

Организация БД в MS SQL Server 2008

Два аспекта организации БД:

1. Логическая
2. Физическая



Логическая структура БД SQL Server

Таблицы (Tables)	Двухмерные матрицы для хранения данных
Ключи (Keys)	Описание первичных и <u>вторичных ключей</u> *
Ограничения (Constraints)	Описания <u>ограничений</u> (Описания ограничений (<u>Check</u>) Описания ограничений (<u>Check</u>)* и значений по умолчанию
Индексы (Indexes)	Структуры, связанные с таблицей или представлением, и позволяющие существенно ускорить поиск в них данных**
Представления (Views)	Виртуальные таблицы, получаемые в результате запроса к таблицам и/или другим представлениям
Хранимые процедуры (Stored Procedures)	Вид программной единицы сервера, доступной клиентскому приложению.
Триггера (Triggers)	Вид программной единицы сервера, выполняемой автоматически при наступлении события
Функции (User defined function)	Вид программной единицы сервера
Пользовательские типы (User defined data type)	Типы данных, создаваемые пользователем
Правила (Rules)	Описание общих условий на столбцы *
Умолчания (Defaults)	Описания общих значений по умолчанию*



Логическая структура БД SQL Server

Логические компоненты в SSMS

Microsoft SQL Server Management Studio

File Edit View Tools Window Community Help

Object Explorer

Connect

VI-SQL2008.ЗаказыЮ - dbo.Заказы

Column Name	Data Type	Allow Nulls
ЗаказID	int	<input type="checkbox"/>
КлиентID	int	<input type="checkbox"/>
НомерНакладной	varchar(10)	<input checked="" type="checkbox"/>
ДатаЗаказа	date	<input type="checkbox"/>

Foreign Key Relationships

Selected Relationship:
FK_ЗаказаноТоваров_Заказы
FK_Заказы_Клиенты

Editing properties for existing relationship.

(General)

Check Existing Data On: Yes

Tables And Columns Sp...

Identity

(Name) FK_ЗаказаноТоваров_Заказы

Description

Table Designer

Enforce For Replication: Yes

Enforce Foreign Key Cor: Yes

INSERT And UPDATE Sp...

Add Delete Close

Full-text Specification: No

Has Non-SQL Server Subscriber: No

Identity Specification: Yes

Indexable: Yes

In Column: No

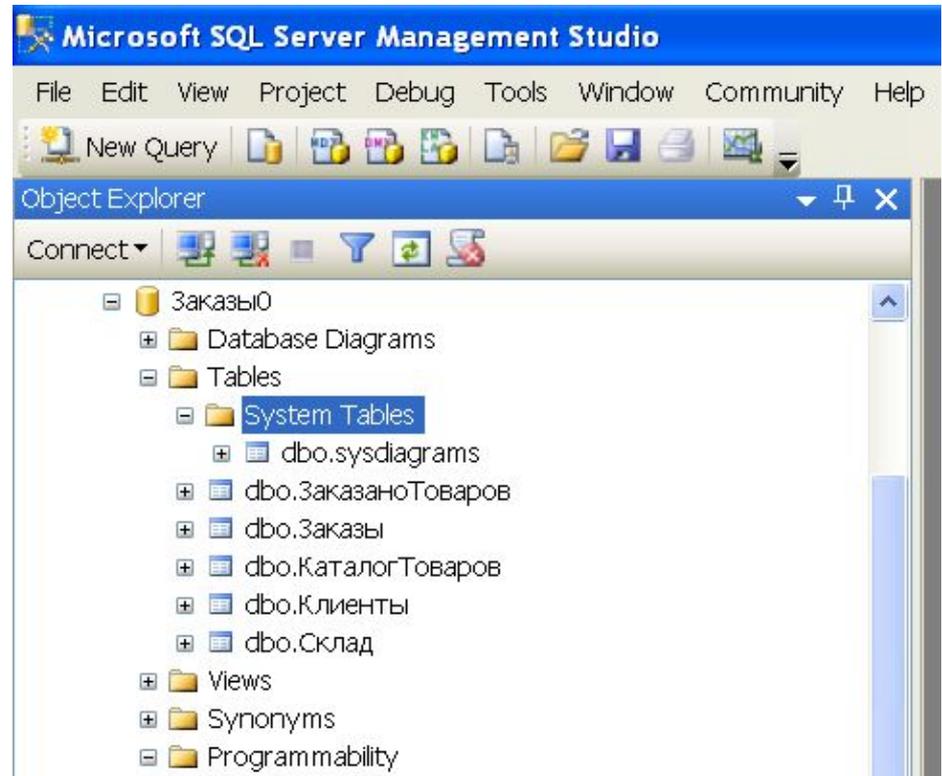
Registered Servers Object Explorer



Логическая структура БД SQL Server

Типы таблиц

- ПОЛЬЗОВАТЕЛЬСКИЕ
- СИСТЕМНЫЕ
- ВРЕМЕННЫЕ



Логическая структура БД SQL Server

Пользовательские таблицы

- создаются пользователем SQL Server(a) после создания БД. Создание таблиц можно выполнить в **SSMS** посредством описания их столбцов и других параметров или в **Query Editor** посредством выполнения команды **Create table**.
- имена пользовательских таблиц произвольные.



Логическая структура БД SQL Server

Системные таблицы

- создаются SQL Server(ом) при создании БД. Они хранят все необходимую информацию базы мета данных (БМД).
- прямой доступ к ним запрещен, т.е. посредством команд SQL (**insert, update, delete**). Для доступа к ним имеются специальные системные хранимые процедуры или системные представления, функции которых охватывают все возможные ситуации управления БД.



Логическая структура БД SQL Server

Временные таблицы

- создаются во время пользовательского сеанса при необходимости временного хранения данных, например, для промежуточных результатов при сложных расчетах;
- создаются в системной БД **tempdb**;
- бывают локальные и глобальные. Локальные таблицы видны только в том контексте, в котором они созданы, например, в хранимой процедуре, глобальные – из различных соединений;
- локальные временные таблицы уничтожаются после завершения программы, глобальные - после завершения соединения.
- имена локальных таблиц должны начинаться с символа #, а глобальные – с двух символов ## .



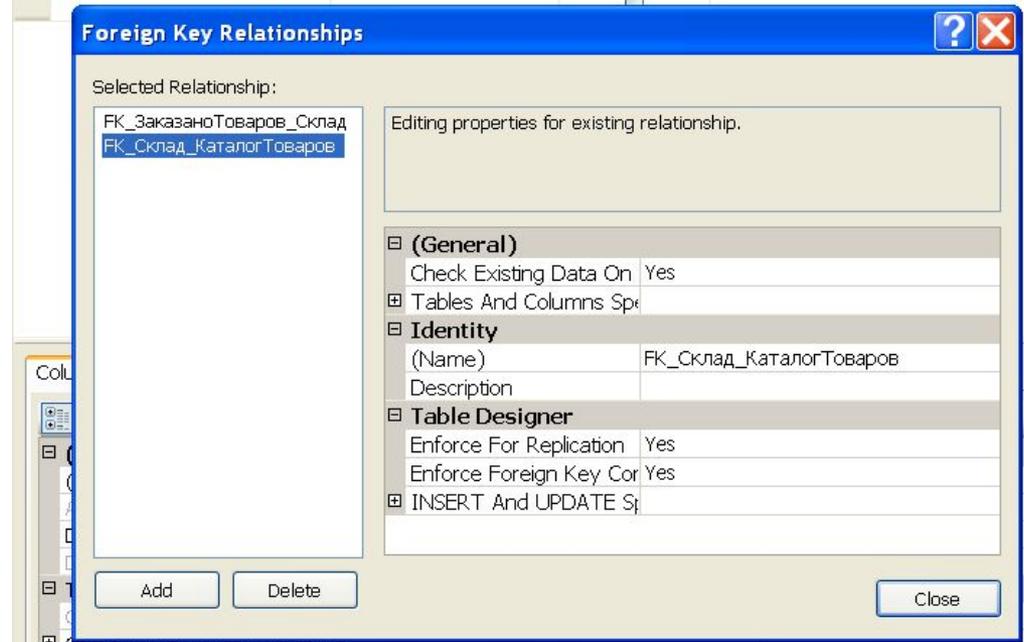
Логическая структура БД SQL Server

***Замечание.**

Объекты описания внешних связей всегда связаны с конкретной таблицей.

В SSMS для просмотра свойств имеющихся связей, их создания и удаление на каждой таблице связей (команда контекстного меню изменения проекта таблицы)

Column Name	Data Type	Allow Nulls
СкладID	int	<input type="checkbox"/>
КодТовара	int	<input type="checkbox"/>
Спецификация	varchar(50)	<input type="checkbox"/>
ЦенаОтпускная	real	<input type="checkbox"/>
Остаток	int	<input type="checkbox"/>
Место	int	<input checked="" type="checkbox"/>



Логическая структура БД SQL Server

***Замечание.**

Объекты описания ограничений check связей всегда связаны с конкретной таблицей.

В SSMS для просмотра свойств имеющихся ограничений check, их создание и удаление на каждой таблице нужно вызывать менеджер проверочных ограничений (команда контекстного меню “**check...**” в окне изменения проекта таблицы)



Логическая структура БД SQL Server

**Замечание

Объекты индексов связаны с конкретной таблицей.

В SSMS для просмотра свойств имеющихся индексов, создания новых или удаления существующих в выбранной таблице **нужно вызывать менеджер индексов (кнопка с буквой I)** в окне изменения проекта

The screenshot shows a table with the following columns:

Column Name	Data Type	Allow Nulls
КодТовара	int	<input type="checkbox"/>
Наименование	varchar(40)	<input type="checkbox"/>

The 'Indexes/Keys' dialog box is open, showing the following properties for the selected primary/unique key or index:

- (General)**
 - Columns: КодТовара (ASC)
 - Is Unique: Yes
 - Type: Primary Key
- Identity**
 - (Name): PK_КаталогТоваров
 - Description:
- Table Designer**
 - Create As Clustered: Yes
 - Data Space Specification: PRIMARY
 - Fill Specification:

Buttons: Add, Delete, Close



Логическая структура БД SQL Server

***Замечание.**

Объекты правила (**Rules**), умолчания (**Defaults**) содержат описания для всех таблиц, которые потом могут быть закреплены за конкретными столбцами в конкретных таблицах.

Они используются для совместимости с предыдущими версиями SQL Server и в будущих версиях Microsoft SQL Server этот компонент предполагается убрать.

Ограничения целостности, которые связаны с этими объектами, могут быть описаны в качестве параметров столбцов (**check и default**) в таблицах и представлениях.



Физическая структура БД SQL Server

БД SQL Server размещается в файлах следующих типов:



primary

Цель размещения БД в структурах операционной т.е файлах.

Основной файл, который содержит системную информацию о самой БД и её объектах (системные таблицы) и, если есть место, сами данные (пользовательские таблицы). В каждой БД имеется только один файл этого типа. Расширение этого файла **mdf**.

secondary

...

secondary

Дополнительные файлы, которые содержит только данные, не разместившиеся в основном файле. Рекомендуемое расширение этого файла **ndf**.

transaction log

...

transaction log

Файл журнала транзакций, которые содержит информацию для восстановления БД. Число таких файлов - от одного и более. Рекомендуемое расширение этого файла **ldf**.

Физическая структура БД SQL Server

Файлы БД имеют два имени:

- логическое имя (**Logical File Name**) Используется в командах T-SQL при ссылках
- Физическое имя (**OS File Name**) Под которым файл храниться на диске

Простая БД может состоять из 2-х файлов:

одного основного **primary** и

одного журнала транзакций **transaction log**

Файл(ы) журнала транзакций необходимо размещать на отдельном (ых) дисковом устройстве повышенной надёжностью и быстроедействием.

Физическая структура БД SQL Server

Физическая структура БД определяется :

- **количество файлов** БД secondary и transaction log
- **начальным размером** всех файлов
- **типом и размером увеличения** файлов

The screenshot shows the 'Database files' tab in SQL Server Enterprise Manager for the database 'Заказы0'. The 'Database files' table is as follows:

Logical Name	File Type	Filegroup	Initial Size (MB)	Autogrowth	Path	File Name
Заказы0_Data	Rows Data	PRIMARY	3	By 10 percent, unrestricted growth	C:\Program Files\Microsoft SQ...	Заказы0_Data.MDF
Заказы0_Log	Log	Not Applic...	1	By 10 percent, restricted growth to...	C:\Program Files\Microsoft SQ...	Заказы0_Log.LDF

A dialog box titled 'Change Autogrowth for Заказы0_Data' is open, showing the following settings:

- Enable Autogrowth
- File Growth:
 - In Percent (10)
 - In Megabytes (10)
- Maximum File Size:
 - Restricted File Growth (MB) (100)
 - Unrestricted File Growth

Buttons: OK, Cancel

Физическая структура БД SQL Server

Все файлы БД должны относиться к какой-либо группе

Размещение таблиц и индексов относится именно к какой-либо группе файлов, а не к файлу

Типы группы файлов:

Primary File Group - основная группа файлов. Включает файл типа **Primary** и все файлы **secondary**, не включенные в другие группы. Может быть только одна основная группа.

User Defined File Group – пользовательская группа файлов. Включает все файлы **secondary**, указанные в параметре **FILEGROUP** команды создания или изменения БД. Можно создавать несколько пользовательских групп с произвольным набором файлов (но один файл может принадлежать только одной группе)

Любая из группы файлов может быть назначена по умолчанию. Исходно это **Primary File Group**

Физическая структура БД SQL Server

Группы файлов

позволяют (при наличии нескольких независимых дисковых устройств) распараллелить операции записи/чтения в файловую группу данных и тем самым повысить производительность системы

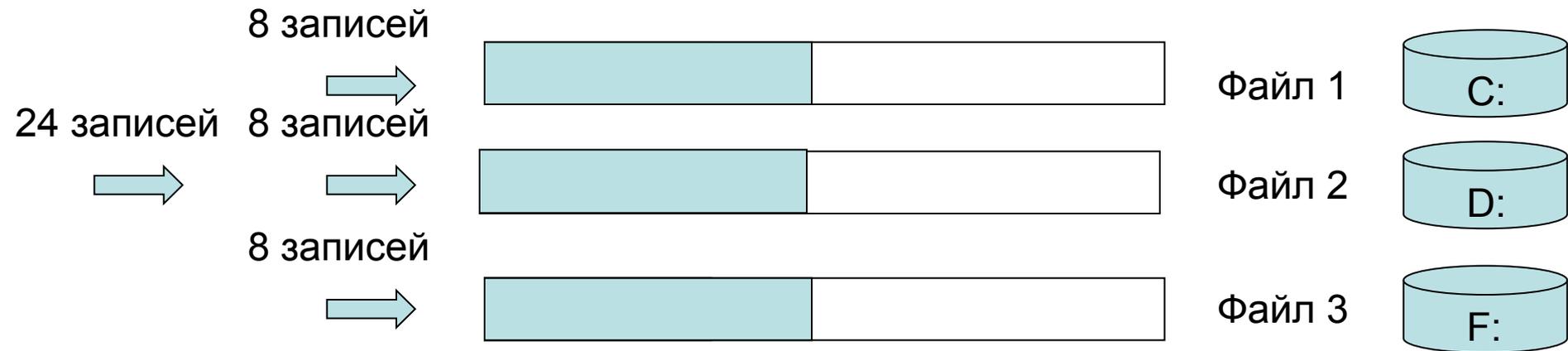
Физическая структура БД SQL Server

Заполнение файлов при организации «основной - дополнительный»



Физическая структура БД SQL Server

Заполнение сгруппированных файлов, размещенных на разных дисковых устройствах



RAID

Производительность, а также и надежность БД может обеспечиваться на системном уровне, используя систему **RAID**.

RAID (Redundant Array of Independent Disks — (избыточный массив независимых дисков) — технология виртуализации данных, которая объединяет несколько дисков в логический элемент.

Конфигурации RAID реализуется как на уровне дискового устройства, так и на логическом уровне ОС

Существуют различные варианты конфигурации RAID, которые обеспечивают определенный уровень производительности и надежности системы хранения данных на файловом уровне.

RAID

Наиболее распространенные уровни **RAID** следующие:

RAID 0 (чередование дисков): хорошая производительность, низкая надежность, избыточность дисков.

RAID 1 (зеркализация): хорошая производительность, высокая надежность, та же избыточность дисков.

RAID 5 (чередованием дисков с контролем четности): низкая производительность, устойчив к отказу одного диска, бо́льшая избыточность дисков.

RAID 10 (зеркализация с чередованием): хорошая производительность, высокая надежность (устойчива к отказу нескольких дисков), высокая избыточность (вдвое больше дисков, чем для **RAID 5**).

Создание БД

Создание БД выполняется после её физического проектирования

Физическая структура БД может быть изменена (добавлены, удалены файлы, группы) в процессе эксплуатации

Создание БД

Создание и изменения БД выполняется с использованием

- SSMS
- T-SQL

Создание БД в SSMS

Диалоговое окно SSMS для описания параметров создаваемой БД

Кнопка для определения свойств расширения файлов БД

Закладка для групп файлов БД

Расширение в процентах

Расширение в МБ

Неограниченное расширения файлов

Кнопка для добавления файла

Имя базы данных: DB
Владелец: <по умолчанию>

Использовать файлы группы:

Логическое имя	Тип файла	Файловая группа	Начальный размер	Авторасширение	Путь	Имя файла
DB	Данные строк	PRIMARY	20	4 МБ, рост не ограни...	D:\SQL_DB	DB_data.mdf
DB_log	Журнал	Не приме...	10	20 %, рост не ограни...	F:\DB_LOG	DB_log.ldf
DB2	Данные строк	DBSEC	20	3 МБ, рост не ограни...	E:\SQL_DB	DB2_data.ndf
DB3	Данные строк	PRIMARY	20	4 МБ, рост не ограни...	D:\SQL_DB	DB3_data.ndf
DB4	Д...	E:\SQL_DB	DB4_data.ndf

Change Autogrowth for Заказ2_ASS1

Enable Autogrowth

File Growth

In Percent (10)

In Megabytes (1)

Maximum File Size

Restricted File Growth (MB) (100)

Unrestricted File Growth

OK Cancel

Добавить Удалить

Создание БД T-SQL

CREATE DATABASE database_name

[**ON** [**PRIMARY**]

[< **filespec** > [, ... n]]

[, < **filegroup** > [, ... n]]

]

[**LOG ON** < **filespec** > [, ... n]]

[**COLLATE** collation_name]

[**FOR LOAD** | **FOR ATTACH**]

< **filespec** > ::= =

([**NAME** = location_file_name ,]

FILENAME = "os_file_name"

[, **SIZE** = size]

[, **MAXSIZE** = { max_size | UNLIMITED]

[, **FILEGROWTH** = growth_increment] ,

< **filegroup** > ::= =

FILEGROUP filegroup_name < filespec > [, ... n]

[Например](#), ...

Пример создания БД на T-SQL

CREATE DATABASE [DB] **ON PRIMARY**

(**NAME** = N'DB', **FILENAME** = N'D:\SQL_DB\DB_Data.MDF' ,
SIZE = 20, **MAXSIZE** = 9807, **FILEGROWTH** = 4) ,

(**NAME** = N'DB2', **FILENAME** = N'E:\SQL_DB\DB2_Data.NDF' ,
SIZE = 20, **MAXSIZE** = 9807, **FILEGROWTH** = 4),

FILEGROUP DBSEC

(**NAME** = N'DB3', **FILENAME** = N'D:\SQL_DB\DB3_Data.NDF' ,
SIZE = 20, **MAXSIZE** = 9807, **FILEGROWTH** = 4),

(**NAME** = N'DB4', **FILENAME** = N'E:\SQL_DB\DB4.NDF' ,
SIZE = 20, **MAXSIZE** = 9807, **FILEGROWTH** = 4)

LOG ON

(**NAME** = N'DB_Log', **FILENAME** = N'F:\DB_LOG\DB_Log.LDF' ,
SIZE = 10, **FILEGROWTH** = 20%)

COLLATE Cyrillic_General_CI_DB

Изменение БД в T-SQL

ALTER DATABASE database_name

ADD FILE < **filespec** > [,...n] [**TO FILEGROUP** filegroup_name]

ALTER DATABASE database_name

ADD LOG FILE < **filespec** > [,...n]

ALTER DATABASE database_name

ADD FILEGROUP filegroup_name

ALTER DATABASE database_name

REMOVE FILE Для удаления файла потребуется выполнить его сжатие. Для чего используется команда

ALTER DATA **DBCC SHRINKFILE** (location_file_name, **EMPTYFILE**)

REMOVE FILEGROUP logical_file_name

ALTER DATABASE

MODIFY NAME

Для изменения имени БД можно использовать системную хранимую

ALTER DATABASE sp_renamedb

SET < optionspec > [,...n] [**WITH** < termination >]

...

Размещение объектов БД в группе файлов

В существующую группу файлов можно поместить следующие типы объектов

- таблицы**
- индексы**
- данные типа TEXT, NTEXT, IMAGE**

Размещение объектов БД в группе файлов

Оператор создания таблицы

CREATE TABLE *имя_таблицы*

(*<описание_элемента_таблицы>* [,...])

[**ON** { имя_группы_файлов | **DEFAULT** }]

[**TEXTIMAGE_ON** { имя_группы_файлов | **DEFAULT** }]

Оператор создания индексов

CREATE [**UNIQUE**] **INDEX** *имя_индекса*

ON *имя_таблицы* (имя_столбца [**ASC** | **DESC**] [,...])

[**ON** имя_группы_файлов]

Дополнительные
предложения