



Contents

Introduction..... 3

Windows Administration..... 4

20,000-Foot View of Active Directory and DRA 7

Distributed, Automated Administration and Policy Enforcement 9

Creating Your Own Views 13

Unlocking the Power of Windows 2000 and Windows NT 15

Managing Windows 2000 & Windows NT 4.0 with Ease

White Paper

October 2000

NetIQ Directory and Resource Administrator (DRA) is an advanced, distributed policy-based directory management application. Directory and Resource Administrator can unlock the power of Windows NT 4 and Windows 2000, providing you the ability to:

- Improve security.
- Assure data integrity and prevention of directory pollution.
- Unify and seamlessly administer Windows NT 4 and Windows 2000.
- Automate provisioning.
- Increase administration flexibility.
- Employ extensive, rich reporting and auditing.
- Reduce administrative costs.

Directory and Resource Administrator effectively and seamlessly manages Microsoft Windows 2000 and Windows NT 4 systems. Built from the ground up to exploit Windows 2000 technologies, Directory and Resource Administrator can drastically reduce administrative costs while protecting your existing technology investments.

Directory and Resource Administrator gives you the administrative power and flexibility to extend native Active Directory delegation across geographic, operating system and organizational unit (OU) hierarchical boundaries. Using a rules-based architecture, Directory and Resource Administrator automatically enforces and propagates policies across both Windows 2000 and Windows NT 4 systems—ensuring data integrity enforcement and increasing security.

First Edition

NetIQ Corporation provides this document “as is” without warranty of any kind, either express or implied, including, but not limited to, the implied warranties of merchantability or fitness for a particular purpose. Some states do not allow disclaimers of express or implied warranties in certain transactions; therefore, this statement may not apply to you.

This document and the software described in this document are furnished under a license agreement or a non-disclosure agreement and may be used only in accordance with the terms of the agreement. This document may not be lent, sold, or given away without the written permission of NetIQ Corporation. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, or otherwise, without the prior written consent of NetIQ Corporation. Companies, names, and data used in this document are fictitious unless otherwise noted.

This document may include technical inaccuracies or typographical errors. Changes are periodically made to the information herein. These changes may be incorporated in new editions of the document. NetIQ Corporation may make improvements in and/or changes to the products described in this document at any time.

Copyright © 1995-2000 NetIQ Corporation, all rights reserved.

U.S. Government Restricted Rights: Use, duplication, or disclosure by the Government is subject to the restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause of the DFARs 252.227-7013 and FAR 52.227-29(c) and any successor rules or regulations.

AppManager, the AppManager logo, AppAnalyzer, Knowledge Scripts, Work Smarter, NetIQ Partner Network, the NetIQ Partner Network logo, Chariot, Pegasus, Qcheck, OnePoint, the OnePoint logo, OnePoint Directory Administrator, OnePoint Resource Administrator, OnePoint Exchange Administrator, OnePoint Domain Migration Administrator, OnePoint Operations Manager, OnePoint File Administrator, OnePoint Event Manager, Enterprise Administrator, Knowledge Pack, ActiveKnowledge, ActiveAgent, ActiveEngine, Mission Critical Software, the Mission Critical Software logo, Ganymede, Ganymede Software, the Ganymede logo, NetIQ, and the NetIQ logo are trademarks or registered trademarks of NetIQ Corporation or its subsidiaries in the United States and other jurisdictions. All other company and product names mentioned are used only for identification purposes and may be trademarks or registered trademarks of their respective companies.

Introduction

This white paper describes how Directory and Resource Administrator (DRA) from NetIQ can organize your views of directories, enforce automated policies and maintain the security and integrity of your Windows 2000 and Windows NT systems.

Directory and Resource Administrator is built from the ground up to give businesses the power to simply and securely manage Windows 2000 and Windows NT 4 systems. Based on the advanced architecture of Microsoft Windows 2000 and Component Object Model (COM), Directory and Resource Administrator version 6.x delivers industry-leading systems management capabilities.

Directory and Resource Administrator three-tiered architecture can manage data from several data sources, including Windows NT SAM, Exchange Directory, Active Directory and computing resources. Directory and Resource Administrator manages the interaction between users, processes and the database—controlling what actions can be taken and ensuring correct workflow and data integrity.

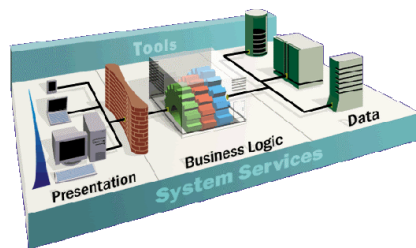


Figure 1 - Three-tiered Architecture

No matter where you are in your Windows evolution—Windows NT 4.0, Windows 2000, or a mixed environment—enterprises need DRA now for the following reasons:

- **Improved security** - DRA can reduce the number of user accounts with Admin permissions and can safely distribute administration and data content ownership throughout your organization.
- **Data integrity and prevention of directory pollution** - Since automation and policies are seamlessly integrated with directory updates, DRA makes it easy to ensure the integrity, consistency and completeness of Active Directory content.
- **Unified and seamless Windows NT 4 and Windows 2000 administration** - DRA improves productivity by giving you a single easy-to-use administrative tool vs. needing to implement and learn to operate multiple tools for multiple operating systems.
- **Automated provisioning** - DRA was built to help users automate administrative tasks and apply policies to directory updates. These updates can extend beyond the directory itself to other applications and databases, making the task of provisioning easy. Many of the Windows-related provisioning tasks are provided out-of-the-box.
- **Increased administration flexibility** - DRA's ActiveView technology seamlessly joins and administers data from multiples sources (e.g. Windows NT SAM, Exchange Directory, Active Directory, computing resources, etc.) and eliminates the problems associated with having to administer data bound within a rigid, hard-to-change hierarchical structure.
- **Extensive, rich reporting and auditing** - DRA addresses the need in Windows NT and Windows 2000 to log and audit all administrative actions and provides extensive and rich reporting.
- **Reduced administrative costs** - DRA provides significant out-of-the-box policy and automation, letting you eliminate many administrative steps.

Windows Administration

Like most enterprises, you've probably invested heavily in the Windows platforms so you can keep up with the ever-changing demands of the modern business world. It's also likely you've come up against difficulties, such as hiring and retraining skilled staff, managing the day-to-day operations and keeping up with the latest technology.

This section will look at some of the challenges you encounter and how Directory and Resource Administrator can help.

Windows NT 4.0

Windows NT 4 has been widely recognized as a valuable computing platform. Even with the arrival of the next generation operating system in the form of Windows 2000, it's a safe bet that there will be a lot of Windows NT 4 around for years to come.

Still, Windows NT 4 comes with its share of headaches, including:

- Delegation
 - If you delegate with native tools, you end up with too many administration user accounts and no control.
 - If you don't, you end up with overloaded Administrators.
- Limited reporting and auditing.
- There is no easy way to find out who can do what to whom.
- Native tools require the use of multiple user interfaces.
- No automation for repetitive associated tasks (I.e., create mailbox, home directory, quota, etc. every time a user account is created),
- No way to unify and integrate the administration of several different data sources.
- No way to enforce standards and policies for data consistency.

Let's take a closer look at a couple of these topics.

Delegation

Windows NT 4.0 provides some delegation capabilities. You can assign individuals to be Account Operators, Administrators, Printer Operators, and Server Operators. Each of these groups grants the person some level of administrative power over the domain. However, most organizations are uncomfortable with the limited granularity and broad scope of the power. For example, if a person is an Account Operator, he may change any of the properties for user accounts and groups. When delegating authority, many organizations require more granularity because they want to restrict which properties a person can change. A common example is giving the Help Desk the ability to reset the password for a user account.

The scope of the power is an issue because it too represents an all or nothing proposition. A person is an Account Operator or Printer Operator for everything in the domain, or for nothing—a result of working in a flat name space. Using the native Windows NT administration tools, you cannot partition authority without creating additional domains and the accompanying administrative, hardware, and software overhead. This presents enterprises with a difficult choice. Should they break up their domains to allow partitioning of authority, or should they live with the security exposure of granting a person more power than his tasks require?

Directory and Resource Administrator enables you to safely delegate permissions using its powers model, which is exposed through its Active Views technology, described in more detail below.

Reporting and Auditing

Businesses run on information. Making sure the right people have access to the right information—and ensuring the wrong people don't—is critical in running today's information systems.

Directory and Resource Administrator provides explicit log entries detailing actions performed, success or failure of the actions and/or which attributes were changed. For example, if Madeline resets the password for Michael's account, the Directory and Resource Administrator log entry will say exactly that, in plain English. This feature provides the easy-to-follow, easy-to-understand electronic trail you need to secure your organization's IT assets. Together with its reporting features, the Directory and Resource Administrator log drastically reduces the time required for audits.

Reliable and secure systems require active monitoring, and monitoring requires reporting. Windows 2000 and Windows NT store much of the data needed for active monitoring, but Directory and Resource Administrator provides the means to easily extract that data. Directory and Resource Administrator comes with more than 40 pre-programmed reports that cover almost every aspect of Directory and Resource Administrator user account, group, Microsoft Exchange mailbox and resource administration. Many of these reports are parameter driven, enabling you to control report content.

In providing the ability to precisely scope the reporting result set, Directory and Resource Administrator enables data mining for systems state, change and management data. This capability is critical to secure operations—because reporting all enterprise information in huge quantities is often worse than providing no reporting at all.

DRA and Windows NT 4

Directory and Resource Administrator is part of a comprehensive administration suite and broader systems management offering from the leading NT/Windows 2000 systems management vendor. We were the first to build and sell a product of this kind (originally called Enterprise Administrator). Because we have been in this market from the onset, we have more customers, more functionality, and more awards than the latecomers. Customers were able to see ROI in days--not years—and were able to have their Windows NT 4 operate much like Windows 2000.

Enterprises need Directory and Resource Administrator for NT4.0 for the following reasons:

- Eliminate Administration security risks:
 - Reduce the number of Administrator user accounts
 - Granular delegation of permissions
 - Every action is logged
 - Extensive, rich reporting and auditing
- Save administration time:
 - Distributed administration based on your delegation model
 - Automation and policy enforcement for data integrity and consistency
 - Unified administration of users, groups, mailboxes, resources, and other data through one consistent UI
- Increase administration flexibility:
 - Relational-like ActiveViews seamlessly 'join' and administer data from multiple sources, e.g., SAM, Exchange, Active Directory, etc.
 - § Can cross domains
 - § Can have policy and automation assigned to ActiveViews
- Simplify and promote distributed administration:
 - Rules-based approach for easy administration dynamically adapts to the latest data
 - Push out administration of common tasks to your enterprise with confidence and security
- Investment carries forward as you migrate to Windows 2000

Windows 2000

Windows 2000 represents a leap in operating systems technologies. Based on Windows NT technologies, it has a solid foundation for reliability, scalability, and application hosting. The question isn't IF you will be going to Windows 2000, but WHEN.

All systems have challenges for administration— Directory and Resource Administrator seeks to add value to Windows 2000 by providing additional management capabilities:

- Adds reporting and strengthens auditing.

- Gives protection against directory pollution
- Strengthens security administration
- Provides automation

Directory and Resource Administrator can be used to protect against incorrect data, or “pollution,” from getting into Active Directory. You want to use automation in your favor to ensure correct information is put into your Active Directory. Delegating content updates without a means of automatically policing the content can be a big problem.

To take advantage of ‘stored procedures’, you need a management application, such as Directory and Resource Administrator, which sits between the user and the Active Directory to do policy enforcement.

Administration of the Active Directory generally involves having to remember to update other data sources from the same transaction. Directory and Resource Administrator can work with a variety of data sources as part of tasks, such as creating a user. Additionally, Directory and Resource Administrator logs all of its actions so you know who did what to whom.

Active Directory is designed to represent information in a hierarchical view. Once you have decided on your directory structure, it is not something you change often. ActiveViews, described in more detail below, eliminate the problems and politics of being bound to a rigid, unchangeable hierarchical structure. They facilitate change (i.e. acquisitions) and can include data across directory boundaries.

It’s useful to think of examples to illustrate why Directory and Resource Administrator is so useful in the Windows 2000 environment. Consider how you would do the following if you didn’t have Directory and Resource Administrator :

- Scenarios where Access Control List (ACL) management isn’t enough:
 - Using the native tools on a two-domain forest with a total of 15,000 objects, identify all users whose passwords Bob can update.
 - Show me all groups whose memberships were changed by MADDOM\Fred yesterday.
 - Apply a delegation policy that spans different domains in the same forest.
 - In an Organizational Unit (OU) containing 1,000 users, show administrative assistant Sam only the four accounts whose password he can change and filter out all the rest from his view.
- Content control: where ACL managers cannot go:
 - Allow a user to create only groups that begin with a specified prefix such as “ATL” for Atlanta-based groups or “CHI” for Chicago-based groups.
 - Enforce an Employee ID attribute that must consist of nine numeric digits.
 - Validate the state field for an employee based on the city entry.
 - Allow a user to see only objects over which he or she has write power.
- Automating infrastructure changes based on directory changes:
 - Whenever a user account is created, automatically create and assign permissions for the user’s home directory, home share and NTFS5 quota.
 - Automatically generate Internet proxy addresses according to a specified format (e.g. firstname.lastname@company.com).
 - Automatically assign a logon script based on the user’s OU membership.
 - Automatically send an e-mail notification to the security office whenever the membership of Domain Admins changes.

DRA and Windows 2000

Built from the ground up to exploit Windows 2000 technologies, Directory and Resource Administrator can drastically reduce administrative costs while protecting your existing technology investments. Use Directory and Resource Administrator with Windows 2000 to:

- Prevent Active Directory pollution
 - Policies and automation can be seamlessly integrated with Directory updates:
 - § Ensures data integrity of the Active Directory – protect the most important database in your company.
 - § Enables data content ownership to be distributed safely throughout your organization.

- Simplify and reduce administrative volume
 - Automated administrative transactions:
 - § Eliminates the multitude of routine administrative tasks associated with User, Group and resource management.
 - § Ensure consistency and completeness.
- Integrate Active Directory with the 'outside world'
 - Active Directory updates/modifications can be unified through policies and automation with other databases, directories and critical administrative data
- Employ auditing and reporting.
- Implement flexible directory structure.

Managing the Transition

Directory and Resource Administrator also plays a critical role during the transition from NT 4.0 to Windows 2000 as the transition itself comes with some problems of its own. Since the skills required for the two environments are different, you want to minimize the effects of dealing with multiple operating systems.

With Directory and Resource Administrator, all policies, automation, administrative ActiveViews, etc. defined in NT 4.0 carry forward to Windows 2000. Directory and Resource Administrator can also manage both NT 4.0 and Windows 2000 as if it were a single seamless Windows environment—same user interface and same toolset.

Put another way: two environments, two sets of challenges, one simple solution.

20,000-Foot View of Active Directory and DRA

Active Directory uses objects to represent network resources, such as users, groups, computers, devices and applications. Objects are grouped together in containers called Organizational Units (OUs), which can then be arranged into a hierarchical structure or tree.

The use of objects to represent network resources in Active Directory simplifies resource management. For example, if user John has four applications that require logons, without Active Directory he must treat each application separately because each application maintains its own security and user identity information, requiring John to either maintain separate passwords or try to keep the passwords between the systems synchronized. However, if the applications use Active Directory for authenticating John, only one account must be maintained for all the applications.

Active Directory as Database

Active Directory is a special purpose, distributed *database*. You are probably familiar with database concepts such as access control and data integrity. You are also likely to know that applications typically provide the business process workflow and interactions with a database.

For example, if a payroll administrator wants to update an employee's salary field, he or she does not work directly with the database, such as entering SELECT and JOIN statements. Instead, the payroll administrator uses a payroll application or human resources application, which in turn communicates with the database. This same type of interaction applies between the Directory and Resource Administrator application and Active Directory.

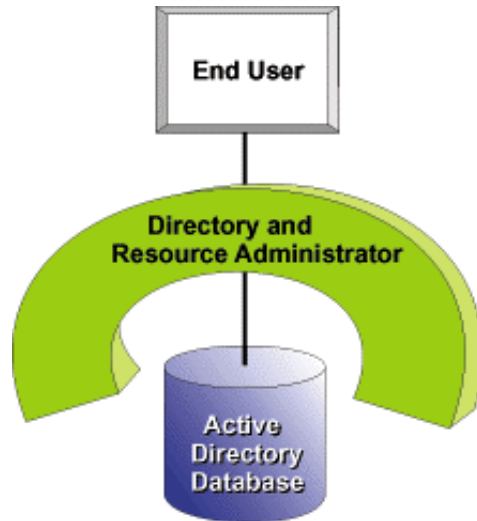


Figure 2 - Management Application Layer

The Directory and Resource Administrator directory management application is situated between administrators and Active Directory and greatly expands the scope of activities administrators can perform beyond the capabilities of Active Directory, such as:

- Unifying administrative workflow
- Enforcing corporate policy
- Protecting data integrity

Active Directory Benefits

Network administrators have long been frustrated by the effort required to maintain correct information about people and resources on their networks. A simple task, such as changing an employee's telephone number, often turns into a daunting chore because of the difficulty of finding all instances of the telephone number on the network. One solution to this problem is for a directory (Active Directory) to serve as a clearinghouse for identity information. With Active Directory for Windows 2000, application developers can leave the maintenance of this sort of information to the directory.

Because Active Directory is available, secure and reliable, application developers can ease their burden by concentrating on developing their programs' primary functions rather than duplicating things Active Directory provides. And administrators benefit because Active Directory becomes the single authoritative source. Active Directory simplifies complex environments, making information available to those who have the proper access rights.

Active Directory provides benefits, such as:

- Secure, centralized place to hold information about users and objects.
- Reliable, available data store.
- Foundation for emerging business applications that rely on identity information (for example, e-commerce businesses).

Security: What s the Worry?

You have taken great pains to make sure your systems are secured and reliable, so why do you need Directory and Resource Administrator for security management? The answer lies in the distinction between *access control* and *content management*.

Access control means the system can *authenticate* identities. Content management means the system can *enforce* policies regarding the information being placed in it.

To draw an analogy: At U.S. airports, you are asked to present some form of identification before you can get a boarding pass. You are *authenticated* when you show your driver's license, passport or other valid identification. After you receive your boarding pass, you must then pass through a security station where you pass your carry-on baggage through an X-ray machine, you go through a metal detector, and where security personnel—not you the passenger—*enforce* security policies.

Airports and airlines have policies in place to ensure the integrity and security of the aircraft, but the policies must be enforced to be of any use. Similarly, to ensure the integrity and security of data entered into your critical directories, you must enforce your business policies.

Directory and Resource Administrator provides proactive methods to safeguard against directory and content pollution—going way beyond simply setting ACLs:

- Directory and Resource Administrator supports a complete distributed administration and transaction model with active script triggers and commit/rollback.
- Directory and Resource Administrator takes advantage of dynamic ActiveViews that span OUs, domains, trees and forests, in both Windows 2000 and Windows NT.
- Virtual Property Objects allow extra-directory attributes to be appended without extending the schema.
- Policies and automation can manage mixed Windows 2000 and Windows NT environments.
- Directory and Resource Administrator exploits the capabilities of Windows 2000 and Active Directory.

Distributed, Automated Administration and Policy Enforcement

Wouldn't it be great to have your business policies encoded and enforced? Imagine if a user didn't have to remember every department code before filling in a field or have the system tell them when the telephone or fax number they just entered is not correct. Don't just *print* your business policies and hope they will be followed—*automate* them with Directory and Resource Administrator .

At a recent industry meeting, a presenter¹ suggested that companies have their IT programmers learn Perl so that they could automate processes and fix bad data. Of course, automating business processes is a good idea, but writing a collection of Perl scripts is probably not the most efficient way for your organization to accomplish this. Directory and Resource Administrator provides a standard, cohesive approach to automating your business processes.

In organizations that use Microsoft Windows (and who doesn't?), Windows scripting languages, such as VBscript are typically preferable over Perl as the means to encode business policies. Directory and Resource Administrator gives you the flexibility to run any scripting language supported by Windows 2000 and Windows NT. You can also run executable programs from Directory and Resource Administrator triggers.

Although directories allow access only to authorized users, security can be weakened if users are given the wrong level of access. Directory and Resource Administrator lets you distribute administrative rights in a granular way. For example, with Directory and Resource Administrator , if you have Help Desk personnel who have rights to reset passwords, you don't have to give them administrator privileges to do that single task. In addition to the Microsoft Management Console (MMC) interface, Directory and Resource Administrator has a user-friendly Web interface for simple tasks such as resetting passwords.

¹ Jenkins, S. (2000, June 26). *Deploying an Enterprise Directory: It Doesn't Help to be a Rocket Scientist*, The Open Group Meetings, Austin, Texas.

Automated Provisioning

[Provisioning provides the essential measures that must be taken in advance to accomplish specific tasks. For example, think of a new employee and the workflow associated with getting them set up—new network ID, phone number, e-mail address, office, desk, and so on. How do you get all these things lined up in your organization? How many people are involved?

A directory is useful in the provisioning process since it is a centralized store for user identity. Still, just because a directory is good at maintaining data in an accessible manner doesn't mean the data itself is correct. For example, think how many typographical errors exist in corporate information banks. Automated processes reduce the chance of new errors creeping into the system.

With Directory and Resource Administrator, you can encode your business processes so that the operation does not complete unless all conditions are met—ensuring that the data in the directory is complete. These naming rules and conventions are usually spelled out in your business policies. Directory and Resource Administrator comes with a number of policies already provided—right out of the box.

Provisioning Example

Let's use an example to show how customers can take advantage of Directory and Resource Administrator in the areas of built-in policy enforcement, custom policy enforcement, and the ability to generate their own automation using Directory and Resource Administrator's pre- and post-task scripting ability. We can show these abilities by performing a common administrative task—creating a new user.

Besides creating the new user object, we also want to do some provisioning: create a home share, home directory, set our home directory disk volume quota and create an Exchange 5.5 mailbox. We will also check to see if they have an employee number assigned in a database maintained by Human Resources (HR) and if that number has already been used.

The Home Volume Disk Quota policy lets us set the maximum limit, a default quota limit, and a default quota warning level.

In addition to the built-in policies used in our example, we can easily create a VBScript policy that will check the company HR database using the Microsoft ActiveX Data Objects (ADO) interface. We'll call our policy "*HR Policy*". Directory and Resource Administrator utilizes COM interfaces to run scripts, so you have a lot of programming power available to you for customizing your scripts.

The HR database contains information about the employee; including name, telephone number, location, address, and so on. In our example, this information has already been placed into the database by HR and is considered to be accurate and complete.

Pre- and post-task triggers handle automation in Directory and Resource Administrator. In our example, the pre-task trigger will obtain data from the HR database so that we can fill in information about the user before the user object is created (e.g. address, telephone number). After the pre-task trigger has been run, the user object is created. The post-task trigger will create the Exchange 5.5 mailbox and will also update the HR database to indicate that the user object has been created. If any step in the process fails, we rollback any changes made up to that point.

To demonstrate how policies work in action, let's create a new user. One of the attributes we see on the properties page for creating a new user is "Employee ID." The value of this field will be used by our VBScript policy to match up with data in the HR database.

To show off the power of Directory and Resource Administrator, let's look at what happens when we do the following:

1. Enter an invalid employee ID
2. Put in an employee ID that has already been used, and
3. Assign a valid employee ID

Our HR database doesn't have an employee ID 999, so if we enter 999 for the employee ID and attempt to finish the creation of the user in Directory and Resource Administrator, we will get an error message. We can tailor the message so it is meaningful to us, i.e. "The Policy object failed with the error: 'employee 999 is not in the database.'"

Next we put in an employee ID that has already been used, employee ID 001. How do we know it's been used? Because our *HR Policy* updates the HR database when a user is created—which is in keeping with a business decision made by our example corporation.

Again, we will get an error message when we try to complete this operation. This time, our error message explains "The Policy object failed with the error: 'A user account already exists for employee ID 001. Employee name: Floyd, Louis. User account: HR001.'" Because the *HR Policy* can return informative error messages, we can even see to whom the ID 001 belongs.

Now we go back to the properties page and put in the correct employee ID. Once we have finished the operation of adding a user, we find that a number of things have happened:

- The user account has been created with properties filled in with information taken from the HR database.
- An Exchange mailbox has been created with information filled in.
- A home directory was created.
- A new home share was created with the share name coming from the policy we set.
- This user's disk quota has been set
- The HR database was updated to indicate this user has been created in Directory and Resource Administrator.

So using out-of-the box and custom policies, we have been able to do complete automated provisioning in conjunction with the task of creating a user object.

The Value of Consistent Data

When you want to search a directory for someone in management by title, do you use *manager*, *mgr*, or what? Do telephone numbers include area codes or country codes? Finding information becomes much easier when rules are available that define how information is specified.

How would you accomplish that without Directory and Resource Administrator? By having a printed manual or a Web site that spell out the business process? Who makes sure the requirement is followed? The correctness of the information is left up to the person performing the operation, who may have to answer the telephone or deal with other interruptions that lead to human errors.

The ability to put your business processes into computer-executable scripts gives you the assurance that processes are followed *exactly*. Users who enter information don't have to interpret business policies because Directory and Resource Administrator maintains and enforces policies. And when users make mistakes, Directory and Resource Administrator displays customized error messages with information about how to correctly enter information.

Reducing the Administrative Burden

The IT skills gap is having a real effect on IT organizations. Even as IT administrators are being asked to do more to keep up with the incredible pace of the Internet-accelerated information age, it's harder to find skilled people ready to take up the challenge. To solve this problem, your organization can improve the efficiency of IT administrators and/or delegate simple tasks to specialized administrators (such as a department administrative assistant, who is authorized only to reset passwords and only for people in that department).

Another aspect of Windows system management has to do with *who* can do *what*. If a small group of administrators has to do everything connected with the directory and its information, the workload can be overwhelming.

A popular method for looking at access control is the CRUD (Create, Read, Update, Delete) matrix. If you look at your critical system resources and determine who can create, read, update and delete, you have a pretty complete picture of how to assign access rights.

When you are dealing with especially important data, you may want use a CRUD+O matrix—O for Owner. Your business policy should specify who the *owner* of the critical information is. For example, when a new employee is hired, who is responsible for the office telephone number being correct? Is it human resources, facilities, communications, a department administrator or someone else? If user Madeline's telephone number is wrong in the directory, who does she call to get it corrected—or can she change it herself?

Rules-based Administration

Directory and Resource Administrator operates by running a series of rules against requested operations, providing an efficient way to specify what should be done, because only those rules that relate to the task at hand are executed.

Many organizations look at moving to Windows 2000 in order to simplify management by collecting the huge volume of network information into Active Directory. Rules let you then simplify and promote distributed administration. A rules-based approach dynamically adapts to the latest data, as well as enabling you to delegate common administration tasks to members of your enterprise with confidence and security. And any rules you create in the Windows NT environment carry forward as you migrate to Windows 2000.

Policy-based Administration

Wouldn't it be nice to not have to involve a network administrator for every change that must be made to a directory? When users change their mailing addresses, wouldn't they like to be able to make those changes themselves? With Directory and Resource Administrator, you are able to say *who* has the ability to change *what*, so you can set things up to allow users to control much of their own information.

You can distribute administration tasks where it makes sense. For example, you might give the power to reset passwords for a department to the administrative assistant for that group. You can keep Help Desks and system administrators from having to do mundane, simple tasks. With the IT skills gap widening all the time, doesn't it make sense to do so?

You can use the rules you define in Directory and Resource Administrator to enforce company policies, such as naming conventions and creation of home directories. Since Directory and Resource Administrator can apply policies for the data, users don't have to become expert at your organization's information policy guidelines in order to use that information.

Access Control

Controlling who has access to what in Active Directory can be a daunting task. You must keep track of access rights that have been delegated and how those rights change over time (for example, new resources added or people changing jobs).

How does access control in DRA differ from that provided by the tools that come with Active Directory? The simple answer is that while you can use the Windows 2000 Delegation of Control wizard to assign permissions to people over various objects, there is no easy way to generate reports that answer the following kind of questions across the entire Active Directory:

1. Whose passwords can Bob change?
2. What groups can Sarah alter the membership of?
3. In which OUs can the New York Help Desk create users?
4. Over which objects does Barney have some write power?
5. Who changed the CEO's password and when?
6. Who tried unsuccessfully to add themselves to the Domain Admins group?
7. How do I enforce a content policy over AD (for example, Employee ID field entries must be nine numeric digits)?

8. How do I ensure that home directories are automatically created, assigned the correct permissions and quotas assigned?
9. How do I prevent the proliferation of administrator access privileges?
10. How do I reduce intrusions (for example, Help Desk person looks at the entire directory to obtain information on infrastructure security settings)

Simply put, if you want to know who can do what to whom, you must be able to manually look at every object. Directory and Resource Administrator and its ActiveView technology can help you in the following ways:

- Reports on all ActiveView delegations of authority (not ACL reporting).
- Auditing of all transactions against the directory when they occur.
- Content control (for example, an employee ID must be at least nine numeric digits in length).
- Simplicity—rules can be assigned to cover or cross OUs, name wildcards, multiple domains or even multiple forests.
- Exposes an ADSI to write to other applications enabled for Active Directory, letting you take advantage of content control policies.
- Assistant administrators can see only the objects they have some power over and that are relevant to their duties.
- Trigger-based automation that can cause changes outside the directory whenever the directory is changed.

Creating Your Own Views

If you look back a bit in history, you find that databases used to be largely hierarchical. So why are relational databases so prevalent today? The answer has to do with their ability to create a particular view into the data specifically matched to the task at hand.

ActiveViews

Active Directory organizes network objects in a *hierarchical* tree, enabling you to arrange the OU containing users and network objects according to geography, departments, functions, or whatever arrangement makes most sense for your organization. However, the hierarchical Active Directory structure has the disadvantage of being difficult to change—the hierarchy is fairly rigid and shouldn't be changed lightly. Administrators must work within the confines of the directory tree structure. On the other hand, with a *relational* database, it is easier to form particular *views*.

For example, if your organization has three major sites: Chicago, Houston and New York, and each site has departments for human resources, sales and marketing, the directory structure may be a simple tree (see figure 3 below) structured according to geography and departments. With this structure, if you want to update information about people who belong to sales departments, you would typically traverse the tree to the sales OU underneath the Chicago and work with those objects, then traverse to sales underneath Houston, and so on.

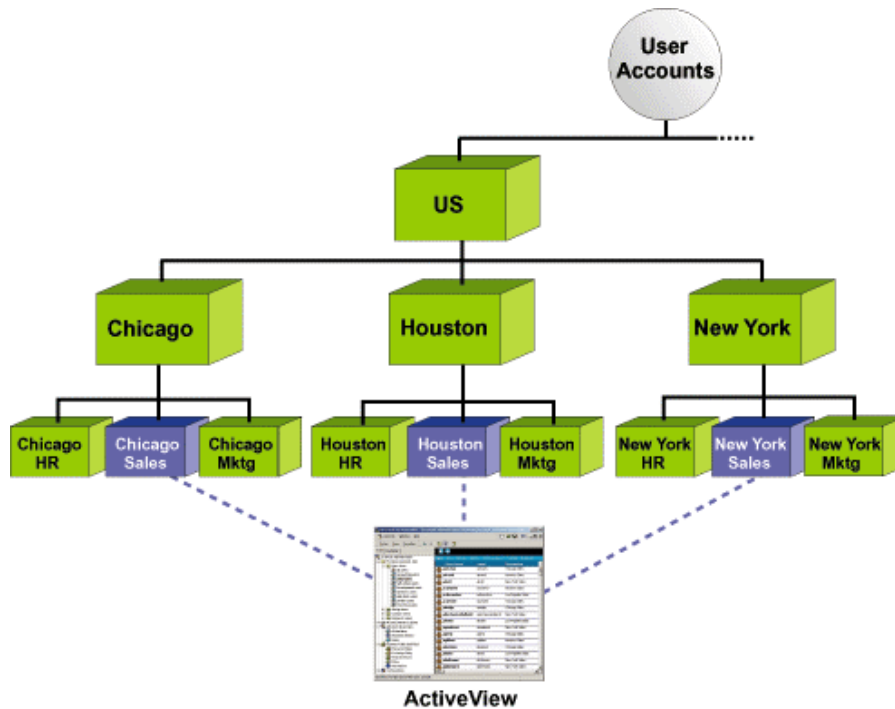


Figure 3 - OU Hierarchy and ActiveViews OU

But what you really want to do is to have a view into the directory which contains the people in the various sales containers. This is what an ActiveView accomplishes—it gives you a view of a *particular set* of objects. For example, you can designate an ActiveView to include everyone in OUs containing *sales*.

ActiveViews can provide multiple views of data in the hierarchical Windows 2000 Active Directory structure, enabling easier access to data in Active Directory and simplifying data views. And ActiveViews can span domains, both Windows 2000 and Windows NT, trees and forests.

For example, one ActiveView could include all objects related to the sales department, even though those objects may exist in Windows NT domains, resource domains and multiple Active Directories. The ActiveView of these objects displays all the objects as though they were in a single container. ActiveViews are “virtual OUs” that are defined based on rules in Directory and Resource Administrator .

ActiveViews can also seamlessly connect and administer data from multiple sources (such as SAM, Exchange and Active Directory) and can span domains. The ability for you to assign policy and automation to ActiveViews gives you additional management power and flexibility.

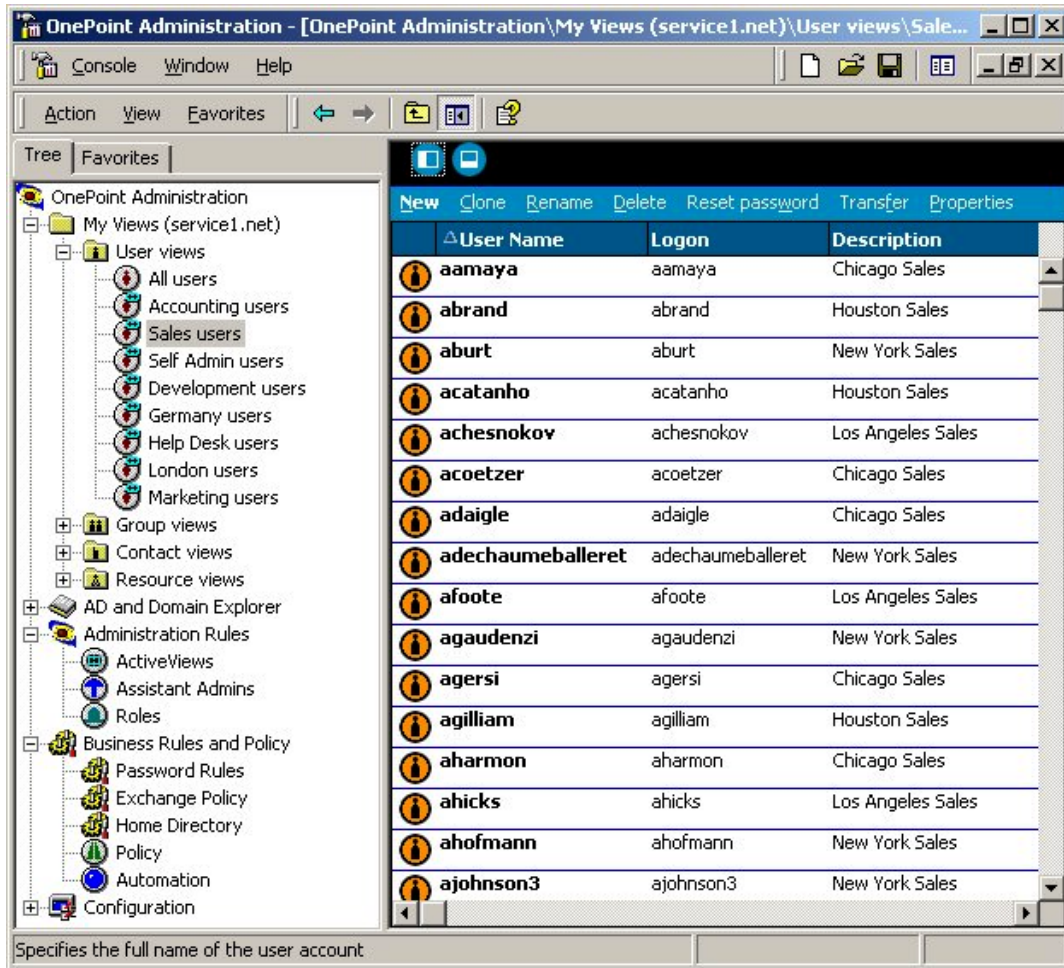


Figure 4 - DRA ActiveView Example

ActiveViews enables you to implement a flexible directory structure, enabling you to:

- Eliminate the problems and politics of being bound to a rigid, unchangeable hierarchical structure.
- Eliminate the need to compromise on allocation of administration resources.
- Facilitate change (such as acquisitions).
- Include data across directory boundaries.

ActiveViews dynamically adjust to schema changes made in the Active Directory. Also, using Virtual Property Objects (VPOs) technology, DRA can manage information not contained in the directory itself, such as disk quota on the volume of a user's home directory.

Unlocking the Power of Windows 2000 and Windows NT

Whether you are running Windows 2000, are in the midst of a move to Windows 2000 or are only thinking about moving to Windows 2000, Directory and Resource Administrator helps you in your administrative work *now*. Even if your organization uses only Windows NT, Directory and Resource Administrator running on a non-intrusive Windows 2000 server can manage your entire Windows NT environment.

In summary, Directory and Resource Administrator can unlock the power of Active Directory, Windows 2000 and Windows NT 4, providing you the ability to:

- Improve security.

- Assure data integrity and prevention of directory pollution.
- Unify and seamlessly administer Windows NT 4 and Windows 2000.
- Automate provisioning.
- Increase administration flexibility.
- Employ extensive, rich reporting and auditing.
- Reduce administrative costs.

You can do all this because Directory and Resource Administrator :

- Prevents directory pollution.
- Unifies administration with multiple data sources.
- Increases administrative flexibility via ActiveViews.
- Provides significant out-of-the-box policy and automation.
- Works seamlessly with both Windows 2000 and Windows NT combined.
- Performs cross-forest management.
- Employs auditing and reporting beyond that provided by Windows 2000.
- Improves productivity.
- Enables safe, distributed administration.

Directory and Resource Administrator is the only product you need for Windows 2000, Windows NT and Active Directory administration—as well as combined NT and Windows 2000 administration. No matter where you are in the Windows evolution, NetIQ Directory and Resource Administrator is there to make your life