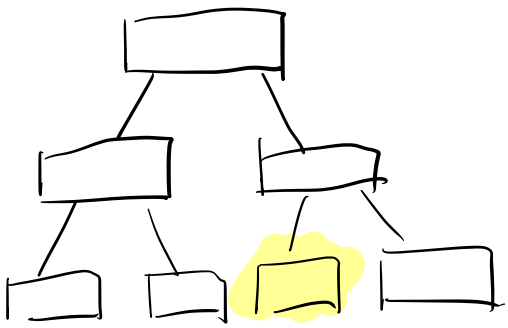
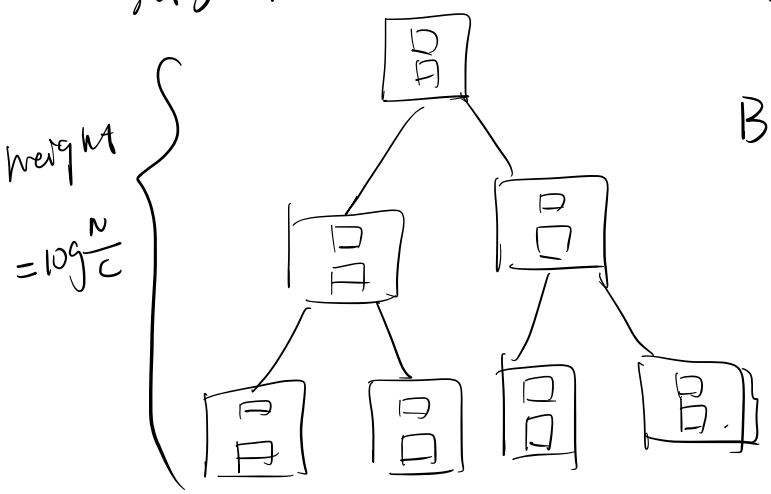
ODS - Btree



查找 $O(m \log_m n)$

m为阶数. $\log n$.

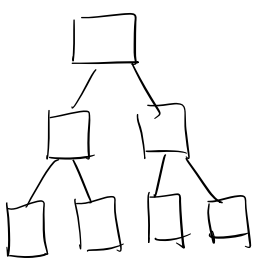
sub-rORAM (Ri : range = 2^i)



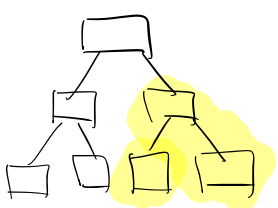
Btree : $n = \frac{N}{2^i}$

$m \log(\frac{mN}{2^i}) \cdot \log_C^N$

sub-rORAM (Rj , $j < i$)



Btree

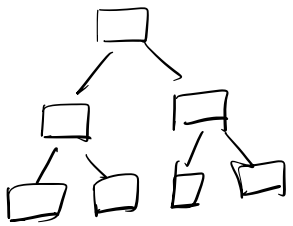


$m \log_m n + (2^{i-j} - 1)$

$[m \log_m n + (2^{i-j} - 1)] \cdot \log_C^N$

用范围为4的 rORAM+ODS-Btree 查大小为4的query

① Btree (假设扇叉为2)



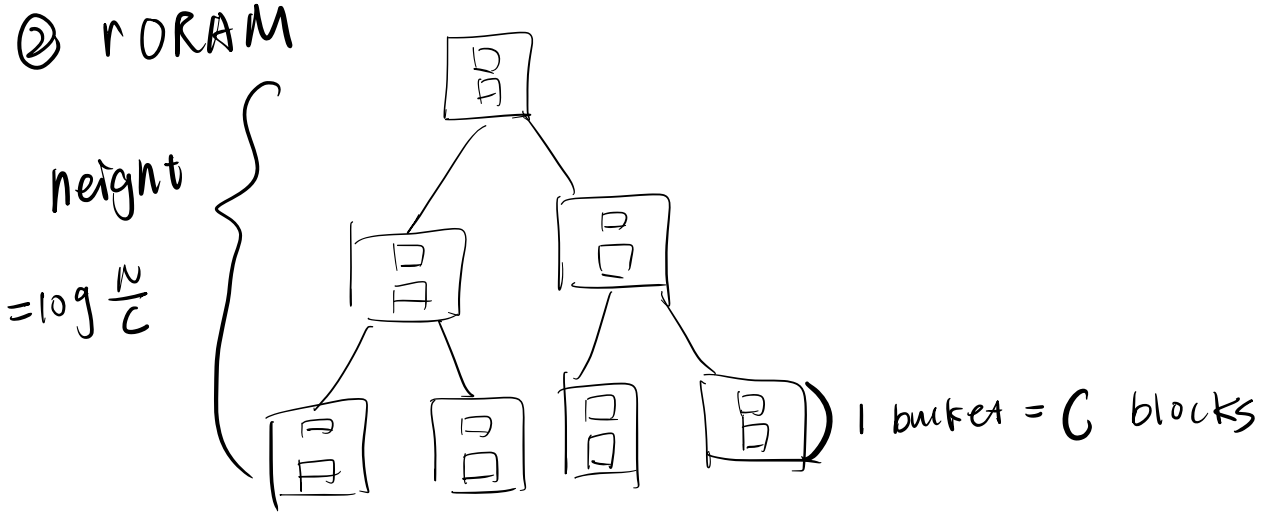
大小为: $\frac{N}{4}$ (N为总共node个数)

层数为 $\log\frac{N}{4} = \log N - 2$

需从Btree提取 $\frac{4}{4} = 1$ 个点

→ 复杂度为 $O(\log N - 2)$

② rORAM

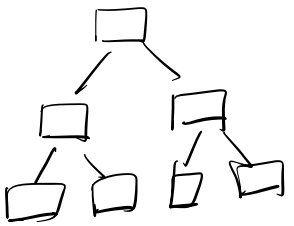


→ 复杂度为 $(\log\frac{N}{C})$

⇒ 总复杂度为 $O((\log N - 2)\cdot\log\frac{N}{C})$

用范围为2的 rORAM+ODS-Btree 查大小为4的query

① Btree (假设扇叶为2)



大小为: $\frac{N}{2}$ (N为总共node个数)

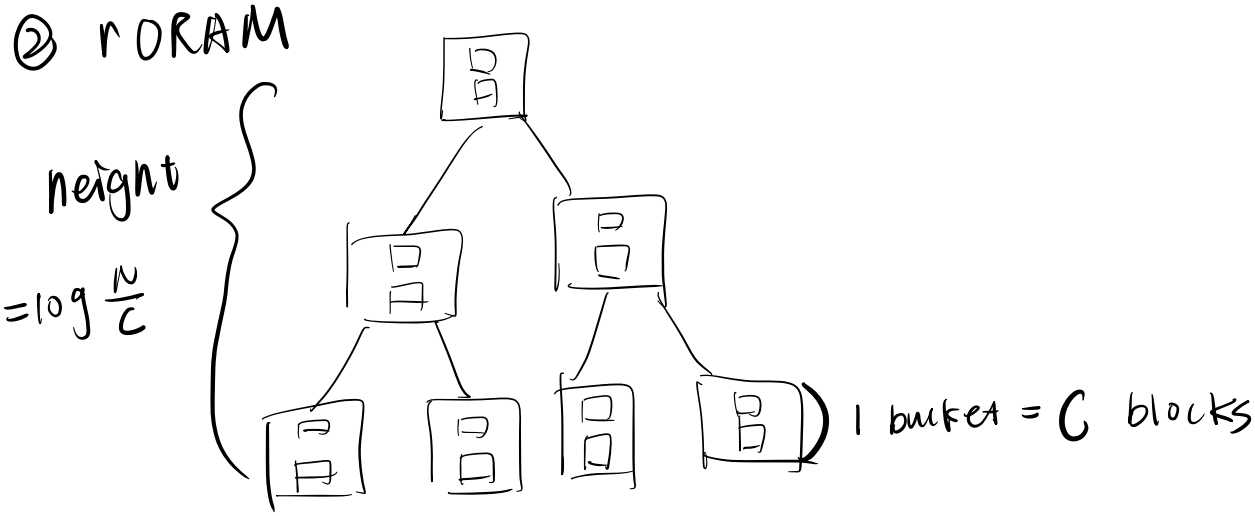
层数为 $\log\frac{N}{2} = \log N - 1$

需从Btree提取 $\frac{4}{2} = 2$ 个点

→ 复杂度为 $O(\log N - 1 + 2)$

找到第1个点　找到范围内下一个点

② rORAM



→ 复杂度为 $(\log\frac{N}{C})$

⇒ 总复杂度为 $O((\log N + 1)\cdot\log\frac{N}{C})$

相差倍数: $\frac{\log N + 1}{\log N - 2}$ (与rORAM无关)

用范围为 i 的 rORAM+ODS-Btree 查大小为 i 的 query

B+tree: $O(\log \frac{N}{i})$

用范围为 j 的 rORAM+ODS-Btree 查大小为 i 的 query $(j < i)$

Btree: 找到第1个点: $\log \frac{N}{j}$

找到后续 $\frac{i}{j} - 1$ 个点: $\frac{i}{j}$

$\rightarrow O(\log \frac{N}{j} + \frac{i}{j})$

用于 rORAM 不影响比例

$\rightarrow$ 差值比为 $\frac{\log N - \log j + \frac{i}{j}}{\log N - \log i}$

当 $j = 1$

$i = \frac{N}{\log^2 N} \rightarrow \frac{\log N + \frac{N}{\log^2 N}}{\log N - \log N + 2\log\log N}$

$= \frac{\log^3 N + N}{2\log\log N \cdot \log^2 N}$