NVIDIA GRID: GRAPHICS ACCELERATED VDI WITH THE VISUAL PERFORMANCE OF A WORKSTATION

White Paper | May 2014

By Alex Herrera, Senior Analyst

TABLE OF CONTENTS

LIST OF FIGURES

EXECUTIVE SUMMARY

NVIDIA's GRID technology is a break-through in remote visualization, delivering interactive, high-performance modern graphics user interfaces (GUIs), 2D, imaging, and 3D graphics … anywhere, any time, on any device. With a GRID GPU in a server, users can for the first time access rich visual content with the largest datasets, remotely-rendered with a local look and feel. They can, whether they're at the office actively working on their corporate-issue PC or workstation, or whether they're off the clock, checking in on progress via their personal smartphone or tablet.

GRID exploits unique, first-time features in NVIDIA's Kepler-class GPUs finally deliver the true "desktop experience" that VDI users have been waiting for. By the end of 2014, both Citrix and VMware will be able to deliver these capabilities in their Hypervisors, integrating GRID's virtual GPU (vGPU) technology. Running on GRID-enabled servers supplied by the most respected OEMs in the industry, vGPU technology enables true professional-caliber interactivity that scales gracefully with a multitude of CCUs (concurrent users).

With the advent of more mature VDI environments, coupled with NVIDIA's GRID vGPU technology, graphics is now ready for a shift to the cloud, to whatever degree — private, hosted, or public — meets the needs of the business.

Figure 1 GRID virtual GPU technology: server-side rendering of rich 3D content, delivered wherever, whenever.

AN INTERACTIVE VISUAL EXPERIENCE ALWAYS MATTERS

GPUs are everywhere, and for good reason. Be it PC, workstation, smartphone or tablet, graphics-rich computing has become pervasive, thanks to the real-time, high-quality experience GPU acceleration delivers. Of course, it hasn't always been this way. Back in the 70's, there was essentially one computing model: a dumb terminal on a desk, connected to a mainframe or minicomputer in the backroom. Primarily limited to text modes, the visual interface made scant use of 2D graphics, and 3D was largely non-existent.

The 80's changed that, as the emergence of the workstation and PC pushed the computation to desktops. Operating systems from Apple (MacOS) and Microsoft (Windows) introduced compelling graphical user interfaces (GUIs) that forever changed the look of personal computing, broadening appeal to the masses. At the time, however, the only means for applications to display on a PC was VGA, a register-level, hardware specification primarily oriented for text. And the only system processor to rely on was the CPU, woefully inadequate for this type of raster processing.

As the 80's gave way to the 90's, the demand for a richer visual experience gave rise to the PC's first hardware graphics accelerators. Graphics APIs (application programming interfaces) like DirectX and industry standards like OpenGL emerged to create a layer of abstraction that allowed graphics vendors like NVIDIA to innovate. And innovate is precisely what the industry did. NVIDIA's own introduction of the GPU in 1999 marked an instrumental step that gave rise to both modern professional-caliber graphic workstations and today's multi-billion dollar PC gaming industry.

A range of visually-starved applications, from gaming to CAD, both exploited those gains and spurred on the push to ever-higher performance levels. Operating systems (Microsoft Windows 7 and Apple's MacOS X) raised the bar on graphics demands as well, dialing up the visual complexity in their GUIs to rely more heavily on GPUs. One generation's advance in capabilities simply opened the door to the next. In the past decade, we’ve witnessed the transformation of the limited, fixed-function 3D rasterizer into a highly flexible, massively-parallel programmable compute engine, whose impact is being felt beyond traditional raster-based graphics, spurring appeal for GPUs in the datacenter as compute engines as well as renderers.

Today, the GPU is ubiquitous ... from the multi-billion dollar gaming industry, to workstation-caliber applications in CAD, digital media entertainment and sciences, to high-resolution graphics and video in the palm of your hand. Even mainstream productivity apps like Microsoft Office 2013 demand non-trivial GPU capabilities, like DirectX 10 compliance. And the emerging influx of rich HTML5 content on the web relies on having a capable GPU under the hood of whatever device is doing the browsing.

The reason is simple: a quality interactive visual experience always matters, and delivering on that experience always requires a GPU. That's a premise validated throughout the evolution of personal computing. And it's a premise that's just as valid for any enterprise's IT plans, whether those plans focus more on GPU-equipped clients like PCs and workstations, or whether it's a future that includes GPUs in the datacenter.

THE NEW FRONTIER: GPUS IN THE CLOUD

Today, server-centric computing is back capturing mindshare in a big way. Granted, it has a different shape than in it has in the past, as well as a variety of names and contexts — cloud computing, virtual desktop infrastructure (VDI), client consolidation infrastructure (CCI) and hosted virtual desktops (HDV). But in essence all imply the same basic idea: moving the data and heavy computation to a central resource, accessible by many rather than one.

The benefits of centralizing data in workstation applications are many, particularly as the sheer volume of visual computing data continues to explode. A few minutes of a Hollywood-caliber scene shot in 4K and captured raw can exceed 100 GBytes. And in oil and gas exploration, surveys of potential drilling fields are both expansive and detailed, resulting in single data sets that can easily push into the Terabytes. For more and more applications, copying the raw data from datacenter to datacenter can waste minutes or hours, time businesses can ill afford in an ever more competitive climate.

When it comes to visualization, it's time for some to rethink the old paradigm of copying models and data from server to client to keep pixel bandwidth local, and consider a new one: leaving big data in a central datacenter, rendering it on a server, and shipping only the rendered pixels to the client. And once that shift is made, several significant advantages emerge, benefiting various types of users engaged in a range of visually rich tasks.

MORE ACCESS FOR THE USERS THAT NEED IT

Business users that need access to that rich visual data can exploit remote visualization to tap into a single, up-to-date project database at any point in the workflow. The "power user" and "knowledge worker", may not be directly engaged in product creation, but their work still depends on being able to quickly, accurately view and mark up project material. Consider the daily needs of the product marketer, sales engineer, and support technician.

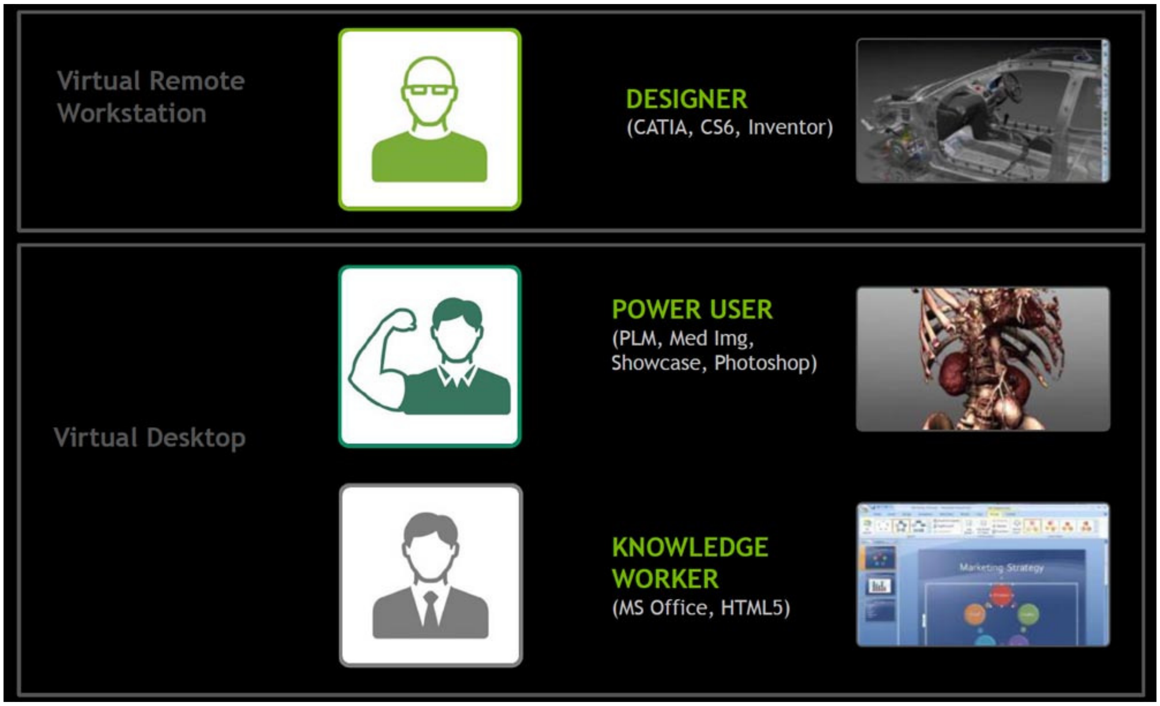


Figure 2 Project complexity and aggressive schedules demand timely accurate access to visual data by all.

HETEROGENEOUS CLIENTS

The incursion of smart digital devices into personal and working lives is undeniable. More and more, businesses need to deal with the increasing use of "BYOD" (Bring Your Own Device) in an effective and reasonable way. In the age of always-connected employees available 24/7, the line between personal and work devices has forever been blurred. With virtual machines running on the server, a client can take any number of the aforementioned forms, from thin-client, to smartphone or tablet, or to conventional deskside or notebook Mac or PC. That gives IT managers more options and more flexibility in supporting the unavoidable trend in BYOD, while at the same time presenting opportunities to cut costs.

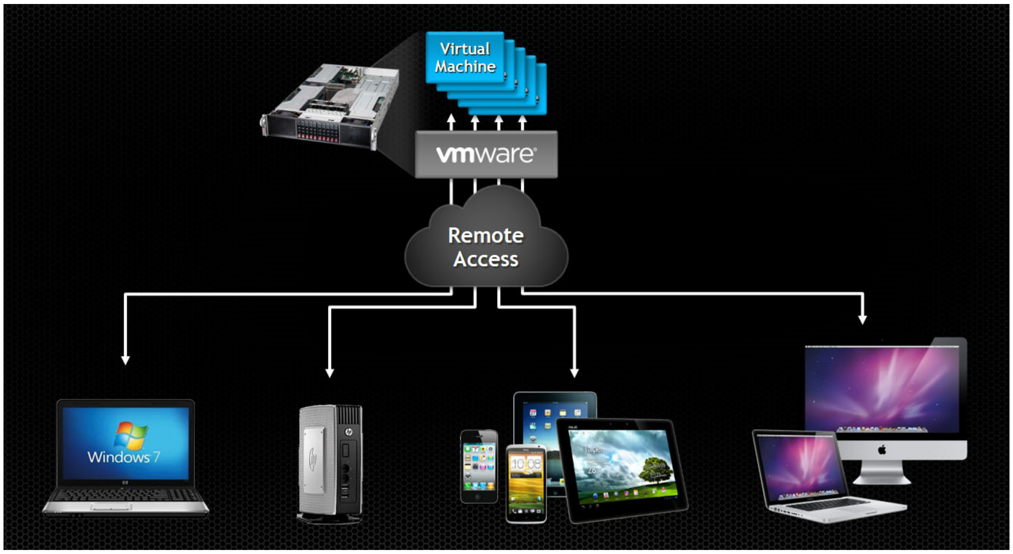


Figure 3 Like it or not, all IT environments are going to have to deal with heterogeneous clients.

24 HOUR PRODUCTIVITY: "FOLLOW THE SUN"

The beauty of a centralized model resource, especially applicable in the era of global enterprises — operations, outsourcing, and supply chains — is that none of those users have to be near company headquarters (or wherever their local enterprise's IT hub happens to be). And that leaves open the elegant promise of individuals and teams, scattered around the world, working in a daytime-driven pipeline, where one team picks up at daybreak just as another is going home at sunset.

Yes, "following the sun" is of course possible by copying source data at the end of the day to the next site that's just getting breakfast. But when the data is big, it may take the whole workday simply to copy the database. The cost of networking and storage might even prohibit creating a shadow data center for smaller regional workgroups, creating another barrier to smooth worldwide workflow. Shipping application renderings around the world, however, can be far less taxing.

BENEFITS FOR FORTUNE 500, BUT EVEN MORE SO FOR SMB

Remote visualization holds just as much value for the SMB community as it does the Fortune 500, perhaps more. Because when once one considers the move to a centralized model, with remote viewing, a whole host of other benefits emerge: security, ease-of-management, and the ability to work from anywhere.

That's good for most any installation relying heavily on professional-caliber visuals. But now think of the small business. They have no or limited IT personnel (maybe none at all), have limited budgets to spread compute-heavy clients around the facility, and they need to quickly expand and contract user environments as work comes in and out. Virtual machines do that a lot quicker than physical machines, with any IT support required easier to outsource.

Consider the small architecture firm traveling weekly to the client's premises, showing work in progress or unveiling the final design. Any on-site collaboration would either mean waiting until they got back to the office to update, or constantly lugging around a big workstation and big disk arrays. Waiting isn't good enough, and the lugging is obviously not an attractive alternative. Rather, visualizing content stored on a remote server means collaborators can make those tweaks on-site, interactively in real-time — without sacrificing visual quality.

THE HISTORICAL STUMBLING BLOCKS TO GPU-ACCELERATED VDI

With so many compelling advantages to anywhere/any-time/any-device remote visualization, graphics-rich VDI should be commonplace. But it's not. While VDI has found broad acceptance in mainstream computing, serving graphics-rich, workstation-caliber applications and data from a server to a remote client is found today only in relatively small niches. Why? Simply put, first-generation GPU-accelerated solutions either fell short in their performance or disappointed in their versatility.

The simplest and most accepted way to implement VDI to date is though a fully-abstracted software implementation of a virtual machine, running on the server: the Soft PC. With no GPU present, the CPU has to process all of the workload, including the graphics, a lot like that mainframe/terminal combination of decades ago.

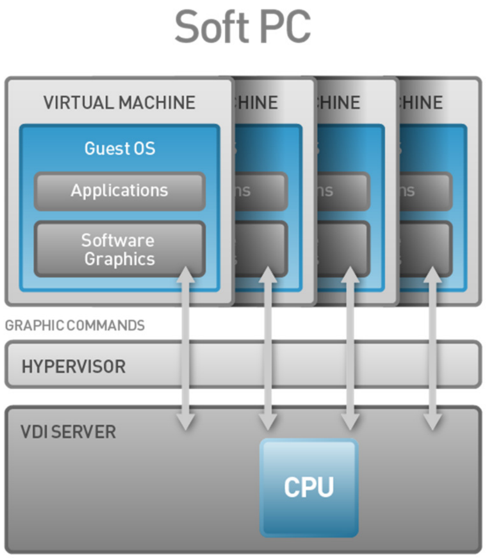
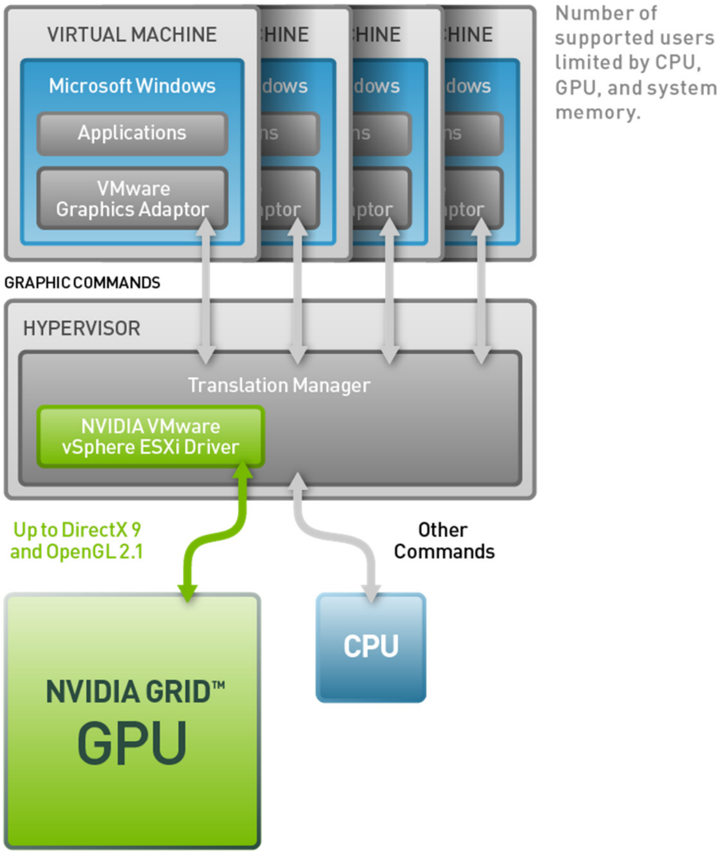


Figure 4 The Soft PC implementation for VDI: no GPU means no rich visual content

Similarly, while a Soft PC implementation can work fine for simple text-based, console-type applications, it can't deliver an interactive user experience with anything but the simplest graphics content. For the same reason GPUs grew to prominence decades ago, GPU acceleration is a necessity for effective visualization in every corner of computing. But how can that be implemented effectively in a Virtual Machine environment? To date, server-based GPU acceleration has come in two basic flavors: GPU Sharing and GPU Pass-through.

DESKTOP GPU SHARING: SCALABILITY WITHOUT THE PERFORMANCE LIMITATIONS

GPU Sharing relies on VDI software to provide a layer of abstraction that lets the client application behave as though it has its own physical, dedicated GPU, while the server's GPU (and driver) can think it's responding to one master host. The VDI hypervisor running on the server intercepts API calls and translates commands, drawing contexts, and process-specific address spaces, before passing along to the graphics driver.



(Source: NVIDIA)
Figure 5 GPU sharing: supports multiple VMs, but compromises graphics performance

GPU Sharing is a reasonable solution for many, but not an ideal solution for all. It can perform effectively with simple applications and visuals and support concurrent users (CCUs), but the extensive compute cycles spent abstracting complex 3D rendering will add latency and reduce performance. Furthermore, the reliance on API translation means 100% application compatibility is impossible to guarantee. For example, applications which leverage features from the most recent OpenGL versions may not run as expected.

GPU PASS-THROUGH: PERFORMANCE FOR DESIGNERS AND POWER USERS

So if the software overhead of GPU Sharing is a problem, then why not go ahead and actually dedicate one physical GPU in the server to each hosted client? Well, that's precisely how systems are configured in servers implementing GPU Pass-through. Unlike the rest of the physical system components, which are represented as multiple virtual instances to multiple clients by the hypervisor, the Pass-through GPU is not abstracted at all, but remains one physical device. Each hosted virtual machine gets its own dedicated GPU, eliminating the software abstraction and the performance penalty that goes with it. For example, a VDI server with 2 NVIDIA GRID K1 boards (4 GPUs per board) can support 8 simultaneous users.

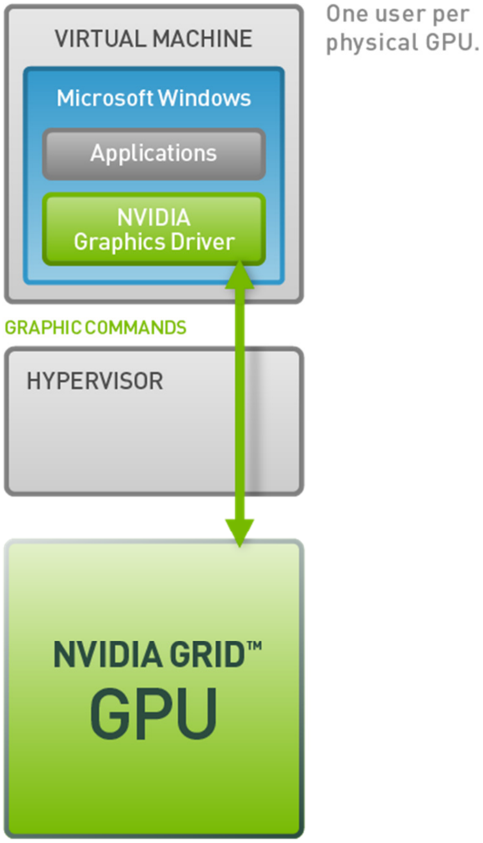


Figure 6 GPU Pass-Through: reasonable performance, but with no ability to share GPU among multiple clients

GPU Pass-through can make a lot of sense serving the power user, whose need for performance already demands a dedicated GPU. Consider the mechanical engineer who demands no-compromise visuals and for whom 100 GB file transfers are a daily occurrence. Instead of a Quadro board at his desk, he's tapping the power of a GRID GPU in the server, and reaping the benefits in data security, remote visualization, and support for BYOD hardware.

A NEW PARADIGM

GRID'S VIRTUALIZED GPU (VGPU) LETS REMOTE SERVERS DELIVER RICH VISUAL CONTENT TO MULTIPLE CONCURRENT USERS, INTERACTIVELY

For the right applications, GPU Sharing and GPU Pass-through can be an effective means to centralize and serve visually rich applications to multiple clients per sever. However, the respective tradeoffs have limited broader adoption for both approaches. The software overhead for GPU Sharing constrains performance, particularly as the CCU count rises. And GPU Pass-through's fixed 1:1 relationship (between server GPUs and clients) misses out on the big advantage of any server-centric topology: supporting multiple clients with one shared computing resource.

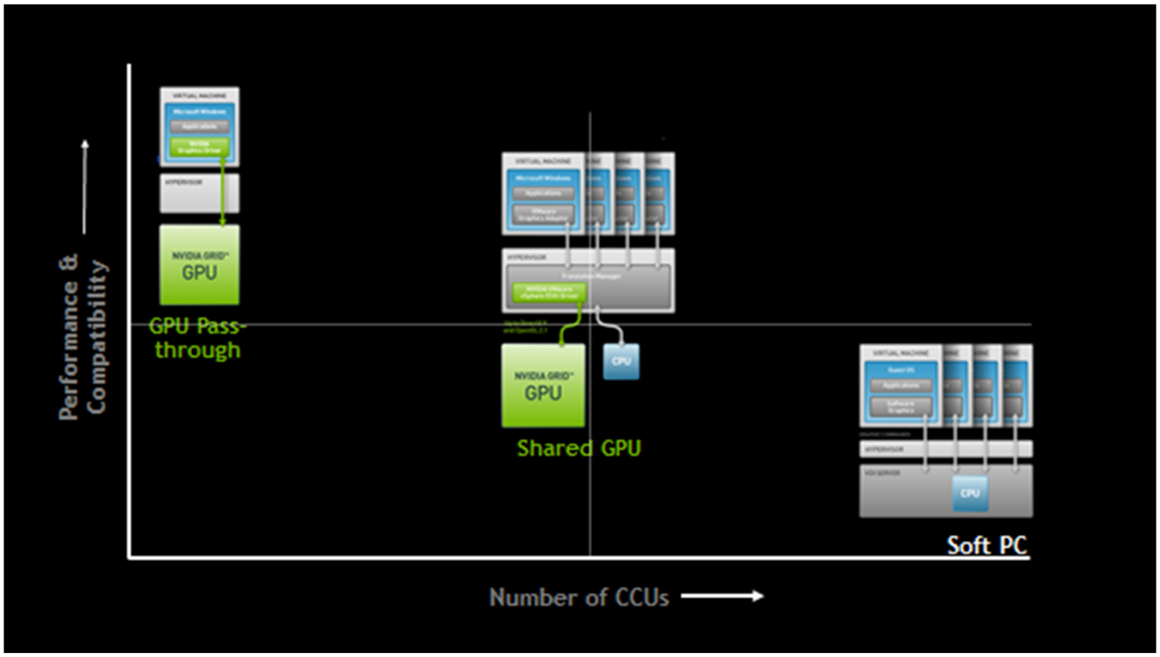


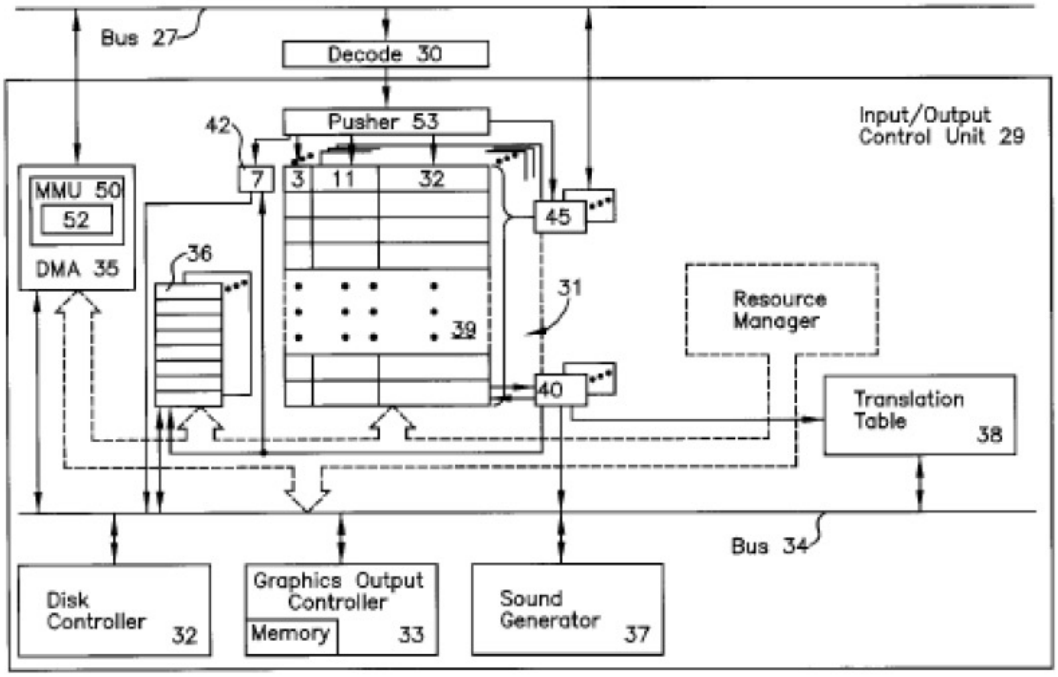
Figure 7 All of server-GPU approaches have their tradeoffs

What's ultimately needed to give server-based, GPU-accelerated, rendering broader appeal is a high-performance, low-latency approach, one that delivers 2D, 3D, imaging and 100% API support, while scaling gracefully with the number of CCUs. Enter NVIDIA's GRID technology.

LEVERAGING ONE GPU FOR MANY IN A LOW-LATENCY, HIGH-PERFORMANCE SOLUTION

Until GRID, the GPU in a VDI-hosted system was the black sheep of what was otherwise a cleanly virtualized system. Every other hardware component was virtualized in a way that let multiple VMs linked to multiple CCUs "own" a virtual representation of that component. But not the GPU. An historically non-virtualizable device, the GPU presented a special case, requiring work-arounds like GPU Sharing and Pass-through to support graphics-accelerated VDI.

But no longer, as NVIDIA has purposely engineered its most recent generation GPU, Kepler, to be the world's first truly virtualizable GPU. Unique in design, Kepler implements a Memory Management Unit (MMU) that maps and translates a host’s virtual address to the system’s physical address. Each process works in its own virtual address space, and the GRID GPU's MMU hardware keeps them physically separate, so no process can step on another’s toes. Working hand in hand with the MMU in Kepler are 256 independent input buffers another, each dedicated to a different host, thereby separating each VM's command stream into independent rendering contexts.



(Source: NVIDIA)
Figure 8 The Kepler GPU’s MMU and multi-channel input buffers enable the first virtualizable GPU

The combination of the address-space unifying MMU and a VM-specific MMU is the linchpin in delivering the worlds' first truly virtualizable GPU, ideally suited to serve multiple CCUs without the performance and latency penalty of excessive software overhead. The Citrix Hypervisor integrates NVIDIA management software, exploiting all of the benefits of GRID vGPU support.

THE GRID ECOSYSTEM

The complete GRID solution consists of three primary components: GRID GPUs (and software), provided by NVIDIA and GRID servers and supporting VDI software, provided by industry-leading partners. Supporting server OEMs include Dell, HP, IBM, Cisco, and Supermicro, while compliant virtualization software comes from premier suppliers Citrix, VMware, and Microsoft.

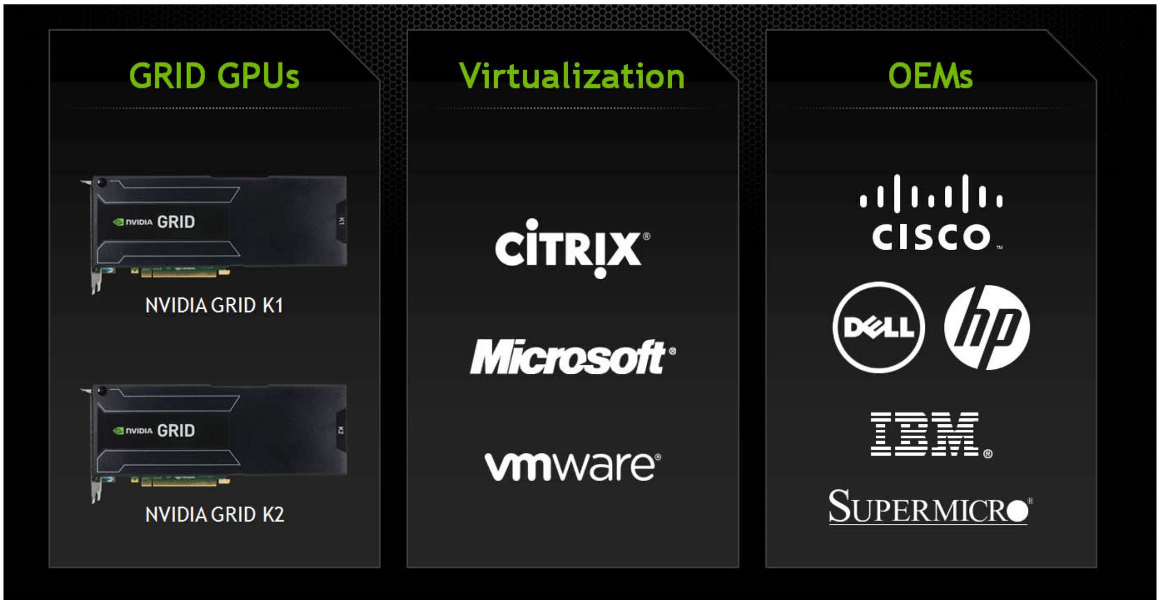


Figure 9 The GRID partner ecosystem

How's it all work? In a VDI environment with vGPU, every VM communicates through the desktop Hypervisor to its own dedicated vGPU driver, for which one instance exists per VM. Each vGPU driver sends command and control to the one physical GPU, using its own dedicated input channel. As frames are rendered, the driver returns rendered frames back to the virtual desktop, which then streams it back to the remote host.

For VMware users, GRID solutions are fully interoperable with from third-party vendors like Teradici, that capture rendered window images, encode them via PCoIP, and stream the window content to the client for subsequent decode.

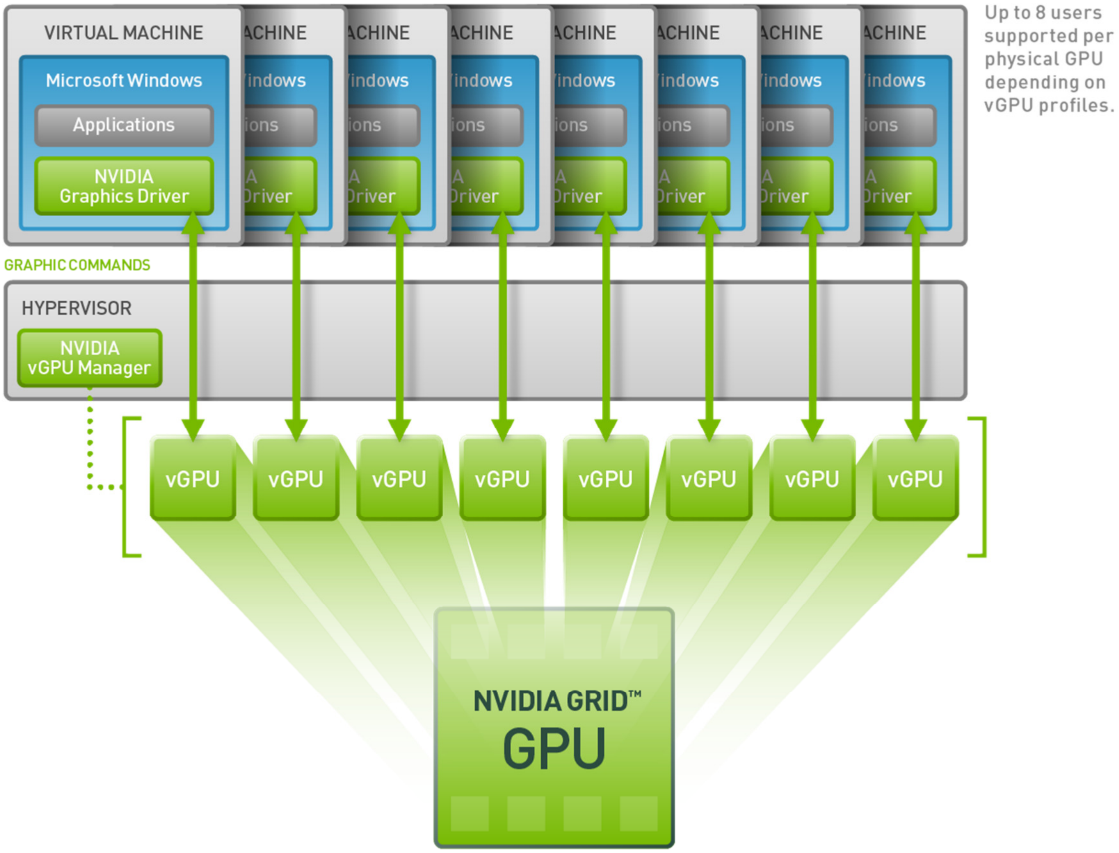


Figure 10 GRID Virtual GPU: all virtual machines share one physical GPU with hardware-based virtualization

GRID ROUNDS OUT A COMPLETE ARSENAL OF VISUALIZATION TOOLS

Today, GPUs are found in virtually every computing device we interact with, be it at work, play or leisure. Rendering graphics on the client device's local GPU is a sensible paradigm that's evolved throughout the history of personal computing — one that's withstood the test of time, and one that shows no sign of fading.

But it's no longer the only paradigm. Because with GRID, NVIDIA is making the GPU location-agnostic, promising interactive, high-performance 3D graphics delivered from a visualization server, for whoever needs it, on whatever device they prefer. GRID's unique vGPU technology streamlines processing to create the first remote visualization solution that can deliver performance, retain full application compatibility, and do so with a multitude of concurrent clients.

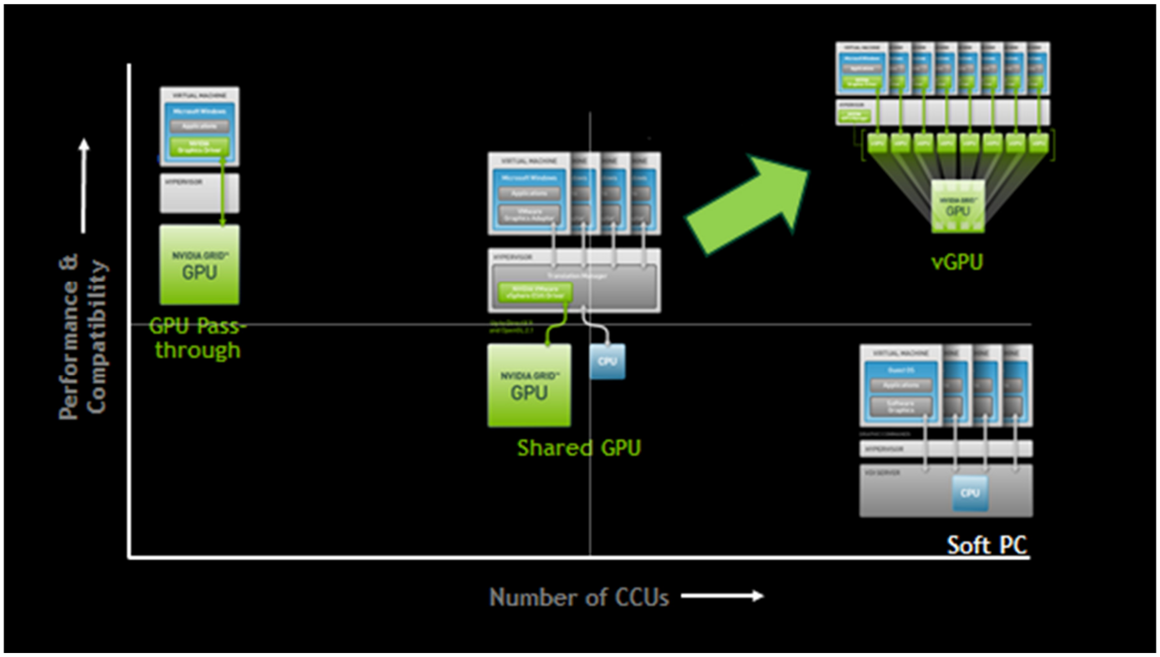


Figure 11 GRID vGPU: server-based graphics performance that can scale

Added to a technology arsenal that includes traditional, client-side Quadro GPUs, GRID now enables a continuum of visualization solutions that leverage the best of both worlds. Where should your visualization IT solution sit on that continuum? Let the needs of your business dictate the answer, not the limits of your technology.

For the latest information on GRID, including information on OEM servers certified for use with GRID, please visit www.nvidia.com/vdi. And for a complete, up-to-date listing of GRID-certified applications, check out www.nvidia.com/gridcertifications.

Copyright