

# Using SourceSafe to Maintain a SourceSafe Database

*by Ted Roche*

## Introduction

Microsoft® Visual SourceSafe® is a version control system use for storing source code, text files, web contents or any other type of documents where ownership, tracking and revision history is important. However, the Visual SourceSafe (VSS) database must have regular maintenance performed to ensure that it is kept in good shape and does not encounter corruption.

Like remembering to get the oil changed or scheduling that appointment with the dentist, it is really easy to let maintenance slide. And like other maintenance, there can be serious consequences to neglect. This article shows how, with a little work up front, an administrator can automate the routine maintenance, keep an eye on what's going on, and be alerted to database problems.

One of the real hassles with using SourceSafe's Analyze utility is interpreting the results intelligently. The method shown here minimizes the noise, so administrators can see right through to any changes they might need to be concerned with.

## The Problem

A SourceSafe database is a collection of files, typically stored on a network share, accessible to all of the client workstations that need to use it. The software to read and write the database runs on the client computers only. There is no software running on the server; it simply works as a file server, responding to file read and write requests. Consequently, any disruption on a workstation or the network infrastructure during one of the write operations can cause corruption in the database.

Microsoft supplies a command-line tool, `Analyze.exe`, to review the structure of a SourceSafe database, examine its contents, and report if there are any problems. Additionally, the analyze program can fix these problems, compact the database, and delete orphaned files no longer in use. Details of how to use the Analyze program are in the Administrator's Guide portion of the help file.

Microsoft recommends that the Analyze utility be run weekly against the database to detect corruption problems. This is a general guideline, and it depends on the amount of work you are doing against the data, the number of users, and the reliability of the infrastructure the source code control system is running within. If there are frequent network glitches, or if you have a large number of transactions running against the database, or if a few hours of downtime to perform recovery are undesirable to you, consider running the analysis more frequently.

At Ted Roche & Associates, LLC, we're developing applications for our clients on an on-going basis. Our source code control repository is over a gigabyte in size, and we want to know as soon as possible when a problem occurs. We run the Analyze utility each night, in its most verbose mode. In addition, we run the longer-running "repair" run on Saturday night.

In the verbose mode, there can be a lot of "informational" messages that are of little or no concern. There are messages about unused space or improperly formatted labels and filenames that will be automatically fixed when the weekly fix process is run. The design of our "minimal

## 2 SourceSafe Maintenance

effort" system is to show these messages only once to the administrator as soon as they are detected, but remove them from subsequent displays, to prevent hiding an important message in the noise of too many messages.

### The Solution

Designing an optimal system for the administrators had a couple of goals in mind:

- **Minimal effort** – the administrators have full-time responsibilities in addition to VSS administration, so everything that can be automated should be. Routine maintenance is scheduled and runs automatically. Results are emailed to the administrators for review or action.
- **Automated scheduling** – set so that routine tasks are performed on a routine schedule, not requiring intervention
- **Clear information** – eliminating extraneous noise so an administrator can immediately detect if there are conditions requiring immediate attention.

### Minimal effort

Once the system is installed and configured, it runs itself. Daily analysis runs are completed, analyzed and mailed to the administrator. Weekly analysis runs, where the underlying database is altered, also complete automatically, but archive any repaired or changed files in the event that older files need to be recovered. Each morning, the administrator is greeted with an email describing the results of the previous evening's process. This report consists of the same set of messages each morning if the process completed without change, or a distinctly different message, letting the administrator know there's something amiss.

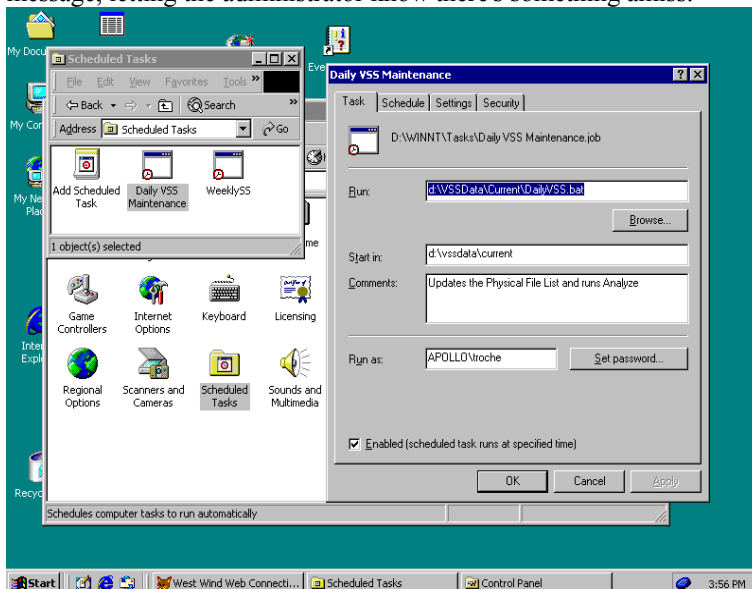


Figure 1: The Scheduled Tasks option in Control Panel lets you automate SourceSafe maintenance batch files.

### Automated scheduling

The Windows Scheduler is used to launch each of the batch files at their designated time. You'll find the Scheduled Tasks interface in the Windows Control Panel (Figure 1). Add a new task, set it to the path to your batch file, specify that it start in the correct directory, and with proper user credentials ("Run as"). Use the second tab of the dialog, "Schedule" (Figure 2) to set the days and times appropriate for your facilities. At our office, we schedule the maintenance to run at 11:30 PM, late enough that we're unlikely to interfere with work, but early enough so that it completes before the daily backup.

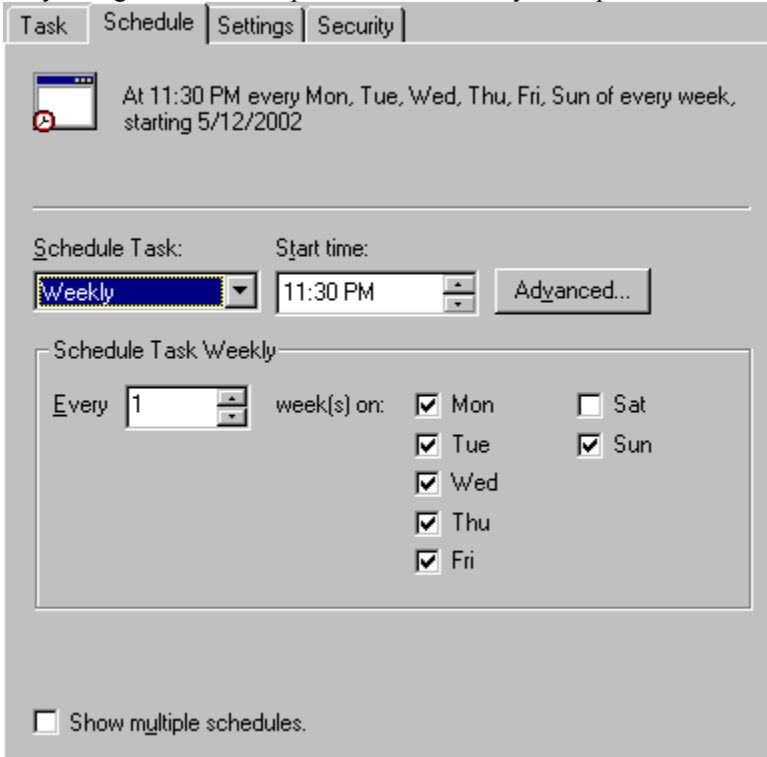


Figure 2: Schedule the maintenance on days and times appropriate for your facility.

### Clear information

There were two processes we wanted automated: a daily analyze that performed a read-only analysis and a weekly analyze that actually attempted to fix problems. Both of the two processes send emails. The formats of the two emails are distinctly different so that administrators will recognize one from the other. In addition, each presents the minimum amount of information necessary for the administrator to recognize if additional action is necessary, or to quickly dismiss the email if there are no problems reported.

### Choosier Administrators Choose DIFF

One of the stickier items was trying to figure out what to do with the voluminous information provided by analyze. Many of the errors reported are cosmetic or trivial enough that they can

## 4 SourceSafe Maintenance

---

be repaired during the next weekly cycle without concern. However, we wanted to capture those errors as close to the point of creation as possible,

*Listing 1: The DailyVSS.BAT file reports any problems in the database.*

(Editor's Note: ↵ symbols show where the text wrapped for printing. These lines should all be one continuous line.)

```
cd d:\vssdata\current

REM Update the physical file list
REM don't start /WAIT Physical.bat - the CMD window hangs!
del Physical.txt
win32\SS PHYSICAL $/ -R -O@Physical.Txt

REM Rename the analyze log to preserve the older results
del data\backup\analyze.old
ren data\backup\analyze.log analyze.old

REM Run the analyze program
start /WAIT win32\ANALYZE.EXE -v4 -i- data

REM Delete the old differences file
del data\backup\LogDiff.txt

REM Determine the difference between the old and new file
win32\ss Diff data\backup\analyze.log data\backup\analyze.old -I- ↵
-O@data\backup\LogDiff.txt -DS -DX-

REM Send the results to the admin
\Install\Blat\BLAT data\backup\LogDiff.txt -to Admin@tedroche.com ↵
-subject "VSS Results" -attacht "data\backup\analyze.log"
```

*Listing 2: The WeeklySS.BAT file cleans and packs the SourceSafe database.*

```
cd d:\vssdata\current

REM Update the physical file list
REM don't start /WAIT Physical.bat - the CMD window hangs!
del Physical.txt
win32\SS PHYSICAL $/ -R -O@Physical.Txt

REM Create a new directory for the results
MD BACKUPS\%DATE%

REM Set the date environment variable
FOR /F "tokens=2-4 delims=/ " %i IN ('date /t') DO SET DATE=%k-%i-%j

REM Run the analyze program, compressing, fixing,
REM deleting, verbose, non-interactive, saving
REM results in the newly-created directory

start /WAIT win32\ANALYZE.EXE -c -d -f -v4 -i- -bBACKUPS\%DATE% data

REM Remind the admins of disk space remaining
```

```

ECHO Analyze and Fix completed. Keep an eye on disk space! ␣
>backups\%DATE%\LogDiff.txt
ECHO >>backups\%DATE%\LogDiff.txt
DIR backups\%date% /w /s >>backups\%date%\logdiff.txt

REM Determine the difference between the old and new file
win32\ss Diff $/Analyze/analyze.log backups\%DATE%\analyze.log -I- ␣
-O@backups\%DATE%\LogDiff.txt -DS -DX-

REM Send the results to the admin
\Install\Blat\BLAT backups\%DATE%\LogDiff.txt -to Admin@tedroche.com ␣
-subject "VSS Repair Results" -attacht "backups\%date%\analyze.log"

REM Check the resulting file into SourceSafe
cd backups
del analyze.log
copy %DATE%\analyze.log .

..\win32\ss CheckIn analyze.log -C"Checked in by Weekly.BAT" -K -P$/Analyze

```

## Outstanding Issues

There still remain two dangers that this process can't overcome. First, administrators might not notice when a pesky email *doesn't* show up! If something breaks the process, either a scheduling task getting turned off or email settings changing, it can be difficult for an administrator to notice that something doesn't happen. Secondly, scheduled tasks need to be re-authenticated whenever the password is changed. If you have a password policy that requires passwords to be changed on a regular basis, you'll need to visit these schedule tasks and re-authenticate them. Best practices would be to create a user with appropriate rights (and limitations, i.e., not allowed to log on remotely) and grant that user alone non-expiring password.

## Conclusion

By automating, scheduling, and adding intelligence to analyzing the raw results of routine maintenance, a SourceSafe database administrator can keep their database in tip-top shape without boring, repetitive drudgery while still maintaining a close watch on the system.

## References

Roche, Ted, *Essential SourceSafe*, Hentzenwerke Publishing, June 2001, ISBN 1-930919-05-0  
The BLAT utility can be found at <http://www.interlog.com/~tcharron/blat.html>

Microsoft best practices white paper

<http://msdn.microsoft.com/library/en-us/dnvss/html/vssbest.asp>

Microsoft KnowledgeBase article on Analyze errors:

INFO: Error Messages from **Analyze Tool** of Visual SourceSafe (Q152807)

<http://support.microsoft.com/default.aspx?scid=kb;en-us;152807>

### About the Author



**Ted Roche** develops data-centric applications for Web, client-server and LAN use. He is a principal in Ted Roche & Associates, LLC, <http://www.tedroche.com>, where he offers consulting, training and mentoring as well as software development services. Ted is author of [Essential SourceSafe](#), co-author of the award-winning [Hacker's Guide to Visual FoxPro 6.0](#), and a contributor to five other FoxPro books.



Copyright © 2005 by Ted Roche. Licensed under the Creative Commons ShareAlike-Attribution License 1.0. See <http://creativecommons.org/licenses/by-sa/1.0/> for details.

Microsoft, Windows, Visual SourceSafe, Natural, and other terms are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.