

Администрирование в информационных системах

Практическое занятие 2.
Системные и пользовательские БД

- На физическом уровне каждая БД SQL Server хранится как минимум в двух файлах:
 - **Файл данных** (data file)
 - **Файл журнала транзакций** (transaction log file)
- Каждая БД имеет один основной файл данных и может иметь дополнительные файлы данных, каждый из которых используется только этой БД.
 - **Основной файл** имеет, как правило, расширение .mdf.
 - Основной файл содержит данные таблиц и индексов, а также служебную информацию, которая используется SQL Server при работе. Он содержит системные таблицы, где хранится описание объектов БД, информацию о расположении дополнительных файлов БД.
 - **Дополнительные файлы** имеют, как правило, расширение .ndf. Дополнительные файлы используются, например, для размещения БД на различных дисках.
 - Файлы данных могут быть объединены в **группы файлов** (FILEGROUPS) для удобства управления данными.

Архитектура базы данных

- SQL Server хранит информацию о расположении всех файлов БД в двух местах:
 - в системной БД master
 - в основном файле БД
- Для управления данными сервер в основном использует информацию из БД master.
- Информация о файлах данных из основного файла используется в случае, когда информация в БД master должна быть изменена:
 - при обновлении сервера;
 - при восстановлении БД master;
 - при добавлении БД к серверу при помощи хранимой процедуры `sp_attach_db`.

Архитектура БД

- Каждый файл данных имеет *логическое имя*, используемое при обработке операторов Transact-SQL и *физическое имя*, используемое Windows.
 - **Логическое имя** должно быть уникально для отдельной БД и должно соответствовать правилам идентификации объектов SQL Server.
 - **Физическое имя** в должно удовлетворять требованиям именования файлов в операционной системе.

Файлы БД

- Дополнительными свойствами файла являются параметры:
 - *начальный размер,*
 - *величина приращения при увеличении,*
 - *максимальный размер файла.*
- Эти данные хранятся на *странице заголовка файла,* которая является первой страницей любого файла данных.

Файлы БД

- Для доступа к данным структура файла данных имеет страничную организацию.
 - Все данные разбиваются на отдельные страницы.
 - SQL Server уникально идентифицирует страницы по номеру файла и номеру страницы.
 - Для нумерации страниц используются последовательные номера, начиная с 0.

Размещение информации в файле

- Для обеспечения хранения информации необходимо выделить свободное место в файле данных и присвоить его соответствующему объекту.
- Свободное пространство, выделяемое для таблиц и индексов, измеряется в *ЭКСТЕНТАХ*.
 - **Экстент** – блок размером 64 кбайта и состоит из 8 страниц по 8 Кб каждая.
 - Экстенты бывают однородные и смешанные.

Выделение пространства для таблиц и индексов

- При создании нового объекта (таблицы или индекса) SQL Server находит смешанный экстенст, содержащий свободную страницу и связывает ее с созданным объектом.
 - Страница содержит данные, относящиеся только к одному объекту.
- Если объекту требуется дополнительное пространство, то SQL Server распределяет свободное пространство из смешанных экстенстов, пока необходимо число страниц не достигнет 8, затем выделяется однородный экстенст для данного объекта.
- Если в файлах данных нет свободного пространства и разрешено их автоматическое увеличение, то SQL Server будет увеличивать их размер по алгоритму циклического обслуживания.

Выделение пространства для таблиц и индексов

- При размещении страницы объекта в однородном или смешанном экстенте используется таблица IAM, чтобы отследить все страницы, выделенные таблице или индексу.
- Каждая страница IAM может указывать на 512000 страниц. Страницы IAM расположены в файле случайным образом и связаны друг с другом: первая страница IAM указывает на вторую и т.д.
- Другие системные страницы:
 - PFS – страница свободного пространства страниц
 - GAM – страница глобальной карты размещений (используется для выделения экстента из свободного пространства)
 - SGAM – страница дополнительной глобальной карты размещений (используется для выделения пространства в смешанном экстенте).

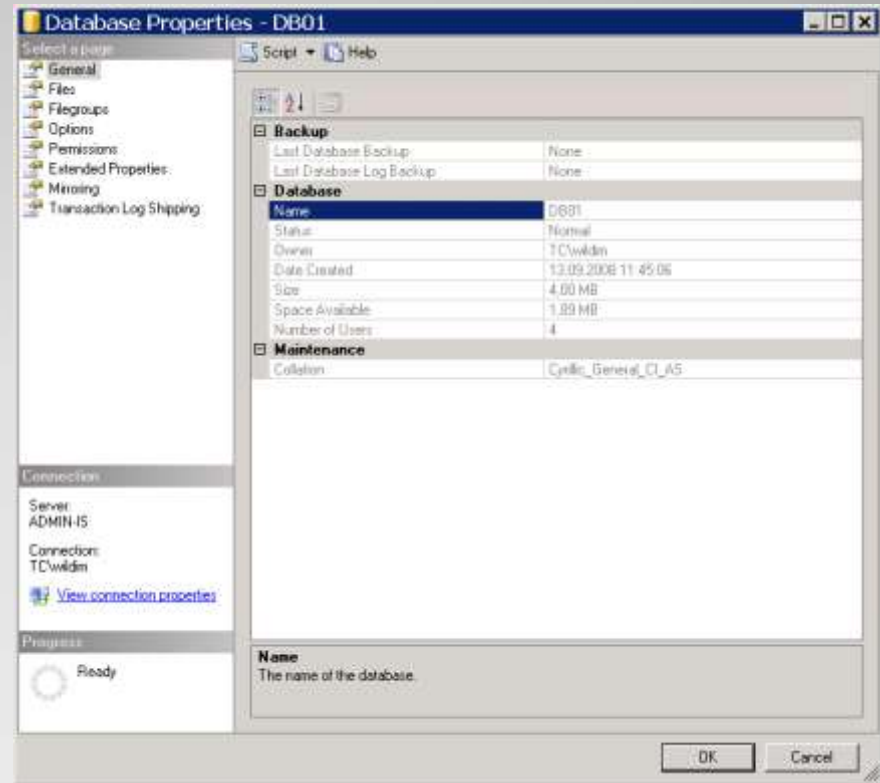
Таблица Index Allocation Map

- Параметры БД определяют различные аспекты работы с системными или пользовательскими базами данных.
- Параметра пользовательских БД при создании наследуют параметры системной БД **model**.
- Выделяют следующие типы параметров:
 - управление автоматическим поведением БД (такие как автоматическое создание и обновление данных статистики);
 - выбор модели восстановления;
 - совместимость с ANSI;
 - режим доступа к БД (например только для чтения и доступа, предоставленный только для владельца БД – dbo).

Настройка параметров БД

Просмотр значений параметров БД

- Просмотр установленные значения параметров БД можно с помощью:
 - SQL Server Management Studio;
 - системной функции **DATABASEPROPERTYEX**
 - например, для определения модели восстановления в БД DB01 необходимо выполнить запрос:
 - `SELECT DATABASEPROPERTYEX ('DB01','RECOVERY')`



- Изменение значений параметров пользовательской БД выполняется либо путем изменения параметров в SQL Server Management Studio, либо путем выполнения оператора Transact-SQL **ALTER DATABASE**.
 - Например, для изменения модели восстановления на модель восстановления результатов импорта необходимо выполнить запрос:
 - **ALTER DATABASE SEMBD SET RECOVERY BULK_LOGGED**

Изменение значений параметров БД

- Для создания новой базы данных используется инструкция T-SQL:
 - CREATE DATABASE *имя_базы_данных*
 - ON
 - (NAME=логическое имя файла, FILENAME=физическое имя файла, SIZE=размер, MAXSIZE=макс_размер, FILEGROWTH=автоувеличение)
 - LOG ON
 - (NAME=логическое имя файла, FILENAME=физическое имя файла, SIZE=размер, MAXSIZE=макс_размер, FILEGROWTH=автоувеличение)

Создание базы данных

- После создания БД возникает необходимость в периодическом изменении ее размеров.
 - При увеличении размеров БД возрастает нагрузка на систему, журналы транзакций увеличиваются быстрее.
- SQL Server поддерживает ряд механизмов управления изменениями БД:
 - автоматические методы контроля размера БД,
 - ручного изменения размеров файлов БД и журналов транзакций.

Изменение размера базы данных

- При создании пользовательской БД по умолчанию выбирается автоматическое увеличение размеров файлов данных и журналов транзакций.
 - При каждом увеличении файлов нагрузка на систему возрастает. Кроме того, увеличение дискового пространства для хранения файлов приводит к дефрагментации диска.
- Необходимо контролировать объемом данных и журнала транзакций и рекомендуется отказываться от модели автоматического увеличения размеров файлов данных и журнала транзакций.

Автоматическое увеличение размера файлов

- Изменение параметров автоматического увеличения размеров файлов данных и журнала транзакций выполняется с помощью графических средств SQL Server Management Studio.
- Другой вариант – использование операторов Transact-SQL.
 - Для отключения автоматического увеличения файла данных БД semdb, необходимо выполнить запрос:
 - **ALTER DATABASE SEMDB
MODIFY FILE (NAME = 'semdb_data', FILEGROWTH = 0)**

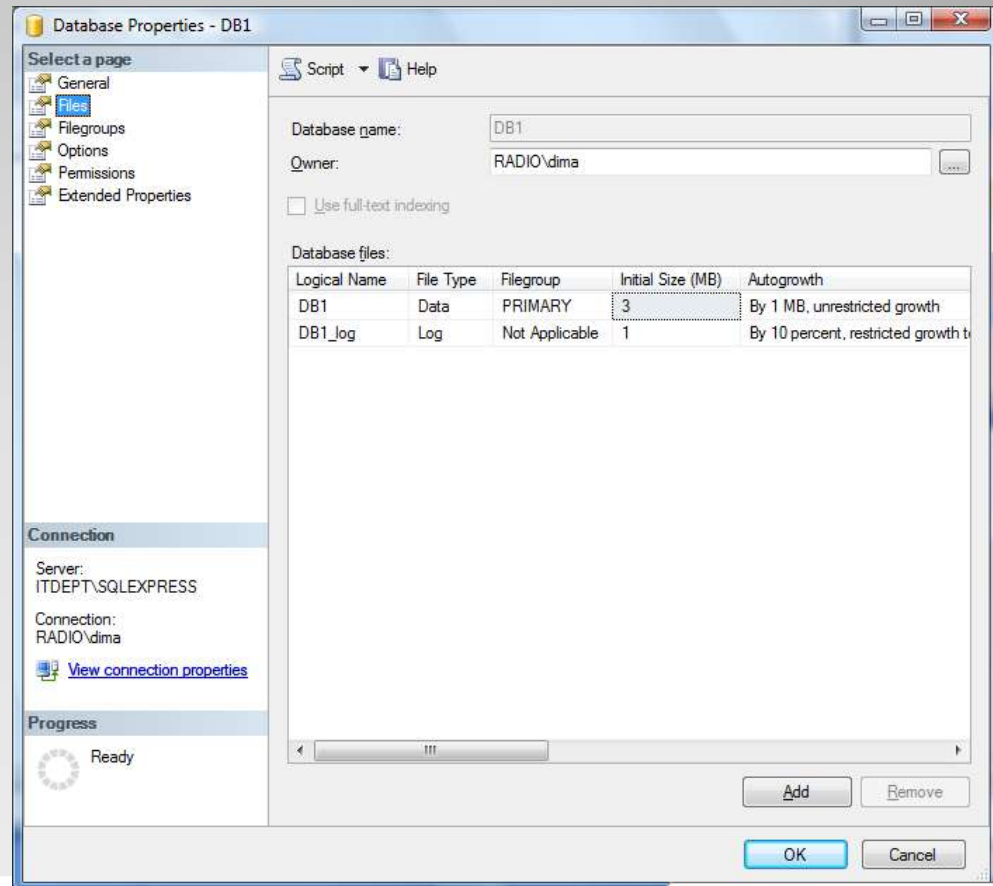
Изменение параметров автоматического увеличения размера файлов

- Размер файлов БД может быть автоматически не только увеличен, но и уменьшен.
- Такой режим может быть установлен, если файл данных или журнала транзакций имеет много свободного места.
 - Режим автоматического уменьшения размера файлов не рекомендуется к использованию, поскольку это уменьшает производительность системы.

Автоматическое уменьшение размеров файлов

Изменение размера файлов данных вручную

- Отслеживая изменения свободного пространства в файлах данных, администратор имеет возможность оценить необходимость в изменении размеров файлов. Такие операции выполняются во время снижения нагрузки на систему.
- Для увеличения размера файлов можно воспользоваться средствами SQL Server Management Studio, выбрав соответствующие закладки в контекстном меню соответствующей БД.



- При использовании средств Transact-SQL необходимо выполнить соответствующий запрос.
- Для установления нового размера файла данных в БД semdb размером 20Мб необходимо выполнить запрос:
 - **ALTER DATABASE SEMDB
MODIFY FILE (NAME = 'semdb_data', SIZE = 25)**
- Для уменьшения размера файла данных вручную используется оператор DBCC SHRINKFILE:
 - **USE SEMDB**
 - **DBCC SHRINKFILE ('semdb_data', 10)**
 - Данный запрос уменьшит размер файла данных в БД semdb до 10 Мб.
- При выполнении оператора DBCC SHRINKFILE операция производится над текущей БД. Во время уменьшения размера файл БД уменьшается с конца. По умолчанию все использованные страницы в сокращаемой части файла данных перемещаются на свободное пространство в начале файла. Файл не может быть уменьшен до размера меньшего, чем объем данных или размера БД model.

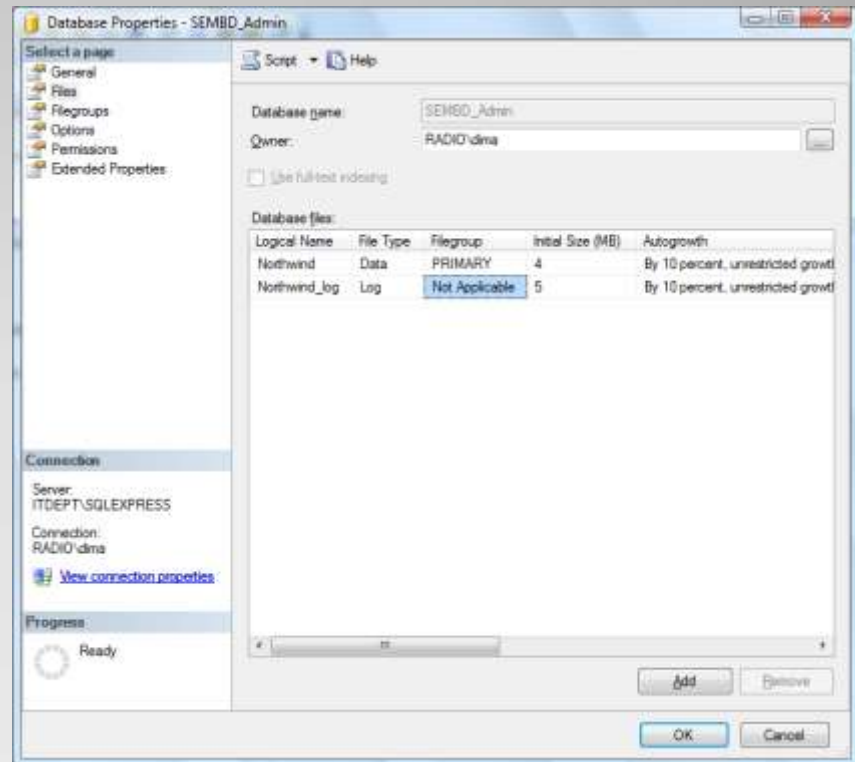
Изменение размера файлов данных вручную

- Для работы БД важно наличие свободного пространства в журнале транзакций.
- При использовании модели восстановления отдельных операций или модели восстановления результатов копирования необходимо выполнение регулярного резервного копирования файлов журнала транзакций.
- Если журналы транзакций заполняются слишком быстро необходимо либо чаще копировать журналы, либо позволять SQL Server автоматически увеличивать размер журнала.
- Изменение размеров файла журнала транзакций БД выполняется подобно изменению размеров файлов данных.

Изменение размера файла журнала транзакций

Создание дополнительных файлов

- При нехватке места на одном диске прибегают к созданию дополнительных файлов данных и журналов транзакций, расположенных на других дисках.
- При создании дополнительных файлов БД можно воспользоваться SQL Server Management Studio.
 - По умолчанию все дополнительные файлы данных добавляются в группу файлов PRIMARY.



- Создание дополнительных файлов возможно с помощью операторов Transact-SQL.
- Для добавления дополнительного файла данных в БД SEMDB необходимо задать:
 - **ALTER DATABASE SEMDB**
ADD FILE (NAME = 'SEMDB2', FILENAME = 'C:\DB\SEMDB2.ndf', SIZE = 10 , MAXSIZE = 25, FILEGROWTH = 5)

Создание дополнительных файлов

- Для обеспечения максимальной производительности, отказоустойчивости и минимизации времени восстановления данных при сбое, необходимо разработать стратегию размещения файлов данных, журналов транзакций и БД tempdb.
- При выборе размещения журнала транзакций необходимо учитывать основное назначение журналов – обеспечение восстановления данных в случае отказа дисков с размещенными на них файлами данных. Отсюда следует основное условие – размещаются файлы транзакций, как правило, на дисках отличных от дисков с файлами данных БД.
- Следующий шаг оптимизации производительности – размещение журналов транзакций на отдельных дисках для каждой БД.
- Еще один способ оптимизации производительности – использование системы RAID 1 (зеркальное хранение данных).

Конфигурирование дисковой подсистемы

- При выборе дисковой системы для хранения файлов данных основные условия – обеспечение недопустимости потери данных и минимизация времени простоя.
- Пути решения – размещения файлов данных и журналов транзакций на разных дисках, размещение файлов данных на разных дисках (особенно для больших БД).
- Использование RAID позволяет увеличить производительность системы.
- Если в системе требуется высокая производительность операций чтения, то рекомендуется использование RAID 5.
- Если требуется высокая производительность операций записи, то рекомендуется использование RAID 0, или RAID 10.

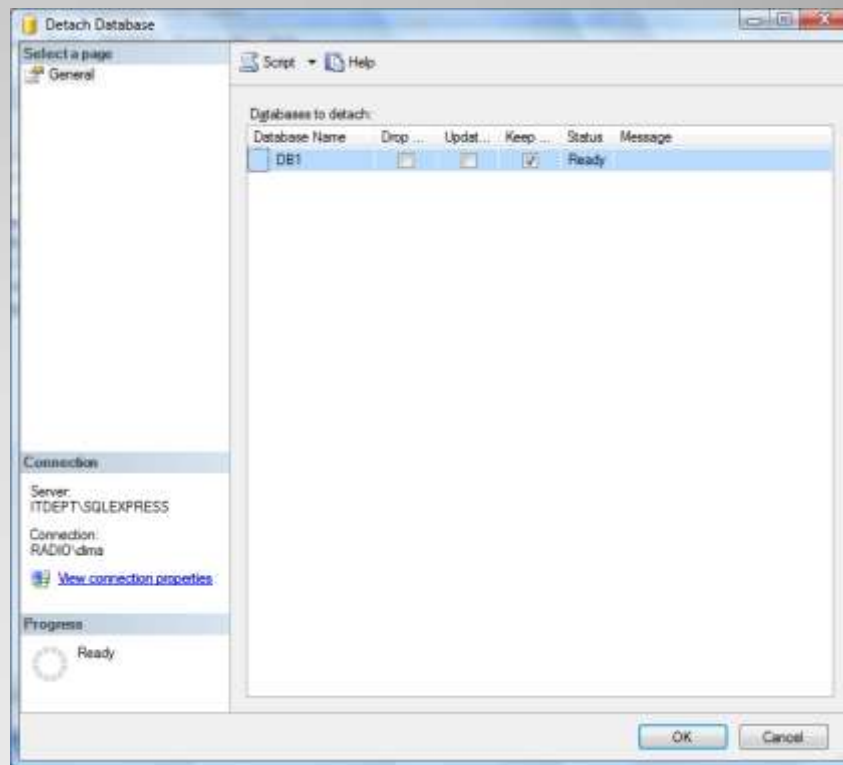
Конфигурирование дисковой подсистемы

- При выборе дисковой подсистемы для размещения БД tempdb необходимо помнить, что данная БД используется для временного хранения рабочих файлов.
- Оптимизация производительности данной БД предполагает возможность обеспечения обработки большого числа операций чтения и записи.
- Оптимизация производительности – размещение БД tempdb на отдельном диске и использование отдельного дискового контроллера.

Конфигурирование дисковой подсистемы

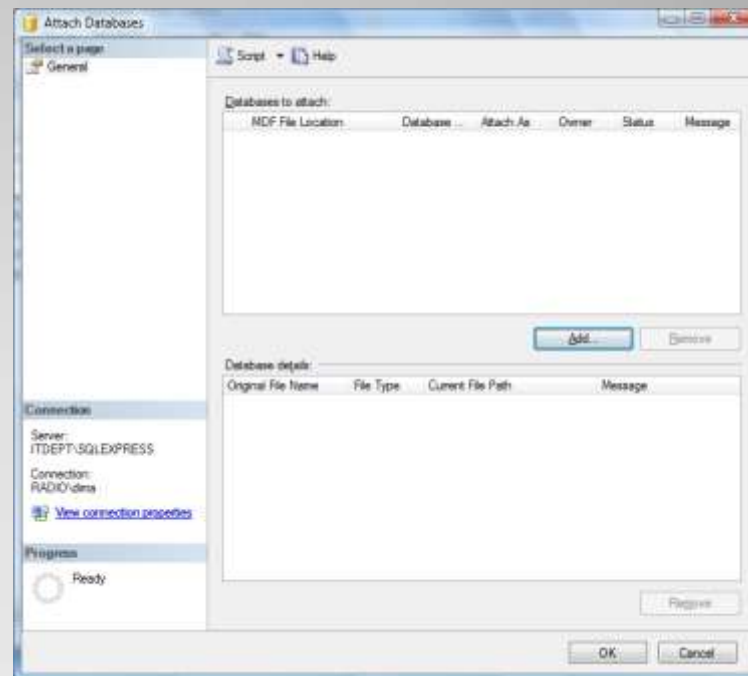
Отсоединение и присоединение БД

- Для отсоединения БД с помощью SQL Server Management Studio используется пункт меню **Все задачи \ Detach Database**.
 - Если с БД работают пользователи принудительное отключение выполняется с помощью кнопки Clear.
 - Рекомендуется обновить статистических сведений о БД.



Отсоединение и присоединение БД

- После отсоединения можно переместить физические файлы БД на новые носители хранения информации.
- Для присоединения БД используется пункт контекстного меню **Все задачи \ Attach Database**.
- В диалоговом окне указывается путь к основному файлу данных.
 - Основной файл содержит информацию о размещении других файлов БД.
 - В поле Current File Location для каждого перемещенного файла указывается его новое размещение.



- Для отсоединения БД можно использовать и операторы Transact-SQL.
- Для этого применяется системная хранимая процедура `sp_detach_db`:
 - **`sp_detach_db SEMDB, TRUE`** – выполняет отсоединение БД `semdb`.
- Для присоединения используется системная хранимая процедура `sp_attach_db`:
 - **`sp_attach_db SEMDB, @filename1 = 'C:\DB\Semdb_data.mdf'`**

Отсоединение и присоединение БД