# How to Select a Virtualization Management Tool

**David M. Davis**

**11 x VEXPERT, VCP, VCAP, AND CCIE, ACTUALTECH MEDIA**

## CONTENTS

## SO MANY TOOL CHOICES

While VMware provides an excellent management tool for your virtual infrastructure in vCenter Server, most VMware admins soon realize that they need to expand their toolset to include resource monitoring, performance analytics, and capacity optimization capabilities such as those found in VMware's vRealize Operations offering.

Let's assume you've decided to consider something more than vCenter Server to manage your growing virtual infrastructure. With so many tools available from VMware and other vendors, how do you tell them apart? How would you narrow down the masses of tools into just a few? And how would you select a single tool to recommend that your company purchase?

As a former IT manager who has been through this process several times, I can tell you from firsthand experience that doing a thorough job is difficult and time-consuming. Comparing and testing products, then choosing one based on quantitative reasons, is challenging; not only do you have to find similarities and differences, but also where tools "are the same but different."

For example, many tools may claim to do capacity planning for virtual infrastructure, but how specifically do they do it, and which approach is better? You can't make an accurate comparison with only a high-level, superficial, check-box matrix.

## DIFFERENTIATING VIRTUALIZATION MANAGEMENT TOOLS

On the surface, all virtualization management tools might look alike. You can walk around the expo at your local VMware User Group Meeting (VMUG) and probably find at least five different virtualization management tools. At VMworld you'll find many, many more than that.

Over the years, I've developed a list of questions I ask which allow me to quickly tell them apart. While you're free to come up with your own questions, here's what I recommend that you ask…

### WHAT'S THE HISTORY?

One of the first questions I ask product managers and engineers (not always salespeople because they often don't know) is about the history and the architecture of the tool.

Where did the code for this tool come from? Is it the same tool I used 10 years ago? If so, has it been retooled and face-lifted? Did the tool used to be an "SNMP element manager"?

Many times, this history will connect dots for you, and you might remember this tool from the past. If nothing else, this knowledge will give you insight into who created the tool, how old it is (which will determine its architecture and whether it was designed for virtualization or physical servers), and what company it came from.

### WHAT'S THE ARCHITECTURE AND DESIGN?

For the most part, virtualization admins aren't developers and, in most cases, they don't think about a tool's architecture until they've already bought it.

However, you don't have to be a software developer to know that the architecture of a tool is what defines its complexity, scalability, and limitations.

For example, if a tool requires that you supply a database connection, you'll obviously have to pay for, manage, and monitor utilization of that database over time. Another consideration is how the tool's designed to obtain its data: Is it agentless, agent-based, or does it allow you to use both? Agentless designs are great, but there are limitations to what they can provide.

Agent-based designs can provide a lot of insight, but they also require you to deploy agents on every monitored host (unnecessarily in many cases). On the other hand, some tools give you the option to use the agentless approach until you need deeper knowledge, at which time you can install agents.

> ## You can't make an accurate comparison with only a high-level, superficial, check-box matrix.

Another example of how the architecture and design of a management solution is important is how it affects the openness of the solution. Some solutions are designed in such a way that they can be easily extended and integrated with other solutions in the data center, whereas closed solutions are designed such that they can only be enhanced by the software vendor who created the solution.

### WHAT'S THE ENTRY-LEVEL PRICE POINT AND PRICING MODEL?

At many companies (especially the small and medium-size private companies I'm most familiar with), the cost of the tool is the primary decision point.

For example, at my previous company, I had a mythical "line in the sand" at $100,000 for any hardware or software. If any single solution cost more than $100K, I would either tell the vendor forget it or break it into smaller chunks.

I knew that my company's owner would laugh at anything that expensive; the alternative would require a massive multi-year campaign to get approved. Thus, I personally preferred tools that were full-featured but priced in smaller, more affordable chunks like per-VM.

While in some cases the per-VM model might cost more than the per-socket model, with per-VM you can select just the VMs you want to monitor instead of being required to purchase a solution to monitor all servers. Your company, on the other hand, might look at the total 5-year cost for the entire virtual infrastructure; in that case, the per-socket cost might be more appealing.

In addition to deciding per-VM or per-CPU, a new consideration these days is workloads moving to public cloud. You may not know upfront how many workloads will be onsite in your data center and how many will live in the cloud. So licensing flexibility, which allows conversion from on-site workloads to public cloud workloads, becomes critical.

> In many cases, it's worth it to pay a little more to get a lot more for a tool that's easier to use, more full featured, or more extensible.

But the most important point about cost is to not make everything about upfront expenditures. In other words, don't rule out the really excellent solutions just because they cost more. In many cases, it's worth it to pay a little more to get a lot more for a tool that's easier to use, more full featured, or more extensible.

Also, it's better to buy the right tool the first time instead of growing out of it and having to "pay double" by buying another tool when the first tool doesn't work as expected.

## WHAT DOES THE TOOL DO FOR YOU?

Tool vendors should be able to quickly give you 3-5 bullet points covering what the tool does and how it helps you. When you ask this question, you shouldn't receive a 10-minute speech. The UVP (Unique Value Proposition) is something that any company should able to quickly answer.

Tools in the virtualization management category typically provide features around:

- **Comprehensive monitoring and alerting.** This means monitoring the most critical resources of the virtual infrastructure—compute, memory, storage, and networking—and alerts on the status of any of those resources. This could mean an alert that a vSphere host is experiencing an "all paths down" storage event or that a vSphere host is unreachable on the network. Admin notifications should be available in a variety of ways.

- **Change tracking.** It's important to be able to track infrastructure changes and quickly identify and roll back those configuration changes if they're the root cause of a problem.

- **Performance analysis.** Analyzing resource utilization across data centers, clusters, hosts, VMs, data stores, and networks allows admins to quickly identify (and perhaps even prevent) performance bottlenecks that cause application slowness or outage.

- **Capacity optimization and planning.** Analyzing long-term capacity utilization can help you determine when additional resources will be needed or where unused resources are sitting idle.

- **Performance and capacity alerting.** Similar to general monitoring and alerting, performance and capacity alerting can inform administrators, for example, that a host is hitting higher than 80% CPU utilization, or that a host is paging to disk because it's out of physical memory.

- **Performance and capacity troubleshooting.** This is important in helping you identify the root cause of performance issues. For example, you'd want to know that storage I/O contention on a host is the root cause of application slowness on a specific VM.

- **Log Analytics.** This involves using log data from a variety of sources to quickly identify the root cause of a problem.

The best tools can do even more than that. For example:

- **What-if scenario analysis.** Wouldn't it be great to get answers to questions like this: "If I doubled the memory on all hosts in a cluster, how many more VMs would I be able to add? And once I did that, where would the next bottleneck be?"

- **Application analysis.** Integrating metrics from applications such as database I/O latency or transactions per second with traditional performance metrics can be a tremendous troubleshooting help.

- **Workload balancing.** This ensures that VMs (and their critical applications) get the resources that they need to perform well both within a cluster—with vSphere's distributed resource scheduler (DRS), for instance—as well as across multiple clusters.

- **Automation.** Once you're comfortable with recommendations from the tool, you want to automatically turn those recommendations into actions. For example, if a VM needs more virtual memory, it could be dynamically added; or if a change caused a performance issue, the tool could reverse the change.

- **Relationship mapping.** This is about being able to visually see the relationships between objects and resources, not just up and down the stack (from the application, through the virtual infrastructure and down to the physical hardware) but also between multiple VMs that make up complex, multi-tiered applications.

- **Cost insights.** Knowing the cost of various infrastructure components such as CPU, memory, storage, and so on, is an obvious, and critical, benefit.

## HOW DOES THE TOOL SHOW MEASURABLE VALUE?

Once you know where a tool came from, how it collects its data, and where it's stored, you want to know how that tool is really going to help you. A tool may do performance monitoring, but if that just means it offers some pretty speedometer dials with CPU and RAM, it won't be very helpful in the big-picture view of virtualization management.

In other words, make sure that the value the tool provides is quantitative. A tool that reduces the time it takes to troubleshoot problems on average by 50% is far better than a tool that just improves efficiency.

Another question to ask around measurable value: "How does your tool intelligently identify problems, and then help to prevent them or quickly solve them?" For example, vCenter Server offers simple thresholds that can be configured (e.g., when a host has > 80% CPU utilization) to point out potential trouble in the virtual infrastructure.

By comparison, smart management tools use more than just thresholds. Smart tools should monitor your typical utilization over time, learn the "normal" for your environment (think "baseline"), then detect anomalies. At that point, it should either automatically remediate the issue or help you intelligently and efficiently take action to solve that problem. You can move on to making the rest of your infrastructure more efficient.

## WHAT TYPE OF INTEGRATION AND EXTENSIBILITY IS INCLUDED?

Many tools only see the virtual infrastructure. While that is, of course, important, the reality is that there's more to the data center—much more.

Tools should be able to monitor your physical servers, storage, networking, applications, public clouds, or just about anything you want them to.

This extensibility gives you great flexibility as a virtualization admin. Ask yourself: can the tool monitor Amazon Web Services (AWS) instances, Oracle databases, vSAN storage, Docker containers, Windows OS, and SAN or NAS arrays? Even if they can, though, just showing you data from those sources on the monitoring interface in a single window (providing the proverbial single-pane-of-glass) isn't enough.

> If you can't deploy the application and realize its value in under 10 minutes, the tool is too complex.

Why not use the data collected from these other sources, bring it into your monitoring database, and be able to correlate it with virtual infrastructure statistics? You're investing in the smartest performance and capacity analysis tool you can find, right? Why not use that investment to analyze other data center resources?

## WHAT'S THE LEARNING CURVE?

All too often, ease of installation and the usability of a tool is overlooked. If you can't deploy the application and realize its value in under 10 minutes, the tool is too complex. Another factor in usability is the user interface. Virtualization admins deserve beautiful, easy-to-use, fast, intuitive, and even fun-to-use tools. *Don't settle for less.*

## WHAT'S THE FUTURE OF THE TOOL?

It's not only about what a tool can do for you today; you want to know how well it will serve you in the future. For instance, what are some of the most requested features? What's planned for the next release? What are the major features you foresee adding in the next 1-2 years? What has the release/new features cadence been?

Virtualization management tool developers should be candid with their end users and be able to answer these types of questions. If the vendor can't offer you a solid roadmap for their tool, look elsewhere. You want a tool that will be around and continue to offer better value for you and your company.

## WHAT'S THE ONE THING THAT MAKES IT UNIQUE?

Every tool should have something that makes it unique. In the case of the vRealize Suite, which includes vRealize Operations and vRealize Log Insight, the fact that it's from VMware means I know it's designed to work with what I already have. For other tools, that unique aspect might be that it's a legacy element manager tool with a pretty face or that it's strong on virtual network analysis in VDI settings.

## SELECTING THE RIGHT TOOL FOR YOU

Using this list of questions, you should be able to narrow the field of available tools down to just a few. From there, you should put those tools to the test in your lab environment.

> If the vendor can't offer you a solid roadmap for their tool, look elsewhere. You want a tool that will be around and continue to offer better value for you and your company.

Hands-on experience should be mandatory before making a choice. You can get an idea of the tool's usability in a quick demo or video; but to really see how usable a tool is, you'll need to install it and spend some time clicking around.

This is also true for showing measurable value. You should be able to do a short proof-of-concept with a tool, using it to monitor (in a read-only mode) your production virtual infrastructure. If it doesn't show you real value in a reasonable period of time, move on to another tool.

## Appendix: Worksheet for Choosing a Virtualization Management Solution

See **Page 6** for a list of questions to ask vendors when shopping for new tools. Note that this isn't called a "checklist" for good reason: checkbox-style questions don't go into enough detail about "how" and "why." Instead, note the answers to these more complex questions on a worksheet with multiple bullets for the answer to each question.

## About the Author

**David Davis** has more than 20 years of experience in enterprise technology. He's been an IT manager, responsible for implementing and administering VMware solutions. He's also authored hundreds of papers, eBooks, and video training courses, that mostly focus on virtualization and cloud computing. He's an 11-time vExpert, VCP, VCAP, & CCIE (#9369). Today, he is a partner at ActualTech Media, where he creates content around enterprise technology, and moderates online events. His blog can be found at ActualTech.io and his video training is available at Pluralsight. You can find him on Twitter at @DavidMDavis.

# Worksheet for Choosing a Virtualization Management Solution

|  | VENDOR #1 | VENDOR #2 |
|---|---|---|
| History |  |  |
| Architecture |  |  |
| Entry-level price |  |  |
| Pricing model |  |  |
| Total cost of ownership (TCO) |  |  |
| What the tool provides/benefits |  |  |
| How they show measurable value |  |  |
| Integration/extensibility |  |  |
| Usability/learning curve |  |  |
| Future direction |  |  |
| What makes the tool unique |  |  |
| Eval/PoC status |  |  |
| **Summary/conclusion** |  |  |