Whether or not to deploy Microsoft Exchange on virtualized infrastructure has been hotly debated over the years. The mission-critical, resource-intensive mail server has left organizations weighing tradeoffs between application performance and server consolidation. However, the challenge of deploying and managing Exchange, whether virtualized or not, is not simply a matter of performance versus utilization. Rather, Exchange requires that both are simultaneously achieved: assuring performance, while maximizing efficiency in the underlying infrastructure.

Exchange has been designed such that performance degradation also occurs when the underlying infrastructure (whether virtualized or not) is over-provisioned. In other words, it’s not only resource-intensive, as is commonly known, but it is finicky as well. Workloads need to get exactly the resources they need, when they need them.

Truly assuring application performance for Microsoft Exchange requires virtualization infrastructure to be at its best. It needs to operate in a Desired State—a state in which application performance is assured, while infrastructure utilization is maximized. Any virtualized application operating in a Desired State will exhibit performance improvements. Exchange’s particular sensitivity to the underlying infrastructure, however, makes maintaining a Desired State all the more critical to assuring its performance.

This ebook discusses the particular challenges that administrators face when managing Exchange in a virtualized environment, the traditional approaches that exacerbate these challenges, and the key requirements for achieving and maintaining a Desired State in virtual and cloud environments.
Microsoft Exchange plays a crucial role in business operations across the globe. When email is down, business is down. It’s an application used daily, if not hourly, by every employee at a company. Add Bring Your Own Device (BYOD) practices to the mix and usage extends to all hours of the day and night, more so for companies with global operations.

As of 2011, the mail server had a worldwide installed base of 360 million mailboxes. (On-premises Exchange accounted for 76% of that installed base.)¹ In a recent survey, Turbonomic found that the mail server is the most commonly cited business-critical application: 68% of respondents consider it business critical, followed by file servers at 62% and website at 52%.

It’s no wonder then, that deploying the mail server on virtualized infrastructure has been hotly debated over the years. Evaluating perceived tradeoffs between performance and server consolidation has been an ongoing dilemma. Nevertheless, VMware customer surveys indicate that business-critical applications, particularly Microsoft Exchange are increasingly being virtualized. It highlights the draw of cutting infrastructure costs and easier upgrades.

---

**Figure 1:** The percentage of business-critical applications that VMware customers have virtualized. For Microsoft Exchange, the jump in 2013 can be attributed to hypervisor developments that scaled back restrictions on how much CPU and memory could be allocated to support a single application. Source: *Why Virtualizing Your Business-Critical Applications Can Lead to a Promotion*, VMware 2015.²
The Draw of Virtualized Microsoft Exchange

Virtualization offers capital and operational benefits that become significant at scale. The most touted among them: server consolidation. Other benefits include easier application upgrades and installs, easier disaster recovery, reduced IT expenses, and so on.

Server Consolidation

Server consolidation was particularly relevant for Microsoft Exchange, which traditionally had multiple pieces or “roles,” each of which was assigned a dedicated server per Microsoft’s recommendations. The roles have since been reduced from several in Exchange 2007/2010 to just two in Exchange 2013—mailbox and CAS—as server hardware has become more powerful. Regardless of the increasingly simplified architecture, the mail server remains a resource-hungry application, which promises to complicate any shared pool of resources.

Application Uptime

Virtualization allows administrators to move applications and provision resources as needed. Moving running applications from one server to another avoids downtime in the case of hardware failure. Meanwhile, adding memory or CPU to stressed applications avoids performance degradation.

Reduced Costs

With multiple applications running on a single server, fewer servers are required, significantly reducing infrastructure expenses. This includes the capital costs of the servers and licenses, as well as the operational expense to power and cool them. Virtualizing the infrastructure has also meant that installs are faster, and that the repetitive and mundane tasks involved in managing physical servers are now reduced, further reducing operational expenses.

For all the touted benefits of virtualizing mission-critical applications, performance is not among them. To the contrary, negative impacts on performance are accepted, albeit begrudgingly. The perception that virtualized applications will see negative impacts on performance are the direct result of:

- **The Break-Fix Mentality:** Reliance on performance degradation to trigger alerts and initiate troubleshooting and root cause analysis (the irony cannot be overstated).

- **Guessing Games:** Approximations when configuring and sizing the underlying infrastructure—particularly detrimental for Microsoft Exchange, which is sensitive to over-provisioning, in addition to under-provisioning.

It is the approach, not virtualization itself, which makes at least some performance degradation inevitable. Given this industry-wide reality, it's no wonder that proposed “best practices” for Exchange recommend against virtualizing it or at least maintaining some artificial boundaries to make management easier. The catch is that by not delivering virtualization at its best—giving application workload demands the exact resources they require—performance degradation will occur in Microsoft Exchange, whether virtualized or not.
Microsoft Says: ‘Don’t Virtualize’

The “preferred architecture” promoted by the Exchange Team at Microsoft recommends against virtualization. Ross Smith IV, Principal Program Manager at Microsoft, writes in his post “The Preferred Architecture” (PA):

In the PA, all servers are physical, multi-role servers. Physical hardware is deployed rather than virtualized hardware for two reasons:

1. The servers are scaled to utilize eighty percent of resources during the worst-failure mode.

2. Virtualization adds an additional layer of management and complexity, which introduces additional recovery modes that do not add value, as Exchange provides equivalent functionality out of the box.

By deploying multi-role servers, the architecture is simplified as all servers have the same hardware, installation process, and configuration options. Consistency across servers also simplifies administration.

Smith emphasizes that the cornerstone concept for the architecture is “simplicity.” Given the complexity of virtualization and the critical role of Exchange, the team's stance against it is understandable. Microsoft Exchange is a resource-intensive, highly dynamic application that adds a significant amount of risk to shared environments. The Exchange team has determined that any efficiency benefits virtualization has to offer are simply not worth the risk to performance.

But, Says Microsoft: ‘If You Must Virtualize, Take Not-Too-Wild Guesses’

A common challenge when deploying Exchange on physical or virtualized servers is sizing. How much CPU? How much memory? As it turns out, assuring application performance is not as easy as over-provisioning, especially when it comes to Exchange. Whether an administrator chooses to follow Microsoft’s guidelines or a 3rd-party’s, VMs are allocated resources based on a mix of “hard” calculations and estimates.

The Exchange team recommends scaling out commodity-class 2U servers with 2 processor sockets for deployment. Why? Because those are the types of servers they use to deploy Exchange Online. The preference for scaling out comes down to anticipating the probability of server failure and mitigating the risk: “…the more nodes in your DAG, the more nodes you can lose while a) still maintaining quorum and b) keeping the load lighter on the nodes that pick up the slack.”

If virtualizing, the Exchange team recommends sticking as close to the Preferred Architecture as much as possible and ensuring that virtual core count and memory size not exceed 24 CPU and 96 GB, respectively. Exchange has been designed with the expectation that it will operate on commodity servers. As a result, in the case of scaling up servers or VMs, or simply having oversized them, Exchange will get caught up in administrative tasks instead of completing transactions:
Many of the issues we see are in some way related to concurrency and reduced throughput due to excessive contention amongst threads. This essentially means that the server is trying to do so much work (believing that it has the capability to do so given the massive amount of hardware available to it) that it is running into architectural bottlenecks and actually **spending a great deal of time dealing with locks and thread scheduling instead of handling transactions associated with Exchange workloads**. Because we architect and tune the product for mid-range server hardware as described above, no tuning has been done to get the most out of this larger hardware and avoid this class of issues. (emphasis own)

Traditionally administrators, with the help of sizing calculators, have taken a best guess approach to determining how much CPU and memory is just enough. But even the best of guesses will likely see performance impacts.

Not quite art, and not quite science. It is no wonder that CIOs are slower to virtualize the mail server versus less critical applications. As Tony Redmond puts it: ⁹

Most experienced people take the output from any general-purpose sizing tool and cast a cold eye over its recommendations to put them into context with the operational and business requirements for a deployment.

Whatever the guess, dedicating resources to specific applications or parts of applications diminishes the benefits of virtualization. With virtualization at its best, workloads get the resources they require, and only the resources they require. Exchange, finicky beast that it is, needs virtualization at its best.

**VMware Says: ‘Virtualize (But Not Too Much)’**

Even VMware, which enthusiastically pushes for virtualizing Microsoft Exchange 2013, is not prepared to push virtualization’s theoretical limits. The industry giant recommends practices that similarly diminish the value of virtualization with arbitrary boundaries: ¹⁰

While we are glad that Microsoft has evolved in some ways and the Exchange team is now more open in discussing the inherent defects in Exchange Server 2013, we cannot but notice that Jeff et al continued to push the "Combined Role" design recommendation, in spite of the fact that such design unnecessarily complicates performance troubleshooting and hinders fault domain isolation. (emphasis own)

VMware would prefer single-role design to multi-role in order to make root cause analysis easier. In other words, virtualize Exchange deployments, but don’t go too far with it or it’ll be too difficult to figure out what happened when things go wrong.

That hardly instills confidence for organizations set on delivering application performance. Remember though, these recommendations are the consequence of an industry built on break-fix approaches. Maintaining artificial boundaries in virtualized environments has also been a means of “managing” the environment.

---

⁹ I mean, seriously, how many people who deploy Exchange 2013 in any sort of large-scale manner (which is where a tool like the server calculator is most useful) will take the output from an Excel worksheet and say “Eureka – now I know what hardware to order!”? Well, maybe there are a couple... but not too many.
Assuring Microsoft Exchange Performance

Let’s First Solve for the Right Problem

Heavy, highly dynamic resource demands, like those of Exchange, are likely to wreak havoc on any environment, let alone one that’s been virtualized and relies on a shared pool of resources. The nature of a shared environment is complex: every workload demand met, impacts the supply of resources available to every other entity in the environment.

Traditional virtualization and cloud approaches focus on analyzing, configuring, and sizing infrastructure supply. They rely on alerts to know when something has gone wrong and go through great effort to figure out what’s caused the issue in the first place. Focusing on supply inevitably leads to this break-fix loop.

The real problem that must be solved is how to exactly match real-time, dynamic workload demand to the underlying infrastructure in a continuously changing environment. In other words, how to keep the environment in a Desired State: assuring application performance, while maximizing infrastructure utilization.

Fortunately, complex, process-driven problems are exactly why software exists. Any approach that truly assures performance must first understand workload demand and then consider what available infrastructure supply will best meet that demand to deliver service. And, given the scale and complexity of today’s virtual and cloud environments, it must scale. The solution must be:

- **Demand-Driven**: Workload placement decisions must be driven by real-time application workload demand.

- **Distributed**: Workload placement decisions must happen independently across the environment in order to scale.

A demand-driven, distributed software solution is the only way administrators can achieve virtualization at its best: where real-time, dynamic workload demands get exactly the resources they need, when they need them. In other words, a Desired State in which application performance is assured, while maximizing efficiency—it is also the only way to truly assure performance in virtualized Microsoft Exchange.

‘Feeding VMs’ (Supply-Driven) vs. ‘Teaching VMs to Hunt’ (Demand-Driven): When VMs (and other data center entities) can independently make resource decisions they become self-sufficient. A solution that enables this self-sufficiency will scale with any environment.
Solving the challenge of real-time intelligent workload placement requires software which can scale along with the applications, infrastructure, and with the ongoing demand. The complexity of virtual and cloud environments, in particular, demands a distributed approach. It is the only way to assure application performance, while maximizing infrastructure utilization at scale.

In 1988, Donald Ferguson and Cliristos Nikolaou at IBM T.J. Watson Research Center and Prof. Yechiam Yemini at Columbia University, suggested in the paper Microeconomic Algorithms for Load Balancing in Distributed Computer Systems that applying microeconomic principles of supply, demand, and pricing can be used to solve the complexity of load balancing across distributed systems, enabling workloads to get the resources they need to perform at scale.

The data center behaves like an economic market. Users put demands on applications, which then put demands on VMs, hosts, data stores, etc. An application will demand vCPU, vMem, and storage from a VM, while that VM demands CPU, memory, IO, CPU ready queue, etc. from the host. Everything in the data center is essentially a buyer and a seller.

Abstracting the data center as an economic market and using algorithms to apply the principles of microeconomics—demand, supply, and price—in virtual and cloud environments enables data center entities to figure out workload placement, sizing, and configuration among themselves. It is because every entity makes decisions independently that the solution is infinitely scalable.
In 2009 Turbonomic commercialized the ideas suggested by Prof. Yemini (one of Turbonomic’s founders) et al, delivering a platform that controls any type of workload on any type of infrastructure anywhere at anytime. Turbonomic’s patented algorithm abstracts the data center and all its interdependent entities into a common data model, a marketplace of buyer and sellers, mapping the end-to-end relationships throughout the entire IT stack and across the entire data center. This market abstraction is core to the Turbonomic platform, delivering a solution that prevents performance degradation at scale.

Turbonomic essentially creates an “Invisible Hand” in the data center. Data center entities independently make purely rational decisions based solely on the price of resources. As a resource becomes constrained, its price increases. For example, if memory on a host is constrained (demand is high), the VMs on that host pay more for it. If a VM gets a better price for resources from another host, it will move there. Price, which is simply a function of supply and demand, allows an entity to make a decision based on a complete understanding of the entire stack and everything in the data center.

Because entities buy only the resources they need, when they need them, application workload demand is satisfied, while infrastructure utilization is maximized—the environment achieves a Desired State. Further validating the performance benefits of the Turbonomic platform and its market abstraction, a Principled Tech study found that it improved application performance (transaction throughput) by 24% when its workload placement decisions were executed in a virtualized environment.11 These decisions can be taken manually or, as most of our customers do, automated.
In the case of assuring performance in virtualized Microsoft Exchange, Turbonomic’s market abstraction is key. Again, every entity in the data center is a buyer and a seller, and every resource has a price. Ultimately, applications sell Quality of Service (QoS) to the business in order to buy resources from the VM (or container).

How Does Turbonomic Abstract Microsoft Exchange?

With Turbonomic, Exchange servers are discovered on VMs within the environment. The software abstracts Exchange as a market entity, which buys and sells resources, as it needs them. It buys vCPU and vMem from the VM. It sells the basic commodities of heap, connections, and threads, as well as the QoS commodities: response time and transactions.

The software monitors the Client Access Server component of Exchange servers to understand workload demand. Meanwhile, Turbonomic’s market abstraction enables Exchange servers to independently and dynamically “buy” the exact resources they need. Driven by this real-time workload demand, Exchange VMs will resize memory and recommendations will be made for active connections, and the thread pool size, to assure application performance.12

Because the market abstraction is demand-driven and keeps the environment operating in a Desired State, Exchange does not get caught up in administrative tasks. It gets the resources it needs and only the resources it needs. More importantly, all workload placement decisions must be a part of delivering service in order to get the revenue required to continue buying resources from the VM. Administrative tasks that do not bring revenue to Exchange will not be executed at the cost of performance.
Managing virtualized Microsoft Exchange is a high-stakes, high-risk challenge. Over-provisioning this business-critical application does not assure performance and can instead increase the risk of degrading it. It requires virtualization at its best: workload demands get exactly the resources they need—no more, no less. In other words, assuring Microsoft Exchange performance is only possible when the virtual environment is in a Desired State.

A Desired State, in which application performance is assured, while maximizing efficiency, can only be achieved with the Turbonomic platform. Modeling the data center as an economic market accomplishes two very important things for applications, particularly Microsoft Exchange:

- The demand-driven nature of the market abstraction ensures that its intelligent workload placement continually drives application performance, while maximizing efficiency—it will not allow Exchange to get caught up in administrative tasks that do not support workload transactions.

- The market abstraction and algorithms create a distributed software solution in which entities independently determine placement, sizing, and configuration among themselves—scaling with today’s rapidly growing virtual and cloud environments.

The Turbonomic platform pushes the theoretical limits of virtualization to drive virtual and cloud environments to a Desired State. Microsoft Exchange is simply one more proof point that validates the advantage of abstracting the data center as an economic market.

A virtualized Microsoft Exchange deployment that operates in a Desired State achieves all the capital and operational benefits of virtualization. And because workload demand gets exactly the resources it needs to deliver service, there are significant performance benefits as well.
About Turbonomic

Turbonomic delivers an autonomic platform where virtual and cloud environments self-manage in real-time to assure application performance. Turbonomic's patented decision engine dynamically analyzes application demand and allocates shared resources to maintain a continuous state of application health.

Launched in 2010, Turbonomic is one of the fastest growing technology companies in the virtualization and cloud space. Turbonomic's autonomic platform is trusted by thousands of enterprises to accelerate their adoption of virtual, cloud, and container deployments for all mission critical applications.

© 2015 Turbonomic, Inc. All Rights Reserved. All trademark names are the property of their respective companies.
Footnotes


4 *The Case for Virtualizing Your Microsoft Exchange Deployment.* Aberdeen Group, June 2012.


6 http://blogs.technet.com/b/exchange/archive/2015/06/19/ask-the-perf-guy-how-big-is-too-big.aspx


12 The actions to resize memory can be automated in the software. Actions for resizing the active connections and the thread pool size will only appear as recommendations in the UI.