

Лекция 12 ЛОГИЧЕСКАЯ СТРУКТУРА И ФИЗИЧЕСКАЯ РЕАЛИЗАЦИЯ БД

Структуру баз данных необходимо рассматривать на двух уровнях: логическом и физическом.

Логическая структура базы данных определяет структуру таблиц, взаимосвязи между ними, список пользователей, хранимые процедуры, правила, умолчания и другие объекты базы данных.

Физическая структура базы данных включает в себя описание файлов и групп файлов данных, журнала транзакций, первоначальный размер файлов, шаг прироста базы данных, ее максимальный размер, параметры конфигурации и другие физические характеристики.

ЛОГИЧЕСКАЯ СТРУКТУРА БД

Все логические компоненты базы данных SQL Server 2008 называются **объектами** и подразделяются на **10 типов**.

Для объектов можно устанавливать различные правила доступа со стороны пользователей.

Функциональное назначение объектов базы данных сервера SQL Server 2008 можно кратко определить следующим образом:

- 1.Table**
- 2. User-defined data type**
- 3.View**
- 4.Stored procedure**
- 5.Trigger**
- 6.Index**
- 7.Rule**
- 8.Constraint**
- 9.Default**
- 10.Function**

ФИЗИЧЕСКАЯ ОРГАНИЗАЦИЯ БД

Обращение к системным базам данных напрямую запрещено. Обращаться к ним можно только с помощью специально разработанных интерфейсов:

- a) С помощью системных хранимых процедур;
- b) С помощью интегрированной среды Enterprise Manager;
- c) Используя программные интерфейсы Transact-SQL (API).

Основная единица хранения данных на уровне файла базы данных – это страница, которая участвует в операциях ввода-вывода как единое целое даже тогда, когда требуется всего одна строка. Размер страницы равен 8 Кбайт. Файл журнала транзакций не имеет страниц и экстендов. Он содержит только последовательность записей транзакций, выполняемых в базе данных. Каждая страница файла базы данных имеет объем 8192 байт. Первые 96 байт страницы отводятся под заголовок, в котором хранится системная

информация: тип страницы, объем свободного места на странице, идентификационный номер таблицы или индекса – владельца страниц: Имеется шесть типов страниц:

1.Data.

2.Index.

3.Text/Image.

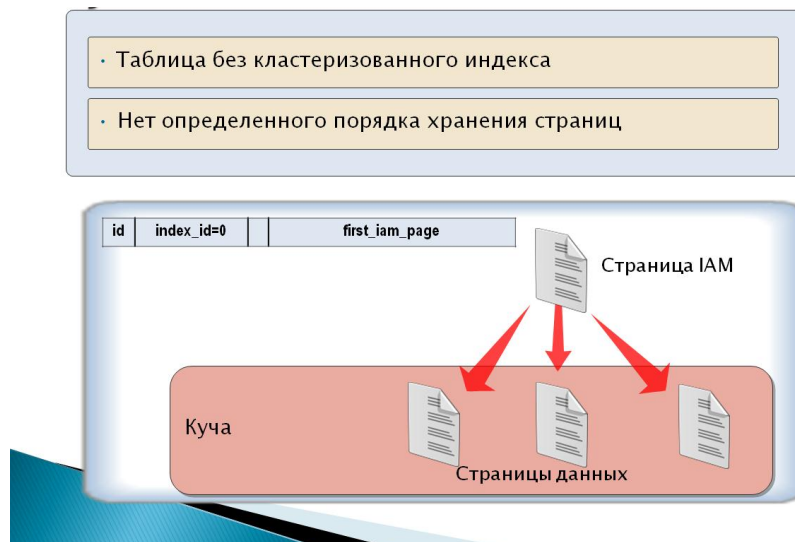
4.Global Allocation Map (GAM).

5.Page Free Space.

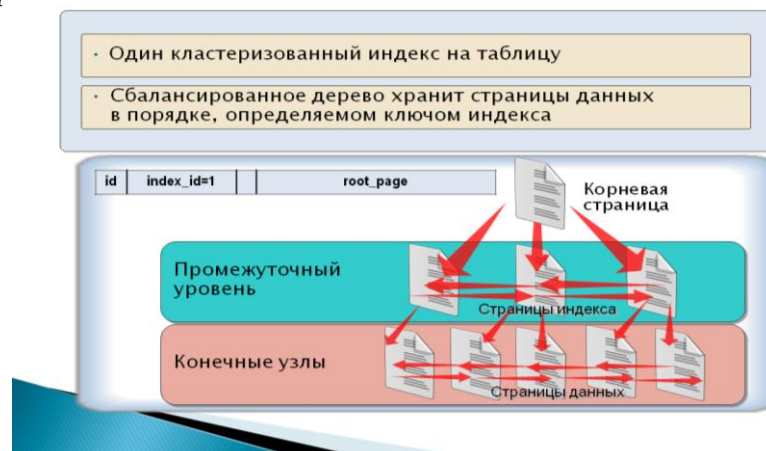
6.Index Allocation Map (IAM).

В SQL Server доступ к данным может осуществляться одним из двух способов: просмотром всех страниц данных в таблице или с помощью индексов.

Куча — это таблица без кластеризованного индекса. Для обслуживания кучи SQL Server использует страницы карты распределения индекса (Index Allocation Map, IAM).

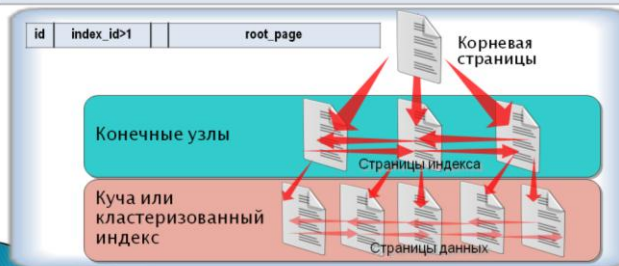


Кластеризованный индекс



Некластеризованный индекс

- Сбалансированное дерево ссылается на кучу или кластеризованный индекс
- До 249 некластеризованных индексов на таблицу



Синтаксис создания индексов

Используйте SQL Server Management Studio
или
выражение Transact-SQL CREATE INDEX

```
CREATE [UNIQUE] [CLUSTERED | NONCLUSTERED ]  
INDEX index_name ON { table | view } ( column [ ASC  
| DESC ]  
[ , . . . n ] )  
INCLUDE ( column [ , . . . n ] )  
[ WITH option [ , . . . n ] ]  
[ ON { partition_scheme (column) | filegroup |  
"default" } ]
```