

ProActive DBA™ SQL Capture™

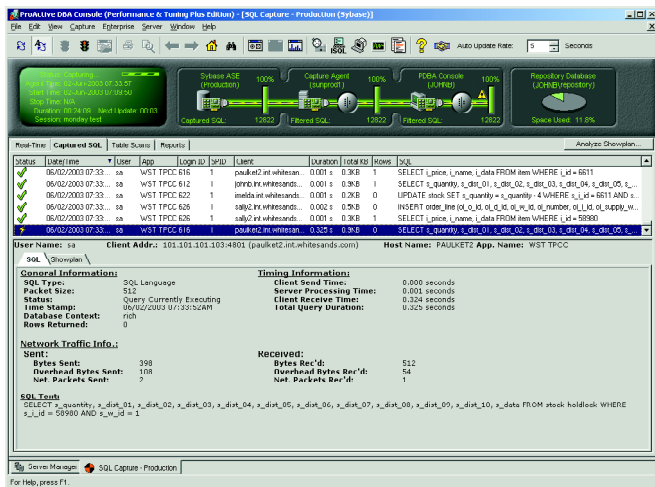
No Impact™ SQL Capture and Monitoring for Sybase ASE and Microsoft SQL Server

Introduction

SQL Capture from White Sands Technology, Inc. provides best-of-breed **No Impact** SQL monitoring for Sybase ASE and Microsoft SQL Server. It lets you:

- Capture **all** or selected SQL sent to the database server with **No Impact** on performance
- Instantly identify **poor-performing SQL**
- Monitor **real-time** database activity **non-intrusively**
- Create **alert actions** based on SQL events or performance metrics
- **Proactively** identify degrading response times and their causes, to anticipate problems before they occur
- Identify SQL in need of **tuning** efforts
- Track end-user **response times / transaction counts**
- Show SQL causing **table scans**
- **Save/load** all or selected SQL automatically to/from flat files for archiving or external processing

All this and more is available through an **easy-to-use** yet powerful graphical user interface.



Capture and view client SQL in real-time, with **No Impact** to the production server!

Why Use a Network-Based Monitor?

Traditional database performance monitors show global system performance metrics (CPU usage, disk I/Os etc.)

While these metrics can be useful for some purposes, they do not give any indication of the performance level being experienced by end-users. For example, your database server's CPU usage may be low, yet you have 100 blocked users who think the server is running very slowly!

Plus, global monitors can only sample the running SQL queries every so often, so by design they will miss a large percentage of executed SQL, and the more often they sample running processes, the more overhead these monitors will impose on your server.

SQL Capture captures performance metrics down to the individual SQL query, cursor or stored procedure call (with parameters), on **every** query. The metrics it reports correlate directly to the end-user's experience of database performance.

When you use it with ProActive DBA Database Auditor, Diagnostic Monitor and Visual Space Manager, you get the most complete performance-tuning solution available!

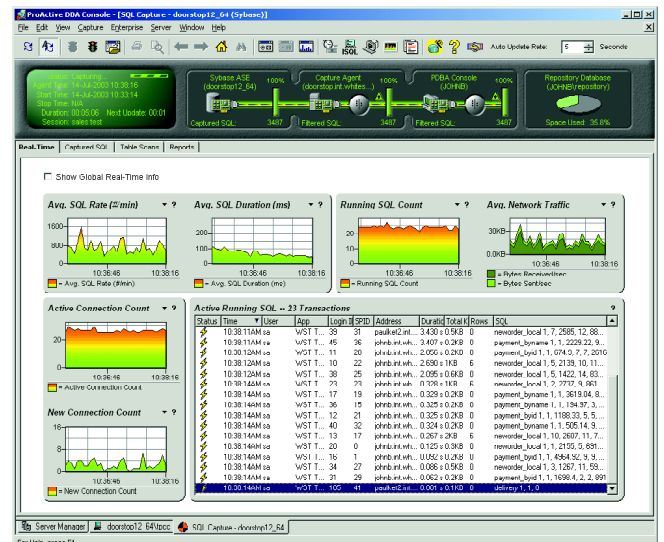
Non-Intrusive 24x7 Monitoring

SQL Capture works by "sniffing" network SQL conversations between clients and the Sybase or Microsoft database server, gathering a wide variety of metrics on the SQL executed by clients.

It can capture transactions 24x7 for full-time monitoring or auditing of SQL activity.

Capturing can occur right on the target server for convenient low-impact monitoring, or on a separate machine for true **No Impact** capturing of database activity.

You can log all or selected SQL detail to a repository database of your choice, and/or to operating system flat files. Canned or custom reports may also be generated during capturing and saved to the repository for later viewing and analysis.



Real-time view shows up-to-the-minute server activity!

ProActive DBA™ SQL Capture™

Real-Time Monitoring

ProActive DBA SQL Capture shows you real-time database activity levels, with **No Impact** to the production server.

SQL Capture's real-time performance metrics include:

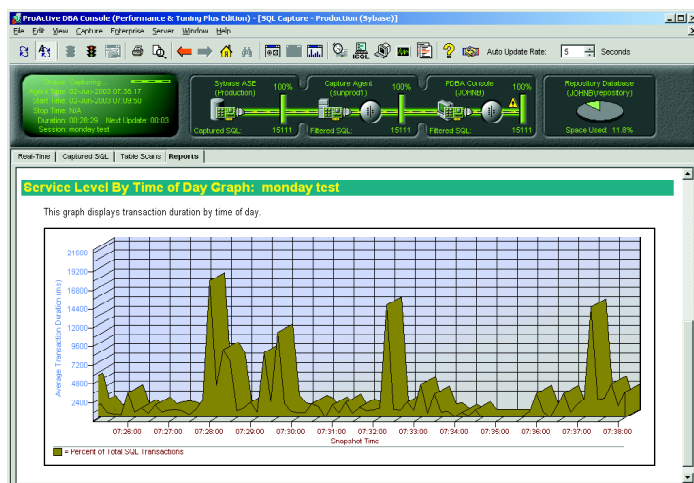
- Currently-active SQL queries
- Average SQL transactions per minute
- Average duration of SQL transactions
- Network activity levels
- Number of logged-in and active users

Graphs with zoom/pan features let you view recent historical performance metrics. Click on any active SQL query to view detailed performance data, showplan information, etc.

When you use **SQL Capture** in conjunction with ProActive DBA Diagnostic Monitor, you get the best of both worlds—SQL statement-level monitoring, as well as global high-level server and operating system monitoring, all within a single integrated framework!

Manage Performance Metrics

Powerful reporting and filtering lets you report on SQL queries by user, application, query type or other factors, for analyzing end-user service levels, transaction rates and other metrics. Produce graphical reports for management of different categories of database usage.



Track end-user response times and transaction rates historically.

Application Tuning

Diagnose application performance problems by examining SQL queries sent to the server. Identify redundant SQL, poor-performing SQL and SQL that can be turned into stored procedures.

Find SQL that returned a specific error, for **application debugging**. Especially useful when a server error (e.g. out of resources, data corruption) occurs at an unexpected time.

Customized Reporting

SQL Capture provides a wide variety of customizable reports, including:

- Longest-running SQL
- Most frequently-executed types of queries
- End-user service level histograms
- Transactions and connections graphed by time-of-day
- Top network usage SQL
- Most frequently-executed stored procedures
- And much more!

You can create unlimited custom versions of any report, based on all or selected SQL queries. All reports have the same flexible set of filtering options, which may include wildcards and include/exclude filters.

The screenshot shows the 'Console-side SQL Capture Filtering Configuration' dialog box. It has a 'Filters' tab. Under 'Date Range', there are fields for 'Start Date/Time' (04/24/2003 14:01:58) and 'Stop Date/Time' (04/24/2003 14:01:58). Under 'Client Connection', there are radio buttons for 'All Connections' and 'Selected Connections'. The 'Selected Connections' section has fields for 'User Name' (sally_m), 'SPID' (empty), and 'Login ID' (empty). There are also fields for 'Host Name', 'Application Name', and 'Client Address'. Under 'Queries', there are checkboxes for 'Show Max. of' (1), 'Net Bytes Over' (0 KB), 'Duration' (0 ms or more), 'Row Count' (0 or More), and 'Error Code(s)' (1205). There is a 'SQL Text Filter' text area and a checkbox for 'Filter Using Standardized SQL'. A 'Database Context' field contains 'sales_p2'. At the bottom, there is a checkbox for 'Filter out Internal SQL' and a 'Defaults' button. 'OK' and 'Cancel' buttons are at the bottom right.

Flexible filtering options let you create customized reports and views.

Reports may be generated dynamically “on-the-fly,” and you can also specify predefined reports which are processed and saved to the repository as SQL queries are captured.

Advanced SQL Filtering and Sorting

SQL detail, reports, exports and alerts can be filtered individually by a wide range of criteria, including SQL text, user name, application name, client IP/hostname, execution time, network usage, rows returned, database context, error code returned and other properties.

Data Warehouses (DSS / OLAP)

SQL Capture benefits data warehouse users by identifying the longest-running SQL and analyzing the query plans to find table scan activity.

OLTP / e-Business

SQL Capture benefits transaction-processing environments by tracking transaction volume versus response time. Ensure end-user service-level agreements are being met, and monitor growth of transaction quantities over time.

Summary
SQL - Most Frequently Executed - sales test
This table lists up to the top 10 most frequently executed SQL batches that were executed by clients during the capture session(s).

Report Parameters:
Number Of Top Entries In Report: 10
Sorted By: # of Executions (Descending)

Rank	# Executions	% of SQL	Accum % of SQL	Total Execution Time	Total Server Time	Total Network Time	SQL
1	1015	10.6%	10.6%	00:00:05.78	00:00:05.78	00:00:00.00	select i_price, i_name, i_data from item where
2	1013	10.6%	21.2%	00:01:21.27	00:01:21.27	00:00:00.00	insert order_line (i_id, o_id, i_id, w_id, o_
3	1013	10.6%	31.7%	00:00:10.23	00:00:08.83	00:00:01.40	select s_quantity, o_qty, o_id, o_id, o_id
4	963	9.9%	41.7%	00:00:22.60	00:00:22.60	00:00:00.00	update stock set s_quantity = s_quantity - o_
5	366	3.7%	45.4%	00:00:02.32	00:00:02.32	00:00:00.00	set rowcount 0
6	363	3.7%	49.1%	00:00:08.78	00:00:08.78	00:00:00.00	commit
7	232	2.4%	51.5%	00:00:00.50	00:00:00.50	00:00:00.00	set transaction isolation level 0
8	217	2.3%	53.8%	00:17:53.94	00:17:53.94	00:00:00.00	select w_tax from warehouse where w_id = 0
9	217	2.3%	56.0%	00:00:37.96	00:00:37.96	00:00:00.00	update district set d_next_o_id = d_next_o_i
10	217	2.3%	58.3%	00:00:00.50	00:00:00.50	00:00:00.00	begin transaction

Detail
SQL - Most Frequently Executed - sales test
Rank: 1 of 10
SQL: select i_price, i_name, i_data from item where i_id = 0
First Execution: 07/14/2003 10:54AM
Last Execution: 07/14/2003 10:56AM

View most frequently-executed SQL statements.

Identify Frequently-Executed SQL

SQL Capture includes reports that identify the **most frequently-executed** SQL patterns and stored procedures, to direct performance tuning efforts where they will have the biggest return-on-investment.

The **Most Frequently Executed SQL** report highlights small queries that get executed many times (often needlessly). Such queries can have a big performance impact, but are often overlooked as performance-tuning targets.

The **Top N SQL By Execution Time** report shows you which SQL patterns or stored procedures account for the most combined end-user response time—a great way to identify queries most in need of performance-tuning.

These reports all group similar SQL or stored procedure invocations together, so you see all executions of the same type of SQL statement (regardless of whitespace, comments, different WHERE-clause parameters, etc.)

Powerful Alerting

SQL Capture can alert you when a SQL query that meets selected criteria is executed, or when performance metrics exceed threshold values. SQL Capture offers the following types of alerts:

- Long-Running SQL
- SQL Returning Large Result Set
- Network-Intensive SQL
- Average Service Level Exceeds Threshold
- Transaction Rate Over/Under Threshold
- Other SQL Properties (SQL Text, Error Code, Etc.)

You can set a customized SQL filter for each alert, and each alert can take a different action—send an email, pager notification, kill an offending process, etc.

Table Scan Analysis

Our automated table scan analyzer shows you which tables were **table-scanned** the most and which queries failed to use an **index**. Combined with Database Auditor's column usage reports, it makes it easy to determine what indexes should be used and which columns to create indexes on, for optimal query performance!

Table Scan Analysis
This report shows the tables scanned by a specific query. It includes information about the query, the tables scanned, and the number of rows returned.

Table Name	Table Size	Index Count	Table Scans	Rows Returned
item	1,015,000	1	1	1,015,000
order_line	1,013,000	1	1	1,013,000
stock	963,000	1	1	963,000
warehouse	217,000	1	1	217,000
district	217,000	1	1	217,000

Query Information:
SQL Text: select i_price, i_name, i_data from item where i_id = 0
Query Completion: 07/14/2003 10:54AM
Query Completion: 07/14/2003 10:56AM
Query Duration: 00:00:05.78
Query Rows Returned: 1,015,000
Query Rows Returned: 1,013,000
Query Rows Returned: 963,000
Query Rows Returned: 217,000
Query Rows Returned: 217,000

Quickly and easily see which tables were table-scanned, so you can focus tuning efforts on the queries that need it most!

Application Engineering

Discover what SQL queries are being executed by legacy or third-party applications or middleware for which you may not have the source code.

As a development tool, SQL Capture is invaluable for seeing how high-level programming environments and middleware (e.g. PowerBuilder, WebObjects, Tuxedo, ODBC/JDBC) actually use the database.

ProActive DBA™ SQL Capture™

Flexible Archiving and Exporting

SQL Capture lets you export captured SQL detail in two ways: automatically during capturing, or manually from the Console.

During capturing, you can choose capture all or selected SQL detail to the repository database and/or to operating system flat files in BCP or comma-delimited text format, for integration with your own scripts or other applications.

And, you can stream SQL detail to archive files for 24x7 collection of historical data. You can set up a script to automatically dump archive files to tape or other offline storage. Then, reload one or more archives at any time to view detail, process reports or perform any other functions on the archived capture session!

Available Reports

- Longest-running SQL
- Most frequently-executed SQL patterns
- Top N SQL patterns, grouped by execution time
- Top N stored procedures executed
- End-user service level by time-of-day
- End-user service level histogram
- SQL transactions by time-of-day
- Database connections by time-of-day
- Top network usage SQL

Note: All reports can be customized and assigned different SQL filters.

All reports can be generated dynamically as “ad-hoc” reports and/or generated automatically during capturing and saved to the repository.

Major Product Features

- Network-Based SQL Capture
- **No Impact** 24x7 Monitoring
- Scheduled, Unattended Capturing
- Full Suite of Filtering Options With Wildcards and Pattern-Matching
- Create and Customize Unlimited Ad-Hoc or “Canned” Reports
- Save / Load SQL Detail to/from Operating System Flat Files and/or Database Repository

- Create Alerts on Specific SQL or Performance Events
- View Showplan and I/O Statistics
- Table Scan and Index Analysis
- Flexible Capture-Side and Display Side Filtering
- Schedule 24x7 Continuous Monitoring Via GUI and/or Command-Line
- Easiest Installation in the Industry!
- Integration With Other ProActive DBA Products (Database Auditor, Diagnostic Monitor, Visual Space Manager, Database Maintenance Manager, Disaster Recovery Toolset) in a Single Framework

Purchase Justifications

- Identify End-User Performance Problems
- Monitor End-to-End Response Times
- Ensure Compliance With End-User Service Level Agreements
- Tune SQL For Optimal Performance
- Discover SQL Executed By Middleware, Third-Party Apps, Etc.

Workstation Requirements

- Windows 2003, XP, 2000, NT 4.0, Me or 98
- Pentium 200MHz or faster CPU
- 128MB or more RAM
- 100MB or more available disk space
- 1024x768 or better display
- Sybase and/or Microsoft database client software installed

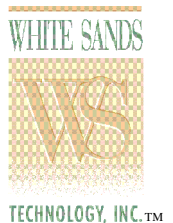
Database Server Platforms Supported

SQL Capture can capture SQL from **any** Sybase ASE platform.

Supports **all** Sybase versions 11.0.x through 12.5.x.

Supports **all** Microsoft SQL Server versions 6.0 through 2000.

Capture agents run on Windows, Sun Solaris, IBM AIX, HP-UX, Tru64 Unix and RedHat Linux.



www.whitesands.com

White Sands Technology, Inc. • 6737 Variel Ave. Suite A • Canoga Park, CA 91303 • 1-818-702-9200 • fax 1-818-702-9100 • sales@whitesands.com

Copyright © 2004 White Sands Technology, Inc. All rights reserved. ProActive DBA, SQL Capture, No Impact, Database Auditor, Diagnostic Monitor, Visual Space Manager, Database Maintenance Manager, Disaster Recovery Toolset and the White Sands logo are trademarks of White Sands Technology, Inc. Other trademarks are the properties of their respective owners.