

Users Manual

# SQLShot<sup>TM</sup>

*For Sybase ASE  
From 11.5 to 15.5*





# SQL SuperVision Release 4.2 User's Guide

<u>About SQL SuperVision</u>	4
<u>SQL SuperVision's Technical Support</u>	5
<u>Enhancements featured in this 4.2 release</u>	6
<u>From Installation to Monitoring</u>	7
<u>Prerequisites</u>	8
Architecture	8
Sybase's Adaptive Server Enterprise	8
Max SQL text monitored	8
Event buffers per engine	9
Sybase's Monitor Server	9
Client applications	9
SQL SuperVision release 4.2	9
Application data	9
%APPDATA%	9
« .MNT » files	9
« .XLS » files	10
« .TXT » files	10
« .FLT » files	10
« .GPH » files	10
« SQL SuperVision.log » file	10
« .tmp » files	10
<u>Installation</u>	11
<u>License key registration</u>	15

**How to start and stop monitoring** **17**

---

**Monitoring SQL Statements** **20**

---

Contents of the session's report	21
Detailed SQL Statement	22
Detail - General	24
Detail - SQL Statement	25
Detail - Batch Text	26
Detail - Execution Plan	27
Detail - Procedure - Trigger	28
Detail - Server Performances	29
Detail - Print	30
SQL Statements' graphs	31
Filters	33
Filter - Spid	35
Filter - Login	36
Filter - Host	37
Filter - Application	38
Filter - Date-Time	39
Filter - SQL Statement	40
Filter - Batch Text	41
Filter - Procedure - Trigger	42
Filter - Execution Plan	44
Filter - Performances	45
Filter - How to manage filter files	45
Display options	45
Sort	47
New Session	48

**Monitoring the Adaptive Server's performances** **49**

---

The Adaptive Server's indicators	50
How to restore a configuration	50
How to add an indicator	52
How to modify an indicator	52
How to remove an indicator	53
How to save the configuration	54
How to refresh the graph	55
SQL Statements related to a time period	55
Graphs' display options	56
New graph	57

**How to save results** **58**

---

How to save to a database	58
Technical recommendations	58
Access management	58
Connecting to the archive database	59
How to save to a file	60
How to save automatically	61
Exports	61
How to export to Excel	61
How to export to a text file	62

<b>How to retrieve results</b>	<b>64</b>
<hr/>	
How to retrieve from a database	64
History	64
Adaptive Server's performances' statistics	65
Statistics on queries	67
Statistics on Execution Plans	68
How to retrieve from a file	69
Automatic refresh	70
<b>Results deletions</b>	<b>72</b>
<hr/>	
<b>Executing SQL SuperVision in a background task</b>	<b>73</b>
<hr/>	
Creating a scheduled task	73
Executing a background task	77
How to shut down a session in background mode	78
<b>SQL SuperVision's archive database model</b>	<b>79</b>
<hr/>	
Tables	80
ssv_stmt	81
ssv_batch	83
ssv_object	84
ssv_textobject	85
ssv_exeplan	86
ssv_textplan	87
ssv_monitor	88
ssv_version	89
ssv_timestamp	90

# About SQL SuperVision

Copyright © 2000- 2010 by DATABASE Architechs

For any additional information about this handbook and/or other products and services thereof, please contact:

DATABASE Architechs  
2, rue Troyon  
92310 Sèvres  
France

Phone : +33 (1) 41 14 60 70  
Fax : +33 (1) 41 14 60 71  
Email: <mailto:support@dbarc.com>  
Web Site : <http://www.dbarc.com>

Sybase, PowerBuilder, Sybase Adaptive Server Enterprise, Sybase Monitor Server, Sybase Open Server, and Sybase Open Client are trademarks of Sybase Corporation. UNIX is a trademark of UNIX Systems Laboratories. Windows, Windows 95, Windows NT, Windows 2000, Windows XP and Windows Vista are trademarks of Microsoft Corporation.

This document is an unpublished work containing confidential information owned by DATABASE Architechs. This document and all attached information thereof are the property of DATABASE Architechs and may not be copied, used, or transferred to anyone, either in part or in full, without the express consent of DATABASE Architechs.

SQL SuperVision's architecture is built on Sybase's Monitor Server. The information collected by SQL SuperVision has been obtained from the values sampled by Monitor Server. SQL SuperVision does not directly collect the performance indicators' values on Sybase Adaptive Server Enterprise. Monitor Server is in charge of accessing the shared memory (For further information, see the documentation of Sybase's Monitor Server).

The SQL Queries, as well as its parameters and all performance information related to the execution context, (performance indicators' values at the Adaptive Server level), are automatically displayed by SQL SuperVision and can be saved in an ASCII file or in a Sybase Adaptive Server database (whose model is provided in the appendix). Thanks to SQL SuperVision, it is possible to consult the performance incidents in real time, as soon as they are detected, including the SQL Statements that are in a pending state.

SQL SuperVision is an application that has been developed using Sybase PowerBuilder and Microsoft Visual C++. It can be deployed on any Windows platform.

For performance monitoring purposes, SQL SuperVision captures the SQL Statements in a pending state exceeding a preset performance threshold and it does so in real time. SQL SuperVision makes it possible to analyze each SQL statement captured. Impact depends on preset performance thresholds. Thanks to SQL SuperVision, it is possible to detect the database design's mistakes and the applications' coding errors before they create major problems during the production phase. SQL SuperVision makes it possible to pinpoint the SQL Queries that have a real impact on the Adaptive Server's performance. Thanks to SQL SuperVision's off-set design, the impact on the performance of monitored systems is linked to a preset performance threshold.

For optimization purposes, SQL SuperVision allows database administrators and project managers to instantly locate Adaptive Server's performance problems that are caused by heavy load in the execution of SQL statements.

For quality control and load tests purposes, SQL SuperVision analyzes the performance of all your applications, whether they are being developed or in a pre-production stage. In the production context, SQL SuperVision captures the SQL Statements exceeding a preset performance threshold. Along with load-test tools, SQL Supervision makes it possible to test your applications in an environment mimicking production conditions and also to identify the causes of saturations related to SQL statements.

For migration purposes, SQL SuperVision is essential to performing migrations from one Adaptive Server Enterprise release to another. SQL SuperVision captures critical SQL Queries sent by your applications in the current Adaptive Server release. The results can then be compared with the results of another monitoring session in the new Adaptive Server release.

# SQL SuperVision's Technical Support

The screenshot shows the Database Architects website with a navigation menu (Services offered, Products, Resources, Company, Partners, Events, Contact) and a main content area. The main content area includes a header with the Database Architects logo and tagline "Architecting for the Future", a navigation menu, and a main content area with a large image of a database structure and text describing their global consulting team. Below this, there are sections for "Publications" (showing books like "Microsoft SQL Server 2005 High Availability"), "Products" (showing "SQL SuperVision DATABASE Tuning"), "Technical Support Documents" (with links to "USER MANUAL" and "Installation Instructions"), "Technical Support Request" (with a form for contacting the team), and "Related" (with links to "P&T Tools", "Events", "Performance & Tuning", etc.). A "News" sidebar on the left lists recent events and articles.

**Database Architects**  
"Architecting for the Future"

Services offered | Products | Resources | Company | Partners | Events | Contact

Database Architects global consulting team has delivered hundreds of large scale, high impact database data architecture, high availability and performance and tuning engagements.

Our strengths are:

- Database Design and Data Analysis/Modeling
- Distributed Data Architectures/Data Replication
- Database Encryption/Security Frameworks
- High Availability
- Performance & Tuning
- Data Standards and Governance
- Delivery on time and within budget
- Extreme Quality of deliverables
- Knowledge Transfer and Education

**Publications**

**Products**

**SQL SuperVision**  
**DATABASE Tuning**

**Technical Support Documents**

Contact Database Architects: you can contact us here or email us

**USER MANUAL** | **Installation Instructions**

**Technical Support Request**

Contact Database Architects team

Firstname:

Lastname:

Company:

Email:

Request:

Message:

The Database Architects Team

**Related**

- P&T Tools
- Events
- Performance & Tuning
- Services Offerings
- Books
- White Papers
- Presentations
- Partners
- Clients
- Customer Testimonials
- Case Studies
- Management Team

**Sign up for our newsletter**

Be the first to hear about new products, events and Database Architects news.

**News**

- Nov. 2-5, 2009 PA & S Summit 2009 Seattle
- Aug. 20, 2009 MS SQL User Group San Diego
- Jul. 27, 2009 Database Tuning version 4.0 available for 30-days Free Trial Website
- Jul. 27, 2009 SQL Supervision version 4.0 available for 30-days Free Trial Website
- Jul. 27, 2009 SQL Shot version 4.0 available for 30-days Free Trial Website
- Jul. 17, 2009 UFE West USA Job/Career Solidarity Initiative SF - Bay Area
- Jul. 1, 2009 Interested in reducing your IT costs? Services
- Jun. 12, 2009 PA & S Summit 2009 - Paul Bertuool Speaker! Seattle
- Jun. 6, 2009 SQL Saturday Portland
- Jun. 6, 2009 SQL Shot Luncheon at SQL Saturday Portland

>more...

Going to <http://www.dbarchitects.com/?page=support> you will have access to the Database Architects' support web-page.

You can then, send us any question about SQL Supervision licensing, Technical Support or Sybase Adaptive Server performance issue.

You can also send us directly an email to [support@dbarchitects.com](mailto:support@dbarchitects.com).

## Enhancements featured in this 4.2 release

Setting of the sampling rate.

This new 4.2 release makes it possible to define the frequency of performance information's snapshots related to queries and metrics of dataserver. These frequencies are configured in milliseconds, meaning that it is now possible to define a frequency of less than one second. Although the access method has been optimized, we would like you to note that the impact on the monitored dataserver is proportional to the frequency selected. We therefore strongly recommend that you avoid to set up the frequency below 500 ms on a production dataserver. For testing and for developments, it is preferable to use a high speed snapshot. In such a situation you need to keep an eye on the activity of the monitor server so as not to pollute the events being observed.

# From Installation to Monitoring

Please follow the steps below to set up and configure SQL SuperVision.

Prerequisites

Installation

License key registration

How to start and stop monitoring

The setting of the network connections do not come with the SQL SuperVision package. Please have your Database Administrator perform these operations beforehand. Although these settings are platform-specific, please find below the information needed to set up SQL SuperVision.

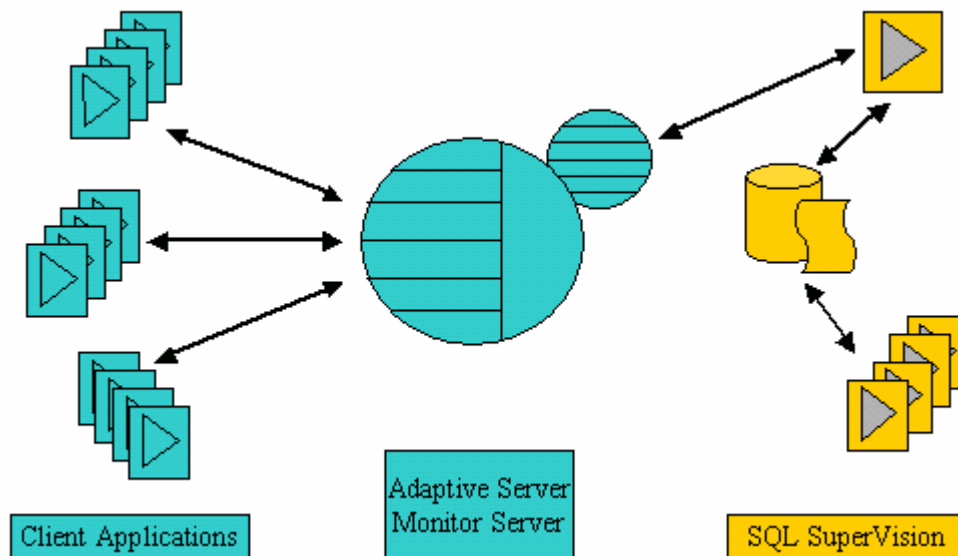


# Prerequisites

## Architecture

Sybase Adaptive Server Enterprise  
 Sybase Monitor Server  
 Client applications  
 SQL SuperVision release 4.2

## Architecture



SQL SuperVision connects to Monitor Server. To make operate the system, you need to install OpenClient 11.1.1+ on the PC workstation hosting SQL SuperVision. You then need to define the network addresses, within the SQL.INI file (using dscedit for example) in order to connect to Monitor and to Adaptive Server Enterprise (ASE).

Caution: the logical names that are used for connections, which are defined in the local SQL.INI file (on the PC WorkStation where SQL SuperVision is installed) must be identical to those present in the interfaces file or in the SQL.INI file of the host machine of the Monitor and the Adaptive Server.

This is illustrated in the following example:

```
[ASE12_PROD]
query=TCP,SERVPROD,5000
master= TCP,SERVPROD,5000
[PROD_MS]
query=TCP,SERVPROD,5002
master= TCP,SERVPROD,5002
```

ASE12\_PROD and PROD\_MS, corresponding respectively to Adaptive Server and Monitor Server, must have the same names in the various interfaces or SQL.INI files (see Sybase documentation: Monitor Server's User Guide).

## Sybase's Adaptive Server Enterprise

All Adaptive Server Enterprise 11.5+ releases installed on any platform are supported by SQL SuperVision. However, two parameters need to be set for SQL SuperVision to work. The operations outlined below are described more precisely in the Monitor Server's User Manual (Sybase documentation).

The parameters in question are as follows : "max SQL text monitored" and "event buffers per engine".

### Max SQL text monitored

Connect to the Adaptive Server, using ISQL, with a grantedg"sa\_role" login and execute the following SQL statement:

```
exec sp_configure "max SQL text monitored", <value>
```

The <value> setting determines the size of SQL text and execution plan text that can be captured by the Monitor Server. We recommend to set it to 8192.

Caution: SQL SuperVision cannot work without this amendment (0 is the default value). When you start a monitoring session, SQL SuperVision verifies the value that has been assigned to this parameter. It cannot be inferior to 1024. This parameter is static and the Adaptive Server needs to be shut down and restarted for the Adaptive Server to acknowledge the new value.

## Event buffers per engine

Connect to the Adaptive Server, using ISQL, with a granted "sa\_role" login and execute the following SQL statement:

```
exec sp_configure "event buffers per engine", <value>
```

SQL SuperVision can work without this amendment (100 is the default value). If this value is set too low, however, the Monitor Server is unable to capture all the events. We recommend that you use the following value: 2000. This parameter is static and the Adaptive Server needs to be shut down and restarted for the Adaptive Server to acknowledge the new value.

## Sybase's Monitor Server

Monitor Server is an essential component of the system. SQL SuperVision cannot directly monitor the Adaptive Server. The release of the Monitor Server must be compatible with the Adaptive Server to which it is dedicated. It also needs to be installed and running. Before launching the Monitor Server, make sure that parameters have been properly configured (see Sybase documentation: Monitor Server's User Guide).

## Client applications

There are no prerequisites for the latter.

## SQL SuperVision release 4.2

To install SQL SuperVision, you will need to have a PC workstation running on Microsoft Windows 2003/XP/VISTA/7. You will also need to install Sybase's Open Client release 11.1.1+. SQL SuperVision will require at least 25 MB of free disk space. This can be located on the PC's hard drive or on a network's shared disk drive.

## Application data

The files used by SQL SuperVision are stored in the directory % APPDATA% (see below), as follows.

See also:

%APPDATA%

« .MNT » files

« .XLS » files

« .TXT » files

« .FLT » files

« .GPH » files

« SQL SuperVision.log » file

« .tmp » files

### %APPDATA%

To display the name of that directory, run the DOS command:

```
echo% APPDATA%
```

### « .MNT » files

These files contain performance data collected by SQL<sub>g</sub>SuperVision. These files are stored by default in the

%APPDATA%\SQL SuperVision\Data Files directory.

### « .XLS » files

By default, the data exported to Microsoft Excel spreadsheets are stored in the %APPDATA%\SQL SuperVision\Data Files directory.

### « .TXT » files

By default, the data exported to text format files are stored in the %APPDATA%\SQL SuperVision\Data Files directory.

### « .FLT » files

By default, the filters' configuration files are created in the %APPDATA%\SQL SuperVision\cfg\flt directory.

### « .GPH » files

By default, the graphs' configuration files are stored in the %APPDATA%\SQL SuperVision\cfg\gph directory.

### « SQL SuperVision.log » file

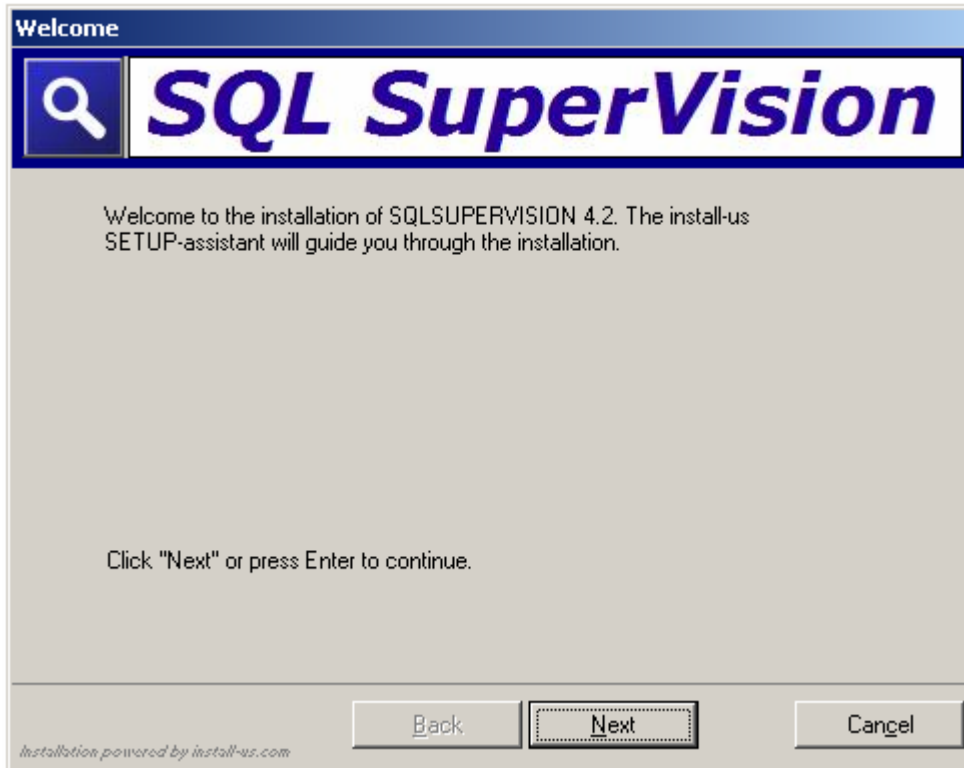
The SQL SuperVision's errorlog file is in the %APPDATA%\SQL SuperVision\log directory.

### « .tmp » files

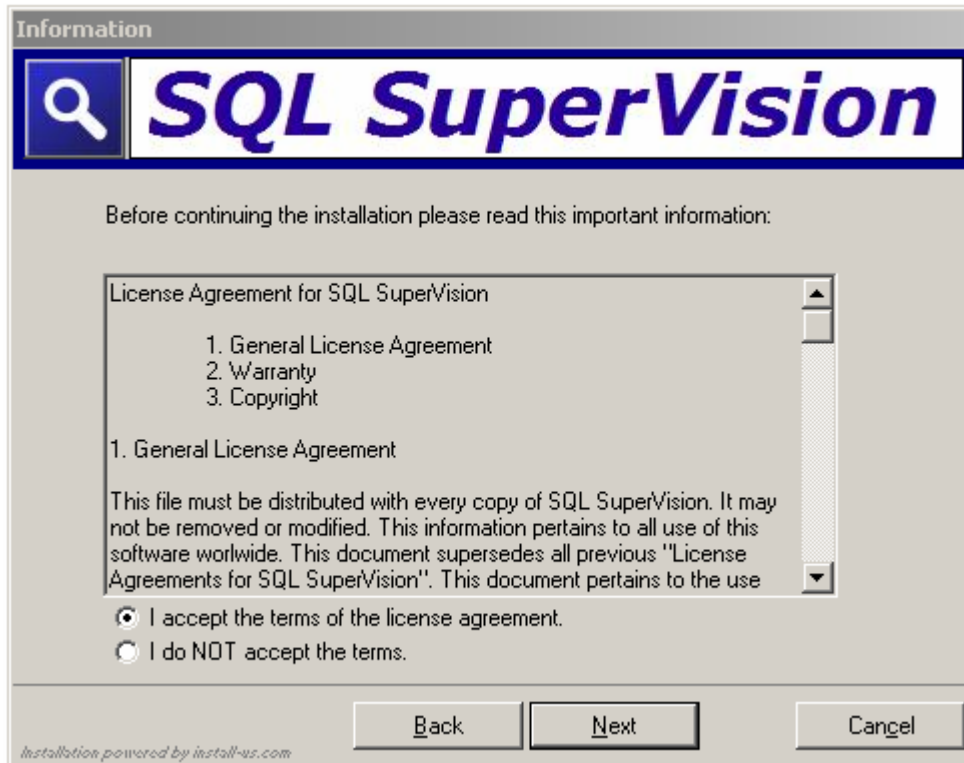
The files used to control background mode executions are created in the %APPDATA%\SQL SuperVision\log directory.

# Installation

Insert your CDROM into the drive. The installation dialog box appears. Alternatively, you can run the following program "<CDROM drive>:\setup.exe".

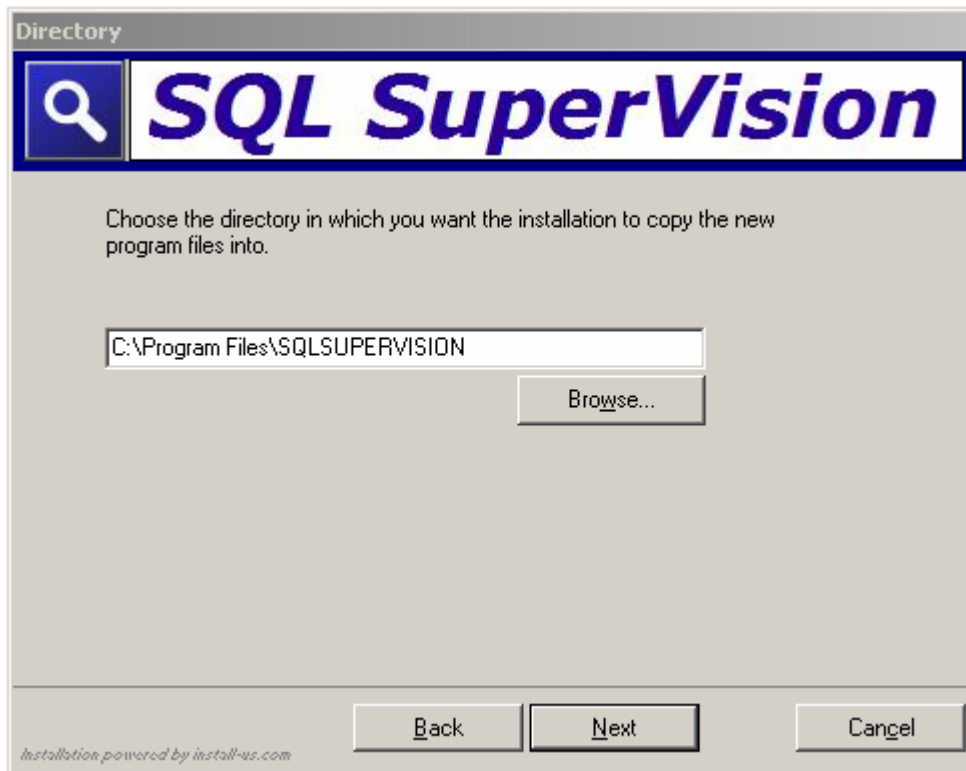


Set up starts.

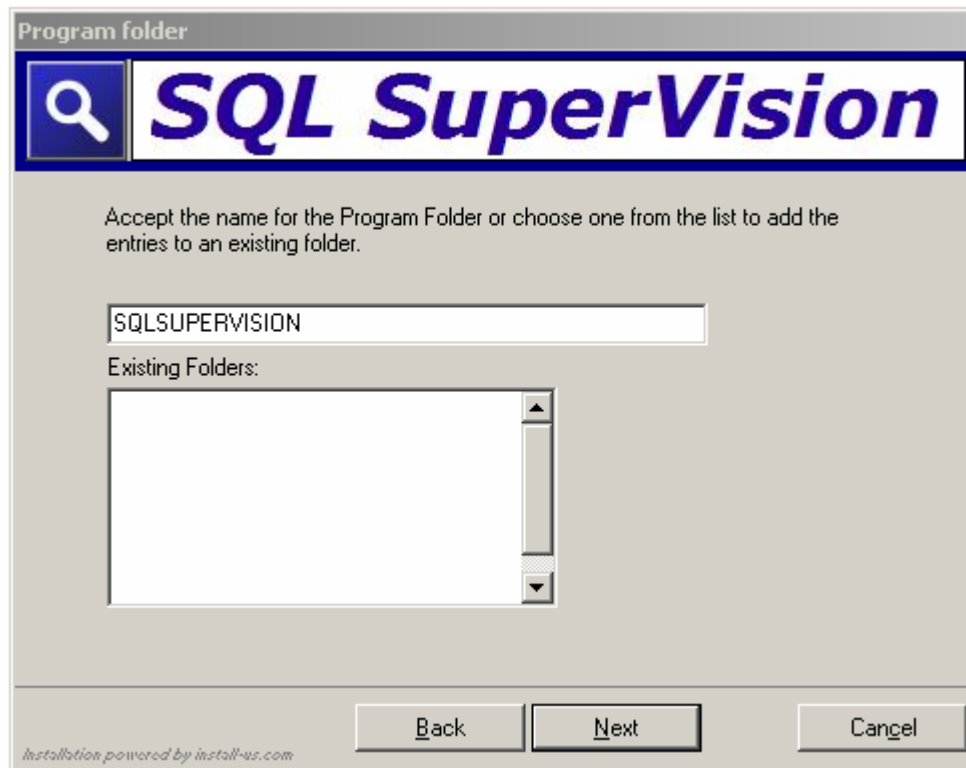


Enter SQL SuperVision's installation directory. SQL SuperVision needs to be installed on a computer that is able to connect to the Monitor Server (see paragraph on "Prerequisites") , or a computer that can access SQL Supervision's files or archive

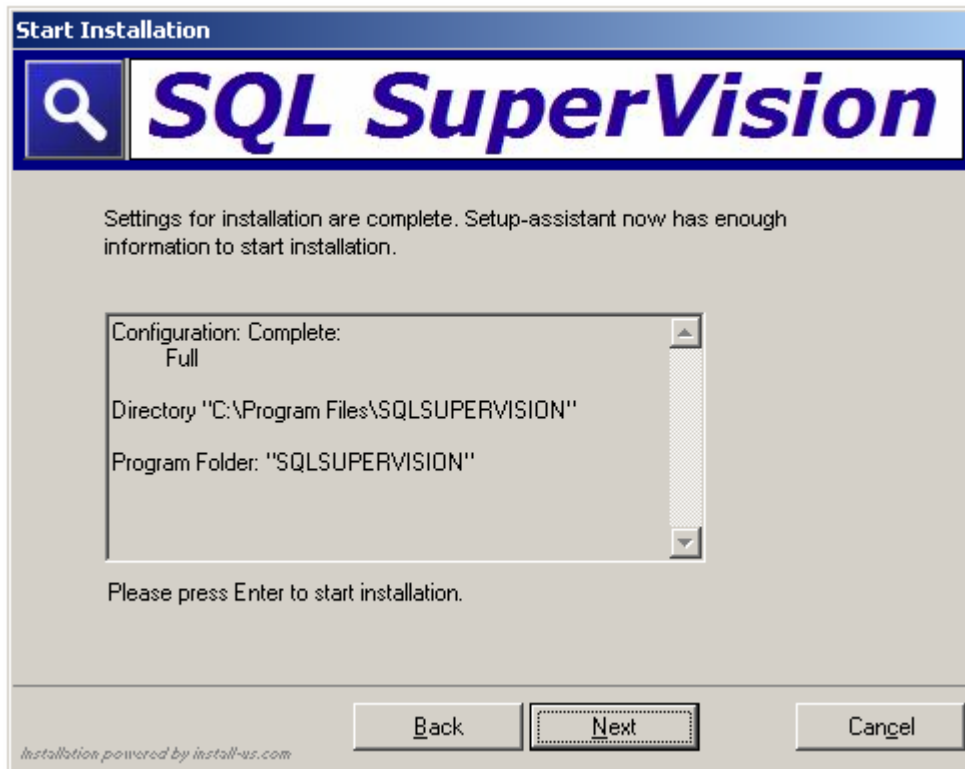
database.



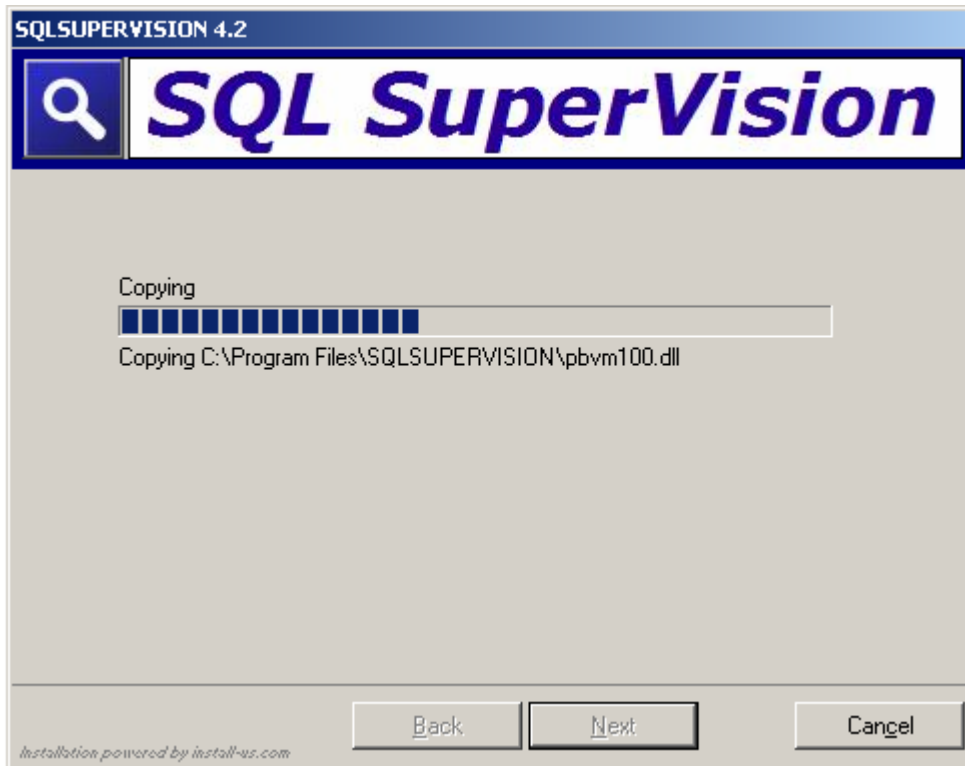
Next, enter the folder where you want to store SQL SuperVision.

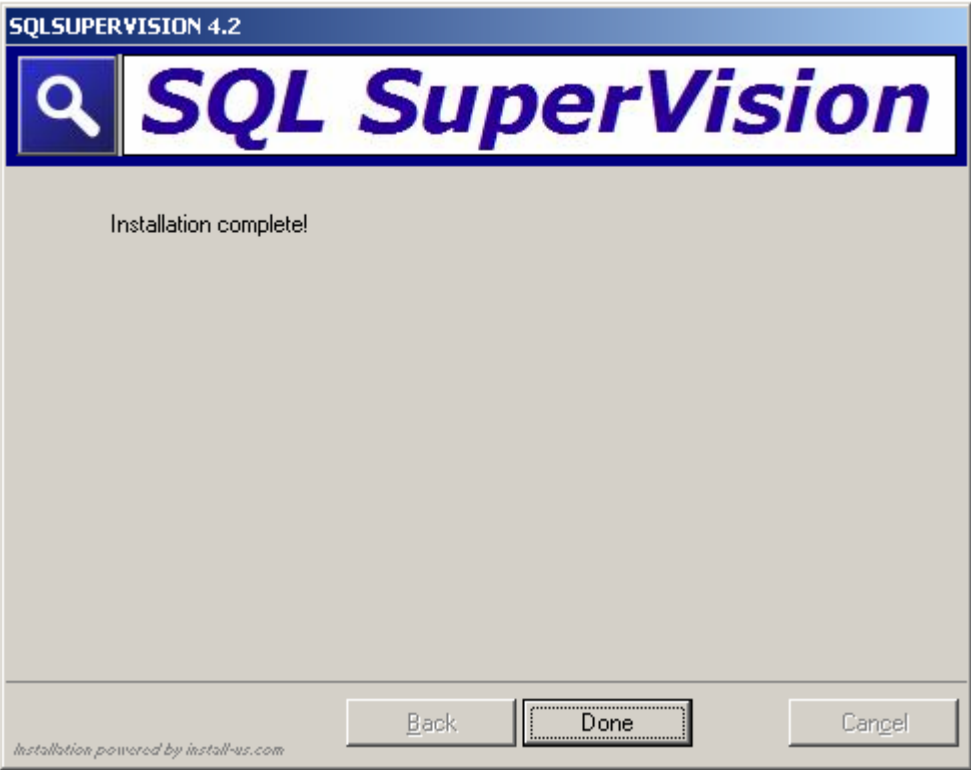


The installation program displays a validation window.



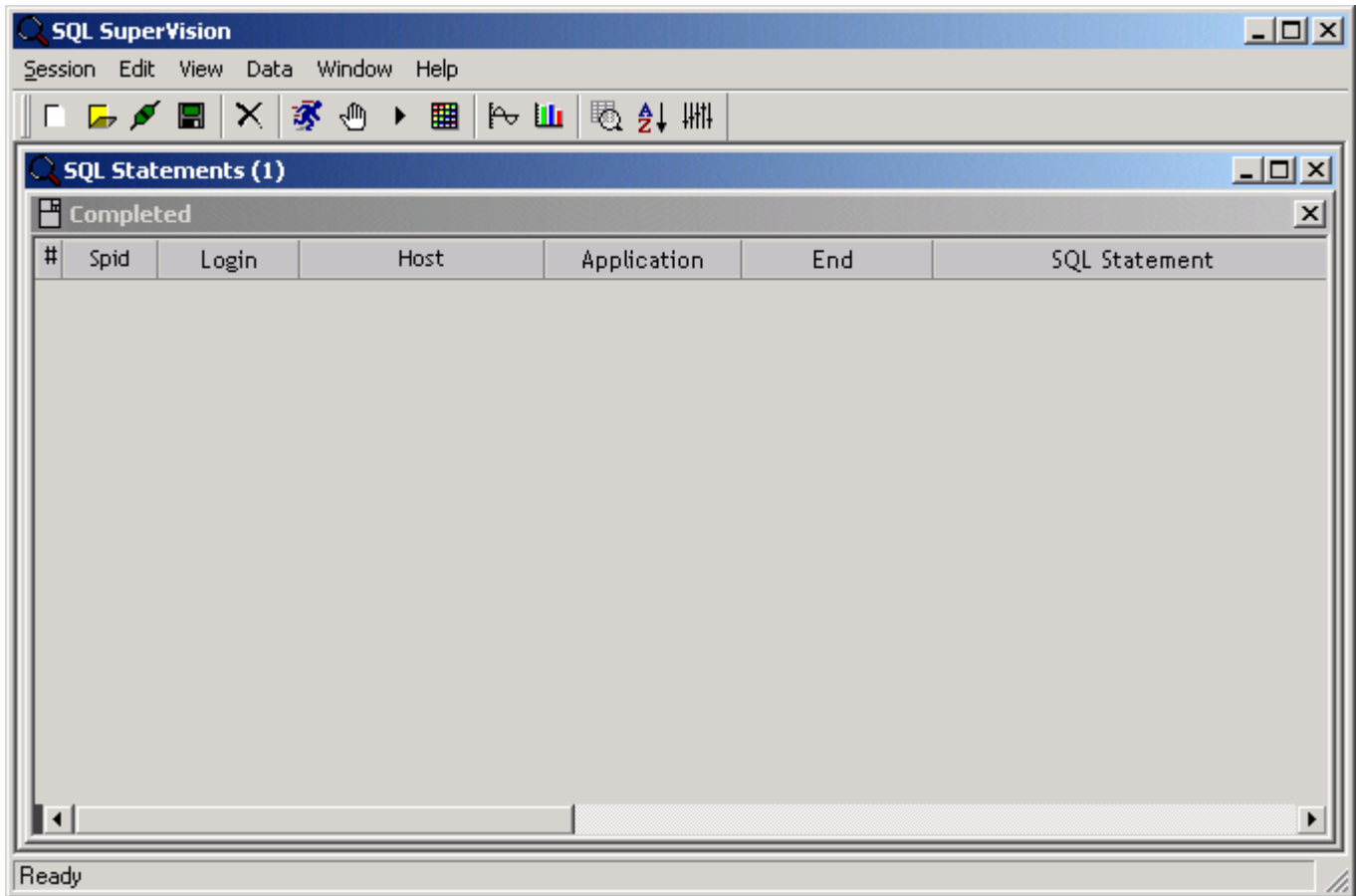
The installation proceeds automatically.



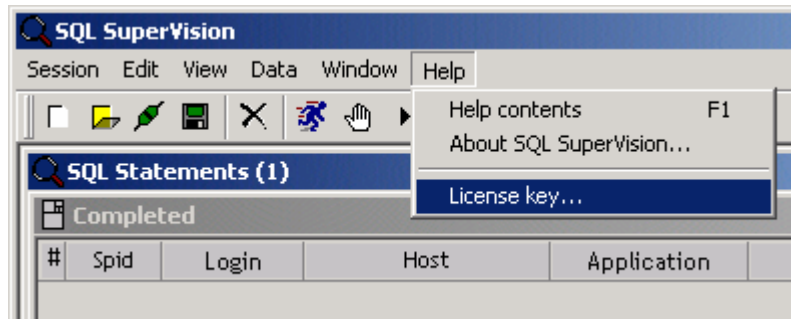


## License key registration

Proceed as follows: Go to the "Start - Programs - SQL SuperVision" menu and, run the "SQL SuperVision" program. The following window appears. Once started, SQL SuperVision accesses its initialization file to use the display configuration. This configuration is saved each time you leave the program.

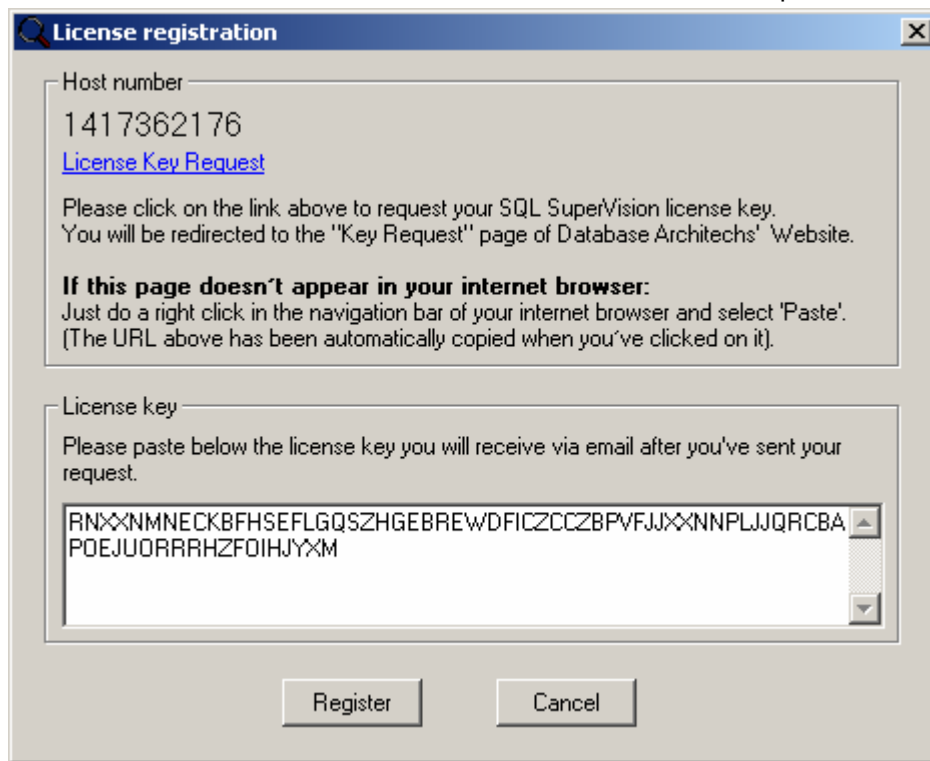


Following the installation, you are asked to register your license key. The latter manages monitoring connections to the Adaptive Server and authorizes a defined number of simultaneous sessions for a specific number of connected users, based on the type of license purchased. To obtain your license key, choose the "Help - License key..." menu item.

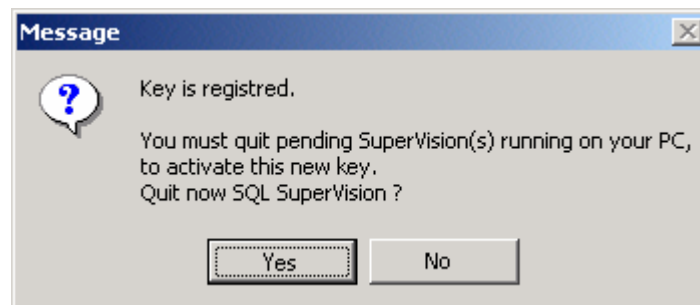


The following dialog box appears. The program displays the PC's identification (host ID). You will need to forward it to <mailto:support@dbarc.com>, so they can give you your personal SQL SuperVision's license key. Next, type your key in the "License key" edit box and click on the "Register" button.



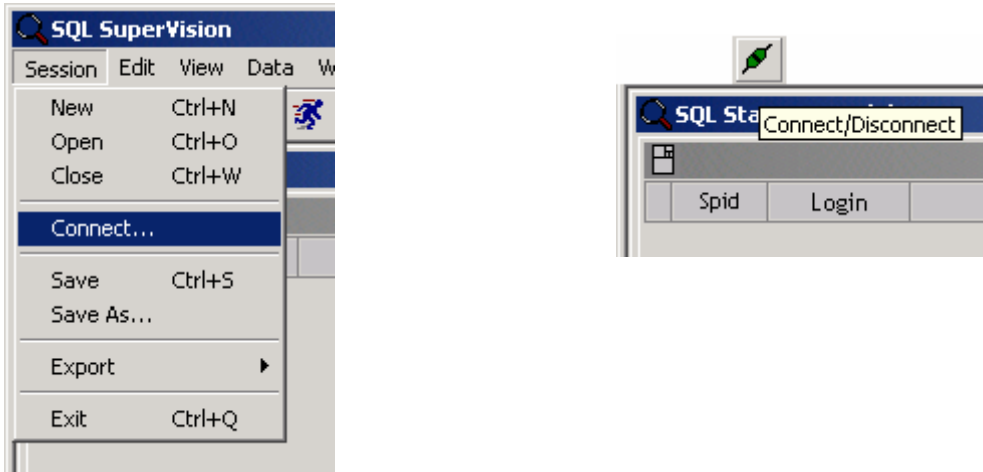


A validation dialog box appears. Quit the program to allow SQL SuperVision to recognize the key.

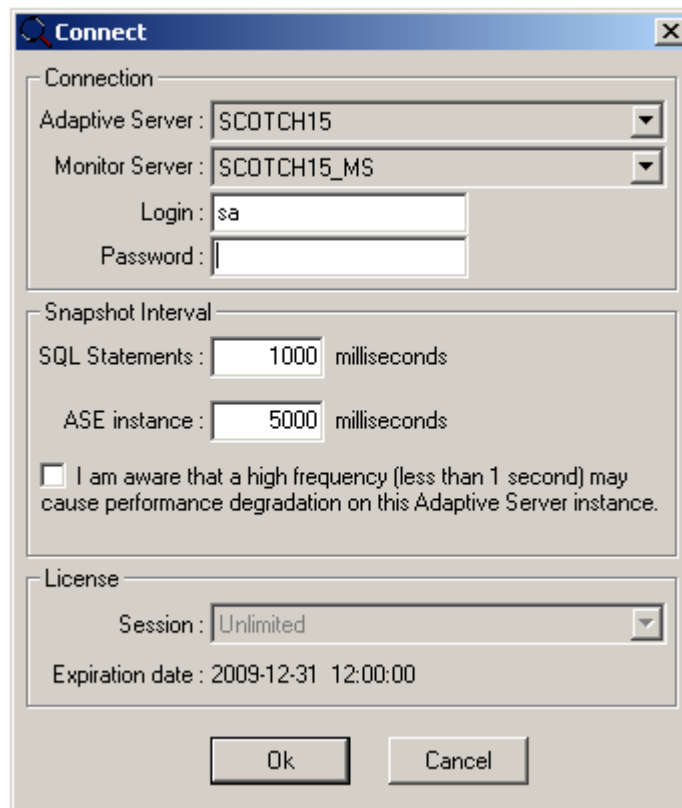


## How to start and stop monitoring

To start a monitoring session on the targeted Adaptive Server, choose the "Session - Connect..." menu item. Alternatively, you can click on the "Connect/Disconnect" menu icon.



The connection dialog box appears.



The Adaptive Server is the targeted Adaptive Server that needs monitoring. The Monitor Server is the name of the associated Monitor Server. You must connect using a granted "sa\_role" login. You can also use the system administrator "sa" login.

Enter the snapshot frequencies for both the SQL Statements and the Adaptive Server instance, but beware of the following: if the frequency is too high, the performance of the Adaptive Server being monitored will be decreased. Choose the session of the license you need to use for the current monitoring session. The list contents depends on the license purchased and varies according to the monitoring sessions that currently active.

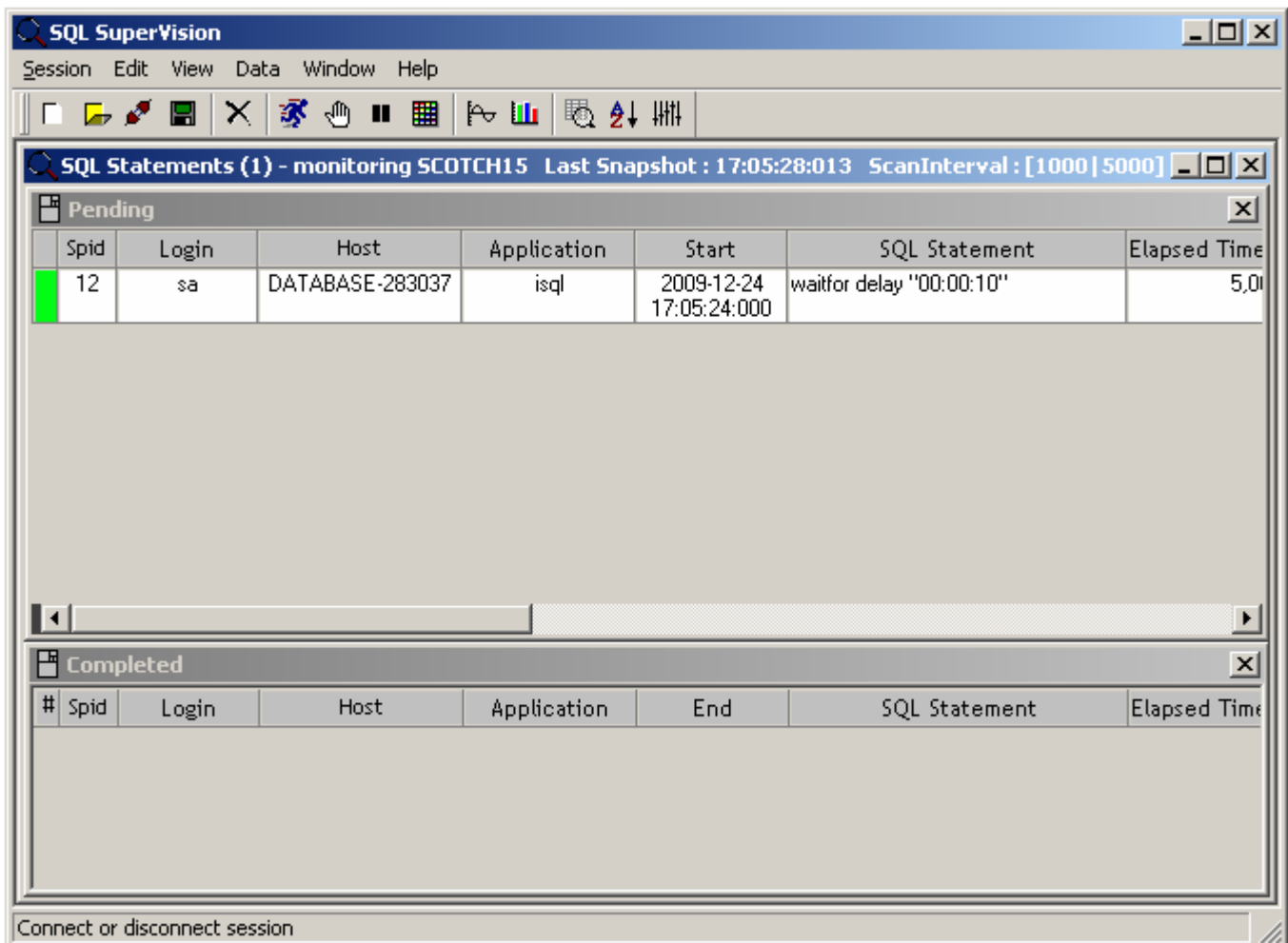
As soon as the session is connected, the window's title changes to display the following information: SQL

Statements (1) monitoring <name of the targeted Adaptive Server> <last snapshot datetime> <scan interval>. The "Connect/Disconnect" menu icon changes to red to indicate that the session is in progress and the "Refresh" menu lets you know that the graphical user interface is displaying information in real time. The child window's upper part (See Title of this window) changes to "Pending" state.



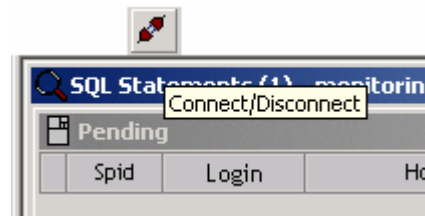
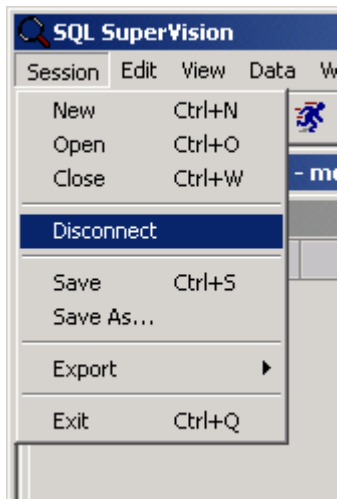
As soon as the session is connected and the Automatic Refresh setting is activated, SQL Statements appear in the session's report (based on the SQL activity of the targeted SQL Server). To ensure that the monitoring is properly set up, simply connect to the targeted Adaptive Server and run the following SQL Statement:

```
waitfor delay "00:00:10"
```



After a few seconds, the SQL Statement appears in the session's report. This shows that the monitoring is operational. If it fails to appear, see Chapter on [SQL SuperVision's Technical Support](#).

To stop a monitoring session, click on the "Connect/Disconnect" menu icon, or select the "Session - Disconnect" menu item.



The window's title changes to display the "SQL Statements (1)", which means that the session is no longer active. The "Connection" and "Automatic Refresh" menu icons are deactivated state and the title located in the upper part of the screen is no longer visible.



# Monitoring SQL Statements

The captured SQL Statements appear in the session's report in real time. In the upper part of the window, quantitative information change with each passing interval until the end of the SQL Statement's execution. At this moment, the SQL Statement moves from the top part ("pending") to the bottom part of the window ("completed").

See also:

[How to start and stop monitoring](#)

[Contents of the session's report](#)

[SQL Statements' graphs](#)

[Filters](#)

[Display options](#)

[Sort](#)

[New session](#)

## Contents of the session's report

See also:  
Detailed SQL Statement

The screenshot shows the SQL SuperVision application window. The title bar reads "SQL SuperVision". The menu bar includes "Session", "Edit", "View", "Data", "Window", and "Help". The toolbar contains various icons for session management and monitoring. The main window is titled "SQL Statements (1) - monitoring SCOTCH15 Last Snapshot : 17:49:23:530 ScanInterval : [1000 | 5000]". It is divided into two sections: "Pending" and "Completed".

**Pending**


Spid	Login	Host	Application	Start	SQL Statement	Elapsed Time
1	Yves	VISIONTEST	isql	2009-12-24 17:49:22:513	waitfor delay '00:00:4' rollback	2,00
30	Thierry	VISIONTEST	isql	2009-12-24 17:49:23:530	update Dbarc_cpteur set id = id + 1	0,98
21	Greg	VISIONTEST	isql	2009-12-24 17:49:20:513	update Dbarc_cpteur set id = id + 1 where code = @code	4,00

**Completed**

#	Spid	Login	Host	Application	End	SQL Statement	Elapsed Time
21	1	Yves	VISIONTEST	isql	2009-12-24 17:49:22:513	update Dbarc_cpteur set id = id + 1 where code = @code	1,05
20	30	Thierry	VISIONTEST	isql	2009-12-24 17:49:22:513	waitfor delay '00:00:4'	3,03
19	18	Greg	VISIONTEST	isql	2009-12-24 17:48:02	update Dbarc_cpteur set id = id + 1 where code = @code	2,01
18	23	Yves	VISIONTEST	isql	2009-12-24 17:48:02	update Dbarc_cpteur set id = id + 1 where code = @code	8,01
17	16	Thierry	VISIONTEST	isql	2009-12-24 17:48:02	waitfor delay '00:00:4'	2,01
16	16	Thierry	VISIONTEST	isql	2009-12-24 17:47:59:140	update Dbarc_cpteur set id = id + 1	7,95

At the bottom of the window, there is a button labeled "Connect or disconnect session".

The session's report contains two parts. The displayed SQL Statements located in the window's upper part are in a pending state. Once completed, these statements are moved to the lower part of the screen. When the option "Automatic Refresh" is





active (icon: ) , the upper part of the window (SQL Statements that are in a pending state) is the only one that is enabled and the bottom part of the window (completed) cannot be selected. The same goes for the SQL Statements' graphs. To click on or scroll through the completed SQL statements' reports, you need to switch off the "Automatic Refresh" option. The monitoring remains active, but the captured pending SQL Statements are no longer refreshed, but are still saved in the background. Once you activate the "Automatic Refresh" option, the results saved in the background are immediately inserted into the session's report and all information regarding the ongoing SQL Statements becomes available again, in real time.

For each captured SQL Statement, the program displays the following information:

Spid :                   The process identifier.  
Login :                 Login name.  
Host :                  Hostname.

Application :	Application name.
Start :	The date and time of the pending SQL Statements. (These data vary according to the time zone of the Adaptive Server and they correspond to the beginning of the SQL Statement's execution).
End :	the date and time at which the execution of the SQL Statement ended. (These data vary according to the time zone of the Adaptive Server and they correspond to the completion of the SQL Statement).
Elapsed Time :	Time needed to execute the SQL Statement.
Batch Text :	The query sent by the client application.
SQL Statement :	The SQL Statement captured by SQL SuperVision. This indicates the performance issue that has been identified.
Execution Plan :	The SQL Statement's execution plan.
Procedure - Trigger :	The Database containing the stored procedure, the trigger or the function, followed by the procedure, the trigger or the function's owner name, followed by the procedure, the trigger or the function's name. When a SQL Statement is not part of a procedure, a trigger or a function (dynamic SQL), the character string "***NoDatabase**.*NoOwner**.*NoObject**" is displayed.
Logical reads :	The quantity of data page reads (in data cache) consumed by the SQL Statement.
Physical reads :	The quantity of data page reads (on physical device) consumed by the SQL Statement.
CPU Time :	The computation time used by the SQL Statement.
Locks :	The quantity of demands for locks requested by the SQL Statement that were not satisfied immediately.

Monitoring (See Paragraph on "How to start and stop monitoring ") and SQL Statements' Automatic Refresh displays (See Paragraph on "Session reports' contents") work independently from each other. This means that it is impossible to view the details on the SQL Statements contained in the completed statements' report when the option "Automatic Refresh" is active

(icon: ) and when the "Completed" report is greyed out. However, it is possible to interrupt the refreshing task () without losing the results of the ongoing monitoring session. During the interruption, the captured SQL Statements are saved in the background and you can view the details of the SQL Statements contained in the reports are viewable. When the option "Automatic Refresh" is active, you can view the pending SQL Statements' details at any time, in real time, as was the case for the former releases. The  icon, located in the "View - SQL Statements - Pending" menu, opens up the session's report containing the pending SQL Statements. The  icon, located in the "View - SQL Statements - Completed" menu, opens up the session's reports containing the executed SQL Statements.

Each session's report can be split in two (at the left of the horizontal scrollbar). This enables you to view the pending SQL Statements' reports as well as the completed SQL Statements' reports. Each report comes with its own horizontal scrollbar. However, it is impossible to view synchronized columns in the window while scrolling horizontally.

## Detailed SQL Statement

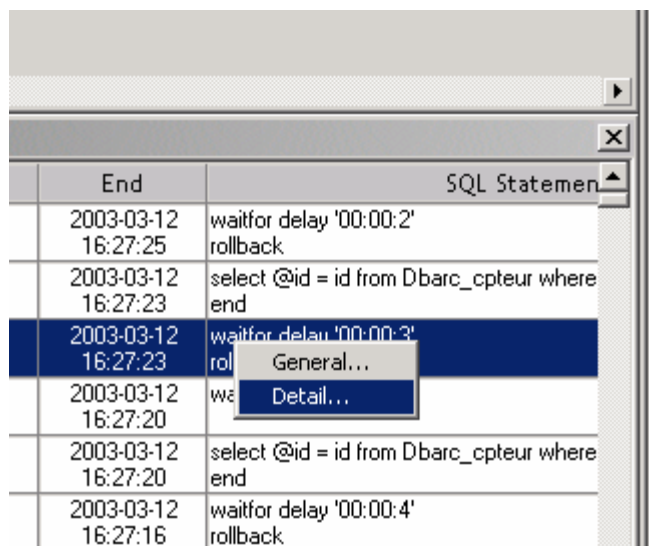
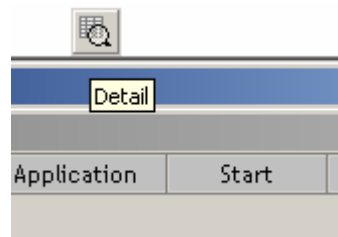
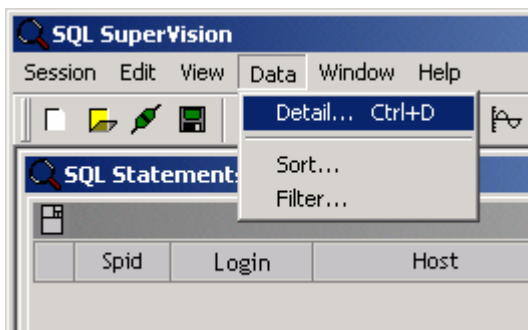
See also:

- Detail - General
- Detail - SQL Statement
- Detail - Batch Text
- Detail - Execution Plan
- Detail - Procedure - Trigger
- Detail - Server Performances
- Detail - Print

To see a detailed SQL Statement, simply click on a specific report's line, in the "#" column. To select the line, make sure it is highlighted. To unselect lines, simply click on any of the report's columns.

#	Spid	Login	Host	Application	End	SQL Statement
139	9	Yves	VISIONTEST	ctisql	2003-03-12 16:27:25	waitfor delay '00:00:2' rollback
138	9	Yves	VISIONTEST	ctisql	2003-03-12 16:27:23	select @id = id from Dbarc_cpteur where end
137	12	Greg	VISIONTEST	ctisql	2003-03-12 16:27:23	waitfor delay '00:00:3' rollback
136	8	Thierry	VISIONTEST	ctisql	2003-03-12 16:27:20	waitfor delay '00:00:4'
135	12	Greg	VISIONTEST	ctisql	2003-03-12	select @id = id from Dbarc_cpteur where

Once you have made your selection, select the "Data - Detail..." menu item. Alternatively, you can click on the "Detail" menu icon (See Figures below), or use the right click on your mouse, to display the context menu. Next, select the "Detail" menu item. You can double-click any column of the selected line.



A window with six tabs is displayed.



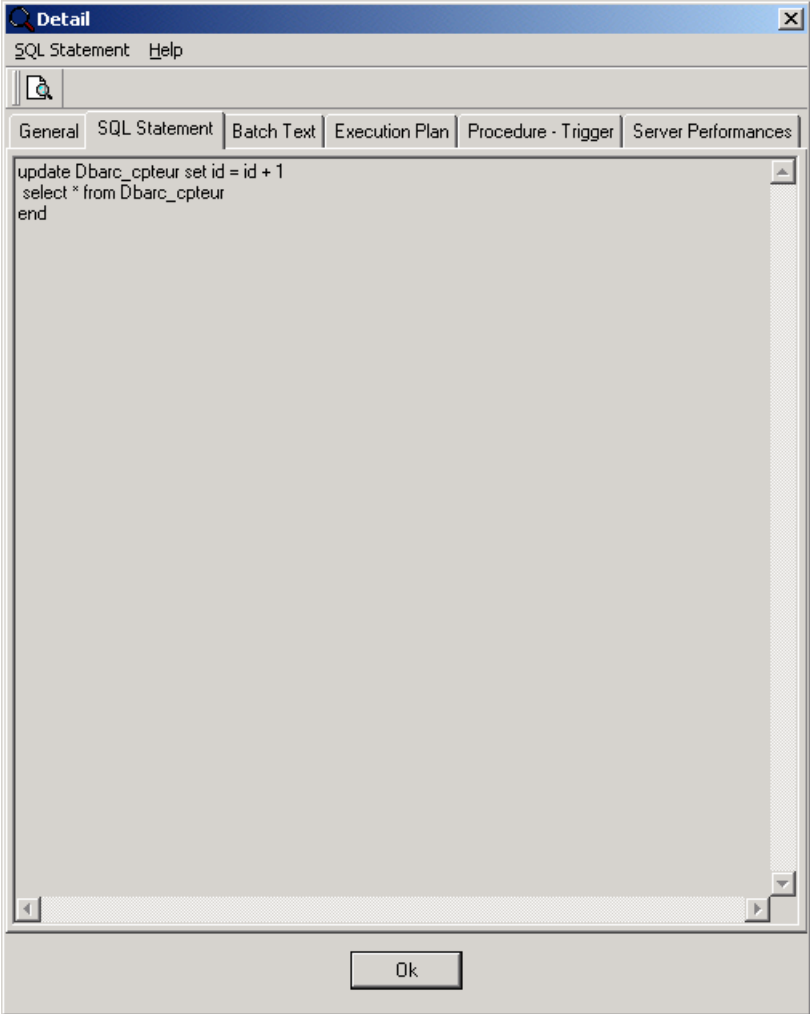
## Detail - General

This window displays the following data: spid, login name, host name, application name, date and time of the execution's start, date and time of the execution's end, and statistics of execution. Moreover, it shows the procedure's name or the trigger's name (if the SQL Statement is part of a Transact SQL object). Whenever there is a dynamic SQL statement, the detail tab reads: **\*\*NoDatabase\*\***. **\*\*NoOwner\*\***. **\*\*NoObject\*\***.



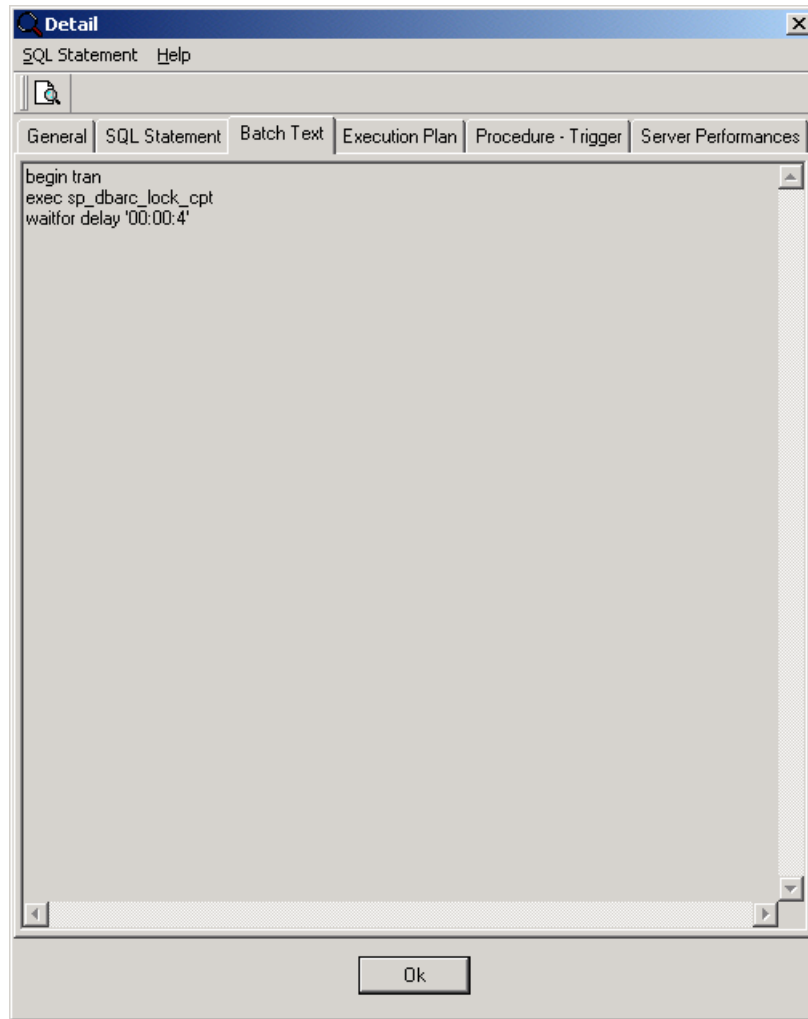
## Detail - SQL Statement

The SQL Statement can either be part of a SQL batch sent by the client application, or part of a stored procedure or a trigger. The execution duration of this SQL Statement exceeded the scan interval. This elapsed time threshold is triggered at the SQL Statement's level, not at the transaction's level. It is not set in motion by the duration for a stored procedure or of a trigger.



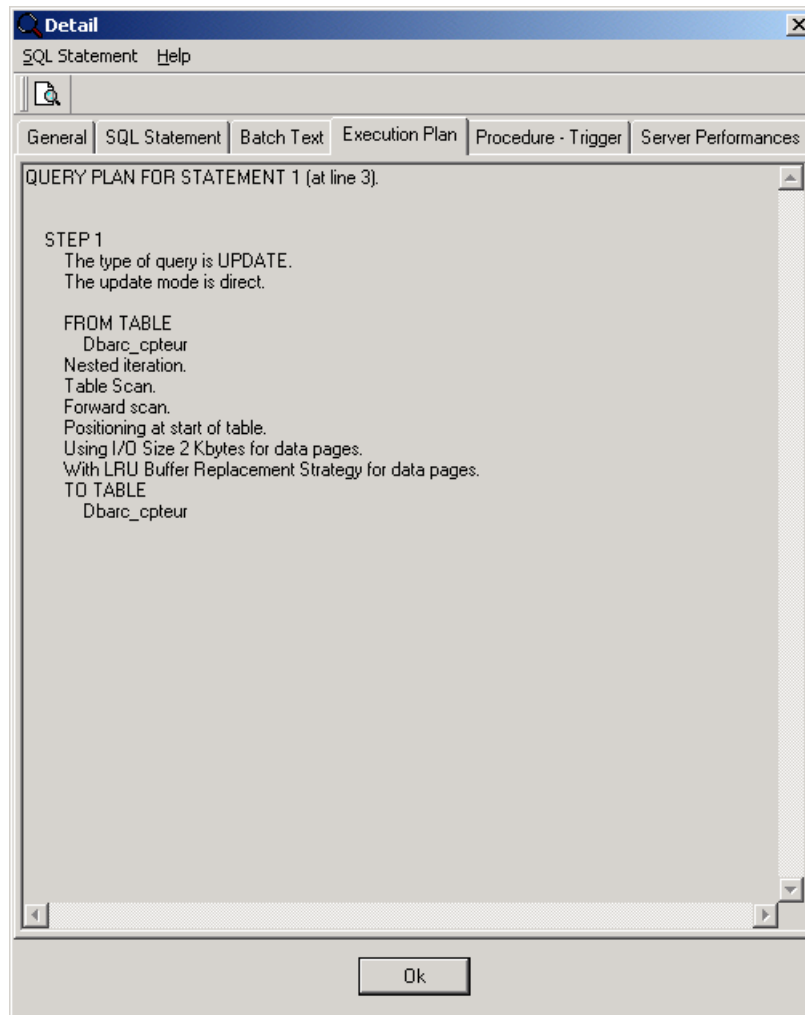
## Detail - Batch Text

This is the SQL batch sent by the Client Application to the SQL Server. (It contains a stored procedure call as well as its parameters, SQL BatchText, RPC, cursors...).



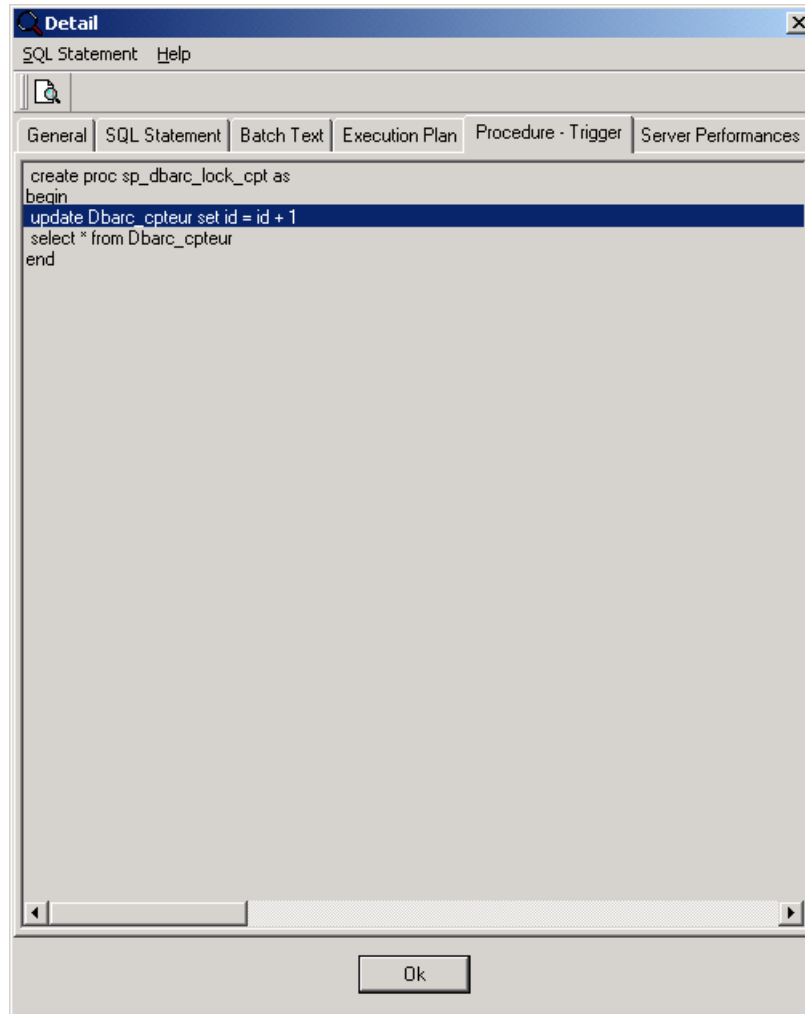
## Detail - Execution Plan

This is the SQL Statement's execution plan as computed by the Adaptive Server's query optimizer.



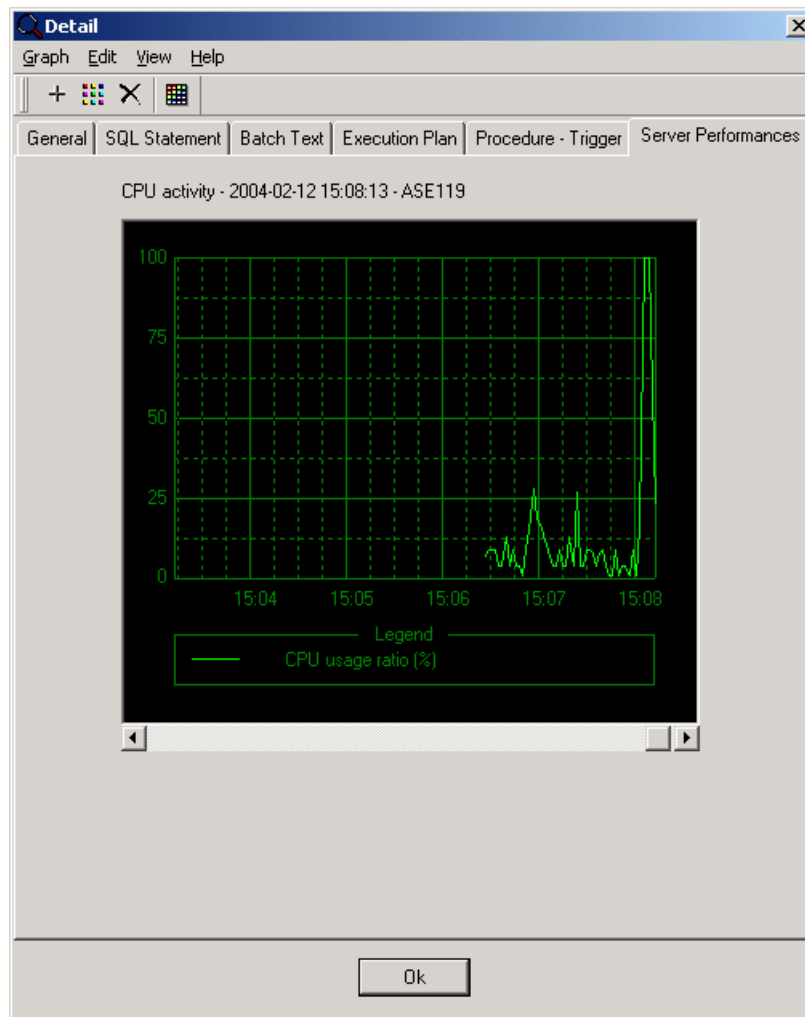
## Detail - Procedure - Trigger

This is the stored procedure or the trigger source code containing the captured SQL Statement (provided the latter is part of a procedure or trigger). In the source code text, the captured SQL Statement is selected. Whenever the SQL Statement is not a part of an Adaptive Server procedure or a trigger, it is a part of the Batch Text, namely a dynamic SQL. When it happens, the tab displays the following character string: "\*\*\*NoDatabase\*\*NoOwner\*\*NoObject\*\*".

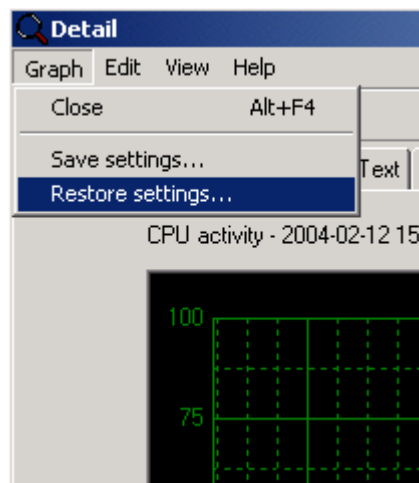


## Detail - Server Performances

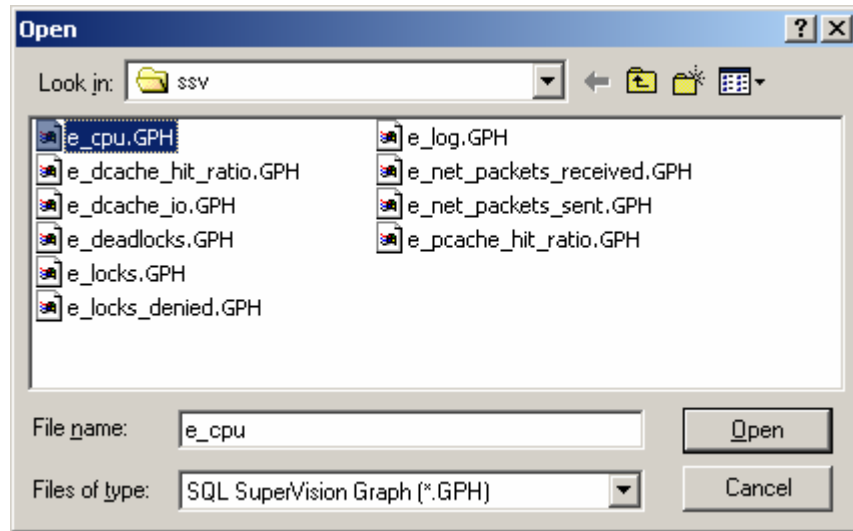
This tab displays the graph related to the chosen Adaptive Server's performance indicator. These are the values that have been collected during the execution of the current SQL Statement.



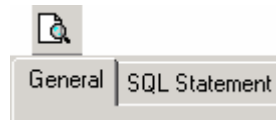
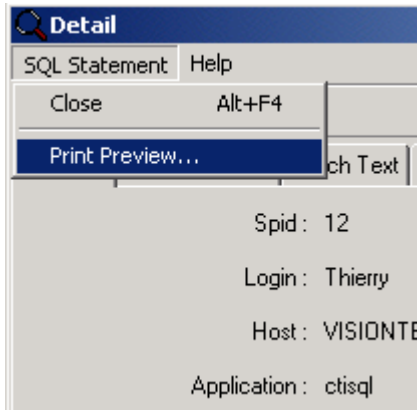
When selecting the "Graph - Restore settings..." menu item, you can view the different performance indicators that are available.



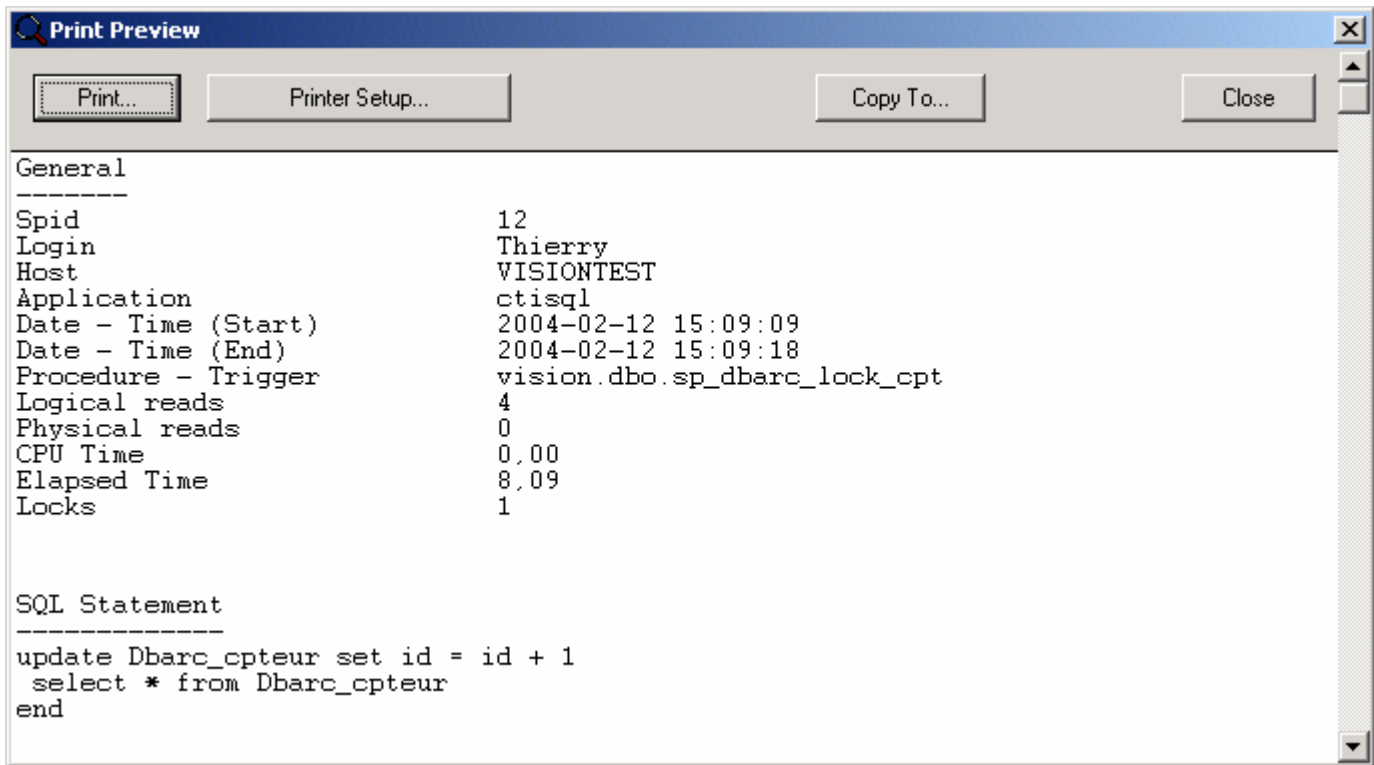
To select, open a configuration file previously configured with the following SQL SuperVision's feature "Adaptive Server's performances' monitoring". To learn how to save and change the graph configurations, see chapter on "Adaptive Server's performances' monitoring".



## Detail - Print

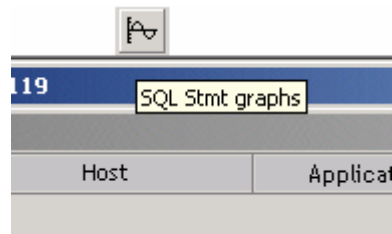
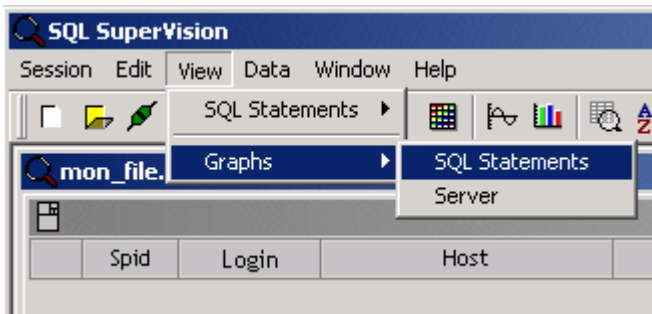


This feature allows you to print the contents of the "General", "SQL Statements", "Batch Text" and "Execution Plan" tabs. You can then select to print, configure the printing settings or copy the preview to a file.



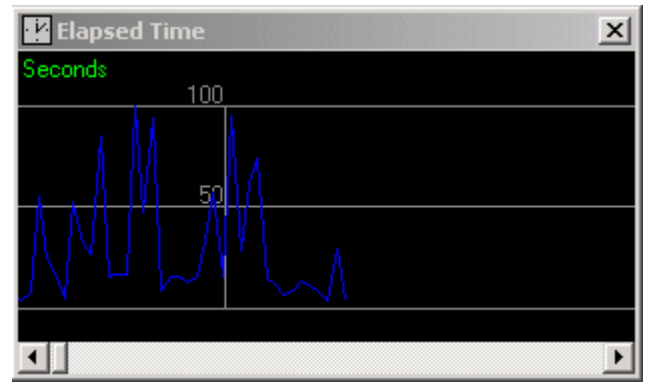
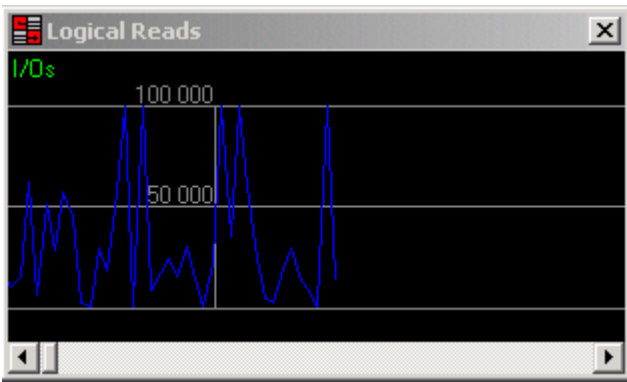
## SQL Statements' graphs

To display the SQL Statements' graphs, please select the "View - Graphs - SQL Statements" menu item. Alternatively, you can click on the "SQL Stmt graphs" menu icon (See Figures below). These graphs are only available for the completed SQL Statements.

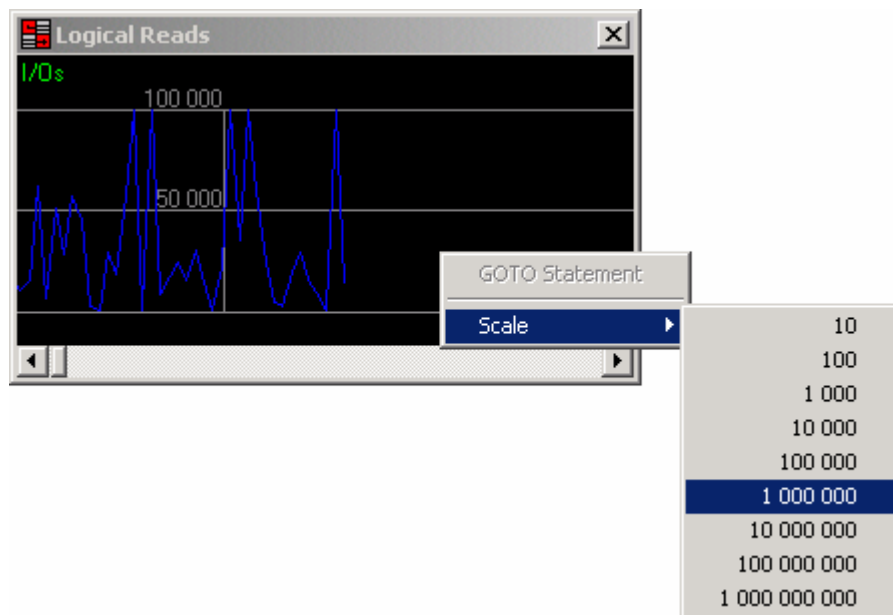


Two graphs are displayed:

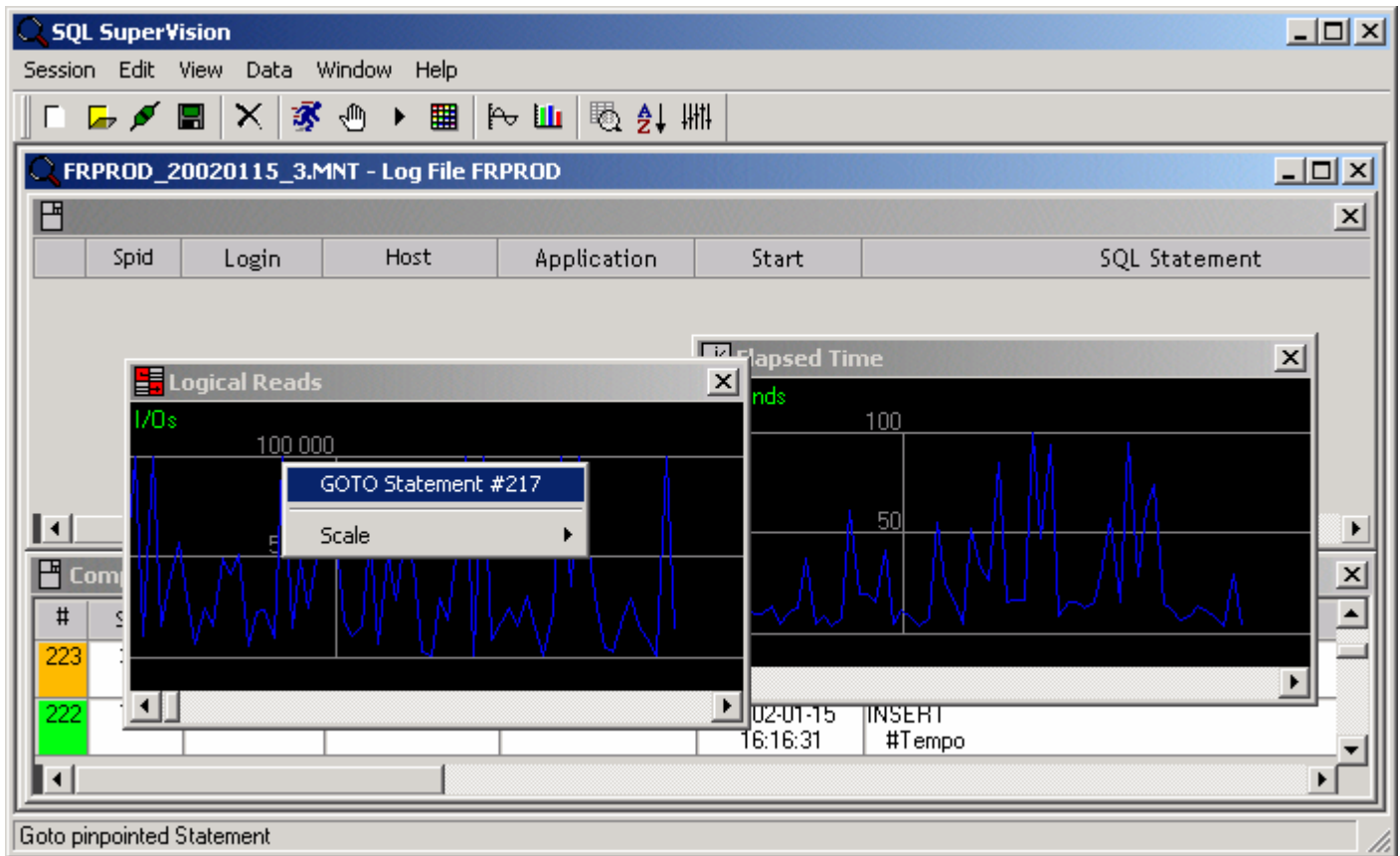




These graphs display the execution duration (elapsed time) in seconds as well as the logical inputs/outputs used by the SQL Statements. They vary according to the sort order selected for the session's report. Each SQL Statement corresponds to a point in the curve.



Depending on the values obtained, you can adjust the scale by clicking on the right button. A contextual menu appears. Select the most suitable scale. Thanks to these graphs, you can identify the peaks showing which SQL Statements use most logical I/Os or Elapsed times. To point to the SQL Statement generating a peak, right-click on the peak in question. Next, select the "GOTO Statement #(i)", from the contextual menu. (i) is the SQL Statement's line number (column "#" in the session's report). Once clicked, the selected SQL Statement is displayed at the top of session's report (in the completed SQL Statements' part).

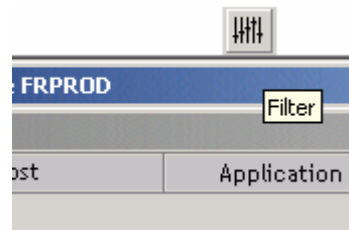
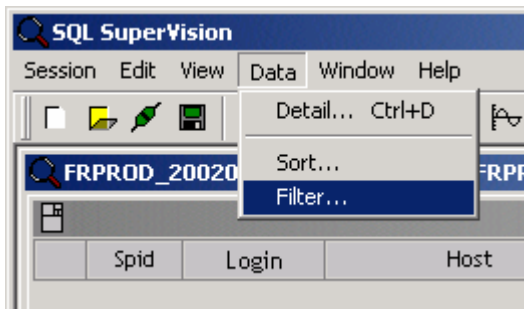


## Filters

See also:

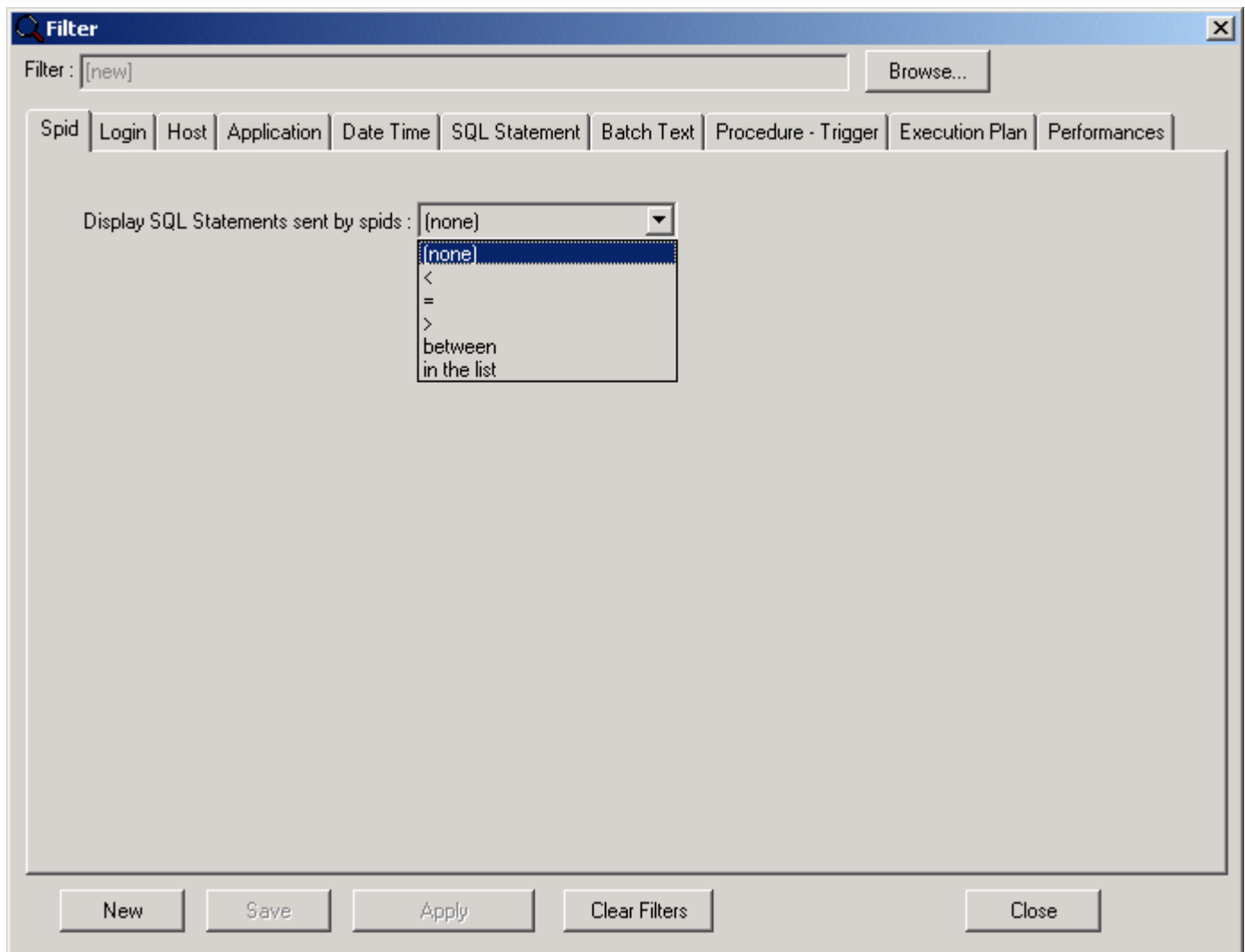
- Filter - Spid
- Filter - Login
- Filter - Host
- Filter - Application
- Filter - Date-Time
- Filter - SQL Statement
- Filter - Batch Text
- Filter - Procedure - Trigger
- Filter - Execution plan
- Filter - Performances
- Filter - How to manage filter files

Filters are used to reduce the quantity of SQL Statements displayed in the session's report or captured by SQL SuperVision. These filters can be applied to a session in progress or to an archive containing the history of previously recorded results. The filters affect completed SQL Statements' reports (bottom part of the window) and pending SQL Statements' reports (upper part of the window). To apply these filters, you need to select the "Data - Filter..." menu item. Alternatively, click on the "Filter" menu icon (See Figures below).



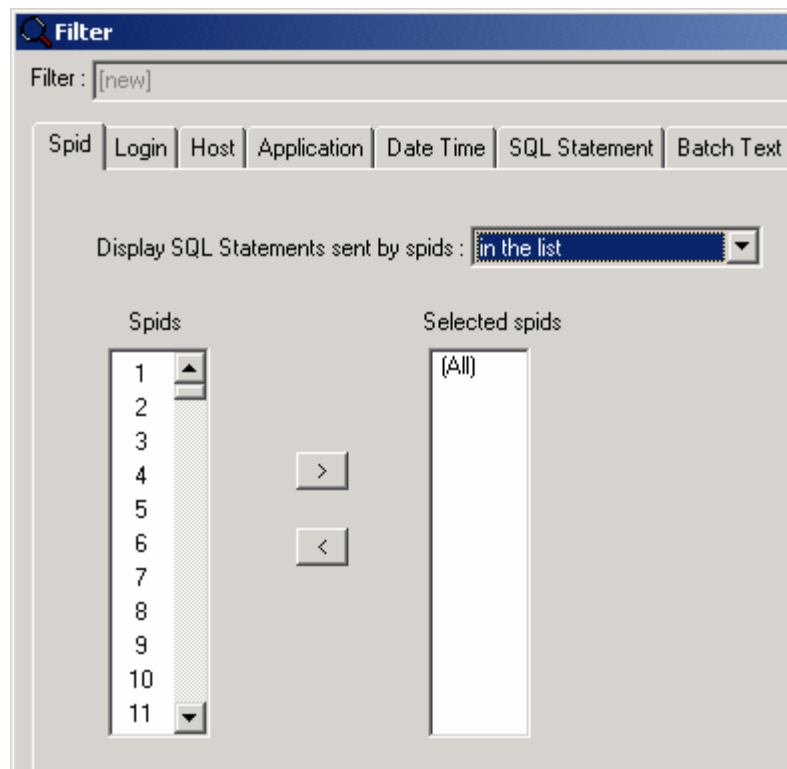
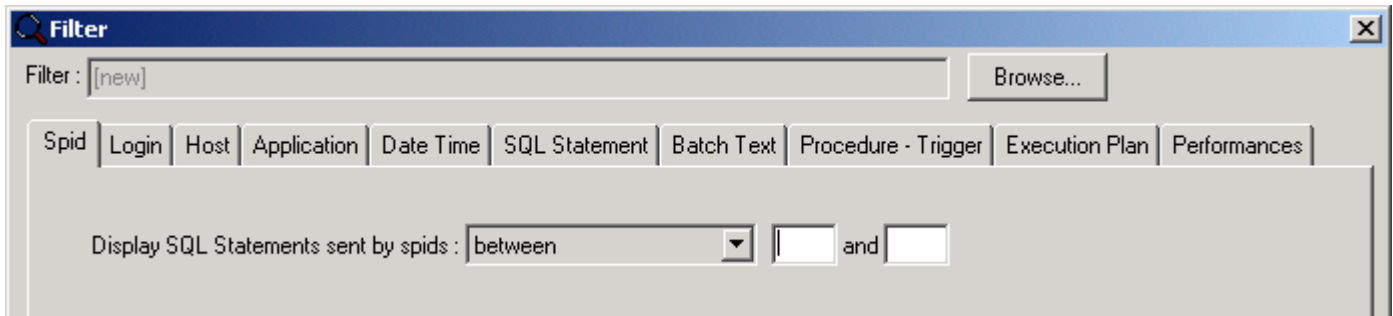
A window with ten tabs is displayed. Each tab represents a particular type of the session report's information. If several filter types are set, the filtered SQL Statements will need to fulfill all the defined filters (with an "AND" operator).

Example: A filter defines the "Paul" login and logical I/Os higher than 5000. The filtered SQL Statements are linked to the SQL Queries sent by the "Paul" login and consuming more than 5000 logical I/Os.



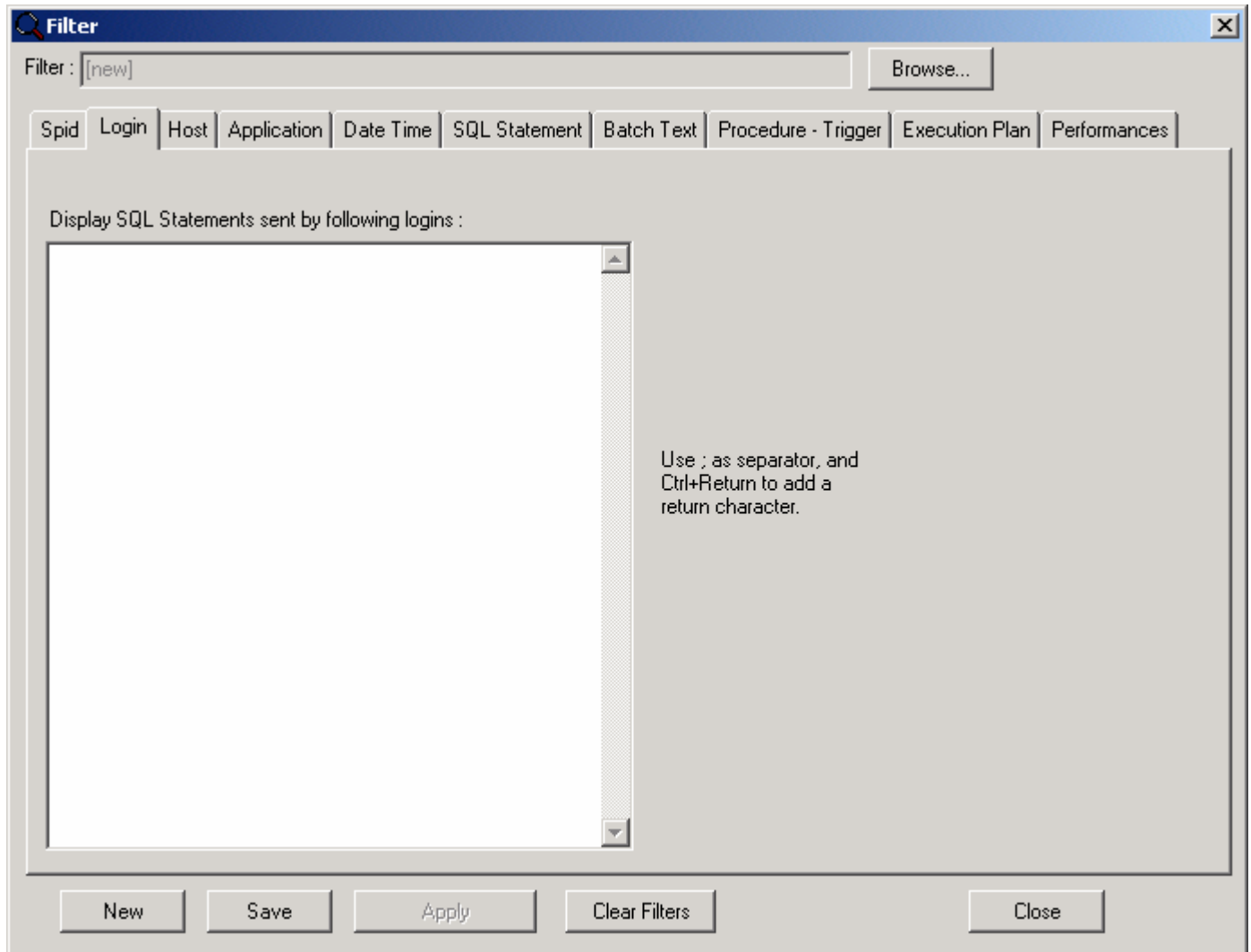
## Filter - Spid

Only the SQL Statements executed by the defined Spid(s) will be displayed. The list of spids can be set within a list or within a values interval.



## Filter - Login

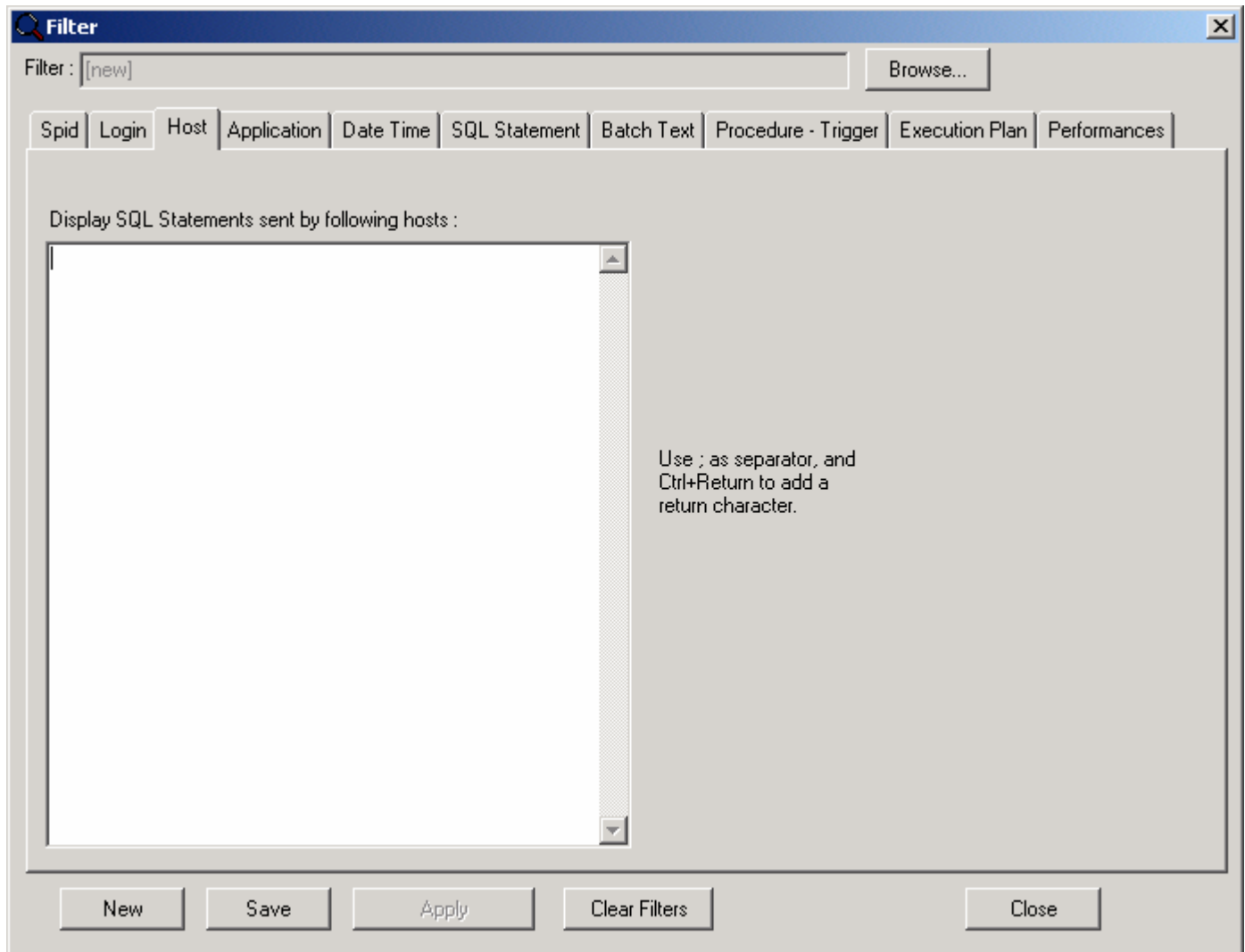
Only the SQL Statements executed by the defined logins will be displayed. The logins are entered in a list and are separated by a semicolon.



## Filter - Host

Only the SQL Statements sent by the defined workstations will be displayed. The hosts are entered in a list and they are separated by a semicolon.

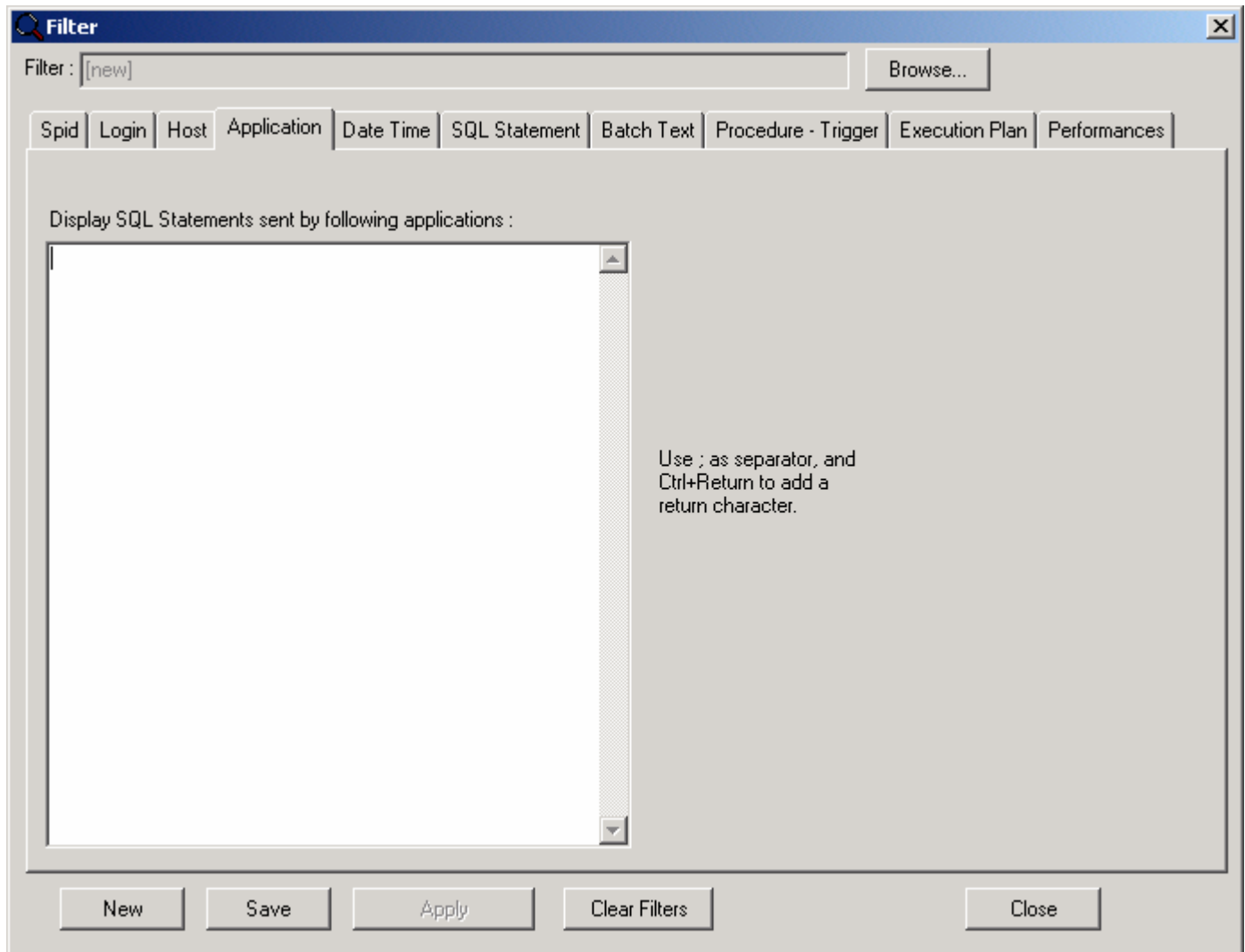
The "Host" information is the property of a given client application's connection. Since this property is optional, it may not always be mentioned. If you select this value as a filter, you need to make sure that the applications being monitored mention this property.



## Filter - Application

Only the SQL Statements executed in the defined applications will be displayed. The application names are entered in a list and are separated by a semicolon.

The "Application" information is a property of a given client application's connection. Since this property is optional, it may not always be mentioned. If you select this value as a filter, you need to make sure that the applications being monitored mention this property.



## Filter - Date-Time

This filter allows you to only display the SQL Statements that are being executed within a specific range of time. You can select to display the SQL Statements starting from a particular date and time or you can select a particular segment between two time ranges.

The screenshot shows a dialog box titled "Filter" with a search icon and a close button (X). At the top, there is a text field labeled "Filter : [new]" and a "Browse..." button. Below this is a horizontal menu with tabs: "Spid", "Login", "Host", "Application", "Date Time", "SQL Statement", "Batch Text", "Procedure - Trigger", "Execution Plan", and "Performances". The "Date Time" tab is selected. The main area contains the text "Display SQL Statements whose Date and Time of execution is" followed by a dropdown menu set to "between". To the right of the dropdown are two columns of input fields. The first column has a "Date" field with "0000-00-00" and a "Time" field with "00:00:00". The second column has a "Date" field with "0000-00-00" and a "Time" field with "00:00:00". At the bottom of the dialog are five buttons: "New", "Save", "Apply", "Clear Filters", and "Close".



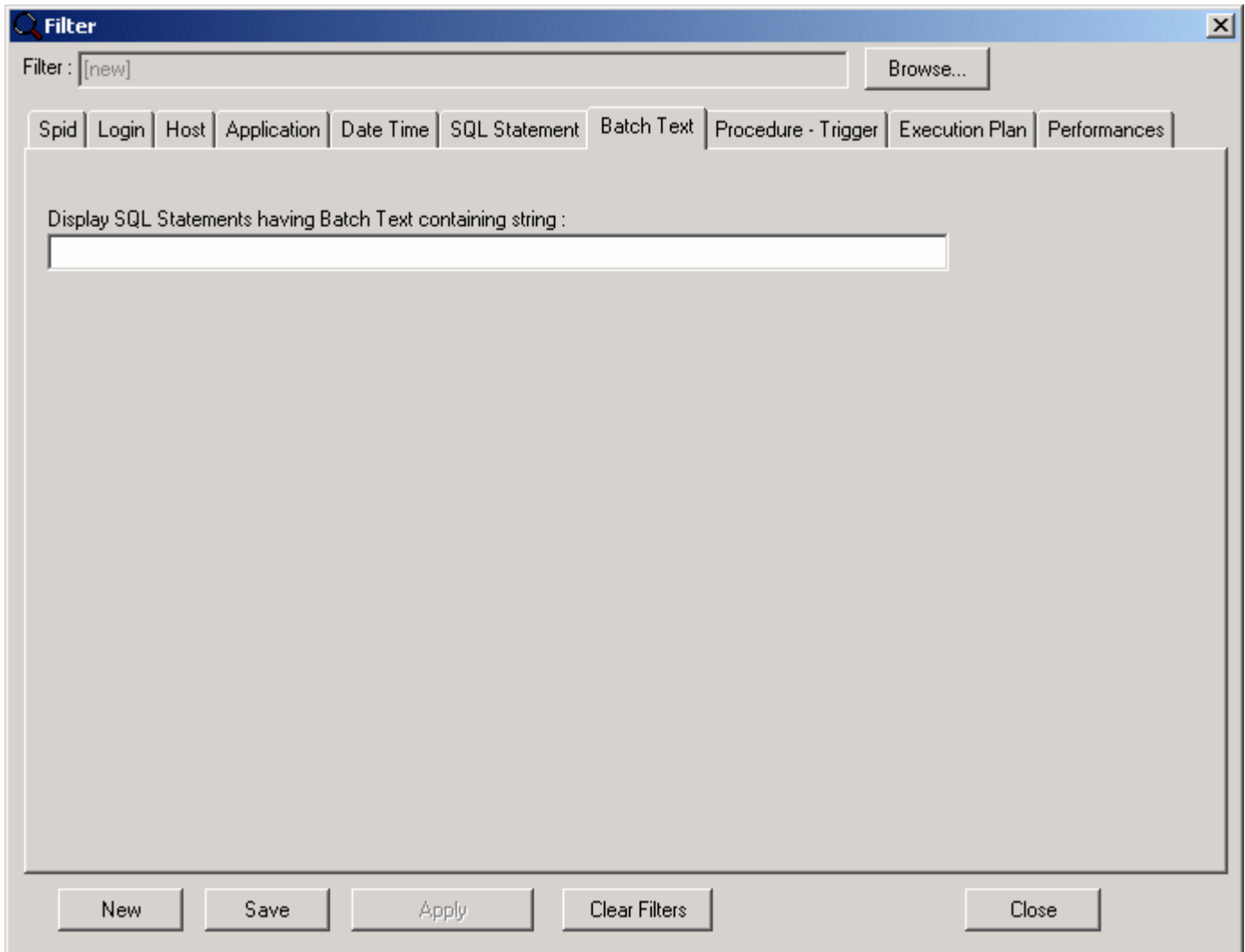
## Filter - SQL Statement

This filter allows you to only display the SQL Statements containing a defined character string (e.g., the name of a given table). Please note that the character string entered needs to match the SQL Statement's text exactly.

The screenshot shows a dialog box titled "Filter" with a search icon and a close button (X) in the top right corner. At the top, there is a text field labeled "Filter :" containing the text "[new]" and a "Browse..." button to its right. Below this is a horizontal row of tabs: "Spid", "Login", "Host", "Application", "Date Time", "SQL Statement", "Batch Text", "Procedure - Trigger", "Execution Plan", and "Performances". The "SQL Statement" tab is currently selected. The main area of the dialog is a large grey rectangle containing the text "Display SQL Statements containing string :" followed by a wide, empty text input field. At the bottom of the dialog, there are five buttons: "New", "Save", "Apply", "Clear Filters", and "Close".

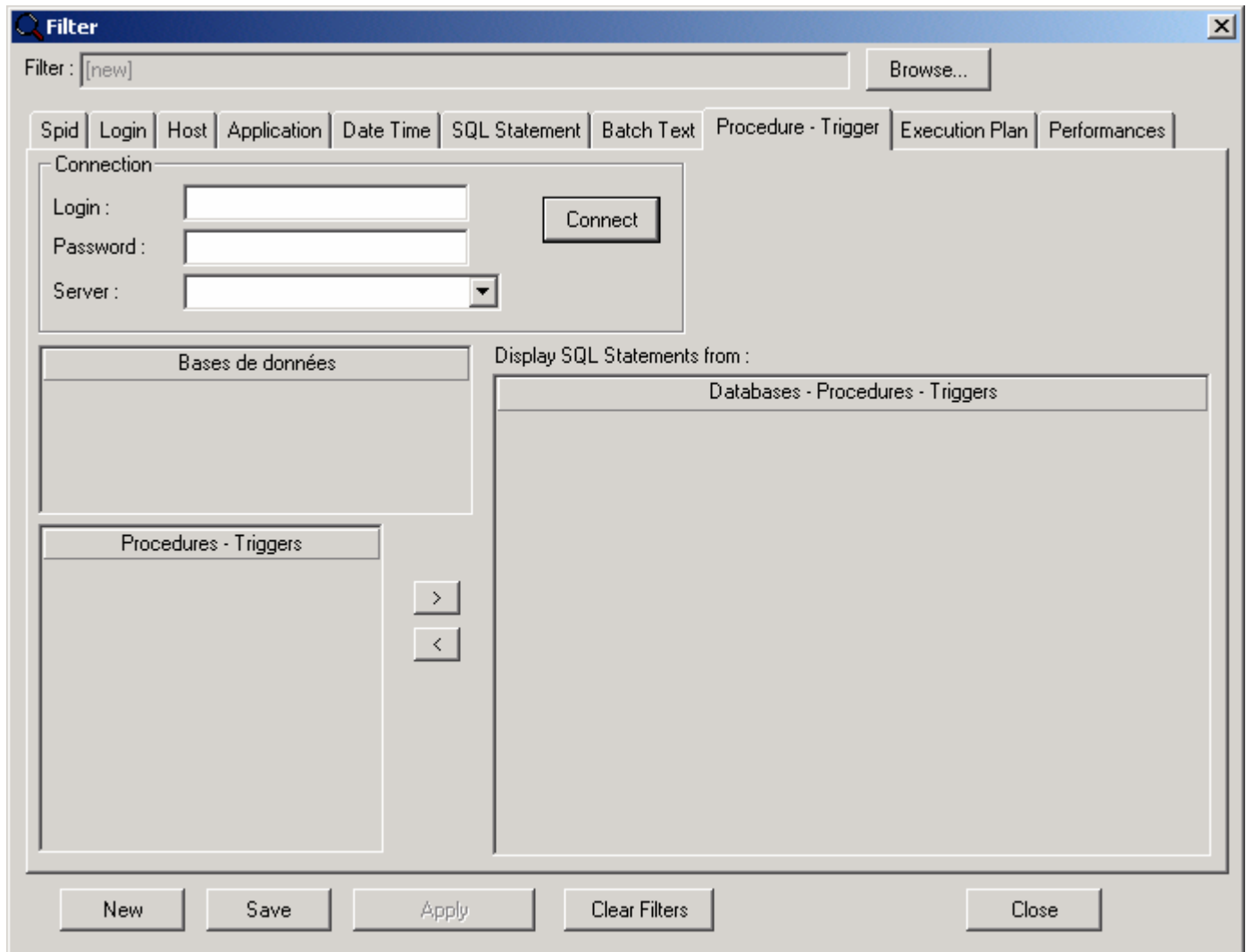
## Filter - Batch Text

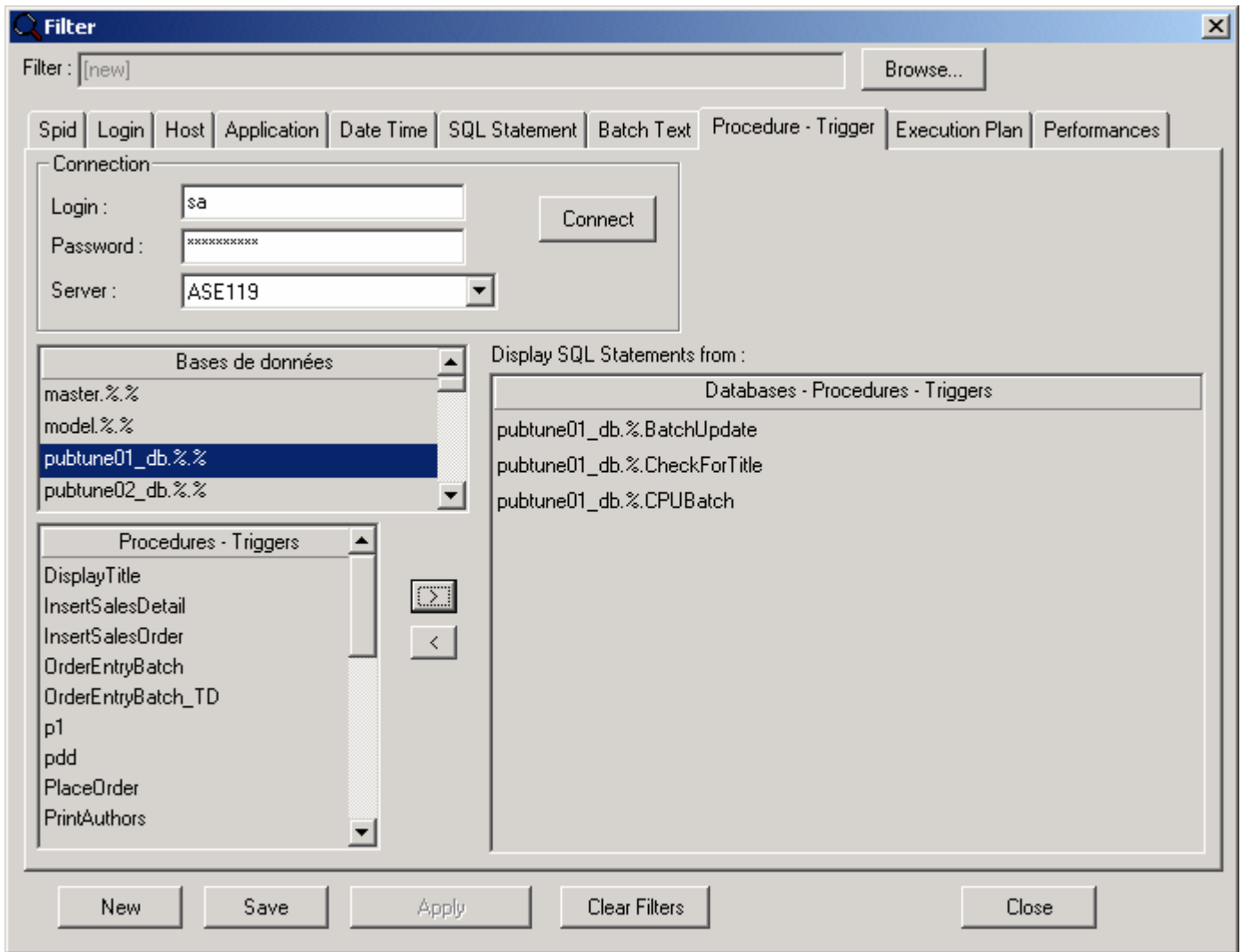
This filter allows you to only display the SQL Statements whose Batch Text contains a defined character string (e.g., the name of a given stored procedure). Please note that the character string entered needs to match the Batch Text exactly.



## Filter - Procedure - Trigger

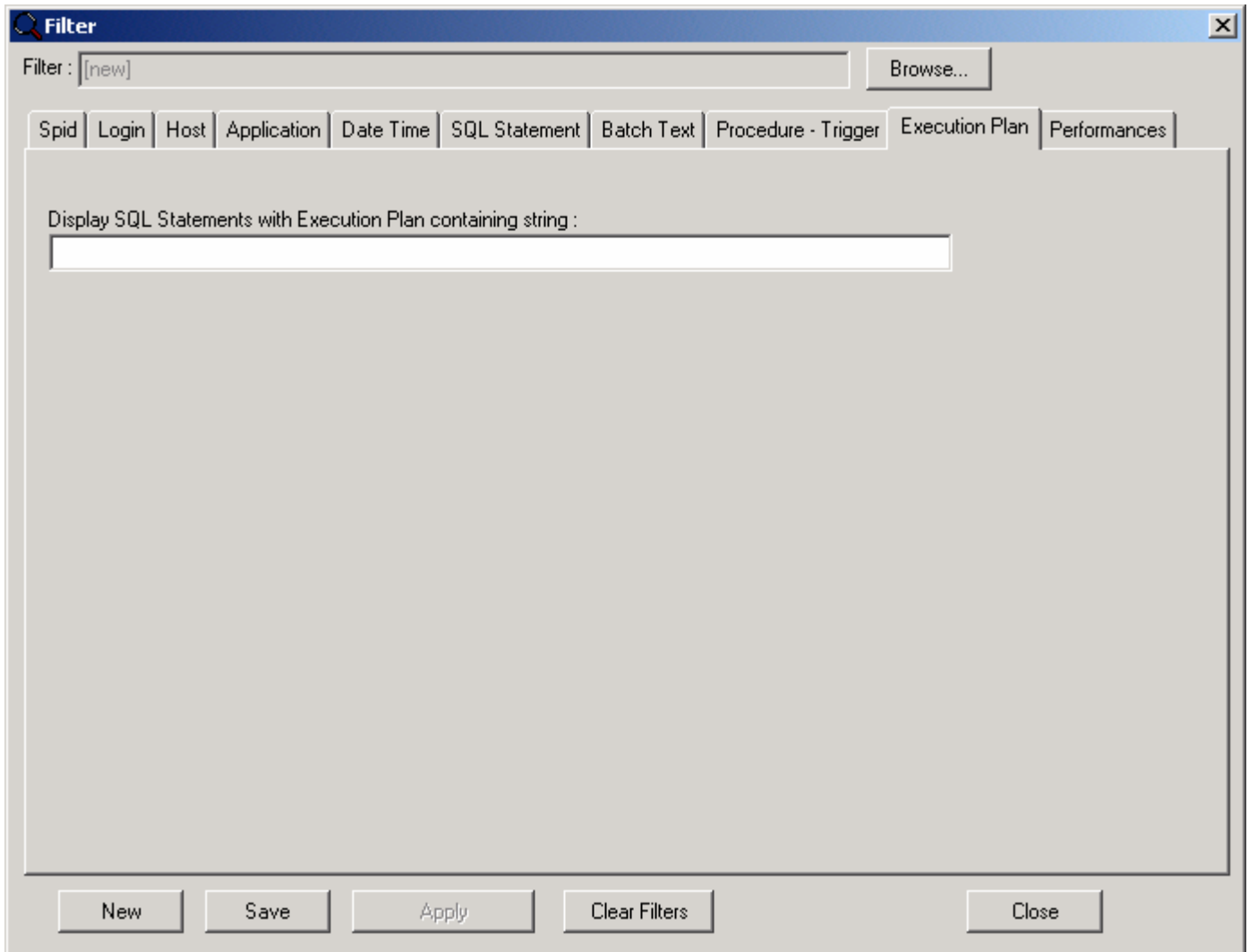
This filter allows you to only display the SQL Statements which are part of a defined list of stored procedures or triggers. To select the list, you first need to connect to the targeted Adaptive Server. Next, select the list of procedures or triggers required.





## Filter - Execution Plan

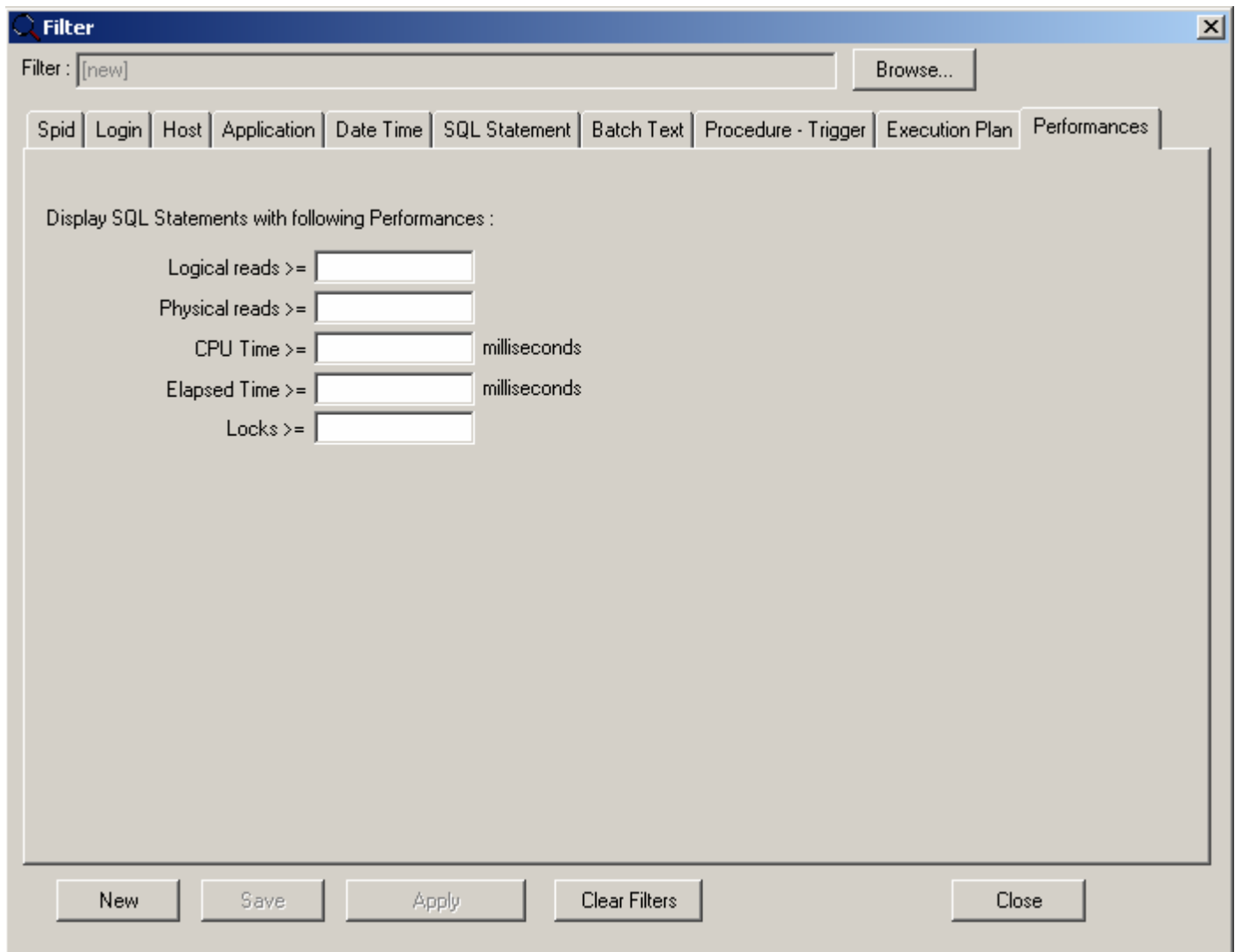
This filter allows you to only display the SQL Statements whose execution plan contains a defined character string (e.g., "Table Scan"). Please note that the character string entered needs to match the execution plan text exactly.



The screenshot shows a dialog box titled "Filter" with a search icon and a close button (X) in the top right corner. The "Filter:" label is followed by a text input field containing "[new]" and a "Browse..." button. Below this is a tabbed interface with the following tabs: "Spid", "Login", "Host", "Application", "Date Time", "SQL Statement", "Batch Text", "Procedure - Trigger", "Execution Plan", and "Performances". The "Execution Plan" tab is currently selected. Inside the dialog, there is a label "Display SQL Statements with Execution Plan containing string :" followed by a large empty text input field. At the bottom of the dialog, there are five buttons: "New", "Save", "Apply", "Clear Filters", and "Close".

## Filter - Performances

These various filters allow you to only display the SQL Statements whose performance indicators exceed the preset thresholds. By default, these thresholds are set to 0, except for the Elapsed Time threshold, which cannot be less than one second.



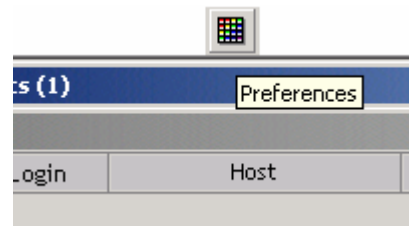
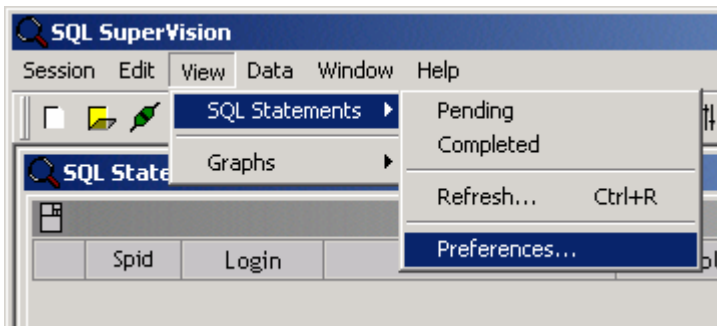
## Filter - How to manage filter files

Once you have defined the elements you need to filter, save these settings in a file. To do this, click on the "Save" button and select a filename with the FLT suffix. After clicking on the "Apply" button, this file is used to apply the filters. To retrieve previous settings, click on the "Browse" button to select a filter file. The "New" button is used to clear all settings from the various tabs. The "No Filter" setting unselects the filtering and displays the monitoring windows without applying any filter. By default, «.FLT» files are stored in %APPDATA%\SQL SuperVision\cfg\flt directory.

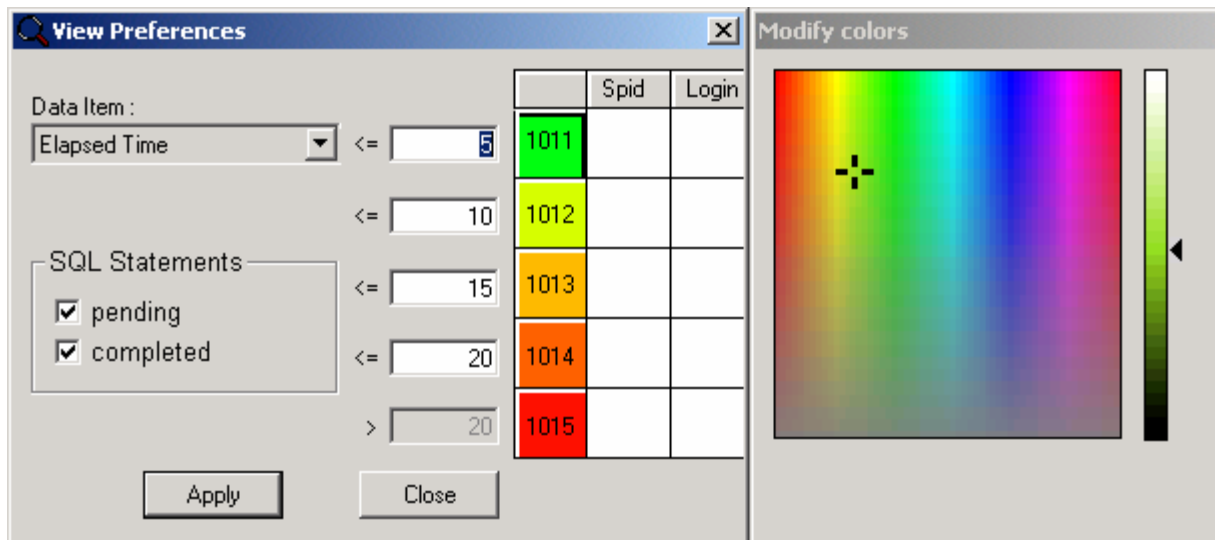
## Display options

For an easy identification of the most consuming SQL Statements, SQL SuperVision allows you to associate colors, defined by thresholds, with a specific performance indicator. By default, these thresholds are set to the "Elapsed Time" indicator and the colors range from green to red, based on the SQL Statement's execution duration.

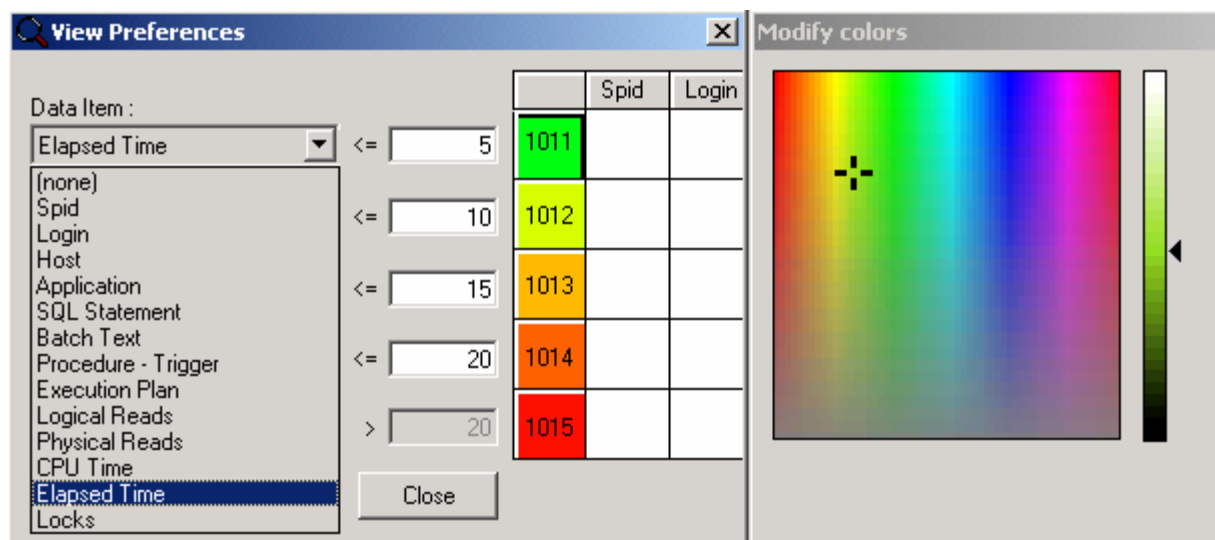
The display options make it possible to choose the performance indicator to which the thresholds will apply and to configure the values and colors associated with each threshold. To modify the display options, select the "View - SQL Statements - Preferences" menu item. Alternatively, click on the "Preferences" menu icon (See Figures below).



The following window is displayed:



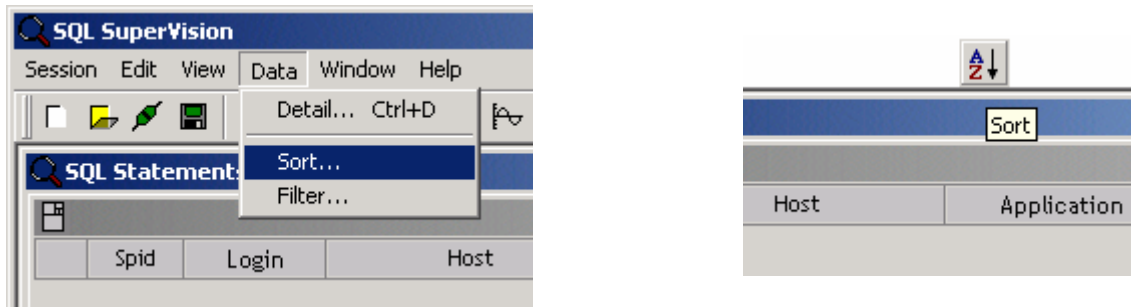
Select the performance indicator to which the thresholds will apply.



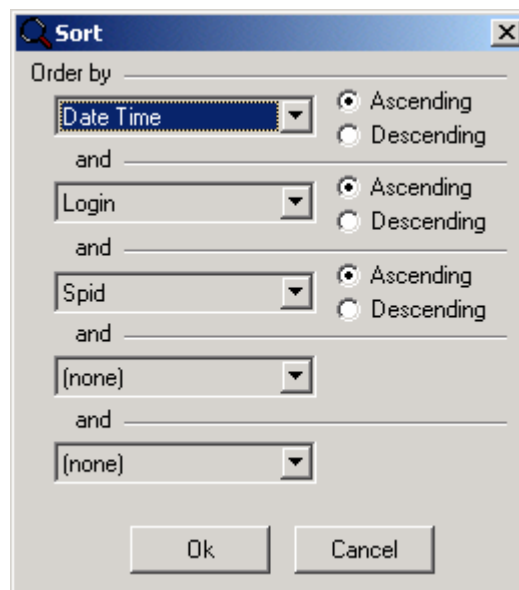
Choose colors as well as values for the various thresholds. Next, apply these display options to the pending SQL Statements, to the completed SQL Statements, or to both.

## Sort

To sort the lines of a given report, select the "Data - Sort..." menu item. Alternatively, click on the "Sort" menu icon (See Figures below).



The following window is displayed:



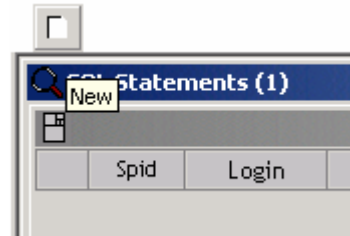
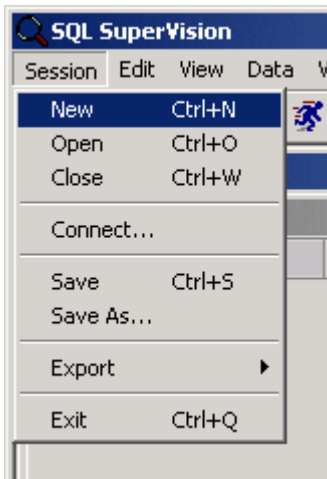
You will need to select the sort keys (up to five levels) as well as the direction (ascending or descending). Each sort key corresponds to a specific column of a given session's report.

Caution: For performance's sake, sorting is not allowed while viewing results in real time (Automatic Refresh).



## New Session

To create a new session, select the "Session - New..." menu item or click on the "New" menu icon (See Figures below). For performance's sake, we recommend that you not create several monitoring sessions simultaneously, but rather that you launch as many SQL SuperVision versions as needed for each monitoring session. The monitoring frequency of each session is evenly distributed by the Windows load-balancing task on the various processors of your computer. Because you can create new sessions, you can view several archive files simultaneously.



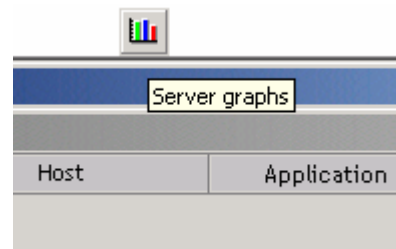
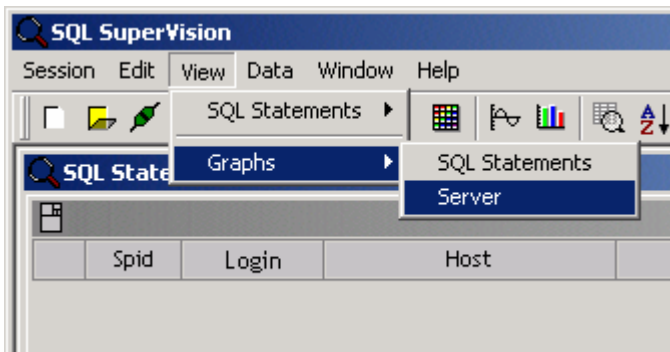
Within a new session, you can create a new connection to the Adaptive Server. To do so, use the ("Session - Connect..." menu. (See Recommendation above). You can also open a historical archive by using the ("Session - Open..." menu).

# Monitoring the Adaptive Server's performances

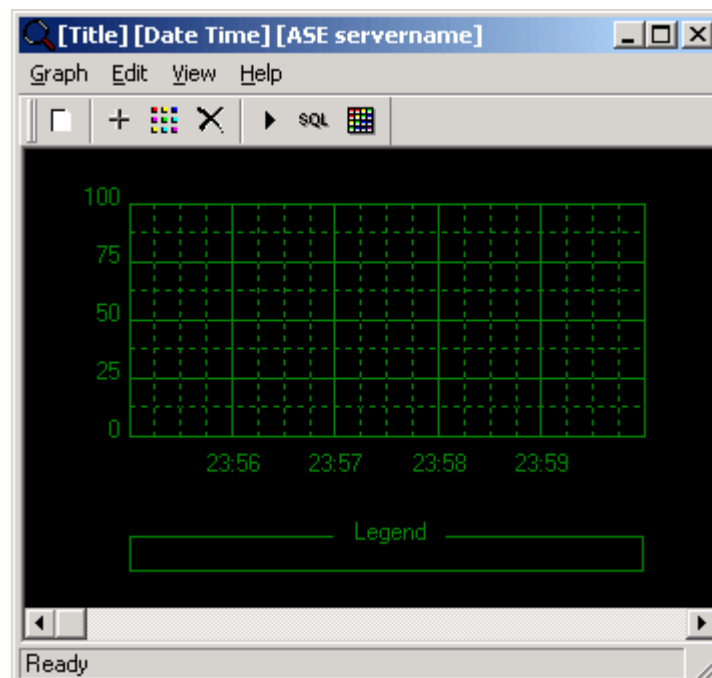
See also:

- How to start and stop a monitoring session
- The Adaptive Server's indicators
- How to restore a configuration
- How to add an indicator
- How to modify an indicator
- How to remove an indicator
- How to save the configuration
- How to refresh the graph
- SQL Statements related to a time period
- Graphs' display options
- New graph

To view the monitoring results relating to particular performance indicators of the Adaptive Server, select the "View - Graphs - Server" menu item from the session's report window. Alternatively, click on the "Server graphs" menu icon (See Figures below).



The following window is displayed:



From this window, you can view the Adaptive Server's performance curves. However, this window is not initialized. You first need to select the indicators that you want to see displayed.<sup>49</sup> You also need to define the appropriate graph's scale.

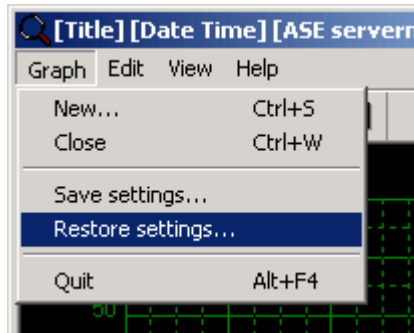
## The Adaptive Server's indicators

Column of the table	Column of the session's report	Description
Stmtid		Primary key, single identifier in this table. This number does not appear in the session's report in the graphical interface of SQL SuperVision. The line number displayed in the session's report corresponds to a numbering method appearing in the tabular window, as in MSExcels, to establish a link with the graph of the SQL Statements.
Keymonitor		SQL Supervision's license key from the PC workstation, which is connected to the Monitor Server. This information can be used to detect if several stations are connected to the same Monitor Server.
Status		"1" = SQL Statement is pending, "0" = SQL Statement is completed, "100" = SQL Statement is removed "logically" and consequently qualifies for a physical delete.
Spid	Spid	The SQL Server Process ID, which executes the SQL Statement.
Kspid		The Kernel SQL Server Process ID, which executes the SQL Statement. It is an internal identifier of the Adaptive Server that is incremented and allotted to each new connection.
Stmt_batchid		The query's number, within a client connection. This batch's identifier is augmented each time a new query is sent by a client PC workstation within a connection.
Login	Login	Login name of the connected user.
Servername	Window 's title	The Server's name of the monitored Adaptive Server name (entry of the SQL.ini file or interfaces file).
Hostname	Host	Name of the client PC workstation that sent the query to the ASE Server.
Appname	Application	Name of the application that sent the query to the ASE Server.
Date_time	Date Time (Start) Date Time (End)	In case of pending SQL Statements, it is the date and time at which the execution started. For completed SQL statements, it is the date and time at which the execution ended.
Batchid		Foreign key used to establish a link with the "ssv_batch" table.
Objectid	Procedure – Trigger (Put the following tables together: « ssv_object » and « ssv_textobject »)	Foreign key used to establish a link with the "ssv_object" and "ssv_textobject" tables. If "objectid" = "-1", the recorded SQL Statements belong neither to a stored procedure nor to a trigger and they correspond to a query sent as part of a dynamic SQL. Please note that the reverse is untrue: a recorded SQL Statement can belong to a stored procedure executed in a trigger that is activated by a query sent as part of a dynamic SQL.

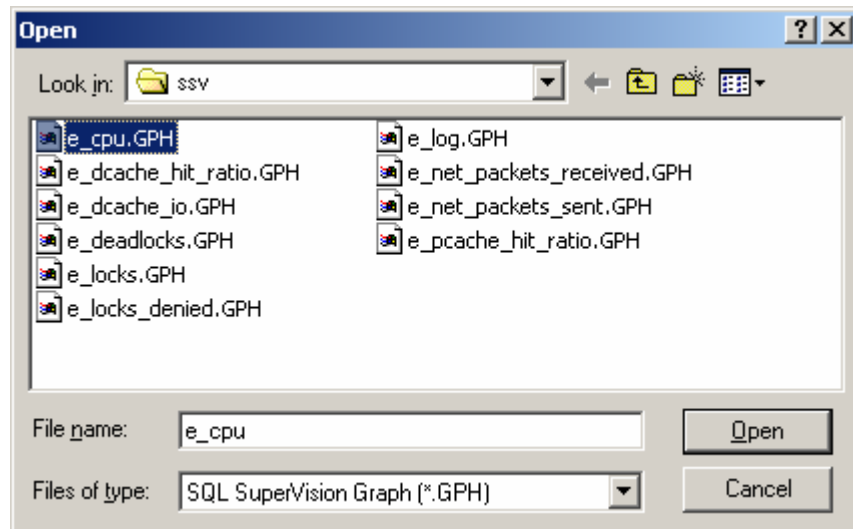
There are two ways to display an indicator. You can either use the preset graphs' configuration files or you can manually define the graph's aspect and its contents (See "How to save the configuration").

## How to restore a configuration

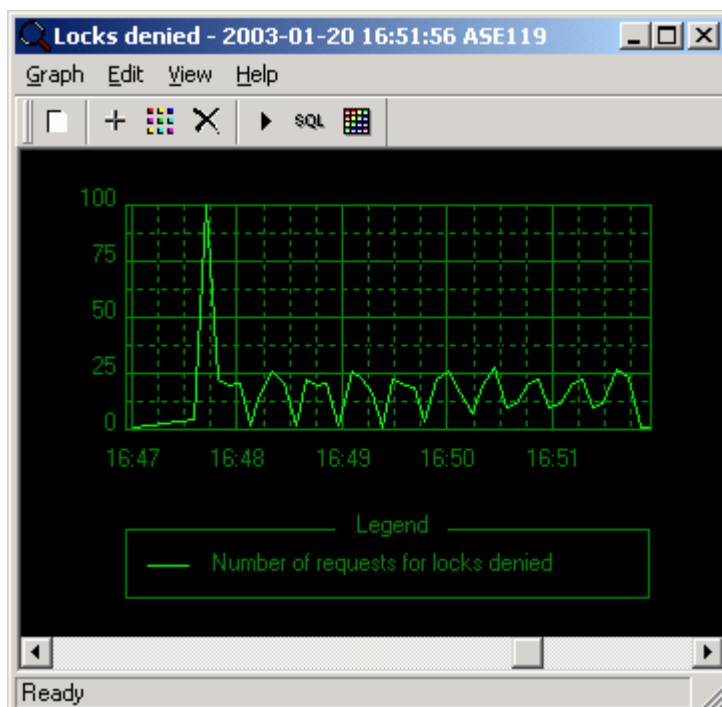
Select the "Graph - Restore settings..." menu item.



Next, determine the location and the name of the file containing the requested graph configuration. The default location is "%APPDATA%\SQL SuperVision\cfg\gph" directory.



When you have selected the indicator you want to view, the graph adapts automatically to the configuration saved in the file (See below). The configuration files mentioned above are provided by default with the SQL SuperVision's installation. However, you can create your own configuration files or modify those provided to suit your needs.

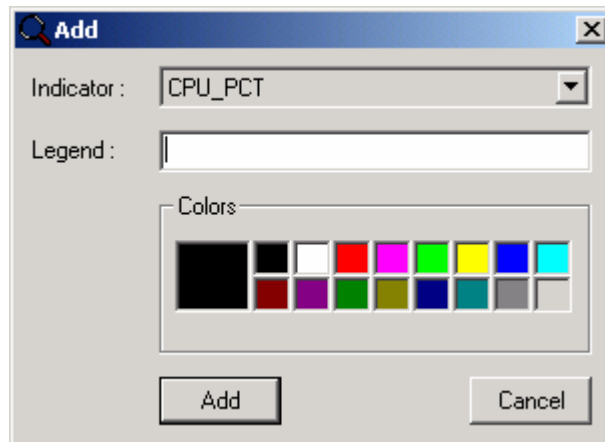


## How to add an indicator

To add a new indicator to your graph, select the "Edit - Add..." menu item. Alternatively, click on the "Add" menu icon, as illustrated below.



The following dialog box appears:

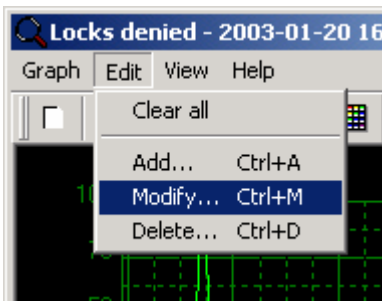


From this window, you can define the indicator you want to view. First, select the "Indicator" dropdown list box containing the various types described above (See "The server's indicators"). Next, enter the text you want to see displayed in the lower part of the graph. You can then select the color of the values' curve. When you click on the "Add" button, the graph changes immediately to reflect the new values recorded by the selected indicator.

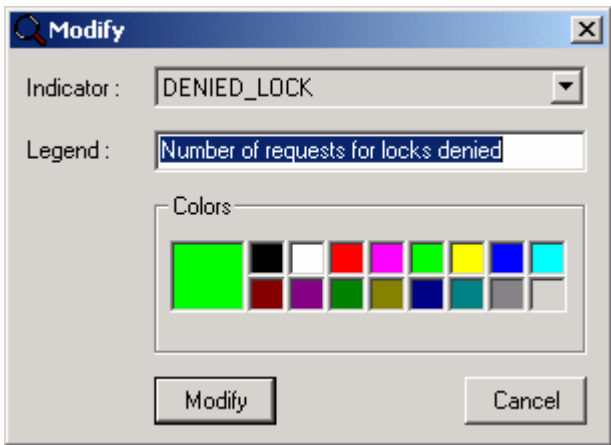
If the dropdown list box does not contain any value, it means that no monitoring session was created or that no values were read in the archive database or file. If the graph doesn't change after you have added an indicator, you need to change the graph's scale (See Paragraph below on "Graphs' display options"). The absence of change in the graph can also mean that there is no value for this indicator in the time range displayed in a given session's report. (If this happens, please contact [SQL SuperVision Technical Support](#) for a better assessment of the case.).

## How to modify an indicator

To modify the display of a specific indicator, select the "Edit - Modify..." menu item. Alternatively, click on the "Modify" menu icon, as illustrated below.



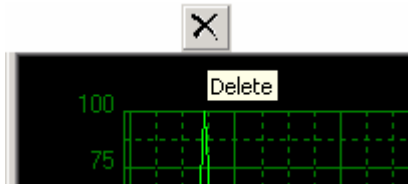
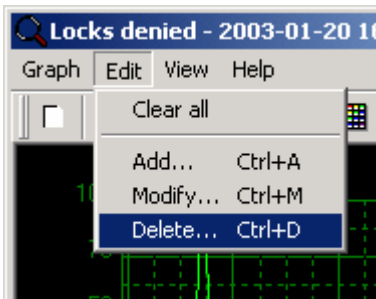
The following dialog box appears :



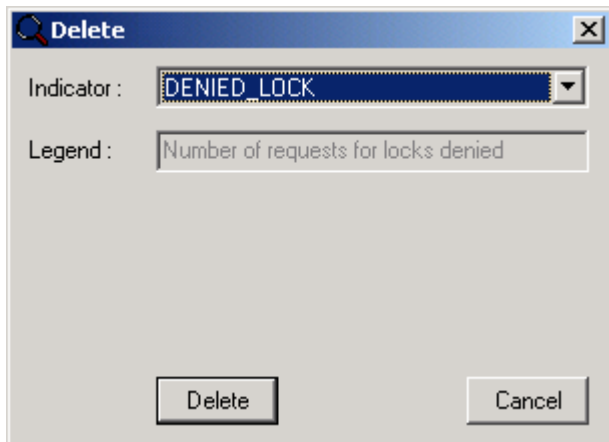
At this stage, you can modify the display characteristics that were previously added to the graph. To modify the texts and/or colors, select the indicators you need to change from the "Indicator" dropdown list box. Next, click on the "Modify" button. The graph changes automatically to reflect the new display settings.

### How to remove an indicator

To remove an indicator from your graph, select the "Edit - Delete..." menu item. Alternatively, click on the "Delete" menu icon, as illustrated below.



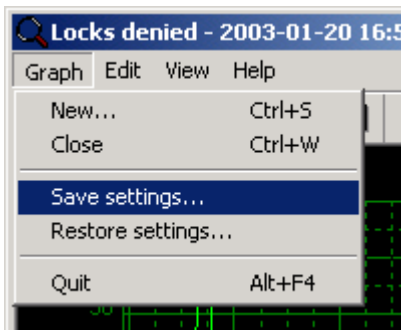
The following dialog box appears:



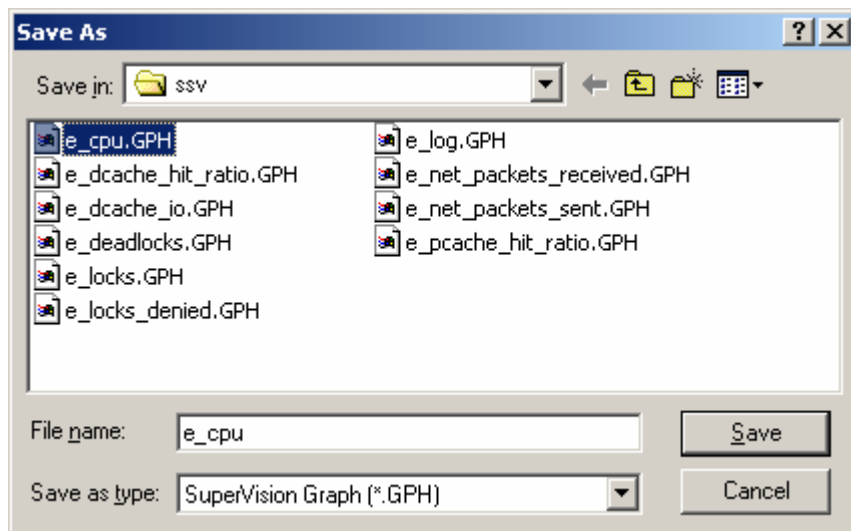
You can now remove indicators that were previously added to the graph. To do so, click on the "Delete" button. The selected indicator's values are immediately removed from the graph.

## How to save the configuration

To save a graph's given configuration, choose the "Graph - Save settings..." menu item.



You now need to determine the location and the filename containing the graph configuration you have just saved. By default, «.GPH» files are stored in %APPDATA%\SQL SuperVision\cfg\gph directory.



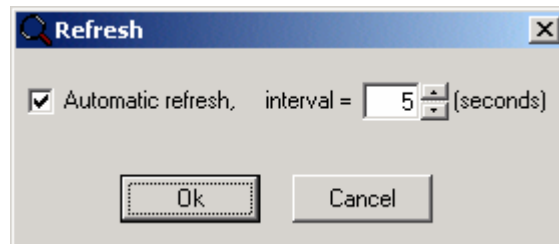
## How to refresh the graph



To perform the automatic refresh of the graph, select the "View - Refresh..." menu item. Alternatively, click on the "Refresh" menu icon (See Figures below).



This option makes it possible to refresh the display of a given report or graph, whether they are being produced locally or by a monitoring session in progress that is executed by another SQL SuperVision (See Save automatically). This option allows you to view the events captured by a workstation dedicated to monitoring from a distant client PC workstation (see SQL SuperVision's execution in a background task).

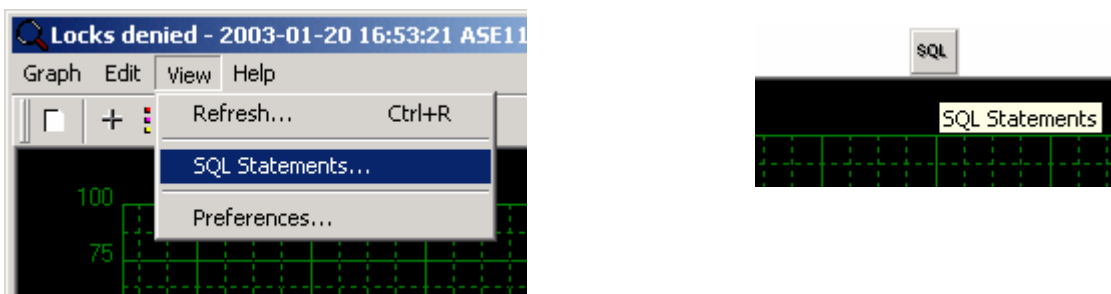
When you activate the automatic refresh function while reading a file or an archive database extract, a dialog box asks you to define the refresh frequency. In direct monitoring mode (SQL SuperVision is connected to an Adaptive Server), the automatic refresh function is performed with the same frequency as the monitoring (i.e. the refreshing frequency cannot be modified). However, you can interrupt the refreshing, if you wish to.



As soon as you activate the automatic function, the refresh icon changes from the "read" mode  to the "pause" mode .

## SQL Statements related to a time period

To view the SQL Statements captured during the time range displayed in the graph's window, select the "View - SQL Statements..." menu item. Alternatively, you can click on the "SQL Statements" menu icon (See Figures below).



The following window appears. It only contains the SQL<sub>55</sub>Statements that have been captured from the beginning



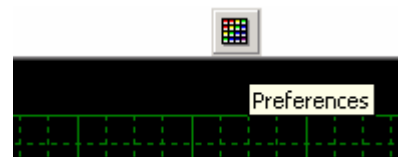
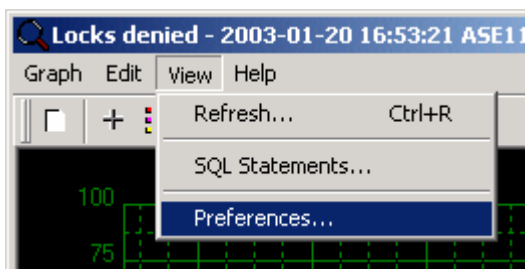
date-time of the graph (left) to the ending date-time of the graph (right).

#	Spid	Login	Host	Application	End	SQL Statement
83	12	Yves	VISIONTEST	ctisql	2003-01-20 16:51:14	waitfor delay '00:00:4' rollback
82	8	Thierry	VISIONTEST	ctisql	2003-01-20 16:51:14	update Dbarc_cpteur set id = id + 1 select * from Dbarc_cpteur
81	10	Vincent	VISIONTEST	ctisql	2003-01-20 16:51:11	waitfor delay '00:00:8' rollback
80	12	Yves	VISIONTEST	ctisql	2003-01-20 16:51:10	update Dbarc_cpteur set id = id + 1 where code = @code
79	1	Greg	VISIONTEST	ctisql	2003-01-20 16:51:09	waitfor delay '00:00:6' rollback
78	8	Thierry	VISIONTEST	ctisql	2003-01-20 16:51:03	waitfor delay '00:00:4'
77	8	Yves	VISIONTEST	ctisql	2003-01-20 16:50:59	waitfor delay '00:00:2' rollback
76	12	Greg	VISIONTEST	ctisql	2003-01-20 16:50:56	waitfor delay '00:00:3' rollback
75	8	Yves	VISIONTEST	ctisql	2003-01-20 16:50:56	update Dbarc_cpteur set id = id + 1 where code = @code
74	12	Greg	VISIONTEST	ctisql	2003-01-20 16:50:53	update Dbarc_cpteur set id = id + 1 where code = @code
73	1	Thierry	VISIONTEST	ctisql	2003-01-20 16:50:53	waitfor delay '00:00:4'

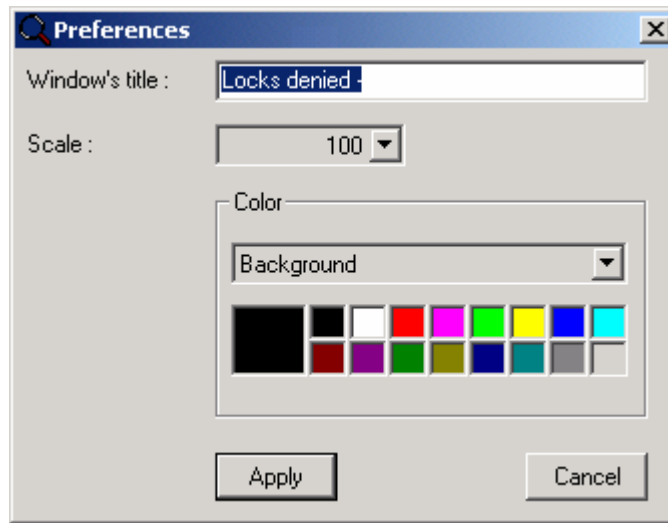
From this window, you can export the data to Excel or to a .txt format file. You can also display the SQL statements' graphs and view the details of a given SQL Statement. These actions obey the same principles as those concerning a monitoring session's report.

## Graphs' display options

You can modify the graphs' display options (including the graph's title and scale, the window's fill color and the graph grid's color). To do so, select the "View - Preferences..." menu item. Alternatively, click on the "Preferences" menu icon (See Figures below).



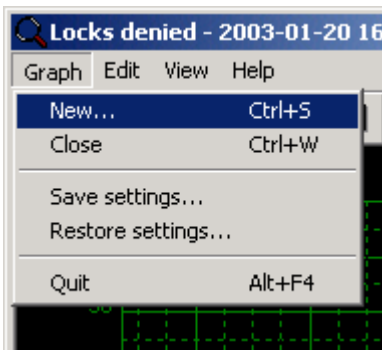
The following dialog box appears :



The preferences defined from this dialog box can be saved in a configuration file (See Paragraph on "How to save the configuration").

## New graph

To create a new graph, select the "Graph - New..." menu item. Alternatively, click on the "New" menu icon (See Figures below). You can display as many graphs as you wish, depending on your PC workstation's capacity.



If at any time you need to re-display a graph that was left in the "close" menu box, simply activate the "Graphs - Server" function from the session's report window.

## How to save results

See also:

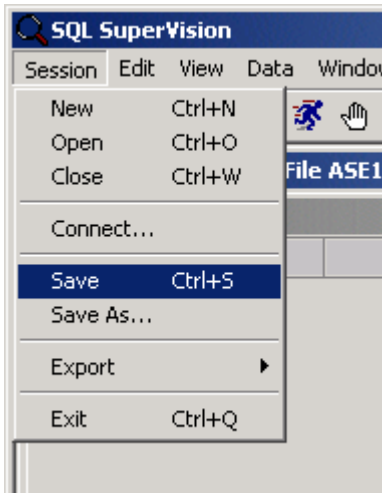
How to save to a database

How to save to a file

How to save automatically

Exports

To save the contents of a session's report, select the "Session - Save/Save As..." menu item. Alternatively, click on the "Save" menu icon (See Figures below).



Next, you have two ways of saving your information: you can save it to a file or to a database.

## How to save to a database

You can save the contents of a session's report to an Adaptive Server's database. There is no technical prerequisite attached to this database. The database's physical model is provided in the appendix (See Database model).

This feature is only available in the SQL SuperVision's Enterprise Edition.

## Technical recommendations

- 1) For performance's sake, the archive database should not be located on a WAN (from the SQL SuperVision collecting the monitoring results).
- 2) We recommend that you not install this database on a production server.
- 3) Volume required for the archive database:

You need to allow for 10 Kb per captured SQL Statement, on average, and for 1 Mb, for Adaptive Servers' indicators values, per monitoring hour. As an example, it is typical to capture more than 5000 SQL Statements per day on a targeted production Adaptive Server experiencing performance problems. This represents a total volume of 50 Mb for the SQL Statements and of 24 Mb for the Adaptive Server's indicators values. Approximately 75 Mb are delivered, each monitoring day. For SQL SuperVision's stored delete procedures, see Chapter on "Results deletions". It is necessary to manage the archive database's free space. (A good way to do this is to set alarm thresholds to the "default" setting and to "logsegment" segments. We recommend that you create a database with a minimum size of 1 Gb for data and of 200 Mb for logs.

## Access management

The login used to record the results in the archive database must be that of the database owner (DBO). The users who view the information are simple "users" in the "public" group of this database. They are able to execute deletes on the results (see Chapter on "Results deletions").

The viewing of information within a centralized database can be of critical importance to your security. If such is your

intention, you can create views as follows:

- 1) You create a login (sp\_addlogin)
  - 2) You allow him to use the archive database (sp\_adduser)
- You modify the view "ssv\_stmt\_user" :
- ```
Drop view ssv_stmt_user
go
create view ssv_stmt_user
as
select * from ssv_stmt
where < criteria of restriction >
```

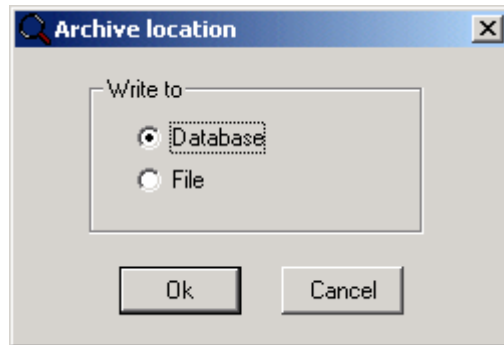
example:

```
drop view ssv_stmt_user
go
create view ssv_stmt_user
as
select * ssv_stmt
where (user_name() = "develop" and servename = "SERVPROD" and appname = "PORTFOLIO")
```

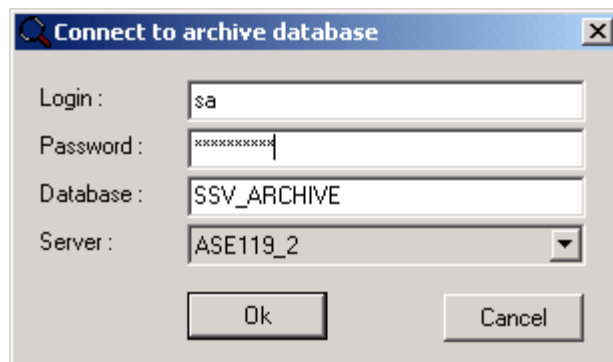
The "PORTFOLIO" application developers can only view the performance results of their own stored procedures. Do not hesitate to contact [SQL SuperVision's technical support](#) to validate your criteria concerning the SQL SuperVision's database model and performances using indexes.

## Connecting to the archive database

From the following dialog box, select the required location for the archive.



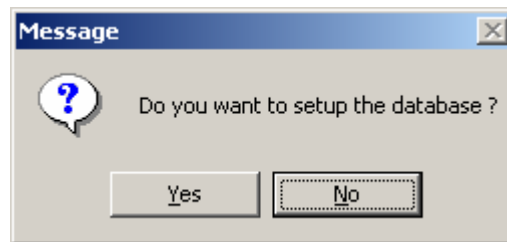
After selecting "Database", you connect to it.



If the database is not yet formatted or if its structure has been modified, SQL SuperVision displays the following message.



Next, SQL SuperVision suggests that you execute the full database setup. The connected login must be that of the DBO (database owner).



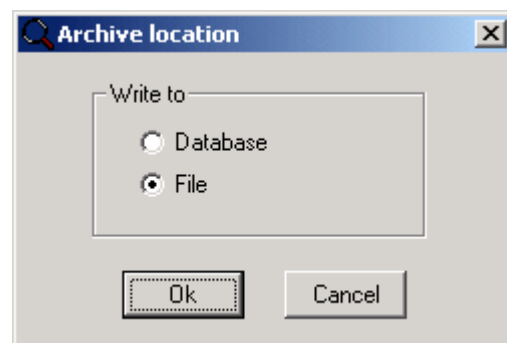
Caution: The SQL SuperVision package does not come with a specific procedure to restore data. If you choose: "yes" (database setup), all previous data will be lost. While saving and restoring Adaptive Server databases, use the tools provided and certified by Sybase.

## How to save to a file

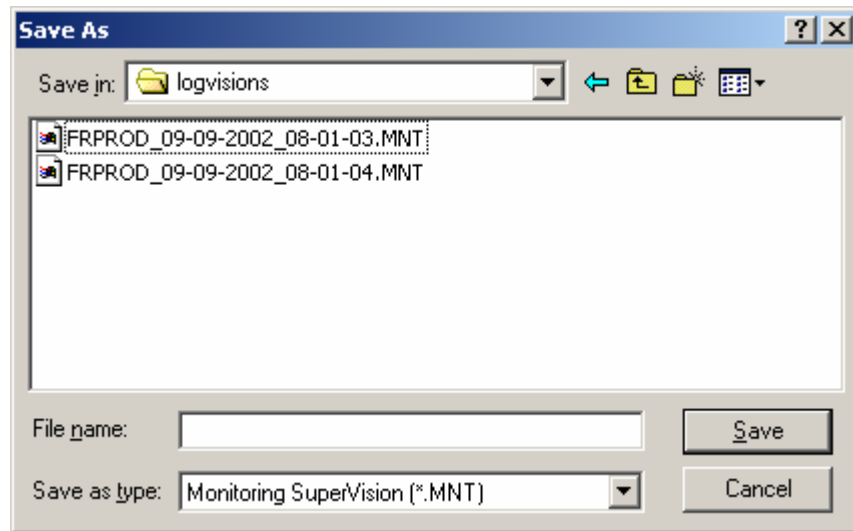
Recommendations:

- 1) For performance's sake, the archive file containing the monitoring results should not be located on a WAN (from the SQL SuperVision version recording the results).
- 2) We recommend that you not create this file on a production server.
- 3) Volume required for the archive file:  
You need to allow for 10 Kb per captured SQL Statement, on average, and for 1 Mb, for Adaptive Servers' indicators values, per monitoring hour. As an example, it is typical to capture more than 5000 SQL Statements per day, on a targeted production Adaptive Server experiencing performance problems. This represents a total volume of 50 Mb for the SQL Statements and of 25 Mb for the Adaptive Server's indicators values. Approximately 75 Mb are delivered, each monitoring day. It is imperative that you manage the free space that is on the archive directories.

From the following dialog box, select the required location for the archive.

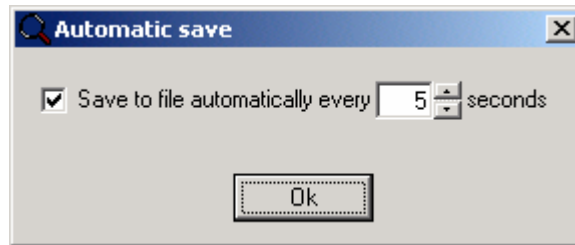


After selecting "File", you need to define the location and name of the file. The default location is "%APPDATA%\SQL SuperVision\Data Files" directory.



## How to save automatically

When a monitoring session is active, a window appears to setup the session's reports automatic-save frequency which is defined in seconds. The newly captured SQL Statements are then automatically saved to the file at the selected frequency. The automatic-save frequency operates independently from the monitoring frequency. (and the latter cannot be modified.).



## Exports

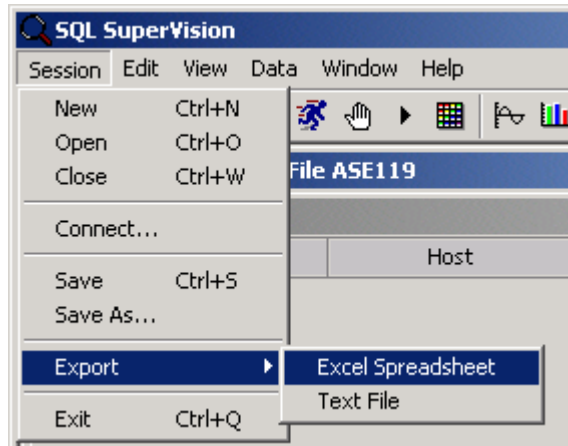
See also:

How to export to Excel

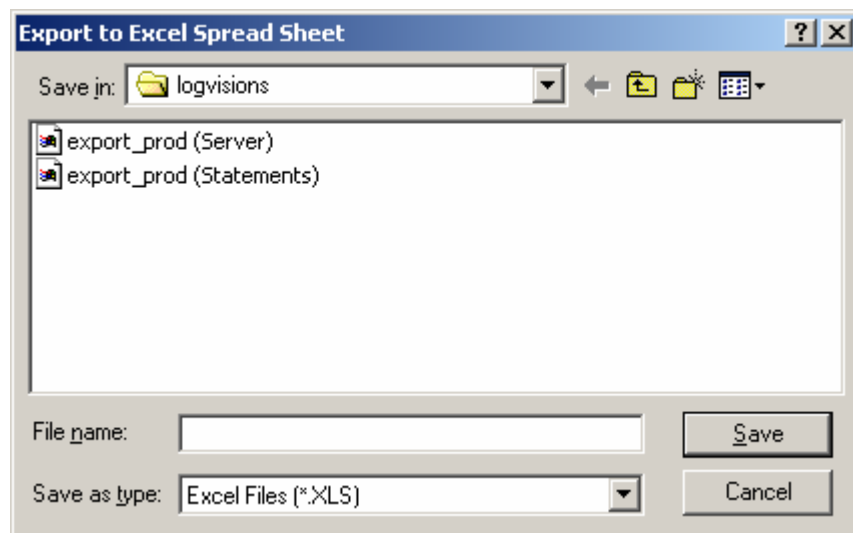
How to export to a text file

SQL SuperVision enables you to export all the SQL Statements or part thereof to a file (in Excel or Text format). Instead of exporting results to a database, you can save them to a database (see Chapter on "How to save to a database"). To do this, select the SQL Statements you wish to export. Next, select the "Session - Export" menu item and determine the file format required to export the data.

### How to export to Excel

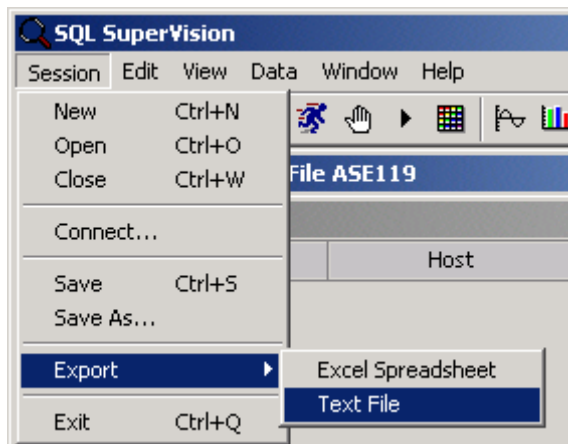


Enter a file name in the directory of your choice. The default location is "APPDATA%\SQL SuperVision\Data Files" directory.

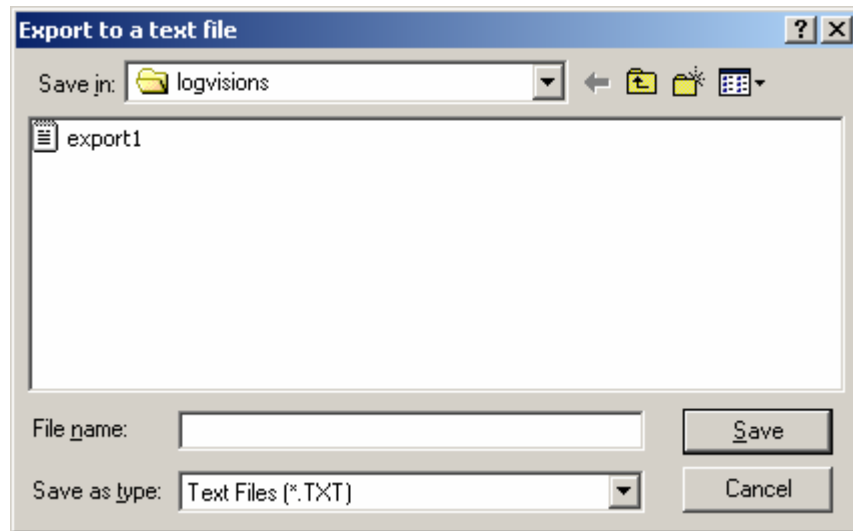


The export generates two files: < file name > (Statements).XLS, which contains the lines of the session's report and < file name > (Server).XLS, which contains the Adaptive Server's performance indicators' values.

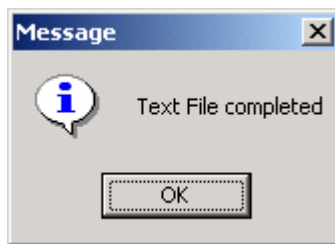
## How to export to a text file



Enter a file name in the directory of your choice. By default, the «.TXT» files containing performance data collected by SQL SuperVision, exported to text format, are stored in the %APPDATA%\SQL SuperVision\Data Files directory.



When the exporting is complete, the following message appears:



The export generates a file: < name of file>.TXT in ASCII format, which contains the lines of the session's report. The Adaptive Server's performance indicators' values are not exported to the text file.



## How to retrieve results

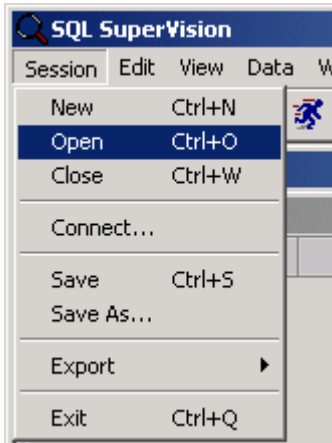
See also:

How to retrieve from a database

How to retrieve from a file

Automatic refresh

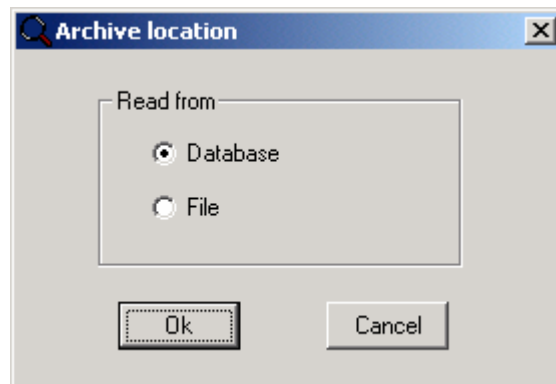
To open an archive, all you need to do is select the "Session - Open..." menu item, or click on the "Open" menu icon (see Figures below).



SQL SuperVision allows the following two archive formats: ASCII files and Adaptive Server's dedicated databases.

## How to retrieve from a database

From the following dialog box, select the required location for the results.



After selecting "Database", you connect to the archive database containing the monitoring results recorded by SQL SuperVision.

See also:

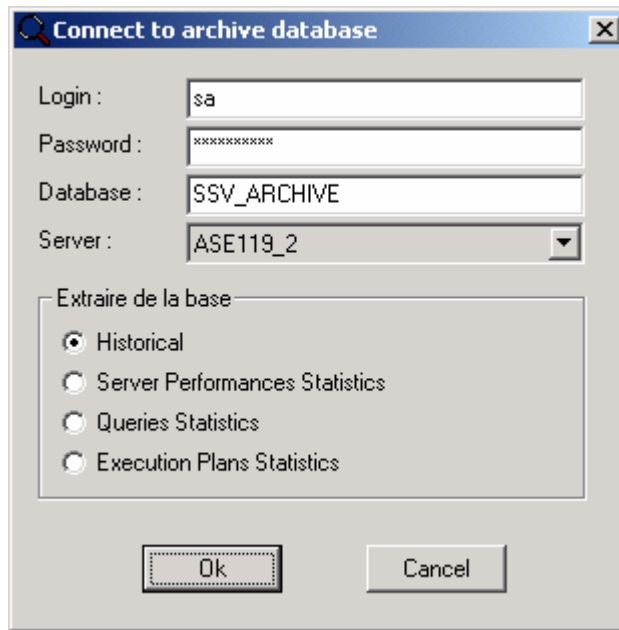
History

Performance statistics of the Adaptive Server

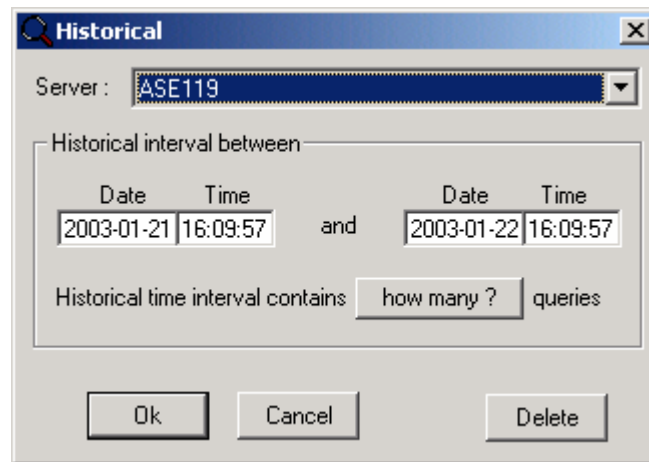
Statistics on queries

Statistics on Execution Plan

## History

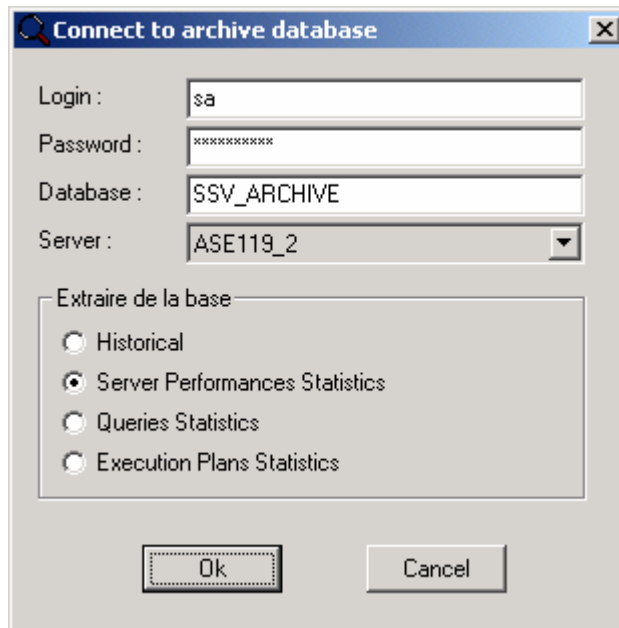


The dialog box below enables you to extract the required data.



The "Server" dropdown list box contains the various Adaptive Servers' names for the monitoring data that exists in the database. Define the time interval needed. Thanks to the "how many?" button, you can find out how many SQL Statements are present in the database according to these criteria. Check that your PC workstation's memory is not saturated with a bulky session's report. When you click on this button, the number of SQL Statements replaces the previous string ("how many?"). The "Delete" feature is explained in the paragraph on "Results deletions".

## Adaptive Server's performances' statistics



Connect to archive database

Login : sa

Password : \*\*\*\*\*

Database : SSV\_ARCHIVE

Server : ASE119\_2

Extraire de la base

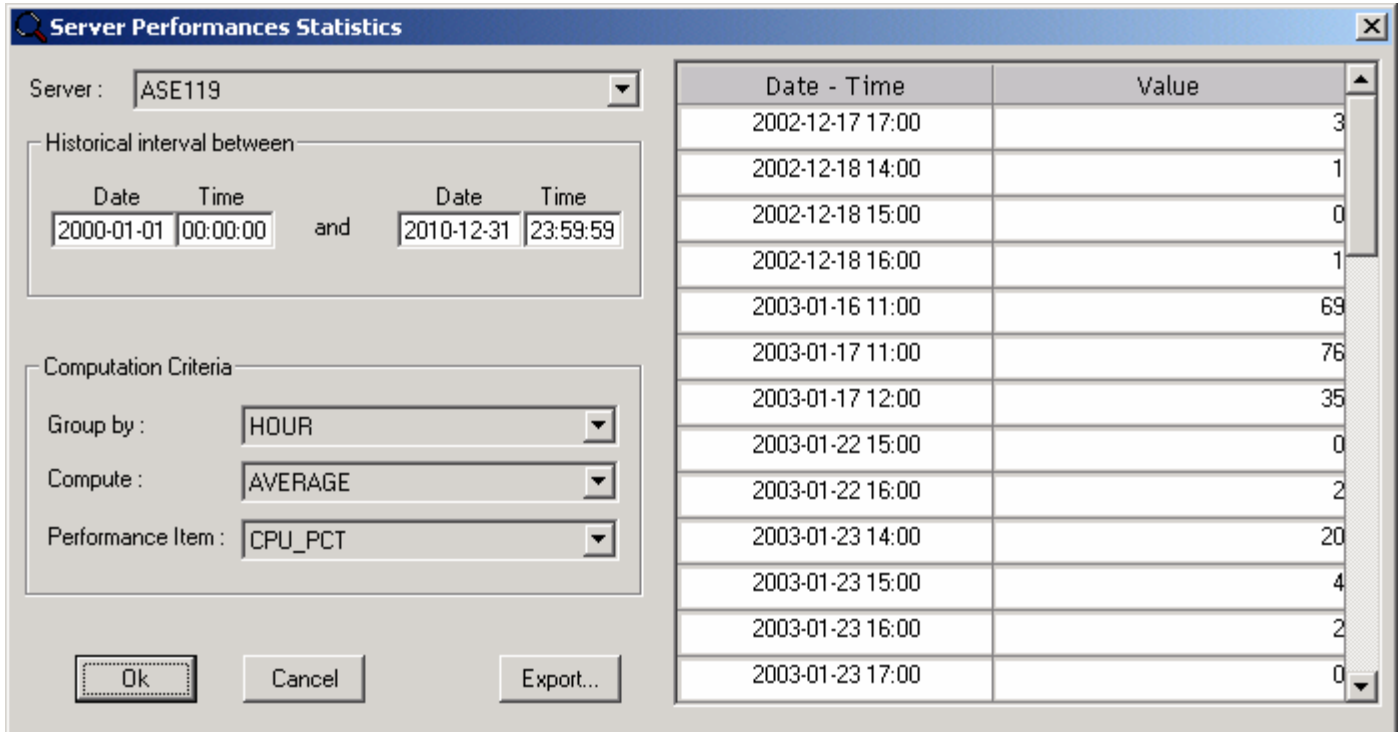
Historical

Server Performances Statistics

Queries Statistics

Execution Plans Statistics

Ok Cancel



Server Performances Statistics

Server : ASE119

Historical interval between

Date Time and Date Time

2000-01-01 00:00:00 and 2010-12-31 23:59:59

Computation Criteria

Group by : HOUR

Compute : AVERAGE

Performance Item : CPU\_PCT

| Date - Time      | Value |
|------------------|-------|
| 2002-12-17 17:00 | 3     |
| 2002-12-18 14:00 | 1     |
| 2002-12-18 15:00 | 0     |
| 2002-12-18 16:00 | 1     |
| 2003-01-16 11:00 | 69    |
| 2003-01-17 11:00 | 76    |
| 2003-01-17 12:00 | 35    |
| 2003-01-22 15:00 | 0     |
| 2003-01-22 16:00 | 2     |
| 2003-01-23 14:00 | 20    |
| 2003-01-23 15:00 | 4     |
| 2003-01-23 16:00 | 2     |
| 2003-01-23 17:00 | 0     |

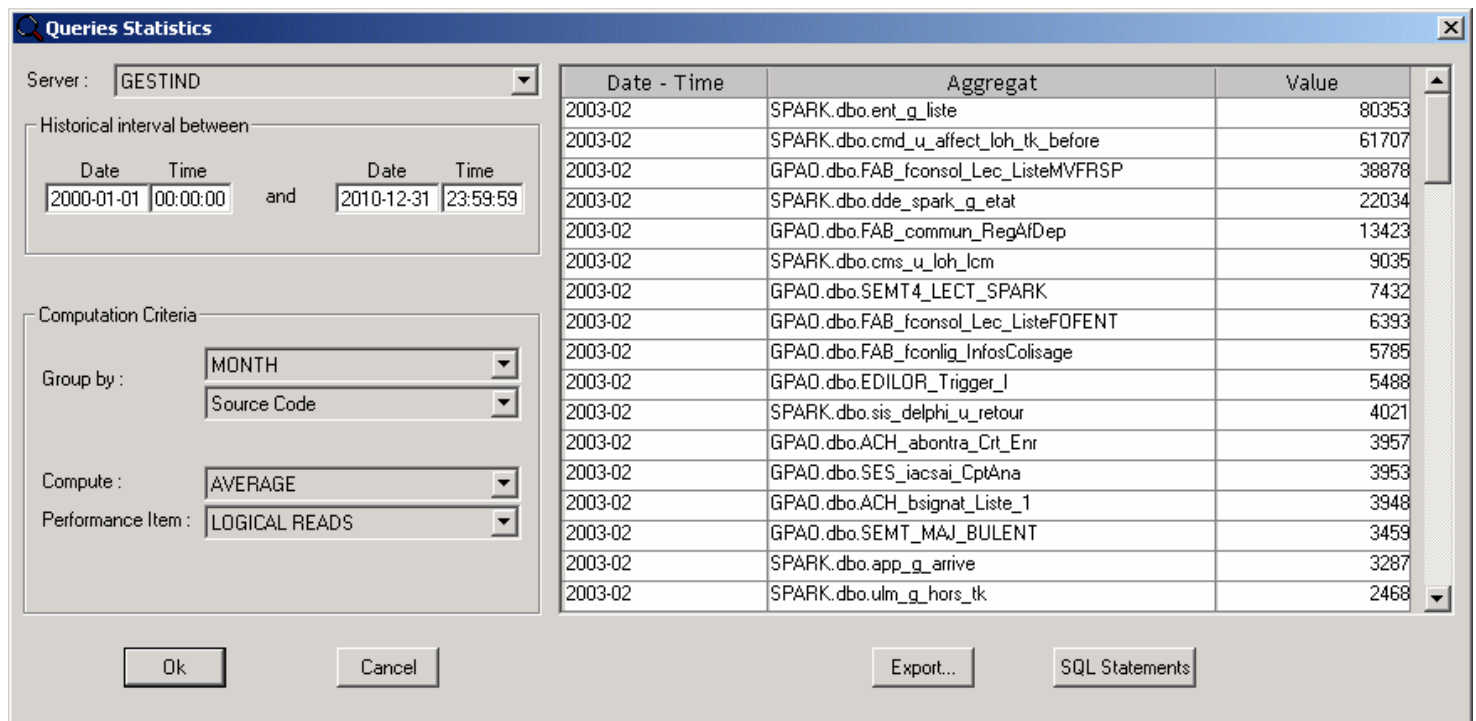
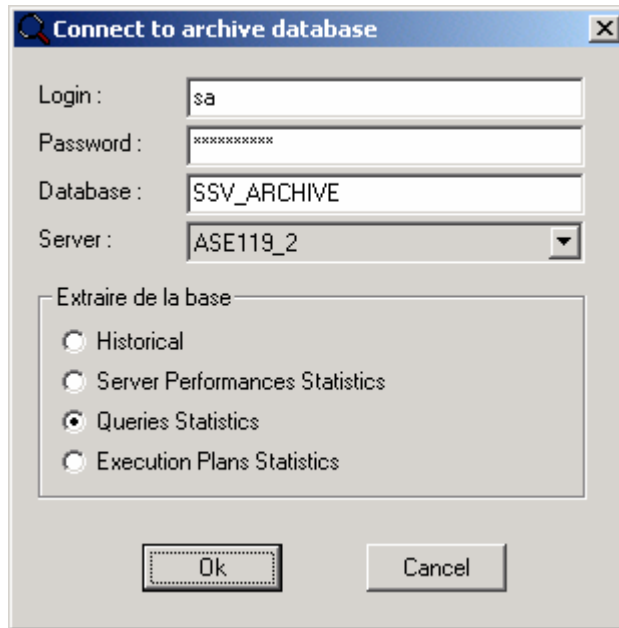
Ok Cancel Export...

This dialog box enables you to calculate the server's performance statistics. To proceed, you need to take the following steps:

- 1) Select the monitored server for which you recorded information in the archive database.
- 2) Determine an observation period that defines the volume of data for which the statistics are being calculated.
- 3) Select the aggregate's size, i.e., the time interval used as a basis for the statistical calculations. You can elect to aggregate per minute, per hour, per day, per month or per year or select "none", which corresponds to the elementary values.
- 4) Select the type of statistical calculations required (sum, average, min value or max value).
- 5) Select the performance item that the calculation applies to (logical reads, physical reads, CPU time, elapsed time).

Next, click on the "Ok" button to obtain the results in the table on the right-hand side. You can then export them to Microsoft Excel, by clicking on the "Export..." button, in order to build a graph or insert the results in a document. You can click on the column titles to sort the data.

## Statistics on queries



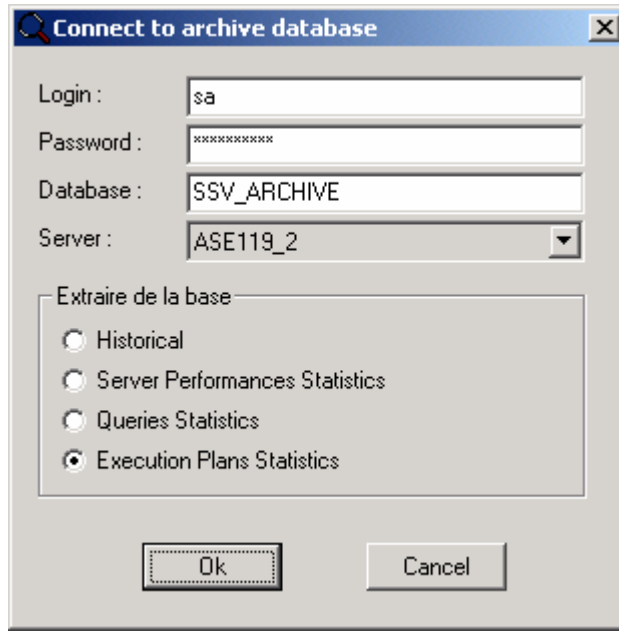
This dialog box enables you to calculate the statistics concerning the queries executions that were captured by SQL SuperVision. To proceed, you need to take the following steps:

- 1) Select the monitored server for which you recorded information in the archive database.
- 2) Determine a period defining the volume of data for which the statistics are being calculated.
- 3) The aggregate is composed of two elements: the size of the aggregate, i.e., the time interval which is used as a basis for the statistical calculations and the axis, which is based on the login, the hostname, the application name or the source code (stored procedure, trigger or dynamic SQL). You can elect to aggregate per minute, per hour, per day, per month, per year or select "none", which corresponds to the elementary values.
- 4) Select the type of statistical calculations required (sum, average, min value or max value).
- 5) Select the performance item to which the calculation applies (logical reads, physical reads, CPU time, elapsed time).

Next, click on the "Ok" button to obtain the results in the table on the right-hand side. You can then export them to Microsoft Excel, by clicking on the "Export..." button, in order to build a graph or to insert the results in a document. You can click on the column titles to sort the data.

By selecting a line from the table of results, you can list the queries from a given session's report that have been captured during the selected period and that match the criteria defined for the aggregate. To do this, you can either double click on the line required or you can select a line and click on the "SQL Statements" button.

## Statistics on Execution Plans



The image shows a dialog box titled "Connect to archive database". It contains the following fields and options:

- Login : sa
- Password : [masked with asterisks]
- Database : SSV\_ARCHIVE
- Server : ASE119\_2 (dropdown menu)
- Extrait de la base (radio buttons):
  - Historical
  - Server Performances Statistics
  - Queries Statistics
  - Execution Plans Statistics
- Buttons: Ok, Cancel

Server : FRbb

Historical interval between  
 Date Time and Date Time  
 2000-01-01 00:00:00 and 2010-12-31 23:59:59

Performance Item : LOGICAL READS

|    | Execution plans                                    | # Exec | sum(logical reads) | avg(logical reads) |
|----|----------------------------------------------------|--------|--------------------|--------------------|
| 1  | QUERY PLAN FOR STATEMENT 19 (at line 75).          | 5      | 34059620           | 6811924            |
| 2  | QUERY PLAN FOR STATEMENT 7 (at line 30).           | 2      | 23180308           | 11590154           |
| 3  | QUERY PLAN FOR STATEMENT 15 (at line 95).          | 1      | 13643816           | 13643816           |
| 4  | QUERY PLAN FOR STATEMENT 3 (at line 9).            | 11     | 10795652           | 981422             |
| 5  | QUERY PLAN FOR STATEMENT 1 (at line 1).            | 2      | 9146132            | 4573066            |
| 6  | QUERY PLAN FOR STATEMENT 5 (at line 27).           | 1      | 7020919            | 7020919            |
| 7  | QUERY PLAN FOR STATEMENT 26 (at line 88).          | 1      | 5716420            | 5716420            |
| 8  |                                                    | 1      | 5331210            | 5331210            |
| 9  | STEP 1                                             | 1      | 4161014            | 4161014            |
| 10 | The type of query is UPDATE.                       | 1      | 3084007            | 3084007            |
| 11 | The update mode is deferred_varcol.                | 1      | 1987052            | 1987052            |
| 12 | FROM TABLE                                         | 1      | 1587630            | 1587630            |
| 13 | SIDLigneEtat                                       | 1      | 1337168            | 1337168            |
| 14 | Nested iteration.                                  | 1      | 1281224            | 1281224            |
| 15 | Table Scan.                                        | 1      | 851172             | 851172             |
| 16 | Forward scan.                                      | 1      | 771165             | 771165             |
| 17 | Positioning at start of table.                     | 1      | 674686             | 674686             |
| 18 | Using I/O Size 16 Kbytes for data pages.           | 1      | 646394             | 646394             |
|    | With LRU Buffer Replacement Strategy for data page | 1      |                    |                    |

This dialog box enables you to calculate the statistics regarding the execution plans of the SQL Statements that have been captured by SQL SuperVision. This functionality is used to rationalize your optimizations. This report lists the statistics compiled by identical execution plans. For example, an UPDATE SQL statement relating to a T1 table present in a stored procedure called P1 can produce the same execution plan as a different UPDATE statement from the same T1 table but in another stored procedure called P2. The optimization effort will therefore deal with the performance issue present in the two stored procedures. If you need to improve the SQL code, this amendment needs to be implemented in both procedures. To obtain these statistics, you need to proceed as follows:

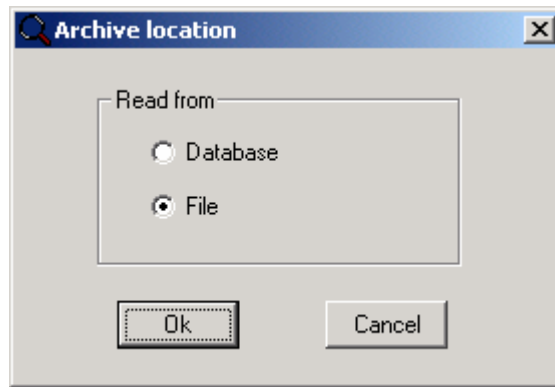
- 1) Select the monitored server for which you recorded information in the archive database.
- 2) Determine a period that defines the volume of data for which the statistics are being calculated.
- 3) Select the performance item to which the calculations apply (logical reads, physical reads, CPU time, elapsed time).

The aggregation criterion is the textual composition of each execution plan present in the archive database.

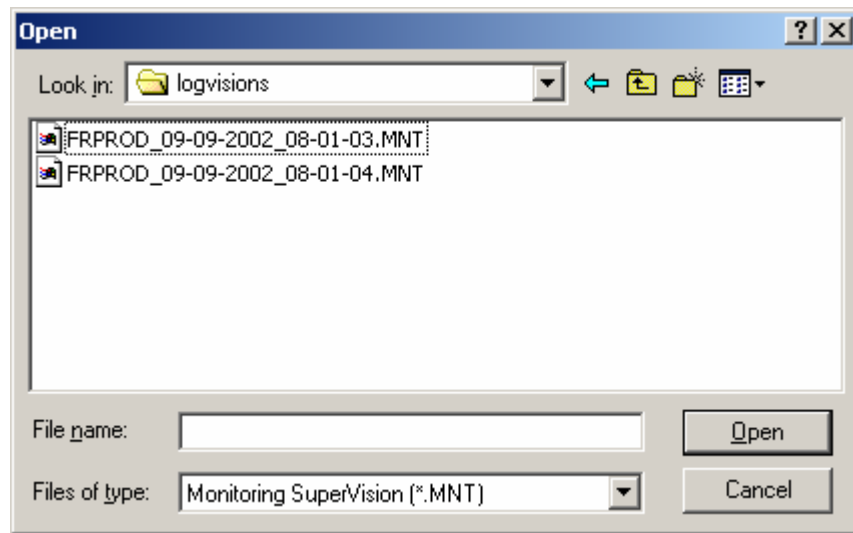
Next, click on the "Ok" button to obtain the results in the table. By selecting a line from the table of results, you can list the queries from a given session's report that have been captured during the selected period and that match the aggregate's criterion. To do this, you can either double-click on the line selected or you can select the line then click on the "SQL Statements" button.

## How to retrieve from a file

From the following dialog box, select the location required.



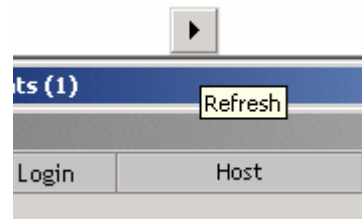
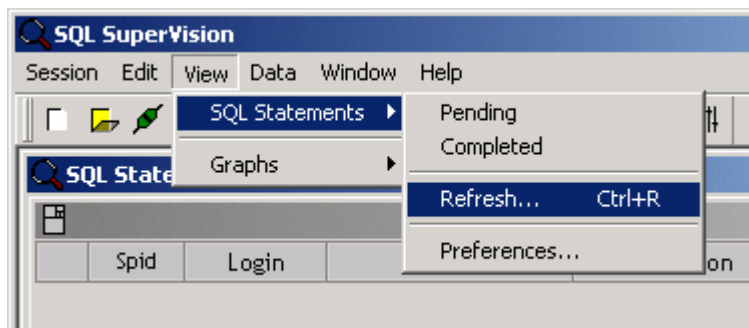
After selecting "File", define the directory and the file name containing the SQL SuperVision's monitoring results. By default, the «.MNT» files containing performance data collected by SQL SuperVision are stored in the %APPDATA%\SQL SuperVision\Data Files directory.



All SQL Statements contained in this file are loaded in the session's report.

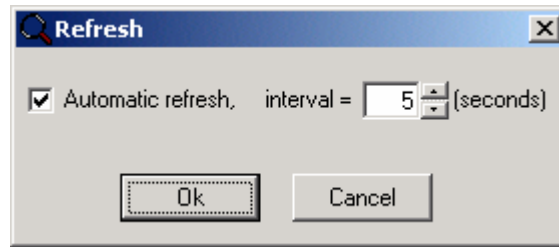
## Automatic refresh





It is possible to read a file that is currently being processed by an active monitoring session running on a different PC workstation. To perform an automatic refresh of the display, select the "View - Refresh..." menu item. Alternatively, you can click on the "Refresh" menu icon (See Figures below).



When you activate the automatic refresh function of the display, a dialog box asks you to define the required refreshing frequency. In the direct monitoring mode (SQL SuperVision is connected to the Adaptive Server), the automatic refresh function is performed with the same frequency as the monitoring. In this case, the refreshing frequency cannot be

modified. However, you can interrupt the refreshing at any time.

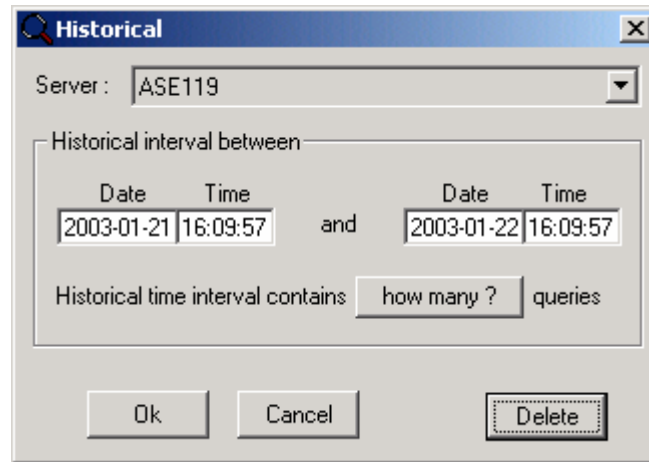


As soon as the automatic refresh function is activated, the refresh icon goes from the "read" mode  to the "pause"  mode. Starting with Release 3 of the software, the session's connection (See Paragraph on "How to start a monitoring session") and the automatic refresh function of SQL Statements' display (See Paragraph on "Contents of the session's report") run independently from each other. What this means is that you cannot view the details of the SQL Statements of the "Completed" report when the Automatic Refresh function is active (icon , and when the "Completed" report is greyed out). However, you can interrupt the automatic refresh and resume it ( icon) without losing the results of the monitoring session in progress. During the interruption, the captured SQL Statements are saved in the background and you can view the details of the SQL Statements in the reports. Former releases allowed you to view the details of the SQL Statements (in a pending state) in real time, while the Automatic Refresh function was active.



## Results deletions

From this window, you can activate the deletion of the monitoring results (SQL Statements and server's indicators' values) of a given archive database (See below: "Delete" button). The SQL Statements and the Adaptive Server's performance indicators' values associated with the server in question and the selected time period are removed.



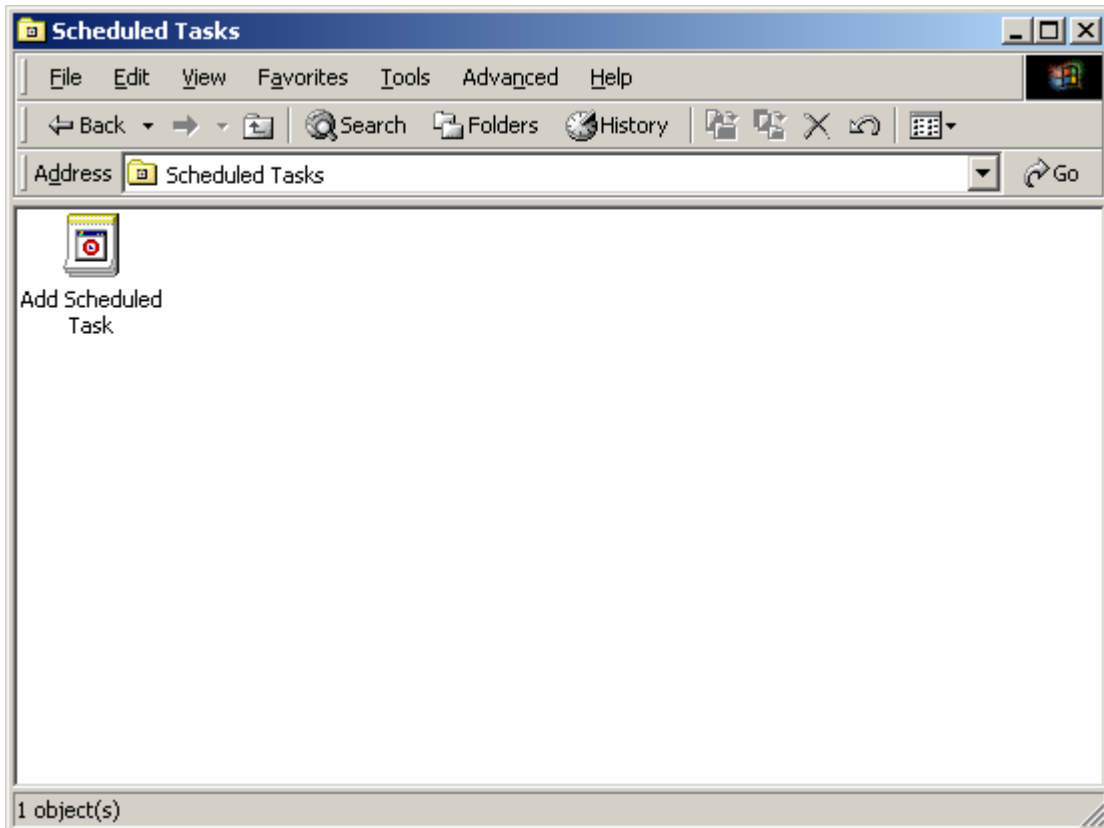
## Executing SQL SuperVision in a background task

You can launch SQL Supervision in a background task. To do so, you need to create a scheduled task. Next, use the background task's execution parameters.

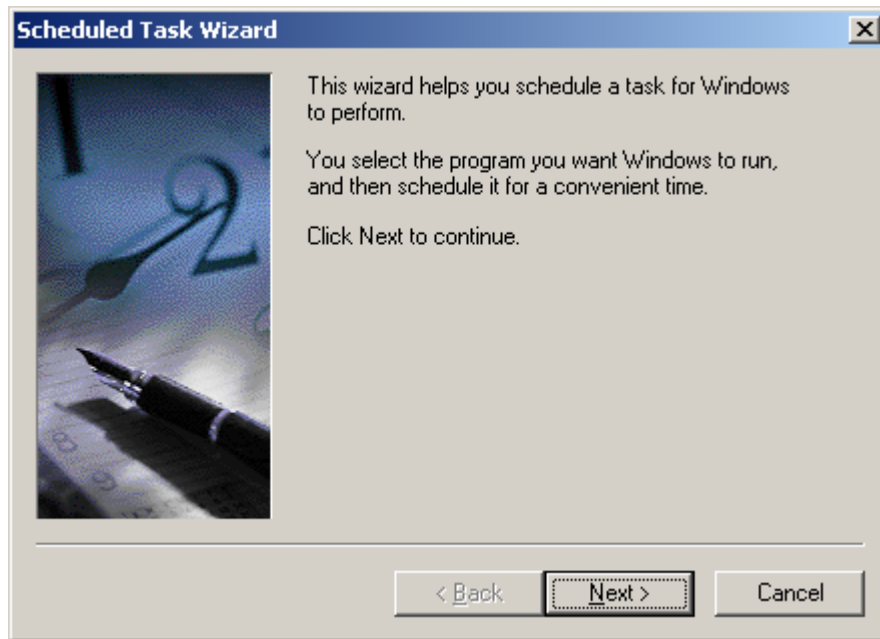
This feature is only available in the SQL Supervision's Enterprise Edition.

### Creating a scheduled task

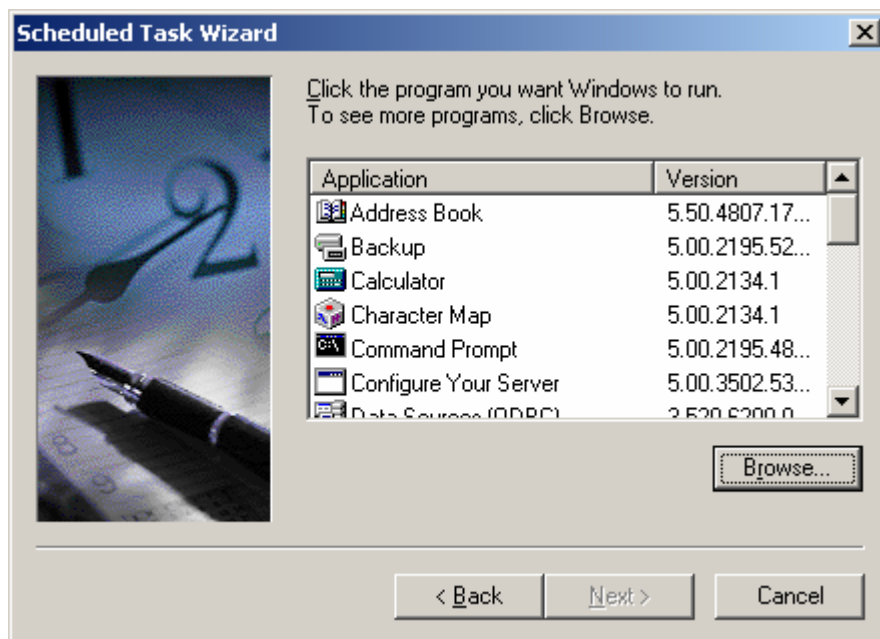
The following example details the steps to follow. From the "Start - Settings - Control panel" menu, select the "Scheduled Tasks" utility (to setup an automatic process execution in a background task of your computer). The window dedicated to the management of scheduled tasks helps you create a new task.



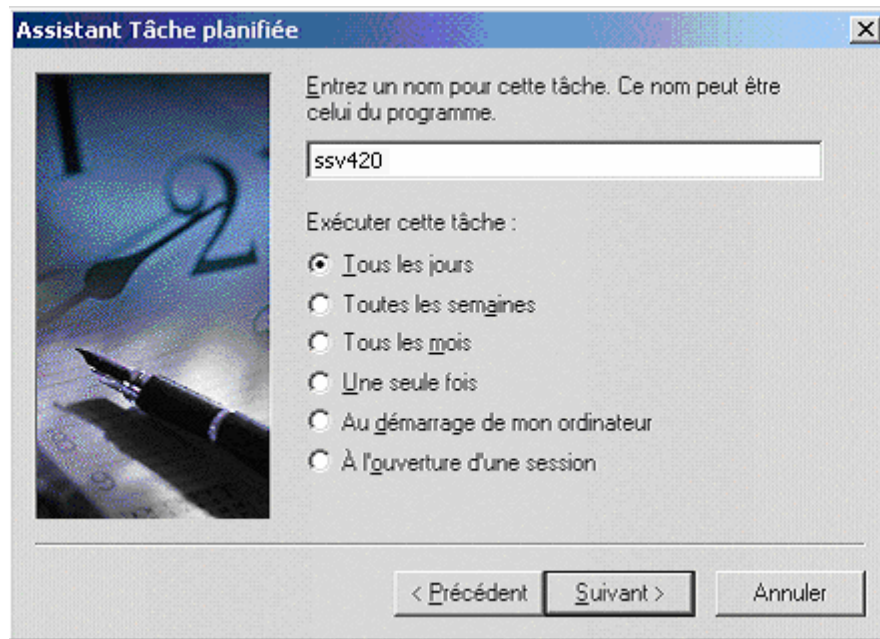
Next, click on "Add Scheduled Task".



Now, click on the "Next>" button.



Click on the "Browse..." button to look for the SQL SuperVision's EXE file (ssv420.exe), which is in the installation directory. Next, select the launch options. We recommend that you define a daily execution, especially if you intend to record the results to a file, so that it is not too bulky.



You can now define the time when you want the process to start.



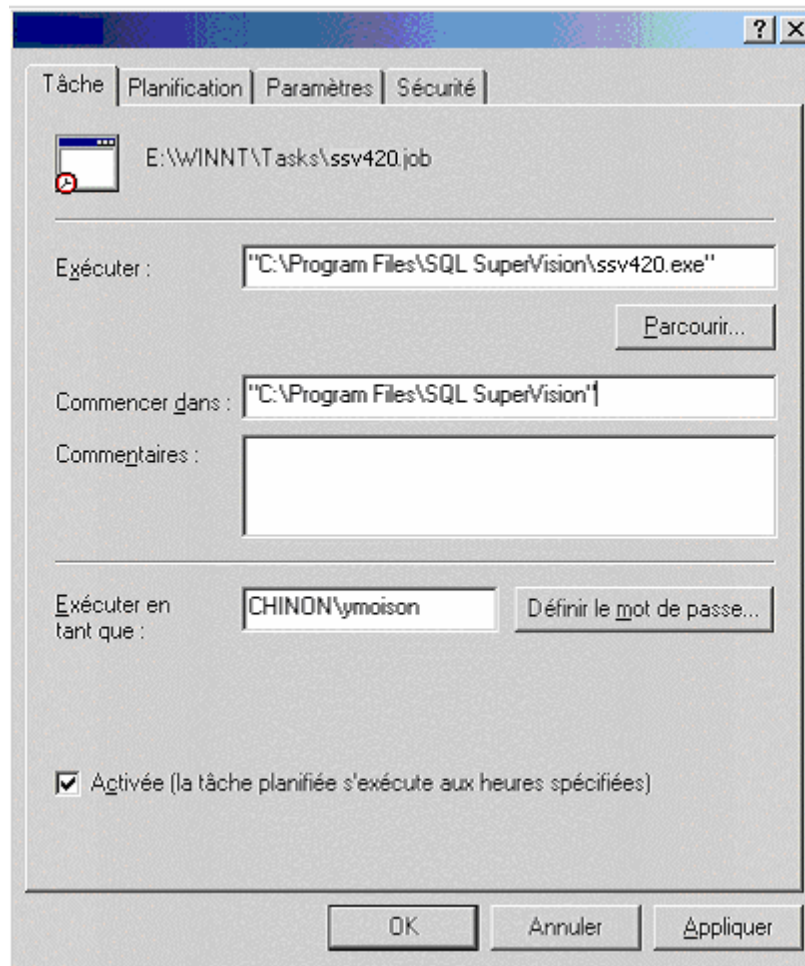
Enter the login of the background task's owner (the person who is launching the scheduled task). It is preferable to use a login that is not set to a specific person, in order to avoid any inappropriate password modification.



The scheduled task creation is almost complete.



You now need to complete the command line (in the following "Run: " window) using SQL SuperVision's background task execution's options described below. The "Start in: "edit line defines the execution's directory. The SQL SuperVision's errorlog file named « SQL SuperVision.log » is stored in the %APPDATA%\SQL SuperVision\log. It contains SQL SuperVision's execution traces and all messages related to potential incidents.



## Executing a background task

- [-D] session's duration (minutes)
- [-F] filters file
- [-L] license level
- [-M] Monitor Server's name
- [-O] archive file  
 \$\$dd\$ day\_month\_year  
 \$\$hh\$ hour-minute-second  
 example:  
 "SERVER\_\$\$dd\$\_\$hh\$.MNT --> SERVER\_05-11-2009\_15-01-03.MNT"
- [-P] password of the login connected to the Monitor Server and to the Adaptive Server
- [-S] monitored Adaptive Server's name
- [-U] login that connects to the Monitor Server and to the Adaptive Server
- [-V] application is visible (YES/NO)
- [-W] login connected to the Adaptive Server Hosting the archive database
- [-X] password of the login connected to the Adaptive Server Hosting the archive database
- [-Y] archive database
- [-Z] Adaptive Server hosting archive database
- [-I] snapshot frequency for the SQL Statements (milliseconds), default value = 1000
- [-J] snapshot frequency for the Adaptive Server's performance indicators' monitoring (milliseconds), default value = 15000
- [-K] Option used to enforce snapshot frequency under 1 second

### Example 1:

The following command creates a session whose task is to monitor the "ASE150" Adaptive Server for 10 hours. The results are recorded in a file located in the execution directory. The name of this file is built dynamically from the current date and hour at which the program is launched.

```
«C:\Program Files\SQL SuperVision\ssv420» -D600 -L5000 -MASE150_MS -OASE150_$$dd$_$hh$.MNT
```

-Ppasswordsa -SASE150 -Usa -VNO

#### Example 2:

The following command creates a session whose task is to monitor the "ASE150" Adaptive Server for 10 hours. The results are recorded in a database named "SSV\_ARCHIVE" and located on another Adaptive Server named "ASE\_RECT". SQL SuperVision connects to the latter with the login "user1". The Snapshot's frequency for the queries is 500 milliseconds, the -K option is mandatory to enable SQL SuperVision to launch a session with that snapshot frequency, i.e., faster than 1 second (see chapter on "How to start and stop monitoring").

```
«C:\Program Files\SQL SuperVision\ssv420» -D600 -L5000 -MASE150_MS -Ppasswordsa -SASE150 -Usa -VNO
-Wutil1 -Xpasswordutil1 -YSSV_ARCHIVE -ZASE_RECT -I500 -K
```

You now need to replace the "Run: " edit line with your own command line, based on the method illustrated in the above examples.

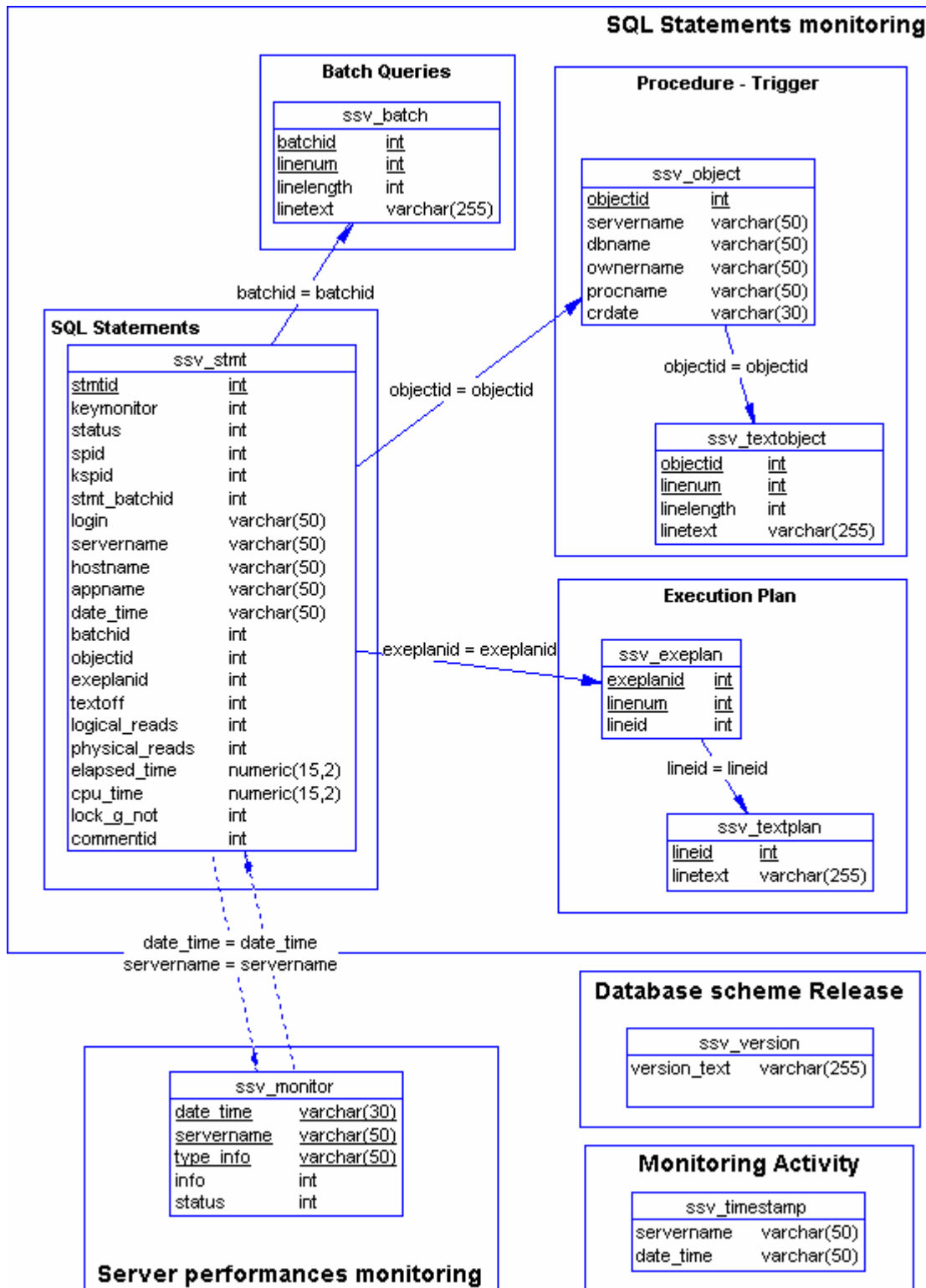
The «.MNT» files containing performance data collected by SQL SuperVision are stored in the default directory named %APPDATA%\SQL SuperVision\Data Files.

## How to shut down a session in background mode

When launched in the background mode, SQL SuperVision creates a file called <Monitored Adaptive Server>.tmp in the "%APPDATA%\SQL SuperVision\log" directory. It is controlled by SQL SuperVision at every cycle of the Scan Interval. If this file is missing, SQL SuperVision stops the session started in the background mode. To stop a SQL SuperVision session launched in the background mode, it is imperative to carry out a manual removal of this file. Please do not cancel the process from the task manager.

# SQL SuperVision's archive database model

See also:  
Tables





## Tables

See also:

ssv\_stmt

ssv\_batch

ssv\_object

ssv\_textobject

ssv\_exeplan

ssv\_textplan

ssv\_monitor

ssv\_version

ssv\_timestamp

## ssv\_stmt

Each line in this table corresponds to a line of the session's report. The following table describes the connection properties associated with each SQL Statement, as well as their performance attributes.

| Column of the table | Column of the session's report                                                                          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|---------------------|---------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Stmtid              |                                                                                                         | Primary key, single identifier in this table. This number does not appear in the session's report in the graphical interface of SQL SuperVision. The line number displayed in the session's report corresponds to a numbering method appearing in the tabular window, as in MSEXcel, to establish a link with the graph of the SQL Statements.                                                                                                    |
| Keymonitor          |                                                                                                         | SQL Supervision's license key from the PC workstation, which is connected to the Monitor Server. This information can be used to detect if several stations are connected to the same Monitor Server.                                                                                                                                                                                                                                             |
| Status              |                                                                                                         | "1" = SQL Statement is pending, "0" = SQL Statement is completed, "100" = SQL Statement is removed "logically" and consequently qualifies for a physical delete.                                                                                                                                                                                                                                                                                  |
| Spid                | Spid                                                                                                    | The SQL Server Process ID, which executes the SQL Statement.                                                                                                                                                                                                                                                                                                                                                                                      |
| Kspid               |                                                                                                         | The Kernel SQL Server Process ID, which executes the SQL Statement. It is an internal identifier of the Adaptive Server that is incremented and allotted to each new connection.                                                                                                                                                                                                                                                                  |
| Stmt_batchid        |                                                                                                         | The query's number, within a client connection. This batch's identifier is augmented each time a new query is sent by a client PC workstation within a connection.                                                                                                                                                                                                                                                                                |
| Login               | Login                                                                                                   | Login name of the connected user.                                                                                                                                                                                                                                                                                                                                                                                                                 |
| Servername          | Window 's title                                                                                         | The Server's name of the monitored Adaptive Server name (entry of the SQL.ini file or interfaces file).                                                                                                                                                                                                                                                                                                                                           |
| Hostname            | Host                                                                                                    | Name of the client PC workstation that sent the query to the ASE Server.                                                                                                                                                                                                                                                                                                                                                                          |
| Appname             | Application                                                                                             | Name of the application that sent the query to the ASE Server.                                                                                                                                                                                                                                                                                                                                                                                    |
| Date_time           | Date Time (Start)<br>Date Time (End)                                                                    | In case of pending SQL Statements, it is the date and time at which the execution started. For completed SQL statements, it is the date and time at which the execution ended.                                                                                                                                                                                                                                                                    |
| Batchid             |                                                                                                         | Foreign key used to establish a link with the "ssv_batch" table.                                                                                                                                                                                                                                                                                                                                                                                  |
| Objectid            | Procedure – Trigger<br>(Put the following tables together:<br>« ssv_object » and<br>« ssv_textobject ») | Foreign key used to establish a link with the "ssv_object" and "ssv_textobject" tables. If "objectid" = "-1", the recorded SQL Statements belong neither to a stored procedure nor to a trigger and they correspond to a query sent as part of a dynamic SQL. Please note that the reverse is untrue: a recorded SQL Statement can belong to a stored procedure executed in a trigger that is activated by a query sent as part of a dynamic SQL. |

ssv\_stmt (continued)

| Column of the table | Column of the session's report                                                           | Description                                                                                       |
|---------------------|------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------|
| Textoff             | SQL Statement (after the address inside the source code has been calculated)             | Position of the impacted SQL Statement within the stored procedure, trigger or dynamic SQL query. |
| Exeplanid           | Execution plan (Put the following tables together: « ssv_exeplan » and « ssv_textplan ») | Foreign key used to establish a link with the "ssv_exeplan table".                                |
| Logical_reads       | Logical reads                                                                            | Number of reads in the memory consumed by the SQL Statement.                                      |
| Physical_reads      | Physical reads                                                                           | Number of physical reads consumed by the SQL Statement.                                           |
| Elapsed_time        | Elapsed Time                                                                             | Elapsed time spent by the SQL Statement.                                                          |
| Cpu_time            | CPU Time                                                                                 | Computing time consumed by the SQL Statement.                                                     |
| Lock_g_not          | Locks                                                                                    | Number of locks not allocated immediately with the execution of the SQL Statement.                |
| Commentid           |                                                                                          | Foreign key which will be used in a later version.                                                |

## ssv\_batch

The lines in this table correspond to the SQL queries that are sent by the client PC workstations and that prompt the execution of the SQL Statements captured by SQL SuperVision. Please note that there is a (1,N) association, where N is the number of lines, within the SQL query, that are located between the "ssv\_stmt" table and the "ssv\_batch" table.

| Column of the table | Column of the session's report                                                                                             | Description                                                                                                                                                     |
|---------------------|----------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| batchid             |                                                                                                                            | This column belongs to the primary key (batchid, linenum). This identifier is used to establish a link with the foreign key "batchid" of the "ssv_stmt" table.  |
| linenum             |                                                                                                                            | This column belongs to the primary key (batchid, linenum). The text of the query is cut out in several lines of 255 characters.                                 |
| linelength          |                                                                                                                            | Length of the text line. This value can be higher than the result of the "datalength (linetext)" function, when the last characters of the linetext are blanks. |
| linetext            | SQL Statement<br>(after the address inside the source code has been calculated, when a query was sent in dynamic SQL mode) | The query's text line.                                                                                                                                          |

## ssv\_object

The lines in this table correspond to the stored procedures or triggers that contain the SQL Statements captured by SQL SuperVision. Please note that there is a (0,1) association between the "ssv\_stmt" table and the "ssv\_object" table. There is no match (0 cardinality) between the two tables when the impacted SQL Statement derives from a dynamic SQL.

| Column of the table | Column of the session's report                                       | Description                                                                                                                                                                                                                     |
|---------------------|----------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| objectid            |                                                                      | Primary key of the table. This identifier is used to establish a link with the foreign key "objectid" of the "ssv_stmt" and "ssv_textobject" tables.                                                                            |
| servername          |                                                                      | Name of the ESA Server containing the object.                                                                                                                                                                                   |
| dbname              | Procedure – Trigger<br>(Put together with the<br>« ssv_stmt » table) | Name of the database containing the object.                                                                                                                                                                                     |
| ownername           | Procedure – Trigger<br>(Put together with the<br>« ssv_stmt » table) | Name of the owner of the stored procedure or trigger.                                                                                                                                                                           |
| procname            | Procedure – Trigger<br>(Put together with the<br>« ssv_stmt » table) | Name of the stored procedure or trigger.                                                                                                                                                                                        |
| crdate              |                                                                      | Date on which the stored procedure or trigger was created. The versions are archived, which makes it possible to compare the performances recorded by SQL SuperVision on the various versions of a stored procedure or trigger. |

## ssv\_textobject

The lines in this table correspond to the source codes of the stored procedures or triggers that contain the SQL Statements captured by SQL SuperVision. There is a (1,N) association, where N represents the number of lines of the source code that are located between the "ssv\_stmt" and the "ssv\_object" tables.

| Column of the table | Column of the session's report                                                                                          | Description                                                                                                                                                                                                               |
|---------------------|-------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| objectid            |                                                                                                                         | This column belongs to the primary key (objectid, linenum). This identifier is used to establish a link with the foreign key "objectid" of the "ssv_stmt" table and the primary key "objectid" of the "ssv_object" table. |
| linenum             |                                                                                                                         | This column belongs to the primary key (objectid, linenum). The text of the source code (stored procedure or trigger) is cut out in several lines of 255 characters.                                                      |
| linelength          |                                                                                                                         | Length of the text line. This value can be higher than the result of the "datalength (linetext)" function, when the last characters of the linetext are blanks.                                                           |
| linetext            | SQL Statement (after the address inside the source code has been calculated, when a query was sent in dynamic SQL mode) | Text line of the source code of the stored procedures and triggers.                                                                                                                                                       |

## ssv\_exeplan

The lines in this table correspond to the execution plans of the SQL Statements captured by SQL SuperVision. There is a (1,N) association, where N represents the number of lines of the execution plan that are located between the "ssv\_stmt" and the "ssv\_exeplan" tables.

| Column of the table | Column of the session's report                                                           | Description                                                                                                                                                                     |
|---------------------|------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| exeplanid           |                                                                                          | This column belongs to the primary key of the table (exeplanid, linenum). This identifier is used to establish a link with the foreign key "exeplanid" of the "ssv_stmt" table. |
| linenum             |                                                                                          | This column belongs to the primary key (exeplanid, linenum). The execution plan is composed of several lines of 255 characters.                                                 |
| lineid              | Execution Plan<br>(Put the following tables together: « ssv_stmt » and « ssv_textplan ») | Foreign key linked to the primary key "lineid" of the "ssv_textplan" table.                                                                                                     |

## ssv\_textplan

The lines of this table correspond to the lines of the execution plans. (Example: To save room and because the literal elements of the execution plans are very redundant, the "STEP 1" line - and the same goes for the other lines - is recorded only once in the archive database.) There is a (1,1) association between the "ssv\_exeplan" table and the "ssv\_textplan" table. Note that the "ssv\_textplan" table contains the character strings of the execution plans of the "ssv\_exeplan" table.

| Column of the table | Column of the session's report                                                         | Description                                                                                                                     |
|---------------------|----------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------|
| lineid              |                                                                                        | Primary key of the table. This identifier is used to establish a link with the foreign key "lineid" of the "ssv_exeplan" table. |
| linetext            | Execution Plan<br>(Put the following tables together: « ssv_stmt » et « ssv_exeplan ») | Standard text line of the execution plan<br>Examples:<br>« STEP 1 »<br>« Table Scan »...                                        |



## ssv\_monitor

The lines in this table contain the values of the performance indicators that are collected at the Adaptive Server's level. These values can be viewed in conjunction with the graphs.

| Column of the table | Description                                                                                                                                                                                                                          |
|---------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| date_time           | Timestamp of the indicator's value.                                                                                                                                                                                                  |
| servername          | The ASE Server being measured                                                                                                                                                                                                        |
| type_info           | Indicator :<br>CPU_CPT<br>DEADLOCK_CNT<br>DENIED_LOCK<br>LOCK_CNT<br>LOG_CONTENTION_PCT<br>NET_PKTS_RCVD<br>NET_PKTS_SENT<br>PAGE_HIT_PCT<br>PAGE_IO<br>STP_HIT_PCT<br>(For further details, See Chapter on « Server's indicators ») |
| info                | Value                                                                                                                                                                                                                                |
| status              | 0 or 100 (since the value has a logical delete status, the line qualifies for a final physical deletion).                                                                                                                            |

## ssv\_version

It contains the release of the installed SQL SuperVision's archive database.

| Column of the table | Description                                                                                                                                                                                                         |
|---------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Version_text        | Character string characterizing the version.<br>Run the following in the database:<br>Select * from ssv_version<br>You obtain the following result:<br>« SQL SuperVision Database - 3.0.00 - Aug 1 2002 » or later. |

## ssv\_timestamp

It contains the status of the monitoring sessions recorded in the base files.

| Column of the table | Description                                                 |
|---------------------|-------------------------------------------------------------|
| servername          | Monitored servername.                                       |
| date_time           | Date, hour, minute, second of the last monitoring snapshot. |