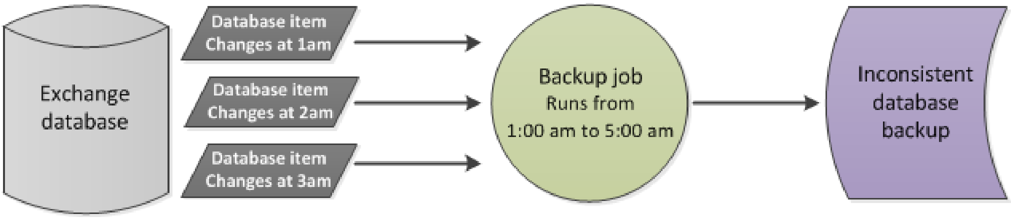
Creating consistent backups with VSS

If you create and administer backups, you will have heard the term VSS. You may even know that VSS runs in the background while a backup takes place. Knowing that is usually enough but if you're customizing your backups or encountering VSS problems it helps to know more. So let's take a closer look at VSS, the virtual shadow copy service.

What does VSS do?

Backups can take a long time, which is a problem because the data you are backing can be changing, sometimes a lot. This means your backup contains data as it was at different points in time—it is not consistent. This inconsistency causes problems for data, especially application data like databases that a constantly changing as shown in the example below.


Exchange database
Database item Changes at 1am
Database item Changes at 2am
Database item Changes at 3am
Backup job Runs from 1:00 am to 5:00 am
Inconsistent database backup

VSS solves the inconsistent data problem by creating and maintaining a point-in–time snapshot of the volume to be backed up. The backup job can then use this snapshot.

VSS can create snapshots with two levels of consistency:

- **Crash consistent**: A snapshot is created of the selected data's volume. The snapshot does not include data that is changing or in memory, so data may be missing or inconsistent.
- **Application consistent**: The application being backed up checks its own files in the snapshot to make sure they are correct. For example, information in memory and uncompleted database transactions are included in the snapshot, making it more accurate and consistent.

BackupAssist has a VSS application selection box that you can tick on the data selection screen. If you only select the application's files, the application will not be aware of the backup and will not make it application consistent.

How does VSS work

The VSS subsystems

The VSS service works with a set of subsystems to create and maintain snapshots. To understand VSS you will need to know what the 3 subsystems are.

- **The VSS requester:** The VSS requester asks for the snapshot to be created and, when it's no longer needed, to be removed. VSS requesters are built into VSS-Aware applications like backup software.
- **The VSS writer:** A VSS writers allow applications to ensure that their own files (in a snapshot) are application consistent. VSS writers are built into VSS-aware applications like Exchange and SQL.
- **The VSS provider:** VSS providers create and maintain snapshots, and come in two types.

  A *VSS software provider* creates a snapshot on the same local disk as the live data. Microsoft operating systems ship with a software VSS provider.

  A VSS *hardware provider* uses dedicated hardware to create a snapshot. 3rd party vendors can make hardware VSS providers for their devices, like a mirrored disk on a NAS that acts as the snapshot.

The VSS process in action

Now that you know about the subsystems used by VSS, let's look at how they work together to create a snapshot. Specifically, we will look at a snapshot created and maintained by a software VSS provider.

A software provider creates a snapshot on the same volume as the live data. If the live data changes, a copy of that data (before it changes) is saved to a location called the *shadow storage*. By doing this, the snapshot is able to maintain a point-in-time version of the live data, without using up much disk space.

So now that you know how the snapshot is maintained, how does the backup job use it? Simple – The backup job asks the VSS service for the selected data and the following happens:

- If the data HAS changed, the snapshot will provide the backup job with a copy of the data as it was before it changed, from the shadow storage.
- If the data HAS NOT changed, the snapshot will provide the backup job with the data from the live volume.

> The process of copying live data before it changes causes a small performance reduction for volume writes.
> The longer a snapshot is needed, the larger it will become; the more data changes the faster the snapshot grows.

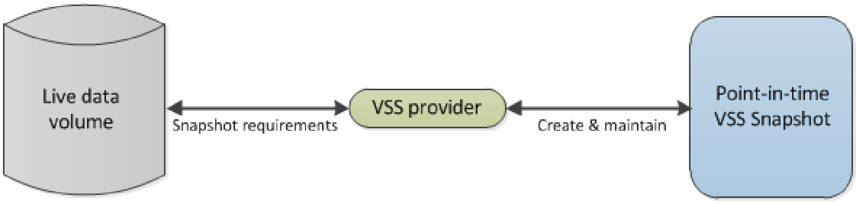
Let's break it down, step by step.

The below diagrams depict the communication between the backup software, the data, the snapshot and subsystems. All of this communication is managed by and through the VSS service.

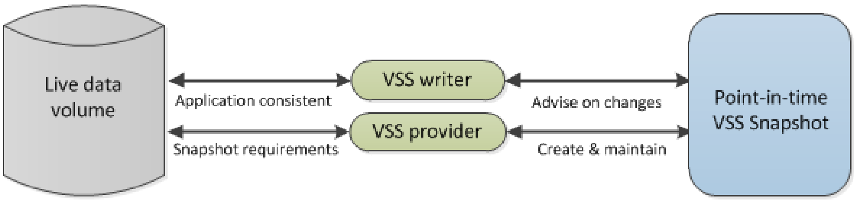
Step 1 The backup job's VSS requester tells the VSS service what it needs and to prepare a snapshot.

Step 2 The VSS service tells the volumes VSS provider(s) what the snapshot requirements are.

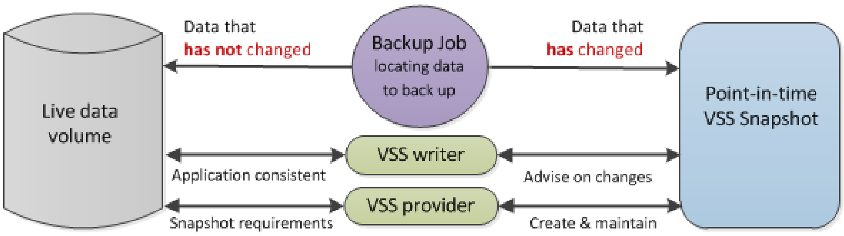
Step 3 A VSS software provider will then make the snapshot and maintain it, as the live data changes.



Step 4 Any assigned VSS writers advise on changes to the snapshot to make it application consistent.



Step 5 The snapshot is complete and ready to use, and control passes back to the VSS requester.



The backup job can now begin requesting data from the VSS service, and create the backup.

Once an application knows that its data has been backed up, its VSS writer will perform some clean-up activities such as clearing its database transaction logs. This can free up disk space and speed up the application.

When you select a VSS application, the backup job uses a variable path that points to both the snapshot and the volume the application is on. The VSS service will determine if the file requested by the backup is in the snapshot or the live data. BackupAssist never knows if the data it backed up came from the live data or the snapshot.

How does BackupAssist System Protection use VSS?

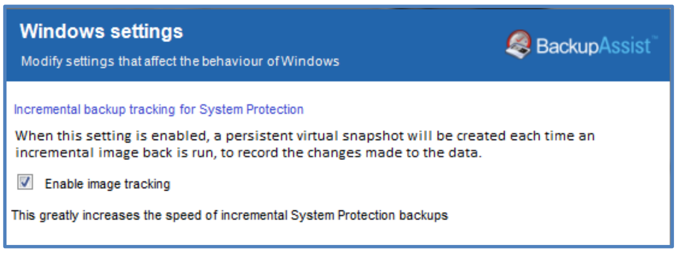
System Protection uses snapshots to produce consistent backups in the way described in the previous sections. However, there are two additional functions that System Protection can use a snapshot for.

Another use for the consistency snapshot

System Protection creates a full image backup when it first runs. If the data selection and destination do not change, the next backup will be incremental. This is achieved by reviewing the data to be backed up and the data in the destination - to see what data changed. What has changed is what is backed up.

Determining what data has changed consumes a lot of time but BackupAssist can do this quickly using snapshots. A snapshot knows what data has changed, because that is what it maintains. So, instead of deleting the snapshot after a backup, System Protection keeps the snapshot and uses it in the next backup to determine what data has changed. This process is called incremental reading.

To maintain a *persistent snapshot* for *incremental reading* select the BackupAssist *Setting* tab >*Windows Settings* > *Enable image tracking*.



Windows settings

Modify settings that affect the behaviour of Windows

Incremental backup tracking for System Protection

When this setting is enabled, a persistent virtual snapshot will be created each time an incremental image back is run, to record the changes made to the data.

☑ Enable image tracking

This greatly increases the speed of incremental System Protection backups

With incremental reading enabled, a 20 min incremental image backup could be completed in 2 min.

This only works if the destination does not change and if data selection does not change. For that reason we advise that the same media is used at least for a week, so the performance gains can be achieved.

Introducing the historical snapshot

The limitation of image backups is that each backup overwrites data that has changed, so you can only restore data from the previous backup. The only way to create a restore point for every backup would be to create a full image backup each time the job runs, and that would waste time and disk space.

This limitation is overcome by creating another persistent snapshot at the backup destination. This 'historical snapshot' is used to makes copies of data that has changed, each time System Protection creates a backup. When a restore is performed, the image backup and these historical snapshots (of data that changed) can be used to provide a restore point for each backup job that ran.

- A historical snapshot can reference the data of previous historical snapshots, meaning they only need to save data that has changed since the last backup job ran.
- The shadow storage space used by the snapshot has a size limit. When the limit is reached, the data referenced by the oldest historical snapshot is deleted.

The historical snapshots are persistent snapshots so the shadow storage space will grow. For this reason it is important that the destination is only used for these backups.