The Ten Cardinal Rules of Capacity Planning

By Rich Schiesser in conjunction with Harris Kern's Enterprise Computing Institute

A formal capacity planning program is usually one of the last processes an infrastructure will implement. There are many reasons for this, but chief among them is that this is a strategic process that belongs in a highly tactical environment. Infrastructures typically shine when it comes to day-to-day problem solving, spur-of-the-moment fire-fighting and short-term resolutions of issues. It isn't so much that infrastructures are adverse to strategic processes. More likely, they rarely get to strategic activities due to the daily onslaught of tactical matters to address. This article describes the ten cardinal rules for planning for the adequate capacity of computer resources within an infrastructure.

Cardinal Rule #1: Agree on a Common Definition of Capacity Planning

Capacity planning means different things to different people. Agreeing on a common, formal definition of the process is key to designing and implementing an effective capacity planning program. An example of such a definition could be: a process to predict the types, quantities, and timing of critical resource capacities that are needed within an infrastructure to meet accurately forecasted workloads. Care should be taken to distinguish the concept of capacity planning which is a long-range, strategically oriented process, from that of capacity management which tends to be a short-term, tactically oriented activity.

A person's perspective plays a key role in capacity planning. For example, a server operating at 60% capacity may be great news to a performance specialist who is trying to optimally tune response times. But, to an IT financial analyst trying to optimize resources from a cost standpoint, this may be disturbing news of unused resources and wasted costs. A formal, common definition of capacity planning can help to explain and bridge these two perspectives.

Cardinal Rule #2: Select a capacity planning process owner.

The second cardinal rule in developing a robust capacity planning process is to select an appropriately qualified individual to serve as the process owner. This person will be responsible for designing, implementing, and maintaining the process and will be empowered to negotiate and delegate with developers and other support groups.

Above all else, this individual must be able to communicate effectively with developers because much of the success and credibility of a capacity plan depends on accurate input and constructive feedback from developers to infrastructure planners. This person also needs to be knowledgeable on systems and network software and components, as well as with software and hardware configurations. Other recommended characteristics include having the ability to think and act strategically and a reasonable working knowledge of an organization's critical applications. The relative preference of these traits will obviously vary from shop to shop, depending on the types of applications provided and services offered.

Cardinal Rule #3: Identify key resources to be measured.

Once the process owner is selected, one of his or her first tasks will be to identify the infrastructure resources that are to have their utilizations or performance measured. This determination is made based on current knowledge about which resources are most critical to meeting future capacity needs. In many shops these resources will revolve around network bandwidth, the number and speed of server processors,

or the number, size, or density of disk volumes comprising centralized secondary storage. Other resources often identified for consideration are:

- Channels
- Tape drives
- Centralized memory in servers
- Centralized printers
- Desktop processors
- Desktop disk space
- Desktop memory

Cardinal Rule #4: Measure the current utilizations of the resources.
The resources identified in cardinal rule #3 should now be measured as to their utilizations or performance. These measurements provide two key pieces of information. The first is a utilization baseline from which future trends can be predicted and analyzed. The second is the quantity of excess capacity available for each component. For example, a critical server may be running at an average of 60% utilization during peak periods on a daily basis. These daily figures can be averaged and plotted on a weekly and monthly basis to enable trending analysis.

Resource utilizations are normally measured using several different tools. Each of these tools contributes a different component to the overall utilization matrix. One tool may provide processor and disk channel utilizations. Another may supply information on disk space utilization, while still another provides insight into how much of that space is actually being used within databases.

This last tool can be very valuable. Databases are often preallocated by database administrators to a size that they feel will support growth over a reasonable period of time. Knowing how full those databases actually are, and how quickly they are filling up, provides a more accurate picture of disk space utilization. In environments where machines are used as database servers, this information is often known only to the database administrators. In these cases it is important to establish an open dialog between capacity planners and database administrators and to obtain access to a tool that provides this crucial information.

Cardinal Rule #5: Compare current utilizations to maximum capacities.
The intent here is to determine how much excess capacity is available for selected components. The utilization or performance of each component measured should be compared to the maximum usable capacity. Note that the maximum usable is almost always less than the maximum possible. The maximum usable server capacity, for example, is usually only 80–90%. Similar limitations apply for network bandwidth and cache storage hit ratios. By extrapolating the utilization trending reports and comparing them to the maximum usable capacity, the process owner should now be able to estimate at what point in time a given resource is likely to exhaust its excess capacity. These comparisons should be updated at least monthly, and if possible, weekly. As utilizations change, the point in time at which existing capacity will be exceeded should similarly be adjusted

Cardinal Rule #6: Collect meaningful workload forecasts from representative users.
This is one of the most critical cardinal rules in the entire capacity planning process, and it is the one over which you have the least control. Developers are usually asked to help users complete IT workload forecasts. As in many instances of this type, the output is only as good as the input coming in. Working with developers and some selected pilot users in designing a simple yet effective worksheet can go a long way to enforcing this cardinal rule. User workload forecast worksheets should be customized and used as much as possible to meet the unique requirements of your particular environment.

Cardinal Rule #7: Transform forecasts into resource requirements.
After the workload forecasts are collected, the projected changes need to be transformed into resource requirements. Sophisticated measurement tools or a senior analyst's expertise can help in changing projected transaction loads, for example, into increased capacity of server processors. The worksheets also allow you to project the estimated time frames during which workload increases will occur. For major application workloads, it is wise to utilize the performance centers that key suppliers of the servers, database software, and enterprise applications now offer. These suppliers are normally more than willing to lend their capacity planning expertise to current or prospective clients, but surprisingly few shops take advantage of these low or no costs services.

Cardinal Rule #8: Map requirements onto existing utilizations.
The projected resource requirements derived from the workload projections of the users in cardinal rule #7 are now mapped onto the charts of excess utilization from cardinal rule #5. This mapping will show the quantity of new capacity that will be needed by each component to meet expected demand. For example, suppose that a company's financial systems are running on two production servers that are each running at 50% utilization. If we presume that maximum usable capacity is 80% then our excess capacity would be 30%. If the projected new resource requirements are for only an additional 25% of capacity, then no new upgrades would needed. But if another 50% of capacity is needed, then either existing servers need to be upgraded or a new server brought in.

Cardinal Rule #9: Predict when the shop will be out of capacity.
The mapping of the quantity of additional capacity needed to meet projected workload demands will also pinpoint the time frame during which these upgraded resources will be required. Using the same example as described in cardinal rule #8, suppose additional 10% of capacity is needed every quarter for the foreseeable future. By the end of the third quarter all available excess capacity will have been exhausted. At this point in time new upgrades or additional servers need to be brought in. This is very advantageous to know since most upgrades requires months of pre-planning to order, configure and install.

Cardinal Rule #10: Update forecasts and utilizations.
The process of capacity planning is not a one-shot event but rather an ongoing activity. Its maximum benefit is derived from continually updating the plan and keeping it current. The plan should be updated at least once per year. Shops that use this methodology best update their plans every quarter. Note that the production acceptance process also uses a form of capacity planning when determining resource requirements for new applications.

These ten cardinal rules will go a long way to formalizing the capacity planning process for an infrastructure, and should help ensure that adequate computer and network resource capacity is always available. This can result in capacity planning becoming one of the first infrastructure processes to be implemented, rather than the last.