



Nota: os textos deste documento foram extraídos dos Manuais On-line do SqlServer 2008 (2008).

Nível de Isolamento de Transação.

O nível de isolamento de transação controla o comportamento de bloqueio e controle de versão de linha das instruções Transact-SQL emitidas por uma conexão com SQL Server. Principais argumentos para o *set transaction isolation level*:

READ UNCOMMITTED

Especifica que as instruções podem ler linhas que foram modificadas por outras transações, mas que ainda não foram confirmadas.

Transações com execução em nível READ UNCOMMITTED não emitem bloqueios compartilhados para impedir outras transações de modificar os dados lidos pela transação atual. Transações READ UNCOMMITTED também não são bloqueadas por bloqueios exclusivos que impediriam a transação atual de ler linhas que foram modificadas, mas não confirmadas, por outras transações. Quando esta opção é definida, é possível ler modificações não confirmadas, chamadas de **leituras sujas**. Os valores nos dados podem ser alterados e linhas podem aparecer ou desaparecer no conjunto de dados antes do término da transação. Essa opção tem o mesmo efeito de definir NOLOCK em todas as tabelas em todas as instruções SELECT em uma transação. Esse é o **menos** restritivo dos níveis de isolamento.

READ COMMITTED

Especifica que as instruções não podem ler dados que foram modificados, mas ainda não confirmados por outras transações. Isto previne leituras sujas. Dados podem ser alterados através de outras transações entre instruções individuais dentro da transação atual, resultando em leituras não repetíveis ou dados fantasmas. Esta opção é o padrão do SQL Server.

REPEATABLE READ

Especifica que as instruções não podem ler dados que foram modificados, mas ainda não confirmados por outras transações e que nenhuma outra transação pode modificar dados que foram lidos pela transação atual até que esta seja concluída.

Bloqueios compartilhados são colocados em todos os dados lidos por cada instrução na transação, sendo mantidos até que a transação seja concluída. Isso impede outras transações de modificar qualquer linha que tenha sido lida pela transação atual. Outras transações **podem inserir** novas linhas que correspondam às condições de pesquisa das instruções emitidas pela transação atual.

SERIALIZABLE

Especifica o seguinte:

- As instruções não podem ler dados que foram modificados, mas ainda não confirmados por outras transações.
- Nenhuma outra transação pode modificar dados lidos pela transação atual até que esta seja concluída.
- Outras transações não podem inserir linhas novas com valores chave que estão no intervalo de chaves lido por alguma instrução da transação atual até que esta seja concluída.

Bloqueios de intervalo são colocados no intervalo de valores chave que corresponde às condições de pesquisa de cada instrução executada em uma transação. Isso bloqueia que outras transações atualizem ou insiram alguma linha que possa se qualificar para alguma das instruções executadas pela transação atual. Isto significa que, se alguma das instruções de uma transação for executada uma segunda vez, ela lerá o mesmo conjunto de linhas. Os bloqueios de intervalo são mantidos até que a transação seja concluída. Esse é o mais restritivo dos níveis de isolamento, pois ele bloqueia intervalos de chaves inteiros até que a transação seja concluída. Como a simultaneidade é menor, use essa opção apenas quando necessário. Essa opção tem o mesmo efeito de definir **HOLDLOCK** em todas as tabelas em todas as instruções **SELECT** em uma transação.

SNAPSHOT

Especifica que os dados lidos por qualquer instrução em uma transação serão a versão transacionalmente consistente que existia no início da transação. A transação pode reconhecer apenas modificações de dados que estavam confirmadas antes do início da transação. Modificações de dados efetuadas por outras transações após o início da transação atual não são visíveis às instruções em execução na transação atual. O efeito será como se as instruções em uma transação obtivessem um instantâneo dos dados confirmados conforme eles se encontravam no início da transação.

Exceto quando um banco de dados está sendo recuperado, transações **SNAPSHOT** não requerem bloqueios ao ler dados. Transações **SNAPSHOT** que lêem dados não bloqueiam outras transações de gravar dados. Transações que gravam dados não bloqueiam transações **SNAPSHOT** de ler dados.

Durante a fase de reversão de uma recuperação de banco de dados, as transações **SNAPSHOT** solicitarão um bloqueio se houver uma tentativa de ler dados que se encontram bloqueados por outra transação que está sendo revertida. A transação **SNAPSHOT** será bloqueada até que aquela transação seja revertida. O bloqueio será liberado tão logo seja permitido.

A opção de banco de dados **ALLOW_SNAPSHOT_ISOLATION** deve ser definida como **ON** para que uma transação que usa o nível de isolamento **SNAPSHOT** seja iniciada. Se uma transação que usa o nível de isolamento **SNAPSHOT** acessar dados em vários bancos de dados, **ALLOW_SNAPSHOT_ISOLATION** deve ser definido como **ON** em cada banco de dados. Por exemplo:

```
ALTER DATABASE pesquisa_qualidade  
SET ALLOW_SNAPSHOT_ISOLATION ON
```

```
ALTER DATABASE pesquisa_qualidade  
SET READ_COMMITTED_SNAPSHOT ON;
```

Uma transação iniciada com outro nível de isolamento não pode ser definida com o nível de isolamento SNAPSHOT; isso causaria a anulação da transação. Se uma transação for iniciada no nível de isolamento SNAPSHOT, você poderá alterar seu nível de isolamento e retorná-la novamente para SNAPSHOT. Uma transação é iniciada na primeira vez em que ela acessa dados.

Uma transação em execução com nível de isolamento SNAPSHOT pode exibir as alterações por ela efetuadas. Por exemplo, se a transação executar um UPDATE em uma tabela e, em seguida, emitir uma instrução SELECT contra a mesma tabela, os dados modificados serão incluídos no conjunto de resultados.

Comentários

Apenas uma única opção de nível de isolamento pode ser definida por vez, permanecendo definida para aquela conexão até que seja explicitamente alterada. Todas as operações de leitura executadas na transação operam sob as regras do nível de isolamento especificado, a menos que uma dica de tabela na cláusula FROM de uma instrução especifique comportamento de bloqueio ou controle de versão diferente para uma tabela.

Os níveis de isolamento da transação definem o tipo de bloqueio adquirido em operações de leitura. Bloqueios compartilhados adquiridos para READ COMMITTED ou REPEATABLE READ geralmente são bloqueios de linha, embora os bloqueios de linha possam ser escalados para bloqueios de página ou tabela se um número significativo de linhas em uma página ou tabela forem consultadas pela leitura. Se uma linha for modificada pela transação depois de ter sido lida, a transação irá adquirir um bloqueio exclusivo para proteger essa linha, sendo mantido até que a transação seja concluída. Por exemplo, se uma transação REPEATABLE READ tiver um bloqueio compartilhado em uma linha e a transação modificar essa linha, o bloqueio de linha compartilhado será convertido em bloqueio de linha exclusivo.

Exceto por uma exceção, é possível alternar de um nível de isolamento para outro durante uma transação. A exceção ocorre ao alterar de qualquer nível de isolamento para o isolamento SNAPSHOT. Isso faz com que a transação falhe e seja revertida. Porém, é possível alterar uma transação iniciada em isolamento SNAPSHOT para qualquer outro nível de isolamento.

Quando você altera uma transação de um nível de isolamento para outro, os recursos lidos após a alteração são protegidos de acordo com as regras do novo nível. Recursos lidos antes da alteração continuam sendo protegidos de acordo com as regras do nível anterior. Por exemplo, se uma transação mudar de READ COMMITTED para SERIALIZABLE, os bloqueios compartilhados adquiridos após a alteração serão, nesse caso, mantidos até o término da transação.

Se você emitir SET TRANSACTION ISOLATION LEVEL em um procedimento armazenado ou em um disparador, quando o objeto retornar, o controle do nível de isolamento será redefinido com o nível que estava em vigor quando o objeto foi invocado. Por exemplo, se você definir REPEATABLE READ em um lote e o lote chamar um procedimento armazenado que define o nível de isolamento como SERIALIZABLE, a configuração de nível de isolamento será revertida para REPEATABLE READ quando o procedimento armazenado retornar o controle para o lote.

Atenção: durante a avaliação procure apresentar exemplos de como esses tipos de isolamento são importantes e recomendáveis para os casos apresentados.



Comandos Relacionados:

```
SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED
SET TRANSACTION ISOLATION LEVEL READ COMMITTED
SET TRANSACTION ISOLATION LEVEL REPEATABLE READ
SET TRANSACTION ISOLATION LEVEL SERIALIZABLE
SET TRANSACTION ISOLATION LEVEL SNAPSHOT
dbcc useroptions
```

```
select * from sys.databases
select * from sys.tables
select * from sys.all_objects where object_id = object_id(<nome do objeto>)
```

```
exec sp_lock
select @@SPID
exec sp_who
```

```
begin tran
select @@TRANCOUNT
```

```
rollback
```

```
select * from aproveitamentos (READUNCOMMITTED)
select * from aproveitamentos (READCOMMITTED)
select * from aproveitamentos (SERIALIZABLE)
select * from aproveitamentos (REPEATABLE READ)
```

```
select * from aproveitamentos (nolock)
select * from aproveitamentos (Xlock)
select * from aproveitamentos (readpast)
select * from aproveitamentos (holdlock)
```

```
<comando SQL> WITH ( { ROWLOCK | TABLOCK | PAGLOCK } )
update aproveitamentos with (paglock) set apr_nota = 10 where dis_codigo =
2 and mat_numero = 1
```

Referência:

Manuais Online do SQL Server 2008. Disponível em: www.msdnbrasil.com.br. Acessado em: 06-03-2009.