VMware Virtual Machine Testing

With SANBlaze VirtualAware Emulation

A SANBlaze Whitepaper

SANBlaze Technology, Inc.

February 2014

Abstract

SANBlaze Technology introduces VirtualAware^{TM} technology for VMware performance and capacity planning.

With increased use of VMware virtual machines running on ever more capable multiprocessor machines, it is trivial to create a scenario where the storage requirements of the virtual machines exceed the capabilities of the storage system.

Mismatches in performance capabilities between VMware host machines and the storage subsystems hosting their respective Datastores become particularly acute in the capacity planning and test environment, where the goal is to maximize the number of virtual machines per physical multiprocessor initiator.

Increasing the performance of the storage subsystem in an enterprise environment may be accomplished by simply "throwing money" at the problem. Large, highly capable storage arrays with multiple high bandwidth 10G - 40G iSCSI or 8G - 16G Fibre Channel interconnects provide the necessary bandwidth and performance to satisfy the requirements of high-end VMware initiators for the purpose of capacity planning and performance test, but at a price point well out of range for most testing scenarios.

SANBlaze has addressed the qualification and capacity planning for this high-end VMware test environment with the VirtualAware Storage Emulation System. VirtualAware technology leverages the industry-leading storage emulation technology of SANBlaze's VLUN product, providing high performance emulated storage, by adding support for virtual VMware Datastores.

VirtualAware technology is VAAI (vStorage API for Array Integration) capable, supporting full hardware acceleration to VMware. VMware "clones" are deduplicated automatically by the instantaneous SANBlaze cloning process, and completed in a fraction of the master’s storage space. For example, an 8GB Windows Virtual Machine can be fully cloned in as little as 100MB, allowing many virtual machine storage devices to live in RAM and operate at line speed.

Tests that stress disk access, latency and bandwidth, such as a virtual machine "boot storm" can be completed in a fraction of the time at a fraction of the cost of previously available methods, for example.

Table of Contents

Abstract 2

Table of Contents 3

List of Figures 3

Virtual Storage for VMware Testing 4

How it Works 5

Line Rate Testing 5

Configuring a VirtualAware SANBlaze System 6

Create the Master LUN on the VirtualAware System 6

Creating the Slave Virtual Machines 10

Saving the Entire Configuration 16

Saving an Individual VirtualAware Configuration........................16

Conclusion 16

List of Figures

Figure 1: Create a LUN for the Master Datastore 7

Figure 2: Create the Master LUN Datastore 8

Figure 3: Mount the Datastore to VMware and Install Guest OS 9

Figure 4: Creating Virtual Snapshot Datastores 11

Figure 6: Configured VirtualAware Datastore 14

Virtual Storage for VMware Testing

This whitepaper introduces the concept and practice of Virtual Storage for VMware performance and capacity planning testing, and the VirtualAware Emulation product from SANBlaze Technology, Inc.

If you are unfamiliar with SANBlaze and the Virtual Storage products they offer, you may be inclined to ask "What is Virtual Storage, and how would I use it?"

SANBlaze VLUN products, with their Virtual Storage, provide a storage subsystem for the purpose of initiator/host testing that looks to the initiator exactly like a "real" disk, except the (virtual) disk is RAM based, and therefore performs line rate transactions with near zero latency.

All user data can be retained by the virtual storage system. Virtual disks can be formatted, mounted and used exactly as one would use a physical disk, with the ability to vary latency, inject errors, present thousands of individual disks, emulate arrays, change operation modes from disk to tape, target to initiator, change disk/tape/SSD attributes and many other operational parameters to aid in the qualification and scalability of initiators and hosts.

VirtualAware emulation brings SANBlaze's leading virtualization technology to the VMware capacity planning and performance test environment.

Using VirtualAware Virtual Emulation, a VMware Datastore is created on a virtual storage device presented from the SANBlaze VLUN system to a VMware virtual machine. Once created, this "Virtual Aware" Datastore can instantly replicate Virtual Machines using the vCenter "Clone" capability. Each clone appears identical to the master, and is capable of saturating multiple 10G Ethernet connections or 4, 8 or 16G Fibre Channel connections.

Connections from the VMware system to the VirtualAware Emulation System can be over iSCSI, SAS, FCoE or native Fibre Channel.

How it Works

The VirtualAware Emulation System is based on the industry proven SANBlaze Target Emulation System with enhancements to optimize VMware capacity planning and performance testing.

VirtualAware allows a VMware Datastore to be created on a virtualized disk presented to the VMware initiator. This disk can be connected via iSCSI, FCoE or a native SAN interface and it will appear to VMware like a native "raw" disk.

The VirtualAware System supports all VAAI optimizations including: Write Same, Extended Copy, Receive Copy Results, Unmap and Compare and Write. The virtual Datastore will be seen as "hardware accelerated" to VMware vCenter.

In addition, the SANBlaze VirtualAware LUN will deduplicate any time a vCenter clone operation occurs, enabling an entire virtual machine to be cloned without any appreciable increase in consumed storage.

Line Rate Testing

The SANBlaze VirtualAware System provides significant configuration and performance advantages to the capacity planning and test environment. After the original virtual machine has been created, it can be cloned multiple times in RAM without exceeding the RAM space available in the SANBlaze emulator.

For example, a single Virtual Machine running Windows(TM) can be created and then instantly replicated to provide additional Windows Virtual Machines each with their own independent virtual disks.

Since all VirtualAware disks reside in memory, with the ability to be backed up to disk files on the SANBlaze appliance, the Virtual Machines see Datastores with low latency and with line rate (10G / 8G FC) performance.

Configuring a VirtualAware SANBlaze System

Configuring a Datastore on a SANBlaze Emulation System consists of a few simple steps:

1. Create a LUN of size N, large enough to contain your Virtual Machines
2. Use VMware to create or copy your Virtual Machine to the LUN created in step 1. above
3. Use the vCenter clone command to copy the Virtual Machine

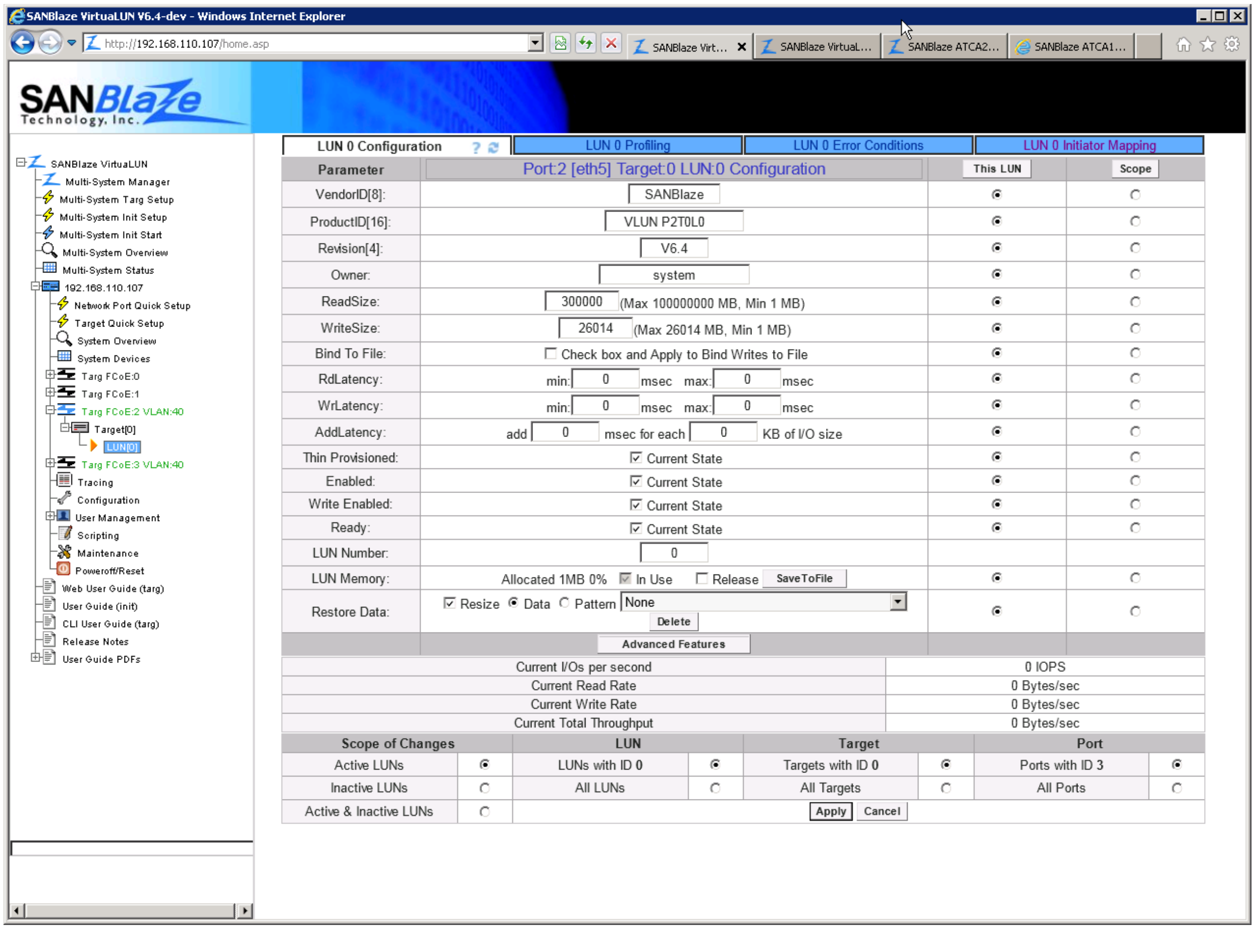
Create the Master LUN on the VirtualAware System

Create a Virtual LUN that will become the Master Virtual Machine. This master will be the basis for all clones in the next step, so it should be as complete as possible. If you want all your virtual machines to have VMware tools, install VMware tools now to the master so that subsequent clones will be complete.

From the SANBlaze system, choose any port that has a physical connection to your VMware Initiator, and create a LUN large enough to contain the Datastore for your VM and its guest operating system. The example uses 26GB.

You should make the Virtual LUN as large as possible, using all available memory in the system, so that it will be able to house as many virtual clones as possible.

Please note that the "thin provisioned" box must be checked and the page must be applied. Ensure the LUN is enabled, write-enabled, and ready.



Parameter	Port:2 [eth5] Target:0 LUN:0 Configuration	This LUN	Scope
VendorID[8]:	SANBlaze	●	○
ProductID[16]:	VLUN P2T0L0	●	○
Revision[4]:	V6.4	●	○
Owner:	system	●	○
ReadSize:	300000 (Max 100000000 MB, Min 1 MB)	●	○
WriteSize:	26014 (Max 26014 MB, Min 1 MB)	●	○
Bind To File:	□ Check box and Apply to Bind Writes to File	●	○
RdLatency:	min: 0 msec max: 0 msec	●	○
WrLatency:	min: 0 msec max: 0 msec	●	○
AddLatency:	add 0 msec for each 0 KB of I/O size	●	○
Thin Provisioned:	☑ Current State	●	○
Enabled:	☑ Current State	●	○
Write Enabled:	☑ Current State	●	○
Ready:	☑ Current State	●	○
LUN Number:	0		
LUN Memory:	Allocated 1MB 0% ☑ In Use □ Release SaveToFile	●	○
Restore Data:	☑ Resize ● Data ○ Pattern None Delete	●	○
	Advanced Features		

Current I/Os per second	0 IOPS
Current Read Rate	0 Bytes/sec
Current Write Rate	0 Bytes/sec
Current Total Throughput	0 Bytes/sec

Scope of Changes		LUN		Target		Port	
Active LUNs	●	LUNs with ID 0	●	Targets with ID 0	●	Ports with ID 3	●
Inactive LUNs	○	All LUNs	○	All Targets	○	All Ports	○
Active & Inactive LUNs	○	Apply Cancel					

Figure 1: Create a LUN for the Master Datastore

VirtuaLUN LUN setup page for VMware testing.

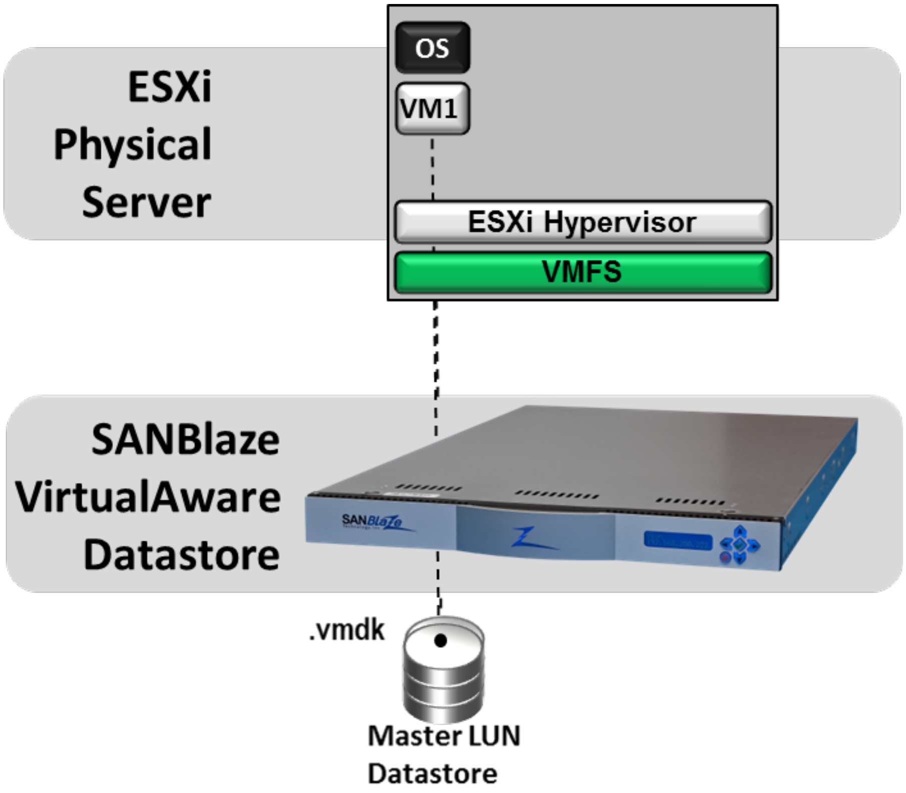


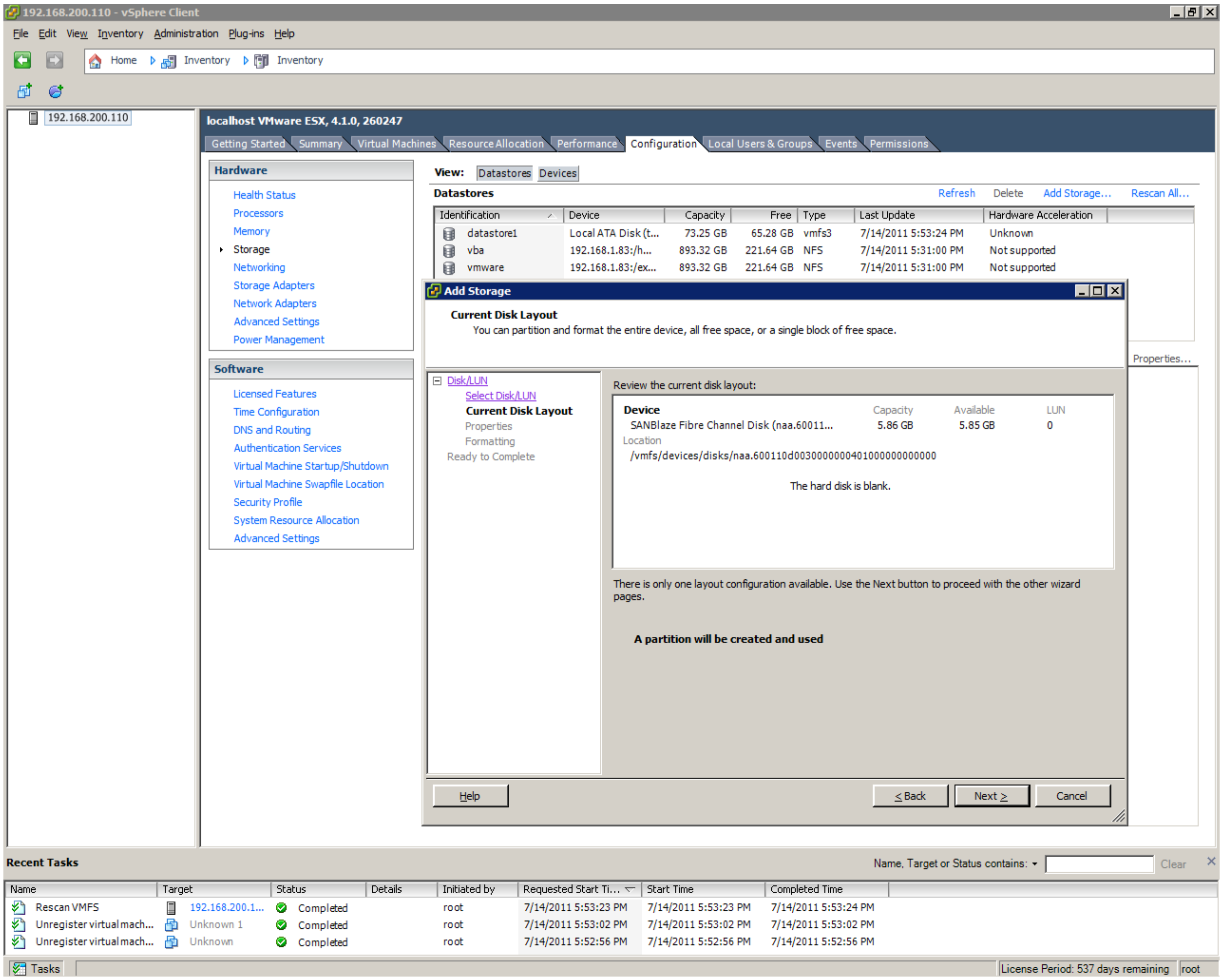
Figure 2: Create the Master LUN Datastore

In the example above, a 26GB LUN has been created at Port2 Target0 LUN0.

When the LUN is created, VMware will be able to access the LUN for use as a Datastore. You may have to initiate a "rescan" for VMware to discover the LUN, by using "rescan all" in vSphere or with the CLI command:

```
esxcfg-rescan -A
```

Use VMware's VSphere Client or CLI to discover the VirtualAware LUN and initialize it.



localhost VMware ESX, 4.1.0, 260247

Datastores

Identification	Device	Capacity	Free	Type	Last Update	Hardware Acceleration
datastore1	Local ATA Disk (t...	73.25 GB	65.28 GB	vmfs3	7/14/2011 5:53:24 PM	Unknown
vba	192.168.1.83:/h...	893.32 GB	221.64 GB	NFS	7/14/2011 5:31:00 PM	Not supported
vmware	192.168.1.83:/ex...	893.32 GB	221.64 GB	NFS	7/14/2011 5:31:00 PM	Not supported

Add Storage

Current Disk Layout

You can partition and format the entire device, all free space, or a single block of free space.

Review the current disk layout:

Device	Capacity	Available	LUN
SANBlaze Fibre Channel Disk (naa.60011...	5.86 GB	5.85 GB	0

Location
/vmfs/devices/disks/naa.600110d0030000000401000000000000

The hard disk is blank.

There is only one layout configuration available. Use the Next button to proceed with the other wizard pages.

A partition will be created and used

Recent Tasks

Name, Target or Status contains: Clear

Name	Target	Status	Details	Initiated by	Requested Start Ti...	Start Time	Completed Time
Rescan VMFS	192.168.200.1...	Completed		root	7/14/2011 5:53:23 PM	7/14/2011 5:53:23 PM	7/14/2011 5:53:24 PM
Unregister virtual mach...	Unknown 1	Completed		root	7/14/2011 5:53:02 PM	7/14/2011 5:53:02 PM	7/14/2011 5:53:02 PM
Unregister virtual mach...	Unknown	Completed		root	7/14/2011 5:52:56 PM	7/14/2011 5:52:56 PM	7/14/2011 5:52:56 PM

License Period: 537 days remaining root

Figure 3: Mount the Datastore to VMware and Install Guest OS

Allow VMware to write a vmfs file system to the device and bring the Datastore online.

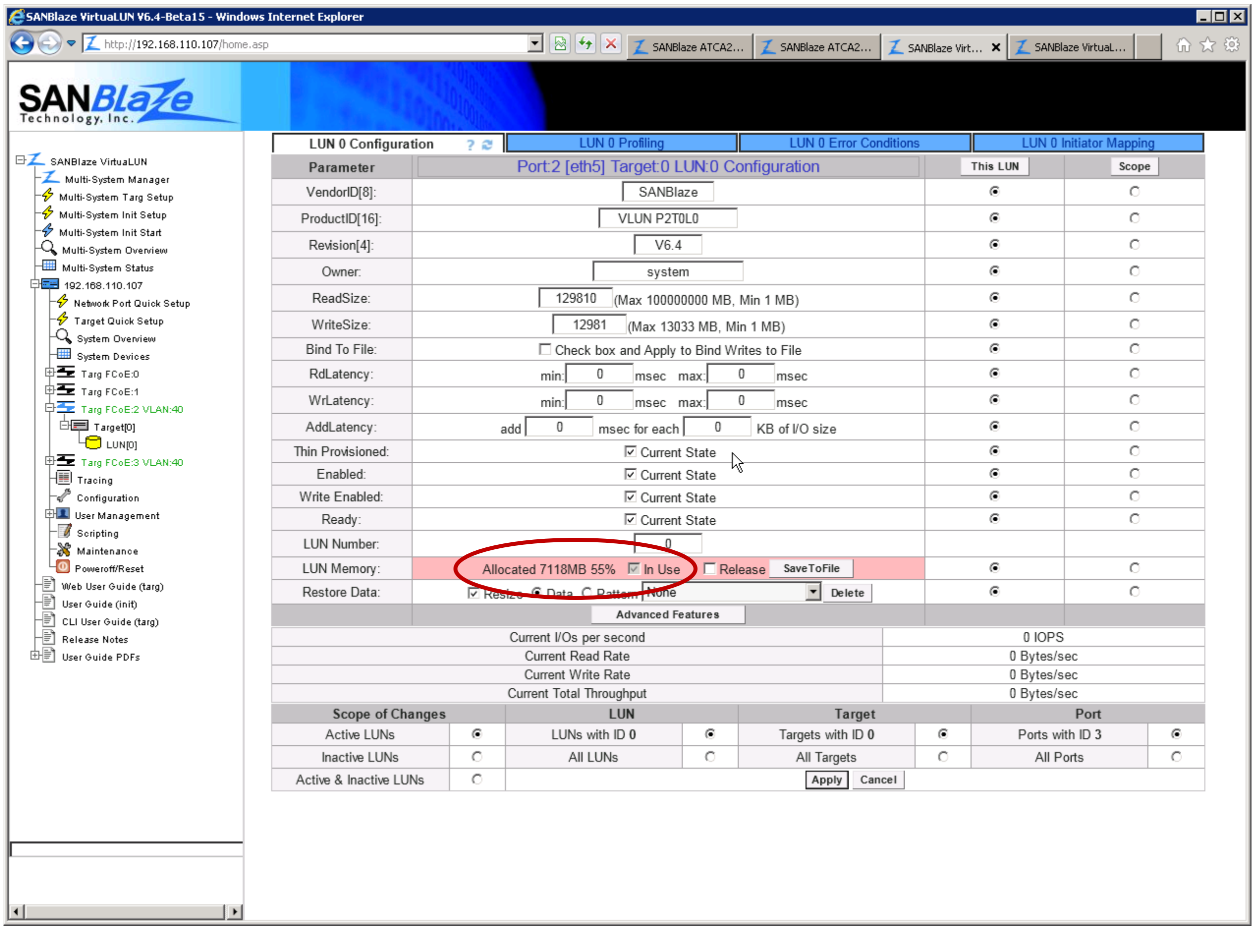
Create a Virtual Machine using this Datastore as the target for the guest operating system and all configuration files. You can use your preferred method for creating the Virtual Machine: installing from a CD made available to the guest machine, using an iso image, or by simply copying a previously created VM to it.

Once you have created this "Master" LUN Virtual Machine's Datastore, power on the Virtual Machine and verify that the functionality is what you want for your test bed machines. If you are doing an installation from scratch (Windows for example), it is a good idea to stop at various points and save the installation to a snapshot file on the SANBlaze system.

To create these check point snapshots, simply use the "Save to File" on the VirtualAware SANBlaze system. Be certain you are not currently writing to the Datastore when you save to file, or the file will have incomplete data.

***Note:* It is recommended to save the Master Virtual Machine LUN to a file before "cloning" additional Virtual Machines.**

A complete installation of Windows7 as a guest operating system as the initial "master" copy consumes 7188 MB of storage space on the VirtualAware SANBlaze LUN.



Parameter	Port:2 [eth5] Target:0 LUN:0 Configuration	This LUN	Scope
VendorID[8]:	SANBlaze	●	○
ProductID[16]:	VLUN P2T0L0	●	○
Revision[4]:	V6.4	●	○
Owner:	system	●	○
ReadSize:	129810 (Max 100000000 MB, Min 1 MB)	●	○
WriteSize:	12981 (Max 13033 MB, Min 1 MB)	●	○
Bind To File:	□ Check box and Apply to Bind Writes to File	●	○
RdLatency:	min: 0 msec max: 0 msec	●	○
WrLatency:	min: 0 msec max: 0 msec	●	○
AddLatency:	add 0 msec for each 0 KB of I/O size	●	○
Thin Provisioned:	☑ Current State	●	○
Enabled:	☑ Current State	●	○
Write Enabled:	☑ Current State	●	○
Ready:	☑ Current State	●	○
LUN Number:	0		
LUN Memory:	Allocated 7118MB 55% ☑ In Use □ Release SaveToFile	●	○
Restore Data:	☑ Resize ● Data ○ Pattern None Delete	●	○
	Advanced Features		

Current I/Os per second	0 IOPS
Current Read Rate	0 Bytes/sec
Current Write Rate	0 Bytes/sec
Current Total Throughput	0 Bytes/sec

Scope of Changes		LUN		Target		Port	
Active LUNs	●	LUNs with ID 0	●	Targets with ID 0	●	Ports with ID 3	●
Inactive LUNs	○	All LUNs	○	All Targets	○	All Ports	○
Active & Inactive LUNs	○			Apply Cancel			

Creating the Slave Virtual Machines

Once the Master Virtual Machine is created on the Datastore, it can be used to create N Virtual Snapshots, or clones, for N virtual machines as shown in the figure below.

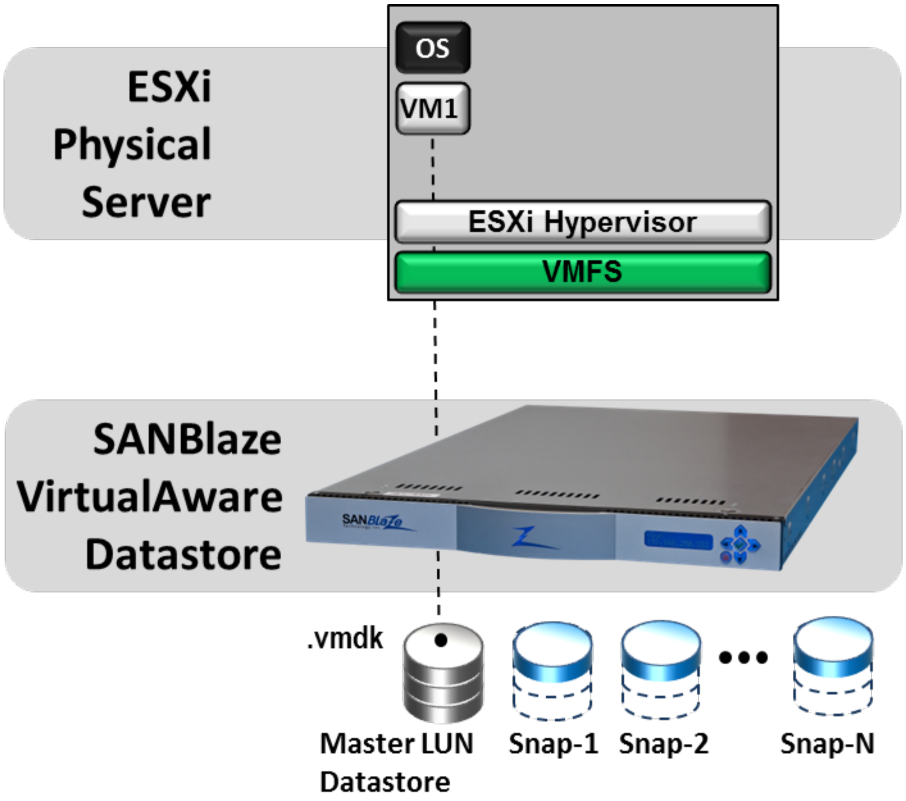
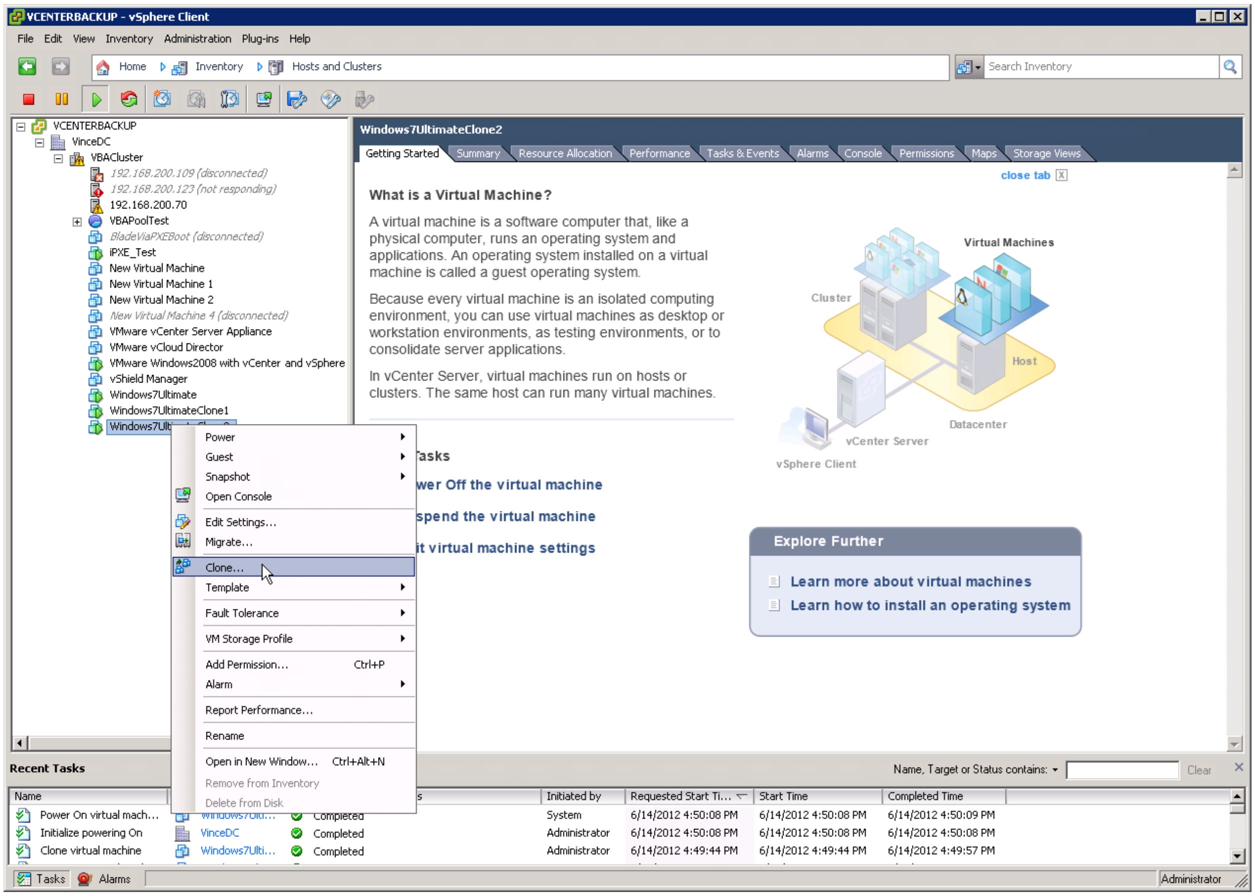


Figure 4: Creating Virtual Snapshot Datastores

From your VMware vCenter, locate the Virtual Machine that uses the SANBlaze VirtualAware device as its Datastore, and use the clone operation as shown in the figure below.

Note that the clone operation is not a native part of ESXi and vSphere. The node must be a member of a vCenter to take advantage of the clone functionality.



What is a Virtual Machine?

A virtual machine is a software computer that, like a physical computer, runs an operating system and applications. An operating system installed on a virtual machine is called a guest operating system.

Because every virtual machine is an isolated computing environment, you can use virtual machines as desktop or workstation environments, as testing environments, or to consolidate server applications.

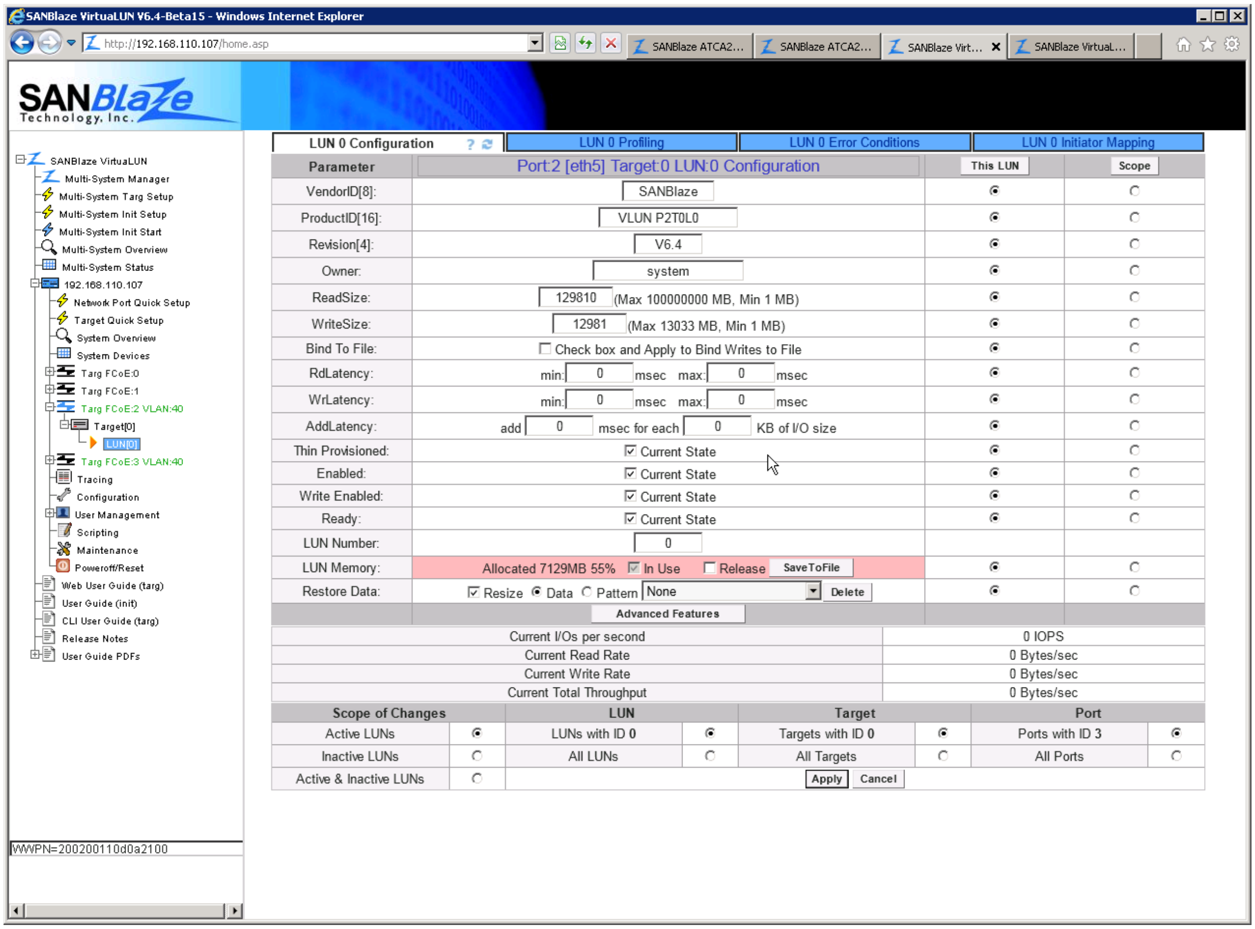
In vCenter Server, virtual machines run on hosts or clusters. The same host can run many virtual machines.

Name		s	Initiated by	Requested Start Ti...	Start Time	Completed Time
Power On virtual mach...	Windows7Ulti...	Completed	System	6/14/2012 4:50:08 PM	6/14/2012 4:50:08 PM	6/14/2012 4:50:09 PM
Initialize powering On	VinceDC	Completed	Administrator	6/14/2012 4:50:08 PM	6/14/2012 4:50:08 PM	6/14/2012 4:50:08 PM
Clone virtual machine	Windows7Ulti...	Completed	Administrator	6/14/2012 4:49:44 PM	6/14/2012 4:49:44 PM	6/14/2012 4:49:57 PM

The clone operation will take advantage of the VAAI primitives implemented on the SANBlaze VirtualAware storage emulator, and the clone operation will occur in seconds.

Using SANBlaze VirtualAware provisioning, the SANBlaze LUN will automatically deduplicate all common data while completing the VMware clone operation, allowing a full clone in a fraction of the original storage space.

As noted above, the first guest operating system, Windows7, required 7118MB; while the clone copy required only an additional 11MB (0.1%) resulting in a total used space of 7129MB as shown in the figure below.



Parameter	Port:2 [eth5] Target:0 LUN:0 Configuration	This LUN	Scope
VendorID[8]:	SANBlaze	●	○
ProductID[16]:	VLUN P2T0L0	●	○
Revision[4]:	V6.4	●	○
Owner:	system	●	○
ReadSize:	129810 (Max 100000000 MB, Min 1 MB)	●	○
WriteSize:	12981 (Max 13033 MB, Min 1 MB)	●	○
Bind To File:	☐ Check box and Apply to Bind Writes to File	●	○
RdLatency:	min: 0 msec max: 0 msec	●	○
WrLatency:	min: 0 msec max: 0 msec	●	○
AddLatency:	add 0 msec for each 0 KB of I/O size	●	○
Thin Provisioned:	☑ Current State	●	○
Enabled:	☑ Current State	●	○
Write Enabled:	☑ Current State	●	○
Ready:	☑ Current State	●	○
LUN Number:	0		
LUN Memory:	Allocated 7129MB 55% ☑ In Use ☐ Release SaveToFile	●	○
Restore Data:	☑ Resize ● Data ○ Pattern None Delete	●	○
	Advanced Features		

Current I/Os per second	0 IOPS
Current Read Rate	0 Bytes/sec
Current Write Rate	0 Bytes/sec
Current Total Throughput	0 Bytes/sec

Scope of Changes		LUN		Target		Port	
Active LUNs	●	LUNs with ID 0	●	Targets with ID 0	●	Ports with ID 3	●
Inactive LUNs	○	All LUNs	○	All Targets	○	All Ports	○
Active & Inactive LUNs	○			Apply Cancel			

Booting the cloned Virtual Machine will result in slightly more space being consumed due to write operations performed as the machine boots, but a fully booted the machine will take only a fraction of the space, and will be a complete copy of the master.

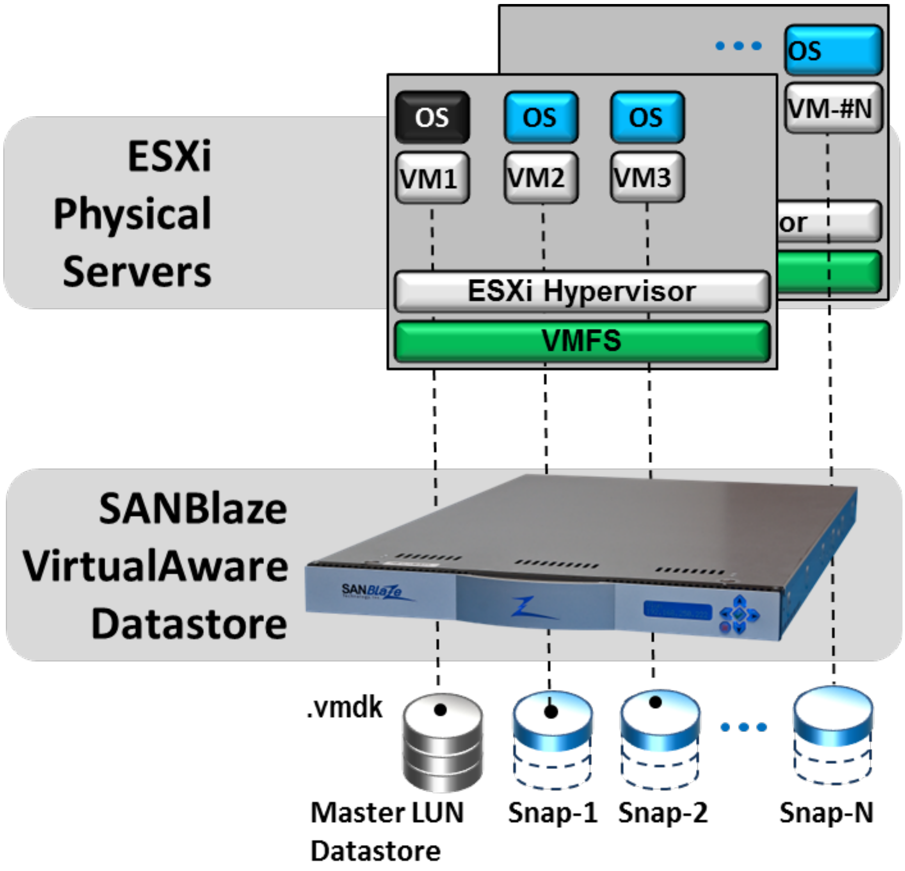


Figure 5: Configured VirtualAware Datastore

VirtualAware technology directs all reads from any Slave to come from a shared memory pool containing the master LUN data, and provides the following significant advantages to the test environment:

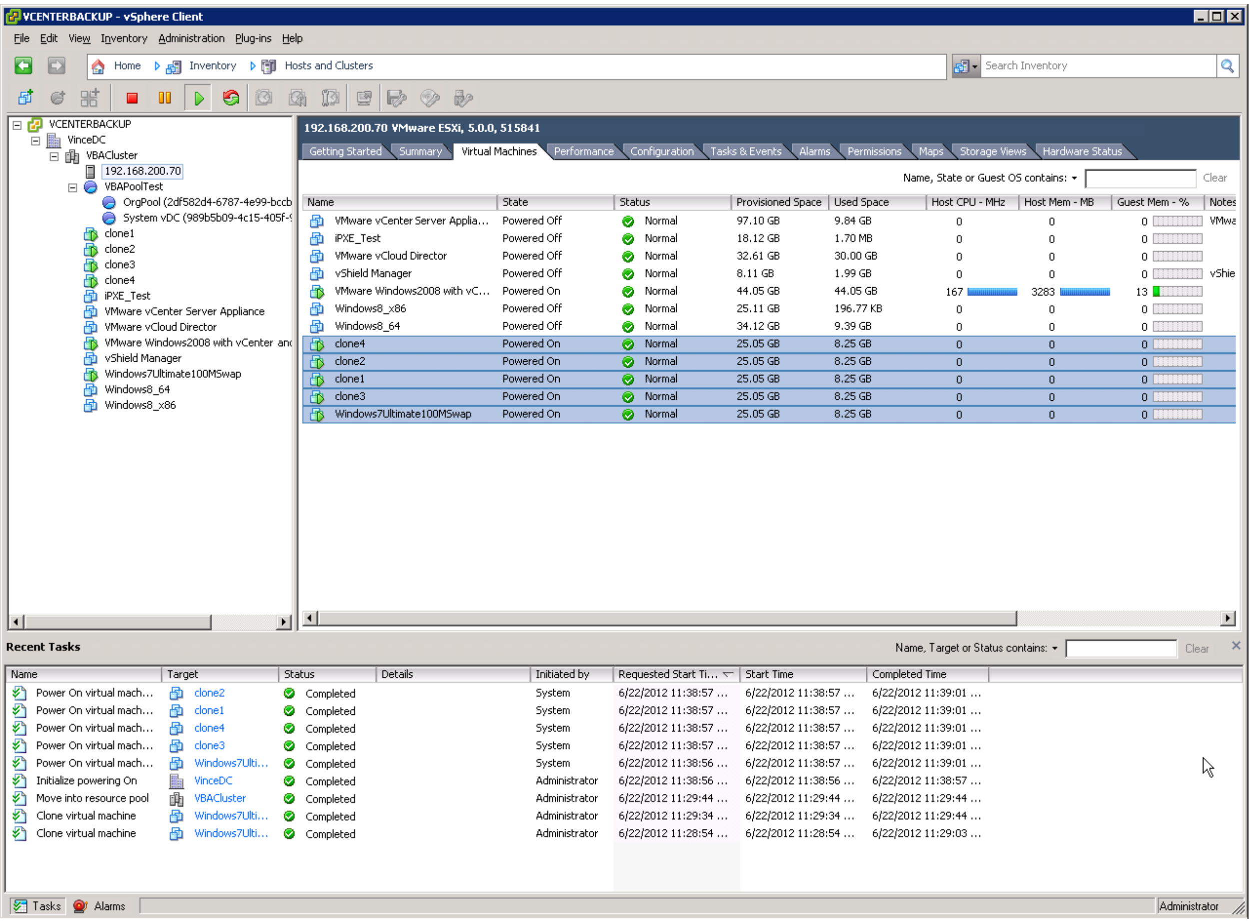
- A single copy of the "guest" operating system can be instantly cloned to several Virtual Machines, automatically de-duplicating any common data between the master and clone copies.
- Each Virtual Machine sees a private copy of its own guest operating system, and is completely independent of all other Virtual Machines
- Each Virtual Machine enjoys RAM speed access to its own guest operating system, limited only by the interconnect speed (currently a multiple 16G, 8G FC or 10G Ethernet) between the VirtualAware Datastore and the VMware Initiator system.

Writes to the VirtualAware system terminate at the individual RAM based clone copy associated with a specific virtual machine, so that each Virtual Machine can be customized without affecting any other. This method has the following significant advantages:

- Each individual VirtualAware based VM appears to the guest operating system as a full installation of a virtual machine, allowing each virtual machine to be customized, becoming a completely independent entity.
- All VirtualAware storage can be saved to file on demand, providing a snapshot in time can be preserved for all N virtual machines
- All virtual machines can be restored to any saved snapshot.
- Customizations to all Virtual Machines can be saved to a named file for permanent retention
- All Virtual Machines can be returned to "factory default" (the master copy) at any time, reverting the test bed to its start of the day condition

The figure below shows the original "master" installation of Windows7 and four fully functional "clones". Each Virtual Machine shows it consumes 8.25GB out of 25GB available. Collectively, the Virtual Machines "see" 125G of capacity, and together consume ~42GB.

In reality, this entire configuration is running on the SANBlaze VirtualAware system using only 7.4G of RAM.



VCENTERBACKUP
VinceDC
VBACluster
192.168.200.70
VBAPoolTest
OrgPool (2df582d4-6787-4e99-bccb
System vDC (989b5b09-4c15-405f-
clone1
clone2
clone3
clone4
iPXE_Test
VMware vCenter Server Appliance
VMware vCloud Director
VMware Windows2008 with vCenter and
vShield Manager
Windows7Ultimate100MSwap
Windows8_64
Windows8_x86

192.168.200.70 VMware ESXi, 5.0.0, 515841

Name	State	Status	Provisioned Space	Used Space	Host CPU - MHz	Host Mem - MB	Guest Mem - %	Notes
VMware vCenter Server Applia...	Powered Off	Normal	97.10 GB	9.84 GB	0	0	0	VMwa
iPXE_Test	Powered Off	Normal	18.12 GB	1.70 MB	0	0	0	
VMware vCloud Director	Powered Off	Normal	32.61 GB	30.00 GB	0	0	0	
vShield Manager	Powered Off	Normal	8.11 GB	1.99 GB	0	0	0	vShie
VMware Windows2008 with vC...	Powered On	Normal	44.05 GB	44.05 GB	167	3283	13	
Windows8_x86	Powered Off	Normal	25.11 GB	196.77 KB	0	0	0	
Windows8_64	Powered Off	Normal	34.12 GB	9.39 GB	0	0	0	
clone4	Powered On	Normal	25.05 GB	8.25 GB	0	0	0	
clone2	Powered On	Normal	25.05 GB	8.25 GB	0	0	0	
clone1	Powered On	Normal	25.05 GB	8.25 GB	0	0	0	
clone3	Powered On	Normal	25.05 GB	8.25 GB	0	0	0	
Windows7Ultimate100MSwap	Powered On	Normal	25.05 GB	8.25 GB	0	0	0	

Recent Tasks

Name	Target	Status	Details	Initiated by	Requested Start Ti...	Start Time	Completed Time
Power On virtual mach...	clone2	Completed		System	6/22/2012 11:38:57 ...	6/22/2012 11:38:57 ...	6/22/2012 11:39:01 ...
Power On virtual mach...	clone1	Completed		System	6/22/2012 11:38:57 ...	6/22/2012 11:38:57 ...	6/22/2012 11:39:01 ...
Power On virtual mach...	clone4	Completed		System	6/22/2012 11:38:57 ...	6/22/2012 11:38:57 ...	6/22/2012 11:39:01 ...
Power On virtual mach...	clone3	Completed		System	6/22/2012 11:38:57 ...	6/22/2012 11:38:57 ...	6/22/2012 11:39:01 ...
Power On virtual mach...	Windows7Ulti...	Completed		System	6/22/2012 11:38:56 ...	6/22/2012 11:38:57 ...	6/22/2012 11:39:01 ...
Initialize powering On	VinceDC	Completed		Administrator	6/22/2012 11:38:56 ...	6/22/2012 11:38:56 ...	6/22/2012 11:38:57 ...
Move into resource pool	VBACluster	Completed		Administrator	6/22/2012 11:29:44 ...	6/22/2012 11:29:44 ...	6/22/2012 11:29:44 ...
Clone virtual machine	Windows7Ulti...	Completed		Administrator	6/22/2012 11:29:34 ...	6/22/2012 11:29:34 ...	6/22/2012 11:29:44 ...
Clone virtual machine	Windows7Ulti...	Completed		Administrator	6/22/2012 11:28:54 ...	6/22/2012 11:28:54 ...	6/22/2012 11:29:03 ...

Saving the Entire Configuration

The entire test bed, Master LUN and all clones can be saved to a named configuration at any time, by selecting the Configuration page from the menu, providing a file name and selecting "Save".

Saving an Individual VirtualAware Configuration

Any individual Virtual Datastore can be saved at any point by selecting the VirtualAware LUN and using the "save to file" function.

Conclusion

SANBlaze VirtualAware technology provides RAM based line rate datastores for high-end VMware initiator capacity planning and performance test environments.

Simple configuration hardware, accelerated "clone" operations and automatic de-duplication allow for large installations to be staged and deployed quickly for maximum test efficiency.

VirtualAware allows instantaneous "clone" operations allowing multiple snapshots of a single master guest operating system on vmfs3 Datastores to be presented as unique Virtual Machine storage to VMware virtual machines.

RAM based operation and high bandwidth interconnects to the VMware initiator provide a perfect test bed for applications that would otherwise be limited by the performance of the storage subsystem, such as the "boot storm" scenario when hundreds of virtual machines try to access their Datastores simultaneously.