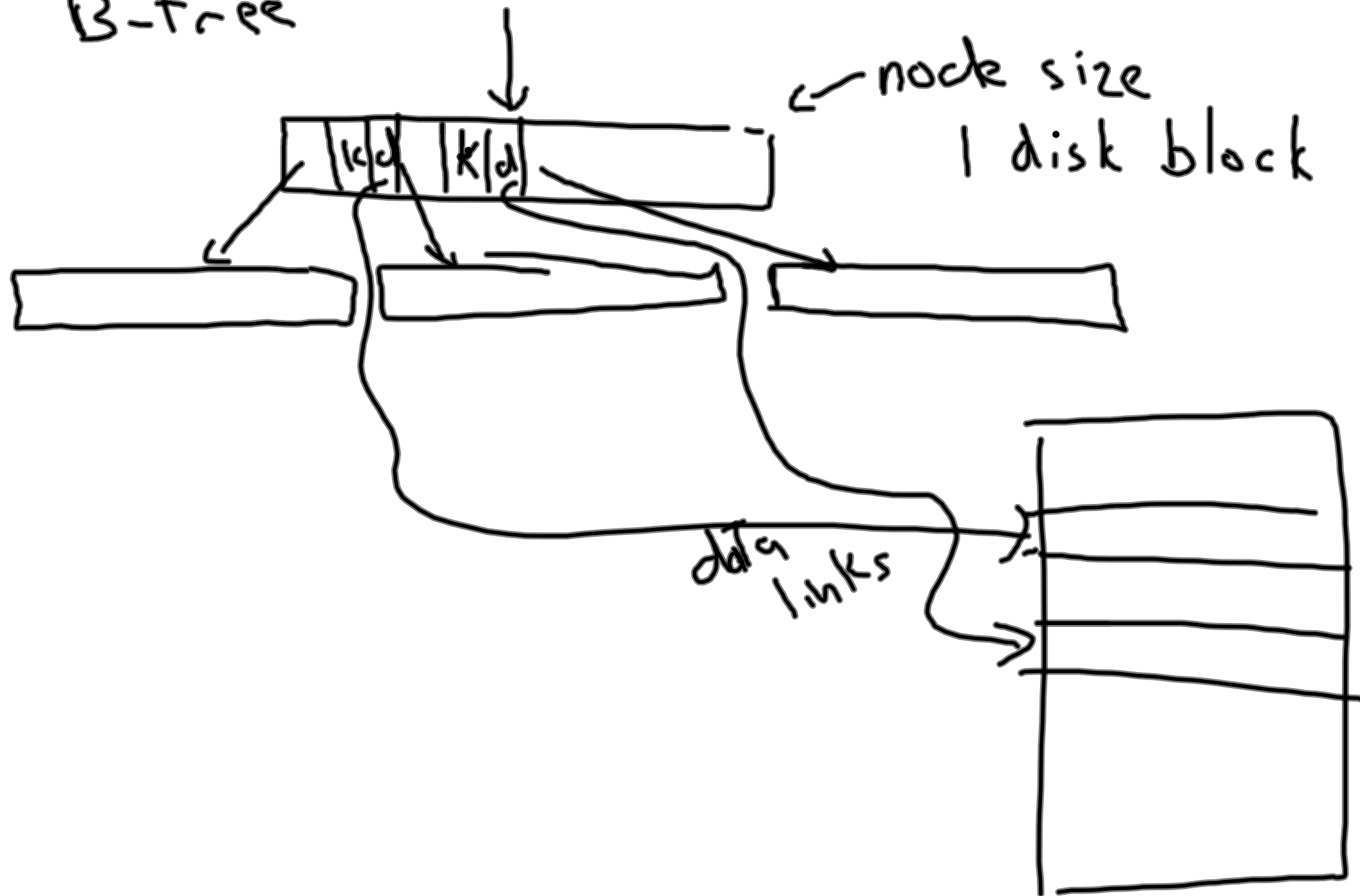


B-tree



DB

act#	balance
------	---------

Deposit (b)

Read(balance)
balance += b
Write(balance)

Transfer (amt)

transaction

Read (bal 1)
Read (bal 2)
bal 1 -= amt
bal 2 += amt
Write (bal 1)
Write (bal 2)

Deposit (500)
acct 1

Transfer (200)
acct 1 to acct 2

read (acct1.bal)
read (acct2.bal)
bal1 -= 200
bal2 += 200

read (acct 1 . bal)
bal += 500
write (acct 1 . bal)

write (acct1.bal)
write (acct2.bal)

Lost update

Transaction operations

- Begin transaction
 - read/write
 - end
-
- commit
 - rollback

System log

start trans. T_id
write T_id, old_value,
new_value, X
read T_id, X ← (item)
commit T_id
abort T_id

Transactions (ACID properties)

Atomicity - transactions cannot be split up

Consistency - Transaction changes DB from one correct state to another

Isolation - updates should not be affected by other transactions

Durability - once changes are committed they should not be lost from a later failure.

Transaction schedule

Recoverable

Serializable

