

Realtime
publishers

VDI or Desktop Virtualization:
What's Right for You?
The Essentials Series

A Look at Traditional VDI's Five Big Failures

sponsored by

WANOVA[®]

Greg Shields

Introduction to Realtime Publishers

by **Don Jones, Series Editor**

For several years now, Realtime has produced dozens and dozens of high-quality books that just happen to be delivered in electronic format—at no cost to you, the reader. We've made this unique publishing model work through the generous support and cooperation of our sponsors, who agree to bear each book's production expenses for the benefit of our readers.

Although we've always offered our publications to you for free, don't think for a moment that quality is anything less than our top priority. My job is to make sure that our books are as good as—and in most cases better than—any printed book that would cost you \$40 or more. Our electronic publishing model offers several advantages over printed books: You receive chapters literally as fast as our authors produce them (hence the “realtime” aspect of our model), and we can update chapters to reflect the latest changes in technology.

I want to point out that our books are by no means paid advertisements or white papers. We're an independent publishing company, and an important aspect of my job is to make sure that our authors are free to voice their expertise and opinions without reservation or restriction. We maintain complete editorial control of our publications, and I'm proud that we've produced so many quality books over the past years.

I want to extend an invitation to visit us at <http://nexus.realtimepublishers.com>, especially if you've received this publication from a friend or colleague. We have a wide variety of additional books on a range of topics, and you're sure to find something that's of interest to you—and it won't cost you a thing. We hope you'll continue to come to Realtime for your educational needs far into the future.

Until then, enjoy.

Don Jones

Introduction to Realtime Publishers..... i

A Look at Traditional VDI’s Five Big Failures..... 1

 Square Peg. Round Hole..... 1

 Traditional VDI’s Five Big Failures 3

 Failure #1: Latent Network Connectivity 3

 Failure #2: Heavy Applications 3

 Failure #3: Video Conferencing, Voice, and Softphones 3

 Failure #4: Highly-Dynamic Application Sets 4

 Failure #5: Offline Use..... 4

Seeking a Middle Ground 4

Copyright Statement

© 2011 Realtime Publishers. All rights reserved. This site contains materials that have been created, developed, or commissioned by, and published with the permission of, Realtime Publishers (the "Materials") and this site and any such Materials are protected by international copyright and trademark laws.

THE MATERIALS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. The Materials are subject to change without notice and do not represent a commitment on the part of Realtime Publishers its web site sponsors. In no event shall Realtime Publishers or its web site sponsors be held liable for technical or editorial errors or omissions contained in the Materials, including without limitation, for any direct, indirect, incidental, special, exemplary or consequential damages whatsoever resulting from the use of any information contained in the Materials.

The Materials (including but not limited to the text, images, audio, and/or video) may not be copied, reproduced, republished, uploaded, posted, transmitted, or distributed in any way, in whole or in part, except that one copy may be downloaded for your personal, non-commercial use on a single computer. In connection with such use, you may not modify or obscure any copyright or other proprietary notice.

The Materials may contain trademarks, services marks and logos that are the property of third parties. You are not permitted to use these trademarks, services marks or logos without prior written consent of such third parties.

Realtime Publishers and the Realtime Publishers logo are registered in the US Patent & Trademark Office. All other product or service names are the property of their respective owners.

If you have any questions about these terms, or if you would like information about licensing materials from Realtime Publishers, please contact us via e-mail at info@realtimepublishers.com.

A Look at Traditional VDI's Five Big Failures

For too long, VDI has been seen as *the next era in desktop delivery*. This attribution implicitly presumes that VDI is to be the panacea for how all desktops are to be provisioned for their users.

Today's thinking is slowly shifting from that belief. First and foremost is the recognition that different use cases have very different requirements, not all of which are solved by virtualizing your desktops and throwing them into the data center. Although some businesses indeed jumped on VDI's bandwagon as early adopters, signs appear to indicate its rate of adoption is slowing. Qualifying that statement, smart businesses recognize that VDI's adoption makes sense for the special cases where it best fits—rather than being everything for everyone.

Rare is the company that has moved its entire employee base off physical computers and onto VDI. One reason for that stagnation could relate to the mismatch between the needs of IT and its users, which the previous article explored. Why do many VDI projects still remain in the pilot phase? A potential explanation is that implementing it to the wrong use cases introduces a set of big failures. What are those failures?

Square Peg. Round Hole.

The irrationally exuberant might loudly proclaim that, "We'll be moving all of our users to VDI in this project!" Yet that same individual might not realize that inside most companies exists a range of user classes. Consider the following example classes that might exist inside your environment:

- **The task worker.** These individuals typically interact with a very limited set of applications. Task workers by definition are paid to accomplish a predetermined set of tasks using a closed set of tools. The daily workflow of task workers typically sees them working with a desktop computer inside the LAN and rarely continuing that work outside. Some task worker situations also use hot desking, where workers do not occupy the same physical space during each work shift.
- **The knowledge worker.** Knowledge workers are typically not tasked so much as handed rough goals and deadlines. Their managerial emphasis leans toward independence and self-determination. As a result, knowledge workers tend to require a larger and more dynamic suite of applications to complete their work. Their greater level of professional freedom also tends to impact their desire for a more customized workspace.

- **The software developer.** Software developers are a special class of knowledge workers in that their applications tend to require low-level administrative access with high resource consumption. Their application set is also highly dynamic and may include applications that don't integrate well into automation solutions.
- **The never-in-the-office worker.** Notwithstanding this worker's class of user, the never-in-the-office worker presents a special case. Their hardware rarely if ever returns to the LAN, which often means they require administrative rights to self-service problems and add applications. Their computers lie outside the LAN's administrative boundary, so they often represent a higher threat level.
- **The sometimes-outside-the-office worker.** Much different than the "never" case is the sometimes-outside-the-office worker. Executives, salespeople, consultants, and the like, these people spend some portion of their job inside the LAN and another portion outside. This group also has special needs due to the constantly varying trust level associated with their hardware as well as their changing access requirements.
- **The outside consultant.** Not an employee, the outside consultant typically enjoys far less freedom inside the office. Outside consultants are typically brought in to accomplish specific tasks, with accesses being highly controlled to prevent exposure. These individuals are only partially trusted, and must be delegated responsibility, applications, and accesses with care.
- **The worker in the conference room.** Every person inside the office occasionally needs to leave their workspace to join others in a conference room. While there, application sets are rarely dynamic; however, maintaining the user's workspace in order to facilitate the meeting is always a desire.
- **The lab trainee.** Finally are individuals in a lab environment, whether for testing or learning. Lab environments represent special cases as well, with rapid rebuild and rapid provisioning being key goals for efficiency.

It should be plainly obvious that these different sets of individuals are best serviced through very different desktop delivery mechanisms. Although task workers, outside consultants, conference room workers, and labs might all present a perfect fit for VDI's desktop provisioning model, the absoluteness of VDI's efficacy grows hazy as one analyzes the other use cases.

Is it better to give the never-in-the-office worker a VDI desktop or allow them to work with their uncontrolled home computer? Will a VDI desktop help the sometimes-in-the-office worker while they're at their desk but significantly impede their activities once they leave? Can the advanced requirements of knowledge workers and software developers be met by VDI's everything-for-everyone delivery model?

Traditional VDI's Five Big Failures

If your answers to those questions have you concerned, know that you're on the right track. VDI's entire delivery approach is well-suited to a particular set of use cases. Individuals with LAN-speed connections, limited need for customization, relatively static application sets, low resource requirements, and a rare requirement for access outside the office all represent round pegs for VDI's round hole.

However, there are use cases where even today's VDI technology advances can still lead to failure. The following list highlights the biggest five to watch out for.

Failure #1: Latent Network Connectivity

Protocols like RDP and ICA are designed to be exceptionally bandwidth tolerant. One can run RDP or ICA through extremely narrow network connections and expect an acceptable user experience.

These protocols, however, also tend to be latency intolerant. The reason relates to the types of activities being done in the session. Clicking a mouse, entering text into a document, or moving a window works very well when latency is effectively zero. That same experience becomes very dissociative when latency grows above 200 or 300 milliseconds. Even with caching technologies, that extra fifth or third of a second between action and reaction usually makes for an unacceptable user experience.

Failure #2: Heavy Applications

VDI gains its cost efficiencies by collocating many virtual machines atop a smaller number of hosts. It rewards smart administration when more users work atop less hardware. Yet those benefits quickly unravel once "heavy" applications are provisioned to desktops.

These heavy applications—Adobe Flash et al, multimedia, CAD/CAM, imaging, and so on—consume comparatively larger quantities of system resources than do their lighter brethren. More still equals more, even considering today's advancements in resource optimization. Thus, supporting heavy applications incurs a comparatively greater cost per provisioned desktop than in the lightweight model.

Failure #3: Video Conferencing, Voice, and Softphones

Communication technologies that bring people together for live conversations require low latency if those conversations will have value. If you've ever placed a call over a satellite phone or poor VoIP connection and heard the multi-second delay between speaking and hearing, you recognize the special challenge network latency creates for these applications.

At issue is not necessarily the technologies that enable these communication mediums to work within VDI. Vendors today are making great strides in caching and other technologies that limit the impact. A much more operational issue is the dynamics of the environment itself. Although 20 virtual desktops on the same host might see their communication platforms perform flawlessly, that same experience can quickly degrade when hardware resources go into contention.

Failure #4: Highly-Dynamic Application Sets

Vendors today have also created mature technologies for rapidly deploying applications to VDI desktops. The process is little different than with physical computers. The sometimes unrecognized hurdle is that every application automation solution first requires prepackaged applications. That packaging process takes time and costs money.

The cost/benefit analysis gets worse when applications are not commonly used. Consider the one-off situation where a single (perhaps homegrown) application is needed by one or a few users. Here, the effort to package can far outweigh just simply installing it via Next, Next, Finish.

Automation grows even less effective when such applications require regular updates. Manual installations don't fit into the VDI approach due to the fact that logged out desktops are typically cleansed and returned to an available pool. In essence, you would need to continually Next, Next, Finish any needed application at every logon. Not good. Creating special cases for one-off users and their applications impacts VDI's cost model as well as its administrative optimizations.

Failure #5: Offline Use

Not every user in a business accomplishes their tasks as they sit at their desks or are attached to a high-speed network connection. Some work in client or partner sites where connectivity isn't permitted. Others travel to places where connectivity is unavailable at worst and spotty at best.

Traditional VDI's solution to offline use often involves a check-in/check-out process whereby a virtual machine is transferred from the data center to the user's laptop. The sheer size of most virtual machines makes this a lengthy process. How much time did your last 40-plus-gigabyte file transfer require? More time than patience, one supposes.

Seeking a Middle Ground

Like so much in life, it seems that a focus on either of the opposing options outlined thus far suggests the real solution lies somewhere in the middle. Whatever product filled that gap would enjoy all the centralization functionality gained by storing desktop images in the data center as well as the flexibility of processing those images on local hardware. The final article in this series discusses how one approach, hybrid desktop virtualization, helps align these two needs under a potential single solution.