

Chris Beckett's VCAP-DTD Notes v0.1

Foreword

These are the exam notes I used to get me through the VCAP5-DTD exam. These are free to the community and you should not have paid anything to get these. If you spot any errors or omissions in this document, feel free to send them to chrisbeckett999@googlemail.com. It's not been proof read, so there may well be a few porkies in here. Not deliberate ones, perhaps some assumptions I made that were wrong or some tech I misunderstood.

Please also note that this is a living document, so the copy you have may be out of date. Always worth checking my blog at <http://beckettsblog.wordpress.com> to see if a newer version exists.

Always use these notes in combination with the exam blueprint from VMware Education. This is obtained free of charge from

http://mylearn.vmware.com/mgrReg/plan.cfm?plan=32566&ui=www_cert

Section 1 : The Exam

The exam is 131 questions and has a duration of 225 minutes (3.75 hours), passing score is 300 out of a maximum of 500.

The exam comprises multiple choice, drag and drop and design item questions.

Remember the design factors - RRAC - Risks, Requirements, Assumptions and Constraints. All should form part of any solution design

The design takes three phases - conceptual, logical and physical. You start at the highest overview level, and as you drop through the phases, you add more and more detail as you go.

Objectives

Objective 1.1 - Identify business requirements for a View design

- Functional vs non-functional requirements. A functional requirement is what a solution will "do" and a non-functional requirement is something that a solution will "be". An example of this is that a functional requirement should be a set of inputs or outputs and be something demonstrably measurable, such as solution must have sufficient capacity for 500 virtual desktops with a 15% growth overhead for the next three years. A non-functional requirement would be that the solution must run on HP servers and NetApp storage, with Cisco switches. As these are "qualities" or characteristics of the solution, these are considered non-functional.

- Conceptual, logical, physical designs. Conceptual designs generally encompass the end goal of a solution without ever detailing the steps or materials required to get to that point. For example, “I want a bunch of virtual desktops for my Finance people to use from their laptops via a client” is a concept, nothing more. From here, the design becomes an iterative process where the next step is to draw a logical design. The logical design will encompass objects such as hosts, storage, networking but not contain details such as how much RAM or CPU a host will have, or what RAID configuration storage will have. It’s a high level drawing of the inter dependencies and relationships of objects required to make the solution whole. The physical design then drops a layer deeper into the design process by fleshing out the logical design by including details of the HP server hardware for the ESXi hosts, for example. This would then be followed by details of the NetApp storage, the RAID configuration (RAID-DP etc.), number of disks in the aggregate, OnTAP versions. The networking would contain details on the switch vendor, switch model, any VLAN trunking, vSwitch properties (standard vSwitch, VDS, switch security, VLAN tagging, Jumbo Frames, etc.). This would be in enough detail that a bill of materials can be derived and an implementation plan can be written
- Business requirements can take a number of different forms, but for a View deployment would generally be aligned to the following topics :-
 - Hosts (new hardware purchase, reprovision existing hardware, limitations of incumbent vendor such as max amounts of CPU, memory)
 - Storage (different tiering - SSD Tier 1, 15K SAS Tier2, 10K SATA Tier 3 etc. and the requirement to use the bare minimum on a cost basis)
 - Networking (taking into account corporate LAN users but also users in branch office scenarios with limited bandwidth)
 - Licensing (to keep costs down, customers may wish to take advantage of SQL Express or the vCenter Appliance, thus also reducing complexity)
- Business Case for desktop virtualisation. There can be many business drivers for VDI, but the main items include:-
 - Security - runs atop vSphere with the inherent security built in, plus if a physical desktop, laptop or tablet is lost or stolen, no corporate data is lost
 - Cost - not always easy to justify, but as control becomes centralised, so costs become more predictable
 - Agility - the ability to quickly provision and maintain desktops is a key business benefit for View. Contrast this with the length of time it would take to requisition a physical laptop/tablet, have the corporate build assigned and then the ongoing patch requirements plus the maintenance overheads in case of physical failure or damage to a hard drive, cracked screen etc.
 - One Management Console - all managed via the web, with no “fat client” requirement. Management can be done via PC, laptop or tablet with no need to expose desktop admins to the vSphere Client
 - PCoIP is automatically tuning and provides superb performance out of the box on the LAN or WAN, much better than traditional RDP
 - View Persona Management - streams down portions of the user’s profile on a “just in time” basis. This reduces issues around profile corruption and slow

logon/logoff times as only the portions of the profiles in use are streamed from the network

- ThinApp - permits the virtualisation of applications. This is a key driver as it places an app in a “bubble”, meaning different versions of software can co-exist on the same device (Internet Explorer) for compatibility reasons without affecting each other. Also has a streaming function for when the application is large and placing it on the master image does not make sense. As the app is started, enough is streamed down from the network share to start the program, while the rest is loaded in the background. These applications may also be configured to pre-cache when a user logs on for quicker startup times.
- View Composer - allows use of linked clones from a master image. Massively reduces provisioning times and also storage footprints, as linked clones are only changes from the master image and so are kept very small, rather than full clone images which can use a lot of expensive storage and drive up costs.
- vShield Endpoint - anti-virus is off loaded to a separate appliance on each host to reduce the overhead of scans and updates on virtual desktops.
- Hardware independence - as desktops are virtualised, this permits access to Windows desktops and apps from platforms such as Mac, iPad etc.
- Hardware lifecycle - means old PC hardware that cannot support Windows 7 (insufficient physical resources, driver compatibility) can be repurposed as thin client devices, meaning the life span of hardware can be elongated, reducing costs
- Identify requirements for conceptual design
 - Number of users
 - Number of virtual machines
 - Storage requirements
 - Networking requirements
 - Future expansion
 - MTR/RPO in case of failure, availability with High Availability
 - Security (anti-virus, HIPS)
 - Management (physical/virtual vCenter, separate clusters)
- Determine availability requirements for View Conceptual Design
 - When must the solution be available? 9-5? 24/7?
 - What do we do in the event of failure?
 - Load balancing options?
- Determine reliability requirements for View Conceptual Design
 - What steps can we take to eliminate points of failure?
 - Shared storage with built in redundancy
 - Striped networked connections between multiple physical switches
 - Multiple ThinApp repositories in case of server loss
 - Use VMware HA to provide service in the event of host loss
 - Where should load balancing devices sit in the overall design
- Determine manageability requirements for a View Conceptual Design
 - What platforms must we manage from? Windows? Mac? iPad?

- Will the desktop admins be separate from vSphere admins?
- Training required for desktop admins?
- From the public internet or just on the corporate LAN?
- Administrators at branch offices?
- Identify interoperability requirements
 - What platform will consumers use? Windows? Mac, iPad, Android?
 - What protocol options are available per platform? So Windows most fully featured, then goes down from Mac to Linux to tablet
 - Also consider interop with enterprise applications such as SCCM and anti-virus/HIPS solutions which may want to interact with virtual desktops
- Determine security requirements for a View Conceptual Design
 - On the corporate LAN or from the public internet too?
 - What about the use of certificate checking? Enforced, mandatory, over rides?
 - Will we allow the use of USB passthrough?
 - Will we permit the use of downloadable desktops via a Transfer Server?
 - What firewall ports will we need?
 - Which AD domains will be allowed access to which pools?
 - Are there any smart card requirements?
 - Use SSL tunneling wherever possible
 - Will we permit the use of copy and paste?
- Identify desktop personas and user experience requirements
 - Are we using local or roaming profiles
 - Are we using mandatory profiles?
 - What storage capacity will we need to store persona managed profiles?
 - Will users maintain multiple profiles if they also use physical desktops?
 - Will users require Aero, 3D rendering? Not just for CAD level users, but also think about Aero being used by MS Office etc. Is there an expectation that Windows 7 VDIs will be identical visually to Windows 7 on a physical desktop? Are there any training gaps for end users?
- Evaluate mobility requirements
 - Will users require access via tablet (iPad/Android?). If so, how to make the client available?
 - Will there be a training need for end users in regard to gesture based actions?
 - What about local mode desktops? Will users need these, consider time it takes for initial download and then policies around check-ins etc
 - What applications will be required by users using Local Mode desktops and what constraints will have on your design, including ThinApp?
- Determine ROI/TCO requirements
 - Use the provided calculators to determine the return on investment and total cost of ownership of any proposed View solution. If the costs are too high, the project may not even be kicked off.

Objective 1.2 - Identify Technical Requirements for a View Design

- Identify current workflow
 - How do users work now? Establish logon sequence, what sort of scenarios users have
 - Try to classify into three broad categories - task worker, knowledge worker, power user - sometimes there is also a VIP category (high value)
 - Which apps do they use and how frequently?
 - Do they have requirements for video/audio?
 - What is the user perception of the current physical environment? Acceptable? Slow? Takes too long to logon etc.? This will help us set expectation of View environment and illustrate “quick wins” to project sponsors/management
 - Are there multiple logons as part of the normal workflow? We should try and use single sign on as much as possible to simplify the experience for the end user
- Evaluate additional peripheral needs
 - How does a user currently print? Local, network, USB? Follow me printing? Something in house or exotic?
 - What requirements are there for “pluggable” devices? USB storage, cameras, headsets, headphones etc?
 - Do users require multiple monitor or large monitor support?
- Evaluate current network bandwidth and latency and determine network I/O requirements
 - Perform a current state analysis of desktop estate
 - Use a capacity planning tool such as VMware Capacity Planner, Liquidware, etc. **Do not guess.** Stats will be fluid depending on user scenario and even time of day/week (initial logon storm first thing in the morning, any daily reports run from a central server etc.). Cumulative and accurate stats will feed into the overall design to ensure you have sufficient bandwidth
 - Again try to categorise users into the three use cases based on their network I/O requirements. In most cases, desktop network I/O requirements are quite low
 - Remember that PCoIP is tuned out of the box, if all users are on the corporate LAN, you probably don’t need to change anything for optimal performance
 - Where users are on slower connections, there are several steps you can take to reduce latency
 - Consider the use of multi-media redirection
 - Use a local cache on the View Client, recommended 1GB cache, this is on Windows clients only, not thin clients or Linux
 - Build to lossless can be turned off, with no real perceptible image quality change to the end user
 - Use the GPO to configure PCoIP settings using pcoip.adm template
 - Configure the maximum audio bandwidth limit to reduce overall bandwidth required while not losing quality on audio. 50-100Kbps is suggested for congested networks where high audio quality is not required. As an additional guide, up to 450Kbps provides quality between that of a phone call and FM radio

- Review and run the commands.txt file provided to tune the Windows desktop for best performance, it may disable a service you need
- Remember that 3D graphics has a higher bandwidth utilisation, so don't enable this for everyone
- Some users may require a hardware solution for 3D graphics, this will require a card putting in at least one host (think Teradici card)
- Use the View Planner tool to help you predict the environment
- Use full duplex network connections end to end if you can
- Try to ensure round trip latency less than 250ms
- Consider using QOS tags for PCoIP traffic
- Don't tunnel PCoIP over SSL VPNs, use a Security Server
- Allow PCoIP traffic to burst where possible
- If you have a traffic shaping device on your network edge, try and configure a PCoIP friendly policy
- Don't use traffic shaping at the vSwitch level
- 3D graphics should have a latency sub 100ms, or users will notice the difference
- In network bandwidth constrained scenarios, consider the use of minimum and maximum image quality values in the GPO template. Don't use minimum value when high degree of accuracy is required, such as X-Rays
- Maximum Frame Rate is also useful, but don't set less than 10-15fps when video is in use. Useful in scenarios where a lot of video work is performed
- Maximum PCoIP Session Bandwidth can be useful where one or more users are hogging network bandwidth and a ceiling can be set
- PCoIP Session Bandwidth Floor allocates a reserved amount of bandwidth which can be called upon immediately, rather than having to wait for bandwidth to become available, however this should only be used in exceptional circumstances
- Evaluate current storage environment and determine storage I/O requirements
 - Again perform a capacity planning exercise - **do not guess**
 - Review the number of IOPS, but also review the breakdown of reads/writes to ensure that the RAID configuration you specify is appropriate for your needs
 - Remember XP has a 64KB block size and Windows 7 1MB block size. **This is important when computing IOPS**
 - Review existing storage for suitability for View, so examine the number of disks, maximum number of IOPS, RAID penalties, maximum storage capacities. Also review if existing storage has enhanced features such as VAAI and VASA support to help improve the solution's performance
 - Review your knowledge of RAID levels, especially RAID0, RAID10 and RAID 5, you will need to know what the write penalties are per configuration, this will form part of your design
 - How much storage will you need? Capacity planning will help analyse disk space currently in use, Composer should help reduce the amount of storage required where workloads are similar (especially in the case of task workers)

- Evaluate the use of the View Storage Accelerator to improve performance during logon storm type events. VMware classify desktop workloads as 90% read and 10% write, capacity planning statistics should bear this out
- Even though you may think that only SSDs will provide the raw performance in terms of IOPS, this will have a big impact on the solution cost, not only the upfront costs, but the lifespan of SSDs is still shorter than spinning disk. Customers may also be reluctant to fund additional SSD on the basis of cost, look for appropriate storage for other objects (such as persistent disks, disposable disks, ThinApp repository, persona management storage) based on performance and capacity vs costs
- Flash based solutions such as NetApp FlashCache may help alleviate any potential performance issues by “front ending” disk operations before flushing them out to disk
- Deduplication may also be an option, but View Composer linked clones largely negates the need for this type of solution
- Disk tiering at SAN/NAS level may also help. If you are using linked clones, remember that the read penalty is based on the replica disk, so place this on high performance storage and linked clones may be sited elsewhere on a cheaper disk which still provides sufficient write IOPS
- Is the current SAN on the vSphere HCL? Don't use it if not
- Based on current environment, determine CPU/Memory/Storage sizing requirements
 - Again, always use metrics obtained from your capacity planning exercise
 - Don't forget to include some overhead into your sizing, in case growth is expected or required as part of the design
 - Remember to include such things as RAID performance penalties, VM swap file requirements (allocated vRAM - vRAM reservation)
 - Look at workload patterns and try to establish where the first performance constraint will kick in (most likely RAM or storage, rather than CPU or network, but this is not a given)
 - Examine the categorisation of user workloads (task worker, knowledge worker etc.) and see how the split breaks down. Generally, you'd expect to find most users in the first two categories, with fewer users in the power user categories
 - Remember that not all storage vendors read and write data to disk in the same way, look for vendor white papers or best practices around View, most of the big vendors have them
 - If using NFS or iSCSI, ensure there is sufficient bandwidth to meet those needs. It may be that 10Gbps networking may be the best option here, as multiple 1Gbps NICs does not provide the same level of performance and only ever uses a single NIC
 - Remember that with transparent page sharing and memory compression techniques, you should be able to reduce your overall physical memory requirement by around 50%
- Evaluate application requirements and dependencies
 - Use the capacity planning tool to evaluate what is in use and by whom

- Use the “long tail effect” to pick on most frequently used applications first
- Can we use this application with Windows 7?
- Which applications can we use in a stateless desktop?
- Are there any specific graphics requirements for any applications (such as Aero, 3D or even a minimum amount of graphics memory, dual screens etc.)
- Do we need to run any legacy applications, or multiple versions of the same application (Internet Explorer, for example)
- Are there any core applications that would make better sense placed into the master image? There are licensing and patching implications to this
- Which applications require Windows services on bootup? What about peripheral connections such as card readers etc, can these applications actually be virtualised?
- If you have Java applications, what JRE versions are required? Can these be virtualised?
- If you have multiple JRE versions in applications, consider how best you should handle this. Multiple JRE versions in one ThinApp, or separate them out and call them as dependencies?
- Evaluate access requirements
 - Which AD domains require access? Which should not be permitted?
 - Are there any logon restrictions that need to be carried over from the existing environment, such as only logon during office hours etc?
 - Any access from the public internet, or only the corporate LAN?
 - Maybe you have some groups of users that should be allowed external access but not others. Think medical records or requirements around intellectual properties.
 - Consider use of tags to prevent certain groups of users from accessing resources
 - Is RSA or token based authentication required?
- Evaluate management and administrative needs and determine user groups and access requirements
 - Again refer to your capacity planning metrics to help you map out users and their workload profile
 - How many pools might we need?
 - Persistent or non-persistent desktops?
 - How many master images? The fewer the better, in general
 - Who needs access to the View Administrator tool? Create an AD group for them
 - Where admins require limited access, use AD groups to enforce this too
 - Create custom roles in View Administrator and give them “just enough permissions to perform”
 - Remember you can use the Inventory Administrators group to provide administrative access to pools and desktops without providing permissions to make system level changes

Objective 2.1 - Create a Desktop Virtual Machine Design

- Determine the appropriate virtual machine configuration(s) that will satisfy design requirements
 - What functional/non-functional requirements do you already have for the desktop?
 - Must be Windows 7
 - Must have Office 2010 Professional
 - Must have 3D graphics and Aero?
 - Analyse your capacity planning report to ensure virtual desktops are right sized in terms of CPU, memory and disk space
 - Remember to account for virtual hardware versions, as this will affect your ability to deliver on some items, especially around Local Mode desktops
- Establish image design requirements
 - Use capacity planning information again to see what software is used and when
 - Look to see what printing configurations are in use
 - How big does the C drive need to be?
 - If there is to be a D drive for user data, how big should that be?
 - Will we need more than 4GB RAM?
 - How will you handle paging and temp file requirements
 - Break out which users will always require the same desktops, same applications and who need to make persistent changes
- Establish OS requirements
 - Windows XP, or Windows 7 or both?
 - 32 bit or 64 bit?
 - What about multi-language requirements? Will we need proofing tools, language packs etc? This will bloat the image size
 - Which version of Windows? Professional? Enterprise? Depends on what license you have and if you have a requirement for the likes of BitLocker etc.
- Based on customer requirements, determine the number of images for the design
 - Use the capacity planning metrics and also any data you have collected with interviewing stakeholders to try and define different use cases
 - Where you can group users into use case profiles, see if you can meet this requirement with a few images as you can
 - Less images require less storage, management, patching
 - Some images may need specific pieces of software embedding, or may need to be in different languages
 - Do you need to maintain different patch levels of software and OS versions to meet legacy application requirements?
 - There may be different hardware requirements for different groups of users, from fairly modest with task workers up to additional memory and/or CPU for power users
- Identify applications to be virtualized or natively installed

- The overarching goal of desktop virtualisation is to “silo” each component, so OS, applications and persona are managed as separate entities which come together to make the complete stack
- Remember that each application you natively install into the master image has to be updated, patched and also bloats the master image size
- Better to use ThinApp if you can so that applications can be more easily patched, plus as licencing changes, you can remove applications from desktops by just removing their sandbox, you need not go into the registry or file system as such, which is quite invasive
- Having a ThinApp repository also means you have a central place of administration for all applications, once they’re packaged they can be used anywhere
- When a user wants an application, point them to the ThinApp repository, rather than having to crack open the master image to install something that not everyone will use
- If you have a pool of 50 desktops each with Office, you may need to purchase 50 licenses, even if 15 copies are only ever used
- Some applications will need to be natively installed if they are constrained by ThinApp functionality, so for example an application cannot be virtualised if a device driver is required
- If you have services installed by an application, this can still be virtualised
- Know the limitations of ThinApp – so there is no 64 bit application support and the behaviour of some apps will not allow virtualisation to happen (device drivers etc.)
- Establish peripheral device requirements
 - Will users require USB “plug in” functionality? Cameras, keyboards, pointing devices, iPads etc?
 - What about printers? Are there any USB connected printers or multifunction devices in use?
 - Do you need any specialist drivers for any of the above?
 - Consider how you might restrict certain USB devices from being plugged in, using Group Policy

Objective 2.2 - Determine Session Integration Requirements for a Desktop Pool Design

- Based on the customer requirements, establish the number and types of pools to be included in the design
 - Based on the capacity planning results, see if you can group the users into the aforementioned workload profiles
 - The goal should be keeping the design as simple as possible
 - Task based workers who use web applications for a high proportion of their work would be good candidates for non-persistent desktops

- These pools can also make use of linked clones to save storage space and allow for automated provisioning when supply and demand increases and decreases, power saving can also be used to reduce costs
- Power users such as developers may require persistent desktops with additional hardware (additional CPU and/or memory) and also require dedicated desktops, so they always get the same VM every time they logon
- Keep in mind the management of the pools, will this need to be delegated out along organisational lines? For example, do the Finance department have an IT technician who needs to be able to refresh and reboot desktops? Have a separate pool here so you can assign permissions at a granular level
- Establish the detailed pool characteristics
 - Evaluate such things as
 - Can the user choose their protocol?
 - What about security features such as USB, copy and paste?
 - Will the pool be automated or not?
 - Will you permit Local Mode desktops, and consider their impact
 - Persistent or non persistent desktops?
 - Persistent disk settings
 - Disposable disk settings
 - View Composer customisation method (SysPrep, QuickPrep)
 - Adobe Flash settings
 - Host cache settings
 - Much of this can be derived from your user workload profiling
- Identify delegated administration of roles
 - Again, review if there are IT technicians who require access to subsets of desktops to reboot, refresh or register desktops
 - This can be achieved using stakeholder analysis meetings
 - Create AD groups for delegated admin and add users to them
 - Add groups into roles in View Administrator
 - AD can then be used to control who has access to what
- Determine monitoring requirements
 - What needs to be monitored?
 - Stakeholder analysis should give you this information
 - Look for host based monitoring if possible
 - Also have monitoring and analysis tools such as Liquidware UX or vCenter Operations Manager for View to provide granular performance statistics
 - It may be the organisation already has an enterprise monitoring solution that you can plug the View deployment into
 - No additional learning curve
 - Helps keep costs down
 - SCOM integration possible for large enterprises
- Establish persona management

- Evaluate the storage requirements by establishing how many users are to be managed
- Review existing storage infrastructure to ensure it is fit for purpose
- VMware recommend one server per 1,000 users
- VMware recommends a volume striped across four disks, each with it's own SCSI controller (virtual or physical) as disk I/O can be high
- 8GB RAM and a minimum 1Gbps NIC is recommended on each persona repository server
- Identify where on the network the repositories will live, ensuring sufficient bandwidth and IOPS
- If there is a requirement for Windows XP, remember you will still need UPHClean
- Define how persona management will work by using the GPO template. Configure such items as folders to preload, which is useful when pre-caching ThinApp sandboxes
- The profile upload interval is 10 minutes by default, there is not really a use case to change this
- Decide if you want to store personas alongside Windows profiles
- Do not remove personas on logout, unless security policies mandate you should
- Establish if any profile folders should be exempted from persona management for any reason
- Remember that persona management must be enabled in the master image when the View Agent is installed
- Identify if there is any specific requirement for co-existence of persona managed and regular Windows profiles
- Don't scan the persona management repository with anti-virus, nor should you use backup software that uses VSS, this will corrupt the files
- Use persistent disks if users have large profiles, this will speed up logon
- One part of the deployment will be the plan to deploy persona management, ensure you have a good knowledge of all the steps required and which order they will need to be performed
 - Create CIFS share
 - Assign appropriate permissions
 - Link to repository in View Admin
 - etc
- Determine an appropriate anti-virus solution for a desktop pool design
 - Where possible, you should use vShield Endpoint for anti virus scans on View
 - It uses the vShield Manager (appliance) and usually one appliance per host which the scans and signature file updates are sent to (third party solutions vary, so you should check)
 - Solution forms part of vSphere and View Premier, but there is an overhead as virtual appliances will need to be installed, configured and managed, plus the cost of any third party software. vShield Endpoint in and of itself does not provide the complete protection, just the framework

- If vShield does not form part of the design and “regular” anti-virus is in use, how can you mitigate performance and reduce overall I/O?
 - Change daily scans to be staggered across all desktops
 - Review signature file update policies to see if these can be applied weekly out of hours, say Saturday or Sunday
- Based on customer requirements, identify appropriate firewall policies for the design
 - Access from the corporate LAN only, or internet too?
 - Consider the use of tagging, so only certain pools can be accessed by certain users
 - Private VLANs may be useful and come with VDS
 - vShield Zones can segregate traffic if required
 - Do you require the Windows firewall to be switched on?
 - Don't forget to use the ESXi host firewall
 - Only allow what ports you need, remember to review View documentation on DMZ type designs and what ports are required
 - For example, 80/443 for the initial connection, 4172 for PCoIP. You don't need to allow 4001 and 8009 as they are “internal only” connections. Do you need USB redirection?
- Establish application management
 - Configure a ThinApp repository
 - Store ThinApps in the repository, associate apps by pool or desktop
 - Manage ThinApps via View Administrator
 - Design a patching and maintenance strategy for each ThinApp, including when you are going to patch them
- Establish OS patching process
 - In an automated pool, it is simple enough to patch the master image and send out the update using recompose, but ensure the users are not disrupted
 - Consider the load on vCenter when a pool is recomposed or refreshed. In a large pool, this is not a trivial operation and a good window of time needs to be allowed.
 - In a full clone environment, you may need to patch each desktop individually
 - Consider solutions such as SCCM or Altiris, much like physical desktops
 - Maintains centralised control of patching
 - You may also have a security team who are in charge of patching who set and enforce organisational policies

Objective 3.1 - Create a View Component Infrastructure Design

- Determine the access infrastructure required
 - How many Connection and Security servers will you need?
 - Minimum of two Connection servers for redundancy, security servers are optional and are not really required if access is from a corporate LAN only
 - How many vCenter servers will you need?

- Review the business requirements and also the constraints. There may be existing hosts and/or storage that will be redeployed for a virtual environment
 - You may be constrained by physical hardware maximums, such as the maximum amount of physical memory in the hosts
- Ensure storage is kept in the appropriate tier - you don't need to use SSD for disposable disks, for example
- How many hosts? This will be determined by the number of desktops, but also what the hardware budget is and whether or not you are repurposing existing hosts.
- Remember limitations of ESXi with regard to max 8 host View clusters with VMFS. This may be mitigated by using NFS.
- Review load balancing requirements. If a load balancer is to be used, review the View documentation to see where that goes in the design and how it should be configured.
- Establish accounts necessary for View services
 - Depending on which components are to be used, ensure service accounts are created in AD for each service with the appropriate permissions (View Composer, vCenter Server etc.)
- Determine the number of View Standard/Replica servers required for the design
 - Should be a minimum of two connection servers per block, maximum is 7 to support up to 2,000 users
 - Remember the ADAM database is peered between the connection servers, meaning a copy of the database should one connection server be lost
 - Also defined by Windows licensing constraints
- Determine the number of View Security Servers required for the design
 - Do you actually need them? Only really for external access, corporate LAN should not require them
 - Again a minimum of two for redundancy, resiliency and load balancing
 - Remember you can pair multiple Security Servers with a single Connection Server
 - Remember to bear in mind Windows licensing constraints
 - Security Servers should always be in your DMZ and the Connection Servers firewalled off
- Determine the number of View Transfer Servers required for the design
 - Decide if you are going to employ local mode desktops, if not, you do not need a Transfer Server
 - Remember that Transfer Servers have very specific hardware requirements and have a high disk I/O profile, which will matter if you are going to virtualise it
 - Remember also that the network bandwidth may get saturated if there are many simultaneous check in/check out events, so additional bandwidth may be required for many users
 - If you plan on keeping several transfer repositories, remember to use network shares so that multiple Security Servers can access the base images. Also ensure there is sufficient disk space to store these base images

- Establish design characteristics of the Virtual Profile repository
 - No idea what this means, just use a network share with sufficient disk space, I guess (seems to be related to persona management, so see previous stuff)
- Determine an appropriate load balancing solution for the design
 - Use stakeholder feedback to find out if a layer 4 load balancing solution already exists, this can be re-used for View deployments
 - There are also other solutions such as round robin DNS, which is not really load balancing and also NLB from Microsoft, which is free but very limited
 - Use customer requirements plus constraints to help you decide what to do here

Objective 3.2 - Determine the Database Architecture for a View Component Infrastructure Design

- Determine the size and placement of vCenter Server database
 - Will this go on the vCenter server and have an all in one VM?
 - Separate database server or even separate shared physical cluster?
 - VMware recommend the database is stored separately from vCenter and View Composer
 - Sizing really depends on the level of statistics you collect and for how long you keep them, generally 20GB will suffice for one year collected at level four
 - Will you use the vCenter Server application for Windows, or will you use the vCenter virtual appliance? Remember you cannot use this for VUM and View Composer (with version 5.0 and older, at least)
- Identify which database servers/platforms will be used in the design
 - This will come back to stakeholder interviews, organisations usually have a preference for a database vendor, be it SQL or Oracle, depending on skills and licensing
 - Ensure the proposed database platform is supported for vCenter, View Composer and the View events database
 - Customers that want to use SQL Express have a limitation of 10GB, you need to factor this in when making your decision on which platform you choose
- Determine the size and placement of the View Events database
 - Keep it away from vCenter Server on a separate DB server, size is generally quite lowball, say 200MB. Depends how much you keep.
 - Remember the Events database will just grow unchecked unless you perform scheduled maintenance. Don't let this limitation bring your database over.
- Determine the size and placement of the View Composer database
 - I can't find any calculator for this, but same rules apply as in keep the database server away from vCenter Server. Forum posts suggest 100MB or so, even in very large use cases

Objective 3.3 - Create a ThinApp Repository Design

- Based on design requirements, determine applications for inclusion in the ThinApp repository
 - Go back to the customer requirements and see if there is a defined need to embed any applications in the master image, such as Office and/or Firefox. Also review your capacity planner output to see which applications should be good candidates for virtualisation, in as much as they are used by a high proportion of users
 - Some applications simply do not virtualise, mainly much older legacy applications
 - If applications use DCOM or kernel mode drivers, you may not be able to virtualise them and as such, they may need to be natively installed into your master image
- Determine storage and networking requirements for the ThinApp repository
 - You can use the RAWC tool to generate sample workloads to give you a rough idea of the bandwidth needed to stream an application such as MS Office
 - As a high proportion of the disk I/O will be read operations, ensure the RAID configuration takes account of this and provides sufficient throughput
 - Networking should be a minimum of 1Gbps, but again you can use the RAWC tool to get an approximate profile of each application's requirements
 - Even though you need good read throughput, the ThinApp repository is not necessarily a good candidate for Tier 2 storage. Also, if disk space requirements are high, SSD type solutions are not particularly cost effective
- Determine placement of application repository
 - The repo needs to be placed on a network share with sufficient capacity, performance and also with sufficient network bandwidth. Don't forget that for streamed applications, additional NTFS permissions will need to be configured
 - VMware recommends using DFS for application repositories
 - Consider latencies between the ThinApp repository and the virtual desktops being used. Place them as close together as you can on the network, especially when applications are being streamed

Objective 4.1 - Integrate a View Design with vSphere

- Determine the appropriate vSphere infrastructure requirements for the View design
 - What do we need to make this work? ESXi, vCenter Server, DRS and HA are all useful. Also consider the additional functionality that the VDS provides. especially around load balancing and Network I/O Control
- Determine host/cluster/resource pool requirements for the design
 - Remember that with View 5.0 and older, there is a maximum of 8 hosts per cluster when Composer is in use, this goes away with 5.1 if you have an NFS datastore
 - Do you need N+1 resilience in your cluster? Remember to factor that in
 - Calculate the total amount of CPU and RAM required from your capacity planning reports, allowing for 80% utilisation on each host

- Do you need resource pools? Remember that these kick in under host resource contention, so some desktops may constrain other desktops of resource. Do not use resource pools as a means of organising your desktops! This is not what they are for!
- Resource pools may become part of the design however if you have a VIP group of users that you want to provide the best possible experience to. Keep in mind though that resource pools have no influence unless the cluster is resource constrained
- Ensure vSphere infrastructure will meet established performance requirements for the design
 - Remember to go back to your capacity planner reports
 - Ensure there is enough CPU, RAM, network and IOPS to cover what is in use already on the physical desktops
 - Remember to add in some slack for N+1 redundancy plus 80% utilisation per host
 - Use all features available to you to improve performance, such as VAAI, VASA and the Storage Accelerator from a storage perspective
 - If you are using 10Gbps Ethernet, use Network I/O control to ensure virtual desktops have enough bandwidth when the link gets saturated
 - Local storage may also be useful in terms of performance. For example, if you are making heavy use of disposable disks there is a high I/O overhead because they host paging files and temp files. These could easily be hosted on local storage, such as 10K drives and this will help keep costs down and make use of existing resources where budgets are tightly constrained
- Determine appropriate storage sizing for the design
 - Use the capacity planning reports again, also ensure you don't stack too many VMs on a single LUN
 - Also, do not place virtual desktops on the same LUN as vCenter server or Composer, this may result in disk contention
 - Use Composer where possible to reduce the storage footprint
 - Ensure there is enough slack for any future planned growth (this should be defined at the outset)
- Ensure vSphere High Availability configuration meets View design requirements
 - Are you planning for N+1 or N+2 resilience?
 - What about admission control? Or will you just try to power on all virtual desktops regardless of free resources?
- Ensure vSphere network configuration meets View display protocol requirements
 - Ensure there is no single point of failure by utilising multiple uplinks into multiple physical switches
 - Use the GPO template to tune the PCoIP protocol
 - No need to make any changes on the corporate LAN, as PCoIP will self adjust according to available bandwidth

- On the WAN, disable build to lossless, use client side caching and implement maximum frame rates, maximum audio bandwidth to help improve performance and reduce bandwidth requirements

Objective 4.2 - Integrate View Design with Infrastructure Services

- Identify the features that vCloud Director uses to provide multi-tenancy
 - vCD uses such things as Organisational DCs to provide segregation between tenants, including the use of Private VLANs to restrict the flow of network traffic
- Describe the way logging works in a multi-tenant environment (vSE, vCD, vSphere etc.)
 - Not sure what they are asking for here and why it has anything to do with View
- Explain user access and network isolation between organisations
 - As each vCD cell connects to a vCenter, you can configure Active Directory connections and authentication
 - Again, deploy Private VLANs if you need to segregate out traffic in a granular manner
- Determine appropriate Active Directory configuration for the design
 - Keep View desktops in their own OU for delegated management but also for application of View specific GPOs
 - Use groups for access to pools and especially for delegated management of administrative roles
- Based on customer requirements, identify Smart Card / RSA configuration
 - Verify that the proposed solution is supported with View
 - Install any software and drivers on the virtual desktop and also at the View client end (if only a subset of users require a smart card, consider using a separate pool with it's own master image)
 - In the Connection Server settings, configure the smart card policy (card required, what do when smart card is removed etc)
 - SecureID can be configured to only use specific Connection Servers, these may be used when a connection is coming in from the internet, for example
 - Know the workflow order and the steps involved with implementing RSA secure tokens, for example. This may be a key project deliverable.
 - Know what the key files are for an RSA integration, including what the RSA architecture is and the role of files such as sdrec.conf
 - Decide how RSA will be used, for example will logons match AD? Is RSA mandatory?

Objective 5.1 - Create a Physical Design for View Infrastructure Storage

- Size the Events database
 - As before, quite small depending on how much information is kept. In general terms, around 200MB
- Size the View Composer database

- The composer database is very small and does not contain much information, 100MB seems like a sensible upper limit, based on some forum postings I found
- Determine storage requirements for the Transfer Server repository
 - The transfer repository holds the compressed and encrypted base images for View Composer, so if there are only a few images in use, this should be quite small, as sparse disks are used
 - The repository is only required if linked clones in local mode are in use
 - Each Transfer Server can perform up to 60 simultaneous operations, so keep this figure in mind when sizing your Transfer Server layer
- Determine storage requirements for infrastructure virtual machines (eg. vCenter servers, View servers etc.)
 - Decide how many components you need, based on customer requirements and also the capacity planning metrics. Don't forget to include considerations for things such as redundancy as well as configuration maximums
 - AS a minimum, you will need a vCenter Server, database (preferably on it's own server) and Connection Server
 - Optionally, you may require any of the following :-
 - View Composer
 - Security Server
 - vShield Endpoint Manager and virtual appliances per host
 - View Transfer Server
 - Additional Connection Servers
 - Product documentation will tell you how to size these virtual machines, including CPU, memory, network and storage requirements

Objective 5.2 - Create a Physical Design for View Pool Storage

- Determine the base image/template requirements for the design
 - Which operating systems do we need to support? Windows XP? 7?
 - 32 bit or 64 bit or both?
 - Multi-language settings?
 - Any embedded applications required, such as MS Office?
 - Do you need different base images for different departments?
- Establish replica requirements
 - Which storage tier does this need to be on? SSD? Is there a flash cache based LUN we can use? Should always go on the fastest disk you have because of high amount of disk reads
- Identify linked clone requirements (eg. OS disk)
 - How many linked clones per pool are we having?
 - How large do you expect the OS disk to grow? If you have a developers group that installs a lot of applications on a dedicated desktop, this might grow quickly
 - VMware recommend a maximum of 64 to 128 linked clones per LUN, though you can go a lot higher than that. Depends largely on the user workload, so try and group users with similar requirements into the same pool.

- Determine the amount of estimated data growth and the impact to the design
 - Remember that user data will be stored on their home drive, so unless a lot of applications are installed that are not in the base image, data requirements should not grow excessively
 - Remember that we can place linked clones on slower, cheaper disks which mean that costs need not be driven up
 - One way to estimate growth would be to look at the size of the current base image (if it exists) and compare this with the size of the OS drives obtained from the capacity planning report. Crude, but useful.
 - The impact would be that as more changes are made outside the base image, so storage requirements go up
 - If you have a high proportion of users using stateless desktops with persona management, the growth factor should be quite low
- Establish persistent disk requirements
 - Persistent disks are used with linked clones to preserve user data when images are refreshed or recomposed.
 - To discover if persistent disks are required, it is worth looking at the capacity planning analysis, especially if there are metrics in there for disk usage. Generally user data would be stored on the network where it can be backed up, but power users with large profiles may benefit from persistent disks
 - Where users need to install their own applications and preserve changes, persistent disks may be a good idea
- Establish disposable disk requirements
 - Disposable disks are generally used for temporary files such as paging files
 - These may be useful where large page files need to be configured on linked clones but do not need to remain permanent on logout
 - They also slow the growth of linked clone machines as the clones do not retain temp files
 - If storage space is not high, then this may prove an attractive option
 - Having disposable disks on lower tier storage may also be helpful if I/O is a concern. Reads and writes can be moved away from OS disks on more expensive storage providing a performance improvement at a relatively low cost
- Determine full clone requirements
 - This use case works best when users install software locally. In a linked clone setup, data may be lost when a pool is refreshed or recomposed
 - Look in your workload profiles for power users, it's most likely this use case will benefit most from full clone desktops
 - The main constraint of the full clone is the amount of disk space that is used. If you need to place these clones in Tier1 or Tier2 storage, this may drive up the cost of the solution
- Establish persona requirements (eg central profile store, View Virtual Profiles etc.)
 - Remember the advice from earlier, and size your persona repository appropriately
 - VMware recommend using DFS

- Stripe your repository across four disks, each with their own SCSI controller. In a large environment, this repository will see a lot of disk I/O activity
- Ensure there is sufficient network bandwidth to cope with demand, so a minimum of 1Gbps
- Remember for XP you still need UPHClean, this is built into Windows 7
- Have a strategy around local, roaming and mandatory profiles. Your capacity planner tool may have information on this, try to break down what is in use already in the physical environment
- Are you going to be using stateless desktops? If you are, you need to set aside enough storage space for the users in that workload profile
- Establish if any profile folders or settings are to be taken out of persona management, or if you will use it as the default solution
- How will you need to apply the GPO template? You can do this by OU or embedding in the master image, which may influence your pool design
- For larger deployments, consider the use of multiple repositories but these are tied to pools, so ensure users have visibility of all repositories if they are entitled to multiple pools
- Evaluate what coexistence you need to keep with existing Windows profiles
- Consider the impact of folder redirection on your design
- Remember that you will need the View Premier license to use persona management

Objective 5.3 - Create a Physical Design for Desktop Application Storage

- Identify the applications required for the design
 - Use your capacity planner reports to look at which applications need the most focus
 - Identify any dependencies that may prevent an application from running under Windows 7
 - Profile each application, starting with the most used and see if the application can be virtualised with ThinApp
 - Stakeholder feedback can be very useful on this front as new versions of software may be being implemented or replaced at the point where Windows 7 or View is deployed, so it may be you need not spend time gathering profiling information for an application
 - Break out applications into their user profile use case - so for example if you have an application that road warriors use on their Local Mode desktops, does that place a constraint on how you package the software?
- Determine the appropriate deployment method for each application (eg ThinApp Streamed, ThinApp deployed, natively installed etc.)
 - Streamed apps generally work best for large ThinApp'd applications, such as MS Office, Visio or Photoshop (for example)

- Where applications have a smaller footprint, both in terms of storage space and changes made in the registry (such as Firefox or 7-Zip), these may be best deployed
- Where there is limited bandwidth between the desktops and the ThinApp repository (for whatever reason), it may make sense to natively install the application. This should always be considered a last resort though, as the base OS image should really be kept as pristine as possible for ease of management and patching. Introducing applications into the base image may provoke DLL conflicts or such things
- Establish application capacity requirements
 - The capacity planning metrics should be able to help here in telling you how much disk space an application uses up. Remember that if you natively install a lot of applications into your base image and you use full clone pools, you will require a lot of storage which in turn will drive up costs

Objective 5.4 - Determine Performance Requirements for a View Physical Storage Design

- Establish IOPS requirements (eg. peak, average, average peak etc.)
 - You will need capacity planning reports to obtain this information, which will help you profile workloads (task workers, knowledge workers etc) as well as helping you calculate the total number of IOPS required for the proposed solution
 - The formula used is **Throughput (MB per second) x 1024 / Block Size (KB) = IOPS**
 - Task workers generate on average 5 IOPS, whereas power users can generate anything up to 20-30 IOPS depending on the workload
 - SSDs average around 6000 IOPS
 - 15K disks average 175 IOPS
 - 10K disks average 125 IOPS
 - And don't forget the RAID penalty which also should be factored in
 - Don't forget also that Windows 7 uses a 1MB block size whereas XP uses a 64KB block size
- Establish throughput requirements
 - Go back to capacity planner and obtain this information, there is no way you can guess this!
- Determine storage network requirements
 - Generally you would want to use fibre channel if you can, but if you need to keep costs down (or have other reasons) then you can use NFS or iSCSI, both of which are fully supported
 - If you aren't using 10Gbps Ethernet, you will never get more than 1Gbps throughput as NFS/iSCSI will only use a single NIC
 - That being said, you can implement Jumbo Frames (must be end to end from guest through to switch port) which will provide improved throughput and lower overhead on the host and network switch

- Try to ensure hosts and storage are on the same IP subnet to reduce network hops and thus reduce latency
- You should also have fully redundant paths for the storage, using multiple storage processors or multiple physical switch uplinks if using IP based storage protocols

Objective 6.1 - Create a Physical View Client Design

- Determine the number and types of access endpoints required for the design (eg. zero client, thin client, fat client etc.)
 - Although you may be able to get this from a capacity planner report, you will most likely obtain this information from a stakeholder interview
 - This is because thin clients will be introduced with View and don't already exist, so it may not necessarily go that the number of physical desktops now will correlate with the number of thin clients being introduced
 - Remember that each type of device carries their own logistics, for example
 - Thin clients will need firmware updates, how do you handle this?
 - If old PCs are re-purposed as thin clients, what about Microsoft patches
 - Some zero clients will place a constraint on the design as there will be a finite number of options for use - i.e. connected devices etc
 - Can you still perform a single sign on, or passthrough logon?
- Identify security requirements for the View Clients
 - Do you need to implement smart card authentication?
 - Will clients be connecting from the internet and need to pass through a Security Server to access their desktop?
 - Do you need to enforce certificate checking?
 - Will certain user groups be exempt from accessing desktops from the internet? Tags are useful for configuring routes for users and their access needs
- Determine connectivity requirements for the View Clients
 - Presumably this refers to what network bandwidth is available to clients
 - Use the capacity planner reports to see the bandwidth used and scale the clients accordingly. I don't know, I'm just guessing here
 - Bear in mind any smart card reader devices and also swipe card readers
- Establish multimedia capabilities
 - Establish if there is a need for streaming video and/or audio, if so, discover what kind of quality is required
 - For example, there may be a need to listen to some brief audio from a voice mail system on an irregular basis, this does not necessarily need a lot of bandwidth
 - However, if video training is regularly used, you may need to tune bandwidth settings when network congestion occurs or if the link speed is slow to the View client
 - If content is Adobe Flash based, will you need to throttle the traffic on high latency low speed links?

- Also, if content is not Flash but say a .MOV file, what will you need in the design in order to take account of this requirement?
- Establish peripheral requirements
 - As mentioned previously, verify what devices are required and how you will set policies to affect this
 - USB cameras
 - Memory sticks/portable storage
 - Smart card readers
 - USB printers
 - With some of these devices, some vendor specific software may be required over and above drivers, so you need to establish who needs this
 - There may be a security requirement to allow only certain types of USB devices, so you need to consider how you are going to configure GPOs with device identifier strings to only permit a certain type or vendor of USB device to be connected

Objective 6.2 - Determine Session Connectivity Requirements for a View Client Design

- Establish session mobility requirements (eg. Follow me etc.)
 - Is follow me printing required? If so, you will need to perform some integration work for location based printing
 - What about users with tablets or mobile devices that need to reconnect to sessions on the move?
 - When a user disconnects, decide what the requirements are on dealing with that, so for example will you force the user to logoff? Or if they are connected via iPad with intermittent coverage, will the user need to be able to reconnect to their existing session without losing any work?
- Establish remote access requirements
 - This is another instance where stakeholder feedback is probably more useful than capacity planning, and would come out of a requirements analysis phase
 - Identify if there are any users who will access View from outside the corporate LAN
 - Identify if you need to deploy a Security Server infrastructure and how the network DMZ is configured
 - Are there any firewall or network security policy constraints?
 - What protocols are required? For example, certain View Clients do not provide PCoIP support, this may need to be factored in. Also, serial port connections may require an RDP connection
- Determine applicable session behaviour requirements for the design (eg. disconnect, logoff, timeout)
 - Stakeholder feedback will define how this will be configured
 - Security policies may dictate that users are automatically logged off if they disconnect a View session for any reason

- Much depends on security policies but also resource requirements - if non persistent desktops are in use, it makes no sense to preserve the session over time and tie up CPU and memory
- Consider the implications on licensing if you need to spin up more desktops than you are licensed for. Remember it is concurrent licensing, so you may be able to meet requirements by provisioning more desktops and still be in license compliance. For example, you may have a requirement for resources to always be available, but in a school lab, you may need to logout and refresh the desktops of each user every hour, when a new user comes along in the next hour, how will you satisfy their need for a virtual desktop immediately?
- Identify display protocol requirements to satisfy the design
 - Identify if there is any constraint or risk preventing you from using PCoIP
 - Where there is restricted bandwidth, you will need to make changes in the GPO template to disable build to lossless and also reduce audio bandwidth to help improve performance

Objective 6.3 - Determine Management Requirements for a View Client Design

- Determine patching requirements
 - How will you keep View clients patched?
 - With thin or zero clients, this tends to be more aligned towards physical hardware firmware updates and can be sent out from a central point. Wyse Device Manager is a good example of this
 - You can use the likes of Altiris or SCCM to patch clients on the corporate LAN, as you'd patch any other enterprise application
 - For the likes of iPad/Android, these are usually updated from AppStore/Google Play and require no administrative intervention
 - What if you have offshore users? If you have no enterprise solution that reaches these users and you have firewalls and security in front of them, you could permit access to the View portal, where they can obtain the latest client themselves
- Establish software distribution requirements
 - Much the same as above, you could deploy the View Client using an enterpriser solution, or if that is not possible, using the View portal and let the user self serve
- Identify client peripheral requirements
 - Establish if the likes of USB connectivity is required, and how this might impact your security design. Remember it is possible to restrict types and vendors of USB devices, this may be important
 - Establish any requirements for the likes of multiple screens or displays
- Establish security requirements
 - Much as above, if you have a security that disallows USB devices to be connected, or does not permit cut and paste, this needs to be factored into your design

- Again consider using tags if you have subsets of users that only have access to certain pools and desktops if security policies mandate this