Emmet для SQL Server Management Studio - <https://www.devart.com/dbforge/sql/sqlcomplete/download.html>

---------------

Проверка включённых портов для подключения к данному инстансу SQL Server

USE master

GO

xp\_readerrorlog 0, 1, N'Server is listening on'

GO

---------------

База для тестов - AdventureWorks

---------------

## Основные сведения о базах данных

На компьютере можно установить один или несколько экземпляров SQL Server . Каждый экземпляр SQL Server может содержать одну или несколько баз данных. В базе данных может содержаться одна или несколько групп объектов владения, которые называются схемами. В каждой схеме присутствуют объекты базы данных, такие как таблицы, представления и хранимые процедуры.

Базы данных SQL Server хранятся в файловой системе в виде файлов. Файлы могут быть объединены в группы файлов.

При получении доступа к экземпляру SQL Server пользователи идентифицируются согласно имени входа. При получении доступа к базе данных пользователи идентифицируются как пользователи базы данных.

Пользователь, имеющий доступ к базе данных, может получить разрешения на доступ к объектам этой базы данных. Хотя разрешения и могут быть предоставлены отдельным пользователям, рекомендуется создавать роли базы данных, добавляя при этом пользователей базы данных к соответствующим ролям, а затем предоставлять разрешения ролям. Предоставление разрешений ролям, а не пользователям позволяет легко и понятно управлять процессом распределения разрешений, несмотря на постоянное изменение и рост числа пользователей.

---------------

**Вопросы :**

- какие типы баз данных бывают?

- сколько и какие базы нужны для работы сиквел-сервера?

- из чего состоит каждая база?

**Ответы:**

- БД бывают системные и пользовательские;

- для работы нужны системные БД (master, msdb, model, resource и tempdb), каждая из них носит сопровождающий характер для работы самого сервера;

- master - все юзеры, связанные сервера(я так понимаю в кластере), параметры конфигурации

системы, тут хранится инфа о зарегистрированных БД и местоплоложении их файлов

- msdb - используется агентом для обработки расписаний заданий

- model - выступает шабоном при создании новой БД, содержит базовую структуру

- tempdb - служит для хранения временных данных (таблиц, процедур, таблиц с

возвращаемыми функциями данными). Все данные хранятся в пределах сеанса работы с сервером

- resource - Я НЕ ПОНЯЛ ЕЁ НАЗНАЧЕНИЕ

- составляющие БД :

- Объектно - коллекции таблиц, представления и хранимые процедуры. Что касается последних двух я не понял что это такое.

- Физически – как минимум два файла, файл с данными и файл журнала(логи).

---------------

**Вопросы :**

- Файлы БД и группы

**Ответы:**

Каждая база данных SQL Server имеет как минимум два рабочих файла: файл данных и файл журнала. Файлы данных содержат данные, таблицы, индексы, хранимые процедуры и представления. Файлы журнала содержат сведения, необходимые для восстановления всех транзакций в базе данных. Файлы данных могут быть объединены в файловые группы для удобства распределения и администрирования.

Типы файлов БД :

- Первичный файл данных содержит сведения, необходимые для запуска базы данных, и ссылки на другие файлы в базе данных. Данные и объекты пользователя могут храниться в данном файле или во вторичном файле данных. В каждой базе данных имеется один первичный файл данных. Для имени первичного файла данных рекомендуется расширение MDF.

- Вторичные файлы могут быть использованы для распределения данных на несколько дисков, в этом случае каждый файл записывается на отдельный диск. Кроме того, если размер базы данных превышает максимальный размер для одного файла Windows, можно использовать вторичные файлы данных, таким образом база данных сможет расти дальше. Для имени вторичного файла данных рекомендуется расширение NDF.

- Файлы журнала транзакций содержат сведения, используемые для восстановления базы данных. Для каждой базы данных должен существовать хотя бы один файл журнала. Для файлов журнала транзакций рекомендуется расширение LDF.

Файловые группы :

У каждой базы данных есть первичная файловая группа. Эта файловая группа содержит первичный файл данных и все вторичные файлы, не входящие в другие файловые группы. Пользовательские файловые группы могут создаваться для удобства администрирования, распределения и размещения данных.

Например, три файла — Data1.ndf, Data2.ndf и Data3.ndf — могут быть созданы на трех дисках соответственно и отнесены к файловой группе fgroup1. В этом случае можно создать таблицу на основе файловой группы fgroup1. Запросы данных из таблицы будут распределены по трем дискам, и это улучшит производительность. Подобного улучшения производительности можно достичь и с помощью одного файла, созданного на чередующемся наборе дискового массива RAID. Тем не менее файлы и файловые группы позволяют без труда добавлять новые файлы на новые диски.

---------------

**Вопросы :**

- Виды страниц

- Экстенты

**Ответы:**

Страницы файлов данных

Страница — основная единица хранения данных в SQL Server. Страницы файлов данных SQL Server нумеруются последовательно; первая страница файла получает нулевой номер (0). Каждый файл базы данных имеет уникальный цифровой идентификатор. Чтобы уникальным образом определить страницу базы данных, необходимо использовать как идентификатор файла, так и номер этой страницы. Первая страница каждого файла — это страница заголовка файла; она содержит сведения об атрибутах данного файла. Некоторые другие страницы, расположенные в начале файла, тоже содержат системные сведения, например карты размещения. Одна из системных страниц, хранимых как в первичном файле данных, так и в первом файле журнала, представляет собой загрузочную страницу базы данных, которая содержит сведения об атрибутах этой базы данных.

Типы страниц :

- Data - Строки со всеми данными, за исключением типов text, ntext, image, nvarchar(max),

varchar(max), varbinary(max) и xml, если для текста в строке установлено значение ON.

- Индекс - Содержимое индекса

- Текст/изображение - Типы данных больших объектов: text, ntext, image, nvarchar(max), varchar(max),

varbinary(max) и данные xml. Столбцы переменной длины, когда размер строки данных

превышает 8 КБ: varchar, nvarchar, varbinary и sql\_variant.

Следующие типы страниц отвечают за размещение экстентов и страниц

- GAM - На GAM-страницах записано, какие экстенты были размещены. В каждой карте GAM

содержится 64 000 экстентов. В карте GAM приходится по одному биту на каждый экстент в

покрываемом им интервале. Если бит равен 1, то экстент свободен; если бит равен 0, то

экстент размещен.

- SGAM - На SGAM-страницах записано, какие экстенты в текущий момент используются в качестве

смешанных экстентов и имеют как минимум одну неиспользуемую страницу. Если бит равен

1, то экстент используется как смешанный экстент и имеет свободную страницу. Если бит

равен 0, то экстент не используется как смешанный экстент, или он является смешанным

экстентом, но все его страницы используются.

- PFS - Сведения о размещении страниц и доступном на них свободном месте. Страницы **Page Free**

**Space (PFS)** записывают состояние размещения каждой страницы, информацию о том, была

ли размещена конкретная страница, а также количество свободного места на каждой

странице. В PFS на каждую страницу приходится по одному байту, хранящему информацию о

том, была ли страница размещена или нет, а если была — то пустая она, или ее заполнение

находится в промежутке от 1 до 50 процентов, от 51 до 80 процентов, от 81 до 95 процентов

или от 96 до 100 процентов.

После размещения экстента на объект компонент Компонент SQL Server Database Engine использует PFS-страницы для записи информации о том, какие страницы в экстенте размещены, а какие свободны. Эти сведения используются при размещении новой страницы.

PFS-страница является первой страницей в файле данных после страницы заголовка файла (страница с ИД 1). За ней следует страница GAM (страница с ИД 2), а затем — страница SGAM (страница с ИД 3). После первой PFS-страницы находится новая PFS-страница с примерно 8000 страницами, а также дополнительные PFS-страницы с последующими интервалами в 8000 страниц. После первой GAM-страницы на странице 2 находится другая GAM-страница с 64 000 экстентов, другая SGAM-страница с 64 000 экстентов находится после первой SGAM-страницы на странице 3 и дополнительные страницы GAM и SGAM с последующими интервалами в 64 000 экстентов.

Экстенты

Экстент — это набор из 8 физически непрерывных страниц.

В SQL Server есть два типа экстентов.

* **Однородные** экстенты принадлежат одному объекту, и все восемь страниц экстента может использовать только этот владеющий объект.
* **Смешанные** экстенты могут находиться в общем пользовании максимум у восьми объектов. Каждая из восьми страниц в экстенте может находиться во владении разных объектов.

---------------

**Вопросы :**

- Файл лога, как работает

**Ответы:**

Каждая база данных SQL Server имеет журнал транзакций, в котором фиксируются все изменения данных, произведенные в каждой из транзакций. Журнал транзакций является критическим компонентом базы данных и в случае системного сбоя может потребоваться для востановления БД

Логически журнал транзакций SQL Server выглядит последовательностью записей в журнале. Каждая запись идентифицируется регистрационным номером транзакции (номер LSN). Каждая новая запись добавляется в логический конец журнала с номером LSN, который больше номера LSN предыдущей записи. Записи журнала хранятся в той последовательности, в которой они были созданы. Каждая запись журнала содержит идентификатор транзакции, к которой она относится. Все записи журнала, связанные с определенной транзакцией, с помощью обратных указателей связаны в цепочку, которая предназначена для ускорения отката транзакции.

Записи журнала для изменения данных содержат либо выполненную логическую операцию, либо исходный и результирующий образ измененных данных. Исходный образ записи — это копия данных до выполнения операции, а результирующий образ — копия данных после ее выполнения.

В журнал транзакций записываются различные типы операций, например:

* начало и конец каждой транзакции;
* любые изменения данных (вставка, обновление или удаление), включая изменения в любой таблице (в том числе и в системных таблицах), производимые системными хранимыми процедурами или инструкциями языка DDL;
* любое выделение и освобождение страниц и экстентов;
* создание и удаление таблиц и индексов.

Кроме того, регистрируются операции отката. Каждая транзакция резервирует в журнале транзакций место, чтобы при выполнении инструкции отката или возникновения ошибки в журнале было достаточно места для регистрации отката. Объем резервируемого пространства зависит от выполняемых в транзакции операций, но обычно он равен объему, необходимому для регистрации каждой из операций. Все это пространство после завершения транзакции освобождается.

Часть файла журнала от первой записи, требуемой для полного отката базы данных, до последней зарегистрированной записи называется активной частью журнала или активным журналом. Этот раздел журнала необходим для выполнения полного восстановления базы данных.

**Физически Журнал транзакций в базе данных состоит из одного или несколько физических файлов, в каждом файле создаётся логический файл на весь размер физического файла, в нём создаются виртуальные файлы**. Для каждой базы данных должен существовать хотя бы один файл журнала. Внутри системы движок SQL Server Database Engine делит каждый физический файл журнала на несколько виртуальных файлов журнала. Виртуальные файлы журнала не имеют фиксированных размеров. Не существует также и определенного числа виртуальных файлов журнала, приходящихся на один физический файл журнала. Компонент Компонент Database Engine динамически определяет размер виртуальных файлов журнала при создании или расширении файлов журнала. Компонент Компонент Database Engine стремится обслуживать небольшое число виртуальных файлов. После расширения файла журнала размер виртуальных файлов определяется как сумма размера существующего журнала и размера нового приращения файла. Администраторы не могут настраивать или устанавливать размеры и число виртуальных файлов журнала.

Рекомендуется назначать файлам журнала значение *size*, которое было бы максимально близким к окончательному требуемому размеру, а также задавать относительно большое значение *growth\_increment*.

ОТ МЕНЯ – Логи – это физические файлы, которые внутри себя имеют логический файл на всё пространство, в котором, в свою очередь, посредством движка сервера (SQL Engine) формируются несколько логических файлов. При создании файла лога, админ с помощью инструкции Alter Database устанавливает значение size и параметр авторасширения. Самая старая запись, от которой начинаются все записи, имеет (MinLSN) минимальный регистрационный номер в БД. Запись с номером MinLSN очень важна, на ней стоит контрольная точка или делался полный бекап и от неё начинается откат при сбое, она необходима для успешного отката на уровне всей базы данных. Если параметр расширения не установлен или не производится контроль за ростом физ. файла, получаем переполнение и ошибку. Если в журнале содержится несколько физических файлов журнала, логический журнал будет продвигаться по всем физическим файлам журнала до тех пор, пока он не вернется на начало первого физического файла журнала.

### **Усечение журнала**

Усечение журнала необходимо для предотвращения переполнения журнала. При усечении журнала удаляются неактивные виртуальные файлы журнала из логического журнала транзакций базы данных SQL Server , что приводит к освобождению пространства в логическом журнале для повторного использования физическим журналом транзакций. Если усечение журнала транзакций не выполняется, со временем он заполняет все доступное место на диске, отведенное для файлов физического журнала. Однако перед усечением журнала должна быть выполнена операция создания контрольной точки. Новая контрольная точка записывает текущие страницы, измененные в памяти (известные как измененные незафиксированные страницы), вместе со сведениями журнала транзакций из памяти на диск. При создании контрольной точки неактивная часть журнала транзакций помечается как неиспользуемая, после чего ее можно освободить путем усечения журнала.

фыв

---------------

**Вопросы :**

- Модели востановления

**Ответы:**

іва

---------------

**Вопросы :**

- что такое инстансы (дефолтные и именованные)

- сервисы в поставке сиквела

- сетевые протоколы и порты

- Collation (пиздец, как важно!!)

- настройки во время и после установки

----------------------------------------------------------------------------------------------------------------------------------------------------

**BACKUP**

Системные БД SQL Server 2016 DevEd :

- master

- msdb Хранятся C:\Program Files\Microsoft SQL Server\MSSQL13.MSSQLSERVER\MSSQL\DATA

- model **Обязательно делать бекап**

- resource - C:\Program Files\Microsoft SQL Server\MSSQL13.MSSQLSERVER\MSSQL\Binn\mssqlsystemresource.mdf/ldf

----------------------------------------------------------------------------------------------------------------------------------------------------

|  |  |  |
| --- | --- | --- |
| DISTINCT | Уникальные записи | SELECT DISTINCT speed, ram  FROM PC; |
| ORDER BY | Сортировка результата выборки по указанному полю/полям. Сортировка по возрастанию ASC или убыванию DESC | SELECT DISTINCT speed, ram  FROM PC  ORDER BY ram DESC;  Сортировка по двум полям  SELECT DISTINCT speed, ram  FROM PC  ORDER BY ram DESC, speed DESC; |
| WHERE | Горизонтальное условие на выборку, установка отбора по условию | SELECT DISTINCT speed, ram  FROM PC  WHERE price < 500  ORDER BY 2 DESC; |
| В запросах, в которых используется DISTINCT, при использовании GROUP BY и ORDER BY в SELECT должны быть включены поля по которым производится групировка и сортировка.  Эту проблемму можно решить используя агрегатную функцию  SELECT model FROM PC  GROUP BY model  ORDER BY MAX(price) DECT; | | |
| MONTH и DAY | Используется для сортировке по месяцу или дню или вместе | SELECT date  FROM Battles  ORDER BY MONTH(date), DAY(date); |
| BETWEEN | Используется в отборе по диапазону условий | exp1 BETWEEN exp2 AND exp3  равносильно  exp1 >= exp2 AND exp1 <= exp3  Пример: Найти модель и частоту процесора ценой от $400 до $600  SELECT model, speed  FROM PC  WHERE price BETWEEN 400 AND 600; |
| IN |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Задачи от Паши :

надо настроить на сервере автоматические бэкапы баз по такому расписанию:

фулл - раз неделю в воскресенье в 20-00

дифф - каждый день кроме воскресенья в 20-00

лог - каждые 4 часа

на диске должна храниться история за последнюю неделю (т.е. джоб должен анализировать и удалять старое)

имя файлов должно быть user-friendly, типа ServerName\_master\_FULL\_20190213\_173120.bak

так же написать скрипт для автоматической генерации скриптов на случай восстановления

**Полезные команды**

**Определение размера базы**

USE AdventureWorks2016;

EXEC sys.sp\_spaceused @updateusage = N'true';