1

Group Policy Essentials

In this chapter, you'll get your feet wet with the concept that is Group Policy. You'll start to understand conceptually what Group Policy is and how it's created, applied, and modified, and you'll go through some practical examples to get at the basics.

The best news is that the essentials of Group Policy are the same in all versions of Windows 2000 onward. So as I stated in the introduction, if you've got Windows XP, Windows Server 2003, Windows Server 2008, Windows Vista, Windows 7—whatever—you're golden.

That's because Group Policy isn't a server-driven technology. As you'll learn in depth a little later, the magic of Group Policy happens (mostly) on the client (target) machine. And when we say "client," we mean anything that can "receive" Group Policy directives: Windows 7, Windows XP, or even Windows Server 2008 or Windows Server 2003; they're "clients" too.

So, if your Active Directory Domain Controllers are a mixture of Windows 2000 and/or Windows 2003 and/or Windows Server 2008, nothing much changes. And it doesn't matter if your domain is in Mixed, Native, or another mode—Group Policy works exactly the same in all of them.

There are occasional odds and ends you get with upgraded domain types. With the Windows 2003 or later schema, you'll get something neat called WMI filters (described in Chapter 4). Also note that in a Windows 2008 Functional mode domain level, the replication of the file-based part of a Group Policy Object (GPO) can be enhanced to use distributed file system (DFS) replication instead of system volume (SYSVOL) replication.

Regardless of what your server architecture is, I encourage you to work through the examples in this chapter.

So, let's get started and talk about the essentials.

Getting Ready to Use This Book

This book is full of examples. And to help you work through these examples, I'm going to suggest a sample test lab for you to create. It's pretty simple really, but in its simplicity we'll be able to work though dozens of real-world examples to see how things work. Here are

the computers you need to set up and what I suggest you name them (if you want to work through the examples with me in the book):

DC01.corp.com This is your Active Directory Domain Controller. It can be any type of Domain Controller, Windows 2000 and later. For this book, I'll assume you've loaded Windows Server 2008 and later on this computer and that you'll create a test domain called Corp.com.

In real life you would have multiple Domain Controllers in the domain. But here in the test lab, it'll be okay if you just have one.

I'll refer to this machine as DC01 in the book. We'll also use DC01 as a file server and software distribution server and for a lot of other roles we really shouldn't. That's so you can work through lots of examples without bringing up lots of servers.

XPPRO1.corp.com This is some user's Windows XP machine, and it's joined to the domain Corp.com. I'll assume you've loaded Windows XP's SP3. Sometimes it'll be a Sales computer, other times a Marketing computer, and other times a Nursing computer. To use this machine as such, just move the computer account around in Active Directory when the time comes. You'll see what I mean. I'll refer to this machine as XPPRO1 in the book.

Win7.corp.com This is some user's Windows 7 machine and it's joined to the domain Corp.com. I'll refer to this machine as WIN7 in the book. Like XPPRO1, this machine will move around a lot to help us "play pretend" when the times arise. Windows XP works a little differently than Windows 7, so having both a Windows 7 and a Windows XP machine in your environment will be good for testing if you are in charge of making both work under the same roof.

Win7management.corp.com This machine is yours—the IT pro who runs the show. You could manage Active Directory from anywhere on your network, but you're going to do it from here. This is the machine you'll use to run the tools you need to manage both Active Directory and Group Policy. I'll refer to this machine as WIN7MANAGEMENT. As the name implies, you'll run Windows 7 from this machine. Note that you aren't "forced" or "required" to use a Windows 7 machine as your management machine—but you'll be able to "manage it all" if you do.

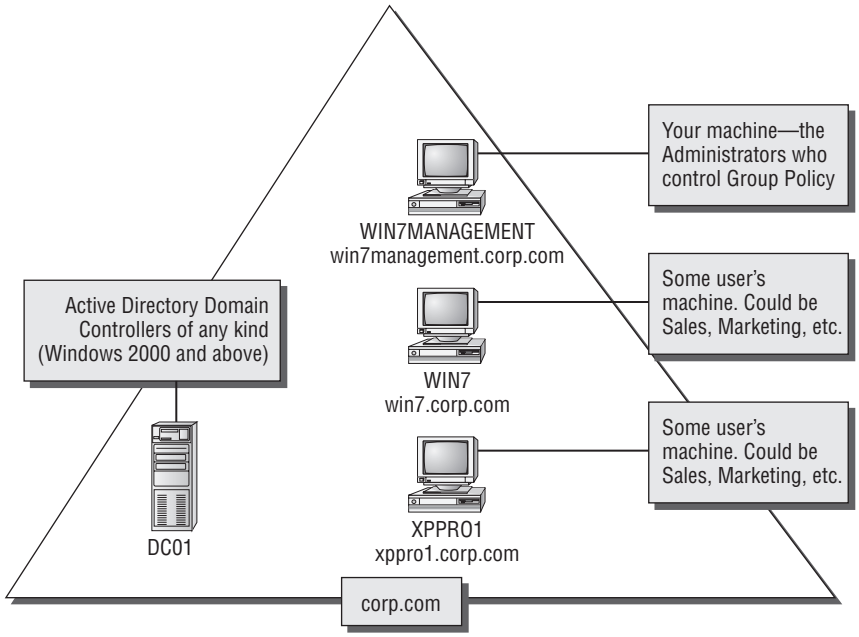
Figure 1.1 shows a diagram of what our test network should look like if you want to follow along.

To save space in the book, we're going to assume you're using a Windows 7 machine as your management machine. If you're forced by some draconian corporate edict to use a Windows Vista or Windows XP (or earlier) machine as a management machine, you'll have to refer to previous editions of the book to get the skinny about using them.

For working through this book, you can build your test lab with real machines or with virtual hardware. Personally, I use VMware Workstation (a pay tool) and VMware Server (a free tool) for my testing. However, Microsoft's tools, like Virtual Server 2005

and Windows Virtual PC, or Sun VirtualBox (all free) are perfectly decent choices as well. Using virtual machines, if you don't have a bunch of extra physical servers and desktops around, you can follow along with all the examples anyway.

FIGURE 1.1 Here's the configuration you'll need for the test lab in this book. Note that the Domain Controller can be 2000 or above, but 2008 is preferred to allow you to work through all the examples in this book.



I suggest you build your test lab from scratch. Get the original media or download each operating system and spin up a new test lab.

You can find Windows Server 2008 trial download versions here:

`www.microsoft.com/windowsserver2008/en/us/trial-software.aspx`

And if you want to get Windows Server 2008 R2, you can find it here:

`http://technet.microsoft.com/en-us/evalcenter/dd459137.aspx`

If you want to get Windows 7 trial versions, here's the URL:

`http://tinyurl.com/ktug5n`

Microsoft usually also makes pre-built virtual hard disk (VHD) images for use with Virtual PC. It's your choice of course, but I prefer to fresh-build my lab instead of using the preconfigured VHD files.

And that's what I'll be doing for my examples in this book. If the URLs I've specified change, I'm sure a little Googling, er, Bing-ing will Bing it, er, bring it right up.

Because Group Policy can be so all-encompassing, I highly recommend that you try the examples in a test lab environment first before making these changes for real in your production environment.

Note that from time to time I might refer to some machine that *isn't* here in the suggested test lab, just to illustrate a point. However, this is the minimum configuration you'll need to get the most out the book.

Getting Started with Group Policy

Group Policy is a big, big place. And you need a road map. Let's try to get a firm understanding of what we're about to be looking at for the next several hundred pages.

Group Policy Entities and Policy Settings

Every Group Policy Object contains two halves: a User half and a Computer half. These two halves are properly called *nodes*, though sometimes they're just referred to as either the *User half* and the *Computer half* or the *User branch* and the *Computer branch.*

A sample Group Policy Object with both the Computer Configuration and User Configuration nodes can be seen in Figure 1.2 (in the upcoming section "Local Group Policy on Pre-Vista Computers"). Don't worry; we'll show you how to get there in just a second.

Just to make things a little more complicated, if you're deploying settings using Active Directory (the most usual case) as opposed to walking up and creating a "local GPO" as we do in Figure 1.2, the interface is a wee bit different and shows the Group Policy Preferences' node. Hang tight for more on that.

The first level under both the User and the Computer nodes contains Software Settings, Windows Settings, and Administrative Templates. If we dive down into the Administrative Templates of the Computer node, underneath we discover additional levels of Windows Components, System, Network, and Printers. Likewise, if we dive down into the Administrative Templates of the User node, we see some of the same folders plus some additional ones, such as Shared Folders, Desktop, and Start Menu and Taskbar.

In both the User and Computer halves, you'll see that policy settings are hierarchical, like a directory structure. Similar policy settings are grouped together for easy location. That's the idea anyway—though, admittedly, sometimes locating the specific policy or configuration you want can prove to be a challenge.

When manipulating policy settings, you can choose to set either computer policy settings or user policy settings (or both!). You'll see examples of this shortly. (See the section "Searching and Commenting Group Policy Objects and Policy Settings" in Chapter 2 for tricks on how to minimize the effort of finding the policy setting you want.)

Most policy settings are not found in both nodes. However, there are a few that overlap. In that case, if the computer policy setting is different from the user policy setting, the computer policy setting generally overrides the user policy setting. But, to be sure, check the Explain text associated with the policy setting.

Wait... I Don't Get It. What Do the User and Computer Nodes Do?

One of the key issues that new Group Policy administrators ask themselves is: "What the heck is the difference between the Computer and User nodes?"

Imagine that you had a combination store: Dog Treats (for dogs) and Candy Treats (for kids). That's right; it's a strange little store with seemingly two types of incompatible foods under the same roof. You wouldn't feed the kids dog treats (they'd spit them out and ignore the treat), and you wouldn't feed the kids' candy to a dog (because the dogs would spit out the sour candy and ignore the treat).

That's the same thing that happens here. Sure, it looks tempting. There are lots of treats on both sides of the store, but only one type of customer will accept each type of treat.

So, in practical terms, the Computer node (the first part of the policy) contains policy settings that are only relevant for computers. That is, if there's a GPO that contains Computer-side settings and it "hits" a computer, these settings will take effect. These Computer-side settings could be items like Startup Scripts, Shutdown Scripts, and how the local firewall should be configured. Think of this as every setting relevant to the *computer itself*—no matter who is logged on at that moment.

The User node (the second part of the policy) contains policy settings that are relevant only for users. Again, if there's a GPO that contains User-side settings and it "hits" a user, these settings will take effect for that user. These User-side items only make sense on a per-user basis, like logon scripts, logoff scripts, availability of the Control Panel, and lots more. Think of this as every setting relevant to the currently logged-on user—and these settings will follow the user to every machine they pop on to.

Feeding users dog treats, er, Computer settings doesn't work. Same thing with feeding computers User-side settings. When a GPO hits user objects with Computer policy settings or computer objects with User policy settings, it simply will *not* do anything. You'll just sit there and scratch your head and wonder why it doesn't work. But it's not that it's not working; this is how it's designed.

Computer settings are for computer objects, and User settings are for user objects. If this is bad news for you, there are two ways to get out of the problem. One way is an in-the-box advanced technique called *loopback processing* that can help you out. Look for more information on loopback processing in Chapter 4. The other way is via a third-party tool called PolicyPak, which (among other things) can permit computers to embrace user-side settings. More on this in Chapter 6.

The 18 (Original) Categories of Group Policy

In this section, you'll learn how to gain access to the interface, which will let you start configuring these categories.

Now, as you're following along working through these examples (or you read Table 1.1 and want to get started right away), you might start to think to yourself, "Jeremy's screen shots don't look exactly like what I have on my screen." There's a simple answer for that.

There are, confusingly, various versions of the Group Policy Editor. If you're still using Windows XP as the place from which you manage Group Policy, your screens are going to look a little different from mine.

More modern clients (starting with Windows Vista/SP1 and onward to Windows 7) use a newer Group Policy Management Console (GPMC), which is found in a download called RSAT. RSAT stands for Remote Server Administration Toolkit and is akin to the old Adminpak download from the Server 2003 era. Inside the RSAT, you'll find tools like Active Directory Users and Computers as well as the GPMC, which we'll use right around the bend. Additionally, only RSAT's GPMC shows subnodes: Policies and Preferences, which will be important for Chapter 5. I'll show you how to get and install the RSAT tool with the updated GPMC in the section "Implementing the GPMC on Your Management Station" coming up a little later in this chapter, so don't worry about it for now.

Note that local policies are *not* split into two subnodes, though—that is because the Preferences node is only available within Active Directory GPOs. More on local policies vs. Active Directory–based policies, right around the corner.

So, if in your real world you're missing the Policies and Preferences nodes within User Configuration and Computer Configuration right now, don't panic. You'll learn how to get those nodes introduced.

In this book, you'll learn about the major categories of Group Policy. Table 1.1 should be helpful if you're looking to get started working right away with a category. Again, your Group Policy Editor won't show the Policies or Preferences nodes unless you're using the RSAT tools with the updated GPMC and you're editing an Active Directory GPO.

TABLE 1.1 The Major Categories of Group Policy

Group Policy Category	Where in Group Policy Interface	Which Operating Systems Support It	Where to Find Information in the Book	Notes
Administrative Templates (also known as Registry Settings)	User or Computer ➢ Policies ➢ Administrative Templates	Windows 2000+	Many examples throughout the book	
Security Settings	Computer or User Configuration ➢ Policies ➢ Windows Settings ➢ Security Settings	Windows 2000+	Chapter 8	
Wired Network (802.3) Settings	Computer Configuration ➢ Policies ➢ Windows Settings ➢ Security Settings ➢ Wired Network (IEEE 802.3) Policies	Windows Vista+ only	Chapter 8	Be sure to read Chapter 8 before attempting to use these settings.
Wireless Network (802.11) Settings	Computer Configuration ➢ Policies ➢ Windows Settings ➢ Security Settings ➢ Wireless Network (IEEE 802.11) Policies	Windows XP and Windows Vista+ (set independently)	Chapter 8	Be sure to read Chapter 8 before attempting to use these settings for Windows Vista+.
Scripts	Computer Configuration ➢ Policies ➢ Windows Settings ➢ Scripts (Startup/Shutdown) and Policies ➢ Windows Settings ➢ Script (Logon/Logoff)		Chapter 12	
Group Policy Software Installation (also known as Application Management)	Computer or User Configuration ➢ Policies ➢ Software Settings	Windows 2000+	Chapter 11	

TABLE 1.1 The Major Categories of Group Policy *(continued)*

Group Policy Category	Where in Group Policy Interface	Which Operating Systems Support It	Where to Find Information in the Book	Notes
Folder Redirection	User Configuration ➢ Policies ➢ Windows Settings ➢ Folder Redirection	Windows 2000+; some additional options for Windows XP; many additional options for Windows Vista+	Chapter 11	
Disk Quotas	Computer Configuration ➢ Policies ➢ Administrative Templates ➢ System ➢ Disk Quotas	Windows 2000+	I don't cover this subject in this book. This content has been removed to make room for other material. Disk quotas have been covered in previous editions.	There is a brief article on disk quotas here: `http://support.microsoft.com/kb/183322`. You'll find another article here: `http://tinyurl.com/35mvny`.
Encrypted Data Recovery Agents (EFS Recovery Policy)	Computer Configuration ➢ Policies ➢ Windows Settings ➢ Security Settings ➢ Public Key Policies ➢ Encrypting File System	Windows 2000+	Not covered	
Internet Explorer Maintenance	User Configuration ➢ Policies ➢ Windows Settings ➢ Internet Explorer Maintenance	Windows 2000+	Chapter 12	
Software Restriction Policies	Computer or User ➢ Policies ➢ Windows Settings ➢ Security Settings ➢ Software Restriction Policies	Windows XP+	Chapter 8	

TABLE 1.1 The Major Categories of Group Policy *(continued)*

Group Policy Category	Where in Group Policy Interface	Which Operating Systems Support It	Where to Find Information in the Book	Notes
Quality of Service (QoS) Packet Scheduler and Policy-Based QoS	Computer or User Configuration ➢ Policies ➢ Windows Settings ➢ Policy-based QoS	Windows XP+; Policy-based Enterprise QoS is Vista+ only.	Not covered	You can start your Windows Vista+ QoS journey here: `http://tinyurl.com/yxglpp`.
IPSec (IP Security) Policies	In XP: Computer Configuration ➢ Policies ➢ Windows Settings ➢ Security Settings ➢ IP Security Policies	Windows 2000+	Chapter 8	In Vista+, this is part of the Windows Firewall with Advanced Security section located under Computer Configuration ➢ Policies ➢ Windows Settings ➢ Security Settings.
Windows Search	Computer Configuration ➢ Policies ➢ Administrative Templates ➢ Windows Components ➢ Search	Windows Vista+	Not covered	See `www.microsoft.com/windows/desktopsearch`.
Deployed Printer Connections	Computer or User Configuration ➢ Policies ➢ Windows Settings ➢ Deployed Printers	Technically, Vista+ only; workaround available for Windows 2000+	Not covered	We'll leverage a better way to zap printers around. We'll learn about it first in Chapter 5, then deeply explore it in Chapter 12. If you want to learn more about the older "Deployed Printer Connections" see GPanswers.com's Newsletter #17.

TABLE 1.1 The Major Categories of Group Policy *(continued)*

Group Policy Category	Where in Group Policy Interface	Which Operating Systems Support It	Where to Find Information in the Book	Notes
Offline Files	Computer or User Configuration ➢ Policies ➢ Administrative Templates ➢ Network ➢ Offline Files	Different Group Policy "moving parts" to make this technology work in Vista+ and Windows Server 2008 than in previous operating systems; feature available in Windows 2000 and later.	Chapter 10	
Group Policy Preference Extensions	Computer or User Configuration ➢ Preferences (not available in local policies, only domain policies)	Group Policy Preference Extensions built in to Windows Server 2008 and Windows 7, but are an additional download and installation for Windows XP and Windows Vista; not supported on Windows 2000 machines.	Chapter 5	Adds 21 additional functions to the Group Policy universe.
Internet Explorer User Accelerators		Windows 7+	Not covered	
Internet Explorer Machine Accelerators		Windows 7+	Not covered	

Group Policy is a twofold idea. First, without an Active Directory, there's one and only one Group Policy available.

Officially, this policy directly on the workstation is called a *local policy*, but it still resides under the umbrella of the concept of Group Policy. Later, once Active Directory is available, the nonlocal (or, as they're sometimes called, *domain-based* or *Active Directory–based*) Group Policy Objects come into play, as you'll see later. Let's get started and explore both options.

While you're plunking around inside the Group Policy Editor (also known as the Group Policy Management Editor, or Group Policy Object Editor), you'll see lots of policy settings that are geared toward a particular operating system. Some are only for specific operating systems, and others are more general. If you happen to apply a policy setting to a system that isn't listed, the policy setting is simply ignored. For instance, policy settings described as working "Only for Windows 7" machines will not typically work on Windows XP machines. Each policy setting has a "Supported on" field that should be consulted to know which operating systems can embrace which policy setting. Many of them will say something like "At least Windows XP" to let you know they're valid for, say, XP and onward.

Understanding Local Group Policy

Before we officially dive into what is specifically contained inside this magic of Group Policy or how Group Policy is applied when Active Directory is involved, you might be curious to see exactly what your interaction with Local Group Policy might look like.

Local Group Policy is best used when Active Directory isn't available, say either in a Novell NetWare environment or when you have a gaggle of machines that simply aren't connected to a domain.

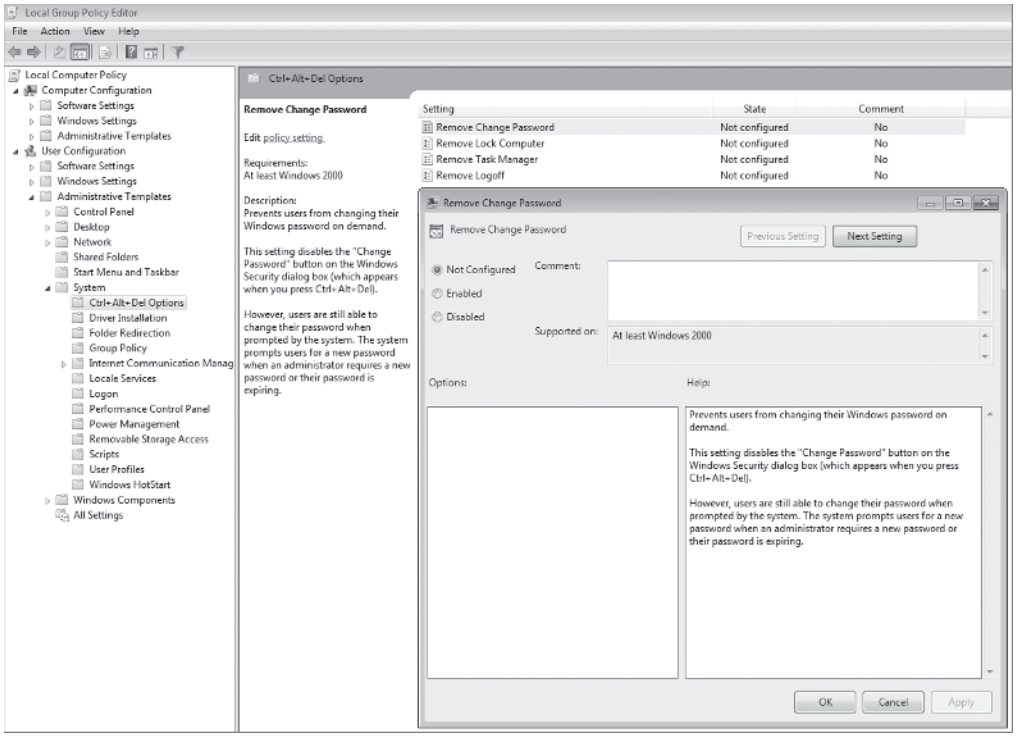
Local Group Policy is different for Windows Vista and later versus the other Windows operating systems. Let's explore Local Group Policy on pre-Vista machines first and then move on to the features specific to Vista and Windows 7.

Local Group Policy on Pre-Vista Computers

The most expeditious way to edit the Local Group Policy on a machine is to click Start ➢ Run and type in **`GPEDIT.MSC`**. This pops up the Local Computer Policy Editor.

You are now exploring the Local Group Policy of this Windows XP workstation. Local Group Policy is unique to each specific machine. To see how a Local Group Policy applies, drill down through the User Configuration ➢ Administrative Templates ➢ System ➢ Ctrl+Alt+Del Options and select **Remove Lock Computer**, as shown in Figure 1.2. Once it's selected, choose the Enabled radio button and click OK.

FIGURE 1.2 You can edit the Local Group Policy using the Local Group Policy Editor (GPEDIT.MSC).



When you do, within a few seconds you should see that if you press Ctrl+Alt+Del, the Lock Computer option is unavailable.

To revert the change, simply reselect **Remove Lock Computer** and select Not Configured. This reverts the change back to the way the operating system works by default.

You can think of Local Group Policy as a way to perform decentralized administration. A bit later, when we explore Group Policy with Active Directory, we'll saunter into centralized administration.

This Local Group Policy affects everyone who logs onto this machine—including normal users and administrators. Be careful when making settings here; you can temporarily lock yourself out of some useful functions. For instance, frequently administrators want to remove Run from the Start menu for Windows XP machines. Then, the first time they themselves want to go to a command prompt, they can't choose Start ➢ Run. It's just gone!

To get that Run command back, you'll have to click the `MMC.exe` icon in Explorer (or via command line/batch file) and manually load the Group Policy snap-in.

As I stated in the introduction, most of the settings we'll explore in this book are available to workstations or servers that aren't joined to an Active Directory domain. However, many functions, like Folder Redirection settings (discussed in Chapter 10), the Software Distribution settings (discussed in Chapter 11), and others are not available to stand-alone machines without Active Directory present.

You can point toward other computers' local policies by using the syntax `gpedit.msc /gpcomputer:"targetmachine"` or `gpedit.msc /gpcomputer:"targetmachine.domain.com"`; the machine name must be in quotes.

Local Group Policy on Vista and Later

It's true that you can also type **`GPEDIT.MSC`** at the Windows Vista or Windows 7 command prompt and get the same Local Computer Policy Editor you just saw in Windows XP.

However, Vista and later has a secret super-power that takes Local Group Policy to the next level.

Remember how, not more than three paragraphs ago, I stated this:

> "This Local Group Policy affects everyone who logs onto this machine—including normal users and administrators. Be careful when making settings here; you can temporarily lock yourself out of some useful functions."

True—for pre-Vista machines. On Vista and later, however, the superpower feature is that you can decide who gets what settings at a local level. This feature is called Multiple Local GPOs (MLGPOs).

MLGPOs are most often handy when you want your users to get one gaggle of settings (that is, desktop restrictions) but you want to ensure that your access is unfettered for day-to-day administration.

Now, in these examples we're going to use Windows 7, but this same feature is available on Vista and later (including Windows Server 2008 and of course Server 2008 R2). It's just not all that likely you'll end up using it on a Windows Server.

Understanding Multiple Local GPOs

The best way to understand MLGPOs is by thinking of the end product. That is, when we're done, we want our users to embrace some settings, and we (administrators) want to potentially embrace some settings. Or perhaps you want just one specific user to embrace a particular combination of settings.

When you type **GPEDIT.MSC** at a command prompt, it's just as if you did it on Windows XP: you're affecting all users—mere mortals *and* administrators.

But with Vista and later, there are actually three "layers" that can be leveraged to ensure that some settings affect regular users and other settings affect you (the administrator).

Let's be sure to understand all three layers before we get too gung-ho and try it out. When MLGPOs are processed, Windows Vista and later checks to see if the layer is being used and if that layer is supposed to apply to that user:

Layer 1 (lowest priority) The Local Computer Policy. You create this by running `GPEDIT.MSC`.

- The settings you make on the Computer Configuration side are guaranteed to affect all users on this computer (including administrators).
- The settings you make on the User Configuration side may be trumped by Layer 2 or Layer 3.

Layer 2 (next highest priority) Is the user a mere mortal *or* a local administrator? (One account cannot be both.) This layer cannot contain Computer Configuration settings.

Layer 3 (most specific) Is this a specific user who is being dictated a specific policy? This layer cannot contain Computer Configuration settings.

You can see this graphically laid out in Figure 1.3.

If no conflicts exist among the levels, the effect is additive. For instance, let's imagine the following:

- Layer 1 (the Local Computer Policy level): The wish is to **Remove Lock Computer** from the Ctrl+Alt+Del area.
- Then, at Layer 2: We say "All local users" will have "Search" gone from the Start menu.
- Then, at Layer 3: We say Fred, a local user, will be denied access to the Control Panel.

The result for Fred will be the sum total of all edicts at all layers.

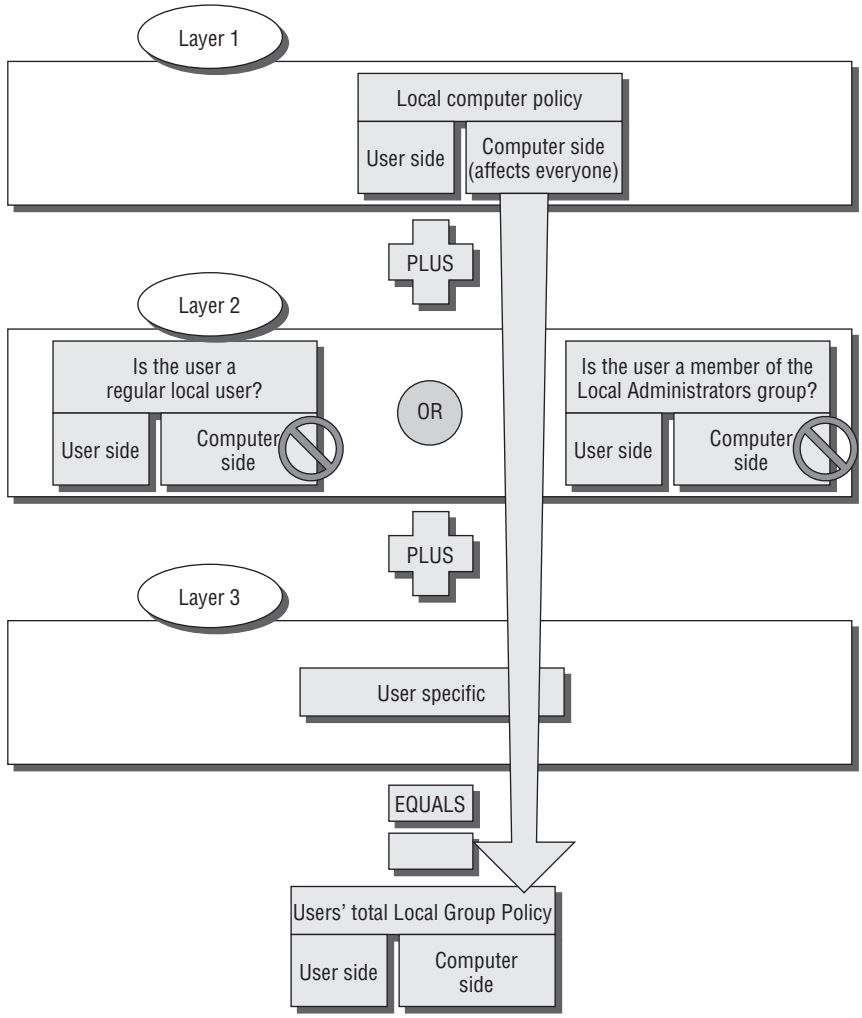
But what if there's a conflict between the levels? In that case, the layer that's "closest to the user" wins (also known as "Last writer wins"). So, if at the Local Computer Policy the wish is to **Remove Lock Computer** from the Ctrl+Alt+Del area, but that area is expressly granted to Sally, a local user on that machine, Sally will still be able to use the Lock command. That's because we're saying that she is expressly granted the right at Layer 3, which "wins" over Layers 1 and 2.

Trying Out Multiple Local GPOs on Windows Vista and Later

Just typing **GPEDIT.MSC** at the Windows Vista and later Start Search prompt doesn't give you the magical "layering" superpower. Indeed, just typing **GPEDIT.MSC** performs the exact same function as it did in Windows XP. That is, every edit you make while you run the Local Computer Policy affects all users logged onto the machine.

To tell Vista and later you want to edit one of the layers (as just described), you need to load the Group Policy Object Editor by hand. We'll do this on WIN7.

FIGURE 1.3 A block diagram of how MLGPOs are applied to a system

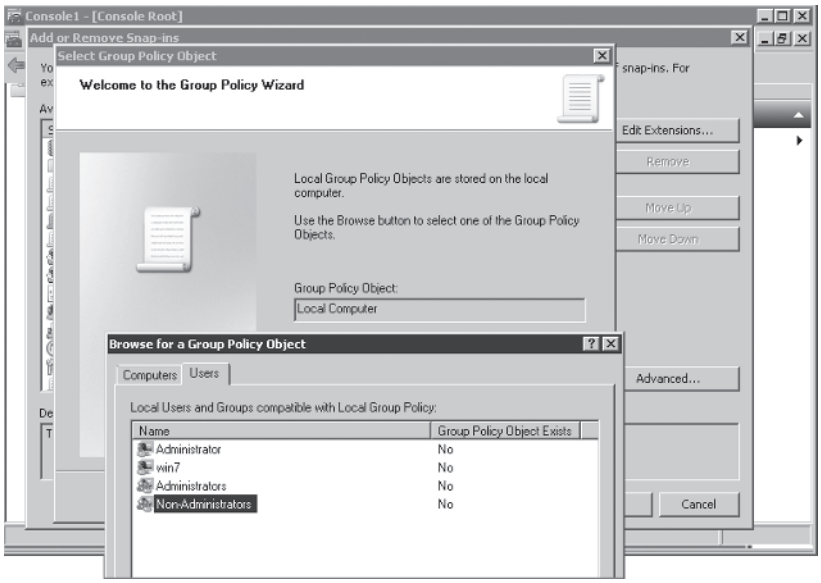


On WIN7, to load the Group Policy Object Editor by hand, follow these steps:

1. Click Start, and then in the Start Search box (which will run things), type **MMC**. A "naked" MMC appears. Note that you will have to approve the User Access Control (UAC) message (UAC is discussed in detail in Chapter 8).
2. From the File menu, choose Add/Remove Snap-in to open the Add/Remove Snap-in dialog box.
3. Locate and select the Group Policy Object Editor Snap-in and click Add (don't choose the Group Policy Management Snap-in, if present—that's the GPMC that we'll use a bit later).

4. At the "Select Group Policy Object" screen, note that the default Local Computer Policy is selected. Click Browse.
5. The "Browse for a Group Policy Object" dialog box appears. Select the Users tab and select the layer you want. That is, you can pick Non-Administrators or Administrators, or click a specific user, or the Administrator account as seen in Figure 1.4.

FIGURE 1.4 Edit specific layers of Windows MLGPOs by first adding the Group Policy Object Editor into a "naked" MMC. Then browse for the Windows Local Group Policy by firing up `GPEDIT.MSC`.



In the Group Policy Object "Exists" column in the Users tab, you can also tell whether or not a local GPO layer is being used.

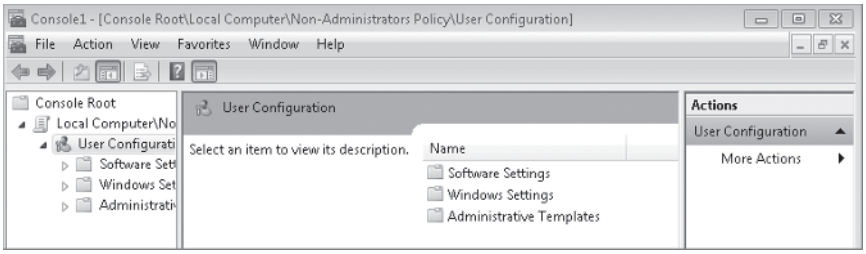
6. At the "Select a Group Policy Object" dialog box, click Finish.
7. At the "Add or Remove Snap-ins" dialog box, click OK.

   You should now be able to edit that layer of the local GPO. For instance, Figure 1.5 shows that I've chosen to edit the Non-Administrators portion of the GPO (which is on level 2).

To edit additional or other layers of the local GPO, repeat the previous steps.

Here's an important point that bears repeating: Layers 2 and 3 of the MLGPO *cannot* contain overriding computer settings from Layer 1. That's why in Figure 1.5 you simply don't see them—they're not there. If you want to introduce a computer-side setting that affects everyone on the machine, just fire up `GPEDIT.MSC` and you'll be off and running. That's Layer 1, and it affects everyone.

FIGURE 1.5 Below the words Console Root, you can see which layer of the local GPO you're specifically editing.



Local GPOs Final Thoughts

You can think of Local Group Policy as a way to perform desktop management in a decentralized way. That is, you're still running around, more or less, from machine to machine where you want to set the Local Group Policy.

The other strategy is a centralized approach. Centralized Group Policy administration works only in conjunction with Active Directory and is the main focus of this book.

For more information, check out the article "Step-by-Step Guide to Managing Multiple Local Group Policy" from Microsoft. At last check, the URL was `http://tinyurl.com/e4e9k`. The specific guide is "Step-by-Step Guide to Managing Multiple Local Group Policy.doc" and is found toward the bottom. The guide is Vista specific, not Windows 7 specific, but all the steps should be the same.

In case you're curious, Local Group Policy is stored in the `%windir%\system32\grouppolicy` directory (usually, `C:\windows\system32\grouppolicy`). The structure found here mirrors what you'll see later in Chapter 7 when we inspect the ins and outs of how Group Policy applies from Active Directory. Windows Vista and later store Level 2 (Admin/Non-Admin Local Policies) and specific Local User Policies (level 3) inside `%windir%\system32\grouppolicyusers`.

You will notice one folder–per-user policy you have created, each named with the Security ID (SID) of the relevant user object.

Active Directory–Based Group Policy

To use Group Policy in a meaningful way, you need an Active Directory environment. An Active Directory environment needn't be anything particularly fancy; indeed, it could consist of a single Domain Controller (Windows 2000 and later) and perhaps just one Windows XP or Windows 7 workstation joined to the domain.

But Active Directory can also grow extensively from that original solitary server. You can think of an Active Directory network as having four constituent and distinct levels that relate to Group Policy:

- The local computer
- The site
- The domain
- The organizational unit (OU)

The rules of Active Directory state that every server and workstation must be a member of one (and only one) domain and be located in one (and only one) site.

In Windows NT, additional domains were often created to partition administrative responsibility or to rein in needless chatter between Domain Controllers. With Active Directory, administrative responsibility can be delegated using OUs.

Additionally, the problem with needless domain bandwidth chatter has been brought under control with the addition of Active Directory sites, which are concentrations of IP (Internet Protocol) subnets with fast connectivity. There is no longer any need to correlate domains with network bandwidth—that's what sites are for!

Group Policy and Active Directory

When Group Policy is created at the local level, everyone who uses that machine is affected by those wishes. But once you step up and use Active Directory, you can have nearly limitless Group Policy Objects (GPOs)—with the ability to selectively decide which users and which computers will get which wishes (try saying that five times quickly). The GPO is the vessel that stores these wishes for delivery.

Actually, you can have only 999 GPOs applied and affecting a user or a computer before the system "gives up" and won't apply any more.

When we create a GPO that can be used in Active Directory, two things happen: we create some brand-new entries within Active Directory, and we automatically create some brand-new files within our Domain Controllers. Collectively, these items make one GPO.

You can think of Active Directory as having three major levels:

- Site
- Domain
- OU

Additionally, since OUs can be nested within each other, Active Directory has a nearly limitless capacity for where we can tuck stuff away.

In fact, it's best to think of this design as a three-tier hierarchy: site, domain, and each nested OU. When wishes, er, policy settings, are set at a higher level in Active Directory, they automatically flow down throughout the remaining levels.

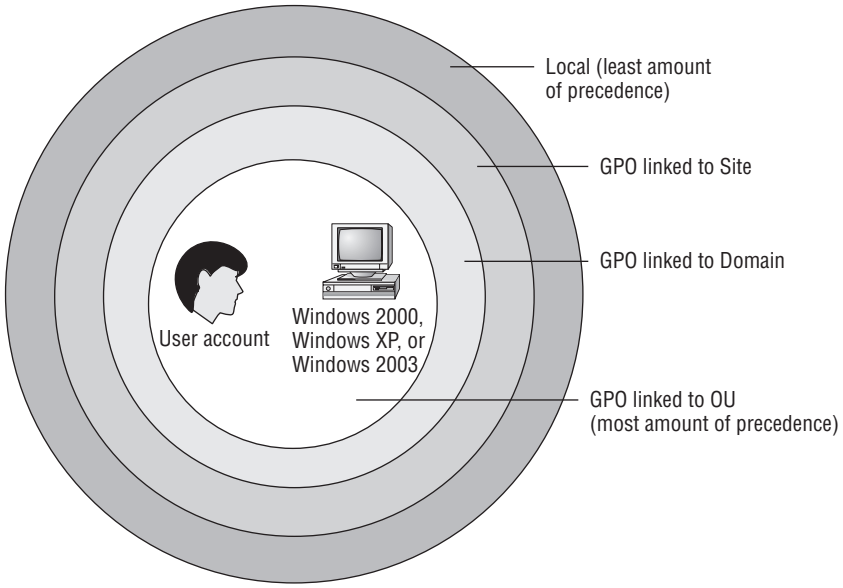
So, to be precise:

- If a GPO is set at the site level, the policy settings contained within affect those accounts within the geography of the site. Sure, their user account could be in one domain and their computer account could be in another domain. And of course, it's likely that those accounts are in an OU. But the account is affected only by the policy settings here because the account is in a specific site. And logically, when a computer starts up in a new site, the User object will also get its site-based Group Policy from the same place. This is based on the IP subnet the user is a part of and is configured using Active Directory Sites and Services.
- If a GPO is set at the domain level, it affects those users and computers within the domain and all OUs and all other OUs beneath it.
- If a GPO is set at the OU level, it affects those users or computers within the OU and all other OUs beneath it (usually just called child or sub-OUs).

By default, when a policy is set at one level, the levels below *inherit* the settings from the levels above it. You can have "cumulative" wishes that keep piling on.

You might wonder what happens if two policy settings conflict. Perhaps one policy is set at the domain level and another policy is set at the OU level, which reverses the edict in the domain. The result is simple: policy settings further down the food chain take precedence. For instance, if a policy setting conflicts at the domain and OU levels, the OU level "wins." Likewise, domain-level settings override any policy settings that conflict with previously set site-specific policy settings. This might seem counterintuitive at first, so bear with me for a minute.

Take a look at the following illustration to get a view of the order of precedence.



The golden rule with Group Policy is truly "Last writer wins." Another way to say it is "If any GPOs conflict, the settings contained in the last written GPO win."

However, don't forget about any Local Group Policy that might have been set on a specific workstation. Recall that for pre-Vista machines, everyone logging onto that workstation is affected by that policy setting. You just learned how Windows Vista's and Windows 7's MLGPOs add up to three layers of user settings (where Windows XP's Local GPOs have just one).

Regardless, once that local policy is determined, only *then* do policy settings within Active Directory (the site, domain, and OU) apply. So, sometimes people refer to the *four* levels of Group Policy: local workstation, site, domain, and OU. Nonetheless, GPOs set within Active Directory always trump the Local Group Policy should there be any conflict.

If this behavior is undesirable for lower Active Directory levels, all the settings from higher Active Directory levels can be blocked with the Block Inheritance attribute on a given OU. Additionally, if a higher-level administrator wants to guarantee that a setting is inherited down the food chain, they can apply the Enforced attribute via the GPMC interface. (Panic not! Chapter 2 explores both Block Inheritance and Enforced attributes in detail.)

Note that you cannot "Block Inheritance" between Local GPOs and Active Directory GPOs. But it is true that anything you set within Active Directory to inverse a Local GPO setting is always honored. Said another way, Active Directory edicts trump local edicts.

Don't sweat it if your head is spinning a little now from the Group Policy application theory. I'll go through specific hands-on examples to illustrate each of these behaviors so that you understand exactly how this works.

Linking Group Policy Objects

Another technical concept that needs a bit of description here is the "linking" of GPOs. When a GPO "appears" to be "created" at the site, domain, or OU level via the GUI (which we'll do in a moment), what's really happening is quite different. It's created in one, set "place," then merely "linked" there. (Yes, I know there are a lot of "quotes" in the last sentence, but sometimes that's how I "write.")

Anyway, when you tell the system, "I want to affect an OU with this new GPO," the system automatically creates the GPO in the fixed location, and then associates that GPO with the level at which you want to affect. That association is called *linking*.

Linking is an important concept for several reasons. First, it's generally a good idea to understand what's going on under the hood. However, more practically, the Group Policy Management Console (GPMC), as we'll explore in just a bit, displays GPOs from their linked perspective.

Let's extend the metaphor a little more.

You can think of all the GPOs you create in Active Directory as children in a big swimming pool. Each child has a tether attached around their waist, and an adult guardian is holding the other end of the rope. Indeed, there could be multiple tethers around a child's waist, with multiple adults tethered to one child. A sad state indeed would be a child who has no tether but is just swimming around in the pool unsecured. The swimming pool in this analogy is a specific Active Directory container named Policies (which we'll examine closely in Chapter 7). All GPOs are born and "live" in that specific domain. Indeed, they're replicated to all Domain Controllers. The adult guardian in this analogy represents a *level* in Active Directory—any site, domain, or OU.

In our swimming pool example, multiple adults can be tethered to a specific child. With Active Directory, multiple levels can be linked to a specific GPO. Thus, any level in Active Directory can leverage multiple GPOs, which are standing by in the domain ready to be used.

Remember, though, unless a GPO is specifically linked to a site, a domain, or an OU, it does not take effect. It's just floating around in the swimming pool of the domain waiting for someone to make use of it.

I'll keep reiterating and refining the concept of linking throughout these first four chapters. And, as necessary, I'll discuss why you might want to "unlink" a policy.

This concept of linking to GPOs created in Active Directory can be a bit confusing. It will become clearer later as we explore the processes of creating new GPOs and linking to existing ones. Stay tuned. It's right around the corner.

An Example of Group Policy Application

At this point, it's best not to jump directly into adding, deleting, or modifying our own GPOs. Right now, it's better to understand how Group Policy works "on paper." This is especially true if you're new to the concept of Group Policy, but perhaps also if Group Policy has been deployed by other administrators in your Active Directory.

By walking through a fictitious organization that has deployed GPOs at multiple levels, you'll be able to better understand how and why policy settings are applied by the deployment of GPOs.

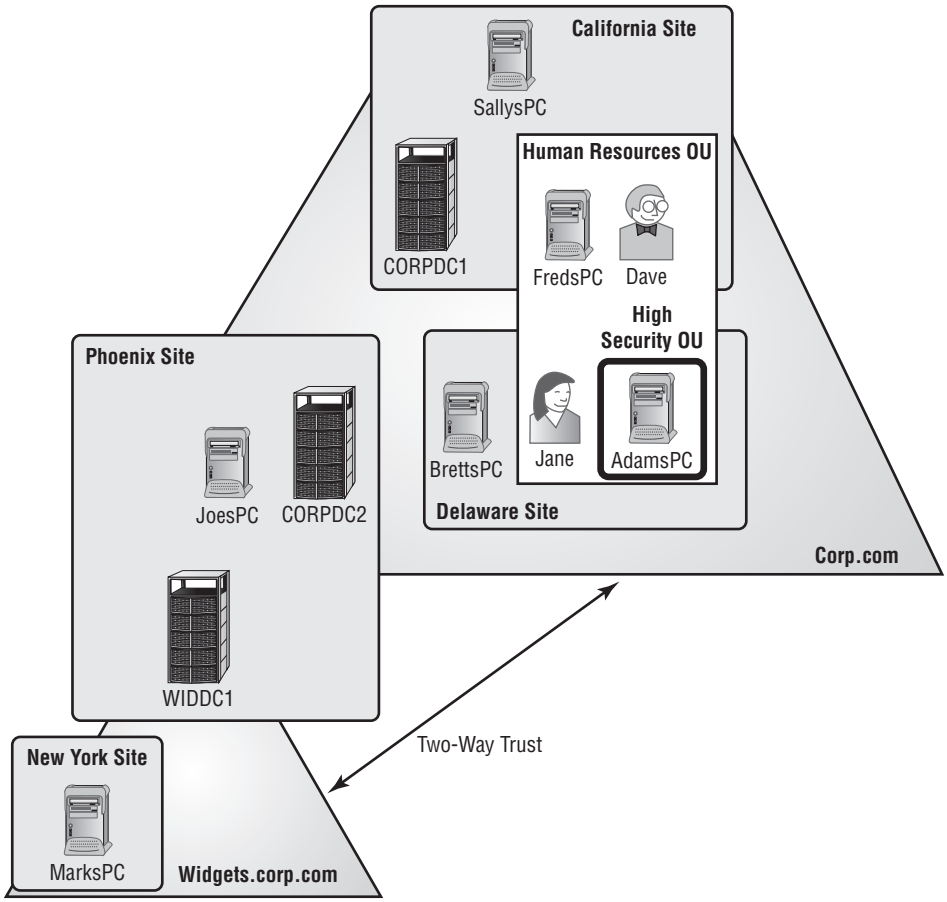
Let's start by taking a look at Figure 1.6, the organization for our fictitious example company, Example.com.

This picture could easily tell 1,000 words. For the sake of brevity, I've kept it down to around 200. There are two domains: Corp.com and Widgets.example.com. Let's talk about Corp.com:

- The domain Example.com has two Domain Controllers. One DC, named CORPDC1, is physically located in the California site. Example.com's other Domain Controller, CORPDC2, is physically located in the Phoenix site.
- As for PCs, they need to physically reside somewhere. SallysPC is in the California site; BrettsPC and AdamsPC are in the Delaware site. JoesPC is in the Phoenix site.

- User accounts may or may not be in OUs. Dave's and Jane's account is in the **Human Resources** OU.
- Computer accounts may or may not be in OUs. FredsPC is in the **Human Resources** OU. AdamsPC is specifically placed within the **High Security** OU. JoesPC, SallysPC, and BrettsPC are hanging out in a container, and aren't in any OUs.

FIGURE 1.6 This fictitious Example.com is relatively simple. Your environment may be more complex.



Using Active Directory Sites and Services, you can put in place a schedule to regulate communication between EXAMPLEDC1 located in California and EXAMPLEDC2 located in Phoenix. That way, the administrator controls the chatter between the two Example.com Domain Controllers, and it is not at the whim of the operating system.

Another domain, called Widgets.example.com, has an automatic transitive two-way trust to Example.com. There is only one Domain Controller in the Widgets.example.com domain, named WIDDC1, and it physically resides at the Phoenix site. Last, there is MarksPC, a

member of the Widgets.example.com domain, and it physically resides in the New York site and isn't in any OU.

Understanding where your users and machines are is half the battle. The other half is understanding which policy settings are expected to appear when they start logging onto Active Directory.

Examining the Resultant Set of Policy

As stated earlier, the effect of Group Policy is cumulative as GPOs are successively applied—starting at the local computer, then the site, the domain, and each nested OU. The end result of what affects a specific user or computer—after all Group Policy at all levels has been applied—is called the *Resultant Set of Policy (*RSoP). This is sometimes referred to as the *RSoP calculation*.

Throughout your lifetime working with Group Policy, you will be asked to troubleshoot the RSoP of client machines.

Many of our dealings with Group Policy will consist of trying to understand and troubleshoot the RSoP of a particular configuration. Getting a good, early understanding of how to perform manual RSoP calculations on paper is important because it's a useful troubleshooting skill. In Chapters 3 and 7, we'll also explore additional RSoP skills—with tools and additional manual troubleshooting.

Before we jump in to try to discover what the RSoP might be for any specific machine, it's often helpful to break out each of the strata—local computer, site, domain, and OU—and examine, at each level, what happens to the entities contained in them. I'll then bring it all together to see how a specific computer or user reacts to the accumulation of GPOs. For these examples, assume that no local policy is set on any of the computers; the goal is to get a better feeling of how Group Policy flows, not necessarily what the specific end state will be.

At the Site Level

Based on what we know from Figure 1.6, the GPOs in effect at the site level are shown here:

Site	Computers Affected
California	SallysPC, EXAMPLEDC1, and FredsPC
Phoenix	EXAMPLEDC2, JoesPC, and WIDDC1
New York	MarksPC
Delaware	AdamsPC and BrettsPC

If we look at the graphic again, it looks like Dave, for instance, resides in California and Jane, for instance, resides in Delaware. But I don't like to think about it like that. I prefer to say that their accounts reside in OUs.

But users are affected by site GPOs *only* when they log onto computers that are at a specific site.

In Figure 1.6, we have Dave's and Jane's accounts in the Human Resources OU. And that's great. But they're only affected by California site-level GPOs if they travel to California. It doesn't matter where they usually reside; again, they're only affected by the site-level GPOs when they're physically present in that site.

So, don't think that user accounts *reside* at the site level. Rather, they reside in the OU level but are using computers in the site and, hence, get the properties assigned to all users at that site.

Sites are defined using the Active Directory Sites and Services tool. IP subnets that constitute a site are assigned using this tool. That way, if a new computer turns on in Delaware, Active Directory knows what site the computer is in.

At the Domain Level

Here's what we have working at the domain level:

Domain	Computers/Users Affected
Example.com Computers	SallysPC, FredsPC, AdamsPC, BrettsPC, JoesPC, EXAMPLEDC1, and EXAMPLEDC2
Example.com Users	Dave and Jane
Widgets.example.com Computers	WIDDC1 and MarksPC

At the OU Level

At the organizational unit level, we have the following:

Organizational Unit	Computers/Users Affected
Human Resources OU Computers	FredsPC is in the **Human Resources** OU; therefore, it is affected when the **Human Resources** OU gets GPOs applied. Additionally, the **High Security** OU is contained inside the **Human Resources** OU. Therefore, AdamsPC, which is in the **High Security** OU, is also affected whenever the **Human Resources** OU is affected.
Human Resources OU Users	The accounts of Dave and Jane are affected when the **Human Resources** OU has GPOs applied.

Bringing It All Together

Now that you've broken out all the levels and seen what is being applied to them, you can start to calculate what the devil is happening on any specific user and computer combination. Looking at Figure 1.6 and analyzing what's happening at each level makes adding things together between the local, site, domain, and organizational unit GPOs a lot easier.

Here are some examples of RSoP for specific users and computers in our fictitious environment:

FredsPC	FredsPC inherits the settings of the GPOs from the California site, then the Example.com domain, and last, the **Human Resources** OU.
MarksPC	MarksPC first accepts the GPOs from the New York site and then the Widgets.example.com domain. MarksPC is not in any OU; therefore, no organizational unit GPOs apply to his computer.
AdamsPC	AdamsPC is subject to the GPOs at the Delaware site, the Example.com domain, the **Human Resources** OU, and the **High Security** OU.
Dave using AdamsPC	AdamsPC is subject to the computer policies in the GPOs for the Delaware site, the Example.com domain, the **Human Resources** OU, and finally the **High Security** OU. When Dave travels from California to Delaware to use Adam's workstation, his user GPOs are dictated from the Delaware site, the Example.com domain, and the **Human Resources** OU.

At no time are any domain GPOs from the Example.com parent domain automatically inherited by the Widget.example.com child domain. Inheritance for GPOs only flows downward to OUs within a single domain—not between any two domains—parent to child or otherwise, unless you explicitly link one of those parent GPOs to a child Domain Container.

If you want one GPO to affect the users in more than one domain, you have four choices:

- Precisely re-create the GPOs in each domain with their own GPO.
- Copy the GPO from one domain to another domain (using the GPMC, as explained in the appendix).
- Use a third-party tool that can perform some magic and automatically perform the copying between domains for you.
- Do a generally recognized no-no called *cross-domain policy linking*. (I'll describe this no-no in detail in Chapter 7 in the section "Group Policy Objects from a Domain Perspective.")

Also, don't assume that linking a GPO at a site level necessarily guarantees the results to just one domain. In this example, as in real life, there is not necessarily a 1:1 correlation between sites and domains. Indeed, without getting too geeky here, sites technically belong to the forest, and not any particular domain.

At this point, we'll put our example Example.com behind us. That was really an on-paper exercise to allow you to get a feel for what's possible in Group Policy land. From this point forward, you'll be doing most items in your test lab and following along.

Group Policy, Active Directory, and the GPMC

The interface used to create, modify, and manipulate Group Policy in the original iteration of Active Directory when Windows 2000 was a pup had led to numerous missteps and head scratching when people try to figure out why something isn't going the way it should.

To make optimal use of Group Policy in an Active Directory environment, the Group Policy team at Microsoft introduced a free, downloadable tool for managing Group Policy in Active Directory in a meaningful way. It's called the Group Policy Management Console (GPMC), as mentioned earlier. The GPMC wasn't part of Windows 2000, or even Windows 2003. And it was "downloadable" for Windows XP operating systems.

It is, however, part of the shipping version of Windows Vista and Windows Server 2008, so no extra effort is required.

But wait!

Turns out, it's "gone again" with Windows Vista when SP1 is installed. Yep—the GPMC is "automatically removed" from Vista when its SP1 is applied. And to *reinstall* it, you have to fetch what's known as the *Remote Server Administration Tools (RSAT)*. Inside RSAT is the "updated GPMC."

And, it's not "in the box" for Windows 7 either.

I know, I know. This is all weird, right? I promise a detailed explanation of why in Chapter 2.

Kickin' It Old School

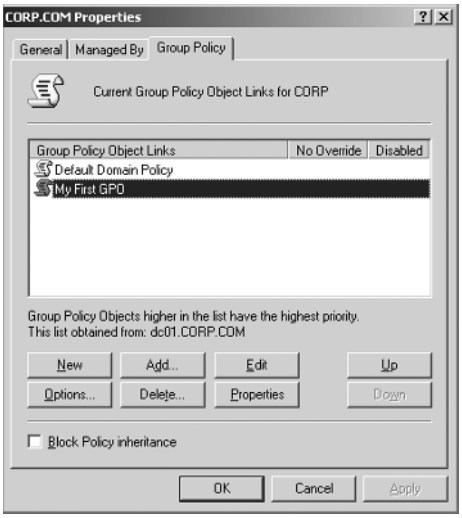
For the love of Pete, please don't tell me you're still using Windows 2000 or Windows 2003 "out of the box" to control your Active Directory.

Because if you are, that means you've never installed the GPMC; the tools built into Windows 2000 and Windows 2003 domains use the old-style interface. This old interface is built into Active Directory Users and Computer and Active Directory Sites and Services.

Hopefully, everyone is past this by now, but if you're not (or, you want a little nostalgia), here's a quick tour of the old-school interface. If you've never seen the old-style interface, you can do so right now before we leave it in the dust for the new GPMC in the next section.

If you don't yet have the GPMC installed on Windows Server 2003, and wish to see the old-style interface and create your first GPO at the domain level, follow these steps:

1. Log onto a Windows 2000 Server or Windows Server 2003 Domain Controller as the Administrator account (which is a Domain Administrator).
2. Choose Start ➢ All Programs ➢ Administrative Tools and select Active Directory Users and Computers. Alternatively, select Start ➢ Run and select `dsa.msc` to open Active Directory Users and Computers.
3. Right-click the domain name and choose Properties from the context menu.
4. Click the Group Policy tab.
5. Click the New button to spawn the creation of your first GPO.
6. For this first example, type **My First GPO**, as shown here.



Highlight the policy and click Edit to open the Group Policy Management Editor.

At this point, things should look familiar, just like the Local Group Policy Management Editor, with the user and computer nodes. For example, if you drill down into the Administrative Templates folder in the User Configuration ➢ Policies folder, you can make a wish at the domain level and all your computers will obey.

For now, don't make any changes; just close the Group Policy Management Editor and read on. This was just a "fantasy walk down memory lane."

In our graphic earlier, we were able to create our first GPO (even though we didn't actually place any policy settings in there). The interface seems reasonable enough to take care of such simple tasks. And, heck, this interface is already part of the operating system, so why move away from it?

The old-school way of viewing and managing Group Policy just isn't scalable over the long haul. This interface doesn't show us any relationship between the GPO we just created and the domain it's in. As you'll see in this chapter, the new interface demonstrates a much clearer relationship between the GPOs you create, the links it takes to use them, and the domains where the GPOs actually "live."

The old-style interface also provides no easy way to figure out what's going on inside the GPOs you create. To determine what changes are made inside a GPO, you need to reopen each GPO and poke around. I've seen countless administrators open each and every GPO in their domain and manually document their settings on paper for backup and recovery purposes.

Indeed, backup and recovery is a really, really big deal, and the old-school mechanism (via NTBACKUP) provided no realistic way to back up and recover GPOs without copious amounts of surgery. The GPMC makes it a snap.

With that in mind, I encourage all of you—(and I know at least a few of you are still out there) if you're currently using the original Windows 2000 or Windows Server 2003 old-school way and haven't yet switched over to the GPMC, now is your time.

GPMC Overview

The Group Policy Management Console (GPMC) was created to help administrators work in a "one-stop-shop" place for all Group Policy management functions. Since 2003, it was freely downloadable as an add-on to either Windows XP or Windows Server 2003 systems. In Windows Vista, it was included in the box. Again, that is, until you load Windows Vista's SP1, when the GPMC is uninstalled until you fetch the RSAT tools. Again, the GPMC is built into Windows Server 2008 and Windows Server 2008 R2, and available for download and installtion for Windows 7 as part of the RSAT tools.

Even though I've said it before, it bears repeating: it doesn't matter if your Active Directory or domains or Domain Controllers are Windows 2000, Windows 2003, or Windows 2008 or whatever. You can use any flavor of the GPMC and create and use Group Policy, regardless of the domain type.

About the GPMC

The GPMC's name says it all. It's the Group Policy Management Console. Indeed, this will be the MMC snap-in that you use to manage the underlying Group Policy mechanism. The

GPMC just helps us tap into those features already built into Active Directory. I'll highlight the mechanism of how Group Policy works throughout the next three chapters.

One major design goal of the GPMC is to get a Group Policy–centric view of the lay of the land. Compared with the old interface (see the sidebar "Kickin' It Old School" earlier), the GPMC does a much better job of aligning the user interface of Group Policy with what's going on under the hood.

The GPMC also provides a programmatic way to manage your GPOs. In fact, the GPMC scripting interface allows just about any GPO operation. The older GPMC that works on XP and 2003 has a way to script using VBScript. The newer GPMC that works on Windows 7 and Windows Server 2008 R2 can use VBScript or PowerShell.

We'll explore scripting with the GPMC and PowerShell in a downloadable bonus chapter. So, if you're interested in scripting, you'll need to have the GPMC bits loaded on the Windows 7 system you want to script.

The GPMC scripts, which were previously part of the downloadable GPMC package, are not included in the newest GPMC. You have to specifically download them from the GPMC scripting center at `http://tinyurl.com/23xfz3` or search for "Group Policy Management Console Sample Scripts" in your favorite search engine.

There are lots of ways you *could* manage your Group Policy universe. Some people walk up to their Domain Controllers, log onto the console, and manage their Group Policy infrastructure there. Others use a *management workstation* and manage their Group Policy infrastructure from their own Windows 7 (suggested) or Windows XP (if you must) management workstation.

I'll talk more about the use and best practices of a Windows 7 management workstation in Chapter 6.

Implementing the GPMC on Your Management Station

As I mentioned, the GPMC isn't built into Windows 7. But it is built into Windows Server 2008 and Windows Server 2008 R2. Remember earlier I stated that you could manage your Active Directory from anywhere. And this is true. You *could* walk up to a Domain Controller, you *could* install the GPMC on a Windows XP or Windows Server 2003 server, or you *could* use Terminal Services to remotely connect to a Domain Controller.

But in this book, you won't be. Your ideal management station is a Windows 7 machine (where we'll manually introduce the GPMC) or a Windows Server 2008 R2 machine (which is ready to go, no pesky downloads needed).

If you must use something else (Windows XP, Windows Server 2003, or Windows Vista RTM), you'll see me pepper in some advice for those. But you'll really want to use the recommended set to get the most out of this book.

Since I'd like to encourage you to utilize "the most modern GPMC" possible, I'm going to specifically shun both Windows Vista and Windows XP from discussion here. Yes, it's true you could use Vista and you could use XP, but it's honestly not a great idea anymore. In Chapter 6, I will cover why this is the case, but for now, let's "get modern" and assume you'll be using a Windows 7 machine.

Using a Windows 7 or Windows Server 2008 R2 Management Station

For this book, and for real life, I recommend that you use what's known as a Windows 7 management station. And, to make use of it to implement Group Policy in your domain, you'll need to introduce the downloadable GPMC on it.

Note that you could *also* use a Windows Server 2008 R2 machine as your management station. Honestly, the Windows 7 GPMC that you'll download and the built-in GPMC for Windows Server 2008 R2 are equals. There's no difference. But it's simply not likely you're going to install Windows Server 2008 R2 on your laptop or desktop.

So, just to be clear, the following two ways to create and manage GPOs are equal:

- Windows 7 and the newly downloadable GPMC (contained within the RSAT tools)
- Windows Server 2008 R2 with built-in GPMC

I'll usually just refer to a Windows 7 management station, and when I say that, I mean what I have in that first bullet point. Just remember that you can use a Windows Server 2008 R2 machine as your management station too.

Now, to be supercrazy, ridiculously clear: you could also use any of the other GPMCs out there, and things will basically "work." I delve into this in serious detail in Chapter 6, but here's the Cliffs Notes, er, Jeremy's Notes version of "What GPMC should I use?":

- Always use Windows 7 (or Windows Server 2008 R2) as your management station and you'll always be able to control all operating systems' settings from all operating systems.
- The next best choice would be Windows Vista/SP1 (or SP2) and RSAT and/or Windows Server 2008. Those two GPMCs are equivalent.
- The next best GPMC would be the downloadable version for Windows Server 2003 and Windows XP.

But, if you have even one Windows 7 or Windows Vista client machine (say in Sales or Marketing), in order to manage all the settings, you're going to need to manage the machine using a "modern" GPMC. So I'm suggesting you just bite the bullet and get yourself a copy of Windows 7 and do your management from there.

Again, more details later, but here's the warning. If you create a GPO using a "newer GPMC" (say, using a Windows 7 or Windows Server 2008 R2 GPMC) but then edit it using an older operating system (say, a Windows XP GPMC), you might not be able to "see" all the configurable options. And what's worse, some settings might be set (but you wouldn't be able to see them!). Only the newest GPMC can see the "stuff" that the newest GPMC puts into the GPO.

What if you're not "allowed" to load Windows 7 on your own management station? Well, you've got another option. Perhaps you can create a Windows 7 or Windows Server 2008 R2 machine to act as your management station, say in the server room. Or, use VMware Workstation or Virtual PC to make an "almost real" management machine. Or, do create a real machine, but set up Terminal Services or Remote Desktop to utilize the GPMC remotely.

Again, in our examples we'll call our machine WIN7MANAGEMENT, but you can use either a Windows 7 or Windows Server 2008 R2 for your best management station experience.

Using a Windows Server 2008 R2 Machine as Your Management Station

The latest GPMC is available in Windows Server 2008 R2. However, it's not magically installed in most cases. The only time it is just "magically there" is when you make your Windows Server 2008 or Windows Server 2008 R2 machine a Domain Controller. In that case, the GPMC is automatically installed for you. You don't need to do the following procedure.

And, if you're following along in the labs, you've likely already made your Windows Server 2008 R2 machine a Domain Controller. But for practice, if you want to learn how to install it for when your Windows Server 2008 and Windows Server 2008 R2 computers are not acting like Domain Controllers, there are two ways to install the GPMC: using Server Manager and also by the command line.

To install the GPMC using Server Manager:

1. Click Start, then point to Administrative Tools and select Server Manager.
2. In the Server Manager's console tree, click Features and then select Add Features.
3. In the Add Feature wizard, select Group Policy Management Console from the list of features.
4. Click Install.

Close Server Manager once you're done.

You can also install the GPMC using the command line:

1. Open a command prompt as an Administrator.
2. In the command prompt, type `ServerManagerCmd -install gpmc`.
3. Close the command prompt when the installation has been completed.

For you PowerShell gurus, you could also use the `"Add-WindowsFeature"` cmdlet in Windows Server 2008 R2.

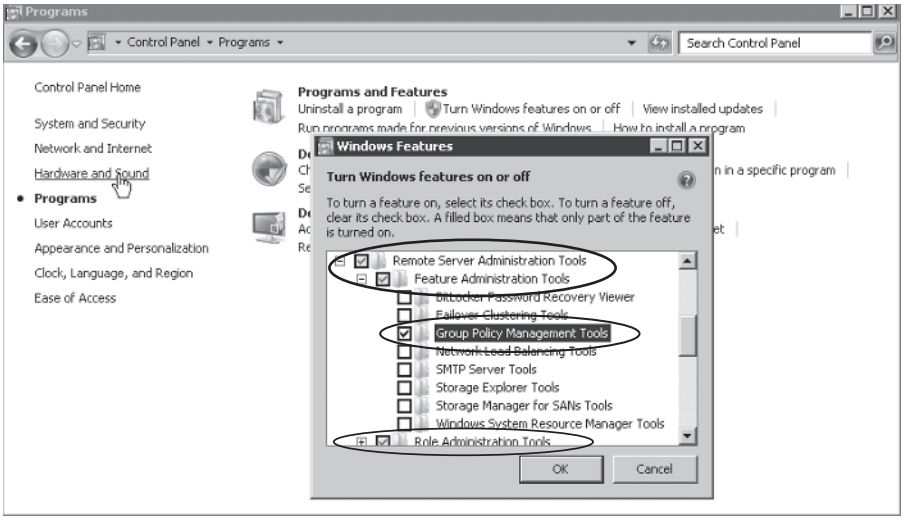
Using Windows 7 as Your Management Machine

The first step on your Windows 7 management-station-to-be is to install Windows 7. Next, you'll need to install RSAT. Find the RSAT for Windows 7 at `http://tinyurl.com/qru5en`.

RSAT installs like a hotfix, and you may or may not need to reboot after installation.

Then, on Windows 7 go to Control Panel and select Programs. Select "Turn Windows features on or off." Locate the Feature Administration Tools and select Group Policy Management Tools. Also, since you're here anyway, locate the Role Administrator Tools, as seen in Figure 1.7, and select Active Directory Administrative Center.

FIGURE 1.7 The RSAT tools install like a hotfix but then must be individually selected using Control Panel ➢ Programs ➢ Turn Windows features on or off.



Once you're done, close the Windows Features window and, if prompted, reboot your Windows 7 machine. The next time you boot, you'll have Active Directory Users and Computers, the GPMC, and other tools available for use in the rest of the book.

If You Must Use a Windows Vista RTM, Windows Vista/SP1 (or SP2) and RSAT, Windows XP, or Windows Server 2003 Management Station

Again, I recommend against using any of the older GPMCs. If you positively cannot use a Windows Vista/SP1 and RSAT machine to be your management station, and you must limp along with a Windows Vista RTM, Windows XP, or Windows Server 2003 machine, you can.

But know that you won't get the full experience, and your screen might look different from my screen shots.

Read Chapter 6 for the full implications of being forced to use an older management machine.

If You Must Use a Windows Vista RTM Machine On your Windows Vista machine, click Start, and in the Start Search prompt, type the **`GPMC.MSC`** command. With Windows Vista RTM, the GPMC will just fire right up.

If You Must Use a Windows Vista/SP1 (or SP2) and RSAT Machine At last check, Vista's RSAT was found at `http://tinyurl.com/3cch2h`. This is preferred over the "in the box" GPMC that came with Windows Vista RTM.

If You Must Use a Windows XP or Windows Server 2003 Management Station Now, what if you really, really cannot use a Windows 7 or, heck, even a Windows Vista or Windows Server 2008 management station? Well, then, sounds like you're stuck with Windows XP or Windows Server 2003. If you're being forced to use Windows XP or Windows Server 2003 as your management station, you can download the older GPMC for free from `http://tinyurl.com/566ru`.

To be honest, I don't know how much longer they'll maintain the original GPMC. I wouldn't be surprised if, some time soon, the only GPMC available will be inside the RSAT packages for Windows Vista and Windows 7. Again, you can install this older GPMC (downloaded as `GPMC.MSI`) on either Windows 2003 or Windows XP with at least SP1.

Creating a One-Stop-Shop MMC

As you'll see, the GPMC is a fairly comprehensive Group Policy management tool. But the problem is that right now the GPMC and the Active Directory Users and Computers snap-ins are, well, separate tools that each do a specific job. They're not integrated to allow you to work on both Users *and* Group Policy at the same time.

Often, you'll want to change a Group Policy linked to an OU and then move computers to that OU. Unfortunately, you can't do so from the GPMC; you must return to Active Directory Users and Computers to finish the task. This can get frustrating quickly. But that's the deal.

As a result, my preference is to create a custom MMC that shows both the Active Directory Users and Computers and GPMC in a one-stop-shop view. You can see what I mean in Figure 1.8.

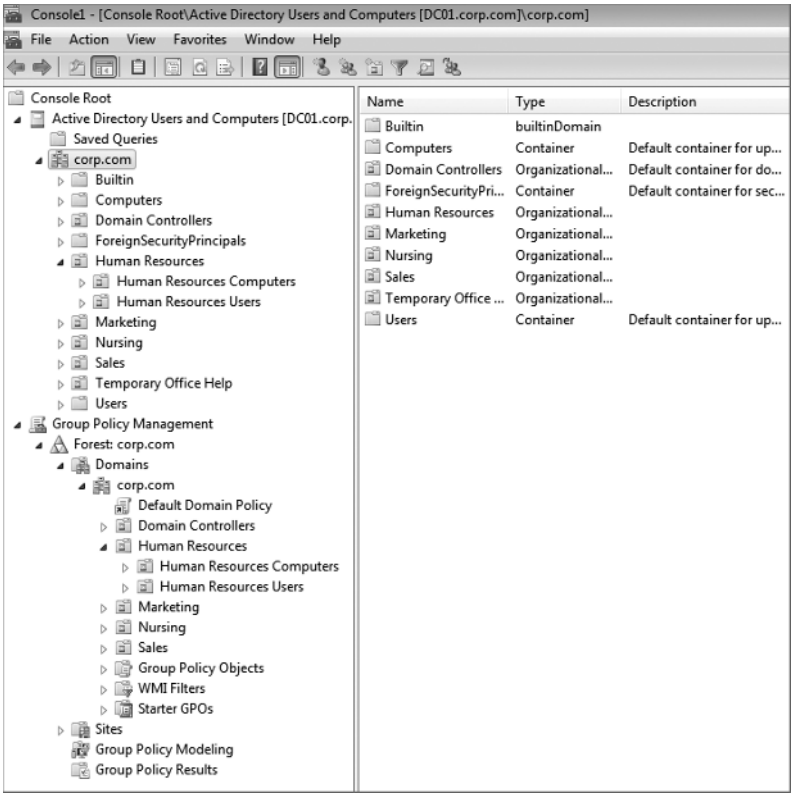
You might be wondering at this point, "So, Jeremy, what are the steps I need in order to create this unified MMC console you've so neatly described and shown in Figure 1.8?"

Just click Start and type **`MMC`** at the Search prompt. Then add in both the Active Directory Users and Computers and Group Policy Management snap-ins, as shown in Figure 1.9.

You won't need the Group Policy Management Editor (which allows you to edit one Group Policy Object at a time), the Group Policy Object Editor (for Local Group Policy), or the Group Policy Starter GPO Editor (which we use in Chapter 2).

Once you have added both snap-ins to your console, you'll really have a near-unified view of most of what you need at your fingertips. Both Active Directory Users and Computers and the GPMC can create and delete OUs. Both tools also allow administrators to delegate permissions to others to manage Group Policy, but that's where the two tools' functionality overlap ends.

FIGURE 1.8 Use the MMC to create a unified console.



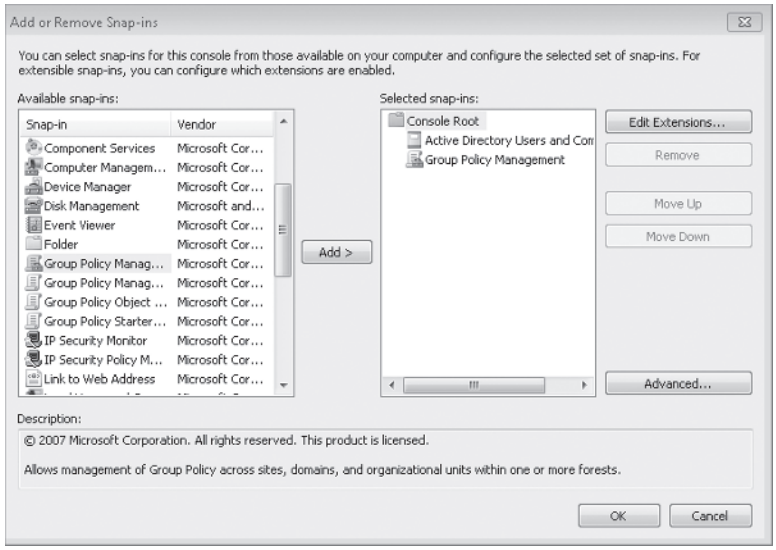
The GPMC won't show you the actual users and computer objects inside the OU, so deleting an OU from within the GPMC is dicey at best because you can't be sure of what's inside!

You can choose to add other snaps-ins too, of course, including Active Directory Sites and Services or anything else you think is useful. The illustrations in the rest of this book will show both snap-ins loaded in this configuration. I suggest you save your "one-stop shop" to the desktop and give it catchy name so you can quickly find it later when you need to.

Group Policy 101 and Active Directory

Let's start with some basics to ensure that things are running smoothly. For most of the examples in this book, you'll be able to get by with just the one Domain Controller and one or two workstations that participate in the domain, for verifying that your changes took place.

FIGURE 1.9 Add Active Directory Users and Computers and Group Policy Management to your custom view.



For the examples in this book, I'll refer to our sample Domain Controller, DC01, which is part of my example Corp.com domain. For these examples, you can choose to rename the Default-First-Site-Name site or not—your choice.

Again, I encourage you to try these examples in your test lab and not to try them directly on your production network. This will help you avoid a CLM (career-limiting move).

For our examples, we'll assume you're using WIN7MANAGEMENT as your management station, which is a Windows 7 and RSAT machine.

Active Directory Users and Computers vs. GPMC

The main job of Active Directory Users and Computers is to give you an Active Directory object–centric view of your domain. Active Directory Users and Computers lets you deal with users, computers, groups, contacts, some of the Flexible Single Master Operations (FSMO) roles, and delegation of control over user accounts as well as change the domain mode and define advanced security and auditing inside Active Directory. You can also create OUs and move users and computers around inside those OUs. Other administrators can then drill down inside Active Directory Users and Computers into an OU and see the computers, groups, contacts, and so on that you've moved to those OUs.

But the GPMC has one main job: to provide you with a Group Policy–centric view of all you control. All the OUs that you see in Active Directory Users and Computers are visible in the GPMC. Think about it—it's the same Active Directory behind the scenes "storing" those details about the OU and its contents.

However, the GPMC just doesn't have a way to "view" the users, computers, contacts, and such. When you drill down into an OU inside the GPMC, you'll see but one thing: the GPOs that affect the objects inside the OU.

In Figure 1.8, you were able to see the Active Directory Users and Computers view as well as the GPMC view—rolled into one MMC that we created earlier. Even though it's not super-obvious from the screen shot, the Active Directory Users and Computers view of **Temporary Office Help** and the GPMC view of the same OU is radically different.

When focused at a site, a domain, or an OU within the GPMC, you see only the GPOs that affect that level in Active Directory. You don't see the same "stuff" that Active Directory Users and Computers sees, such as users, computers, groups, or contacts.

The basic overlap in the two tools is the ability to create and delete OUs. If you add or delete an OU in either tool, you need to refresh the other tool by pressing F5 to see the update. For instance, in Figure 1.8 you could see that my Active Directory has several OUs, including one named **Temporary Office Help**.

Deleting an OU from inside the GPMC is generally a bad idea. Because you cannot see the Active Directory objects inside the OU (such as users and computers), you don't really know how many objects you're about to delete. So be careful!

If I delete the **Temporary Office Help** OU in Active Directory Users and Computers, the change is not reflected in the GPMC window until it's refreshed.

And vice versa.

So, let's summarize with three key points:

- Understanding that the two tools are "separate" and work on the same underlying database is key.
- Understanding that what you do in one tool (i.e., delete an OU) affects the other tool (because it's affecting the same underlying database) is also key.
- The final key is realizing that you will need to occasionally "refresh" the view of each tool. This is because other administrators might be "doing stuff" to the GPOs and/or Active Directory user accounts. You won't see their changes until you refresh *your* view.

Adjusting the View within the GPMC

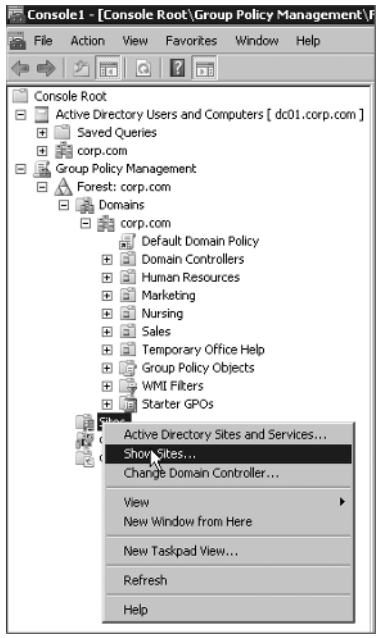
The GPMC lets you view as much or as little of your Active Directory as you like. By default, you view only your own forest and domain. You can optionally add in the ability to see the sites in your forest as well as the ability to see other domains in your forest or domains in other forests, although these views might not be the best for seeing what you have control over.

Here's how to view the various other items you may need to within the GPMC.

Viewing sites in the GPMC When you create GPOs, you won't often create GPOs that affect sites. The designers of the GPMC seem to agree; it's a bit of a chore to apply GPOs to sites. To do so, you need to link an *existing* GPO to a site. You'll see how to do this a bit later in this chapter.

However, you first need to expose the site objects in Active Directory. To do so, right-click the Sites object in GPMC, choose Show Sites from the context menu (see Figure 1.10), and then click the check box next to each site you want to expose.

FIGURE 1.10 You need to expose the Active Directory sites before you can link GPOs to them.



In our first example, we'll use the site level of Active Directory to deploy our first Group Policy Object. At this point, go ahead and enable the Default-First-Site so that you can have it ready for use in our own experiments.

Viewing other domains in the GPMC To see other domains in your forest, drill down to the Forest folder in Group Policy Management, right-click Domains, choose Show Domains, and select the other available domains in your forest. Each domain will now appear at the same hierarchical level in the GPMC.

Viewing other forests in the GPMC To see other forests, right-click the root (Group Policy Management) and choose Add Forest from the context menu. You'll need to type the name of the Active Directory forest you want to add. If you want to add or subtract domains within that new forest, follow the instructions in the preceding paragraph.

You can add forests with which you do not have a trust. However, GPMC defaults will not display these domains as a safety mechanism. To turn off the safety, choose View ➢ Options to open the Options dialog box. In the General tab, clear Enable Trust Detection and click OK.

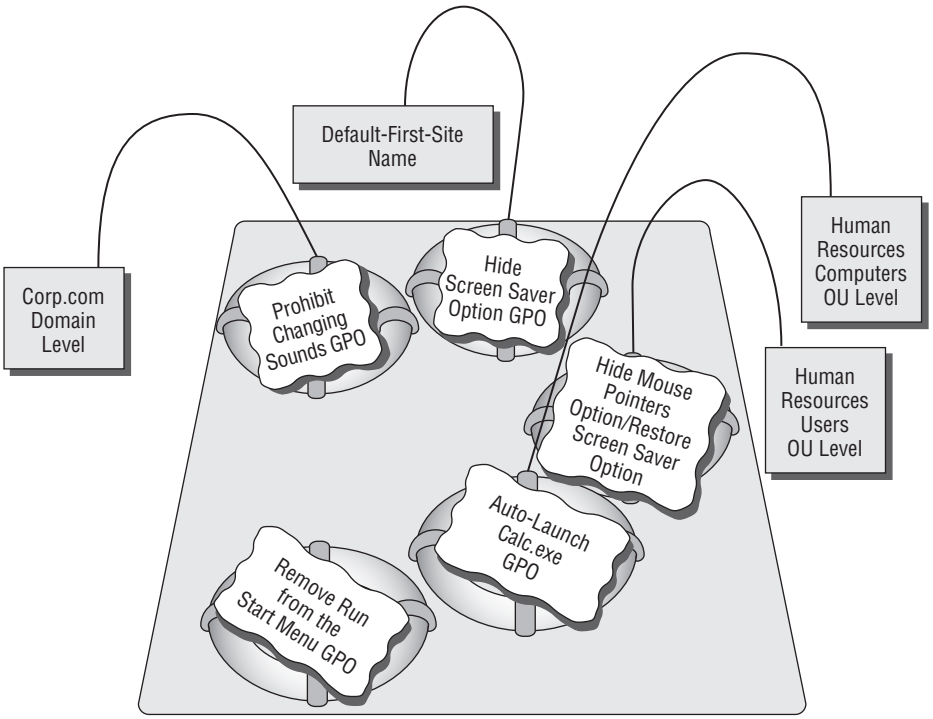
Now that we've adjusted our view to see the domains and forests we want, let's examine how to manipulate our GPOs and GPO links.

The GPMC-centric View

As I stated earlier, one of the fundamental concepts of Group Policy is that the GPOs *themselves* live in the "swimming pool" inside the domain. Then, when you want to utilize a GPO from that swimming pool against a level in Active Directory, you simply link a GPO to that level.

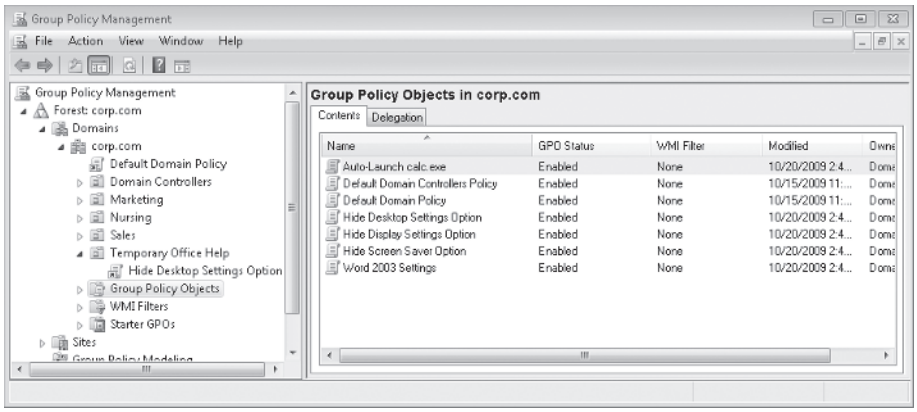
Figure 1.11 shows what our swimming pool will eventually look like when we're done with the examples in this chapter.

FIGURE 1.11 Imagine your about-to-be-leveraged GPOs as just hanging out in the swimming pool of the domain.



Our swimming pool will be full of GPOs, with various levels in Active Directory "linked" to those GPOs. To that end, you can drill down, right now, to see the representation of the swimming pool. It's there, waiting for you. Click Group Policy Management ➢ Forest ➢ Domains ➢ Corp.com ➢ Group Policy Objects to see all the GPOs that exist in the domain (see Figure 1.12).

FIGURE 1.12 The Group Policy Objects folder highlighted here is the representation of the swimming pool of the domain that contains your actual GPOs.



If you're just getting started, it's not likely you'll have more than the "Default Domain Controllers Policy" GPO and "Default Domain Policy" GPO. That's OK. You'll start getting more GPOs soon enough. Oh, and for now, please don't modify the default GPOs. They're a bit special and are covered in great detail in Chapter 8.

All GPOs in the domain are represented in the Group Policy Objects folder. As you can see, when the **Temporary Office Help** OU is shown within the GPMC, a relationship exists between the OU and the "Hide Desktop Settings Option" GPO. That relationship is the tether to the GPO in the swimming pool—the GPO link back to "Hide Desktop Settings Option." You can see this linked relationship because the "Hide Desktop Settings Option" icon inside **Temporary Office Help** has a little arrow icon, signifying the link back to the actual GPO in the domain. The same is true for the "Default Domain Policy," which is linked at the domain level, but the actual GPO is placed below the Group Policy Objects folder.

Our Own Group Policy Examples

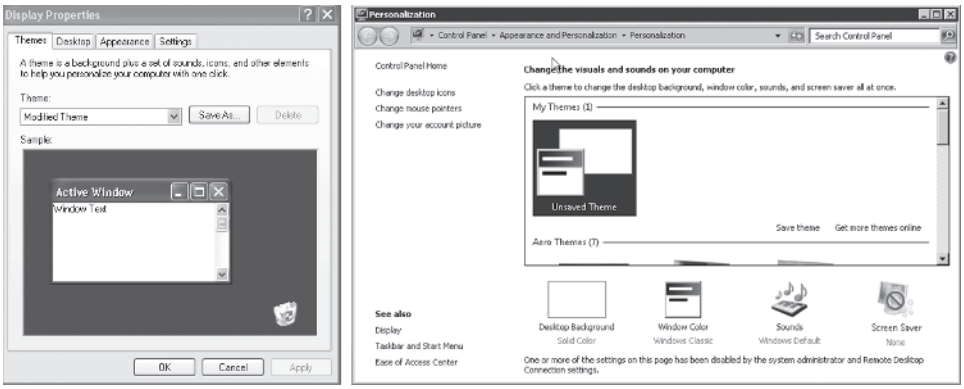
Now that you've got a grip on honing your view within the GPMC, let's take it for a quick spin around the block with some examples!

For this series of examples, we're going after the users who keep fiddling with their display doo-dads in Windows 7 and Windows XP.

If you want to see these examples in action using Windows XP, first start out on XPPRO1 by checking out the default Display Properties dialog box. Just right-click the Desktop and choose Properties from the context menu. You'll see several tabs, including Screen Saver, Appearance, and Settings, as shown in Figure 1.13 (left screen).

If you want to see these examples in action using Windows 7, first start out on WIN7 by looking at the "Personalize appearance and sounds" page, which is located by right-clicking the Desktop and choosing Personalize. You'll see several entries, including Screen Saver, Windows Color and Appearance, and Display Settings, as shown in Figure 1.13 (right screen).

FIGURE 1.13 In Windows XP, all the tabs in the Display Properties dialog box are available by default (left screen). In Windows 7, we can see lots of available areas in the Personalization screen, shown on the right.



Since they're called tabs in Windows XP and entries or options in Windows Vista and Windows 7, I'll just generally call them options from here on out.

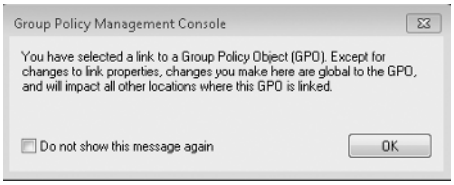
For our first use of Group Policy, we're going to produce four "edicts" (for dramatic effect, you should stand on your desk and loudly proclaim these edicts with a thick British accent):

- At the site level, there will be no ability to change screen savers.
- At the domain level, there will be no ability to change Windows' sounds.
- At the **Human Resources Users** OU level, there will be no way to change the mouse pointers. And, while we're at it, let's bring back the ability to change screen savers!
- At the **Human Resources Computers** OU, we'll make it so whenever anyone uses a Human Resources computer, `calc.exe` automatically launches after login.

Following along with these concrete examples will reinforce the concepts presented earlier. Additionally, they are used throughout the remainder of this chapter and the book.

Understanding GPMC's Link Warning

As you work through the examples, you'll do a lot of clicking around. When you click a GPO link the first time, you'll get this message:



This message is trying to convey an important sentiment—that is, multiple levels in Active Directory may be linked back and use the exact same GPO. The idea is that multiple levels of Active Directory could use the exact same Group Policy Object contained inside the Group Policy Objects container—but just be linked back to it.

What if you modify the policy settings by right-clicking a policy link and choosing Edit from the context menu? All instances in Active Directory that link to that GPO embrace the new settings. If this is a fear, you might want to create another GPO and then link it to the level in Active Directory you want. More properties are affected by this warning, and we'll explore them in Chapter 5.

If you've squelched this message by selecting "Do not show this message again," you can get it back. In the GPMC in the menus, choose View ➢ Options and select the General tab, then select "Show confirmation dialog box to distinguish between GPOs and GPO links," and click OK.

More about Linking and the Group Policy Objects Container

The GPMC is a fairly flexible tool. Indeed, it permits the administrator to perform many tasks in different ways. One thing you'll do quite a lot in your travels with the GPMC is create your own Group Policy Objects. Again, GPOs live in a container within Active Directory and are represented within the Group Policy Objects container (the swimming pool) inside the domain (seen in Figure 1.11, earlier in this chapter). Any levels of Active Directory—site, domain, or OU—simply link back to the GPOs hanging out in the Group Policy Objects container.

To apply Group Policy to a level in Active Directory using the GPMC, you have two options:

- Create the GPOs in the Group Policy Objects container first. Then, while focused at the level you want to command in Active Directory (site, domain, or OU), manually add a link to the GPO that is in the Group Policy Objects container.
- While focused at the level you want to command in Active Directory (domain or OU), create the GPOs in the Group Policy Objects container and automatically create the link. This link is created at the level you're currently focused at *back* to the GPO in the Group Policy Objects container.

Which is the correct way to go? Both are perfectly acceptable because both are really doing the same thing.

In both cases the GPO itself does not "live" at the level in Active Directory at which you're focused. Rather, the GPO itself "lives" in the Group Policy Objects container. The link back to the GPO inside the Group Policy Objects container is what makes the relationship between the GPO inside the Group Policy Objects container swimming pool and the level in Active Directory you want to command.

To get the hang of this, let's work through some examples. First, let's create our first GPO in the Group Policy Objects folder. Follow these steps:

1. Launch the GPMC. Click Start, and then in the search box, type **`GPMC.MSC`**.
2. Traverse down by clicking Group Policy Management ➢ Forest ➢ Domains ➢ Corp.com ➢ Group Policy Objects.
3. Right-click the Group Policy Objects folder and choose New from the context menu, as shown in Figure 1.14, to open the New GPO dialog box.
4. Let's name our first edict, er, GPO, something descriptive, such as "Hide Screen Saver Option."
5. Once the name is entered, you'll see the new GPO listed in the swimming pool. Right-click the GPO, and choose Edit, as shown in Figure 1.15, to open the Group Policy Management Editor.
6. To hide the Screen Saver option, drill down by clicking User Configuration ➢ Policies ➢ Administrative Templates ➢ Control Panel ➢ Personalization. Double-click the **Prevent Changing Screen Saver** policy setting to open the policy setting. Select the Enabled setting, and click OK. You can see the new Windows 7 policy editor user interface in the sidebar "Old Policy Settings UI vs. New Policy Settings UI."
7. Close the Group Policy Management Editor.

Note that in earlier iterations of the GPMC, this setting was named differently and placed in another node. It used to be called **Hide Screen Saver Tab** and was located in the Display node within Control Panel. As you can see, as the operating system changes, so must the GPMC. Which is why it's pretty important to always use the "latest, greatest" GPMC, like we are using in this book.

FIGURE 1.14 You create your first GPO in the Group Policy Objects container by right-clicking and choosing New.

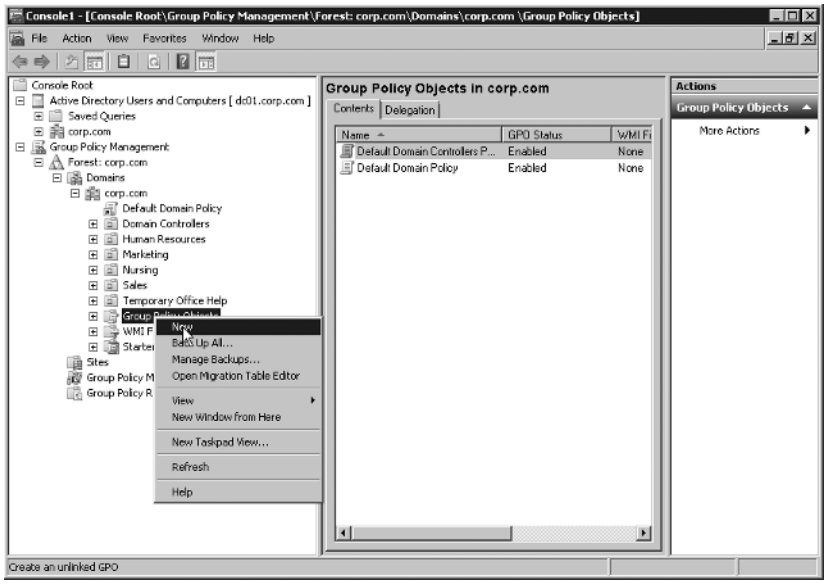
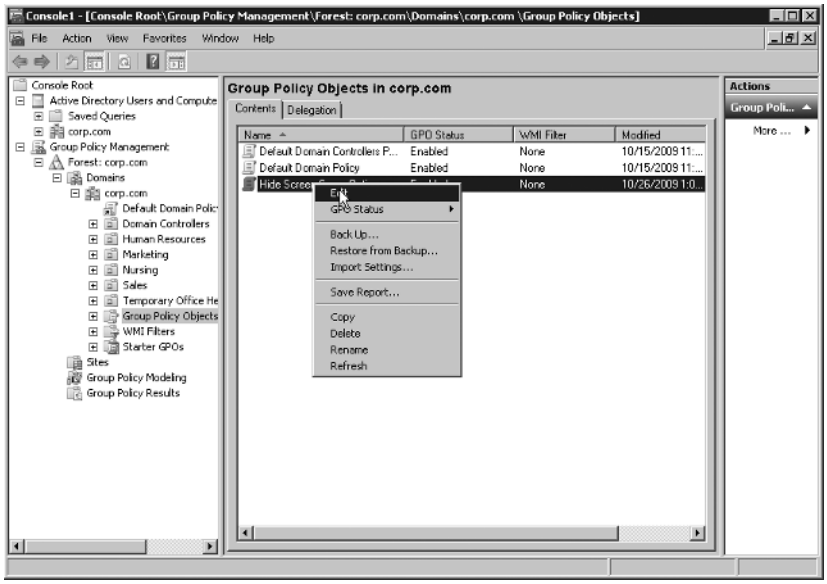
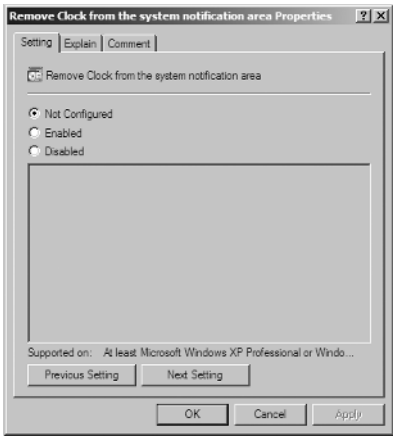


FIGURE 1.15 You can right-click the GPO in the Group Policy Objects container and choose Edit from the context menu to open the Group Policy Management Editor.



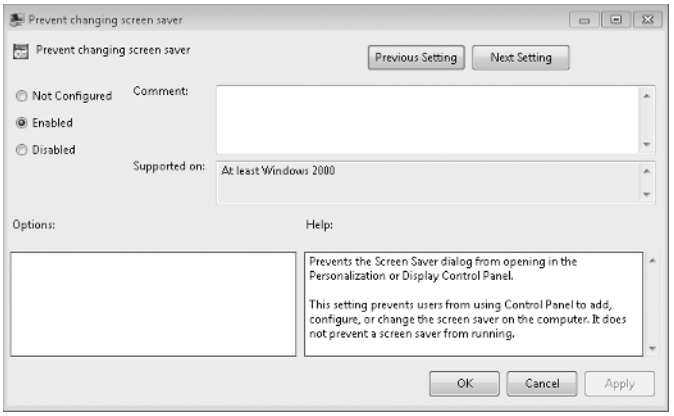
Old Policy Settings UI vs. New Policy Settings UI

If you've spent any amount of time in the Group Policy Editor over the years, this picture will be a familiar sight. Starting with Windows Vista, the Comment tab was added (and, don't worry; we'll examine it thoroughly in the next chapter.) But the problem with this UI is that it really wasn't "one-stop shop." It had two (now, more recently three) tabs to click through to see everything about the policy setting at once.



Now, starting with Windows 7 and Windows Server 2008 R2's GPMC, the UI has changed so that everything about the policy setting is in one quick "overview." All sections previously hidden behind the three tabs are all here.

This might freak out some seasoned Group Policy admins when they see it the first time. I know it took me a little while to get used to it. But in time, I think you'll grow to like it. Oh, and as an added bonus, you can resize the policy setting window. Neat.



Understanding Our Actions

Now that we have this "Hide Screen Saver Option" edict, er, GPO floating around in the Group Policy Objects container—in the representation of the swimming pool of the domain—what have we done? Not a whole lot, actually, other than create some bits inside Active Directory and on the Domain Controllers. By creating new GPOs in the Group Policy Objects folder, we haven't inherently forced our desires on *any* level in Active Directory—site, domain, or OU.

To make a level in Active Directory accept our will, we need to *link* this new Group Policy Object to an existing level. Only then will our will be accepted and embraced. Let's do that now.

Applying a Group Policy Object to the Site Level

The least-often-used level of Group Policy application is at the site. This is because it's got the broadest stroke but the bluntest application. And more and more organizations use high-speed links everywhere, so it's not easy to separate computers into individual sites because (in some organizations) Active Directory is set up to see the network as just one big site!

Additionally, since Active Directory states that only members of the Enterprise Administrators (EAs) can modify sites and site links, it's equally true that only EAs (by default) can add and manipulate GPOs at the site level.

When a tree or a forest contains more than one domain, only the EAs and the Domain Administrators (DAs) of the root domain can create and modify sites and site links. When multiple domains exist, DAs in domains other than the root domain cannot create sites or site links (or site-level GPOs).

However, site GPOs might come in handy on an occasion or two. For instance, you might want to set up site-level GPO definitions for network-specific settings, such as Internet Explorer proxy settings or an IP security policy for sensitive locations. Setting up site-based settings is useful if you have one building (set up explicitly as an Active Directory site) that has a particular or unique network configuration. You might choose to modify the Internet Explorer proxy settings if this building has a unique proxy server. Or, in the case of IP security, perhaps this facility has particularly sensitive information, such as confidential records or payroll information.

Therefore, if you're not an EA (or a DA of the root domain), it's likely you'll never get to practice this exercise outside the test lab. In upcoming chapters I'll show you how to delegate these rights to other administrators, like OU administrators around the bend.

For now, we'll work with a basic example to get the feel of the Group Policy Management Editor.

We already stood on our desks and loudly declared that there will be no Screen Saver options at our one default site. The good news is that we've already done two-thirds of what we need to do to make that site accept our will: we exposed the sites we want to manage, and we created the "Hide Screen Saver Option" GPO in the Group Policy Objects container.

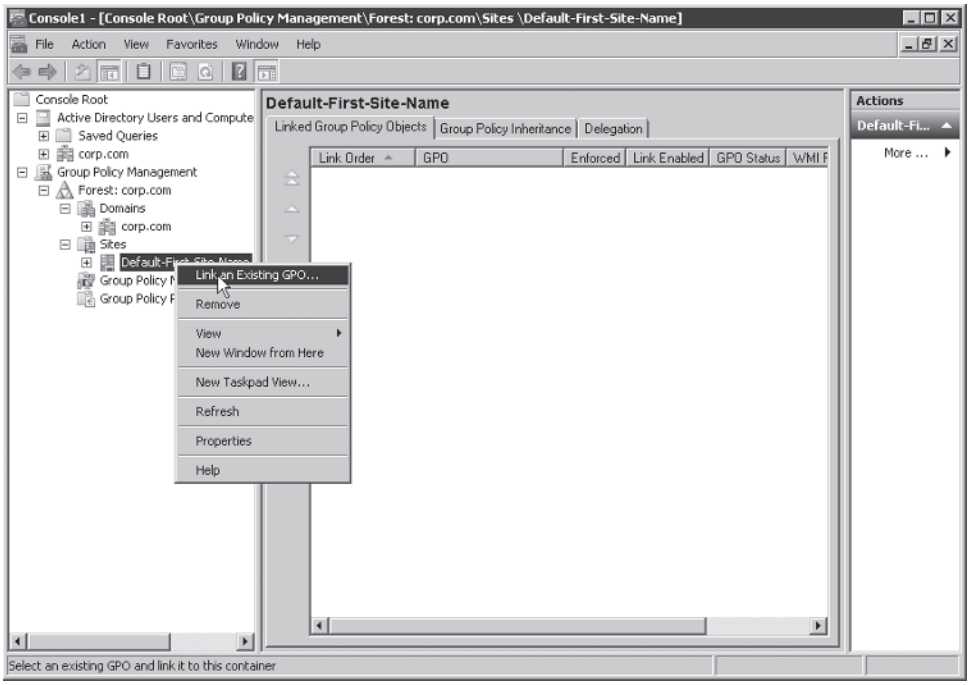
Implementing GPOs linked to sites can have a substantial impact on your logon times and WAN (wide area network) traffic if not performed correctly. For more information, see Chapter 7 in the section "Group Policy Objects from a Site Perspective."

Now all we need do is to tether the GPO we created to the site with a GPO link.

To remove the Screen Saver option using the Group Policy Management Editor at the site level, follow these steps:

1. Inside the GPMC snap-in, drill down by clicking the Group Policy Management folder, the Forest folder, and the Sites folder.
2. Find the site to which you want to deliver the policy. If you have only one site, it is likely called Default-First-Site-Name.
3. Right-click the site and choose "Link an Existing GPO," as shown in Figure 1.16.

FIGURE 1.16 Once you have your first GPO designed, you can link it to your site.



4. Now you can select the "Hide Screen Saver Option" GPO from the list of GPOs in the Group Policy Objects container within the domain.

Once you have chosen the GPO, it will be linked to the site.

Again, there is a good reason why GPOs for sites must be pre-created first. Since Sites does not belong to a specific domain, but rather the forest, you cannot assume which "domain swimming pool" they should be added to. By creating them this way, you know which domain you created them in first, and then to what site you want them linked.

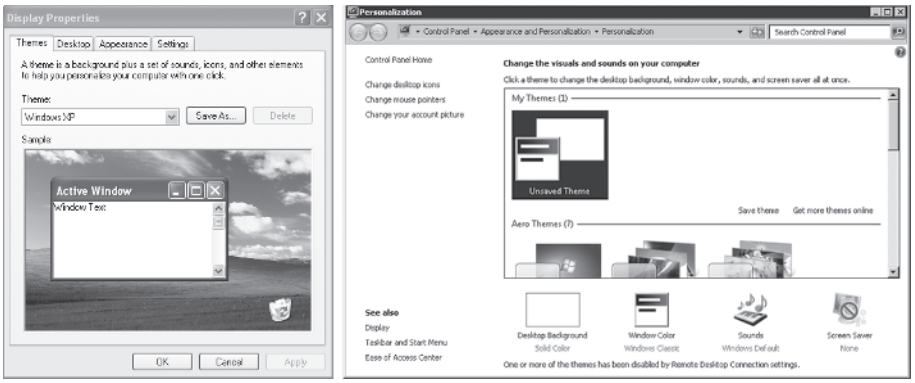
Did you notice that there was no "Are You Sure You Really Want To Do This?" warning or anything similar? The GPMC trusts that you set up the GPO correctly. If you create GPOs with incorrect settings and/or link them to the wrong level in Active Directory, you can make boo-boos on a grand scale. Again—this is why you want to try any setting you want to deploy in a test lab environment first.

Verifying Your Changes at the Site Level

Now, log onto any workstation or server that falls within the boundaries of the site to which you applied the sitewide GPO. If you didn't change any of the defaults, you should be able to log onto any computer in the domain (say, XPPRO1 or WIN7) as any user you have defined—even the administrator of the domain.

By right-clicking the Desktop and selecting Personalize (for Windows 7) or Properties (for Windows XP), you'll see that the Screen Saver option is, well, if not "gone" exactly, at least grayed out and cannot be selected, as shown in Figure 1.17.

FIGURE 1.17 The Screen Saver tab in Windows XP (shown on the left) is missing because the site policy is affecting the user. In Windows 7 (shown on the right), the Screen Saver entry on the Personalization page is disabled (grayed out).



Don't panic if you do not see the changes reflected the first time you log on. See Chapter 3, in the section "Background Refresh Policy Processing," to find out how to encourage changes to appear. To see the Screen Saver tab disappear right now, log off and log back on. The policy should take effect.

This demonstration should prove how powerful Group Policy is, not only because everyone at the site is affected, but more specifically because administrators are not immune to Group Policy effects. Administrators are not immune because they are automatically members in the Authenticated Users security group. (You can modify this behavior with the techniques explored in Chapter 3.)

Applying Group Policy Objects to the Domain Level

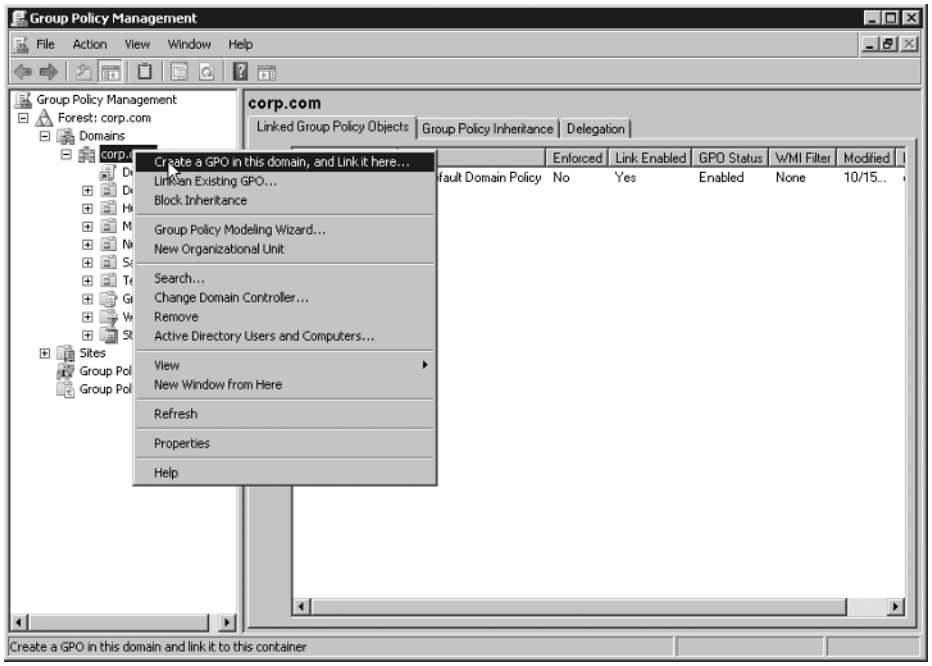
At the domain level, we want to deliver an edict that says that the Desktop Settings option in the Windows 7 Personalization page (or Desktop tab in the Display Properties dialog box for Windows XP) should be removed.

Active Directory domains allow only members of the Domain Administrators group the ability to create and link Group Policy directly on the domain level. Therefore, if you're not a DA (or a member of the EA group), or you don't get delegated the right, it's likely that you'll never get to practice this exercise outside the test lab. (We'll talk more about how to give others' rights to create and link GPOs besides Domain Admins a bit later.)

To apply the edict, follow these steps:

1. In the GPMC, drill down by clicking Group Policy Management ➢ Forest ➢ Corp.com.
2. Right-click the domain name to see the available options, as shown in Figure 1.18.

FIGURE 1.18 At the domain level, you can create the GPO in the Group Policy Objects container and then immediately link to the GPO from here.



"Create a GPO in this domain, and Link it here" vs. "Link an Existing GPO"

In the previous example, we forced the site level to embrace our "Hide Screen Saver Option" edict. First, we created the GPO in the Group Policy Objects folder, and then in another step we linked the GPO to the site level. However, at the domain level (and, as you're about to see, the OU level), we can take care of both steps at once via the "Create a GPO in this domain, and Link it here" command. (Note, in previous versions of the GPMC, this was confusingly called "Create And Link A GPO Here." Being a grammar snob, this was a personal wish of mine to have clarified, and I'm happy to see Microsoft agreed and corrected it.)

This command tells the GPMC to create a new GPO in the Group Policy Objects folder and then automatically link the new GPO back to this focused level of Active Directory. This is a time-saving step so we don't have to dive down into the Group Policy Objects folder first and then create the link back to the Active Directory level.

So why is the "Create a GPO in this domain, and Link it here" option possible only at the domain and OU level and not the site level? Because Group Policy Objects linked to sites can often cause excessive bandwidth troubles when the old-school way of doing things is used. With that in mind, the GPMC interface makes sure that when you work with GPOs that affect sites, you're consciously choosing from *which* domain the GPO is being linked.

Don't panic when you see all the possible options. We'll hit them all in due time; right now we're interested in the first two: "Create a GPO in this domain, and Link it here" and "Link an Existing GPO."

Since you're focused at the domain level, you are prompted for the name of a new Group Policy Object when you right-click and choose "Create a GPO in this domain, and Link it here." For this one, type a descriptive name, such as **Prohibit Changing Sounds**. Your new "Prohibit Changing Sounds" GPO is created in the Group Policy Objects container and, automatically, a link is created at the domain level from the GPO to the domain.

Take a moment to look in the Group Policy "swimming pool" for your new GPO. Simply drill down through Group Policy Management ➢ Forest ➢ Domains ➢ Corp.com and locate the Group Policy Objects note. Look for the new "Prohibit Changing Sounds" GPO.

Right-click the link "Prohibit Changing Sounds" (or the GPO itself) and choose Edit to open the Group Policy Management Editor. To make your wish come true and affect the Windows 7 Personalization page, drill down through User Configuration ➢ Policies ➢ Administrative Templates ➢ Control Panel ➢ Personalization, and double-click **Prevent Changing Sounds**. Change the setting from Not Configured to Enabled, and click OK. Close the Group Policy Management Editor to return to the GPMC.

Note that the policy setting will only affect Windows 7 or Windows Server 2008 R2 machines, so your Windows XP machines (if you have any) will ignore the policy setting.

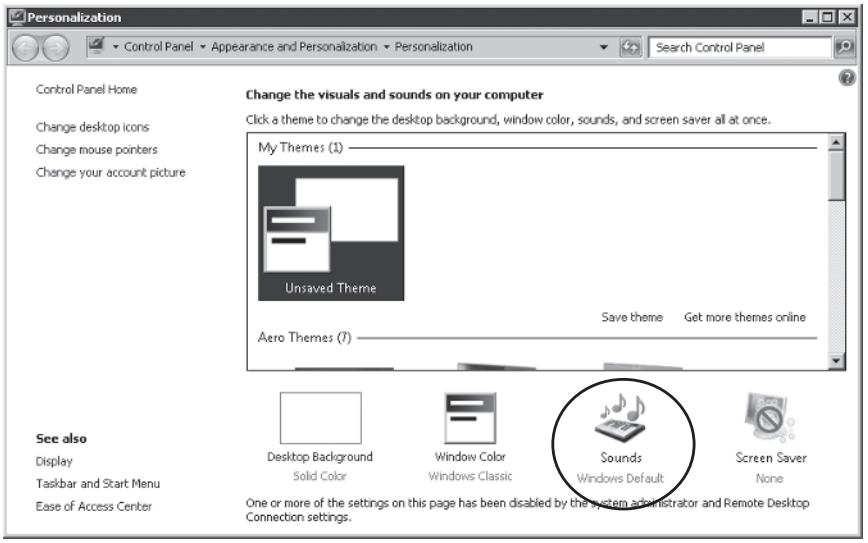
Verifying Your Changes at the Domain Level

Now, log on as any user in the domain. You can log onto any computer in the domain (say, WIN7) as any user you have defined—even the administrator of the domain.

On WIN7, right-click the Desktop and click Personalize.

You'll see (in your tests) that the Sounds area is grayed out, as seen in Figure 1.19. Well, you might not be able to see it, exactly, in Figure 1.19, but, again, it's "locked out" for users.

FIGURE 1.19 The Sounds area is now grayed out because the user is affected by the domain-level policy.



Once again, administrators are not immune to Group Policy effects. You can change this behavior, as you'll see in Chapter 2.

Applying Group Policy Objects to the OU Level

OUs are wonderful tools for delegating away unpleasant administrative duties, such as password resets or modifying group memberships. But that's only half their purpose. The other half is to be able to apply Group Policy.

You'll likely find yourself making most Group Policy additions and changes at the OU level, because that's where you have the most flexibility and the OU is the most refined instrument to affect users. Once OU administrators become comfortable in their surroundings, they want to harness the power of Group Policy.

Preparing to Delegate Control

To create a GPO at the OU level, you must first create the OU and a plan to delegate. For the examples in this book, we'll create three OUs that look like this:

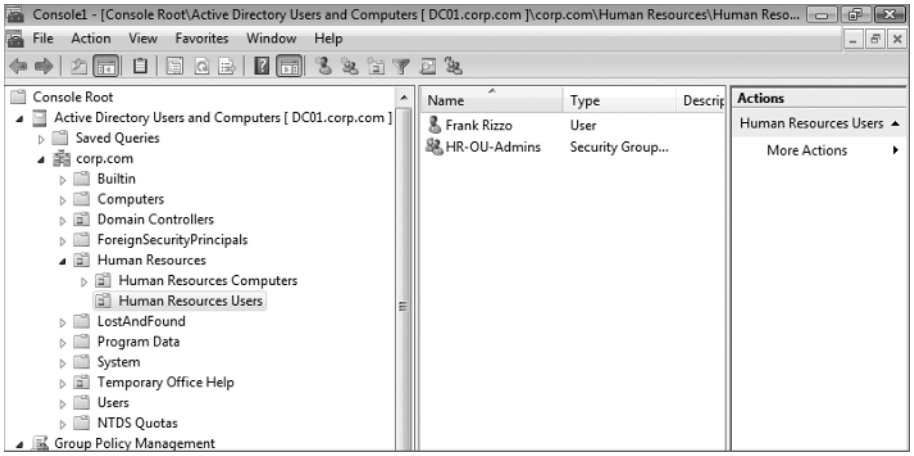
- Human Resources
  - Human Resources Users
  - Human Resources Computers

Having separate OUs for your users and computers is a good idea—for both delegation of rights and also GPO design. Microsoft considers this a best practice. In the **Human Resources Users** OU in our Corp.com domain, we'll create and leverage an Active Directory security group to do our dirty work. We'll name this group HR-OU-Admins and put our first users inside the HR-OU-Admins security group. We'll then delegate the appropriate rights necessary for them to use the power of GPOs.

To create the **Human Resources Users** OU using your WIN7MANAGEMENT machine, follow these steps:

1. Earlier, you created a "unified console" where you housed both Active Directory Users and Computer and the GPMC. Simply use Active Directory Users and Computers, right-click the domain name, and choose New ➢ Organizational Unit, which will allow you to enter a new OU name. Enter **Human Resources** as the name. (Note that newer versions of Active Directory Users and Computers will ask you if you want to "Protect container from accidental deletion." It's your choice. I typically deselect the check box.)
2. Inside the **Human Resources** OU, create two more OUs—**Human Resources Computers** and **Human Resources Users**, as shown in Figure 1.20.

FIGURE 1.20 When you complete all these steps, your Human Resources OU should have a Human Resources Users OU and Human Resources Computers OU. In the users' side, put Frank Rizzo and the HR-OU-Admins.



Alternatively, you can create the OU in the GPMC. Just right-click the domain and choose New Organizational Unit from the context menu.

To create the HR-OU-Admins group, follow these steps:

1. In Active Directory Users and Computers, right-click the new **Human Resources Users** OU and choose New ➢ Group.
2. Create the new group HR-OU-Admins as a new Global Security group.

To create the first user to go inside HR-OU-Admins, follow these steps:

1. In Active Directory Users and Computers, right-click the **Human Resources Users** OU and choose New ➢ User.
2. Name the user **Frank Rizzo**, with an account name of **frizzo**, and click Next.
3. If you've established a Windows 2003 or later domain, you must now enter a complex password for a user. My suggested password in all my books is p@ssw0rd. That's a lowercase *p*, the at sign, an *s*, an *s*, a *w*, a zero, then *r*, and *d*.
4. Finish and close the wizard.

If you're following along, Frank Rizzo's login will be `frizzo@corp.com`.

Easily Manage New Users and Computers

The Computers folder and Users folder in Active Directory Users and Computers are *not* OUs. They are generic *containers*. You'll notice that they are not present when using the GPMC to view Active Directory. Because they are generic containers (and not OUs), you cannot link Group Policy Objects to them. Of course, these objects will receive GPOs if linked to the domain, because the containers are still *in* the domain. They just aren't OUs in the domain.

These folders have two purposes:

- If you ever did an upgrade from NT 4 domains to Active Directory, these User and Computer accounts would wind up in these folders. (Administrators are then supposed to move the accounts into OUs.)
- The two folders are the default location where older tools drop new accounts when creating new users and computers. Additionally, command-line tools, such as `net user` and `net group`, will add accounts to these two folders. Similarly, the Computers folder is the default location for any new client workstation or server that joins the domain. The same goes when you create computer accounts using the `net computer` command.

So, these seem like decent "holding pens" for these kinds of objects. But ultimately, you don't want your users or computers to be in these folders—you want them to end up in OUs. That's where the action is because you can apply Group Policy to OUs, not to these folders! Yeah, sure, these users and computers are affected by site- and domain-level GPOs. But the action is at the OU level, and you want your computer and user objects to be placed in OUs as fast as possible—not sitting around in these generic Computers and Users folders.

To that end, domains that are at least at the "Windows 2003 functional level" have two tools to redirect the default location of new users and computers to the OUs of your choice. For example, suppose you want all new computers to go to a **NewComputers** OU and all new users to go to a **NewUsers** OU. And you want to link several GPOs to the **NewUsers** and **NewComputers** OUs to ensure that new accounts immediately have some baseline level of security, restriction, or protection. Without a little magic, new user accounts created using older tools won't automatically be placed there.

Starting with Windows 2003 Active Directory, Microsoft provided `REDIRUSR` and `REDIRCMP` commands that take a distinguished name, like this:

```
REDIRCMP ou=newcomputers,dc=corp,dc=com and/or
REDIRUSR ou=newusers,dc=corp,dc=com
```

Now if you link GPOs to these OUs, your new accounts will get the Group Policy Objects dictating settings to them at an OU level. This will come in handy when users and computers aren't specifically created in their final destination OUs.

To learn more about these tools, see the Microsoft Knowledge Base article 324949 at `http://support.microsoft.com/kb/324949`.

To add Frank Rizzo to the HR-OU-Admins group, follow these steps:

1. Double-click the HR-OU-Admins group.
2. Click the Members tab.
3. Add Frank Rizzo.

When it's all complete, your OU structure with your first user and group should look like Figure 1.20, shown previously.

Delegating Control for Group Policy Management

You've created the **Human Resources** OU, which contains the **Human Resources Users** OU and the **Human Resources Computers** OU and the HR-OU-Admins security group. Now, put Frank inside the HR-OU-Admins group, and you're ready to delegate control.

Performing Your First Delegation

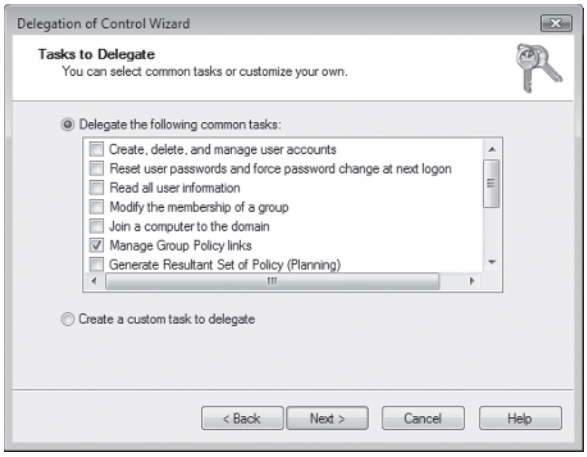
You can delegate control to use Group Policy in two ways: using Active Directory Users and Computers and using the GPMC.

For this first example, we'll kick it old school and do it the Active Directory Users and Computers way. Then, in Chapter 2, I'll demonstrate how to delegate control using the GPMC.

To delegate control for Group Policy management, follow these steps:

1. In Active Directory Users and Computers, right-click the top-level **Human Resources** OU you created and choose Delegate Control from the context menu to start the "Delegation of Control Wizard."
2. Click Next to get past the wizard introduction screen.
3. You'll be asked to select users and/or groups. Click Add, add the HR-OU-Admins group, and click Next to open the "Tasks to Delegate" screen, as shown in Figure 1.21.
4. Click Manage Group Policy Links, and then click Next.
5. At the wizard review screen, click Finish.

FIGURE 1.21 Select the Manage Group Policy Links task.



You might want to click some or all the other check boxes as well, but for this example, only "Manage Group Policy Links" is required. Avoid selecting "Generate Resultant Set of Policy (Planning)" and "Generate Resultant Set of Policy (Logging)" at this time. You'll see where these options come into play in Chapter 2.

The "Manage Group Policy Links" delegation assigns the user or group Read and Write access over the `gPLink` and `gPOptions` properties for that level. To see or modify these permissions by hand, open Active Directory Users and Computers and choose View ➢ Advanced Features. If later you want to remove a delegated permission, it's a little challenging. To locate the permission that you set, right-click the delegated object (such as OU), click the Properties tab, click the Security tab, choose Advanced, and dig around until you come across the permission you want to remove. Finally, delete the corresponding access control entry (ACE).

Adding a User to the Server Operators Group (Just for This Book)

Under normal conditions, nobody but Domain Administrators, Enterprise Administrators, or Server Operators can walk up to Domain Controllers and log on. For testing purposes only, though, we're going to add our user, Frank, to the Server Operators group so he can easily work on our DC01 Domain Controller when we want him to.

To add a user to the Server Operators group, follow these steps:

1. In Active Directory Users and Computers, double-click Frank Rizzo's account under the **Human Resources Users** OU.
2. Click the Member Of tab and click Add.
3. Select the Server Operators group and click OK.
4. Click OK to close the Properties dialog box for Frank Rizzo.

Normally, you wouldn't give your delegated OU administrators Server Operators access. You're doing it solely for the sake of this example to allow Frank to log on locally to your Domain Controllers.

Testing Your Delegation of Group Policy Management

At this point, on your WIN7MANAGEMENT machine, log off as Administrator and log in as Frank Rizzo (`frizzo@corp.com`). Heck, if you're using Windows 7 or Windows Server 2008 and later you can also use Switch User and stay logged in and just flip-flop back and forth as needed.

Now follow these steps to test your delegation:

1. Choose Start and type in **`GPMC.MSC`** at the Start Search prompt to open the GPMC.
2. Drill down through Group Policy Management, Domains, Corp.com, and Group Policy Objects. If you right-click Group Policy Objects in an attempt to create a new GPO, you'll see the context menu shown in Figure 1.22.

As you can see, Frank is unable to create new GPOs in the swimming pool of the domain. Since Frank has been delegated some control over the **Human Resources** OU (which also contains the other OUs), let's see what he *can* do. If you right-click the **Human Resources** OU in the GPMC, you'll see the context menu shown in Figure 1.23.

FIGURE 1.22 Frank cannot create new GPOs in the Group Policy Objects container.

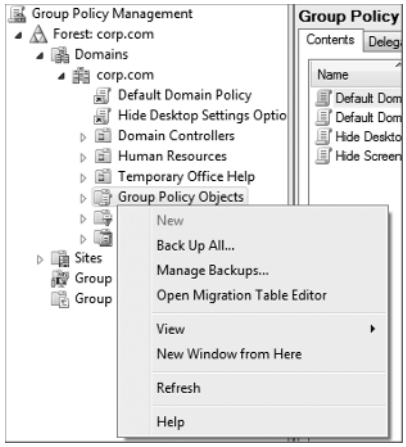
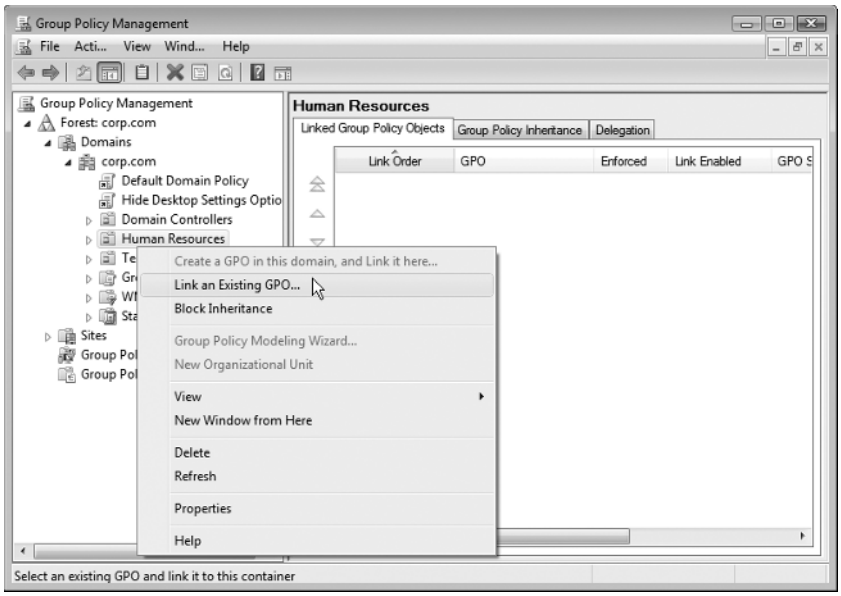


FIGURE 1.23 Frank's delegated rights allow him to link to existing GPOs but not to create new GPOs.



Because Frank is unable to create GPOs in the swimming pool of the domain (the Group Policy Objects container), he is also unable by definition to "Create a GPO in this domain, and Link it here." Although Frank (and more specifically, the HR-OU-Admins) has been delegated the ability to "Manage Group Policy Links," he cannot *create* new GPOs. Frank (and the other potential HR-OU-Admins) has only the ability to *link* an existing GPO.

Understanding Group Policy Object Linking Delegation

When we were logged on as the Domain Administrator, we could create GPOs in the Group Policy Objects container, and we could "Create a GPO in this domain, and Link it here" at the domain or OU levels. But Frank cannot.

Here's the idea about delegating the ability to link to GPOs: someone with a lot of brains in the organization does all the work in creating a well-thought-out and well-tested GPO. Maybe this GPO distributes software, maybe it sets up a secure workstation policy, or perhaps it runs a startup script. You get the idea.

Then, others in the organization, like Frank, are delegated just the ability to *link* to that GPO and use it at their level. This solves the problem of delegating perhaps too much control. Certainly some administrators are ready to create their own users and groups, but other administrators may not be quite ready to jump into the cold waters of Group Policy Object creation. Thus, you can design the GPOs for other administrators; they can just link to the ones you (or others) create.

When you (or someone with the right to link GPOs) selects "Link an Existing GPO," as seen in Figure 1.23, you can choose a GPO that's already been created—and hanging out in the domain swimming pool—the Group Policy Objects container.

In this example, the HR-OU-Admins members, such as Frank, can leverage any currently created GPO to affect the users and computers in their OU—even if they didn't create it themselves. In this example, Frank has linked to an existing GPO called "Word 2003 Settings." Turns out that some other administrator in the domain created this GPO, but Frank wants to use it. So, because Frank has Manage Group Policy Links rights on the **Human Resources** OU (and OUs underneath it), he is allowed to link to it.

But, as you can see in Figure 1.24, he cannot edit the GPOs. Under the hood, Active Directory doesn't permit Frank to edit GPOs he didn't create (and therefore doesn't own).

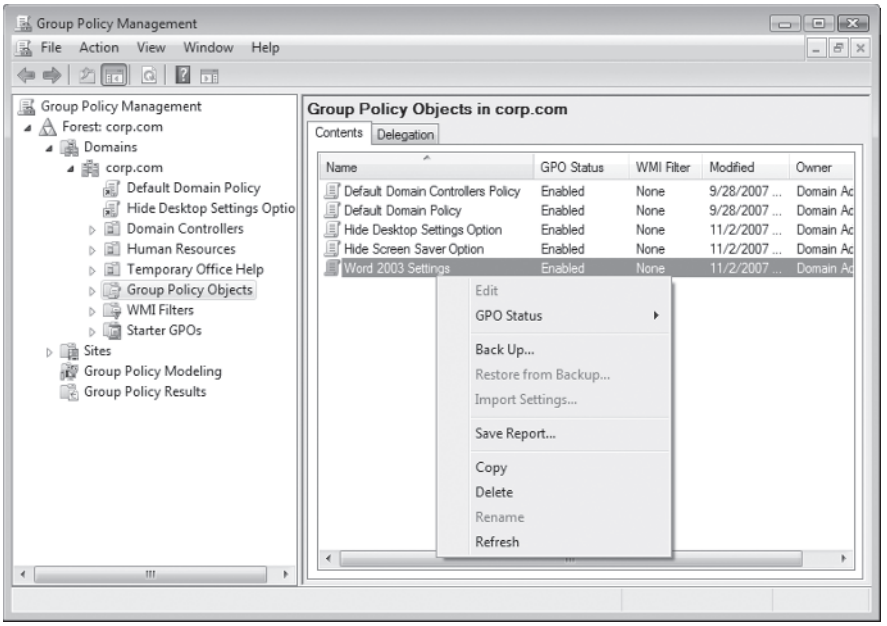
In Chapter 2, I'll show you how to grant specific rights to allow more than just the original creator (and now owner) of the object to edit specific GPOs.

Giving the ability to just link to existing GPOs is a good idea in theory, but often OU administrators are simply given full authority to create their own GPOs (as you'll see later). For this example, don't worry about linking to any GPOs. Simply cancel out of the Select GPO screen, close the GPMC, and log off from the server as Frank Rizzo.

Granting OU Admins Access to Create New Group Policy Objects

By using the "Delegation of Control Wizard" to delegate the Manage Group Policy Links attribute, you performed half of what is needed to grant the appropriate authority to Frank (and any additional future HR-OU-Admins) to create GPOs in the Group Policy Objects container and link them to the **Human Resources** OU, the **Human Resources Users** OU, or the **Human Resources Computers** OU. (Though we really don't want to link many GPOs directly to the **Human Resources** OU.)

FIGURE 1.24 The GPMC will not allow you to edit an existing GPO if you do not own it (or do not have explicit permission to edit it).



You can grant the HR-OU-Admins the ability to create GPOs in the Group Policy Objects container in two ways. For now, I'll show you the old-school way; in Chapter 2, I'll show you the GPMC way.

One of Active Directory's built-in security groups, Group Policy Creator Owners, holds the key to the other half of our puzzle. You'll need to add those users or groups whom you want to have the ability to create GPOs to a built-in group, cleverly named Group Policy Creator Owners. To do so, follow these steps:

1. Switch-User back to Domain Administrator.
2. Fire up Active Directory Users and Computers.
3. By default, the Group Policy Creator Owners group is located in the Users folder in the domain. Double-click the Group Policy Creator Owners group and add the HR-OU-Admins group and/or Frank Rizzo.

You will not be able to add the HR-OU-Admins group until the domain mode has been switched to at least Windows 2000 Native or Windows 2003 Functional level. Switch the domain by using Active Directory Domains and Trusts (or Active Directory Users and Computers). Switching the domain mode is a one-way operation, and disallows older operating systems as Domain Controllers. If you are not prepared to make the switch to Native mode, you'll only be able to add individual members—such as Frank Rizzo—and not a group.

In Chapter 2, you'll see an alternate way to allow users to create GPOs.

Creating and Linking Group Policy Objects at the OU Level

At the site level, we hid the Screen Saver option. At the domain level, we chose to get rid of the Sounds option in the Windows 7 Personalization page.

At the OU level, we have two jobs to do:

- Prevent users from changing the mouse pointers (a new Windows 7–only policy setting)
- Restore the Screen Saver option that was taken away at the site level

To create a GPO at the OU level, follow these steps:

1. Since you're on WIN7MANAGEMENT, log off as Administrator and log back on as Frank Rizzo (`frizzo@corp.com`).
2. Choose Start and type **`GPMC.MSC`** in the Start Search prompt.
3. Drill down until you reach the **Human Resources Users** OU, right-click it, and choose "Create a GPO in this domain, and Link it here" from the context menu to open the New GPO dialog box.
4. In the New GPO dialog box, type the name of your new GPO, say **Hide Mouse Pointers Option /Restore Screen Saver Option**. This will create a GPO in the Group Policy Objects container and link it to the **Human Resources Users** OU.
5. Right-click the Group Policy link and choose Edit from the context menu to open the Group Policy Management Editor.
6. To hide the Mouse Pointers Option in the Windows 7 Personalization page, drill down through User Configuration ➢ Policies ➢ Administrative Templates ➢ Control Panel ➢ Personalization and double-click the **Prevent Changing Mouse Pointers** policy setting. Change the setting from Not Configured to Enabled, and click OK.
7. To restore the Screen Saver setting for Windows 7 (or Screen Saver tab in Windows XP), double-click the **Prevent Changing Screen Saver** policy setting. Change the setting from Not Configured to Disabled, and click OK.
8. Close the Group Policy Management Editor to return to the GPMC.

By disabling the **Hide Screen Saver Tab** policy setting, you're reversing the Enable setting set at a higher level. See the sidebar "The Three Possible Settings: Not Configured, Enabled, and Disabled" later in this chapter.

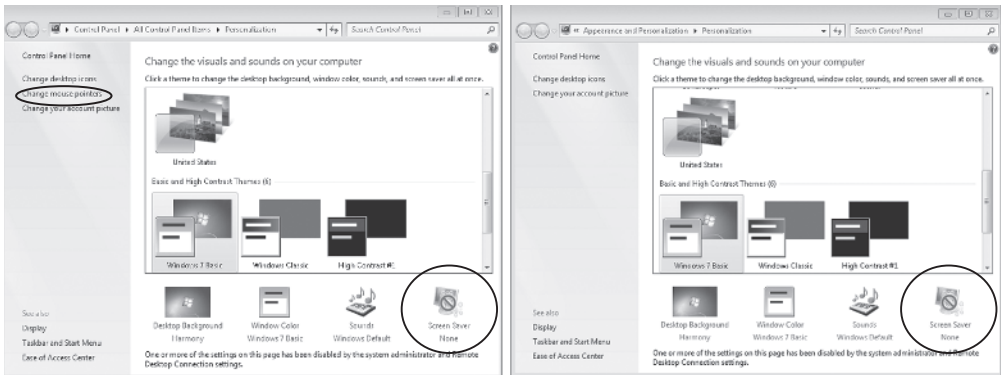
Verifying Your Changes at the OU Level

On your test WIN7 machine, log back on as Frank. Because Frank's account is in the OU, Frank is destined to get the Site, Domain and now the new OU GPOs with the policy settings.

On WIN7, right-click the Desktop and choose Personalize from the context menu to open the Display Properties dialog box. In Figure 1.25, you can see the before (left) and after (right) when the policy is applied. Look closely, and note that the "Change mouse pointers" option is removed and that that the Screen Saver option is no longer grayed out and is now available.

In the book we've highlighted the areas that are affected. But because the book is printed in black and white it could be hard to see that the "Screen Saver" selection is, indeed, no longer grayed out, and, yes, quite clickable again, as seen in the right in Figure 1.25.

FIGURE 1.25 On the left, we have Frank's Personalization page before the policy applies. You can see the Screen Saver icon is unavailable, and the ability to change mouse pointers is still present. On the right, we have Frank's Personalization page after the policy applies. The ability to manipulate screen savers has returned, but he is now prevented from changing mouse settings.



This test proves, once again, that even OU administrators are not automatically immune from policy settings. Chapter 2 explains how to change this behavior.

Group Policy Strategy: Should I Create More or Fewer GPOs?

At times, you'll want to lock down additional functions for a collection of users or computers. For example, you might want to specify that no users in the **Human Resources Users** OU can use the Control Panel.

At the **Human Resources Users** OU level, you've already set up a GPO that contained a policy setting to hide the mouse pointers option in the Windows 7 Personalization page. You can create a new GPO that affects the **Human Resources Users** OU, give it a descriptive name—say, **No One Can Use Control Panel**—and then drill down through User Configuration ➢ Policies ➢ Administrative Templates ➢ Control Panel and enable the policy setting **Prohibit Access to Control Panel**.

Or you could simply modify your existing GPO, named Hide Mouse Pointers Option/Restore Screen Saver Option, so that it contains additional policy settings. You can then rename your GPO to something that makes sense and encompasses the qualities of all the policy changes—say **"**Our Human Resources Users' Desktop Settings."

Here's the quandary: the former method (one policy setting per GPO) is certainly more descriptive and definitely easier to debug should things go awry. If you have only one policy set inside the GPO, you have a better handle on what each one is affecting. If something goes wrong, you can dive right into the GPO, track down the policy setting, and make the necessary changes, or you can disable the ornery GPO (as discussed in Chapter 2 in the section "Stopping Group Policy Objects from Applying.")

The second method (multiple policy settings per GPO) is a teeny-weeny bit faster for your computers and users at boot or logon time because each additional GPO takes some miniscule fraction of additional processing time. But if you stuff too many settings in an individual GPO, the time to debug should things go wrong goes up exponentially. Group Policy has so many nooks and crannies that it can be difficult to debug.

So, in a nutshell, if you have multiple GPOs at a particular level, you can do the following:

- Name each of them more descriptively.
- Debug them easily if things go wrong.
- Disable individually misbehaving GPOs.
- Associate that GPO more easily to a WMI filter (explored in Chapter 6).
- More easily delegate permissions to any specific GPO (explored in Chapter 2).

If you have fewer GPOs at a particular level, the following is the case:

- Logging on is slightly faster for the user (but really only slightly).
- Debugging is somewhat more difficult if things go wrong.
- You can disable individually misbehaving GPOs or links to misbehaving GPOs. (But if they contain many settings, you might be disabling more than you desire.)

So, how do you form a GPO strategy? There is no right or wrong answer; you need to decide what's best for you. Several options, however, can help you decide.

One middle-of-the-road strategy is to start with multiple GPOs and one lone policy setting in each. Once you are comfortable that they are individually working as expected, you can create another new GPO that contains the sum of the settings from, in this example, **Prevent Changing Mouse Pointers** and **Prohibit Access to Control Panel** and then delete (or disable) the old individual GPO.

Another middle-of-the-road strategy is to have a single GPO that contains only the policy settings required to perform a complete "wish." This way, if the wish goes sour, you can easily address it or disable it (or whack it) as needed.

Here's yet another strategy. Some Microsoft documentation recommends that you create GPOs so that they affect only the User half or the Computer half. You can then disable the unused portion of the GPO (either the Computer half or the User half). This allows you to group together policy settings affecting one node for ease of naming and debugging and allows for flexible troubleshooting. However, be careful here because after you disable half the GPO, there's no iconic notification (though there is a column labeled GPO Status that does show this). Troubleshooting can become harder if not performed perfectly and consistently. In all, I'm not a huge fan of disabling half the GPO.

Note that the policy changes we are making to the Windows 7 clients in the domain also take effect on Windows Server 2008 R2 machines.

Creating a New Group Policy Object Affecting Computers in an OU

For the sake of learning and working through the rest of the examples in this section, you'll create another GPO and link it to the **Human Resources Computers** OU. This GPO will autolaunch a very important application for anyone who uses these machines: `calc.exe`.

The setting we're about to play with also exists under the User node, but we'll experiment with the Computer node policy.

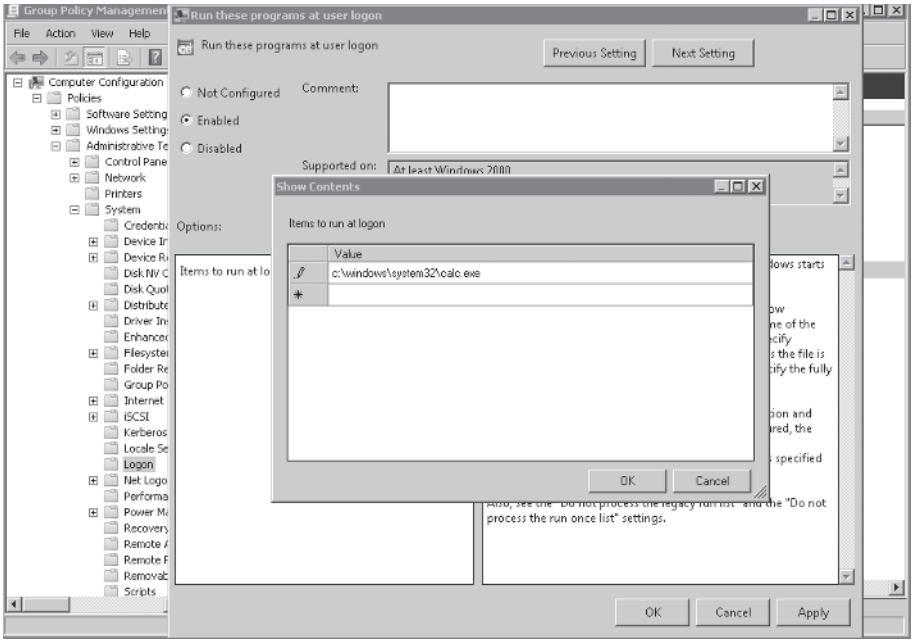
First, you'll need to create the new GPO and modify the settings. You'll then need to move some client machines into the **Human Resources Computers** OU in order to see your changes take effect.

To autolaunch `calc.exe` for anyone logging into a computer in the **Human Resources Computers** OU, follow these steps:

1. If you're not already logged in as Frank Rizzo, the **Human Resources** OU administrator, do so now on WIN7MANAGEMENT.
2. Choose Start and type **`GPMC.MSC`** in the Start Search prompt.
3. Drill down until you reach the **Human Resources Computers** OU, right-click it, and choose "Create a GPO in this domain, and Link it here" from the context menu.

4. Name the GPO something descriptive, such as **`Auto-Launch calc.exe`**.
5. Right-click the GPO, and choose Edit to open the Group Policy Management Editor.
6. We want to affect our client computers (not users), so we need to use the Computers node. To autolaunch `calc.exe`, drill down through Computer Configuration ➢ Policies ➢ Administrative Templates ➢ System ➢ Logon, and double-click **Run these programs at user logon**. Change the setting from Not Configured to Enabled.
7. Click the Show button, and the Show Contents dialog box appears. You'll see this policy setting has a little "table editor" associated with it. In the first "row," simply enter the full path to `calc.exe` as **`c:\windows\system32\calc.exe`** and click OK, as shown in Figure 1.26. Click OK to close the Show Contents dialog box, and click OK again to close the **Run these programs at user logon** policy setting.

FIGURE 1.26 When this policy setting is enabled and `calc.exe` is specified, all computers in this OU will launch `calc.exe` when a user logs in.



8. Close the Group Policy Management Editor to return the GPMC.

Be aware of occasional strange Microsoft verbiage when you need to enable a policy to *disable* a setting. Since Windows 2003, most policy settings have been renamed to "Prohibit <*whatever*>" to reflect the change from confusion to clarity.

Moving Computers into the Human Resources Computers OU

Since you just created a policy that will affect computers, you'll need to place a workstation or two inside the **Human Resources Computers** OU to see the results of your labor. You'll need to be logged on as Administrator on DC01 or WIN7MANAGEMENT to do this.

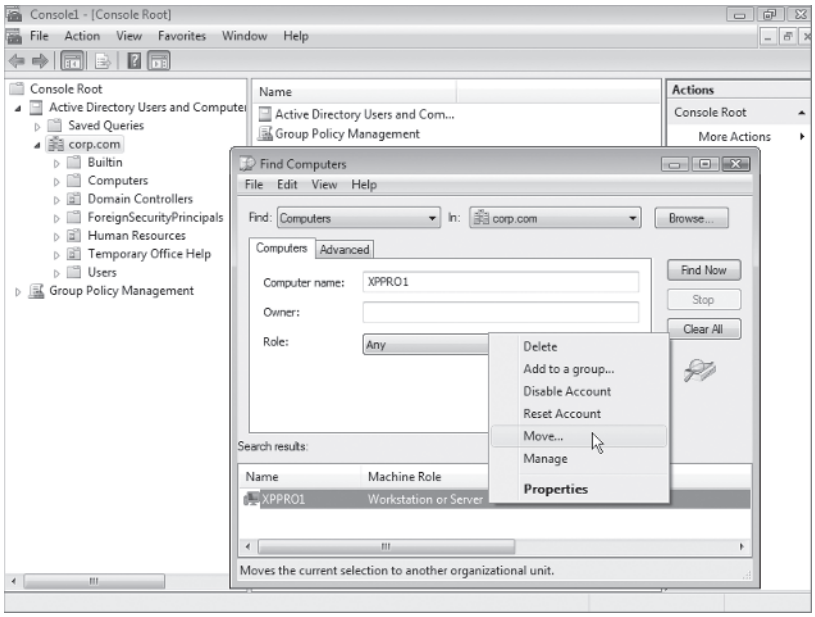
Quite often computers and users are relegated to separate OUs. That way, certain GPOs can be applied to certain computers but not others. For instance, isolating laptops, desktops, and servers is a common practice.

In this example, we're going to use the Find command in Active Directory Users and Computers to find a workstation named XPPRO1 and the Windows Vista workstation named WIN7 and move it into the **Human Resources Computers** OU.

To find and move computers into a specific OU, follow these steps:

1. In Active Directory Users and Computers, right-click the domain, and choose Find from the context menu to open the Find Users, Contacts, and Groups dialog box.
2. From the Find drop-down menu, select Computers. In the Name field, type **WIN7** to find the computer account of the same name. Once you've found it, right-click the account and choose Move from the context menu, as shown in Figure 1.27. Move the account to the **Human Resources Computers** OU.

FIGURE 1.27 Use the Find command to find computers in the domain, then right-click on the entry and select Move to move them.



Repeat these steps for XPPRO1 and all other computers that you want to move to the **Human Resources Computers** OU.

3. Now that you've moved WIN7 (and maybe also XPPRO1) into the new OU, be sure to reboot those client computers.

After you move the computer accounts into the **Human Resources Computers** OU, it's very important to reboot your client machines. As you'll see in Chapter 3, the computer does not recognize the change right away when computer accounts are moved between OUs.

As you can see in this example (and in the real world), a best practice is to separate users and computers into their own OUs and then link GPOs to those OUs. Indeed, underneath a parent OU structure, such as the **Human Resources** OU, you might have more OUs (that is, **Human Resources Laptops** OU, **Human Resources Servers** OU, and so on). This will give you the most flexibility in design between delegating control where it's needed and the balance of GPO design within OUs. Just remember that for GPOs to affect either a user or computer, that user or computer must be within the scope of the GPO—site, domain, or OU.

Verifying Your Cumulative Changes

At this point, you've set up three levels of Group Policy that accomplish multiple actions:

- At the site level, the "Hide Screen Saver Option" GPO is in force for users.
- At the domain level, the "Prohibit Changing Sounds" GPO is in force for users.
- In the **Human Resources Users** OU, the "Hide Mouse Pointers Option/Restore Screen Saver Option" GPO is in force for users.
- In the **Human Resources Computers** OU, the "Auto-Launch calc.exe" GPO is in force for computers.

At this point, take a minute to flip back to Figure 1.11 (the swimming pool illustration) to see where we're going here. To see the accumulation of your policy settings inside your GPOs, you'll need to log on as a user who is affected by the **Human Resources Users** OU and at a computer that is affected by the **Human Resources Computers** OU. Therefore, log on as Frank Rizzo at WIN7.

If you're using Windows 7, right-click the Desktop and choose Personalize. Note that the "Change mouse pointers" option is still missing from the previous exercise (and the Screen Saver entry is restored). And, when you logged in, did the computer GPO autolaunch Windows Calculator?

These tests prove that even OU administrators are not automatically immune from GPOs and the policy settings within. Under the hood, they are in the Authenticated Users security group. See Chapter 2 for information on how to modify this behavior.

The Three Possible Settings: Not Configured, Enabled, and Disabled

As you saw in Figure 1.2 earlier in this chapter, nearly all administrative template policy settings can be set as Not Configured, Enabled, or Disabled. These three settings have very different consequences, so it's important to understand how each works.

Not Configured The best way to think about Not Configured is to imagine that it really says, "Don't do anything" or even "Pass through." Why is this? Because if a policy setting is set to Not Configured, then it honors any previously set setting (or the operating system default).

Enabled When a specific policy setting is enabled, the policy will take effect. In the case of the **Prohibit Changing Sounds** policy setting, the effect is obvious. However, lots of policy settings, once enabled, have myriad possibilities *inside* the specific policy setting! (For a gander at one such policy setting, use the Group Policy Management Editor and drill down to User Configuration ➢ Policies ➢ Administrative Templates ➢ Windows Components ➢ Internet Explorer ➢ Toolbars and select the policy setting named **Configure Toolbar Buttons**.) So, as we can see, Enabled really means "Turn this policy setting on." Either it will then do what it says or there will be more options inside the policy setting that can be configured.

Disabled This setting leads a threefold life:

- Disabled usually means that if the same policy setting is enabled at a higher level; reverse its operation. For example, we chose to enable the **Prevent Changing Screen Saver** policy setting at the site level. If at a lower level (say, the domain or OU level), we chose to *disable* this policy setting, the Screen Saver option will pop back at the level at which we *disabled* this policy. You can think of Disabled (usually) as "reverse a policy setting coming from a higher level."
- Additionally, Disabled often forces the user to accept the administrator's will. That is, if a policy setting is disabled, some default behavior of the policy setting is enforced and the user cannot change it. To see an example policy setting like this, use the Group Policy Management Editor and drill down through User Configuration ➢ Policies ➢ Administrative Templates ➢ Start Menu and Taskbar and select the policy setting named **Force classic Start Menu** (a setting meant for XP and Vista, but not Windows 7). Once this policy setting is set to Disabled, the policy forces Windows XP users to use Start Menu in the XP task-based style (as opposed to the old Windows 2000 style). The point here is that the Disabled setting is a bit tricky to work with. You'll want to be sure that when you disable a policy setting, you're doing precisely what you intend.

- Disabled sometimes has a special and, typically, rare use. That is, something might already be hard-coded into the Registry to be "turned on" or work one way, and the only way to turn it off is to select Disabled. One such policy setting is the **Shutdown Event Tracker**. You disable the policy setting, which turns it off, because on servers since Windows 2003 it's already hard-coded on. In workstations Windows XP and later, it's already hard-coded off. Likewise, if you want to kill Windows XP or Windows 7's firewall, you need to set **Windows Firewall: Protect All Network Connections** to Disabled. (You can find that policy setting at Computer Configuration ➢ Policies ➢ Administrative Templates ➢ Network ➢ Network Connections ➢ Windows Firewall ➢ Domain Profile (and also Standard Profile) while editing GPOs on Windows XP/Service Pack 2 and above.) Again, that's because the firewall's defaults are hard-coded to on, and by disabling the policy setting, you're "reverting" the behavior back.

So, think of Not Configured as having neither Allow nor Deny being set. Enabled will turn it on and possibly have more functions. Disabled has multiple uses, and be sure to first read the Help text for each policy setting. Most times it's simply directly spelled out what Enabled and Disabled does for that particular setting. Lastly, test, test, test to make sure that once you've manipulated a policy setting, it's doing precisely what you had in mind.

Final Thoughts

The concepts here are valid regardless of what your domain is running. It doesn't matter if you have a pure or mixed Active Directory domain with various and sundry Domain Controller types. The point is that to make the best use of Group Policy, you'll need an Active Directory.

You'll also need a Windows 7 or Windows Server 2008 R2 management station to do your Group Policy work. Again, we talk more about why you need a Windows 7 management station in Chapters 3, 6, and elsewhere.

Remember, the GPMC is built into Windows Server 2008 R2, but it's not installed unless the machine is also a Domain Controller. The GPMC isn't built into Windows 7 and is only available through the downloadable RSAT tools.

The more you use and implement GPOs in your environment, the better you'll become at their basic use while at the same time avoiding pitfalls when it comes to using them. The following tips are scattered throughout the chapter but are repeated and emphasized here for quick reference, to help you along your Group Policy journey:

GPOs don't "live" at the site, domain, or OU level. GPOs "live" in Active Directory and are represented in the swimming pool of the domain called the Group Policy Objects container. To use a GPO, you need to link a GPO to a level in Active Directory that you want to affect: a site, a domain, or an OU.

GPOs apply locally and also to Active Directory sites, domains, and OUs. There is a local GPO that can be used with or without Active Directory. Everyone on that computer must embrace that local GPO. Then, Active Directory Group Policy Objects apply: site, domain, and then OU. Active Directory GPOs "trump" any local policy settings if set within the Local Group Policy. Active Directory is a hierarchy, and Group Policy takes advantage of that hierarchy.

Avoid using the site level to implement GPOs. Users can roam from site to site by jumping on different computers (or plugging their laptop into another site). When they do, they can be confused by the settings changing around them. Use GPOs linked to the site only to set up special sitewide security settings, such as IPsec or the Internet Explorer Proxy. Use the domain or OU levels when creating GPOs whenever possible.

Implement common settings high in the hierarchy when possible. The higher up in the hierarchy GPOs are implemented, the more users they affect. You want common settings to be set once, affecting everyone, instead of having to create additional GPOs performing the same functions at other lower levels, which will just clutter your view of Active Directory with the multiple copies of the same policy setting.

Implement unique settings low in the hierarchy. If a specific collection of users is unique, try to round them up into an OU and then apply Group Policy to them. This is much better than applying the settings high in the hierarchy and using Group Policy filtering later.

Use more GPOs at any level to make things easier. When creating a new wish, isolate it by creating a new GPO. This will enable easy revocation by unlinking it should something go awry.

Strike a balance between having too many and too few GPOs. There is a middle ground between having one policy setting within a single GPO and having a bajillion policy settings contained within a single GPO. At the end of your design, the goal is to have meaningfully named GPOs that reflect the "wish" you want to accomplish. If you should choose to end that wish, you can easily disable or delete it.

As you go on your Group Policy journey… Don't go at it alone. There are some nice third-party independent resources to help you on your way. I run `www.GPanswers.com`, which has oodles of resources, downloads, a community forum, downloadable eChapters, video tutorials, links to third-party software, and my in-person and online versions of my hands-on training seminars. Think of it as your secret Group Policy resource.

My pal (and technical editor for a previous edition) Darren Mar-Elia runs `www.GPOguy.com`.

My pal (and technical editor for a previous edition) Jakob Heidelberg has a lot of great articles (mostly on Group Policy topics) at `http://www.windowsecurity.com/Jakob_H_Heidelberg`, (`http://tinyurl.com/ypar82`).