IBM Rational ClearCase 4.x
and Active Directory
ABSTRACT: Like any enterprise-scale application in a Windows network, Rational ClearCase will be affected when the network is converted from using Windows NT domains to using Active Directory domains. This document describes how a conversion to active directory affects ClearCase 4.x communities that host VOBs and views on Windows, and how to manage ClearCase users, groups, hosts, and data during and after the conversion. (ClearCase communities that host VOBs and views on UNIX will not have to take any special steps when converting to active directory.)

1 Introduction

Active Directory is the directory service included with Microsoft Windows 2000. When your organization decides to convert from Windows NT domains to Active Directory, the conversion affects ClearCase users, groups, and hosts in several ways. If you understand how the conversion affects ClearCase and plan accordingly, you can ensure a minimum of disruption for ClearCase users.

NOTE: ClearCase 2002.05.00 (ClearCase 5.0) is fully compatible with Active Directory and provides additional features, including preservation of dynamic views during migration and added capabilities in *vob_sidwalk*. Rational recommends upgrading to ClearCase or ClearCase LT 2002.05.00 before you convert to Active Directory. If you are using or plan to deploy ClearCase 2002.05.00, please read the material on ClearCase and Windows Domains in the *Administrator’s Guide* for Rational ClearCase 2002.05.00 for details on how this release can simplify conversion of a ClearCase network to Active Directory.

This white paper deals primarily with issues that arise when a ClearCase 4.x network that is using Windows NT domains and hosting its VOBs and views on Windows is converted to using Active Directory. If you are already using an Active Directory environment but have not yet deployed ClearCase in it, much of the material in this document will not apply to you, and you may go directly to *ClearCase in an Existing Active Directory Environment* on page 13 for information on configuring Active Directory user and group accounts.

While any ClearCase 4.x or ClearCase LT release can be installed and used in an Active Directory domain environment, only ClearCase 4.1 and later releases can be converted from using Windows NT domains to using Active Directory. ClearCase releases prior to 4.0 are not compatible with Active Directory. Table 1 describes ClearCase compatibility with Active Directory.
This document was written for experienced ClearCase administrators who are familiar with Windows NT, Windows 2000, and Active Directory. We recommend that this document also be reviewed by domain administrators responsible for conversion of Windows NT domains to Active Directory, and that these administrators work closely with ClearCase administrators to plan a conversion that will have the least impact on the ClearCase user community.

### If All of Your VOBs and Views Are Hosted on UNIX

VOB and views that are hosted on UNIX servers do not store SIDs in their databases. Instead, they authenticate Windows users based on their UNIX user name and group memberships and store UNIX user and group IDs (uids and gids) in the VOB and view databases. ClearCase communities that host VOBs and views on UNIX will not have to take any special steps when converting ClearCase clients to active directory.

### Understanding Active Directory

Microsoft provides tools and documentation to facilitate conversion of a Windows network from Windows NT domains to Active Directory. In this document, we assume you have read the applicable documents from Microsoft and are familiar with the terminology they use and the procedures they describe. In particular, we assume you have read the Microsoft white paper entitled *Planning Migration from Microsoft Windows NT to Microsoft Windows 2000*, which is distributed as part of the Windows 2000 Support Tools and is also available on Microsoft’s Web site. That document and related documents introduce several key concepts—including native mode, mixed mode, domain upgrade, domain migration, SID history, and cloning of principals—that we use throughout this document.
How Active Directory Affects ClearCase

ClearCase relies on the operating system to provide identity information it uses when determining a user’s rights to access artifacts under ClearCase control. In an Active Directory environment, some of the details of user and group identity are handled differently than they are in a Windows NT domain environment. Depending on how your Windows NT domain environment is configured, where your ClearCase user and group accounts exist in this domain structure, and how your organization plans to convert Windows NT domains to Active Directory domains, you may need to take steps during and after the conversion process to maintain user access to artifacts under ClearCase control.

Conversion to Active Directory affects ClearCase in several ways:

➤ In Active Directory, trust relationships between domains are created and maintained differently than they are in Windows NT domains. During and after the conversion to Active Directory, these differences will affect ClearCase communities where users from multiple domains access a common set of VOBs and views.

➤ Windows Security Identifiers (SIDs) for users and groups can change in some conversion scenarios. Because ClearCase stores SIDs in VOB databases (to represent owners of objects), VOBs will have to be updated with new SIDs in these scenarios.

In general, sites that have the simplest domain structure (all ClearCase users and hosts in a single domain) will encounter very few issues during the conversion process. Sites with more complex domain structures (users from multiple domains accessing a common set of VOBs and views) will be able to benefit from Active Directory’s improved inter-domain security features after they modify some existing user and group account information.

2 Prerequisites

Before you begin converting a ClearCase network to Active Directory, you must understand the impact of various conversion scenarios on ClearCase and prepare ClearCase hosts for the transition.

Plan Your Active Directory Upgrade or Migration Strategy

Microsoft provides tools and documentation to facilitate conversion of domains from Windows NT to Active Directory. The conversion can take either of two forms:

➤ An upgrade (often referred to as an in-place upgrade), in which a Windows NT domain controller is converted to an Active Directory domain controller operating in mixed or native mode. After an upgrade, all users, groups, and resources have the same SIDs as they had in their original Windows NT domain.
A migration, in which user, group, and resource accounts migrate (using a process referred to as cloning) from a Windows NT domain to an Active Directory domain. After a migration is complete, all users, groups, and resources have different SIDs than they had in their original Windows NT domain. Because a native mode Active Directory maintains information about each principal’s current and former SIDs (referred to by Microsoft as the principal’s SID history), both types of domains can be used together for as long as needed.

We recommend that a knowledgeable ClearCase administrator who has reviewed this document and applicable documents from Microsoft, and who understands the impact of various conversion or migration strategies on ClearCase, be familiar with (and if possible help plan) your organization’s conversion from Windows NT domains to Active Directory.

CAUTION: Microsoft supplies tools for converting the SIDs stored in NTFS ACLs. You should never use these tools (or any tools that change native file system protection information) on a VOB or view storage directory. Only ClearCase utilities should be used to convert SIDs in VOB or view storage directories. See Migrating Multiple Domains on page 10 for details.

Prepare ClearCase Hosts

Before you begin the conversion process, your ClearCase hosts must be configured for use in an Active Directory environment.

➤ All ClearCase clients and servers must be running ClearCase version 4.0 or later.

➤ All ClearCase hosts must also be updated with the following patch bundles or ones that supersede them:

  ➤ clearcase_p4.0.NT-24
  ➤ clearcase_p4.1.NT-18
  ➤ clearcase_p4.2.NT-1
  ➤ ClearCase LT Service Release 2001A.04.01.293.005

➤ All VOBs must be at schema version 54. Schema version 54 stores Windows user and group identity information in SID form to better support Active Directory’s improved handling of user and group authentication.

➤ The user environment variable CLEARCASE_PRIMARY_GROUP must be defined for all users. We recommend that the value of this variable be a domain-qualified group name (DOMAIN_NAME\group_name).

Verify that all ClearCase hosts have been configured as described in this section and that ClearCase is operating normally for all users and hosts before you proceed with the conversion to Active Directory.
3 How ClearCase Uses Domain Groups

ClearCase depends on the user and group information stored in a domain account as well as on the trust relationships between domains to ensure the appropriate level of access to artifacts under ClearCase control. This section provides an overview of how ClearCase uses this information, and how this information differs between Windows NT and Active Directory.

Windows NT Domain Groups

As described in Administering ClearCase, each VOB and every object in it is owned by a user and a group. Dynamic views and many of the objects they contain (including view-private files and DOs) also have an owner and a group. In a Windows NT domain environment, ClearCase users must observe several rules relating to group membership:

➤ Every user who requires access to a common set of VOBs must be a member of an agreed-upon domain group, commonly referred to as the ClearCase users group, typically named clearusers.

NOTE: The ClearCase users group is not the same as the ClearCase administrators group (typically named clearcase), a group to which only a few administrators and the ClearCase server process user (clearcase_albd) belong.

➤ The ClearCase users group must be a domain global group.

➤ Each member of the ClearCase users group must set the environment variable CLEARCASE_PRIMARY_GROUP to the name of the ClearCase users group.

➤ In environments where users from multiple Windows NT domains must access a common set of VOBs, proxy groups must be used, as described in Administering ClearCase.

These rules also apply in a mixed-mode Active Directory environment. In a native-mode Active Directory domain, ClearCase users can take advantage of additional types of groups.

Active Directory Groups

Windows NT domains support two types of groups:

➤ Local groups can include any user or group in a trusted domain, but can be granted rights only on the local computer. (Local groups created on computers that are domain controllers can be granted rights on any domain controller in the domain, but not on other computers in the domain.) The ClearCase users group cannot be a local group.

➤ Domain global groups can include any user in the domain and can be granted rights on any computer in the domain. In a Windows NT domain, the ClearCase users group must be a domain global group.

Native-mode Active Directory domains support two additional types of groups:
➤ Domain local groups can include any user or group in a forest, but can be granted rights only in the domain in which they were created. The ClearCase users group cannot be a domain local group.

➤ Universal groups can include any user or group from any domain in a forest and can be granted rights in any domain in a forest.

Because Active Directory universal groups can contain other groups, they provide a superior alternative to the proxy group scheme that must be used when ClearCase communities include users from multiple Windows NT domains. See Converting Proxy Groups on page 8 for details.

The Primary Group

Although you can designate a user’s primary group using Active Directory account maintenance tools, we continue to recommend setting the user environment variable CLEARCASE_PRIMARY_GROUP to guarantee an unambiguous definition of the group that ClearCase should consider the user’s primary group.

In environments where all ClearCase users and hosts are members of a single Active Directory domain, the ClearCase users group should be a domain global group because there is no need to provide users from other domains with the ability to access artifacts under ClearCase control. A domain global group does not incur the Active Directory replication overhead associated with a universal group, and can be added to a universal group later if necessary.

In environments where users who are members of more than one Active Directory domain must access a common set of VOBs and views, the ClearCase users group must be a universal group. A universal group cannot be designated as a user’s primary group using Active Directory administration tools. In this case, users must set the CLEARCASE_PRIMARY_GROUP environment variable to the name of the universal group.

4 Domain Upgrade Scenarios

This section describes several scenarios in which one or more Windows NT domains are upgraded to Active Directory domains using the in-place upgrade procedure defined by Microsoft. If your site is not using this procedure, see Domain Migration Scenarios on page 9.

NOTE: In this section, when we refer to upgrading a domain, we mean upgrading the primary domain controller of that domain.

Upgrading a Single Domain

Use this procedure if all of your ClearCase users, groups, and hosts are members of a single Windows NT domain:

1. Prepare ClearCase hosts as described in Prepare ClearCase Hosts on page 4.
2. If you do not have a backup domain controller on-line during the upgrade, stop ClearCase on all ClearCase server hosts to prevent ClearCase operations during the upgrade process.

3. Use the procedures defined by Microsoft to perform an in-place upgrade of the Windows NT domain to an active directory domain.

4. After the domain upgrade is complete, shut down and restart all ClearCase client and server hosts.

## Upgrading a Master Domain and its Resource Domains

Use this procedure if all ClearCase users and groups are members of a single Windows NT domain (the master domain) that is trusted by one or more Windows NT resource domains to which ClearCase hosts belong. Because an upgrade changes the trust relationship between a resource domain and a master domain, user and group identity information stored in dynamic view databases will be invalid after the resource domain upgrade. Before you upgrade a resource domain, remove any dynamic views that exist on hosts in that resource domain. The views can be recreated as needed after the upgrade is complete. Snapshot views, which do not store user and group identity information in the same way as dynamic views, do not need to be changed.

Use the following procedure to upgrade a master domain and its resource domains:

1. Prepare ClearCase hosts as described in *Prepare ClearCase Hosts* on page 4.

2. If you do not have backup domain controllers for the master and resource domains on-line during the upgrade, stop ClearCase on all ClearCase server hosts to prevent ClearCase operations during the upgrade process.

3. Use the procedures defined by Microsoft to perform an in-place upgrade of the Windows NT master domain to an Active Directory domain.

4. After the master domain has been upgraded, you may begin to upgrade the resource domains. As long as you do not alter existing trust relationships between domains that have been upgraded and those that have not, you may upgrade the resource domains in any order, and on any schedule that is appropriate for your organization.

   **NOTE:** Before you upgrade any resource domain, remove all dynamic views that reside on hosts in that resource domain.

5. After the upgrade of a resource domain is complete, shut down and restart all ClearCase client and server hosts in that resource domain.

6. Create new dynamic views as needed on hosts in the upgraded resource domain.

## Upgrading Multiple Master and Resource Domains

Use this procedure if ClearCase users and groups are members of more than one Windows NT domain and you are using proxy groups to enable users from these domains to access a common
set of VOBs and views. The domains can include resources as well as users and groups, or can be master domains trusted by resource domains.

Because an upgrade changes the trust relationship between a resource domain and a master domain, user and group identity information stored in dynamic view databases will be invalid after the resource domain upgrade. Before you upgrade a resource domain, remove any dynamic views that exist on hosts in that resource domain. The views can be recreated as needed after the upgrade is complete. Snapshot views, which do not store user and group identity information in the same way as dynamic views, do not need to be changed.

After the upgrade is complete, we recommend converting the Active Directory domains to native mode so that you can use an Active Directory universal group to eliminate the need for proxy groups and domain mapping.

Use the following procedure to upgrade multiple master and resource domains:

1. Prepare ClearCase hosts as described in Prepare ClearCase Hosts on page 4.

2. If you do not have a backup domain controller for each of the master and resource domains on-line during the upgrade, stop ClearCase on all ClearCase server hosts to prevent ClearCase operations during the upgrade process.

3. Use the procedures defined by Microsoft to perform an in-place upgrade of the first Windows NT master domain to an Active Directory domain. Upgrade the domain in which the ClearCase users group is defined before upgrading the domains in which the proxy groups are defined.

4. Upgrade the remaining master domains.

5. After the master domains have been upgraded, you may begin to upgrade the resource domains. As long as you do not alter existing trust relationships between domains that have been upgraded and those that have not, you may upgrade the resource domains in any order, and on any schedule that is appropriate for your organization.

   NOTE: Before you upgrade any resource domain, remove all dynamic views that reside on hosts in that resource domain.

6. After the upgrade of a resource domain is complete, shut down and restart all ClearCase client and server hosts in that resource domain.

7. After all domains have been upgraded and the Active Directory domain has been converted to native mode, follow the procedure described in Converting Proxy Groups on page 8 to eliminate proxy groups and domain mapping.

8. Create new dynamic views as needed on hosts in the upgraded resource domain.

Converting Proxy Groups

Use the following procedure to replace the proxy groups required in a Windows NT domain environment with a ClearCase users group that is an Active Directory universal group. The universal group includes the groups formerly used as proxy groups.
NOTE: You can only use this procedure in an Active Directory domain that is operating in native mode. The ClearCase users group and the proxy groups must all exist in the same forest.

1. Use Active Directory management tools to change the type of the ClearCase users group from global to universal.

2. Use Active Directory management tools to add the former proxy groups as members of the (universal) ClearCase users group.

3. After the proxy groups have been added to the (universal) ClearCase users group, disable domain mapping on all ClearCase client and server hosts. To disable domain mapping on a ClearCase host, use a registry editor to navigate to the registry key HKEY_CURRENT_USER\SOFTWARE\Atria\ClearCase\CurrentVersion, then remove the DWORD value DomainMappingEnabled. (Some ClearCase hosts may store this value in HKEY_LOCAL_MACHINE\SOFTWARE\Atria\ClearCase\CurrentVersion.)

4. Shut down and restart all ClearCase client and server hosts.

5. Ask users to set the value of CLEARCASE_PRIMARY_GROUP to the name of the universal ClearCase users group.

## 5 Domain Migration Scenarios

Sites for which a domain upgrade is impossible can use a domain migration process. This process uses Microsoft tools to populate a native-mode Active Directory domain with user, group, and resource accounts that have been cloned from existing accounts in a Windows NT domain. Both types of domain can operate in parallel and, if the appropriate trust relationships exist between the Windows NT and Active Directory domains, users and groups in either type of domain will have equivalent ClearCase access rights.

Migration can take place over an extended period if necessary. When all user and group accounts have been migrated to the Active Directory domain, the migration process can be completed and the Windows NT domains decommissioned. After a migration, all users, groups, and hosts have new SIDs. This has several implications for ClearCase:

- Dynamic views hosted on computers that are members of a Windows NT resource domain must be removed before the master domain is decommissioned.

- After a host has become a member of an Active Directory domain, you must reconfigure the host’s albd_server process to log on using the ClearCase server process account (typically called clearcase_albd) and ClearCase administrators group (clearcase) that exist in the Active Directory domain. The names of the accounts are still the same, but their SIDs have changed.

- VOB storage directories must be re-protected so that they include the new SIDs of the VOB owner in the directory ACL.

- VOB databases must be updated with the new SIDs. The ClearCase patches listed in Prepare ClearCase Hosts on page 4 include a utility program, vob_sidwalk, that replaces old SIDs.
Detailed instructions for performing these procedures are included in the remainder of this section.

NOTE: Ownership-preserving MultiSite replicas must also be updated with new SIDs. See Using -delete_groups With Ownership-Preserving Replicas on page 17 for more information about this procedure.

Migrating Multiple Domains

All migration scenarios are essentially the same, differing only in their level of complexity as measured by the number of domains being migrated and the trust relationships among them. The procedure defined in this section can be used whether you are migrating a single domain or multiple master and resource domains.

NOTE: Before you begin any migration process, prepare all ClearCase hosts as described in Prepare ClearCase Hosts on page 4. Even though the hosts may not be migrating right away, they will not be accessible by migrated users and groups until you have taken this step.

Migrating Users and Groups

We recommend that you begin any domain migration by migrating users and groups from the Windows NT domains in which they were created to new Active Directory domains. Follow these steps to migrate ClearCase users and groups:

1. Use the procedures defined by Microsoft to migrate users and groups from the Windows NT domains to the Active Directory domains. Be sure to include the ClearCase users group, ClearCase administrators group, and clearcase_albd account in the migration.

2. In many migration scenarios, there will be a period when users logged into the Active Directory domain will share a VOB with users logged into a Windows NT domain. To ensure access to VOBs by users in either type of domain, do one of the following:

   ➣ Add the domain-qualified name of the ClearCase users group that has been migrated to the Active Directory domain to the VOB’s supplemental group list. For example, you could use the cleartool protectvob command as shown here to add the clearusers group in the Active Directory domain AD-DOMAIN to the group list of the VOB with storage on the VOB server host at C:\vobsvr\vobstg\srcs.vbs:

      cleartool protectvob -add_group AD-DOMAIN\clearusers C:\vobstg\srcs.vbs

   ➣ Ask users logged into an Active Directory domain to set their CLEARCASE_PRIMARY_GROUP environment variable to the string representation of the SID of the ClearCase users group in the Windows NT domain. To find the SID string, run the creds command—on a computer that is a member of the Windows NT domain or a domain that trusts the Windows NT domain—as shown in this example:
ClearCase group info:
Name: NT-DOMAIN\clearusers
GID:  0x100423
SID credentials NT:S-1-5-21-103034363-981818062-1465874335-1064

In this case, the user should set `CLEARCASE_PRIMARY_GROUP` to the value
NT:S-1-5-21-103034363-981818062-1465874335-1064

3. Because migrated accounts include SID history, user accounts in the Active Directory domain include twice as many group memberships as they had in the Windows NT domain. (Each user’s group list includes groups from both domains.) Users who are members of multiple groups in a Windows NT domain may find that their group list includes more than 32 groups after migration. Because ClearCase only recognizes 32 groups, these users should set their `CLEARCASE_GROUPS` environment variable to include the SID string that represents the ClearCase users group in the Windows NT domain, as well as the name of the ClearCase users group in the Active Directory domain. See Step #2 of this procedure for information on how to use `creds` to obtain the SID string.

If the user environment variable `CLEARCASE_GROUPS` exists for any user, ClearCase will consider the semicolon-separated list of groups specified in the value of this variable first when determining (or displaying) which groups a user belongs to. This example includes the name of the ClearCase users group from the Active Directory domain and the SID of the ClearCase users group from the Windows NT domain.

```
CLEARCASE_GROUPS=AD-DOMAIN\clearusers;NT:S-1-5-21-103034363-981818062-1465874335-1064
```

If You Must Add a New User While Migration is In Progress

If a new ClearCase user must be created while a domain migration is in progress, use either of the following methods:

➤ Create the user’s account in the Active Directory domain and ask the user to set `CLEARCASE_PRIMARY_GROUP` environment variable to the domain-qualified name of the ClearCase users group that has already been cloned and exists in the Active Directory domain.

➤ Create the user in the Windows NT domain, then migrate the user account.

Like any other ClearCase user, these users may need to change their `CLEARCASE_PRIMARY_GROUP` and `CLEARCASE_GROUPS` environment variables as described in Migrating Users and Groups.

Migrating Individual ClearCase Hosts

When all users and groups have been migrated and no users are required to log into the Windows NT domain, take the following steps to migrate ClearCase hosts to the Active Directory domain. If you cannot migrate all your ClearCase hosts at the same time, we
recommend migrating VOB servers first. (Registry and license servers can migrate at any time, since they do not store SIDs in their databases.)

NOTE: The procedures in this section require you to use the new `vob_sidwalk` utility. Before executing any of these procedures, we recommend that you review Section 9 of this document to get a better understanding of the capabilities of `vob_sidwalk`.

To migrate an individual ClearCase host:

1. Remove any dynamic views on the host.
2. Stop ClearCase on the host.
3. Use the procedures defined by Microsoft to migrate the host to the Active Directory domain.
4. After you migrate the host, reconfigure its `albd_server` to log in as the (migrated) ClearCase server process user in the Active Directory domain. You can do this by reinstalling ClearCase on the host and specifying the new account, or you can do it manually as described in `Administering ClearCase`.
5. If the host is a VOB server that will also be accessed by UNIX clients, reset the credentials mapping domain in the Options tab of the ClearCase Properties application in Control Panel. The credentials mapping domain should be one in which user and group account names match those of UNIX users that will access VOBs on this server.
6. Restart ClearCase on the host.
7. Reprotect any VOB and snapshot view storage on the host by running the ClearCase `fix_prot` utility on each VOB or view storage directory as shown in the following example

   `ccase-home-dir\etc\utils\fix_prot -replace storage-dir-pname`

   where `storage-dir-pname` is the pathname to the VOB or view storage directory.
8. Use the `vob_sidwalk` program to update VOB databases with the new SIDs representing the cloned user and group accounts.
   a. Log into the VOB server host as the VOB owner or privileged user.
   b. Lock the VOB for all users but yourself (`-nusers you`).
   c. Replace the old SIDs with the new ones. Use the `vob_sidwalk` utility as shown in the following example, where `vob-tag` is the VOB-tag of the VOB you are updating and `SIDfile-path` is the name of a file where `vob_sidwalk` will log the changes it makes:

   `ccase-home-dir\etc\utils\vob_sidwalk -sidhistory -execute vob-tag SIDfile-path`
   d. Unlock the VOB.
9. If any of the VOBs on the host is an ownership-preserving MultiSite Replica, see `Using -delete_groups With Ownership-Preserving Replicas` on page 17 for additional steps you must take with this and other replicas in that VOB’s replica family.
NOTE: If you had been using proxy groups to enable users from multiple domains to access a common set of ClearCase artifacts, we recommend using an Active Directory universal group to eliminate the need for proxy groups and domain mapping. Follow the procedure described in Converting Proxy Groups on page 8

If VOB Servers Cannot Migrate When Clients Do

If any VOB server cannot migrate when its clients do and you need to preserve the clients’ ability to access VOBs on that server, you must use `vob_siddump` to establish the mapping between new SIDs and old ones. After the mapping has been established, you can use `vob_sidwalk` to update the VOB database with the new SIDs.

1. Log into a client that has been migrated to the Active Directory domain.
2. Lock the VOB for all users but yourself (`-nusers you`).
3. Run `vob_siddump` to generate a map file. (You must use `vob_siddump` because you cannot run `vob_sidwalk` from a remote host.) In the following example, `vob-tag` is the VOB-tag of a VOB on a server that is still in the Windows NT resource domain, and `SIDfile-path` is the pathname to the map file that `vob_siddump` will generate. (If `SIDfile-path` cannot be created on a drive that is accessible to the VOB server host, you must copy it to the VOB server host before you execute Step #4 of this procedure.)

   \[vob_siddump -sidhistory vob-tag SIDfile-path\]

4. Log into the VOB server that hosts the VOB whose tag you used in Step #3 of this procedure. With the VOB still locked for all users but yourself, run `vob_sidwalk` to update the SID information stored in the VOB

   \[vob_sidwalk -execute -map mapfile-path vob-tag SIDfile-file\]

   In this example, `mapfile-path` is the map file you generated in Step #3 of this procedure and `SIDfile-path` is the name of a file where `vob_sidwalk` will log the changes it makes. See Using `vob_sidwalk` on page 15 for more on this topic.

5. Unlock the VOB.

NOTE: If you do not leave the VOB locked from the time you begin Step #3 until the time you complete Step #4 of this procedure, users may create new objects in the VOB between the steps, which will require you to perform both steps again.

6 ClearCase in an Existing Active Directory Environment

If you are already using an Active Directory environment but have not yet deployed ClearCase in it, most of the material in Administering ClearCase regarding domain administration will not apply to you. However, the fundamental rules of account administration for ClearCase user and group accounts are similar
➤ Every user who requires access to a common set of VOBs and views must be a member of an agreed-upon domain group, commonly referred to as the ClearCase users group.

➤ If all ClearCase users are members of the same Active Directory domain, the ClearCase users group can be either a domain global group or a universal group. We recommend using a domain global group to simplify administration and minimize Active Directory replication overhead.

➤ If ClearCase users who access a common set of VOBs and views are members of multiple Active Directory domains in a single forest, the ClearCase users group must be a universal group. Each domain global group to which ClearCase users belong must be a member of this universal group.

NOTE: Universal groups are not supported unless the Active Directory is operating in native mode.

➤ Each member of the ClearCase group must set the environment variable CLEARCASE_PRIMARY_GROUP to the domain-qualified name of the ClearCase users group.

➤ Certain Active Directory domain configurations may make dynamic views on some hosts inaccessible to some users. This is most likely to be the case when a user from one domain accesses a view on a host in another domain and that host’s domain has a transitive trust relationship with the user’s domain. Several measures to mitigate these problems are described in Section 7 of this document.

7 Issues With Dynamic Views

If all ClearCase users, groups, and hosts are members of a single Windows NT domain, dynamic views can be preserved during an upgrade or migration and used without any changes after the upgrade or migration is complete.

If all ClearCase users and groups are members of a single Windows NT domain (the master domain) that is trusted by one or more Windows NT resource domains to which ClearCase hosts belong, dynamic views must be removed before the resource domains are upgraded or the resources migrated, but can be re-created as needed after the upgrade or migration is complete.

In other scenarios, changes in trust relationships that happen during a conversion to Active Directory can affect the accessibility of dynamic views. ClearCase communities that are affected in this way can take either of the following steps to mitigate problems with dynamic views:

➤ When creating new dynamic views on hosts in the Active Directory domain, make sure that the views have read and write access for everyone. On UNIX hosts, set your umask to 0 before creating the view. On Windows hosts, set the user environment variable CLEARCASE_MKVIEW_MODE to a value of 0777 before creating the view. CLEARCASE_MKVIEW_MODE can be used on any host patched as described in Prepare ClearCase Hosts on page 4 to change the default permissions with which a view is created.

NOTE: This strategy requires directory elements that are writable by everyone and file elements that are readable by everyone. ClearCase normally creates directory and file
elements with these permissions, so unless you have changed the default protections on
directory and file elements, this strategy will provide everyone with access to dynamic
views. (Directory writability determines only whether users can create view-private files
within a directory element in a dynamic view.)

➤ Create a direct trust relationship between each of the Active Directory master and resource
domains that ClearCase users and hosts belong to. Active Directory transitive trust
relationships cannot provide the user and group identity information needed by the view
server when it receives a request from a user in another domain to access an object in the
view.

8 Using vob_sidwalk

The vob_sidwalk command can be used in a number of migration scenarios to read or remap
SIDs stored in a VOB database. This section provides examples of several use cases involving
vob_sidwalk. See Section 9 for reference information on vob_sidwalk. (vob_sidwalk is not
usually needed in domain upgrade scenarios.)

Mapping SIDs

In any migration scenario, after a VOB server host migrates from a Windows NT domain to an
Active Directory domain, none of the SIDs stored in the VOB database resolve to user or group
accounts in the Active Directory domain. You can use vob_sidwalk to address this issue in any
of several ways.

Whenever SIDs must be remapped, we recommend that you start by running vob_siddump or
vob_sidwalk without the -execute option, then examining the output to determine which objects
in the VOB would get new owners, before running vob_sidwalk with the -execute option to
actually remap the SIDs. The output SID file is written in comma-separated-value (.csv) form, so
it can be viewed and changed with any text editor or any spreadsheet program that can read a
file written in this format.

Mapping Old SIDs to New SIDs

To replace the old SIDs for migrated users and groups with the new ones, use a command line
like this:

vob_sidwalk -sidhistory -execute vob-tag SIDfile-path

When invoked with the -sidhistory option, vob_sidwalk uses the following algorithm to
determine SID history.

1. Look up a SID to find the account name
2. Look up the account name found in Step #1 to find its SID
3. If the SID returned in Step #2 is different from the SID looked up in Step #1, the SID looked up in Step #1 is assumed to be an historical SID, and the SID returned in Step #2 is written to the new-SID field of the current line of the SID file. (An historical SID is the SID associated with a security principal before migration, and is stored in the principal’s sIDHistory attribute in an Active Directory domain)

If `vob_sidwalk` was invoked with the `-execute` option, the SID used in Step #1 is replaced with the SID returned in Step #2.

**Reassigning Ownership to the VOB Owner**

To reassign ownership of objects in the VOB by mapping all old SIDs to the new SIDs of the VOB owner and group, use a command line like this:

`vob_sidwalk -unknown -execute` 
`vob-tag` 
`SIDfile-path`

When invoked with the `-unknown` and `-execute` options, `vob_sidwalk` maps unresolvable user SIDs to the SID of the VOB owner and maps unresolvable group SIDs to the SID of the VOB’s group.

**NOTE:** If you are using UCM, you may not want to reassign ownership with `-unknown`. Reassigning an open activity to the VOB owner will make it unusable by its creator (unless it was created by the VOB owner).

**Custom Reassignment of Ownership**

To reassign ownership of objects in the VOB by mapping old SIDs to new SIDs using a mapping table you supply, generate a SID file using a command line like this:

`vob_sidwalk -sidhistory` 
`vob-tag` 
`SIDfile-path`

You may edit the file in `SIDfile-path` to change the mapping. See Section 9 for information on the fields in this file, their meaning, and what changes are permitted. When the file has been edited, you can use it as the `mapfile-path` in a command line like this:

`vob_sidwalk -execute -map` 
`mapfile-path` 
`vob-tag` 
`SIDfile-path`

**Resetting VOB Storage Directory Protections**

If VOB storage directory ACLs have been damaged during a migration (or by any other event), you can use `vob_sidwalk` as shown here to correct the ACLs on the VOB storage directory and container files:

`vob_sidwalk -recover_filesystem` 
`vob-tag` 
`SIDfile-path`

When used with the `-recover_filesystem` option, `vob_sidwalk` will also correct the SIDs for the VOB’s supplementary group list.
Using -delete_groups With Ownership-Preserving Replicas

ClearCase MultiSite customers who use ownership-preserving replicas (created with \texttt{mkreplica -preserve}) must take several additional steps when they migrate those replicas’ hosts from Windows NT domains to Active Directory.

Because the changes in SIDs made by \texttt{vob_sidwalk} are not oplogged (and therefore not propagated by replication), the MultiSite administrator must run \texttt{vob_sidwalk} on each ownership-preserving replica in a replica family when the server that hosts the replica is migrated to Active Directory. When it is run on an ownership-preserving replica, \texttt{vob_sidwalk} preserves the original SIDs on VOB’s group list, so that operations which require container creation will continue to succeed whether or not all ownership-preserving replicas in a family have been updated. After all ownership-preserving members of a replica family have been updated, the administrator must run \texttt{vob_sidwalk} again using the \texttt{-delete_groups} option to remove these historical group SIDs. We recommend removing historical SIDs because a VOB has a limit of 32 groups on its group list. Keeping unused historical SIDs on the list may cause the list to overflow as new groups are added.

\textbf{NOTE:} This procedure assumes you have already migrated user and group accounts for all users of all replicas to Active Directory, and that all users have set their \texttt{CLEARCASE\_PRIMARY\_GROUP} environment variable to the name of the ClearCase group in the Active Directory domain.

1. Synchronize all replicas in the family to be sure that each replica includes the same set of user and group SIDs.

2. Follow the procedure described in \textit{Migrating Individual ClearCase Hosts} on page 11 to migrate hosts. All ownership-preserving replicas in a family must be processed using the same \texttt{vob_sidwalk} options. If the \texttt{-map} option is used, you can save time if you generate one mapping file and use it on all ownership-preserving replicas in a family.

3. After the replica has been synchronized again with other replicas that have had their SIDs updated as described in Step #2 of this procedure, run the command:

\texttt{vob_sidwalk -sidhistory \texttt{vob-tag SIDfile-path}}

and examine the resulting SID file to see if any new SID mappings are needed (because new user or group identities have been added to the replica). If new SID mappings are required, run \texttt{vob_sidwalk} again using the options you used in Step #2 of this procedure.

4. After all ownership-preserving replicas have been updated as described in Step #2 of this procedure and the SID file generated in Step #3 of this procedure shows that no new SID mappings are needed, run \texttt{vob_sidwalk -execute -delete_groups} on each of the replicas. This deletes historical group SIDs from the VOB’s group list.

9 \texttt{vob_sidwalk Reference Page}

\texttt{vob_sidwalk, vob_siddump}
Read or change Windows security identifiers (SIDs) in schema version 54 VOB databases

**SYNOPSIS**

```
vob_sidwalk  [-p profile profile-path ] | [ -s idhistory ] | [ -u unknown ]
             [ -m ap mapfile-path ] | [ -l log logfile-path ] | [ -e xecute ] | [ -delete_groups ]
             vob-tag SIDfile-path
vob_sidwalk  --recover_filesystem vob-tag SIDfile-path
```

```
vob_siddump  [-p profile profile-path ] | [ -s idhistory ] | [ -u unknown ]
             [ -m ap mapfile-path ] | [ -l log logfile-path ] vob-tag SIDfile-path
```

**DESCRIPTION**

**vob_sidwalk** and **vob_siddump** are administrative utilities that can be used to read or change Windows security identifiers stored in VOB databases formatted with schema version 54. **vob_sidwalk** and **vob_siddump** are not compatible with VOB databases formatted with schema version 53.

The programs are typically needed when you:

- move a VOB from one Windows NT domain to another Windows NT domain
- move a VOB from a Windows NT domain to an Active Directory domain

**vob_siddump** is a read-only version of **vob_sidwalk**. It can be executed on the VOB server or any client to list the SIDs stored in a VOB and optionally generate mapping files that can be used as input to **vob_sidwalk**.

**vob_sidwalk** must be executed on the VOB server. It has all of the capabilities of **vob_siddump**, and can also change SIDs in the VOB database. In addition, **vob_sidwalk** can be executed with the **-recover_filesystem** option to reset the protections on a VOB storage directory to be consistent with the SID of the VOB’s owner and group.

**RESTRICTIONS**

There are no restrictions on **vob_siddump**.

**vob_sidwalk** has the following restrictions:

**Identities:** You must have one of the following identities to run **vob_sidwalk**:

- VOB owner
- Member of the ClearCase group (ClearCase on Windows)
- Local administrator of the ClearCase LT server (ClearCase LT on Windows)

**Locks:** An error occurs if the VOB is locked.

**Other:** You must enter this command on the VOB server host.

**OPTIONS AND ARGUMENTS**

**READ OR MAP SIDS** Default: None. These options are allowed with both **vob_siddump** and **vob_sidwalk**.

- **-s idhistory**
  Find historical SID information stored in the VOB database. Write the current name and SID for each account to the **new-name** and **new-SID** fields of the SID file in **SIDfile-path** and write the historical name and SID to the **old-name** and **old-SID** fields. If either command is invoked without this option, it writes the current name and SID for each account to the **old-name** and **old-SID** fields of the SID file, and the **new-name** field is set to **IGNORE**.
--unknown
Map SIDs that cannot be resolved to an account in the domain. Any user SID that cannot
be resolved is mapped to the SID of the VOB owner. Any group SID that cannot be
resolved is mapped to the SID of the VOB’s primary group. The mappings are written to
the SID file.

--profile profile-path
Write a list of all SIDs found in the VOB along with the database identifiers that describe
objects owned by each SID. The list is written to the file in profile-path. Each line of the
file has the format:

metatype,dbid,user-name,user-SID,group-name,group-SID,mode,container...

where each field has the form:

metatype
The VOB meta-type name, or one of the special names
ROOT, TREE, or FILE for file system objects that have no
dbid
database identifier for this VOB object
user-name
user name of the object’s owner
user-SID
string representation of user SID
group-name
group name of the object’s group
group-SID
string representation of group SID
mode
the object’s access mode
container...
pathname of the object’s container file, if applicable

This option can generate a large file in profile-path and consume significant resources on
the VOB server host. This option cannot be used with any other option.

--map mapfile-path
Force remapping of all SIDs in a VOB database as specified in the mapping file at
mapfile-path. Details about the SID remappings for the VOB at vob-tag are written to
SIDfile-path.

The mapping file contains one or more lines in the following format.

old-name,type,old-SID,new-name,type,new-SID

where each field has the form:

old-name
domain-name\account-name
new-name
one of domain-name\account-name, IGNORE, DELETE
type
one of USER, GROUP, GLOBALGROUP,
LOCALGROUPONDC, LOCALGROUP
old-SID, new-SID
string representation of SID

You can use a SID file from a previous run of vob_sidwalk or vob_siddump as the
mapping file. If you need to change the existing mapping (to reassign ownership of
objects), you can edit the file to make any of the following changes:

Change the new-name field to IGNORE to prevent reassignment of ownership.
Change the new-name field to DELETE to reassign ownership of the object to the VOB
owner
Specify a different SID in the new-SID-string field to map ownership to the new user or group. In this case, vob_sidwalk ignores the contents of the new-name and associated type field, then verifies that the type of the new-SID-string matches the type of the old-SID-string.

**UPDATE SIDS** Default: Only read or map SIDs. Do not change anything in the VOB database unless the -execute option is present. These options are not allowed with vob_siddump.

- **execute**
  Modify SIDs stored in the VOB database. Unless the -execute option is used, vob_sidwalk logs, in the SID file, the changes that would have been made but does not actually change anything in a VOB database.

- **delete_groups**
  Remove any historical SIDs found in the group list of an ownership-preserving replica. Historical SIDs are always removed from the group list of a non-replicated VOB or a non-ownership-preserving replica.

**LOGGING** Default: No logging.

- **log logfile-path**
  Write a log of SID reassignments. Each line of the file at logfile-path has the format:
  metatype,dbid,container,old-SID,reserved,new-SID
  where each field has the form:
  metatype The VOB meta-type name, or one of the special names ROOT, TREE, or FILE for file system objects that have no dbid
dbid database identifier for this VOB objectcontainer old-SID string representation of old SIDreserved reserved for future usenew-SID string representation of new SID

**FIXING STORAGE DIRECTORY PROTECTIONS** Default: Do not change protections.

- **recover_filesystem**
  Fix protections on VOB storage directory. This option is not supported with vob_siddump. With vob_sidwalk, it cannot be used with any other option.

**VOB-TAG** Default: none

- vob-tag
  A VOB-tag specifying the VOB to which to operate.

**SID FILE** Default: none

- SIDfile-path
  A pathname at which the command should write the SID file. An error is returned if SIDfile-path exists or is not specified. Each line of the SID file has the format:
  old-name,type,old-SID,new-name,type,new-SID,count
  where each field has the form:
  old-name domain-name\account-name
### ClearCase 4.x and Active Directory

<table>
<thead>
<tr>
<th>new-name</th>
<th>one of domain-name\account-name, IGNORE</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>one of USER, GROUP, GLOBALGROUP, LOCALGROUPONDC, LOCALGROUP</td>
</tr>
<tr>
<td>old-SID, new-SID</td>
<td>string representation of SID</td>
</tr>
<tr>
<td>count</td>
<td>number of objects with this owner</td>
</tr>
</tbody>
</table>

You can use the SID file as the mapping file when running either command with the `-map` option.
IBM software integrated solutions
IBM Rational supports a wealth of other offerings from IBM software. IBM software solutions can give you the power to achieve your priority business and IT goals.

- **DB2®** software helps you leverage information with solutions for data enablement, data management, and data distribution.
- **Lotus®** software helps your staff be productive with solutions for authoring, managing, communicating, and sharing knowledge.
- **Tivoli®** software helps you manage the technology that runs your e-business infrastructure.
- **WebSphere®** software helps you extend your existing business-critical processes to the Web.
- **Rational®** software helps you improve your software development capability with tools, services, and best practices.

Rational software from IBM
Rational software from IBM helps organizations create business value by improving their software development capability. The Rational software development platform integrates software engineering best practices, tools, and services. With it, organizations thrive in an on demand world by being more responsive, resilient, and focused. Rational’s standards-based, cross-platform solution helps software development teams create and extend business applications, embedded systems and software products. Ninety-eight of the Fortune 100 rely on Rational tools to build better software, faster. Additional information is available at www.rational.com and www.therationaledge.com, the monthly e-zine for the Rational community.

Rational is a wholly owned subsidiary of IBM Corp. (c) Copyright Rational Software Corporation, 2003. All rights reserved.

IBM Corporation
Software Group
Route 100
Somers, NY 10589
U.S.A.

Printed in the United States of America
01-03 All Rights Reserved.
Made in the U.S.A.

IBM the IBM logo, DB2, Lotus, Tivoli and WebSphere are trademarks of International Business Machines Corporation in the United States, other countries, or both.

Rational, and the Rational Logo are trademarks or registered trademarks of Rational Software Corporation in the United States, other countries or both.

Microsoft and Windows NT are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States, other countries, or both.

UNIX is a trademark of The Open Group in the United States, other countries or both.

Other company, product or service names may be trademarks or service marks of others.

The IBM home page on the Internet can be found at ibm.com.