Abstract
This technical brief reviews three tasks critical to Active Directory (AD) maintenance:
• Monitoring AD replication
• Recovering AD objects
• Cleaning AD’s database
As we’ll see, accomplishing any of these tasks using native tools can be a difficult, manual task. But Active Administrator from Dell makes it easy to proactively maintain your AD environment.

Introduction
Regular maintenance is as important for AD as it is for your car. Maintaining Active Directory can be a lot like maintaining a vehicle. Simple maintenance is common knowledge: You wash your car every so often, put air in the tires when they get low, and add the occasional quart of oil. But that’s probably the extent of your mechanical prowess. You could maybe change the oil in a pinch, or even do a bit of minor electrical work, but anything more requires time and expertise you might not have.

For most IT pros, maintaining AD happens in much the same way. You’re skilled in account management, you understand the security basics, and you can occasionally create the errant AD Sites and Services object. But what about the activities going on under AD’s hood? How’s your replication? Any DNS issues? Are AD backups easily recoverable? Is the database free from orphaned objects?

Too often the answer is, “I don’t know, and I generally don’t care—as long as everything’s working.” That attitude might work for your car, but it can be incredibly short-sighted with Active Directory. Neglect AD’s maintenance tasks, and your entire IT environment is only asking to get left on the side of the road.

Avoid costly AD issues with three key maintenance tasks. Just as keeping your car in good condition helps you avoid unnecessary repair costs, AD preventative maintenance is critical to avoiding costly AD issues.
Just as keeping your car in good condition helps you avoid unnecessary repair costs, AD preventative maintenance is critical to avoiding costly AD issues.

Save yourself the headaches with a few key tasks:
- Monitoring AD replication
- Recovering AD objects
- Cleaning AD’s database

These tasks are subject to the old adage "An ounce of prevention is worth a pound of cure." By doing just an "ounce of prevention," you’ll avoid needing to use a "pound of cure" to fix the big AD maintenance problems down the road.

A variety of third-party solutions exist that greatly automate these maintenance tasks, but Windows offers tools as well. Those tools might seem free at first blush, but as you’ll see, their ounce of up-front cost can sometimes create a pound of downstream effort.

**Task #1: Monitoring AD replication**
Your AD’s availability is a moving target. Every change replicates to every domain controller (DC), which requires assurance those changes have propagated correctly. This behavior makes monitoring AD replication a mission-critical activity. So, here’s the question: How healthy is your AD replication?

For such a critical activity, finding the answer isn’t a trivial task. Indeed, there are plenty of third-party solutions that deliver a heads-up green light/red light, but what answers does Windows itself deliver?

**Pound of cure: Repadmin and PowerShell**
Windows offers a few tools to keep replication in check. For example, Microsoft has provided the repadmin command-line tool for quite some time. This tool does everything from replicating a single object to seeing the current replication state between partnered DCs. It can even force replication when you need an instant verification that objects are correctly moving around.

But like some command-line tools, repadmin comes with almost too many options. It also delivers an abundance of resulting data, so getting actionable information out of it is quite a chore.

For the script-savvy, the PowerShell cmdlet Get-ADReplicationFailure (Figure 1) performs similar functions to repadmin. It can query and return AD object and replication information. Its results are arguably more useful since they’re delivered as objects rather than simple text, but this is still no wholesale monitoring solution. You’ll still need to parse through cmdlet output or build scripts to keep an eye out for replication issues.

Both of these native solutions also suffer from a key omission: They don’t tell you why the replication is failing. These tools provide data, but you’ll need to research...
error codes to convert their data into knowledge of what you should fix. Finally, while these tools do assist with checking the status of AD replication, they don’t provide an easy and automated way to do it. And don’t forget all that free time you don’t have to set this up in the first place.

Ounce of prevention: Active Administrator
Active Administrator helps you understand on an ongoing basis whether replication between servers is possible. It finds replication errors throughout servers across forests and sites and displays them on a graphical interface (see Figure 2). Administrators can then dig into each error to ascertain more detail.

Task #2: Recovering AD objects
Backing up Active Directory can be an insidiously incomplete activity. Your backups are probably running, but can you easily restore their data? How much time will it take to resurrect an object or a group of objects? And, most importantly, when was the last time you actually tested a restore—on actual hardware?

While all of AD needs to be backed up, a focus on three basic components is central to a smart recoverability strategy:
- AD objects
- Object permissions
- Group Policy Objects (GPOs)
All three of these components are very likely on your backup tapes (or disks) today. But the question these days isn’t so much about the successful capture of their data. It is the capture of that data in an easily restorable fashion that’s become the mission-critical requirement.

Objects, their permissions and GPOs are all considered AD components, but traditional backup solutions store them on tapes or disks in very different ways. Backing them up might be easy, but recovering them quickly might not.

Think for a minute about each of these in your environment. Does an object restore in your environment require Directory Services Restore Mode and DC downtime? What exact steps does your recovery plan need to resurrect a lost GPO? How might you reconstruct an object’s permissions without touching the object itself?

Figure 2. Active Administrator tests server replication and shows errors in an easy-to-read interface.
Accomplishing these tasks is inordinately tedious with Windows alone, but it can be done. Third-party alternatives have already invested the effort in creating automation. Which fits your need?

Pound of cure: DSRM, PowerShell and cmdlets

Virtually every IT pro has a backup solution that keeps copies of AD data. The three kinds of data can require three different approaches:

- AD objects in Windows domains that haven’t been updated to today’s domain and forest functional levels can only be recovered via Directory Services Restore Mode. DSRM is a special mode for domain controllers that requires DC downtime, non-authoritative restores, and subsequent adjustments to the objects needing recovery. Windows Server 2008 R2 and Windows Server 2012 improve AD object recovery slightly with the introduction of the AD Recycle Bin, which has a reasonable GUI for locating and recovering AD objects. But the recovery process in Windows Server 2008 R2 requires some complex PowerShell command-line action. Search for “Get-ADObject | Restore-ADObject” for all the gory details.

- AD object permissions operate much like NTFS permissions, but with a devious twist: screw them up, and you can screw up AD entirely. Complicating matters further is the complex nature of AD object permissions. Knowing what the correct permission should be isn’t always obvious. Microsoft offers the dsacls command and the set-acl cmdlet to reset object permissions. They’ll get the job done, but at what cost?

- Group Policy Objects can be restored via the Group Policy Management Console—but only if they’ve been backed up using that same console. Console-based GPO backups, depending on your version of Windows, can also be cumbersome to manage. Console backups can be a manual activity, and some GPO data (WMI filters, IPSec policies, and GPO links) sometimes won’t get captured. PowerShell offers process automation with the Restore-GPO cmdlet, with all the usual caveats that accompany home-built scripts.

Ounce of prevention: Active Administrator

To mitigate the risk of Active Directory mishaps, Active Administrator provides both backup and recovery features. When an object needs to be restored, you can select the object from a list and restore it, either with all the attributes it possessed when it was backed up, or only the attributes you select. In the case of a container object, you have the option of restoring either all objects it contains or all objects it contains of a particular type.

You can preview the object before it is restored, as well as compare the attributes of the selected object in the archive with those of the same object in Active Directory. Active Administrator
also provides the ability to recover entire GPOs, or simply roll back a GPO to a previous edit.

Restoring objects is easy with Active Administrator: Simply select the object from a list and restore it, either with all the attributes it possessed when it was backed up, or only the attributes you select.

**Task #3: Cleaning AD’s database**

Take a quick peek at your Active Directory database. Now answer the question, “Which objects are still actually in use?”

Which computer objects correspond with actual computers still connecting to Active Directory? Which user objects are used by employees that are still receiving paychecks? Active Directory’s database might be a beautiful solution for authentication and resource authorization, but it’s always suffered a missing piece: AD objects never get scavenged.

The Windows 98 computer you joined to the domain in 1998 might still be there. Bob from Accounting might have left a decade ago, but his user account never went anywhere. An unscavenged Active Directory database is a notorious warehouse of no-longer-active objects, orphans standing side by side with others still in use.

And every single one of those objects is a security risk just waiting to happen. You can install a third-party solution that exposes those objects in an easy-to-read dashboard. Such a solution might deliver a heads-up display of each object’s age, along with all the automations necessary to remove those you’ve been informed are no longer relevant. Or, you can build something yourself out of the tools Windows provides. Let’s take a look at what those tools are.

**Pound of cure: PowerShell & command-line tools**

Microsoft supplies two tools for finding those nasty inactive objects. The first, dsquery, is a command-line tool that is generally executed on a domain controller itself. The dsquery tool has been around for a number of OS versions, and is even included with Windows Server 2012.

The dsquery tool is equipped with a variety of features. Its information can be piped into other tools like dsrm to create an automatic all-or-nothing removal solution. Try entering this at a domain controller command prompt:

```
dquery computer -inactive 30 | dsrm
```

On second thought, don’t. The code snippet above will query Active Directory for objects that have been inactive for 30 days or more. It will then immediately remove any objects it finds. While wielding such power might seem great, there’s plenty of accompanying responsibility.

Using PowerShell to accomplish this task offers greater flexibility, but at the cost of greater potential for error in constructing
the command you need. The two-part PowerShell command shown below first creates a variable called $ThirtyDays to define the days of inactivity, and then uses that variable with Get-ADUser and ultimately Remove-ADUser to delete any objects it finds. Figure 4 shows a result, without the actual object removal.

$ThirtyDays = (get-date).adddays(-30); Get-ADUser -filter {(lastlogondate -notlike "*" -OR lastlogondate -le $ThirtyDays) -AND (passwordlastset -le $ThirtyDays) -AND (enabled -eq $True)} -Properties lastlogondate, passwordlastset | Remove-ADObject

As with the previous command, be extra careful here. PowerShell commands like these are indiscriminate in the work they do. A misplaced keystroke might mean the loss of all kinds of AD objects (and a quick backtrack to this paper’s Task #2). More important than these commands, though, is the reality of how a Windows environment operates. Rare is the Active Directory domain where every object can be removed at an exact number of days. More useful than mass deletion is fine-point control that can locate and remove inactive objects at the appropriate time. For that, we must look to a third-party solution.

**Ounce of prevention: Active Administrator**

Active Administrator simplifies the maintenance of accounts, sites and trusts. You can easily monitor account activity, purge inactive user and computer accounts, send password reminders and more. Active Administrator’s activity dashboard (Figure 5) gives you a high-level view of activity across multiple domains, and its ability to purge dormant users and computers saves you manual effort.

![Figure 5. Monitor activity and purge dormant accounts with Active Administrator.](image)

PowerShell offers greater flexibility in locating inactive objects—but a misplaced keystroke in a command could cause the loss of all kinds of AD objects.
Maintaining AD: Time for the weigh-in

Maintaining AD requires everyday effort. Some aspects are easy, even with Windows’ native tools; other aspects might seem easy but are actually hard. Figuring out which tasks you can manage unaided and which are best left to third-party solutions like Active Administrator requires weighing your options carefully. Just as you do with your automobile, seek balance with your AD maintenance activities and leave the heavy lifting to the experienced professionals.

Conclusion

Implementing an ounce of prevention is the clear choice over a pound of cure. Native tools get you only so far when it comes to Active Directory maintenance. On the other hand, Active Administrator delivers the complete automation and flexible granularity you need to best enhance the performance of Active Directory.

The smart move? Save yourself maintenance struggles down the road. Take a hard look at the entire solution your Active Directory needs for ongoing maintenance, and remember that an ounce of prevention is worth a pound of cure.
For More Information
© 2013 Dell, Inc. ALL RIGHTS RESERVED. This document contains proprietary information protected by copyright. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording for any purpose without the written permission of Dell, Inc. (“Dell”).

Dell, Dell Software, the Dell Software logo and products—as identified in this document—are registered trademarks of Dell Inc. in the U.S.A. and/or other countries. All other trademarks and registered trademarks are property of their respective owners.

The information in this document is provided in connection with Dell products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Dell products. EXCEPT AS SET FORTH IN DELL’S TERMS AND CONDITIONS AS SPECIFIED IN THE LICENSE AGREEMENT FOR THIS PRODUCT, DELL ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL DELL BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF DELL HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Dell makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. Dell does not make any commitment to update the information contained in this document.

About Dell
Dell Inc. (NASDAQ: DELL) listens to customers and delivers worldwide innovative technology, business solutions and services they trust and value. For more information, visit www.dell.com.

If you have any questions regarding your potential use of this material, contact:

Dell Software
5 Polaris Way
Aliso Viejo, CA 92656
www.dell.com
Refer to our Web site for regional and international office information.