

Administração de BD

José Antônio da Cunha

CEFET-RN

09/10/2000

Filegroups

- **Utilizando filegroups para ganho de performance e gerenciamento de espaço**

Filegroups - são estruturas lógicas que sustentam os arquivos de dados em um database. Um database padrão possui um arquivo de dados e um arquivo de log; o arquivo de dados está associado a um filegroup chamado PRIMARY. Pode-se criar outros arquivos de dados assim como outros filegroups, mas porque, como e quando criar outros filegroups?

Filegroups

- **Arquitetura de um database**

Apesar do nome singular, um database é uma estrutura formada por pelo menos dois arquivos: um para armazenamento de dados (Master Data File, extensão .MDF) e outro reservado para o log de transações (Log Data File, extensão .LDF). Veja a **Figura 1**.

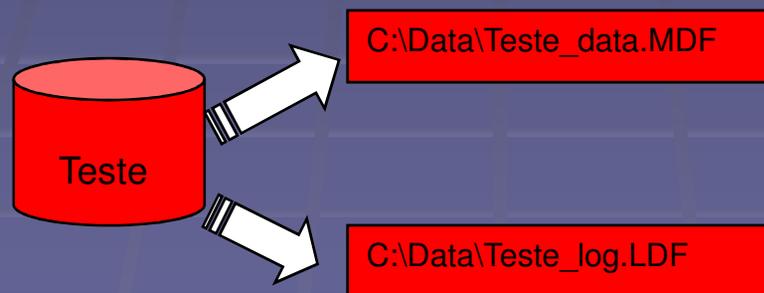


Figura 1: Estrutura típica de um database.

Filegroups

- **Arquitetura de um database**

além dos arquivos **.MDF** e **.LDF**, é possível criar outros arquivos para armazenamento de dados. Estes arquivos secundários possuem extensão **.NDF**, de Secondary Data Files e podem ser criados no mesmo filegroup do arquivo **.MDF** (PRIMARY) ou em outro filegroup. A decisão de utilizar o mesmo filegroup ou criar um novo depende da finalidade do arquivo secundário. A seguir estão listados algumas situações cuja resolução baseia-se na implementação de filegroups:

Filegroups

- **Arquitetura de um database**

- O arquivo de dados principal atingiu um tamanho que extrapola a capacidade da unidade de fita DLT utilizada no backup. Esse problema pode ser resolvido com a realocação de tabelas em outro filegroup, distribuindo o backup final em partes menores que não ultrapassem a capacidade da fita. Nesse caso, deve-se criar outro filegroup para armazenar o arquivo secundário.

Filegroups

■ Arquitetura de um database

- Você precisa criar um database com tamanho inicial de 35GB, mas não possui esse espaço em uma única unidade de disco. A distribuição é a seguinte: 20GB na unidade C e 15GB na unidade D. qual a solução? Crie um Master Data File com 15GB em C e um Secondary Data File com 20GB em D. os dois arquivos constituirão uma unidade única de gerenciamento de espaço. O arquivo secundário criado na unidade D será visto pelo SQL Server como uma extensão natural do arquivo primário. Nesses casos, o arquivo secundário deverá ser criado no mesmo filegroup do arquivo que se deseja expandir (PRIMARY).

Filegroups

■ Arquitetura de um database

- Você pode melhorar a performance de um database criando índices e tabelas em filegroups distintos, localizados em unidades e controladores específicos. Se as páginas de dados das tabelas forem armazenadas em unidades diferentes daquela utilizada para os índices (por exemplo C e D), pesquisas que utilizam índices serão beneficiadas por leituras executadas em paralelo. Nesse caso, deve-se criar outro filegroup para armazenar o arquivo secundário.

A **Figura 2** é um retrato de um database que utiliza um filegroup secundário para armazenamento de índices.

Filegroups

- Arquitetura de um database

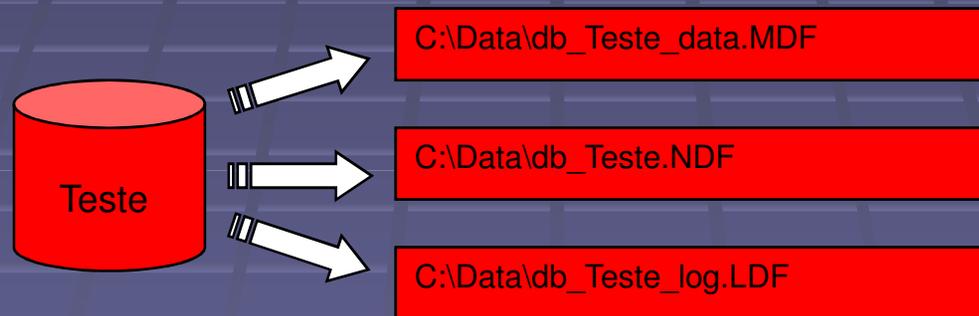


Figura 2: Database que utiliza arquivo secundário (.NDF) para armazenamento de índices

Filegroups

- **Criando um database através de comandos T-SQL**

```
CREATE DATABASE [Teste] ON PRIMARY
(
  NAME                = N'Teste_Data',
  FILENAME             = N'c:\Data\Teste_Data.MDF',
  SIZE                 = 1,
  FILEGROWTH           = 10%
)
LOG ON
(
  NAME                = N'Teste_Log',
  FILENAME             = N'c:\Data\Teste_Log.LDF',
  SIZE                 = 1,
  FILEGROWTH           = 10%
)
```

Filegroups

- **Criando um database através de comandos T-SQL**

No exemplo anterior, criamos o database Teste_data no filegroup PRIMARY representado por c:\Data\Teste_data.MDF. o database também possui um arquivo de Log em c:\Data\Teste_Log.LDF, mas ainda não criamos arquivos secundários.

Filegroups

- **Criando arquivo secundário**

```
ALTER DATABASE Teste_Data
    ADD FILE
    (...
    FILENAME = N'c:\Data\Teste_Data2.NDF',
    SIZE = 1,
    ...)
    FILEGROWTH = 10%
) TO FILEGROUP [PRIMARY]
```

```
CREATE DATABASE Exemplo
ON PRIMARY
( NAME = exemplo_prim,
  FILENAME = 'C:\Dadossql\exemplo\exemplo_prim.mdf',
  SIZE = 10,
  MAXSIZE = 50,
  FILEGROWTH = 15%),
( NAME = exemplo_sec1,
  FILENAME = 'C:\Dadossql\exemplo\exemplo_sec1.ndf',
  SIZE = 10,
  MAXSIZE = 50,
  FILEGROWTH = 15%),
FILEGROUP VendasGroup1 --File grupo secundário
( NAME = exemplo_sec2,
  FILENAME = 'C:\Dadossql\exemplo\exemplo_sec2.ndf',
  SIZE = 10,
  MAXSIZE = 50,
  FILEGROWTH = 5%),
```

```
( NAME = exemplo_sec3,  
  FILENAME = 'C:\Dadossql\exemplo\exemplo_sec3.ndf',  
  SIZE = 10,  
  MAXSIZE = 50,  
  FILEGROWTH = 15%),  
  
FILEGROUP VendasGroup2 --segundo filegroup  
( NAME = exemplo_sec4,  
  FILENAME = 'C:\Dadossql\exemplo\exemplo_sec4.ndf',  
  SIZE = 10,  
  MAXSIZE = 50,  
  FILEGROWTH = 5),  
( NAME = exemplo_sec5,  
  FILENAME = 'C:\Dadossql\exemplo\exemplo_sec5.ndf',  
  SIZE = 10,  
  MAXSIZE = 50,  
  FILEGROWTH = 15%)
```

LOG ON

```
( NAME = exemplo_log,  
  FILENAME = 'C:\Dadossql\exemplo\exemplo_lo.ldf',  
  SIZE = 5M,  
  MAXSIZE = 25M,  
  FILEGROWTH = 5M)
```

Filegroups

■ Conclusão

No SQL Server (versões 7.0 ou superiores), temos o conceito de Filegroup. Este conceito muitas vezes não é utilizado na prática por falta de entendimento do que ele significa, como pode ser utilizado e quais as vantagens em utilizar filegroups.

A utilização de um filegroup permite que os arquivos de um Banco de Dados sejam agrupados para facilitar o gerenciamento, bem como a distribuição ao longo de volumes redundantes e de melhor desempenho (RAID-0, RAID-1, etc.). A utilização de filegroup pode ser um auxiliar valioso na melhoria do desempenho de um Banco de Dados, ao permitir que o Banco de Dados (através dos seus diversos arquivos) seja criado em múltiplos discos, múltiplas controladoras ou em sistemas do tipo RAID. Também podemos fazer com que uma tabela ou índice seja criada em um disco específico, simplesmente associando a tabela ou índice com um filegroup.

Filegroup

NOTA: Os arquivos de log não fazem parte de nenhum filegroup, sendo que estes são gerenciados separadamente pelo Banco de Dados.

FIM