# Operating System Foundations - Solution 2001

p10q8
JMB

Buffer only - write through synchronously to disc
- read: block may be in memory from a previous read, if not, read disc.

Disc block cache: read as ^, write to block in memory
write to disc when ~~block~~ buffer needed to store some other block
(and probably periodically eg UNIX) 2

Pro - fast response to write

Con - "done" to application on write does not ⇒ data on disc
∴ can't use for tx. system. May be lost on a crash.
Writes could be re-ordered + interaction with crashes. 3

Concurrency Control

App. (top-down) access to buffer must wait for device drive
to fill buffer before a read.

Access to all data such as free list + block headers must take place under mutual exclusion to avoid race condition / dual allocation etc.

6 Must wait for a dirty block to be written out before filling

typical header of a disc block buffer:

filesystem / block (VID)
owner: PID.
dirty (changed)
valid? (being filled / emptied)
locked (for investigation)
links for hash chain + free list

eg: Read (bytes) → block ID → in cache? → YES → valid ⟨ N wait / Y. read
↘ N.

(waiting involves sleep + wakeup when event signalled)

N: lock free chain - try first block - dirty? ↗ Y. write out try next. ↘ N allocate → read (mark inval wait (valid)

Write (bytes) → block ID → in cache? → YES → valid $\xrightarrow{Y}$ write mark dirty
↘ N. N ↘ wait ↗

9 N: get a new buffer as for read.

< expand ).

(20)