables`, `External Tables`, `Graph Tables`, `dbo.belongsTo`, `dbo.EmployeesNode`, `dbo.reportsTo`, `dbo.TerritoriesNode`, `dbo.Categories`, `dbo.CustomerCustomerDer`, `dbo.CustomerDemographic`, `dbo.Customers`, and `dbo.Employees`.

The **Query Editor** pane in the center contains the following T-SQL script:

```
SELECT FirstName, LastName, NULL AS ManagerFirstName, NULL AS ManagerLastName
FROM EmployeesNode
WHERE ReportsTo IS NULL
UNION ALL
SELECT e.FirstName, e.LastName, m.FirstName AS ManagerFirstName, m.LastName AS ManagerLastName
FROM EmployeesNode AS e, reportsTo, EmployeesNode AS m
WHERE MATCH (e-(reportsTo)->m);
```

The **Results** pane at the bottom displays the query results:

FirstName	LastName	ManagerFirstName	ManagerLastName
Andrew	Fuller	NULL	NULL
Nancy	Davolio	Andrew	Fuller
Janet	Leverling	Andrew	Fuller
Margaret	Peacock	Andrew	Fuller
Steven	Buchanan	Andrew	Fuller
Michael	Suyama	Steven	Buchanan
Robert	King	Steven	Buchanan
Laura	Callahan	Andrew	Fuller
Anne	Dodsworth	Steven	Buchanan

(9 row(s) affected)

The status bar at the bottom shows the session information: `17CTP2-1(14.0 CTP2.1) | SQL2017CTP2-1\Nimda (55) | Northwind | 00:00:00 | 9 rows`.

MATERIAL EXCLUSIVO PARA ALUNO UDEMY - DESENVOLVIDO

Ready POR ANDRÉ ROSA - IT FOREST Col 31 Ch 31 INS

SYSTEM DATABASES

Banco	Descrição
Master	Registra toda a informações de nível de sistema por uma instância do SQL Server.
MSDB	É usado pelo SQL Server Agent para programar alertas e trabalhos.
MODEL	É usado como modelo de todos os bancos de dados criados na instância do SQL Server. As modificações feitas no banco de dados modelo, como tamanho, ordenação, modelo de recuperação, e outras opções de bancos de dados, são aplicadas a qualquer banco de dados criados em seguida
TEMPDB	É um workspace para reter objetos temporários ou conjuntos de resultados intermediários.
RESOURCE	É um banco de dados do tipo somente leitura que contém objetos de sistema incluídos no SQL Server. Os objetos de sistema são fisicamente persistentes no banco de dados Recurso , mas aparecem logicamente no esquema sys de todo banco de dados.

SYSTEM DATABASES

Propriedades físicas de Resource

Os nomes dos arquivos físicos do banco de dados do Resource são **mssqlsystemresource.mdf** e **mssqlsystemresource.ldf**.

Localizado em <drive>:\Program Files\Microsoft SQL Server\MSSQL<version>.<instance_name>\MSSQL\Binn\

Não devem ser movidos.

Cada instância do SQL Server tem um e apenas um arquivo **mssqlsystemresource.mdf** associado, e as instâncias não compartilham esse arquivo.

CONFIGURE

INFORMAÇÃO

Exibe ou altera parâmetros de configuração global para o servidor atual.

sp_configure não aceita novos valores de opção de configuração fora dos intervalos válidos documentados para cada opção de configuração.

O padrão das permissões de **CONFIGURE** é concedida aos possuidores da permissão ALTER SETTINGS. As funções de servidor fixas **sysadmin** e **serveradmin** contêm esta permissão implicitamente.

<https://docs.microsoft.com/pt-br/sql/t-sql/language-elements/reconfigure-transact-sql?view=sql-server-2017>

RECONFIGURE

INFORMAÇÃO

Atualiza o valor configurado atualmente (de uma opção de configuração alterada com o procedimento armazenado do sistema **sp_configure**). Como algumas opções de configuração requerem uma parada e um reinício do servidor para atualizar o valor em execução no momento, RECONFIGURE nem sempre atualiza o valor em execução no momento

sp_configure não aceita novos valores de opção de configuração fora dos intervalos válidos documentados para cada opção de configuração.

O padrão das permissões de **RECONFIGURE** é concedida aos possuidores da permissão ALTER SETTINGS. As funções de servidor fixas **sysadmin** e **serveradmin** contêm esta permissão implicitamente.

<https://docs.microsoft.com/pt-br/sql/t-sql/language-elements/reconfigure-transact-sql?view=sql-server-2017>

Todas Configurações

INFORMAÇÃO

<https://docs.microsoft.com/pt-br/sql/database-engine/configure-windows/server-configuration-options-sql-server?view=sql-server-2017>

CONFIGURAÇÃO DE MEMÓRIA

Mínimos e Máximos

A configuração padrão de `min server memory` é 0 e a configuração padrão de `max server memory` é 2.147.483.647 MB (megabytes). Por padrão, o SQL Server pode alterar seus requisitos de memória dinamicamente com base nos recursos do sistema disponíveis.

A quantidade mínima de memória permitida para a **memória máxima do servidor** é de 128 MB.

CONFIGURAÇÃO DE MEMÓRIA

Observação

As opções memória mínima do servidor e memória máxima do servidor são opções avançadas.

Se você estiver usando o procedimento armazenado no sistema **sp_configure** para alterar essas configurações, será possível alterá-las apenas quando **show advanced options estiver definida como 1**.

Essas configurações entram em vigor imediatamente sem a reinicialização do servidor.

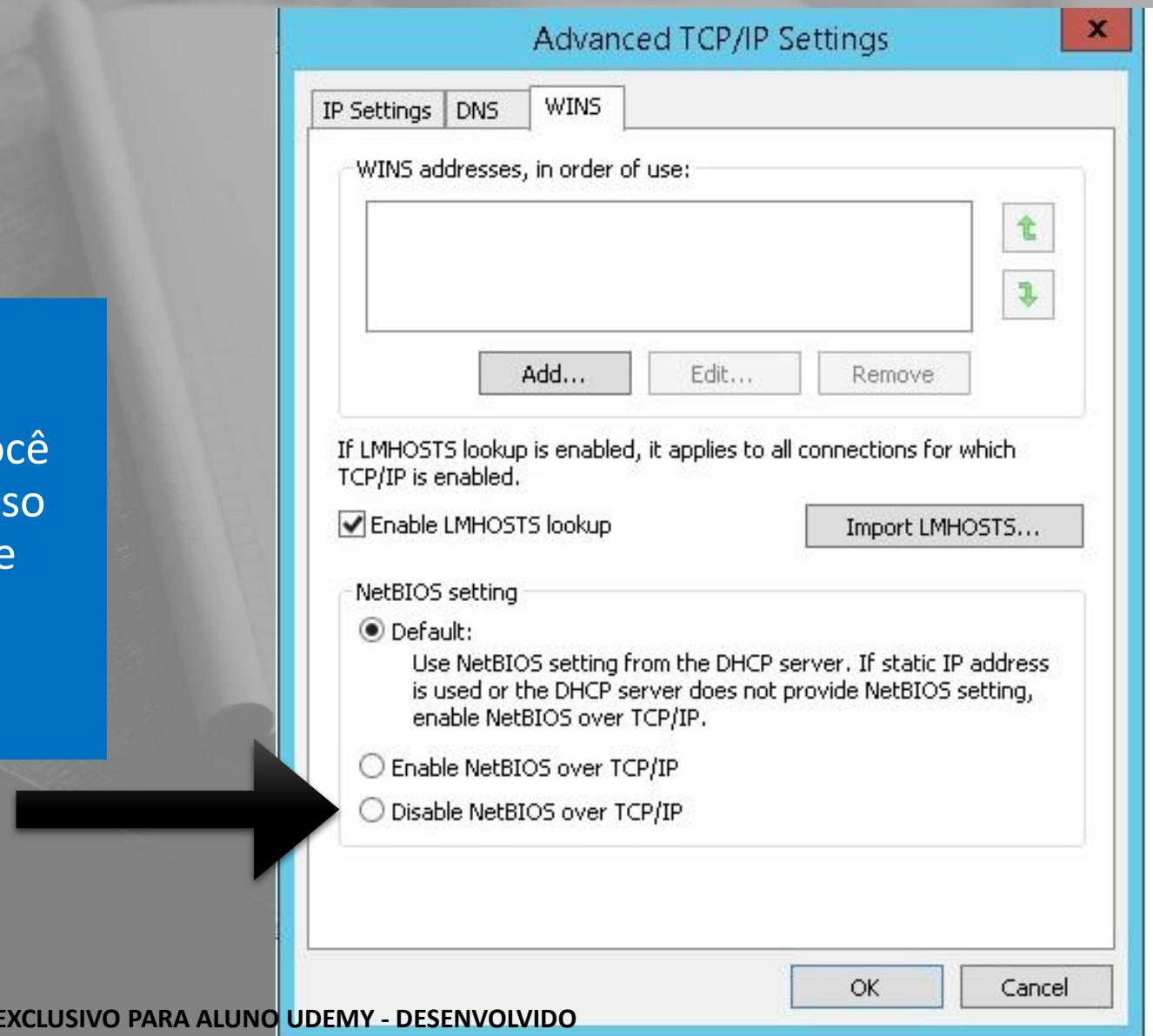
Antes de instalar o SQL Server

Práticas recomendadas

- ✓ Aprimore a segurança física
- ✓ Use firewalls
- ✓ Isole serviços
- ✓ Configure um sistema de arquivos seguro(NTFS)
- ✓ Desabilite os protocolos NetBIOS e SMB
- ✓ Instalando o SQL Server em um controlador de domínio(Por motivos de segurança, é recomendável não instalar o SQL Server em um controlador de domínio.)

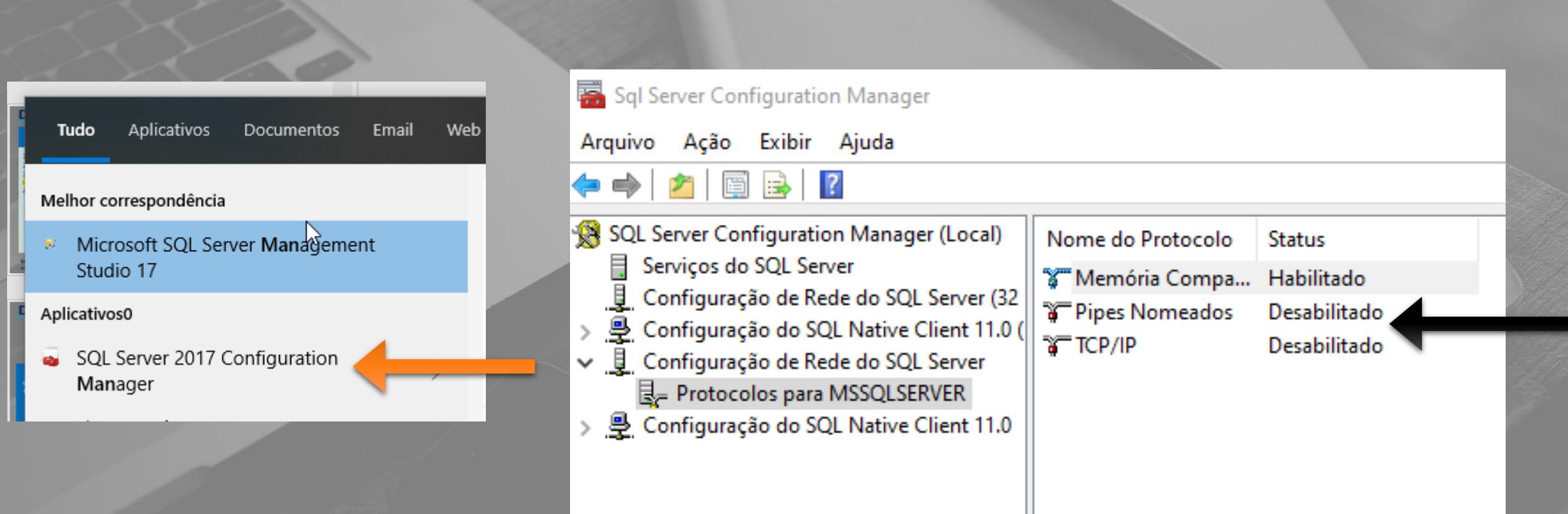
Disable NetBIOS and Server Message Block

O sistema deixa como padrão, porém você pode configurar como para Desativar caso não use ou Ativar caso use o serviço de NETBIOS.



HABILITANDO TCP/IP E NAMED PIPES

C:\Windows\SysWOW64\mmc.exe /32 C:\WINDOWS\SysWOW64\SQLServerManager14.msc



<https://docs.microsoft.com/pt-br/sql/database-engine/configure-windows/enable-or-disable-a-server-network-protocol?view=sql-server-2017>

Disable NetBIOS and Server Message Block

PROTOCOLOS

Todos os servidores na rede de perímetro devem ter os protocolos desnecessários desabilitados, incluindo NetBIOS e SMB.

O NetBIOS usa as seguintes portas:

- ✓ UDP/137 (serviço de nome do NetBIOS)
- ✓ UDP/138 (serviço de datagrama do NetBIOS)
- ✓ TCP/139 (serviço de sessão do NetBIOS)

O SMB usa as seguintes portas:

- ✓ TCP/139
- ✓ TCP/445

```
C:\Users\Administrator>netstat -a | find "445"
TCP      0.0.0.0:445          SRV-DB-01:0          LISTENING
TCP      [::]:445             SRV-DB-01:0          LISTENING
```

Disable NetBIOS and Server Message Block

Desativando SMB

- 1- Ir até Serviços
- 2- Desativar e parar serviço “Server”

Outras opções

Bloqueio de Firewall –porta 445

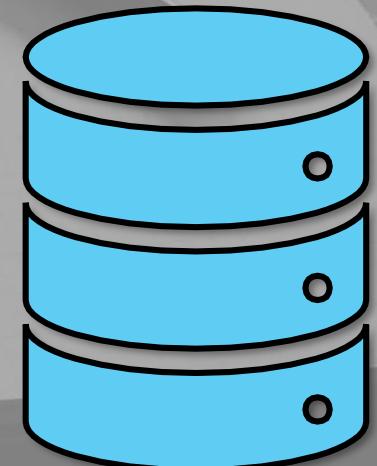
Desinstalar Recurso SMB

Arquivos do banco de dados

Primários

Secundário

Log de Transações



Arquivos do banco de dados

Arquivo	Descrição	Extensão
Primário	<p>O arquivo de dados primário contém as informações de inicialização do banco de dados e aponta para os outros arquivos no banco de dados.</p> <p>Dados do usuário e objetos podem ser armazenados neste arquivo ou em arquivos de dados secundários.</p> <p>Todo banco de dados possui um arquivo de dados primário.</p>	mdf

<https://docs.microsoft.com/pt-br/sql/relational-databases/databases/database-files-and-filegroups?view=sql-server-2017>

Arquivos do banco de dados

Arquivo	Descrição	Extensão
Secundário	<p>Os arquivos de dados secundários são opcionais, definidos pelo usuário, e armazenam dados do usuário.</p> <p>Arquivos secundários podem ser usados para distribuir os dados entre os diversos discos, colocando cada arquivo em uma unidade de disco diferente.</p> <p>Além disso, caso um banco de dados exceda o tamanho máximo em um único arquivo Windows, será possível usar arquivos de dados secundários, assim, o banco de dados continuará a crescer.</p>	ndf

Arquivos do banco de dados

Arquivo	Descrição	Extensão
Log de Transações	<p>Os arquivos de log de transações armazenam as informações de log usadas para recuperar o banco de dados.</p> <p>Deve haver, no mínimo, um arquivo de log para cada banco de dados.</p>	Ldf

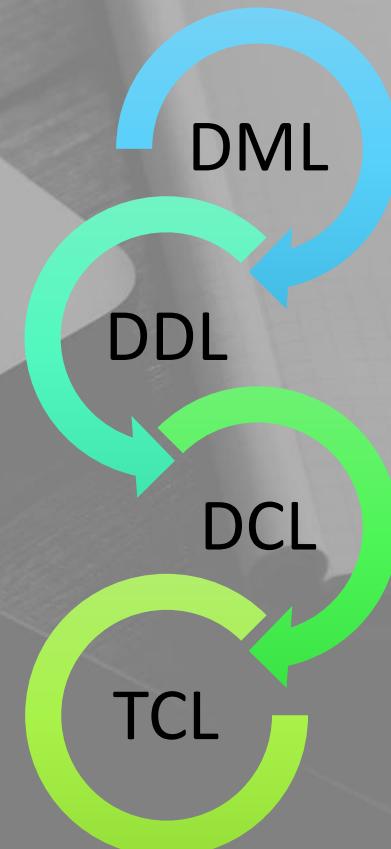
<https://docs.microsoft.com/pt-br/sql/relational-databases/databases/database-files-and-filegroups?view=sql-server-2017>

DEFINIÇÕES DA LINGUAGEM

A linguagem *SQL* é dividida em quatro tipos de instruções de linguagem primárias: **DML**, **DDL**, **DCL** e **TCL**.

Usando estas declarações, podemos definir a estrutura de um banco de dados através da *criação* e *alteração* de objetos de banco de dados, e podemos *manipular* dados em uma tabela através de *atualizações* ou *eliminações*.

Nós também podemos controlar qual usuário pode *ler / escrever* dados ou gerencia operações



- 1. **DML (Data Manipulation Language)**
- 2. **DDL (Data Definition Language)**
- 3. **DCL (Data Control Language)**
- 4. **TCL (Transaction Control Language)**

DEFINIÇÕES DA LINGUAGEM

LANGUAGE STATEMENTS

DML

- SELECT
- INSERT
- UPDATE
- DELETE

DDL

- CREATE
- ALTER
- DROP
- TRUNCATE

DCL

- GRANT
- REVOKE
- DENY

TCL

- BEGIN TRANSACTION
- COMMIT
- SAVE TRANSACTION
- ROLLBACK

DEFINIÇÕES DA LINGUAGEM

DDL

- 2. DDL (Data Definition Language)

São usadas para definir a estrutura de banco de dados ou esquema. Alguns exemplos:

DDL

- ✓ **CREATE** - para criar objetos no banco de dados, o próprio banco de dados, tabelas, indexes, procedures, views, functions e triggers.
- ✓ **ALTER** – altera a estrutura da base de dados, o próprio banco de dados, tabelas, indexes, procedures, views, functions e triggers.
- ✓ **DROP** – apaga objeto do banco de dados, o próprio banco de dados, tabelas, indexes, procedures, views, functions e triggers.
- ✓ **TRUNCATE** – remover todos os registros de uma tabela, incluindo todos os espaços alocados para os registros são removidos.

CONSTRAINTS

Constraint são Utilizadas para especificar regras de armazenamentos de dados nas tabelas e garantir integridade.

Tipo Constraint	Descrição
NOT NULL	Garante que uma coluna não receberá valor NULL
UNIQUE	Garante que os valores em uma coluna sejam diferentes.
PRIMARY KEY	Chave única, linha exclusiva com, combinação com
FOREIGN KEY	Referencia o valor de um campo em determinada linha a outra tabela
DEFAULT	Define um valor padrão para uma coluna quando nenhum valor é especificado
INDEX	Usado para criar e recuperar dados do banco de dados com melhor performance

DBA – SEGURANÇA-SECURITY

Visão geral da segurança.

Tipos de segurança disponíveis.

- ✓ Servidor/Instância 
- ✓ Banco de dados 
- ✓ Objetos 



DBA – SEGURANÇA-SECURITY

A segurança no SQL Server é baseada nos seguintes conceitos:

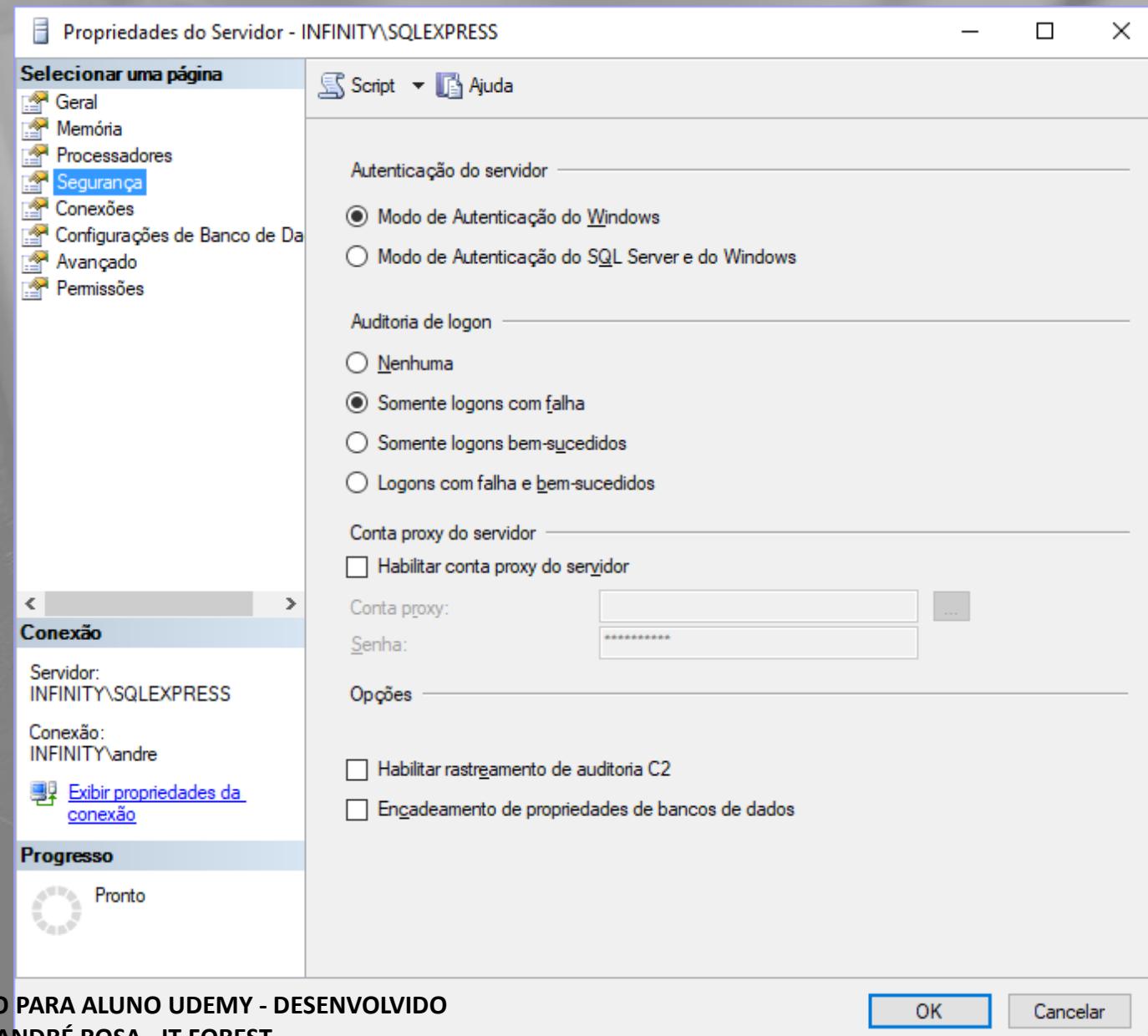
- ✓ Logins.
- ✓ User Accounts
- ✓ Schemas.
- ✓ Roles. (Server e Database)
- ✓ Permissões (DCL)



Segurança a Nível de Servidor

Os modos de autenticação possíveis são os seguintes:

- ✓ Windows Authentication mode.
- ✓ SQL Server and Windows Authentication mode.



DBA – SEGURANÇA-SECURITY

Permissões para Acessar os Objetos do Banco de Dados

Fazer o Logon no SQL Server não garante acesso a um ou mais Banco de Dados.

Você precisa ter **permissão de acesso a(os) Banco(s) de Dados** e, além do mais, você precisa ter **permissão de acesso aos objetos do Banco de Dados**.

O usuário precisa passar por **2** níveis de segurança:

- ✓ Permissão para fazer a conexão com o SERVIDOR\INSTÂNCIA.
- ✓ Permissão para acessar um ou mais Bancos de Dados.

USUÁRIO X LOGIN

Logins e usuários **são duas coisas diferentes** no SQL Server.

- ✓ Um **login** concede ao principal acesso ao servidor.
- ✓ Um **usuário** concede um acesso de login a um banco de dados específico.

Portanto, o **login** tem permissão para se conectar ao SQL Server. No entanto, nesse estágio, nenhum dos bancos de dados no servidor tem um **usuário** associado a esse **login**.

USUÁRIO X LOGIN

1.

- Criar login

2.

- Criar usuário (DB)

3.

- Associar Login X User

4

- Especificar acesso

ROLES - Segurança a Nível de Servidor

Permissões associadas com as principais Server Roles.

ROLES	Permissões para os membros dessa Role
SYSADMIN	Poderes totais sobre todos os objetos do servidor.
SECURITYADMIN	Pode gerenciar logins do servidor.
SERVERADMIN	Pode configurar a maioria das opções do servidor.
DISKADMIN	Gerenciar os arquivos de um banco de dados.
DBCREATATOR	Criar e alterar Banco de Dados.
PROCESSADMIN	Gerenciar processos rodando no SQL Server. (KILL)
SETUPADMIN	Pode gerenciar e configurar a replicação entre servidores(Linked Server)
BULKADMIN	Essa permissão permite a importação de arquivos externos para o SQL Server (txt, excel e etc).
PUBLIC	Todo login no SQL pertence a role public. É uma role default que permite que o login criado se connect nas databases, dentre outras coisas, contudo, o login não consegue fazer nada na base de dados se não for dado acessos específicos.

SERVER LEVEL ROLES AND PERMISSIONS: 9 fixed server roles, 34 server permissions

sysadmin fixed server role

CONTROL SERVER: Has all permissions in the server

bulkadmin fixed server role

ADMINISTER BULK OPERATIONS

dbcreator fixed server role

ALTER ANY DATABASE
CREATE ANY DATABASE

setupadmin fixed server role

ALTER ANY LINKED SERVER

securityadmin fixed server role

ALTER ANY LOGIN

Important: The securityadmin role should be treated as equivalent to the sysadmin role.

diskadmin fixed server role

ALTER RESOURCES

serveradmin fixed server role

ALTER SETTINGS

SHUTDOWN

ALTER ANY ENDPOINT

CREATE ENDPOINT

processadmin fixed server role

ALTER ANY CONNECTION

VIEW ANY DEFINITION

public fixed server role

The other server level permissions, are not granted to any fixed server role except sysadmin.

There are no server-level permissions inherent in the public server role, however some server permissions are present by default. Specifically, VIEW ANY DATABASE, and CONNECT permission to the endpoints. These permissions can be revoked.

Segurança a Nível de Database

Permissões associadas com as principais Fixed Databases Roles.

ROLES	Permissões para os membros dessa Role
db_owner	Tem poderes totais sobre o banco de dados.
db_accessadmin	Pode adicionar e remover usuários ao Banco de Dados.
db_datareader	Pode ler dados em todas as tabelas de usuário do BD.
db_datawriter	Pode adicionar, alterar ou excluir dados em todas as tabelas de usuário do BD.
db_ddladmin	Pode adicionar, modificar ou excluir objetos do BD.
db_securityadmin	Pode gerenciar roles e adicionar ou excluir usuários às roles do BD. Pode gerenciar as permissões para objetos do BD.
db_backupoperator	Pode fazer o backup do BD.
db_denydatareader	Não pode consultar dados em nenhuma das tabelas do BD, mas pode efetuar alterações na estrutura do BD e de seus objetos.
db_denydatawriter	Não pode alterar dados no Banco de Dados.

Segurança a Nível de Database

DATABASE LEVEL ROLES AND PERMISSIONS: 11 fixed database roles, 77 database permissions

db_owner fixed database role

CONTROL DATABASE: Has all permissions in the database

db_datareader

GRANT SELECT ON DATABASE::<name>

db_denydatareader

DENY SELECT ON DATABASE::<name>

db_datawriter

GRANT INSERT ON DATABASE::<name>

db_denydatawriter

DENY INSERT ON DATABASE::<name>

GRANT UPDATE ON DATABASE::<name>

DENY UPDATE ON DATABASE::<name>

GRANT DELETE ON DATABASE::<name>

DENY DELETE ON DATABASE::<name>

db_accessadmin

CREATE SCHEMA

ALTER ANY USER

CONNECT

db_securityadmin

ALTER ANY ROLE, CREATE ROLE

ALTER ANY APPLICATION ROLE

VIEW DEFINITION

CREATE SCHEMA

ALTER ANY ROLE, CREATE ROLE

ALTER ANY APPLICATION ROLE

VIEW DEFINITION

db_backupoperator

BACKUP DATABASE

BACKUP LOG

CHECKPOINT

db_ddladmin

ALTER ANY ASSEMBLY
ALTER ANY ASYMMETRIC KEY
ALTER ANY CERTIFICATE
ALTER ANY CONTRACT
ALTER ANY DATABASE DDL TRIGGER
ALTER ANY DATABASE EVENT NOTIFICATION
ALTER ANY DATASPACE
ALTER ANY FULLTEXT CATALOG
ALTER ANY MESSAGE TYPE
ALTER ANY REMOTE SERVICE BINDING
ALTER ANY ROUTE
ALTER ANY SCHEMA
ALTER ANY SERVICE
ALTER ANY SYMMETRIC KEY
CHECKPOINT
CREATE AGGREGATE
CREATE DEFAULT
CREATE FUNCTION
CREATE PROCEDURE
CREATE QUEUE
CREATE RULE
CREATE SYNONYM
CREATE TABLE
CREATE TYPE
CREATE VIEW
CREATE XML SCHEMA COLLECTION
REFERENCES

public

There are no database-level permissions inherent in the public database role, however some database permissions are present by default. Specifically, VIEW ANY COLUMN MASTER KEY DEFINITION, VIEW ANY COLUMN ENCRYPTION KEY DEFINITION, and SELECT permission on many individual system tables. These permissions can be revoked.

There are various special purpose roles in the msdb database

MATERIAL EXCLUSIVO PARA ALUNO UDEMY - DESENVOLVIDO
to any fixed database role except db_owner.
POR ANDRÉ ROSA - IT FOREST

SQL Server 2017

Segurança a Nível de Objetos

DCL

- 3. DCL (Data Control Language)

São usadas para definir acesso/controle dos dados/objetos: Alguns exemplos:

DCL

- ✓ **GRANT** – atribui privilégios de acesso do usuário a objetos do banco de dados.
- ✓ **REVOKE** – remove os privilégios de acesso aos objetos obtidos com o comando GRANT.
- ✓ **DENY** - nega permissão a um usuário ou grupo para realizar operação em um objeto ou recurso.

SCHEMAS

Schema - Principal

Principal: um principal é considerado qualquer objeto que possa solicitar acesso a recursos do SQL Server. por exemplo, usuários e grupos do windows, logins e roles do SQL Server, e applications roles.

Schema: um schema é um conjunto de objetos, sendo que todos os objetos pertencentes a um schema têm como dono o mesmo principal.
No SQL Server todos os objetos de um Banco de Dados têm como dono um schema.

Servidor.Banco_de_Dados.Schema.Objeto

SRV-DB-01\MSSQLSERVER001.MINICRM.**dbo**.CLIENTE

SCHEMAS

- ✓ Um schema é um container para objetos.
- ✓ Todo objeto pertence a um schema.
- ✓ Todos os objetos de um schema têm o mesmo **dono(owner)**, que é o dono do schema.
- ✓ As permissões podem ser atribuídas para o schema e também para os objetos dentro do schema.

TRABALHANDO COM SCHEMAS

Object Explorer

Connect ▾

- Databases
 - System Databases
 - Database Snapshots
 - AdventureWorks**
 - Database Diagrams
- Tables
 - System Tables
 - dbo.AWBuildVersion**
 - dbo.DatabaseLog**
 - dbo.ErrorLog**
 - HumanResources.Department**
 - HumanResources.Employee**
 - HumanResources.EmployeeAddress**
 - HumanResources.EmployeeDepartmentHistory**
 - HumanResources.EmployeePayHistory**
 - HumanResources.JobCandidate**
 - HumanResources.Shift**
 - Person.Address**
 - Person.AddressType**
 - Person.Contact**
 - Person.ContactType**
 - Person.CountryRegion**
 - Person.StateProvince**
 - Production.BillOfMaterials**
 - Production.Culture**



MODOS DE MANUTENÇÃO

Modos de Manutenção

São estados que banco de dados, normalmente utilizados para realizar algum tipo de **manutenção** no banco de dados. Sempre é interessante deixar o banco de dados **indisponível** para o usuário, **quando for criar, tabelas, indexes ou realizar algum outro procedimento que ação do usuário possa interferir no resultado.**



- ✓ OFFLINE
- ✓ ONLINE
- ✓ SINGLE_USER
- ✓ READ_ONLY
- ✓ MULT_USER
- ✓ READ_WRITE

- ✓ SUSPECT
- ✓ EMERGENCY



MODOS DE MANUTENÇÃO

- ✓ **ONLINE**: O banco de dados está disponível para acesso. Aqui podemos acessar e gravar informações;
- ✓ **OFFLINE**: O banco de dados está indisponível. O serviço do SQL Server não coloca o banco de dados nesse estado automaticamente. É um estado que somente o administrador do banco de dados o coloca via comando ALTER DATABASE [database] OFFLINE;
- ✓ **RESTORING**: O banco de dados está indisponível. Um ou mais arquivos do grupo de arquivos que compõe o banco de dados está sendo restaurado por uma ação de restore;

MODOS DE MANUTENÇÃO

- ✓ **SINGLE_USER**: Este modo especifica que apenas **um usuário** pode acessar o banco de dados por vez e, normalmente é usado para ações de manutenção;
- ✓ **READ_ONLY**: Os usuários podem **ler dados** do banco de dados, mas não os modificar.;
- ✓ **READ_WRITE**: O banco de dados está disponível para operações de **leitura e gravação**;
- ✓ **MULT_USER**: Todos os usuários com permissões apropriadas para se conectar ao banco de dados são permitidos.;

MODOS DE MANUTENÇÃO

- ✓ **RECOVERING:** O banco de dados está sendo recuperado. Ele ficará disponível assim que a recuperação estiver completa. Caso não consiga recuperar (colocar o banco de dados em ONLINE), o mesmo entra em estado SUSPECT e fica inacessível. O estado “RECOVERING” acontece toda vez que o serviço do SQL Server é iniciado para testar se todos os objetos e seus datafiles estão íntegros;
- ✓ **RECOVERY_PENDING:** O banco de dados está com algum erro na recuperação e uma ação adicional é exigida do usuário para resolver o erro e permitir que o processo de recuperação seja concluído. O banco fica inacessível enquanto esse erro não for sanado;

MODOS DE MANUTENÇÃO

SUSPECT: Alguns dos arquivos do grupo do banco de dados estão corrompidos. Na inicialização do SQL Server, este verifica se os arquivos estão íntegros. Se ele encontra algum problema nesses arquivos, ele coloca o banco de dados no estado “**SUSPECT**”. Esse estado necessita da ação direta do Administrador de Banco de Dados e é o estado que trabalharemos nesse artigo;

EMERGENCY: Nesse estado o banco fica em modo de usuário único e marcado como **READ_ONLY**, além do arquivo de log de gravação do banco ficar desabilitado. Nesse estado é que trabalharemos para solucionar o estado do banco de dados **SUSPECT** e para tornar ele **ONLINE**.

BACKUP E RESTORE

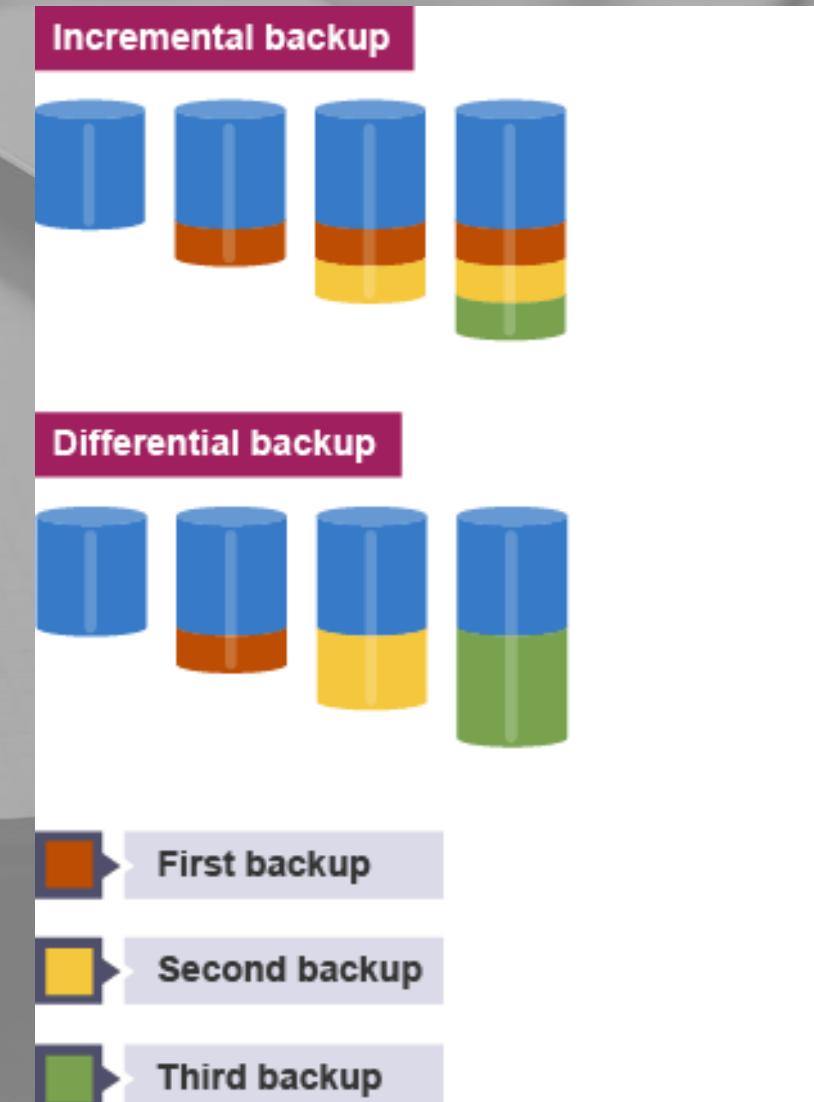
- ✓ Dominar tarefas de backup e restore é imprescindível para administrador de banco de dados
- ✓ Deve se definir estratégias/políticas para o mesmos
- ✓ Teste de restauração/validação.



BACKUP E RESTORE

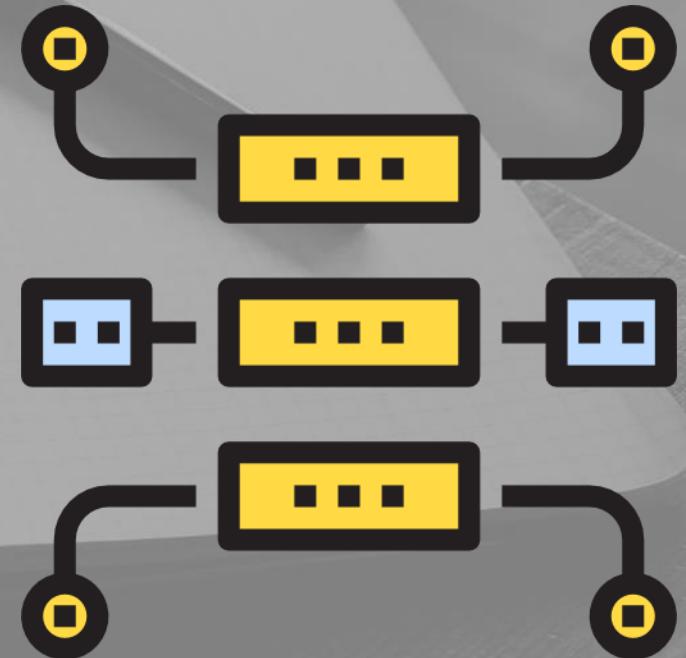
Tipos de Backup SQLSERVER

- **FULL** – Completo
- **INCREMENTAL** – Vários arquivos de backup, com alterações apenas feitas após ultimo backup.
- **DIFERENCIAL** -é um backup cumulativo de todas as alterações feitas desde o último backup completo



BACPAC

- Um aplicativo da camada de dados (DAC) é uma entidade de gerenciamento de banco de dados lógico que define todos os objetos SQL Server.
- Um BACPAC é um artefato relacionado que encapsula o esquema de banco de dados e também os dados armazenados no banco de dados.



COMO SABER VERSÃO E EDIÇÃO EM USO

Conectar-se à instância do SQL Server e, em seguida, execute a seguinte consulta:

Select @@version

Ou

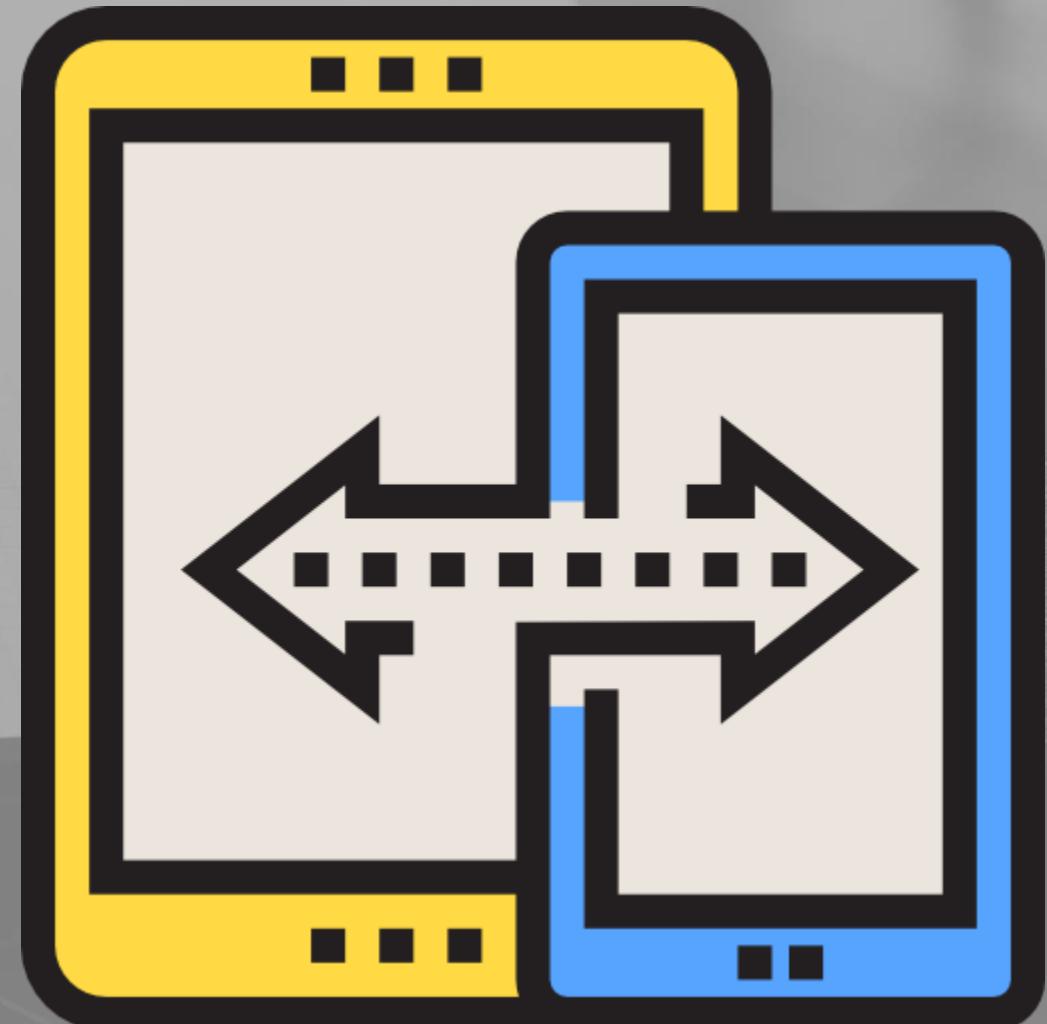
```
SELECT SERVERPROPERTY('productversion'),  
       SERVERPROPERTY ('productlevel'),  
       SERVERPROPERTY ('edition')
```

DBA – ATTACH/DETACH (Anexa/Desanexa)

ATTACH/DETACH

através do recurso de administração do SQLSERVER, é possível **anexar ou desanexar** através dos modos :

- ✓ Gráfico
- ✓ Linha de comando.



SHRINK

Comandos/tarefas utilizadas para redução do tamanho de banco de dados ou arquivos.

É possível reduzir cada arquivo dentro de um banco de dados para remover páginas sem utilização.

Embora o Mecanismo de Banco de Dados reutilize efetivamente os espaços, há momentos em que um arquivo não necessita mais ser tão grande quanto era; pode ser necessário reduzir o arquivo. É possível reduzir dados e arquivos de log de transações, estas ações podem ser automáticas ou manuais.

--Redução automática do banco de dados

--Quando a opção **AUTO_SHRINK** do banco de dados estiver definida como ON,

ALTER DATABASE AULA_BK SET AUTO_SHRINK OFF --(Desliga)

ALTER DATABASE AULA_BK SET AUTO_SHRINK ON --(Liga)

FRAGMENTAÇÃO DE INDICES

Reorganizando e recompilando índices

O Mecanismo de Banco de Dados do SQL Server modifica os índices automaticamente sempre que são realizadas operações **de entrada, atualização ou exclusão nos dados subjacentes**. No decorrer do tempo, essas modificações podem fazer com que as informações do índice sejam dispersadas pelo banco de dados (**fragmentadas**).

--Detectando a fragmentação em índices de rowstore

Usando a função de sistema `sys.dm_db_index_physical_stats`

Valor avg_fragmentation_in_percent	Instrução corretiva
> 5% e < = 30%	<code>ALTER INDEX REORGANIZE</code>
> 30%	<code>ALTER INDEX REBUILD WITH (ONLINE = ON)</code> ¹

Lock, Block e Deadlock

LOCK

- Ocorre quando uma sessão (Ex: sessão 77) está realizando alguma alteração (de dados ou de estrutura) em algum objeto. Para garantir o **ACID** o SQL **trava** este objeto para outras sessões não modificarem este objeto.

BLOCK

- Semelhante ao Lock, ocorre no cenário que existe um lock no objeto que 1 ou mais sessões estão tentando alterar o objeto, estas sessões, ficam em **espera(wait)** em fila.

DEADLOCK

- O cenário de Deadlock é semelhante ao cenário do Block, atentando-se que no **Deadlock**, a sessão **blockeadora** também está sendo **blockada**.

DEADLOCK



MATERIAL EXCLUSIVO PARA ALUNO UDEMY - DESENVOLVIDO
POR ANDRÉ ROSA - IT FOREST

O que são statistics?

Estatísticas (statistics) são objetos que detêm informações importantes sobre a distribuição dos dados dentro de tabelas e views indexadas. As estatísticas são de extrema importância para o SQL Server, uma vez que o Otimizador de Consulta (Query Optimizer) utiliza as estatísticas para analisar a seletividade e cardinalidade dos dados, a fim de criar um plano de alta qualidade traçando a melhor “rota” para execução da query.

Existem três modos que uma estatística pode ser criada:
automática, explícita e implicitamente

Por default **AUTO_CREATE_STATISTICS** é **ON**. Isso permite que, caso necessário, o SQL Server crie estatísticas em operações de manipulação de dados.

Ao executar o Select, teoricamente forçará o SQL Server a criar um **plano de execução** e como não existe nenhuma estatística para a tabela, internamente e automaticamente será criada para o predicado.

Estatísticas criadas implicitamente

É quando criamos um índice – seja ele clustered ou não – e por consequência são criadas estatísticas implicitamente para os campos chaves do índice.

```
ALTER TABLE alunos  
ADD CONSTRAINT pkAluno PRIMARY KEY CLUSTERED (Id)
```

Estatísticas criadas manualmente

Também podemos criá-las manualmente, utilizando o comando
CREATE STATISTICS

```
CREATE STATISTICS St_id_nome  
ON alunos (aluno_id,aluno_nome);
```

DBA – DBCC (DataBase Console Commands)

DBCC

são comandos que fazem a consistência lógica e física do banco de dados.

Apesar da maioria dos comandos ser somente para retornar informações sobre o banco temos alguns comandos podem ajudar a resolver problemas.

CATEGORIA	DESCRIÇÃO
Manutenção	Tarefas de manutenção em um banco de dados, índice ou grupo de arquivos.
Miscellaneous	Tarefas diversas, como habilitar sinalizadores de rastreamento ou remover uma DLL da memória.
Informações	Tarefas que reúnem e exibem vários tipos de informações.
Validação	Operações de validação em um banco de dados, tabela, índice, catálogo, grupo de arquivos ou alocação de páginas de banco de dados.

DBA – DBCC (DataBase Console Commands)

DBCC - Manutenção

Comando	Descrição
DBCC TABLE CHECK	A consistência lógica e física dos objetos no banco de dados é verificada nessa fase.
DBCC TABLE REPAIR	As correções de banco de dados são executadas nessa fase, se REPAIR_FAST, REPAIR_REBUILD ou REPAIR_ALLOW_DATA_LOSS for especificado e houver erros de objeto que precisem ser corrigidos.
DBCC SYS CHECK	As tabelas do sistema de banco de dados são verificadas nessa fase.
DBCC SYS REPAIR	As correções de banco de dados são realizadas nessa fase se REPAIR_FAST, REPAIR_REBUILD ou REPAIR_ALLOW_DATA_LOSS for especificado e houver erros de tabelas do sistema que precisem ser corrigidos.
DBCC FREEPROCCACHE	Limpa cache de procedures

DBA – DBCC (DataBase Console Commands)

DBCC – Diversas Miscellaneous

Comando	Descrição
DBCC HELP	Retorna informações de sintaxe para o comando especificado DBCC.
DBCC dllname (FREE)	Descarrega o procedimento armazenado estendido DLL especificado da memória
DBCC TRACEON	Habilita os sinalizadores de rastreamento especificados
DBCC TRACEOFF	Desabilita os sinalizadores de rastreamento especificados.

DBA – DBCC (DataBase Console Commands)

DBCC – Validação

Comando	Descrição
DBCC CHECKDB	Verifica a integridade lógica e física de todos os objetos do banco de dados especificado com a execução das seguintes operações:
DBCC CHECKTABLE	Verifica a integridade de todas as páginas e estruturas que compõem a tabela ou a exibição indexada
DBCC CHECKCONSTRAINTS	Inspeciona a integridade de uma restrição especificada ou de todas as restrições em uma tabela especificada no banco de dados atual.
DBCC CHECKALLOC	Verifica a consistência de estruturas de alocação de espaço em disco para um banco de dados especificado

DBA – DBCC (DataBase Console Commands)

DBCC – Informações

Comando	Descrição
DBCC SHOWCONTIG	Exibe informações de fragmentação para os dados e índices da tabela ou exibição especificada
DBCC SHOW_STATISTICS	Exibe as estatísticas de otimização de consulta atuais de uma tabela ou exibição indexada
DBCC PROCCACHE	Exibe informações em um formato de tabela sobre o cache de procedimento
DBCC OPENTRAN	Ajuda a identificar as transações ativas que podem impedir o truncamento do log

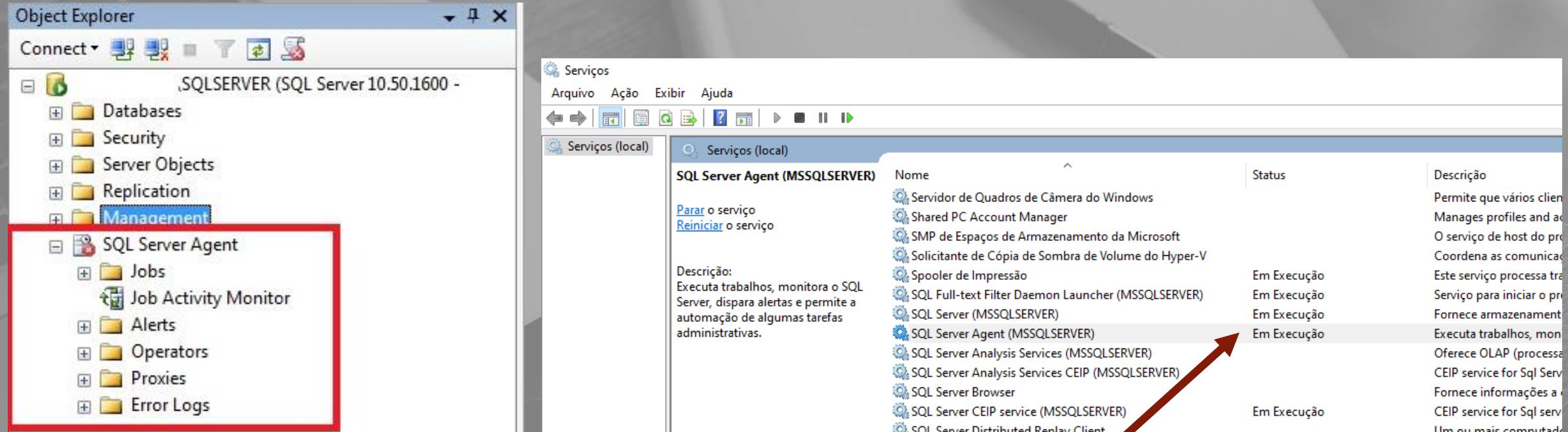
DBA – SQL AGENT

SQL Server Agent é um serviço do SQLServer que executa tarefas/Jobs agendadas.
EX: Backups, Rotinas Administrativas do DB.

1. Estabeleça quais tarefas administrativas ou eventos de servidor ocorrem regularmente e se essas tarefas ou eventos podem ser administrados via programação. Uma tarefa é uma boa candidata para automação sempre que envolve uma sequência previsível de etapas e ocorre em um horário específico ou em resposta a um evento específico.
2. Defina um conjunto de trabalhos, agendas, alertas e operadores usando o SQL Server Management Studio, o Transact-SQL scripts ou o SMO (SQL Server Management Objects).
3. Execute os trabalhos do SQL Server Agent que você definiu.

DBA – SQL AGENT

SQL Server Agent é um serviço do SQLServer que executa tarefas/Jobs agendadas.
EX: Backups, Rotinas Administrativas do DB



DBA – DMV (Dynamic Management Views)

As exibições e funções de gerenciamento dinâmico **retornam informações do estado do servidor** que podem ser usadas para **monitorar** a saúde da instância do servidor, **diagnosticar** problemas e **ajustar** o desempenho.

DBA – DMV (Dynamic Management Views)

Por que usar DMV?

- ✓ **Informações Relevantes para “N” Situações**
- ✓ **Otimiza seu tempo desenvolvendo query complexas para realizar diagnósticos.**

DMVs estão categorizadas
em grupos específicos
baseados nas áreas.

Categoria	Prefixo
AlwaysOn Availability Group	sys.dm_hadr_*
Change Data Capture Related	sys.dm_cdc_*
Change Tracking Related	sys.dm_tran_*
Common Language Runtime Related	sys.dm_clr_*
Database Mirroring Related	sys.dm_db_mirroring_*
Database Related Dynamic	sys.dm_db_*
Execution Related Dynamic	sys.dm_exec_*
Extended Events	sys.dm_xe_*
Filestream and FileTable (Transact-SQL)	sys.dm_filestream_*
Full-Text Search and Semantic Search	sys.dm_fts_*
I/O Related	sys.dm_io_*
Index Related	sys.dm_db_index_*
Object Related	sys.dm_sql_*
Query Notifications Related	sys.dm_qn_*
Replication Related	sys.dm_repl_*
Resource Governor	sys.dm_resource_governor_*
Security Related	sys.dm_audit_*/ sys.dm_cryptographic_*
Service Broker Related	sys.dm_broker_*
SQL Server Operating System Related	sys.dm_os_*
Transaction Related	sys.dm_tran_*

DBA – DMV (Dynamic Management Views)

Todas Categorias

Grupos de disponibilidade Always On, exibições de gerenciamento dinâmico e funções (Transact-SQL)	Exibições de gerenciamento dinâmico de tabela otimizada em memória (Transact-SQL)
Exibições de gerenciamento dinâmico relacionadas à captura de dados de alterações (Transact-SQL)	Funções e exibições de gerenciamento dinâmico relacionadas ao objeto (Transact-SQL)
Controle de alterações relacionadas a exibições de gerenciamento dinâmico	Exibições de gerenciamento dinâmico relacionadas às notificações de consulta (Transact-SQL)
Common Language Runtime relacionados exibições de gerenciamento dinâmico (Transact-SQL)	Exibições de gerenciamento dinâmico relacionadas à replicação (Transact-SQL)
Exibições de gerenciamento dinâmico relacionadas ao espelhamento de banco de dados (Transact-SQL)	Administrador de recursos relacionados a exibições de gerenciamento dinâmico (Transact-SQL)
Banco de dados relacionados a exibições de gerenciamento dinâmico (Transact-SQL)	Funções e exibições de gerenciamento dinâmico relacionadas à segurança (Transact-SQL)
Funções e exibições de gerenciamento dinâmico relacionadas à execução (Transact-SQL)	Exibições e funções de gerenciamento dinâmico relacionadas ao servidor (Transact-SQL)
Exibições de gerenciamento dinâmico de eventos estendidos	Exibições de gerenciamento dinâmico relacionadas ao Service Broker (Transact-SQL)
FileStream e exibições de gerenciamento dinâmico de FileTable (Transact-SQL)	Exibições de gerenciamento dinâmico e funções relacionam a dados espaciais (Transact-SQL)
Pesquisa de texto completo e funções e exibições de gerenciamento dinâmico de pesquisa semântica (Transact-SQL)	SQL Data Warehouse e Parallel Data Warehouse exibições de gerenciamento dinâmico (Transact-SQL)
Funções e exibições de gerenciamento dinâmico de replicação geográfica (banco de dados SQL do Azure)	Sistema operacional SQL Server relacionados exibições de gerenciamento dinâmico (Transact-SQL)
Índice de funções e exibições de gerenciamento dinâmico relacionadas ao (Transact-SQL)	Stretch Database exibições de gerenciamento dinâmico (Transact-SQL)
Eu O relacionadas a funções e exibições de gerenciamento dinâmico (Transact-SQL)	Funções e exibições de gerenciamento dinâmico relacionadas à transação (Transact-SQL)

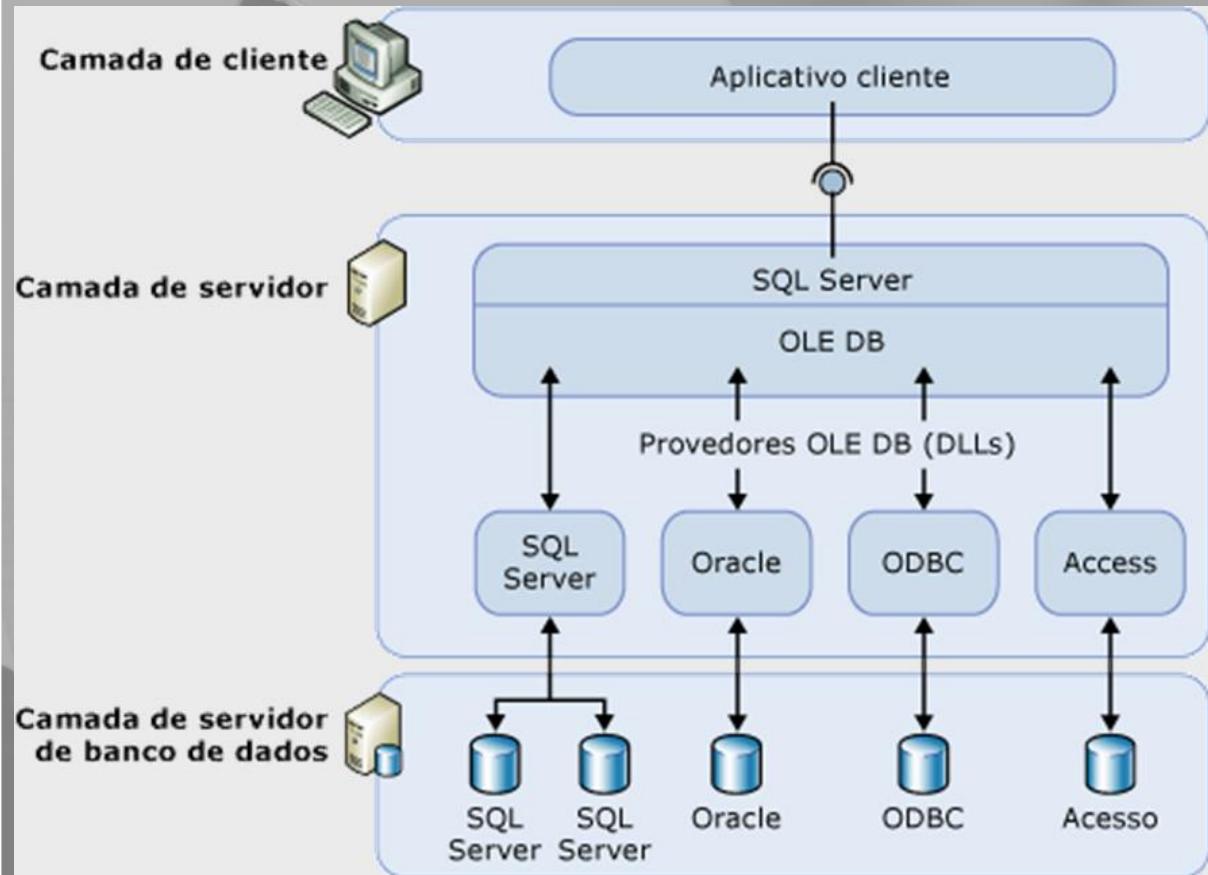
DBA – LINKED SERVER

LINKED SERVER

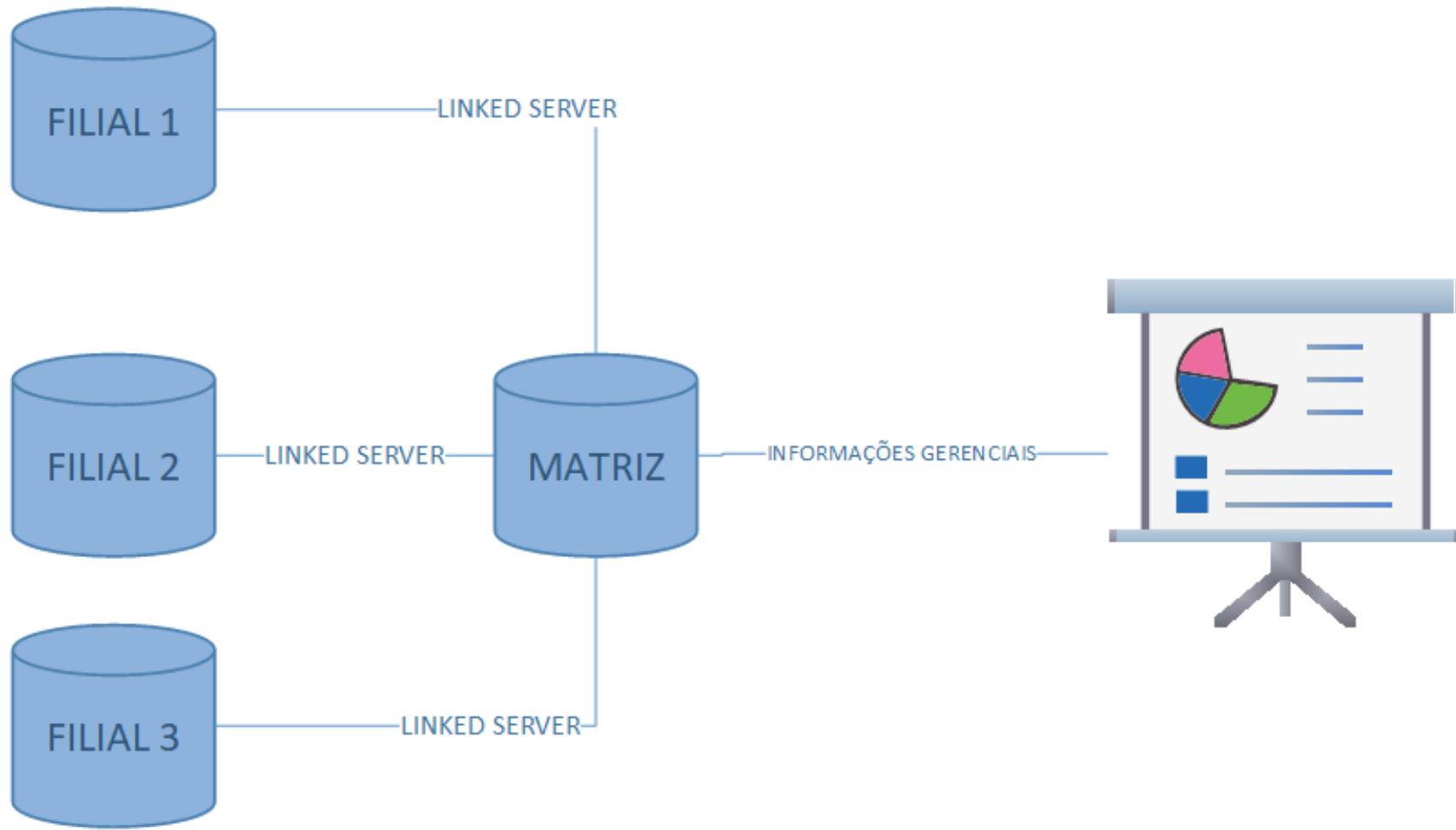
O Linked Server é uma funcionalidade do SQL Server muito útil para estabelecer conexão segura entre dois ou mais servidores.

Administradores de banco de dados e desenvolvedores utilizam muito esse recurso para realizarem consultas em outros servidores.

Essa “ponte” de comunicação é possível entre fontes OLE DB, como Excel, Access, servidores SQL Server e até servidores com outro SGBDs como Oracle, Mysql, Sybase. As principais vantagens de se utilizar Linked Server é a facilidade de executar consultas distribuídas, comandos de atualizações que envolvem diversas bases em servidores diferentes.



POSSIVEL CENÁRIO LINKED SERVER



DBA – LINKED SERVER

LINKED SERVER-PONTOS DE ATENÇÃO.

- ✓ Configuration Manager Pipes Nomeado e TCP/IP -> Habilitados
- ✓ Configurações Firewall -> Portas 1433, 1434 -> Liberadas, Regras.
- ✓ Conexão Remota na Instância -> Permitir
- ✓ Conexão entre as redes -> Funcionando

DBA – UTILITÁRIO SQLCMD

SQLCMD

O sqlcmd utilitário permite que você insira instruções Transact-SQL, procedimentos do sistema e arquivos de script no prompt de comando

PARÂMETROS

- a packet_size
- A (dedicated administrator connection)
- b (terminate batch job if there is an error)
- c batch_terminator
- C (trust the server certificate)
- d db_name
- e (echo input)
- E (use trusted connection)
- f codepage | i:codepage[,o:codepage] | o:codepage[,i:codepage]
- g (enable column encryption)

DBA – UTILITÁRIO SQLCMD

SQLCMD

O sqlcmd utilitário permite que você insira instruções Transact-SQL, procedimentos do sistema e arquivos de script no prompt de comando

PARÂMETROS

- G (use Azure Active Directory for authentication)**
- h rows_per_header**
- H workstation_name**
- i input_file**
- I (enable quoted identifiers)**
- j (Print raw error messages)**
- k[1 | 2] (remove or replace control characters)**
- K application_intent**
- l login_timeout**
- L[c] (list servers, optional clean output)**

DBA – UTILITÁRIO SQLCMD

SQLCMD

O sqlcmd utilitário permite que você insira instruções Transact-SQL, procedimentos do sistema e arquivos de script no prompt de comando

PARÂMETROS

- m error_level
- M multisubnet_failover
- N (encrypt connection)
- o output_file
- p[1] (print statistics, optional colon format)
- P password
- q "cmdline query"
- Q "cmdline query" (and exit)
- r[0 | 1] (msgs to stderr)
- R (use client regional settings)

DBA – UTILITÁRIO SQLCMD

SQLCMD

O sqlcmd utilitário permite que você insira instruções Transact-SQL, procedimentos do sistema e arquivos de script no prompt de comando

PARÂMETROS

- s col_separator
- S [protocol:]server[instance_name][,port]
- t query_timeout
- u (unicode output file)
- U login_id
- v var = "value"
- V error_severity_level
- w column_width
- W (remove trailing spaces)
- x (disable variable substitution)

DBA – UTILITÁRIO SQLCMD

SQLCMD

O sqlcmd utilitário permite que você insira instruções Transact-SQL, procedimentos do sistema e arquivos de script no prompt de comando

PARÂMETROS

- X[1] (disable commands, startup script, environment variables, optional exit)**
- y variable_length_type_display_width**
- Y fixed_length_type_display_width**
- z new_password**
- Z new_password (and exit)**
- ? (usage)**

DBA – UTILITÁRIO SQLCMD

SQLCMD

O sqlcmd utilitário permite que você insira instruções Transact-SQL, procedimentos do sistema e arquivos de script no prompt de comando

```
C:\Users\andre>sqlcmd -S infinity\sqlexpress -d curso -q "select * from cidades" -s "|" -o "C:\Cursos\Database\Log\cidades.txt"
1> go
1> exec sp_spaceused
2> go
```

<https://docs.microsoft.com/pt-br/sql/tools/sqlcmd-utility?view=sql-server-2017>

DBA – OTIMIZANDO PERFORMANCE

PROCESSAMENTO LÓGICO

Ordem de processamento lógico da instrução SELECT

- 1-FROM
- 2-ON
- 3-JOIN
- 4-WHERE
- 5-GROUP BY
- 6-WITH CUBE ou WITH ROLLUP
- 7-HAVING
- 8-SELECT
- 9-DISTINCT
- 10-ORDER BY
- 11-INÍCIO

Esta sequência geralmente é verdadeira. No entanto, há casos incomuns em que a sequência pode ser diferente.

<https://docs.microsoft.com/pt-br/sql/relational-databases/query-processing-architecture-guide?view=sql-server-2017#sql-statement-processing>

DBA – OTIMIZANDO PERFORMANCE

Ordem de processamento lógico da instrução SELECT

5

- SELECT <CAMPOS>

1

- FROM <TABELAS>

2

- WHERE <FILTROS>

3

- GROUP BY <CAMPOS AGREGADOS>

4

- HAVING <FILTRO AGREGADOS>

6

- ORDER BY <ORDENAÇÃO CAMPOS>

<https://docs.microsoft.com/pt-br/sql/relational-databases/query-processing-architecture-guide?view=sql-server-2017#sql-statement-processing>

DBA – CICLO DE AJUSTE DE PERFORMANCE



DBA – DICAS DE PERFORMANCE

ETAPAS

1. Armazene as informações relevantes e necessárias no banco de dados em vez de estrutura ou matriz de aplicativos.
2. Use tabelas **normalizadas** no banco de dados. Tabelas menores são melhores do que tabelas normalizadas genéricas, o que as tornam grandes.
3. Use **números** ao invés de strings para ID's nas tabelas.
4. Mantenha a chave primária de **caracteres menores ou inteiros**. É mais fácil processar chaves de pequena largura.
5. **Armazene caminhos de imagem ou URLs** no banco de dados em vez de imagens. Tem menos sobrecarga.
6. Use tipos de banco de **dados apropriados para os campos**. Se data_log no banco de dados é arquivado para usar data e hora como tipos de dados, não use VARCHAR (20).
7. Especifique nomes de colunas em vez de usar ***** na instrução SELECT.

DBA – DICAS DE PERFORMANCE

ETAPAS

8. Use a cláusula **LIKE** adequadamente. Se você estiver procurando a correspondência exata use “**=**” em vez disso.
9. Escreva as palavra-chave do SQL em letras maiúsculas para fins de legibilidade.
10. **Usar JOIN é melhor para desempenho** do que usando **sub-consultas ou consultas aninhadas**.
11. Utilize **Stored Procedures, Functions ou views**. Elas são mais rápidas e ajudam na manutenção e segurança do banco de dados.
12. Criação de **index adequada melhora a velocidade das operações** no banco de dados.
13. Analyse Query in Database Engine Tuning Advisor
14. Keywords que você **deve** usar(TOP, DISTINCT , COUNT)
15. Keywords que você **deve evitar**(LIKE, IN)
16. LOCK ou **NOLOCK**
17. Execute Plano de execução.

Clusterizado vs Não-Clusterizado

Clusterizado

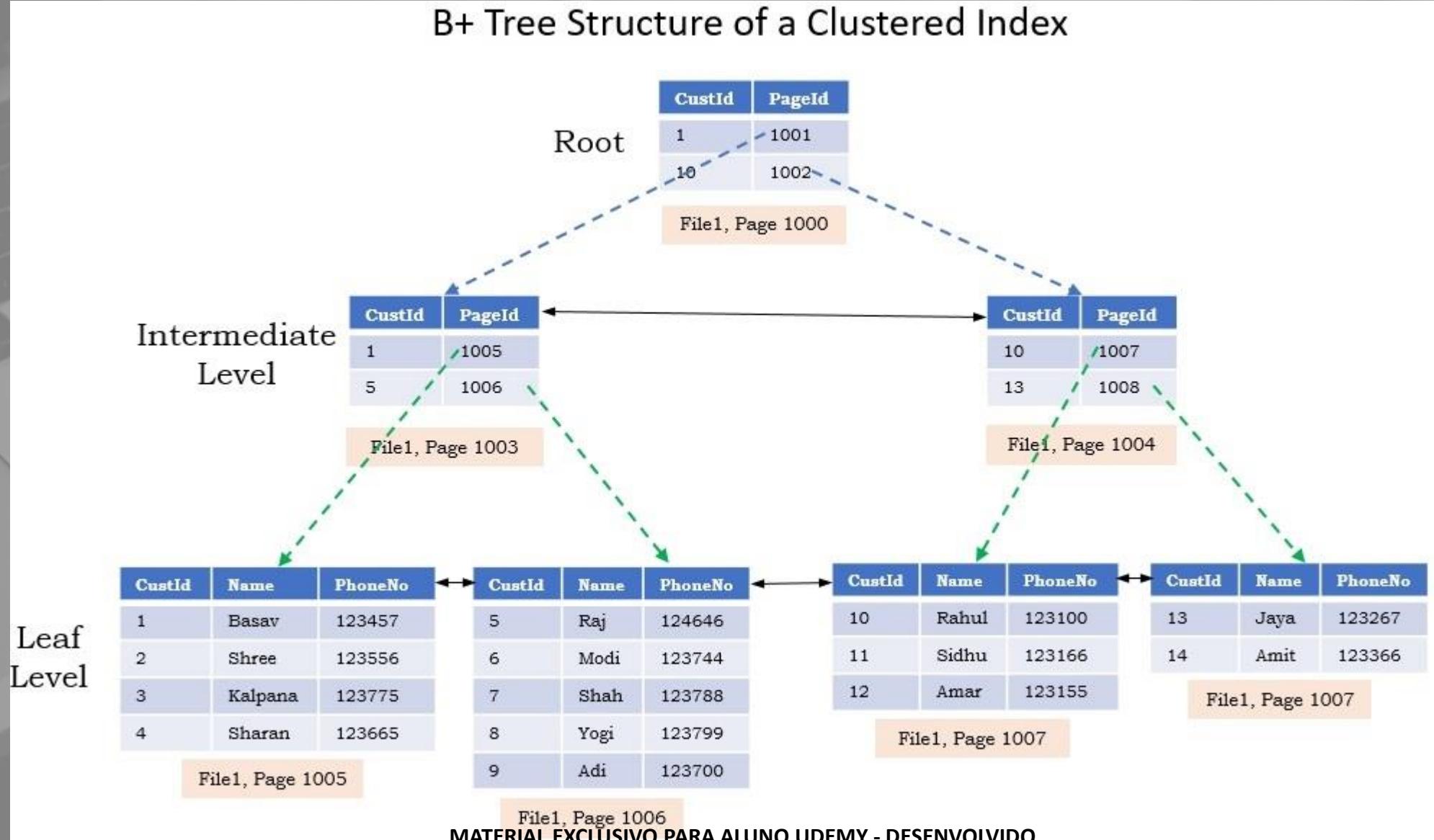
- Determina a ordem em que as linhas de uma tabela são armazenadas no disco físico.
- Cada tabela pode ter apenas 1 índice clusterizado.

Não-Clusterizado

- Ordena somente o índice em si, e não as linhas

DBA - INDICES - INDEX

B+ Tree Structure of a Clustered Index



DBA – ALWAYS ON (HA-DR)

ALTA DISPONIBILIDADE (HA)

RECUPERAÇÃO DE DESASTRE (DR)

DBA – ALWAYS ON (HA-DR)



FALHAS EM DISCO

MATERIAL EXCLUSIVO PARA ALUNO UDEMY - DESENVOLVIDO
POR ANDRÉ ROSA - IT FOREST

DBA – ALWAYS ON (HA-DR)



DBA – ALWAYS ON (HA-DR)



MANUTENÇÃO PREVENTIVA

MATERIAL EXCLUSIVO PARA ALUNO UDEMY - DESENVOLVIDO
POR ANDRÉ ROSA - IT FOREST

DBA – ALWAYS ON (HA-DR)



RECUPERACAO DE DESASTRES

MATERIAL EXCLUSIVO PARA ALUNO UDEMY – DESENVOLVIDO
POR ANDRÉ ROSA - IT FOREST

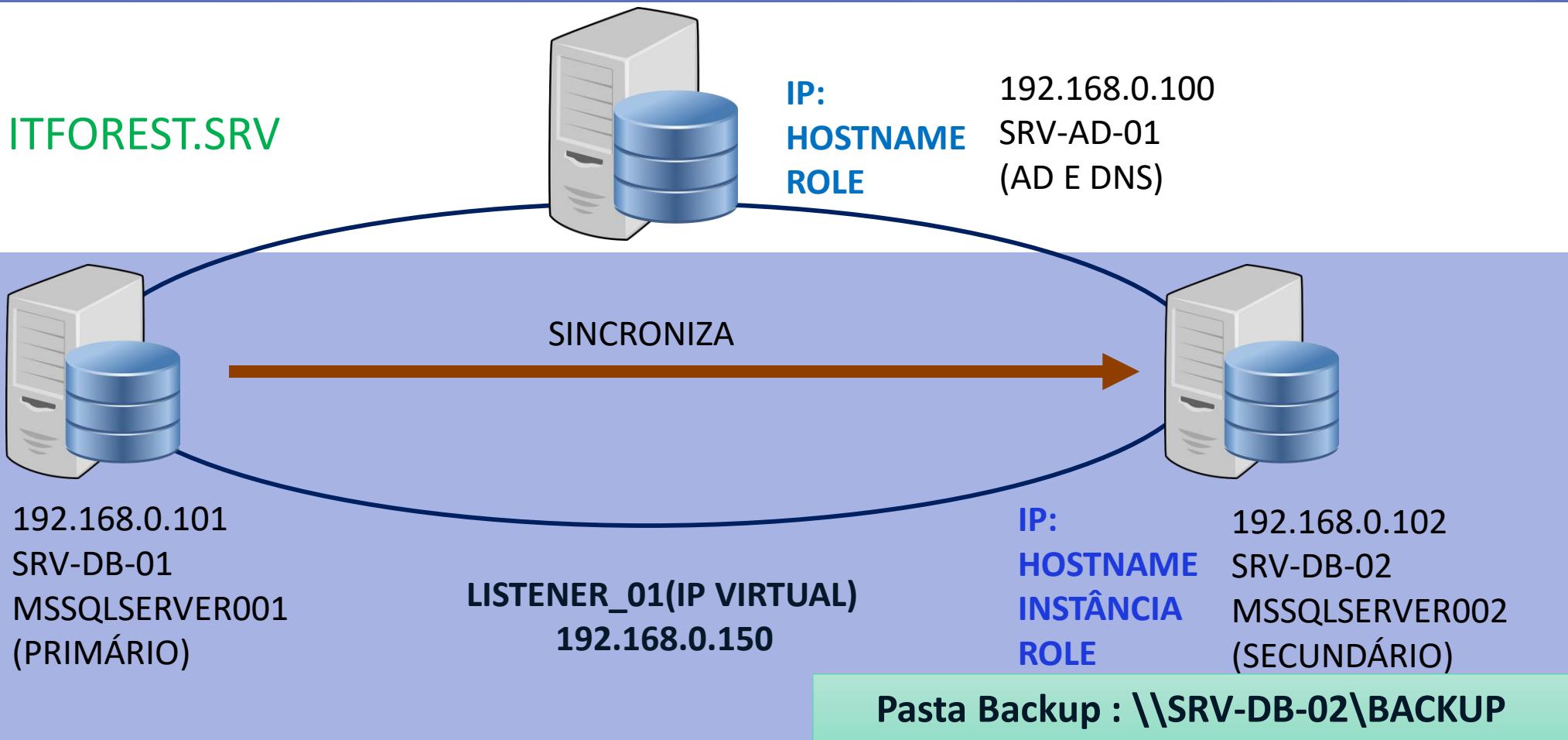
DBA – ALWAYS ON (HA-DR)

O **Always On Availability Groups** é uma solução de alta disponibilidade e **recuperação de desastres** de nível corporativo, introduzida no SQL Server 2012 para maximizar a disponibilidade de um ou mais bancos de dados de usuários.

Os AlwaysOn Availability Group **exigem** que as instâncias do SQL Server residam nos nós do **Windows Server Failover Clustering (WSFC)**.

DBA – ALWAYS ON CENÁRIO (CURSO)

DOMINIO = ITFOREST.SRV

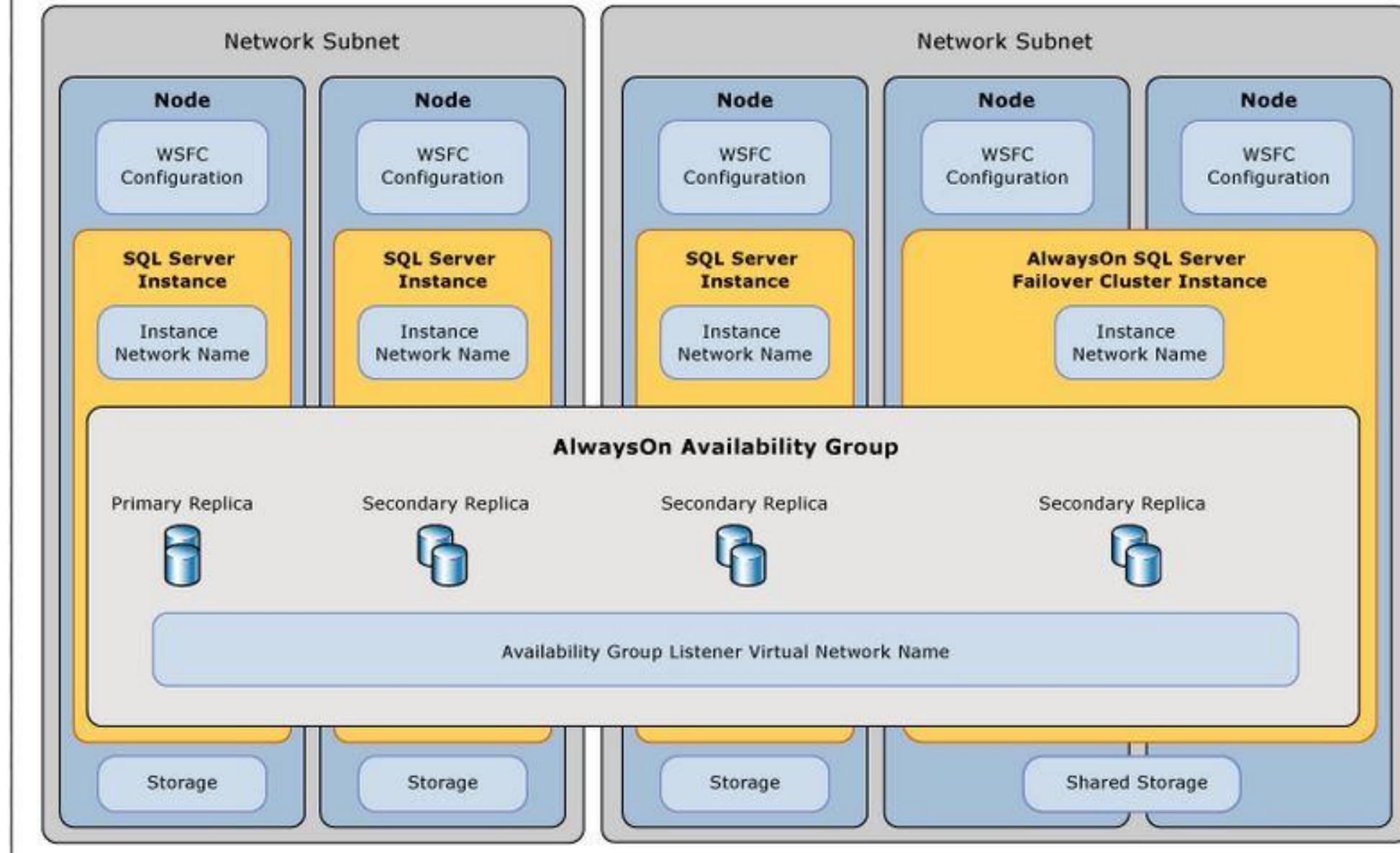


Grupo de disponibilidade (FG_DB_01)

QUORUM : \\SRV-AD-01\QUORUM

(WSFC)FAILOVER CLUSTERING(CLUSTER_DB) 192.168.0.99

Windows Server Failover Clustering (WSFC) Cluster



MATERIAL EXCLUSIVO PARA ALUNO UDEMY – DESENVOLVIDO

WSFC Quorum Share POR ANDRÉ ROSA IT FOREST

DBA - ALWAYS ON - PRÉ-REQUISITOS



VERIFIQUE SE O SISTEMA NÃO É UM CONTROLADOR DE DOMÍNIO.



VERIFIQUE SE CADA COMPUTADOR ESTÁ EXECUTANDO O WINDOWS SERVER 2012 OU VERSÕES POSTERIORES. (.NET FRAMEWORK 3.5)



VERIFIQUE SE CADA COMPUTADOR É UM NÓ EM UM CLUSTER DO WINDOWS SERVER FAILOVER CLUSTERING (WSFC).



INSTALAÇÃO AUTÔNOMA DO SQL SERVER EM CADA NÓ.

DBA – ALWAYS ON - ETAPAS



Configura Windows Server Failover Clustering (WSFC).



Configura Quorum (Witness)



Configura Grupo de Alta de Disponibilidade.



Configura Listener.

DBA – ALWAYS ON – PONTOS DE ATENÇÃO

Comunicação entre os NODES (DNS) e
IP's

Instalar .net Framework 3.5

WINDOWS ATUALIZADO

DBA - ALWAYS ON - PONTOS DE ATENÇÃO

FIREWALL (LIBERAR PORTAS DO SERVICOS) (1433,5022)

INICIAR SQL SERVER COM USUÁRIO DO DOMÍNIO

HABILITAR PROTOCOLO NAMED PIPES E IP (CONFIGURATION MANAGER)

DBA - ALWAYS ON - PONTOS DE ATENÇÃO

PERMISSÃO DE PASTAS COMPARTILHADAS

INICIAR O SERVIÇO
(SQL SERVER BROWSER)

MAQUINAS PRIMÁRIAS E SECUNDÁRIOS
MESMA CONFIGURAÇÃO

Modo de Failover

Exibe o modo de failover para o qual a réplica está configurada. Os valores possíveis do modo de failover são:

Automático. Indica que uma ou mais réplicas estão em modo de failover automático.

Manual. Indica que nenhuma réplica está em modo de failover automático.

Modo de Disponibilidade

Indica a propriedade de réplica que você define separadamente para cada réplica de disponibilidade. Esse valor é ocultado por padrão. Os valores possíveis são:

Assíncrono. A réplica secundária nunca é sincronizada com a réplica primária.

Synchronous. Ao ficar em dia com o banco de dados primário, um banco de dados secundário entra nesse estado e permanece em dia, desde que a sincronização continue para o banco de dados.

Estado da Sincronização

Indica se uma réplica secundária está sincronizada no momento com a réplica primária. Essa coluna é mostrada por padrão. Os valores possíveis são:

Não Sincronizado. Um ou mais bancos de dados na réplica não estão sincronizado ou ainda não foram unidos ao grupo de disponibilidade.

Sincronizando. Um ou mais bancos de dados na réplica estão sendo sincronizados.

Sincronizado Todos os bancos de dados na réplica secundária são sincronizados com os bancos de dados primário correspondentes na réplica primária atual, se houver, ou na réplica primária mais recente.

Modos de Conexão no Secundário

O modo de conexão para cada servidor secundário pode ser:

Dissallow Connections (Não Permitir Conexões) – As réplicas secundárias não permitem que seja realizada nenhuma conexão.

Allow Only Read-Intent Connections – A réplica permite somente a leitura de conexões que tem a intenção de ler e passam pelo native client do SQL Server.

Allow all Connections – É permitido qualquer conexão.