Enezis

Disk queueing fundamental laws

This note is intended to present the basic issues of I/O bottleneck removals and the fundamental laws on which the models are based. The laws that dictate queuing modelling are described in many books and cannot be explained here at length. Only the formulas described below are used in the model spreadsheets.

DISKS

Definitions :

- TransferTime : time to transfer a block from the disk to the disk controller
- Block Size : disk I/O size
- $\overline{S}_d$: Disk service time = average time spent at the controller plus disk to access a block from a disk.
- TransferRate : Rate (in MB/s) at which data is transferred to/from a disk
- ControllerTime : time to transfer a block from the disk to its controller
- $Seek_{rand}$: Average seek time for a request to a random cylinder. Seek time is the the time to position the disk arm at the proper cylinder.
- DiskRevolutionTime : Time for a disk to complete a full revolution (60/DiskSpeed)
- RunLength : sequential workloads exhibit subsequences, called runs, of requests to consecutive blocks on the disk. RunLength is the average run length observed in a workload.
- λ_d : Arrival rate (aka I/O/s)
- U_d : Utilisation of a disk (as seen in sar –d on host attached disk). Percentage of time a disk is busy servicing a request. $U_d = \lambda_d S_d$

Single disks

Utilization law : $U_d = \lambda_d \overline{S_d}$

TransferTime= $\frac{BlockSize}{10^6 \times TransferRate}$

Random workloads :

$$\overline{S}_d = ControllerTime + Seek_{rand} + \frac{DiskRevolutionTime}{2} + TransferTime$$

Sequential workloads

$$\overline{S}_d = ControllerTime + \frac{Seek_{rand}}{RunLength} + \frac{[1/2 + (RunLength - 1)(1 + U_d)/2] \times DiskRevolutionTime}{RunLength} + \frac{TransferTime}{RunLength}$$

Disk arrays

Définitions :

- StripeUnit : size in bytes of a stripe unit
- StripeGroupSize : size in bytes of the stripe group assumed to be a multiple of StripeUnit
- n_r : number of stripe units read by a read request
- n_w : number of stripe units modified by a write request
- λ^r_{array} : arrival rate of read requests to a disk array
- λ^w_{array} : arrival rate of writes requests to a disk array
- λ^r_{disk} : arrival rate of read requests to any of the disks in the array
- λ^w_{disk} : arrival rate of write requests to any of the disks in the array
- N : number of disks in the array
- S^r_{disk} : average service time at a disk array for read requests
- S^w_{disk} : average service time at a disk array for write requests
- $rw(n_w)$: number of stripe units read as a result of a request to write n_w stripe units.

Number of reads generated by writes on a RAID 5 4+1 array (please read section 1.4.2 for RAID5 performance considerations) :

n_w	$rw(n_w)$	Explanation
1	2	Read one stripe unit and the parity block
2	2	Read two additional stripe units to compute the parity
3	1	Read one more stripe units to compute the parity
4	0	No additional reads needed

- R^r_{diski} : average response time of read requests at disk i.
- R^w_{diski} : average response time of write requests at disk i.

Arrival rate of read and write requests to component disks :

$$\lambda_{disk}^{r} = \frac{n_r}{N} \times \lambda_{array}^{r} + \frac{rw(n_w)}{N} \times \lambda_{array}^{w}$$

$$\lambda_{disk}^{w} = \frac{(n_w + 1)}{N} \times \lambda_{array}^{w}$$

Service time for reads and writes for the disk array :

$$S_{disk}^{r} = \max_{i=1}^{n_r} \{R_{diski}^{r}\}$$

$$S_{disk}^{w} = \max_{i=1}^{rw(n_r)} \{R_{diski}^{r}\} + \max_{i=1}^{n_w+1} \{R_{diski}^{w}\}$$

Utilisation of individual disks

$$U_d = (\lambda_{disk}^{r} + \lambda_{disk}^{w})[Seek_{rand} + \frac{DiskRevolutionTime}{2} + \frac{StripeUnit(bytes)}{10^6 \, xTransferRate}]$$