**Понимание моделей восстановления SQL Server**

Каждая база данных на SQL Server имеет установку модели восстановления. SQL Server имеет три различных модели восстановления: простая (Simple), полная (Full) и с неполным протоколированием (Bulk-Logged). Установка модели восстановления определяет, какие варианты создания резервных копий и восстановления доступны для базы данных, а также то, каким образом ядро базы данных обрабатывает сохранение записей в журнале транзакций.

Журнал транзакций представляет собой подробный файл, который

используется для записи обновлений, выполняемых в каждой транзакции. Ядро SQL Server использует этот журнал для поддержания целостности базы данных. В случае, если транзакция требует отката, или база данных повреждается по какой-то причине, журнал транзакций используется для восстановления базы данных в согласованное состояние. Модель восстановления для базы данных определяет, как много данных требуется записать SQL Server в журнал транзакций, и может ли быть выполнено восстановление к некоторой точке во времени. Начальная установка модели восстановления для новой базы данных определяется моделью восстановления системной базы данных *model*. Настройка модели восстановления базы данных может быть легко изменена либо с помощью SSMS, либо кодом T-SQL. Чтобы лучше понять детали каждой модели восстановления и то, как они влияют на доступные варианты резервирования и восстановления, позвольте мне сделать обзор каждой из имеющихся моделей восстановления.

**Простая модель восстановления**

Simple recovery model является самой простой из моделей восстановления. Когда используется эта модель, каждая транзакция по-прежнему записывается в журнал транзакций. Записи журналов транзакций будут автоматически удаляться , если используется простая модель восстановления. Этот процесс удаления происходит для всех завершенных транзакций, когда устанавливается контрольная точка. Поскольку записи журнала удаляются при возникновении контрольной точки, резервирование журналов транзакций не поддерживается при использовании простой модели восстановления. Это означает, что восстановление к заданному моменту времени не может быть выполнено, если для базы данных модель восстановления установлена в SIMPLE. Поскольку в данном режиме журнал транзакций автоматически очищается, это позволяет сохранять его малый размер и избежать контроля над его ростом.  
  
Используя простую модель восстановления, вы можете делать только полные и дифференциальные резервные копии. Поскольку эта модель не поддерживает использование резервирования журналов транзакций, вы можете восстанавливать базу данных только к моменту времени, когда было выполнено полное или дифференциальное резервирование. Если требуется обеспечить восстановление базы данных к какому-нибудь другому моменту времени, необходимо использовать другую модель восстановления.

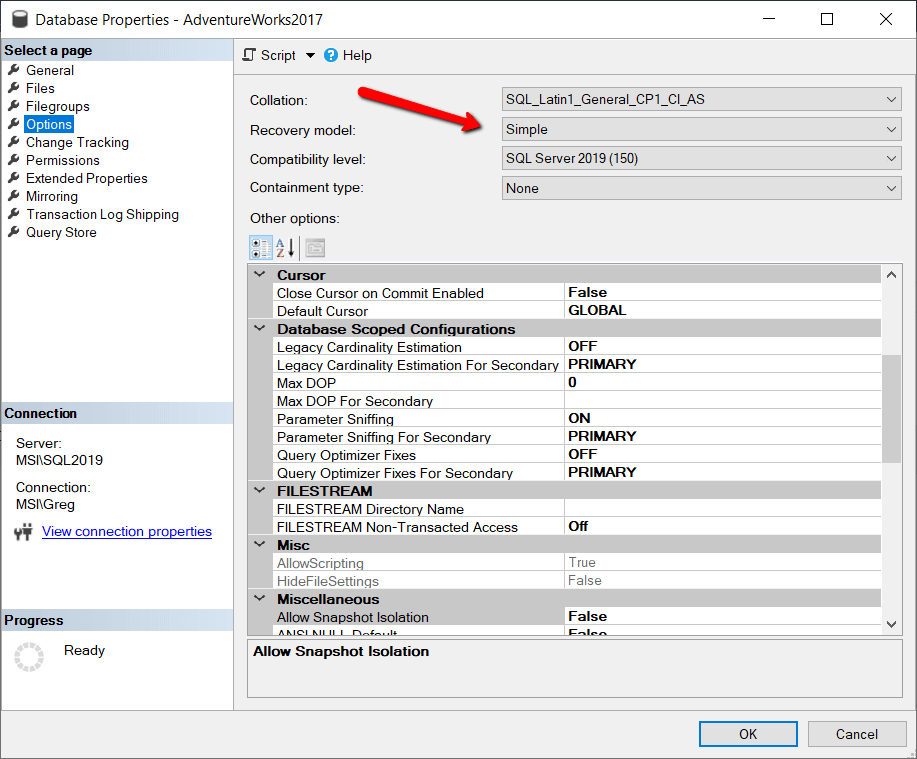
**Полная модель восстановления**

Как следует из названия, полная модель восстановления поддерживает все варианты для создания резервных копий и восстановления базы данных. Подобно простой модели восстановления, все транзакции записываются в журнал транзакций, но они остаются в журнале транзакций и после завершения транзакции. Записи остаются в журнале транзакций до тех пор, пока не будет сделана резервная копия журнала. При выполнении резервирования журнала транзакций для базы, которая находится в режиме полного восстановления, записи журнала записываются в бэкап журнала транзакций, и записи завершенных транзакций удаляются из журнала транзакций.  
  
Поскольку каждая транзакция записывается в журнал транзакций, полная модель восстановления поддерживает восстановление к некоторому моменту времени, т.е. база данных, которая полностью журнализируется, может бы восстановлена к любому моменту времени. Но это также означает, что журнал транзакций неизбежно должен быть достаточно большим, чтобы обеспечивать запись всех транзакций, пока не будет выполнен бэкап журнала. Если приложение выполняет множество транзакций, то существует опасность переполнения журнала транзакций. Когда журнал транзакций заполняется, база данных перестает принимать транзакции до тех пор, пока не будет сделан бэкап журнала транзакций, увеличен размер журнала транзакций, или журнал не будет усечен. Следовательно, при использовании для базы данных полной модели восстановления, вам вам необходимо делать резервные копии журнала достаточно часто, чтобы удалять завершенные транзакции до того, как журнал транзакций заполнится.  
  
Помимо транзакций вставки и обновления, в журнал записывается множество информации, связанной с другими операциями, подобными созданию/изменению индекса и массовой загрузки. Если вы обнаружили, что ваш журнал транзакций заполняется вследствие операций с индексами или массовой загрузки (например, SELECT INTO), вы можете рассмотреть вариант перехода на модель восстановления с неполным протоколированием, пока эти операции выполняются.

**Модель восстановления с неполным протоколированием**

Модель восстановления с неполным протоколированием минимизирует использование пространства журнала транзакций при операциях с неполным протоколированием, подобных BULK INSERT, SELECT INTO или CREATE INDEX. Функцилнальность этой модели подобна полной модели восстановления за исключением того, что записи в журнал транзакции минимально протоколируются при выполнении указанных операций. Минимальное протоколирование помогает поддерживать журнал меньших размеров, но при этом журнализируется не так много информации.  
  
Модель восстановления с неполным протоколированием улучшает производительность операций с загрузкой больших объемов данных посредством сокращения количества журнализируемой информации. Кроме того, поскольку транзакции с неполным протоколированием не полностью журнализируются, это сокращает количество места при записи в журнал транзакций, что уменьшает шансы переполнения журнала транзакций. Поскольку операции с неполным протоколированием минимально журнализируются, это оказывает влияние на восстановление к определенному моменту времени.  
  
Восстановление к определенному моменту времени все еще может быть выполнено при использовании модели восстановления с неполным протоколированием. Если база данных повреждается во время выполнения операции с минимальным протоколированием, базу данных можно восстановить только к последнему бэкапу журнала транзакций, созданному до первой операции с минимальным протоколированием. Если бэкап журнала транзакций содержит операции с неполным протоколированием, опции с остановкой не могут использоваться. Если операции с минимальным протоколированием вообще не выполнялись во время использования модели восстановления с неполным протоколированием, то вы все еще можете выполнять восстановление к определенному моменту времени так же, как и для полной модели восстановления.  
  
Модель восстановления с неполным протоколированием является отличным способом минимизировать объем журнала транзакций и улучшить производительность массовых операций с неполным протоколированием. Однако следует иметь в виду, что во время, когда была выполнена операция массовой загрузки, восстановление к определенному моменту времени не может быть выполнено. Следовательно, чтобы минимизировать потерю данных при использовании операций массовой загрузки, вам нужно сделать бэкап журнала транзакций непосредственно перед этой операцией, а затем еще один сразу после завершения операции. При этом восстановление к определенному моменту времени может быть выполнено при использовании любых бэкапов журнала транзакций, как сделанных до операции массовой загрузки, так и выполненных после специальной резервной копии журнала, выполненной после завершения операции массовой загрузки.

**Какая модель восстановления используется?**

Существует много способов для определения того, какая модель восстановления для базы данных используется. Одним вариантом определения модели восстановления является SQL Server Management Studio. Для этого сначала щелкните правой кнопкой на базе данных и выберите пункт "Properties" (свойства) из меню. В окне свойств выберите пункт "Options" (опции) в левом контекстном меню. Выполнив эти действия, вы увидите нижеприведенное окно.  
  
  
Рис.1: Опция Recovery Model  
  
Рисунок 1 показывает, что для базы данных AdventureWorks2017 модель восстановления установлена в значение Simple.  
  
Другим способом отображения свойств восстановления базы данных является выполнение кода T-SQL, показанного в листинге 1.  
  
Листинг 1: Код для отображения модели восстановления

SELECT name, recovery\_model\_desc

FROM sys.databases

WHERE name = 'AdventureWorks2017' ;

**Изменение модели восстановления**

С течением времени может потребоваться изменить модель восстановления. Это может произойти, когда приложению требуется больше или меньше вариантов восстановления базы данных. Если требуется изменить модель восстановления, это можно легко сделать с помощью кода в листниге 2.  
  
Листинг 2: Изменение модели восстановления с помощью кода T-SQL

USE master;

GO

ALTER DATABASE AdventureWorks2017 SET RECOVERY FULL;

GO

В листинге 2 модель восстановления для базы данных AdventureWorks2017 была изменена на FULL. Кроме того, вы можете изменить модель восстановления базы данных с помощью страницы свойств SSMS, показанного на рисунке 1. Для этого просто выберите требуемый вариант в поле Recovery Model, а затем щелкните OK.

**Восстановление баз данных для каждой модели восстановления**

Модель восстановления определяет вид резервных копий, которые могут быть сделаны для базы данных, который, в свою очередь, определяет имеющиеся варианты для восстановления поврежденной базы данных. Давайте рассмотрим то, как каждая модель восстановления может повлиять на варианты восстановления для типичной проблемы, приводящей к повреждению базы данных. Типичная проблема, которую я рассмотрю, заключается в некорректном обновлении программистом T-SQL базы данных, который затем просит администратора баз данных восстановить базы данных к некоторому моменту времени, предшествующему ошибочному процессу обновления, повредившего базу данных. В реальных условиях администратору чаще приходится восстанавливать базу данных в результате выполнения некорректного кода, чем в результате повреждения из-за аппаратных сбоев. Ниже я исследую, по крайней мере, один вариант, который может быть использован для восстановления поврежденной базы данных в результате программной ошибки для каждой из различных моделей восстановления.

**Простая модель восстановления**

Когда база данных использует простую модель восстановления, вы можете использовать полный бэкап или полный и дифференциальный бэкап для восстановления поврежденной базы данных в результате, например, программистской ошибки. Поскольку при использовании простой модели восстановления нельзя взять бэкапы журнала транзакций, то восстановления к определенному моменту времени выполнить невозможно, за исключением момента конца диффренциала, если он имеется. Следовательно, чтобы восстановить поврежденную базу данных, вам нужно восстановить полный бэкап базы данных и по-возможности дифференциальный бэкап базы данных, который был сделан как раз перед повреждением. Восстановление полного и дифференциальных бэкапов приводят базу данных в состояние, в котором она находилась на момент, когда были сделаны бэкапы. Все обновления базы данных, сделанные после этих бэкапов будут потеряны вместе с обновлением, выполненным ошибочным кодом.

**Полная модель восстановления**

В отличие от простой модели восстановления, полная модель восстановления предлагает много способов восстановления от ошибочного обновления или удаления. Первый вариант - это использовать последний полный бэкап. Если используется только последний бэкап, то восстановленная база данных потеряла бы такое же количество обновлений, как и при восстановлении по простой модели, описанной выше.  
  
Другим вариантом является использование опций восстановления к определенному моменту времени, доступных для полной модели восстановления. Восстановление к определенному моменту времени означает, что администратор баз данных может восстановить базу данных к моменту времени (минуте), непосредственно предшествующему выполнению ошибочного обновления. Любые обновления, которые были сделаны с базой данных после полного бэкапа, но перед моментом восстановления, не будут потеряны, в отличие от простой модели восстановления.  
  
Для того, чтобы восстановить базу данных к моменту времени после полного бэкапа, база данных должна иметь, по крайней мере, один бэкап журнала транзакций, сделанный после полного бэкапа. Если журнал транзакций не велся до возникновения повреждения, то единственным вариантом устранить повреждение будет восстановление базы данных с помощью только лишь последнего полного бэкапа. В этом случае количество утерянных данных будет тем же, что и для простой модели восстановления.  
  
Имеются и другие варианты, которые могут быть использованы для восстановления к определенному моменту времени после полного бэкапа. Одним из таких вариантов может быть использование дифференциального бэкапа. Дифференциальный бэкап - это бэкап, который содержит все изменения с базой данных, произошедшие после последнего полного бэкапа.

**Модель восстановления в неполным протоколированием**

Подобно полной модели восстановления, модель восстановления с неполным протоколированием поддерживает восстановление к определенному моменту времени, если момент времени восстановления не содержится в бэкапе журнала транзакций, содержащего операции неполного протоколирования. Таким образом, если база данных повреждается в результате программной ошибки при использовании модели восстановления с неполным протоколированием, администратор по-прежнему может восстановить базы данных , если повреждение произошло до или после бэкапа журнала транзакций, который содержит операции с неполным протоколированием. Если обновление происходит, когда выполняется операция с неполным протоколированием, то лучшее, что может сделать администратор, это восстановить к моменту времени последнего бэкапа журнала транзакций, выполненного до операции с неполным протоколированием. Даже если администратор не сможет восстановить базу к моменту времени, входящему в бэкап журнала транзакций, когда он содержит операцию с неполным протоколированием, он, по крайней мере, может это сделать, если журнал транзакций не содержит каких-либо операций с неполным протоколированием.