

SQL Anywhere 17: Monitoring and Tuning



Which tools are available ?

SQL Anywhere 17: Monitoring and Tuning



Tool	Answer to	Purpose
SA Monitor	Which database servers are currently available	To see the health and availability of your SA resources (database servers, Mobilink Serves, server farms, etc,)
SA Cockpit	What your database server is doing	To see what is occurring in database server components (connected users, database properties, server messages, etc.)
SA Profiler	Why your database server is slow?	To troubleshoot and tune your applications

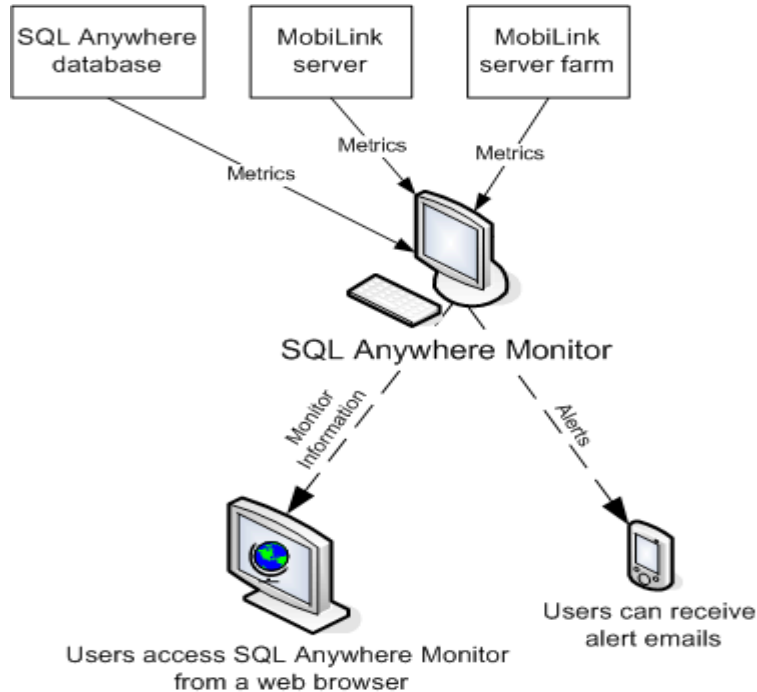
SQL Anywhere Monitor



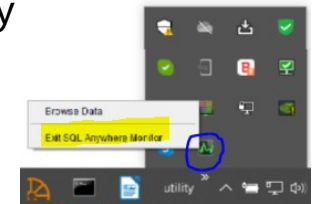
SQL Monitor provides the following features:

- Browser-based interface (based on flash)
- Constant data collection
- Email alert notification
- Monitor multiple databases servers
- Minimal performance impact

SQL Anywhere (SA) Monitor Architecture



- to start SA Monitor: from the Start Menu type SQL Anywhere → Administration Tools → SA Monitor
- to stop SA Monitor: from the System Tray right-click the Monitor icon and click *Exit SQL Anywhere Monitor*



SQL Anywhere Monitor: Resources



- You can monitor the following type of Resources

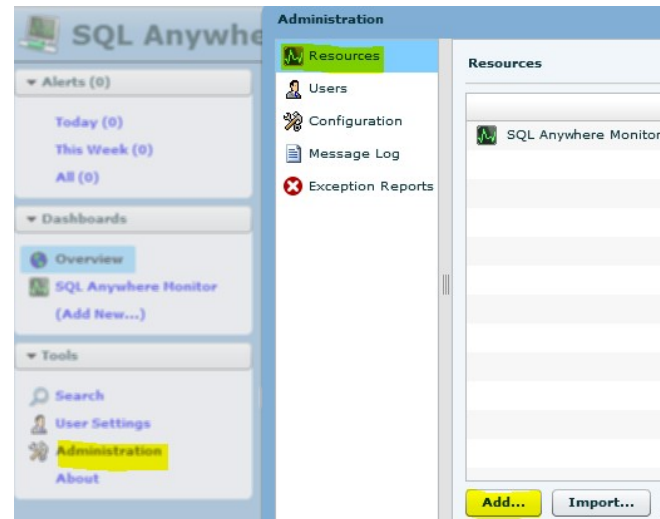
-  SQL Anywhere Server
-  MobiLink Server
-  MobiLink Server Farm
-  Web Service - web server, proxy, or host

To start monitoring a resource, i.e SQL Anywhere:

- click on Administration → Resources
- click on Add
- select Sql Anywhere Server
- follow the wizard's indications to configure the connection to the database server

Configure Resource

Name	Server	Collection Interval	Custom Metrics	Alert Thresholds	Operators	Blackouts
Give the resource a name so that users can easily identify it.						



SQL Anywhere Monitor: Dashboards and Widgets



- A dashboard is a container of one or multiples widgets
 - The Overview dashboard provides an overview of the health and availability of the resources being monitored.
 - By default, the Overview dashboard contains the Resource List widget and the Alert List widget.
- In each dashboard you can add(delete or modify) the following widget types:

Type of widget	Description
Key Performance Metrics	Displays information gathered for a resource you are monitoring.
Alerts	Displays a list of alerts for the resource
Resources	Displays a list of resources.
SQL Anywhere Connections	Displays a list of database connections.
SQL Anywhere Scale-Out Topology	Displays a topology of mirroring and scale-out systems.

SQL Anywhere Monitor: Metrics



- The monitor collects many metrics
 - Whether a resource is running
 - Whether a resource is listening and processing requests
 - Busy resources: long running queries or blocked users
 - # of synchronizations performed by MobyLink over a period of time
 - Average time taken by a backend server to process an HTTP request

SQL Anywhere Monitor: Alerts and Thresholds



- An alert is a condition about a resource that should be brought to the Monitor administrator's attention
 - Alerts are detected by the Monitor based on metrics that are collected.
 - When an alert condition is met, the alert is listed in the Alert List widget for the specified resource.
 - By default, alerts appear in the Alert List widgets and they include informations about the cause of the problem
 - In the Resource List the resource's status changes to reflect the existence and severity of the alert
- As a Monitor administrator, you can configure the thresholds that are used to trigger alerts, for example, thresholds for database resources are triggered when:
 - CPU usage exceed the given threshold, memory usage reaches X% of the max cache size
 - Free disk space per dbspace is less than X MB on the disk
 - A connection has been blocked for longer than X seconds, a query has run for longer than X secs.
 - The # of unscheduled requests reaches X, etc.

SQL Anywhere Monitor: Custom Metrics



- You can customize a metric and collect data from a resource and receive alerts for that metric
 - Alerts based on the user-specified alert criteria appear in the Alerts List widget
 - The Monitor records custom metric data by calling the value function in the SQL Anywhere resource every collection period
- Procedure in the resource database:
 - create a user-defined function
 - Grant the sa_monitor_user_EXECUTE privilege for this function
- In the Monitor:
 - In the tools pane click Administration , select the resource and click Configure
 - Click the Customer Metrics tab and then click New
 - Specify the setting for the user-defined function, click Ok , click SAVE, click Close

SQL Anywhere Monitor: Custom Metrics 2



Example : A database user wants to monitor the numbers of orders in a table and raise an alert when the total number of orders exceeds 100000.

- In the resource database, the user creates the following user-defined function:

```
CREATE FUNCTION "dba"."get_order_count"()
  RETURNS INT
  NOT DETERMINISTIC
  BEGIN
  DECLARE "currentCount" INT;
  SELECT COUNT(*) INTO currentCount FROM SalesOrders;
  RETURN "currentCount";
END;
```

- In the Monitor, the user configures a custom metric with the following settings:

Name	OrderCount
Display units	Orders
Minimum	0
Maximum	
Data type	Integer
Function owner	dba
Function name	get_order_count

SQL Anywhere Cockpit



The cockpit is a database server monitoring tool that:

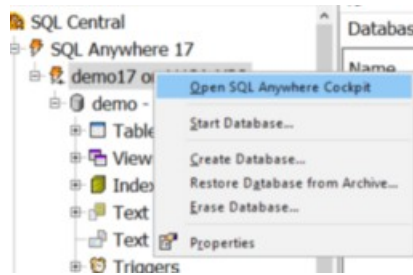
- provides an up-to-date view of the availability and capacity of a database server
- issues alerts whenever detects predefined conditions that could indicate possible problems. The alerts can be optionally notified by email.
- allows to control the threshold values which trigger the alerts
- allows to perform simple administration tasks such as starting backups and dropping connections
- uses a temporary or permanent database where to save the configuration setting. You can switch between temporary to permanent
- Is available only with version 16 or 17

SQL Anywhere Cockpit: starting, stopping and connecting

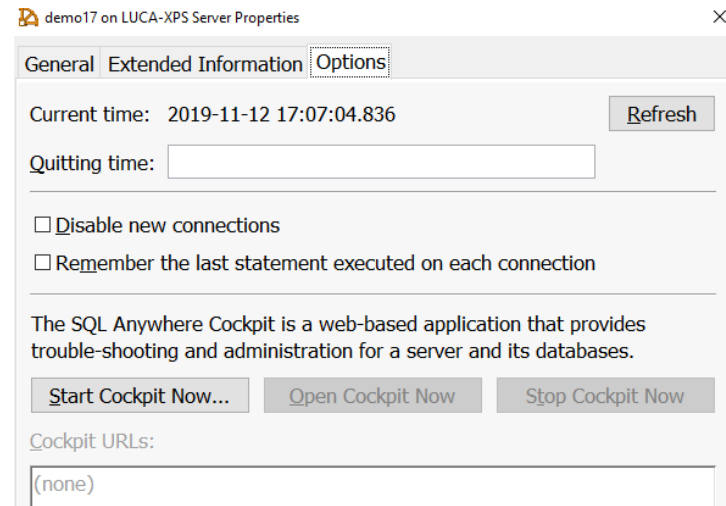


You can start and stop the Cockpit, only over a running database server from :

- Sql Central



or



- Isql by calling the system procedure **sa_server_option**

SQL Anywhere Cockpit: starting, stopping and connecting



At starting point you have the option to:

- Use an existing cockpit configuration permanent database
- Create a new cockpit configuration database
- Use a temporary database – No alerts and configuration will be saved after shutdown

A screenshot of a Windows-style dialog box titled "Start SQL Anywhere Cockpit". The dialog contains the following text: "Server: demo17 on LUCA-XPS", "The SQL Anywhere Cockpit is a web-based application that provides trouble-shooting and administration for a server and its databases.", and "The SQL Anywhere Cockpit uses its own database file to store thresholds for raising alerts and information about historical alerts. You can use the Cockpit with a temporary database file, in which case alert thresholds and historical alert information will be lost when you shut down the server or stop the Cockpit. To preserve these settings, create a permanent Cockpit database and specify this database file each time the Cockpit is started." Below the text are three radio button options: "Use this existing database file:" (with a text input field and a "Browse..." button), "Create a new database to this file:" (with a text input field and a "Browse..." button), and "Use a temporary database file". At the bottom right are "OK", "Cancel", and "Help" buttons.

Start SQL Anywhere Cockpit

Server: demo17 on LUCA-XPS

The SQL Anywhere Cockpit is a web-based application that provides trouble-shooting and administration for a server and its databases.

The SQL Anywhere Cockpit uses its own database file to store thresholds for raising alerts and information about historical alerts. You can use the Cockpit with a temporary database file, in which case alert thresholds and historical alert information will be lost when you shut down the server or stop the Cockpit. To preserve these settings, create a permanent Cockpit database and specify this database file each time the Cockpit is started.

Use this existing database file:

Create a new database to this file:

Use a temporary database file

SQL Anywhere Cockpit: starting, stopping and connecting



You connect through a browser-based interface (based on Javascript) to the running Cockpit at the url:

- automatically provided by SQL central
- retrieved by isql with the query **SELECT PROPERTY ('CockpitURL');**

The user chosen to connect to must be granted the COCKPIT_ROLE user-defined role. For convenience the COCKPIT_ROLE is populated with the following system privileges (on ver. 16 the COCKPIT_ROLE has to be built by hand).

- MONITOR system privilege
- DROP CONNECTION system privilege
- BACKUP DATABASE system privilege
- SERVER OPERATOR system privilege
- ACCESS DISK INFORMATION system privilege

SQL Anywhere Cockpit: landing page



Once connected you are redirect to (the first tab of) the tabbed panel landing page consisting of 5 tabs:

- Home
 - A summary dashboard
- Monitoring: where to see alerts and configure the relative thresholds
- Connections : where to see the connections properties and eventually kill them
- Database : where to see the database properties and perform backups
- Messages: where to see the error log messages

The screenshot shows the SQL Anywhere Cockpit landing page. At the top, there are navigation tabs: HOME, MONITORING, CONNECTIONS, DATABASES, and MESSAGES. The main dashboard area contains three summary cards:

- Alert Severity:** Shows 'OK' in green text.
- Total Connections:** Shows '2' in blue text. Below it, a breakdown: Blocked: 0, Waiting: 0, Idle: 1, Executing: 1.
- CPU Usage:** Shows '0 %' in blue text with a progress bar below it.

Below these cards is a section titled 'Last 10 Server Errors' with a table header:

Time	Message
Nessun dato	

At the bottom, there is a 'Server' section with a 'Show Details' button. The server information is as follows:

Name: demo17	Version: 17.0.10.5855
Connections: 2	Platform: Windows 10 Build 18362
Databases: 1	Host: LUCA-XPS
Cache size: 55.91 MB	Up since: 2019-11-13 11:29:46.291000

SQL Anywhere Cockpit: alert thresholds



- The predefined alerts cover all the main aspects of the server's bottleneck:
 - Cpu utilization
 - Number of connections
 - Unsheduleded requests
 - Long running operations , Connection blocking
 - Checkpoint Urgency (the time that has elapsed since the last checkpoint, as a percentage of the checkpoint time setting of the database)
 - Recovery Urgency (a heuristic to estimate the amount of time needed to restore the database if it failed right now)
 - Cache panics (the number of times/per minute the target server failed to find a cache page to allocate)
 - Cache hit percentage
 - Low disk space, Temporary file usage, database file fragments
- For each predefined alert type you can set three different threshold values to handle three severity level:
 - Low
 - Medium
 - High

SA Cockpit: configure the thresholds



Thresholds [Email Notification](#)

CPU UTILIZATION

Raise a high severity alert when CPU utilization is over % for at least seconds.

Raise a medium severity alert when CPU utilization is over % for at least seconds.

Raise a low severity alert when CPU utilization is over % for at least seconds.

NUMBER OF CONNECTIONS

Raise a high severity alert when there are or more connections to the server.

Raise a medium severity alert when there are or more connections to the server.

Raise a low severity alert when there are or more connections to the server.

UNSCHEDULED REQUESTS

Raise a high severity alert when or more requests are unscheduled.

Raise a medium severity alert when or more requests are unscheduled.

Raise a low severity alert when or more requests are unscheduled.

CHECKPOINT URGENCY

Raise a high severity alert when the checkpoint urgency is % or more.

Raise a medium severity alert when the checkpoint urgency is % or more.

Raise a low severity alert when the checkpoint urgency is % or more.

RECOVERY URGENCY

Raise a high severity alert when the recovery urgency is % or more.

Raise a medium severity alert when the recovery urgency is % or more.

Raise a low severity alert when the recovery urgency is % or more.

CACHE PANICS

Raise a high severity alert when the rate of cache panics is at least per minute for at least seconds.

Raise a medium severity alert when the rate of cache panics is at least per minute for at least seconds.

Raise a low severity alert when the rate of cache panics is at least per minute for at least seconds.

CONNECTION BLOCKING

Raise a high severity alert when a connection has been blocked for at least seconds.

Raise a medium severity alert when a connection has been blocked for at least seconds.

Raise a low severity alert when a connection has been blocked for at least seconds.

LONG RUNNING OPERATIONS

Raise a high severity alert when an operation has been running for or more seconds.

Raise a medium severity alert when an operation has been running for or more seconds.

Raise a low severity alert when an operation has been running for or more seconds.

DATABASE FILE FRAGMENTS

Raise a high severity alert when the database file contains or more fragments.

Raise a medium severity alert when the database file contains or more fragments.

Raise a low severity alert when the database file contains or more fragments.

LOW DISK SPACE

Raise a high severity alert when a drive has less than MB of free space.

Raise a medium severity alert when a drive has less than MB of free space.

Raise a low severity alert when a drive has less than MB of free space.

TEMPORARY FILE USAGE

Raise a high severity alert when temporary file usage exceeds MB for at least seconds.

Raise a medium severity alert when temporary file usage exceeds MB for at least seconds.

Raise a low severity alert when temporary file usage exceeds MB for at least seconds.

CACHE HIT PERCENTAGE

Raise a high severity alert when the cache hit percentage is or less for at least seconds.

Raise a medium severity alert when the cache hit percentage is or less for at least seconds.

Raise a low severity alert when the cache hit percentage is or less for at least seconds.

SQL Anywhere Cockpit: Connections



- You are presented with a list of connection to the database and you are able to select one to drop it

or

- To inspect all the connection attributes

SQL Anywhere Cockpit - 'demo17' | Logged in as: 'dba' to database 'demo' | Help |

HOME MONITORING **CONNECTIONS** DATABASES MESSAGES

At a Glance:

Total Connections

3

Blocked: 0 Waiting: 0
Idle: 2 Executing: 1

Drop

Conn ID	User ID	Name	Database	Blocked On	Communication Link	Last Request Time
1	DBA	SQL_Central 1	demo	0	local	2019-11-15 15:54:58.058
1405	DBA	SQL_DBC_14a31194	demo	0	local	2019-11-15 15:39:49.049
100000000		INT: StmtPerfMngtCor	demo	0	NA	

CONNECTION

DBA
ID: 1405

Overview Properties **Locking and Blocking** History

LOCKS THIS CONNECTION HOLDS

Owner	Table	Position ...	Row Locks	Table Lo...	Schema ...
GROUP0	Contacts	0	0	0	1

CONNECTIONS BLOCKED BY THIS CONNECTION

Connection Id	User Id	Connection...	Table	Blocked Since
---------------	---------	---------------	-------	---------------

SQL Anywhere Cockpit: Databases



- You are presented with a list of databases and you are able to select one and inspect all the database attributes

HOME MONITORING CONNECTIONS **DATABASES** MESSAGES

Database	Last Backup Time	Size
demo		4.14 MB

SQL Anywhere Cockpit - 'demo17'

DATABASE
demo

Overview Properties Backup

DESCRIPTION

File size:	4.14 MB
Page size:	4096
Database file:	C:\Users\Public\Documents\SQL Any
Log file:	C:\Users\Public\Documents\SQL Any
Log mirror file:	(none)
Temporary file:	C:\Users\luca\AppData\Local\Temp\sc
ID:	0
Capability ID:	3DF8C0C6E00C3000F010FEDFF7FE
Mirror role:	(not mirrored)

CONNECTIONS

Number of connections:	3
Connections disabled:	Off

SQL Anywhere Cockpit: Messages



- Very useful especially when the engine is running on a server and you don't have any access to it

HOME MONITORING CONNECTIONS DATABASES **MESSAGES**

Save

	Time	Se	Text	Category	Database	
	2019-11-15 16:01:24.024		Finished checkpoint of "demo"	Checkpoint	demo	^
	2019-11-15 16:01:24.024		Starting checkpoint of "demo"	Checkpoint	demo	
	2019-11-15 15:41:24.024		Finished checkpoint of "demo"	Checkpoint	demo	
	2019-11-15 15:41:24.024		Starting checkpoint of "demo"	Checkpoint	demo	
	2019-11-15 15:21:23.023		Finished checkpoint of "demo"	Checkpoint	demo	
	2019-11-15 15:21:23.023		Starting checkpoint of "demo"	Checkpoint	demo	
	2019-11-15 15:02:24.024		SQL Anywhere Cockpit URL: h	Message	SQLACockpit	
	2019-11-15 15:02:24.024		SQL Anywhere Cockpit URL: h	Message	SQLACockpit	

SQL Anywhere Profiler



A tool that logs the activities that occur in your database in real time and analyzes the information for performance issues such as:

- Deadlocks and blocked connections
- Long-running and expensive queries, as well as repeatedly run statements
- Expensive hidden procedures, for example, triggers, events, and nested stored procedure calls
- Potential problem areas within the body of a procedure

SQL Anywhere Profiler: Comprehensive & targeted profiling



Choose between two profiling options depending on the analysis granularity of your system:

- Comprehensive profiling:
 - it collects all activity that occurs in your database, useful when you are investigating a performance problem that has a number of potential causes
 - the performance of the database being monitored is affected
 - it collects the most information about your workload, if you need in production time use it for small amount of time
- Targeted profiling :
 - best choice when you know the characteristics of the SQL statements that are likely to affect performance
 - because it collects less information, performance is minimally affected
 - it profiles stored SQL objects , more suitable in a production environment

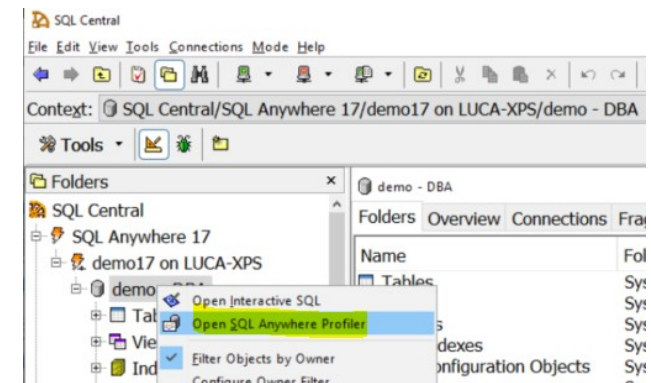
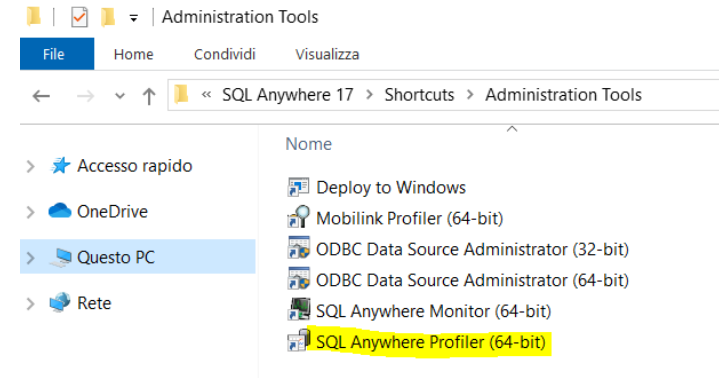
SQL Anywhere Profiler: Starting a profiling session



You can start a profile session either from:

- the command line:

- or SQL Central:



SQL Anywhere Profiler: Starting a profiling session



Whatever you choose you are presented with the profiler option panel:

Profiling Options

Operations | Disk Space | User Defined Events

What kind of profiling do you want to do?

Comprehensive
Record all database activity. This includes information about connections, statements, events, console messages, web server messages, and internal server operations.

Targeted
Record information for selected statements only.

Describe the statements you want to profile:

Statement is active for at least

Statement blocks for at least

Other criteria:

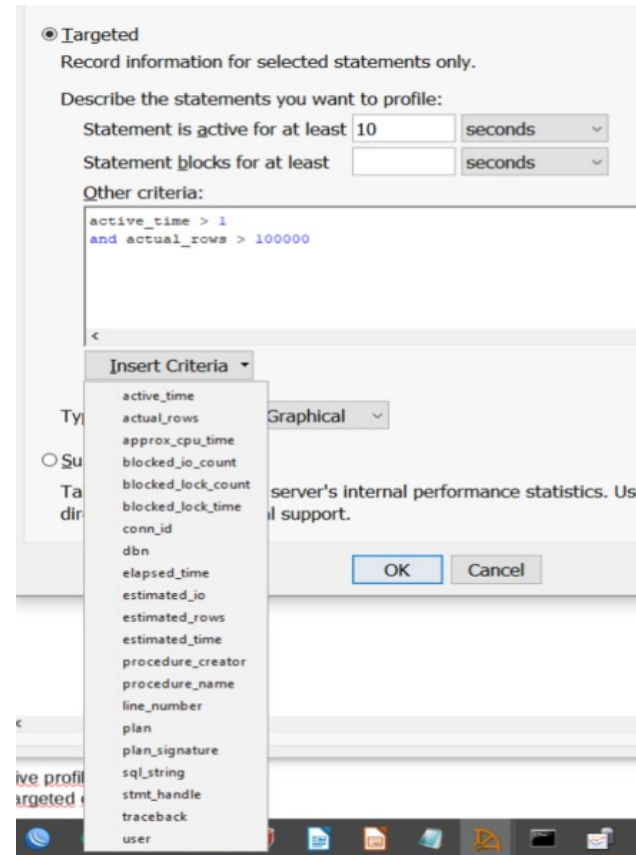
Type of plan to collect:

Support
Take a snapshot of your server's internal performance statistics. Use this option when directed by SAP technical support.

SQL Anywhere Profiler: Starting a profiling session



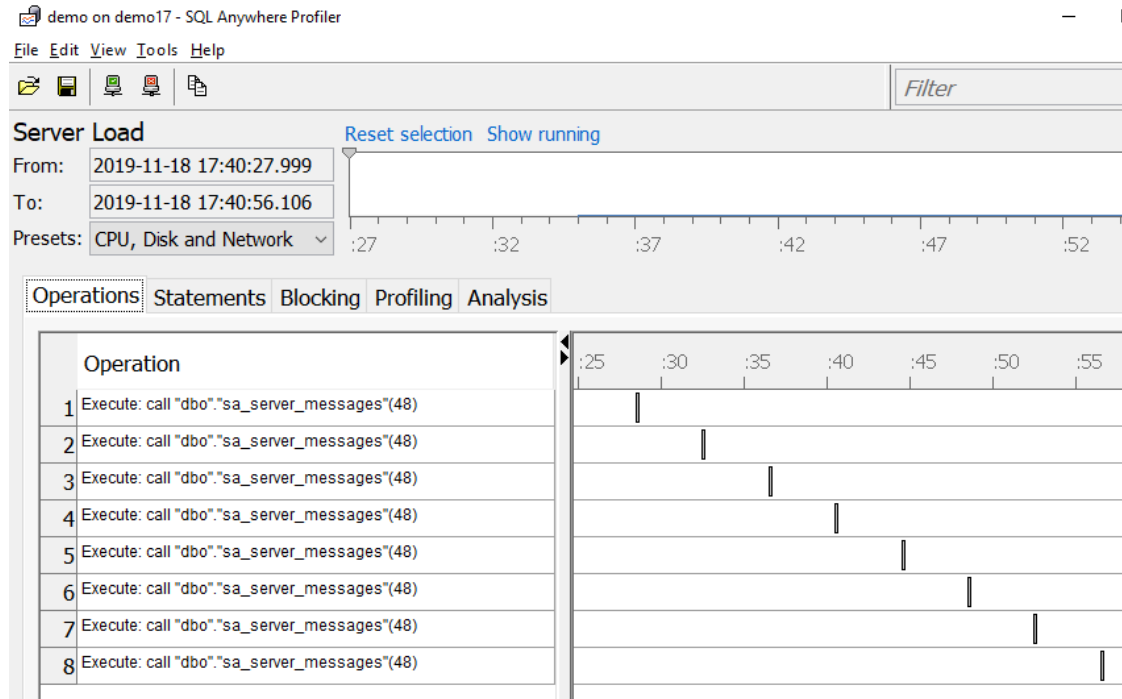
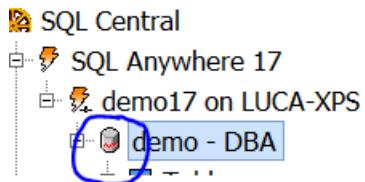
- The comprehensive profile option has no parameters
- the targeted profile option has many:



SQL Anywhere Profiler: Starting a profiling session



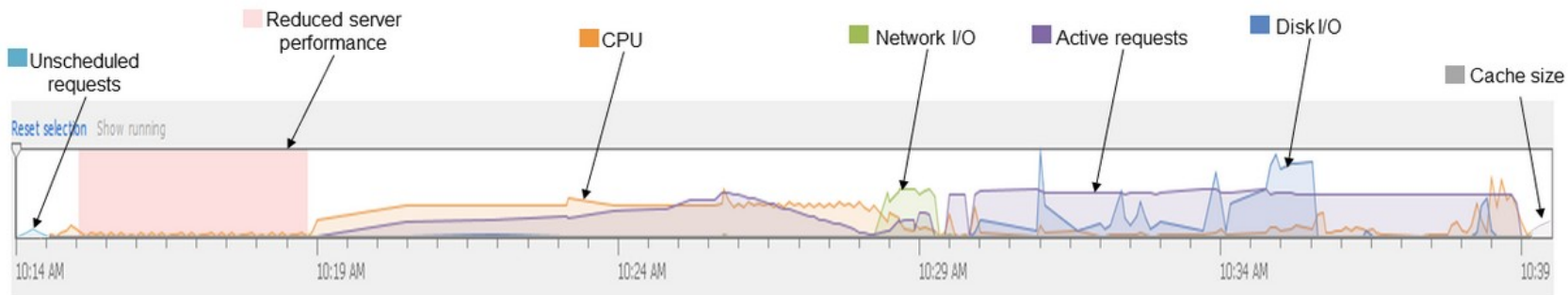
- Once a profiling type has been selected and the OK button is clicked the acquisition process starts
- The red line indicates the db is under profiling :



SQL Anywhere Profiler: the Server Load panel




- It tracks the percentage of CPU, disk and network I/O usage, number of active or unscheduled requests and the cache size



SQL Anywhere Profiler: the Operations , Statements and Blocking tabs



- You can analyze the single operation(aka the full statement) or the grouped statements with the underlying operations (aka the statement profile)
- By clicking on a single operation you can access to the following informations 
- You can check the blocking tab for issues related to blocking, blocked or deadlocked connections

Statement Profile				
	Count	Avg Exec	Max Exec	Avg Blocked
1 call "dbo"."sa_server_messages"	849	0 ms	140 ms	0 ms
2 select "COUNT()" from "dbo"."EXCLUDEOBJECT"	1	0 ms	0 ms	0 ms

Operations for Selected Profile				
	Full Statement	Execution Time	Blocked	Started
1	call "dbo"."sa_server_messages"(37)	140 ms	0 ms	Today 16:39:35
2	call "dbo"."sa_server_messages"(37)	16 ms	0 ms	Today 16:36:47

Execute Statement Properties

General	SQL	Plan	Stored SQL	Blocking	Connection
Operation type: Execute Statement					
Date: Today 15:21:06					
Execution time: 0 ms					
Connection: 1 (DBA)					
Statement timing: The statement was too short to meas...					
Statement handle: 327884					
Cursor handle: 131281					

SQL Anywhere Profiler: the Operations tab example Index analyzer



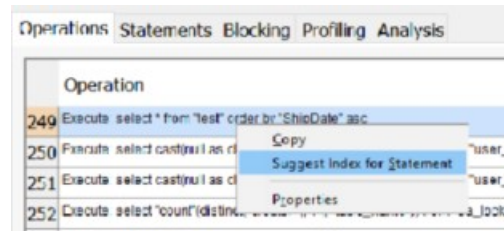
Is not so easy to see at work the **Index Consultant**. Try this:

- Create a test table by taking data from an existing table
- Populate the test table with more and more rows
- Run a query with an order by clause
- On the *Operations* tab select the query executed and right click the menu

```
SELECT * into test FROM  
"GROUPO"."SalesOrderItems"
```

```
( Execute for 10 times:)  
insert into test select * FROM test
```

```
SELECT * FROM test  
order by ShipDate
```



SQL Anywhere Profiler: the Profiling tab



- The top panel shows the SQL objects that have executed since profiling started.



- By clicking on a row of the top panel specific statistics of the selected SQL object appear in the lower panel
- If the SQL object is executed multiple times the table also shows the time differences between the baseline and any subsequent executions
- BY default the first execution of a stored SQL object is considered as the baseline until you specify a different execution

The screenshot shows the 'Profiling' tab in the SQL Anywhere Profiler. The interface has a top navigation bar with tabs for 'Operations', 'Statements', 'Blocking', 'Profiling', and 'Analysis'. Below this is a table with columns: 'Name', 'Owner', 'Count', 'Time', 'Type', 'Trigger Table', and 'Foreign Table'. The table is currently empty. Below this table is a horizontal separator, followed by another table with columns: 'Count', 'Time', '%', 'Line', and 'Source'. This second table is also empty.

SQL Anywhere Profiler: Comparing the results of different executions of stored procedures



Example (taken from a of SQL profiler tutorial on the SA User Guide)

- Create the following stored procedure:



```
CREATE OR REPLACE PROCEDURE LONGPROC ()
BEGIN
DECLARE LOCAL TEMPORARY TABLE RET (pkey INT, fkey INT);
FOR lp AS CURSOR FOR SELECT EmployeeID AS emp_id FROM GROUPO.Employees
DO
    INSERT INTO ret
    SELECT emp_id, so."ID"
    FROM GROUPO.SalesOrders so
    WHERE so.SalesRepresentative = emp_id;
END FOR;
SELECT *
FROM ret;
END;
```

- Execute the stored procedure by executing the statement
`CALL DBA.longproc ();`

- Set it as baseline to compare the results

- Modify the stored procedure with a smarter query



```
CREATE OR REPLACE PROCEDURE LONGPROC ()
BEGIN
SELECT e.EmployeeID, so."ID" FROM GROUPO.Employees e, GROUPO.SalesOrders so
WHERE so.SalesRepresentative = e.EmployeeID;
END;
```

- Execute the changed stored procedure by executing the statement
`CALL DBA.longproc ();`

SQL Anywhere Profiler: the Analysis tab



- **Workload Summary**
Useful to know if hardware resources are a limiting factor for performance. It is the starting point of the analysis.
- **Connections**
aggregate operation and statement counts for each connection with indication of the statement with longest execution time
- **Users**
aggregate operation and statement counts for each user with indication of the statement with longest execution time
- **Server**
A static description of the main attributes of the server environment:
 - server machine , Sql Anywhere engine and the profiled database
- **Statement Performance Summary** : see the next slide

SQL Anywhere Profiler: Statement Performance Summary



The *statement performance summary* feature uses the **sp_top_k_statements** and **sp_find_top_statements** system procedures (together with GTSYSPERFCACHESTMT and GTSYSPERFCACHEPLAN system views) to report the statement/plan combinations that take the longest time to run. Use to answer questions like:

- Is this statement running slower today than it was before?
- Is the amount of data being returned or modified by the statement the same today as it was yesterday?
- Has the execution plan for the statement changed?
- Has one execution plan been used more yesterday than today?
- Is the maximum runtime for the statement much higher than the average runtime?
- Is the maximum/average runtime for the statement for one plan very different from the maximum/average runtime for the other statement? If so, do invocations with one plan process or return more rows than those with another plan?

SQL Anywhere Profiler: Statement Performance Summary



For example statement #3 was executed with two different plans

[Workload Summary](#) | [Connections](#) | [Users](#) | [Server](#) | [Statement Performance Summary](#)

Statements Data is 5 min 7 sec old | [Refres](#)

Text	Total Time	Executed	Plans
1 select IGTSYSPERFCACHESTMT.stmt_hash,IGTSYSPERFCACHESTMT.stmt_text,IGTSYSPERFCACHESTMT.updated from SYS.IGTSYSPERFCACHESTMT	14 ms	4	1
2 select #SQLAProfiler_profiling.object_type,#SQLAProfiler_profiling.object_name,#SQLAProfiler_profiling.owner_name,#SQLAProfiler_profiling.line_num,#SQLAProfiler_profiling	48 ms	63	1
3 select cast(null as char(128)) as PROCEDURE_CAT,su.user_name as PROCEDURE_SCHEM, b.proc_name as PROCEDURE_NAME,cast((select count() from SYS.ISYSP	30 ms	4	2
4 select cast(null as char(128)) as TABLE_CAT,su.user_name as TABLE_SCHEM, tab.table_name as TABLE_NAME, col.column_name as COLUMN_NAME,(select cast(coale	46 ms	6	1

Performance Summary for Statement #3

Statement Text

```
select cast (null as char(128)) as PROCEDURE_CAT,su.user_name as PROCEDURE_SCHEM,
b.proc_name as PROCEDURE_NAME,cast ((select count ()
from SYS.ISYSPROCPARM as pp where pp.parm_mode_in = 'Y' and pp.proc_id = b.proc_id) as integer) as NUM_INPUT_PARAMS,cast ((select count ()
from SYS.ISYSPROCPARM as pp where pp.parm_mode_out = 'Y' and pp.parm_type <> 1 and pp.proc_id = b.proc_id) as integer) as NUM_OUTPUT_PARAMS,cast ((select count ()
from SYS.ISYSPROCPARM as pp where (pp.parm_type = 1 or
```

Plan #1 (Slowest) [Show this Plan](#)

Execution count: 1 Plan was last used: Today 16:24:11
Total rows returned: 530

Total execution time: 30 ms Total blocked time: 9 ms
Average execution time: 30 ms Average blocked time: 9 ms
Maximum execution time: 30 ms (Today 16:24:11) Maximum blocked time: 9 ms

Plan #2 (Fastest) [Show this Plan](#)

Execution count: 3 Plan was last used: Today 16:02:19

SQL Anywhere Profiler: Documentation and Tutorials



Online you are able to find more documentation and some tutorials available on these topics:

The screenshot shows the SAP DocCommentXchange interface. At the top left is the SAP logo. To its right is the text 'DocCommentXchange'. Below this is a navigation bar with buttons for 'Contents', 'Index', 'Search', and 'Comments', followed by a 'Scope: all' dropdown menu. The main area displays a tree view of documentation topics. The 'SQL Anywhere Profiler' section is expanded, showing several sub-topics. The 'Tutorial: Profiling procedures (Profiler)' item is highlighted with a light blue background.

- Database servers
 - SQL Anywhere database server executable (dbsrv17, dbeng17)
- Database configuration
- Database maintenance
- Database administration tools and utilities
- Performance improvements, diagnostics, and monitoring
 - SQL Anywhere Monitor, Cockpit, and Profiler comparison
- Monitoring
- Performance
- Diagnostics
 - SQL Anywhere Profiler
 - Running a comprehensive profiling session (Profiler)
 - Running a targeted profiling session (Profiler)
 - Criteria for specifying the SQL statements to collect during a targeted profiling session (Profiler)
 - Comparing the results of different executions of stored procedures, functions, events, and triggers (Profiler)
 - Sending SAP Support a snapshot of your database server's diagnostics (Profiler)
 - Troubleshooting: Detect when hardware resources affect performance by using the Workload Summary and Server Load graph (Profiler)
 - Troubleshooting: Application logic problems (Profiler)
 - Performance profiling tutorials (Profiler)
 - Tutorial: Collecting profiling information (Profiler)
 - Tutorial: Diagnosing blocked connections and deadlocks (Profiler)
 - Tutorial: Profiling procedures (Profiler)
 - Other diagnostic tools and techniques

SQL Anywhere Profiler: Enjoy your profiling sessions



- This presentation will be available to all the participants
- For further information you can contact me by email
luca.casavola@softpi.com