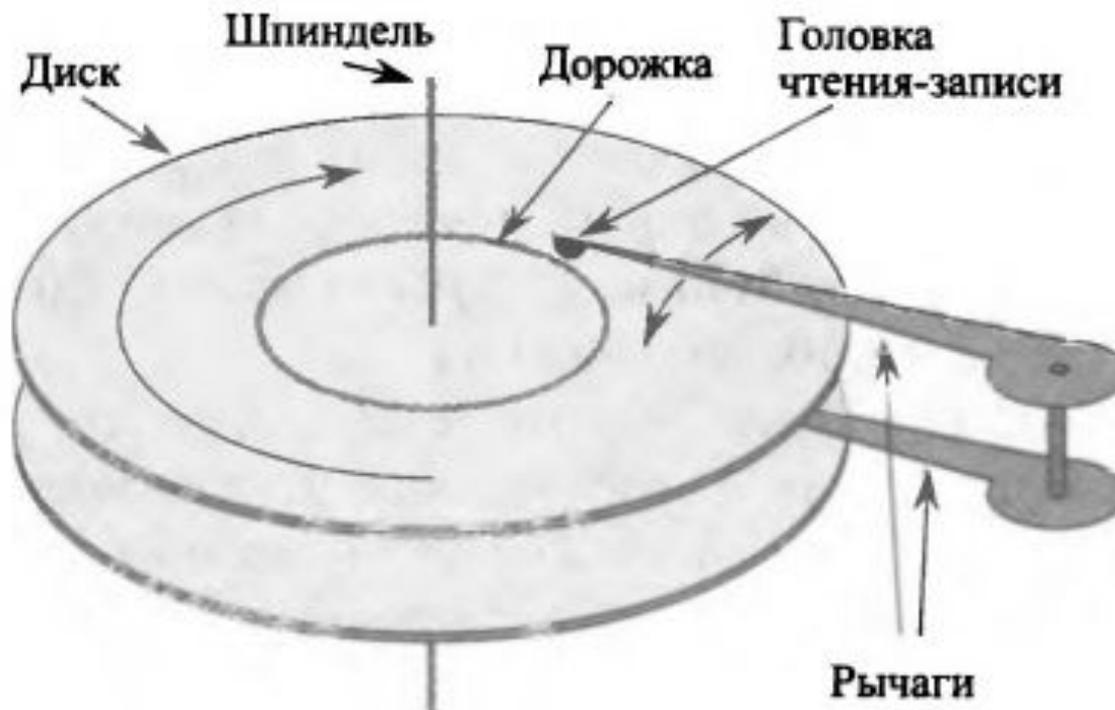


Структуры данных во вторичной памяти

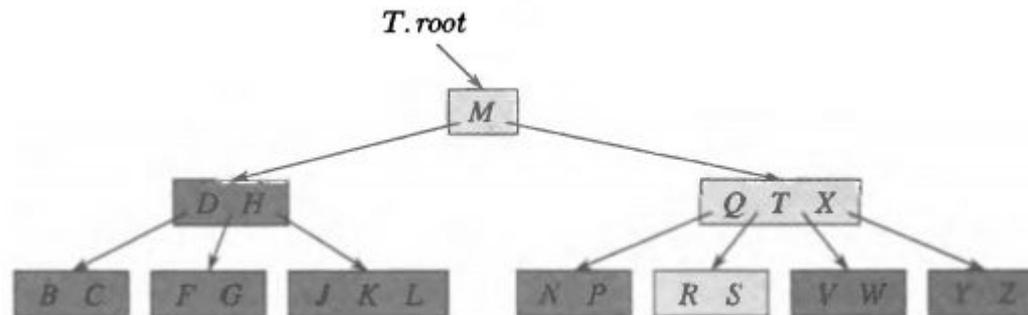


Типичный дисковый накопитель

Определение B-деревьев

B-дерево T представляет собой корневое дерево (корень которого $T.root$), обладающее следующими свойствами.

1. Каждый узел x содержит следующие атрибуты:
 - а) $x.n$, количество ключей, хранящихся в настоящий момент в узле x ;
 - б) собственно $x.n$ ключей, $x.key_1, x.key_2, \dots, x.key_{x.n}$, хранящихся в неубывающем порядке, так что $x.key_1 \leq x.key_2 \leq \dots \leq x.key_{x.n}$;
 - в) булево значение $x.leaf$, равное TRUE, если x представляет собой лист, и FALSE, если x является внутренним узлом.



Определение B-деревьев

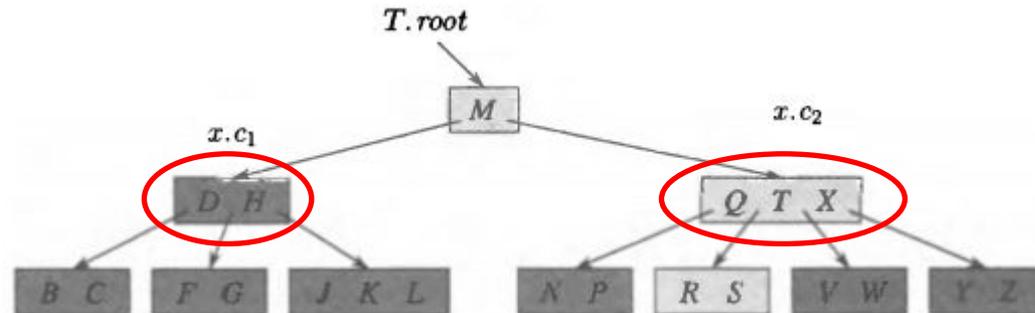
2. Кроме того, каждый внутренний узел x содержит $x.n + 1$ указателей $x.c_1, x.c_2, \dots, x.c_{x.n+1}$ на дочерние узлы. У листьев дочерних узлов нет, так что значения их атрибутов c_i не определены.
3. Ключи $x.key_i$ разделяют поддиапазоны ключей, хранящихся в поддеревьях: если k_i является произвольным ключом, хранящимся в поддереве с корнем $x.c_i$, то

$$k_1 \leq x.key_1 \leq k_2 \leq x.key_2 \leq \dots \leq x.key_{x.n} \leq k_{x.n+1}.$$

D - $x.key_1$

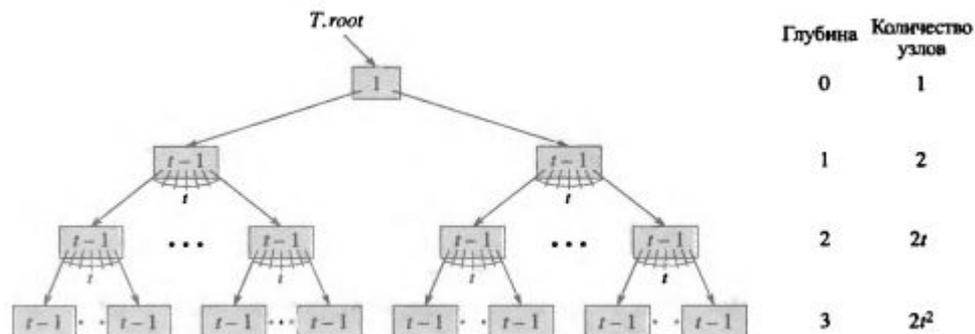
H - $x.key_2$

B - k_1



Определение B-деревьев

4. Все листья расположены на одной и той же глубине, которая равна высоте дерева h .
5. Имеются нижняя и верхняя границы количества ключей, которые могут содержаться в узле. Эти границы могут быть выражены с помощью одного фиксированного целого числа $t \geq 2$, называемого *минимальной степенью* (minimum degree) B-дерева:
 - а) каждый узел, кроме корневого, должен содержать как минимум $t-1$ ключей. Каждый внутренний узел, не являющийся корневым, имеет, таким образом, как минимум t дочерних узлов. Если дерево не является пустым, корень должен содержать как минимум один ключ;
 - б) каждый узел содержит не более $2t-1$ ключей. Таким образом, внутренний узел имеет не более $2t$ дочерних узлов. Мы говорим, что узел *заполнен* (full), если он содержит ровно $2t-1$ ключей².



Поиск в B-дереве

B-TREE-SEARCH(x, k)

```
1  $i = 1$ 
2 while  $i \leq x.n$  и  $k > x.key_i$ 
3    $i = i + 1$ 
4 if  $i \leq x.n$  и  $k == x.key_i$ 
5   return ( $x, i$ )
6 elseif  $x.leaf$ 
7   return NIL
8 else DISK-READ( $x.c_i$ )
9   return B-TREE-SEARCH( $x.c_i, k$ )
```

x - указатель на корневой элемент поддерева

k - ключ, поиск которого ведется

