Database servers have hardware requirements different from other infrastructure software, specifically unique demands on I/O and memory. This presentation covers these differences and various I/O options and their benefits.

https://momjian.us/presentations

Creative Commons Attribution License

Last updated: April 2022
Outline

• CPU
• Multi-threading
• GHz
• Pipelining
• SMP
• NUMA
Nope!

- CPU
- Multi-threading
- GHz
- Pipelining
- SMP
- NUMA
Normal Server Priorities

- I/O
- Memory
- CPU
Database Server Priorities

- I/O
- Memory
- CPU
Traditional servers are often CPU constrained because of:

- Network overhead (http)
- Text processing (email)
- Virtual machines (application servers)
- Application code
Database Server’s Unique Requirements

- Sequential scans of large tables
- Index scans causing random I/O
- Unpredictable query requirements
- Reporting

These do not require major CPU resources.
ACID (D = durability) requires committed transactions to be stored permanently. Few other server facilities must honor this requirement.
Magnetic Disk I/O Stack

- **PostgreSQL Shared Buffer Cache**
- **Kernel Disk Buffer Cache**
- **HBA/RAID Cache**
- **Disk Cache**
- **Magnetic Disk**

- **Write-Ahead Log**
  - fsync

- **PostgreSQL Shared Buffer Cache**
  - fsync

- **Kernel Disk Buffer Cache**
  - fsync

- **HBA/RAID Cache**
  - write-back

- **Disk Cache**
  - write-through
Magnetic Disk I/O Stack With BBU

- PostgreSQL Shared Buffer Cache
- Kernel Disk Buffer Cache
- HBA/RAID Cache
- Disk Cache
- Magnetic Disk

- Write-Ahead Log
  - fsync
- BBU

- fsync
  - write-back
Flash (NAND) Storage I/O Stack

- PostgreSQL Shared Buffer Cache
- Kernel Disk Buffer Cache
- HBA/RAID Cache
- Disk Staging Cache
- Flash (NAND) Solid State Disk (SSD)

**Write-Ahead Log**
- fsync

**Kernel Disk Buffer Cache**
- fsync

**Flash (NAND) Solid State Disk (SSD)**
- write-back
DRAM Storage I/O Stack

- PostgreSQL Shared Buffer Cache
- Kernel Disk Buffer Cache
- HBA/RAID Cache
- DRAM Solid State Disk (SSD)

- Write-Ahead Log
  - fsync

- DRAM Storage I/O Stack

- fsync
Write-Back vs. Write-Through Caching

- Write-back caching returns write success before passing data to lower storage layers
- Write-through caching waits for write acknowledgement from lower storage layers before returning success
• HBA/RAID cache behavior is usually controlled by the HBA/RAID firmware, often conditionally based on the health of the BBU
• Storage drive cache behavior can be set by utility commands or by using certain operating system calls
• Enterprise/SAS storage devices usually default to write-through, while consumer/SATA devices usually default to write-back
• HBA/RAID controllers often set storage drive caching mode to write-through
• With an HBA/RAID non-volatile cache, there is little advantage to using write-back mode on storage drives
Durable Cache: Battery and Supercapacitor-Backed Units

- Verify battery or supercapacitor (supercap) existence visually
- Most write the cache to local flash memory on power failure
- Battery-backed units (BBU)
  - Detected battery failure can disable write-back cache mode
  - Requires failure monitoring and replacement
Battery-Backed Unit on RAID Controller

https://www.flickr.com/photos/jeminus/
Supercapacitor-Backed Unit on SSD

Also called “power loss protection”
Magnetic Disk Selection

- More small spindles is better than fewer large spindles
- RAID 5/6 is too slow for database writes
- RAID 10 is popular
- Make sure SMART reporting is fully supported
- SAS/SCSI disks are usually designed for enterprise workloads, unlike SATA/ATA
  - reliability
  - error reporting
  - 24-hour operation
  - heat
  - vibration
SSD Selection

- **Flash (NAND) vs. DRAM**: [https://appuals.com/ssd-buying-guide/](https://appuals.com/ssd-buying-guide/)
- Write staging area — it is not just cache
- Running a flash (NAND) SSD in write-through mode can reduce its usable life because of increased write cycles
• Best for WAL and random I/O, e.g., indexes
• Set `random_page_cost = 1.1`
• Set `effective_io_concurrency` to 256 for SATA, 1,000 for NVMe
• xfs or ext4 over ext3
• Reduce file system logging, particularly for /pg_wal directory
• Disable access (atime) recording
Shared Storage

- SAN and NAS replace direct-attached storage (DAS) with shared storage
- Often used for easier storage management
- Shared I/O resource
- Databases often wait for I/O completion, meaning they have to contend with shared resource contention
- SAN serves block devices, NAS serves file systems
• The more RAM, the better; this reduces I/O requirements
• Ideally, five minutes of your working set
• The more RAM, the more possibility of RAM failure
• Use ECC (Error Correction Codes) RAM
  • detect errors
  • correct errors
  • report faulty memory
  • cosmic radiation
• Parallel query allows a single session to use multiple CPUs
• Heavy use of server-side functions might generate significant CPU load
• CPUs can become a bottleneck if the entire database fits in RAM and the workload is read-only
Not the Same

Just because something has the same interface doesn’t mean has the same capabilities. Compatible computer hardware is not all the same.

https://www.flickr.com/photos/cdevers/