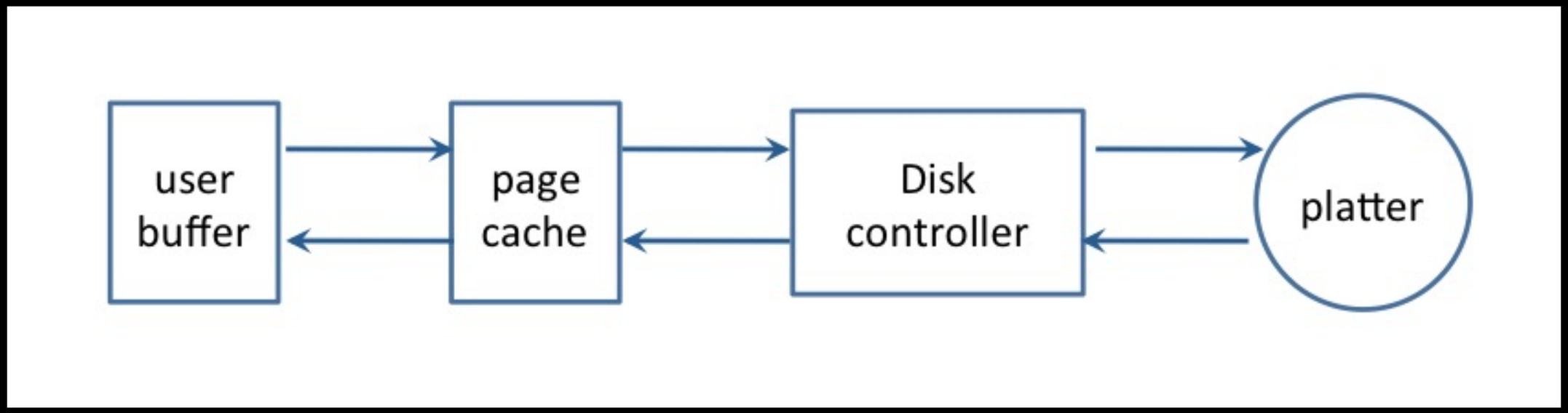
Disk and Page cache

chenzongzhi

Write path



Write Path

- The client sends a write command to the database (data is in client's memory).
- The database receives the write (data is in server's memory).
- The database calls the system call that writes the data on disk (data is in the kernel's buffer).
- The operating system transfers the write buffer to the disk controller (data is in the disk cache).
- The disk controller actually writes the data into a physical media (a magnetic disk, a Nand chip, ...).

Numbers

Numbers Everyone Should Know

L1 cache reference	0.5 ns
Branch mispredict	5 ns
L2 cache reference	7 ns
Mutex lock/unlock	25 ns
Main memory reference	100 ns
Compress 1K bytes with Zippy	3,000 ns
Send 2K bytes over 1 Gbps network	20,000 ns
Read 1 MB sequentially from memory	250,000 ns
Round trip within same datacenter	500,000 ns
Disk seek	10,000,000 ns
Read 1 MB sequentially from disk	20,000,000 ns
Send packet CA->Netherlands->CA	150,000,000 ns

Page cache

```
[chenzongzhi@mdb9921:~]$ sudo sysctl -a | grep dirty
[sudo] password for chenzongzhi:
vm.dirty_background_ratio = 10
vm.dirty_background_bytes = 0
vm.dirty_ratio = 20
vm.dirty_bytes = 0
vm.dirty_writeback_centisecs = 500
vm.dirty_expire_centisecs = 3000
```

Page Cache

- vm.dirty_background_ratio

Page cache

- vm.dirty_ratio

Example

Page Cache

- vm.dirty_writeback_centisecs

Page Cache

- vm.dirty_expire_centisecs

Situation I

- Decreasing the Cache
- vm.dirty_background_ratio = 5
- vm.dirty_ratio = 10

Situation II

- Increasing the Cache
- vm.dirty_background_ratio = 50
- vm.dirty_ratio = 80

Situation III

- Online Server
- vm.dirty_background_ratio = 5
- vm.dirty_ratio = 80

Why

Page Cache History

History

- Before 2.6.32: Linux used “Pdflush”
- After 2.6.32: Linux used Backing Device Info(BDI) flush threads
- After 3.10.0: Linux used kworker

Why