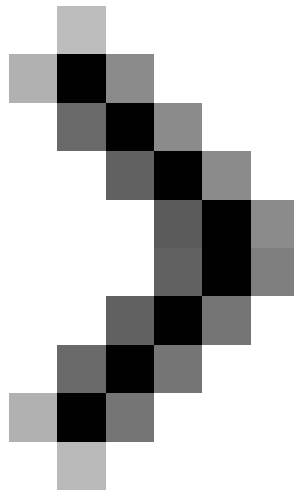


# Transfer mailboxes in Exchange 2007 through Exchange Management Console (Part 2)

In Part 1 of this article, we introduced the Move Mailbox utility that is included in the Exchange Management Console as well as other event logs recorded when the mailboxes were moved in Exchange 2007. Although the Move Mailbox utility provides one. The interface is simple to perform mailbox migration between databases and servers but it still lacks some



## Transfer mailboxes in Exchange 2007 through Exchange Management Console (Part 1)

*Neil Hobson*

**In Part 1 of this article, we introduced the Move Mailbox utility that is included in the Exchange Management Console as well as other event logs recorded when the mailboxes were moved in Exchange 2007.** Although the Move Mailbox utility provides one. The interface is simple to perform mailbox migration between databases and servers, but it still lacks some of the usual settings that administrators need. These report

parameters can be obtained by using Exchange Management Shell to move mailboxes. At the end of Part 2 of this article, I will show you some of the main parameters that can be used. Now, let's look at the log files generated by the mailbox transfer process.

## Log files of Move Mailbox

Exchange 2007's mailbox migration process generates several XML and LOG files every time a transfer is made. You can find these files in the default folder *Program Files\Microsoft\Exchange Server\Logging\MigrationLogs*, see Figure 14.

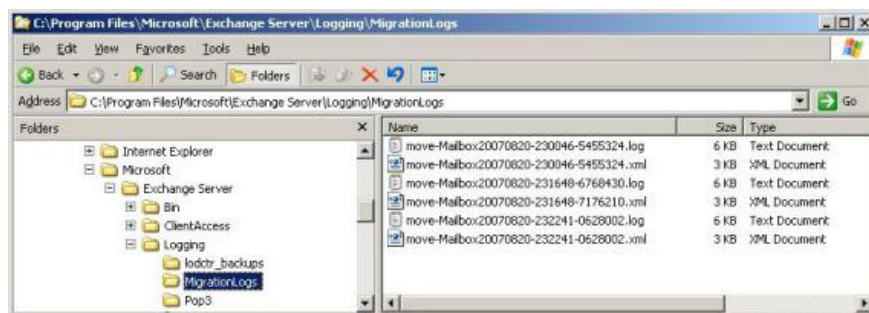


Figure 14: Log files of Exchange 2007 Move Mailbox

Both types of files have information about the mailbox transfer process that has just been performed. Examples of .log and .xml files are shown in Figures 15 and 16.

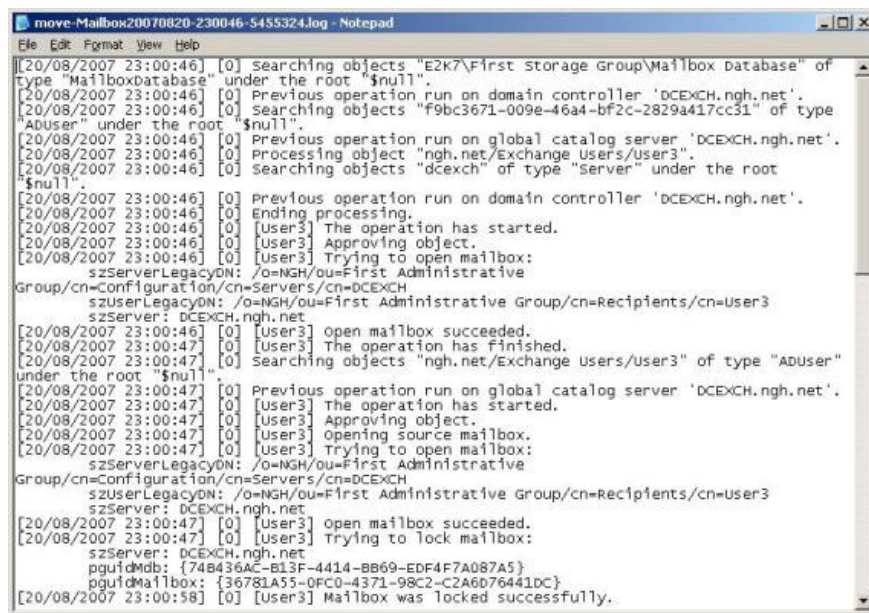


Figure 15: Content of log files

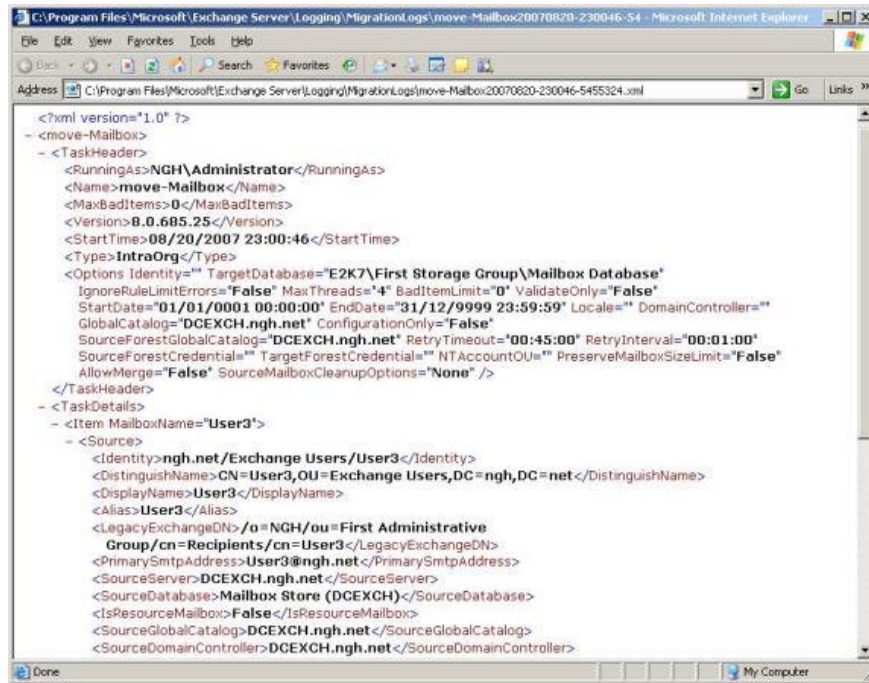


Figure 16: Content of XML files

Reading log files is easier than XML files in raw format because this is not always the easiest thing to do. One answer to that problem is to apply a style sheet. We will re-introduce Exchange 2003 a bit so that we can show you an example of this. The Exchange mailbox process of Exchange 2003 generates XML files in a manner similar to those implemented in Exchange 2007. The Exchange Server 2003 SDK (downloadable here) has an XSLT style sheet that you can use to replace Change the way you view XML files generated during mailbox migration. This issue will give you an idea of ??what you can do with style sheets. Obviously the style sheet provided with the SDK only works with Exchange 2003 XML files, so you will need to create for yourself the Exchange 2007 XML files, or perhaps a similar style sheet will be released by Microsoft. If you download and install the Exchange Server 2003 SDK, you will find the style sheet called **TaskWizardRun.xslt** below the default *Program Files\Exchange SDK\Samples\TaskWizard* folder . To apply the style sheet to the XML file, all you need to do is change the XML file. For example, Figure 17 illustrates the original Exchange 2003 XML file that was opened in Notepad.

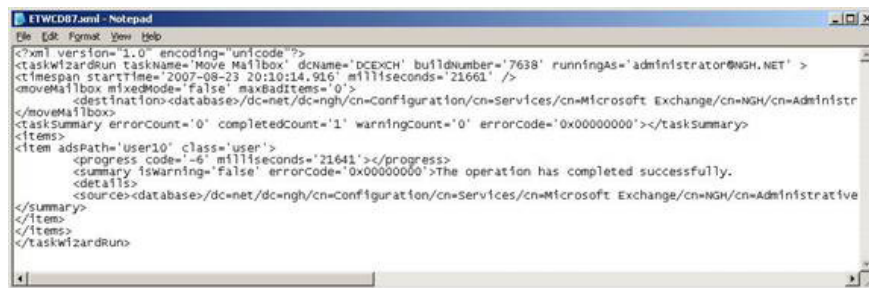
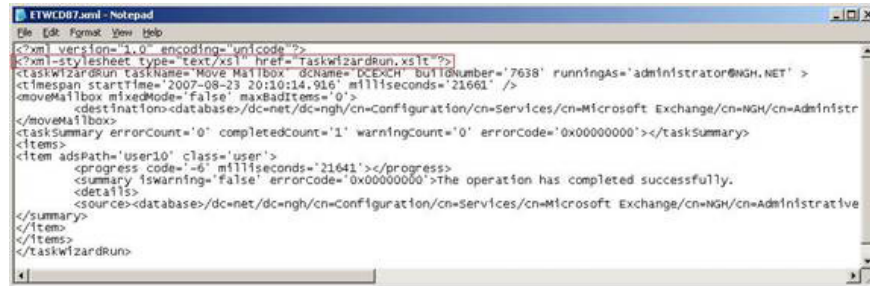


Figure 17: Exchange 2003 XML file in Notepad

Notice the second line is started with `em="">` . To apply style sheets, simply add the following line in front of the 'taskWizardRun' line:

The final XML file will look like the file shown in Figure 18, with the added text already highlighted.



```
<?xml version='1.0' encoding='unicode'?>
<?xml-stylesheet type='text/xsl' href='TaskWizardRun.xslt'?>
<taskWizardRun taskName='Move Mailbox' BCName='DCEXCH' ButtonNumber='7638' runningAs='administrator@NGH.NET' >
<timespan startTime='2007-08-23 20:10:14.916' milliseconds='21661' />
<moveMailbox mixedMode='False' maxBadItems='0' >
  <destination><database>/dc=net/dc=ngch/cn=Configuration/cn=Services/cn=Microsoft Exchange/cn=NGH/cn=Administrative
</moveMailbox>
<taskSummary errorCount='0' completedCount='1' warningCount='0' errorCode='0x00000000'></taskSummary>
<items>
<item adsPath='user10' class='user' >
  <progress code='-6' milliseconds='21641'></progress>
  <summary warning='False' errorCode='0x00000000'>The operation has completed successfully.
  <details>
  <source><database>/dc=net/dc=ngch/cn=Configuration/cn=Services/cn=Microsoft Exchange/cn=NGH/cn=Administrative
</item>
</items>
</taskWizardRun>
```

Figure 18: The XML file of Exchange 2003 has been changed

This means that, when you open the XML file in Internet Explorer, it will look like the example shown in Figure 19. See if you made a typo with the style sheet.

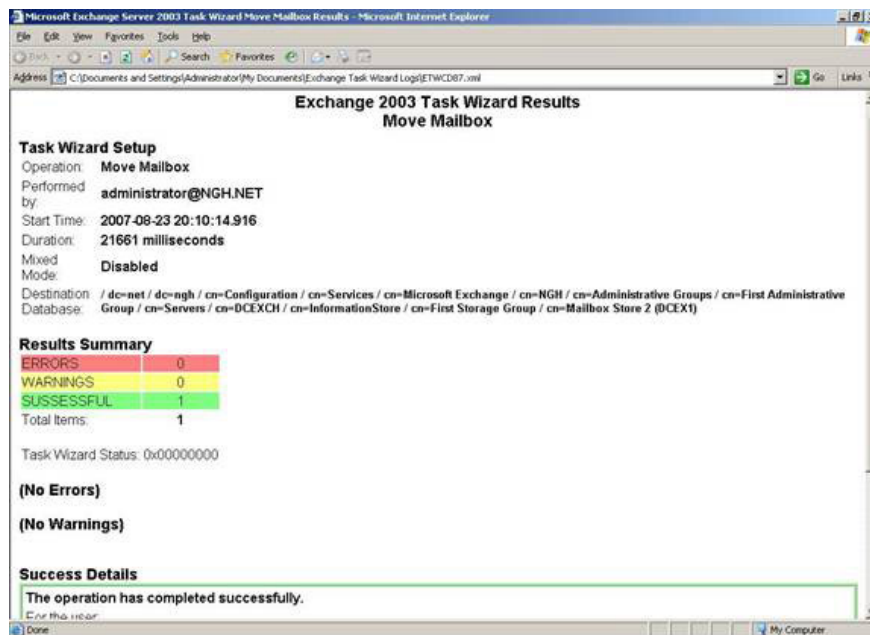


Figure 19: XML file of Exchange 2003 with style sheet

## Transfer Mailbox through Exchange Management Shell

When using Exchange Management Shell to move mailboxes, the cmdlet to use is the Move-Mailbox cmdlet. When using this cmdlet, there are many parameters provided to Exchange administrators that are not provided in the Exchange Management Console. In the last part of this article, we will introduce some interesting parameters that you can refer to and use.

First, let's look at the problem of simply migrating a mailbox from an Exchange 2003 server to an Exchange 2007 server in the same forest. In its basic format, the cmdlet to perform this task for the mailbox of User3 is:

**Move-Mailbox User3 -TargetDatabase 'E2K7First Storage GroupMailbox Database' -Confirm: \$false**

Obviously, the main parameter here is TargetDatabase, this parameter in the above example is configured to move the mailbox to another mailbox database named *Mailbox Database* inside the storage group called *First Storage Group* on the machine. Exchange 2007 host is called *E2K7*. If you execute this cmdlet, you will see the status of the operation when it proceeds through the directory and passes the messages through it. This problem is clearly shown in Figure 20. Note the use of the parameter `-Confirm: $false`, this is the parameter to make sure you are not prompted to ask for sure 'Are you sure?' before the mailbox is actually moved.

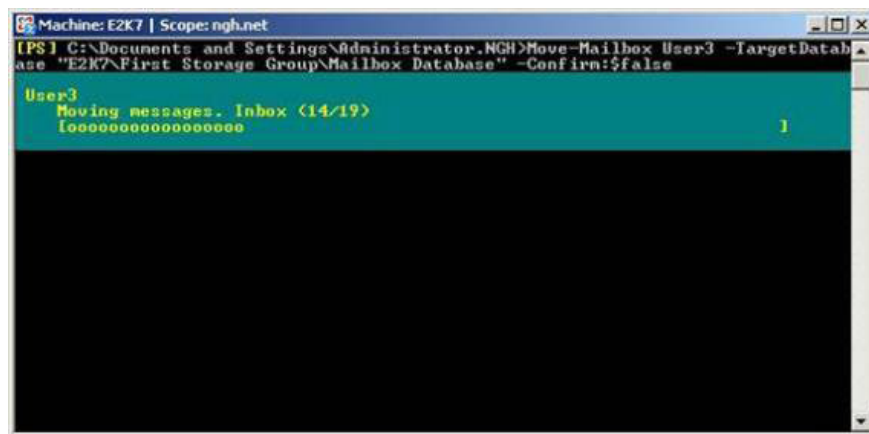


Figure 20: Mailbox is being transferred through Management Shell

The parameter `-ValidateOnly` is a very useful parameter if you want to see which mailboxes can be transferred without transferring it. In Figure 21, you can see the Move-Mailbox cmdlet running with the parameter `-ValidateOnly`, the *Status Message* attribute is shown as 'This mailbox can be moved to the target database' (This mailbox can be transferred to the database). target data).

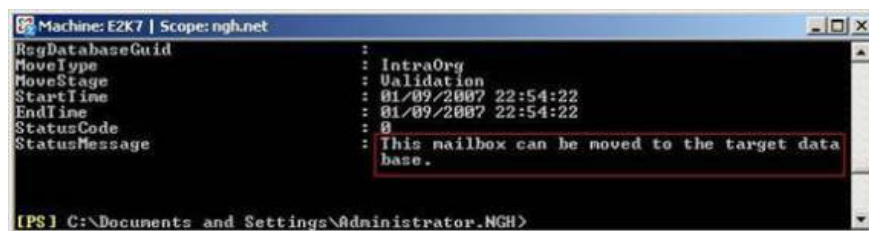


Figure 21: Results obtained from ValidateOnly parameter

After a successful mailbox move, the Status Message property will be set to 'This mailbox has been moved to the target database' (This mailbox has been moved to the target database).

You need to know that the Move Mailbox utility running from within Exchange Management Console is multithreaded and limited to the maximum number of threads of 4. You can increase this number if you use Exchange Management Shell to move mailboxes. The specific parameter that you need to use is the parameter -MaxThreads, which has a maximum value of 30. However, before you set this parameter to the maximum value (30), consider What about the bottleneck if it is done and what your benefits will be.

Now let's look at the case where you want to move all mailboxes on the default mailbox store of an Exchange 2003 server named DCEXCH to an Exchange 2007 server named E2K7. The mailbox store name on the Exchange 2003 server in this example is *Mailbox Store (DCEXCH)*. In this case, we want to move 10 mailboxes simultaneously. Cmdlet to execute will be:

**Get-Mailbox -Database 'DCEXCHFirst Storage GroupMailbox Store (DCEXCH)' | Move-Mailbox -TargetDatabase 'E2K7First Storage GroupMailbox Database' --MaxThreads 10**

The first part uses the Get-Mailbox cmdlet to retrieve all mailboxes that have been stored on the Exchange 2003 server database. This issue is allowed by the Move-Mailbox cmdlet, but note the parameter further. -MaxThreads 10 at the end. The results of the implementation process are shown in Figure 22.

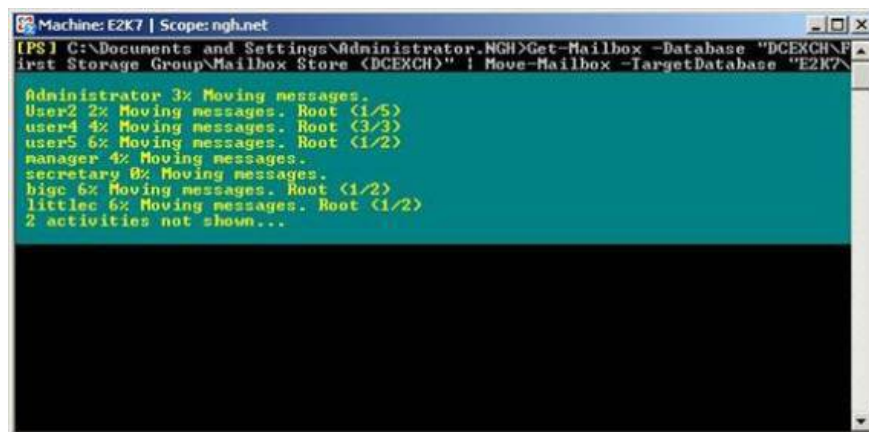


Figure 22: Transferring to more than 4 mailboxes simultaneously

As mentioned earlier, in this article we have described in detail the XML files generated when mailboxes are moved and the default location of these files is the *Program FilesMicrosoftExchange ServerLoggingMigrationLogs* folder. If you want to change the location of the XML log file, you can do it with Exchange Management Shell via the -ReportFile parameter. Note that you need to include the full path and file name in that parameter. If the folder name does not exist, it will be created. An example of the cmdlet in moving mailboxes of User3 and writing the results into *C: MigLogsmailboxes.xml* is:

**Move-Mailbox User3 -TargetDatabase 'E2K7First Storage GroupMailbox Database' -ReportFile C: MigLogsmailboxes.xml -Confirm: \$ false**

Finally, you can remember from part one of this article that the Move Mailbox utility in Exchange Management Console allows you to specify the number of modified items that you will be affected when moving mailboxes. Obviously such a setting also exists within the Exchange Management Shell under the -BadItemLimit parameter form. We also mentioned in part 1 of this article that the largest value you can set is 2,147,483,647, the

minimum value is 0.

## **Conclude**

In the conclusion of part one, we said that moving mailboxes through the Exchange Management Console can be done quite easily. In addition to moving on Exchange Management Shell, there is nothing really complicated and there is nothing that we have to suspect that using Shell will take precedence by many administrators due to configuration options. extensions are provided inside it.

You finished reading the article "**Transfer mailboxes in Exchange 2007 through Exchange Management Console (Part 2)**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.