

Quest® Migration Manager for Active Directory
8.14

Resource Kit User Guide



© 2019 Quest Software Inc. ALL RIGHTS RESERVED.

This guide contains proprietary information protected by copyright. The software described in this guide is furnished under a software license or nondisclosure agreement. This software may be used or copied only in accordance with the terms of the applicable agreement. No part of this guide may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's personal use without the written permission of Quest Software Inc.

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

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

Quest Software Inc.

Attn: LEGAL Dept

4 Polaris Way

Aliso Viejo, CA 92656

Refer to our Web site (<https://www.quest.com>) for regional and international office information.


Patents


Quest Software is proud of our advanced technology. Patents and pending patents may apply to this product. For the most current information about applicable patents for this product, please visit our website at <https://www.quest.com/legal>.

Trademarks

Quest, the Quest logo, and Join the Innovation are trademarks and registered trademarks of Quest Software Inc. For a complete list of Quest marks, visit <https://www.quest.com/legal/trademark-information.aspx>. All other trademarks and registered trademarks are property of their respective owners.

Legend

 **CAUTION:** A CAUTION icon indicates potential damage to hardware or loss of data if instructions are not followed.

 **IMPORTANT, NOTE, TIP, MOBILE, or VIDEO:** An information icon indicates supporting information.

Migration Manager for Active Directory Resource Kit User Guide

Updated - February 2019

Version - 8.14

Contents

Introduction	1
Profile Updating Utilities	1
Account and Computer Updating Utilities	1
Miscellaneous Utilities	2
System Requirements	3
ExportProfile	4
Pre-requisites for Using ExportProfile	4
Using ExportProfile	4
Format of the ExportProfile Parameter File	6
ExportProfile Return Codes	7
Format of the ExportProfile Log File	7
ChangeProfile	9
Using ChangeProfile	9
Format of the ChangeProfile Parameter File	10
ChangeProfile Return Codes	13
Format of the ChangeProfile Log File	13
Remarks	14
RegWalker	15
Using RegWalker	15
Format of the RegWalker Parameter File	16
Task Description File Format	18
OS	18
BaseKey	19
Group	19
Find	19
Replace	20
Create	21
Copy	21
Move	21
Delete	22
SetVal	22
DelVal	22
RenVal	23
CopyVal	23
Output	23

String Variables	24
RegWalker Return Codes	25
Computer List File Format	26
Migrated Users Map File	26
Format of the RegWalker Output File	27
Format of the RegWalker Log File	27
RegWalker Usage Examples	27
GetRsl	30
Using GetRsl	30
OCSMigrator	31
OCSMigrator System Requirements	31
Using OCSMigrator	31
Exporting User Data	33
Generating Vmover.ini File	33
Importing User Data	34
Updating the Contacts That Point to Migrated User Accounts	35
Clearing the Migration Database	36
InputBox	37
Using InputBox	37
GPMCEXport	38
GPO Migration Procedure	38
CertMgr	42
CertMgr Configuration	42
Using CertMgr	43
SSN Creator	45
Active Directory Cleanup Utility	46
Using Active Directory Cleanup Utility	46
About us	47
Technical support resources	47

Introduction

Migration Manager Resource Kit is a growing collection of utilities that can work alone or in concert to customize the migration process. Each utility has been developed in close cooperation with our customers and partners. Using one or more of these utilities can significantly simplify your migration.

The utilities can be classified as follows:

- [Profile Updating Utilities](#)
- [Account and Computer Updating Utilities](#)
- [Miscellaneous Utilities](#)

Each utility is described in detail in its own topic.

Profile Updating Utilities

Although SIDHistory takes care of access to resources, it does not take care of user profiles, which control such things as the desktop appearance, preferences, and programs. In order for users to have the same profiles in the target domain as they had in the source domain, Quest offers two utilities that can be distributed through user logon scripts and update profiles as appropriate. Each utility works in a similar fashion and supports a controlled and complete profile migration. The two utilities do not rely on each other, but when used together, they ensure that users will not be disrupted because of profile problems.

The **ExportProfile** utility should be run while the users are still logging on to the source domain, but their accounts have been migrated to the target domain along with SIDHistory. The utility dynamically creates a mapping of the current source account to new target account and updates profile links. This operation is totally transparent to the end user, and all results can be centrally logged. We anticipate this utility will be used for about 90% of the migrated user profiles. If the utility has already been run on a machine for the user logging in, it will not run again.

The **ChangeProfile** utility should be run when the users begin logging into the new domain. This utility will update a profile if it was not updated by ExportProfile, which would be the case if the user did not log in to that machine while ExportProfile was in use. ChangeProfile requires that a user log out once ChangeProfile has updated the user's profile.

Account and Computer Updating Utilities

The **RegWalker** is a command line tool used to search and replace textual information in a Windows 9x/NT/2000/XP/2003 registry database. The search operation can be performed on keys, values, and data. The

replace operation can be performed on only values. The utility can be deployed through logon scripts to automate mass client reconfigurations. Some situations in which the utility might be used include the moving of Windows 9x workstations to a new domain, and the reconfiguring of Windows NT/2000/XP/2003 and Windows 9x for a new application or file server during server consolidation.

The **GetRsl** is a command-line utility that reports current registry size and the registry size limit for a remote computer.

The **OCSMigrator** utility helps you migrate OCS/LCS accounts after you migrate the corresponding Active Directory accounts. The utility also migrates group lists, contact lists and block lists.

Miscellaneous Utilities

InputBox provides a graphical user interface to command-line utilities. It can be used with Migration Manager Resource Kit utilities or any other applications that accept their parameters from the command line.

GPMCExport custom add-in allows administrators to migrate group policy objects (GPOs) when performing the domain migration.

CertMgr utility can be used to automate the encryption certificate migration used to give access to encrypted NTFS files.

SSNCreator utility can export the migration account mapping to the Domain Migration Wizard session format. This format can be used to re-permission resources by other applications such as Quest Consolidator.

System Requirements

Migration Manager Resource Kit can be installed on an Intel-based computer where Migration Manager is installed running any of the following operating systems:

- Microsoft Windows Server 2019
- Microsoft Windows Server 2016
- Microsoft Windows Server 2012 R2
- Microsoft Windows Server 2012
- Microsoft Windows Server 2008 R2 Service Pack 1 or higher
- Microsoft Windows Server 2008 Service Pack 1 or higher (x64 edition)
- Microsoft Windows Server 2008 Service Pack 1 or higher (x86 edition)
- Microsoft Windows 7 Service Pack 1 or higher (x86 edition)
- Microsoft Windows 7 Service Pack 1 or higher (x64 edition)

ExportProfile

This utility associates the current source user's profile with the future (migrated) account. As a result, the same profile will be shared between the source and target accounts.

i **NOTE:** The utility links the new account to the old profile. However, it does not grant the new account the permissions needed to use the profile. This has to be done manually or with other Migration Manager components. See Migration Manager for Active Directory Resource Processing Guide for details..

Pre-requisites for Using ExportProfile

The following requirements must be satisfied before ExportProfile can be used:

- The user account must exist in the target domain.
- The explicit right to modify the following registry key must be granted to the user account:

```
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\  
CurrentVersion\ProfileList
```

- A two-way trust relationship must be set up between the source and the target domains.
- The source domain must be visible in the target domain and vice versa. You can check that by running the net view command from the command prompt:

```
net view /domain:SourceDomain
```

Using ExportProfile

ExportProfile uses the following command line syntax:

```
ExportProfile.exe IniFileFull
```

where IniFileFull is the path to the parameter file.

The format of the parameter file is described in the Format of the ExportProfile Parameter File topic.

The program will accept either a full UNC path to the parameter file or a relative path. If the file is specified in relative format, ExportProfile assumes the file is located in the Windows directory—not in the current directory. To work around this limitation, use the following syntax from a logon script or a batch file:ExportProfile.exe %0\..\IniFile

Or use the following syntax from the command line:

```
ExportProfile.exe .\IniFile
```


When looking for the SIDHistory, the ExportProfile queries the global catalog of the forest. To process the query, the computer where the utility is executed has to be a Windows NT 4.0 computer and a member of a native mode Windows 2000 domain, or a Windows 2000 computer. If the computer is a Windows NT 4.0 system in Windows NT 4.0 domain or a mixed-mode Windows 2000 domain, UseOldNames in the parameter file should be set to 1.

If a mapping is found, the program creates a key for the migrated account in the following location:

```
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\  
CurrentVersion\ProfileList
```

ExportProfile fills the new key with values taken from the old key of the logged-on user.

If a key for the migrated account already exists in the registry, then only the link to the **ProfileImagePath** is changed. To simplify troubleshooting, rollback, and cleanup, the old value is saved under the name **ProfileImagePath.BAK**.

When the program is run from a logon script, the INI file can be located in the Netlogon share, along with logon script itself.

If the operating system supports the %LOGONSERVER% system variable, you can use it in the argument line and put the parameter file in any share with read access.

i **NOTE:** If the account was renamed during migration and SIDHistory was not used, ExportProfile cannot link the new account to the old profile.
If the target user was already logged in on the domain controller, ExportProfile cannot link the target account to the old profile. The utility will have access denied because the users don't have rights on the other users' registry branch on the DC. There are no problems if logging in on a workstation or a member server..

Examples

```
> ExportProfile.exe ExportProfile.ini  
> ExportProfile.exe %0\..\ExportProfile.ini
```

For a local run:

```
> ExportProfile.exe .\ExportProfile.ini
```

ExportProfile and ChangeProfile can be used in one custom add-in file in the following way:

```
>ChangeProfile.exe %0\..\ChangeProfile.ini  
>if ERRORLEVEL 0 ExportProfile.exe %0\..\ExportProfile.ini
```

Since ChangeProfile determines the absence of a profile for the new account and returns 0, ExportProfile can be executed to export the current user settings to the future user profile.

Checking the return code of ChangeProfile helps to prevent running ExportProfile when there is no need for both programs to be executed. Running ExportProfile in this case guarantees all the users will have their profiles updated.

The setting of two keys in the registry does a marking of the execution:

```
[HKEY_CURRENT_USER\Software\Aelita\RegUtil\ExportProfile]  
AttempNumber=Number of last attempt
```

and

```
[HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\  

```

CurrentVersion\ProfileList\TextSID] MiniProfileMigration=2

If the link is successfully changed, the program saves the old link under a new name called ProfileImagePath.BAK. An administrator or a power user could revise this value to locate the unused profiles and clear disk space allocated to them.

Format of the ExportProfile Parameter File

[Task]

Domain=Migration Target Domain

LogFileName=%LOGONSERVER%\WRITE_SHARE\ExportProfile.log

AttemptNumber=number

UseOldNames={0|1}

Explanation:

Parameter	Description
Domain	Target domain. It is assumed that the user has been migrated from the source domain to the specified domain.
LogFileName	<p>The file to which the program will write the ERRORLEVEL code description. This parameter can contain environment-variable strings to dynamically control the location of the log file. For example:</p> <p>%LOGONSERVER%\WRITE_SHARE\ExportProfile.log %UserProfile%\ExportProfile.log</p> <p>If the %LOGONSERVER% variable is not supported, then you can specify a path to a share with write access (same for all domain controllers). The Logon server will be resolved from the %PATH% system variable of the custom add-in and added to the beginning of the LogFileName. Use the following syntax:</p> <p>LogFileName=WRITE_SHARE\ExportProfile.log</p>
AttemptNumber	The program will attempt to run for the same account this number of times. After each run, the program increases its own counter in the registry and stops when the value given in the AttemptNumber parameter is reached. If the program is successful, it sets the counter in the registry to the value of AttemptNumber directly. The default is 0 .
UseOldNames	<p>One of the following:</p> <p>0 - Assume the migrated user names changed and use SIDHistory to establish account mapping. This is the default.</p> <p>1 - Assume the migrated user names didn't change.</p>

Parameter File Example

Below is an example of an ExportProfile.ini file:

[Task]

```
Domain=trust
LogFileName=d:\ExportProfile.log
AttemptNumber=8
UseOldNames=1
```

ExportProfile Return Codes

The following program return codes are set depending on the work results.

Return Code	Description
-2	Unknown exception.
-1	Unknown error.
0	Success
1	Not a Windows NT/2000 OS.
2	Unrecognized or missing INI file.
3	No domain controllers found.
4	No new account found.
5	Current and new profile paths are equal.
6	Error accessing the registry.
7	MiniProfileMigration value is set for the target profile—the link hasn't changed.
9	Subsequent run for the same profile.
10	Failed to load a required DLL.
11	Failed to retrieve the current user's SID.
12	Failed to find the current profile. ³
13	Failed to find the new account at the target domain.
14	The resolved new SID is the same as the old SID.
15	Failed to copy the profile path.

Format of the ExportProfile Log File

The log file contains records for each run. The fields of each record are tab-delimited.

The records contain the following data:

```
Hour:Minutes:Seconds
```

```
Day/Month/Year
```

```
Computer
```

Account

Profile

Error code

Error Message

Auxiliary Error Message

Key

\\Domain Controller

Here Profile is the logon profile of the Account, Key is the key in the registry for the migrated account, and Domain Controller is the computer used to obtain the Key.

ChangeProfile

This utility associates the profile of the user currently being migrated with the source account. As a result, the same profile will be shared between the source and target accounts.

i **NOTE:** The utility links the new account to the old profile. However, it does not grant the new account the permissions needed to use the profile. This has to be done manually or with other Migration Manager components. See the *Resource Updating* section of the **Migration Manager for Active Directory User Guide** for details.

Using ChangeProfile

ChangeProfile uses the following command line syntax:

```
ChangeProfile.exe IniFileFull
```

where IniFileFull is the full path to the parameter file. The format of the parameter file is described in the [Format of the ChangeProfile Parameter File](#) topic.

The program accepts either a full UNC path to the parameter file or a relative path. If the file is specified in relative format, ChangeProfile assumes the file is located in the Windows directory—not in the current directory. To work around this limitation, use the following syntax from a logon script or a batch file:

```
ChangeProfile.exe %0\..\IniFile
```

Or use the following syntax from the command line:

```
ChangeProfile.exe .\IniFile
```

Examples

```
> ChangeProfile.exe ChangeProfile.ini
> ChangeProfile.exe %LOGONSERVER%\READ_SHARE\
ChangeProfile.ini
```

If some of the parameters are not specified in the INI file, the default settings are used. When the program runs with its default settings, it does the following, depending on the number of old SIDs found in the SIDHistory:

Number of Old SIDs	Description
number of old SIDs = 0	Marks execution in the registry and quits.
number of old SIDs =	Changes the link to the found profile, marks execution and performs an immediate

Number of Old SIDs	Description
1	logoff.
number of old SIDs > 1	Marks execution in the registry and quits.

Marking of the execution is accomplished by setting two keys in the registry:

```
[HKEY_CURRENT_USER\Software\Aelita\RegUtil\ChangeProfile]
```

```
AttempNumber=Number of last attempt
```

and

```
[HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\
```

```
CurrentVersion\ProfileList\TextSID] MiniProfileMigration=1
```

Another tool (ExportProfile) sets the value of MiniProfileMigration to 2.

When the link is successfully changed, the program sets the value of **AttemptNumber** to the value given in the parameter file and saves the old link under a new value called ProfileImagePath.BAK. An administrator or a power user could revise this value to locate the old profile folder and remove the folder to clear disk space being occupied by unused profiles.

If the link to the new (migrated) account is not successfully changed, the **AttemptNumber** value is increased by 1. When it reaches the maximum value contained in the parameter file, the program stops attempting to change the profile link.

Format of the ChangeProfile Parameter File

```
[Task]
```

```
DisableLogoff={0|1}
```

```
DisableMessage={0|1}
```

```
LogoffMessage=Message text
```

```
LogoffMessageYN={0|1}
```

```
OnDuplicatedSid=URL
```

```
TryToLoadProfile={0|1}
```

```
DialogPosX=number
```

```
DialogPosY=number
```

```
SelectProfile={0|1}
```

```
SleepTime=number
```

```
LogoffTimeout=number
```

```
AttemptNumber=number
```

LogFileName=%LOGONSERVER%\WRITE_SHARE\ChangeProfile.log

Explanation:

Parameter	Description
DisableLogoff	One of the following: 0 - Initiate logoff after the profile link has been changed. This is the default. 1 - Do not initiate logoff, even if the profile link has been changed for the currently logged-on account.
DisableMessage=0	One of the following: 0 - Display a message box prompting for a logoff if the profile link has been changed successfully. This is the default. 1 - Do not show a message box prompting for a logoff, even if the profile link has been changed successfully.
LogoffMessage=...	One of the following: 0 - Include an OK button in the message box. If this parameter is 0, the DisableLogoff parameter is ignored. This is the default. 1 - Include Yes and No buttons in the message box. Logoff occurs only if the user clicks Yes .
OnDuplicatedSID=URL	Use the URL address specified if there are duplicates in SIDHistory (this is considered to be an error if SelectProfile is set to 0 and no profile re-mapping is done.) By default the value of OnDuplicatedSID is empty.
TryToLoadProfile=0	One of the following: 0 - Bypass loading the profile before changing the profile link. This is the default. 1 - Load the profile using the RegLoadHive API before changing the profile link. The link will be modified only if the profile can successfully be loaded. See the Remarks topic.
DialogPosX=N	Set the horizontal coordinate in pixels of the top-left corner of the dialog boxes to be shown to N. The default is 0.
DialogPosY=N	Set the vertical coordinate in pixels of the top-left corner of the dialog boxes to be shown to N. The default is 0.
SelectProfile=0	One of the following: 0 - Treat duplicates in SIDHistory as an ambiguous situation and forward the user to the URL given in the OnDuplicatedSID parameter. This is the default. 1 - Show a select dialog box allowing the user to pick one of the available old profiles to set as the current profile if there are duplicates in SIDHistory.
SleepTime=N	Wait N seconds before either showing a logoff message box or logging off, depending on other settings. The default is 0.
LogoffTimeout=N	Have the logoff prompt dialog box wait N seconds for the user to confirm his or her choice. After the time elapses the logoff is performed.

Parameter	Description
	The default is 0.
AttemptNumber	<p>The program will attempt to run for the same account this number of times. After each run the program increases its own counter in the registry and stops when the value given in this parameter is reached. If the program is successful, it sets the counter in the registry to this value directly.</p> <p>The default is 0.</p>
LogFileName	<p>The ERRORLEVEL code description will be written to this file. This parameter can contain environment-variable strings to dynamically control the location of the log file. For example:</p> <p>%LOGONSERVER%\WRITE_SHARE\ ChangeProfile.log %UserProfile%\ChangeProfile.log</p> <p>If the %LOGONSERVER% variable is not supported, then you can specify a path to a share with write access (same for all domain controllers). The logon server will be resolved from the %PATH% system variable of the custom add-in and added to the beginning of the LogFileName. Use the following command:</p> <p>LogFileName=WRITE_SHARE\ChangeProfile.log</p>

Parameter File Example

Below is an example of a ChangeProfile.ini file:

```
[Task]
DisableLogoff=1
DisableMessage=0
LogoffMessage="Please log off or restart your computer"
LogoffMessageYN=0
;OnDuplicatedSid=URL
TryToLoadProfile=0
DialogPosX=150
DialogPosY=150
SelectProfile=1
SleepTime=10
LogoffTimeout=40
AttemptNumber=5
LogFileName=\\server\share\ChangeProfile\
ChangeProfile.log
```


ChangeProfile Return Codes

The following program return codes are set depending on the work results.

Return Code	Description
-2	Exception raised.
-1	Unknown error.
0	No links to an old profile were found.
1	Not a Windows NT/2000 OS.
2	Subsequent run for the same profile.
3	Duplicate SIDs exist in SIDHistory.
4	Profile link has been modified. User refused to logoff when prompted.
5	Profile link has been modified followed by user logoff.
6	Error while using registry API.
7	New and old SIDs already use the same profile.
8	Cannot load profile with TryToLoadProfile set to 1.
9	MiniProfileMigration value is set for this profile.
10	Error while calling the API functions.
11	Link to the profile has been changed with no logoff because it is disabled in the parameter file.
12	Argument (INI) file not found.
13	User canceled selection of profile.

Format of the ChangeProfile Log File

The log file contains records for each run. The fields of each record are tab-delimited.

The records contain the following data:

Hour:Minutes:Seconds

Day/Month/Year

Computer

Account

Profile

SIDsInHistory

Error

where Profile is the logon profile of the Account and Key is the key in the registry for the migrated account.

A log file indicating the result of a run is created on the target workstation. The path of the log file is as follows:

Remarks

The following problem arises if an account is used for purposes such as running a Windows 2000 service or scheduled task.

Conditions

SRC\User is the old account.

TARGET\User is the new account.

On workstation WST there is a service running under the SRC\User account.

TARGET\User logs on to WST and ChangeProfile.exe is executed.

Results

TARGET \User gets logged off by the utility and is assigned a new profile instead of the old one, and the procedure is cycled out.

Explanation

When TARGET \User logs on, Windows decides to use the profile of SRC\User. The procedure is unsuccessful because the profile is already in use by the service (the same thing will happen if RegLoadHive fails for any other reason). Windows decides to create and assign a new profile to the user.

Solution

1. Set the TryToLoadProfile option to 1 in the parameter file. The program will abort with the 'Cannot load profile' error (see error codes).
2. The program creates the registry value MiniProfileMigration (REG_DWORD) under the following path:

HKLM\Software\Microsoft\Windows NT\CurrentVersion\ProfileList\TextSID

where TextSID is the textual SID of the currently logged on user. This prevents the program from cycling.

RegWalker

When you collapse multiple domains or consolidate servers, a mass update of clients should take place to reflect name changes. The Windows operating system and third-party applications tend to store domain and server names in multiple locations in the registry database.

The **RegWalker** is an advanced tool that significantly reduces the cost of client update through the automation of mass-registry analysis and reconfiguration. Whether you change a server's name, move data to a different file or application server, or migrate to a different domain, RegWalker significantly decreases total reconfiguration costs by ensuring zero end-user downtime during the transition.

The **RegWalker** is a command-line tool used to perform most of the typical operations with the registry on computers running Windows 9x/NT/2000/XP/2003 operating systems. Such operations include search and replace of text data, copying, moving and deleting of the registry keys and values, and registry keys and values text data output to the external file.

In Windows NT/2000, the operations that change the registry require a local administrator's level of access. In Windows 9x, the operations on the registry are performed regardless of the user's privileges.

Using RegWalker

RegWalker can be started from a command prompt or from user logon scripts, and the command line syntax and file path interpretation differs accordingly. You can use the migrated users map file generated by Migration Manager to automatically find corresponding source and target users regardless of the start mode you choose for RegWalker.

Command Prompt Start Mode

In command prompt start mode, RegWalker interprets the paths to the parameter file and other necessary files as either relative or UNC paths. In this mode RegWalker can process remotely a number of specified computers. To start RegWalker from the command prompt, use the following command:

```
RegWalker.exe Filename.ini
```

where `Filename.ini` is the full path to and the name of the parameter file.

Logon Script Start Mode

In logon script start mode, RegWalker interprets the paths to the parameter file and other necessary files as either relative to the domain controller the user logs on to or UNC paths.

To start RegWalker from a logon script, use the following command:

```
RegWalker.exe "[\\server_name\]share_name\Filename.ini" -s
```

where:

- `\\server name` is an optional parameter. If `\\server name` is not specified, it is replaced with the name of the DC that the user logs on to.

- Filename.ini is the parameter file.

Format of the RegWalker Parameter File

```
[RegWalker]
DomainName = CDTEST
MapFile = vmover.ini
TaskFile = task.xml
OutputPath = Results
ComputerListFile = computers.lst
DateFormat = "yy '-'MMM-'dd"
TimeFormat = "h':'mm':'ss tt"
{User defined variable name}=user defined variable value
```

Explanation:

Parameter	Description
DomainName	Default domain name, which is used for processing computers running Windows 9x and for resolving the computer names if the domain name is not specified in the ComputerListFile parameter.
MapFile	Full path and the name of the migrated users map file.
TaskFile	Full path and the name of the task description file. This file is described in further detail below.
OutputPath	Path to the folder where output and log files will be created. If this parameter is not specified, files are created in the folder where RegWalker is located.
ComputerListFile	Full path and the name of a file with a list of users and computers for which registries will be processed remotely.
DateFormat	The way the date will look in the output and log files.
TimeFormat	The way the time will look in the output and log files.
{User defined variable name}	The name and value of a user-defined variable.

Parameter File Example

In command prompt start mode, the only required parameter is the path to the task file. For logon script start mode, a path to the share for output files is also required:

```
[RegWalker]
TaskFile = task.xml
OutputPath = netlogon\Results
```

Date and Time Display Formats

A DTP format string consists of a series of elements that represent a particular piece of information and define its display format. The elements will be displayed in the order they appear in the format string.

Date and time format elements will be replaced by the actual date and time. They are defined by the following groups of characters:

Element	Description
"d"	The one- or two-digit day.
"dd"	The two-digit day. Single-digit day values are preceded by a zero.
"ddd"	The three-character weekday abbreviation.
"dddd"	The full weekday name.
"h"	The one- or two-digit hour in 12-hour format.
"hh"	The two-digit hour in 12-hour format. Single-digit values are preceded by a zero.
"H"	The one- or two-digit hour in 24-hour format.
"HH"	The two-digit hour in 24-hour format. Single-digit values are preceded by a zero.
"m"	The one- or two-digit minute.
"mm"	The two-digit minute. Single-digit values are preceded by a zero.
"M"	The one- or two-digit month number.
"MM"	The two-digit month number. Single-digit values are preceded by a zero.
"MMM"	The three-character month abbreviation.
"MMMM"	The full month name.
"t"	The one-letter AM/PM abbreviation (for example, AM is displayed as "A").
"tt"	The two-letter AM/PM abbreviation (for example, AM is displayed as "AM").
"yy"	The last two digits of the year (for example, 1996 would be displayed as "96").
"yyyy"	The full year (for example, 1996 would be displayed as "1996").

To make the information more readable, you can add body text to the format string by enclosing it in single quotes. Spaces and punctuation marks do not need to be quoted.

i | **NOTE:** Characters other than those listed in the table above that are not delimited by single quotes will result in unpredictable display by the DTP control.

For example, to display the current date with the following format:

Today is: 04:22:31 Tuesday Dec 12, 2002

the format string is

'Today is: 'hh':'m':'s dddd MMM dd', 'yyyy

To include a single quote in your body text, use two consecutive single quotes. For example, suppose you want output that looks like the following :

Don't forget Dec 12, 2002

You would use the following format string:

'Don"t forget' MMM dd', 'yyyy

It is not necessary to use quotes with the comma, so the following string is also valid and produces the same output:

```
'Don"t forget' MMM dd, yyyy
```

Task Description File Format

The task description file specifies the type and order of the operations that will be performed with the registries on the computers. The format of the file complies with the XML standard except for the following:

- Tag and attribute names are case insensitive.
- Only UNICODE character coding is used.

All operations are specified with tags and parameters and are set with corresponding attributes.

The top-level elements are **OS** and **BaseKey**. All relative registry key paths are relative to the path specified in the **Name** attribute of the **BaseKey** element.

The full path to the registry must begin with the registry hive name

(**HKEY_CLASSES_ROOT**, **HKEY_CURRENT_CONFIG**, **HKEY_CURRENT_USER**, **HKEY_LOCAL_MACHINE**, **HKEY_USERS**) and can be preceded by the computer name (e.g., `\\Server\HKEY_CLASSES_ROOT`).

Operations specified in the task file are executed in the following order:

1. All operations defined before the first **Find** tag that do not depend on the search results
2. Once the first **Find** tag is reached, all find and replace operations within one **BaseKey** element
3. Operations that depend on the search results in the order the specified conditions are met
4. All operations defined after the **Find** tag in the order of appearance

CAUTION: Only the following named entities can be used in the task description file: `"`, `&`, `<`, `>`. Other special characters must be specified with their decimal or hexadecimal code (for example, `` or ``).

The following is a detailed description of the task file elements, their attributes, and values.

OS

Specifies the operating system to work with. All operations defined by the nested elements will be performed on computers with the specified operating system.

Attributes:

Name	Mandatory	Purpose	Values
Type	Yes	Specifies the OS type for which the specified operations will be performed	Winnt - Windows NT 3.51, Windows NT 4.0, Windows 2000 and later. Win - Windows 95, Windows 98, or Windows Me

Possible nested elements:

BaseKey

BaseKey

Specifies a registry hive name to work with. All operations defined by the nested elements will be performed on the specified registry hive and its subtrees. Relative registry key paths are relative to the path specified in this element.

Attributes:

Name	Mandatory	Purpose	Values
Name	Yes	Specifies the base path to the registry key. Most of the operations are performed with paths relative to the base path.	You can precede the path to the base registry key with the NetBIOS name of the computer.
Subkey	No	Specifies the depth of the search operation.	1 - Search among all subtree keys. This is the default attribute's value 0 - Search among current registry folder keys.

Possible nested elements:

Group, Find, Replace, Copy, Move, Create, Delete, Setval, Renval, Delval, CopyVal, Output

Group

Combines a number of search operations. The replace operation will be performed only if all search operations within the group have been successful.

No attributes

Possible nested elements:

Find, Replace

Find

Defines the search operation. This element can combine other elements that depend on the search results.

Attributes:

Name	Mandatory	Purpose	Values
String	Yes	Search string.	Symbols
Keys	No	Specifies whether to search among key names.	1 - Search among key names 0 - Do not search among key names
Values	No	Specifies whether to search among value names.	1 - Search among value names 0 - Do not search among value names
Data	nO	Specifies whether to search among string data.	1 - Search through data 0 -Do not search through data*

Name	Mandatory	Purpose	Values
WholeString	nO	Specifies whether the search string must be matched exactly or can be part of a longer string.	<p>1 - The search is successful if the search string exactly matches the found string.</p> <p>0 - The search is successful if the search string exactly matches the found string or the found string is a part of a larger string.*</p>
MatchCase	nO	Distinguishes between uppercase and lowercase characters. Note that this attribute is ignored when searching through values and keys.	<p>1 - The case of characters is taken into account.</p> <p>0 - The case of characters is not taken into account*</p>

i **NOTE:**To enumerate and write to the output file all values of a key, specify an empty line in the String attribute and set the Keys attribute to 1, as follows:
`<find string="" keys="1">`

Possible nested elements:

i **NOTE:**If the string expressions specified in the nested elements are successfully found, then the {FoundPath}, {FoundKey}, {FoundValue}, {FoundData} variables are replaced with relative path, registry key name, registry value name, and found value data values, respectively.

Copy, Move, Create, Delete, Setval, Renval, Delval, CopyVal, Output

Replace

Defines the search and replace operations in the registry values' data. This element can combine other elements that depend on the results of the search operation.

Attributes:

Name	Mandatory	Purpose	Values
String	Yes	Search string	Symbols
With	Yes	Replacement string	Symbols
WholeString	No	Specifies whether the search string must be matched exactly or can be part of a longer string	<p>1 - The search is successful if the search string exactly matches the found string.</p> <p>0 - The search is successful if the search string exactly matches the found string or the found string is a part of a longer string.</p>
MatchCase	No	Distinguishes between uppercase and lowercase characters	<p>1 - The case of characters is taken into account.</p> <p>0 - The case of characters is not taken into account.*</p>

Possible nested elements:

Copy, Move, Create, Delete, Setval, Renval, Delval, CopyVal, Output

i **NOTE:** If a found registry value matches a number of Replace operations within one BaseKey element, only the first Replace operation will be performed.

If the string expressions specified in the nested elements are successfully found, the {FoundPath}, {FoundKey}, {FoundValue}, {FoundData} variables are replaced with relative path, registry key name, registry value name, and value data values, respectively.

If a search string matches a number of values within the Multi-String values in the registry, nested operations will be performed for each string.

Create

Specifies the new registry subtree creation.

Attributes:

Name	Mandatory	Purpose	Values
String	Yes	Relative path and the name of the created subtree.	Symbols
Class	No	Class of the created subtree	Symbols

Cannot contain nested elements.

Copy

Specifies the copy operation of the registry subtree.

Attributes:

Name	Mandatory	Purpose	Values
Src	Yes	Relative path to the registry subtree that must be copied	Symbols
Dest	Yes	A path and a name of the registry subtree copy that will be created	Symbols
Permissions	No	Specifies whether to copy permissions from the source registry subtree keys	1 - Permissions are copied from the source subtree keys. 0 - Default permissions are set.

Cannot contain nested elements.

Move

Moves the registry subtree to a specified location.

Attributes:

Name	Mandatory	Purpose	Values
Src	Yes	Relative path to the registry subtree that must be moved	Symbols
Dest	Yes	A path and a name of the registry subtree copy that will be created	Symbols
Permissions	No	Specifies whether to copy permissions from the source registry subtree keys	1 -Permissions are copied from the source key. 0 - Default permissions are set.

Cannot contain nested elements.

Delete

Deletes the registry subtree.

Attributes:

Name	Mandatory	Purpose	Values
Name	Yes	Relative path to the registry subtree that must be deleted	Symbols

Cannot contain nested elements.

SetVal

Sets the registry key value.

Attributes:

Name	Mandatory	Purpose	Values
Key	Yes	Relative path to the registry key.	Symbols
Name	Yes	Value name	Symbols
Value	Yes	Value to be set	Symbols

Cannot contain nested elements.

i | **NOTE:** If the specified registry key name is not found it will be created. To set an unnamed default value for the key, leave the **Name** attribute blank.

DelVal

Deletes the registry key value.

Attributes:

Name	Mandatory	Purpose	Values
Key	Yes	Relative path to the registry key.	Symbols
Name	Yes	Value name	Symbols

Cannot contain nested elements.

i | **NOTE:** To delete the unnamed default value, leave the **Name** attribute blank.

RenVal

Renames the registry key value.

Attributes:

Name	Mandatory	Purpose	Values
Key	Yes	Relative path to the registry key.	Symbols
Name	Yes	Value name	Symbols
NewName	Yes	New value name	Symbols

Cannot contain nested elements.

i | **NOTE:** To rename the unnamed default value, leave the **Name** attribute blank.

CopyVal

Copies the registry key value.

Attributes:

Name	Mandatory	Purpose	Values
Src	Yes	Relative path to the registry key.	Symbols
Dest	Yes	Path and name of the registry key to which the value will be copied	Symbols
Name	Yes	Value name	Symbols
NewName	No	Copied value name	Symbols

Cannot contain nested elements.

i | **NOTE:** To copy the unnamed default value, leave the **Name** attribute blank.

Output

Specifies the text information to be written to the output file.

Attributes:

Name	Mandatory	Purpose	Values
String	Yes	Text to be written to the output file	Symbols

Cannot contain nested elements.

String Variables

You can use predefined string variables in the attributes' string values. The exact values that such variables take on depend on the result of the previous search operation, migrated users map file processing, and some other parameters.

i | **NOTE:** To type braces ({}) in the attribute's value, use double braces, as follows:
 {{{}}

Predefined String Variables

Name	Description
{UserName}	<p>Processed user name. This variable can take on different values depending on the RegWalker start mode (logon script or command prompt):</p> <ul style="list-style-type: none"> • Current user name when started from logon script or if the ComputerListFile is not specified • User from ComputerListFile if it is specified
{UserRID}	<p>RID of the processed user. This variable can take on different values depending on the RegWalker start mode (logon script or command prompt):</p> <ul style="list-style-type: none"> • Current user RID when started from logon script or if the ComputerListFile is not specified • RID of the user from ComputerListFile if it is specified. <p>Applicable for Windows 9x users only if they are present in the migrated users map file. If the domain name is not specified in the file, the domain name from INI file will be used.</p>
{Domain}	<p>NetBIOS domain name of the processed user:</p> <p>Current user domain when started from logon script or if the ComputerListFile is not specified Domain name of the user from ComputerListFile if it is specified For Windows 9x users, if the domain name is not specified in the file, the name from INI file will be used.</p>
{DomainSID}	<p>SID of the processed user domain:</p> <ul style="list-style-type: none"> • Current user domain SID when started from logon script or if the ComputerListFile is not specified <p>Domain SID of the user from ComputerListFile if it is specified</p>
{SrcUserName}	<p>User name before the migration. The value is the result of the migrated users map file processing.</p>
{SrcUserRID}	<p>User RID before the migration. The value is the result of the migrated users map file</p>

Name	Description
	processing.
{SrcDomain}	NetBIOS name of the domain the user was migrated from. The value is the result of the migrated users map file processing.
{SrcDomainSID}	SID of the domain the user was migrated from. The value is the result of the migrated users map file processing.
{ComputerName}	NetBIOS name of the computer that will be processed: <ul style="list-style-type: none"> • Current user's computer name when started from logon script or if the ComputerListFile is not specified <p>Computer name from ComputerListFile if it is specified</p>
{BaseKey}	Path to the base registry key. The value is specified by the Name attribute of the BaseKey element.
{FoundPath}	Path to the found registry key. The value is changed if the search operation is successful and can be used within Find and Replace elements only.
{FoundKey}	Name of the found registry key. The value is changed if the search operation is successful and can be used within Find and Replace elements only.
{FoundValue}	Name of the found registry key value. The value of the variable is changed if the search operation is successful and can be used within Find and Replace elements only.
{FoundData}	Found in the registry data. The value of the variable is changed if the search operation within the registry value names and data is successful and can be used within Find and Replace elements only.
{ReplacedData}	Data's state after replace operation. The value of this variable is changed if replace operation was successful and can be used within Replace elements only.
{Date}	Current date. The date format is specified in the parameter file.
{Time}	Local time. The time format is specified in the parameter file.

In addition to the predefined variables, you can use custom variables, which must be specified in the parameter file. To specify a custom variable, create a key with the variable name and assign a value to it. For example:

```
{NewProxyName} = proxy1
{NewProxyIP} = 10.0.0.10
```

RegWalker Return Codes

The following program return codes are set depending on the work results.

Return Code	Description
0	Success.
1	Unknown error.
2	Invalid startup parameters.
3	Parameter file was not found.
4	Could not resolve domain controller name.
5	Migrated users map file was not found
6	Migrated users map file format error.
7	Computer list file was not found.
8	Task description file was not found or invalid.
9	Can't write to log file.

Computer List File Format

This file can be used in the command prompt start mode to process the specified computers remotely. The file must be a plain text file containing a list of computers and users in the following format:

```
Computer_name;[domain\]user_name
```

The selected operations will be performed on the computers specified in the computer list file if the path to the registry subtree does not contain the computer name.

The string variable {ComputerName} is replaced with the currently processed computer name. {UserName} and {Domain} variables are replaced with user and domain names taken from the list, and {DomainSID} and {UserRID} variables are replaced with the SID and RID of the specified domain respectively.

If a domain name is not specified in the computer list file, the domain name will be taken from the parameter file. If the parameter file does not contain the domain name, the domain name of the user account under which RegWalker runs will be used.

The user that starts the process must have sufficient access rights on the registry.

Computer List File Example

```
ALEX;RESEARCH\Alex
```

```
EDITOR;RESEARCH\Betty
```

```
MARTIN;RESEARCH\Martin
```

```
SERVER;RESEARCH\Serge
```

Migrated Users Map File

This file is generated by Resource Updating Manager and contains source-target user mappings. The name of the file is specified in the Resource Updating Manager; the default name is vmover.ini. RegWalker can work with uncompressed files only. This file is used by RegWalker to find correspondence between source and target users.

Format of the RegWalker Output File

The output files contain information specified by the Output operation. Each file name consists of the domain name, computer name, and user name, delimited by semicolons.

Format of the RegWalker Log File

The log file contains tab-delimited text records that represent information on work results and errors description. The number and the names of log files depend on the RegWalker start mode.

In command prompt start mode, RegWalker logs its activities into one log file, regwalker.log.

In logon script start mode, a separate log file is created for each processed user. The name of each log consists of the domain name, computer name, and user name, delimited by semicolons.

Each record in the log contains the following tab-delimited data:

- Date
- Time
- Computer name
- Record type (Information/Error)
- Path to the registry key that caused the error
- Error code
- Error description

RegWalker Usage Examples

The following are several RegWalker usage examples.

Example 1

RegWalker can be used to automate the post-migration update of Microsoft Client for Netware settings on the user desktop computers.

The Microsoft Client for Netware settings are stored in the following registry key under the user SID:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\NWCWorkstation\Parameters\Option\<SID>
```

When users are migrated, their SIDs change; therefore the users do not have access to these settings. Regwalker creates the appropriate Novell NDS Tree and Context registry settings on selected users' computers for the migrated users enumerated in the migrated users map file generated by DMW (default name vmover.ini).

In this example RegWalker is started from the command prompt. To use RegWalker:

1. In a text editor such as Notepad, create a parameter file for RegWalker. In the file specify the paths to the migrated user map file (the default name is vmover.ini) and the RegWalker task file:

```
[RegWalker]
TaskFile="Novell Settings Registry Update.xml"
OutputPath=Reports
```

MapFile=vmover.ini

Save the file under the name **regwalker.ini**.

2. In a text editor, create a task file. You can copy the provided example directly to the file and save it as Novell Settings Registry Update.xml.

Task file example:

```
<!--
The following Novell Settings Registry file tells RegWalker to the Microsoft
Client for Netware settings on the user computers in the following registry key
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\
Services\NWCWorkstation\Parameters\Option\<SID>
-->
<os type="winnt">
<basekey name="HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\
Services\NWCWorkstation\Parameters\Option" subkeys="0" >
<find string="{SrcDomainSid}-{SrcUserRid}" keys="1" wholestring="1" >
<copy src="{FoundKey}" dest="{BaseKey}\{DomainSid}-{UserRid}" />
<output string="{ComputerName}&#9;{UserName}&#9;Novell settings have been
moved." />
</find>
</basekey>
</os>
```

3. Run RegWalker.exe from a command prompt, specifying the path to the ini file you created:

```
RegWalker.exe regwalker.ini
```

RegWalker will process the registries on the selected computers and copy the necessary settings from the source user profiles to the corresponding target user profiles.

i | **NOTE:** In this example, all files are located in the same folder together with regwalker.exe file.

Example 2

In this example RegWalker is started from the command prompt and used to update the print server name in the users' profiles. Take the following steps:

1. In a text editor such as Notepad, create a parameter file for RegWalker. In the file specify the paths to the RegWalker task file and the output path, and define two variables:

```
[RegWalker]
TaskFile = task2.xml
OutputPath = Results
ComputerListFile = domaincomputers.lst
{OldPrintServer}="\\PrintServer"
{NewPrintServer}="\\PrintServer2"
```

Save the file under the name **regwalker.ini**.

2. In a text editor, create a task file. You can copy the provided example directly to the file and save it as task2.xml.

Task file example:


```

<!--
The following task file tells RegWalker to change the name of the print server in
the user profile. RegWalker will change Server key value from {OldPrintServer} to
{NewPrintServer} in the following registry key
HKEY_USERS\"SID"\Printers\Connections\ "*"\
-->
<os type="winnt">
<basekey name="HKEY_USERS\{DomainSID}-{UserRID}\Printers" SubKeys="1" >
<group>
<find string="Connections" keys="1" WholeString="1" />
<find string="Server" values="1" WholeString="1" />
<replace string="{OldPrintServer}" with="{NewPrintServer}" WholeString="1" >
<output string="{Data}&#9;{Time}&#9;{ComputerName}&#9;Change printer setting for
user {UserName} rename server from server replaced from {OldPrintServer} to
{NewPrintServer}" />
</replace>
</group>
</basekey>
</os>

```

3. In a text editor, create a file with a list of computers to process and save the file as domaincomputers.lst. For example:

```

ALEX;RESEARCH\Alex
EDITOR;RESEARCH\Betty
MARTIN;RESEARCH\Martin
SERVER;RESEARCH\Serge

```

4. Run RegWalker.exe from a command prompt, specifying a path to the ini file you created:

```
RegWalker.exe regwalker.ini
```

RegWalker will process registries on the users' computers and change the old print server name to the new one in the users' profiles.

GetRsl

Every computer has a Registry Size Limit (RSL) set for it. It is a universal maximum for registry space that prevents an application from filling the paged pool with registry data. The total amount of space that can be consumed by registry data (hives) is restricted by the registry size limit.

When the registry size gets close to the limit, the operating system may prevent applications from modifying the registry. This may prevent Resource Updating Manager and VMover from updating profiles and registries on remote computers.

GetRsl displays the current registry size and the registry size limit for a remote computer. If the registry size is close to the limit, this might be the cause of some problems you experience. In this case, modify the computer's RSL and retry updating the computer.

Using GetRsl

GetRsl uses the following command line syntax:

```
GetRsl.exe ComputerName
```

where ComputerName is the name of the remote computer.

The utility produces output in the following format:

```
ComputerName RegistrySize RSL
```

The numbers are reported in megabytes, for example:

```
> GetRsl.exe Setanta
```

```
Setanta 24.3 69.3
```

OCSMigrator

The OCSMigrator utility helps you migrate OCS/LCS accounts after you migrate the corresponding Active Directory accounts. The utility also migrates contact lists including groups and block lists.

! **CAUTION:** The utility does not migrate Access Level settings for the contacts.

To map the accounts, the utility uses the vmover.ini file generated by Migration Manager. All migration parameters are specified in the command line.

OCSMigrator System Requirements

The **OCSMigrator** utility requires the following:

- Microsoft .Net Framework 2.0
- Microsoft Office Communications Server 2007 or Microsoft Live Communications Server 2005
- Microsoft SQL Server 2000 or higher

! **CAUTION:** All procedures described below refer to both Microsoft Office Communications Server 2007 and Microsoft Live Communications Server 2005 (unless the certain product is specified explicitly).

The SQL Server hosting migration database must be created manually before you run the utility and must be available for access both from the source and target OCS/LCS servers.

The account under which you run the utility must be a member of the RTCDomainUserAdmins group in the current domain (for LCS 2005) or a member of the RTCUniversalUserAdmins group (for OCS 2007).

Using OCSMigrator

The **OCSMigrator** utility package includes files as follows:

1. **OCSMigrator.exe** (version **8.6.0.1**) - OCS/LCS Migrator utility
2. **default.values.txt** - text file with object properties' default values; required for migration from LCS 2005 to OCS 2007 (see [Importing User Data](#) topic for details)

Using the Utility in Inter-Forest Migration

The utility currently supports for LCS to OCS account migration.

To migrate user accounts, take the following steps:

1. Export user data to the migration SQL database - for that, run the utility on the source server as described in the [Exporting User Data](#) topic.
2. Generate the mapping file (**Vmover.ini**) as described in the [Generating Vmover.ini File](#) topic.
3. Import user data as described in the [Importing User Data](#) topic; use the **–external** key. Wait until these changes take effect (you can check the **dbo.resource** table in LCS/OCS Database).
4. Import user data once again with OCSMigrator; use the **– external** key.
5. Update the contacts that point to migrated users - for that, run the utility locally on each server that keeps user accounts whose contacts must be updated, as described in the [Updating the Contacts That Point to Migrated User Accounts](#) topic.

! **CAUTION:** When supplying command line parameters to the utility, you should enter the same value for **<session name>** parameter regardless of the use case - export, import, update or cleanup.

i **NOTE:** During the first import, a contact list is usually imported only partially, so import should be run once again after LCS/OCS User Replicator processes the objects, and Active Directory replication is complete. During the consequent import, the following messages can appear in the log:

Put MSFT_SIPESUserSetting HRESULT:-1007911672. Exception.

These messages occur due to the attempts to re-create a user account in LCS/OCS; they can be safely ignored.

Using the Utility in Intra-Forest Migration

The utility currently supports for intra-forest migration from LCS to LCS.

To migrate user accounts, take the following steps:

1. Export user data as described in the [Exporting User Data](#) topic.
2. Generate the mapping file (**Vmover.ini**) as described in the [Generating Vmover.ini File](#) topic.
3. Delete source user accounts from LCS/OCS.

! **CAUTION:** Wait for LCS/OCS User Replicator to process the objects and for Active Directory replication to complete.

4. Import user data as described in the [Importing User Data](#) topic; use the **–external** key and the **–sourcesip** key.
5. Wait until these changes take effect (you can check the **dbo.resource** table in OCS Database).

! **CAUTION:** Wait for LCS/OCS User Replicator to process the objects and for Active Directory replication to complete.

6. Import user data once again with OCSMigrator; use the **– external** key and the **–sourcesip** key.

! **CAUTION:** When supplying command line parameters to the utility, you should enter the same value for **<session name>** parameter regardless of the use case - export, import, update or cleanup.

i **NOTE:** During the first import, a contact list is usually imported only partially, so import should be run once again after LCS/OCS User Replicator processes the objects, and Active Directory replication is complete. During the consequent import, the following messages can appear in the log:
Put MSFT_SIPESUserSetting HRESULT:-1007911672. Exception.
These messages occur due to the attempts to re-create a user account in LCS/OCS; they can be safely ignored.

Exporting User Data

To export user data to the migration database, do the following:

1. Copy the utility to the source OCS/LCS server.
2. Run the utility from the command line in the following format:

```
OCSMigrator.exe -export <session name> -log <logfile.txt> -domain <source domain or dc> -sql <SQL server> -db <Database> [-sqluser <SQL username> -sqlpass <SQL password>]
```

Where:

- **<session name>** - character string identifying current migration, for example: SOURCE_OCS_SERVER-TARGET_OCS_SERVER
- **<logfile.txt>** - log file name
- **<source domain or dc>** - name of the source domain or DC
- **<SQL server>** - name of the SQL server hosting the migration database
- **<Database>** - name of the migration database
- **<SQL username>** - account name to connect to the SQL Server (If not specified, SQL Integrated Security will be used)
- **<SQL password>** - account password

Generating Vmover.ini File

To generate the vmover.ini file, do the following:

1. In the Resource Updating Manager console, create a new collection.
2. Right-click on the collection and choose **Create Task | Processing**.
3. In the **Create Processing Task** wizard, specify the processing settings. Do not start the task.
4. Click on the **Tasks** tab in the right pane.
5. Right-click on the newly created task and select **Export Settings to File**.
6. Save the INI file in the desired location as *vmover.ini*.
7. Open the INI file in any text editor and verify that the settings are accurate and the file contains the desired objects only.
8. Finally, copy the vmover.ini file to the network share specified above.

CAUTION: If new users or groups are migrated after the file is exported, a vmover.ini file needs to be re-exported in order to contain the new users or groups.

NOTE: It is recommended to generate the Vmover.ini file for the utility usage with only one of the source-target domain pairs.

Importing User Data

To import user data from the migration SQL database to OCS/LCS, do the following:

1. Copy the utility to the target OCS/LCS server.
2. Run the utility from the command line in the following format:

```
OCSMigrator.exe -import <session name> -map <vmover.ini> -log <logfile.txt> -domain <target domain or dc> -sql <SQL server> -db <Database> [-sqluser <SQL username> -sqlpass <SQL password>] [-pools <pools mapping text file>] [-external] [-default <default.values.txt>] [-sourcesip] [-errorretry <number of retries>] [-errorwait <time interval>]
```

Where:

- **<session name>**—character string identifying current migration, for example: SOURCE_OCS_SERVER-TARGET_OCS_SERVER
- **<logfile.txt>**—log file name
- **<target domain or dc>** - name of the target domain or DC
- **<SQL server>**—name of the SQL server hosting the migration database
- **<Database>**—name of the migration database
- **<SQL username>**—account name to connect to the SQL Server (if not specified, SQL Integrated Security will be used)
- **<SQL password>**—account password
- **<pools mapping text file>**—text file containing pool pairs in the following format: **<source pool FQDN> = <target pool FQDN>**. Each pool pair you specify must be located in a separate line.

NOTE: This parameter is optional. If you do not specify it, the first enumerated pool will be used as a target pool.

- **external**—sets whether the source user contacts (pointing to the users of the other domains) must be added to the target user profile
- **<default.values.txt>**—text file containing default values of the properties that target objects have but source objects have not; this file is required for migration from LCS to OCS because source LCS objects miss some properties that target OCS objects should have.

The file must contain a list of values (one per line) in the following format: **<WMI class name>.<property name>=<value>**

For example:

```
MSFT_SIPESUserSetting.EnabledForEnhancedPresence=true
```

- **<-sourcesip>**—sets target user's SIP URI to the same value as source user's. This key facilitates intra-forest migration - if used with the utility, then the procedure of updating user data can be omitted.
- **<number of retries>**—the default number of retries in case of error is 5; you can set a higher value if you get many errors
- **<time interval>**—the default time interval between retries is 60000 milliseconds (that is, 1 minute); you can set a shorter interval if the import takes too long

The following contacts are added to the target user profile:

1. Contacts pointing to the target users which have matching source users included in the **vmover.ini** file.
2. If the **-external** key is used:
 - the source user contacts pointing to the user accounts from other domains, which are not involved in the migration process, and
 - the source user contacts (in the source domain) which are not included in mapping (without any contact modification)

There is no need to run the OCSMigrator utility for every source-target pair. To export users from all the forest pools to the temporary database, run the user data export on one of the servers. Next, run the user data import on one of the servers and specify the source-target OCS/LCS matching in the pool mapping file.

The **-pools** option is mandatory for the intra-forest migration, otherwise the target users can get into on of the source pools.

During the intra-forest migration an issue may occur when the target users are created, but the corresponding contacts are not processed. This is resulted by the delay in data replication between the OCS/LCS servers. To resolve the issue, wait until the newly created users are available in the OCS/LCS snap-in, and run the utility in the import mode once again on the target OCS/LCS server.

After the computer is moved to the target domain, the user must manually change the OCS server name in the OCS connection settings. This can be done automatically by logon script. The names of the internal and external OCS servers (for the Microsoft Office Communicator 2007) are stored, respectively, in the **ServerAddressInternal** and **ServerAddressExternal** values of the **HKEY_CURRENT_USER\Software\Microsoft\SharedUcClient** registry key.

Updating the Contacts That Point to Migrated User Accounts

If run with the parameters described in this section, the OCSMigrator utility replaces the migrated source users' SIPs (in the contact lists of all users) with the corresponding target users' SIPs. This operation should be performed right after importing target users' data (see above).

To update the contacts pointing to migrated users, do the following:

1. Copy the utility to the OCS server for which the contacts should be updated.
2. Run the utility from the command line with the following syntax:

```
OCSMigrator.exe -update <session name> -log <logfile.txt> -domain <domain or dc> -sql <SQL server> -db <Database> [-sqluser <SQL username> -sqlpass <SQL password>]
```

Where:

- **<session name>** - character string identifying current migration, for example: SOURCE_OCS_SERVER-TARGET_OCS_SERVER
- **<logfile.txt>** - log file name
- **<domain or dc>** - name of the domain or DC
- **<SQL server>** - name of the SQL server hosting the migration database
- **<Database>** - name of the migration database
- **<SQL username>** - account name to connect to the SQL Server (if not specified, SQL Integrated Security will be used)
- **<SQL password>** - account password

Clearing the Migration Database

To clear the migration database, run the utility with the following syntax:

```
OCSMigrator.exe -delete <session name> -log <logfile.txt> -sql <SQL server> -db  
<Database> [-sqluser <SQL username> -sqlpass <SQL password>]
```


InputBox

InputBox provides a graphical user interface to command-line utilities. It can be used with Migration Manager Resource Kit utilities or any other applications that accept their parameters from the command line.

Using InputBox

Use the following command-line parameters:

```
InputBox[.exe] [{/prompt:text | /p:text}] [{/title:caption | /t:caption}]  
{/command:argument | /c:argument} [{/remain | /r}] [{/default:value | /d:value}] [/  
{interactive | /i}]
```

```
/prompt: or /p:text
```

Specifies the message to be displayed in the dialog box.

```
/title: or /t:text
```

Specifies the string to be displayed in the title bar of the dialog box.

```
/command: or /c:parameter
```

The program you want to run and its command line arguments. Arguments may include %1, which will be replaced by the user input from the dialog box.

```
/remain: or /r
```

Don't close the command interpreter window after executing the command.

```
/default: or /d:value
```

Specifies the string to be displayed in the text box as the default response if no other input is provided.

```
/interactive: or /i
```

Show the command interpreter window.

Example

The following command runs the Netdom utility and prompts the user for a new name for the computer. You can write the command in a CMD file and send this file to users so they can rename their computers as they like without having to use the command line.

```
InputBox /p:" Please input new computer name" /t:"Rename computer" /d:%computername%  
/c:"netdom renamecomputer %computername% /newname:%1" /r /i
```

i | **NOTE:** If you run InputBox from a CMD or BAT file, use %%1 instead of %1, because %1 is interpreted as a parameter of the command file.

GPMCEXport

The GPMCEXport custom add-in allows administrators to migrate group policy objects (GPOs) when performing domain migration, as follows:

1. The custom add-in retrieves account mapping information from the migration project and stores it in a file that can be imported into Microsoft Group Policy Management Console (GPMC).
2. When migrating GPOs with GPMC, simply select the mapping file. All source security principals will automatically be translated into the corresponding target objects.

The following group policy settings contain security principals (users, security groups, and computers) and can be updated using the mapping file:

- Security policy settings of the following types:
 - User rights assignment
 - Restricted groups
 - System services
 - File system
 - Registry
- Advanced folder redirection policy settings.
- The GPO Discretionary Access Control List (DACL), if you choose to preserve it during a copy operation.
- The DACLs on software installation objects, if any. These DACLs are preserved only if the option to copy the GPO DACL is specified.

GPMCEXport allows GPMC to translate all of these settings during GPO migration. Thus, all settings within GPOs, GPO security filtering (restricting GPO application), and GPO delegation (permissions on the GPO objects) can be set to the correct target domain accounts.

GPO Migration Procedure

GPO migration consists of two steps:

1. Export account mapping from Migration Manager.
2. Migrate GPOs with GPMC.

Step 1. Export Account Mapping from Migration Manager for Active Directory

After you have completed the domain migration with Migration Manager for Active Directory, the migration project has account mapping information for all users, groups, and computers that were migrated.

Use the following procedure to export the mapping information:

1. Make sure that your current account has at least Read access to the migration project. You can check this by running the Migration Manager console.
2. In a Windows command prompt, change the directory to the Migration Manager 'Resource Kit' folder.
3. Run GPMCEExport.vbs with the following parameters:
4. GPMCEExport.vbs /host:computer /pdn:partitiondn [/port:number]

Where:

- host is the computer on which the migration project database is installed.
- pdn is the project partition ('CN=ProjectName').
- port is the port to be used to connect to the ADAM containing the database (389 by default).

For example:

```
GPMCEExport.vbs /host:MyServer /pdn:CN=Migration /port:389
```

GPMCEExport will produce the following files:

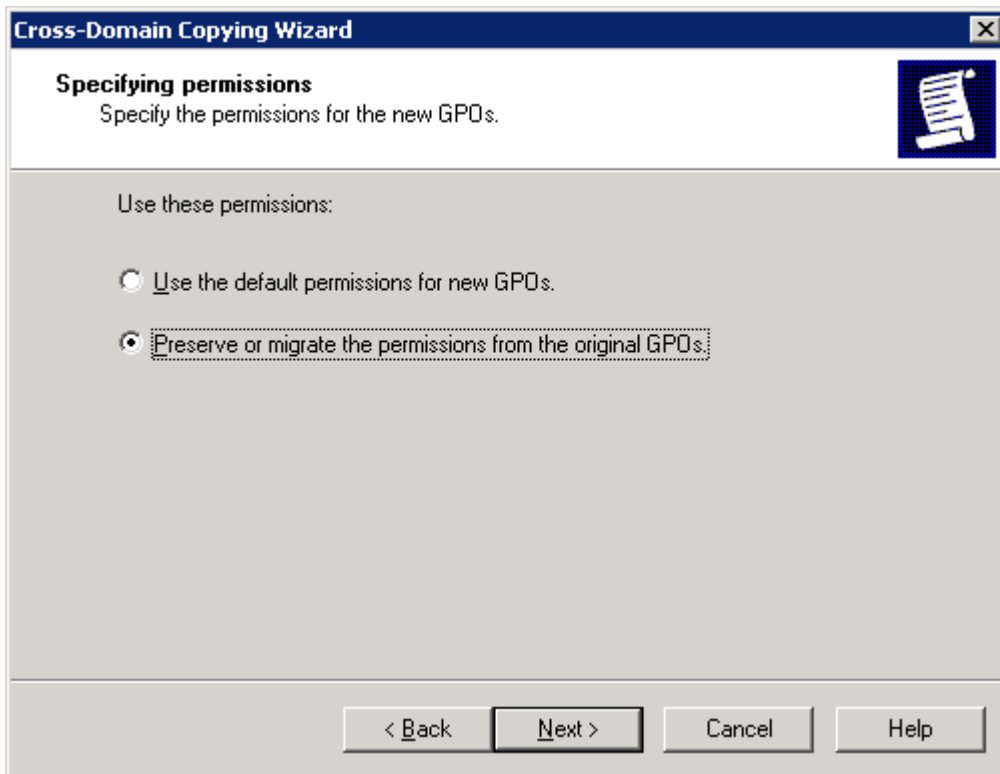
- GPMCEExport.log, the custom add-in log file.
- For each source/target domain pair, a file named SourceDomainName_TargetDomainName.migtable. If such a file already exists, it will be overwritten.

All files will be created in the folder in which the custom add-in is located.

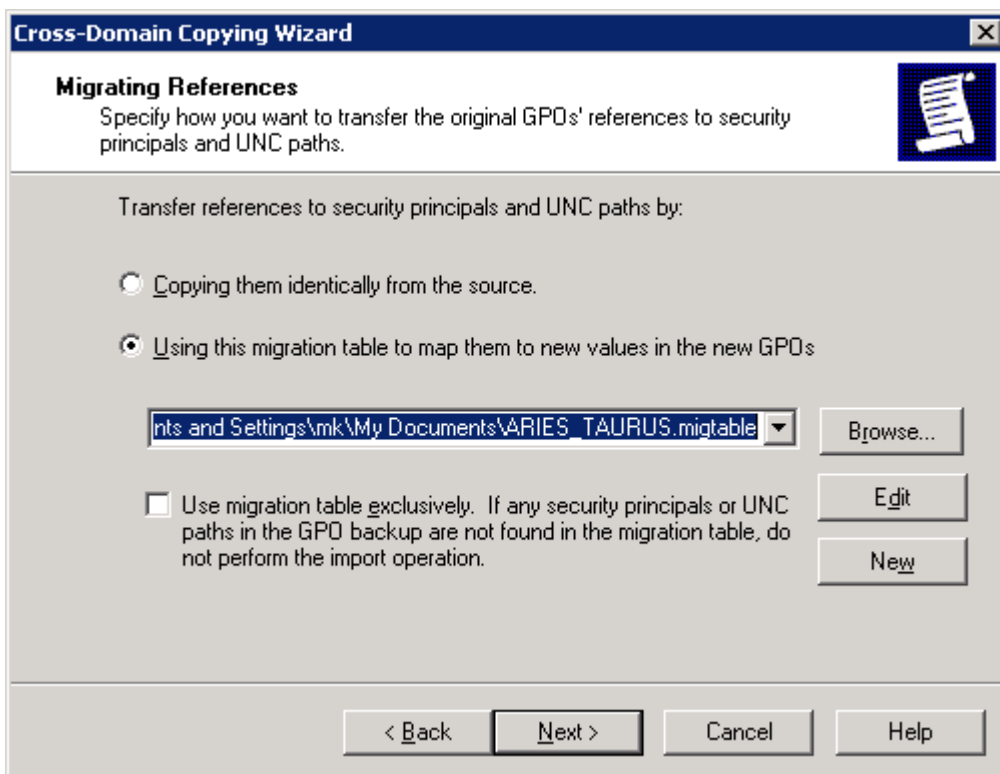
Step 2. Migrate GPOs with GPMC

Use the following procedure to migrate GPOs:

1. Run GPMC in the target forest (connect to a domain controller within the target forest).
2. Add the source forest.
3. Browse to the GPOs located in the Group Policy Objects node within the source domain.
4. Use the GPMC user interface to copy and paste (or drag and drop) the desired GPOs from the source domain to the target domain. The Cross-Domain Copying Wizard will start.
5. If you want the Security Filtering and Delegation lists to be migrated and translated with the Migration Manager mapping, choose the Preserve or migrate the permissions from the original GPOs option. Click Next to continue.



6. In the Migrating References step, click **Browse...** and select the mapping file created by the GPMCEExport custom add-in.



Finish the wizard. The GPOs will be migrated to the target forest. All the account settings that used to be set for the source accounts are translated to the corresponding target domain accounts.
After that you can link the GPOs to the appropriate domains, sites, and OUs in the target forest.

CertMgr

In Active Directory environments, users can use encryption certificates to give access to encrypted NTFS files. The certificates can be managed from the Certificates Microsoft Management Console (MMC) snap-in. If a source account certificate is not migrated to the target account, the user loses the corresponding capabilities, such as access to the encrypted file system. Certificates marked as exportable (which is the default option) can be migrated. The certificate migration can be done manually, with each user exporting and then importing the certificate from the Certificates snap-in. Alternatively, the CertMgr utility can be used to automate the procedure.

CertMgr Configuration

The CertMgr utility is installed to the Resource Kit folder. To export the certificates. Run the following command:

```
certmgr /mode:e
```

To import previously-exported certificates, run the following command:

```
certmgr /mode:i
```

Other options are taken from the CertMgr.ini file. You can create the file by following this template:

```
; ***** CertMgr.ini *****  
  
[CertMgr]  
CertDir=  
  
; The folder in which CertMgr will put the certificates during export  
; and from which certificates will be taken during import.  
; The default is the folder in which CertMgr is located.  
; You can specify a UNC (network) or local path.  
; CertMgr supports environment variables (e.g., CertDir=%SYSTEMROOT%\certs)  
; and relative paths (e.g., CertDir=..\certs or CertDir=.\certs).  
; If the folder does not exist, it will be created automatically.  
  
Exportable=yes|no
```

; Specify whether CertMgr should mark the certificates as exportable when it imports them.

! CAUTION: By default CertMgr marks the processed certificates as exportable (Exportable=yes).

[Log]

LogFileDir=

; Output folder for the CertMgr.log file.

; The default is the folder in which CertMgr is located.

; You can specify a UNC (network) or local path.

; CertMgr supports environment variables and relative paths,

; If the folder does not exist, it will be created automatically,

; If the file already exists, it will be appended.

Filter=-1

; Bit mask for log details level.

; ERROR = 1,

; WARNING = 0x2,

; INFORMATIONAL = 0x4,

; SUMMARY = 0x8,

; INTERNAL_ERROR = 0x10,

; TRACE_MSG = 0x20,

; The default is -1, which means that all the information is logged.

Using CertMgr

To export the certificates installed locally, run the following command:

```
certmgr /mode:e
```

To import previously-exported certificates on the local computer, run the following command:

```
certmgr /mode:i
```

To automate certificate export or import for multiple users, you can create logon scripts to run the command “certmgr /mode:e” for all users in the source domain, and “certmgr /mode:i” for users in the target domain.

For each user, CertMgr will create a certificates file in the folder specified in the CertMgr.ini file. The file will not be password-protected, so please use NTFS permissions to guarantee certificate safety.

If export is run twice for the same user, it will simply overwrite the file it previously created.

CertMgr replaces the certificate if it already exists in the store.

! CAUTION: Keep in mind that only one user account can own a certificate. Once a certificate is imported by the target account, it can no longer be used by the source account.

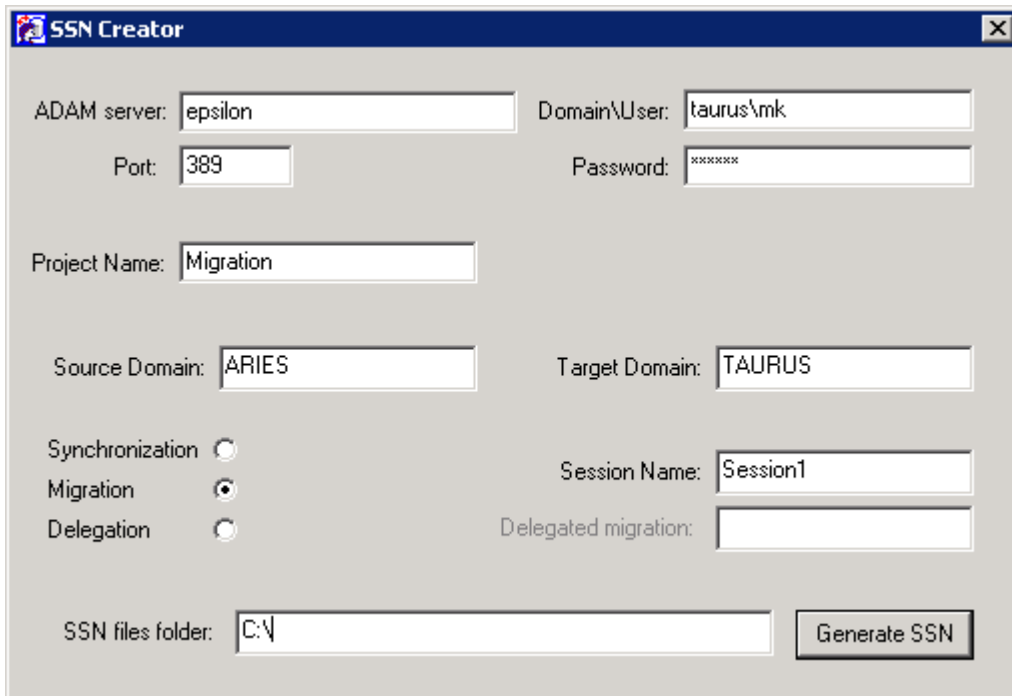
The tool does not delete the certificate files after import. Administrators need to delete the files themselves after they are no longer needed.

SSN Creator

The SSN Creator utility can export the migration account mapping to the Domain Migration Wizard session format. This format can be used to re-permission resources by other applications such as Quest Consolidator.

To export the mapping, take the following steps:

1. Run SSNExport from the Resource Kit folder.
2. Fill in the parameters and click **Generate SSN**.



The screenshot shows the SSN Creator utility window with the following fields and values:

- ADAM server:
- Domain\User:
- Port:
- Password:
- Project Name:
- Source Domain:
- Target Domain:
- Synchronization:
- Migration:
- Delegation:
- Session Name:
- Delegated migration:
- SSN files folder:
- Generate SSN button

Active Directory Cleanup Utility

This utility is designed to help you clean up either or both of the following:

- Migrated objects' Active Directory service attributes used by Migration Manager
- Microsoft Exchange mail redirection addresses

Using Active Directory Cleanup Utility

The utility provides the Active Directory Cleanup Wizard and should be launched locally from the computer where it is installed.

The account used to start the Active Directory Cleanup Wizard must have Full Admin rights to the migration project. To delegate these rights, perform the following steps:

1. In Migration Manager Console, right-click the **Migration Project** node and select **Delegate**.
2. Browse for the account to which you want to delegate the permissions.
3. Select the **Full Admin** role for that account from the role list.
4. Click **Add Account**. This will add the account to the **Delegated accounts** list.
5. Click **OK**.

To start the Active Directory Cleanup Wizard, run the **ADCleaner.exe** file from the installation folder. Follow the steps offered by the wizard to specify where and what to clean up. The following options on the Cleanup Options step of the wizard control what kind of data to clear:

- Cleanup service attributes
- Cleanup mail redirection addresses

Select either or both of the options, as necessary.

About us

Quest provides software solutions for the rapidly-changing world of enterprise IT. We help simplify the challenges caused by data explosion, cloud expansion, hybrid datacenters, security threats, and regulatory requirements. We are a global provider to 130,000 companies across 100 countries, including 95% of the Fortune 500 and 90% of the Global 1000. Since 1987, we have built a portfolio of solutions that now includes database management, data protection, identity and access management, Microsoft platform management, and unified endpoint management. With Quest, organizations spend less time on IT administration and more time on business innovation. For more information, visit www.quest.com.

Technical support resources

Technical support is available to Quest customers with a valid maintenance contract and customers who have trial versions. You can access the Quest Support Portal at <https://support.quest.com>.

The Support Portal provides self-help tools you can use to solve problems quickly and independently, 24 hours a day, 365 days a year. The Support Portal enables you to:

- Submit and manage a Service Request
- View Knowledge Base articles
- Sign up for product notifications
- Download software and technical documentation
- View how-to-videos
- Engage in community discussions
- Chat with support engineers online
- View services to assist you with your product