



Do It Yourself End-User Computing

Version 1

created by Tony Foster on Aug 13, 2014 3:58 PM, last modified by Tony Foster on Sep 11, 2014 11:11 AM

Interested in building an End User Computing solution on your own, getting your hands dirty, and intimately knowing all the parts? You've made a good choice and its clear you like to understand everything happening in your environment. In the following sections we hope you can find some great information on your journey.

As we take this trip into building a VDI solution example scenarios are provided. These examples do not come from any one customer but come from almost 10 years of designing, implementing, and troubleshooting virtual environments. Any similarities are just a coincidence. This is also a very short version of getting started building a VDI solution. To get all the information to build a VDI solution would take at least one book and probably more.

Goal

A good place to start is by documenting the specific things you want to achieve from your EUC solution. The more specific you can be the better your results will be. For example "I want to consolidate IT resources and reduce desktop costs..." is a really generic goal. A better example would be "I want to consolidate all computer labs in our three largest high schools into a VDI solution to foster a better learning environment and improve lab uptime to 99.99%" is much more specific and can help you reach your goal quicker.

You can then take the that goal and start breaking it down further. An example of some questions would be:

- How many users per site?
- How much bandwidth do I have between sites?
- How will these desktops be used?

As you keep breaking down the questions you will find the you develop a very complete list of criteria for testing and implementing your EUC solution. In addition to this you are able to use this to help prevent project creep so that you can get a solution up and running faster for less money. If you have an awesome vendor you like to work with they may even be able to help you develop and flesh out the goals of your implementation.

Parts List - Part I

When you know what you want then you can start to put together the infrastructure you need for your solution. Some of the areas you may want to include in your list are the following:

- Storage
 - For user data
 - For desktop images
 - For users persona
 - For supporting VMs
- Compute
 - Desktops
 - Management
- Hypervisor
- Networking
 - To connect the physical servers for the project
 - To connect the internal users to the virtual environment
 - To connect users outside the organization to the virtual environment.
- Security Appliances (if applicable)
- Backup (BC/DR)

Parts List - Part II

Now how do you figure out how much of all of this stuff is needed? The easiest way I've found is to use a building block approach. A good way to do this is look at reference architectures (RAs) companies have built and made publicly available. A good place to find these is the [Consolidated List of Current EMC VDI Reference Architectures](#). Here you can see how solutions have been designed and sized for a specific application.

It's also a good idea to look at the compatibility guides for your virtualization product. No one wants to plan out an entire implementation only to find out that the hardware is not supported with the hypervisor you want to use.

Definition

Reference Architecture (RA): Is a document that describes a previously built and validated solution.

Using the reference architecture from above, EMC solution partners, the virtualization community, and other tools freely available on the web you can start to figure out how many and what types of hardware you might need. Doing this helps solidify your design. It also prepares you for the next step.

Continuing the example from above it may only be feasible to do a single school at a time and each school has 450 lab seats. So it may make sense to start with an array that can be used for multiple uses, such as a VNX. This could allow you to provide storage to many different projects from a single array. Making it possible to do more with less and stretch IT budgets farther than before. Maybe you can do all three schools at once and it makes more sense to use an all flash based array, such as a XtremIO, to house all of the desktops for the schools. It just depends on what is best for your needs. This is only an example and the design can and should vary depending on your needs, not the sales persons.

Some Assembly Required

We've figured out what parts we need and all the math works out. The team feels really accomplished. All the parts are sitting on the shipping dock waiting for us. Its time to start building our solution. This is where the rubber meets the road. Before you start putting everything together here are a few reminders.

- Do you have enough power in the rack or DC to support your new hardware? There is nothing more painful than getting all the new hardware racked and powered on and then tripping breakers.
 - If installing new racks for your VDI project does the power come in from the top of the rack or the bottom? Can the power distribution panels (power strips) reach the power drops?
- Do you have all the network drops you need to get everything on the network? Often times its a good thing to count all of the network ports on the equipment before you start racking just to be sure you have the right number of drops.
- Do you have all the tools you need to rack all of the equipment? Screwdrivers, crescent wrenches, drills, socket set, etc.
- Do you have all the licenses? Hypervisor, management VM's, desktop licenses, etc.
- This sounds obvious, will the rack and all of the other equipment fit through the door?
- *If you have a good one be sure to add it in the comments below.*

Once you have everything racked, powered on, and burned in you've tackled the easy part. Now its time to start installing the software and testing. The procedures for this vary from product to product and its best to consult the install guides specific to your hypervisor. It's also worth noting you may need special builds or drivers to take advantage of all the options your compute hardware has to offer.

Where's my Desktop?

With the hypervisor installed and VDI almost a reality there are a few more things you may want to consider. In all but the smallest VDI deployments you will find that management servers (vCenter, etc.) will be given dedicated resource reservations or separate hardware. Some will say this doesn't make sense. The reason you do this is to prevent a chicken and an egg scenario.

Going back to the school example. Lets say the desktops and the management VM's were all put together and there were no reservations, separate pools, or separate hardware. Then lets say a few students stand up a game server on their desktops, a few more are streaming some YouTube Video's, and several others are compiling some code. Since all of the resources are shared equally amongst the desktops and the management servers a DOS attack has just inadvertently happened. All of the resources are being consumed by the desktops. The management systems can't get any resources to do their management and they stop responding. You now cant get in to manage the desktops and figure out what's happening. For this reason and many others separate your management applications from your desktops. Everyone will thank you later.

By now desktops should be up and running and everything is function as you would expect it. Time to test and make sure it will run as you expect it to. Grab a VDI session and start using it just like you would expect anyone else to. It's actually very important that anyone who implements a VDI solution also get to use it. There are several reasons for this. First is if there are any performance issues you will know about them. Secondly it shows that you are willing to use what you recommended others to use. And lastly you get to use all the great features that a VDI solution brings to your organization.

To Be Continued

Now that your VDI solution is up and running you can sit back and relax. Until that next set of patches come out. Then you need to test and verify that they wont crash your VDI environment. Or wait awhile and let others take that risk before you update your systems. Other than that you are set and ready to go.

Conclusion

There is a lot that goes into building your own VDI solution from the ground up. There are lots of resources out there to help you develop your VDI solution so that it meets your needs and creates a win for your organization. If you don't have time or think you could make better use of your time you may want to look at other ways to get a VDI solution up faster. This could mean engaging an EMC partner and implementing a [VSPEX](#) solution. Or maybe you want a fully converged solution that is tested and ready for you to use it when you wheel it into your datacenter. In which case you might want to look at a [Vblock](#). Or if you don't need the ability to expand your VDI environment at a moments notice you may want to consider a [Desktop as a Service \(DaaS\)](#) solution.

Obviously this isn't a comprehensive guide to do it yourself VDI. It's just a starting point to provide you with some information on how to do it yourself. If there are other important points (I know I left out a lot for the sake of space) add them in the comments section. If you have additional questions about VDI please use the forums or reach out to your EMC representative.

269 Views

Categories: End User Computing , vSphere Integration

Tags: xen , citrix , view , horizon , euc , vdi

(0 ratings)

0 Comments

There are no comments on this document.