UNITED STATES DISTRICT COURT
WESTERN DISTRICT OF TEXAS
WACO DIVISION

Intellectual Ventures II LLC, Plaintiff, v. VMware, Inc. Defendant.	)	Civil Action No. 6:20-cv-220 **JURY TRIAL DEMANDED**

COMPLAINT FOR PATENT INFRINGEMENT

Plaintiff, Intellectual Ventures II LLC ("IV"), for its complaint against defendant, VMware, Inc. ("VMware"), hereby alleges as follows:

THE PARTIES

1. Intellectual Ventures II LLC ("Intellectual Ventures II") is a Delaware limited liability company having its principal place of business located at 3150 139th Avenue SE, Bellevue, Washington 98005.

2. Upon information and belief, VMware is a Delaware corporation with its headquarters located at 3401 Hillview Avenue, Palo Alto, California. VMware has regular and established places of business in this District, including two in Austin, Texas with over 700 employees. VMware also has at least two other offices in Texas—one in Coppell, Texas, and the other in Farmers Branch, Texas. Defendant may be served with process through its registered agent, Corporation Service Company, d/b/a CSC-Lawyers Incorporating Service Company, 211 E. 7th Street, Ste. 620, Austin, Texas 78701-3136

JURISDICTION

3. This is a civil action for patent infringement under the patent laws of the United States, 35 U.S.C. § 271, *et seq.* This Court has subject matter jurisdiction under 28 U.S.C. §§ 1331 and 1338(a).

4. This Court has general personal jurisdiction over VMware because VMware is engaged in substantial and not isolated activity at its regular and established places of business within this judicial district. This Court has specific jurisdiction over VMware because VMware has committed acts of infringement giving rise to this action and has established more than minimum contacts within this judicial district, such that the exercise of jurisdiction over VMware in this Court would not offend traditional notions of fair play and substantial justice.

5. Venue is proper in this judicial district pursuant to 28 U.S.C. §§ 1391(b)-(c) and 1400(b) because VMware maintains regular and established places of business and has committed acts of patent infringement within this judicial district.

FACTUAL BACKGROUND

6. Intellectual Ventures Management, LLC ("Intellectual Ventures") was founded in 2000. Since then, Intellectual Ventures has been involved in the business of inventing. Intellectual Ventures fosters inventions and facilitates the filing of patent applications for those inventions; collaborates with others to develop and patent inventions; and acquires and licenses patents from individual inventors, universities, corporations, and other institutions. A significant aspect of Intellectual Ventures' business is managing the plaintiff in this case, Intellectual Ventures II.

7. One founder of Intellectual Ventures is Nathan Myhrvold, who worked at Microsoft from 1986 until 2000 in a variety of executive positions, culminating in his appointment as the company's first Chief Technology Officer ("CTO") in 1996. While at Microsoft, Dr. Myhrvold

founded Microsoft Research in 1991, and was one of the world's foremost software experts. Between 1986 and 2000, Microsoft became the world's largest technology company.

8. Under Dr. Myhrvold's leadership, IV acquired more than 70,000 patents covering some important inventions of the Internet era. Many of these inventions coincided with Dr. Myhrvold's successful tenure at Microsoft.

9. Two significant accomplishments of the Internet era are the related technologies of cloud computing and virtualization. Cloud computing enables ubiquitous access to shared pools of configurable computing system resources, such as memory, and higher-level services, such as virtual desktops or applications that run on these system resources. These resources and services, often collectively referred to as "the cloud," can be rapidly provisioned with minimal management effort, often over the Internet. Virtualization allows for the creation of multiple software environments of dedicated resources that each simulate for its users a computing environment that was traditionally physically distinct such as a server or an operating system.

10. Three of the many beneficial consequences of cloud computing and virtualization are the consolidation of server resources that businesses require to run a varied suite of software applications, the increased continuity of those server resources, and the outsourcing of those server resources to companies that operate large server clouds. Server resources are consolidated by replacing large numbers of physically distinct computer servers with a smaller number of computer servers, where each can host many virtual servers (also known as virtual machines). Each virtual machine (or "VM") can execute the functions previously delivered by an entire computer server. Many VMs can run on a single physical server, which results in server consolidation and promotes scalability. Increased continuity is achieved in large part by enabling VMs to seamlessly move from one computer server to another computer server, for example, in

the event that the first server becomes overloaded. Outsourcing of server resources requires technology that provides several different customers (i.e., users) access to a pool of those resources across a network, while providing each of those customers fault tolerance, performance and security isolation from the other customers accessing that same pool of resources.

11. The consolidation of servers by replacing a physical server with a virtual server has enabled services to migrate into giant centralized "clouds," such as the Google or Amazon cloud, and more generally allows access to shared pools of configurable computing resources and services over the Internet. This enables services to be delivered to a given customer using centralized servers that seem as though they were located just a few feet away from the user, even if they are actually located hundreds or even thousands of miles away (hence the terms "virtual service" and "virtual server").

12. The concept of transferring VMs, in the event of a computer server overload, to a different computer server has made VMs far less prone to outages, which in turn has changed how companies that provide virtualization technologies such as VMware are perceived. Such virtualization technology companies were previously associated with the limited function of server consolidation and thought of as relatively unreliable. With the more resilient architecture afforded by VMs that can move from physical server to physical server, however, virtualization companies are now also recognized as critical enablers of business continuity and availability.

13. Outsourcing of server resources to pools of physical servers, such as those found in data centers, enables large-scale and inexpensive delivery of virtual server services to a wide range of customers scattered across the Internet. The availability of large-scale and inexpensive virtual server services to the public at large has further enabled customers to focus less on building and maintaining their own network of physical computer servers and more on running their core

businesses. It has also allowed small and mid-size businesses to enjoy the sorts of cutting-edge computing technologies that were previously reserved for larger companies that could afford to build and maintain their own sophisticated networks of physical servers.

14. VMware provides virtualization solutions and services to its customers. VMware's product offerings include: vSphere (including vCenter Server, vMotion and ESX/ESXi hypervisor), VMware's foundational virtualization platform; Horizon/Horizon View; SD-WAN/VeloCloud, VMware's SD-WAN solution; NSX, VMware's network virtualization solution; and vSAN, VMware's storage virtualization product. With 117 worldwide offices, VMware markets and sells these solutions and services throughout the globe, including in the United States and Texas.

THE PATENTS-IN-SUIT

15. On May 10, 2016, the Patent and Trademark Office (PTO) issued United States Patent No. 9,338,217 ("the '217 patent"), titled METHOD AND APPARATUS FOR COMPUTING WITHIN A WIDE AREA NETWORK. The '217 patent is valid and enforceable. A copy of the '217 patent is attached as Exhibit A.

16. The '217 patent relates to a system, method and/or apparatus for an improved way to provide remotely hosted virtual machines to users.

17. On June 20, 2017, the PTO issued United States Patent No. 9,686,378 ("the '378 patent"), titled CONTENT MANAGEMENT AND TRANSFORMATION SYSTEM FOR DIGITAL CONTENT. The '378 patent is valid and enforceable. A copy of the '378 patent is attached as Exhibit B.

18. On July 28, 2015, the PTO issued United States Patent No. 9,092,546 ("the '546 patent"), titled CONTENT MANAGEMENT AND TRANSFORMATION SYSTEM FOR

DIGITAL CONTENT. The ’546 patent is valid and enforceable. A copy of the ’546 patent is attached as Exhibit C.

19. On March 21, 2006, the PTO issued United States Patent No. 7,016,963 (“the ’963 patent”), titled CONTENT MANAGEMENT AND TRANSFORMATION SYSTEM FOR DIGITAL CONTENT. The ’963 patent is valid and enforceable. A copy of the ’963 patent is attached as Exhibit D.

20. The ’546, ’378 and ’963 patents relate to systems, methods and/or apparatus for an improved way to invoke and deliver server-based applications to remote client devices based on unique client characteristics.

21. On November 9, 2004, the PTO issued United States Patent No. 6,816,464 (“the ’464 patent”), titled METHOD, SYSTEM, AND COMPUTER PROGRAM PRODUCT FOR ROUTE QUALITY CHECKING AND MANAGEMENT. The ’464 patent is valid and enforceable. A copy of the ’464 patent is attached as Exhibit E.

22. The ’464 patent relates to a system, method and/or apparatus for an improved way to test candidate network routes and select a subset thereof for interconnecting gateways across a wide area network (WAN).

23. Intellectual Ventures II is the owner and assignee of all rights, title and interest in and to the ’217 patent, the ’378 patent, the ’546 patent, the ’963 patent, and the ’464 patent, and holds all substantial rights therein, including the rights to grant licenses, to exclude others, and to enforce and recover past damages for infringement of those patents.

COUNT I
(VMware’s Infringement of U.S. Patent No. 9,338,217)

24. Paragraphs 1-23 are reincorporated by reference as if fully set forth herein.

25. The elements claimed by the ’217 patent, taken alone or in combination, were not well-understood, routine or conventional to one of ordinary skill in the art at the time of the invention. Rather, the ’217 patent claims and teaches, *inter alia*, an improved way to provide remotely hosted virtual machines to users, which was not present in the state of the art at the time of the invention. The invention improved upon then existing remote computing technology by providing a hosting platform capable of delivering virtual machines to remote clients over a network that each appear to clients as a fully functional computer. The invention further improved on prior art solutions by storing a user’s personal state on the remote hosting platform and allowing migration of the virtual machine across hosts while maintaining said personal state.

26. Instead of providing downloadable applets which had to be run on a user’s local machine, or allowing limited remote control of conventional computers in a master/slave configuration, the inventions of the ’217 patent allowed for delivery and use of remotely hosted virtual machines via a network, which were able to maintain user personal state and migrate from host to host. Thus, the resulting virtual machines were more reliable, untethered to a particular host or client machine, and provided full functionality to users relative to the prior art systems discussed above.

27. The invention represented a technical solution to an unsolved technological problem. The written description of the ’217 patent describes, in technical detail, each of the limitations in the claims, allowing a person of skill in the art to understand what those limitations cover, and therefore what was claimed, and also understand how the non-conventional and non-generic ordered combination of the elements of the claims differ markedly from what had been conventional or generic in the industry at the time of the inventions of the ’217 patent. More specifically, the claims of the ’217 patent each recite receiving at a server from at least one

network accessible computer a request, the host configured to implement a network accessible stored personal state which is remotely stored at a remote server and accessible via the internet, the network accessible stored personal state including a network accessible stored prior state of a workspace from a previously used network accessible computer, the host computer being posted for collaboration. Further, the claims recite that the server determines the workspace for the at least one network accessible computer from the network accessible stored personal state, loads the workspace into the at least one network accessible computer, wherein the workspace includes one or more windows that were open at the time of previous use, receives another request via the network from the client computer to collaborate with the host computer, and facilitate the collaboration of the host computer and the client computer over the network.

28. The system covered by the asserted claims, therefore, differs markedly from the conventional and generic systems in use at the time of this invention, which *inter alia* lacked the claimed combination of network accessible computers capable of providing fully functional virtual machines to remote clients. Further, the claimed inventions differ from prior art systems by remotely storing a personal state allowing for failover and migration, and also facilitating collaboration between the host computer and a client computer.

29. As described above, the ’217 patent is drawn to solving a specific, technical problem arising in the context of remotely hosted virtual machine delivery and operation. Consistent with the problem addressed being rooted in such remotely hosted virtual computing environments, the ‘217 patent’s solutions consequently are also rooted in that same technology and cannot be performed with pen and paper or in the human mind.

30. VMware has directly infringed, and continues to directly infringe, literally and/or by the doctrine of equivalents, individually and/or jointly, at least claims 1, 2, 4, 7, 8, 14-17, and

19 of the ’217 patent by making, using, testing, selling, offering for sale and/or importing into the United States products and/or services covered by one or more claims of the ’217 patent. VMware’s products and/or services that infringe the ’217 patent include, but are not limited to, vSphere, ESX/ESXi, Horizon/View, vCenter Server, Connection Server, vMotion and vSAN, and any other VMware products and/or services, either alone or in combination, that operate in substantially the same manner.

31. Claim 14 of the ’217 patent is reproduced below:

14. A method comprising:
receiving a first request over a network at a server from at least one network accessible computer of a plurality of network accessible computers, the host computer configured to:
implement a network accessible stored personal state, wherein the network accessible stored personal state is remotely stored at a remote server such that the network accessible stored personal state is accessible via internet, which permits the at least one network accessible computer to operate as a host computer for a client computer over the internet, the network accessible stored personal state including a network accessible stored prior state of a workspace of a previously used network accessible computer, and
post the host computer for collaboration;
determining, by the server, the workspace for the at least one network accessible computer from the network accessible stored personal state;
loading, by the server, the workspace into the at least one network accessible computer, wherein the workspace includes one or more windows that were open at a time the previously used network accessible computer was previously used;
receiving a second request over the network at the server from the client computer to collaborate with the host computer; and
facilitating by the server the collaboration of the host computer and the client computer over the network.

32. As one non-limiting example, VMware Horizon as integrated with vSphere practices a method of receiving a first request over a network at a server from at least one network accessible computer of a plurality of network accessible computers. The vSphere back-end infrastructure, such as the View Connection Server and vCenter Server facilitate the delivery of virtual desktops and applications which run on ESX/ESXi servers. The ESX/ESXi hosts communicate requests, over a network, to the View Connection Server and vCenter Server, as seen below:

VMware Horizon 7 is the leading platform for virtual desktops and applications.

Provide end users access to all of their virtual desktops, applications, and online services through a single digital workspace.

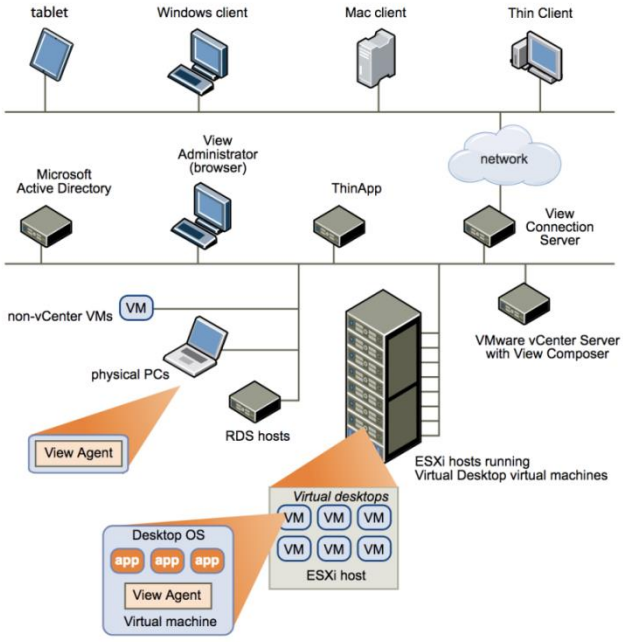
View Connection Server

This software service acts as a broker for client connections. View Connection Server authenticates users through Windows Active Directory and directs the request to the appropriate virtual machine, physical or blade PC, or Windows Terminal Services server.

View Connection Server provides the following management capabilities:

- Authenticating users
- Entitling users to specific desktops and pools
- Assigning applications packaged with VMware ThinApp to specific desktops and pools
- Managing local and remote desktop sessions
- Establishing secure connections between users and desktops

VMware vCenter Server – Manages the vSphere hosts and VMs. Also provides provisioning tasks and is used by Connection Servers and Composer to create VMs and clones, and App Volumes to mount AppStacks and user writeable volumes.



33. VMware Horizon as integrated with vSphere includes a host computer configured to implement a network accessible stored personal state, which is remotely stored at a remote server such that it is accessible via the internet, which permits the at least one network accessible computer to operate as a host computer for a client computer over the internet. For instance, the

ESX/ESXi hosts and virtual machines hosted thereon, can use vSAN shared storage. The shared storage allows for virtual machine state information, including personalized information, to be stored at the vSphere back-end and available to all ESX/ESXi hosts and virtual machines they host, as seen below:

Figure 1 shows how multiple vSphere hosts with several virtual machines running on them can use VMFS to share a common clustered pool of storage.

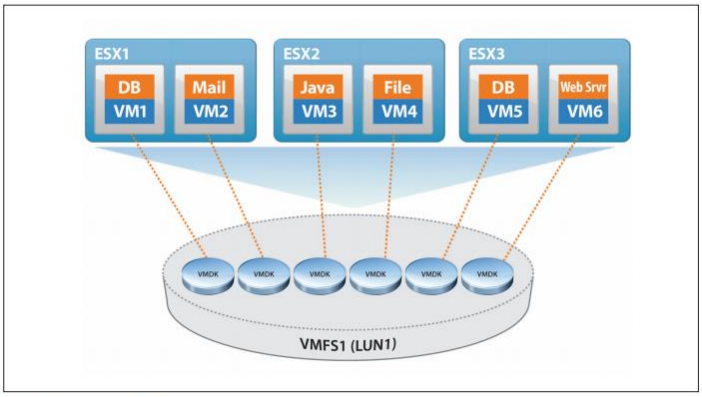
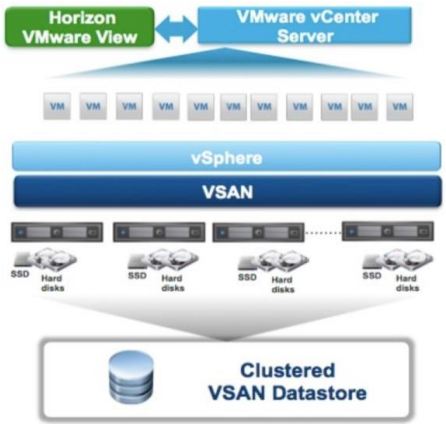


Figure 1. VMFS as a Common Pool of Storage

VMFS empowers IT organizations to greatly simplify virtual machine provisioning by efficiently storing the entire machine state in a central location. It enables multiple instances of vSphere hosts to access shared virtual machine storage concurrently. It also enables virtualization-based distributed infrastructure services such as vSphere DRS, vSphere HA, vMotion, vSphere Storage DRS and Storage vMotion to operate across a cluster of vSphere hosts. In short, VMFS provides the foundation that enables the scaling of virtualization beyond the boundaries of a single system.

VMFS enables portability of virtual machines across vSphere hosts to provide high availability while lowering management overhead. As a CFS and cluster volume manager (CVM), VMFS enables unique virtualization services that leverage live migration of running virtual machines from one vSphere host to another. VMFS also facilitates automatic restart of a failed virtual machine on a separate vSphere host, and it supports clustering virtual machines across different vSphere hosts. File-level lock management provides the foundation needed for the multiserver virtualization that enables vSphere HA, vSphere DRS, vMotion, vSphere Storage DRS, Storage vMotion and VMware vSphere Fault Tolerance (FT), causing less downtime and faster recovery.



34. The network accessible stored personal state provided by VMware Horizon as integrated with vSphere further includes a network accessible stored prior state of a workspace of a previously used network accessible computer. The virtual machine state information stored on the vSAN shared storage reflects stored personalized setting and preferences as well as the real-time state of running virtual machines, including all programs and processes running at that time, as seen below:

> Before a virtual machine can be started, its files must be accessible from one of the active cluster hosts that the master can communicate with over the network

Desktop Pools for Specific Types of Workers

View provides many features to help you conserve storage and reduce the amount of processing power required for various use cases. Many of these features are available as pool settings.

The most fundamental question to consider is whether a certain type of user needs a stateful desktop image or a stateless desktop image. Users who need a stateful desktop image have data in the operating system image itself that must be preserved, maintained, and backed up. For example, these users install some of their own applications or have data that cannot be saved outside of the virtual machine itself, such as on a file server or in an application database.

Stateful desktop images Also known as persistent desktops, these images might require traditional image management techniques. Stateful images can have low storage costs in conjunction with certain storage system technologies. Backup and recovery technologies such as VMware Consolidated Backup and VMware Site Recovery Manager are important when considering strategies for backup, disaster recovery, and business continuity.

> **First**, the entire state of a **virtual machine is encapsulated** by a set of files stored on shared storage. VMware's clustered Virtual Machine FileSystem (VMFS) allows multiple installations of ESX Server to access the same virtual machine files concurrently.

Persistent Disks for Dedicated Desktops

When you create dedicated-assignment desktop pools, View Composer can also optionally create a separate persistent virtual disk for each virtual desktop. The end user's Windows profile and application data are saved on the persistent disk. When a linked clone is refreshed, recomposed, or rebalanced, the contents of the persistent virtual disk are preserved. VMware recommends that you keep View Composer persistent disks on a separate datastore. You can then back up the whole LUN that holds persistent disks.

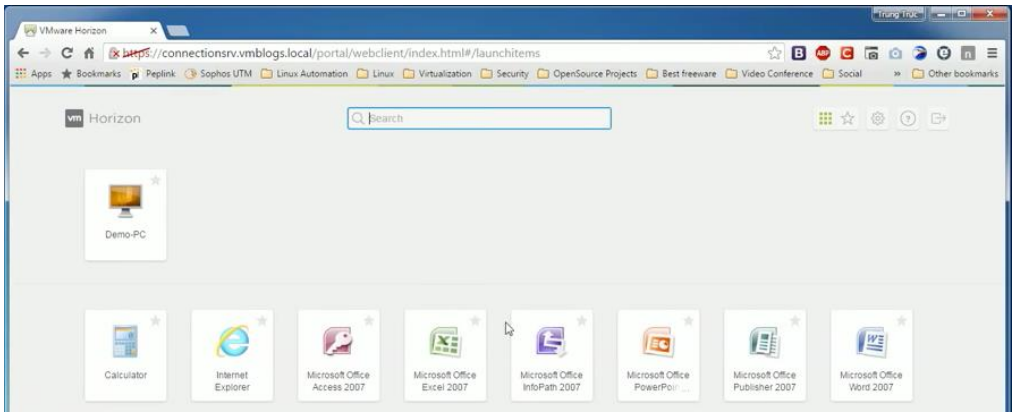
35. VMware Horizon as integrated with vSphere further posts the host computer for collaboration, as seen below:

Automated Pools That Contain Full Virtual Machines

To create an automated desktop pool, View dynamically provisions machines based on settings that you apply to the pool. View uses a virtual machine template as the basis of the pool. From the template, View creates a new virtual machine in vCenter Server for each desktop.

Worksheet for Creating an Automated Pool That Contains Full Virtual Machines

When you create an automated desktop pool, the View Administrator Add Desktop Pool wizard prompts you to configure certain options. Use this worksheet to prepare your configuration options before you create the pool.



Host or cluster	Select the ESXi host or cluster on which the virtual machines run. In vSphere 5.1 or later, you can select a cluster with up to 32 ESXi hosts.

36. VMware Horizon as integrated with vSphere determines, by the server, the workspace for the at least one network accessible computer from the network accessible stored personal state. Each time a virtual machine is instantiated on an ESX/ESXi host the vSphere back-end retrieves the stored state information from vSAN storage that corresponds with the particular VM end user. The same process also occurs when recovering a VM from a failed state or migrating a VM from one host to another using vMotion, as seen below:

VMFS enables portability of virtual machines across vSphere hosts to provide high availability while lowering management overhead. As a CFS and cluster volume manager (CVM), VMFS enables unique virtualization services that leverage live migration of running virtual machines from one vSphere host to another. VMFS also facilitates automatic restart of a failed virtual machine on a separate vSphere host, and it supports clustering virtual machines across different vSphere hosts. File-level lock management provides the foundation needed for the multiserver virtualization that enables vSphere HA, vSphere DRS, vMotion, vSphere Storage DRS, Storage vMotion and VMware vSphere Fault Tolerance (FT), causing less downtime and faster recovery.

VMFS empowers IT organizations to greatly simplify virtual machine provisioning by efficiently storing the entire machine state in a central location. It enables multiple instances of vSphere hosts to access shared virtual machine storage concurrently. It also enables virtualization-based distributed infrastructure services such as vSphere DRS, vSphere HA, vMotion, vSphere Storage DRS and Storage vMotion to operate across a cluster of vSphere hosts. In short, VMFS provides the foundation that enables the scaling of virtualization beyond the boundaries of a single system.

First, the entire state of a **virtual machine is encapsulated** by a set of files stored on shared storage. VMware's clustered Virtual Machine FileSystem (VMFS) allows multiple installations of ESX Server to access the same virtual machine files concurrently.

37. VMware Horizon as integrated with vSphere loads, by the server, the workspace into the at least one network accessible computer, wherein the workspace includes one or more windows that were open at a time the previously used network accessible computer was previously used. The vSphere back-end infrastructure, such as the View Connection Server and vCenter Server load the appropriate virtual machine state files stored in the vSAN shared storage onto the virtual machine instantiated on the ESX/ESXi hosts. In the case of a recovery from a failed state or a migration from one host to another, the stored state information includes all programs, processes and windows being used at the time of the fail or migration, as seen below:

VMFS enables portability of virtual machines across vSphere hosts to provide high availability while lowering management overhead. As a CFS and cluster volume manager (CVM), VMFS enables unique virtualization services that leverage live migration of running virtual machines from one vSphere host to another. VMFS also facilitates automatic restart of a failed virtual machine on a separate vSphere host, and it supports clustering virtual machines across different vSphere hosts. File-level lock management provides the foundation needed for the multiserver virtualization that enables vSphere HA, vSphere DRS, vMotion, vSphere Storage DRS, Storage vMotion and VMware vSphere Fault Tolerance (FT), causing less downtime and faster recovery.

Desktop Pools for Specific Types of Workers

View provides many features to help you conserve storage and reduce the amount of processing power required for various use cases. Many of these features are available as pool settings.

The most fundamental question to consider is whether a certain type of user needs a stateful desktop image or a stateless desktop image. Users who need a stateful desktop image have data in the operating system image itself that must be preserved, maintained, and backed up. For example, these users install some of their own applications or have data that cannot be saved outside of the virtual machine itself, such as on a file server or in an application database.

> **Second**, the active **memory and precise execution state of the virtual machine is rapidly transferred** over a high speed network. This allows the virtual machine to instantaneously switch from running on the source ESX Server to the destination ESX Server. VMotion keeps the transfer period imperceptible to users by keeping track of on-going memory transactions in a bitmap. Once the entire memory and system state has been copied over to the target ESX Server, VMotion suspends the source virtual machine, copies the bitmap to the target ESX Server, and resumes the virtual machine on the target ESX Server. This entire process takes less than two seconds on a Gigabit Ethernet network.

38. VMware Horizon as integrated with vSphere receives a second request over the network at the server from the client computer to collaborate with the host computer. The vSphere back-end infrastructure, such as the View Connection Server and vCenter Server facilitate connection and re-connection requests by a Horizon client, including after a migration or failure situation, as seen below:

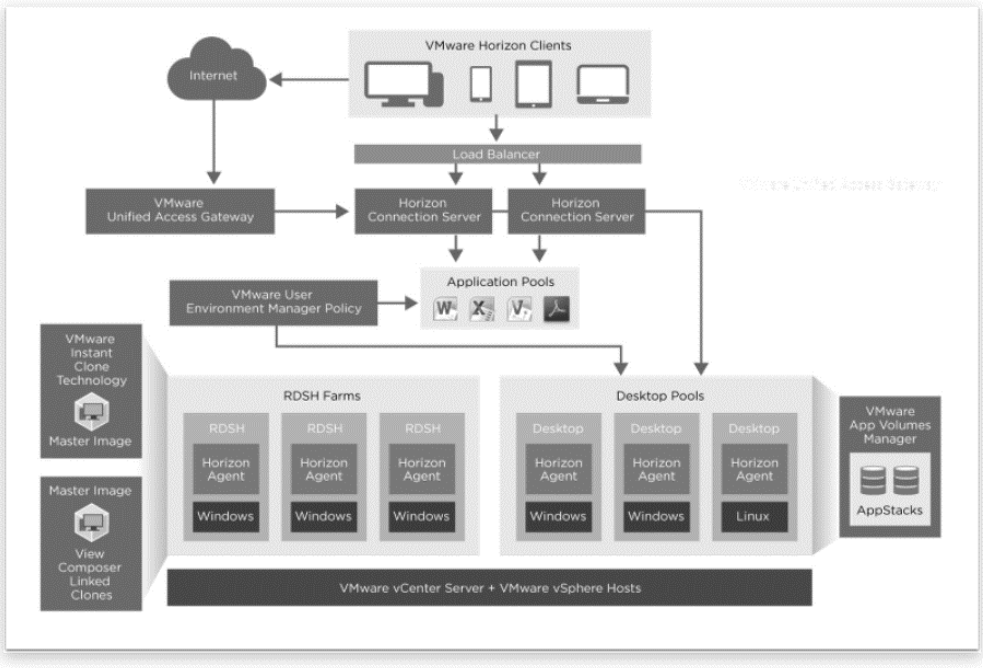
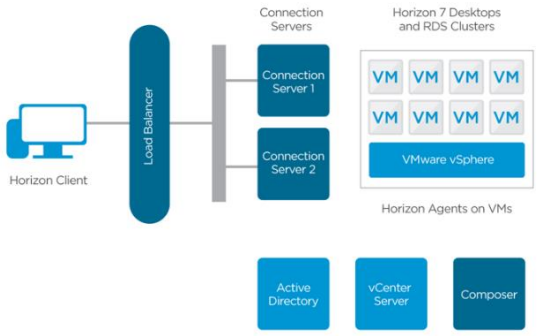


Figure: Horizon 7 Architecture Overview

39. VMware Horizon as integrated with vSphere facilitate by the server the collaboration of the host computer and the client computer over the network. The vSphere back-

end infrastructure, such as the View Connection Server and vCenter Server facilitate connection and re-connection requests by a Horizon client, which instantiates or re-instantiates the virtual machines, including the stored personal state information, as seen below:

Connection Servers broker client connections, authenticate users, and direct incoming requests to the correct endpoint. Although the Connection Server helps form the connection, it typically does not act as part of the data path after the connection is established.



View Connection Server

This software service acts as a broker for client connections. View Connection Server authenticates users through Windows Active Directory and directs the request to the appropriate virtual machine, physical or blade PC, or Windows Terminal Services server.

View Connection Server provides the following management capabilities:

- Authenticating users
- Entitling users to specific desktops and pools

Client endpoints communicate with a Connection Server or security server host over secure connections.

The initial client connection, which is used for user authentication and remote desktop and application selection, is created over HTTPS when a user provides a domain name to Horizon Client. If firewall and load balancing software are configured correctly in your network environment, this request reaches the Connection Server or security server host. With this connection, users are authenticated and a desktop or application is selected, but users have not yet connected to the remote desktop or application.

When users connect to remote desktops and applications, by default the client makes a second connection to the Connection Server or security server host. This connection is called the tunnel connection because it provides a secure tunnel for carrying RDP and other data over HTTPS.

When users connect to remote desktops and applications with the PCoIP display protocol, the client can make a further connection to the PCoIP Secure Gateway on the Connection Server or security server host. The PCoIP Secure Gateway ensures that only authenticated users can communicate with remote desktops and applications over PCoIP.

40. Additionally, VMware has been, and currently is, an active inducer of infringement of the ’217 patent under 35 U.S.C. § 271(b) and contributory infringement of the ’217 patent under 35 U.S.C. § 271(c) either literally and/or by the doctrine of equivalents.

41. VMware has actively induced, and continues to actively induce, infringement of the ’217 patent by intending that others use, offer for sale, or sell in the United States, products and/or services covered by one or more claims of the ’217 patent, including but not limited to vSphere, ESX/ESXi, Horizon/View, vCenter Server, Connection Server, vMotion and vSAN, and any VMware product and/or service, alone or in combination, that operates in materially the same manner. VMware provides these products and/or services to others, such as customers, resellers and end-user customers, who, in turn, use, provision for use, offer for sale, or sell in the United States products and/or services that directly infringe one or more claims of the ’217 patent.

42. VMware has contributed to, and continues to contribute to, the infringement of the ’217 patent by others by knowingly providing products and/or services that, when installed and configured result in a system as intended by VMware, directly infringes one or more claims of the ’217 patent.

43. VMware knew of the ’217 patent, or should have known of the ’217 patent, but was willfully blind to its existence. Upon information and belief, VMware has had actual knowledge of the ’217 patent since at least as early as the service upon VMware of this Complaint. By the time of trial, VMware will have known and intended (since receiving such notice) that its continued actions would infringe and actively induce and contribute to the infringement of one or more claims of the ’217 patent.

44. VMware has committed, and continues to commit, affirmative acts that cause infringement of one or more claims of the ’217 patent with knowledge of the ’217 patent and

knowledge or willful blindness that the induced acts constitute infringement of one or more claims of the ’217 patent. As an illustrative example only, VMware induces such acts of infringement by its affirmative actions of intentionally providing hardware and or software components that when used in their normal and customary way as desired and intended by VMware, infringe one or more claims of the ’217 patent and/or by directly or indirectly providing instructions on how to use its products and/or services in a manner or configuration that infringes one or more claims of the ’217 patent, including those found at one or more of the following:

- https://docs.vmware.com/en/VMware-Horizon-7/7.1/view-71-setting-up-virtual-desktops.pdf
- https://docs.vmware.com/en/VMware-Horizon-7/7.2/view-72-architecture-planning.pdf
- https://www.vmware.com/products/vsphere.html
- https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/whitepaper/vmware-vsphere-vmfs-best-practices-whitepaper.pdf
- https://blogs.vmware.com/virtualblocks/2015/05/29/20-common-vsan-questions/
- https://techzone.vmware.com/quick-start-tutorial-series-vmware-horizon-7

45. VMware has also committed, and continues to commit, contributory infringement by, *inter alia*, knowingly selling products and/or services that when used cause the direct infringement of one or more claims of the ’217 patent by a third party, and which have no substantial non-infringing uses, or include a separate and distinct component that is especially made or especially adapted for use in infringement of the ’217 patent and is not a staple article or commodity of commerce suitable for substantial non-infringing use.

46. As a result of VMware’s acts of infringement, Plaintiff has suffered and will continue to suffer damages in an amount to be proved at trial.

COUNT II
(VMWare's Infringement of U.S. Patent No. 9,686,378)

47. Paragraphs 1-46 are reincorporated by reference as if fully set forth herein.

48. The elements claimed by the '378 patent, taken alone or in combination, were not well-understood, routine or conventional to one of ordinary skill in the art at the time of the invention. Rather, the '378 patent claims and teaches, *inter alia*, an improved way to deliver components of server-based applications to remote client devices that provide interfaces to the applications at the client devices based on unique client characteristics, which were not present in the state of the art at the time of the invention. The invention improved upon existing server-based application delivery technology by enabling the dynamic creation of customized application interfaces adapted for compatibility with a variety of client platform types.

49. Instead of storing one or more static versions of server-based applications to which clients could be re-directed, the server-based applications of the present invention are customized based on client variables so that components of the applications implementing interfaces are delivered to a remote client in a manner optimized for that client's specific environment. For example, the present invention allows for application component(s) residing on a server to be invoked and transformed into an application interface presentable to clients, where said clients would otherwise be incapable of or have difficulty displaying such components. Thus, the resulting server-based application delivery systems and methods enabled a more flexible and dynamic approach relative to prior art systems that were limited to providing static, pre-configured application content.

50. The invention represented a technical solution to an unsolved technological problem. The written description of the '378 patent describes, in technical detail, each of the limitations in the claims, allowing a person of skill in the art to understand what those limitations

cover, and therefore what was claimed, and also understand how the non-conventional and non-generic ordered combination of the elements of the claims differ markedly from what had been conventional or generic in the industry at the time of the inventions of the '378 patent. More specifically, the claims of the '378 patent each recite receiving, by a server, an indication of an actuation of a user interface provided on a client device, wherein the client device is associated with client variables indicating an environment of the client device, and in response to receiving the indication, generating an updated version of the user interface for display in the environment of the client device by invoking a version of an application component that is configured for the environment of the client device, wherein the invoking of the application component processes the indication of the actuation, and packaging the updated version of the user interface for communication to the client device. Further embodiments claimed in the '378 patent include producing interim data in response to the invocation of the application component and using the interim data to create an updated version of the user interface for display in the environment of the client device.

51. The system covered by the asserted claims, therefore, differs markedly from the conventional and generic systems in use at the time of this invention, which *inter alia* lacked the claimed combination of using client variables to determine a client environment and invoking an application component of a server-based application from said variables and dynamically creating a customized application interface based on the client's specific environment. Embodiments of the present invention further include updating the customized application interface in response to a user actuation and using interim data to create the updated interface.

52. As described above, the '378 patent is drawn to solving a specific, technical problem arising in the context of server-based application delivery to remote clients. Consistent

with the problem addressed being rooted in such server-based application delivery, the '378 patent's solutions consequently are also rooted in that same technology and cannot be performed with pen and paper or in the human mind.

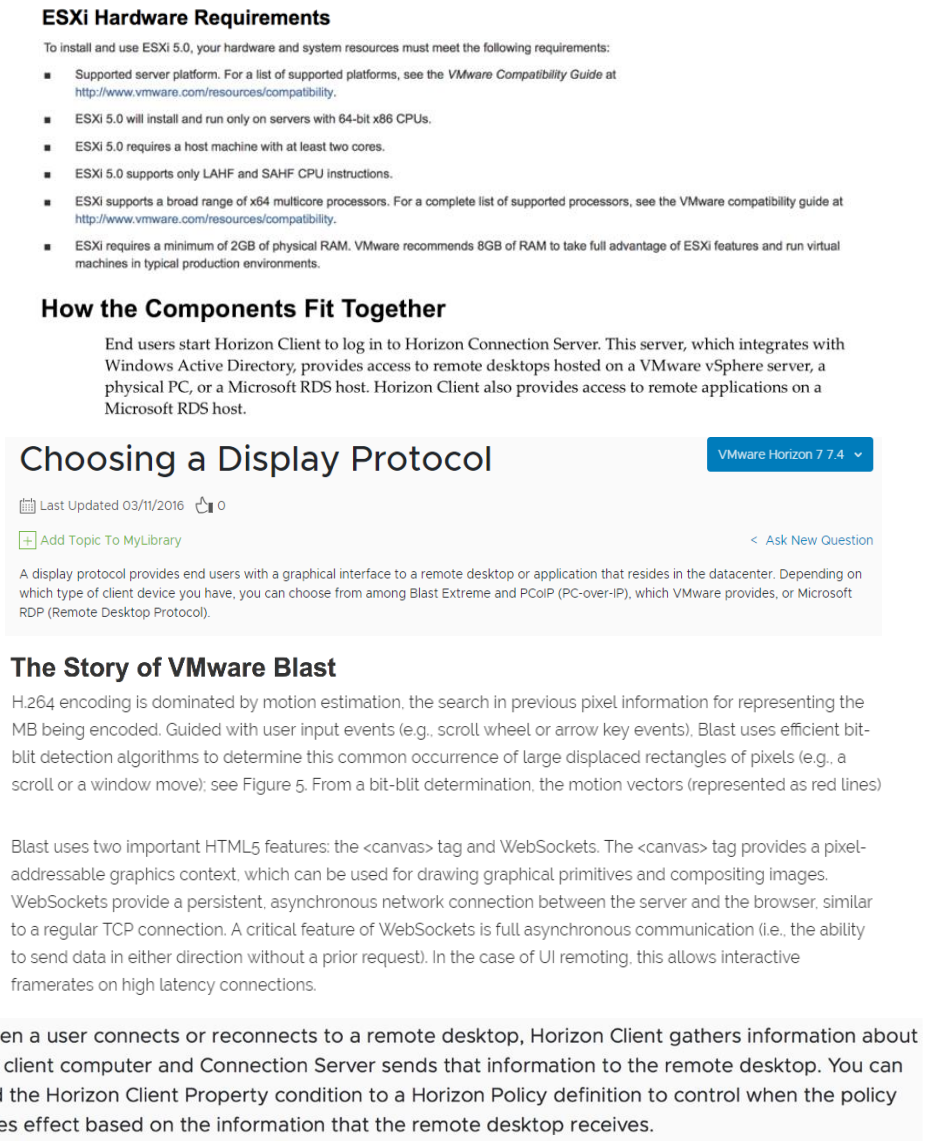
53. VMware has directly infringed, and continues to directly infringe, literally and/or by the doctrine of equivalents, individually and/or jointly, at least claims 1, 3, 4, 5, 6, 9, 10, 12, 13, 15, 16, 17, 18, 19, 20 of the '378 patent by making, using, testing, selling, offering for sale and/or importing into the United States products and/or services covered by one or more claims of the '378 patent. VMware's products and/or services that infringe the '378 patent include, but are not limited to, Horizon/View, Horizon Client/Horizon HTML 5 Client, vSphere, ESX/ESXi, vCenter Server, Connection Server, and any other VMware products and/or services, either alone or in combination, that operate in substantially the same manner.

54. Claim 1 of the '378 patent is reproduced below:

1. A method implemented by a server, the method comprising:
receiving, by the server from a client device, an indication of an actuation of a user interface provided on the client device, wherein the client device is associated with client variables defining an environment of the client device; and
in response to receiving the indication of the actuation:
invoking, by the server, a version of an application component that is configured for the environment of the client device, wherein the invoking of the application component processes the indication of the actuation and, in response, produces interim data representing results of the application component processing the indication of the actuation;
creating, by the server, based on the determined environment of the client device and using the interim data, an updated version of the user interface for display in the environment of the client device; and
packaging, by the server, the updated version of the user interface for communication to the client device.

55. As one non-limiting example, vSphere and ESX/ESXi in combination with VMware Horizon practice a method implemented by a server, comprising receiving, by the server from a client device, an indication of an actuation of a user interface provided by the client device, wherein the client device is associated with client variables defining an environment of the client device. For instance, when a Horizon Client interacts with the virtual machine or application

which it is using (e.g., scroll, keyboard input or touch gestures) the vSphere back-end infrastructure receives an indication of such interaction, which is specific to the client's environment, as seen below:



ESXi Hardware Requirements

To install and use ESXi 5.0, your hardware and system resources must meet the following requirements:

- Supported server platform. For a list of supported platforms, see the *VMware Compatibility Guide* at http://www.vmware.com/resources/compatibility.
- ESXi 5.0 will install and run only on servers with 64-bit x86 CPUs.
- ESXi 5.0 requires a host machine with at least two cores.
- ESXi 5.0 supports only LAHF and SAHF CPU instructions.
- ESXi supports a broad range of x64 multicore processors. For a complete list of supported processors, see the VMware compatibility guide at http://www.vmware.com/resources/compatibility.
- ESXi requires a minimum of 2GB of physical RAM. VMware recommends 8GB of RAM to take full advantage of ESXi features and run virtual machines in typical production environments.

How the Components Fit Together

End users start Horizon Client to log in to Horizon Connection Server. This server, which integrates with Windows Active Directory, provides access to remote desktops hosted on a VMware vSphere server, a physical PC, or a Microsoft RDS host. Horizon Client also provides access to remote applications on a Microsoft RDS host.

Choosing a Display Protocol

VMware Horizon 7 7.4

Last Updated 03/11/2016 0

Add Topic To MyLibrary < Ask New Question

A display protocol provides end users with a graphical interface to a remote desktop or application that resides in the datacenter. Depending on which type of client device you have, you can choose from among Blast Extreme and PCoIP (PC-over-IP), which VMware provides, or Microsoft RDP (Remote Desktop Protocol).

The Story of VMware Blast

H.264 encoding is dominated by motion estimation, the search in previous pixel information for representing the MB being encoded. Guided with user input events (e.g., scroll wheel or arrow key events), Blast uses efficient bit-blit detection algorithms to determine this common occurrence of large displaced rectangles of pixels (e.g., a scroll or a window move); see Figure 5. From a bit-blit determination, the motion vectors (represented as red lines)

Blast uses two important HTML5 features: the <canvas> tag and WebSockets. The <canvas> tag provides a pixel-addressable graphics context, which can be used for drawing graphical primitives and compositing images. WebSockets provide a persistent, asynchronous network connection between the server and the browser, similar to a regular TCP connection. A critical feature of WebSockets is full asynchronous communication (i.e., the ability to send data in either direction without a prior request). In the case of UI remoting, this allows interactive framerates on high latency connections.

When a user connects or reconnects to a remote desktop, Horizon Client gathers information about the client computer and Connection Server sends that information to the remote desktop. You can add the Horizon Client Property condition to a Horizon Policy definition to control when the policy takes effect based on the information that the remote desktop receives.

56. vSphere and ESX/ESXi in combination with VMware Horizon also practice the step of in response to receiving the indication of the actuation, invoking, by the server, a version of an application component that is configured for the environment of the client device, wherein the invoking of the application component processes the indication of the actuation and, in response, produces interim data representing results of the application component processing the

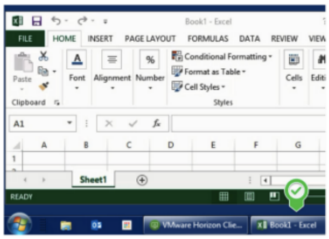
indication of the actuation. For example, actuation information can be sent to the vSphere back-end infrastructure via the Blast protocol using a WebSocket packet. The vSphere back-end then processes the actuation information and invokes the necessary components of the application that have been selected based on client variables. Pixel data representing an updated interface based on the actuation is then generated, as seen below:

Using View for Application Delivery

View also includes a feature called Unity Touch, which provides a gesture-based option for accessing Windows applications on iPads or other tablets. Because of small screen size, using mobile devices for navigating applications and files can be challenging. Unity Touch makes that easier. Unity Touch allows users to browse, search for, and open virtualized and hosted Windows applications, and switch between multiple open applications, without using the Windows Start menu or task bar. The file manager, task bar, and Start menus are also responsive to touch navigation. For more information, see Introducing Unity Touch.

The app appears and looks just like it would if it were a locally installed app.

In this example, the application icon for the published app Excel appears in the taskbar just as it would for a locally installed app.



This step described using the Windows-based client and seamless integration into the Windows user experience. If you install Horizon Client on other operating systems, such as a macOS or an Android tablet, the experience of using published apps is likewise integrated into those operating systems and their OS-specific features.

Using High Resolution Mode

You can enable High Resolution Mode from the Horizon Client Settings window. Tap **Settings** at the bottom of the Horizon Client window, tap **Resolution**, and tap to toggle the **High Resolution Mode** setting to on. To enable High Resolution Mode if you are using a remote desktop or application, tap to expand the Horizon Client Tools radial menu icon, tap the **Settings** (gear) icon, tap **Resolution**, and tap to toggle the **High Resolution Mode** setting to on.

The High Resolution Mode feature has the following requirements and limitations:

- You cannot use the High Resolution Mode feature for existing sessions. You must log out and log in to a new session for the feature to take effect.
- You must have an iPad Pro, or an iPad or iPad mini with Retina display, to use the High Resolution Mode feature.

VMware Blast is the VMware UI remoting technology in VMware Horizon. Blast uses standardized encoding schemes, including JPG/PNG and H.264 for pixel encoding, and Opus for audio. Unlike proprietary encoding schemes, these standard formats are supported natively, hence efficiently, in browsers and mobile devices.

Blast-JPG/PNG shipped in the Fall of 2013 in support of browser clients and in early 2015 in support of Linux virtual machines. Blast-H.264 shipped in March 2016 with Horizon 7, as Blast Extreme, with feature and performance parity with PCoIP. Much was written about Blast Extreme since. Here, we provide background and more in-depth technical details.

> UIs exhibit high frame coherence (i.e., a small portion of pixels varies from one frame to another). For example, text input, scrolling or pull-down menus all result with localized pixel changes. So, the large portions of the screen that remain unchanged do not need to be reexamined by the encoder. For a given frame, Blast rapidly determines changed regions and plumbs the resulting change-map information into the encoder. Typical H.264 encoders do not take advantage of this optimization, since in common video content most pixels change in every frame.

57. vSphere and ESX/ESXi in combination with VMware Horizon practice the step of creating, by the server, based on the determined environment of the client device and using the interim data, an updated version of the user interface for display in the environment of the client device. As noted above, Blast uses pixel encoding to generate an updated version of the user interface based on the actuation and client variables, which represents the updated view to be sent to the client, as seen below:

The Story of VMware Blast

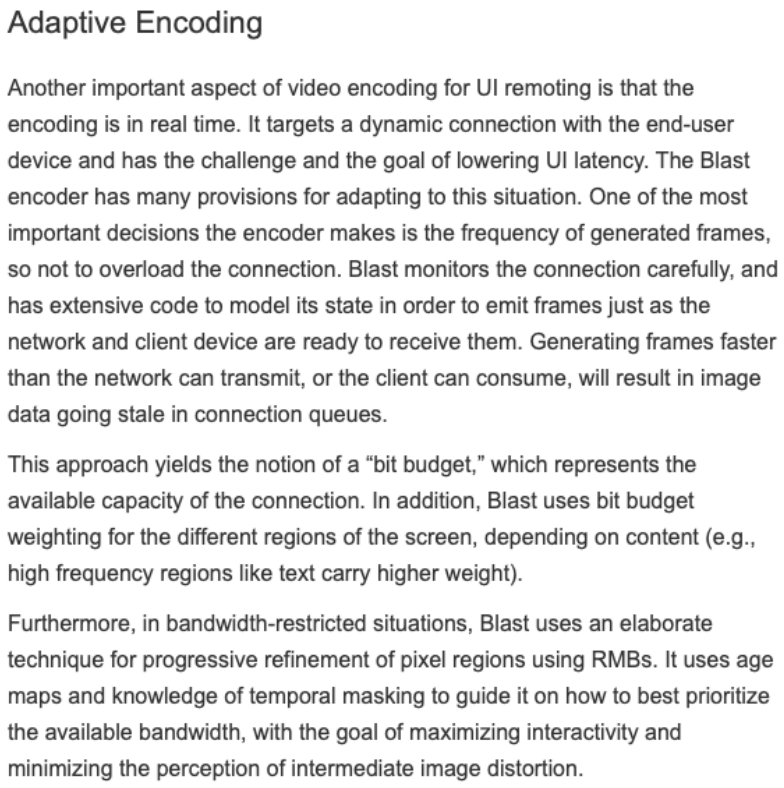
H.264 encoding is dominated by motion estimation, the search in previous pixel information for representing the MB being encoded. Guided with user input events (e.g., scroll wheel or arrow key events), Blast uses efficient bit-blit detection algorithms to determine this common occurrence of large displaced rectangles of pixels (e.g., a scroll or a window move); see Figure 5. From a bit-blit determination, the motion vectors (represented as red lines)

> UIs exhibit high frame coherence (i.e., a small portion of pixels varies from one frame to another). For example, text input, scrolling or pull-down menus all result with localized pixel changes. So, the large portions of the screen that remain unchanged do not need to be reexamined by the encoder. For a given frame, Blast rapidly determines changed regions and plumbs the resulting change-map information into the encoder. Typical H.264 encoders do not take advantage of this optimization, since in common video content most pixels change in every frame.

Figure 5. Moving the calculator is detected as a bit-blit, so its MBs are encoded efficiently based on the previous frame. The red lines are highly correlated and represent the motion vectors referencing previous pixel regions.

58. vSphere and ESX/ESXi in combination with VMware Horizon practice the step of packaging, by the server, the updated version of the user interface for communication to the client device. For example, the updated pixel data is encoded and transmitted to the client via the network, as seen below:

Figure 5. Moving the calculator is detected as a bit-blit, so its MBs are encoded efficiently based on the previous frame. The red lines are highly correlated and represent the motion vectors referencing previous pixel regions.



Adaptive Encoding

Another important aspect of video encoding for UI remoting is that the encoding is in real time. It targets a dynamic connection with the end-user device and has the challenge and the goal of lowering UI latency. The Blast encoder has many provisions for adapting to this situation. One of the most important decisions the encoder makes is the frequency of generated frames, so not to overload the connection. Blast monitors the connection carefully, and has extensive code to model its state in order to emit frames just as the network and client device are ready to receive them. Generating frames faster than the network can transmit, or the client can consume, will result in image data going stale in connection queues.

This approach yields the notion of a "bit budget," which represents the available capacity of the connection. In addition, Blast uses bit budget weighting for the different regions of the screen, depending on content (e.g., high frequency regions like text carry higher weight).

Furthermore, in bandwidth-restricted situations, Blast uses an elaborate technique for progressive refinement of pixel regions using RMBs. It uses age maps and knowledge of temporal masking to guide it on how to best prioritize the available bandwidth, with the goal of maximizing interactivity and minimizing the perception of intermediate image distortion.

59. Additionally, VMware has been, and currently is, an active inducer of infringement of the '378 patent under 35 U.S.C. § 271(b) and contributory infringement of the '378 patent under 35 U.S.C. § 271(c) either literally and/or by the doctrine of equivalents.

60. VMware has actively induced, and continues to actively induce, infringement of the '378 patent by intending that others use, offer for sale, or sell in the United States, products and/or services covered by one or more claims of the '378 patent, including but not limited to Horizon/ Horizon View, Horizon HTML 5 Client, vSphere, ESX/ESXi, vCenter Server, Connection Server, and any VMware product and/or service, alone or in combination, that operates in materially the same manner. VMware provides these products and/or services to others, such as customers, resellers and end-user customers, who, in turn, use, provision for use, offer for sale, or sell in the United States products and/or services that directly infringe one or more claims of the '378 patent.

61. VMware has contributed to, and continues to contribute to, the infringement of the ’378 patent by others by knowingly providing products and/or services that, when installed and configured result in a system as intended by VMware, directly infringes one or more claims of the ’378 patent.

62. VMware knew of the ’378 patent, or should have known of the ’378 patent, but was willfully blind to its existence. Upon information and belief, VMware has had actual knowledge of the ’378 patent since at least as early as the service upon VMware of this Complaint. By the time of trial, VMware will have known and intended (since receiving such notice) that its continued actions would infringe and actively induce and contribute to the infringement of one or more claims of the ’378 patent.

63. VMware has committed, and continues to commit, affirmative acts that cause infringement of one or more claims of the ’378 patent with knowledge of the ’378 patent, and knowledge or willful blindness that the induced acts constitute infringement of one or more claims of the ’378 patent. As an illustrative example only, VMware induces such acts of infringement by its affirmative actions of intentionally providing hardware and or software components that when used in their normal and customary way as desired and intended by VMware, infringe one or more claims of the ’378 patent and/or by directly or indirectly providing instructions on how to use its products and/or services in a manner or configuration that infringes one or more claims of the ’378 patent, including those found at one or more of the following:

- https://docs.vmware.com/en/VMware-vSphere/index.html
- https://my.vmware.com/web/vmware/info?slug=desktop_end_user_computing/vmware_horizon_clients/4_0

- https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/products/horizon/vmware-horizon-7-datasheet.pdf
- https://docs.vmware.com/en/VMware-Horizon-7/7.0/com.vmware.horizon-view.planning.doc/GUID-5C5C9E6B-263E-4DA0-895F-4F1415F37947.html
- https://docs.vmware.com/en/VMware-Horizon-7/7.1/view-71-configuring-remote-features.pdf
- https://docs.vmware.com/en/VMware-Horizon-7/7.4/horizon-architecture-planning/GUID-CFAABEB9-9CF2-4098-A01D-1CA118D4B6BD.html
- https://blogs.vmware.com/euc/2016/09/story-of-vmware-blast.html
- https://docs.vmware.com/en/VMware-Horizon-7/7.1/com.vmware.horizon.remote.features.doc/GUID-F23C9960-3399-4FBE-9910-393D8E4A91F8.html
- https://docs.vmware.com/en/VMware-Horizon-Client-for-iOS/4.6/com.vmware.horizon.ios-client-46-user-guide.doc/GUID-926E5CA8-6C40-45B4-AB31-6579F3906CDA.html

64. VMware has also committed, and continues to commit, contributory infringement by, *inter alia*, knowingly selling products and/or services that when used cause the direct infringement of one or more claims of the ’378 patent by a third party, and which have no substantial non-infringing uses, or include a separate and distinct component that is especially made or especially adapted for use in infringement of the ’378 patent and is not a staple article or commodity of commerce suitable for substantial non-infringing use.

65. As a result of VMware’s acts of infringement, Plaintiff has suffered and will continue to suffer damages in an amount to be proved at trial.

COUNT III
(VMWare's Infringement of U.S. Patent No. 9,092,546)

66. Paragraphs 1-65 are reincorporated by reference as if fully set forth herein.

67. The elements claimed by the '546 patent, taken alone or in combination, were not well-understood, routine or conventional to one of ordinary skill in the art at the time of the invention. Rather, the '546 patent claims and teaches, *inter alia*, an improved way to invoke and deliver components of server-based applications to remote client devices that provide interfaces to the applications at the client devices based on unique client characteristics, which were not present in the state of the art at the time of the invention. The invention improved upon existing server-based application delivery technology by enabling the dynamic creation of customized application interfaces adapted for compatibility with a variety of client platform types.

68. Instead of storing one or more static versions of server-based applications to which clients could be re-directed, the server-based applications of the present invention are customized based on client variables so that components of the applications implementing interfaces are delivered to a remote client in a manner optimized for that client's specific environment. For example, the present invention allows for application component(s) residing on a server to be invoked and transformed into an application interface presentable to clients, where said clients would otherwise be incapable of or have difficulty displaying such components. Thus, the resulting server-based application delivery systems and methods enabled a more flexible and dynamic approach relative to prior art systems that were limited to providing static, pre-configured application content.

69. The invention represented a technical solution to an unsolved technological problem. The written description of the '546 patent describes, in technical detail, each of the limitations in the claims, allowing a person of skill in the art to understand what those limitations

cover, and therefore what was claimed, and also understand how the non-conventional and non-generic ordered combination of the elements of the claims differ markedly from what had been conventional or generic in the industry at the time of the inventions of the '546 patent. More specifically, the claims of the '546 patent each recite receiving at a server a request from a client device that identifies an application and one or more client variables. In response, the server obtains an application component configured to operate in a particular client environment based on the client variable(s), invokes the application component resulting in interim data, and creates an application interface tailored to the client environment using the interim data. In certain claimed embodiments, the created application interface is further packaged and/or transmitted to the remote client for delivery.

70. The system covered by the asserted claims, therefore, differs markedly from the conventional and generic systems in use at the time of this invention, which *inter alia* lacked the claimed combination of using client variables to determine a client environment and invoking an application component of a server-based application from said variables and dynamically creating a customized application interface based on the client's specific environment.

71. As described above, the '546 patent is drawn to solving a specific, technical problem arising in the context of server-based application delivery to remote clients. Consistent with the problem addressed being rooted in such server-based application delivery, the '546 patent's solutions consequently are also rooted in that same technology and cannot be performed with pen and paper or in the human mind.

72. VMware has directly infringed, and continues to directly infringe, literally and/or by the doctrine of equivalents, individually and/or jointly, at least claims 1, 4-7, 9-11, 13-18, 19 and 21 of the '546 patent by making, using, testing, selling, offering for sale and/or importing

into the United States products and/or services covered by one or more claims of the ’546 patent. VMware’s products and/or services that infringe the ’546 patent include, but are not limited to, Horizon View, vSphere, ESX/ESXi, vCenter Server, Connection Server, and any other VMware products and/or services, either alone or in combination, that operate in substantially the same manner.

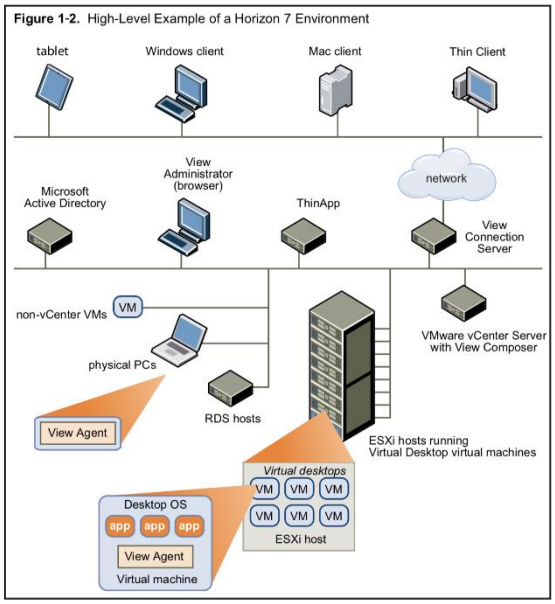
73. Claim 1 of the ’546 patent is reproduced below:

1. A method implemented by a server, the method comprising:
receiving, by the server, a request for an application from a client device, the request identifying one or more client variables that describe the client device;
in response to receiving the request, obtaining a version of an application component of the application, the application component selected from among a plurality of available versions of the application, the versions being configured to operate in different environments, the obtained version of the application component being selected for the client device based on at least the one of the client variables;
determining, by the server, a destination environment of the client device, the determining based on at least one of the client variables;
invoking, by the server, the obtained version of the application component;
creating, by the server, based on the determined destination environment and using interim data resulting from the server invoking the obtained version of the application component, an application interface for the application component; and
packaging, by the server, the application interface for communication to the client device.

74. As one non-limiting example, vSphere and ESX/ESXi in combination with VMware Horizon comprises a method, implemented by a server as claimed in the ’546 patent. For example, vSphere in conjunction with ESXi provides hosted virtual machine(s) (“VMs”) for creating and delivering server-based application functionality to remote users via the Horizon platform, as is explained further below:

Horizon 7: Delivering Desktops and Applications as a Service

Horizon 7 enables IT to centrally manage images to streamline management, reduce costs, and maintain compliance. With Horizon 7, virtualized or hosted desktops and applications can be delivered through a single platform to end users. These desktop and application services—including RDS hosted apps, packaged apps with ThinApp, SaaS apps, and even virtualized apps from Citrix—can all be accessed from one unified workspace to provide end users with all of the resources they want, at the speed they expect, with the effi ciency business demands.



VMware vSphere is VMware's virtualization platform, which transforms data centers into aggregated computing infrastructures that include CPU, storage, and networking resources. vSphere manages these infrastructures as a unified operating environment, and provides you with the tools to administer the data centers that participate in that environment.

The two core components of vSphere are ESXi and vCenter Server. ESXi is the virtualization platform where you create and run virtual machines and virtual appliances. vCenter Server is the service through which you manage multiple hosts connected in a network and pool host resources.

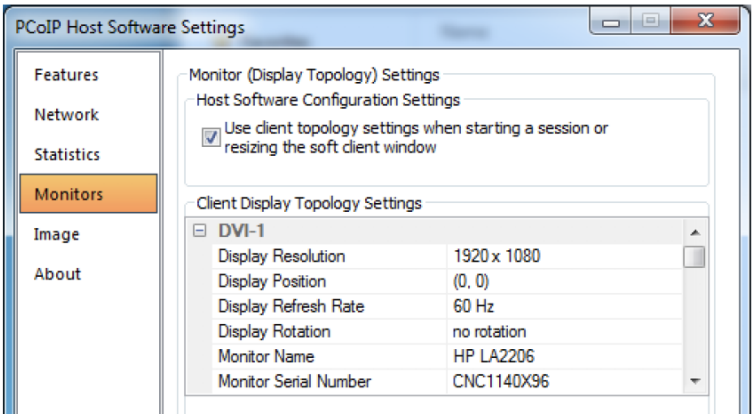
75. vSphere and ESX/ESXi in combination with VMware Horizon receive, by the server, a request for an application from a client device, the request identifying one or more client variables that describe the client device. For instance, using Horizon Client, a user is presented with various applications and desktops which are remotely hosted on VMware's vSphere

platform. When a hosted application or desktop is selected via Horizon Client, the remote vSphere platform receives a request which includes information relating to the client machine, as shown below:

6. To launch an application or desktop, double-click the icon for the application or desktop.

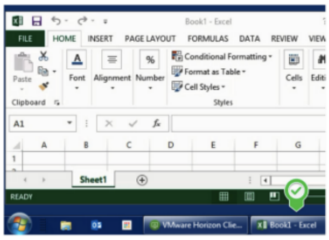
Deliver a Better Desktop Experience

Horizon View delivers the best end-user experience across locations and devices. Horizon View with PCoIP adapts to the end user's network connection to provide a high-quality customized desktop experience over the LAN and WAN. Users can flexibly connect to their Horizon View desktop from a variety of devices, including desktops, thin or zero clients and mobile devices. Horizon View supports 3D graphics rendering, unified communications, multi-monitor configurations, and audio and video content with multi-media redirection (MMR). The results are a rich, seamless user experience and maximum productivity for all types of end users.



The app appears and looks just like it would if it were a locally installed app.

In this example, the application icon for the published app Excel appears in the taskbar just as it would for a locally installed app.



This step described using the Windows-based client and seamless integration into the Windows user experience. If you install Horizon Client on other operating systems, such as a macOS or an Android tablet, the experience of using published apps is likewise integrated into those operating systems and their OS-specific features.

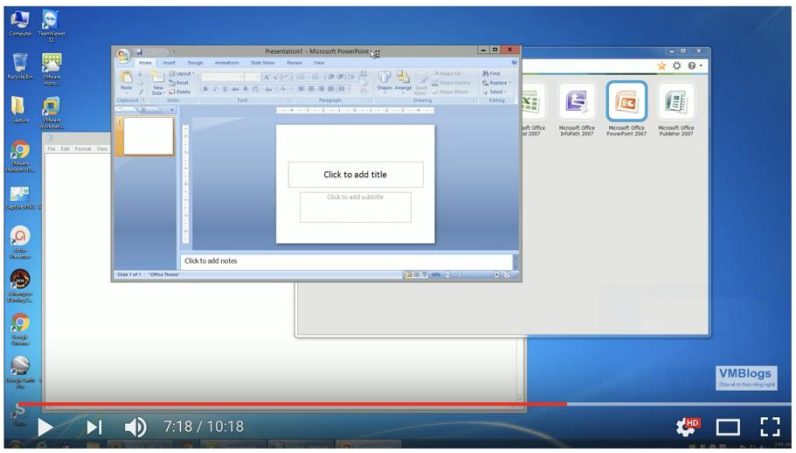
76. vSphere and ESX/ESXi in combination with VMware Horizon, in response to the receiving the request, obtain a version of an application component of the application, the application component selected from among a plurality of available versions of the application, the versions being configured to operate in different environments, the obtained version of the application component being selected for the client device based on at least one of the client variables. When an application or desktop is selected via the Horizon Client, Horizon passes client variables to the vSphere environment that hosts the selected application/desktop. Based on these variables, a version of the application/desktop is selected for the specific client, for example, Windows version supporting HD display and touch capable, as shown below:

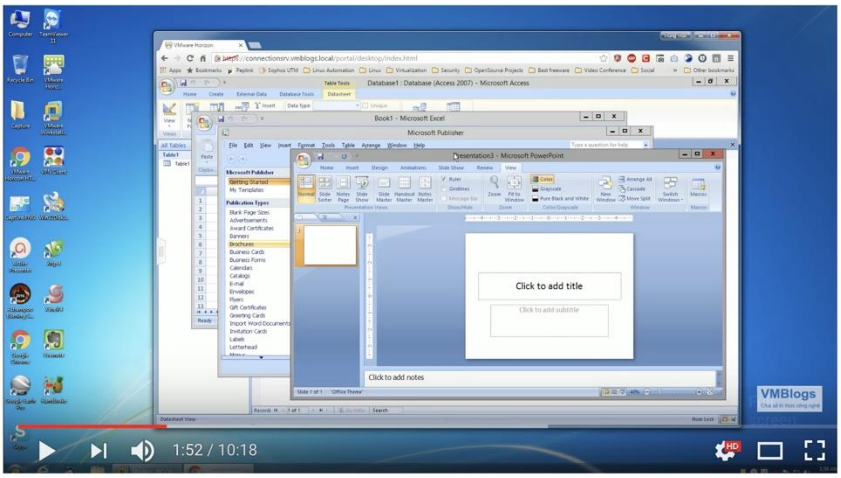
1.1 VMware User Environment Manager

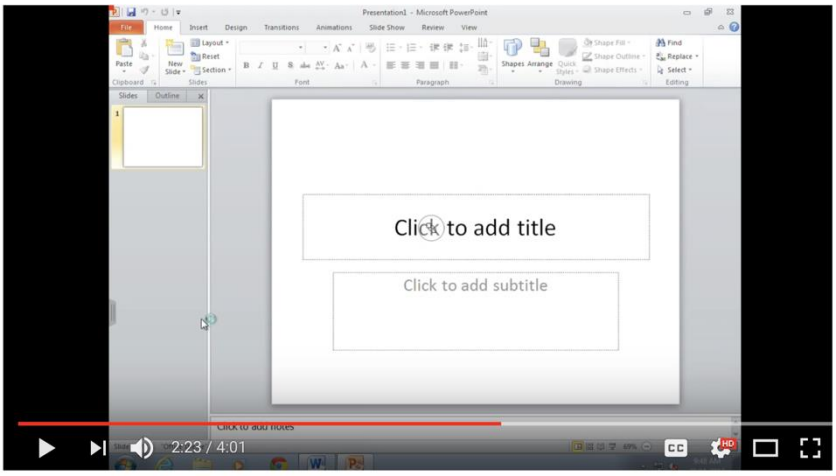
VMware User Environment Manager offers a desktop that adjusts to the actual situation of the end user, providing access to the IT resources that are required, based on a user's role, device and location. Many organizations suffer from hidden productivity loss as a result of ad hoc activities like manually mapping network drives and printers or providing application shortcuts to end users. This so-called distortion not only impacts IT departments but also affects end users. The relevant user experience that User Environment Manager offers, significantly eliminates this distortion.

Using View for Application Delivery

View also includes a feature called Unity Touch, which provides a gesture-based option for accessing Windows applications on iPads or other tablets. Because of small screen size, using mobile devices for navigating applications and files can be challenging. Unity Touch makes that easier. Unity Touch allows users to browse, search for, and open virtualized and hosted Windows applications, and switch between multiple open applications, without using the Windows Start menu or task bar. The file manager, task bar, and Start menus are also responsive to touch navigation. For more information, see Introducing Unity Touch.





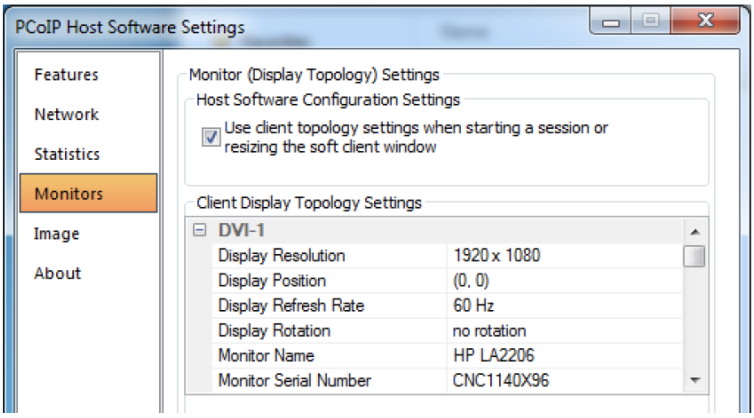


77. vSphere and ESX/ESXi in combination with VMware Horizon determine, by the server, a destination environment of the client device, the determining based on at least one of the client variables. The vSphere platform uses the information about the client environment, such

as, client operating system, display capabilities, network capabilities or device type in order to determine the specific client environment, as shown below:

Deliver a Better Desktop Experience

Horizon View delivers the best end-user experience across locations and devices. Horizon View with PCoIP adapts to the end user's network connection to provide a high-quality customized desktop experience over the LAN and WAN. Users can flexibly connect to their Horizon View desktop from a variety of devices, including desktops, thin or zero clients and mobile devices. Horizon View supports 3D graphics rendering, unified communications, multi-monitor configurations, and audio and video content with multi-media redirection (MMR). The results are a rich, seamless user experience and maximum productivity for all types of end users.



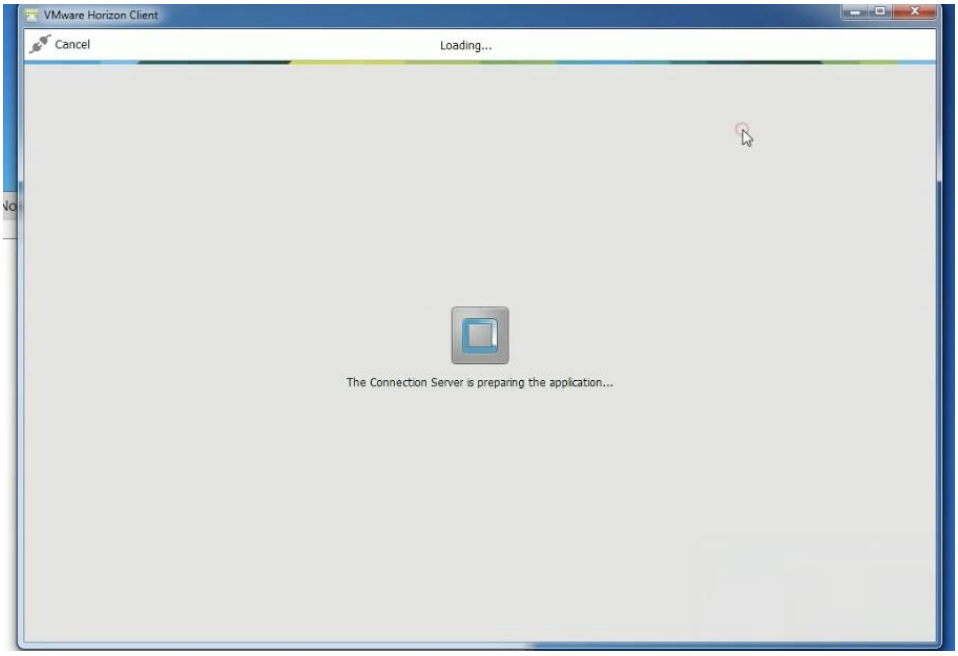
1.1 VMware User Environment Manager

VMware User Environment Manager offers a desktop that adjusts to the actual situation of the end user, providing access to the IT resources that are required, based on a user's role, device and location. Many organizations suffer from hidden productivity loss as a result of ad hoc activities like manually mapping network drives and printers or providing application shortcuts to end users. This so-called distortion not only impacts IT departments but also affects end users. The relevant user experience that User Environment Manager offers, significantly eliminates this distortion.

78. vSphere and ESX/ESXi in combination with VMware Horizon invoke, by the server, the obtained version of the application component. When the client environment has been determined, the vSphere platform executes the application on the remote host, as seen below:

A virtual machine is a software computer that, like a physical computer, runs an operating system and applications. The virtual machine consists of a set of specification and configuration files and is backed by the physical resources of a host. Every virtual machine has virtual devices that provide the same functionality as physical hardware but are more portable, more secure, and easier to manage.

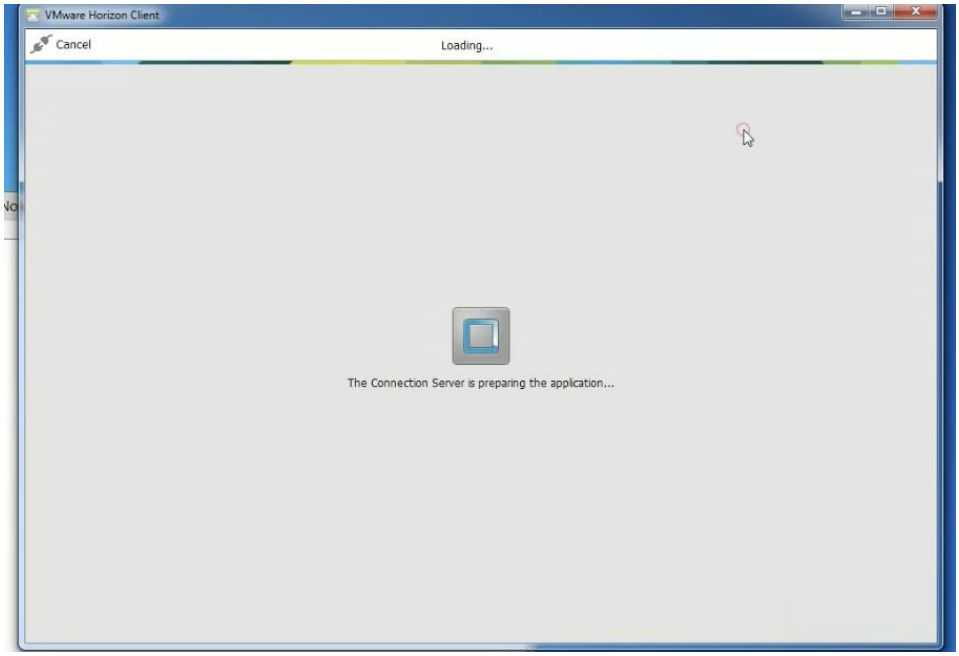
A multitude of application-delivery options have been developed in response to this growing need, as well as new terms to talk about those options. For the purposes of this paper, *software application delivery*—also called *app delivery*—refers to any method used by IT administrators to make applications available to their end users. For this discussion, app delivery includes methods such as streaming, where the application is not actually placed on the endpoint device at all.

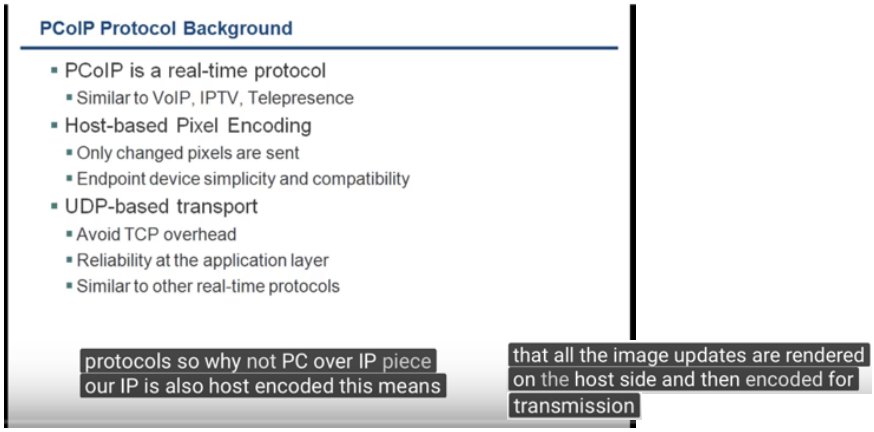


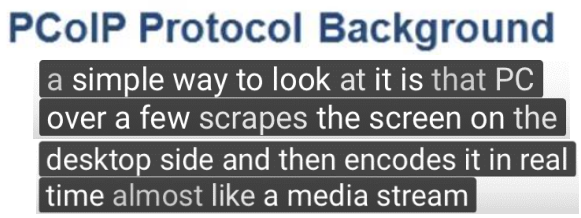
79. vSphere and ESX/ESXi in combination with VMware Horizon create, by the server, based on the determined destination environment and using interim data resulting from the server invoking the obtained version of the application component, an application interface for the application component. For example, based on the determined environment and invoked application component, the vSphere platform creates an application interface by encoding a binary set of pixels resulting from said invocation, as illustrated below:

Deliver a Better Desktop Experience

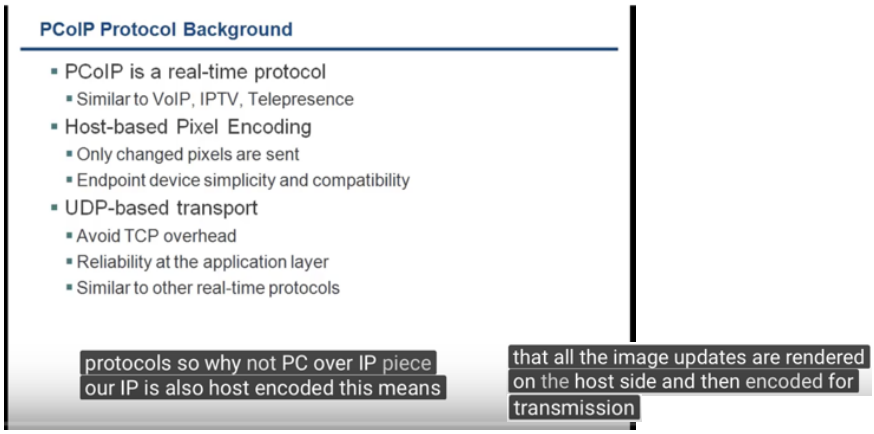
Horizon View delivers the best end-user experience across locations and devices. Horizon View with PCoIP adapts to the end user's network connection to provide a high-quality customized desktop experience over the LAN and WAN. Users can flexibly connect to their Horizon View desktop from a variety of devices, including desktops, thin or zero clients and mobile devices. Horizon View supports 3D graphics rendering, unified communications, multi-monitor configurations, and audio and video content with multi-media redirection (MMR). The results are a rich, seamless user experience and maximum productivity for all types of end users.

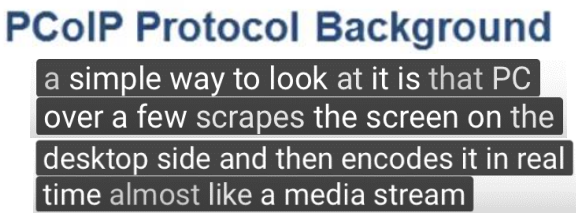


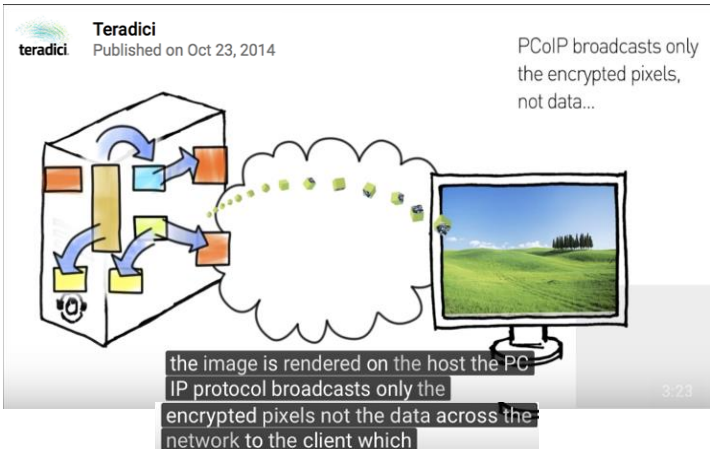




80. vSphere and ESX/ESXi in combination with VMware Horizon packages, by the server, the application interface for communication to the client device. The encoded set of pixels are sent to the client device over a network where it can be decoded and presented on the client device in human-viewable form, as illustrated below:







81. Additionally, VMware has been, and currently is, an active inducer of infringement of the '546 patent under 35 U.S.C. § 271(b) and contributory infringement of the '546 patent under 35 U.S.C. § 271(c) either literally and/or by the doctrine of equivalents.

82. VMware has actively induced, and continues to actively induce, infringement of the '546 patent by intending that others use, offer for sale, or sell in the United States, products and/or services covered by one or more claims of the '546 patent, including but not limited to Horizon View, vSphere, ESX/ESXi, vCenter Server, Connection Server, and any VMware

product and/or service, alone or in combination, that operates in materially the same manner. VMware provides these products and/or services to others, such as customers, resellers and end-user customers, who, in turn, use, provision for use, offer for sale, or sell in the United States products and/or services that directly infringe one or more claims of the '546 patent.

83. VMware has contributed to, and continues to contribute to, the infringement of the '546 patent by others by knowingly providing products and/or services that, when installed and configured result in a system as intended by VMware, directly infringes one or more claims of the '546 patent.

84. VMware knew of the '546 patent, or should have known of the '546 patent, but was willfully blind to its existence. Upon information and belief, VMware has had actual knowledge of the '546 patent since at least as early as the service upon VMware of this Complaint. By the time of trial, VMware will have known and intended (since receiving such notice) that its continued actions would infringe and actively induce and contribute to the infringement of one or more claims of the '546 patent.

85. VMware has committed, and continues to commit, affirmative acts that cause infringement of one or more claims of the '546 patent with knowledge of the '546 patent and knowledge or willful blindness that the induced acts constitute infringement of one or more claims of the '546 patent. As an illustrative example only, VMware induces such acts of infringement by its affirmative actions of intentionally providing hardware and/or software components that when used in their normal and customary way as desired and intended by VMware, infringe one or more claims of the '546 patent and/or by directly or indirectly providing instructions on how to use its products and/or services in a manner or configuration that infringes one or more claims of the '546 patent, including those found at one or more of the following:

- https://docs.vmware.com/en/VMware-vSphere/index.html
- https://my.vmware.com/web/vmware/info?slug=desktop_end_user_computing/vmware_horizon_clients/4_0
- https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/products/horizon/vmware-horizon-7-datasheet.pdf
- https://docs.vmware.com/en/VMware-Horizon-7/7.0/com.vmware.horizon-view.planning.doc/GUID-5C5C9E6B-263E-4DA0-895F-4F1415F37947.html
- https://docs.vmware.com/en/VMware-Horizon-7/7.1/view-71-configuring-remote-features.pdf
- https://www.youtube.com/watch?v=6ddJGbxpwxc
- https://www.youtube.com/watch?v=KQXB6Y_db28

86. VMware has also committed, and continues to commit, contributory infringement by, *inter alia*, knowingly selling products and/or services that when used cause the direct infringement of one or more claims of the ’546 patent by a third party, and which have no substantial non-infringing uses, or include a separate and distinct component that is especially made or especially adapted for use in infringement of the ’546 patent and is not a staple article or commodity of commerce suitable for substantial non-infringing use.

87. As a result of VMware’s acts of infringement, Plaintiff has suffered and will continue to suffer damages in an amount to be proved at trial.

COUNT IV
(VMWare’s Infringement of U.S. Patent No. 7,016,963)

88. Paragraphs 1-87 are reincorporated by reference as if fully set forth herein.

89. The elements claimed by the ’963 patent, taken alone or in combination, were not well-understood, routine or conventional to one of ordinary skill in the art at the time of the

invention. Rather, the '963 patent claims and teaches, *inter alia*, an improved way to invoke, deliver, and transform interfaces to server-based applications that are delivered to remote client devices based on unique client characteristics and real-time user inputs received by the client, which were not present in the state of the art at the time of the invention. The invention improved upon existing server-based application delivery technology by enabling the dynamic creation of customized application interfaces adapted for compatibility with a variety of client platform types.

90. Instead of storing one or more static versions of server-based applications to which clients could be re-directed, the server-based applications of the present invention are customized based on client variables so that components of the applications implementing interfaces are delivered to a remote client based on real-time user input received by the client and in a manner optimized for that client's specific environment. For example, the present invention allows for application component(s) residing on a server to be invoked and transformed into an application interface presentable to clients, where said clients would otherwise be incapable of or have difficulty displaying such components. Thus, the resulting server-based application delivery systems and methods enable a more flexible and dynamic approach relative to prior art systems that were limited to providing static, pre-configured application content.

91. The invention represented a technical solution to an unsolved technological problem. The written description of the '963 patent describes, in technical detail, each of the limitations in the claims, allowing a person of skill in the art to understand what those limitations cover, and therefore what was claimed, and also understand how the non-conventional and non-generic ordered combination of the elements of the claims differ markedly from what had been conventional or generic in the industry at the time of the inventions of the '963 patent. More specifically, the claims of the '963 patent each recite receiving an invocation request from a client

device, the invocation request identifying the client device and a server-based application, selecting an application interface particular to the client device, providing rendering information for the selected application interface to the client device, invoking the server-based application, receiving an action request from the client device, interpreting the action request to identify changes required to the application interface, providing rendering for the changes to the application to the client device, and invoking a server-based application command in accordance with the action request.

92. The system covered by the asserted claims, therefore, differs markedly from the conventional and generic systems in use at the time of this invention, which *inter alia* lacked the claimed combination of receiving an invocation request identifying a client device and server-based application, selecting an application interface particular to the client device, providing rendering information for the interface, invoking the server-based application and receiving an action request from the client device. Further the prior art additionally lacked interpreting the action request from the client device to identify changes to the interface, providing rendering for the changes to the interface and invoking the server-based application in accordance with the action request.

93. As described above, the ’963 patent is drawn to solving a specific, technical problem arising in the context of server-based application delivery to remote clients. Consistent with the problem addressed being rooted in such server-based application delivery, the ’963 patent’s solutions consequently are also rooted in that same technology and cannot be performed with pen and paper or in the human mind.

94. VMware has directly infringed, and continues to directly infringe, literally and/or by the doctrine of equivalents, individually and/or jointly, at least claim 2 of the ’963 patent by

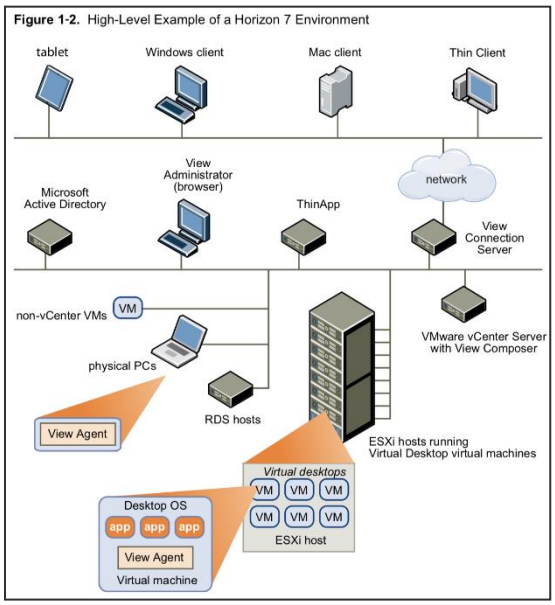
making, using, testing, selling, offering for sale and/or importing into the United States products and/or services covered by one or more claims of the ’963 patent. VMware’s products and/or services that infringe the ‘963 patent include, but are not limited to, Horizon View, vSphere, ESX/ESXi, vCenter Server, Connection Server, and any other VMware products and/or services, either alone or in combination, that operate in substantially the same manner.

95. Claim 2 of the ’963 patent is reproduced below:

2. A method for allowing various platforms to execute server-based applications through a network, the method comprising the steps of:
receiving an invocation request from a client device, the invocation request identifying the client device and a server-based application;
selecting an application interface particular to the client device;
providing rendering information for the selected application interface to the client device;
invoking the server-based application;
receiving an action request from the client device;
interpreting the action request to identify changes required to the application interface;
providing rendering for the changes to the application to the client device; and
invoking a server-based application command in accordance with the action request.

96. As one non-limiting example, vSphere and ESX/ESXi in combination with VMware Horizon practice a method for allowing various platforms to execute server-based applications through a network. For example, vSphere in conjunction with ESXi provide hosted virtual machine(s) (“VMs”) for creating and delivering server-based application functionality to remote users on a variety of client devices via the Horizon platform, as is explained further below:

> Horizon 7: Delivering Desktops and Applications as a Service
> Horizon 7 enables IT to centrally manage images to streamline management, reduce costs, and maintain compliance. With Horizon 7, virtualized or hosted desktops and applications can be delivered through a single platform to end users. These desktop and application services—including RDS hosted apps, packaged apps with ThinApp, SaaS apps, and even virtualized apps from Citrix—can all be accessed from one unified workspace to provide end users with all of the resources they want, at the speed they expect, with the effi ciency business demands.



VMware vSphere is VMware's virtualization platform, which transforms data centers into aggregated computing infrastructures that include CPU, storage, and networking resources. vSphere manages these infrastructures as a unified operating environment, and provides you with the tools to administer the data centers that participate in that environment.

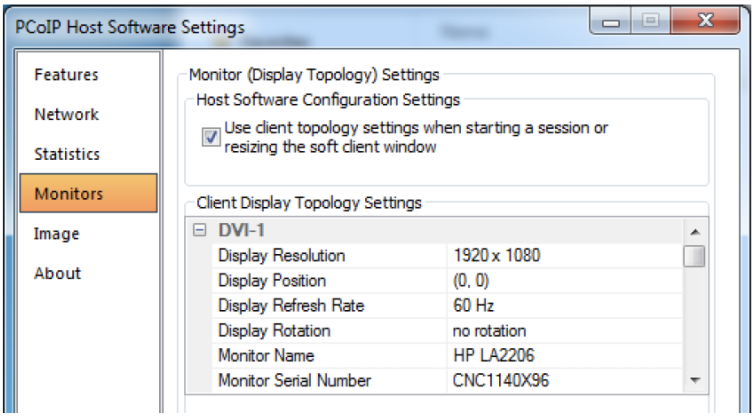
The two core components of vSphere are ESXi and vCenter Server. ESXi is the virtualization platform where you create and run virtual machines and virtual appliances. vCenter Server is the service through which you manage multiple hosts connected in a network and pool host resources.

97. vSphere and ESX/ESXi in combination with VMware Horizon practice the step of receiving an invocation request from a client device, the invocation request identifying the client device and a server-based application. For instance, using Horizon Client, a user is presented with various applications and desktops which are remotely hosted on VMware's vSphere platform. When a hosted application or desktop is selected via Horizon Client, the remote vSphere platform receives a request which includes information relating to the client machine and selected application, as shown below:

6. To launch an application or desktop, double-click the icon for the application or desktop.

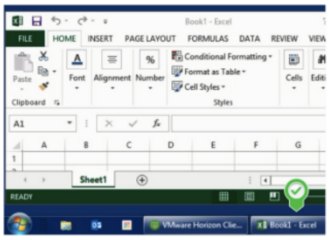
Deliver a Better Desktop Experience

Horizon View delivers the best end-user experience across locations and devices. Horizon View with PCoIP adapts to the end user's network connection to provide a high-quality customized desktop experience over the LAN and WAN. Users can flexibly connect to their Horizon View desktop from a variety of devices, including desktops, thin or zero clients and mobile devices. Horizon View supports 3D graphics rendering, unified communications, multi-monitor configurations, and audio and video content with multi-media redirection (MMR). The results are a rich, seamless user experience and maximum productivity for all types of end users.



The app appears and looks just like it would if it were a locally installed app.

In this example, the application icon for the published app Excel appears in the taskbar just as it would for a locally installed app.



This step described using the Windows-based client and seamless integration into the Windows user experience. If you install Horizon Client on other operating systems, such as a macOS or an Android tablet, the experience of using published apps is likewise integrated into those operating systems and their OS-specific features.

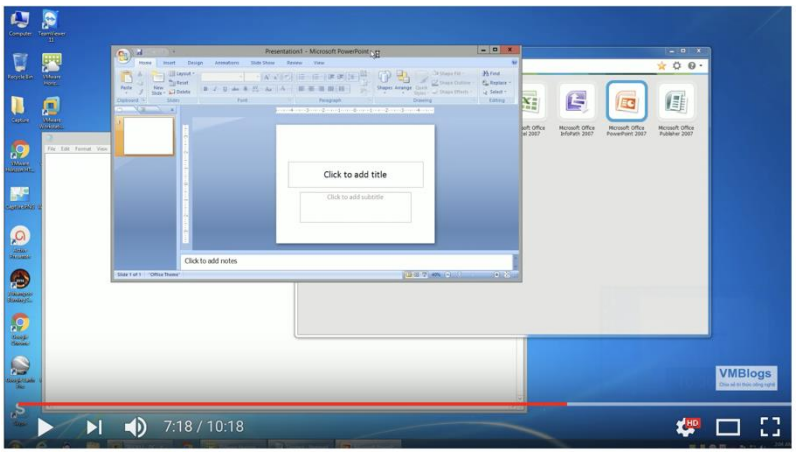
98. vSphere and ESX/ESXi in combination with VMware Horizon also practice the step of selecting an application interface particular to the client device. When an application or desktop is selected via the Horizon Client, Horizon passes client variables to the vSphere platform that hosts the selected application/desktop. Based on these variables, a version of the application/desktop is selected for the specific client, for example, a Windows desktop supporting an HD display and touch capable, as shown below:

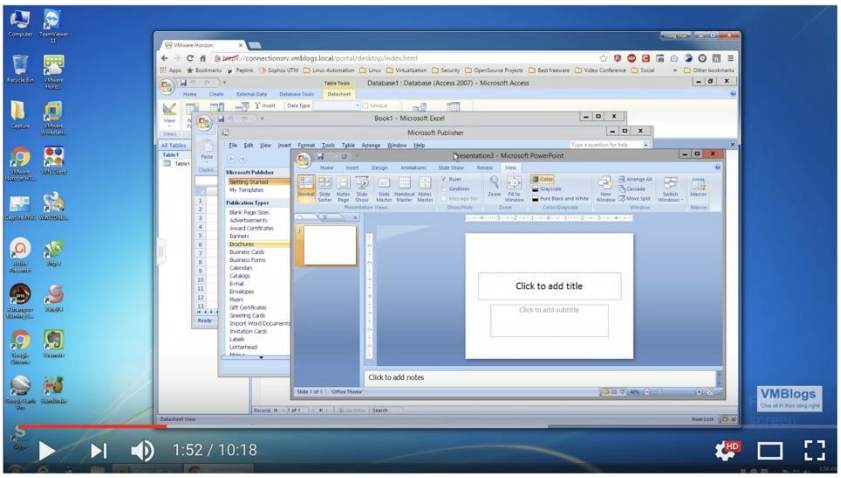
1.1 VMware User Environment Manager

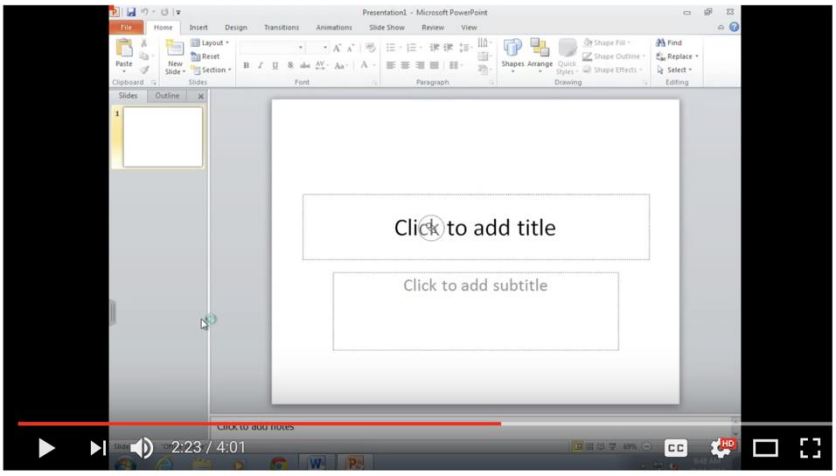
VMware User Environment Manager offers a desktop that adjusts to the actual situation of the end user, providing access to the IT resources that are required, based on a user's role, device and location. Many organizations suffer from hidden productivity loss as a result of ad hoc activities like manually mapping network drives and printers or providing application shortcuts to end users. This so-called distortion not only impacts IT departments but also affects end users. The relevant user experience that User Environment Manager offers, significantly eliminates this distortion.

Using View for Application Delivery

View also includes a feature called Unity Touch, which provides a gesture-based option for accessing Windows applications on iPads or other tablets. Because of small screen size, using mobile devices for navigating applications and files can be challenging. Unity Touch makes that easier. Unity Touch allows users to browse, search for, and open virtualized and hosted Windows applications, and switch between multiple open applications, without using the Windows Start menu or task bar. The file manager, task bar, and Start menus are also responsive to touch navigation. For more information, see Introducing Unity Touch.





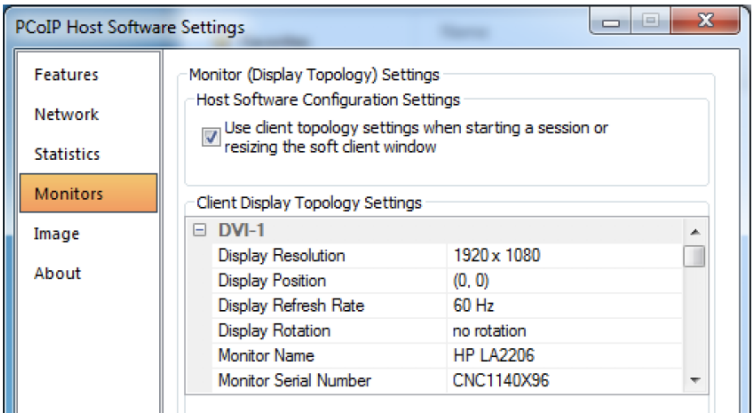


99. vSphere and ESX/ESXi in combination with VMware Horizon further practice the step of providing rendering information for the selected application interface to the client device. The vSphere platform uses the information about the client environment, such as, client operating

system, display capabilities, network capabilities or device type in order to provide information regarding the rendering of the application interface to the client device, as shown below:

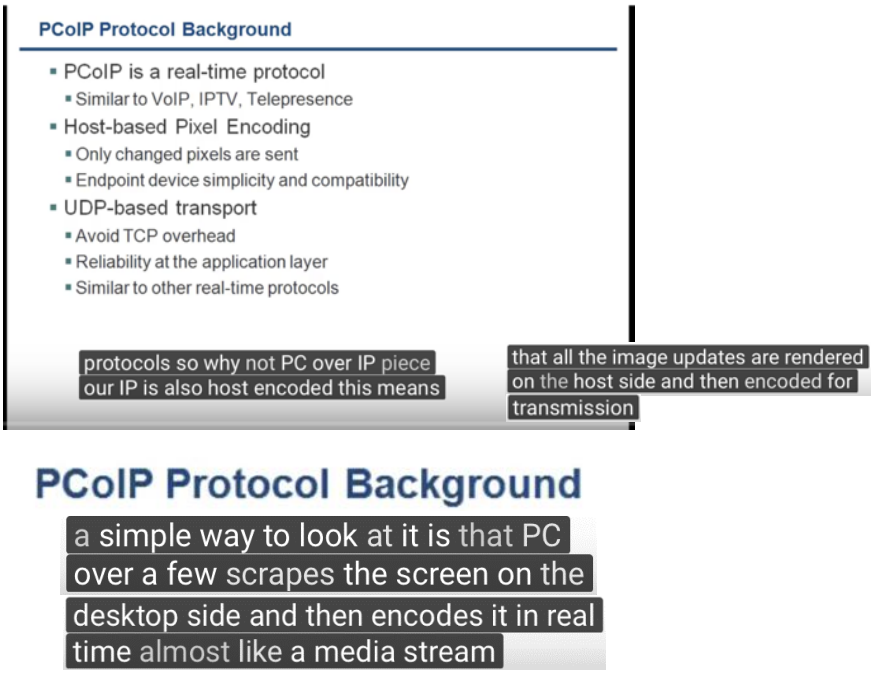
Deliver a Better Desktop Experience

Horizon View delivers the best end-user experience across locations and devices. Horizon View with PCoIP adapts to the end user's network connection to provide a high-quality customized desktop experience over the LAN and WAN. Users can flexibly connect to their Horizon View desktop from a variety of devices, including desktops, thin or zero clients and mobile devices. Horizon View supports 3D graphics rendering, unified communications, multi-monitor configurations, and audio and video content with multi-media redirection (MMR). The results are a rich, seamless user experience and maximum productivity for all types of end users.

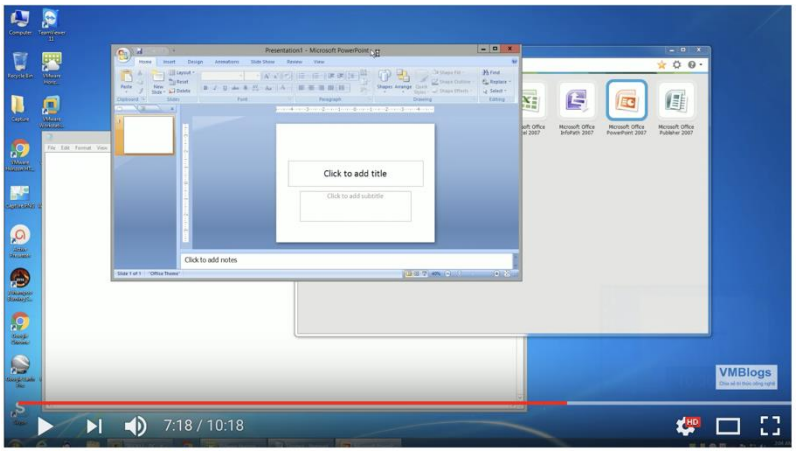


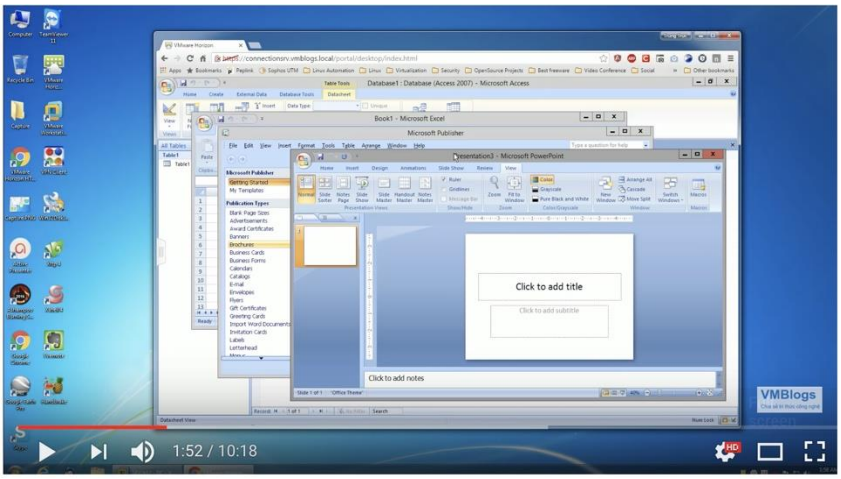
1.1 VMware User Environment Manager

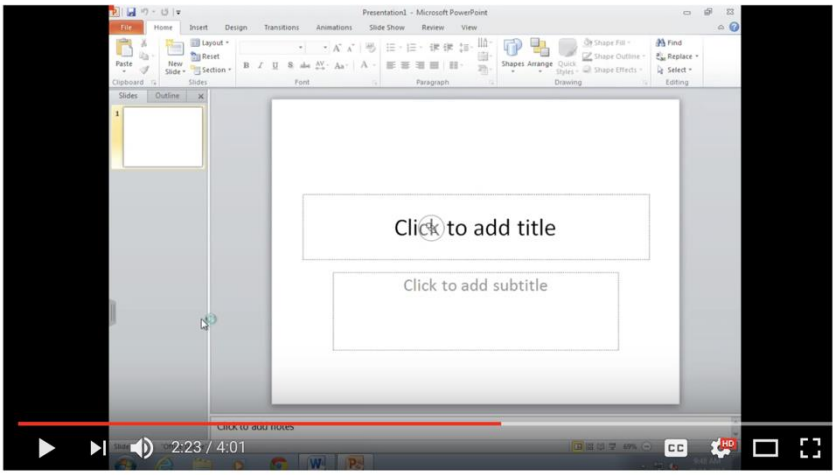
VMware User Environment Manager offers a desktop that adjusts to the actual situation of the end user, providing access to the IT resources that are required, based on a user's role, device and location. Many organizations suffer from hidden productivity loss as a result of ad hoc activities like manually mapping network drives and printers or providing application shortcuts to end users. This so-called distortion not only impacts IT departments but also affects end users. The relevant user experience that User Environment Manager offers, significantly eliminates this distortion.



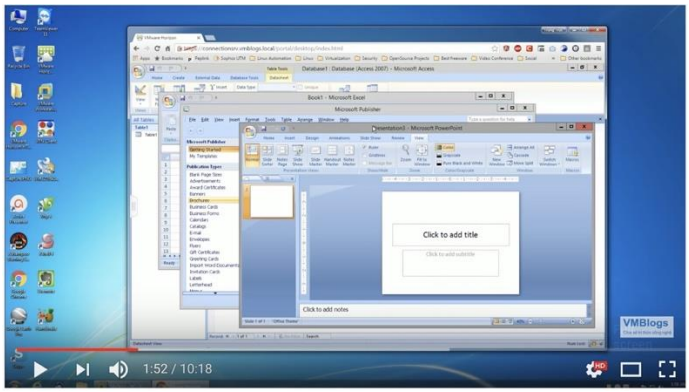
100. vSphere and ESX/ESXi in combination with VMware Horizon practice the step of invoking the server-based application. For example, the application is launched on the vSphere back-end and rendering information sent to the client device, as seen below:

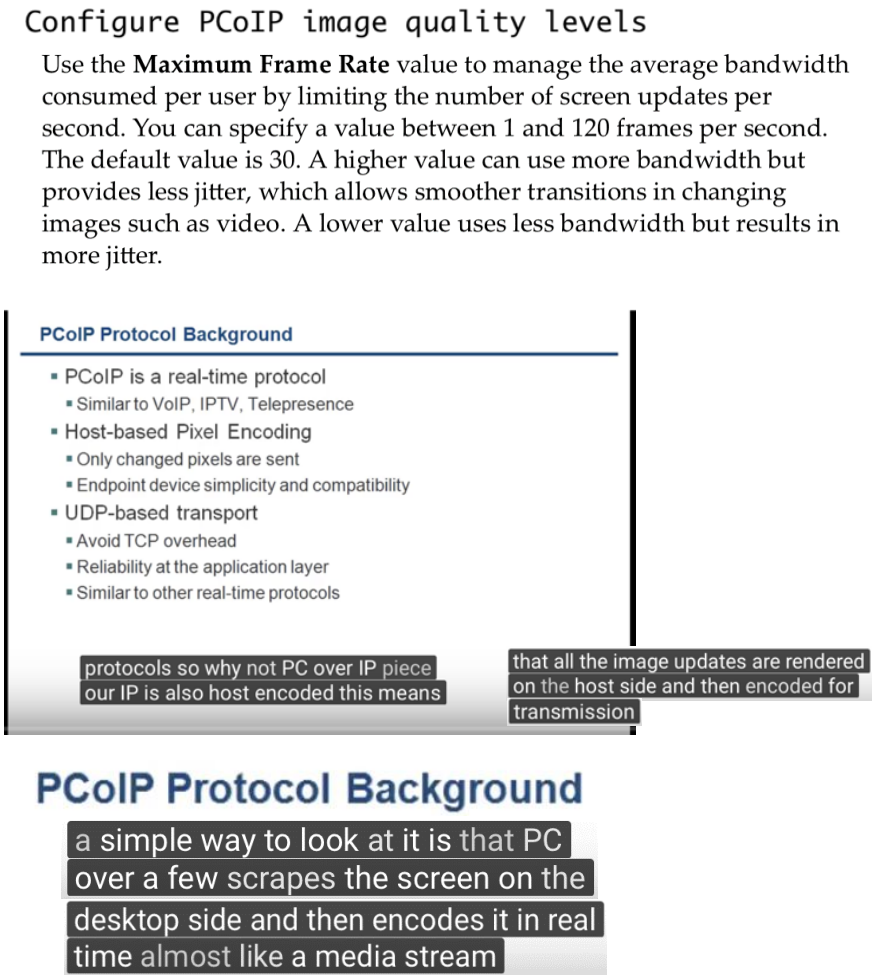




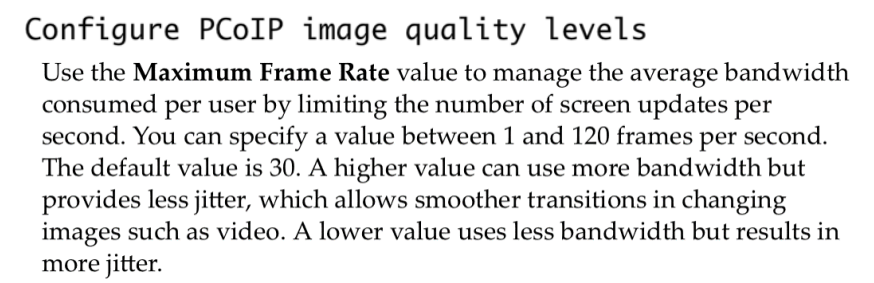


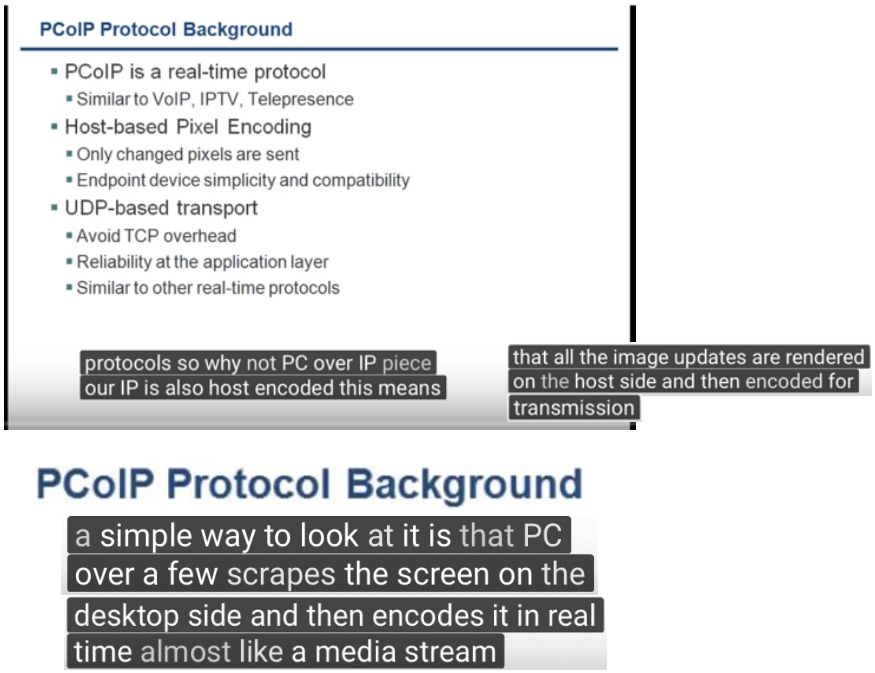
101. vSphere and ESX/ESXi in combination with VMware Horizon practice the step of receiving an action request from the client device. For instance, when a client device is using a remotely hosted application or virtual machine, the display protocol can monitor user input activity, such as mouse clicks and keyboard inputs, as seen below:



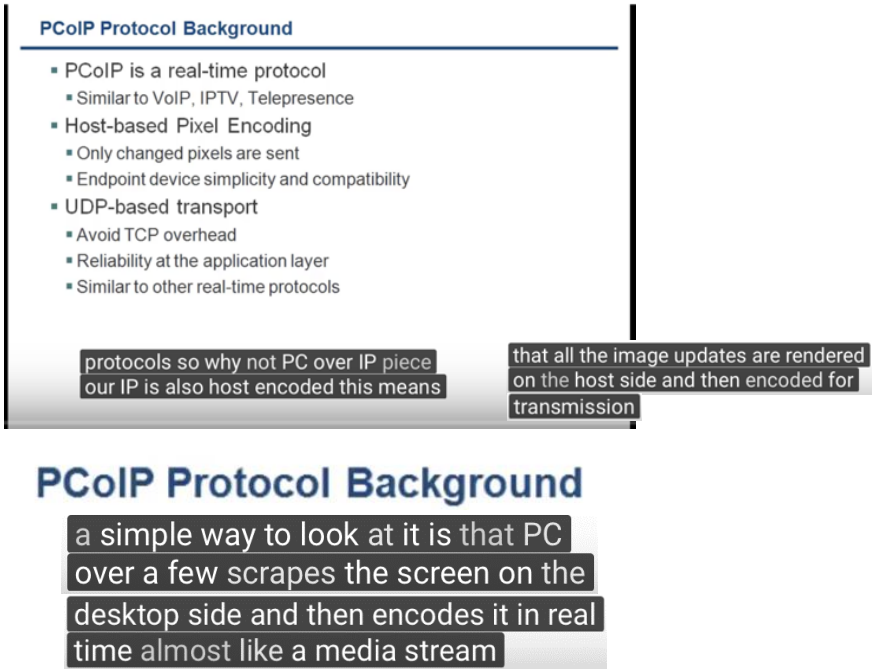


102. vSphere and ESX/ESXi in combination with VMware Horizon practice the step of interpreting the action request to identify changes required to the application interface. When the vSphere back-end receives information indicating interface actuation, such as mouse clicks and keyboard input from the client device, it interprets how the application interface display must change, as seen below:

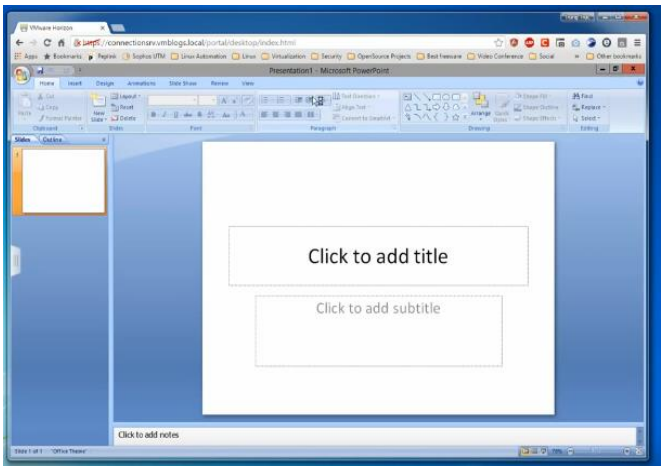


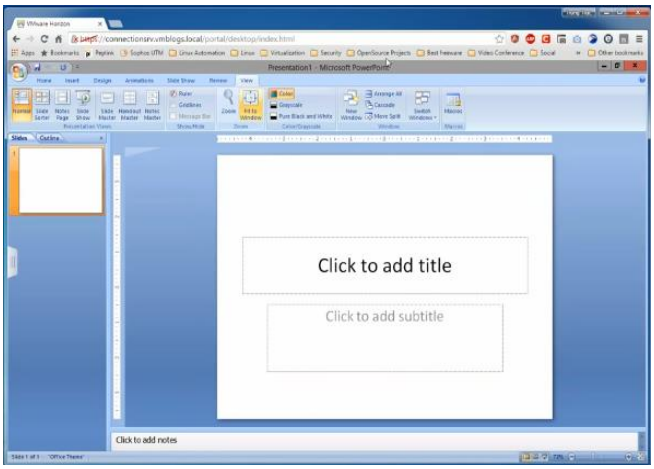


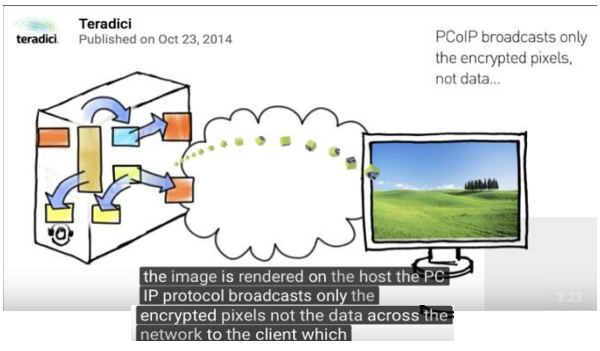
103. vSphere and ESX/ESXi in combination with VMware Horizon practice the step of providing rendering for the changes to the application interface to the client device. For example, the application interface changes are rendered on the host side and then encoded and transmitted to the client device for display, as seen below:



104. vSphere and ESX/ESXi in combination with VMware Horizon practice the step of invoking a server-based application command in accordance with the action request. Once the vSphere back-end has processed and rendered the change associated with the action request, the host invokes the application command associated with the request, as seen below:







105. Additionally, VMware has been, and currently is, an active inducer of infringement of the ’963 patent under 35 U.S.C. § 271(b) and contributory infringement of the ’963 patent under 35 U.S.C. § 271(c) either literally and/or by the doctrine of equivalents.

106. VMware has actively induced, and continues to actively induce, infringement of the ’963 patent by intending that others use, offer for sale, or sell in the United States, products and/or services covered by one or more claims of the ’963 patent, including but not limited to Horizon/ Horizon View, Horizon HTML 5 Client, vSphere, ESX/ESXi, vCenter Server, Connection Server, and any VMware product and/or service, alone or in combination, that operates in materially the same manner. VMware provides these products and/or services to others, such as customers, resellers and end-user customers, who, in turn, use, provision for use, offer for sale, or sell in the United States products and/or services that directly infringe one or more claims of the ’963 patent.

107. VMware has contributed to, and continues to contribute to, the infringement of the ’963 patent by others by knowingly providing products and/or services that, when installed and configured result in a system as intended by VMware, directly infringes one or more claims of the ’963 patent.

108. VMware knew of the ’963 patent, or should have known of the ’963 patent, but was willfully blind to its existence. Upon information and belief, VMware has had actual knowledge of the ’963 patent since at least as early as the service upon VMware of this Complaint. By the time of trial, VMware will have known and intended (since receiving such notice) that its continued actions would infringe and actively induce and contribute to the infringement of one or more claims of the ’963 patent.

109. VMware has committed, and continues to commit, affirmative acts that cause infringement of one or more claims of the ’963 patent with knowledge of the ’963 patent and knowledge or willful blindness that the induced acts constitute infringement of one or more claims of the ’963 patent. As an illustrative example only, VMware induces such acts of infringement by its affirmative actions of intentionally providing hardware and or software components that when used in their normal and customary way as desired and intended by VMware, infringe one or more claims of the ’963 patent and/or by directly or indirectly providing instructions on how to use its products and/or services in a manner or configuration that infringes one or more claims of the ’963 patent, including those found at one or more of the following:

- https://docs.vmware.com/en/VMware-vSphere/index.html
- https://my.vmware.com/web/vmware/info?slug=desktop_end_user_computing/vmware_horizon_clients/4_0
- https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/products/horizon/vmware-horizon-7-datasheet.pdf
- https://docs.vmware.com/en/VMware-Horizon-7/7.0/com.vmware.horizon-view.planning.doc/GUID-5C5C9E6B-263E-4DA0-895F-4F1415F37947.html
- https://docs.vmware.com/en/VMware-Horizon-7/7.1/view-71-configuring-remote-features.pdf
- https://www.youtube.com/watch?v=6ddJGbxpwxc
- https://www.youtube.com/watch?v=KQXB6Y_db28

110. VMware has also committed, and continues to commit, contributory infringement by, *inter alia*, knowingly selling products and/or services that when used cause the direct infringement of one or more claims of the ’963 patent by a third party, and which have no

substantial non-infringing uses, or include a separate and distinct component that is especially made or especially adapted for use in infringement of the ’963 patent and is not a staple article or commodity of commerce suitable for substantial non-infringing use.

111. As a result of VMware’s acts of infringement, Plaintiff has suffered and will continue to suffer damages in an amount to be proved at trial.

COUNT V
(VMWare’s Infringement of U.S. Patent No. 6,816,464)

112. Paragraphs 1-111 are reincorporated by reference as if fully set forth herein.

113. The elements claimed by the ’464 patent, taken alone or in combination, were not well-understood, routine or conventional to one of ordinary skill in the art at the time of the invention. Rather, the ’464 patent claims and teaches, *inter alia*, an improved way to test candidate network routes and select a subset thereof for interconnecting gateways across a wide area network (WAN), which were not present in the state of the art at the time of the invention. The invention improved upon existing route monitoring and selection techniques by providing for automatic route monitoring, scoring, and evaluation on a per communication link basis and in real time. The invention can further implement user-customizable routing preferences in the course of providing said route scoring and evaluation, allowing for fine-grained user control over the routing process.

114. Compared to the prior art, the claimed approach implements improved routing intelligence by providing route monitoring, scoring, and evaluation that automatically accounts for a plurality of candidate routes and measurements on a per communications link basis while also considering user-customizable routing preferences.

115. The invention represented a technical solution to an unsolved technological problem. The written description of the ’464 patent describes, in technical detail, each of the

limitations in the claims, allowing a person of skill in the art to understand what those limitations cover, and therefore what was claimed, and also understand how the non-conventional and non-generic ordered combination of the elements of the claims differ markedly from what had been conventional or generic in the industry at the time of the inventions of the '464 patent. More specifically, the claims of the '464 patent recite methods and systems for assessing network routes for use in establishing a communications link including identifying candidate routes and associated terminal gateway(s), transmitting quality measurement packets to determine route quality metrics on each candidate route, and receiving quality measurement packets to determine route quality statistics from each candidate route. The claimed systems and methods further recite determining route statistics based on routing information in the quality measurement packets, configuring a route ordering schedule based on user input, and scoring the candidate routes based on the routing statistics and the route ordering schedule to configure a scoring table that includes a quality score and packet loss, jitter, and/or delay.

116. The system covered by the asserted claims, therefore, differs markedly from the conventional and generic systems in use at the time of this invention, which inter alia lacked the claimed combination of identifying a plurality of candidate routes, transmitting and receiving quality measurement packets, determining route statistics, configuring a route ordering schedule based on user input, and scoring the candidate routes to configure a scoring table that includes a quality score and packet loss, jitter, and/or delay. Embodiments of the present invention further teach operation across packet-switched networks (such as the Internet); inclusion of databases that store and allow for consideration of historical routing information; graphical user interfaces ("GUIs") for accepting user routing preferences; and advanced settings for configuring route measurement properties, timings, and statistical analysis.

117. As described above, the ’464 patent is drawn to solving a specific, technical problem arising in the context of loss, latency and jitter sensitive route selection in packet-switched networks. Consistent with the problem addressed being rooted in such packet-switched network environments, the ’464 patent’s solutions consequently are also rooted in that same technology and cannot be performed with pen and paper or in the human mind.

118. VMware has directly infringed, and continues to directly infringe, literally and/or by the doctrine of equivalents, individually and/or jointly, at least claims 1, 2, 4, 5, 6, 7, 8, 10, 11, 12, 13, 19, 20, 21, 22, of the ’464 patent by making, using, testing, selling, offering for sale and/or importing into the United States products and/or services covered by one or more claims of the ’464 patent. VMware’s products and/or services that infringe the ’464 patent include, but are not limited to, VMware NSX and SD-WAN by VeloCloud (“VeloCloud”), and any other VMware products and/or services, either alone or in combination, that operate in substantially the same manner.

119. Claim 1 of the ’464 patent is reproduced below:

1. A method for assessing network routes for use in establishing a communications link within a communications network, comprising the steps of:

(1) identifying a plurality of candidate routes that can be used to establish said communication link, wherein a terminating gateway associated with each of said plurality of candidate routes is identified;

(2) transmitting quality measurement packets for each of said candidate routes, wherein said quality measurement packets can be used to determine at least one route quality metric;

(3) receiving returned quality measurement packets for each of said candidate routes, wherein said returned quality measurement packets can be used to determine route statistics;

(4) determining route statistics, wherein said route statistics are based on routing information contained within said quality measurement packets;

(5) configuring a route ordering schedule based on user set levels of route characteristics; and

(6) scoring each of said candidate routes based on route statistics and said route ordering schedule, wherein a scoring table is configured that includes a quality score and one or more of packet loss, average delay, and average jitter.

120. As one non-limiting example, VMware's VeloCloud practices a method for assessing network routes for use in establishing a network communications link.

> The VMware NSX® SD-WAN Edge by VeloCloud™ appliance is a compact, thin edge device that is zero-touch provisioned from the cloud for secure, optimized connectivity to applications and data. The NSX SD-WAN Edge is also available as a VNF (virtual network function) for instantiation on a virtual CPE platform. The NSX SD-WAN Edge with Dynamic Multi-Path Optimization (DMPO) and deep application recognition aggregates multiple links (e.g. Private, Cable, DSL, 4G-LTE) and steers traffic over the optimal links to other on-premises NSX SD-WAN Edges in branch offices, private data centers, campuses, and headquarters. The NSX SD-WAN Edge can also optionally connect to the system of global VMware NSX® SD-WAN Gateway by VeloCloud™ as shown in Figure 2 to provide performance, security, and visibility for cloud services (SaaS, IaaS, B2B Internet).

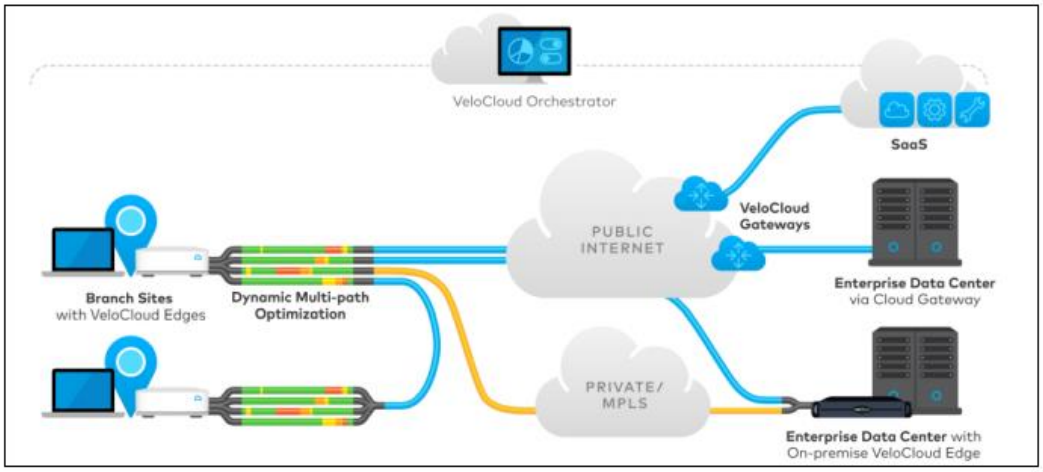


Figure 2: NSX SD-WAN Service

Continuous Link Monitoring

- Drives automation and optimization

Dynamic Per Packet Steering

- Sub-second steering without session drops
- Aggregated bandwidth for single flows

On Demand Remediation

- Protects against concurrent degradation
- Enables single link performance

121. VeloCloud further includes functionality for identifying a plurality of candidate routes that can be used to establish said communication link, wherein a terminating gateway

associated with each of said plurality of candidate routes is identified. For example, the dynamic multi-path optimization ("DMPO") functionality allows VeloCloud to monitor a plurality of traffic paths across multiple available connection types (e.g., MPLS, broadband, LTE, etc.) between VeloCloud-enabled customer endpoints (e.g., customer edge and gateway devices including VCEs and VCGs), as illustrated below.

SD-WAN dynamically utilizes multiple available connections (MPLS, broadband, LTE) to find the optimal delivery path for traffic across the entire network, shaping the bandwidth as needed to eliminate jitter and dropped data packets, thereby delivering an optimal user experience regardless of location.

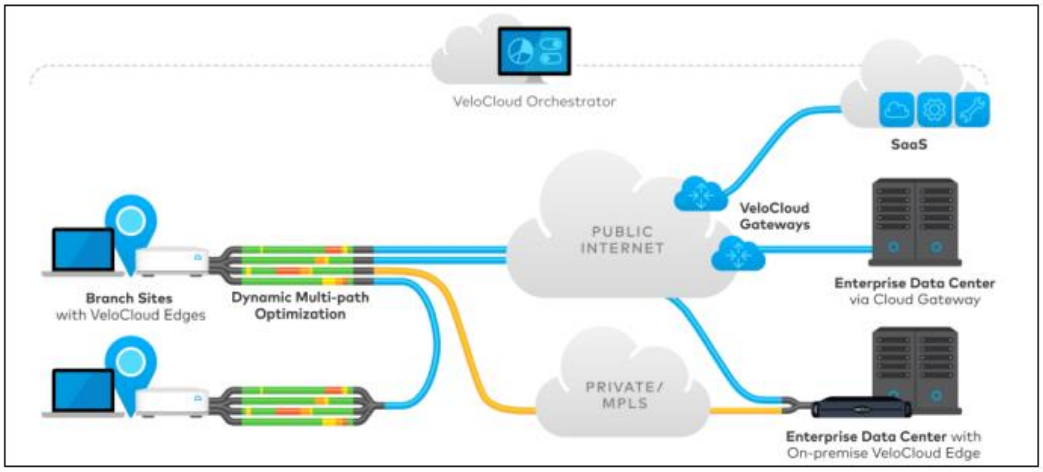


Figure 2: NSX SD-WAN Service

The NSX SD-WAN solution has three main components:

Edge (VCE) - highlight location flexibility (cloud, dc, branch), form factor flexibility

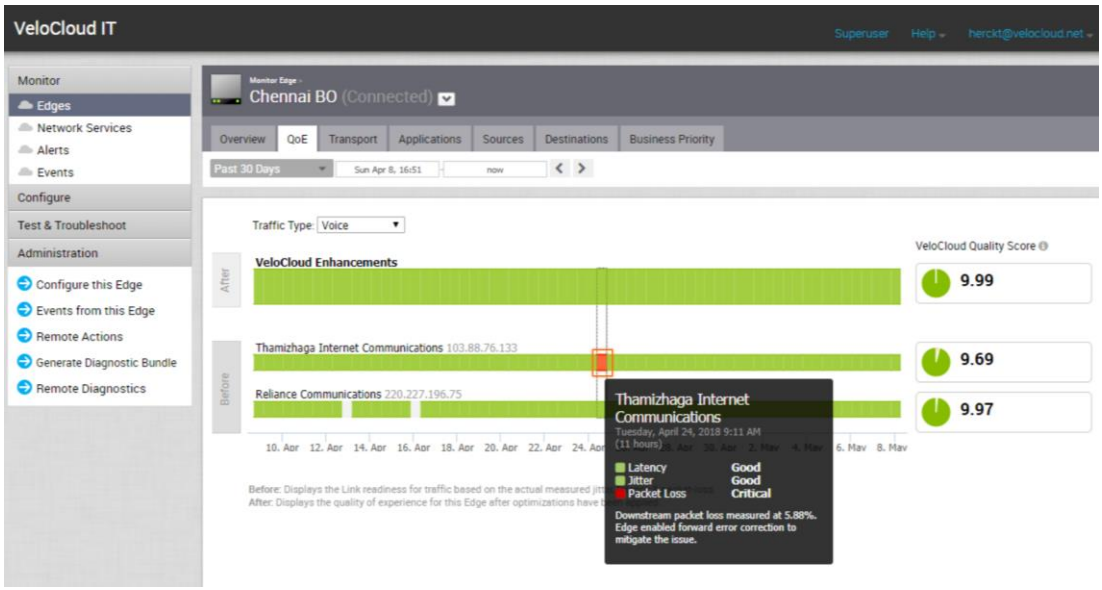
Orchestrator (VCO) - Virtual, Multi-tenant, highlight simplicity and no CLI, enables fast ramp of IT teams, less need for skilled resources, monitoring and troubleshooting are key, API Integration (eg AT&T leverages APIs)

Gateway (VCG) - Virtual, Multi-tenant with functions on data plane and control plane, VCG has global presence with partnerships with major service providers. Supports both cloud and on-premise model.

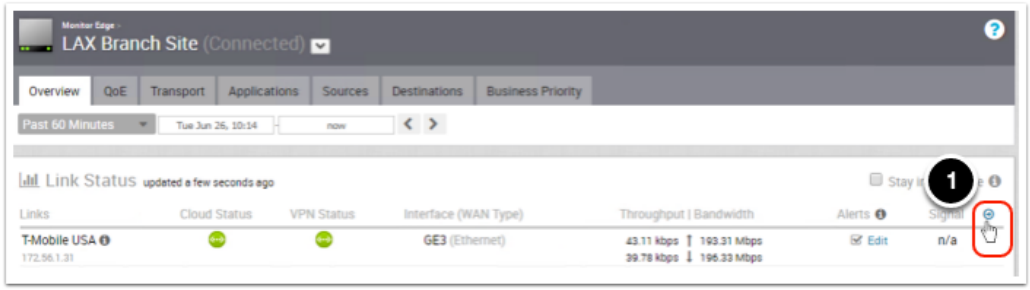
122. VeloCloud further includes functionality for transmitting quality measurement packets for each of said candidate routes, wherein said quality measurement packets can be used to determine at least one route quality metric. For example, DMPO is a monitoring protocol capable of transmitting packets in the uplink direction to measure various link performance attributes, as shown below.

Continuous Path Monitoring

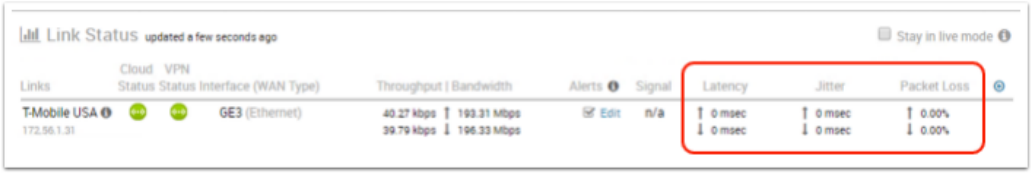
DMPO performs continuous, uni-directional measurements of performance metrics — loss, latency and jitter of every packet on every tunnel between any two DMPO endpoints, VCE or VCG. VeloCloud's per-packet steering allows independent decisions in both uplink and downlink directions without introducing any asymmetric routing.



There are real time measurements available that characterize the links, latency, jitter and packet loss behavior.



1. Click on the blue arrow to get the details on the WAN performance parameters like Latency, Jitter, %Packet loss.

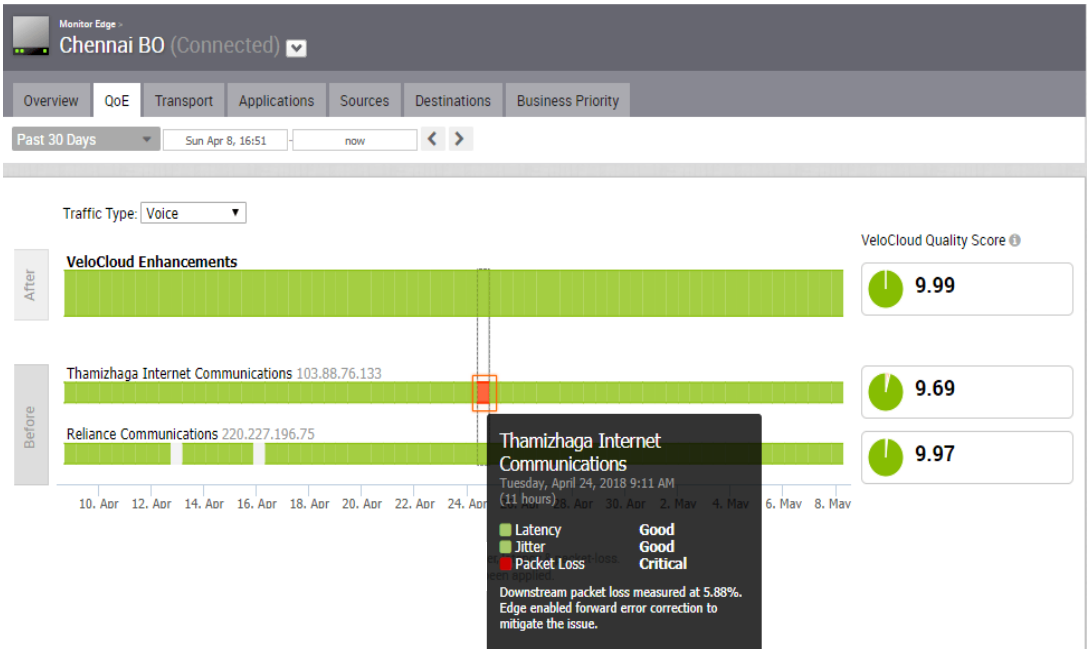


123. VeloCloud further includes functionality for receiving returned quality measurement packets for each of said candidate routes, wherein said returned quality

measurement packets can be used to determine route statistics. For example, DMPO is a monitoring protocol capable of receiving packets in the downlink direction to measure various link performance attributes, as shown below.

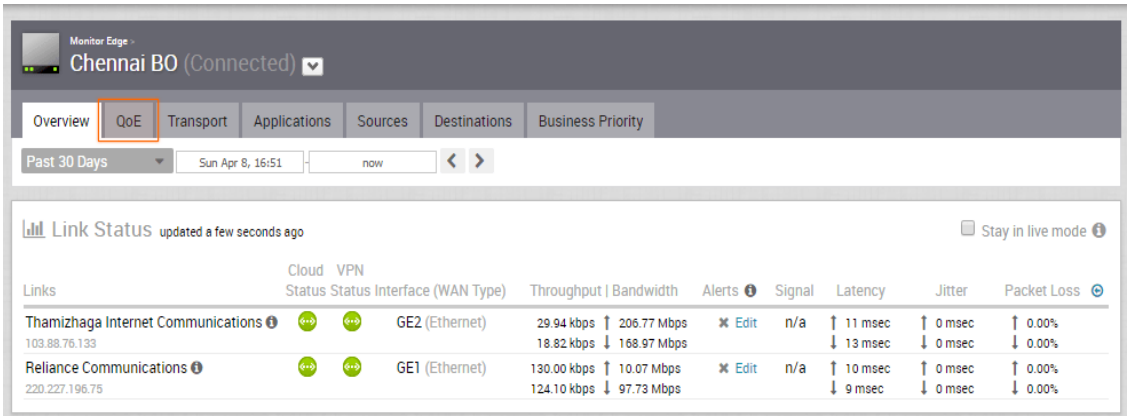
Continuous Path Monitoring

DMPO performs continuous, uni-directional measurements of performance metrics — loss, latency and jitter of every packet on every tunnel between any two DMPO endpoints, VCE or VCG. VeloCloud's per-packet steering allows independent decisions in both uplink and downlink directions without introducing any asymmetric routing.



Before: Displays the Link readiness for traffic based on the actual measured jitter, latency & packet-loss.
After: Displays the quality of experience for this Edge after optimizations have been applied.

124. VeloCloud further includes functionality for determining route statistics, wherein said route statistics are based on routing information contained within said quality measurement packets. For example, VeloCloud can determine and display routing statistics for each available path based on the DMPO monitoring, as shown below.



There are also real time measurements available that characterize the links' latency, jitter and packet loss behavior.

- Click on the Link Status detail icon to explore real time link statistics

This is a critical component to understand what the links are capable of transporting and what the impact of these conditions are on applications.

When on the Edge Overview page, the edges are also instructed to stream real time throughput numbers to the Orchestrator to give administrators a better sense of the utilization of the Edge.

125. VeloCloud further includes functionality for configuring a route ordering schedule based on user set levels of route characteristics. For example, Dynamic Application Steering functionality enables VeloCloud to store and automatically consider user-level preferences related to route selection when performing path evaluation for particular application flows, as shown below.

Dynamic Application Steering

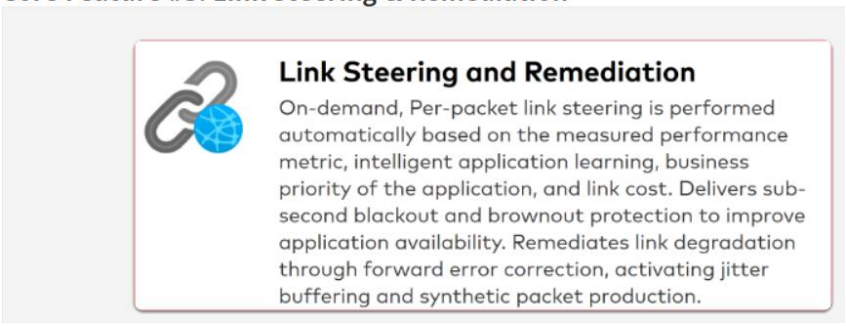
Applications are automatically recognized and steered to the optimal link(s) based on business priority, built-in knowledge of application network requirements, and real-time link performance and capacity metrics. Dynamic per packet steering can move a session, for example a voice call, mid-stream to avoid link degradation without any call drop or even voice quality glitch. Single high bandwidth flows can utilize aggregated bandwidth for faster response times.

Enterprise-wide Business Policies

NSX SD-WAN makes setting policy as simple as a single click. Enterprises or their managed service providers can define business level policies that apply enterprise wide across many Edges, all through a centralized, cloud based NSX SD-WAN Orchestrator. Link steering, link remediation and QoS are all applied automatically based on the business policies; however specific configuration overrides may also be applied. The centralized NSX SD-WAN Orchestrator also provides an enterprise wide view and configurability of routing in an overlay flow control table, eliminating complex node by node route configurations.

126. VeloCloud further includes functionality for scoring each of said candidate routes based on route statistics and said route ordering schedule, wherein a scoring table is configured that includes a quality score and one or more of packet loss, average delay, and average jitter. For example, VeloCloud can assign scoring tables for available traffic paths based on the above DMPO monitoring and Dynamic Application Steering intelligence, as shown below.

Core Feature #3: Link Steering & Remediation



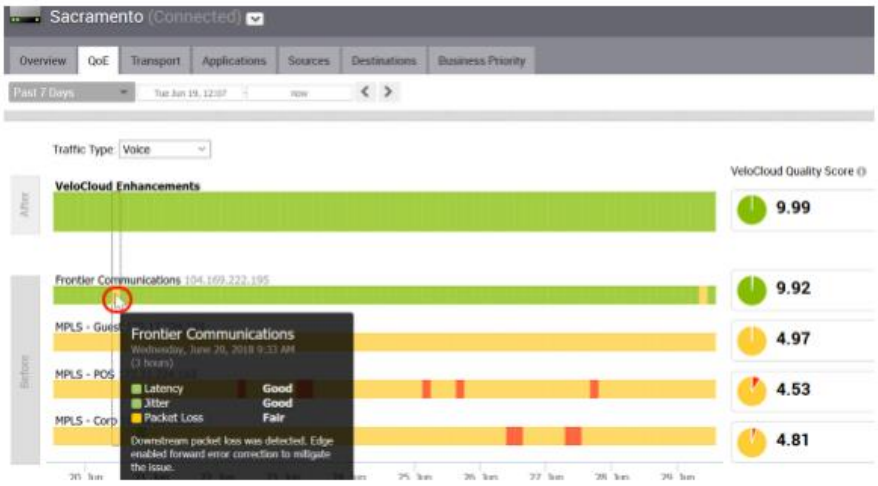
Another way to determine the link quality is to look at the Quality of Experience (QoE) rating

- Click on the QoE tab

The screen shows the VeloCloud Quality Sore (VQS) for each of the links and rates them on a scale from 0 through 10 on how well the links perform to carry a certain type of traffic (Voice in the default case)

Quality of Experience

Another way to determine the link quality is to look at the Quality of Experience (QoE) rating.



127. Additionally, VMware has been, and currently is, an active inducer of infringement of the ’464 patent under 35 U.S.C. § 271(b) and contributory infringement of the ’464 patent under 35 U.S.C. § 271(c) either literally and/or by the doctrine of equivalents.

128. VMware has actively induced, and continues to actively induce, infringement of the ’464 patent by intending that others use, offer for sale, or sell in the United States, products and/or services covered by one or more claims of the ’464 patent, including but not limited to VMware NSX and VeloCloud, and any VMware product and/or service, alone or in combination, that operates in materially the same manner. VMware provides these products and/or services to others, such as customers, resellers and end-user customers, who, in turn, use, provision for use, offer for sale, or sell in the United States products and/or services that directly infringe one or more claims of the ’464 patent.

129. VMware has contributed to, and continues to contribute to, the infringement of the ’464 patent by others by knowingly providing products and/or services that, when installed and

configured result in a system as intended by VMware, directly infringes one or more claims of the ’464 patent.

130. VMware knew of the ’464 patent, or should have known of the ’464 patent, but was willfully blind to its existence. Upon information and belief, VMware has had actual knowledge of the ’464 patent since at least as early as the service upon VMware of this Complaint. By the time of trial, VMware will have known and intended (since receiving such notice) that its continued actions would infringe and actively induce and contribute to the infringement of one or more claims of the ’464 patent.

131. VMware has committed, and continues to commit, affirmative acts that cause infringement of one or more claims of the ’464 patent with knowledge of the ’464 patent and knowledge or willful blindness that the induced acts constitute infringement of one or more claims of the ’464 patent. As an illustrative example only, VMware induces such acts of infringement by its affirmative actions of intentionally providing hardware and or software components that when used in their normal and customary way as desired and intended by VMware, infringe one or more claims of the ’464 patent and/or by directly or indirectly providing instructions on how to use its products and/or services in a manner or configuration that infringes one or more claims of the ’464 patent, including those found at one or more of the following:

- https://www.vmware.com/products/sd-wan-by-velocloud.html
- https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/product/vmware-deliver-sd-wan-solution-overview.pdf
- https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/product/vmware-enterprise-wan-agility-solution-overview.pdf
- http://docs.hol.vmware.com/HOL-2019/hol-1940-01-net_html_en/

- https://partners.lantelligence.com/wp-content/uploads/2018/07/dmpo-brief.pdf

132. VMware has also committed, and continues to commit, contributory infringement by, inter alia, knowingly selling products and/or services that when used cause the direct infringement of one or more claims of the ’464 patent by a third party, and which have no substantial non-infringing uses, or include a separate and distinct component that is especially made or especially adapted for use in infringement of the ’464 patent and is not a staple article or commodity of commerce suitable for substantial non-infringing use.

133. As a result of VMware’s acts of infringement, Plaintiff has suffered and will continue to suffer damages in an amount to be proved at trial.

PRAYER FOR RELIEF

Plaintiff requests that the Court enter judgment against VMware as follows:

(A) finding that VMware has infringed one or more claims of each of the above patents-in-suit, directly and/or indirectly, literally and/or under the doctrine of equivalents;

(B) awarding damages sufficient to compensate Plaintiff for VMware’s infringement under 35 U.S.C. § 284;

(C) finding this case exceptional under 35 U.S.C. § 285 and awarding Plaintiff its reasonable attorneys’ fees;

(D) awarding Plaintiff its costs and expenses incurred in this action;

(E) awarding Plaintiff prejudgment and post-judgment interest; and

(F) granting Plaintiff such further relief as the Court deems just and appropriate.

DEMAND FOR JURY TRIAL

Plaintiff demands trial by jury of all claims so triable under Federal Rule Of Civil Procedure 38.

Date: March 25, 2020

Respectfully submitted,

/s/ Derek Gilliland
DEREK GILLILAND
STATE BAR NO. 24007239
SOREY, GILLILAND & HULL, LLP
P.O. Box 4203
109 W. Tyler St.
Longview, Texas 75601
903.212.2822 (telephone)
903.212.2864 (facsimile)
derek@soreylaw.com

KARL RUPP
State Bar No. 24035243
NIX PATTERSON L.L.P.
1845 Woodall Rodgers Fwy., Suite 1050
Dallas, Texas 75201
972.831.1188 (telephone)
972.444.0716 (facsimile)
krupp@nixlaw.com

OF COUNSEL:

Paul J. Hayes
phayes@princelobel.com
Matthew Vella
mvella@princelobel.com
Robert R. Gilman
rgilman@princelobel.com
Jonathan DeBlois
jdeblois@princelobel.com
Alex Breger
abreger@princelobel.com
PRINCE LOBEL TYE LLP
One International Place, Suite 3700
Boston, MA 02110
Tel: (617) 456-8000
Fax: (617) 456-8100

COUNSEL for PLAINTIFF