

## Chapter 1

# Data Protection Concepts

*He who laughs last probably made a backup.*

—*Murphy's Laws of Computing*

As a civilization, we humans tend to get attached to our stuff—all flavors, shapes, and sizes of it. A lot of this stuff is probably not worthy of the amount of time and attention we devote to it. When something happens to part us from our stuff, we get in a snit for a while, and then an amazing thing happens: we gradually realize that most (if not all) of our stuff is just clutter, and that our lives are still going on just as nicely as they were before. True, there are some tangible objects that are important and necessary, but in the aftermath of disasters or other life-changing events, we find that suddenly a lot of our stuff just doesn't seem as important as it used to seem. How many PEZ dispensers or fast-food sports-team 32-oz. cups does one person need, anyway?

When we start dealing with the intangible type of stuff we call "data," however, losses can quickly become more catastrophic. Leaving aside threats and dangers such as identity theft, data and information are critical commodities for many businesses. Many workers don't spend a lot of time dealing with data as part of their duties; a barista doesn't need to have a computer to create and serve a 20-oz. triple-shot mocha with whipped cream; a carpenter spends more time cutting, planning, and hammering than sending email—at least, we hope they do!

For those of us who are information technology (IT) pros or information workers, however, data is the lifeblood of the information with which we deal. As with physical objects, not all data is of equal value. Consider the relative value of the following types of information. Be sure to think about the impact on your organization if you were to lose access to this data, and the amount of effort required to reconstruct this data if it were missing:

- ◆ All of the accounts, passwords, and settings for all users in your organization
- ◆ The contents of your mailbox, calendar, and contacts
- ◆ The databases supporting your CRM deployment
- ◆ The databases supporting your ERM deployment
- ◆ Accounting spreadsheets and other financial files on your internal servers

Depending on your organization, some of these types of information will be more critical to your needs than others. As an example, at 3Sharp we would have a minor amount of discomfort if we lost our user accounts; re-creating the list of active user accounts would represent a few minutes' worth of work, and we would collectively spend another handful of hours dealing with issues such as fixing access permissions. However, the loss of our Exchange mailboxes—and years of contact information, documents, and knowledge stored within the hundreds of thousands of messages—would be catastrophic for us.

When we think about protecting our physical assets, we often spend a lot of time and money to do the job. If you don't believe me, just spend a few minutes thinking about how much time we spend both personally and corporately on such tasks as drawing up and paying for insurance policies and premiums, generating and reconciling various types of inventory, or installing and maintaining access control mechanisms such as burglar alarms, deadbolts, and antitheft systems.

All too often, we don't take the same amount of time to adequately protect our data assets. Backup systems have been a key part of IT infrastructures for decades now, but hardly a week goes by without a new story about a backup failure. The concept of disaster recovery has been pushed by vendors, consultants, authors, and speakers for years, yet few organizations have a completely, tested, trustworthy plan for rebuilding critical IT resources and infrastructure from the ground up. If our data is so important to us—and is harder to replace than physical objects, which at least can be covered by insurance policies—shouldn't we as IT professionals take a corresponding amount of effort to protect this information and data, beyond slapping a tape drive onto a server, loading up some backup software, and calling it good?

Microsoft's System Center line of products is designed to give IT pros better tools for managing their IT infrastructure, and the System Center Data Protection Manager (DPM) 2007 product is an important part of this lineup. If you're interested in finding out how to fully protect your critical IT resources—not just performing backups that you're not certain are really worth their time and expense—then this is the book for you. We're going to dig into DPM and reveal all its secrets for you, and give you practical guidance on putting it to work for you and getting the most out of it.

Before we introduce you to DPM in detail, get together over drinks, and make it your new best friend, we need to ensure that we all understand what we're talking about. Let's take some time—we're not in a hurry, after all—and go over some basic concepts that relate to data protection. Do you know the difference between data protection and backup, for instance? If not, no fear—you will after you're done with this chapter. Once we've done that, we'll go on to explore some foundation concepts for DPM that you'll need before we move on to other chapters.

In this chapter, you will learn to:

- ◆ Understand general data protection concepts
- ◆ Identify new concepts introduced by DPM
- ◆ Identify the components in the DPM architecture

## General Concepts

In order to get really excited about all of the benefits that DPM provides—an essential part of making the decision to deploy it—you need to understand how it changes the playing field from the previous generation of products. Because we don't have any way to know what your level of experience in this field is, we're going to start with the basics; we want to ensure that we're all on the same page before diving into the new material.

In this section, we're going to have a brief discussion about the following topics:

- ◆ An exploration of backups and restores: what they are, how they work, what purpose they serve, what benefits they provide, and the weaknesses they have
- ◆ An overview of tape-based backup: why it was originally used, why it's still used, and why it may no longer be suitable for all backup operations

**FOR EXPERIENCED BACKUP ADMINISTRATORS**

At the very least, give this section a quick read-through, so that if we're coming at the topic from a different angle, you won't be surprised down the road when things don't line up the way you might expect. If you're already familiar with these introductory concepts, we beg your indulgence; we know we run the risk of boring you. Even if you've got good backup and restore practices down cold, you may discover a few implications or questions that you might not have fully considered.

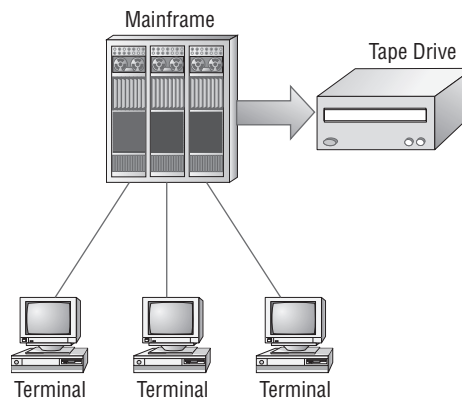
- ◆ A discussion of disk-based backup: what advantages it offers over tape, the potential drawbacks, and how it enables new modes of data protection
- ◆ A handy comparison table of the benefits and drawbacks of tape-based backup versus disk-based backup, suitable for showing skeptical coworkers and managers
- ◆ An overview of the Volume Shadow Copy Service, one of the key technologies used by DPM as well as by other modern backup suites
- ◆ A discussion of data replication: its advantages and disadvantages, as well as a description of common replication mechanisms
- ◆ An introduction to data protection: how it's more than just a good backup plan and what additional areas of concern it includes

We've got a lot of ground to cover, so let's get to it!

**Backups and Restores**

In the abstract sense, "backup and restore" is a simple concept that is nothing more or less than the most basic of common sense: your data is valuable, so make sure you have regular backup copies. This is probably why so many people have spent so much combined time and money over the years (an impressively large amount of money and an even more impressive number of man-hours) to put this simple idea into practice. For many years, backups meant using tape drives (see the "Tape Backup" section for an in-depth discussion of tape drive technology); this strategy worked well when data processing systems were centralized mainframes that held everyone's data (as shown in Figure 1.1) on into the early years of the PC and networking revolution.

**FIGURE 1.1**  
Centralized backups  
on mainframes

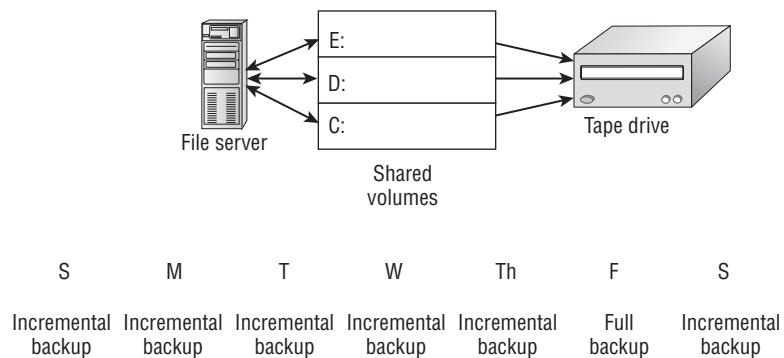


Yet, what starts as a simple idea time and again ends up eating resources and being a continual pain point in our networks. Once data processing moved onto an ever-increasing number of servers and workstations, a centralized tape backup strategy started to be less than optimal. In all the authors' combined years of systems administration, we think it's fair to say that probably the most hated duty is that of overseeing the backup and restore infrastructure. It's the source of a lot of stress and angst, because inevitably the goal is "make sure everything we care about is backed up all the time, but don't spend any money" (or so it seems). The reality is that backing up everything takes too long and costs too much, so you have to start making compromises in your design—and it's all too easy to be criticized for your compromises.

Here are two scenarios we've seen that illustrate the types of design questions and compromises backup administrators must face:

- ◆ A start-up company has a single file server (shown in Figure 1.2). In the event of some sort of hardware failure, management wants as little work lost as possible. In an ideal world, this would mean some sort of continuous backup strategy, but this has been ruled out as too expensive. At the same time, only a small amount of data on the server changes. A daily backup strategy is designed with the following characteristics on a single tape drive: a full backup of the file server once a week written to a separate tape and daily capture each night of files that have changed files since the previous day written to a new tape each week. A manager takes last week's full backup tape to a safety deposit box. In the event of a simple hardware failure, such as the motherboard or a hard drive, the previous night's copy of the data can be restored by retrieving the offsite copy and restoring from each day's backup. If something happens to the entire site, such as a fire, the offsite tape copy can be used to restore the data up to the preceding weekend. Tapes are rotated on a three-month basis to limit the cost of new tapes.

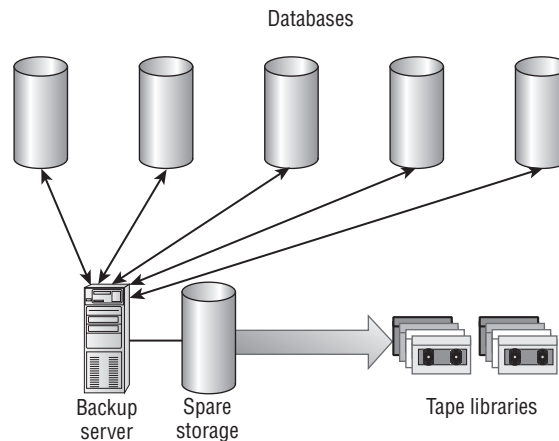
**FIGURE 1.2**  
Backup scenario 1: a single file server



- ◆ A utility company (shown in Figure 1.3) has a large farm of database servers that store a complete set of detailed three-dimensional mapping data for their service area—over a terabyte of data. This database is constantly referred to around the clock by service people, who must not only be able to read the locations of the utility's equipment, but must also be able to make changes to entries based on the status of their work orders. Because the data is located in a database, it is difficult to capture just the data that has changed from day-to-day. While specialized backup applications exist for this type of database server, they conflict with the mapping and work order applications; a native backup interface is used to ensure a nightly backup can be performed without taking the databases offline. However, this

interface is slow, so a full dump of the database is written to spare disk storage every night; this copy is then backed up to two tape libraries that each contain two tape drives for a total of four simultaneous tape streams. Each morning, the previous night's tapes (anywhere from six to ten in a normal day) are picked up by an offsite storage courier, who also returns the tape created three weeks previously. Archived tapes are not rotated, but are kept in a locked room to comply with retention directives established by the legal department.

**FIGURE 1.3**  
Backup scenario 2: a  
database server farm



The previous examples show just a small selection of the difficulties you face when designing the right backup strategy. Rather than spend the entire book talking about the rest of them, we'll summarize these issues for you in Table 1.1.

**TABLE 1.1:** Common Issues Affecting Backup Design

ISSUE	DESCRIPTION
Bandwidth	If you're backing up a single server, this isn't as much of an issue. However, when you start to consolidate backup operations among multiple servers, the amount of available network bandwidth between the machine that holds the data being backed up and the machine doing the backup can become a bottleneck.
Capacity	Each backup tape (or disk volume, if you're using a disk-to-disk strategy as shown in the second example) has a finite amount of space for data. Once the amount of data that you need to back up is larger than this capacity, you need to use multiple backup volumes (tape or disk), multiple backup devices, or both. Using multiple volumes in turn means that either someone must manually load new volumes into the backup device or a more expensive loader device must be used. Multiple devices and volumes complicate both the backup software used as well as the restore process, as data may now be spread across multiple volumes, and the backup software must have some sort of indexing capability (which in turn becomes sensitive data that must be backed up).
Cost	Backup software, devices, and media are not inexpensive. Generally, the more flexibility or capacity you need, the more you can expect to pay. While you can find bargains, they're generally lower in capacity or quality.

**TABLE 1.1:** Common Issues Affecting Backup Design (*CONTINUED*)

ISSUE	DESCRIPTION
Location	Most people think only of network bandwidth here, but there's a lot more to consider. When your data to be backed up lives on another machine, you need to answer a lot of questions. What user account is the backup system using, and does it have access to the data? If the destination is a user workstation, is the user logged off or are critical files (such as financial data) being held open? If the destination is a server, can a single user lock data so that it cannot be accessed by the backup process or are backups performed with a technology that can bypass locks? Early backup designs at 3Sharp were notorious for failing to capture a specific set of highly important files that were often held open by a user application.
Metadata	This is data that is not technically part of the data that is to be backed up, but is related to it in some important way. Examples include access control lists (ACLs) on Windows NTFS volumes, 8.3 filename mappings on shared folders, the backup database indexes used by high-end backup solutions (as mentioned in the Capacity entry), or system state configuration. If you're backing up database or mailbox data, this can also include information from relevant directory services; while it's not directly related to the primary data being backed up, the primary data is useless without the secondary data.
Reliability	Different backup technologies have varying rates of reliability. And there is not always a direct correlation with price; older devices and technologies that survive the market usually do so because they've proven to be trustworthy. The reliability of both the devices and the media must be established separately; when multiple vendors provide the same technology, there may be a marked difference in the reliability between their offerings. Ultimately, the only defense against unreliable media is a regular program of testing the fresh backup copy during the backup process, which in turn reduces the amount of time during the backup window. Media can also go bad during storage or thanks to rough handling, which can diminish the chances of a successful restore.
Security	As mentioned in the Location entry, capturing which user accounts are used (and the access granted to them) is an issue. However, Windows provides specialized access rights for backup and restore operations that bypass many of the typical access restrictions. Accounts with these rights are often valuable targets and susceptible to increased scrutiny. Another issue is whether the backup software allows encryption of protected data or otherwise performs some sort of access control, or does the software allow anyone listening on the network (or who can get their hands on an archived backup volume) to retrieve sensitive data.
Service-level agreements	While not a technical issue, this is often one of the key driving factors in the backup system design. A service-level agreement (SLA) is essentially a commitment to perform a given backup or restore operation within a certain timeframe; it allows the user to know that in the event of a data outage they will have to wait no longer than a well-defined period before getting access to their data again. Quicker SLAs make users and managers happier, but demand more of the backup system and administrators; they must be established with a realistic eye toward the limitations of the technologies used and take into consideration the amount of stress the backup administrators will be under during an emergency. Good SLAs also define the priority level for each type of data and give extra time allowances for the restoration of lower-priority data when higher-priority operations are waiting.

**TABLE 1.1:** Common Issues Affecting Backup Design (*CONTINUED*)

ISSUE	DESCRIPTION
Speed	Backup technologies operate at varying speeds. Often, backup operations must take place within a limited time window during off-peak hours, so backups must be performed as quickly as possible within the budget. Faster technologies and devices tend to cost more, depending on the underlying media.

Beginning in Windows NT and moving forward with Windows Server, Windows XP and Windows Vista offer the following capabilities:

- ◆ The ability to back up and restore to both tape and disk, including removable devices that present themselves as disk volumes at the operating system level
- ◆ The ability to perform a System State backup to capture specialized server-level data such as the Registry, Active Directory databases (if the server is a domain controller), the IIS meta-base, and other critical system data repositories
- ◆ The ability to create Automated System Restore backup sets, which allow a bare-metal backup capability if a server must be completely rebuilt
- ◆ The ability to handle basic backup and restore operations across the network

While Windows Backup goes away in the upcoming Windows Server 2008 (formerly code-named "Longhorn" Server), it will be replaced by the new Windows Server Backup feature, which has most of the same functionality but includes a lot of nice new features.

#### USING WINDOWS BACKUP

Don't automatically turn your nose up at Windows Backup. True, it lacks a lot of the bells and whistles found in more sophisticated and costly packages, such as the ability to schedule operations within the program; you must use the Scheduled Tasks tool provided within Windows or some other third-party utility. However, it covers the basics and can be used to surprising effect. In fact, Microsoft has often used Windows Backup within its own IT infrastructure to help perform daily backup operations on mission critical datasets, usually in conjunction with some enterprise backup package.

If you're upgrading to Windows Sever 2008, or have backups from Windows Server in the data you're protecting, you may be interested to know that the Windows Server Backup replacement will not read the Windows Backup .bkf format. Instead, you'll need to download a free add-on utility, the Windows NT Backup–Restore Utility, from Microsoft Download.

#### Tape Backup

We all know the routine with tape backups:

- ◆ Identify the data that we need to protect by backing it up.
- ◆ Configure the necessary backup filters, schedules, and media.

- ◆ Rotate the tapes so that we reuse tapes containing older data we no longer need.
- ◆ Archive the old tapes we still need for retrieval or historical purposes.
- ◆ Order more tapes to replace the ones that have worn out.

No matter how boring the daily backup routine is, tapes have been with us for a long time; we know the technology, and we know the associated routines. We do our backups, validate the media, test the consistency of our restore processes (at least I hope we all do), and we grumble the whole time. Most of the administrators we know agree that protecting data is one of the most important parts of their jobs. Coincidentally, these same administrators all say it is one of the least appealing parts of their job.

Tape backups have long been the primary choice for enterprises wanting to protect their data. Until recently, no other method offered the flexibility or affordability of tape backups. This, coupled with a thirty-plus-year history of being a known and trusted solution, significant advances in tape technology, and advances in backup software have kept pace with the demands for data protection.

Tape backups have historically offered some important benefits to the typical enterprise environment:

- ◆ **Reliability.** Tapes are known and trusted technology; you've probably already got some sort of tape backup solution in your organization. Tape backup has been around for a long time—since the era of mainframes, computer technology's own era of dinosaurs—and most companies have their tape backup and archival routines firmly in place.
- ◆ **Portability.** Tapes are easily transportable, making it easy to ensure that a copy of critical data can be available in the event of a catastrophe. As mentioned in our second backup example, there are a wide variety of data storage courier services that will come to your location to pick up and drop off tapes.
- ◆ **Scalability.** The scalability of a tape backup system is limited only by two factors: the number of tapes you have and the number of drives you have. With the addition of tape libraries and robots, all of the cumbersome tape handling and labeling tasks can be automated when required in larger organizations.
- ◆ **Cost.** Historically, tape backup was an inexpensive method of protecting data. However, with the cost per gigabyte of disk space dropping rapidly, tape technologies have been hard-pressed to keep up with storage costs, even when factoring in features such as compression support.

Of course, no technology is perfect; tape backup solutions have their share of headaches and drawbacks:

- ◆ **Speed.** Tape backups for larger enterprises can take up a significant amount of time, possibly affecting necessary services on the machines being backed up.
- ◆ **Management.** In many larger environments, compensating for the length of time required to perform backups has led to purchasing multiple tape devices. This leads to very complex backup and restore scenarios due to the additional overhead of the multiple devices, and in the case of restores, collecting all of the correct media for a restore.
- ◆ **Retention.** In the long term, tape media may degrade, rendering blocks of data corrupt or missing and making the tape unsuitable for a restore operation. The risk of tape failures increases over the lifetime of the cartridge; all tape models have a limited number of read-write cycles before they must be replaced.

- ◆ **Testing.** Testing the restore operation of a tape backup system can be complicated, and due to time restrictions in some environments, may require the purchase of additional tape devices, which can be expensive.
- ◆ **Incompatibility.** Tape technology has changed significantly over the years. In enterprises that have existed for a significant length of time, several different types of backup media archive require devices capable of reading the media.

Tape backups have incorporated several different methods for rotating media. Typically, a company will do one full backup of their data each week and a differential backup for daily backup needs. Once a month, a company may archive that month's data offsite, retiring that media from rotation to keep a long term copy for regulatory or corporate policy reasons. This scheme is also known as the Son-Father-Grandfather method, shown in Figure 1.4.

**FIGURE 1.4**

A common tape rotation

M	T	W	Th	F	
D1	D2	D3	D4	W1	Week 1
D1	D2	D3	D4	W2	Week 2
D1	D2	D3	D4	W3	Week 3
D1	D2	D3	D4	M1	Week 4
D1	D2	D3	D4	W1	Week 5
		⋮			
		4 daily tapes		(fifth is used for weekly/monthly)	
		3 weekly tapes		(fourth is used for monthly)	
		13 monthly tapes		(4 × 13 = 52 weeks)	

There are several other common tape rotation schemes, but discussing them is outside the scope of this book.

#### LABELING YOUR BACKUP MEDIA

It never ceases to amaze me how many organizations fail to adequately label their backup media. Imagine going to a library and finding that only every third book had the title and author on the cover and spine. Be sure to label and track all of your media.

#### Disk Backup

In the past, the thought of not having to worry about tape rotation (or the high restore times) has been known to make many administrators weep from sheer joy. Their jubilation, however, was short-lived, only to be crushed under the heel of cost. These sad, depressed administrators went back to their normal tape backup routines, never realizing that the promised land of their proposed disk solution would almost certainly have presented its own difficulties.

Disk-based backups have been historically available for very high-end systems that demanded extreme levels of data availability. These solutions, used when the data was extremely critical, replicate the data from one disk system to another using a block-level copy strategy that makes real-time, synchronous updates. This technology is functionally equivalent to a RAID-1 mirror, but was often used to ensure that a live copy of data was stored in another data center. These solutions were

incredibly expensive, aside from the direct cost of the disk media, placing them out of the grasp of even most enterprise-size organizations.

The price of hard disk technology has rapidly dropped in the last several years, and in response disk-based backups have started to appear in more organizations. One key difference between these solutions and the block-level mirroring described previously is that there is no need for the copying to be synchronous; an asynchronous process is completely sufficient, because the organization's own backup process is the primary consumer for the copied data, rather than user requests or high-demand production applications.

Disk-based backups have become increasingly popular when the amount of data to back up overwhelms the available backup window, or when restoration service-level agreements (SLAs) demand a faster restore time than is possible with tape. Because tape is a serial medium, all of the data written to the tape before the desired set of data must be spooled through; disk, on the other hand, is a random-access medium that permits the restore process direct access to any data that needs to be restored.

Disk-based backups provide a number of benefits to an organization:

- ◆ **Performance.** Disk-based backup solutions tend to offer faster read and write times than their tape counterparts—both due to the inherently faster read/write speed as well as the random access mechanism. Although most disk systems are focused toward increasing the I/O performance, disk-based backup solutions tend to have modest performance requirements, allowing the use of less expensive drives with lower power and cooling specifications.
- ◆ **Availability.** In the event a restore operation needs to be performed, there is no need to physically locate the correct media; instead, you simply need to specify the data to be restored. Where the archive data lives is determined by your organization's own management policies; it can be located on direct-attached storage (DAS), network-attached storage (NAS), or even some sort of storage area network (SAN) technology, making it easy to locate the corresponding backup disk volume.
- ◆ **Lifetime.** Disk, like tape, has advanced significantly over the years; today's high-end server-level drive interfaces offer important performance, feature, and reliability increases (and even workstation-level drives have gotten faster, smarter, and more trustworthy). However, disk interfaces and architectures are generally backward compatible with previous drive generations. Even when a new interface technology is in use, older interfaces can be supported in parallel with little cost or effort. This level of support for older standards makes it considerably easier to transfer archives to newer drives when needed.
- ◆ **Familiarity.** Disk technology, also like tape, has been around for a long time and is well understood by the IT community. Every computer has a hard drive; the principles and techniques of hard drive management are readily available and mastered. Even the care and feeding of advanced disk configurations, such as RAID arrays, has become a commonly available skill in the wake of inexpensive RAID controller solutions intended for small and medium workgroup-level servers.

However, disks have their own drawbacks; they aren't right for all situations, and there are definitely indicators that they may not be right for you:

- ◆ **Lack of portability.** Although tapes are easily moved or shipped from one locale to another, trying to do the same thing with disks can present a difficult challenge. It's easy to take a disk, place it in a padded envelope, and ship it via your favorite overnight service, but we don't recommend

that you do this; antistatic and shock precautions are very important to ensuring the survival of the data at the other end. Simply matching drive interface technologies is often not enough; the lower-level formats produced by controllers of different make and model can be incompatible. If the disks are part of a RAID array, this problem becomes even more pronounced; there may be difficulties in getting another RAID controller to recognize the array.

- ◆ **High initial cost.** The initial cost of disk as media is higher than the cost for an equivalent amount of tapes. Over time, disk's higher level of reliability and support for multiple overwrites makes it the clear winner in the dollar per gigabyte comparison, but the up-front costs of disk controllers, enclosures, and hard drives can be harder for companies on a strict budget to justify.
- ◆ **Increased power and cooling consumption.** Power and waste heat management is a critical part of modern data center operations. Disks are one of the biggest power consumers in a computer, and they contribute a significant amount of the system's total waste heat. Because most of our servers and computer are always powered up, the addition of more drives to the backup solution (or to associated NAS and SAN devices) can have a big impact on the total power and cooling budget.

### Tape versus Disk

Traditionally, enterprises wishing to protect their data from loss relied solely on backup to magnetic tape. In larger environments, the backup process can take eight or more hours. Additionally, the tapes can be quite expensive, take up storage space, and must be changed often. On the upside, tape media is portable, allowing for offsite archival of data.

So how do you know which media is best for you to use in your backup deployment? In Table 1.2 we compare the relative advantages and disadvantages of tape and disk.

**TABLE 1.2:** Comparing Tape and Disk Backups

MEDIA	ADVANTAGES	DISADVANTAGES
Tape	<p>Tape cartridges are portable and require relatively little storage space, which lends itself well to offsite archival.</p> <p>Tape drives and cartridges are available in a number of formats, capacities, and capabilities; it is easy to find a combo that is right for your organization.</p> <p>Tape is a well-known technology with an established history and record of trust; most administrators have ample experience with it.</p> <p>Tape is probably already present in your environment; it represents a significant investment in materials, experience, and archived data.</p>	<p>Tape data must be refreshed or moved from one format of tape to another over long periods of time, and tapes in constant used should be replaced.</p> <p>Tape storage is an expensive storage medium when the total cost (dollar per gigabyte) is considered; drives, cartridges, and replacement drives and media must all be factored together.</p> <p>Tape best practices are not always followed because they increase the time and cost of backup efforts, so many administrators don't know how to minimize and handle media failures.</p> <p>Tape drives usually require an additional interface such as SCSI or SATA, as well as specialized software applications and agents.</p>

**TABLE 1.2:** Comparing Tape and Disk Backups (*CONTINUED*)

MEDIA	ADVANTAGES	DISADVANTAGES
Disk	Tape formats allow for decent levels of compression and storage capacity.	Tape is serial storage; backup, verification, and retrieval are all slow.
	Disk data storage can be reliable for long periods of time, especially if the drives are held to a low duty cycle.	Disks include integrated electronics and take more storage space than tape cartridges.
	Disk provides both random access storage and higher data transfer levels and throughput; backup and restore operations are significantly faster.	Bringing additional drive capacity online usually requires increasingly expensive infrastructure such as RAID controllers and arrays.
	Disk is a very familiar technology for all administrators, so using it for backups doesn't require any additional skills to be learned.	Because disk is so familiar, it may be harder for administrators to develop the proper habits and procedures for volumes used for backup.
	Disk cost has dropped significantly recently, making it a clear winner in the dollars per gigabytes metric.	Disk systems require more electricity and generate more heat than tape systems do.
	Reading archived disks doesn't require special hardware or software, as disk interfaces and formats are generally supported for many years.	Disk restores require all volumes of the relevant disk array to be online at the same time, which may require a compatible array or server configuration to be available.

The recent trend in most enterprise environments is to use a combination of disk and tape to provide full protection:

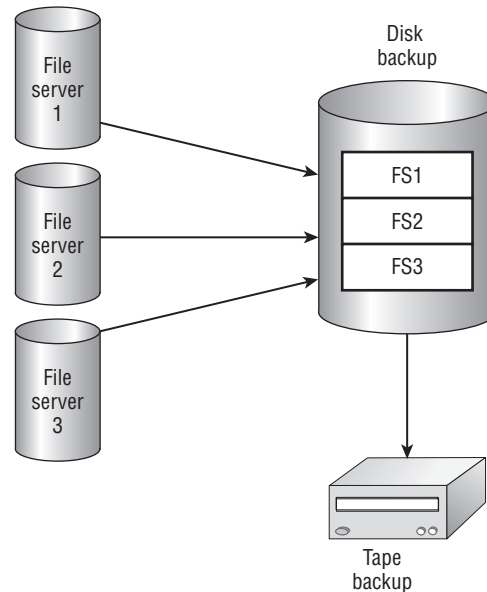
- ◆ Disk offers several advantages that make it a superior choice of media for immediate and short-term backups; its speed allows the backup routines to run more quickly and restore the servers to production in less time while permitting quick, random access restoration of data when it is immediately required.
- ◆ Tape's portability makes it ideal for allowing archival (on or off-site) for longer periods of time such as years; the data can be transferred from disk to tape on the backup server, where it won't impact production use and can be scheduled when administrators are present to oversee operations.

There are several ways that disk and tape can be combined into a single backup solution. The most common deployment option is known as *disk to disk to tape* (D2D2T), which is shown in Figure 1.5.

D2D2T uses one or more disk volumes to perform the initial backup of the live data from the production resource. This backup copy can be generated using a multitude of tools, including tools native to the operating system such as Windows Backup, or using the same software that handles the tape archival. The disk backup can be on a local volume on the protected server or be located across the network on a central machine. Once the disk copy has been created, it is at some point then written

to tape; until the next disk-based backup is created, the previous backup set is available for immediate use in restore activities. In some configurations, multiple generations of disk backups are held on the archive volume and are written to disk only after a suitable time has passed; perhaps only a portion of the data, such as full backups, are archived to tape while incremental and differential backups are removed from the backup volume.

**FIGURE 1.5**  
Disk to disk to tape



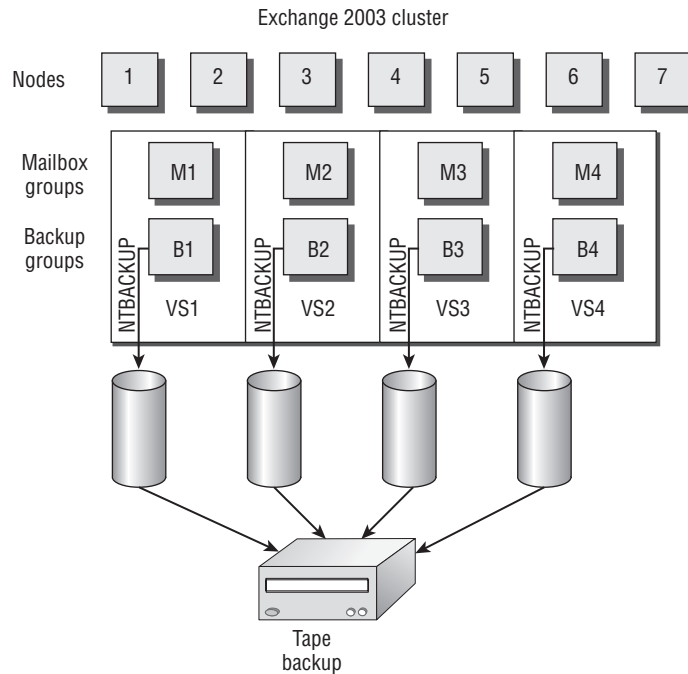
One of our favorite real-world examples of a D2D2T solution can be found in Microsoft's own internal deployment of Exchange Server 2003. Microsoft's Information Technology Group (ITG) faces a lot of unique challenges not seen by many other IT departments of equivalent size; two are particularly relevant for their backup solution:

- ◆ **Dogfooding.** The general Microsoft product development strategy dictates that ITG must use prerelease builds of all major Microsoft applications in their production network as part of the final testing and acceptance trials, a process known as "eating their own dog food." Because they trust production-level data to preproduction software builds, having a reliable backup solution is an absolute necessity.
- ◆ **Heavy usage.** Microsoft's users rely on their email to a degree you have to see to believe. As a result, while there are definite peak times on their mailbox server, there is no good time for downtime such as that caused by typical backup-related outages.

Combined with their aggressive mailbox restore SLAs, ITG deployed a D2D2T solution for backing up their Exchange 2003 mailbox clusters. What's most surprising is what software they used to produce the disk-based backup: Windows Backup. Windows Backup, when run on an Exchange Server, provides support to back up from and restore to storage groups as well as to individual mailbox databases.

ITG's solution, shown in Figure 1.6, uses a Windows Backup instance on each mailbox cluster to back up the relevant mailbox databases to a SAN-based disk volume. This volume is then mounted on a separate server, which is loaded with the backup agent used by their tape archival solution; the backup files created by Windows Backup are copied to disk on a daily basis.

**FIGURE 1.6**  
ITG's Exchange 2003  
backup solution



If ITG needs to perform an immediate restore of a mailbox database, mailbox, or storage group, they can use Exchange's Recovery Storage Group feature to quickly recover the selected data to a live server from the local disk copy of the backup files. From there, they can then move the relevant data back to the online resources. At the same time, they have the long-term data protection afforded by tape, combined with offsite archival.

You can read all of the details of their solution at: <http://www.microsoft.com/technet/itshowcase/content/exchbkup.mspx>.

### The Volume Shadow Copy Service

The Volume Shadow Copy Service (VSS) is a feature first introduced into the Windows server operating system in Windows 2003. VSS is designed to create multiple *shadow copies*, known more commonly as *snapshots*, of one or more volumes. A snapshot is a copy of a set of files and directories as they were at a specific point in time.

By exploiting the ability for the operating system and VSS-aware applications to create multiple snapshots, administrators can produce point-in-time images of critical data as a complete set, ensuring a consistent picture of the data at the time the snapshot was created. These snapshots can then be read by backup applications, allowing this same consistent view of the data to be transferred to long-term storage media while full access continues on the production filesystem.

VSS nicely sidesteps one of the common irritations of conventional backup programs, which can often be negatively affected by users or applications that open files with an exclusive lock. These files can be accessed only by the process that opened them and cannot be read or manipulated in any form by other processes, including backup systems. Exclusive file locks are a constant headache for backup administrators; not only are they a nuisance, they can jeopardize the viability of the

entire backup if the locked files (which are skipped by the backup process) are part of a larger set of data. Imagine the havoc that would be caused by the two following scenarios:

- ◆ An Exchange mailbox database backup captures the .STM database file but not the matching .EDB database file. While the loss of the .STM file can be compensated for, the primary database structure is held in the .EDB file.
- ◆ A SQL Server database backup captures the transaction log file but not the actual database file.

We realize that the above examples are not common: if you're doing Exchange or SQL Server backups, you're almost certainly not trying to do them from the filesystem level against live targets. At the very least, you've taken the protected resource offline before doing this sort of *offline backup*. However, we have seen examples of just these types of mishaps. For backups against live production targets, you're almost certainly using supported backup interfaces to ensure that the backup software will make sure to handle all relevant locks for you and skip the unpleasantness of this file-based approach.

Nevertheless, we brought up these scenarios to make the point that file locks can be more than just a nuisance; they can cause real data loss in your applications. VSS is the answer; a VSS snapshot works at a lower operating system level than the typical filesystem access request, and it creates point-in-time copies of all files on the protected resource. This in turn permits backup applications to use the snapshot to ensure they have a complete, consistent copy of all relevant files in the dataset, whether the application (and backup system) supports a common custom API.

Table 1.3 provides an overview of the various components of VSS.

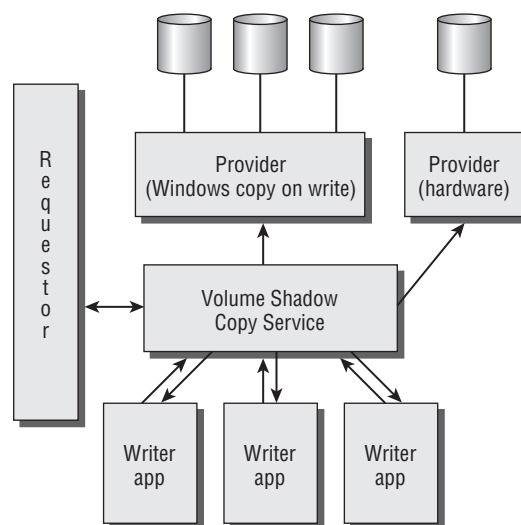
**TABLE 1.3:** Volume Shadow Copy Service Components

COMPONENT	DESCRIPTION
Volume Shadow Copy Service	The service containing the components necessary to create consistent snapshots of one or more volumes
Requestor	Applications such as backup applications that request a volume shadow copy be taken.
Writer	A component of an application, such as SQL or such as SQL or Exchange server, that stores persistent information on one or more volumes participating in shadow copy synchronization. System services like Active Directory can also be VSS writers.
Provider	The provider is a component that creates and maintains the Shadow Copies.
Storage Volume	Shadow copy storage files are placed on the storage volume by the System Copy-On-Write provider. (Note that the storage volume does not need to be the same as the source volume).

A shadow copy snapshot is created by the following process, illustrated in Figure 1.7:

1. The requestor queries the Volume Shadow Copy Service for a list of the writers and gathers the metadata to prepare for shadow copy creation.
2. The writer creates a description in XML of the backup components for the Volume Shadow Copy Service and defines the restore method, and then notifies the writer to make its data ready for a shadow copy.
3. The writer prepares the data via different methods depending upon the data type—completing open transactions, rolling transaction logs, and flushing caches, for example. The writer notifies the Volume Shadow Copy Service when the data is prepared.
4. The “commit” shadow copy phase is initiated by the Volume Shadow Copy Service.
5. The Volume Shadow Copy Service halts I/O write actions on the volume by telling the writers to quiesce their data and freeze requestor writes for the duration required by VSS to create the snapshot. During this time I/O read requests are still allowed; they will not affect the consistency of the data. The application freeze is not allowed to exceed 60 seconds. VSS also flushes the file system buffer to ensure file system metadata consistency.
6. VSS tells the provider to create a shadow copy. The maximum limit on this is 10 seconds.
7. After the shadow copy is created, VSS then releases the writers from their frozen state and all queued write I/Os are completed.
8. The writers are queried by VSS to confirm that the write I/Os were successfully held.
9. In the event that a writer reports that the write I/Os were not successfully held, the shadow copy is deleted and a notification is sent to the requestor.
10. If the I/Os were not successfully held, the requestor can restart the process from the beginning or notify an administrator.
11. In the event of a successful copy, VSS gives the location information for the shadow copy back to the requestor.

**FIGURE 1.7**  
The VSS snapshot  
process



Although the underlying VSS architecture may take a little bit of work to understand, rest assured that Windows and your applications are doing the hard part. The benefits of VSS-aware backups are clear; more and more application and backup vendors are modifying their products to support the use of VSS. A well-written application will hide this complexity from you but make the job of successfully protecting your data (not to mention being able to restore it) much easier.

### MORE ABOUT VSS

VSS was first included with Windows XP. The primary difference between Windows XP's VSS implementation and the Windows Server VSS implementation is that Windows XP can support only non-persistent snapshots, where only one snapshot or shadow copy can exist at a time. Persistent snapshots, on the other hand, permit multiple snapshots to exist simultaneously, giving servers the ability to store multiple point-in-time copies which can then be individually accessed by applications and end users (see Chapter 5, "End User Recovery," for more details). Windows Server 2003 allows VSS-aware applications to create up to 64 simultaneous snapshots per volume.

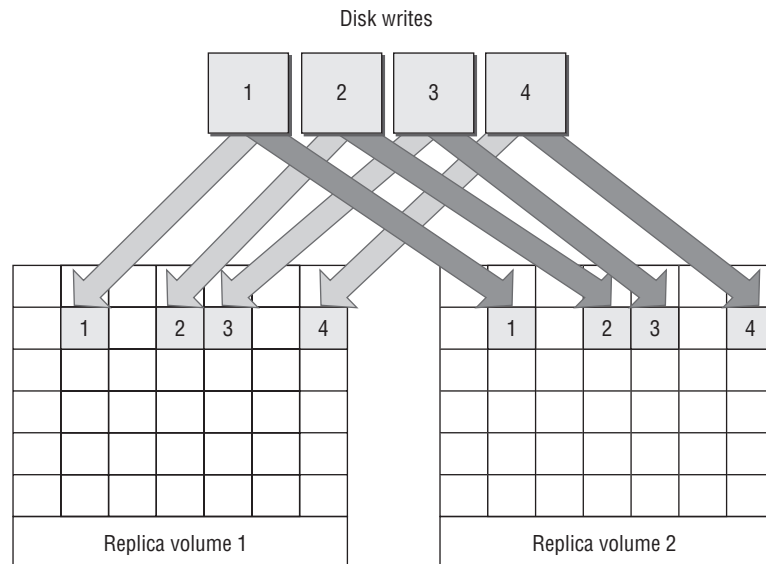
### Replication

In the discussion of disk-based backup, we briefly mentioned the concept of data mirroring, which is an example of *replication*. Unlike a backup, which takes a copy of the data as it exists at a specific point, replication is an ongoing process that keeps a copy of the data synchronized with the original data source.

Of course, it's not that simple (what is?); there are multiple variants and options:

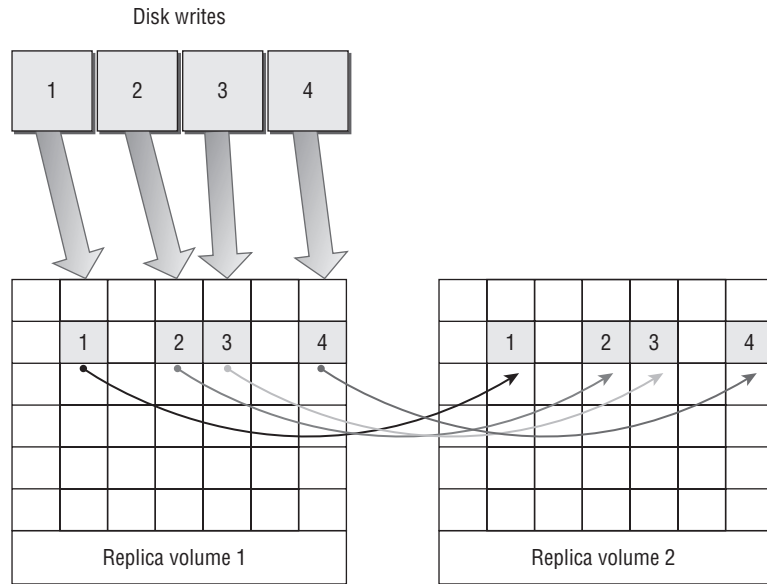
- ◆ Synchronous or continuous replicas (shown in Figure 1.8) write updates and changes to the copy at the same time as they are written to the primary data source. This type of replication requires ample bandwidth between the two replicas and is usually quite a bit more expensive than the alternatives, but both replicas of the data are always up to date—no writes or changes are ever lost.

**FIGURE 1.8**  
Synchronous replicas



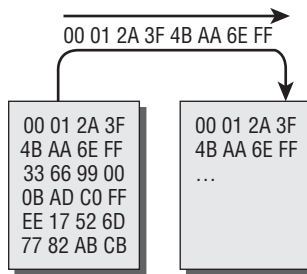
- ◆ Asynchronous replicas (shown in Figure 1.9) are created at specified intervals, such as every 15 minutes; during each interval, changes to the source are queued up for transmission to the replica at the appropriate time. These replicas are a trade-off between the expense of continuous replication and the potential loss of data; the replication interval is set at a value that represents an acceptable compromise.

**FIGURE 1.9**  
Asynchronous replicas



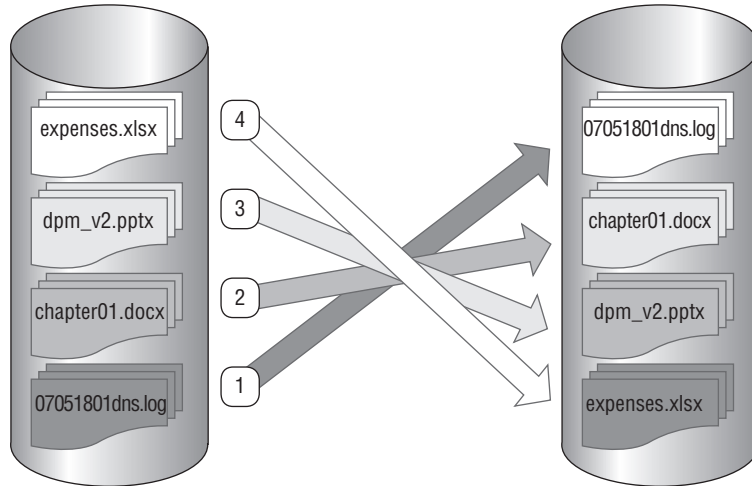
- ◆ Byte-level replicas (shown in Figure 1.10) track all changes in the source at the level of the individual byte. This type of replica requires specialized hardware and software to allow capture of changes at this level of granularity but produces the least amount of replication traffic. They are pretty much unheard of in typical implementations and are usually reserved for very expensive or very important storage installations, such as those used to store critical military data.

**FIGURE 1.10**  
Byte-level replicas



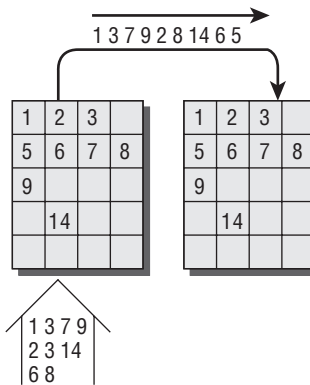
- ◆ File-level replicas (shown in Figure 1.11) operate at the file-system level, tracking changes at a file level. At this level, changes are easy to track and can be performed by unsophisticated software, but it's also fairly inefficient (or even impossible) for certain types of data such as Exchange and SQL Server. The change of a single byte within a file will result in the entire file being recopied to the target replica.

**FIGURE 1.11**  
File-level replicas



- ◆ Block-level replicas (shown in Figure 1.12) represent a compromise between file-level and byte-level. Almost all forms of computer data storage, whether on an NTFS filesystem or inside an Exchange or SQL Server database, are organized into discrete units known as *pages* or *blocks* (usually between 512 and 2,048 bytes). By using a special filter, replication programs can track which blocks have been updated and transmit only those blocks to the target replica. Although the entire block will be transmitted even if only a single byte is updated, most on-disk files consist of hundreds or even thousands of blocks, making this a more than acceptable compromise for almost all situations.

**FIGURE 1.12**  
Block-level replicas



Replication is only tangentially useful for traditional backup and restore processes, and it is commonly seen more in availability solutions such as Windows Distributed File System (DFS).

## Data Protection

We've now set the stage to have a meaningful discussion about the concept of *data protection*, which is a merging of technologies between mere backups and high-end disk mirroring solutions. Data protection is more than just taking the occasional (or regular) copy of your data "just in case."

All of the capabilities we've just discussed have their place in a full Windows-based data protection solution:

Capability	Benefits
Backup and restore	These capabilities are important for ensuring business continuation.
Tape backup	Tape archives help ensure long-term storage capability balanced with portability.
Disk backup	Disk archives provide rapid short-term restoration capabilities and ensure shorter backup windows.
Disk to tape	D2D2T provides a reliable transition between short-term and long-term archival media.
VSS	VSS provides an underlying mechanism to enhance data consistency without compromising service or data availability.
Replication	Replication allows automatic capture of critical data sources balanced by only a small amount of data loss if a server or site is lost.

These benefits are a good starting point, but by themselves, they don't offer much beyond a modern backup solution. However, there are still some key benefits provided by a true data protection solution, and they are still not in place:

- ◆ A set of consistent, repeatable, management capabilities, usually implemented through a policy-base configuration engine
- ◆ The ability to consistently define and apply protection schedules across multiple data sources
- ◆ A single, centralized interface to protect multiple types of data sources in a consistent manner

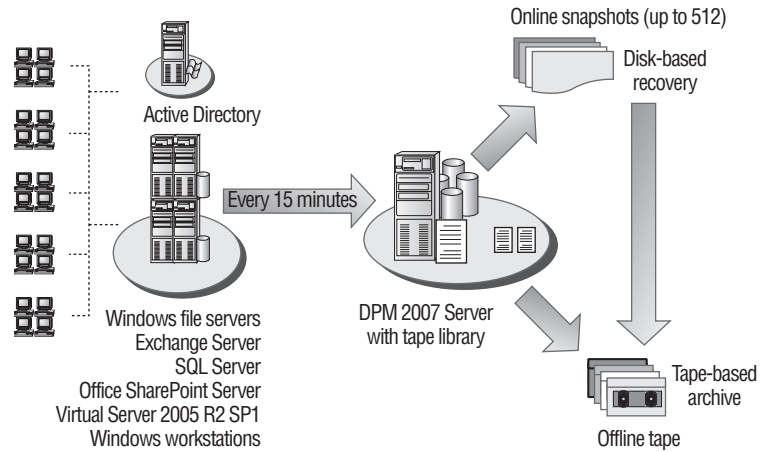
Happily, DPM gives us these benefits and more—as we'll see in the next section.

## DPM Concepts

Now that we've discussed data protection in the abstract, let us move on to an examination of how these concepts are implemented in DPM.

DPM provides a combination of data replication and archival functionality. It incorporates many features commonly seen in advanced backup applications, such as D2D2T and centralized management for multiple data payloads such as Microsoft SQL Server, Exchange Server, Microsoft Virtual Server, and Microsoft SharePoint Services. Unlike traditional backup solutions, however, it combines long-term tape archival with seamless short-term disk replication and storage management, as shown in Figure 1.13.

**FIGURE 1.13**  
A typical DPM solution



Before you can design and implement a DPM solution, there are several building blocks you must understand:

- ◆ The creation of replicas
- ◆ The DPM storage pool
- ◆ The use of protection groups
- ◆ The creation of recovery points
- ◆ The use of end-user recovery

We will examine these concepts in further detail.

## Replicas

The feature that most distinguishes DPM from a typical backup application is its integrated replication engine. When installed into an organization, DPM creates replicas of protected data sources and performs regular asynchronous replication with these sources. All further protection operations take place on these server-side replicas, allowing the original data sources to enjoy continuous uptime.

DPM can use several replication strategies, depending on circumstances and the type of data being protected. Each of these methods has implications on the use of available storage space, which we will cover in more detail in later sections.

## Storage Pool

In some data backup setups, the tape-rotation scheme is enough to make you cringe at the amount of space that goes unused during the daily backups. At some point, you might even develop a complex scheme of doing daily backups to disk and then archiving the results to tape. This is usually more trouble than it's worth. Your backups may not even be completely reliable because you're backing up the backups. You could move from that system to a large robotic tape library, with a third-party backup solution that offers some nice features, but that solution is expensive and slow. This is a typical example of the kind of balancing act IT professionals have to perform time and again.

**TABLE 1.4:** DPM Replication Strategies

REPLICATION STRATEGY	DATA TYPE	DESCRIPTION
Express full backup	All	This method is used both to create the initial replicas when a new data source is first added to DPM protection. It is also used to provide regular updates to an existing replica, usually corresponding with the creation of a new VSS snapshot. When this is performed against an existing replica, DPM uses block-level replication to minimize the replication traffic.
Replication	File data Databases Virtual machines System state	The Protection Agent monitors file writes via a volume filter and performs block-level replication at defined intervals to capture the changes in protected files and folders.
	Exchange	The Exchange storage group transaction logs are captured at regular intervals and copied to the DPM server. With these logs, the DPM server can perform log replay to any specified time and perform data recovery.
	SQL Server	SQL Server transaction logs are captured at regular intervals and copied to the DPM server. With these logs, the DPM server can perform log replay to any specified time and perform data recovery.

Fortunately, DPM removes these problems, allowing us to focus on the more important matters at hand—such as which of our servers has enough spare resource overhead to take on the task of functioning as our Quake server.

The *storage pool* is a key DPM concept. It is a collection of the available disk volumes on which DPM stores all of the data associated with protected resources, such as shadow copies, replicas, and transfer logs. A DPM server requires at least two physical disks volumes, one for the operating system (OS) and program files, and one (or more) for the disk pools. DPM will not add any disk that contains operating system or DPM files to the pool.

The main advantage the DPM storage pool provides is reduced administration. By default, DPM will manage the storage pool for you, taking care to reserve space for protected resources as required. When you get low on space, all you have to do to expand the storage pool is add another volume to

### WHAT IS A PHYSICAL DISK VOLUME, ANYWAY?

When we refer to a *physical disk volume*, it means any disk volume that shows up as a separate device in disk management. This includes direct attached disks, SAN LUNs, iSCSI LUNs, and other disk types.

the pool; DPM will automatically allocate the new space for protection. If you like to fine-tune things manually, it is possible to change the allocation yourself. Generally, you want to change the storage recommendations only if your storage pool space is mostly utilized and you can't quickly add new volumes after the fact.

DPM recommends and allocates storage pool space for a protection group based on the amount of data to be protected. The replica, snapshots, and transfer logs are all stored in allocated space within the storage pool, and Microsoft recommends not changing the default allocations unless they do not meet the needs of your organization. These recommended values are not static, however, and may be modified according to the following limitations of the three protection components:

- ◆ **Replica and shadow copies.** Allocation for the replica and shadow copies may be increased, but not decreased. If you need to increase this allocation, we recommend that you first verify that the increase is going to be a consistent and ongoing change. If not, try to find a way to reduce the amount of data being protected before reallocating the space.
- ◆ **Synchronization log.** Space for the synchronization log can be increased or decreased, but bear in mind that it resides on a disk volume on the protected server. Changing this value may negatively impact storage space on the server you're trying to protect.
- ◆ **Transfer log.** The allocated space for the transfer log cannot be directly modified. DPM, however, does adjust the allocation based upon the other allocation settings. Due to this, the changes you make to the other components will affect the transfer log allocation.

DPM also includes tools to monitor the utilization of the storage pool and generate reports against the usage data. These tools are important aids for capacity planning.

### Protection Groups

As Windows administrators, we should all be familiar with the concept of groups. We see groups and containers all over the place in a Windows network:

- ◆ Standalone Windows servers use *local groups* to hold collections of users; Active Directory uses *security groups* and *distribution groups* to hold collections of users, computers, contacts, or other types of objects. You can use these groups as placeholders when assigning permissions, determining recipients of an email, or many other uses.
- ◆ Active Directory also uses a special type of LDAP object called an *organizational unit (OU)* that is tied to a specific point in the LDAP hierarchy. Unlike a group, OUs cannot be used to assign permissions, but they can be used as administrative boundaries to delegate management permissions and apply Group Policy Objects.
- ◆ Exchange Server 2003 defines the *administrative group (AG)*, which acts in a similar fashion to an Active Directory OU. When you delegate permissions to the AG, or create a policy and apply it to the AG, all of the Exchange servers within that AG are affected by the action.

- ◆ Exchange Server 2003 also defines the *routing group* (RG), which defines a collection of Exchange servers that have sufficient local bandwidth between them that they can always send messages to each other directly. Two Exchange servers in different RGs will always use connectors and bridgeheads to route messages and never contact each other directly (unless, of course, they happen to be the bridgeheads for their respective RGs).
- ◆ SQL Server 2000 and SQL Server 2005 define the *user group*, which allows SQL administrators to easily assign permissions on databases and other SQL objects to multiple users at the same time—just like Windows/AD groups.

By looking at all of these examples (and others too numerous to list here), we see three useful concepts taking form:

- ◆ Groups act as a multiplier of effort; we take a single action, such as granting permission or defining a policy, on a group; that action is then replicated to all members of the group. This can drastically reduce the time we must put into management.
- ◆ Groups act to guarantee consistency; once we know how one member of the group will behave when under conditions relating to the group's definition, we know how they all will act. This can radically reduce the effort we must put into troubleshooting.
- ◆ Groups act to simplify expansion; once the group is defined bringing new resources into congruence with existing resources is simply a matter of adding the new resource to the right group. This gives us enhanced protection from configuration errors.

DPM uses these basic precepts of taking several objects and putting them together in a container object so that the same policies may be applied to them. A *protection group* is an administratively defined group of data sources that share the same protection configuration and schedule. The elements of a protection group are shown in Table 1.5:

**TABLE 1.5:** Elements That Define a DPM Protection Group

ELEMENT	DESCRIPTION
Members	<p>A member is a data source you want to protect. A single server may have more than one data source.</p> <p>A file server has one or more volumes, each of which has one or more shares.</p> <p>A SQL server has one or more instances, each of which has one or more databases.</p> <p>An Exchange server has one or more storage groups.</p> <p>A SharePoint farm has one or more servers, which can be spread out over multiple tiers.</p> <p>A Virtual Server host has one or more virtual machines.</p>
Data Protection Method	<p>The method used by the protection group to protect your data. The method can be disk and/or tape.</p>

**TABLE 1.5:** Elements That Define a DPM Protection Group (*CONTINUED*)

ELEMENT	DESCRIPTION
Short Term Objectives	The parameters control how often recovery points are created, how long they're retained, and when express full backups occur.
Disk Allocation	The amount of storage pool space allocated to the protection group.
Replica Creation Method	This specifies when replication of the protected data to the DPM server takes place. The default option is to have it happen automatically. Additionally, you can schedule it or specify manual replication using removable media.

The biggest key to comprehending why DPM protection groups are so neat is a clear understanding of the term “data sources”:

- ◆ A single file server in your organization may have multiple data sources on it to protect. For example, if you have a file server with multiple volumes, and one volume sees much more activity than the other, you may want the creation of recovery points to occur more frequently on that volume. In that case, the volumes would be members of separate protection groups.
- ◆ An Exchange storage group would be considered a single data source. If you have multiple storage groups, they could each be members of separate protection groups. Even though storage groups can contain multiple databases, you cannot choose to protect individual databases because each storage group shares a single set of transaction logs.
- ◆ An individual SQL database also qualifies as a data source. If you have a single SQL server hosting multiple databases with differing protection requirements, then it would make sense to house them in different protection groups.
- ◆ A SharePoint farm is a single data source, no matter how many servers you have as part of that farm or how many content databases are included. You cannot split the databases or servers within a farm into multiple protection groups.
- ◆ On a Microsoft Virtual Server host, each separate virtual machine image is a separate data source. You can place virtual machines that are on the same host in separate protection groups to best match the protection needs of the data and applications they host.
- ◆ Workstations running Windows XP and Windows Vista may have multiple data sources, just like file servers. However, unlike file servers, the various data sources on a workstation must all be protected in the same protection group; you can't split them into multiple protection groups.

One natural application of protection groups is to define a separate group for each type of data source that you are protecting. This is an instinctive choice for many organizations and administrators; mailbox data often has different requirements than database data, and both are to be treated differently than file server or SharePoint data.

For ease of administration, you may find it simpler to group data by purpose and protection requirements, rather than trying to lump many disparate types together. It all comes down to your environment. This technique permits the application of a single protection policy to multiple types of data sources when their protection characteristics and priorities are the same, in turn simplifying the creation and management of your protection policies.

For example, if you have a SQL server with multiple databases or instances (as in a hosting environment), your data protection requirements may differ between databases. In this case, narrowing down the requirements to a few groups and putting the databases into the group that represents the best fit for their requirements keeps administration simple.

There are a few restrictions to keep in mind when planning your protection groups:

- ◆ Data sources can be members of only one protection group.
- ◆ File shares residing on the same volume cannot be members of separate protection groups.
- ◆ When you select a folder or share for protection, its children are automatically selected and cannot be deselected.
- ◆ When you select a location that contains a *reparse point* (mount points and junction points are two examples of reparse points), DPM will prompt you to protect the target location, but will not replicate the reparse point itself. After recovering the data, you must manually recreate the reparse point.
- ◆ If you select system volumes or program folders, DPM will not be able to protect the system state of the machine as a separate data source.

The general rule of thumb when designing your protection groups is to use as few as you need; unless you have a specific reason why you shouldn't include a resource in an existing protection group, don't create a new protection group.

## Recovery Points

There comes a point in every administrator's life when data recovery needs to occur. It could be due to any number of causes: hardware failure, unrecoverable software issue, a security problem, a natural disaster, or even Godzilla attacking your data center. Whatever the cause, you need to be able to get your data back; otherwise, why are you even bothering with those tedious backups in the first place?

A *recovery point* is a snapshot that represents the state of data at a point in time. The use of persistent snapshots by VSS and DPM means that there can be more than one version of the data available for restore. Note that recovery points are not tied directly to the underlying VSS snapshots, depending on the payload being protected; Exchange data, for example, creates a recovery point every 15 minutes even though it doesn't perform an express full backup that frequently. Instead, DPM captures the transaction logs and can replay them to duplicate the state of the protected database.

Recovery points are specific to a protection group. During the creation of a protection group, you will be asked to specify a protection policy. This means that you will be asked to set the following parameters:

- ◆ **Retention range.** This parameter determines how long a snapshot should exist on the DPM server's storage pool. When the age of the data exceeds this value, DPM will transfer it to tape (if the policy permits).

- ◆ **Synchronization frequency.** This parameter specifies the synchronization schedule for the replica and controls how often the protection agent will send the block-level updates to the DPM server, up to every 15 minutes.
- ◆ **Recovery points.** This parameter specifies how often recovery points are created, up to every 15 minutes. There is always a “Latest” recovery point, representing the last synchronization performed that does not correspond to a defined recovery point.

When you create a new protection group, DPM by default creates three daily recovery points: 8:00 AM, noon, and 6:00 PM. Depending on how often data changes in your environment and what level of data loss you deem acceptable, you may want to change this schedule. For example, in a business that keeps mostly to an 8 AM to 5 PM, Monday through Friday schedule, the default values probably wouldn't make much sense; such an organization might want two or three restore points spread through the day on working days.

While protection groups associate settings and schedules with protected data and locations, the recovery process remains blissfully ignorant of this arrangement. Microsoft has made data recovery simple with DPM, while at the same time giving it flexibility found in few traditional backup applications. While in the recovery section of the administration console, data is organized by server. When you choose some protected data to recover, you don't have to know the ins and outs of the protection group it was a member of; you just need to know what data you need and where it should be. DPM automatically populates the recovery points for the data you are trying to recover, so you can select the appropriate point in time to recover.

#### WHY DO I NEED BOTH SYNCHRONIZATION FREQUENCY AND RECOVERY POINTS?

At first, it seems as though specifying both your synchronization frequency and explicit recovery points is redundant. After all, if you're replicating your data every 15 or 30 minutes, isn't that good enough? Depending on your organization, the answer may very well be, “Yes.” For many, though, that's not the case.

When you go to restore data from a DPM protection group member, you will be asked to pick which recovery point you want to use. No matter what schedule you've set, you will always see the “Latest” entry. This entry represents the last replication of the protected member *that does not correspond to a recovery point*. This is important, as DPM will not permanently store intervening synchronizations. That is, a DPM recovery point is roughly comparable to a traditional full backup; when you restore from a recovery point, DPM doesn't need to take any other replica or synchronization data into account. Extending this metaphor, a synchronization is analogous to a traditional differential backup; you first restore from the full backup (recovery point), then you restore the latest differential backup (synchronization) to get the latest version of the protected data.

This will be a lot easier to understand with an example, so let's examine the case of a protection group configured with a 15-minute synchronization schedule and the default 8:00 AM/12:00 PM/6:00 PM recovery point schedule:

- ◆ At 8:00 AM, the protection agent replicates the changed data to the DPM server. Because this also happens to be a recovery point, DPM creates a new replica to write the data to. For the next 15 minutes, “Latest” and “8:00 AM” will both point to the same replica.
- ◆ At 8:15 AM, the protection agent replicates the next batch of changed data. “8:00 AM” is still a discrete recovery point, so DPM allocates a new chunk of storage to hold the next replica of the data. This replica can be restored by choosing “Latest.”

- ◆ At 8:30 AM, the protection agent replicates the next batch of changed data. The last set of data is from 8:15 AM, which is not a configured recovery point, so DPM updates that replica with these changes. The “8:00 AM” recovery point is still available, but “Latest” now points to the data from the 8:30 AM replication. The data from 8:15 AM can no longer be selected, even indirectly, as a recovery point. This continues every 15 minutes up through 11:45 AM, with DPM applying the updates to the latest replica.
- ◆ At 12:00 PM, the protection agent replicates the changed data to the DPM server. Because this also happens to be a recovery point, DPM creates a new replica to write the data to. For the next 15 minutes, “Latest” and “12:00 PM” will both point to the same replica.
- ◆ At 12:15 PM, the protection agent replicates the next batch of changed data. “12:15 PM” is still a discrete recovery point, so DPM allocates a new chunk of storage to hold the next replica of the data. This replica can be restored by choosing “Latest.” All further replications through 5:45 PM will update this replica.
- ◆ At 6:00 PM, the protection agent replicates the changed data to the DPM server. Because this happens to also be a recovery point, DPM creates a new replica to which to write the data. For the next 15 minutes, “Latest” and “6:00 PM” will both point to the same replica.
- ◆ At 6:15 PM, the protection agent replicates the next batch of changed data. “6:15 PM” is still a discrete recovery point, so DPM allocates a new chunk of storage to hold the next replica of the data. This replica can be restored by choosing “Latest.” All further replications through 7:45 AM the next morning will update this replica.

Clear? Put in very simple terms, separating the replication and recovery point schedules permits you to define how much data you’re willing to lose in the event of an outage, right up to the limit of DPM’s 15-minute granularity, while at the same time giving you explicit control over how much storage to use for your recovery points.

Having explained that, we don’t think that it makes a lot of sense to create a recovery point both at the end of the day and the beginning of the next day. If nobody’s changing the data during those hours, what is the point of using up storage space for another replica? We’ll talk later in Chapter 12, “Advanced DPM,” about the specific considerations you’ll want to review when picking appropriate times for recovery points.

## End-User Recovery

Most of us have had to deal with non-disaster-related recoveries of user data. We get the request, say a few less-than-polite words about the user under our breath, find the media, read it into the drive, and recover the data. Things have gotten a bit better with the ability to make shadow copies via the VSS functionality in Windows Server 2003. This enables us to deploy the VSS recovery client to end users, give them a little education on its use, and let them recover from accidental loss themselves.

Data Protection Manager can be enabled for end-user recovery of data that exists in protected locations. Using the VSS client application, users can browse through previous versions of files and folders by the corresponding restore point.

This functionality is similar to what you get when you turn on VSS for a volume on a server. The main difference here is that the snapshots are not taking up space on your production file volumes.

End-user recovery brings many benefits that are not present with traditional backup and restore scenarios including:

- ◆ **Self service.** Users don’t have to contact IT to recover their data. It’s a simple process that they can accomplish themselves.

- ◆ **Instant recovery.** The recovery of the files when initiated by users happens when they initiate it, not after submitting a request to IT and waiting for an administrator to find the time to retrieve the media, load it, and find the data the user wanted.
- ◆ **Improved efficiency.** Because the IT department does not have to be contacted, their time is spent more efficiently.

However, bear in mind that end-user recovery does have some potential drawbacks:

- ◆ Only SharePoint documents and files and folders on protected file shares can be enabled for end-user recovery. Other data types such as SQL Server and Exchange cannot be enabled for end-user recovery.
- ◆ End-user recovery relies on the VSS client tool. This means that you have to deploy it in your environment, as well as train your users how it is used and make sure they understand its limitations.
- ◆ Users may inadvertently overwrite the current version of a file with an older version.

We will discuss end-user recovery in detail in Chapter 5, “End User Recovery.”

## DPM Architecture

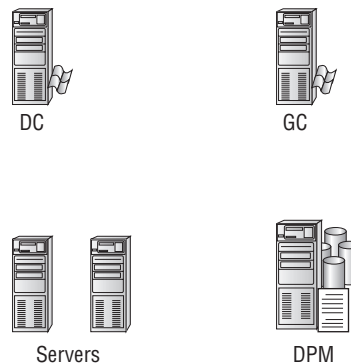
With a better understanding of how DPM works, we can finish the chapter with a look at how the pieces of the DPM solution fit together.

DPM uses the following tiers:

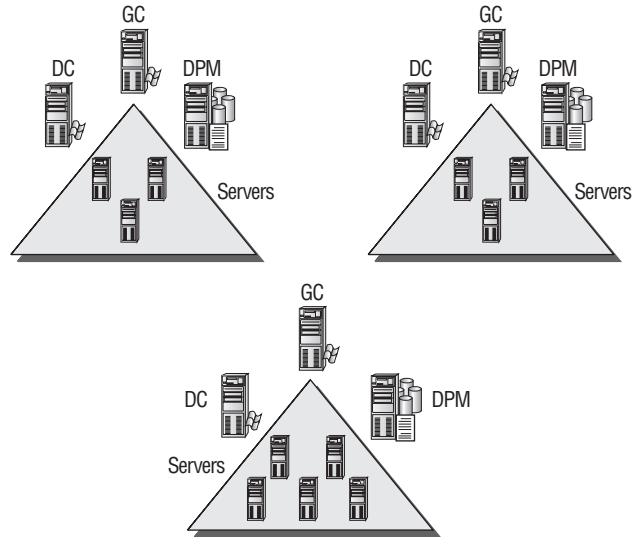
- ◆ The protection agent is a service that resides on each protected server, performing replication and restore operations on behalf of DPM.
- ◆ The DPM server application runs on one or more dedicated servers, providing centralized scheduling, policy creation, and management, as well as serving as the repository for replicas of protected data and the location of tape operations.
- ◆ Optionally, DPM can interact with third-party backup software, providing an additional level of protection.

Figure 1.14 shows a typical single domain DPM deployment, and Figure 1.15 depicts a more complicated enterprise installation.

**FIGURE 1.14**  
A single domain DPM deployment



**FIGURE 1.15**  
A complex DPM deployment



Let's examine each tier in more detail.

### The Protection Agent

Chances are, if you're reading this, that you're familiar with traditional backup methods, and more than one third-party backup application. Many third-party applications include "agents." Agents, in these cases, are small software applications that reside on client machines and target specific data types. Most backup solutions require a separate agent to cover Exchange, SQL Server, and any data other than flat files.

In DPM there is only one agent, the *protection agent*, which handles all of the protection responsibilities on protected servers. The protection agent is a small client application installed on servers being protected by DPM. It uses a special disk volume filter that hooks into the Windows Server storage drivers, allowing it to track the block-level changes to resources it has been configured to protect. A client with the protection agent installed may be managed by only one DPM server and cannot be protected by multiple DPM servers.

The protection agent performs the following functions:

- ◆ Maintains the synchronization logs and records changes to selected resources on the protected server. The protection agent maintains a separate synchronization log for each protected volume. The synchronization log is located in a hidden folder on the volume to which it pertains.
- ◆ Copies the synchronization log to the DPM according to the configured schedule. Once DPM has a copy of the synchronization log, the data can be synchronized with the DPM server's replica and appropriate recovery points and VSS snapshots created.
- ◆ Performs express full backup synchronizations when scheduled or requested, including when a new data source on the protected server is first included in a protection group.
- ◆ Handles communications with DPM to allow DPM to browse the shares, volumes, and folders on the protected server during recovery operations.

There are two components to the protection agent; the agent itself and the *Agent Coordinator*. The Agent Coordinator component is temporary software that is used during installation, upgrade, or uninstallation of the protection agent.

### DPM Server

If the protection agent tier is the eyes and ears, the DPM server tier is the brains. The server tier performs the following functions:

- ◆ Hosts the DPM application. This application provides the policy engine, scheduler, and data repository.
- ◆ Hosts the SQL Server instance that provides the index of all data sources, replicas, protection groups, and other DPM configuration information.
- ◆ Manages the storage pool, which must be presented to the server tier as some sort of locally attached storage.
- ◆ Creates and updates the replicas with the data received by the protection agents.
- ◆ Captures the VSS snapshots on each replica according to the configured schedule of the appropriate protection group.
- ◆ Moves replica data to tape according to the long-term retention policies defined by the appropriate protection group.
- ◆ Retrieves replica data from tape and disk in response to recovery operations and sends it to the appropriate protection agent.

As mentioned previously, DPM should be installed on its own server. As you can tell from the previous list, it is responsible for performing a large number of tasks even in a simple environment; when you are protecting multiple sources, the DPM server will become very busy. The major bottleneck for most DPM operations is the disk subsystem, but RAM and CPU are important as well, especially for the underlying SQL Server instance DPM uses to track protected data. In larger environments, you can configure DPM to use a separate SQL Server instance.

We'll cover the particulars of installing DPM in all its glory in Chapter 2, "Installing DPM."

### Third-Party Backup Software

The final tier in a DPM system is completely optional. Out of the box, DPM provides native support for reading and writing protected data to tape devices ranging from single tape drives to large, expensive libraries. It will even do its best to create an appropriate tape rotation schedule for you, relieving you of the burden of having to decide the best way to handle your tape volumes.

However, DPM does suffer from a serious limitation: it protects only Windows servers. If your organization is all Windows all the time, this probably doesn't concern you that much; DPM offers enough functionality that it makes a compelling choice to protect all of your Windows servers and data (and Microsoft certainly hopes that's how you'll use DPM).

In many companies, though, they've got all sorts of other servers and operating systems, ranging from legacy mainframes to Unix and Linux servers and more. Many of the bigger backup solutions offer a high degree of cross-platform capability; they can support backup and restore operations from a large number of operating systems, hardware architectures, and third-party applications. For example, a single backup application might handle the following workloads:

- ◆ Windows desktops and laptops running a variety of versions of Windows, including Windows 95, Windows 98, Windows ME, and Windows 2000

- ◆ Windows NT4 and 2000 domain controllers
- ◆ Windows NT4 and 2000 member servers
- ◆ Sun Solaris servers running Oracle databases
- ◆ UNIX or Linux workstations running various user applications
- ◆ Novell eDirectory servers providing directory, file, and print services
- ◆ Novell Groupwise messaging servers

If your organization uses one of these applications, you've likely got a large amount of time and effort (not to mention money) invested in it.

DPM is designed to provide value to you even if you've already got a backup solution you want to keep. Use DPM to protect your Windows assets; use your existing application to back up the data from the DPM replicas. This integration provides several benefits:

- ◆ Your centralized tape-handling procedures need no modification; all tape operations are still done the same way you're currently doing them. However, your Windows machines still get all the benefits of DPM, including the ability to offer end-user recovery for appropriate payloads.
- ◆ You can reduce the number (and type) of expensive backup agents for your tape backup solution. Instead of having to buy a license to back up each SQL Server machine or Exchange server (let alone Windows file servers), you simply need a file agent to back up each of your DPM servers.
- ◆ Windows administrators can perform their own local restores from the DPM replicas without having to drag the central backup administrators into it. This is an especially lovely prospect for Exchange and SQL Server administrators; it makes recovering deleted executive mailboxes much less painful for all parties concerned.

One caveat you need to keep in mind when using DPM with a third-party tape solution: your tape agent must support the use of VSS. DPM replicas rely heavily on VSS capabilities, and you will not be able to perform reliable backups without it. This shouldn't be a problem; almost every backup solution out there with Windows support now offers VSS compatibility.

#### **PROTECTING NON-WINDOWS SERVERS WITH DPM?**

When we say DPM protects only Windows servers, we're not being completely accurate. We feel bad about that, so let's set the record straight: DPM can protect any server that runs in a Microsoft Virtual Server virtual machine (VM). So, you *can* protect VMs running any supported operating system such as Red Hat Linux or Novell's SuSE Linux. However, not many people run all of their non-Windows servers in Virtual Server VMs—and if you're one of the odd ones who do, there are still some caveats we'll explore further in Chapter 10, "Protecting Virtual Servers."

## The Bottom Line

**Understand general data protection concepts.** Understanding the concepts that apply to any data-protection scenario makes it easier for you to identify the challenges you face in your environment.

### Master It

1. Name the common factors affecting the design of traditional backup and restore solutions.
2. What are the two common storage technologies used for backup and restore? What are two advantages and disadvantages for each technology?
3. Describe how D2D2T (disk-to-disk-to-tape) works.
4. Name the two replication strategies and explain how they differ.
5. Describe the three levels of replication.

**Distinguish new concepts introduced by DPM.** DPM presents a whole new way of thinking about data protection, but it introduces several new concepts to master.

### Master It

1. List which of the following members can be included in the same protection group: a shared volume on a file server, a virtual machine on Virtual Server, a SQL database, a SharePoint farm, and an Exchange storage group.
2. Missy, an Exchange administrator, has two mailbox databases for which she needs to design separate protection policies. To do this, she must put them into separate protection groups. What must she first do in order to permit this configuration?
3. Tom, a SQL Server administrator, has two SQL databases that he needs to protect with DPM. How many protection groups does he need to protect them?
4. You are protecting your department's file server and have it as a member of a protection group defined to synchronize every 30 minutes and create recovery points at 7:00 AM, 3:00 PM, and 11:00 PM. At 3:07 PM, your manager saves changes to an important spreadsheet on the file server. At 3:31, his secretary makes changes to the spreadsheet but the file is corrupted. Up until what time will you be able to recover his saved version before it is overwritten? (Hint: reread the "Why Do I Need Both Synchronization Frequency and Recovery Points?" sidebar.)

**Identify the components in the DPM architecture.** While DPM attempts to mask the complexity of its protection operations, you still need to know the underlying components of your DPM deployment.

### Master It

1. Name the tiers of the DPM application.
2. Does DPM require the use of a separate tape backup solution?

