

# Virtualized Simulation for Military Concept Development and Experimentation: The Cerebro Battle Lab, a Case Study

**Jan Jaap Knobbout**  
Royal Netherlands Aerospace Centre (NLR)  
Amsterdam  
[jan.jaap.knobbout@nlr.nl](mailto:jan.jaap.knobbout@nlr.nl)

## ABSTRACT

In recent years, virtualization technologies have become standard practice in many industries, mainly due to their ability to provide a flexible computing environment at a fraction of the cost and time required for traditional approaches. This paper presents a case study of a unique application of virtualization in the military domain, specifically in the area of concept development and experimentation (CD&E). The Netherlands Aerospace Centre has invested in the development of a virtualized facility, called the Cerebro Battlelab, to support research into new military concepts and the simulation of large-scale scenarios for the Netherlands Armed Forces. The authors describe the design and implementation of the battlelab that hosts various types of simulation software, including flight simulators, on virtual machines in a secure vSphere server infrastructure. This allows operators and researchers to run simulations and experiments in a controlled and repeatable manner, while providing greater flexibility and scalability compared to alternative methods, like standard desktops or keyboard-video-mouse (KVM) setups. This paper also describes the trade-off analysis that led to the decision to use a virtual desktop infrastructure (VDI), as well as the technical aspects of implementing a VDI with low-latency and graphics-intensive requirements, and the operational benefits and challenges encountered during use. This case study provides valuable insights for anyone who is interested in the virtualization of simulation systems in the military and defense industry, especially those looking to implement similar solutions in their own organizations.

## ABOUT THE AUTHORS

**Jan Jaap Knobbout** has a background in Aerospace Engineering from Delft University of Technology. He works as an R&D Engineer at the Royal Netherlands Aerospace Centre, with a focus on the interoperability of simulation systems. He has been involved with the Cerebro project for the past two years.

# Virtualized Simulation for Military Concept Development and Experimentation: The Cerebro Battlelab, a Case Study

Jan Jaap Knobbout  
Royal Netherlands Aerospace Centre (NLR)  
Amsterdam  
jan.jaap.knobbout@nlr.nl

## INTRODUCTION

In his 2022 address to defense counsellors from NATO nations, General Philippe Lavigne, NATO's Supreme Allied Commander Transformation, said they are "working with partners across NATO to embrace Multi-Domain Operations with increased focus on cyberspace and space – enabling the Military Instrument of Power to prepare, plan, orchestrate and execute synchronized activities, across all operational domains and environments, at scale and speed" (NATO, 2022). This doctrine shift towards distributed, multidomain, and joint all domain operations has not gone unnoticed by the military modelling and simulation (M&S) industry, which is expected to experience an increase in market share by USD 6.11 billion between 2021 and 2026 (Technavio, 2022). Battlefield simulations are rapidly growing larger and more complex, both in terms of the number of interconnected systems and military domains involved. That same upwards trend can therefore be found in the time and effort required to prepare for, execute, monitor, and evaluate these simulations in traditional desktop environments. As mentioned by PLEXSYS solutions strategy analyst Ethan Wilson at the 2023 IT2EC conference: "Every time you add a domain to simulation, you add an exponential growth in resource" (Wilson & Elsey, 2023). To avoid runaway costs, governments and industry alike are forced to look for solutions that will streamline this process while simultaneously supporting operators during every simulation phase. The Royal Netherlands Aerospace Centre (NLR) has therefore decided to invest in a solution that they believe will do exactly that; Cerebro: a battlelab powered by a virtualized infrastructure.

While the concept and application of virtualization technologies is nothing new, even in the military M&S domain, information regarding the development and implementation of a virtualized desktop infrastructure (VDI) that is capable of providing a smooth end-user experience for human-in-the-loop jet fighter simulations, or graphics-intensive applications in general, is severely lacking. Resources for setting up office environments to allow thousands of employees to simultaneously access their email and perform regular office duties are plentiful, but the 3D acceleration required by many simulation applications requires a different approach. The infrastructures used by the cloud gaming industry most closely resemble the type of infrastructure required for a battlelab environment, as cloud gaming providers like Nvidia's GeForce Now are able to deliver a smooth 3D-accelerated gaming experience to their customers all over the world. However, due to the highly competitive nature of this relatively new industry, they naturally do not publicly offer information or tutorials on how to set up a similar service on a smaller and localized scale.

The purpose of this paper is therefore to offer a starting point for those who are interested in implementing a similar solution in their own organization. Topics covered will include the trade-off analysis that led to the decision to virtualize the battlelab's infrastructure, the design and implementation of the hardware and software used in the battlelab, and challenges encountered and lessons learned during this entire process.

## THE CEREBRO BATTLELAB

Since its conception, NLR has been *the* strategic partner of the Royal Netherlands Air Force (RNLAf). The dedicated support provided by NLR to the Dutch Armed Forces encompasses a wide range of activities, including concept development, threat analyses, and technical assistance during large-scale international exercises. NLR frequently deploys its individual simulator facilities and supporting systems for use at defense sites with network connections to other NATO countries, enabling their integration into distributed simulation experiments. Recognizing the value and potential of consolidating its resources, NLR has taken the strategic decision to establish its own battlelab. This decision allows NLR to integrate its advanced simulation facilities and tools with more efficiency, which in turn

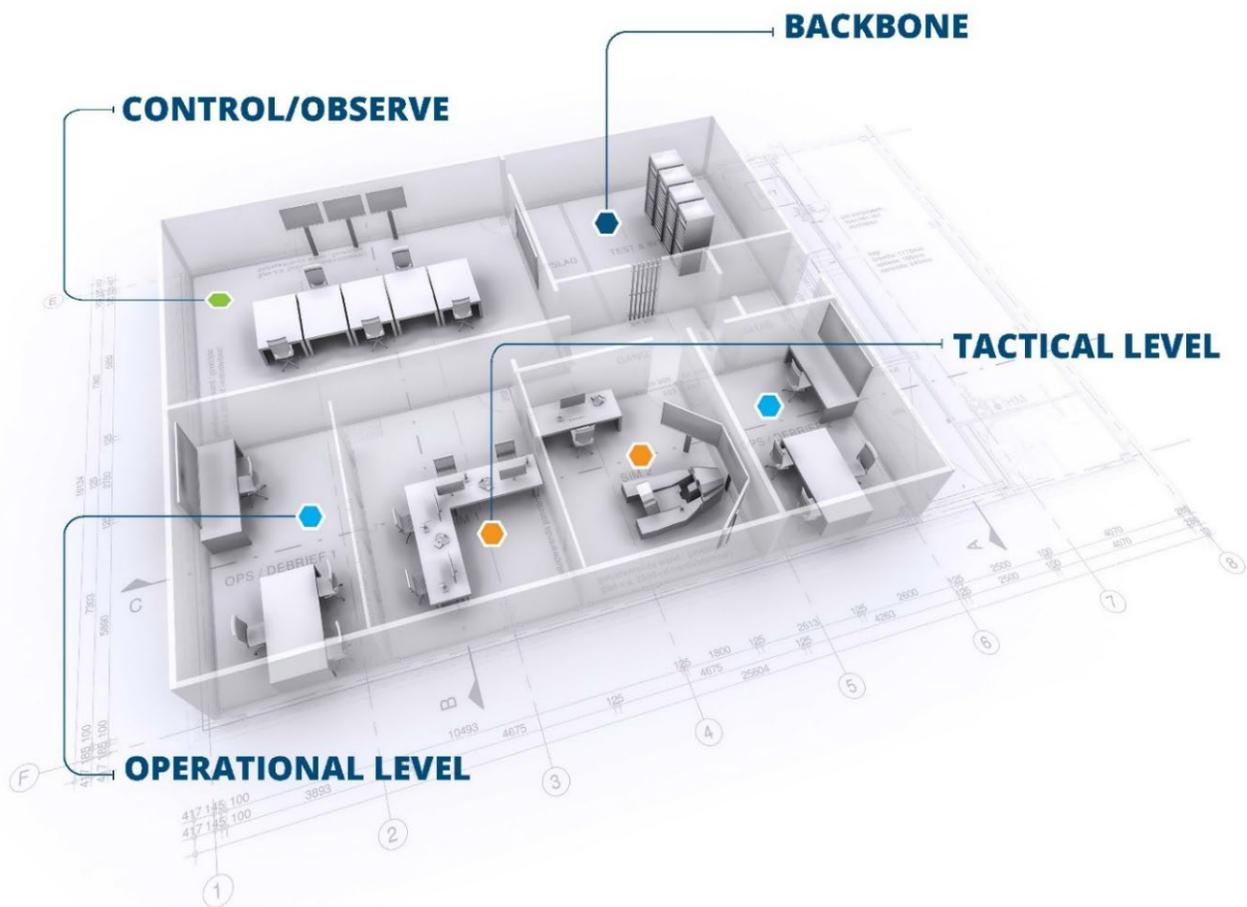
enables the development and testing of new applications in a more streamlined manner. Overall, the creation of a battlelab ensures a proactive approach to staying at the forefront of technological advancements in the dynamic landscape of air and space power.

**The battlelab: an overview**

In this paper, the definition of a battlelab is two-fold: It is a digital environment in which tactical, operational, and potentially strategic military concepts can be tested and evaluated. It is also a physical location where systems and operators converge, supported by essential network connections that enable seamless integration with off-site systems and other battlelabs.

A conceptual rendering of the battlelab was devised to clarify the main purpose of the Cerebro battlelab, which is to be an environment in which concepts for air and space power and information-driven operations can be developed, tested, and evaluated in a simulated setting (see Figure 1). It would require a dedicated space for conducting tactical human-in-the-loop simulations, a command and control (C2) area from which the operational level of the simulation can be operated, and a control/observation room for (de)briefings, analysis, and to maintain a general overview during a simulation exercise.

Most of the components required for such a battlelab were already present at NLR, including simulation facilities and software, supporting tools such as Common Operational Pictures (COP), recording and analysis tools, and the technical expertise to bring them all together. The remaining question was what the backbone of the Cerebro facility would look like.



**Figure 1 A conceptual rendering of the Cerebro Battlelab**

## **TO VDI OR NOT TO VDI**

One of the reasons that VDI came up as a possible solution in the context of the battlelab's infrastructure is the fact that a VDI offers exceptional flexibility. Operators and researchers can access their designated virtualized desktop environments from any of the available endpoints in the Cerebro area. This eliminates the need for physical desktop machines to be relocated or reconfigured based on specific simulation exercises.

A downside of a VDI in this context is the upfront cost. On larger scales and for regular office duties, the cost savings of a VDI are immediately apparent, as users could use their own devices to connect to their personal virtual machines (VMs). For a battlelab, for which remote work is generally not a possibility due to the nature of the simulations performed there, a company would still need to purchase endpoints for everyone who requires access to a VM during a simulation exercise. Also, hardware that was used for an experiment at a higher classification cannot be used for subsequent experiments at lower classifications, requiring a new set of hardware for every classification that will be used in the battlelab. Cost savings will therefore only become apparent over time, as the reduction in setup time, configuration changes, and simulation management significantly minimizes operational expenses.

### **Trade-off analysis**

Comparing the advantages and disadvantages of a VDI in this manner and comparing them to other solutions results in a decision matrix, or trade-off analysis. A similar approach was used at NLR to determine that Cerebro will be powered by a VDI solution. The requirements of the solution and the weights attributed to every requirement will differ from organization to organization. At NLR, the following solutions were considered:

- Traditional desktop solution
- A hybrid solution, with desktop endpoints and server-based applications and services running on a centralized server.
- A pre-built VDI solution
- A custom VDI solution, where system administrators at NLR would assemble the servers with individually bought components.

Metrics like application performance and input latencies were evaluated for each type of solution, along with more subjective requirements, like the ease with which the environment can be maintained or secured.

### **Automation**

Up until recently, every simulation facility at NLR had been running on traditional desktop solutions. Occasionally, a simulation host would have to run on an operating system (OS) that differs from the OS of the simulation image generator, in which case a VM with that specific OS would be run on the machine with the other OS, requiring only a single desktop machine. Applications like license servers or terrain databases would be installed on a separate desktop. However, even in these use-cases, where positions of operators will never change between scenarios and the software being used is rarely updated, booting up every machine manually and starting up the required software can be arduous.

A main requirement for Cerebro's backbone therefore pertained to configuration management. In an ideal scenario, a technical lead of a simulation exercise could press a single button, which would result in the endpoints automatically booting up in their specified environments. After successful logins by the operators, applications would subsequently start with the correct scenarios and other configurations in place. A single button would then also shut down the exercise and endpoints, after which a new exercise can be started again. A VDI solution would lend itself perfectly for this type of use case.

Another time-consuming task that most simulation operators will be familiar with is preparing machines for experiments and making sure that the OS, applications, and settings are properly configured. With a VDI, only a single OS image will need to be kept up-to-date for every OS distribution maintained by your organization. Instead of updating an Nvidia driver on twenty desktop machines, it is updated once on the golden Windows image, after which new VMs will all be deployed with that new driver.

### **Performance**

One area where a traditional desktop will always have an edge over VDIs is the performance of applications. If the organization's specific requirement is being able to run applications at a smooth 120 frames per second (FPS), a VDI could struggle to achieve that due to limitations imposed by connection brokers like VMware Horizon utilizing its Blast Extreme connection protocol, which has a 60 FPS limit. There will also need to be a trade-off between picture quality and smoothness. For instance, in a hospital setting, a VDI should prioritize maximizing picture quality for radiologists to accurately assess lossless and uncompressed CT scans, where preserving every detail is essential. However, in a battlelab environment, prioritizing picture quality might lead to occasional stutters in 3D applications. Consequently, if the organization emphasizes maximizing picture quality as a requirement for a jet fighter simulation, a VDI might not be the most suitable solution. Settings like these, however, can be defined on a per-VM basis. The VM that hosts a fighter-jet simulation could prioritize smoothness over picture quality, whereas the VM that is used to display and assess satellite imagery could make sure that no details are lost. But if that fighter-jet simulation should also maintain maximum picture quality, the VDI solution will prove to be insufficient.

### **Latency**

Moreover, the network connection between the endpoint and the VM introduces input latencies that may surpass the latency requirements of flight simulators, depending on their use case. While a 150ms input latency would be acceptable for concept development and experimentation, it might prove too high for training purposes. Before purchasing a pre-built VDI solution, it is recommended to conduct tests on available VDI configurations provided by suppliers to ensure that picture quality and input latencies are acceptable for the use cases foreseen by the organization.

For Cerebro, the VDI solution proved to be sufficiently capable of handling current use cases and other potential activities for the foreseeable future. The decision of whether to opt for a pre-built VDI solution from a supplier or to construct a custom-made solution using individual components was delegated to the department that would be responsible for managing the VDI. It was swiftly determined that the potential cost savings of a self-built solution were outweighed by the convenience of a pre-built system accompanied by round-the-clock support available on the same day.

## **HARDWARE CONSIDERATIONS**

A VDI is only as good as the hardware it runs on. When confronted with performance issues in a traditional desktop environment, there are established troubleshooting steps one can follow, such as monitoring CPU or GPU loads, to identify the underlying cause of the problem. With a VDI, the added hypervisor layer drastically increases the number of variables that could be causing performance issues. Selecting suitable hardware to ensure that it can handle the organization's use-cases is the first step in ensuring optimal performance and stability in a VDI setup.

### **Hyper-converged infrastructure**

When selecting hardware it will quickly become apparent that not all hardware components are inherently compatible with each other. Many suppliers of these components offer compatibility matrices that outline which other components, when integrated into the same system, will function properly and which combinations may lead to unexpected failures. Although it is possible to carry out thorough research to identify compatible hardware components, complications may still arise, requiring the need for support, and in such scenarios manufacturers may deflect their ownership of the problem, which ultimately hinders the constructive resolution of the issues at hand. Currently, there are companies providing what is known as a converged infrastructure solution. This solution involves bundling multiple components into a thoroughly tested and optimized system package. Although this approach restricts the range of available components and may introduce a delay in accessing, for example, the latest Nvidia GPU, one significant advantage of this approach is that in the event of component failures, there is a single point of contact for support and assistance.

During the hardware selection process, NLR opted not just for a converged infrastructure, but a hyper-converged infrastructure, or HCI. The difference between a converged infrastructure and an HCI is that the storage area network (SAN) is implemented virtually at a software level (vSAN), allowing for the central management of resources across all server instances.

## Graphics

Despite the restricted selection of components for an HCI, the majority of vendors do offer a range of options to choose from, as they will have tested various configurations. This paper does not intend to provide a full tutorial on designing and sizing a VDI environment in general. As mentioned before, there are numerous resources available on that subject, such as the freely available "VDI Design Guide" by Johan van Amersfoort from ITQ (Amersfoort, 2018). And while all components together will determine how a VDI performs, there are some aspects to a VDI in this particular context that will differ from a regular VDI implementation, which will be discussed in this paper.

One of those main difference between a 'regular' VDI and the VDI-powered battlelab is the abundance of GPUs. Without a GPU, most graphics-intensive applications will simply refuse to start. Although virtualization offers the option of using software-accelerated graphics within a VM, which will assign CPU resources to handle tasks typically managed by the GPU, this approach severely limits not only the VM's performance, but also the connection between the VM and the endpoint. Utilizing dedicated GPUs on the other hand enables the CPU to offload tasks that it is not well-suited for, resulting in improved VM performance, as well as faster encoding of frames sent to the endpoint.

In the world of professional GPUs, there are two main vendors to choose from at the time of writing: AMD and Nvidia. The HCI chosen by NLR was only validated with Nvidia's A40 GPUs, so this is one decision that was made for us. If the solutions considered by your organization do offer multiple types of GPUs, take into account that advancements in virtualization technologies move at an incredibly fast pace. It is therefore advisable to check which GPU will offer the best performance at the time of reading this paper. The GPU component within the VDI system is not one to compromise on though. As new applications emerge, they demand increasingly robust GPUs with larger VRAM capacities. It is worth noting that GPUs available today are more than twice as fast as those from five years ago, highlighting the rapid pace of development in this field as well.

Another thing that needs to be considered is whether every virtual machine will get a single dedicated GPU (passthrough) or if GPU resources will be shared across VMs (virtual GPU, or vGPU). Both approaches have their advantages and disadvantages, but the increased flexibility of vGPUs made for an easy decision in the case of Cerebro, where VMs are created and destroyed before and after every simulation exercise. Further details on this topic will be discussed later. Note that the use of virtual GPUs will require Nvidia GRID licensing, which is something to take into account when comparing GPU costs.

## Endpoints

Endpoints serve as the devices through which users log into the VMs provided by the VDI environment. Their primary function is to decode the video frames generated by the VM and display them, while also relaying user inputs from devices such as keyboards, mice, and joysticks. As all the processing takes place on the server, the endpoints can be as compact as a Raspberry Pi Zero. However, for graphics-intensive applications, it is advisable to opt for slightly more powerful endpoints to ensure efficient decoding of incoming video frames. Nowadays, most CPUs come equipped with integrated graphics units that assist in this task. While certain devices even incorporate dedicated GPUs, those could be excessive for a battlelab use case. Unless your organization intends to fill their battlelab environment with flight simulators featuring quad-monitor setups for image generators and other displays, like the United States Air Force Academy's Multi-Domain Laboratory (USAFA MDL) that has a total of 24 simulators like those displayed in Figure 2, all completely virtualized (ZedaSoft, 2021). In that case, an endpoint without a dedicated graphics card might struggle to properly decode the large-resolution frames that are presented to it.



Figure 2 ZedaSoft Zuse Simulation Stations at the USAFA MDL

When it comes to selecting an endpoint, there are numerous options available, and the quality of the accompanying client management software can vary significantly. The IGEL OS, a Linux-based OS designed specifically for virtualized environments, comes with their Universal Management Suite. No other software for managing endpoints has come close in terms of providing the same level of flexibility and user-friendliness. However, it is important to

note that USB driver support is a crucial requirement for a battlelab environment to ensure full functionality of various input devices such as joysticks and headsets. In this regard, the Windows operating system stands out with its exceptional out-of-the-box driver support. As a result, the Linux-based endpoint solutions offered by IGEL, Stratodesk, and 10ZiG might not be as suitable for a use case like Cerebro. Nevertheless, this aspect of the industry progresses rapidly as well, and driver support is a feature that can be expanded relatively quickly. Similar to the server hardware, it is advisable to evaluate trial versions of the endpoint clients being considered by your organization to ensure they can effectively handle the tasks required of them.

## **SOFTWARE DESIGN AND IMPLEMENTATION**

Whether or not the hardware will perform properly will also depend on the software installed on the hardware, and how that software is configured. In the case of the HCI solution chosen by NLR, this includes the hypervisor software, the management interface, and the software required to establish connections between the VMs and the endpoints. The software deployed on this specific HCI consists of VMware vSphere. The server nodes are therefore equipped with VMware's own hypervisor software, ESXi. Other hypervisors like Citrix's XenServer or Microsoft's Hyper-V provide similar functionalities, but for NLR it made sense to adhere to the VMware vSphere product suite.

The central management interface provided with VMware vSphere is called vCenter Server. Through this interface, multiple server nodes can be grouped into clusters to facilitate resource sharing among nodes. It enables the creation and monitoring of VMs, the establishment of network file shares on the vSAN layer, and offers various other functionalities. It serves as the initial point of setup for any VMware-based virtualization environment. In the context of a virtualized battlelab, there are not many configurations in vCenter that need to be adjusted differently compared to regular virtualized environments, apart from the configuration of GPUs for either passthrough or shared mode. In the case of shared mode, the vGPU profile assigned to VMs needs to be sufficiently large to accommodate for any software GPU requirements.

### **Securing the Horizon**

VMs can be manually created, but VMware vSphere tightly integrates with VMware Horizon, which allows for the automatic provisioning of entire desktop pools, or groups of VMs, based on one or more golden images created with the vCenter interface. This process is well-documented in VMware's online knowledge base, so the creation of desktop pools will not be discussed in further detail, but it is worth noting that the use of desktop pools offers additional security measures.

First of all, there are settings available in VMware Horizon that will ensure that once a user logs out of a VM, the entire VM along with all of its contents is destroyed. This feature is particularly crucial for military simulations conducted at higher classifications, where access is often strictly limited to those with a need-to-know. If a VM persists between completely different simulations, there is a good chance that information from the previous simulation will be visible to the operators performing subsequent simulations, as they could be utilizing the same VM. By destroying the VM after every simulation and forcing Horizon to create new VMs, the likelihood of presenting data from other simulations to operators during subsequent experiments is eliminated. It is therefore also of critical importance to securely backup any simulation data required for after action review, debriefing, or analysis to a central storage before a user logs out, as this data will otherwise be lost as well. Pre-shutdown scripts can be initiated to automate this backup process.

Additionally, users must possess the appropriate permissions to utilize a VM within a desktop pool. They can be assigned to one or multiple desktop pools, meaning that when they log into the thin client without the necessary rights to any desktop pool, the VMware Horizon Client will display an empty list of available desktop pools. This ensures that even if an unauthorized individual gains access to a thin client, they will be unable to access server-side VMs due to the absence of proper rights assignments.

### **Blast Extreme**

The VMware Horizon Client mentioned above is the software installed on an endpoint that allows a user to log into a VM. Administrators have the flexibility to configure the protocol used for this connection, although leaving it up to the user's discretion is not recommended, especially in the battlelab use case. VMware Horizon offers three protocol options: Microsoft's Remote Desktop Protocol (RDP), Teradici's PC-over-IP (PCoIP), and VMware's own Blast Extreme. When testing these connection protocols with graphics-intensive applications, it becomes evident that RDP

is not suitable for such tasks. RDP was primarily designed for basic remote access to desktop environments rather than delivering high-performance graphics over a remote connection.

Blast Extreme incorporates adaptive techniques and encoding algorithms that dynamically adjust the display quality based on network conditions, which gives it an edge over PCoIP when it comes to the delivery of graphics-intensive applications. It should also be noted that PCoIP is scheduled to be phased out from the list of available Horizon protocols by the end of 2028 (VMware, 2023). Therefore, if your organization intends to adopt VMware solutions, it is advisable to prioritize optimizing the Blast Extreme settings to ensure an optimal user experience.

One of the only resources available that will provide a starting point for optimizing Blast Extreme for use cases like these is the VDI Design Guide Part II (Amersfoort, 2021). For Windows-based VMs, Blast Extreme settings are defined in the system registry. Managing registry settings in distributed environments, such as a battlelab, can be simplified by utilizing a Group Policy Object (GPO) with the Blast Extreme ADMX template. This template is available to all VMware customers through their customer portal. For Linux-based VMs, the settings are straightforwardly defined inside a file. The recommended initial values for these settings are provided by Table 1.

**Table 1 Recommended initial values for Blast Extreme settings (Amersfoort, 2021, pp. 271-273)**

Name	Value
EncoderH264Enabled	1
EncoderNvidiaH264Enabled	1
H264minQP	10
H264maxQP	21
MaxBandwidthKbpsPerMegaPixelSlope	100000
UdpEnabled	1
EncoderHEVCEnabled	1
EncoderNvidiaHEVCEnabled	1

Note that Blast Extreme, by default, is limited to 30 FPS. This can be increased to a maximum of 60 FPS with the *MaxFPS* setting. It is worth noting that the impact of modifying any of these settings is observable in real-time, facilitating the process of determining the optimal values for your organization's VDI solution.

All of these Blast Extreme settings are implemented inside the VMs on the server side. However, there is an additional client-side setting related to VMware Horizon that significantly affects the responsiveness of USB devices, particularly joysticks and other gamepad-like devices. VMware Horizon, by default, buffers its audio and video to minimize audio-video synchronisation issues in latency sensitive or bandwidth-restricted environments. As a result, there may be a delay of up to half a second before video frames are displayed on the screen. This delay can cause joysticks, used in applications such as flight simulators, to exhibit a perceived latency of approximately 300-400ms, which is generally considered unacceptable.

Attempting to troubleshoot this problem without prior knowledge of Horizon's buffering behavior could potentially lead to a series of time-consuming tests, such as evaluating various joystick models and other gamepads, conducting network bandwidth assessments, exploring alternative remote desktop connection software and protocols, and more. Fortunately, the solution to this issue is documented in a VMware knowledge base article (VMware, 2022), which simply involves adding the following line to a VMware configuration file on the endpoint:

*RemoteDisplay.disableAudioAndVideoBuffering = FALSE*

By implementing and fine-tuning all of the aforementioned settings according to your specific VDI environment, applications should exhibit a seamless performance on the endpoints, akin to running on a physical desktop.

### Application management

The implementation of a VDI opens up new possibilities for application management. The vSphere suite of virtualization software includes something called AppVolumes. In a typical scenario, you would install the desired software on a regular desktop or VM. This can be done manually or by employing an automated method to handle the installation process. However, if the same software is needed on another machine, the entire installation process must be repeated. This includes any updates or removals of the software, which would need to be performed on each machine where the software is deployed.

Using VMware AppVolumes, software installation and updates are streamlined by performing them only once on a designated *packaging* VM. Within the AppVolumes interface, the creation of a software package can be initiated by assigning it a name and a virtual disk template. The template defines the type of virtual disk on which the application(s) will be installed, with larger applications necessitating larger virtual disks. Once the process is initialized and you log into the packaging VM, the virtual disk is attached to the VM, and any modifications made to the system, including software installations, are recorded onto that virtual disk. Once you have completed the setup, the virtual disk is finalized and ready for deployment. You can provide access to the created software package on a per user basis.

When users log into a VM and they have the correct entitlements, AppVolumes rapidly assigns a copy of the virtual disk associated with the software package during the login process, typically within seconds. Consequently, the installed applications become instantly accessible to users upon logging in. If an update is required for the software, a single update to the virtual disk suffices, simplifying the update process.

### The Cerebro Portal

One of the requirements for the Cerebro battlelab was the ‘single button startup’. A single button press that would boot up thin clients, provision the required virtual machines, and automatically start any applications with the correct scenarios and configurations based on user profiles, ready for use. While this is a possibility with VDIs, it is not a standard feature.

To fulfill this requirement, the Cerebro project team made the decision to develop a web portal. This portal enables Cerebro administrators to configure, initiate, monitor, and shut down the entire battlelab environment, providing a centralized and user-friendly interface for managing the various components.

Most of the vSphere products, including VMware Horizon, AppVolumes, and vCenter Server, provide an API (Application Programming Interface). Through these APIs, nearly every action and configuration available in the user interface can be accessed programmatically. Tasks such as destroying a virtual machine or granting users access to software packages can be automated using these APIs.

In the configuration stage, the administrator can associate users to Horizon desktop pools and a collection of AppVolumes applications, or *appstack*. The administrator is then also able to match those applications to certain configuration, see Figure 3.

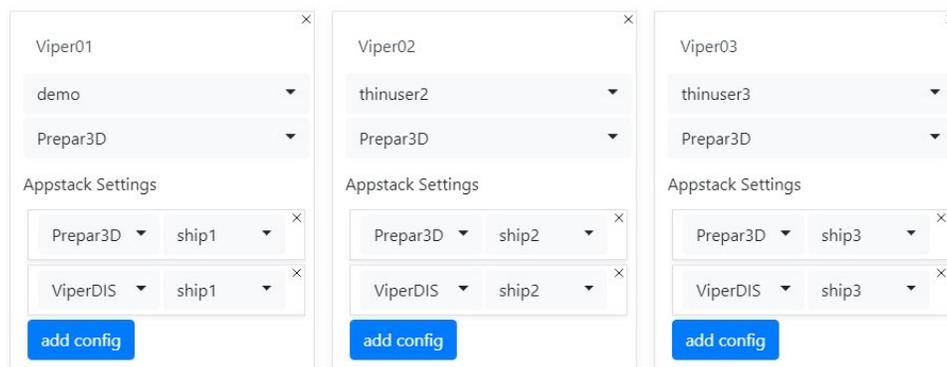


Figure 3 Scenario configuration with the Cerebro Portal

These configurations decide, for example, the initial positions of Viper01, Viper02, and Viper03, as these need to be unique. The same is true for their respective DIS entity IDs; a shared default configuration would not work in this case.

Once all the roles for the operators have been defined, the scenario can be saved to a database. From the deployment page, scenarios can be selected and started. The phases of the deployment phase are depicted in Figure 4, which shows a portion of the deployment page presented to the experiment lead. Since every user starts without any rights to desktop pools or software packages, the first steps in the automated deployment phase consist of assigning users the necessary rights for their assigned desktop pools and software packages through the Horizon and AppVolumes APIs. When the users are given the proper entitlements, Horizon automatically starts to create new VMs.



**Figure 4** The different deployment phases as shown inside the Cerebro Portal

When all the VMs are provisioned, a Wake-On-LAN message is broadcast to the endpoints, triggering their startup and enabling operators to log in using their credentials. The VMware Horizon Client launches automatically, and since each operator is entitled to only one desktop pool, the client seamlessly connects them to a VM from their assigned desktop pool without requiring any user interaction.

Upon user login, AppVolumes attaches the required software packages to the corresponding VM, making the applications instantly accessible. Subsequently, the portal transfers the necessary scenario and configuration files to the appropriate locations within the VM. The applications are then automatically launched. All with a single press of the *play* button.

During the simulation experiment, individual applications can be restarted if any issues arise, and scenario and configuration files can be edited and reloaded as needed. Once the experiment concludes, any essential data is backed up before issuing a shutdown command to the endpoints. Simultaneously, the virtual machines are destroyed, and the desktop pools are disabled to prevent the provisioning of new VMs. Finally, all user rights to desktop pools and software packages are revoked, ensuring a clean slate for the next simulation experiment.

This entire process takes only a matter of minutes, allowing for quick turnaround between simulation experiment deployments. The portal makes optimal use of the VDI environment and its software components, showcasing the incredible flexibility and convenience of managing simulation experiments in a battlelab powered by a VDI solution.

## CONCLUSION

Even without a custom-made portal that ties all of a VDI's components together, this type of infrastructure greatly simplifies hosting simulation experiments compared to traditional setups. However, it is important to reiterate that a VDI will never deliver the same application performance. A desktop will always outperform a virtual machine. A trade-off analysis is vital to ensure that the performance impact of running applications on a VDI remains within acceptable margins. Test the applications that your organization intends to use to determine whether or not the VM can run them at the required number of frames per second. Check whether the input latencies of the devices your organization intends to plug into the endpoints are acceptable, because a very fluent display will do nothing to improve a user's experience if their joystick inputs are delayed by half a second.

It might be tempting to deploy hypervisors and other virtualization software on desktops that your organization has no use for, in an effort to quickly perform these tests before investing in hardware that is dedicated to handling VDI workloads. The earliest version of Cerebro ran on a similar setup with a similar goal. If there is one thing to take away from reading this paper, it is that this would be a large waste of time, since the performance of such a setup is incomparable to the performance of dedicated VDI hardware. VDI vendors will often have their own test

environments, allowing you to install the software you want to test and connect any input device you wish to evaluate. If the tests are successful, the risk of investing in hardware that will not be capable of properly running your applications will be practically eliminated.

For NLR, in the case of Cerebro, every one of these tests proved that a VDI is capable of handling the use cases foreseen for the battlelab. Graphics-intensive applications would run at a steady 60 FPS, even during intense manoeuvring, and input latencies of the joysticks used by NLR did not exceed 150ms, which is the maximum allowed input latency for concept development and experimentation purposes.

The true advantages of a VDI become evident in the management of the simulation environment, where the time required for maintaining and updating operating systems and applications is significantly reduced compared to the traditional desktop environment. Additionally, the downtime between multiple experiments is minimized. That on its own would suffice to call this endeavor a resounding success. But Cerebro has yet to touch the full potential of a VDI, as it lends itself perfectly for hosting Kubernetes clusters for MSaaS (Modelling & Simulation as a Service) applications, or the facilitation of high-definition rendering pipelines to process locally hosted terrain and culture data for more immersive simulation experiences. The versatility of this infrastructure allows organizations to tailor its utilization according to their specific needs. Hence, it is highly advisable for organizations aiming to establish a simulation center to explore the potential of utilizing a VDI.

## REFERENCES

- Amersfoort, J. (2018). *VDI Design Guide: A comprehensive guide to help you design VMware Horizon, based on modern standards*. ITQ.
- Amersfoort, J. (2021). *VDI Design Guide Part II*. ITQ.
- NATO. (2022, 11 17). Retrieved from ACT.NATO.INT: <https://www.act.nato.int/articles/defence-counsellors-discuss-multi-domain-operations-nato-2030-ukraine>
- Technavio. (2022). *Military Simulation and Virtual Training Market by Product and Geography - Forecast and Analysis 2022-2026*. Technavio.
- VMware. (2022, 11 22). *USB redirected Devices experience Input Data Latency with Blast (67759)* . Retrieved from VMware Knowledge Base: <https://kb.vmware.com/s/article/67759>
- VMware. (2023, 3 4). *Announcing end of support for PCoIP in VMware Horizon*. Retrieved from VMware: <https://blogs.vmware.com/euc/2023/03/announcing-end-of-support-for-pcoip-in-vmware-horizon.html>
- Wilson, E., & Elsey, D. (2023). *Adapt or Die: Lessons in MDS and Training in Modern Warfare*. IT2EC. Rotterdam.
- ZedaSoft. (2021, 9 9). *Introducing the United States Air Force Academy (USAF) Multi-Domain Laboratory (MDL)*. Retrieved from ZedaSoft: <https://www.zedasoft.com/news/introducing-the-united-states-air-force-academy-usafa-multi-domain-laboratory-mdl/>
- ZedaSoft. (2023, 6 1). *Zuse*. Retrieved from ZedaSoft: <https://www.zedasoft.com/product/zuse/>