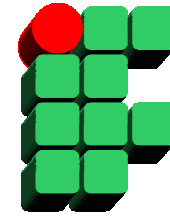


Formação de DBAs SQL Server 2008

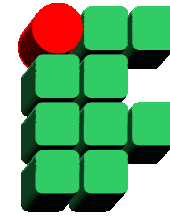
Parte 5: Backup e Restore

Por que realizar backups?



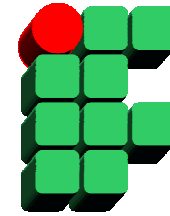
- Em muitas situações, o backup é a principal alternativa para restaurar uma situação aceitável dos dados, após uma falha de hardware, software, um incidente de segurança ou uma catástrofe;
- Definir uma política de backup eficaz e testá-la regularmente é fundamental para garantir a continuidade de um banco de dados.

Tipos de Falha



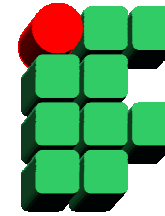
- Falha de Hardware:
 - Discos defeituosos podem corromper os dados nele armazenados. Se o sistema não prevê a utilização de RAID ou de outro mecanismo de redundância, os dados podem ser perdidos definitivamente;
 - Falhas em múltiplos componentes, especialmente relacionados aos subsistemas de I/O, podem comprometer os dados, mesmo em um ambiente com redundância;
 - Catástrofes como incêndios e inundações podem destruir completamente um data center, deixando poucas alternativas de recuperação dos dados.

Tipos de Falha

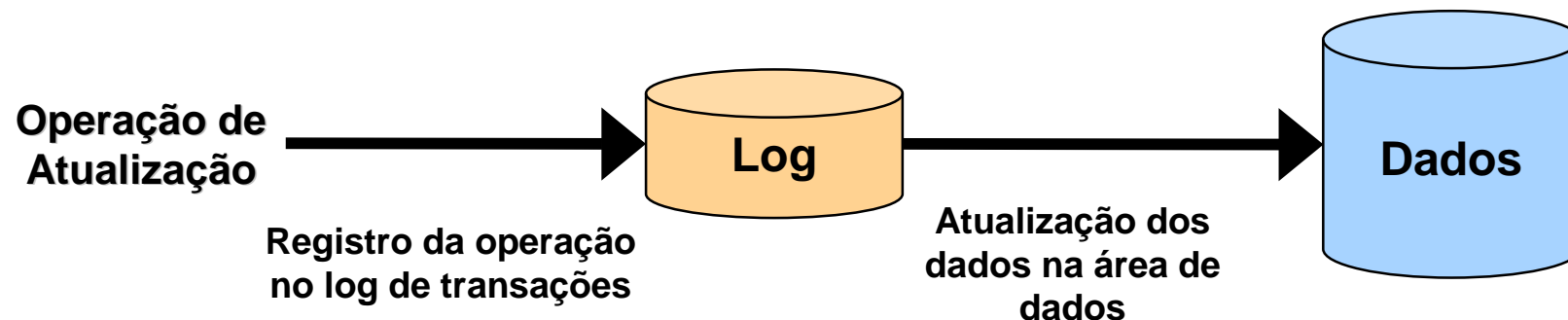


- Falha de Software:
 - Sistemas operacionais, aplicações e o próprio SQL Server 2008 podem conter bugs que ocasionem a ocorrência de falhas, podendo comprometer os dados.
- Falha Humana:
 - Uma pessoa com acesso autorizado (DBA ou desenvolvedor) pode excluir ou atualizar acidentalmente o banco de dados.
- Falha de Segurança:
 - Uma pessoa sem autorização pode obter acesso indevidamente às informações, modificá-las ou destruí-las.

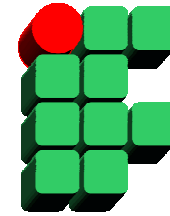
Log de Transações



- O log de transações registra as operações de atualização executadas no banco de dados;
- O processo de recuperação automática (recovery) depende do log de transações;
- Para garantir a integridade dos dados, os registros das operações são gravados no log antes dos dados serem atualizados na área de dados (gravação write-ahead).

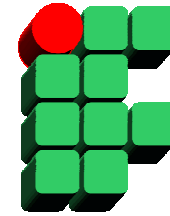


Modo de Recuperação



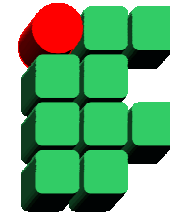
- O modo de recuperação (recovery model) é uma propriedade de cada banco de dados que determina a forma como os registros do log de transação serão realizados;
- O modo de recuperação influencia o processo de recuperação automática (recovery) e as possibilidades de backup;
- Existem 3 modos possíveis de recuperação no SQL Server 2008: simple, bulk-logged e full.

Modo de Recuperação



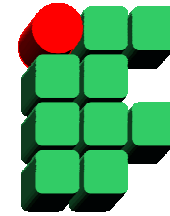
- Modo Simple
 - Modo mais simplificado de registro das operações no log de transações;
 - A cada checkpoint, os registros das transações concluídas são excluídos automaticamente, economizando espaço em disco e facilitando a administração;
 - Nesse modo não é possível realizar backups do log de transações, logo, restaurações para um momento específico no tempo (point in time recovery) não podem ser realizadas.

Modo de Recuperação



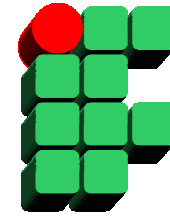
- Modo Full
 - Modo mais completo de registro das operações no log de transações, incluindo operações de bulk-load e indexação;
 - Os registros das transações concluídas serão excluídos somente após a execução de um backup do log, logo, uma rotina de backup periódico do log de transações é fundamental quando o modo de recuperação está configurado para FULL;
 - Consome mais espaço por registrar mais operações e exigir uma área para armazenamento para os backups de log;
 - Nesse modo é possível realizar restaurações para um momento específico no tempo (point in time recovery).

Modo de Recuperação



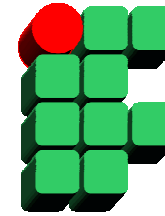
- Modo Bulk-Logged
 - Modo intermediário entre o SIMPLE e o FULL. Operações de bulk-load e indexação são armazenadas de forma resumida, enquanto as demais operações são registradas completamente;
 - Os registros das transações concluídas serão excluídos somente após a execução de um backup do log, logo, uma rotina de backup periódico do log de transações é fundamental quando o modo de recuperação está configurado para BULK-LOGGED;
 - Economiza espaço em disco para as operações de bulk-load e indexação, mas consome o mesmo espaço que o modo FULL para as demais operações;
 - Nesse modo é possível realizar restaurações para um momento específico no tempo (point in time recovery), desde que os backups de log não contenham operações de bulk-load e indexação.

Tipos de Backup

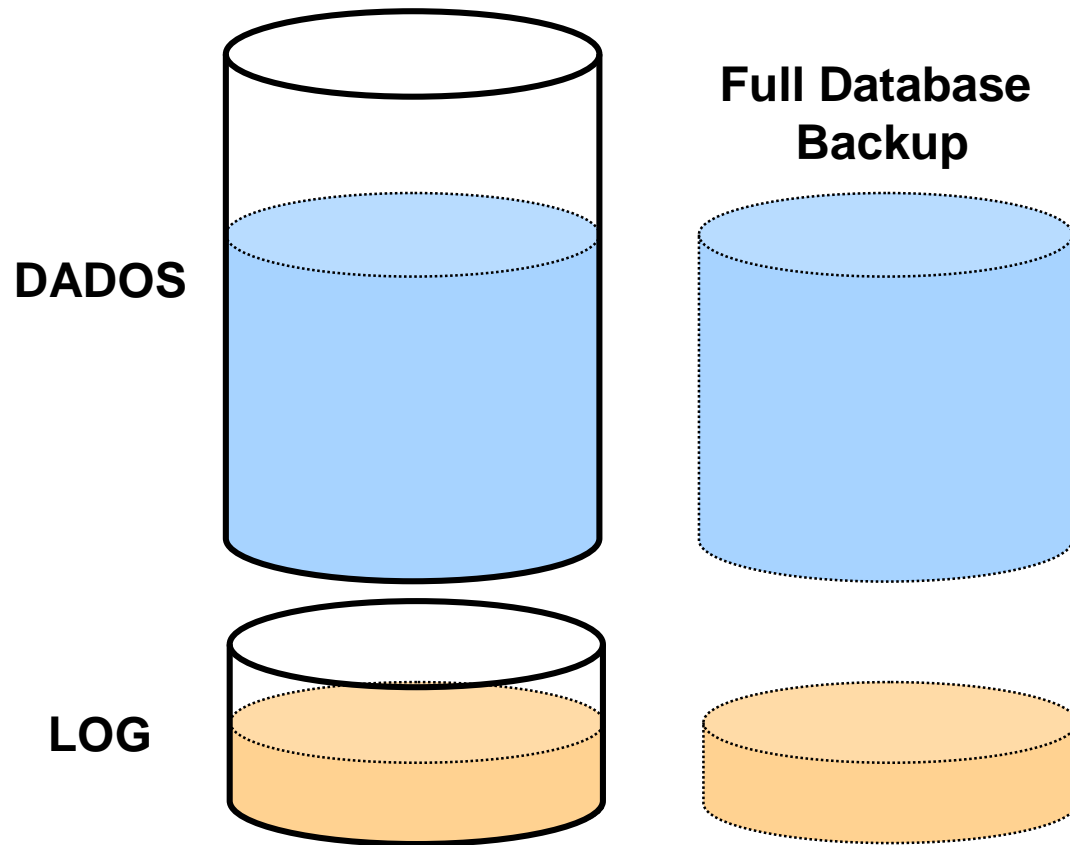


- Full Database Backup:
 - Backup completo dos dados de uma banco de dados e dos registros das transações gerados até a conclusão do backup;
 - Serve de base para os demais tipos de backup e estará presente em qualquer estratégia de backup a ser definida;
 - Ocupa mais espaço em disco, acarreta uma sobrecarga maior e demora mais tempo para ser executado do que os demais tipos de backup.

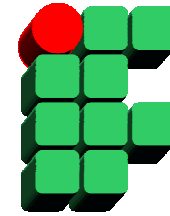
Tipos de Backup



- Full Database Backup:

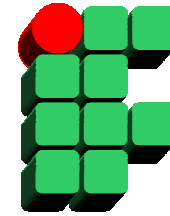


Tipos de Backup

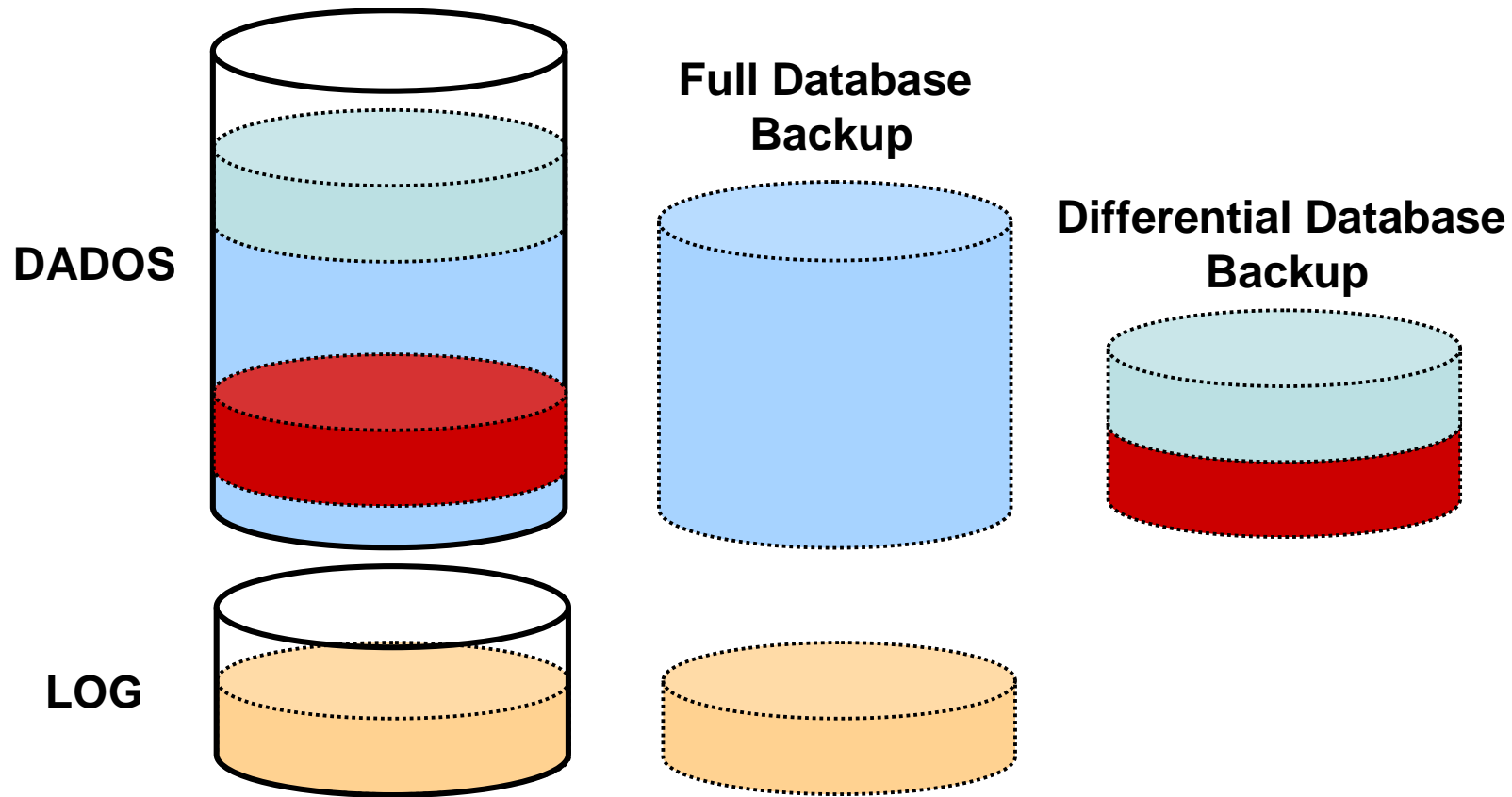


- Differential Database Backup:
 - Backup de todos os dados atualizados desde o último full database backup;
 - Depende do último full database backup para ser restaurado;
 - Tende a ocupar menos espaço em disco, acarretar uma sobrecarga menor e demorar menos tempo para ser executado do que o full database backup.

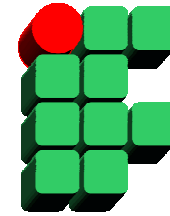
Tipos de Backup



- Differential Database Backup:

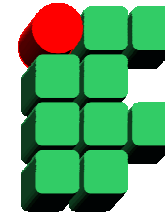


Tipos de Backup

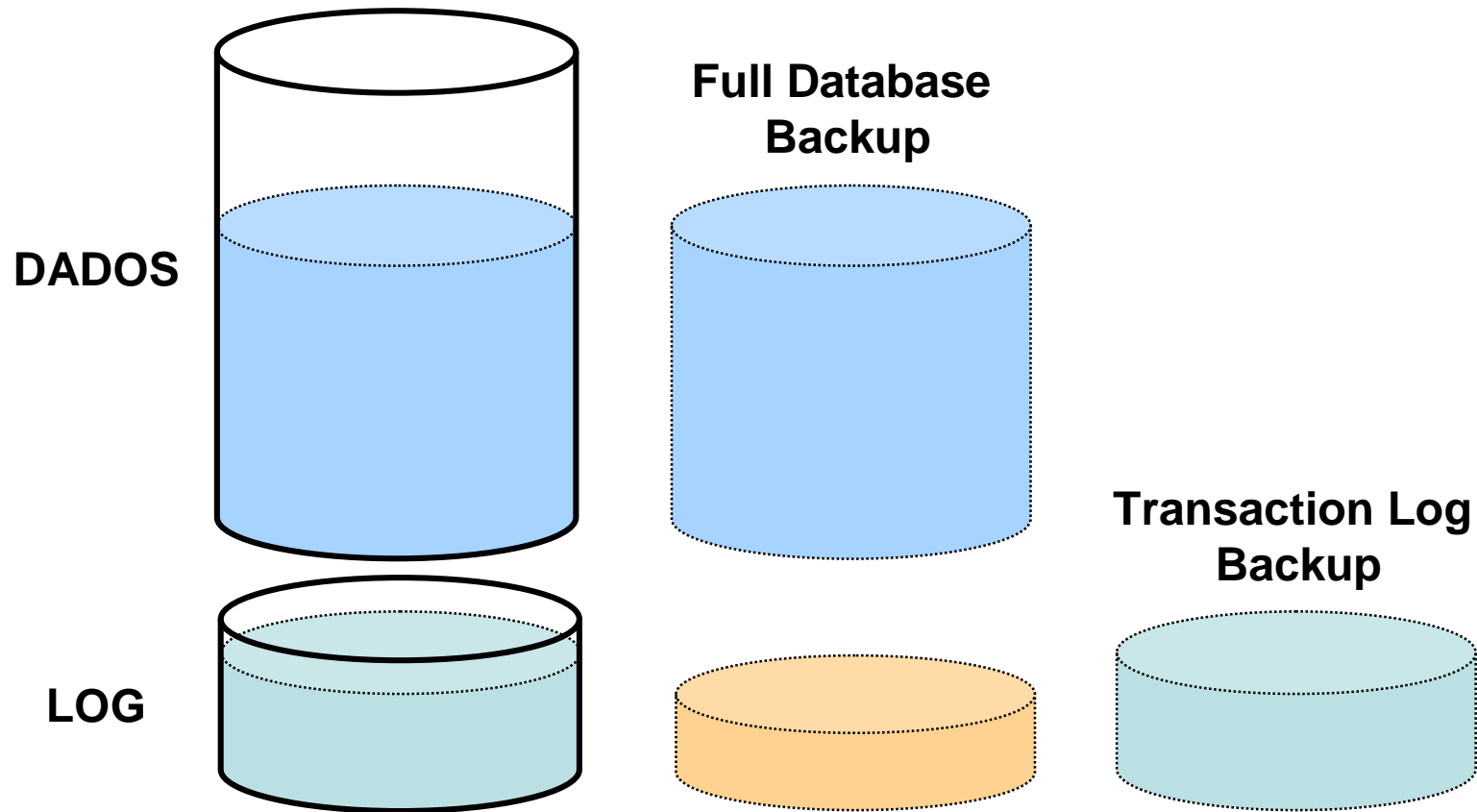


- Transaction Log Backup:
 - Backup dos registros do log de transações;
 - Depende do último full database backup para ser restaurado;
 - Permite que seja realizado um restore até um ponto determinado no tempo;
 - O espaço em disco ocupado, a sobrecarga de execução e a duração dependem do volume de atualizações no banco de dados, da frequência de execução do backup, do modo de recuperação adotado e do tamanho do log de transações;
 - Não pode ser realizado quando modo de recuperação SIMPLE está configurado.

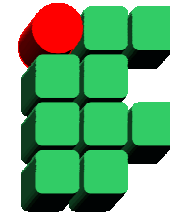
Tipos de Backup



- Full Database Backup:

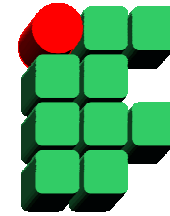


Outros Tipos de Backup



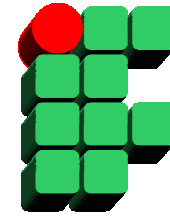
- Partial Database Backup:
 - Backup do primary filegroup e de todos os filegroups que não estejam marcados como READ ONLY;
 - Utilizado para evitar a cópia de dados somente leitura e reduzir o tamanho total do backup;
 - Depende do último full database backup para ser restaurado.
- File Backup ou Filegroup Backup:
 - Backup de arquivos ou filegroups específicos;
 - Utilizado em grandes bancos de dados para dividir a carga do backup em diferentes partes, minimizando a sobrecarga da execução de backup;
 - Utilizado em conjunto com backups de log para manter a consistência com os demais filegroups.

Outros Tipos de Backup



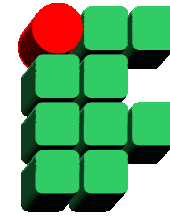
- Copy-Only Backup:
 - Similar ao full database backup, mas não fica registrado no histórico de backup, logo, não serve de referência para futuros backups.
 - Utilizado para realizar uma cópia do banco de dados sem interferir na rotina periódica de backup;
- Tail Log Backup:
 - Quando o log de transações não foi corrompido quando uma falha ocorreu, é recomendada a execução de um último backup de log, de forma a possibilitar a recuperação dos dados até o momento da falha. Esse backup é chamado de Tail Log Backup.

Estratégias de Backup

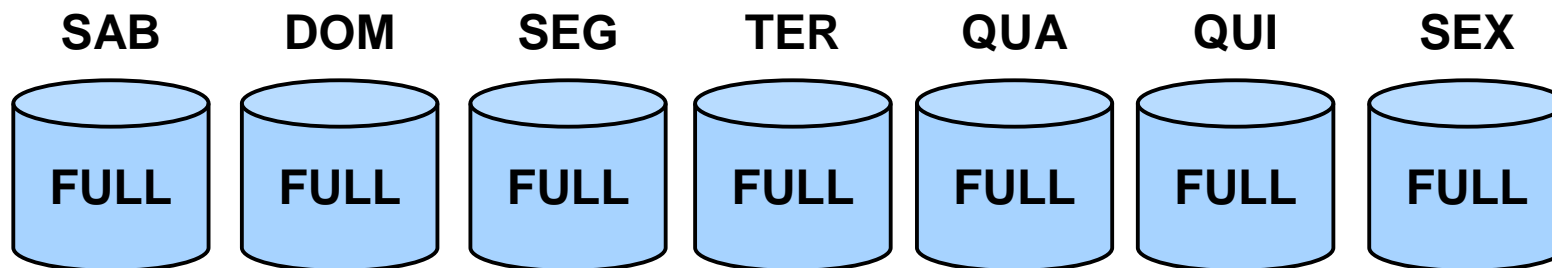


- Questionamentos a serem considerados para definição de uma estratégia de backup:
 - Qual a importância (criticidade, valor) da informação para o negócio?
 - Existe alguma janela de execução para o backup?
 - Existem porções somente leitura dos dados que podem ser facilmente separadas das demais?
 - Qual o volume de dados envolvido?

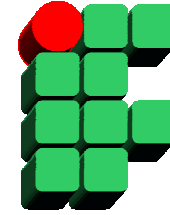
Sugestões de Estratégias de Backup



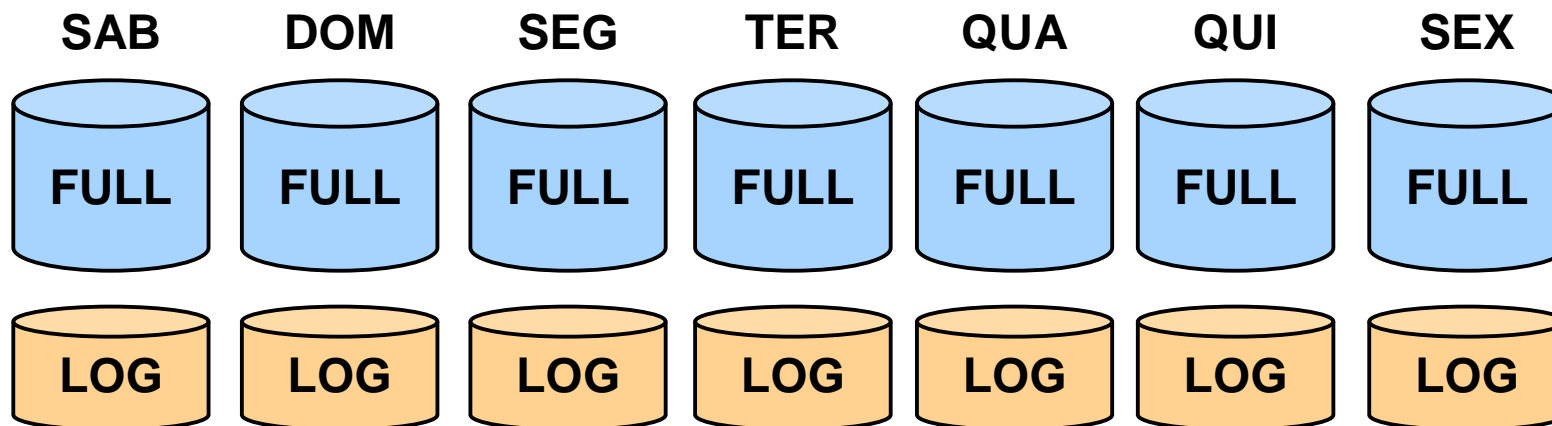
- Full Database diário:
 - Indicado para ambientes com baixa frequência de atualização, com bancos de dados pequenos e informações de pouco valor.



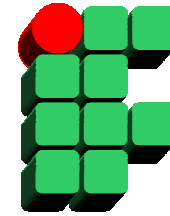
Sugestões de Estratégias de Backup



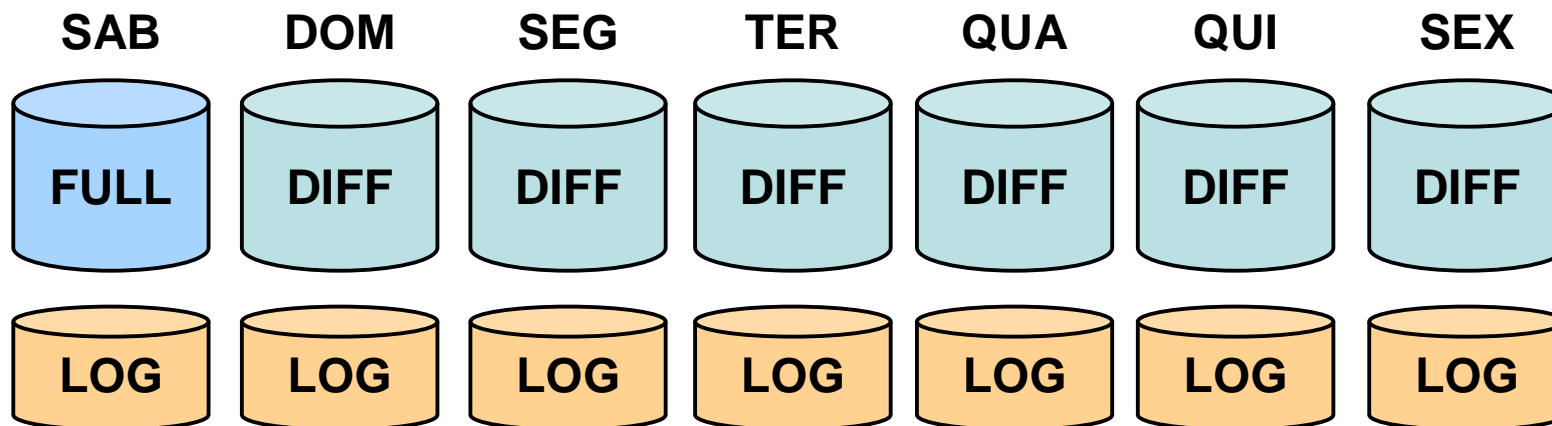
- Full Database (diário) + Log (30 min):
 - Indicado para ambientes com alta frequência de atualização, com bancos de dados pequenos e informações críticas.



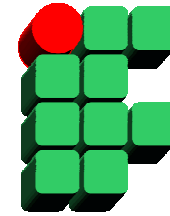
Sugestões de Estratégias de Backup



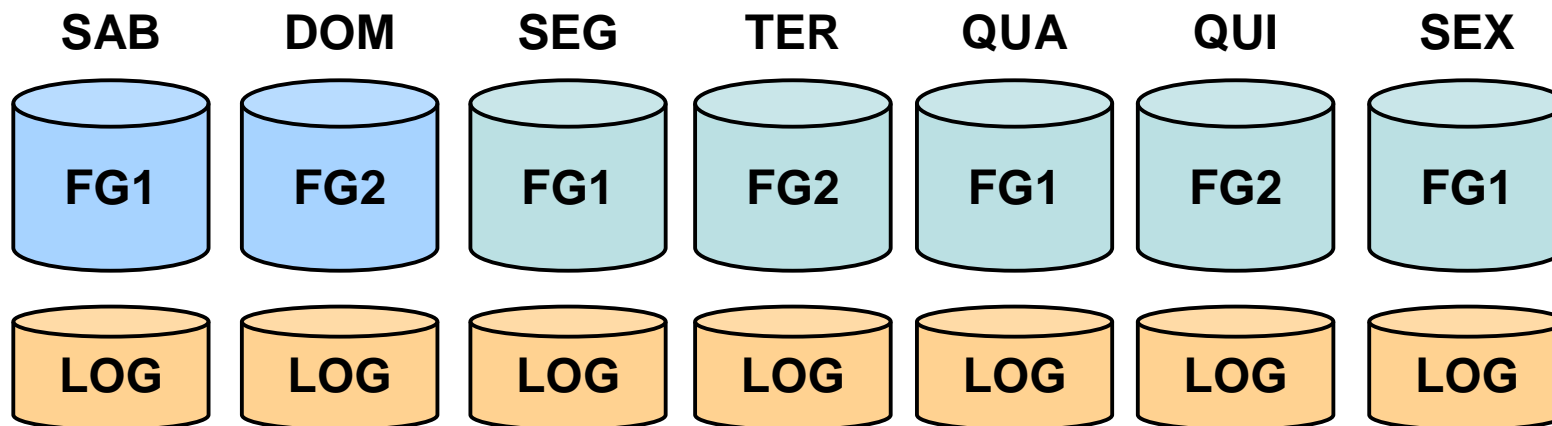
- Full Database (Semanal) + Diferencial (Diário) + Log (30 min):
 - Indicado para ambientes com alta frequência de atualização, com bancos de dados grandes e informações críticas.



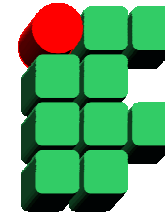
Sugestões de Estratégias de Backup



- Full Filegroup 1 (Semanal) + Full Filegroup 2 (Semanal)
Diferencial Filegroup 1 (2 dias) +
Diferencial Filegroup 2 (2 dias) +
Log (30 min):
 - Indicado para ambientes com alta frequência de atualização, com bancos de dados muito grandes e informações críticas.



Restaurando Bancos de Dados



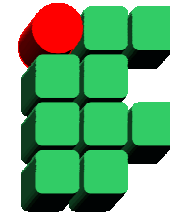
- Restore Path
 - Conjunto completo de backups necessários para restaurar um banco de dados;
 - Exemplos:
 - Restore a partir de um modelo FULL + LOG:



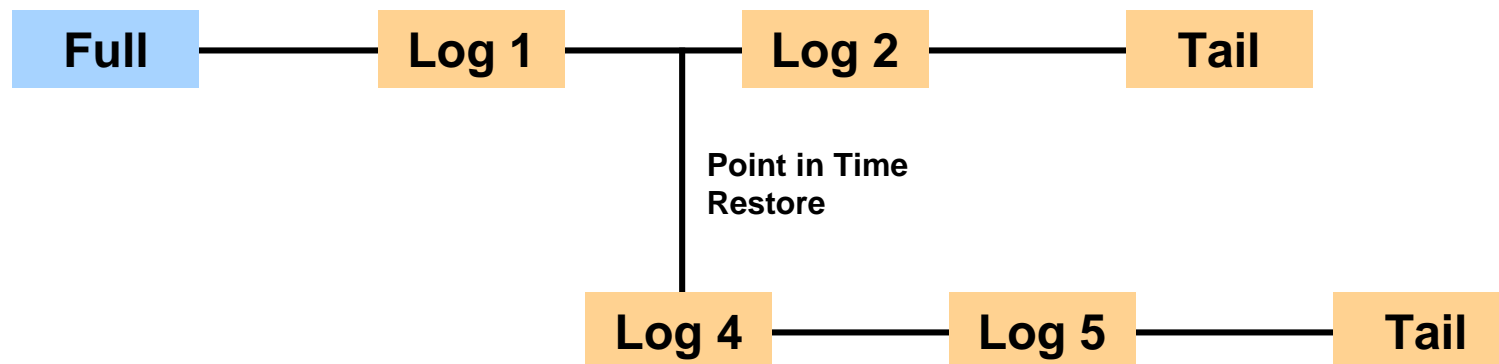
- Restore a partir de um modelo FULL + DIFF + LOG:



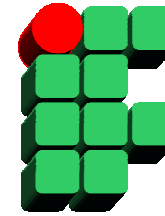
Restaurando Bancos de Dados



- Forked Path
 - Durante restaurações parciais um restore path pode ser quebrado (forked path), gerando mais de uma alternativa de restore.



Restaurando um Banco de Dados



- Execução de um restore path:
 - FULL + LOG, a partir do device dv_backup01:

Full

```
RESTORE DATABASE AdventureWorks  
FROM dv_backup01  
WITH FILE = 8, NORECOVERY;
```

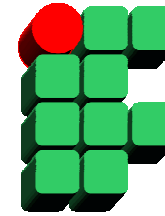
Log 1

```
RESTORE LOG AdventureWorks  
FROM dv_backup01  
WITH FILE = 9, NORECOVERY;
```

Log 2

```
RESTORE LOG AdventureWorks  
FROM dv_backup01  
WITH FILE = 10, RECOVERY;
```

Restaurando um Banco de Dados



- Execução de um restore path:
 - FULL + DIFF + LOG, a partir do arquivo **backup_extra.bak**:

Full

```
RESTORE DATABASE AdventureWorks  
FROM 'D:\SQL\backup_extra.bak'  
WITH FILE = 1, NORECOVERY;
```

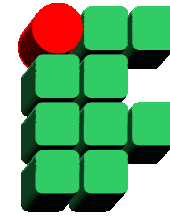
Diff

```
RESTORE DATABASE AdventureWorks  
FROM 'D:\SQL\backup_extra.bak'  
WITH FILE = 2, NORECOVERY;
```

Log

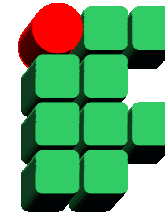
```
RESTORE LOG AdventureWorks  
FROM 'D:\SQL\backup_extra.bak'  
WITH FILE = 3, RECOVERY;
```

Point in Time Restore

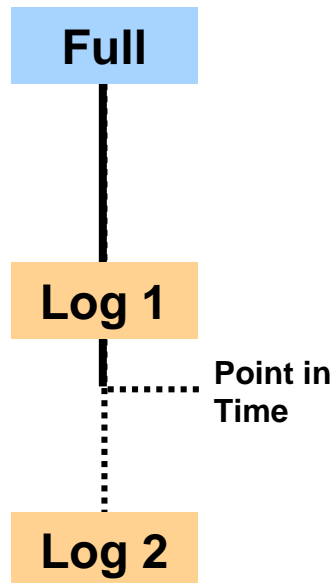


- Permite restaurar o banco de dados até um momento específico no tempo;
- Depende da execução de backups de log para ser realizado;
- Acarreta a formação de forked paths até que o próximo backup seja realizado.

Point in Time Restore



- Exemplo:

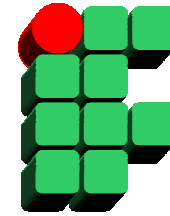


```
RESTORE DATABASE AdventureWorks  
FROM dv_backup01  
WITH STOPAT '2007/06/20 15:34:00', NORECOVERY;
```

```
RESTORE LOG AdventureWorks  
FROM dv_backup01  
WITH STOPAT '2007/06/20 15:34:00', NORECOVERY;
```

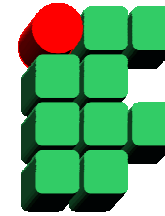
```
RESTORE LOG AdventureWorks  
FROM dv_backup01  
WITH STOPAT '2007/06/20 15:34:00', RECOVERY;
```

Filegroup Restore



- Permite a execução de uma parte de um banco de dados (primary filegroup + filegroups específicos);
- Os filegroups não restaurados ficam indisponíveis;
- A consistência entre os filegroups é mantida através da restauração de backups de log, assim, para bancos de dados com filegroups read-write, o modo de recuperação escolhido precisa ser FULL ou BULK-LOGGED.

Filegroups Restore



```
RESTORE DATABASE AdventureWorks  
FILEGROUP = FG1  
FROM dv_backup_fg  
WITH NORECOVERY;
```

```
RESTORE DATABASE AdventureWorks  
FILEGROUP = FG2  
FROM dv_backup_fg  
WITH NORECOVERY;
```

```
RESTORE LOG AdventureWorks  
FROM dv_backup_fg  
WITH RECOVERY;
```

Primary

FG1

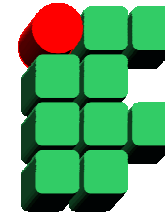
FG2

FG3

FG4



Page Restore



- Utilizado para restaurar páginas de dados corrompidas;
- O ID do arquivo de dados e o ID da página precisam ser fornecidos para execução desse tipo de restore.

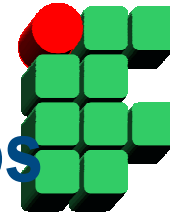
```
RESTORE DATABASE AdventureWorks  
PAGE '1:89' FROM dv_backup  
WITH NORECOVERY;
```

```
RESTORE LOG AdventureWorks  
FROM dv_backup_fg WITH NORECOVERY;
```

```
BACKUP LOG AdventureWorks TO dv_backup_fg;
```

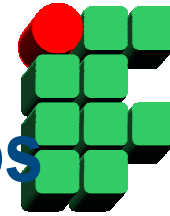
```
RESTORE LOG AdventureWorks  
FROM dv_backup_fg WITH RECOVERY;
```

Desanexando e Anexando Bancos de Dados



- SP_DETACH_DB
 - Desanexa o banco de dados de uma instância, permitindo que seus arquivos sejam copiados ou movidos;
 - As entradas do catálogo (master database) correspondentes ao banco de dados são removidas;
 - O banco de dados precisa estar em modo SINGLE_USER para que o comando possa ser realizado;
 - Caso o banco de dados seja publicado, a replicação precisa ser desativada previamente;
 - Exemplo:
`sp_detach_db AdventureWorks`

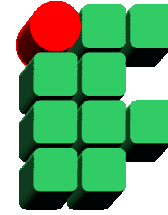
Desanexando e Anexando Bancos de Dados



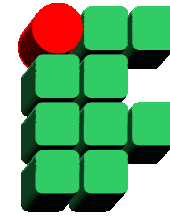
- SP_ATTACH_DB
 - Anexa o banco de dados a uma instância, bloqueando seus arquivos de acesso externo;
 - As entradas do catálogo (master database) correspondentes ao banco de dados são criadas;
 - Permite especificar a localização dos arquivos que compõem o banco de dados;
 - Para anexar bancos de dados com mais de 16 arquivos deve ser utilizado o comando CREATE DATABASE FOR ATTACH;
 - Exemplo:

```
sp_attach_db AdventureWorks,  
            'D:\SQL2008\DATA\AdventureWorks_Data.mdf',  
            'E:\SQL2008\LOG\AdventureWorks_Log.ldf'
```

Gerando Scripts do Banco de Dados

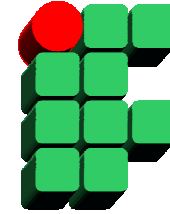


- A geração de scripts de um banco de dados permite recriar objetos como tabelas, constraints, índices, procedures, funções, triggers, assim como as permissões associadas;
- Os scripts não incluem a geração dos dados, mas servem de backup das estruturas do banco de dados.



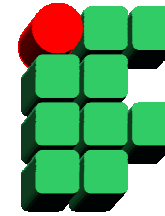
Trabalhando com Bancos de Dados de Sistema

A importância dos BDs de sistema



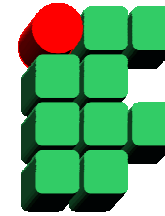
- Diversas operações executadas nos bancos de dados de usuário não pode ser executadas nos bancos de dados de sistema:
 - Adicionar arquivos ou filegroups;
 - Mudar o collation;
 - Mudar o database owner (dbo);
 - Criar full-text catalogs ou full-text indexes;
 - Criar triggers em tabelas de sistema;
 - Excluir o banco de dados;
 - Excluir o guest user;
 - Participar de espelhamento (database mirroring);
 - Mudar o status do banco para OFFLINE;
 - Mudar o status do primary filegroup para READ_ONLY.
- Os bancos de dados de sistema precisam seguir procedimentos especiais para operações de backup e restore, ou para serem movidos de lugar.

Backup e Restore do master database



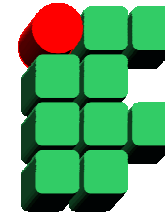
- Uma instância do SQL Server 2008 depende do master database para funcionar, logo, se o master database for comprometido, a instância toda estará comprometida;
- O master database precisa de backup sempre que alguma configuração referente à instância for modificada (criação/alteração/exclusão de bancos de dados, criação/alteração/exclusão de usuários, execução da procedure `sp_configure`, etc);
- Ao restaurar o master database todas as modificações realizadas desde o último backup serão perdidas e precisarão ser refeitas;
- Para restaurar o master database, a instância precisa ser iniciada em modo single-user (parâmetro de inicialização `-m` no Configuration Manager);
- Se o master database estiver corrompido não houver nenhum backup disponível, é necessário realizar um procedimento de rebuild.

Movendo o master database



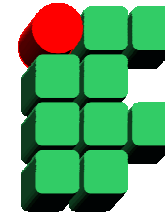
- Para mover os arquivos que compõem o master database, siga os passos abaixo:
 - Verifique a localização dos arquivos do master database no catálogo (sys.master_files);
 - Ajuste os parâmetros de inicialização -d (arquivo de dados), -l (arquivo de log) e -e (log de erros), através do SQL Server Configuration Manager, indicando a nova localização;
 - Pare instância do SQL Server 2008;
 - Mova os arquivos para a nova localização;
 - Inicie a instância em modo master-only;
 - Ajuste o caminho dos arquivos do resource database para a nova localização;
 - Mova os arquivos do resource database para a nova localização;
 - Mude o status do resource database para READ_ONLY;
 - Reinicie a instância do SQL Server 2008;
 - Verifique a nova localização dos arquivos do master e do resource database no catálogo (sys.master_files);
 - Realize um backup do master database.

Backup e Restore do msdb e do model database



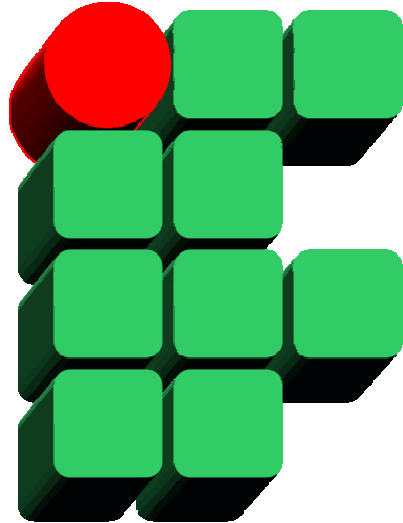
- Um backup periódico do msdb database é recomendado para garantir a continuidade dos seguintes recursos:
 - Tarefas agendadas, alertas e operadores;
 - Replicação;
 - Históricos de backup e restore;
 - Pacotes do SQL Server Integration Service (SSIS).
- O model database é alterado somente pelo usuário, logo, pode ter um processo de backup manual;
- O processo de restore do msdb e do model database é o mesmo que o de um banco de dados de usuário. Para o msdb database, o SQL Server Agent deve ser desativado antes da execução do restore;
- Sempre que o master database passar por um rebuild, o msdb e o model ficarão inaccessíveis e precisam ser restaurados.

Movendo o msdb, o model ou o tempdb



- Para mover os arquivos que compõem o msdb, o model ou o tempdb, siga os passos abaixo:
 - Verifique a localização dos arquivos do banco de dados no catálogo (sys.master_files);
 - Para cada arquivo a ser movido, execute o comando abaixo, indicando a nova localização:

```
ALTER DATABASE database_name  
  MODIFY FILE ( NAME = logical_name , FILENAME = 'new_path/os_file_name' );
```
 - Pare instância do SQL Server 2008;
 - Mova os arquivos para a nova localização;
 - Inicie a instância do SQL Server 2008;
 - Verifique a nova localização dos arquivos do banco de dados no catálogo (sys.master_files);
 - Realize um backup do master database.



Formação de DBAs SQL Server 2008

Parte 5: Backup e Restore