Updated DSMerge - Merging two trees together *(Last modified: 24JAN2000)*

This document (**2912750**) *is provided subject to the disclaimer at the end of this document.*

Issue

To prepare for a tree merge using DSMERGE:

NOTE DSMERGE will not work in Pure NDS 8 Environments.
If you are using NDS4NT and the domain(s) exist in the tree to be merged, you must change the tree name in the registry of your domain controllers. Follow TID 2939465 to do this.

1. Optional: To shorten the time required for the merge, you may want to make the O= level on the source tree its own partition. If there are more than one O= on this tree, make each it's own partition. This is the largest part of the merge and will save a lot of time.

2. Make sure the O= (IE. organizational container) on each tree is unique (different name). Since all O's will become O's in the new tree. Also verify that the tree names are different. If not, change the name of your source tree using DSMERGE | Rename this tree. Renaming the source tree is wisest, since the tree name will change after the merge anyway.

3. Make sure DS on all servers in both trees are at version 5.01 or higher if 4.10, and 5.73 or higher (excluding 5.81 or 5.89) if 4.11.

4. Make sure you are using the latest DSMERGE.NLM, found in the DS410x.EXE or DS411x.EXE files at http://support.novell.com | Minimum Patch List. It resolves the issues people had with past versions of DSMERGE.NLM.

5. Rebuild operational schema on the master replica of [Root] for both trees. DSREPAIR | Advanced options menu | Repair local DS database | Change the "Rebuild operation schema" to Yes | press F10

6. Make sure that DSREPAIR | Report Synchronization Status on the master of both trees has Total errors: 0. Resolve ANY errors before beginning. It would be advisable to use TID#2913292 - General Health to make sure all errors are resolved and the network is stable.

7. Optional: You should also create a database dump file on each master server in each tree: A simple way: load DSREPAIR -RC. Or, DSREPAIR | Advanced Options | Create a Database Dump File. This creates an archive of the SYS:_NETWARE directory. If there was a serious problem, Novell could dial-in via PCAnywhere and put you right back where you were before you started.

8. Verify DSREPAIR | Time synchronization in both trees. It is a good idea to have servers in your source tree referencing time from your target tree. To do this, load SERVMAN on your main time source in your source tree | Server parameters | Time, and modify the following:

TIMESYNC ADD Time Source	Your target tree's main time source (Ex. NYFS001)
TIMESYNC Configured Sources	On
TIMESYNC Directory Tree Mode	Off (allows time to be taken from another tree)
TIMESYNC Type	SECONDARY (if it was SINGLE, or PRIMARY if it was REFERENCE)
Default Time Server Type	SECONDARY (if it was SINGLE, or PRIMARY if it was REFERENCE)

Escape twice, and when prompted to Update AUTOEXEC.NCF and STARTUP.NCF now? and Update TIMESYNC.CFG now? select both options to update both sets of files.

9. If the schemas are mismatched, and you will know as you run the DSMERGE pre-check, use DSREPAIR | Advanced options menu | Global schema operations | Import remote schema, on the master of each tree, one at a time, and import the schema from the other tree. [both ways] Document 1005795 also references this option. It is a good idea to do this multiple times as the schema doesn't always fully import the first time.

Most merges go fine, as long as things are working properly before you start. The biggest problem people have is they don't patch the servers first, they don't make sure the lan drivers and support files (msm, ethertsm, nbi) are up to date, UNTIL they have a problem. This is imperative.

If you have a very large tree merge to perform see the readme.txt for DS410x.EXE or DS411x.exe and verify that everything is configured properly before you begin.

Follow the instructions below for more detailed instructions:

Introduction:

   DSMERGE is used to join two directory services trees together so that the new combined tree can be accessed by clients of both of the previous trees. A source tree and target tree are selected for the merge operation. The source tree is merged into the target tree at the root and becomes part of the target tree. The tree names of all the servers in the source tree are renamed with the name of the target tree.

   If more than two trees are merged together, then each of the source trees are individually merged into the target tree until a single tree remains. If two trees have been created with the same tree name, then one of the two trees must have its tree name renamed before the merge can be performed. In order to perform a merge or rename operation, DSMERGE.NLM must be loaded on the source tree on the server that contains the master replica of the root partition. If it is loaded on another server, you will be told which server contains the master replica of the root partition.

   DSMERGE also has a diagnostic tool to check time synchronization within a tree. This tool will display differences in the synchronized time between servers in the tree, report the status of time synchronization, and show whether or not time is synchronized.

   Users currently logged in to servers which are being merged will not be affected.

The Merge Process:

   For security reasons, you must in login to both trees as an administrator with supervisory rights to the root of the tree to ensure that you have proper authority to perform a merge.

   The merge process occurs in four phases. The first step of a tree merge is the check phase which looks for problems which would prevent the merge from completing successfully. If any are found, the operator is notified to correct them before continuing.

   The second step is to prepare the source tree for the tree merge. The preparation phase affects all servers in the source tree that have a replica of the root partition. After the preparation phase, the source tree is ready to be merged, but no operations to actually merge the two trees have been performed. If the merge is canceled at this point, each tree will continue to operate independently.

   The third step is to perform modifications to the target tree. These modifications all occur within a single transaction. If something goes wrong while modifying the directory database within a transaction, the transaction is canceled and all modifications are backed out. This guarantees that the database isn't corrupted if the merge fails. If there is a problem modifying the target tree, the transaction is canceled, the target tree remains unchanged, and the merge is stopped.

   If the target tree modification is successful, then the fourth step is performed which is to modify the source tree. If the source tree transaction fails then the merge is canceled and the operations performed on the target tree will be automatically cleaned up over time. Both trees will continue to operate independently of each other and the merge can be tried again when the cause of the failure is resolved. If the transaction on the source tree also completes, then the trees are merged.

   Although technically the trees are now merged, other operations, such as the placement of subordinate references when the synchronization process runs, will occur after the merge. DSMERGE will then wait for a replica of the target tree's root partition to be placed on the source tree server. DSMERGE will then rename all the servers in the source tree with the new tree name, which is the target tree's name.

Deciding Which Tree Should Be Merged Into the Other:

   There are several items to consider when deciding which tree should merge into the other.

1:   The source tree servers will all have their tree name renamed to the target tree name, and this may affect clients that login to these servers if they are using the PREFERRED TREE statement in their NET.CFG file as described under "NET.CFG" later in this article. If this will affect your clients, you may want to merge the tree with fewer clients into the one with more to save some work later.

2:   Most of the time spent during a merge is in the preparation phase in which partitions are split. Selecting the tree with the fewer number of subordinate objects to the root object and fewer replicas of the root partition will speed up the preparation phase. Depending on the number of server in the

replica rings, it can be advantageous to remove replicas prior to starting the merge. The default of one Master and two R/W replicas should work effectively if all or most are local. More replicas or replicas placed on servers over WAN links will delay the process completion.

3:    Since only objects in the root replica of the target tree are actually copied between trees during a merge, the number of overall objects in a tree does not affect the time to perform the merge.

Things to Do Before Merging the Trees:

  Before merging two trees together, both trees should be operating properly, synchronizing replicas without errors, and all servers from both trees should be synchronizing time from the same time source. If possible, all servers in both trees should be up and operating. If this isn't possible, at least all servers in both trees that contain replicas of the root partition must be up as they are needed for the partition operations.

  In the target tree the servers that contain a replica of the root partition must also be version 4.1. Any 4.0x servers that contain a replica of the root partition on the source tree will be removed by DSMERGE in the preparation phase.

DSREPAIR. You should run DSREPAIR on both trees before running DSMERGE. DSREPAIR checks that all replicas in a tree are synchronizing properly and identifies and corrects synchronization problems. Merging two trees that are not operating properly or that have database damage may cause problems.

  Removing servers from the tree improperly, or directory services administration operations such as partition join, partition split, moving objects or subtrees, and replica placement that have failed due to communications or hardware failure may also cause problems.

Time Synchronization. Time synchronization configuration and the placement of Secondary, Primary and Reference servers should be reassessed before merging trees together. Both DSMERGE and DSREPAIR have tools to check time synchronization, and need to be run on a server that has a replica of the root partition in order to check all the servers in a tree. A healthy directory services tree should have time synchronized on every server in the tree.

  When performing a merge, time synchronization requires consideration since all servers in both the source and target tree should be synchronized to the same time and preferably to the same time source before the merge. This would require that time synchronization is setup using configured sources' instead of SAP (Service Advertising Protocol) for at least one of the trees. After the merge completes, there will be only one tree, and all servers in that tree will need to be time synchronized.

  The DSMERGE and DSREPAIR tools cannot report the time source for each server and verify that all servers derive time from the same source. The important thing, however, is that all servers in the tree have the same time. Since time differences between the servers are displayed, it is easy to identify which servers need attention. Before the merge, each tree must be checked independently since the tools do not access servers across tree boundaries.

Time synchronization on large trees will be organized using "configured sources" with a one Reference server and multiple Primary servers distributed across WAN links. The Secondary servers will only synchronize to the Primary servers in their configuration list. Tree boundaries are not an issue with "configured sources."

Small trees with several servers can be configured to rely on SAP to find a time source. This is the default configuration when the first server in a new tree is installed, and its time synchronization type is set to "Single Source." As additional servers are installed into the tree they become Secondary servers by default and use SAP to find a time source within their tree.

To preserve this time configuration method, manually synchronize the time on the two Single Source servers in each tree and check that all the Secondary servers are synchronized to them. Then change the Single Source server in one tree to a Secondary just before the merge (see step 8 at the beginning). After the merge, there will be only one Single Source advertising time synchronization services to the other Secondary servers in the tree.

If there is more than one Single Source server found in a tree, the TIMESYNC.NLM will respond with alerts on the server console until the condition is resolved. It is preferable to use configured sources instead of SAP, and preparation for a tree merge is a good time to change this.

Schema. DSMERGE does not resolve discrepancies in the schema between two trees. Since the trees are being merged, the schema on the source and target tree must match or the trees cannot be merged. If you have installed applications that have modified the schema on a tree, install the same application on the other tree so that the schema on both trees will match.

DSMERGE will report discrepancies between the schema on each tree being merged and will not allow a merge if they do not match identically. A future release of DSREPAIR provides tools to converge the schema between two trees. It is found under the "Advanced Options" menu, "Global Schema Options", function "Import remote schema". This operation may need to be run on both trees to make the schema in both trees match.

Duplicate Names. Since the trees are merged at the root, all objects directly subordinate to the root in each tree must have unique names. If there are duplicate names between the two trees, DSMERGE will report them and stop. You must rename the duplicate object in one of the trees before proceeding with the merge.
Access Control. Access control to the root object in the target tree is preserved after the tree is merged. However, objects granted access control to the root of the source tree may not be preserved. Use NETADMIN or NWADMIN to see what objects have trustee rights to the root of the source tree before the merge so that they can be restored after the merge if needed.

Preparing the Source Tree Server

DSMERGE prepares the source tree so that the merge can be performed by modifying only one server in the source tree and one server in the target tree. To do this, all replicas of the root partition of the source tree are removed and only a single copy is left on the server with the master replica. Because this violates the guideline that there should be at least two replicas of each partition, the

subordinate objects of the root are first split into their own partitions. This leaves a single object in the root partition, which is the root object itself, but does not change the replica distribution of all partitions subordinate to it.

The fault tolerance and distribution of the objects that were in the root partition are preserved in the new partitions. When only one object remains in the root partition, all replicas of the root are removed except for the master.

DSMERGE also ensures that this server contains a replica of the partition that its own server object is in. If the server's object is in a partition that it does not have a replica of, then a replica is moved to the server. If this replica is not a master, it is made the master replica of the partition.

This is done because immediately after the merge completes, the target tree server will only be able to authenticate to the source tree server, and it needs access to the master replica of that partition to link in its new subordinate reference.

Only container objects can exist subordinate to the root object. Other objects and aliases must be deleted or moved. These are object classes Organization (O=name) and Country (C=name), unless new classes have been defined.

Operations on the Target Tree Server

Several operations on the target tree root server all occur within a single transaction. An external reference object is created for the source tree server below a partition root of type subordinate reference which represents the partition root of the same partition in the source tree. The external reference is created because the target tree server needs a local ID to identify the source tree server's object in data structures in its database.

Net address and public key properties are stored on the external reference of the source tree server object so that the target tree server can authenticate to the source tree server. Since we ensured that the master replica of the subordinate reference just created is on the source tree server, and since we can authenticate to the source tree server, we can link the subordinate reference in with the master replica immediately after the merge has completed. Subordinate references are created under the root object for all partition roots under the root object on the source tree.

This subordinate reference placement ensures connectivity of the new tree. These subordinate reference partition roots will link in with the master replicas of their respective partitions over time as the replica synchronization process runs. The replica ring of the root partition on the target tree is modified to place a new replica on the source tree server.

If the transaction on the target tree fails, all modifications made to the target tree are backed out and the source tree transaction is not performed. The source tree is left in the prepared state, with the split operations on the subordinates of the root and removal of all replicas but the master. If the merge operation is not going to be performed again, then at least several replicas of the root partition should be distributed to other servers in the tree to restore normal replica distribution.

Operations on the Source Tree Server

Several operations on the source tree root server also occur in a single transaction. An external reference object is created for the target tree server below a partition root of type subordinate reference which represents the partition root of the same partition in the target tree. The external reference is created because the source tree server needs a local ID to identify the target tree servers object in data structures in its database.

Net address and public key properties are stored on the external reference of the target tree server object so that the source tree server can authenticate to the target tree server. The access control properties of the root object are preserved and all other properties of the object are removed. The replica ring of the root partition is modified to place a new replica from the target tree server.

If the transaction on the source tree fails, all modifications made to the source tree described above are backed out. The target tree will be left with subordinate references and external references to objects in the source tree. The external references will be purged after the backlink process fails to backlink the external reference to a real object over a period of time.

When this happens, the replica property of the root object which is used to place a new replica on the source tree root server will be removed as well. The subordinate references left in the target tree can be removed with DSREPAIR.

Completing the Merge and Renaming the Tree

At the conclusion of the merge, the source tree server will receive a replica of the target trees root partition. With the exception of placing several strategic external references and subordinate reference objects, this is the only time during the merge that objects are actually copied between servers. This means that the merge will occur very quickly once it begins.

DSMERGE waits for about two minutes for a new replica of the root partition to synchronize to the source tree root server. If this does not complete in the allocated time, a message reports that there may be a synchronization problem between the two servers. If the root replica is large and contains lots of objects, it may be that not enough time was allowed for the new replica to synchronize.

Use DSREPAIR to check synchronization of the root replica ring to determine if there is a problem. In either case, the merge has been completed and cannot be backed out.

There are now two ADMIN user objects in the tree. The ADMIN user from the source tree should be deleted.

Renaming the Tree for the Source Tree Servers

After the final transaction on the source tree is completed and the merge has been performed, the

servers that were in the old source tree must have their tree name renamed to the tree name of the target tree. This tree name is stored in the directory services database on each server.

DSMERGE makes a list of the addresses of all of these servers before the merge begins, and uses it to connect to each server and request that it changes its tree name. The servers will then verify the request by authenticating to the new tree, and if successful, they will change their tree name and migrate to the new tree.

A server in the source tree may not receive this request to change its tree name for the following reasons.

1:   The server could have been down during the merge operation.
2:   There could be a communications failure.
3:   The server may mistakenly not have been identified as being part of the source tree.

Regardless of the reason, the server will need to change its tree name before it can operate properly in the new tree, since the server will only perform directory services operations with other servers in the same tree.
NetWare 4.1 servers have the ability to find the new tree and migrate to it automatically, even if several tree merge or tree name rename operations have been performed while the server was off line.
NetWare 4.02 servers however will only change their tree name if they receive the initial request after the merge operation.
NetWare 4.01 servers will change their tree name from the initial request, but will then require directory services to be restarted before resuming normal communications with the other servers. This can be done by either running DSREPAIR which shuts down directory services while it runs and then restarts it afterwards, or by downing the server and booting it again. If any of the NetWare 4.0x servers do not receive the initial request they will need help to find the new tree.

This is done with the latest version of DSREPAIR for 4.0x servers which is shipped on the NetWare 4.1 distribution CDROM in directory \NW402\NLS\ENGLISH, or it can be downloaded from Novell's Support Connection Web Site (http://www.support.novell.com) or Novell's ftp server on the internet. Run this version of DSREPAIR on the 4.0x server and select option 7, Move this server to the correct tree after a merge tree operation. Then select the tree that the server belongs to from a list of all available trees. Security requires you to login to the destination tree.

Note: This cannot be used to move a server from one tree to another. If the tree selected is not the tree that this server belongs to, then none of the servers in that tree will be able to authenticate to it or perform directory services operations.

NET.CFG. After all the servers in the source tree have a new tree name, clients that connect to these servers may need some maintenance. If the NET.CFG file used to load the NetWare drivers specifies a PREFERRED SERVER, then no other action is required since the server still exists with its original server name. However, if the PREFERRED TREE statement is used, it must be changed to contain the name of the new tree, since the old tree no longer exists. Then when the drivers load, the first server from the specified tree that responds makes the initial connection.

If the preferred server or the preferred tree does not exist, the client will randomly connect to any

---

server in any tree that responds first when the drivers are loaded. If there is only one tree available, then the server found will be part of the desired tree and things will appear to work, until another tree appears in the future.

If the initial server connection is made to a server that is not in the correct tree, then the default context for the client will not be found and a login attempt will fail. Using the command line syntax for login to specify the tree will override the current server connection and login to the desired tree, i.e., login tree-name/user-name /tree.

Note: If both PREFERRED SERVER and PREFERRED TREE are specified, the PREFERRED TREE statement takes precedence, and the PREFERRED SERVER statement is ignored even if the PREFERRED TREE is not found.

The same rules above apply to individual tree rename operations as well.

After the Merge Has Completed

You can use the tools in DSMERGE and DSREPAIR to check the new tree after the merge has completed. These tools use information found only on a server that contains a replica of the root partition to be able to find all the servers in the tree.

Another method would be to walk the entire tree looking for server objects, but this which would take a long time in a large tree. This method requires that all subordinate references have been placed and external reference objects have been properly back linked. It may take a few minutes before a server with a replica of the root partition has enough information to find all of servers in the combined tree.

If you run the operations to verify the severs or to check time synchronization to soon after the merge, you may not get a complete list of all the servers in the tree.

Check that all servers in the tree have the correct tree name, and also that they have the same time. When a NetWare 4.1 server is found that does not have the correct tree name, a request will automatically be sent to that server to change its tree name.

If DSMERGE identifies a server that has the wrong tree name, the DSREPAIR function "check time synchronization' will additionally display the current tree name of that server.

Reference: March 95 Appnotes
Search: Merging Trees Tree Merge DSMerge DS Merge Two Combining precheck, precheck

Document Title:	DSMerge - Merging two trees together
Document ID:	2912750
Document Revision:	17
Creation Date:	01SEP1996

Modified Date:	24JAN2000
Novell Product Class:	End of Life
	Groupware
	NetWare
	Novell Directory Services
Novell Product and Version:	NetWare-EOL
	GroupWise
	NetWare 3.12
	NetWare 4.1
	NetWare 4.11
	NetWare 4.2
	NetWare 5
	NetWare for Small Business 4.2
	Novell Clients
	intraNetWare 4.11
	NDS 8

DISCLAIMER

The Origin of this information may be internal or external to Novell. Novell makes every effort within its means to verify this information. However, the information provided in this document is for your information only. Novell makes no explicit or implied claims to the validity of this information.

Any trademarks referenced in this document are the property of their respective owners. Consult your product manuals for complete trademark information.