Using VSS to Obtain a Full Backup Of a Zen Database

A White Paper From

For more information, see our web site at
http://www.goldstarsoftware.com

Using VSS to Obtain a Full Backup of a Zen Database

Last Updated: June 2022

The Actian Zen database engine (formerly known as Actian PSQL) stores its data directly inside binary files, which are stored in a standard file system on a local hard disk on the server (or SAN). While the Zen v15 database engine is running, these files are opened by the engine and can be rapidly changing as users are inserting, deleting, or updating records within the environment.

Because the files may be always open while users are in the system, you must take some special precautions to back up these database files. There are three common options to consider:

1. Force all users to exit the system at the end of the day, then perform a normal full or incremental backup on the database folder and files.
2. Place all open files into Continuous Operations Mode with either BUTIL or the Backup Agent. Doing this tells the database engine that a backup is coming and it snapshots all of the files properly, redirecting any further disk writes to a series of Delta files until the backup is done. Once the backup finishes, the roll-out process begins, and changes are moved from the delta files back to the live files again.
3. Leverage an operating system feature called Volume Shadowcopy Services (VSS) and the VSS Writer component (which is built into the Server Engines starting with Pervasive PSQL v11 and above) to snapshot the data, and then copy that snapshot to obtain a full backup.

The first option is self-explanatory and simple enough. In some cases, however, users may forget to exit the application at the end of the day, or the system may have uptime requirements such that getting everyone out is not simply not possible. When users are still in the system, the backups fail, so this is not really a fool-proof solution.

The second option (ContOps) has some limitations, as well. The biggest issue is that it can be complicated to set up, and both mechanisms (BUTIL or Backup Agent) have their benefits and downfalls. More importantly, ContOps is simply not compatible with all application environments, because it requires database files with unique base filenames. Finally, ContOps also has some manual recovery steps when the system crashes, which makes it less hands-off as well. If you are interested in this solution, check our web site for that paper for additional details about Continuous Operations Mode and how to avoid the pitfalls.

The final option is to use a VSS Snapshot, which we will discuss further here.

What Is VSS?

VSS, or Volume Shadowcopy Services, is a built-in part of the Windows operating system environment that can facilitate backups of data on a server. It is commonly used for features such as “Previous Versions” (allowing you to track old versions of files) and the snapshots that take place when you install software or OS updates (allowing you to roll back the system if something doesn’t work correctly).

The actual implementation of VSS is a bit complicated, but it can be likened to taking a picture (or “snapshot”) of the hard disk as of a specific point in time. This snapshot is considered “frozen in time”

and made available for a specific period of time where you can use it to obtain a copy of all of the database files from the same moment in time, ensuring that the database backup is "crash consistent." While this VSS snapshot exists, the backup operation can read data from the snapshot, and this copy will represent a perfect moment in time image of the database files.

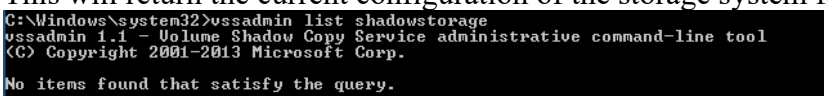
Of course, as users are still in the system, changes are still accruing inside the database engine cache. The database engine is able to write these changes to the disk with no issues because the operating system is ALSO maintaining a copy of the "previous" disk block in a special location (the VSS storage space), which may be located on the same volume or on a second volume with additional disk space. Because of this "copy-on-write" process, there is some overhead involved with VSS Snapshots in both CPU and disk activity, but this overhead can usually be minimized by doing your backups at the quietest time of the day and keeping the snapshots around for the smallest time possible.

So, as the backup is occurring, the database engine continues to operate normally, and users are not aware of the backup process at all. Once the backup is complete, the VSS snapshot is released, and all of the old disk blocks are released to free space once again, and the overhead of the VSS snapshot process is gone as well.

As mentioned above, this is a simplified view of VSS. The actual implementation details are a bit more complicated, requiring a discussion of "VSS Writers", filter drivers, and more. For a more thorough review of the topic, check the Microsoft web site. Here's a good place to start:
https://docs.microsoft.com/en-us/windows-server/storage/file-server/volume-shadow-copy-service

Setting Up a Server for VSS

The first thing that you have to do to make a VSS-based backup is to make sure that VSS is properly configured for your database volume. As with anything you do nowadays, there may be a variety of ways to complete the process, and there may be some differences to the process depending on your specific operating system. The screenshots here have been created from a Windows Server 2019 environment, so if you have a different release, check the Microsoft documentation for differences.

1) Because VSS is a system administration function, we need to have elevated privileges in order to work on this. So, start by launching a Windows Command Prompt **As Administrator**.
2) We then want to check to see if VSS is already configured. This is done with the VSSAdmin utility through this command:
   ```
   VSSAdmin List ShadowStorage
   ```
   This will return the current configuration of the storage system for VSS. If you get this back:
   
   ```
   C:\Windows\system32>vssadmin list shadowstorage
   vssadmin 1.1 - Volume Shadow Copy Service administrative command-line tool
   (C) Copyright 2001-2013 Microsoft Corp.

   No items found that satisfy the query.
   ```
   then you do not currently have VSS configured and you need to set it up first, so go on to the next step. If you DO already have storage configured, then you can jump to the next section.
3) VSSAdmin is also used to add VSS storage space, using a command like the following:
   ```
   VSSAdmin Add ShadowStorage /For=ForVolumeSpec /On=OnVolumeSpec
   /MaxSize=MaxSizeSpec
   ```
   Assuming that you have a data volume on drive E, sufficient space for the VSS storage also on drive E, and that you want to allow the VSS storage to use up to 10GB of disk space, then you can use this version of the command:
   ```
   VSSAdmin Add ShadowStorage /For=E: /On=E: /MaxSize=10GB
   ```

The MaxSizeSpec value can also be a percentage of disk space (like "20%") or it can be set to "UNBOUNDED", which means that all available free space can be used for VSS storage.

4) As a double-check, run the "VSSAdmin List ShadowStorage" command again, and you should see that VSS is now available on your data volume:

```
C:\Windows\system32>vssadmin list shadowstorage
vssadmin 1.1 - Volume Shadow Copy Service administrative command-line tool
(C) Copyright 2001-2013 Microsoft Corp.

Shadow Copy Storage association
   For volume: (E:)\\?\Volume{b048d5ad-ff73-11e0-a752-00505689000b}\
   Shadow Copy Storage volume: (E:)\\?\Volume{b048d5ad-ff73-11e0-a752-0050568900
0b}\
   Used Shadow Copy Storage space: 0 bytes (0%)
   Allocated Shadow Copy Storage space: 0 bytes (0%)
   Maximum Shadow Copy Storage space: 10.0 GB (10%)
```

You now have VSS configured for the E: volume, though no shapshots are present yet.

How Much VSS Storage Space Is Enough?

One common concern is having enough storage space. Remember that the amount of space needed will depend on the duration of time that a snapshot exists, as well as the number of disk writes that take place during this time period. If you are able to keep your VSS snapshots short-lived, and if you are able to do the backups at a relatively quiet time on the server, then you may not need much space at all.

However, if you are doing frequent VSS backups during the busy times of the day, then you may need more disk space that you might expect. You can use the command "VSSAdmin ReSize ShadowStorage" to adjust the size accordingly.

Just remember that if VSS runs out of space during a backup, then the snapshot may be discarded and the benefits of the VSS snapshot backup may be lost. For this reason, we do recommend that you use the **/MaxSize=UNBOUNDED** option when setting up VSS storage space whenever possible.

Testing VSS With a Manual Snapshot

The VSSAdmin tool allows you to create a snapshot right from the command line for a specific volume. This is valuable for testing the overall process to make sure that everything is working as expected.

1) Start an Administrative Command Prompt.
2) Issue the command (using your own drive letter, of course):

```
VSSAdmin Create Shadow /For=E:
```

If this is successful, then you should get a screen back like this:

```
C:\Windows\system32>vssadmin create shadow /For=E:
vssadmin 1.1 - Volume Shadow Copy Service administrative command-line tool
(C) Copyright 2001-2013 Microsoft Corp.

Successfully created shadow copy for 'E:\'
    Shadow Copy ID: {134debed-6efd-4cb0-934a-a06734483777}
    Shadow Copy Volume Name: \\?\GLOBALROOT\Device\HarddiskVolumeShadowCopy2
```

3) You can now verify that the snapshot exists with this command:

```
VSSAdmin List Shadows
```

Which should show you the existing snapshots:

```
E:\>vssadmin list shadows
vssadmin 1.1 - Volume Shadow Copy Service administrative command-line tool
(C) Copyright 2001-2013 Microsoft Corp.

Contents of shadow copy set ID: {638ff964-798a-425e-ae13-c1ddb97fff46}
   Contained 1 shadow copies at creation time: 6/2/2022 4:55:27 PM
      Shadow Copy ID: {134debed-6efd-4cb0-934a-a06734483777}
         Original Volume: (E:)\\?\Volume{b048d5ad-ff73-11e0-a752-00505689000b}\
         Shadow Copy Volume: \\?\GLOBALROOT\Device\HarddiskVolumeShadowCopy2
         Originating Machine:
         Service Machine:
         Provider: 'Microsoft Software Shadow Copy provider 1.0'
         Type: ClientAccessible
         Attributes: Persistent, Client-accessible, No auto release, No writers,
 Differential
```

4) Assuming that the system is being actively used while you have the snapshot active, you can even monitor the disk space used by this shadow copy through the **List ShadowStorage** command as before:

```
E:\>vssadmin list shadowstorage
vssadmin 1.1 - Volume Shadow Copy Service administrative command-line tool
(C) Copyright 2001-2013 Microsoft Corp.

Shadow Copy Storage association
   For volume: (E:)\\?\Volume{b048d5ad-ff73-11e0-a752-00505689000b}\
   Shadow Copy Storage volume: (E:)\\?\Volume{b048d5ad-ff73-11e0-a752-0050568900
0b}\
   Used Shadow Copy Storage space: 1.08 MB (0%)
   Allocated Shadow Copy Storage space: 512 MB (0%)
   Maximum Shadow Copy Storage space: 10.0 GB (10%)
```

5) When you are done, you can remove the shadow copy snapshot that you just created:

```
VSSAdmin Delete Shadows /Shadow={134debed-6efd-4cb0-934a-a06734483777}
```

Note that the large number is the Shadow Copy ID that was shown above in step 3.

```
E:\>VSSAdmin Delete Shadows /Shadow={134debed-6efd-4cb0-934a-a06734483777}
vssadmin 1.1 - Volume Shadow Copy Service administrative command-line tool
(C) Copyright 2001-2013 Microsoft Corp.

Do you really want to delete 1 shadow copies (Y/N): [N]? y

Successfully deleted 1 shadow copies.
```

After you set this up, you will also want to verify that the database engine is being notified as well. Whenever a VSS snapshot is initiated which includes a database volume, you should see a message in the engine's log file (PVSW.LOG or ZEN.LOG) that looks like this:

```
NTDBSMGR64.EXE  SERVER       I     PSQL VSS Writer state: Frozen
```

This tells you that the VSS Writer received the notification about the impending snapshot, completed all pending disk writes, quiesced (i.e. stopped using) the disk, and let the VSS coordinator know that the snapshot was clear to complete. When the coordinator hears back from all of the writers, it completes the snapshot process and then releases the writers, when you will see this message:

```
NTDBSMGR64.EXE  SERVER       I     PSQL VSS Writer state: Thawed
```

Typically, this message follows only a few seconds after the first.

Using VSS from Backup Software

If you are using a backup solution that already understands VSS, then there should be little else that you need to do. Simply signal to the backup software to use a VSS snapshot (which is usually a configuration value located on some setup screen), and it should handle the rest of the process for you.

One caveat to remember, though, is that the VSS Writer does not force new timestamps to the database files during a VSS snapshot. For this reason, you should ALWAYS perform a FULL BACKUP ONLY of the database files when done through a VSS snapshot. Attempting to perform a differential or incremental backup may result in currently-open (and changed) files NOT getting backed up properly,

as their file timestamps may be out of date. If you have a backup solution that relies on some *other* method for tracking changes, such as CBT or USN, then an incremental backup *may* be valid. If you are not sure, contact the application vendor or perform your own system tests.

Accessing a VSS Snapshot Yourself

If you need a bit more control over your backup process, then you may be interested to know that it is ALSO possible to access the snapshot data from a script, such as PowerShell, VBScript, or even a simple batch file.

While you can do this manually, a great automation tool is available under the unwieldy name of *Volume Shadow Copy Simple Client*, or VSCSC. You can find that tool here:

http://vscsc.sourceforge.net/

This web page contains all the tools and information you need to access the snapshot via a tool called DOSDEV, which allows you to mount a device as an accessible volume. The "device", in this case, is the snapshot volume itself, which can be mounted as drive B:, for example. Getting the device name is the hard part, since it changes with each snapshot – but this is where the VSCSC tool comes into play – it knows the snapshot name, and it can pass that to another batch file for the actual backup itself.

Once the snapshot is mounted, the secondary batch file fires up, and you can copy the (now static) data off of drive B: at your convenience using your favorite data synchronization tool, such as XCopy, RoboCopy, Beyond Compare, or just about anything else. See the code samples provided in Appendix A for an idea on how easy this actually is!

Again, as mentioned above, be sure that you only use a FULL backup of PSQL/Zen database files, especially if you have left the System Cache function disabled (the default for Server engines).

Finding More Help

If you need some additional hand-holding, Goldstar Software may be able to assist you further as well. You can contact us at 1-708-647-7665 or via the web at http://www.goldstarsoftware.com.

Appendix A: Sample Backup Scripts

The following command leverages the VSCSC to create a snapshot of the E: drive and then fire a secondary batch file to perform the actual copy operation. When the secondary batch file completes, control returns to VSCSC and the snapshot is dropped automatically, restoring the system performance back to normal levels.

```
vscsc -exec=backupzendata.bat e:
```

Then, the batch file BackupZenData.BAT can be configured to make the backup copy. Note that you are mounting the snapshot as drive B: here, so your SOURCE should always be the B: drive. The target folder can be any local or remote volume (such as the shared folder on a NAS, as shown here), but we do NOT recommend writing the copy to another folder on the E: drive, because all of those disk writes will need to be shadowed.

```
DOSDEV B: %1
XCOPY B:\MyApp\Data\*.* \\NAS\Backups\MyApp\Data /E
DOSDEV /D B:
```