



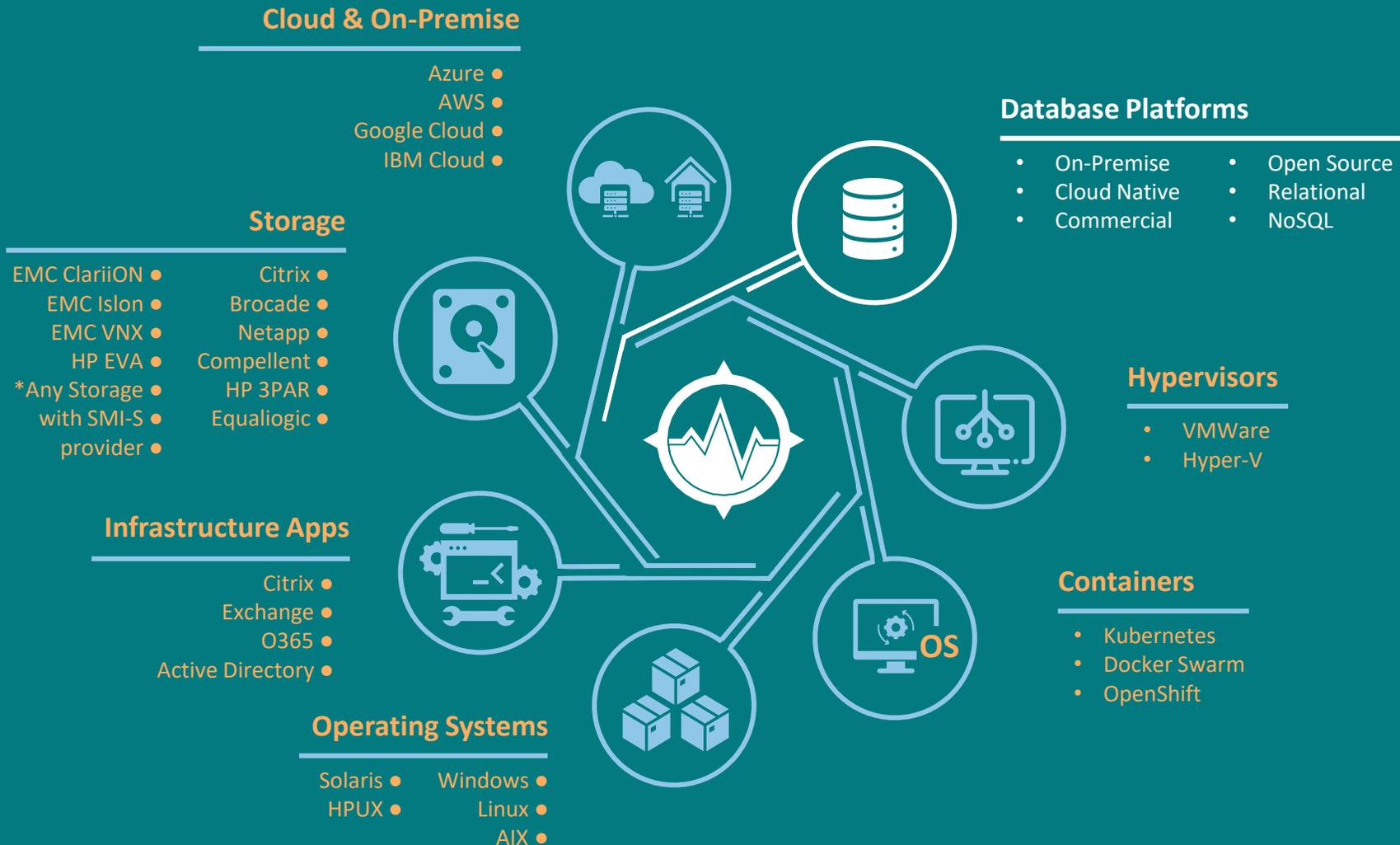
Quest[®]

Where Next Meets Now.

Foglight: Monitoring and Optimization

Tim Fritz, Foglight Product Manager, Quest Software: tim.fritz@quest.com

Multiple Platforms, Maximum Insights



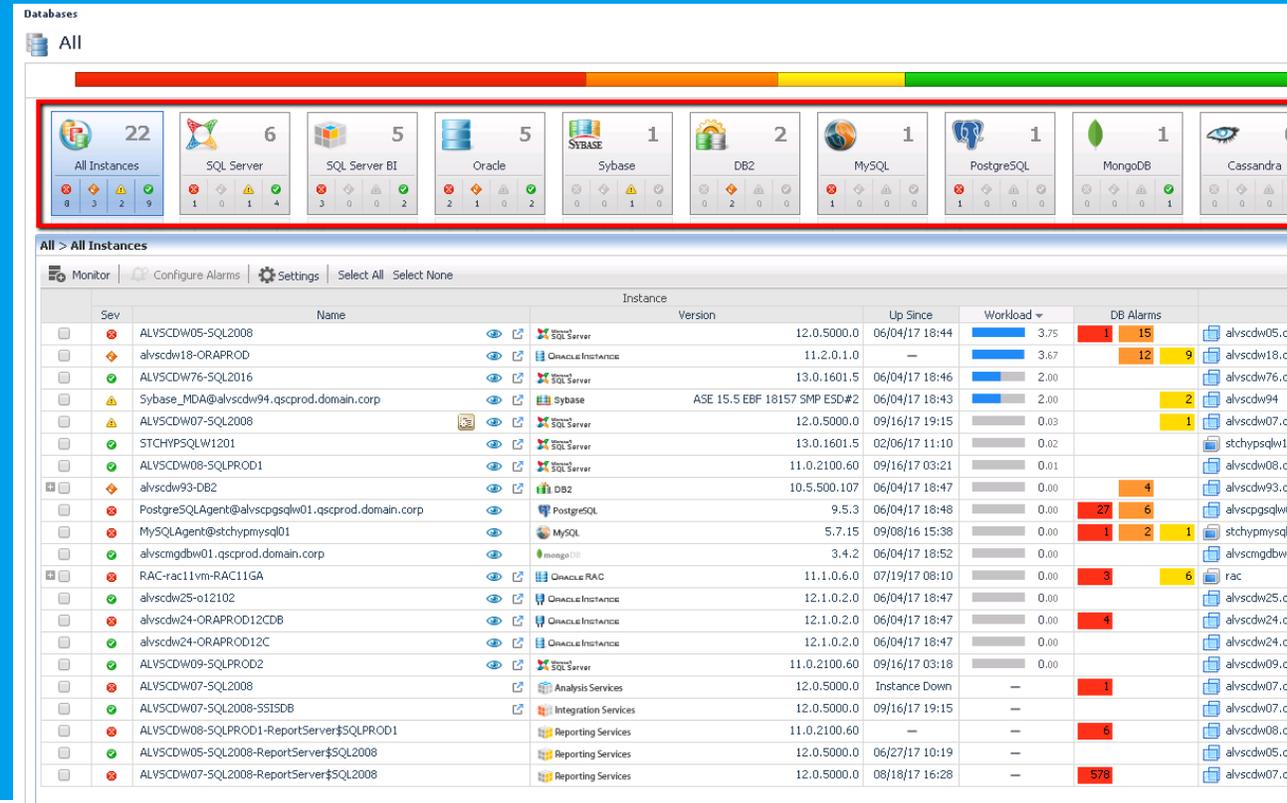
Capabilities

- Cost & Budget Management
- Reporting
- Automated Action
- Automated Remediation
- Change Analysis
- Capacity Management
- Capacity Planning
- Resource Optimization & Tuning
- DevOps & CI/CD
- SLA & OLA
- RPO & RTO

FOGLIGHT

Visibility over your entire database environment

- Single Pane of Glass with drill downs
- Selecting an environment to investigate, Foglight will display high level metrics which will be arranged by the degree of severity of its issues.
- The red, orange and yellow alarms enables us to find our biggest problems quickly and work through them efficiently.
- The tiles on top offer a view of the Database platforms. Here you see SQL Server, Azure, Oracle, SyBase, DB2, MySQL, PostgreSQL, MongoDB, Cassandra.



FOGLIGHT

Database workload optimization

To help you focus performance tuning efforts, Foglight quickly locates resource-intensive SQL statements, statements that have had execution plan changes with adverse results, and statements that are using more CPU or experiencing long wait time.

It also helps you visualize the database process model so you can see where bottlenecks are occurring or where configuration changes might help.

If a resource contention or consumption problem is identified, Foglight will guide the investigation and finding root cause and eliminating it is the goal.

The screenshot displays the Foglight SQL Activity tool interface. At the top, there are navigation tabs for SQL Statements, TSQL Batches, and Databases. The main window is divided into several sections:

- Summary:** Shows a list of the top 10 SQL statements ordered by CPU usage. The top statement is an UPDATE query with a CPU wait of 206.97 and CPU usage of 3,658.
- Resource Consumption:** A grid of six line graphs showing metrics like Response Time, Cache Hit Rates, Call Rates, and Connected Sessions over time.
- SQL Instance Summary:** A code editor showing the SQL statement: `select * from [Production].[TransactionHistory]`.
- Performance Tree:** A tree view on the left showing the instance structure, including SQL Statements, TSQL Batches, Databases, Programs, Users, Client Machines, Context Infos, Command Types, Sessions, Locked Objects, Objects I/O, Files, and Disks.
- Advanced Analytics:** A section with a dimension filter set to 'Instance View' and a resource consumption graph.
- Change Tracking / Advisories:** A table listing advisories with their priority and descriptions. One advisory is highlighted: 'CPU Usage Deviation (6)' with a low priority, occurring between 09/16/21 06:29 PM and 09/16/21 07:29 PM.

CPU Usage Deviation

Name
Overall CPU usage exceeds the

Description
The total instance CPU usage ex performance problem, the perfo or unexpected activity and sho

Figure 1 illustrates the total CPU the timeframe. Essentially, this timeframe, relative to a calculat

Additional information
Figure 2 illustrates how CPU

FOGLIGHT

Database workload optimization

Automated Tuning

1. Automated SQL scan, optimization,
2. Index optimization, and benchmarking;
3. Rewrite inefficient SQL

The screenshot shows the SQL Rewrite tool interface. The top pane displays the original SQL query:

```
SELECT Location.[Name], Location.LocationID, Product.[Name], Product.ProductNumber FROM (AdventureWorks.Production.ProductInventory ProductInv INNER JOIN AdventureWorks.Production.Location Location ON (ProductInventory.LocationID = Location.LocationID) INNER JOIN AdventureWorks.Production.Product Product ON (ProductInventory.ProductID = Product.ProductID))
```

The middle pane shows the execution plan for the original query, including a Hash Match / Inner Join and a Clustered Index Scan.

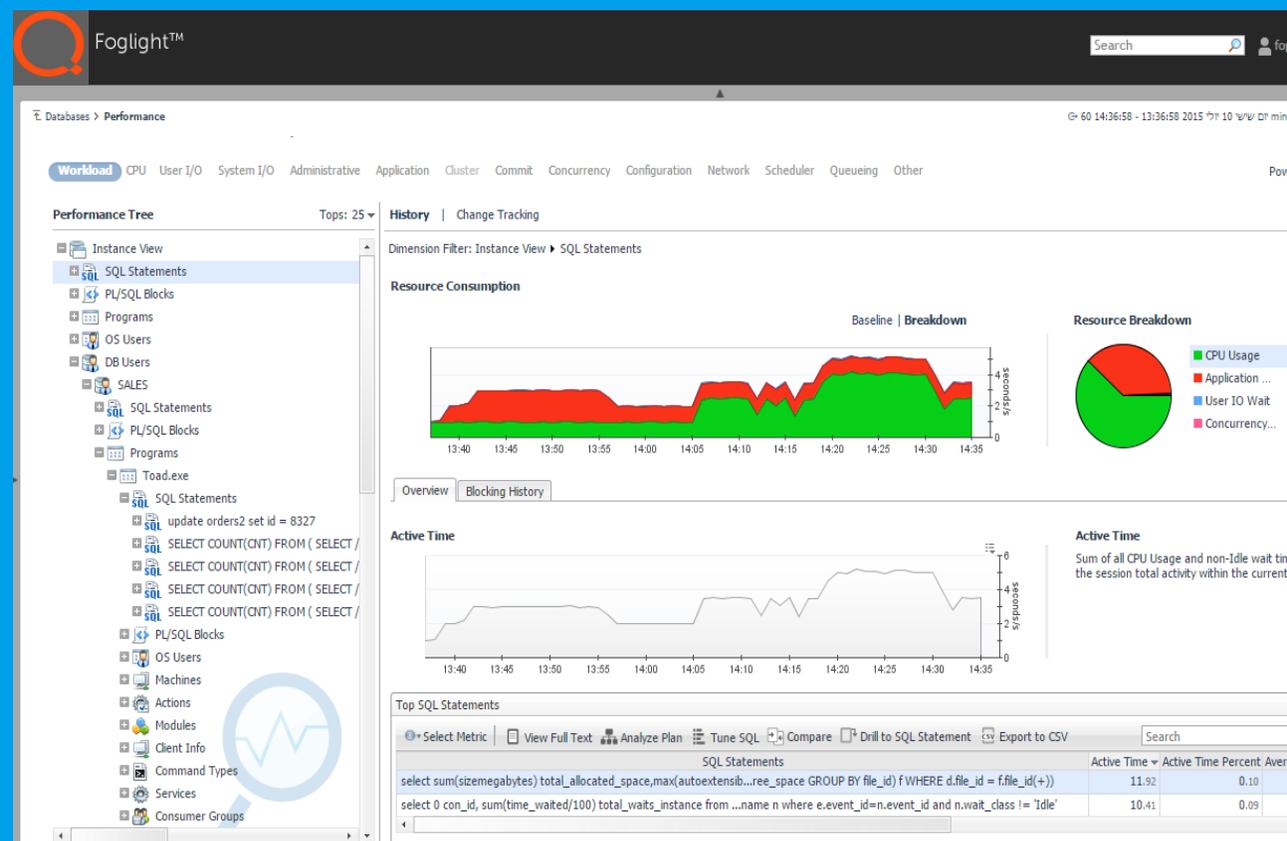
The bottom pane shows a table of alternatives:

Scenario Name	Plan Cost	Executions	Record Count	Status	Total Elapsed Time	Total Elapsed Time	Execution Elapsed Time	Execution CPU Time	Complete Elapsed Time	Complete CPU
Original	0.09438580	3	1,069		00:00:00.023	00:00:00.023	00:00:00.019	00:00:00.005	00:00:00.004	00:00:00.000
Index1	0.07509439	3	1,069		00:00:00.003	00:00:00.003	00:00:00.003	00:00:00.000	00:00:00.000	00:00:00.000
AR1	0.27909650	3	1,069		00:00:00.009	00:00:00.009	00:00:00.005	00:00:00.005	00:00:00.004	00:00:00.000
AR2	0.06802380	3	1,069		00:00:00.006	00:00:00.006	00:00:00.003	00:00:00.000	00:00:00.003	00:00:00.000
AR3	0.10708870	3	1,069		00:00:00.004	00:00:00.004	00:00:00.003	00:00:00.005	00:00:00.001	00:00:00.000
AR4	0.11873410	3	1,069		00:00:00.009	00:00:00.009	00:00:00.004	00:00:00.005	00:00:00.005	00:00:00.000
AR5	0.09237212	3	1,069		00:00:00.005	00:00:00.005	00:00:00.003	00:00:00.005	00:00:00.002	00:00:00.000
AR6	0.13142580	3	1,069		00:00:00.005	00:00:00.005	00:00:00.004	00:00:00.005	00:00:00.001	00:00:00.000
AR7	0.27921180	3	1,069		00:00:00.014	00:00:00.014	00:00:00.004	00:00:00.010	00:00:00.012	00:00:00.000

FOGLIGHT

Deep-dive Multi-Dimensional workload analysis

- Analyze workload elements and wait stats by database, application, user, host or statement.
- Enables you to find your biggest workload offenders; categorized by programs, users, databases, clients, stored procedures...
- You can “drill down” to the SQL statement to find your worst performing SQL and fix it using our SQL Optimizer.
- Customers use this to audit their users to ensure that resources are being allocated appropriately.



FOGLIGHT

Compare configuration and performance statistics

- Offers you the ability to quantify large workloads, if it's within Normal bounds or it's an abnormal activity.
- for example, the Tuesday after Memorial Day this year to the Tuesday after Memorial Day last year.
- Another example is 2 different instances with similar configuration. Compare them side by side to see where the breakdown is occurring.



Foglight™

Databases > Performance > Compare
ISRAIX07-01124 | Summary | SQL Performance | Activity | Storage | Configuration | Alert Log

Comparison Parameters

Comparison results based on: Workload

Time Range: 07/10/15 02:29 PM - 07/10/15 04:15 PM
Instance : ISRAIX07-01124

[Modify](#)

Compared To

Time Range: 07/10/15 02:29 PM - 07/10/15 04:15 PM
Instance : ISRVMRH541-o12102cdb

[Start New Comparison](#)

Workload



Comparison: Overview | Configuration

Statistics

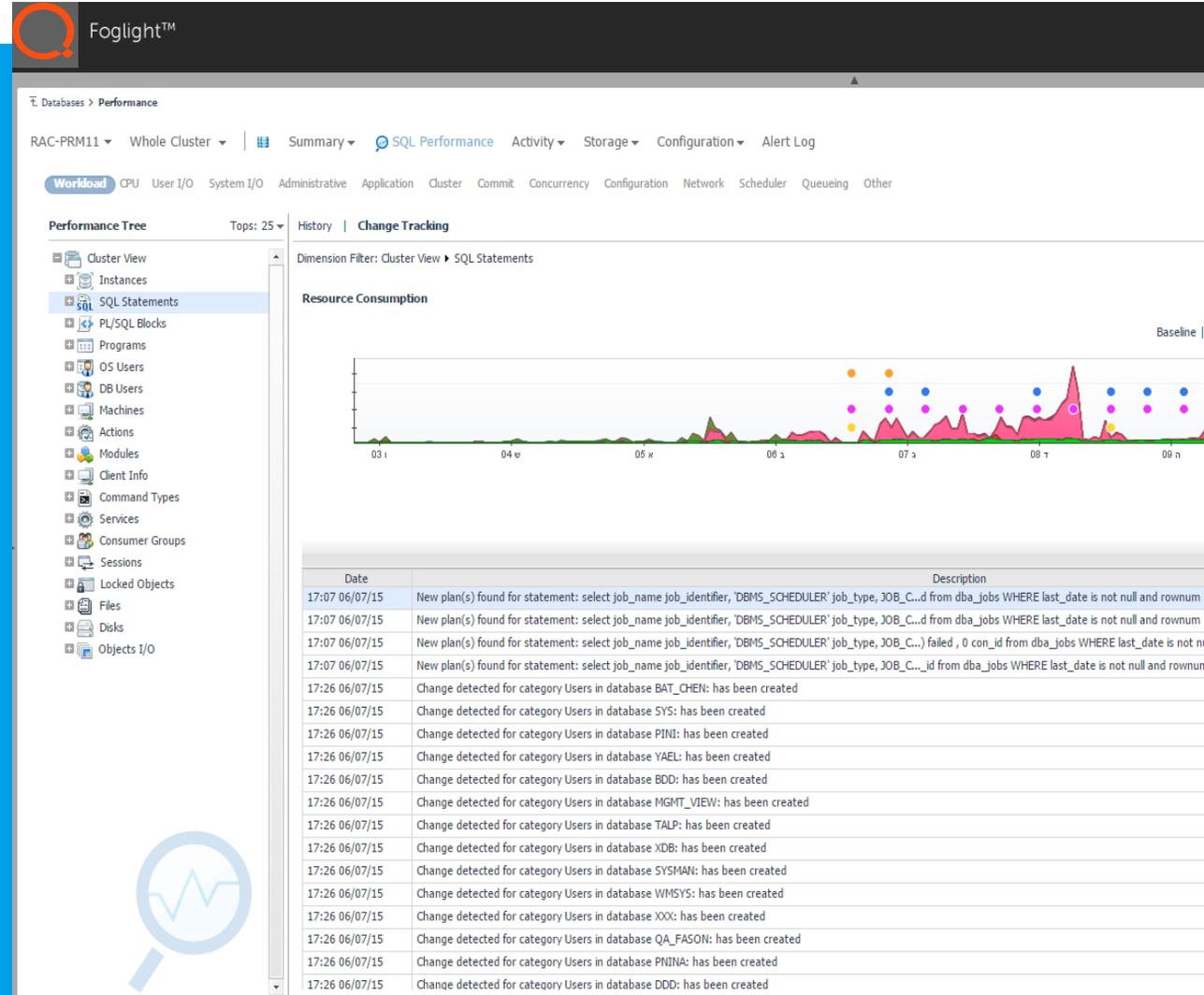
Active Time (second)	14,641	3,239
Average SQL Response Time (seconds)	0.00	1.6
Buffer Gets (operations)	13,171,942	1,874,691
Bytes sent via SQL*Net to client (byte)	7,594,584	11,219,666
Direct Writes (writes)	74,268,829	22,045
Disk Reads (reads)	74,268,829	22,045
Elapsed Time (second)	15,330	10,861
Executions	74,262,008	5,404
Fetches	74,268,829	22,045
Parse Count (hard) (parses)	11	105
Parse Count (total) (parses)	5,444,142	6,326
Recursive Calls (calls)	5,545,822	89,818
Rows Processed	74,344,928	198,566
Sorts	10,419	3,341
SQL*Net roundtrips to/from client (roundtrips)	18,362	28,412
User Commits (commits)	2.7	10

FOGLIGHT

Track changes and visually correlate them to workload

Provides the ability to track changes for your DevOps practice

- Track and assess the performance of continuous delivery of code deployed into production
- Changes that have an adverse affect on performance can be viewed quickly through multiple dimensions, including who made the change



The screenshot displays the Foglight Performance interface. On the left, a 'Performance Tree' lists various system components, with 'SQL Statements' selected. The main area features a 'Resource Consumption' chart showing performance metrics over time. Below the chart is a table of detected changes.

Date	Description
17:07 06/07/15	New plan(s) found for statement: select job_name job_identifier, 'DBMS_SCHEDULER' job_type, JOB_C...d from dba_jobs WHERE last_date is not null and rownum
17:07 06/07/15	New plan(s) found for statement: select job_name job_identifier, 'DBMS_SCHEDULER' job_type, JOB_C...d from dba_jobs WHERE last_date is not null and rownum
17:07 06/07/15	New plan(s) found for statement: select job_name job_identifier, 'DBMS_SCHEDULER' job_type, JOB_C... failed, 0 con_id from dba_jobs WHERE last_date is not null and rownum
17:07 06/07/15	New plan(s) found for statement: select job_name job_identifier, 'DBMS_SCHEDULER' job_type, JOB_C..._id from dba_jobs WHERE last_date is not null and rownum
17:26 06/07/15	Change detected for category Users in database BAT_CHEM: has been created
17:26 06/07/15	Change detected for category Users in database SYS: has been created
17:26 06/07/15	Change detected for category Users in database PINI: has been created
17:26 06/07/15	Change detected for category Users in database YAEL: has been created
17:26 06/07/15	Change detected for category Users in database BDD: has been created
17:26 06/07/15	Change detected for category Users in database MGMT_VIEW: has been created
17:26 06/07/15	Change detected for category Users in database TALP: has been created
17:26 06/07/15	Change detected for category Users in database XDB: has been created
17:26 06/07/15	Change detected for category Users in database SYSMAN: has been created
17:26 06/07/15	Change detected for category Users in database WMSYS: has been created
17:26 06/07/15	Change detected for category Users in database XXX: has been created
17:26 06/07/15	Change detected for category Users in database QA_FASON: has been created
17:26 06/07/15	Change detected for category Users in database PNINA: has been created
17:26 06/07/15	Change detected for category Users in database DDD: has been created

FOGLIGHT

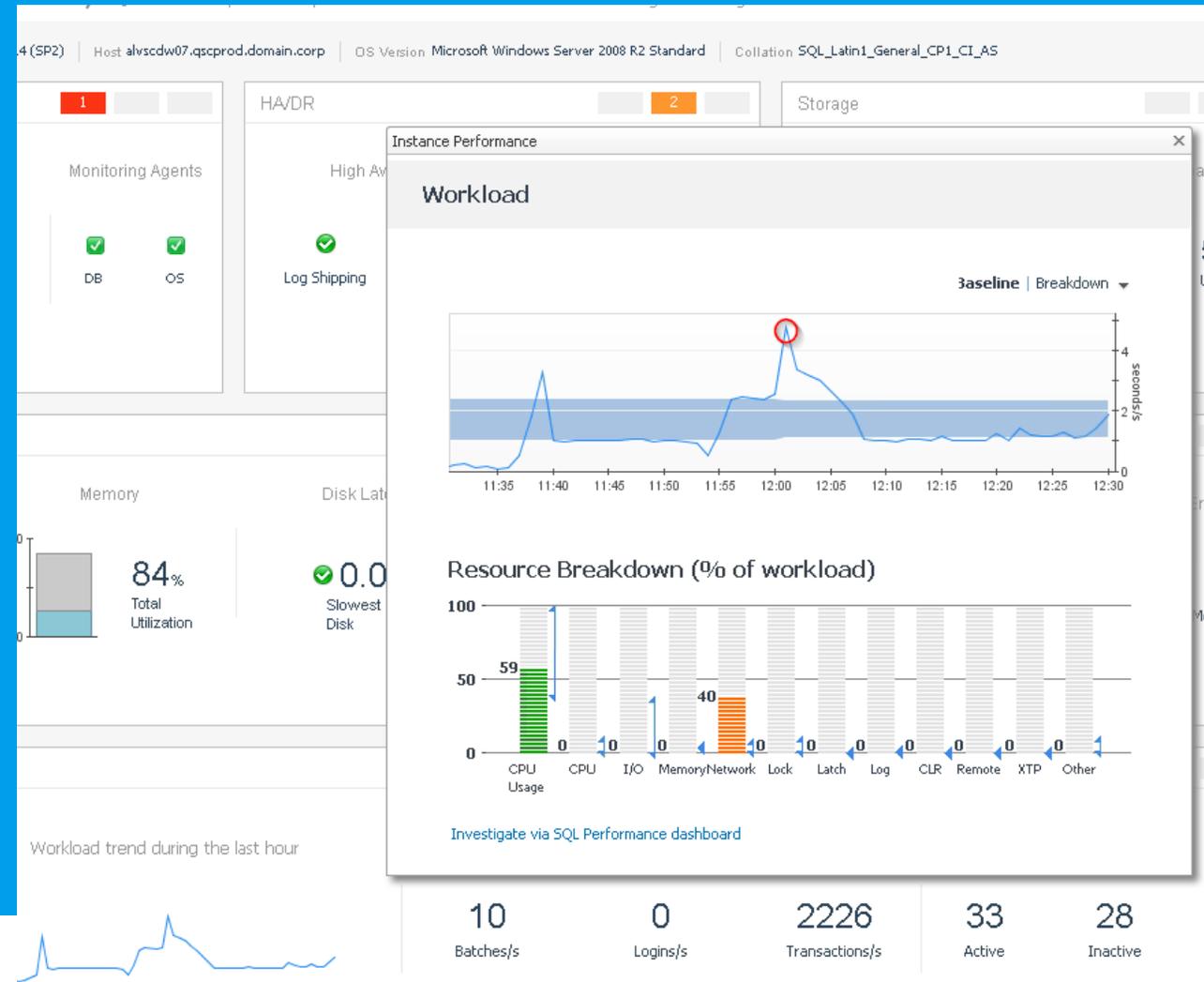
Baselining – What is normal?

Learn what is “normal” and know when performance is out of normal range.

Removes the mystery of what “normal” looks like for your environment.

Foglight automatically learns the behavior of your environment and creates a baseline to show that your system is operating “normally”.

When you know what normal looks like, it gives you the power to pinpoint spikes or breakdowns in performance

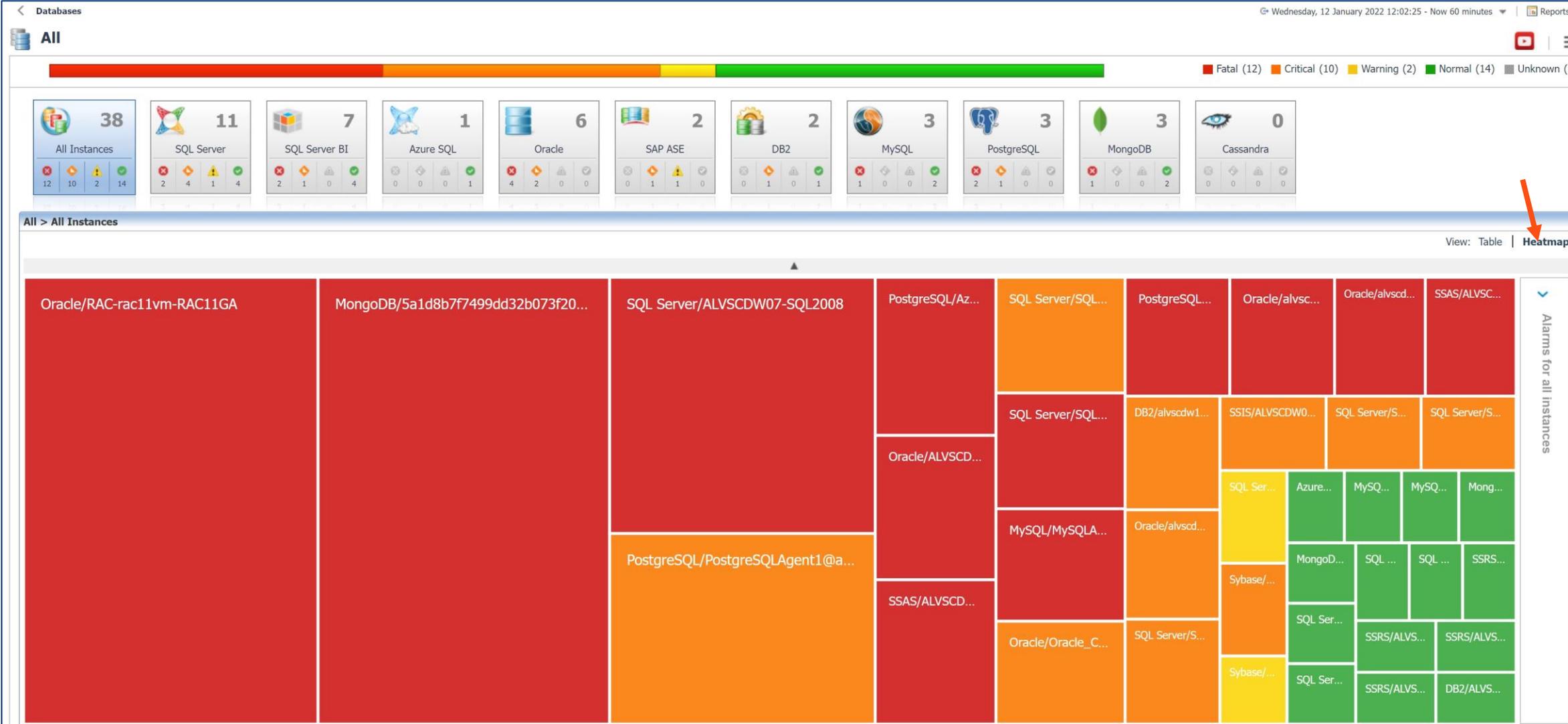




Demo

Quest

Cross Platform Databases Global View



Cross Platform Databases Global View

The dashboard provides a comprehensive overview of database health and performance. At the top, a status bar indicates the overall system health with a color-coded progress bar and a legend: Fatal (7), Critical (12), Warning (5), Normal (13), and Unknown (0). Below this, a grid of database type tiles shows the count of instances for each category: All Instances (37), SQL Server (10), SQL Server BI (6), Azure SQL (1), Oracle (7), SAP ASE (2), DB2 (2), MySQL (2), PostgreSQL (3), MongoDB (3), and Cassandra (1).

The main section, titled 'All > All Instances', displays a detailed table of database instances. The table includes columns for severity, name, instance details, version, uptime, workload, database alarms, host information, and system utilization metrics (CPU Load, Memory, Disk Busy, and Host Alarms). An orange arrow points to the 'View: Table' button in the top right corner of the table area.

Sev	Name	Instance	Version	Up Since	Workload	DB Alarms	Host	System Utilization	Host Alarms		
								CPU Load (%)	Memory (%)	Disk (% Busy)	
Fatal	ALVSCDW07-SQL	Microsoft SQL Server	12.0.5207.0	01/10/21 19:30	2.73	1 Fatal, 31 Critical, 1 Warning	alvscdw07	40%	15%	89%	1 Critical
Critical	SQLSERVER-RD	Microsoft SQL Server	14.0.3049.1	01/28/20 07:56	1.98	2 Critical	ec2amaz	-	-	-	0
Warning	SAPASE01	SAP ASE	ASE 16.0 SP03 PL02 EBF 27415 SMP	02/01/21 08:04	1.00	1 Warning	sapase01	9%	6%	32%	0
Critical	ALVSCDW08-SQL	Microsoft SQL Server	11.0.2100.60	04/15/21 18:40	0.10	2 Critical	alvscdw08	12%	92%	31%	0
Critical	SQLPIREP01	Microsoft SQL Server	15.0.2000.5	01/10/21 19:17	0.04	8 Critical, 1 Warning	sqlpirep01	3%	7%	24%	0
Critical	alvscdw162-DB	DB2	11.5.0.1077	01/10/21 19:27	0.01	4 Critical	alvscdw162	1%	35%	76%	0
Critical	alvscdw93-DB2	DB2	10.5.500.107	01/10/21 19:27	0.00	2 Critical	alvscdw93	1%	48%	0%	0
Critical	SQLNODE2.QS	Microsoft SQL Server	15.0.2070.41	01/10/21 19:30	0.00	2 Critical, 3 Warning	sqlnode2	1%	6%	64%	0
Critical	alvscdw230.qsc	ORACLE Instance	12.2.0.1.0	01/10/21 19:28	0.00	1 Fatal, 3 Warning	alvscdw230	-	-	-	0
Critical	alvscdw18-ORA	ORACLE Instance	11.2.0.1.0	02/16/21 13:32	0.00	2 Critical, 6 Warning	alvscdw18	-	-	-	0
Warning	demoCluster	Cassandra	3.11.2	03/03/21 06:09	0.00	4 Warning	demoCluster	-	-	-	0
Normal	JASONAZURESQL	SQL Azure	12.0.2000.8	05/31/17 06:19	0.00	0	jasonaz	-	-	-	0
Critical	PostgreSQL2	PostgreSQL	12.4	01/10/21 19:28	0.00	1 Fatal, 1 Warning	pgsqlw	-	-	-	0
Critical	PostgreSQLAge	PostgreSQL	9.5.3	01/10/21 19:29	0.00	28 Critical	alvscpg	-	-	-	0
Critical	AzurePostgreSQL	PostgreSQL	9.6.16	03/19/21 15:33	0.00	1 Fatal, 5 Warning	jasonaz	-	-	-	0
Normal	MySQLAgent@	MySQL	5.7.15	06/11/20 18:10	0.00	0	stchypn	-	-	-	0
Normal	MariaDBAmazon	MariaDB	10.2.21	04/08/21 04:23	0.00	0	ip-10-2	-	-	-	0
Normal	alvscmgdbw01	MongoDB	3.4.2	01/10/21 19:27	0.00	0	alvscmg	10%	88%	0%	0
Critical	ALVSCDW24-ORA	ORACLE Instance	12.1.0.2.0	02/26/21 07:23	0.00	2 Critical, 10 Warning	alvscdw24	-	-	-	0
Normal	alvscdw24-ORA	ORACLE Instance	12.1.0.2.0	02/26/21 07:23	0.00	0	alvscdw24	-	-	-	0
Warning	RAC-rac11vm-F	ORACLE RAC	11.1.0.6.0	02/28/21 09:26	0.00	0	rac	6%	80%	1%	0
Critical	Oracle_Cosmos	ORACLE Instance	12.1.0.2.0	09/21/20 17:47	0.00	5 Critical, 2 Warning	oracle	-	-	-	0
Normal	STCHYPSQLW1	Microsoft SQL Server	13.0.1601.5	04/15/21 05:31	0.00	0	stchyps	1%	27%	0%	1 Critical
Normal	APEXSQLMW01	Microsoft SQL Server	14.0.2027.2	01/10/21 19:27	0.00	0	apexsqlm	12%	17%	0%	1 Warning

Cross Platform Databases Alarm Analysis (from "Alarms" navigation option)

Alarms

This dashboard shows the information of system alarms and changes, and facilitates the investigation of top issues in your environment.

Alarms by Time | Heatmap | **Alarm Analysis** | Foglight Today | Foglight Now | Blackouts

Alarms by Source

Alarm Source	Alarm Count	Severities			Alarm Duration		
		F	C	W	Min	Max	Avg
VMW Virtual Machine VMware Tools	87			87	1.2 d	2.2 d	2.2 d
PostgreSQL Index Bloat	31		31		4.8 hr	4.8 hr	4.8 hr
VMW Virtual Machine Logical Drive Availability	18	9	1	8	4.8 hr	1.4 d	6.7 hr
DBSS - Days Since Last Backup - No Backup	14			14	20 hr	3.2 d	2.3 d
VMW Virtual Machine Logical Drive Estimated Fill Time	11	1	3	7	13 hr	2.5 d	1.5 d
DBSS - File Group Utilization	10	6	2	2	7.2 hr	2.4 d	1.0 d
DBSS - Connection Time	9		8	1	10.0 min	7.0 hr	1.5 hr
DBSS - Days Since Last Backup	9		1	8	8.4 hr	3.2 d	16 hr
DBSS - Page Splits	7			7	4.8 min	3.9 hr	1.3 hr
DBO - Tablespace Used Percentage	7		6	1	4.2 hr	4.6 hr	4.4 hr
Kubernetes Container Memory Utilization	7	4	3		4.9 min	2.5 d	19 hr
File System Utilization Copy	6	5		1	4.8 hr	11 hr	9.7 hr
File System Capacity	6	5		1	4.8 hr	11 hr	9.7 hr
MongoDB Replica Member Unreachable	6	6			11 hr	11 hr	11 hr
MongoDB Replication Oplog Lag	6	6			2.4 d	3.2 d	2.8 d
VMware Virtual Machine OS reboot	5	5			1.1 min	10 min	6.3 min
DBO - Usability Connection Availability	5	5			4.7 hr	11 hr	6.0 hr
Azure Region Quota Usage Utilization	5		5		1.4 d	2.2 d	1.6 d
DB2 - Database Memory Pool Utilization	5		5		45 min	2.2 d	1.3 d
VMW Phantom Snapshots	5			5	13 hr	2.5 d	1.9 d
DBO - Usability Availability Single	4	4			4.7 hr	11 hr	7.8 hr

Counts by Severity

Sev	Name	Count
Warning	Warning	190
Critical	Critical	101
Fatal	Fatal	88

Max Number of Evaluated Alarms:

Market Leading Monitoring & Analytics for SQL Server

The screenshot displays the Quest SQL Server monitoring interface for instance ALVSCDW07-SQL2008. The dashboard is divided into several sections:

- Availability:** Shows the instance is 'Up' for 4m 25d 21h with 100% availability during the last hour. Monitoring agents for DB and OS are active.
- HA/DR:** High Availability (Log Shipping, Replication) and Disaster Recovery (Database Missing Backups) are all in a 'Good' state.
- Storage:** Data Space utilization is at 84% and Log space utilization is at 9%.
- Infrastructure:** CPU total utilization is 20%, Memory total utilization is 34%, and CPU overhead is 0%.
- Operational:** Shows 0 failures, 0 long-running jobs, and 0 error messages.
- Instance Performance:** Includes a resource breakdown pie chart, a workload trend graph, and throughput metrics: 6 Batches/s, 0 Logins/s, 29 Transactions/s, 32 Active sessions, 33 Inactive sessions, and 0 Blocked sessions.

A blue box highlights the 'Order by: Active Time' dropdown menu, which lists the following sorting options: SQL Statement, CPU, I/O, Lock, and Active Time. The 'Active Time' option is currently selected.

Below the dropdown, a table of query results is visible, showing the following queries and their execution times:

Query	Time
BACKUP database [AdventureWorks] TO VIRTUAL_DEVICE='VD1_362FDE92...' MAXTRANSFERSIZE=524288, NAME='NAdventureWorks - diff Backup'	3,645.86
SELECT LineTotal FROM Sales.SalesOrderDetail WHERE UnitPrice < @counter GROUP BY LineTotal ORDER BY LineTotal	343.72
SELECT CU.CustomerID, QQ.* FROM Sales.Customer CU INNER JOIN (SE...D.CustomerID, SD.ProductID) QQ ON QQ.CustomerID = CU.CustomerID	59.51
SELECT CU.CustomerID, SD.ProductID, SUM(SD.OrderQty) AS Units, S...D.ProductID = @ProductID GROUP BY CU.CustomerID, SD.ProductID	49.65
SELECT top (@P0) t2.spid, t2.login_time, t2.request_id, t1.start...sion_id > 0 where (t2.spid > 50 and t2.status <> 'background')	16.77
SELECT COUNT(*) FROM Sales.SalesOrderDetail WHERE UnitPrice > @holder	15.48
SET @holder = (SELECT MIN(UnitPrice) FROM Sales.SalesOrderDetail WHERE UnitPrice > @holder)	15.35

Market Leading Workload Analytics for SQL Server

The screenshot displays the SQL Server Workload Analytics interface. On the left, the Performance Tree shows the instance view expanded to SQL Statements. The main area features a Resource Consumption graph and a Resource Breakdown pie chart. Below these is the Overview tab, which contains a table of Top SQL Statements.

Resource Breakdown

Resource	Percentage
Other Wait	58.09%
CPU Usage	34.42%
Network Wait	3.30%
CPU Wait	2.98%

Top SQL Statements

SQL Statements	Query Hash	Active Time	CPU Usage	Average SQL Response Time	Executions
SELECT LAST_NAME, STREET INTO #tmp_sales FROM EMPLOYEE, ORDER_LI...AND CITY = 'H2J' ORDER BY LAST_NAME, STREET OPTION (MAXDOP 1)	0x7bbfef38ddd45161	610.25	549.01	95.81	6.00
FETCH getsumamount INTO @CUSTNME, @ORDERNBR, @PRODESC, @AMOUNT;	0x26d1d39d00000000	487.45	464.69	2.39	204.00
SELECT LineTotal FROM Sales.SalesOrderDetail WHERE UnitPrice < @counter GROUP BY LineTotal ORDER BY LineTotal	0x4655e53c2874f0bc	424.51	382.94	0.16	2,696.00
INSERT INTO #GETSUMAMOUNT(CUSTOMER_NAME, ORDER_NUMBER, PRODUCT_...ION, AMOUNT) VALUES(@CUSTNME, @ORDERNBR, @PRODESC, @AMOUNT)	0x74053ced6c237b0d	358.79	343.91	0.00	9,649,155.00
OPEN getsumamount;	0x6eb5197c00000000	436.07	228.22	121.44	1.00
SELECT LTRIM(RTRIM(LAST_NAME)) + ', ' + LTRIM(RTRIM(FIRST_NAME))...O.CUSTOMER_ID = C.CUSTOMER_ID AND P.PRODUCT_ID = L.PRODUCT_ID	0x8c4bc2e786dfd63a	121.38	45.93	20.48	1.00
SELECT /* TUNING */ e.LAST_NAME, d.DEPARTMENT_DESCRIPTION, j.JOB...d.DEPARTMENT_DESCRIPTION, j.JOB_DESCRIPTION OPTION (MAXDOP 1)	0x4d10191ae01d34f0	266.37	45.65	51.57	5.00
SELECT COUNT(*) FROM Sales.SalesOrderDetail WHERE UnitPrice > @holder	0xc3ec1a852931da03	46.87	29.02	0.21	225.00
SET @holder = (SELECT MIN(UnitPrice) FROM Sales.SalesOrderDetail WHERE UnitPrice > @holder)	0x2f996f8d4a302444	45.79	28.09	0.20	223.00
SELECT CustomerID, TotalPurchaseYTD, DateFirstPurchase, BirthDat...erCarsOwned FROM AdventureWorks.Sales.vIndividualDemographics	0x2f51f94a21b73aaf	26.41	24.23	13.21	2.00
SELECT /* INDEX */ a.ORDER_ID, a.AMOUNT, b.PRODUCT_DESCRIPTION I...ROM ORDER_LINE a, PRODUCT b WHERE a.PRODUCT_ID = b.PRODUCT_ID	0x9b8137fad269851b	18.47	18.15	9.24	2.00
WHILE @@FETCH_STATUS = 0	0x4336e8d000000000	19.22	17.92	1.07	18.00
WAITFOR	0x1417556400000000	16.44	16.44	250.04	14.00

Market Leading Workload Analytics for SQL Server

The screenshot displays the SQL Server Workload Analytics interface for instance 'ALVSCDW07-SQL2008'. The interface includes a navigation menu on the left, a top navigation bar, and several data visualization components:

- Performance Tree:** A tree view on the left showing the instance structure, including SQL Statements, Databases, Programs, Users, Client Machines, Context Infos, Command Types, Sessions, Locked Objects, Objects I/O, Files, and Disks.
- Resource Consumption Chart:** A line chart showing resource consumption over time from 04:20 to 12:00. A shaded blue area represents the baseline, and a red box with an arrow points to it with the text: "Baselines: shaded blue areas are calculated time-sensitive normal ranges of behavior; deviations are a red flag".
- Resource Breakdown Pie Chart:** A pie chart showing the distribution of resource usage:

Resource	Percentage
Other Wait	59.41%
CPU Usage	33.54%
Network Wait	3.82%
CPU Wait	2.26%
- CPU Usage Chart:** A line chart showing CPU usage over time from 04:20 to 12:00. A red arrow points to a peak in the usage.
- Workload related Metrics Table:** A table listing various metrics and their values:

Metric	Resource	Total
Executions	Workload	71,567,240.00
Active Time	Workload	54,999.69
CPU Usage	CPU	18,438.14
Wait Time Percent	Workload	66.46
Batches Rate	Workload	8.38
Logins Rate	Workload	0.36
Average SQL Response Time	Workload	< 0.01



Q&A

Quest

Thank You!

QuestTM

To learn more :

<https://www.quest.com/foglight/>