Performance Engineering Cookbook
Ingredients for Performance and Capacity Success

Capacity – How Full is Your Plate?

Peter van Eijk

This is a series of brief articles explaining the basic concepts of systems performance and capacity planning. Motivated by the Computer Measurement Group, these concepts are applicable to IT systems and beyond.

Capacity

In the theory of computation, computers have infinite capacity. In reality, that capacity is limited. Hence, in the practical world we have to worry about those limits and the effect they have on the performance of our systems. To me, capacity and performance are two sides of the same medal: the more of the available capacity is used, the worse any performance dependent on that capacity typically will be. Later in this series we will investigate that relationship further.

According to ITIL, capacity is "The maximum Throughput that a Configuration Item or IT Service can deliver whilst meeting agreed Service Level Targets. For some types of CI, Capacity may be the size or volume, for example a disk drive."

There are a number of ways capacity is expressed. Most often it is in units per time, such as Megabits per second, transactions per second, helpdesk calls per hour or IOPS (Disk IO per second). In the case of storage, capacity can also indicate the size, for example in Megabytes.

A related concept is utilization, oftentimes expressed as a percentage of capacity, in particular when it is about CPU utilization.

There are a few confusing things about capacity. To begin with, different disciplines have different conventions. For example, to a communications engineer 64 KB is 64000 bits, whereas to a computer engineer 64 KB equals 65536 bytes or 524288 bits. Some effort in cleaning up this sloppiness is going on (see link section on binary prefix). Proper use would be 64 kilobits and 64 kilobytes (KiB), respectively.

Another important issue is that capacity is sometimes a fundamental physical property, such as the size of a hard disk, but can also be a dynamically emerging behavior, such as the number of operations a hard disk or permanent storage system can handle. That 'capacity' is very dependent on the workload, i.e. writes are often faster than reads, and sequential access is mostly faster than random access.

Finally, an often neglected source of performance problems is the capacity of so-called software resources. Examples of these include buffers, connection pools and address spaces. These can also run out, and applications then often manifest 'hick-ups', or worse.

Link farm

Introductory:
Wikipedia definition: http://en.wikipedia.org/wiki/Capacity_planning
Wikipedia on units: http://en.wikipedia.org/wiki/Binary_prefix

Note to readers: are there any concepts here that need further elaboration? We want volunteers to find more link-worthy pages in sources such as the CMG archives, Wikipedia, and for linking back from Wikipedia to these pages. Please write to the author: Dr. Peter HJ van Eijk at pveijk@nlcmg.nl.