

# Automating the Process of ASE Errorlog Checking

By Rob Verschoor

Checking the ASE errorlog for error messages is a DBA task that can be performed automatically by using some simple (and harmless) `dbcc` commands. A shell script providing a complete implementation of checking the ASE errorlog can be downloaded for free (see URL below).

## Why Check the ASE Errorlog ?

Checking the ASE errorlog for error messages should be a regular DBA task, as it is important for a DBA to be aware of any problems that have been logged. However, as DBAs tend to be rather busy, in practice this errorlog check is often skipped. Also, because this check normally has to be done manually (i.e., visually inspecting the errorlog file contents), this isn't exactly every DBA's favorite job. It is therefore useful to automate the process of checking the ASE errorlog as much as possible.

In its simplest form, this could be implemented by a shell script searching the errorlog file for certain strings indicating problems, such as "error," "corrupt," "suspect," etc. By running this script as a scheduled job (for example, a "cron" job in a Unix environment) and emailing the search results, the DBA will automatically be notified of any suspected problems requiring attention. By running this job early in the morning each day, the DBA will come into work to find an email waiting with the errorlog messages.

However, this means that not only are the new error messages mailed to the DBA every day, but also all messages from previous days or weeks, so the DBA would receive an ever-increasing list of error messages, with the older ones repeated every time. Obviously, we would prefer that only new error messages (those that were logged since the previous errorlog check), would be found and mailed to the DBA. This requires that we somehow keep track of how far the ASE errorlog has been examined for error messages. This can be achieved by placing a unique text string in the ASE errorlog marking the location up to where the errorlog has been checked.

## Writing a Marker to the ASE Errorlog

Placing a marker string in the ASE errorlog can be done easily using the command `dbcc logprint`. Note that this

command is formally not documented or supported by Sybase; in practice, however, it can be used without risk.

```
1> dbcc logprint ("hello world!")
2> go
```

Running the above command will result in the following line being written to the errorlog:

```
(...)server background task error -1: hello world!
```

Note that the message "background task error -1:" does *not* indicate an error, but is harmless and can be ignored.

`dbcc logprint` works in all ASE versions. When running ASE 12.0, the same function can be performed through the command `dbcc printolog`; this command is similar to `dbcc logprint`, but does not include the "background task error" string:

```
1> dbcc printolog ("hello world!")
2> go
```

Running this command will result in the following line being written to the errorlog:

```
(...)server hello world!
```

We can now create a script for checking the errorlog in which subsequent invocations will be incremental—they will find only new error messages that occurred after the previous check, while old messages will not be found again.

- When checking the ASE errorlog, only the section following the last occurrence of the marker string should be searched for error messages (this is quite simple using Unix utilities like "sed" or "awk");
- After checking the ASE errorlog, a unique marker string is written to the errorlog to mark the place up to where the log has been searched.

It is important that a unique text string is used for the marker, which should not occur in the errorlog as part of any ASE message. In the ready-to-use shell script which I provide, the string `_Marker_For_Checking_Errorlog_` is used. This is arbitrary, though—any other unique string could be used as well.

Note that keeping track of the location in the errorlog could also be done differently; for example, by storing the last-checked errorlog line in a file and using it to find the location the next time the errorlog check is performed. However, the solution described in this article is self-contained and does not need any external files.

### Determining the Pathname of the ASE Errorlog File

The user-friendliness of our errorlog-check script can be improved still further. In order to search the ASE errorlog file, the exact pathname of this file must somehow be determined first. The user could specify this pathname explicitly, but it is much easier if the script determines it automatically.

Fortunately, the solution lies in the rather obscure command `dbcc resource` (again, this is not formally documented or supported, but is without risk in practice). While most of the output of this command is not useful for a DBA, it also contains the actual errorlog pathname for the ASE server in question (in boldface below):

```
1> dbcc traceon(3604)
2> go
1> dbcc resource
2> go

(...)
(...) rinterfpath=/opt/sybase/interfaces
(...) rerrfile=/opt/sybase/install/PROD.log
(...)
```

The errorlog pathname can be extracted from the output of `dbcc resource` using tools such as “sed” and “awk.”

Determining the errorlog pathname through `dbcc resource` works in all ASE versions. In ASE 12.0, the errorlog pathname is also available directly through the (undocumented) global variable `@@errorlog`; this is simpler and more convenient than using `dbcc resource`.

Note that `dbcc resource` also displays the pathname of the interfaces file used by the server (behind the string `rinterfpath=`); while not relevant in the context of this article, this information can be useful when troubleshooting connectivity problems.

### Putting It Together

By combining the features discussed above, a script for checking the ASE errorlog can be created that works as follows:

- The script first determines the location of the ASE errorlog file using `dbcc resource` (or `@@errorlog` in ASE 12.0)
- Next, the script searches the errorlog file to find the last occurrence of a special marker string, indicating the point where the previous check was completed; the remaining part of the errorlog is searched for messages indicating errors or potential problems (note that this requires that the script runs on the same host on which the ASE errorlog file is located)
- Any messages found are emailed to the DBA(s); this can be usually be done through the standard command-line mail utilities such as “mail” or “mailx” (or, if these aren’t available, through “sendmail”)
- Any messages found are emailed to the DBA(s)
- Finally, the marker string is written to the ASE errorlog to indicate the completion of this particular check using `dbcc logprint`

The script should be scheduled for regular (preferably daily) execution; for example, through “cron.”

### Download the Script

A ready-to-use, free Bourne shell script performing this functionality can be downloaded from [www.sypron.nl/chklog.html](http://www.sypron.nl/chklog.html). The script is named “check\_errorlog.sh” and should be invoked as follows (for an ASE server named PRODUCTION):

```
check_errorlog.sh PRODUCTION sa mypassword
```

This script works almost immediately—only minimal configuration is required, such as adding the email address(es) to which the email messages should be sent. Further instructions and details are described in the header of the script.

The script also contains some added functionality, such as the ability to check the errorlog even when the server is not running by explicitly specifying the errorlog pathname on the command line. Comments and suggestions are welcome at [rob@sypron.nl](mailto:rob@sypron.nl). ■

---

*Rob Verschoor is a freelance consultant in The Netherlands, specializing in Sybase ASE. He can be reached at [rob@sypron.nl](mailto:rob@sypron.nl).*