

15. Database Recovery, Backup, and Restoration

Em todo sistema de gerenciamento de banco de dados, sempre existe a possibilidade de falha de hardware ou software. Um SGBD pode ser vítima de falhas sem aviso prévio. Após a ocorrência de uma falha, um SGBD deve ter algum método de recuperação das informações. Esta é uma das principais vantagens que um SGBD tem sobre os antigos sistemas baseados em arquivos que eles substituíram.

O DBMaker incorpora recursos avançados de proteção de dados para evitar perda de dados e tempo de inatividade devido a falhas. Esses recursos permitem que o DBMaker garanta a confiabilidade de um banco de dados e a consistência dos dados, fornecendo funcionalidades de recuperação, backup e restauração.

15.1 Types of Database Failures

As falhas de banco de dados podem ser divididas em dois tipos: falhas de sistema e falhas de mídia. Quando ocorre qualquer um desses tipos de falha, existe a possibilidade de inconsistência ou perda de dados em um banco de dados. Um SGBD deve fornecer recursos para a recuperação de falhas e substituição de um banco de dados danificado por uma cópia de backup.

System Failures

Uma falha de sistema, conhecida como falha de instância, é uma falha da memória principal em um sistema de computador. As falhas de sistema podem ser causadas por uma falha de energia, uma falha de aplicativo ou do sistema operacional, um erro de memória ou outra razão. O resultado é a terminação inesperada do SGBD.

Aplicativos e transações ativas podem terminar de forma anormal quando ocorre uma falha de sistema. Como o estado exato de uma transação em andamento ou de uma transação que não foi completamente escrita no disco não pode ser determinado de forma confiável após uma falha de sistema, esses tipos de transações requerem recuperação. O método mais comum de proteção contra falhas de sistema é o uso de um log de transações ou um arquivo de diário.

Media Failures

A falha de mídia (por exemplo, falha de disco) é uma falha do sistema de armazenamento em disco de um sistema de computador. As falhas de mídia geralmente são causadas por traumas físicos no próprio disco, como um choque de cabeça, incêndio ou exposição a vibração ou forças g fora dos seus limites operacionais físicos.

Nada pode evitar a perda de dados em um disco afetado quando ocorre uma falha de mídia. Um ou mais arquivos podem ser fisicamente danificados devido à falha, exigindo a restauração do banco de dados. No entanto, o banco de dados pode ser restaurado com sucesso se ele fornecer facilidades de backup e restauração

15.2 Recovery from Database Failures

Os objetivos da recuperação após uma falha de banco de dados são garantir que as transações confirmadas sejam refletidas no banco de dados, assegurar que as transações não confirmadas não sejam refletidas no banco de dados e retornar à operação normal o mais rápido possível, isolando os usuários dos problemas causados pela falha.

O DBMaker usa arquivos de diário e pontos de verificação para atingir esses objetivos. Os arquivos de diário e os pontos de verificação trabalham juntos para garantir que todas as transações sejam recuperadas no menor tempo possível, com o mínimo impacto para os usuários.

Journal Files

Os arquivos de diário fornecem um registro histórico em tempo real de todas as alterações feitas em um banco de dados e o status de cada alteração. No caso de uma falha de sistema, o registro histórico de alterações mantido no arquivo de diário permite que o DBMaker recupere e refaça as alterações feitas por transações que foram concluídas, mas não foram escritas no disco, ou desfaça as alterações feitas por transações que terminaram de forma anormal.

Se um banco de dados estiver em modo de backup, os arquivos de diário também armazenarão informações adicionais que o DBMaker pode usar para restauração. Essas informações permanecerão nos arquivos de diário até que sejam feitos backups deles; após o backup dos arquivos de diário, o DBMaker liberará esse espaço para uso por novas transações.

Durante o processo de restauração, o DBMaker adiciona informações dos arquivos de diário de backup a uma cópia de backup do banco de dados. Portanto, apenas os arquivos de diário que contêm alterações feitas no banco de dados entre backups completos precisam de backup.

Checkpoint Events

Um ponto de verificação (checkpoint) é um evento do sistema que traz o banco de dados para um estado limpo. O DBMaker grava todos os registros de diário e todas as páginas de dados sujas de seus buffers de memória interna no disco e recupera blocos de diário que não são mais necessários para fins de backup ou recuperação. O DBMaker pode recuperar blocos de diário que contêm transações não ativas que foram concluídas antes do início da transação ativa mais antiga.

O tempo de inicialização após uma falha de instância é reduzido após a realização de um ponto de verificação. O DBMaker grava o horário do último ponto de verificação e uma lista de todas as transações ativas no momento do ponto de verificação no cabeçalho do arquivo de diário. Durante a recuperação do banco de dados, o DBMaker usa essas informações para determinar quais transações devem ser desfeitas, quais devem ser refeitas e quais devem ser ignoradas.

O DBMaker fará automaticamente um ponto de verificação quando os arquivos de diário estiverem cheios, para tentar recuperar alguns blocos de diário para reutilização. Se o ponto de verificação não puder recuperar espaço suficiente para concluir a transação atual, a transação será abortada. O DBMaker também fará automaticamente um ponto de verificação quando o banco de dados iniciar e desligar, e quando um backup online for realizado.

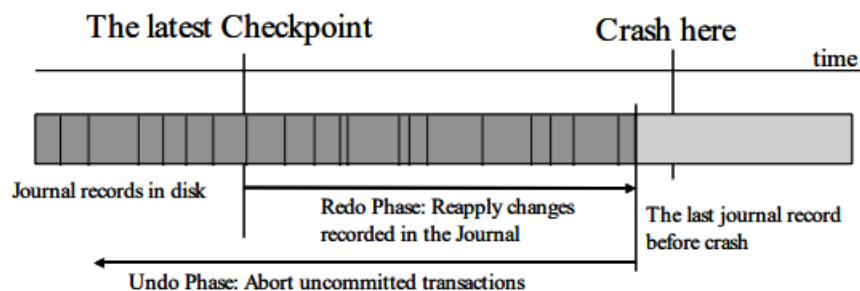
Os administradores de banco de dados podem iniciar manualmente um ponto de verificação executando o comando CHECKPOINT. O intervalo ideal entre dois pontos de verificação depende da frequência de atividade do banco de dados, do tamanho médio das transações e do tamanho e número de arquivos de diário. Como esses fatores podem variar significativamente de banco de dados para banco de dados, o intervalo ideal é melhor determinado através da experiência. Pontos de verificação manuais reduzem o tempo necessário para iniciar, terminar e fazer backup de um banco de dados, bem como a possibilidade de encontrar um diário cheio.

Os pontos de verificação podem exigir uma quantidade significativa de tempo para serem concluídos, dependendo do tamanho e do número de transações desde o último ponto de verificação. Qualquer transação que esteja ativa quando um ponto de verificação ocorrer precisa aguardar que o DBMaker calcule quais registros de diário

ele pode recuperar, mas não precisa aguardar enquanto o DBMaker realmente grava registros de diário e páginas de dados sujas no disco.

Recovery Steps

O DBMaker fornece suporte para recuperação automática quando o banco de dados é iniciado após uma falha de sistema ou quando ocorre um erro durante a inicialização. Durante o processo de recuperação, o DBMaker sempre executa duas etapas separadas: refazer (redo) e desfazer (undo).



O primeiro passo no processo de recuperação é refazer (ou reaplicar) todas as alterações feitas no banco de dados que estão registradas no diário. Este passo é necessário, pois é possível que uma transação tenha sido concluída antes da falha do sistema, sem que todas as alterações feitas pela transação tenham sido escritas no banco de dados. No entanto, essas alterações estão armazenadas no diário e podem ser escritas no banco de dados durante esta etapa. Após este passo, o banco de dados contém as alterações feitas por todas as transações confirmadas e as alterações feitas por todas as transações não confirmadas.

O segundo passo no processo de recuperação desfaz (ou reverte) as alterações feitas por transações que não foram concluídas antes da ocorrência da falha do sistema. Este passo é necessário, pois o estado exato de uma transação em andamento não pode ser determinado de forma confiável em caso de falha de sistema. Essas transações incompletas devem ser removidas, uma vez que uma transação é autossuficiente por definição e deve ou ser concluída com sucesso e alterar os dados, ou falhar e deixar os dados inalterados. Ao final desta etapa, o banco de dados contém as alterações feitas por todas as transações confirmadas, mas não contém nenhuma alteração feita por transações não confirmadas.

O DBMaker também suporta a inicialização de um banco de dados após uma falha de mídia ou após uma falha de sistema, o que causa inconsistências em um banco de dados que não podem ser corrigidas durante o processo de recuperação automática. Nesses casos, o banco de dados falha ao iniciar e normalmente seria necessário restaurar seu banco de dados a partir de um backup. No entanto, se você nunca fez backup do seu banco de dados, pode forçar a inicialização do banco de dados

definindo o modo de inicialização forçada usando a palavra-chave DB_ForcS no arquivo dmconfig.ini. Isso permite iniciar o banco de dados e descarregar os dados não afetados. Para mais informações sobre o modo de inicialização forçada, consulte Forçando a Inicialização do Banco de Dados.

Forcing Database Startup

O DBMaker realiza automaticamente operações de recuperação se erros ocorrerem quando um banco de dados inicia normalmente. Se o banco de dados não conseguir iniciar, pode haver alguns erros no disco. Erros de disco exigem que o banco de dados seja restaurado a partir do backup mais recente para ser corrigido. Se o banco de dados não tiver backups e não conseguir iniciar, use o modo de inicialização forçada fornecido pelo DBMaker.

O DBMaker oferece uma opção de inicialização forçada para essa situação. Para ativar o modo de inicialização forçada, use a palavra-chave DB_ForcS no arquivo dmconfig.ini. Definir essa palavra-chave como 1 ativa o modo de inicialização forçada, e defini-la como 0 desativa-o.

Quando o modo de inicialização forçada está ativado, o DBMaker ignora erros ao iniciar o banco de dados.

Se o banco de dados ainda não conseguir ser iniciado, há uma alternativa restante fornecida no procedimento abaixo. No entanto, antes de realizar este procedimento, faça backup de todos os dados e arquivos de diário.

Para iniciar um banco de dados quando ele não iniciar no modo de inicialização forçada:

1. Defina o Modo de Inicialização Forçada como desativado no dmconfig.ini (DB_ForcS = 0).
2. Defina o Modo de Inicialização para Novo Modo de Diário no dmconfig.ini (DB_SMode = 2).
3. Reinicie o banco de dados.
4. Redefina o Modo de Inicialização de volta ao normal no dmconfig.ini (DB_SMode = 1).

O DBMaker fornece a opção de usar um novo diário para forçar a inicialização do banco de dados sem quaisquer operações de recuperação. Portanto, se ocorrerem erros graves o suficiente para impedir que o banco de dados inicie, o banco de dados pode estar em um estado inconsistente.

Após iniciar o banco de dados com este método, verifique a consistência do banco de dados. Para mais informações sobre a verificação de consistência do banco de dados, consulte a seção 6.12 Verificação da Consistência do Banco de Dados.

15.3 Types of Backups

Backups são utilizados para proteger um banco de dados contra falhas de mídia ou outros erros de mídia. Após uma falha de mídia, um ou mais arquivos de banco de dados podem estar danificados e inutilizáveis. Use o backup mais recente para substituir os arquivos danificados e reconstruir o banco de dados.

Os backups de banco de dados consistem em sequências de backup. Uma sequência de backup inclui um backup completo, todos os backups diferenciais associados ao backup completo e os backups incrementais realizados após o backup completo.

Full Backups

Um backup completo é qualquer backup que cria uma cópia de todos os arquivos de dados e de diário, fornecendo uma cópia completa do banco de dados em um ponto no tempo. Um backup do arquivo `dmconfig.ini` também pode ser criado, preservando qualquer configuração personalizada que possa existir para o banco de dados. O administrador de banco de dados pode realizar um backup completo enquanto o banco de dados está online ou offline.

Backups completos arquivam todo o banco de dados, exigindo, portanto, uma grande quantidade de espaço de armazenamento. No entanto, um banco de dados pode ser restaurado relativamente rapidamente usando um backup completo. Um backup completo pode ser usado para restaurar um banco de dados ao ponto no tempo em que o backup completo foi realizado.

Um backup completo válido será atribuído a um ID de backup completo. O ID de backup completo é um carimbo de data e hora. O ID de backup garante que backups completos, diferenciais e incrementais sejam corretamente associados em uma sequência de backup. Observe que um backup diferencial registra apenas os dados que mudaram desde o backup completo mais recente. A base diferencial é necessária ao restaurar a partir de um backup diferencial. Um backup diferencial sozinho é insuficiente para reconstruir um banco de dados. Todos os backups incrementais entre o backup completo válido atual e o próximo pertencem apenas ao backup completo válido atual. Tentar restaurar backups incrementais contra sequências anteriores (e quaisquer outras) falhará. As sequências de backup são gerenciadas pelo DBMaker.

A reparação de um banco de dados, a restauração de um banco de dados, o início de um banco de dados no novo modo de diário ou a mudança do modo de backup exigirão um novo backup completo válido. Existem três métodos principais para realizar backups completos. O primeiro é através do servidor de backup, e é discutido com mais detalhes na seção 15.6 Servidor de Backup. Backups completos pelo servidor de backup podem ser realizados com o dmSQL ou com a utilidade JServer Manager. O segundo método utiliza um backup completo interativo. O backup completo interativo não exige que o Servidor de Backup esteja iniciado. O JServer Manager é o método recomendado para realizar esse tipo de backup completo. Para orientações sobre como realizar um backup completo interativamente, consulte o Guia do Usuário do JServer Manager. O terceiro método para realizar um backup completo é o backup completo offline. Consulte a seção 15.5, Backups Completos Offline, para mais informações.

Differential Backups

Backups diferenciais ajudam a economizar tempo e espaço em disco. Ao contrário de um backup completo que simplesmente copia todos os arquivos, backups diferenciais usam uma abordagem diferente.

Um backup diferencial é baseado no backup completo mais recente. Esse backup completo é a base do diferencial. Uma base diferencial deve existir antes que um backup diferencial seja criado.

Backups diferenciais contêm apenas os dados que mudaram desde o momento em que o backup base diferencial foi criado. Uma única base diferencial é tipicamente usada para vários backups diferenciais sucessivos. Posteriormente, se for necessário restaurar o banco de dados, tanto o backup completo (o backup base diferencial) quanto o backup diferencial serão necessários.

Arquivos de dados (todos os arquivos DB e BB) estão incluídos durante os backups diferenciais. Arquivos de diário, devido às suas frequentes mudanças, são copiados apenas os blocos de diário úteis.

Um backup diferencial registra apenas os dados que mudaram desde o backup completo mais recente. Isso permite que os usuários realizem backups mais frequentes, pois os backups diferenciais são menores que os completos. Realizar backups frequentes diminui o risco de perda de dados. Considere usar backups diferenciais quando:

- Apenas uma parte relativamente pequena do banco de dados mudou desde o último backup completo. Backups diferenciais são particularmente eficazes quando os mesmos dados são modificados muitas vezes.

- Backups frequentes são desejados, mas não backups completos frequentes.
- Minimizar o tempo de avanço do diário de transações ao restaurar um banco de dados.

Existem dois métodos para realizar backups diferenciais. O primeiro utiliza o servidor de backup. A configuração de várias palavras-chave é necessária antes de iniciar o banco de dados. Veja a Seção 15.6, Servidor de Backup. O segundo método é chamado de backups diferenciais online. A utilidade JServer Manager é recomendada para esse método, e os detalhes estão explicados no Guia do Usuário do JServer Manager.

Incremental Backups

Um backup incremental é qualquer backup que cria uma cópia apenas dos arquivos de diário que foram alterados desde o último backup incremental, diferencial ou completo. Backups incrementais só podem ser realizados após um backup completo ou diferencial ter sido realizado. A realização de um novo backup completo inicia uma sequência de backup. Backups incrementais subsequentes são parte dessa sequência e não podem ser usados com qualquer outro backup completo ou diferencial. Observe que um backup incremental é composto por arquivos de diário que registram todas as transações desde que o modo de backup (DB_BMode) esteja ativado. Quando o Banco de Dados DBMaker está operando no modo normal (DB_SMode = 1), antes de realizar um backup incremental, deve ter sido feito um backup completo ou diferencial; quando no modo de replicação (DB_SMode = 4), um backup incremental pode ser feito sem um backup completo ou diferencial. A sequência de arquivos de backup incremental fornece uma cópia das alterações feitas no banco de dados desde o último backup completo. O administrador do banco de dados pode realizar um backup incremental apenas enquanto o banco de dados está online.

Backups incrementais arquivam apenas arquivos de diário, portanto, exigem apenas uma pequena quantidade de espaço de armazenamento. No entanto, pode levar mais tempo do que um backup completo para restaurar um banco de dados, já que o DBMS deve reverter todas as transações nos arquivos de diário de backup. Use backups incrementais junto com um backup completo e diferencial para restaurar um banco de dados a qualquer ponto no tempo entre o backup completo ou diferencial anterior e o momento em que o último backup incremental foi concluído.

Existem dois métodos para realizar backups incrementais (há um terceiro método, Backup incremental para o atual, que é considerado um tipo diferente de backup). O primeiro método é o backup incremental por servidor de backup. O servidor de backup deve estar iniciado no banco de dados para usar esse método. Para mais

informações sobre como realizar um backup incremental por servidor de backup, consulte a seção 15.6 Servidor de Backup. O segundo método é o backup incremental interativo. Esse tipo de backup incremental não exige que o servidor de backup esteja iniciado. O método recomendado para realizar backup incremental interativo é com a utilidade JServer Manager. O backup incremental interativo é explicado no Guia do Usuário do JServer Manager.

Offline Backups

Um backup offline é qualquer backup que deve ser realizado após um banco de dados ter sido desligado. O administrador do banco de dados deve agendar um horário para desligar o banco de dados e notificar todos os usuários para que eles possam se desconectar do banco de dados. Backups offline podem ser inconvenientes para os usuários, uma vez que eles devem lembrar de concluir todas as transações ativas e se desconectar do banco de dados antes que ele seja desligado. O administrador do banco de dados pode realizar apenas backups completos enquanto o banco de dados está offline.

Um SGBD não precisa fornecer a capacidade de fazer backup de um banco de dados offline, pois você pode fazer o backup do banco de dados com comandos do sistema operacional após ele ser desligado. O administrador do banco de dados pode realizar um backup offline usando esse método, ou utilizando o JServer Manager, uma ferramenta gráfica fácil de usar que realiza backups offline sem recorrer a comandos do sistema operacional.

Online Backups

Um backup online é qualquer backup realizado enquanto um banco de dados está em execução. O administrador do banco de dados não precisa desligar o banco de dados, e os usuários não precisam se desconectar. Backups online são mais convenientes para os usuários, já que nenhuma ação é necessária da parte deles. O administrador do banco de dados pode realizar backups completos, diferenciais e incrementais enquanto o banco de dados está online.

Um SGBD deve fornecer a capacidade de realizar backups online, uma vez que o banco de dados ainda está em operação e com usuários conectados. O DBMaker permite que backups online sejam realizados manualmente usando o dmSQL e comandos do sistema operacional, mas também oferece o JServer Manager, uma ferramenta gráfica fácil de usar que permite realizar backups online sem recorrer a comandos do sistema operacional.

Online Incremental to Current Backups

O DBMaker também suporta um tipo adicional de backup chamado incremental online até o atual.

A diferença entre um backup incremental online e um backup incremental online até o atual em um banco de dados com vários arquivos de journal é pequena, mas importante. Em um backup incremental online, o DBMaker fará o backup de todos os arquivos de journal que foram usados desde o último backup, excluindo o arquivo de journal ativo. Em um backup incremental online até o atual, o DBMaker fará o backup de todos os arquivos de journal que foram usados, incluindo o arquivo de journal ativo. Isso significa que um backup incremental online pode restaurar um banco de dados até o ponto no tempo em que a última transação confirmada foi gravada no último arquivo de journal completo, enquanto um backup incremental online até o atual pode restaurar um banco de dados até o ponto no tempo em que o arquivo de journal ativo foi respaldado.

Em um banco de dados com apenas um único arquivo de journal, um backup incremental online e um backup incremental online até o atual são iguais. Nesse caso, o único arquivo de journal é o arquivo de journal ativo. O DBMaker fará o backup desse único arquivo de journal em ambos os tipos de backup incremental.

15.4 Backup Modes

O modo de backup determina se o DBMaker pode realizar backups incrementais online e o tipo de dados que serão incluídos durante um backup incremental. Ele também define quando o DBMaker liberará blocos de journal que pertencem a transações inativas para uso por outras transações. O DBMaker possui três modos de backup: **NONBACKUP**, **BACKUP-DATA** e **BACKUP-DATA-AND-BLOB**.

Backup Mode	TableSpace Backup Mode	User Defined TableSpace (DATA)	User Defined TableSpace (BLOB)	System TabeSpace (DATA and BLOD)
No Bacukp		No	No	No
Backup Data		Yes	No	Yes

Backup Data and Blob	Backup BLOB Off	Yes	No	No
	Backup BLOB On	— Yes	Yes	Yes

NONBACKUP Mode

O modo **NONBACKUP** não oferece proteção para dados que foram inseridos ou atualizados desde o último backup completo. Nesse modo, backups incrementais online não podem ser realizados. Um banco de dados pode usar o journal para se recuperar completamente de uma falha de instância, mas uma falha de mídia pode resultar na perda de dados. Blocos de journal que não estão sendo usados por uma transação ativa podem ser reutilizados imediatamente após um checkpoint, mas uma vez que são sobrescritos, o banco de dados só pode ser restaurado até o ponto no tempo do último backup completo.

BACKUP-DATA Mode

O modo **BACKUP-DATA** oferece proteção para dados (excluindo dados BLOB) que foram inseridos ou atualizados desde o último backup completo. Nesse modo, um administrador de banco de dados pode realizar um backup incremental online, mas apenas dados que não são BLOB serão armazenados nos arquivos de backup. Um banco de dados pode usar o journal para se recuperar completamente de uma falha de instância e pode se recuperar parcialmente de uma falha de mídia. Embora o último backup possa ser usado para restaurar o banco de dados até o ponto no tempo da falha de mídia, quaisquer alterações nos dados BLOB serão perdidas. Blocos de journal que não estão em uso por uma transação ativa só podem ser reutilizados após a realização de um checkpoint e o backup do arquivo de journal.

BACKUP-DATA-AND-BLOB Mode

O modo **BACKUP-DATA-AND-BLOB** fornece proteção para todos os dados (incluindo dados BLOB) que foram inseridos ou atualizados desde o último backup completo. Nesse modo, um administrador de banco de dados pode realizar um backup incremental online, e todos os dados serão armazenados nos arquivos de backup. O banco de dados pode usar o journal para se recuperar completamente de uma falha de instância e se recuperar totalmente de uma falha de disco. O último backup pode ser usado para restaurar o banco de dados completamente até o ponto no tempo da falha de mídia, incluindo todos os dados BLOB. Blocos de journal que não estão em

uso por uma transação ativa só podem ser reutilizados após a realização de um checkpoint e o backup do arquivo de journal.

Tablespace BLOB Backup Mode

Isso pode dificultar para o administrador do banco de dados decidir qual modo de backup deve escolher. Para evitar que esse tipo de situação ocorra, o DBMaker permite que o administrador do banco de dados modifique o modo de backup do banco de dados para tabelas individuais ao criá-las.

Para fazer backup dos dados BLOB em um tablespace específico, utilize a opção **BACKUP BLOB ON** ao executar o comando **CREATE TABLESPACE**. Para evitar o backup de dados BLOB em um tablespace específico, utilize a opção **BACKUP BLOB OFF** ao executar o comando **CREATE TABLESPACE**.

O modo de backup de cada tablespace dependerá então da combinação do modo de backup do banco de dados e do modo de backup do tablespace da seguinte forma:

- Se o banco de dados estiver em modo **BACKUP-DATA-AND-BLOB** e um tablespace foi criado com a opção **BACKUP BLOB ON**, o DBMaker fará o backup dos dados BLOB nesse tablespace.
- Se o banco de dados estiver em modo **BACKUP-DATA-AND-BLOB** e um tablespace foi criado com a opção **BACKUP BLOB OFF**, o DBMaker não fará o backup dos dados BLOB nesse tablespace.
- Se o banco de dados estiver em modo **BACKUP-DATA**, o DBMaker não fará o backup dos dados BLOB, independentemente de o tablespace ter sido criado com a opção **BACKUP BLOB ON** ou **BACKUP BLOB OFF**.

O DBMaker usa o modo **BACKUP BLOB ON** por padrão para os tablespaces recém-criados. Todas as alterações nos dados BLOB nesse tablespace serão registradas no arquivo de journal quando o banco de dados estiver em modo **BACKUP-DATA-AND-BLOB**.

NOTA: Um novo backup completo é necessário após um tablespace ser alterado para somente leitura, pois os arquivos de dados em tablespaces somente leitura não são incluídos durante os backups diferenciais.

Backup Mode of File Objects

Além de fazer backup dos dados regulares e BLOB no banco de dados, os usuários podem optar por fazer backup de objetos de arquivo. Objetos de arquivo são incluídos apenas durante backups completos automáticos iniciados pelo daemon de backup. Os usuários devem primeiro iniciar o servidor de backup, definir o cronograma de

backup completo e configurar o diretório de backup. Para mais informações sobre as configurações de backup completo, consulte a seção 15.6, Servidor de Backup.

Existem dois tipos de objetos de arquivo: objetos de arquivo do usuário e objetos de arquivo do sistema. O administrador do banco de dados pode optar por fazer backup apenas dos objetos de arquivo do sistema, tanto de objetos de arquivo do sistema quanto do usuário, ou de nenhum deles. A palavra-chave DB_BkFoM no arquivo dmconfig.ini especifica o Modo de Backup dos Objetos de Arquivo.

- DB_BkFoM = 0: Não fazer backup de objetos de arquivo
- DB_BkFoM = 1: Fazer backup apenas de objetos de arquivo do sistema
- DB_BkFoM = 2: Fazer backup de objetos de arquivo do sistema e do usuário

Ao fazer backup de objetos de arquivo (DB_BkFoM = 1, 2), o servidor de backup copia todos os arquivos externos dos objetos de arquivo para o subdiretório "FO" sob o diretório especificado pela palavra-chave DB_BkDir. O cronograma segue o cronograma de backup completo especificado por DB_FBkTm e DB_FBkTv.

Exemplo: Um trecho de um arquivo dmconfig.ini contendo palavras-chave relacionadas é mostrado abaixo:

```
[MyDB]
DB_BkSvr = 1 ; starts the backup server
DB_FBkTm = 01/05/01 00:00:00 ; begins from midnight at May 1, 2001.
DB_FBkTv = 1-00:00:00 ; interval is every one day.
DB_BkDir = /home/dbmaker/backup ; backup directory
DB_BkFoM = 2 ; backup both system and user file objects
```

Como o Modo de Backup de Objetos de Arquivo está definido como 2, o servidor de backup copiará todos os objetos de arquivo externos do banco de dados para o diretório "/home/dbmaker/backup/FO". Se o subdiretório FO não existir, o servidor de backup o criará.

Os arquivos no subdiretório FO são renomeados com um número sequencial. Por exemplo, se o nome do arquivo externo original é "/DBMaker/mydb/FO/ZZ000123.bmp", o servidor de backup o copiará para o subdiretório FO e o renomeará para 'fo0000000344.bak', indicando que é o 344º objeto de arquivo. O mapeamento entre o nome do arquivo completo original e seu novo nome é registrado no arquivo de lista de mapeamento de objetos de arquivo, dmFoMap.his. Para mais informações sobre o arquivo de lista de mapeamento de objetos de arquivo, consulte a seção 15.7, Arquivos de Histórico de Backup.

O servidor de backup também moverá a versão anterior dos objetos de arquivo para o subdiretório FO sob o diretório de backup antigo especificado por DB_BkOdr.

Os administradores de banco de dados devem considerar que habilitar o backup de objetos de arquivo requer mais tempo para um backup completo. O custo de um backup completo inclui (1) copiar o backup completo anterior se DB_BkOdr estiver definido; (2) copiar todos os arquivos do banco de dados; (3) copiar todos os arquivos de journal; e (4) copiar todos os objetos de arquivo se DB_BkFoM estiver definido. Além disso, é necessário garantir que haja espaço suficiente no diretório de backup especificado por DB_BkDir para todos os arquivos de backup, a fim de evitar falhas no backup.

Backup Mode of Stored Procedures

Existem três tipos de procedimentos armazenados: procedimentos armazenados SQL, procedimentos armazenados ESQL e procedimentos armazenados JAVA. Como os códigos-fonte são escritos em um banco de dados, os procedimentos armazenados SQL são incluídos no backup como dados regulares durante um backup completo. Além de fazer backup dos dados regulares, dados BLOB e objetos de arquivo no banco de dados, os usuários podem optar por fazer backup de procedimentos armazenados ESQL ou procedimentos armazenados JAVA. Esses dois tipos de procedimentos armazenados são incluídos apenas durante backups completos automáticos iniciados pelo daemon de backup. Os usuários devem primeiro iniciar o Backup Server, definir o cronograma de backup completo e configurar o diretório de backup. Para mais informações sobre as configurações de um backup completo, consulte a seção 15.6, Backup Server.

Os usuários podem especificar o modo de backup dos procedimentos armazenados definindo o parâmetro DB_BkSPm:

- DB_BkSPm = 0: Não fazer backup de procedimentos armazenados ESQL e procedimentos armazenados JAVA
- DB_BkSPm = 1: Fazer backup de procedimentos armazenados ESQL e procedimentos armazenados JAVA

Exemplo:

Um trecho de um arquivo dmconfig.ini contendo palavras-chave relacionadas é mostrado abaixo:

```
[MyDB]
DB_BkSvr = 1 ; starts the backup server
DB_FBkTm = 14/05/01 00:00:00 ; begins from midnight at May 1, 2014
DB_FBkTv = 1-00:00:00 ; interval is every one day
DB_BkDir = /home/dbmaker/backup ; backup directory
```

DB_BkSPm = 1 ; backup ESQL stored procedures and JAVA
stored procedures

O diretório padrão dos arquivos de backup dos dois tipos de procedimentos armazenados é o subdiretório denominado SP dentro do diretório especificado pela palavra-chave DB_BkDir. Se os usuários configuraram a palavra-chave DB_BkOdr no arquivo dmconfig.ini, durante o processo de realização de um backup completo, as sequências de backup anteriores dos dois tipos de procedimentos armazenados serão movidas para o subdiretório também denominado SP dentro do antigo diretório de backup especificado por DB_BkOdr, e, em seguida, essas sequências de backup serão excluídas do diretório padrão de procedimentos armazenados.

Durante o processo de backup dos dois tipos de procedimentos armazenados, o Backup Server primeiro gera um arquivo chamado dmSpBk.his. Em seguida, o Backup Server copia os arquivos dos procedimentos armazenados e, ao mesmo tempo, arquivos com a extensão .s0 e .b0, usados para carregar os procedimentos armazenados copiados, são criados. Para os procedimentos armazenados ESQL, os arquivos que precisam ser copiados são os arquivos fonte e os arquivos objeto; para os procedimentos armazenados JAVA, apenas os arquivos objeto precisam ser copiados. Para facilitar a reconstrução dos procedimentos armazenados, os arquivos fonte dos procedimentos armazenados ESQL serão renomeados no formato snameowner.ec. Por exemplo, se o nome do procedimento armazenado ESQL for y1 e for de propriedade de SYSADM, os arquivos fonte copiados serão renomeados como y1SYSADM.ec.

O arquivo dmSpBk.his é utilizado para registrar informações de backup. Para mais informações sobre o arquivo de lista de backup dos procedimentos armazenados, consulte a seção 15.7, Arquivos de Histórico de Backup.

Os administradores de banco de dados devem considerar que a ativação do backup de objetos de arquivo exige mais tempo para um backup completo. O custo de um backup completo inclui (1) copiar o backup completo anterior, se DB_BkOdr estiver configurado; (2) copiar todos os arquivos do banco de dados; (3) copiar todos os arquivos de journal; (4) copiar todos os objetos de arquivo, se DB_BkFoM estiver configurado; (5) copiar os procedimentos armazenados ESQL e JAVA, se DB_BkSPm estiver configurado. Além disso, assegure-se de que haja espaço suficiente em disco no diretório de backup especificado por DB_BkDir para todos os arquivos de backup a fim de evitar falhas no backup.

Compressing Backup Files

Os arquivos de banco de dados podem se tornar muito grandes e uma grande quantidade de espaço livre é necessária para armazenar os arquivos de backup. O DBMaker agora suporta a compressão de arquivos de backup. Para ativar ou desativar esse recurso, você pode definir a palavra-chave `DB_BkZip` no arquivo `dmconfig.ini` ou alterar `BKZIP` com o procedimento armazenado do sistema `SetSystemOption` enquanto o banco de dados estiver em execução.

- **DB_BkZip = 1:** Comprime os arquivos de backup.
- **DB_BkZip = 0:** Os arquivos de backup não são comprimidos (padrão).

O formato de compressão é GZIP, portanto, você pode usar qualquer ferramenta compatível com GZIP para ler o arquivo comprimido.

NOTA: Arquivos FO não são comprimidos, mesmo que você defina `DB_BkZip` para habilitar a compressão de arquivos de backup.

Setting the Backup Mode

O DBMaker oferece vários métodos para definir o modo de backup. O método que você escolher depende de se o seu banco de dados está online ou offline, e se você se sente mais confortável editando o arquivo de configuração diretamente, usando a ferramenta de linha de comando `dmSQL`, ou usando a ferramenta gráfica `JServer Manager`.

Modificar o modo de backup de um banco de dados para fornecer um nível mais alto de proteção de backup (por exemplo, de `NONBACKUP` para `BACKUP-DATA`, ou de `BACKUP-DATA` para `BACKUP-DATA-AND-BLOB`) afeta o uso do journal. O journal começa a registrar mudanças em dados que não eram registrados antes da modificação do modo de backup. Como resultado, é necessário realizar um backup completo ou um backup diferencial quando você muda o modo de backup. Isso fornece um ponto de partida para os arquivos de journal de backup atualizarem durante o processo de restauração.

Não são necessárias etapas extras ao modificar o modo de backup de um banco de dados para fornecer um nível mais baixo de proteção de backup (por exemplo, de `BACKUP-DATA` ou `BACKUP-DATA-AND-BLOB` para `NONBACKUP`), já que o journal simplesmente para de registrar mudanças em dados. O DBMaker usará o backup completo ou diferencial anterior como ponto de partida para os arquivos de journal de backup atualizarem durante o processo de restauração. No entanto, algumas mudanças nos dados podem ser perdidas se o banco de dados for restaurado após a mudança para um nível mais baixo de proteção de backup.

O administrador do banco de dados pode alterar o modo de backup do banco de dados enquanto estiver offline usando o arquivo dmconfig.ini ou o JServer Manager. Como o modo de backup afeta o uso do journal, um backup completo offline deve ser realizado antes de iniciar o banco de dados com a nova configuração de modo de backup. Os modos de backup podem ser alterados de um modo para outro sem restrições quando offline, desde que um backup completo seja feito ao passar de um nível mais baixo de proteção de backup para um nível mais alto. Para mais informações sobre a realização de um backup completo offline, consulte "Backups Completos Offline" mais adiante neste capítulo.

Um administrador de banco de dados pode alterar o modo de backup de um banco de dados online usando o dmSQL. Como o modo de backup afetará o uso do journal, o modo de backup deve ser alterado de um nível mais baixo de proteção de backup para um nível mais alto (por exemplo, de NONBACKUP para BACKUP-DATA, ou de BACKUP-DATA para BACKUP-DATA-AND-BLOB) entre o início e o término de um período de backup completo. Durante o tempo de execução, os usuários não podem alterar diretamente o modo de backup de NONBACKUP para BACKUP-DATA-AND-BLOB, ou de BACKUP-DATA-AND-BLOB para BACKUP-DATA. No entanto, os usuários podem alterar o modo de backup de BACKUP-DATA-AND-BLOB para NONBACKUP a qualquer momento.

Exemplo

Para usar o dmSQL para alterar o modo de backup online:

```
dmSQL> BEGIN BACKUP;  
dmSQL> SET DATA BACKUP ON;  
dmSQL> END BACKUP DATAFILE;  
dmSQL> END BACKUP JOURNAL;
```

ou:

```
dmSQL> BEGIN BACKUP;  
dmSQL> END BACKUP DATAFILE;  
dmSQL> SET DATA BACKUP ON;  
dmSQL> END BACKUP JOURNAL;
```

DBMaker não permite que o banco de dados passe de um nível mais alto de proteção de backup para um nível mais baixo, a menos que primeiro seja alterado para o modo NONBACKUP. Para mudar de BACKUP-DATA-AND-BLOB para BACKUP-DATA, primeiro altere para o modo NONBACKUP e depois siga as regras acima para mudar

de um nível inferior de proteção de backup para um nível superior. O modo de backup pode ser alterado de BACKUP-DATA-AND-BLOB ou BACKUP-DATA para NONBACKUP a qualquer momento; não é necessário que isso seja feito entre o início e o término de um período de backup completo.

USING THE DMCONFIG.INI CONFIGURATION FILE

Se o banco de dados estiver offline, altere o modo de backup diretamente usando a palavra-chave DB_BMode no arquivo dmconfig.ini. Na próxima vez que o banco de dados for iniciado, o novo modo de backup será usado. Se o banco de dados estiver online, alterar o valor da palavra-chave DB_BMode não terá efeito até que o banco de dados seja desligado e reiniciado. Lembre-se de realizar um backup completo offline se o modo de backup for alterado de NONBACKUP para BACKUP-DATA ou BACKUP-DATA-AND-BLOB ou de BACKUP-DATA para BACKUP-DATA-AND-BLOB. Para definir o modo de backup usando o arquivo dmconfig.ini:

1. Abra o arquivo dmconfig.ini no servidor de banco de dados usando qualquer editor de texto ASCII.
2. Localize a seção de configuração do banco de dados.
3. Altere o valor da palavra-chave DB_BMode para um dos seguintes valores:

```
0 - NONBACKUP mode
1 - BACKUP-DATA mode
2 - BACKUP-DATA-AND-BLOB mode
```

4. Reinicie o banco de dados para começar a usar o novo modo de backup.

Se a palavra-chave DB_BMode não estiver presente na seção de configuração do banco de dados, você terá que adicioná-la à seção de configuração do banco de dados. Você pode adicionar a palavra-chave em uma linha separada em qualquer lugar entre o início da seção de configuração do banco de dados e o início da próxima seção de configuração; a ordem em que as palavras-chave aparecem não é importante. Se você não especificar um valor para DB_BMode, o valor padrão de 0 (modo NONBACKUP) será usado.

USING DMSQL

Se o banco de dados estiver online e você estiver confortável usando a linha de comando do dmSQL, você pode alterar o modo de backup usando o comando SQL SET. Você deve executar este comando durante um backup completo ou diferencial online. O novo modo de backup será habilitado assim que o comando for executado.

Para definir o modo de backup usando a linha de comando do dmSQL:

1. Conecte-se ao banco de dados para alterar o modo de backup usando o dmSQL.
2. Inicie um backup completo online usando o comando BEGIN BACKUP.
3. Altere o modo de backup durante o período de backup completo emitindo um dos seguintes comandos SET no prompt do dmSQL:

```
dmSQL> SET BACKUP OFF;  
dmSQL> SET DATA BACKUP ON;  
dmSQL> SET BLOB BACKUP ON;
```

1. Complete o backup completo online.

O comando SET BACKUP OFF corresponde ao modo NONBACKUP, o comando SET DATA BACKUP ON corresponde ao modo BACKUP-DATA, e o comando SET BLOB BACKUP ON corresponde ao modo BACKUP-DATA-AND-BLOB.

USING JSERVER MANAGER

Se o banco de dados estiver offline, você pode mudar o modo de backup usando a ferramenta gráfica JServer Manager. O JServer Manager mudará automaticamente o valor da palavra-chave DB_BMode no arquivo dmconfig.ini. Na próxima vez que você iniciar o banco de dados, o novo modo de backup será utilizado. Se o banco de dados estiver online, mudar o valor da palavra-chave DB_BMode não terá efeito até que o banco de dados seja desligado e reiniciado. Você deve lembrar de realizar um backup completo offline se estiver mudando do modo NONBACKUP para o modo BACKUP-DATA ou BACKUP-DATA-AND-BLOB ou do modo BACKUP-DATA para o modo BACKUP-DATA-AND-BLOB. Para obter instruções sobre como definir o modo de backup offline usando a ferramenta gráfica JServer Manager, consulte o Guia do Usuário do JServer Manager.

15.5 Offline Full Backups

Os backups completos offline utilizam comandos do sistema operacional para fazer backup do banco de dados. Embora o DBMaker ofereça essa opção, o uso do servidor de backup é recomendado. Backups completos offline exigem que o banco de dados seja desligado; além disso, gerenciar a sequência de backup é um processo mais complexo.

Para realizar um backup completo offline, você deve ter permissão de leitura nos arquivos do banco de dados a partir do sistema operacional e permissão de escrita

no diretório de backup a partir do sistema operacional. Se for necessário desligar o banco de dados primeiro, você deve ter privilégios de segurança DBA, SYSDBA ou SYSADM.

Você pode realizar um backup completo offline independentemente do modo de backup; o banco de dados pode estar operando em modo NON-BACKUP, BACKUP-DATA ou BACKUP-DATA-AND-BLOB. Usando um backup completo offline, é possível restaurar o banco de dados até o momento em que foi desligado.

Observe que o backup completo offline usando o JServer Manager não faz backup de objetos de arquivos. Os objetos de arquivos devem ser copiados manualmente. Certifique-se de replicar exatamente a estrutura de arquivos e diretórios se estiver restaurando um banco de dados a partir de um backup completo offline. Para obter instruções sobre como realizar um backup completo offline usando o JServer Manager, consulte o Guia do Usuário do JServer Manager.

Offline Full Backup Using dmSQL

Para realizar um backup completo offline usando dmSQL:

1. Notifique todos os usuários que o banco de dados será desligado em um horário especificado e peça que se desconectem antes desse horário.
2. Se o banco de dados estiver em execução, desligue-o usando o comando `TERMINATE DB`. Se houver erros ao desligar o banco de dados, reinicie-o, corrija o problema e desligue-o novamente.
3. Examine o arquivo `dmconfig.ini` e liste todos os arquivos e diretórios relevantes, incluindo o diretório de objetos de arquivos, que necessitam de backup.
4. Use comandos ou utilitários do sistema operacional para copiar os arquivos do banco de dados, incluindo arquivos de dados, arquivos de diário, objetos de arquivos e o arquivo `dmconfig.ini`, para o diretório de backup ou dispositivo de backup.

Offline Full Backup Using JServer Manager

Se o banco de dados estiver offline, você pode realizar um backup completo offline usando a ferramenta gráfica JServer Manager. Note que o backup completo offline usando o JServer Manager não faz backup de objetos de arquivos. Os objetos de arquivos devem ser copiados manualmente. Certifique-se de replicar exatamente a estrutura de arquivos e diretórios ao restaurar um banco de dados a partir de um backup completo offline. Para instruções sobre como realizar um backup completo offline usando o JServer Manager, consulte o Guia do Usuário do JServer Manager.

15.6 Backup Server

Embora o DBMaker forneça métodos para fazer backups de bancos de dados manualmente, você deve se lembrar de realizar backups regularmente. Para ajudar, o Backup Server oferece uma maneira conveniente de automatizar totalmente os backups online completos, diferenciais e incrementais.

NOTA: *O Backup Server só pode realizar um backup online, pois ele só inicia após o início do banco de dados.*

O Backup Server é executado em segundo plano e realiza backups online completos, diferenciais e incrementais em uma agenda definida, conforme os arquivos de jornal se tornam cheios, ou ambos. Essa flexibilidade é possível porque o Backup Server e o servidor de banco de dados se comunicam para determinar quando um backup deve ocorrer, o tipo de backup incremental a ser realizado e quais opções de backup alterar. O Backup Server inicia ao mesmo tempo que o servidor de banco de dados e continua executando até que você o pare ou desligue o servidor de banco de dados.

Ao realizar backups completos, o Backup Server copiará o último backup completo do diretório de backup para o diretório antigo. Em seguida, ele copiará todos os arquivos do banco de dados, incluindo arquivos de jornal e dmconfig.ini, para o diretório de backup, sobrescrevendo o backup completo anterior.

Ao realizar backups diferenciais, o Backup Server copia apenas arquivos de dados (DB e BB). Arquivos de jornal são excluídos porque mudam com muita frequência. Durante os backups diferenciais, apenas blocos úteis de jornal são copiados. Além disso, arquivos de dados em tablespaces somente leitura são excluídos dos backups diferenciais.

Ao realizar backups incrementais, o Backup Server copiará os arquivos de jornal necessários para o diretório de backup.

Existem várias opções usadas para configurar o Backup Server. Essas opções controlam o formato do nome do arquivo de backup, a localização do diretório de backup, a localização do diretório antigo, a agenda que o Backup Server usa para realizar backups, o intervalo e o número máximo de backups diferenciais a serem mantidos após um backup completo, a quantidade de um arquivo de jornal que deve ser preenchida antes que o Backup Server realize um backup incremental, e a maneira como o Backup Server salva os arquivos de backup.

O Backup Server também permite que as configurações de backup sejam feitas durante a execução com o procedimento armazenado SetSystemOption do dmSQL, ou seja, BKSVR, BKDIR, BKITV, DBKTV, BKTIM, BKFUL, BKFOM, BKZIP, BKCMP,

BKRTS, BKCHK, FBKTM, FBKTV, DBKMX, BKODR, BKFRM podem ser alterados durante a execução com o procedimento armazenado do sistema SetSystemOption.

Starting Backup Server

Iaro! Aqui está a tradução:

- Os usuários podem definir palavras-chave relacionadas no dmconfig.ini antes de iniciar o banco de dados. Na próxima vez que os usuários iniciarem o banco de dados, o servidor de backup usará essas palavras-chave para inicializar os parâmetros associados.
- Se o banco de dados já estiver iniciado, os usuários podem usar a Configuração em Tempo de Execução no Jserver Manager para alterar os valores dos parâmetros. Além disso, os usuários podem definir parâmetros com a chamada SetSystemOption('option_name','value'). Observe que parâmetros individuais só podem ser definidos com a sintaxe set, como set backup OFF; set data backup ON; set blob backup ON.

Quando o Backup Server estiver ativado e os parâmetros de backup apropriados estiverem configurados no arquivo de configuração dmconfig.ini, você pode chamar o procedimento armazenado do sistema SetSystemOption para iniciar um backup. O procedimento armazenado pode ser utilizado por qualquer ferramenta de cliente ou aplicação de usuário.

A sintaxe para realizar backup completo, diferencial e incremental online é:

```
dmSQL> CALL SETSYSTEMOPTION('STARTBACKUP','1'); //do full backup
dmSQL> CALL SETSYSTEMOPTION('STARTBACKUP','2'); //do incremental
backup
dmSQL> CALL SETSYSTEMOPTION('STARTBACKUP','3'); //do differential
backup
```

USING DMCONFIG.INI TO START BACKUP SERVER

Se o banco de dados estiver offline, você pode ativar o Backup Server diretamente usando a palavra-chave DB_BkSvr no arquivo dmconfig.ini. Na próxima vez que você iniciar o banco de dados, o Backup Server também será iniciado. Se o banco de dados estiver online, alterar o valor da palavra-chave DB_BkSvr encontrada no arquivo de configuração dmconfig.ini não terá efeito até que o banco de dados seja desligado e reiniciado.

Para iniciar o Backup Server usando o arquivo dmconfig.ini:

1. Abra o arquivo dmconfig.ini no servidor de banco de dados usando qualquer editor de texto ASCII.
2. Localize a seção de configuração do banco de dados para o banco de dados no qual você deseja ativar o Backup Server.
3. Certifique-se de que o modo de backup do banco de dados seja BACKUP-DATA ou BACKUP-DATA-AND-BLOB. O banco de dados está no modo BACKUP-DATA se o valor de DB_BMode estiver definido como 1, e no modo BACKUP-DATA-AND-BLOB se o valor de DB_BMode estiver definido como 2.
4. Altere o valor da palavra-chave DB_BkSvr para 1 para ativar o Backup Server.
5. Reinicie o banco de dados para começar a usar o Backup Server.

USING DMSQL TO START BACKUP SERVER

Quando um banco de dados está online, o Backup Server pode ser habilitado dinamicamente usando a ferramenta de linha de comando dmSQL, conforme mostrado abaixo.

```
dmSQL> CALL SETSYSTEMOPTION('BKSVR','1');
```

Os usuários podem alterar BkSvr com o comando **Call SetSystemOption('BkSvr', '1')**. Para alterar BkSvr e gravar o valor de DB_BkSvr no arquivo de configuração dmconfig.ini ao mesmo tempo, os usuários podem usar o comando **Call SetSystemOptionW('option', 'value')**.

Quando o Backup Server está ativado e os parâmetros de backup apropriados estão configurados no arquivo de configuração dmconfig.ini, você pode chamar o procedimento armazenado do sistema SetSystemOption para iniciar um backup. Esse procedimento armazenado pode ser utilizado por qualquer ferramenta de cliente e aplicações de usuário.

```
dmSQL> CALL SETSYSTEMOPTION('STARTBACKUP','1'); //do full backup
dmSQL> CALL SETSYSTEMOPTION('STARTBACKUP','2'); //do incremental
backup
dmSQL> CALL SETSYSTEMOPTION('STARTBACKUP','3'); //do differential
backup
```

A sintaxe para alterar o intervalo de backup incremental é:

USING JSERVER MANAGER TO START BACKUP SERVER

Independentemente do status online do banco de dados, você pode habilitar o Backup Server dinamicamente usando a ferramenta gráfica JServer Manager. O JServer Manager altera automaticamente o valor da palavra-chave DB_BkSvr no arquivo de configuração dmconfig.ini. Se o banco de dados estiver offline, na próxima vez que você iniciar o banco de dados, o Backup Server também será iniciado. Para obter instruções sobre como iniciar o Backup Server enquanto o banco de dados está offline usando o JServer Manager, consulte o Guia do Usuário do JServer Manager.

Differential Backup Filename Format

Este é o formato do nome do arquivo de backup diferencial: DTimeStamp_DataFileName.dif(2) D — Identificação obrigatória do backup diferencial TimeStamp — Número de segundos desde 1º de janeiro de 1970 (00:00:00 GMT) DataFileName — Nome do banco de dados do arquivo de dados .dif — Extensão de arquivo usada para arquivos de backup diferencial. Se não existir um arquivo de backup diferencial correspondente para um backup completo específico, a extensão do arquivo deve ser .dif2.

Aqui está um exemplo:

Se um primeiro backup diferencial for realizado em 01/12/2009 às 14:11, os nomes dos arquivos de backup diferencial serão gerados da seguinte forma:

D1259647860_DBNAME.BB.dif

D1259647860_DBNAME.DB.dif

D1259647860_DBNAME.SBB.dif

D1259647860_DBNAME.SDB.dif

O nome do arquivo de diário será:

D1259647860_DBNAME.JNL

Incremental Backup Filename Format

O formato do nome do arquivo de backup é <I><TimeStamp><><DB_BkFrm>, por exemplo: I1234567890%2F%4N%4B.JNL. O comprimento total do nome do arquivo não pode exceder 256 caracteres. O timestamp é um dado numérico de 10 dígitos válido no tempo do sistema, e o <DB_BkFrm> pode incluir tanto constantes de texto

quanto sequências de formato (por exemplo, sequências de escape), que representam cadeias de caracteres especiais.

Um nome de arquivo de backup incremental deve consistir em pelo menos três cadeias de caracteres especiais: o ID do backup completo, o nome do banco de dados e o número de identificação do backup. O Backup Server atribui um ID de backup completo ao nomear arquivos incrementais em uma sequência de backup. Ao restaurar um banco de dados, o DBMaker usa o ID de backup completo para recriar corretamente a sequência de backup. O nome do banco de dados identifica corretamente o banco de dados ao qual um arquivo de backup incremental pertence. O número de identificação do backup identifica a posição relativa do arquivo de backup incremental na sequência de backup.

As sequências de formato têm três partes: o caractere de escape, o valor do comprimento e o caractere de formato. As sequências de formato válidas são: %[x]F — O ID do backup completo. A variável x pode ter valores de 1 a 4, onde os valores representam os seguintes formatos: 1: ID do backup completo mostrado como AAAAMMDD, por exemplo, 20010917

2: ID do backup completo mostrado como MMDD, por exemplo, 0917

3: ID do backup completo mostrado como MMDDhhmm, por exemplo, 09171305

4: ID do backup completo mostrado como DDhhmmss, por exemplo, 17130558

%[n]B — Número de identificação do backup

%[n]N — O arquivo de diário pertence a este banco de dados

O caractere de escape identifica o início da sequência de formato e é representado pelo símbolo %. Se você quiser incluir o símbolo % como uma constante de texto no formato do nome do backup, deve usar dois símbolos % juntos (ou seja, %%). Um único dígito ou um dos caracteres de formato válidos mostrados acima deve seguir imediatamente o símbolo %. Se qualquer outro caractere seguir o símbolo %, o formato do nome do backup é inválido, e o DBMaker retornará um erro.

O valor do comprimento n é um valor inteiro entre um e nove que determina o comprimento da cadeia de caracteres gerada pela sequência de formato. Se a sequência de formato retornar uma cadeia que pode ser representada em menos caracteres do que o valor do comprimento fornece, zeros serão adicionados a ela. O nome do banco de dados tem zeros adicionados à direita do nome, enquanto todos os outros valores têm zeros adicionados à esquerda. Se a sequência de formato retornar uma cadeia que requer mais caracteres do que o valor do comprimento fornece, ela será truncada. O nome do banco de dados é truncado pela direita,

enquanto todos os outros valores são truncados pela esquerda. Os colchetes que envolvem o valor do comprimento indicam que o valor do comprimento é opcional; não inclua os colchetes ao inserir a sequência de formato. Se você não fornecer um valor para o comprimento, o Backup Server usará o comprimento total da cadeia de caracteres gerada pela sequência de formato.

O caractere de formato identifica o tipo de cadeia de caracteres especial que a sequência de formato retornará. O caractere de formato deve ser F, B ou N; usar qualquer outro caractere resultará em um formato de nome de backup inválido, e o DBMaker retornará um erro. Um caractere de formato válido que não segue imediatamente o caractere de escape ou o caractere de escape e um único dígito será tratado como uma constante de texto.

Os valores de data e hora são obtidos do sistema. Esses valores só estarão corretos se a data e a hora do sistema estiverem corretas. O valor do número de identificação do backup é a posição ordinal do arquivo de diário de backup na sequência de backup. O DBMaker fornece automaticamente esse número para cada arquivo de diário que é salvo pelo Backup Server.

O DBMaker fornece vários métodos diferentes para definir o formato do nome do arquivo de backup. O método que você escolhe depende de se você está mais confortável editando o arquivo de configuração diretamente ou usando a utilidade gráfica JServer Manager.

USING DMCONFIG.INI TO SET BACKUP FILE NAME FORMAT

Se o banco de dados estiver offline, você pode definir diretamente o formato do nome do arquivo de backup usado pelo Backup Server usando a palavra-chave DB_BkFrm no arquivo dmconfig.ini. Da próxima vez que você iniciar o banco de dados, o Backup Server aplicará esse formato de nome de arquivo de backup a todos os arquivos de diário de backup. Se o banco de dados estiver online, alterar o valor da palavra-chave DB_BkFrm não terá efeito até que o banco de dados seja desligado e reiniciado.

□ Para definir o formato do arquivo de backup usando o arquivo de configuração dmconfig.ini:

1. Abra o arquivo dmconfig.ini no servidor de banco de dados usando qualquer editor de texto ASCII.
2. Localize a seção de configuração do banco de dados para um banco de dados.
3. Altere o valor da palavra-chave DB_BkFrm para uma string contendo o formato a ser usado para o formato do nome do arquivo de backup. **NOTA:** A *string* pode conter quaisquer sequências de formato válidas e constantes de texto,

mas o comprimento total do nome do arquivo resultante não deve exceder 256 caracteres.

4. Reinicie o banco de dados para começar a usar o novo formato de nome de arquivo de backup.

USING DMSQL TO SET BACKUP FILE NAME FORMAT

O procedimento armazenado SetSystemOption pode ser usado para alterar o formato do nome do arquivo de backup enquanto o banco de dados está em execução. A sintaxe geral do comando é:

Exemplo: Para alterar o formato do nome do arquivo de backup para I1234567890_%2F%4N%4B.JNL, digite a seguinte linha no prompt de comando do dmSQL.

```
dmSQL> CALL SETSYSTEMOPTION('bkfrm', 'I1234567890_%2F%4N%4B.JNL');
```

USING JSERVER MANAGER TO SET BACKUP FILE NAME FORMAT

O formato do nome do arquivo de backup pode ser definido para bancos de dados offline ou online usando a utilidade gráfica do JServer Manager. O JServer Manager alterará automaticamente o valor da palavra-chave DB_BkFrm no arquivo dmconfig.ini. Da próxima vez que você iniciar o banco de dados, o Backup Server aplicará esse formato de nome de arquivo de backup a todos os arquivos de diário de backup. Para obter instruções sobre como definir o formato do arquivo de backup usando o JServer Manager, consulte o Guia do Usuário do JServer Manager.

Backup Directory

O diretório de backup especifica onde o Backup Server colocará os arquivos de backup. O DBMaker suporta um caminho de arquivo de backup único e múltiplos caminhos de arquivo de backup para os usuários. O Backup Server criará automaticamente o BkDir. No entanto, você deve escolher um ou mais diretórios de backup em um disco diferente dos arquivos do banco de dados para evitar a perda tanto do banco de dados quanto dos arquivos de backup em caso de erro de mídia.

O diretório de backup é especificado pela palavra-chave DB_BkDir no arquivo dmconfig.ini. O valor da palavra-chave DB_BkDir pode conter um caminho completo ou relativo para o diretório de backup. Se você não especificar um diretório de backup, o Backup Server criará automaticamente um diretório de backup padrão chamado "backup" no diretório do banco de dados. O diretório do banco de dados é

especificado pela palavra-chave DB_DbDir no arquivo dmconfig.ini. O comprimento total do caminho do diretório de backup não deve exceder 256 caracteres.

No entanto, se o banco de dados estiver em modo de replicação (mestre ou escravo), BKDIR deve ser um único caminho. Se você definir BKDIR com múltiplos caminhos, apenas o primeiro será utilizado e o tamanho dos caminhos adicionais será ignorado. Além disso, não é uma boa ideia permitir que o Backup Server crie e use o diretório de backup padrão se você tiver mais de um banco de dados no mesmo diretório. Nesse caso, as informações de histórico de backup de um banco de dados podem sobrescrever ou ser adicionadas ao histórico de backup de outro banco de dados, tornando um ou ambos os backups inutilizáveis. Para evitar esse tipo de problema, você pode colocar cada banco de dados em um diretório de banco de dados diferente ou especificar explicitamente um diretório de backup para cada banco de dados. Colocar cada banco de dados em um diretório de banco de dados diferente é o método preferido, pois isso permite ver exatamente quais arquivos pertencem a qual banco de dados.

O DBMaker fornece vários métodos diferentes para definir o diretório de backup. O método que você escolhe depende de se seu banco de dados está online ou offline e de sua preferência por editar o arquivo de configuração diretamente ou usar a utilidade gráfica JServer Manager.

USING DMCONFIG.INI TO SET BACKUP DIRECTORY

Se o banco de dados estiver offline, você pode definir o diretório de backup usado pelo Backup Server diretamente usando a palavra-chave DB_BkDir no arquivo dmconfig.ini. Da próxima vez que você iniciar o banco de dados, o Backup Server usará esse diretório como o diretório de backup. Se o banco de dados estiver online, alterar o valor da palavra-chave DB_BkDir não terá efeito até que o banco de dados seja desligado e reiniciado.

Para definir o diretório de backup usando o arquivo de configuração dmconfig.ini:

1. Abra o arquivo dmconfig.ini no servidor de banco de dados usando qualquer editor de texto ASCII.
2. Localize a seção de configuração do banco de dados para um banco de dados.
3. Altere o valor da palavra-chave DB_BkDir para uma string contendo o nome de um diretório existente para definir o diretório de backup.
4. Reinicie o banco de dados para começar a usar o novo diretório de backup.

USING DMSQL TO SET BACKUP DIRECTORY

O procedimento armazenado SetSystemOption pode ser usado para alterar o diretório de backup enquanto o banco de dados está em execução. A sintaxe geral do comando é:

```
CALL SETSYSTEMOPTION('bkdir', 'path')
```

Path é o caminho completo do novo diretório de backup. O comprimento da string em path não deve exceder 256 caracteres.

Exemplo:

Para alterar o caminho do diretório para E:\storage\database\backup\WebDB, digite a seguinte linha no prompt de comando do dmSQL.

```
dmSQL> CALL SETSYSTEMOPTION('bkdir',  
'E:\\storage\\database\\backup\\WebDB');
```

USING JSERVER MANAGER TO SET BACKUP DIRECTORY

Se o banco de dados estiver offline, você pode definir o diretório de backup offline usado pelo Backup Server utilizando a ferramenta gráfica JServer Manager. O JServer Manager alterará automaticamente o valor da palavra-chave DB_BkDir no arquivo dmconfig.ini. Na próxima vez que você iniciar o banco de dados, o Backup Server usará esse diretório como o diretório de backup. Se o banco de dados estiver online, o JServer Manager pode alterar o diretório de backup imediatamente com a Configuração em Tempo de Execução ou adiar a mudança até a próxima vez que você reiniciar o banco de dados ao fazer um backup interativo. Em ambos os casos, o JServer Manager também fará uma cópia do arquivo de histórico de backup no novo diretório de backup. Para obter instruções sobre como definir o diretório de backup usando o JServer Manager, consulte o Guia do Usuário do JServer Manager.

Setting Multiple Backup Paths

O DBMaker também suporta múltiplos caminhos de arquivo de backup para os usuários. Esta função é útil quando um usuário tenta salvar em um caminho de backup, mas o caminho de backup não fornece espaço suficiente para que o backup seja concluído. Se a opção de múltiplos backups estiver configurada, o DBMaker redirecionará os dados restantes a serem backupados para locais de backup secundários, de modo que o backup possa ser realizado corretamente. Os usuários

podem utilizar múltiplos caminhos de backup em backups completos, diferenciais ou incrementais. O DBMaker possui as seguintes restrições ao fazer backup de informações usando múltiplos caminhos de backup:

- Quando um sistema de banco de dados tenta fazer backup de arquivos, ele tentará armazenar os arquivos nos caminhos um por um para cada arquivo. Por exemplo, ao armazenar um arquivo no diretório de backup 1 e o diretório não tiver espaço suficiente para armazenar o arquivo, então o arquivo será redirecionado para o diretório de backup 2, e assim por diante. Se todos os diretórios de backup estiverem cheios, uma mensagem de erro será retornada.
- Apenas um diretório de backup pode ser usado para fazer backup de arquivos nos sites escravos.
- Os FOs devem fazer backup no primeiro diretório de backup.
- O número máximo de caminhos de backup é 32.

Exemplo:

Ao configurar múltiplos caminhos de backup, o DBMaker segue a seguinte estrutura:

```
DB_BkDir = <BKDIR 1> <SIZE 1> < BKDIR 2> <SIZE 2> < BKDIR 3> <SIZE 3>...  
< BKDIR n > : the n's backup path  
< SIZE n > : the size of the n's backup path
```

Portanto, ao configurar múltiplos caminhos de backup para o banco de dados DB1, você precisa definir os caminhos no DB_BkDir.

```
DB_BkDir = /home/usr/dbmaker/bk 5000 /home2/backup 1000
```

Quando o espaço disponível em /home/usr/dbmaker/bk estiver cheio, o banco de dados fará o backup em /home2/backup.

Setting the Old Directory

O diretório antigo é um diretório ou um grupo de diretórios (até 32) e é utilizado para salvar uma sequência de backup que é a imediatamente anterior à última. Você deve escolhê-lo em um disco diferente dos arquivos do banco de dados para evitar a perda tanto dos arquivos do banco de dados quanto dos arquivos de backup em caso de erro de mídia.

O diretório antigo é especificado pela palavra-chave DB_BkOdr no arquivo dmconfig.ini. Se você não o especificar, o Servidor de Backup descartará a sequência de backup anterior.

USING DMCONFIG.INI TO SET OLD DIRECTORY

Você pode definir o diretório antigo usado pelo Servidor de Backup diretamente utilizando a palavra-chave DB_BkOdr no arquivo dmconfig.ini. Na próxima vez que você iniciar o banco de dados, o Servidor de Backup usará este diretório como o diretório antigo. Se o banco de dados estiver online, alterar o valor da palavra-chave DB_BkOdr não terá efeito até que o banco de dados seja desligado e reiniciado.

USING DMSQL TO SET OLD DIRECTORY

O procedimento armazenado SetSystemOption pode ser usado para alterar o diretório de backup antigo enquanto o banco de dados está em execução. A sintaxe geral para o comando é:

```
CALL SETSYSTEMOPTION('bkodr', 'path')
```

Path é o caminho completo do novo diretório de backup antigo. O comprimento da string no caminho não deve exceder 256 caracteres.

Exemplo:

Para alterar o caminho do diretório antigo para E:\storage\database\backup\WebDB, insira a seguinte linha no prompt de comando dmSQL.

```
dmSQL> CALL SETSYSTEMOPTION('bkodr',  
'E:\\storage\\database\\backup\\WebDB');
```

USING JSERVER MANAGER TO SET OLD DIRECTORY

Se o banco de dados estiver offline, você pode definir a localização do backup anterior usando a ferramenta gráfica JServer Manager. O JServer Manager mudará automaticamente o valor da palavra-chave DB_BkOdr no arquivo dmconfig.ini. Na próxima vez que você iniciar o banco de dados, o Servidor de Backup usará este diretório como o diretório de backup. Se o banco de dados estiver online, o JServer Manager pode alterar o diretório de backup antigo imediatamente ou adiar a mudança até a próxima vez que você reiniciar o banco de dados. Para obter instruções sobre

como definir o diretório de backup antigo enquanto estiver offline usando o JServer Manager, consulte o Guia do Usuário do JServer Manager.

Differential Backup Settings

A programação de backup diferencial especifica quando o Servidor de Backup realiza backups diferenciais online. A programação inclui o horário inicial do backup e o intervalo de tempo. O horário inicial do backup é a data e hora em que o Servidor de Backup realizará o primeiro backup diferencial. O intervalo de tempo especifica o tempo de espera entre backups diferenciais subsequentes.

O horário inicial do backup completo é especificado pela palavra-chave `DB_FBkTm`, encontrada no arquivo `dmconfig.ini`. O valor da palavra-chave `DB_FBkTm` deve ser uma data e hora no seguinte formato: `YY/MM/DD HH:MM`. Não há valor padrão para o horário inicial do backup; no entanto, ao habilitar o Servidor de Backup com o JServer Manager, um valor padrão é adicionado ao arquivo `dmconfig.ini`.

O intervalo de tempo é especificado pela palavra-chave `DB_DBkTv`, encontrada no arquivo `dmconfig.ini`. O primeiro backup diferencial é realizado em `DB_FBkTm + DB_DBkTv`. O valor da palavra-chave `DB_DBkTv` deve ser um intervalo de tempo no seguinte formato: `D-HH:MM`. Não há valor padrão para o intervalo de tempo; no entanto, ao habilitar o Servidor de Backup com o JServer Manager, um valor padrão de `1-00:00:00` é adicionado ao arquivo `dmconfig.ini`.

Por fim, a palavra-chave `DB_DbKmx` especifica o número máximo de backups diferenciais a serem mantidos após um backup completo. O Servidor de Backup remove o backup diferencial mais antigo quando o número de backups diferenciais, após um backup completo, excede `DB_DbKmx`. O procedimento armazenado do sistema `SetSystemOption` pode ser usado para alterar `DBKMX` enquanto o banco de dados está em execução. Além disso, a palavra-chave `DB_BkChk` especifica se deve verificar o banco de dados antes do backup diferencial, e o procedimento armazenado do sistema `SetSystemOption` pode ser usado para alterar `BKCHK` enquanto o banco de dados está em execução.

USING DMCONFIG.INI TO CHANGE DIFFERENTIAL BACKUP SETTINGS

Quando um banco de dados está offline, sua programação de backup pode ser definida usando as palavras-chave `DB_FBkTm` e `DB_DBkTv`, encontradas no arquivo de configuração `dmconfig.ini`. Na próxima vez que o banco de dados for iniciado, o Servidor de Backup usará essas configurações para a programação de backups diferenciais. Se o banco de dados estiver online, alterar o valor das palavras-chave

DB_FBkTm e DB_DBkTv não terá efeito até que o banco de dados seja desligado e reiniciado.

Para definir a programação de backup usando o arquivo dmconfig.ini:

1. Abra o arquivo dmconfig.ini no servidor do banco de dados usando qualquer editor de texto ASCII.
2. Localize a seção de configuração do banco de dados desejado.
3. Altere o valor da palavra-chave DB_FBkTm para um valor de data e hora usando este formato: YY/MM/DD HH:MM.
4. Altere o valor da palavra-chave DB_DBkTv para um valor de intervalo de tempo usando este formato: ndias-HH:MM.
5. Reinicie o banco de dados para ativar a nova programação de backup.

USING DMSQL TO CHANGE DIFFERENTIAL BACKUP SETTINGS

O procedimento armazenado SetSystemOption pode ser usado para ativar o Servidor de Backup. O comando é:

```
dmSQL> CALL SETSYSTEMOPTION('BKSVR','1');
```

Quando o Servidor de Backup é ativado, inicia um backup diferencial chamando o procedimento armazenado do sistema SetSystemOption:

```
dmSQL> CALL SETSYSTEMOPTION('STARTBACKUP','3');
```

A sintaxe para alterar o intervalo de backup diferencial é:

```
CALL SETSYSTEMOPTION('dbktv','Interval')
```

USING JSERVER MANAGER TO CHANGE DIFFERENTIAL BACKUP SETTINGS

Se o banco de dados estiver offline, a programação de backup diferencial pode ser definida usando a ferramenta gráfica JServer Manager. O JServer Manager altera automaticamente os valores das palavras-chave DB_FBkTm e DB_DBkTv, encontradas no arquivo dmconfig.ini. Na próxima vez que o banco de dados for iniciado, o Servidor de Backup usará essas configurações como a nova programação de backup diferencial. Se o banco de dados estiver online, o JServer Manager pode alterar imediatamente a programação de backup ou, opcionalmente, adiar as

mudanças até a próxima vez que o banco de dados for reiniciado. Para mais detalhes, consulte o Guia do Usuário do JServer Manager.

Incremental Backup Settings

O cronograma de backup incremental especifica os horários em que o Backup Server realizará um backup incremental online. O cronograma é composto por duas partes: o horário do backup inicial e o intervalo de tempo. O horário do backup inicial determina a data e a hora em que o Backup Server realizará o primeiro backup incremental, enquanto o intervalo de tempo determina a duração da espera entre os backups incrementais subsequentes.

Você pode combinar o cronograma de backup incremental com o valor do gatilho do diário para fazer o backup do seu banco de dados tanto em um cronograma regular quanto quando os arquivos de diário atingirem uma porcentagem especificada. Se você não especificar um cronograma de backup incremental, o Backup Server não fará o backup do banco de dados em um cronograma regular.

O horário do backup inicial é especificado pela palavra-chave `DB_BkTim` no arquivo `dmconfig.ini`. Você deve inserir o valor da palavra-chave `DB_BkTim` como uma data e hora no formato `YY/MM/DD HH:MM`. Não há um valor padrão para o horário do backup inicial.

O intervalo de tempo é especificado pela palavra-chave `DB_BkItv` no arquivo `dmconfig.ini`. Você deve inserir o valor da palavra-chave `DB_BkItv` como um intervalo de tempo no formato `D-HH:MM`. Não há um valor padrão para o intervalo de tempo. No entanto, se você usar o JServer Manager para ativar o Backup Server, o JServer Manager fornecerá um valor padrão de `1-00:00:00` para você e escreverá esse valor no arquivo `dmconfig.ini`.

O DBMaker oferece vários métodos diferentes para definir o cronograma de backup incremental. O método que você escolher dependerá de o seu banco de dados estar online ou offline e se você se sente mais confortável editando o arquivo de configuração diretamente ou usando a ferramenta gráfica JServer Manager.

USING DMCONFIG.INI TO CHANGE INCREMENTAL BACKUP SETTINGS

Se o banco de dados estiver offline, você pode definir o cronograma de backup utilizado pelo Backup Server diretamente usando as palavras-chave `DB_BkTim` e `DB_BkItv` no arquivo de configuração `dmconfig.ini`. Na próxima vez que você iniciar o banco de dados, o Backup Server usará essas configurações para o cronograma de

backup incremental. Se o banco de dados estiver online, alterar o valor das palavras-chave DB_BkTim e DB_BkItv não terá efeito até que o banco de dados seja desligado e reiniciado.

Para definir o cronograma de backup usando o arquivo dmconfig.ini:

1. Abra o arquivo dmconfig.ini no servidor do banco de dados usando qualquer editor de texto ASCII.
2. Localize a seção de configuração do banco de dados para o banco de dados que deseja alterar o cronograma de backup.
3. Altere o valor da palavra-chave DB_BkTim para uma data e hora usando o formato de valor YY/MM/DD HH:MM.
4. Altere o valor da palavra-chave DB_BkItv para um intervalo de tempo usando o formato de valor DDDD-HH:MM.
5. Reinicie o banco de dados para começar a usar o novo cronograma de backup.

USING DMSQL TO CHANGE INCREMENTAL BACKUP SETTINGS

A stored procedure **SetSystemOption** pode ser utilizada para alterar o horário de início e o intervalo do backup incremental enquanto o banco de dados está em execução. A sintaxe geral para mudar o horário de início do backup incremental é:

```
CALL SETSYSTEMOPTION('bktim', 'StartTime')
```

A sintaxe geral para alterar o intervalo do backup incremental é:

```
CALL SETSYSTEMOPTION('bkItv', 'Interval')
```

StartTime é o momento em que começa o primeiro backup incremental e tem o formato YY:MM

HH:MM. **Interval** é o intervalo de tempo em que os backups incrementais ocorrem e tem o formato D-HH:MM. Quando o

Backup Server é ativado, chame a store procedure do sistema **SetSystemOption** para iniciar um backup incremental.

Exemplo:

Para definir o intervalo de backup incremental para 1 hora, insira a seguinte linha no prompt de comando do dmSQL.

```
dmSQL> CALL SETSYSTEMOPTION('bkltv','0-1:00:00');
```

USING JSERVER MANAGER TO CHANGE INCREMENTAL BACKUP SETTINGS

Se o banco de dados estiver offline, você pode definir o cronograma de backup incremental utilizado pelo **Backup Server** usando a ferramenta gráfica **JServer Manager**. O **JServer Manager** irá alterar automaticamente os valores das palavras-chave **DB_BkTim** e **DB_Bkltv** no arquivo **dmconfig.ini**. Na próxima vez que você iniciar o banco de dados, o **Backup Server** usará essas configurações como o novo cronograma de backup incremental. Se o banco de dados estiver online, o **JServer Manager** pode alterar o cronograma de backup imediatamente ou adiar a mudança até a próxima vez que você reiniciar o banco de dados. Para instruções sobre como definir o cronograma de backup incremental usando o **JServer Manager**, consulte o **Guia do Usuário do JServer Manager**.

Journal Trigger Value Settings

O valor da trigger do diário especifica a porcentagem que um arquivo de diário deve preencher antes que o **Backup Server** realize um backup incremental online. Você pode combinar o valor do gatilho do diário com o cronograma de backup para fazer backup do seu banco de dados em um cronograma regular e quando os arquivos de diário atingirem a porcentagem especificada.

O valor do gatilho do diário é especificado pela palavra-chave **DB_BkFul** no arquivo **dmconfig.ini**. O valor da palavra-chave **DB_BkFul** pode ser um valor inteiro no intervalo de 50 a 100, ou zero. Valores entre 50 e 100 representam a porcentagem que um arquivo de diário deve preencher antes que o **Backup Server** realize um backup. Um valor de zero faz com que o **Backup Server** realize um backup sempre que um arquivo de diário estiver completamente cheio. Definir o valor como 0 é efetivamente o mesmo que defini-lo como 100, uma vez que ambos farão com que o **Backup Server** realize um backup sempre que um arquivo de diário estiver completamente cheio (100% cheio). Se você não especificar um valor para o gatilho do diário, o **Backup Server** usará o valor padrão de 90.

O **DBMaker** fornece vários métodos diferentes para definir o valor do gatilho do diário. O método que você escolher depende de o seu banco de dados estar online ou offline, e se você se sente mais confortável editando o arquivo de configuração diretamente ou usando a ferramenta gráfica **JServer Manager**.

USING DMCONFIG.INI TO CHANGE JOURNAL TRIGGER VALUE

Se o banco de dados estiver offline, você pode definir o valor do gatilho do diário usado pelo **Backup Server** diretamente utilizando a palavra-chave **DB_BkFul** no arquivo **dmconfig.ini**. Na próxima vez que você iniciar o banco de dados, o **Backup Server** usará essa configuração para o valor do gatilho do diário. Se o banco de dados estiver online, mudar o valor da palavra-chave **DB_BkFul** não terá efeito até que o banco de dados seja desligado e reiniciado.

Para definir o valor do gatilho do diário usando o arquivo **dmconfig.ini**:

1. Abra o arquivo **dmconfig.ini** no servidor de banco de dados usando qualquer editor de texto ASCII.
2. Localize a seção de configuração do banco de dados para alterar o valor do gatilho do diário.
3. Altere o valor da palavra-chave **DB_BkFul** para um valor inteiro entre 50 e 100, ou defina-o como zero.
4. Reinicie o banco de dados para começar a usar o novo valor do gatilho do diário.

USING DMSQL TO CHANGE JOURNAL TRIGGER VALUE

O procedimento armazenado **SetSystemOption** pode ser usado para alterar o valor do gatilho do diário enquanto o banco de dados está em execução. A sintaxe geral para alterar o valor do gatilho do diário é:

```
CALL SETSYSTEMOPTION('bkful', 'n')
```

Onde **n** é 0 ou um valor entre 50 e 100. Definir **n** como 0 ativará o servidor de backup sempre que um arquivo de diário estiver cheio. Definir **n** entre 50 e 100 especifica a porcentagem que um arquivo de diário deve preencher antes que o servidor de backup seja ativado. Exemplo:

Para definir o valor do gatilho do diário para 75 por cento, insira a seguinte linha no prompt de comando do dmSQL:

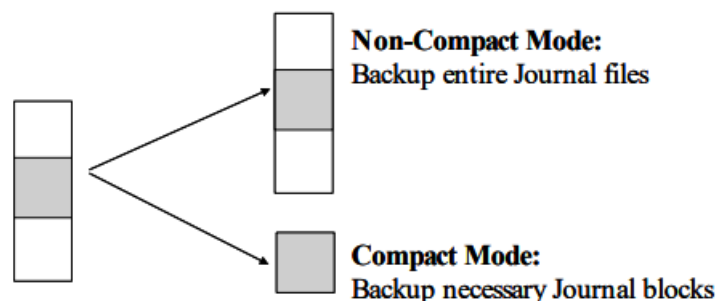
```
dmSQL> CALL SETSYSTEMOPTION('bkful', '75');
```

USING JSERVER MANAGER TO CHANGE JOURNAL TRIGGER VALUE

Se o banco de dados estiver offline, você pode definir o valor do gatilho do diário usado pelo **Backup Server** utilizando a ferramenta gráfica **JServer Manager**. O **JServer Manager** irá alterar automaticamente o valor da palavra-chave **DB_BkFul** no arquivo **dmconfig.ini**. Na próxima vez que você iniciar o banco de dados, o **Backup Server** usará essa configuração como o novo valor do gatilho do diário. Se o banco de dados estiver online, o **JServer Manager** pode alterar o valor do gatilho do diário imediatamente ou adiar a mudança até a próxima vez que você reiniciar o banco de dados. Para instruções sobre como definir o valor do gatilho do diário usando o **JServer Manager**, consulte o **Guia do Usuário do JServer Manager**.

Compact Backup Mode Settings

O modo de backup compacto especifica se o **Backup Server** fará backup de arquivos de diário inteiros ou apenas dos blocos de diário completos ao realizar um backup incremental ou diferencial online. Isso é possível porque nem todo bloco de diário contém dados necessários para restaurar um banco de dados, então o **Backup Server** fará backup apenas dos blocos de diário necessários ao realizar um backup. Isso permite economizar espaço de armazenamento no seu dispositivo de backup, mas também significa que a restauração do banco de dados pode levar mais tempo.



O modo de backup compacto é especificado pela palavra-chave **DB_BkCmp** no arquivo de configuração **dmconfig.ini**. O valor da palavra-chave **DB_BkCmp** pode ser zero ou um. Definir o valor como um habilita o modo de backup compacto, e definir como zero desabilita o modo de backup compacto. Se você não especificar um valor para o modo de backup compacto, o **Backup Server** usará o valor padrão de um (habilitado).

O **DBMaker** fornece vários métodos diferentes para definir o modo de backup compacto. O método que você escolher depende de o seu banco de dados estar online ou offline, e se você se sente mais confortável editando o arquivo de configuração diretamente ou usando a ferramenta gráfica **JServer Manager**.

USING DMCONFIG.INI TO SET COMPACT BACKUP MODE

Se o banco de dados estiver offline, você pode definir o modo de backup compacto utilizado pelo **Backup Server** diretamente usando a palavra-chave **DB_BkCmp** no arquivo **dmconfig.ini**. Na próxima vez que você iniciar o banco de dados, o **Backup Server** usará essa configuração para o modo de backup compacto. Se o banco de dados estiver online, mudar o valor da palavra-chave **DB_BkCmp** não terá efeito até que o banco de dados seja desligado e reiniciado.

Para definir o modo de backup compacto usando o arquivo de configuração **dmconfig.ini**:

1. Abra o arquivo **dmconfig.ini** no servidor de banco de dados usando qualquer editor de texto ASCII.
2. Localize a seção de configuração do banco de dados para alterar o modo de backup compacto.
3. Altere o valor da palavra-chave **DB_BkCmp** para um (1) para habilitar o modo de backup compacto, ou para zero (0) para desabilitar o modo de backup compacto.
4. Reinicie o banco de dados para começar a usar o novo valor do gatilho do diário.

USING DMSQL TO SET COMPACT BACKUP MODE

O procedimento armazenado **SetSystemOption** pode ser usado para alterar o modo de backup compacto enquanto o banco de dados está em execução. Um backup bem-sucedido não requer todos os blocos de diário em um arquivo de diário. Se a palavra-chave **DB_BkCmp** estiver configurada para 1, o servidor de backup fará backup apenas dos blocos de diário que necessitam de backup. O comando é:

```
dmSQL> CALL SETSYSTEMOPTION('bkcmp','1');
```

USING JSERVER MANAGER TO SET COMPACT BACKUP MODE

Se o banco de dados estiver offline, você pode definir o modo de backup compacto utilizado pelo **Backup Server** usando a ferramenta gráfica **JServer Manager**. O **JServer Manager** irá alterar automaticamente o valor da palavra-chave **DB_BkCmp** no arquivo **dmconfig.ini**. Na próxima vez que você iniciar o banco de dados, o **Backup Server** usará essa configuração como o novo modo de backup compacto. Se o banco de dados estiver online, o **JServer Manager** pode alterar o modo de backup compacto imediatamente ou adiar a mudança até a próxima vez que você reiniciar o

banco de dados. Para instruções sobre como definir o modo de backup compacto usando o **JServer Manager**, consulte o **Guia do Usuário do JServer Manager**.

Full Backup Schedule

O cronograma de backup completo especifica os horários em que o **Backup Server** realizará um backup completo online. O cronograma é composto por duas partes: o horário do backup inicial e o intervalo de tempo. O horário do backup inicial determina a data e a hora em que o **Backup Server** realizará o primeiro backup completo, enquanto o intervalo de tempo determina o tempo a ser aguardado entre os backups completos subsequentes.

Você pode combinar cronogramas de backup completo e diferencial com um backup incremental para proteger seu banco de dados. Se você não especificar um cronograma de backup completo, o **Backup Server** não realizará backups completos em uma programação regular.

O horário do backup inicial é especificado pela palavra-chave **DB_FBkTm** no arquivo **dmconfig.ini**. Você deve inserir o valor da palavra-chave **DB_FBkTm** como uma data e hora no formato **YY/MM/DD HH:MM**. Não há valor padrão para o horário do backup inicial.

O intervalo de tempo é especificado pela palavra-chave **DB_FBkTv** no arquivo **dmconfig.ini**. Insira o valor da palavra-chave **DB_FBkTv** como um intervalo de tempo no formato **D-HH:MM**. Não há valor padrão para o intervalo de tempo.

Por fim, a palavra-chave **DB_BkChk** especifica se deve ser realizada uma verificação no banco de dados antes do backup completo e do backup diferencial, e a palavra-chave **DB_BkRTs** especifica se o servidor de backup inclui os arquivos de tablespace somente leitura ao realizar um backup completo. Para habilitar ou desabilitar esses dois recursos, você pode definir as palavras-chave **DB_BkChk** e **DB_BkRTs** no **dmconfig.ini**, ou alterar **BKCHK** e **BKRTS** usando o procedimento armazenado **SetSystemOption** enquanto o banco de dados está em execução.

USING DMCONFIG.INI TO SET FULL BACKUP SCHEDULE

Para definir o cronograma de backup completo usando o arquivo de configuração **dmconfig.ini**:

1. Abra o arquivo **dmconfig.ini** no servidor de banco de dados usando qualquer editor de texto ASCII.
2. Localize a seção de configuração do banco de dados para alterar o valor do gatilho do diário.

3. Defina o parâmetro de configuração **DB_FBkTm** para um valor no formato **YY/MM/DD HH:MM**, e **DB_FBkTv** para um valor no formato **D-HH:MM**.
4. Reinicie o banco de dados para começar a usar o novo cronograma de backup completo.

USING DMSQL TO SET FULL BACKUP SCHEDULE

O procedimento armazenado **SetSystemOption** pode ser usado para alterar o horário de início e o intervalo do backup completo enquanto o banco de dados está em execução. A sintaxe geral para alterar o horário de início do backup completo é:

```
CALL SETSYSTEMOPTION('fbktm','StartTime')
```

A sintaxe geral para alterar o intervalo do backup completo é:

```
CALL SETSYSTEMOPTION('fbktv','Interval')
```

Onde **StartTime** é o horário para iniciar o primeiro backup completo, com o formato **YY/MM/DD HH:MM**. **Interval** é o intervalo de tempo em que os backups completos ocorrem, com o formato **D-HH:MM**.

Quando o **Backup Server** está ativado, chame o procedimento armazenado **SetSystemOption** para iniciar um backup completo.

```
dmSQL> CALL SETSYSTEMOPTION('STARTBACKUP','1');
```

Para definir o intervalo do backup completo para 1 hora, insira a seguinte linha no prompt de comando do **dmSQL**:

```
dmSQL> CALL SETSYSTEMOPTION('fbktv','0-1:00:00');
```

USING JSERVER MANAGER TO SET FULL BACKUP SCHEDULE

Você pode definir o cronograma de backup completo com o **JServer Manager** usando a ferramenta de configuração de início do banco de dados. O **JServer Manager** irá alterar automaticamente os valores das palavras-chave **DB_FBkTm** e **DB_FBkTv** no arquivo **dmconfig.ini**. Na próxima vez que você iniciar o banco de dados, o **Backup Server** usará essas configurações como o novo cronograma de backup completo. Para instruções sobre como definir o cronograma de backup

completo usando o **JServer Manager**, consulte o **Guia do Usuário do JServer Manager**.

Backup Mode of File Objects

O Modo de Backup de Objetos de Arquivo permite que o administrador do banco de dados decida se o **Backup Server** fará o backup de objetos de arquivo durante um backup completo. Também é possível especificar ao **Backup Server** para fazer o backup apenas de objetos de arquivo do sistema ou de objetos de arquivo do sistema e do usuário.

É possível definir o Modo de Backup de Objetos de Arquivo de várias maneiras. A palavra-chave de configuração **DB_BkFoM** determina a configuração durante a inicialização do banco de dados, mas também pode ser modificada durante o tempo de execução com o **dmSQL** ou a utilidade **JServer Manager**.

O **Backup Server** moverá todos os arquivos do backup anterior para o diretório de backup antigo especificado por **DB_BkOdr**.

Iniciar o backup de objetos de arquivo fará com que o banco de dados precise de mais tempo para concluir um backup completo, dependendo de quantos objetos de arquivo estão no banco de dados. O custo total de um backup completo inclui: (1) copiar o backup completo anterior, se **DB_BkOdr** estiver definido; (2) copiar todos os arquivos do banco de dados; (3) copiar todos os arquivos de diário; e (4) copiar todos os objetos de arquivo, se **DB_BkFoM** estiver definido. Certifique-se de que haja espaço em disco suficiente disponível no diretório de backup especificado por **DB_BkDir** (e **DB_BkOdr**, se aplicável) para todos os arquivos de backup mencionados, a fim de evitar falhas no backup.

Os objetos de arquivo são copiados em um diretório **FO** que é criado no diretório de backup no momento em que um backup completo é realizado. Os objetos de arquivo são renomeados sequencialmente quando são copiados para o diretório de objetos de arquivo respaldados. Os arquivos no subdiretório **/FO** são renomeados começando com as letras **FO**, seguidas de um número de série de dez dígitos. Todos os objetos de arquivo de backup são anexados com a extensão de arquivo **.BAK**. O mapeamento entre o nome e o caminho do arquivo de origem e o nome do arquivo de backup é registrado no arquivo de mapeamento de objetos de arquivo **dmFoMap.his**.

BACKUP FILE OBJECT MAPPING FILE

O arquivo de mapeamento de objetos de arquivo **dmFoMap.his** é criado no diretório **DB_BkDir/FO**. Trata-se de um arquivo de texto puro em ASCII que registra o nome

do arquivo externo original e o nome do arquivo de backup. O formato é semelhante a:

```
Database Name: DBSAMPLE5
Begin Backup FO Time: 2013/04/12 09:21:32
FO Backup Directory:
C:\\DBMaker\\5.4\\SAMPLES\\DATABASE\\backup\\FO\\
[Mapping List]
s, fo0000000000.bak,
"C:\\DBMaker\\5.4\\SAMPLES\\DATABASE\\backup\\FO\\ZZ000001.bmp"
u, fo0000000001.bak,
"C:\\DBMaker\\5.4\\SAMPLES\\DATABASE\\backup\\FO\\image.jpg"
...
s, fo0000002345.bak,
"C:\\DBMaker\\5.4\\SAMPLES\\DATABASE\\backup\\FO\\ZZ00AB32.txt"
```

O conteúdo antes de "[Mapping List]" é apenas uma descrição para referência do usuário. Cada linha após "[Mapping List]" representa um registro que mostra o tipo de objeto de arquivo (s = objeto de arquivo do sistema, u = objeto de arquivo do usuário), o novo arquivo no subdiretório **/fo** e seu nome e caminho de arquivo originais. Este arquivo de mapeamento é necessário para a restauração dos objetos de arquivo.

USING DMCONFIG.INI TO SET BACKUP MODE OF FILE OBJECTS

A configuração do arquivo de parâmetros de backup de objetos de arquivo é determinada pela palavra-chave **DB_BkFoM** no arquivo **dmconfig.ini**:

- **DB_BkFoM = 0**: Não fazer backup de objetos de arquivo.
- **DB_BkFoM = 1**: Fazer backup apenas de objetos de arquivo do sistema.
- **DB_BkFoM = 2**: Fazer backup de objetos de arquivo do sistema e do usuário.

Se **DB_BkFoM = 1** ou **2**, o servidor de backup copiará todos os objetos de arquivo para o subdiretório **/fo** sob o diretório de backup. O agendamento segue o cronograma de backup completo.

Exemplo:

Um exemplo de entrada no arquivo **dmconfig.ini** para especificar os parâmetros de backup de objetos de arquivo poderia ser:

```
[MyDB]
DB_BkSvr = 1 ; starts the backup server
DB_FBkTm = 01/05/01 00:00:00 ; begins at midnight, May 1, 2001
```

```
DB_FBkTv = 1-00:00:00 ; interval is once every day
DB_BkDir = /home/dbmaker/backup ; backup directory
DB_BkFoM = 2 ; backup both system and user file objects
```

Como o modo de backup é 2, o servidor de backup copiará todos os arquivos externos (objetos de arquivo do usuário) e os objetos de arquivo do sistema para o diretório **/home/dbmaker/backup/FO**. Se o subdiretório **FO** não existir, o servidor de backup o criará automaticamente.

USING DMSQL TO SET BACKUP MODE OF FILE OBJECTS

O procedimento armazenado **SetSystemOption** pode ser usado para alterar o modo de backup dos objetos de arquivo enquanto o banco de dados está em execução. A sintaxe geral para mudar o Modo de Backup dos Objetos de Arquivo é:

```
CALL SETSYSTEMOPTION('bkfom','n')
```

Onde **n** é 0, 1 ou 2. Definir **n** como 0 desativará o Modo de Backup dos Objetos de Arquivo. Definir **n** como 1 configura o servidor de backup para fazer backup de todos os objetos de arquivo do sistema durante um backup completo. Definir **n** como 2 configura o servidor de backup para fazer backup de todos os objetos de arquivo do sistema e do usuário durante um backup completo.

Exemplo:

Para configurar o Backup Server para realizar um backup completo de todos os objetos de arquivo de usuário e do sistema, insira a seguinte linha no prompt de comando do dmSQL:

```
dmSQL> CALL SETSYSTEMOPTION('bkfom','2');
```

USING JSERVER MANAGER TO SET BACKUP MODE OF FILE OBJECTS

As configurações sob o Modo de Backup de Objetos de Arquivo afetam como os objetos de arquivo são copiados durante o processo de backup completo. Selecionar **Não Fazer Backup de Objetos de Arquivo** desativa o backup de arquivos durante o processo de backup completo. Selecionar **Fazer Backup Apenas de Objetos de Arquivo do Sistema** faz backup apenas de objetos de arquivo do sistema durante backups completos automáticos. Selecionar **Fazer Backup de Objetos de Arquivo**

do Sistema e do Usuário fará backup tanto dos objetos de arquivo do sistema quanto dos objetos de arquivo do usuário, que serão copiados para o diretório de backup durante os backups completos automáticos. Para instruções sobre como definir o Modo de Backup de Objetos de Arquivo durante a inicialização do banco de dados ou com o diálogo de Configurações em Tempo de Execução no JServer Manager, consulte o Guia do Usuário do JServer Manager.

Backup Mode of Stored Procedures

O Modo de Backup de procedimentos armazenados permite que o administrador do banco de dados decida se o Backup Server fará o backup de procedimentos armazenados ESQL e JAVA durante um backup completo.

É possível definir o Modo de Backup de procedimentos armazenados de várias maneiras. A palavra-chave de configuração **DB_BkSPm** determina a configuração durante a inicialização do banco de dados, mas também pode ser modificada durante a execução com **dmSQL** ou a utilidade **JServer Manager**.

O Backup Server moverá o backup anterior de procedimentos armazenados para o diretório de backup antigo especificado por **DB_BkOdr**. Iniciar o backup de procedimentos armazenados fará com que o banco de dados precise de mais tempo para concluir um backup completo, dependendo de quantos objetos de arquivo estão no banco de dados. O custo total de um backup completo inclui:

1. Copiar o backup completo anterior se **DB_BkOdr** estiver definido;
2. Copiar todos os arquivos do banco de dados;
3. Copiar todos os arquivos de log;
4. Copiar todos os objetos de arquivo se **DB_BkFoM** estiver definido;
5. Copiar procedimentos armazenados ESQL e JAVA se **DB_BkSPm** estiver definido.

Além disso, é importante garantir que haja espaço suficiente em disco no diretório de backup especificado por **DB_BkDir** para todos os arquivos de backup, a fim de evitar falhas no backup.

Os procedimentos armazenados são copiados para o subdiretório **SP**, que é criado no diretório de backup no momento em que um backup completo é realizado. Os procedimentos armazenados são renomeados ao serem copiados para o diretório para procedimentos armazenados em backup. As informações de backup sobre os procedimentos armazenados copiados são registradas no arquivo **dmSpBk.his**.

BACKUP STORED PROCEDURE INFORMATION FILE

As informações de backup são compostas por três partes: o nome do banco de dados, o horário do backup e a lista de backup que registra quais linhas foram copiadas. O arquivo de informações de backup **dmSpBk.his** é criado no subdiretório **SP** sob o diretório especificado por **DB_BkDir**. Ele é um arquivo de texto puro em ASCII que registra informações de backup sobre os procedimentos armazenados copiados. O formato do arquivo é semelhante ao seguinte:

```
Database Name: MYDB
Begin Backup SP time: 2014/06/20 09:13:06
[Backup List]
ESQLSP, SYSADM, A4, A4SYSADM.dll, A4SYSADM.ec
....
JAR, "", "", employee.jar, ""
```

O conteúdo antes de "[Backup List]" serve apenas como uma descrição para referência do usuário. Cada linha após "[Backup List]" representa um registro que mostra o tipo de procedimento armazenado, o proprietário do procedimento armazenado, o nome do procedimento armazenado, o nome dos arquivos de objeto dos procedimentos armazenados e o nome dos seus respectivos arquivos de origem. Este arquivo de informações de backup é necessário para a restauração dos procedimentos armazenados.

USING DMCONFIG.INI TO SET BACKUP MODE OF STORED

PROCEDURES

A configuração da palavra-chave **DB_BkSPm** determina o modo de backup dos procedimentos armazenados:

- **DB_BkSPm = 0:** Não realizar backup de procedimentos armazenados ESQL e JAVA.
- **DB_BkSPm = 1:** Realizar backup de procedimentos armazenados ESQL e JAVA.

Exemplo:

Uma entrada no arquivo **dmconfig.ini** para especificar os parâmetros de backup dos procedimentos armazenados pode se parecer com o seguinte:

```
[MyDB]
DB_BkSvr = 1 ; starts the backup server
DB_FBkTm = 14/05/01 00:00:00 ; begins at midnight, May 1, 2014
DB_FBkTv = 1-00:00:00 ; interval is once every day
DB_BkDir = /home/dbmaker/backup ; backup directory
DB_BkSPm = 1 ; backup ESQL stored procedures and JAVA
stored procedures
```

Como o valor de **DB_BkSPm** é 1, o Backup Server fará o backup dos procedimentos armazenados ESQL e JAVA no subdiretório **SP** sob o diretório especificado por **DB_BkDir**. Se o subdiretório **SP** não existir, o Backup Server o criará automaticamente. Isso garante que todos os procedimentos armazenados sejam copiados e possam ser restaurados quando necessário.

USING DMSQL TO SET BACKUP MODE OF STORED PROCEDURES

A stored procedure **SetSystemOption** pode ser utilizada para alterar o modo de backup dos procedimentos armazenados enquanto o banco de dados está em execução. A sintaxe geral para mudar o modo de backup dos procedimentos armazenados é:

```
CALL SETSYSTEMOPTION('BKSPM', 'n')
```

Os usuários podem atribuir **0** ou **1** a **n**. Se **0** for atribuído a **n**, o Backup Server não fará backup de procedimentos armazenados ESQL e JAVA; se **1** for atribuído a **n**, o Backup Server fará backup de procedimentos armazenados ESQL e JAVA durante um backup completo. Essa configuração é útil para gerenciar quais componentes do banco de dados devem ser incluídos no processo de backup, permitindo uma maior flexibilidade e eficiência na administração do banco de dados. Exemplo: To configure the Backup Server to perform a full backup on all stored procedures, you would enter the following command at the **dmSQL** command prompt:

```
dmSQL> CALL SETSYSTEMOPTION('BKSPM', '1');
```

USING JSERVER MANAGER TO SET BACKUP MODE OF STORED PROCEDURES

Independentemente do status online de um banco de dados, você pode habilitar o backup dinâmico de procedimentos armazenados ESQL e procedimentos armazenados JAVA usando a ferramenta gráfica JServer Manager. O JServer

Manager altera automaticamente o valor da palavra-chave DB_BkSPm no arquivo de configuração dmconfig.ini. Se o banco de dados estiver offline, na próxima vez que você iniciar o banco de dados, essa nova configuração entrará em vigor. Para obter instruções sobre como definir o Modo de Backup de procedimentos armazenados durante a inicialização do banco de dados ou com a caixa de diálogo de Configurações de Tempo de Execução no JServer Manager, consulte o Guia do Usuário do JServer Manager.

Inactivate Backup Server

O DBMaker iniciará automaticamente o Backup Server ao iniciar o banco de dados. O Backup Server está desativado por padrão. Você pode controlar o estado do Backup Server com o DB_BkSvr. Quando o DB_BkSvr está definido como 0, o Backup Server está inativo; quando o DB_BkSvr está definido como 1, o Backup Server está ativo. Quando você não quiser mais que o Backup Server esteja ativo, pode definir o valor da palavra-chave DB_BkSvr como 0 no arquivo dmconfig.ini ou mudar BkSvr com a chamada `setsystemoption('bksvr','0')` após o banco de dados ser iniciado.

USING DMCONFIG.INI TO INACTIVATE BACKUP SERVER

Se o banco de dados estiver offline, você pode desativar o Backup Server diretamente usando a palavra-chave DB_BkSvr no arquivo dmconfig.ini. Na próxima vez que você iniciar o banco de dados, o Backup Server não será iniciado. Se o banco de dados estiver online, alterar o valor da palavra-chave DB_BkSvr não terá efeito até que o banco de dados seja desligado e reiniciado.

Para desativar o Backup Server usando o arquivo dmconfig.ini:

1. Abra o arquivo dmconfig.ini no servidor do banco de dados usando qualquer editor de texto ASCII.
2. Localize a seção de configuração do banco de dados para alterar o modo de backup.
3. Altere o valor da palavra-chave DB_BkSvr para 0 para desativar o Backup Server.
4. Reinicie o banco de dados.

USING DMSQL TO INACTIVATE BACKUP SERVER

A stored procedure `SetSystemOption` pode ser usada para alterar o estado do Backup Server enquanto o banco de dados está em execução. A sintaxe geral para mudar o estado do Backup Server é:


```
CALL SETSYSTEMOPTION('bksvr','n')
```

Onde **n** é 0 ou 1. Configurar **n** para 0 desativará o Backup Server. Configurar **n** para 1 ativará o Backup Server.

Exemplo:

Para desativar o Backup Server, insira a seguinte linha no prompt de comando do dmSQL:

```
dmSQL> CALL SETSYSTEMOPTION('bksvr','0');
```

USING JSERVER MANAGER TO INACTIVATE BACKUP SERVER

Se o banco de dados estiver offline, você pode desativar o Backup Server usando a interface gráfica do JServer Manager. O JServer Manager mudará automaticamente o valor da palavra-chave **DB_BkSvr** no arquivo de configuração **dmconfig.ini**. Na próxima vez que você iniciar o banco de dados, o Backup Server não será iniciado. Se o banco de dados estiver online, desativar o Backup Server não terá efeito até que o banco de dados seja desligado e reiniciado. Para instruções sobre como desativar o Backup Server usando o JServer Manager, consulte o Manual do Usuário do JServer Manager.

15.7 Backup History Files

Os backups automáticos usando o Backup Server podem armazenar informações sobre quais arquivos de diário foram copiados, quando foram copiados e onde os arquivos de backup estão localizados no arquivo de histórico de backup de forma automática.

Locating the Backup History File

O arquivo de histórico de backup é um arquivo de texto localizado no primeiro diretório especificado pela palavra-chave **DB_BkDir** no arquivo **dmconfig.ini**. Este arquivo é criado no caminho de backup online e é nomeado como **dmBackup.his**. O arquivo será utilizado automaticamente durante a restauração de um banco de dados, enquanto o backup offline é registrado com **offBackup.his**.

Understanding the Backup History File

Os arquivos de histórico de backup contêm todas as informações referentes ao número de identificação, nomes de arquivos e a hora e data em que os backups foram realizados. O DBMaker utiliza o arquivo de histórico de backup para rastrear sequências de backup e garantir a consistência dos backups completos, diferenciais e incrementais dentro de cada sequência. A seguir está o formato do arquivo de histórico de backup:

```
<backup_id>: file_name -> archive_file_name, time, event
```

Isso indica que um arquivo chamado `file_name` foi copiado para um arquivo de arquivo chamado `archive_file_name` no momento devido ao evento. O evento é uma string de texto que indica a razão do backup. Essa string pode ser JOURNAL-FULL, TIME-OUT, ON-LINE-FULL-BACKUP-BEGIN, ON-LINE-FULL-BACKUP ou ON-LINE-FULL-BACKUP-END. A string JOURNAL-FULL indica que um backup incremental foi realizado porque o diário estava cheio. A string TIME-OUT indica que um backup diferencial ou incremental foi realizado porque o intervalo de backup agendado expirou. A string ON-LINE-FULL-BACKUPxxxx significa que é um backup completo.

Using the Backup History File

Se o diário cheio ocorrer com frequência, diminua a porcentagem de diário de backup cheia ou reduza o intervalo de tempo. Além disso, descubra se o intervalo de backup é muito curto verificando o arquivo de histórico de backup. Se o mesmo arquivo de diário for copiado consecutivamente no arquivo de histórico de backup, o intervalo de tempo pode ser muito curto. Essa situação desperdiçará espaço em disco, pois cada arquivo pode conter apenas alguns blocos alterados. Para evitar isso, habilite o modo de backup compacto ou aumente o intervalo de tempo de backup. Se muitos arquivos de diário forem copiados toda vez, isso pode significar que o intervalo de tempo é muito longo. Essa situação é mais perigosa devido à possibilidade de perder mais dados quando um disco falha. Para evitar isso, os usuários devem reduzir o intervalo de tempo de backup. Realizar backups completos regularmente reduzirá o tempo de recuperação após falhas de mídia, mesmo ao usar o Backup Server. Isso também reduz a quantidade de armazenamento de backup necessário.

Understanding the File Object Backup History File

O arquivo de histórico de objetos de backup, `dmFoMap.his`, mantém um registro de todos os objetos de arquivo que foram copiados, configurando o parâmetro de configuração de backup de objetos de arquivo. O `dmFoMap.his`, localizado no diretório

<DB_BkDir>\FO, é um arquivo de texto ASCII puro que registra o nome original do arquivo externo e o nome do arquivo de backup. A seguir está o formato do arquivo:

```
Database Name: MYDB
Begin Backup FO Time: 2001.5.13 2:33
FO Backup Directory: /DBMaker/mydb/backup/FO (i.e. DB_BkDir/FO)
[Mapping List]
s, fo0000000000.bak, "/DBMaker/mydb/fo/ZZ000001.bmp"
u, fo0000000001.bak, "/home2/data/image.jpg"
....
s, fo0000002345.bak, "/DBMaker/mydb/fo/ZZ00AB32.txt"
```

Na primeira coluna, 's' ou 'u' representam, respectivamente, objetos de arquivo do sistema ou do usuário. A segunda coluna fornece o nome do backup, e a terceira coluna fornece o nome completo e o caminho do objeto de arquivo original.

Understanding the Stored Procedure Backup History File

O arquivo de histórico de backup de procedimentos armazenados, dmSpBk.his, mantém um registro de procedimentos armazenados em ESQL e JAVA que foram copiados configurando o parâmetro de configuração de backup de procedimentos armazenados. O dmSpBk.his, localizado no subdiretório SP sob o diretório especificado por DB_BkDir, é um arquivo de texto ASCII puro que registra informações de backup de procedimentos armazenados em ESQL e JAVA. A seguir está o formato do arquivo:

```
Database Name: MYDB
Begin Backup SP time: 2014/06/20 09:13:06
[Backup List]
ESQLSP, SYSADM, A4, A4SYSADM.dll, A4SYSADM.ec
....
JAR, "", "", employee.jar, ""
```

A primeira coluna indica o tipo de procedimentos armazenados; a segunda coluna indica o proprietário do procedimento armazenado; a terceira coluna indica o nome dos procedimentos armazenados; e a quarta e a quinta colunas indicam, respectivamente, o nome dos arquivos de objeto dos procedimentos armazenados e o nome de seus arquivos-fonte correspondentes.

15.8 Backup on Replication Databases

Em ambos os bancos de dados normais e mestre, mas não no banco de dados escravo, os usuários podem fazer backup completo, backup diferencial e backup incremental. O método é o mesmo de antes. No entanto, o JServerManager não pode fazer backup incremental interativamente no banco de dados mestre. Além disso, ele não pode limpar os arquivos de backup incremental ao fazer o backup completo interativamente no banco de dados mestre.

Observe que, em um banco de dados mestre, potencialmente, muitos arquivos de backup incremental anteriores a um backup completo ainda permanecem na sequência de backup para replicação. Enquanto isso, o servidor de replicação pode não limpar os arquivos de backup incremental por causa do backup completo, então talvez um grande número de arquivos exista em DB_BkDir se o próximo backup completo não for feito em um longo período de tempo.

Em resumo, o servidor de replicação deve cooperar bem com o servidor de backup, e eles não podem atrapalhar um ao outro. Por um lado, o backup não deve prejudicar a replicação, ou seja, o servidor de replicação deve sempre ser capaz de replicar todas as transações para os sites escravos, independentemente de um backup completo ou um backup diferencial ter sido feito ou estar sendo feito. Por outro lado, a replicação não pode prejudicar a sequência de backup.

É possível restaurar o banco de dados mestre com a sequência de backup. No entanto, após o banco de dados mestre ser restaurado, a replicação do banco de dados não continuará. Se os usuários quiserem continuar replicando o banco de dados, todos os bancos de dados escravos devem ser substituídos pelo novo banco de dados mestre, ou seja, os usuários devem copiar os arquivos do banco de dados mestre para substituir todos os arquivos dos bancos de dados escravos.

Existem algumas restrições para backup em banco de dados de replicação:

- Quando o banco de dados mestre for iniciado, BMODE e BKSVR devem estar ativados.
- BMODE, BKSVR, BKDIR não podem ser alterados durante o tempo de execução tanto no banco de dados mestre quanto no banco de dados escravo. Por exemplo, chamar `setsystemoption('bkdir', 'novo_bkdir')` retornará um erro.
- Tanto no banco de dados mestre quanto no banco de dados escravo, a palavra-chave DB_BkDir no arquivo `dmconfig.ini` deve ser um único caminho. Se os usuários definirem a palavra-chave DB_BkDir como multi-caminho, apenas o primeiro caminho será usado e o tamanho do caminho será ignorado.

- Em um banco de dados mestre, é desativado fazer backup incremental interativamente pelo JServerManager.
- Em um banco de dados escravo, é desativado fazer backup completo, backup diferencial e backup incremental.

15.9 Recovery Options

Restaurar um banco de dados recria o banco de dados como ele existia no momento do backup completo mais recente, mais as alterações aplicadas pelos arquivos de diário de backup.

Analyzing Recovery Options

Quais operações de recuperação estão disponíveis? A resposta a esta pergunta é determinada pelo fato de o banco de dados estar ou não em modo BACKUP.

- Se o banco de dados estiver operando em modo NONBACKUP, a única opção para restauração após uma falha de disco é restaurar o backup completo mais recente e reiniciar o banco de dados. Todo o trabalho realizado desde o último backup completo será perdido e deverá ser refeito. Se este for o caso, não há necessidade de responder às perguntas seguintes.
- Se o banco de dados estiver operando em modo BACKUP (BACKUP-DATA ou BACKUP-DATA-AND-BLOB), várias opções de recuperação estão disponíveis para reconstruir o banco de dados danificado.

Preparing for Restoratio

Antes de restaurar um banco de dados após um erro de disco, responda às seguintes perguntas:

- **Para qual ponto no tempo você deseja restaurar o banco de dados?**

Se sua resposta for o momento em que o erro de disco ocorreu, faça o backup de todos os arquivos de diário do banco de dados danificado. Esses arquivos ajudarão o DBMaker a restaurar o banco de dados ao tempo mais atual.

- **Quais arquivos foram previamente copiados?**

Descubra onde o backup completo mais recente e todos os backups diferenciais e incrementais subsequentes estão localizados. Por exemplo, suponha que você realize um backup completo no dia 30 de cada mês, um backup diferencial a cada dia 15 e um backup incremental a cada 10 dias. Se seu sistema for danificado em 25 de maio,

you will need the full backup of 30 de abril, the differential backup of 15 de maio and the incremental backups of 10 and 20 de maio, in addition to the log backup files of 25 de maio. After locating these files, the DBMaker can restore your database to the state it was in before the failure on 25 de maio. A sequence of backup files, composed of a full backup file and a series of differential and incremental backup files, is essential for the restoration. The online backup sequence is identified by a file named `dmbackup.his`, and the offline backup sequence is identified by a file named `offbackup.his`. This makes the log backup file especially important, as the DBMaker uses it to obtain these details when restoring a database.

Performing a Restoration

When executing the restoration process, the DBMaker will perform the following actions:

- Copy all backup files, including data files, BLOB files, and log files, to the directory specified by the keyword `DB_DbDir` in `dmconfig.ini`. This operation will overwrite the original files in the database. Therefore, it is strongly recommended that the user manually copy the original files to another location before running the restoration tools, at least ensuring that the log files are saved, to ensure that, in the event of a restoration failure, there is another chance to restore the database to the most current state.
- Apply the differential or incremental backup files, or both, to the database.

When using the restoration tools, users can specify:

- If you want to restore the database files to the `dmconfig.ini` file in the system. To restore it, specify the full path to the `dmconfig.ini` file to be restored.
- If you want to use a backup sequence to restore the database to a different location than the original, modify the keywords in the path to the data files, including `DB_DbDir`, `DB_DbFil`, `DB_UsrDb`, and so on. If the backup sequence is moved to another location or computer, consider the following when restoring the database:
 - The names of the databases in `dmconfig.ini` must be consistent with the name of the database backup.
 - Define the keywords `BKDIR` and others for data files and BLOB files as necessary.

- O valor de DB_JnFil deve ser definido se houver mais de um arquivo de diário. Certifique-se de que esse valor corresponda ao número de arquivos de diário no banco de dados de backup.
- Se os arquivos de backup estiverem localizados em várias pastas, a palavra-chave DB_BkDir deve ser configurada para incluir todas as pastas onde os arquivos de backup estão localizados. O arquivo dmbbackup.his deve estar localizado no primeiro BKDIR.

NOTA: Os usuários podem copiar um arquivo de configuração dmconfig.ini existente da pasta onde a sequência de backup existia e, em seguida, configurar um novo arquivo dmconfig.ini modificando as palavras-chave relevantes.

- **Caminho completo do arquivo de histórico de backup** dmBackup.his ou offBackup.his, caso o dmBackup.his ou offBackup.his não esteja localizado no diretório padrão.
- **Hora de restauração (RTime).** RTime indica a que hora o banco de dados deve ser restaurado; isso determinará se a sequência de backup atual está disponível ou não, e quais arquivos de backup diferencial e incremental serão aplicados ao banco de dados. O usuário pode especificá-lo nas ferramentas de restauração ou adicionar a palavra-chave DB_RTime no sistema dmconfig.ini ou no arquivo dmconfig.ini de backup que será restaurado. Se RTime não for especificado, o valor padrão é a hora atual.

O DBMaker fornece dois métodos para realizar a restauração: um é pelo **JServer Manager Tool** e o outro pelo **comando de linha Rollover**. Para mais informações sobre o uso do JServer Manager, consulte o **Guia do Usuário do JServer Manager**. E para mais informações sobre o uso da ferramenta de linha de comando Rollover, consulte a próxima seção, '**usar rollover para restaurar o banco de dados**'.

Restoring database by Rollover

O usuário também pode usar o **Rollover**, que é uma ferramenta de linha de comando para restaurar o banco de dados. Seu princípio é o mesmo que o da Restauração de Banco de Dados do JServer Manager. O uso do Rollover é o seguinte:

```
rollover database_name [-C] [-i inifile] [-r rtime] [-h hisfile] [-m foMapfile]
[-t SpResTyp] [-f FOtype] [-p restoreMode] [-B bkDir] [-D newDbdir]
[-M restoreType] [-s sysFoDir] [-u userFodir] [-y]
```

Existem quatorze parâmetros opcionais entre colchetes:

- **C:** Especifica que não deve copiar o arquivo de backup completo para restaurar; é utilizado quando o tipo de arquivo não é comprimido.
- **i:** Especifica o caminho completo do arquivo dmconfig.ini. Se o usuário especificar o dmconfig.ini a ser restaurado, o Rollover substituirá a seção do banco de dados no sistema dmconfig.ini pela seção correspondente no dmconfig.ini especificado; caso contrário, o DBMaker não restaurará o dmconfig.ini.
- **r:** Indica a hora à qual o banco de dados deve ser restaurado. A opção **-r** é o primeiro método para especificar o RTime; o segundo método é adicionar a palavra-chave **DB_RTime** no sistema dmconfig.ini ou no arquivo dmconfig.ini de backup que será especificado para restaurar o banco de dados. Se nem a opção **-r** nem a palavra-chave **DB_RTime** forem fornecidas, o RTime será a hora atual.
- **h:** Fornece o caminho completo do arquivo dmBackup.his ou offBackup.his. O padrão é **DB_BkDir\dmBackup.his** ou **DB_BkDir\offBackup.his**.
- **m:** Fornece o caminho completo do arquivo dmFoMap.his. O padrão é **DB_BkDir\FO\dmFoMap.his**.
- **f:** Especifica qual tipo de arquivos FO devem ser restaurados. Existem quatro valores: o valor 0 significa que nenhum arquivo FO será restaurado; o valor 1 restaurará o FO do sistema; o valor 2 restaurará o FO do usuário; e o valor 3 restaurará todos os FOs. O valor padrão é 3.
- **t:** Especifica se as stored procedures devem ser restauradas. Existem dois valores: se o valor for definido como 0, nenhuma stored procedure será restaurada; se definido como 1, todas as stored procedures serão restauradas. O valor padrão é 1.
- **p:** Especifica o modo de restauração do banco de dados. O valor 0 significa restaurar o banco de dados no modo antigo; o valor 1 significa restaurar o banco de dados com a nova funcionalidade; e o valor 2 significa descompactação paralela.
- **B:** Fornece o caminho completo do diretório de backup onde a sequência de backup reside. O usuário pode especificar múltiplos caminhos, por exemplo, **-B "dir1" -B "dir2"**.
- **D:** Fornece um novo diretório para restaurar o banco de dados.
- **M:** Indica o tipo de restauração: master/slave/bkcrash (restaurar o slave a partir do master).
- **s:** Fornece um novo diretório para restaurar o FO do sistema. Após a restauração, é necessário executar **mvsysfo.sql** para corrigir **SYSFILEOBJ**.
- **u:** Fornece um novo diretório para restaurar o FO do usuário. Após a restauração, é necessário executar **mvusrfo.sql** para corrigir **SYSFILEOBJ**.
- **y:** Fornece "YES" em vez de solicitar a confirmação do usuário.

