Category	Requirement
Alerting	The system should generate alerts as and when thresholds are violated. Multiple alarm delivery methods should be supported - email, text messages (SMS), SNMP traps, pager messages, etc.
Alerting	Alerts should be sent only to administrators/users responsible for a specific problem (e.g., network alerts to network team, ESX alerts to VMware team).
Alerting	A history of events and duration of events should be provided. Users should be able to filter the events based on severity, based on the servers affected, based on duration of events, etc.
Alerting	If an alert is not resolved with in a certain time period, it should be escalated to upper management. The number of levels of management and who receives an alert should be configurable.
Alerting	The monitoring system should include a high availability option, so that alerting can be done 24*7.
Alerting	Administrators must be able to indicate to the monitoring system times when maintenance of the servers or networks is being done, so alerts are suppressed during such periods. Maintenance periods should be configurable for each server or groups of servers.
Alerting	The monitoring system should be able to generate alerts into our trouble ticketing system.
Alerting	Alerts should be generated when a problem is detected. Users should also be informed when a problem is resolved.
Alerting	User should be associated with shift periods - time periods when they will receive alerts. Users can also choose to receive email alerts during one period, and text messages during another.
Alerting	Users should be able to acknowledge alerts to provide notes on a problem, to indicate who is working on a problem, etc.
Alerting	Alarm de-duplication capability should be included - the same alert should not occur twice, and alarm floods should be avoided.
Application Monitoring	The monitoring solution should auto-discover applications running in each guest VM of an ESX server
Application Monitoring	The monitoring solution should be able to monitor applications running inside the guest VMs of an ESX server either using agents or in an agentless manner.
Application Monitoring	Monitoring should include but not be limited to monitoring application availability and response times.
Application Monitoring	Administrators should be able to quickly determine if an application is running on a physical machine or a VM and if it is running on a VM, then the physical server that the VM is hosted on should be traceable.
Business Service Analysis	The monitoring solution should be able to monitor the infrastructure end-to-end, reporting service quality from the user perspective.
Business Service Analysis	The monitoring solution should be able to analyze the infrastructure from the business service context, and be able to translate a service performance problem into an actionable operational event that the IT staff can take action on. This analysis should be done considering VM to ESX server and ESX server to cluster relationships. Application to VM mappings should also be considered when available.
Connection Brokers	Monitor availability and responsiveness of Connection Brokers (Leostream, VMware VDM, etc.)
Connection Brokers	Monitor the workload of the Connection Broker (e.g., work queue length)
Connection Brokers	Monitor utilization of the virtual desktops managed by the Connection Broker (e.g., collect metrics regarding total desktops, total desktops assigned to users, desktops available for assignment to users)
Connection Brokers	Monitor Connection Broker processes and TCP ports to detect any failures
Dashboard	Includes a dashboard from where we can monitor VMs, ESX servers, Clusters, etc.
Dashboard	Dashboard is updated in real-time and is web-based
Guest OS	If the physical CPU usage of a VM is high, identify from the user interface which application process(es) running inside the VM were consuming many CPU cycles

Category	Requirement
Guest OS	If the active memory of a VM is high, identify from the user interface which application process(es) running inside the VM were consuming memory
Guest OS	If the disk or network activity of a VM is high, identify from the user interface which application process(es) running inside the VM were causing this activity
Guest OS	Discover the disk partitions in a guest OS and report on the capacity of each partition and the space available for each partition. Alert if the disk space available is below a pre-specified limit.
Guest OS	Monitor TCP traffic to and from the guest OS and determine if the TCP retransmission ratio is within acceptable limit
Guest OS	Monitor contention for CPU resources in the guest OS by tracking the run queue of processes in the guest OS
Guest OS	Support guest OS monitoring for Linux, Solaris, Windows (XP, 2000 server, 2003 server, Vista, 2008 server) guests
Guest OS	Support guest VMs that may be in different Windows domains or in one or more workgroups
Guest OS	If a guest VM has multiple network interfaces, report traffic on each of the network interfaces. Identify periods of congestion where network traffic is high.
Hardware	Monitor ESX, Connection Broker, VirtualCenter hardware by integration with existing hardware monitoring tools - HP Insight, Dell OpenManage, Sun ManagementCenter, etc.
Licensing	The monitoring solution should be licensed per ESX server monitored, independent of the number of CPUs, sockets, cores on the ESX server
Licensing	For monitoring the virtual infrastructure, the monitoring solution must not be licensed based on VMs hosted on each ESX server
Licensing	Monitoring licenses used for monitoring physical servers should be re-usable for monitoring virtual servers. Thus, when migrating applications from physical to virtual machines, or to monitor the servers hosting the virtual machines, existing monitoring licenses can be reused.
Licensing	The monitoring licenses should be transportable across virtualization platforms - i.e., the same license can be used to monitor either VMware ESX or Citrix XenServer.
Monitoring Architecture	Support agent-less monitoring of ESX servers
Monitoring Architecture	Provide Guest OS metrics without requiring agents to be installed on each guest
Monitoring Architecture	Can monitor ESX servers using Virtual Center, if available
Monitoring Architecture	Can collect metrics by connecting to multiple Virtual Center servers; Provide a single integrated view across VirtualCenters
Monitoring Architecture	Can have a single monitoring console monitoring ESX servers and Virtual Center servers in multiple locations; Use HTTP/HTTPS to communicate metrics to the management console
Monitoring Architecture	Impact on the performance of the ESX servers being monitored should be minimal
Monitoring Architecture	Frequency of the performance monitoring should be configurable from a central location
Monitoring Architecture	Should be able to turn the monitoring on and off on specific servers as required
Monitoring Architecture	The monitoring solution should be administered centrally from a central location.
Monitoring Architecture	The monitoring solution should allow for multiple levels of thresholds to be configured (e.g., minor alert at 90% disk usage, major at 95%), thereby allow early warning (proactive) alerts to be generated.
Monitoring Architecture	The monitoring solution should allow users to override the default settings. Thresholds should be settable for each metric being collected.
Monitoring Architecture	Administrators should have the option to decide which monitors should be turned on and which should be turned off. They should also be able to decide which VMs are to be monitored and to turn off the monitoring for specific VMs.
Monitoring Architecture	Automatic, time-varying baselines should be computed for the metrics collected, allowing norms of the infrastructure to be determined automatically

Category	Requirement
Monitoring Architecture	Since time-varying baselines can lead to unnecessary alerting in some situations (e.g., test environments where the workload may not be time variant), the monitoring solution should allow a combination of user settable (i.e., static) thresholds and automatically determined thresholds, to minimize false alerts.
Monitoring Architecture	Should support multiple ESX versions - ESX 3.0, 3.5, 3i, 3.5i, 4, 4i, 4.1
Monitoring Architecture	Support a highly available clustered monitoring architecture that includes a cluster of management servers so that if one of the managers fails, the other can take over the monitoring functions
Monitoring Architecture	The monitoring solution must be able to integrate with our existing enterprise management system (Integration with Tivoli, NetCool, OpenView, BMC, etc. should be supported). SNMP traps or other integration mechanisms should be supported.
Monitoring Architecture	If any access privileges are required for monitoring, such privileges will be encrypted in a highly secure format and stored by the monitoring system - 128 bit encryption will be used.
Monitoring Architecture	The monitoring system should be able to correlate VM performance with the performance of the ESX servers. The alerts should identify if the bottleneck is in the VM or in the ESX layers.
Monitoring Architecture	The monitoring system should be able to correlate VM and ESX server performance with other external infrastructure components - e.g., network, SAN, etc., and determine times when ESX/VM performance is impacted by network issues, SAN issues, etc.
Monitoring Architecture	The monitoring system should be able to correlate the performance of the VMs, ESX servers, networks, applications, etc. and be able to automatically pin-point where the bottleneck lies.
Monitoring Architecture	Root-cause analysis should be used to prioritize alerts. Based on alert severities, administrators can determine where they need to focus to resolve issues.
Monitoring Architecture	The correlation and root-cause diagnosis should be done automatically and should not involve weeks of customization and configuration to build correlation rules.
Monitoring Architecture	The correlation and root-cause diagnosis should take into account the dynamic nature of virtual infrastructures. Vmotion Live Migration and DRS should be accounted for.
Monitoring Architecture	The monitoring solution should be easy to install and setup. Reboot of production servers should not be required for the system to work.
Monitoring Architecture	Any and all changes to the monitoring system (e.g., addition of servers for monitoring, changing thresholds, adding new users, etc.) are audited and made accessible through an audit log capability. Historical searches and reports on the audit log can be done.
Monitoring Architecture	The monitoring solution should be able to monitor multiple virtualization platforms - VMware ESX, Citrix XenServer, Microsoft Hyper-V, Solaris LDoms, etc., and provide a single dashboard from where these different virtualization platforms can be monitored and managed.
Monitoring Architecture	The monitoring solution must be able to monitor virtual and physical servers and provide a single dashboard for monitoring the virtual and physical infrastructure.
Monitoring Architecture	Alerts should be generated to the enterprise management system when a problem is detected and also when a problem has been resolved.
Monitoring Architecture	The monitoring system should include a knowledge base to track previous problem resolutions, or it should be able to integrate with other knowledge management solutions.
Monitoring Architecture	Offer option for agent-based and agentless monitoring of ESX servers and applications
Monitoring Architecture	Monitoring solution should be extensible - can add new metrics ourselves without requiring programming and a lot of effort.
Monitoring Architecture	Can monitor our routers, switches, firewalls, proxy servers, DNS, SAN, NAS, and other infrastructure elements

Category	Requirement
Monitoring Architecture	Can monitor ESX servers without requiring Virtual Center
Monitoring Architecture	The monitoring solution should come pre-configured with thresholds based on industry-standard best practices.
Monitoring Architecture	Single console from where we can monitor Citrix and Virtual Desktop deployments
Monitoring Architecture	The metrics collected should be stored in a relational database for historical analysis. We should have access to the database schema for our own analysis.
Monitoring Architecture	The data stored should include raw data, trends, and daily/weekly summaries. The amount of data stored in the database must be configurable and the system must be able to automatically retain/delete old data.
Monitoring Architecture	Overhead for maintaining the monitoring system should be low. Should be able to push upgrades from a central console, rather than individually upgrade each of the monitors.
Monitoring Architecture	The monitoring system should be scalable. Expected deployment should include 1000s of ESX servers and tens of thousands of VMs. The monitoring architecture should support such scale.
Monitoring Architecture	The management server should be installable on a VM
Monitoring Architecture	A standard database server (Oracle, Microsoft SQL, etc.) should be used by the monitoring solution and it should be hosted on a VM.
Network	Monitor the network connectivity to the ESX server host; identify and alert if the ESX server is not reachable
Network	Monitor network traffic to and from each of the VM NICs
Physical Memory	Monitor total physical memory of the ESX server
Physical Memory	Monitor memory used by the VM Kernel
Physical Memory	Monitor memory used by the Service Console
Physical Memory	Monitor the total memory granted to the virtual machines by the VM kernel
Physical Memory	Monitor the total balloon memory in use
Physical Memory	Monitor the swap usage over time
Physical Memory	Monitor the amount of memory that is actively used
Physical Memory	Monitor free physical memory on the ESX server
Physical Memory	Monitor used physical memory on the ESX server
Processor	Discover the number of processors on the ESX server and track usage of each of the processors
Processor	Track physical CPU utilization of core components like VM kernel, service console, drivers, and vmotion
Processor	Report CPU utilization of core components in MHz to allow for capacity planning
Reporting	Provide in-depth reports that can be used to analyze the performance of each ESX server. Include graphs showing the performance of the hypervisor (OS), the VMs, the network for a common time window, thereby allowing cross correlation of performance across layers
Reporting	Provide in-depth reports on VirtualCenter performance, showing workload on the VC server (user connections), performance of the system OS, network connectivity, etc.
Reporting	Provide comparison reports that allow metrics to be compared across ESX server, VC servers, etc. in the network, to identify bottleneck areas
Reporting	Provide service level reports documenting the percentage of time that the VM infrastructure is meeting the service level expectation.
Reporting	Provide top-N reports that can list ESX servers, VC servers, etc. based on specific metrics (e.g., Top 10 ESX servers with registered VMs, Top 10 ESX servers with physical CPU utilization, etc.)

Category	Requirement
Reporting	Provide reports that can be used to analyze the performance of VMs. Track VMs as they move from one ESX host to another, yet provide a comprehensive report that includes resource usage for the VM across all ESX servers that it has resided on. Optionally, provide a breakdown of VM performance and usage for each ESX server that it has been hosted on
Reporting	Allow reports to be configured for specific time periods of a day (e.g., 9am to 5pm only)
Reporting	Allow reports to be saved in PDF format
Reporting	Allow reports to be automatically generated and emailed to users
Reporting	Allow users to choose which reports that are to be automatically delivered
Reporting	The reports must be customizable. Administrators should be able to choose which metrics appear on the report.
Resource Pools	Track the percentage of physical CPU used by VMs in the resource pool; also report CPU usage in MHz
Resource Pools	Track the CPU reservation for VMs in each resource pool and compare with the CPU used by the VMs in the resource pools
Resource Pools	Track the memory overhead incurred by each of the resource pools
Resource Pools	Discover the resource pools for each ESX server
Resource Pools	Identify the child resource pools (if any) for each resource pool
Resource Pools	Discover the VMs directly assigned to each resource pool
Resource Pools	Identify the VMs in each resource pool that are powered on
Resource Pools	Track the memory used by VMs in each resource pool and compare with the memory reservation for each resource pool
Storage Adapters	For each storage adapter, monitor physical reads and writes to disk
Storage Adapters	For each storage adapter, monitor issued commands and commands aborted
Storage Adapters	For each storage adapter, monitor data reads and writes in MB/sec
Storage Adapters	For each storage adapter, monitor read and write latencies at the kernel, physical disk, and command queue
Storage Adapters	For each storage adapter, monitor the number of bus resets
Storage Adapters	For each storage adapter, monitor the number of bus resets
Storage Adapters	Report overall data transfers to and from the storage adapters
User Views	The monitoring system should support multiple user roles. Admin users should be able to setup/alter the monitoring configuration. Operators should have read-only access to the monitored infrastructure (i.e., they can see alerts but should not be able to alter the monitoring configuration). Executives should have access to only reports.
User Views	When supporting multiple users, the monitoring solution should have the ability to tie into our Active Directory environment, so user passwords are not required to be stored separately in the monitoring system.
User Views	The monitoring system must provide a web-based interface for monitoring the infrastructure. Web access over SSL must be provided for secure access.
User Views	The monitoring system should be able to show a pictorial geographic map showing the affected parts of the infrastructure.
User Views	The monitoring system should provide different views for different users - so a user only sees the infrastructure (e.g., ESX servers, VC servers, network devices, etc.) that he/she is responsible for.
Virtual Center	Monitor CPU, disk, memory, and network resource usage on the VirtualCenter host
Virtual Center	Monitor the windows service(s) and process(es) supporting VirtualCenter
Virtual Center	Monitor the availability and responsiveness of the VirtualCenter database
Virtual Center	Collect in-depth metrics regarding the usage of the VirtualCenter database
Virtual Center	Monitor network availability to the VirtualCenter host
Virtual Center	Monitor number of sessions currently handled by each VirtualCenter
Virtual Center	Provide details of current sessions on VirtualCenter - e.g., user logged in, login time, etc.

Category	Requirement
Virtual Desktops	Should be able to monitor the login and logout times of users to the Virtual Desktops
Virtual Desktops	Provide an easy way to determine which ESX server and VM a user is logged on to
Virtual Desktops	Be able to track what applications a user is accessing on his/her virtual desktop
Virtual Desktops	Identify which users are logged on to VMs on each ESX server
Virtual Desktops	Provide reports comparing the number of simultaneously powered on virtual desktops on each ESX server. Identify if load balancing is working well.
Virtual Desktops	Provide reports showing the top user sessions by duration for a specific time period.
Virtual Desktops	Provide reports showing the typical CPU, memory, network resources used by each user
Virtual Machine	Indicate the disk capacity available to a VM irrespective of whether it is powered on or off. Comparing the disk capacities of VMs can give an idea of which VM(s) are taking up much of a datastore's storage capacity
Virtual Machine	Indicate the % of physical CPU used by each VM that is powered on
Virtual Machine	Provide the actual physical CPU usage of a VM in MHz to allow what-if migration analysis
Virtual Machine	Monitor the percentage of CPU waits for each powered on VM
Virtual Machine	Monitor the percentage of extra physical CPU used up by a VM
Virtual Machine	Monitor Ready time for each VM - i.e., percentage of time the guest was ready to run but was not able to execute because of Processor contention at hypervisor level
Virtual Machine	Monitor CPU throttled time per VM to identify VMs that do not have proper CPU limits set. The throttled (or max limited) time is the percentage of time the ESX host deliberately did not run the VM because of CPU limit settings for the VM.
Virtual Machine	Track the configured physical memory for each VM and identify times when the configuration is changed
Virtual Machine	Monitor the physical memory consumed by each VM
Virtual Machine	Track the shared memory used by a VM and the balloon memory of the VM to identify memory pressure
Virtual Machine	Monitor disk read and write rates for each VM to identify VMs that are contributing to disk activity of the ESX server
Virtual Machine	Track disk commands issued and aborted for each VM
Virtual Machine	Monitor network traffic to and from each VM on an ESX server
Virtual Machine	Provide an easy way to track which ESX server a VM is running on currently
Virtual Machine	Monitor the uptime of each VM and alert when a VM is rebooted
Virtual Machine	Monitor the percentage of extra CPU used up by the VM
Virtual Machines	Auto discover the virtual machines registered on each ESX server host
Virtual Machines	For each registered VM, report its IP address and guest operating system
Virtual Machines	Dynamically rediscover the VMs registered on each ESX server host to account for Vmotion Live Migration
Virtual Machines	Identify the state of each VM so as to determine which VMs are powered on and which are not
Virtual Machines	Report the number of VMs registered on each ESX server host
Virtual Machines	Report the number of VMs powered on for each ESX server host
Virtual Machines	Monitor the number of VMs added to an ESX server (either new VMs or as a result of Vmotion)
Virtual Machines	Provide details of the VMs added to an ESX server and the time when the VM was added
Virtual Machines	Track the number of VMs removed from an ESX server during a measurement period
Virtual Machines	Provide details of the VMs removed from an ESX server and the time when the VM was removed
Virtual Machines	Provide history of VM additions and removals from each ESX server

Category	Requirement
Virtual Machines	Monitor the network connectivity to each of the VMs on an ESX server and alert when a VM loses connectivity to the network (e.g., because it has blue-screened)
Virtual Machines	Auto discover the mapping of virtual machines to datastores
Virtual Machines	Generate alerts to indicate which VMs are impacted by a datastore failure
Virtual Machines	Report the number of powered-on VMs with users logged in (for VDI support)
Virtual Machines	Report the number of powered-on VMs with no users logged in (for VDI)
VM Clusters	Automatically discover the VM clusters configured for each VirtualCenter server
VM Clusters	Automatically discover the ESX servers mapped to each cluster
VM Clusters	Monitor the hardware associated with each cluster - number of cores, physical CPU available for the cluster, memory available for the cluster
VM Clusters	Monitor and report on the total CPU utilization of the cluster
VM Clusters	Monitor and report on the total memory usage of the cluster
VM Clusters	Discover resource pools associated with a cluster and the memory and CPU usage of each of the resource pools in the cluster
VM Clusters	Automatically discover the VMs in each cluster; Report on total number of VMs registered for the cluster and the number simultaneously powered on; Provide details of which VMs are powered on
VMFS Datastore	Discover VMFS datastores on an ESX server
VMFS Datastore	Monitor space usage of VMFS datastores on the ESX server host
VMFS Datastore	Monitor availability of VMFS datastores on the ESX server and alert when a datastore is not available