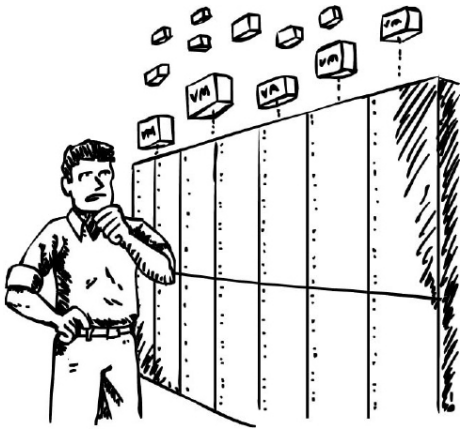
CAPACITY MANAGEMENT *in the* MODERN DATA CENTER

TURBONOMIC WHITE PAPER

turbonomic

EXECUTIVE SUMMARY



Capacity management as an operational discipline has existed since the advent of server-based computing, dating back to the age of the mainframe. Commercial tools to support this discipline have existed for more than 30 years with each successive generation of server platform creating its own unique requirements. As the data center evolved from mainframes to midrange computing and from client server to virtualized, the demand for capacity management tools has evolved in tandem.

The introduction of virtualization in particular introduced the **Intelligent Workload Management** (IWM) problem where capacity management was no longer a sufficient solution to assuring application performance. In particular, traditional capacity management solutions suffer from the following fundamental shortcomings in the modern data center:

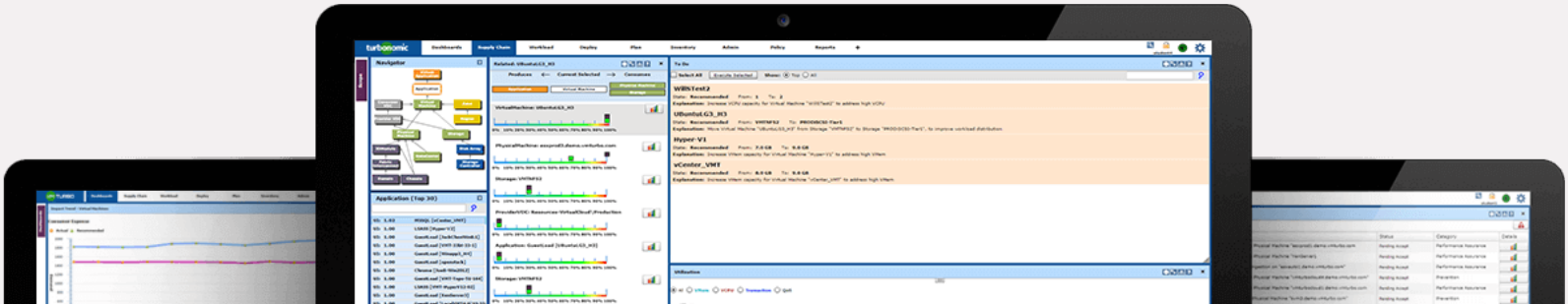
Traditional platforms are inadequate for real-time operations

- Central exponential analytics force them to execute periodically in batch, as such they cannot adapt to continuously fluctuating application demands
- They rely exclusively on historical data and therefore cannot deal with unpredictable application demand patterns
- Recommendations they produce are often obsolete before they can be executed
- They rely on historical data, which is inappropriate for cloud-native application workloads

Traditional platforms focus on infrastructure while ignoring application performance

- They use inappropriate analytics algorithms that focus exclusively on infrastructure utilization and do not consider application performance
- They do not have the semantics to associate workload demand with infrastructure supply to assure application performance

Guaranteeing application performance in the modern data center requires a real-time control system that solves the **Intelligent Workload Management** problem. The design of the software-defined data center with the advent of virtualization **does not include this system**.



DEFINING CAPACITY MANAGEMENT

Gartner defines **Capacity Management Tools** as follows:

> *IT infrastructure-capacity-management tools can generate infrastructure-capacity-related reports, are able to perform historical data analysis and capacity-related analytics, and have IT and business scenario-planning abilities.*
>
> *These tools are distinguished by the breadth of their capabilities in terms of their integration with data from a variety of domain-specific tools (e.g., real-time performance-monitoring tools); by their ability to provide forecasts, advice and automation for a wide variety of different types of infrastructure components; by the depth of their analysis of the underlying factors contributing to the performance of the infrastructure; and their support for what-if scenarios and their integration with online analytical processing (OLAP) business-reporting tools.*
>
> Gartner Market Guide to Capacity Management Tools

The goal of capacity management tools is to answer questions such as:

- Do I have sufficient infrastructure capacity to support my current and future workloads? If not, when must I acquire additional capacity and of what type?
- What is the impact of changing the capacity or configuration of my infrastructure?
- What is the best way for me to migrate workload between environments?

vmware
solarwinds
DELL
IBM
bmc
cirba
ca technologies
TeamQuest
NetIQ

TRADITIONAL
CAPACITY MANAGEMENT TOOLS

A Brief History of Capacity Management

Capacity management tools were originally developed to support IBM mainframes. The primary driver was the fact that mainframe hardware was excessively expensive and as a result, a great deal of effort went into determining precisely how much hardware was required.

With the advent of mid-range servers, capacity management was de-emphasized. Although it was still important to determine how much hardware should be purchased, two trends made this exercise less critical. First, hardware became less expensive, and thus precision in capacity purchases became less important. Second, while mainframes ran many applications on a single server, mid-range systems tended to run a single application per server. This simplified the planning process and reduced the need for sophisticated tools.

Next, the transition away from midrange UNIX systems to client-server systems based on the Wintel platform changed the dynamics yet again. The price of servers began to plunge, and most servers remained single application. This continued to erode the value of capacity management tools.

With the advent of virtualization, the capacity management problem started to look more like the mainframe problem again. Thanks to virtualization, it was once again the norm to run multiple applications on a single server. In addition, although the cost of a single server continued to decline, the number of servers had increased dramatically.

Despite this need, Gartner estimated that as of 2014 less than 5% of enterprises were using IT infrastructure capacity management tools. They further estimated that through 2018 only 30% of enterprises will adopt these tools – a compound annual growth rate of only 5%. Given that the category is mature, the obvious question is, “why is adoption so low?”. Further, given such low penetration, why is adoption growing so slowly?

Capacity Management versus Workload Management

With the advent of virtualization, although multiple applications were executing concurrently on a single server, they were no longer executing in a single operating system instance. The hypervisor handles resource sharing instead of the operating system. The scope of the problem expanded from compute resources to include storage and network resources.

In addition, the intelligent workload management capabilities needed to assure application performance were left out of the hypervisor layer. While capacity management was still a useful planning exercise, it was not a sufficient complement to the hypervisor for performance assurance.

Guaranteeing Application Performance in the Modern Data Center

The primary goal of any operations team is to assure the performance of their applications while maximizing the utilization of the required infrastructure resources. Every activity that is undertaken in the operation of a modern data center including provisioning, monitoring, capacity management, and automation supports this primary objective.

While some claim that capacity management supplemented by automation can address the intelligent workload management problem, this is not correct. It is true that capacity management is a useful exercise in determining future capacity needs and planning migrations, but adding automation as an afterthought does not provide an appropriate platform for assuring application performance. It does not fill the Intelligent Workload Management gap that was left out of the hypervisor layer. Solutions adopting this approach suffer from the following shortfalls:

1. They use inappropriate analytics algorithms that focus exclusively on infrastructure utilization and do not consider application performance
2. They rely exclusively on historical data and therefore cannot deal with applications that experience unpredictable demand patterns
3. Their brute-force analytics force them to execute their analytics in batch and automate only periodically, which prevents them from reacting to changing demand
4. They produce recommendations that are often obsolete before they can be executed
5. They rely on historical data, which is inappropriate for cloud native application workloads

More recently, some of these capacity management tools have added the ability to generate recommendations based on their analysis and in some cases the ability to action those recommendations through scripts or integration with external orchestration systems.

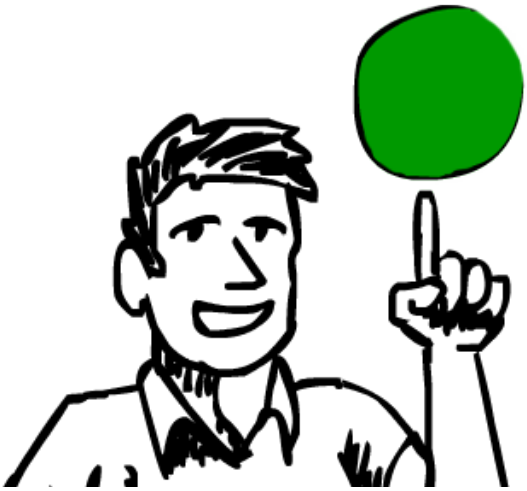
However, in all cases, the analytics used by such capacity management tools are focused on improving infrastructure utilization and not on assuring application performance. This is highly problematic because reconfiguring one's infrastructure for efficiency without accounting for performance can lead to serious application performance issues.

When it comes to VM placement, capacity management solutions rely on a bin-packing algorithm wherein utilization peaks are matched with valleys in order to optimize the density of the infrastructure in question. There are several fundamental problems with this unsophisticated approach.

Cannot Execute In Real Time

In computational theory, bin-packing algorithms are categorized as a combinatorial NP-hard problem. This means that finding the solution to the problem is very computationally intensive and as a result analytics relying on bin packing algorithms must be run periodically in batch versus continuously in real time. Therefore, the resultant automated actions produced by the analytics are executed periodically rather than continuously. This is analogous to how disk defragmentation used to occur before write optimization was built into the file system itself.

The core issue with this approach is that it fundamentally cannot assure application performance because only real-time automation can deal with fluctuating application demand by continually configuring the infrastructure supply to meet current application demand.



Cannot Handle Unpredictable Demand

Because the analytics are run periodically in batch they are based only on historical data, and therefore are only accurate if future demand closely reflects historical demand.

While this approach may be sufficient for periodic capacity management, it is entirely inappropriate for real-time application performance control. Many modern applications have unpredictable demand patterns that make historical analysis insufficient.

For example, virtual desktop workloads do not have consistent historical data. Even traditional transaction processing applications experience unpredictable demand spikes, and it is precisely these scenarios that negatively impact business processes. In order for an analytics engine to assure application performance, it must consider both historical and current real-time workload demand.

Further, because automated actions such as placement decisions are only executed periodically, and cannot account for unpredictable demand, they must rely on headroom allocations to allow sufficient spare capacity to deal with unexpected demand spikes. This headroom allocation actually reduces the efficient use of the underlying infrastructure and is not a sufficient solution to dealing with fluctuating demand. Using the headroom approach one must choose between leaving sufficient unused capacity to deal with any anticipated spike or risking performance issues. Appropriate solutions respond to fluctuating demand in real time, eliminating the difficult choice between overprovisioning and introducing performance risk.

Does Not Scale

Because the bin-packing algorithm is NP-hard, it does not scale easily as multiple dimensions are added. In fact, in the domain of infrastructure, as the algorithm is extended to consider not just compute, but storage, network, and applications, the time and resources required to execute the analytics increase exponentially. As a result, not only does the algorithm not scale, but also it cannot be converted to execute in real-time and therefore can never assure application performance. Finally, it is very difficult to extend across multiple domains – not just compute but also network, storage, and applications.

Automation Is an Afterthought

Legacy capacity management tools predate the software-defined data center and were not initially conceived with automation in mind. As a result, the execution of the analytics, the production of an action plan, and the execution of the action plan are independent phases executed in serial. Often the automation is accomplished through bolt on scripts or third party orchestrators, which dramatically complicates deployment, configuration, and maintenance of the solution. In addition, because the automation can only occur after completion of the analytics, it cannot be executed in real-time.

Unreliable Action Plans

Action plans produced by capacity management tools suffer from a fatal flaw – they can be, and often are, unusable. Because the analytics operate in batch from historical data, all of the actions that they generate are based on an assumption that when the actions are executed the environment is in the same state as it was at the time the data for the analytics was captured. As a result, if the environment has changed in any way between the time that the data was captured and the time the actions are executed those actions are invalid.

Further, because all of the actions are interdependent, a single change (such as a moved VM) can invalidate the entire action plan. This change could happen while the analytics are executing (a process that often takes hours because of the computational intensity of the algorithm) or even while the action plan itself is executing. This is further exacerbated by the fact that there is no way to determine in advance if any invalidating change has occurred before attempting to execute the action plan. As a result, any attempt to execute a produced action plan in dynamically changing infrastructure is unreliable.

Inappropriate for Cloud Native Workloads

Finally, batch capacity management based on historical analysis is completely inappropriate for cloud native workloads. Increasingly applications are being architected to scale horizontally using microservices deployed in containers. These container-based microservices are continually created and destroyed in real time based on application demand – as a result there is insufficient historical data to perform batch capacity analysis. Traditional batch capacity management is completely inappropriate for cloud native workloads, which means they face obsolescence in the near future. In fact, cloud native workloads can only be managed by a real-time control system.

CONCLUSION

As we have seen, capacity management tools are inappropriate for assuring application performance because they cannot execute in real time, cannot handle unpredictable demand, do not scale, produce fundamentally unreliable action plans, and are entirely inappropriate for cloud native workloads.

What is needed to assure application performance in the modern data center is a real-time control system that solves the intelligent workload management problem that was left out of the design of the software-defined data center with the advent of virtualization.

About Turbonomic

Turbonomic's autonomic platform is trusted by enterprises around the world to guarantee the performance of any application on any cloud or infrastructure. Turbonomic's patented decision engine dynamically analyzes application demand and automatically allocates shared resources to all applications maintaining a perpetual state of health.

Launched in 2010, Turbonomic is one of the fastest growing technology companies on the market. Leveraging Turbonomic's autonomic platform, customers can confidently accelerate their adoption of cloud, virtual, and container deployments accelerating transformation.

With Turbonomic, customers drive real-time performance, guarantee a Quality of Service, build confident agility, and minimize OpEx/CapEx spend. To learn more, visit **turbonomic**.com.