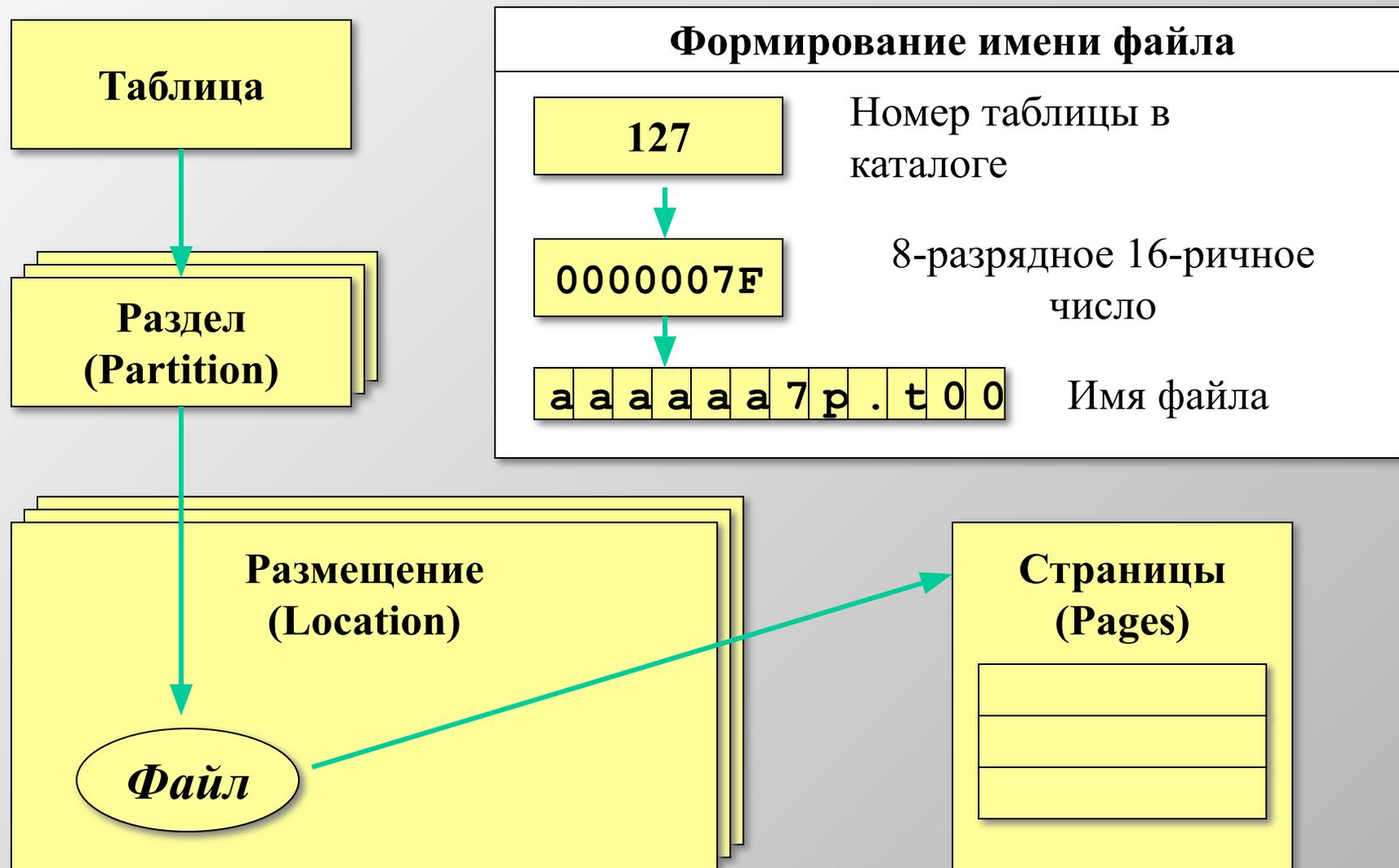




# Тема 4. Физические модели данных

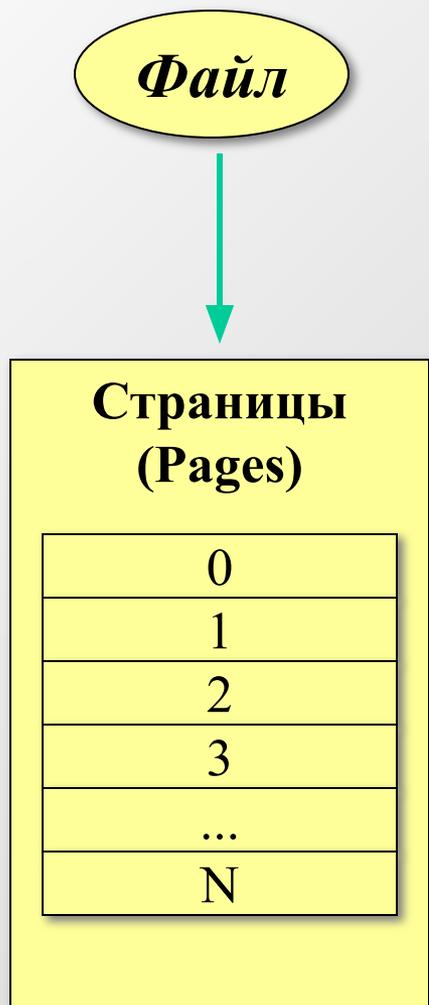


## Элементы физической модели





## Понятие страницы



| Размер страницы | Всего байт | Зарезервировано | Доступно |
|-----------------|------------|-----------------|----------|
| 2 Кбайт         | 2048       | 40              | 2008     |
| 4 Кбайт         | 4096       | 108             | 3988     |
| 8 Кбайт         | 8192       | 108             | 8084     |
| 16 Кбайт        | 16384      | 108             | 16276    |
| 32 Кбайт        | 32768      | 108             | 32660    |
| 64 Кбайт        | 64536      | 108             | 65428    |

Файл разбивается на страницы фиксированного размера, который задается при создании таблицы. Часть пространства страницы доступна для хранения данных, часть – зарезервирована для внутреннего использования.

Каждая страница может содержать определенное число записей (оно зависит от размера записи, но не более 512). Записи не могут пересекать границы страниц.



## Идентификатор записи

$tid = (npage * 512 + nrow)$ ,  
где  $npage$  – номер страницы;  
 $nrow$  – номер записи на странице

Каждая запись на странице данных имеет уникальный идентификатор записи  $tid$ . Эти идентификаторы используются для внутреннего управления данными, например в механизме вторичных индексов или в структуре хранения данных  $btree$ .

**Пример:**  
страница 2К,  
запись 500 байт

|   |      |
|---|------|
| 0 | 0    |
|   | 1    |
|   | 2    |
|   | 3    |
| 1 | 512  |
|   | 513  |
|   | 514  |
|   | 515  |
| 2 | 1024 |
|   | 1025 |
|   | 1026 |
|   | 1027 |



## Структуры хранения данных

Структура хранения определяет способ размещения данных на страницах файла. От выбора структуры хранения может зависеть эффективность выполнения запросов к таблице.

**HEAP**

**HASH**

**ISAM**

**BTREE**



## Размещение данных в структуре HEAP

В структуре heap записи всегда добавляются на последнюю страницу таблицы. Поэтому эта структура хранения является самой быстрой для добавления данных. Структура heap также является самой медленной для чтения, поскольку при любом запросе просматриваются все страницы таблицы. При удалении записей высвободившееся пространство не используется, пока таблица не будет реорганизована.

|                   | Номер | Фамилия     | Возраст | Зарплата  | Комментарий |
|-------------------|-------|-------------|---------|-----------|-------------|
| <i>Страница 0</i> | 7     | Иванов      | 23      | 28 000.00 |             |
|                   | 4     | Петров      | 33      | 32 000.00 |             |
|                   | 10    | Сидоров     | 34      | 35 000.00 |             |
|                   | 1     | Климов      | 40      | 43 000.00 |             |
| <i>Страница 1</i> | 8     | Федоров     | 53      | 46 000.00 |             |
|                   | 5     | Сергеев     | 23      | 22 000.00 |             |
|                   | 11    | Николаев    | 34      | 32 000.00 |             |
|                   | 2     | Александров | 50      | 55 000.00 |             |
| <i>Страница 2</i> | 9     | Никитин     | 43      | 41 000.00 |             |
|                   | 6     | Новиков     | 61      | 35 000.00 |             |
|                   | 12    | Андреев     | 41      | 22 000.00 |             |
|                   | 3     | Савельев    | 44      | 40 000.00 |             |



## Размещение данных в структуре HASH

Для структуры хранения HASH таблица должна иметь ключ, состоящий из одного или нескольких атрибутов. Записи размещаются на страницах, номера которых определяются по значению ключа с использованием специального алгоритма. Если на странице, вычисленной по значению ключа, нет места для размещения записи, то запись размещается на странице переполнения, добавляемой в конец таблицы.

|            | Номер | Фамилия     | Возраст | Зарплата  | Комментарий |
|------------|-------|-------------|---------|-----------|-------------|
| Страница 0 | 10    | Сидоров     | 34      | 35 000.00 |             |
| Страница 1 | 1     | Климов      | 40      | 43 000.00 |             |
|            | 11    | Николаев    | 34      | 32 000.00 |             |
| Страница 2 | 2     | Александров | 50      | 55 000.00 |             |
|            | 12    | Андреев     | 41      | 22 000.00 |             |
| Страница 3 | 3     | Савельев    | 44      | 40 000.00 |             |
| Страница 4 | 4     | Петров      | 33      | 32 000.00 |             |
| Страница 5 | 5     | Сергеев     | 23      | 22 000.00 |             |
| Страница 6 | 6     | Новиков     | 61      | 35 000.00 |             |
| Страница 7 | 7     | Иванов      | 23      | 28 000.00 |             |
| Страница 8 | 8     | Федоров     | 53      | 46 000.00 |             |
| Страница 9 | 9     | НИКИТИН     | 43      | 41 000.00 |             |



## Размещение данных в структуре ISAM

|            |    |             |    |           |  |
|------------|----|-------------|----|-----------|--|
| Страница 0 |    |             |    |           |  |
| Страница 1 | 1  | Климов      | 40 | 43 000.00 |  |
|            | 2  | Александров | 50 | 55 000.00 |  |
|            | 3  | Савельев    | 44 | 40 000.00 |  |
|            | 4  | Петров      | 33 | 32 000.00 |  |
| Страница 2 | 5  | Сергеев     | 23 | 22 000.00 |  |
|            | 6  | Новиков     | 61 | 35 000.00 |  |
|            | 7  | Иванов      | 23 | 28 000.00 |  |
|            | 8  | Федоров     | 53 | 46 000.00 |  |
| Страница 3 | 9  | Никитин     | 43 | 41 000.00 |  |
|            | 10 | Сидоров     | 34 | 35 000.00 |  |
|            | 11 | Николаев    | 34 | 32 000.00 |  |
|            | 12 | Андреев     | 41 | 22 000.00 |  |
| Страница 4 | 13 | Смирнов     | 39 | 32 000.00 |  |



## Особенности структуры ISAM

При приведении таблицы к структуре isam записи сортируются по значению ключа. Индекс организуется в виде двоичного дерева, в узлах которого находятся значения ключа записей, начинающих новую страницу. Число уровней индекса определяется числом страниц таблицы как степень числа 2, необходимая для покрытия общего числа страниц. При добавлении в таблицу или изменении данных индекс не изменяется. Эта структура хранения эффективна при доступе по точному значению ключа, по диапазону значений ключа и по левой части составного ключа. Структура isam не эффективна при быстром росте таблицы, поскольку при этом добавление записей идет на страницы переполнения.



## Размещение данных в структуре ВТРЕЕ

| Страница 0 |    |             | Стр.<br>1 | 1         | 153 | Стр. 2 | 7  | 205 |
|------------|----|-------------|-----------|-----------|-----|--------|----|-----|
|            |    |             |           | 2         | 153 |        | 8  | 205 |
|            |    |             |           | 3         | 153 |        | 9  | 256 |
|            |    |             |           | 4         | 185 |        | 10 | 200 |
|            |    |             |           | 5         | 204 |        | 11 | 256 |
|            |    |             |           | 6         | 204 |        | 12 | 256 |
|            |    |             |           | 9         | 3   |        |    |     |
| Страница 3 | 1  | Климов      | 40        | 43 000.00 |     |        |    |     |
|            | 2  | Александров | 50        | 55 000.00 |     |        |    |     |
|            | 3  | Савельев    | 44        | 40 000.00 |     |        |    |     |
|            | 4  | Петров      | 33        | 32 000.00 |     |        |    |     |
| Страница 4 | 5  | Сергеев     | 23        | 22 000.00 |     |        |    |     |
|            | 6  | Новиков     | 61        | 35 000.00 |     |        |    |     |
|            | 7  | Иванов      | 23        | 28 000.00 |     |        |    |     |
|            | 8  | Федоров     | 53        | 46 000.00 |     |        |    |     |
| Страница 5 | 9  | Никитин     | 43        | 41 000.00 |     |        |    |     |
|            | 10 | Сидоров     | 34        | 35 000.00 |     |        |    |     |
|            | 11 | Николаев    | 34        | 32 000.00 |     |        |    |     |
|            | 12 | Андреев     | 41        | 22 000.00 |     |        |    |     |



## Особенности структуры BТREE

В структуре btree индекс растет при увеличении таблицы, что исключает проблемы, связанные с появлением страниц переполнения, которые характерны для структуры isam. В структуре btree таблица состоит из четырех частей: страницы заголовка, индексных страниц, страниц списка (leaf pages) и страниц данных. В этой структуре, в отличие от isam, индекс указывает не на страницы данных, а на страницы списка (leaf pages), содержащие пары: значение ключа – tid записи на странице данных. Индекс и страницы списка перестраиваются при необходимости по мере увеличения таблицы.

Так же, как и isam, структура хранения btree эффективна при доступе по точному значению ключа, по диапазону значений ключа и по левой части составного ключа. Поскольку в структуре btree индекс указывает на компактные страницы списка, то число ступеней в индексе будет значительно меньше, чем в структуре isam. К недостаткам btree можно отнести динамическое перестроение таблицы в процессе работы (может замедлять работу пользователей при интенсивном обновлении таблицы), а также риск возникновения дедлока.



## Сравнение структур хранения данных

| Характеристика                           | HEAP | HASH | ISAM | BTREE |
|--|------|------|------|-------|
| Наличие ключа                            | -    | +    | +    | +     |
| Страницы переполнения                    | -    | +    | +    | -     |
| Быстрое добавление                       | +    | -    | -    | -     |
| Быстрый доступ по точному значению ключа | -    | +    | +    | +     |
| Быстрый доступ по диапазону ключа        | -    | -    | +    | +     |
| Быстрый доступ по левой части ключа      | -    | -    | +    | +     |
| Перестроение индекса                     | -    | -    | -    | +     |
| Риск дедлока                             | -    | -    | -    | +     |