

How VembuHIVE, a backup repository as a file system is changing the dynamics of data protection



Dr. Jack Fegreus

Founder of openBench Labs.

Copyright Notice

Copyright © 2019 Vembu Technologies. All rights reserved. No part of this Whitepaper can be reproduced or used in any manner whatsoever without the permission of the publisher.

This whitepaper was initially published in December 2015 based on the test results and metrics obtained by using the Vembu BDR Suite v3.5 and these test results and metrics were updated on December 2018 by conducting the same set of tests using the Vembu BDR Suite v4.0

Snapshot of Findings

In this analysis, openBench Labs assesses the performance and functionality of the Vembu Backup & Disaster Recovery (BDR) host-level (a.k.a. agentless) data protection solution in a **VMware vSphere 6.5 HA Cluster**. For this test we utilized a vSphere VM configured with three logical disks located on separate datastores to support an Exchange server with two mailbox databases. Each of the mailbox databases was configured to support 1,000 user accounts.

This paper provides technically savvy IT decision makers with the detailed performance and resource configuration information needed to analyze the trade-offs involved in setting up an optimal data protection and business continuity plan to support a service level agreement (SLA) with line of business (LoB) executives.

To test backup performance, we created 2,000 AD users and utilized LoadGen to generate email traffic. Each user received 120 messages and sent 20 messages over an 8-hour workday. Using this load level, we established performance baselines for a data protection using direct SAN-based agentless VM backups.

In this scenario, we were able to:

- Finish crash-consistent incremental agentless backups in 18 minutes, while processing our base transaction load of 12 Outlook TPS.
- Restore a fully functional VM in less than 5 minutes as a Hyper-V VM capable of sustaining an indefinite load of 4 Outlook TPS
- Recover all user mailboxes as .pst files from a host-level agentless VM backup with no need to schedule a Windows Client backup initiated within the VM's guest Windows OS.

Rather than store image backups of VMs and block-level backups of physical and VM guest host systems as a collection of backup files, Vembu BDR utilizes a document-oriented database as a backup repository, dubbed VembuHIVE, which Vembu virtualizes as a file system.

Redefining Data Protection on a Virtual File System

This paper examines how to use Vembu BDR Suite to implement distributed backup and disaster recovery (DR) operations in a centrally managed data protection environment with an ingenious twist. Rather than store image backups of VMs and block-level backups of physical and VM guest host systems as a collection of backup files, Vembu BDR utilizes a document-oriented database as a backup repository, dubbed VembuHIVE, which Vembu virtualizes as a file system.

Documents in a document-oriented database encapsulate information encoded in value-key pairs using a language, such as XML or JavaScript Object Notation (JSON). Like a file, a document can store any data without following a strict schema. In addition, every document in a document-oriented database can be retrieved using a unique key for the document and can be queried on its content using the encoding language as a query language.

Moreover, the value-key construct for documents creates a database that is highly scalable through the simple addition of storage and compute resources. The scalability of a document-oriented database has been leveraged by a number of large commercial websites, including eBay.

During a backup, the Vembu BDR service, which handles all backup and recovery functions on the Vembu BDR server, de-duplicates and compresses data from an image and block-based backups. Next, Vembu BDR encodes the processed data with content metadata and streams the new collection of processed data and content metadata as documents into the VembuHIVE document-oriented database using very large data blocks. During a full backup of the Exchange VM, the Vembu BDR service streamed the processed data into VembuHIVE using blocks that averaged just under 3 MB.

Backup Specific Restore Anywhere

By replacing structural metadata related to a VM's host file system with content metadata, before Vembu BDR commits the data to a document in VembuHIVE, enables Vembu to virtualize VembuHIVE as a file system, with respect to backup documents. With VembuHIVE acting as a virtual file system, the functionality of the Vembu BDR product can be extended by introducing modules that mimic advanced OS file system utilities that provide such feature as de-duplication, error correction, and version control.

In particular, by applying formatting utilities to VembuHIVE documents, Vembu BDR is able to present a full disk image associated with a VM backup in multiple disk formats, such as .vhd, .vhdx, .vmdk, and .img, on a virtual drive created on the Vembu BDR server. More importantly, Vembu BDR server is able to leverage the presence of disk images with full read/write access on demand in a number of significant ways, including the need of many data protection packages to run backups directly on a VM to protect and recover application-level data items.

By mounting the logical disks of a vSphere VM in a virtual drive as local disk files, Vembu BDR is able to implement application-level backup and recovery functions that would typically require a full backup agent installed on the original VM's host OS. These disk images can also be used to instantly boot a backed-up VM as a Hyper-V VM, without regard to the original VM's host. Using Vembu Quick VM Recovery option simplifies overhead tasks by eliminating the need to mount a network datastore containing read-only pointers to backup data, remap disk writes to a cache or redo logs, and consolidate the pointers and logs into a standard configuration.

For an Instant-boot of a VM image, the Vembu BDR service creates a persistent document within VembuHIVE that can be read, modified, and saved. We were able to recover our Exchange VM by choosing a backup time and booting the respective image using a fully automated recovery process that completed in well under five minutes. Vembu BDR utilizes the local server's Hyper-V defaults to configure the new VM. Consequently, we were able to customize the settings of the Exchange VM using Hyper-V Manager and comply with an SLA to restore Exchange in about 5 minutes to a state representing a loss of no more than 30 minutes of email processing.

Backups and Business Continuity

To implement host-level VM image backups—often dubbed agentless backups—in a vSphere virtual infrastructure (VI), Vembu BDR utilizes VMware application programming interfaces (APIs), including the vSphere Storage APIs for Data Protection (VADP). In particular, VADP provide a snapshot-based framework for VM backup, which Vembu BDR leverages using the latest release of VMware Virtual Disk Development Kit (VDDK 6.0) to access, manipulate, and transfer VM data.

By combining tight integration with vSphere for VM image backups with unified block-level OS and application backups of physical systems, Vembu BDR provides a critical business value to any CIO working with line of business (LoB) executives. For LoB executives, the most important function of IT is the ensuring of business continuity for key business applications. Moreover, these executives drive the growing demand on IT to comply with a service level agreement (SLA) for business continuity. Pivotal components in such an SLA are a Recovery Point Objective (RPO), which limits the amount of data that can be lost, and a Recovery Time Objective (RTO), which limits the amount of time taken to recover after a system outage.

Adding Application Awareness

For data protection, a Virtualized Infrastructure (VI) provides IT with greater flexibility; however, a VI simultaneously presents IT with radically different logical constructs from a typical physical infrastructure. A unique duality characterizes a VI. From a physical perspective, a VI is a collection of host servers running a common hypervisor and supporting a set of applications. From a logical perspective, each hypervisor application is a VM running a distinct OS and hosting its own set of applications.

While VI management software attempts to make VM duality transparent, data protection operations continue to remain difficult for IT administrators to master. IT is able to provide highly efficient hypervisor-level data protection by backing up VMs as unique entities. Nonetheless, such a data protection scheme on its own fails to support the needs of users. LoB users focus exclusively on data objects associated with the applications running within a VM, such as a user's mailbox in an Exchange mailbox database.

For an IT administrator to perform data protection tasks, such as application data recovery and log truncation, a host-level VM backup must invoke APIs within the guest OS to quiesce VM application I/O activity by committing all current transactions and freezing new transactions. Application quiescence creates a crash-consistent backup within the guest OS. In the case of a host-level backup of a VM running Exchange, Microsoft explicitly recommends using Windows Volume Shadow Service (VSS) Writer to quiesce Exchange, truncate logs, and avoid data loss.

To quiesce VM guest OS applications, Vembu provides App-aware, a VMware Tools extension, which is frequently referred to as a VSS requestor agent. IT administrators install App-aware on any VM running an application requiring log truncation after a backup. VSS requestor agents are frequently used to call APIs in a VM guest OS; however, some competitors, such as Veeam, download and install a VSS requestor agent at run-time, and then remove it after the backup is completed.

Vembu's App-aware agent uses the VSS Writer to implement Redirect on Write (RoW) snapshots within a Windows guest OS, rather than Copy on Write (CoW) snapshots, which imposes less total I/O overhead on an incremental VM backup.

Critical RPO and RTO Success Factors

Minimizing data loss for an application means maximizing the number of backups created for an application. To meet an aggressive RPO for a critical application, IT operations must be able to schedule frequent backups, that occur as the application runs throughout the workday. To support fast incremental VM backups that have a minimal impact on application processing, vSphere implements a Changed Block Tracking (CBT) mechanism, which explicitly maps all of the modified data blocks for an incremental backup. In addition, the updated VMware VDDK 6.0 significantly reduces the overhead associated with an ESXi Copy on Write (CoW) snapshot.

For VM logical disks on ESXi datastores, CoW snapshots are highly spaced efficient. To represent a snapshot of a logical disk, an ESXi host is able to create an empty file instantly in the VM's datastore. Only when data needs to be written to a logical disk, does the host actually write data into the snapshot? In particular, the host reads the current data, writes that data to the snapshot, and then writes the new data to the original location. For a VM, the presence of a CoW snapshot results in performing three logical I/O operations for each new write to the file representing a VM logical disk.

The overhead for writes associated with a CoW snapshot escalates dramatically when a business critical application with a high level of I/O activity is running on a VM with Windows guest OS. As part of the backup process, a VSS requestor agent on the VM will need to invoke the VSS Writer to quiesce the application and create snapshots of logical disks. In this process, most VSS requestor agents double down on I/O write overhead by also implementing the Windows Server guest OS snapshots as CoW snapshots that are encapsulated within the ESXi host CoW snapshot.

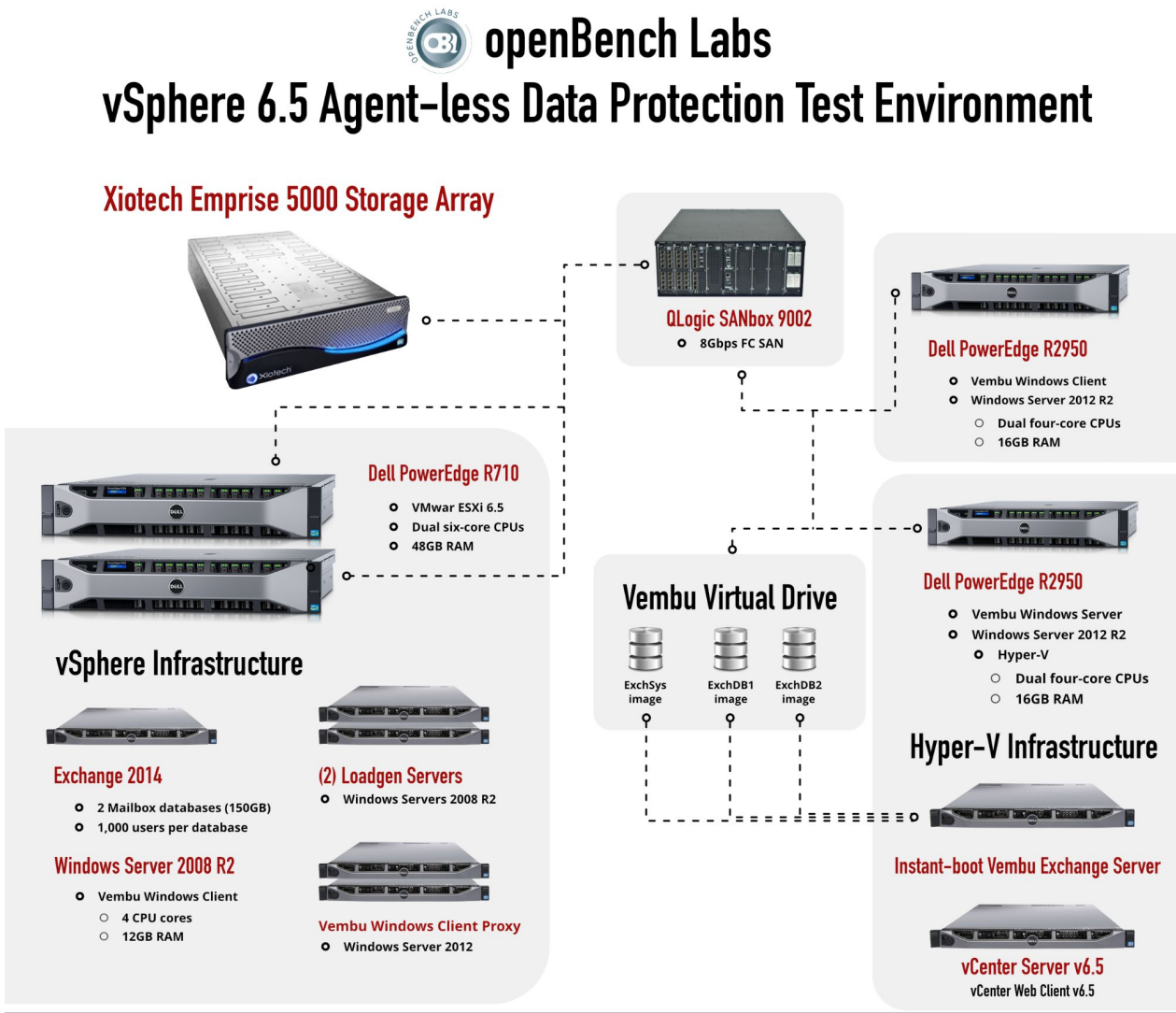
Lowering Snapshot Overhead Through Redirection

Vembu BDR leverages all of the VADP optimization features in performing an ESXi CoW snapshot of the VM; however, Vembu's App-aware agent uses the VSS Writer to implement Redirect on Write (RoW) snapshots within a Windows guest OS, rather than CoW snapshots, which imposes less total I/O overhead on an incremental VM backup. While a RoW snapshot provides the same space-efficiency as a COW snapshot, a RoW snapshot critically does not double the number of logical write operations.

Like a CoW snapshot, a RoW snapshot starts as an empty container. A RoW snapshot, however, does not copy existing data into a snapshot file before writing new data to the original location. A RoW snapshot process writes new data directly to the snapshot file and sets up a pointer to redirect access around the old data, which remains in place. Given the lower overhead associated with an active RoW snapshot, this scheme has been adopted by a number of vendors, including NetApp.

For a highly active VM—even when using the new VMware VDDK 6.0—removing an ESXi snapshot that encapsulates a CoW VSS snapshot can take twice as long as copying CBT data in an incremental VM backup. In contrast, the overhead impact of RoW snapshots is only manifested when unwinding pointers to remove a long chain of snapshots. Since App-aware leaves only one RoW VSS snapshot open for log truncation during an ESXi snapshot, there is no chain of RoW snapshots to unwind when completing an incremental VM backup. Consequently, the issue of RoW overhead extending the time window of an incremental VM backup using Vembu BDR is moot.

Test Bed Infrastructure



Vembu BDR Exchange VM Backup Scenario

We built our test infrastructure using two Dell PowerEdge R710 servers to host a vSphere 6.5 datacenter cluster that supported an Exchange messaging service, which was the central focus of our Vembu BDR testing. We deployed a Dell PowerEdge 2950 system with dual four-core CPUs and 16GB RAM running Windows Server 2012 R2 to support Vembu BDR Server, which includes the Vembu BDR service and VembuHIVE. In addition, we installed Hyper-V on the Dell 2950 server to test Vembu BDR’s advanced cross-platform support features for VM recovery.

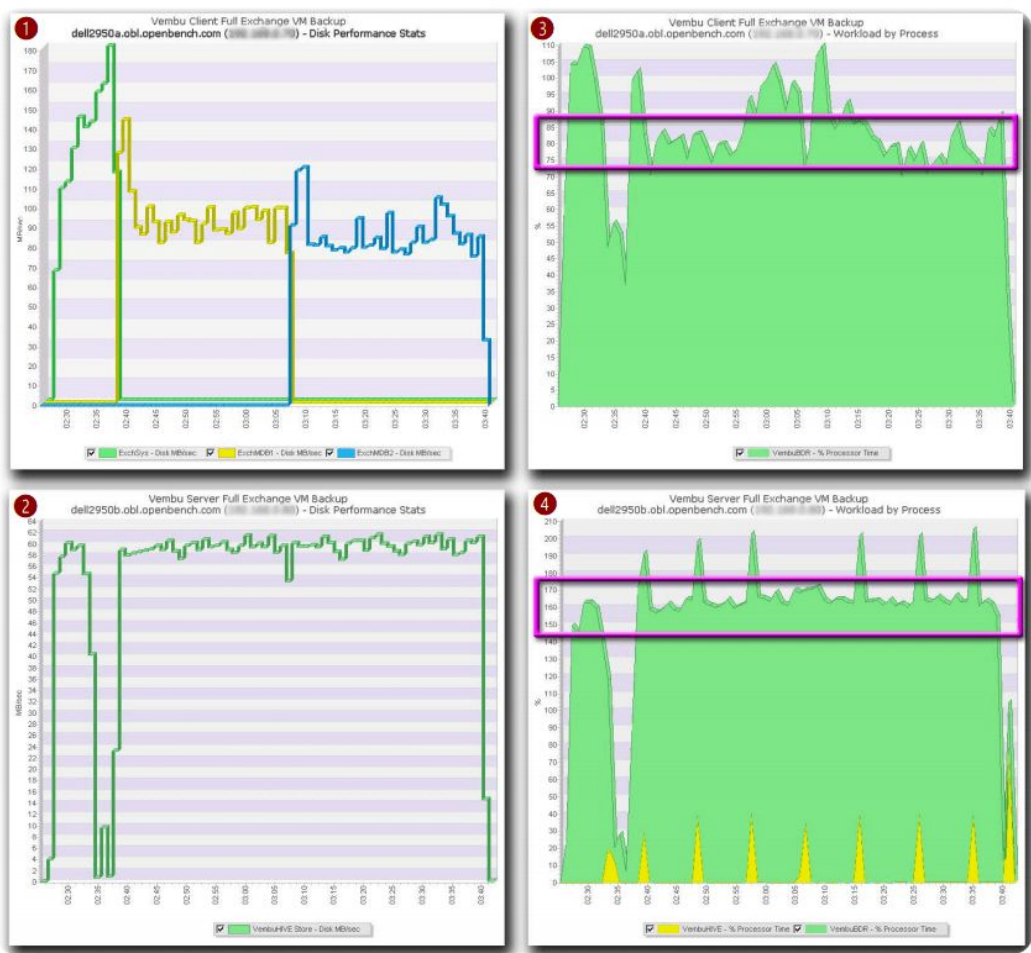
We also set up a second Dell 2950 server running the Vembu Windows client as a SAN-based proxy for host-level VM backups.

Email now represents over 90 percent of business communications, which accounts for the plethora of stakeholders for any corporate email service. Furthermore, email is legally classified as financial data and a disaster recovery (DR) plan can involve over 10,000 regulations from sources including the Federal Rules of Civil Procedures.

To leverage the resources needed to deal with these issues, IT has made email a premier high-profile VI application. Running Exchange on a VM, however, complicates an agentless data protection scheme. As Microsoft continues evolving internal Exchange functions to meet growing performance demands, changes made to data structures alter I/O patterns and modify resource requirements for any VM supporting Exchange. In particular, we tested Vembu BDR using Exchange 2014, which features a number of critical structural changes that impact VM backup.

To sustain email processing during elevated activity, new transactions are written to a memory cache before being written to log files from where transactions become available to users before being moved into a mailbox database. Consequently, tight integration with the Microsoft Messaging Application Programming Interface (MAPI) is requisite to truncate logs using the VSS Writer and recover items from user mailboxes. In addition, Microsoft eliminated Single Instance Storage (SIS) tables from mailboxes to stream transaction I/O using large data blocks. By avoiding data normalization, Microsoft reduces more overhead than it adds by writing more data per transaction with redundant data. Nonetheless, processing more transactions containing more data directly impacts CoW snapshot overhead

Infrastructure Scalability



Vembu Full Backup Scalability

To gain an insight into distributed backup processing with Vembu, we began with a SAN-based full backup of our VM running Exchange. During the full backup, the Vembu client¹ streamed data in 512KB blocks, rather than the traditional 1MB blocks used by most SAN-based backup schemes. This choice optimized Vembu BDR processing, which maintained a steady 60 MB per second stream of writes using 2.5MB blocks to VembuHIVE².

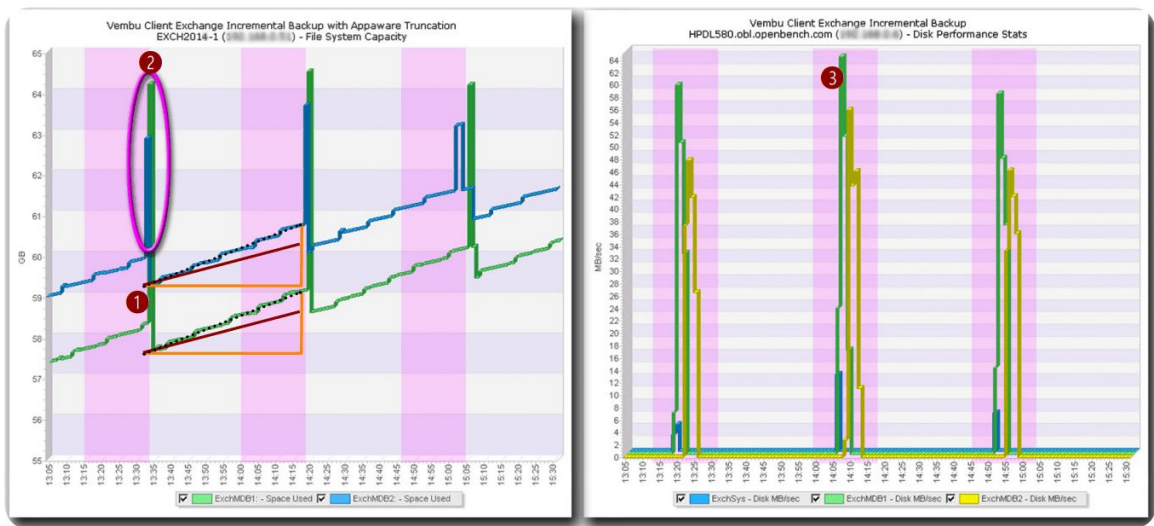
By optimizing end-to-end backup throughput, the Vembu client consumed only 80% of a single core³, while the Vembu server utilized 160% of a core⁴. What's more, a backup window was the same as that of competitors processing the VM's logical disks in parallel, which consumed five cores.

DR Support for Dynamic Business Applications

To test Vembu BDR’s capability to protect an active Exchange email service in a vSphere VI, we provisioned a VM with four CPUs, 12GB of RAM, and three logical disks to support 2,000 users with Exchange 2014. We optimized VM I/O by setting up two mailbox databases containing 1,000 user mailboxes per database, located the databases on separate logical VM volumes, and stored those volumes on dedicated ESXi datastores on separate arrays.

We simulated Outlook message traffic using two VMs running LoadGen. Every Exchange user was assigned a LoadGen profile to receive 120 messages and send 20 messages over an 8-hour workday. This profile generated a message load of 12.6 Outlook TPS on the VM Exchange server.

Meeting SLA Objectives



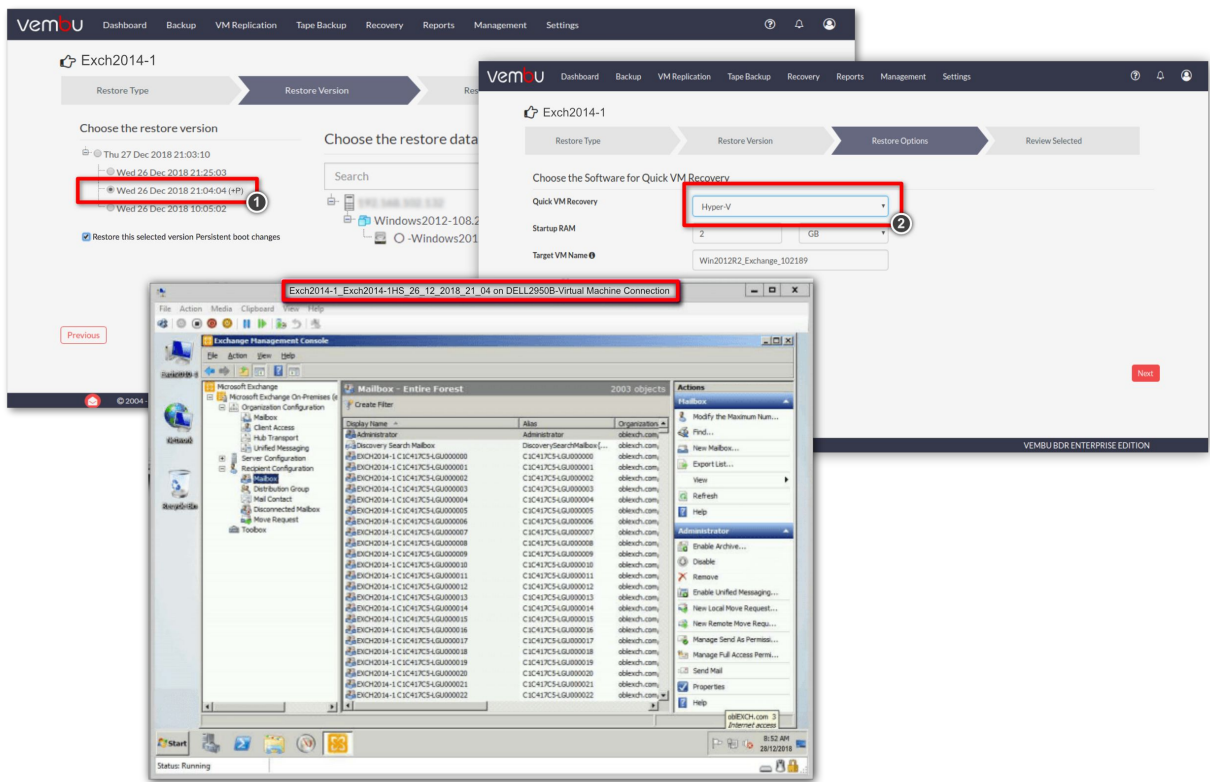
Vembu Incremental Exchange VM Backups

Given all the corporate stakeholders for email, any SLA for an Exchange server needs to address a number of critical issues including from minimum RPO and RTO goals for business continuity. Based on our transaction load, we set an RPO goal of 30 minutes, which equated to the processing of 22,575 Outlook user transactions. To meet that goal, we ran an incremental every 30 minutes. Every backup utilized App-aware to quiesce the Exchange mailbox databases and generate a RoW snapshot via the VSS Writer. As a result, we were able to safely truncate mailbox database logs after every backup with minimal impact on Exchange transaction processing.

While running our base Outlook transaction load, the total amount of data contained in mailbox databases and logs increased by 3GB every 30 minutes^①. At the completion of an incremental backup, log truncation is triggered and removed over 500MB of data from the log files of each mailbox database. In completing incremental backups and removing the ESXi CoW and VSS RoW snapshots, we also observed a unique transitory increase in the total volume of data^②. More importantly, that 3GB of new mailbox data represented only about 15 percent of the totally new and modified data transferred during every incremental backup. Incremental Vembu backups^③ typically read 20GB of CBT data and wrote 6GB of de-duplicated and compressed data in about 17 minutes.

SAN-based incremental backups with competitive products typically read data using 1MB blocks, which transfer data faster; however, large block reads include redundant data, which inflates the amount of data transferred by about 40 percent and increases the amount of data stored by 75% on every backup. In addition, the use of CoW VSS snapshots results in a snapshot removal process that is longer than the time gained by faster backup throughput.

Minimizing RTO With VembuHIVE



Vembu Cross Platform Instant Boot

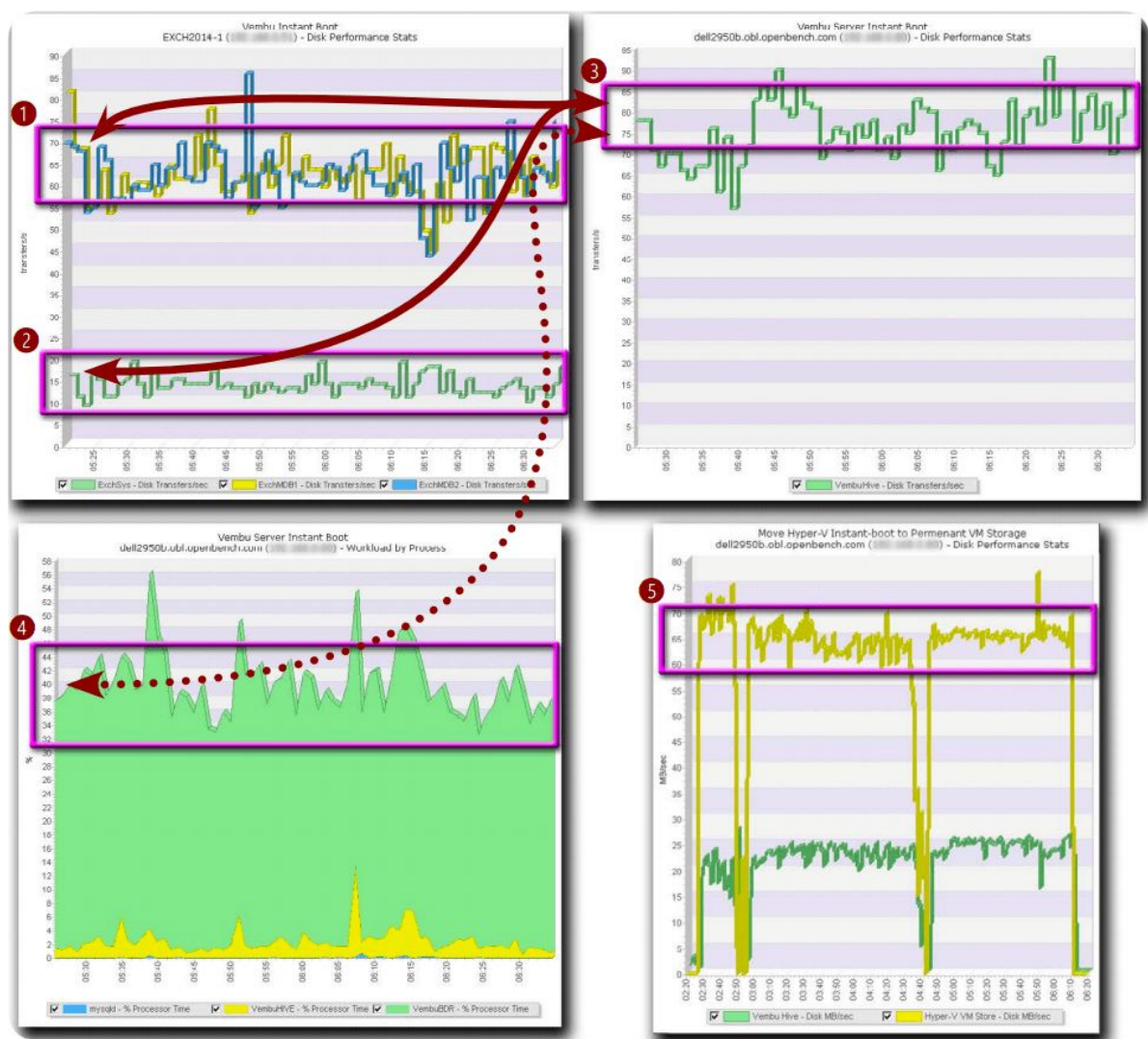
For LoB executives, the value of backup and restore lies entirely in data recovery processes and they measure that value with respect to RTO and RPO. For marketing and sales executives, computer downtime represents more than lost revenue, today computer downtime also equates to potential losses in customer confidence and market share.

To minimize RTO and bring systems online within minutes, the Vembu BDR provides Vembu Quick VM Recovery option, an option that enables an IT administrator to boot a VM into a Hyper-V^① VI directly from a backup document in VembuHIVE. To leverage this feature, IT only needs to install Hyper-V on the server running Vembu BDR. Unlike booting a VM from a backup file, Vembu Quick VM Recovery option does not require IT to configure a specialized subnet with a fenced network topology based on IP masquerading within the production VI.

To launch a VM with Quick VM Recovery, an IT administrator chooses a VM and selects either a backup time^② or the “recent Instant-boot version^③”, which is a persistent document in VembuHIVE. Unlike the boot schemes of competitive products, Vembu Quick VM Recovery option does not rely on read-only pointers to a static backup file. By representing logical disks with documents that can be modified and saved, the Vembu BDR service imposes minimal overhead as it manages interactions between a logical VM disk and VembuHIVE, which acts as a VM datastore.

Minimizing RTO With VembuHIVE

To assess Quick VM Recovery performance, we ran multiple LoadGen scenarios that generated different transaction rates across all user mailboxes serviced. Our goal was to determine the highest Outlook transaction load that our Exchange service could sustain before the number of transactions queued in the cache for log writes grew at a faster rate than the rate at which transactions could be written to disk without pausing the arrival of transactions.



Outlook Transaction Processing with Instant-boot

With all logical disks supported by VembuHIVE, our VM Exchange server was able to sustain an Outlook transaction load of 4.5 TPS. With respect to disk I/O operations per second (IOPS) over the VM's three logical drives, our transaction load generated 65 IOPS^① using 48KB data blocks on each logical disk with a Mailbox database and 15 IOPS^② using 32KB blocks on the VM system disk. To sustain the total 145 IOPS load, the Vembu BDR service interacting with VembuHIVE generated 80 IOPS^③ using 456 KB block transfers from the VembuHIVE disk. What's more, Vembu BDR and VembuHIVE consumed just 40 percent of a single core^④ to process the Outlook transactions.

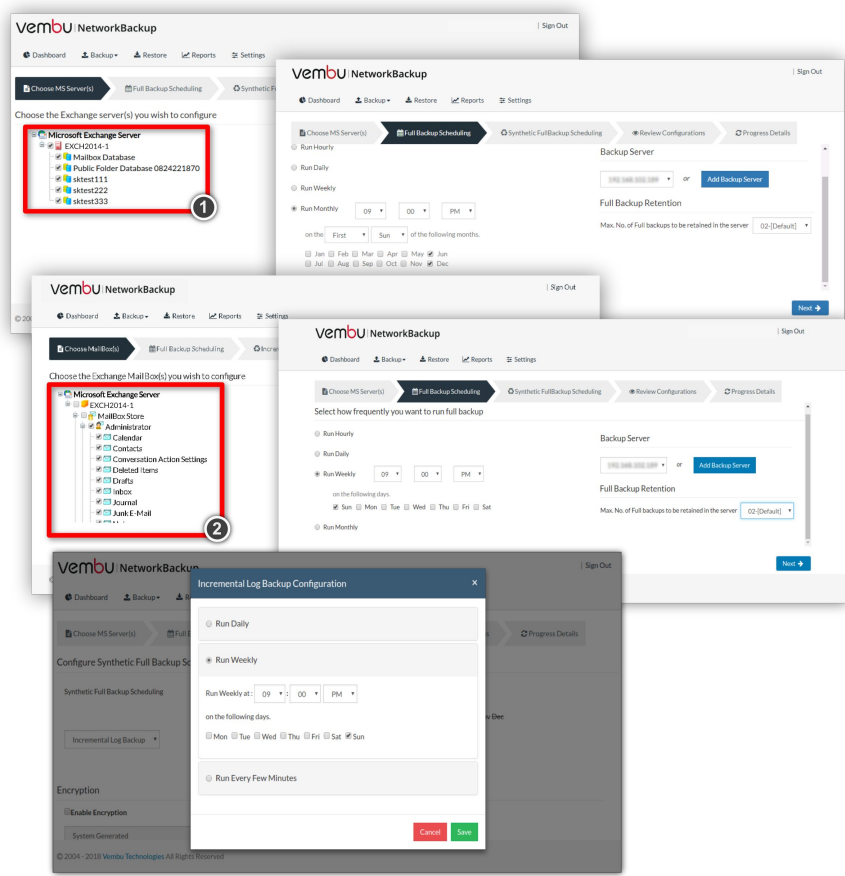
To enable the Hyper-V VM to process Outlook transactions at the same 12.5 TPS rate as the original vSphere VM, we used a Hyper-V move storage operation to copy the VembuHIVE disk images exposed on the Vembu virtual drive to a local disk. Since a Hyper-V move attempts to delete each disk from its original location, we needed to run a separate process for every drive. In every process, the new disk files were written to a local Dell 2950 volume^⑤ at 65 MB per second in large 400 KB data blocks.

Vembu BDR avoids the need for agent-based VM backups by leveraging two important capabilities: presentation of VM logical disks contained in VembuHIVE locally in any image format, and direct extraction of user mailboxes from a mailbox database via MAPI.

Application Data-Item Recovery

For IT administrators, however, the most prevalent day-to-day email problems are related to issues associated with the recovery of user messages and accounts. The reasons behind requests to recover mailbox data items are legion, ranging from inadvertent user deletions to issues of legal discovery. Consequently, an IT operations staff requires a robust set of tools to deal with quick mailbox data retrieval and flexible packaging of the results.

Traditional User Mailbox Recovery



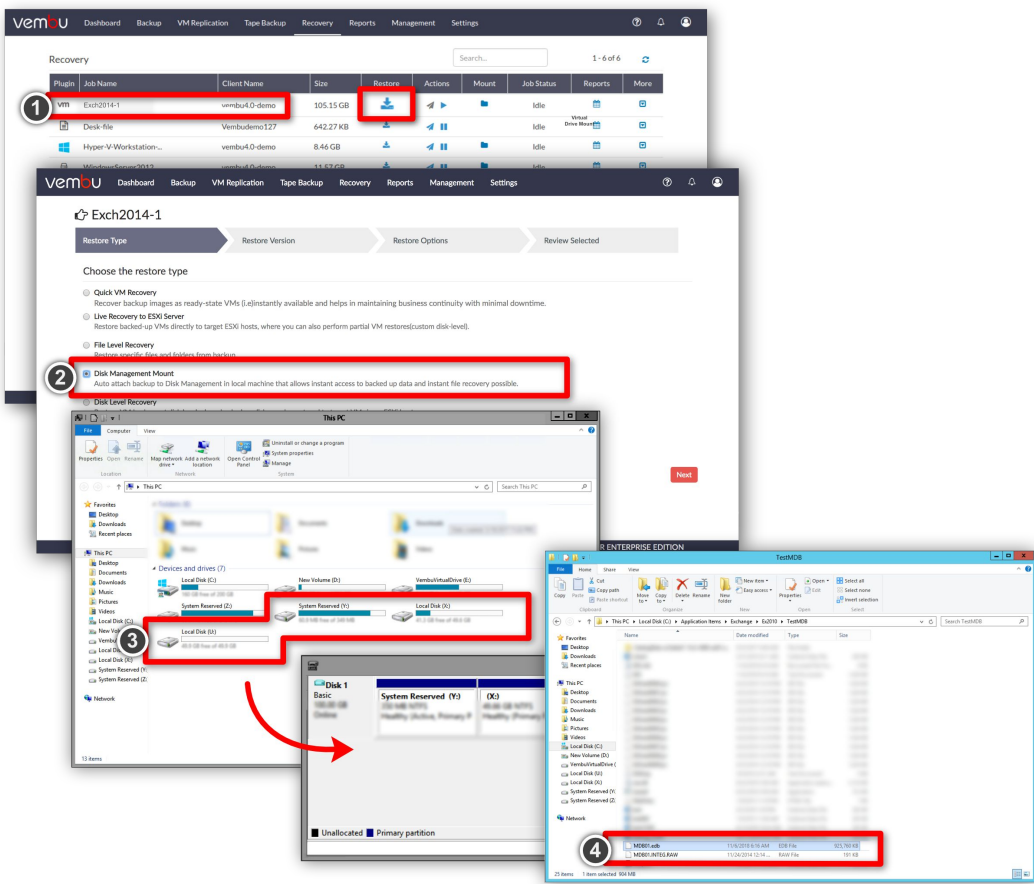
Vembu Windows Client Backup Modules

For physical servers, Vembu resolves email message-recovery issues by adding Windows Client backup modules with rich MAPI capabilities for Exchange. With a full Vembu Windows Client installed on our VM, we were able to back up full Exchange mailbox databases¹ using a full, Incremental log, or differential log process and backup constituent user mailboxes and folders² within a mailbox database as independent .pst files.

Many popular data protection packages with host-level VM backup, such as Symantec's Backup Exec, require agent-based backups of Exchange on a VM to provide mailbox-level protection. Vembu BDR avoids the need for agent-based VM backups by leveraging two important capabilities:

- Presentation of VM logical disks contained in VembuHIVE locally in any image format,
- Direct extraction of user mailboxes from a mailbox database—an .edb file—via MAPI

User Mailbox Recovery From Host-level Backups

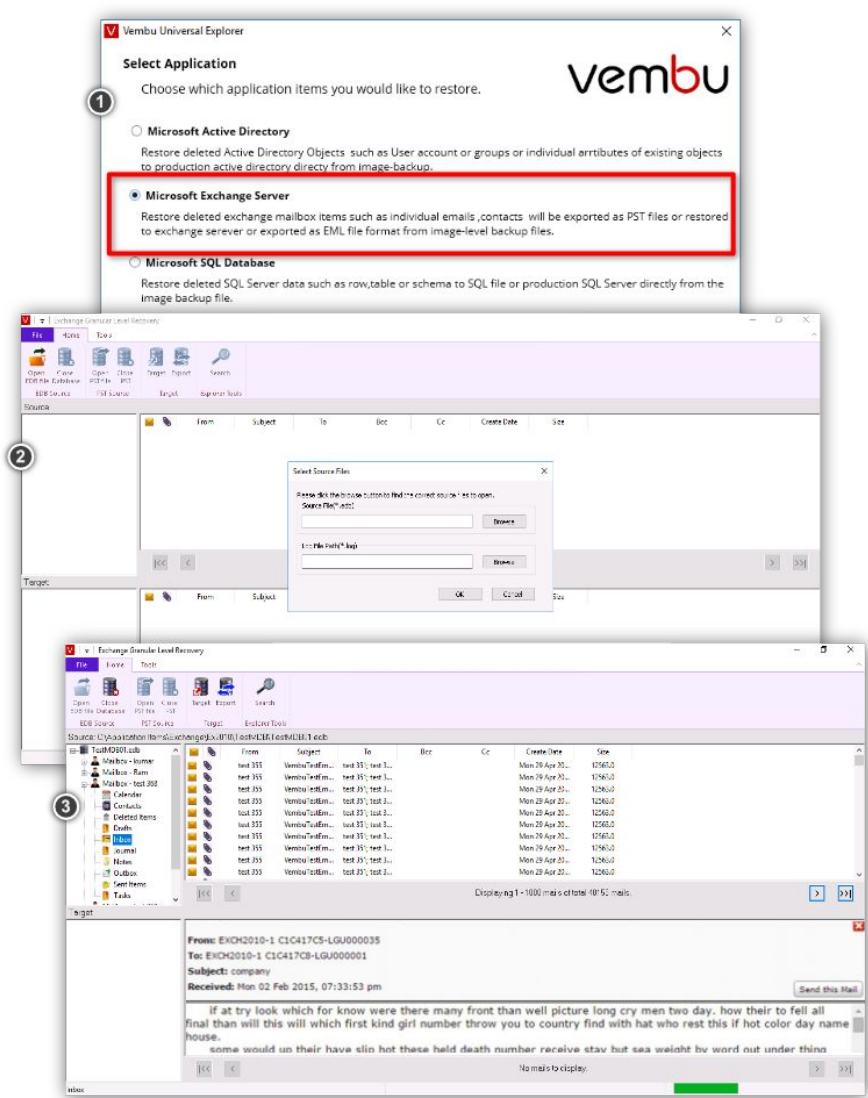


Local Exchange MailBox Database Access

To recover Exchange data items directly from a user mailbox contained in a VM image backup, we went to the recovery window of Vembu BDR Server¹ and it is a two-stage process that first utilizes the Disk Management Mount recovery option² to instantly present images of all of the logical disks associated with a designated VM in the local disk management³ of the Vembu BDR Server. In particular, the first stage in recovering user-level Exchange data items applies this process is applied to a VM running Exchange.

In the first step of this stage, the VembuVirtualDrive is populated with a time-stamped chain of .vhd and .vhdx disk images associated with full and incremental VM backups of a VM running Exchange. Then these disk images are virtually mounted on the server running Vembu BDR exposes the files located on the VM virtual disk⁴.

Consequently, the second stage of the process, which involves installing of Vembu Universal Explorer, can now be opened locally¹ on the Vembu BDR server without involving the production VM.



Local Exchange MailBox Database Access

Once a logical VM disk containing an Exchange Mailbox .edb file has been mounted locally via VembuHIVE, an IT administrator is able to invoke the Microsoft Exchange Server option in Vembu Universal Explorer. This provides a browser for an administrator to locate an instance of an Exchange mailbox database (.edb file)² on a local disk to open. Then the Vembu BDR process is able to leverage MAPI to open the database file and expose all of the individual components. In particular, an administrator is able to scroll through all of the mailboxes to recover mailboxes, mailbox folders, or individual messages as .pst files³.

During VM image backups, the Vembu Backup Server removes all file system metadata from backup data stored as documents, in order to virtualize the VembuHIVE document-oriented database as a sui generis file system.

Customer Value

Vembu BDR Feature Benefits

- 1) Vembu BDR Data Store Maintains a Virtual File System:** Vembu BDR maintains a highly scalable document-oriented database, dubbed VembuHIVE, rather than an archive of backup files to radically improve the dynamics of backup and restore operations.
- 2) Document-oriented Database Enables Restore Anywhere for VMs:** VembuHIVE removes file system metadata from backup data, which enables on-demand datastore images to be immediately mounted as a logical disk for multiple hypervisors.
- 3) Document-oriented Database Minimizes RTO:** VembuHIVE leverages on-demand cross-platform mounting of datastores to automate the creation and booting off a backed-up VM.
- 4) App-aware VMware Tools Extension Optimizes RPO:** Vembu implements RoW VSS snapshots, which do not incur the I/O overhead of a CoW snapshot, enabling incremental backups to complete quicker and IT to schedule incremental backups more frequently.
- 5) Vembu Universal Explorer Integrates with MAPI and enables Recovery of Exchange Application Objects from a Host-level VM backup:** VM image backups can be exported as virtual disk images in any format. In particular, VM disks can be mounted as local disks, which then can be manipulated on the Vembu BDR Server without requiring any interaction on the original VM.

For CIOs, the top-of-mind issue is how to reduce the cost of IT operations. With storage volume, the biggest cost driver for IT, all storage management functions are directly in the spotlight. At the same time, corporate concerns over the expanding reliance of virtually all key corporate processes on IT is also focusing attention on IT operations as a pivotal component of business continuity.

What's more, the concerns of LoB executives over business continuity are helping to drive the next wave of IT projects. In a competitive 24 x 7 x 365 environment, computer downtime represents more than lost revenue to sales and marketing executives. LoB executives equate computer outages with potential losses in customer confidence and market share and expect IT to meet an RTO and an RPO measured in minutes or hours rather than days.

Vembu BDR never creates an incremental backup file from a CBT-based VM backup. Instead of storing a discrete set of backup files, Vembu BDR creates a virtual file system using VembuHIVE, a document-oriented database, for the universe of protected systems. As a result, a full system image can always be navigated within VembuHIVE for every recovery point of every protected system.

During VM image backups, the Vembu BDR server removes all file system metadata from backup data stored as documents, in order to virtualize the VembuHIVE document-oriented database as a sui generis file system. By using the Vembu BDR server to apply formatting utilities to VembuHIVE documents during a restore, it is able to restore a VM datastore in a format compatible with any hypervisor.

The data stores of a VM created with Vembu Quick VM Recovery option are treated as new persistent documents with read/write access, without involving read-only pointers or redo logs for IT to manage and consolidate

In a DR scenario, Vembu leverages the ability to restore a VM in any format, to provide an instant-boot functionality. When Vembu BDR Suite is installed on a server that is concurrently running Hyper-V, Vembu exports the datastores associated with a VM backup as Hyper-V disks and configures a VM to boot from the datastores.

The use of a document-oriented database and its virtualization as a file system also has important long-term implications. As the volume of data continues to expand with double-digit growth, so too grows the storage resources dedicated to backup, which currently has no other value than serving as a recovery medium. With backup data stored as documents in a document-oriented database, the door is open to analyzing that data along multiple dimensions to begin projecting computer usage trends and create an understanding the business value of that computer usage.