IDERA® | Solution Brief

GETTING STARTED WITH SQL DM FOR SQL SERVER

HOW TO QUICKLY DEPLOY, CONFIGURE, AND BENEFIT FROM SQL DIAGNOSTIC MANAGER

PURPOSE OF THIS DOCUMENT

Due to the depth and potential for customization of SQL Diagnostic Manager, it is possible to overlook features during the initial trial period. This text highlights often-missed and modifiable areas that give more complete control over management and reporting in SQL Server environments. For additional product information, visit the SQL Diagnostic Manager wiki page.

INTRODUCTION

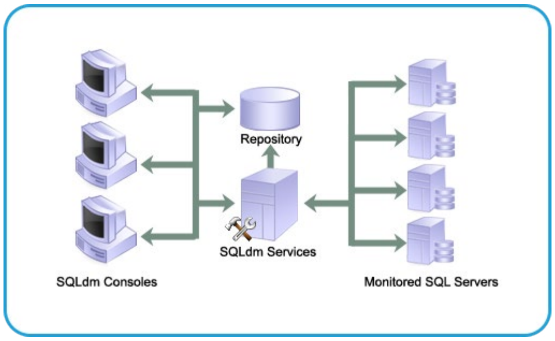
SQL Diagnostic Manager is a powerful performance monitoring and diagnostics solution that proactively alerts administrators to health, performance, and availability problems within SQL Server environments.

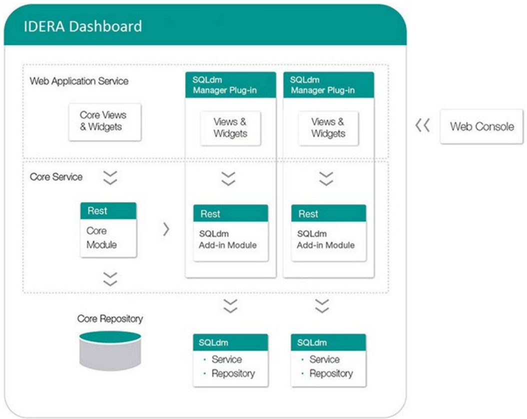
WHY USE SQL DIAGNOSTIC MANAGER?

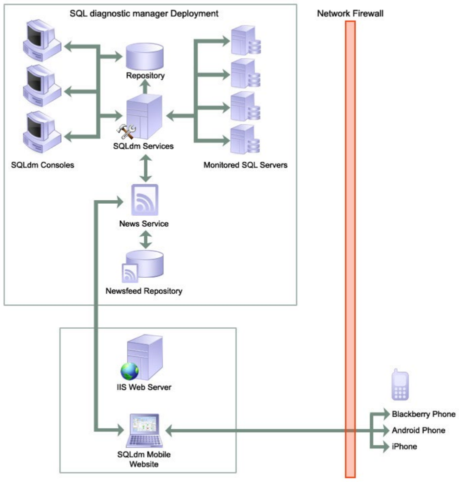
- Easily manage and organize monitored instances of SQL Server.
- Identify the impact of slow-running queries and analyze query performance.
- View in-depth analysis of sessions running on monitored instances of SQL Server.
- Alert predictively, automate alert actions and responses, set up server baselines, and analyze alert patterns.
- Analyze prescriptively with expert recommendations and executable scripts.
- View past performance and plan future capacity.
- Monitor tempdb to identify and resolve contention and performance issues.
- Connect to other tools by sending alerts to other systems and integrating with SCOM.
- Access the mobile console to see the entire SQL Server environment on mobile devices from anywhere.
- Deploy Web-Based Reports via SSRS for comprehensive auditing.
- Adapt to specific needs with custom counters, dashboards, and reports.
- Audit activity by viewing the main actions performed in the SQL Diagnostic Manager environment.
- Manage large environments by accessing multiple repositories and publishing reports from multiple data sources.

ARCHITECTURE

SQL Diagnostic Manager consists of a light, unobtrusive architecture that easily runs in SQL Server environments with minimal configuration. All components of SQL Diagnostic Manager run outside and separate from SQL Server processes. Refer also to the documentation SQL Diagnostic Manager components and architecture, IDERA Dashboard components and architecture, and SQLDM Mobile components and architecture including IDERA Newsfeed.

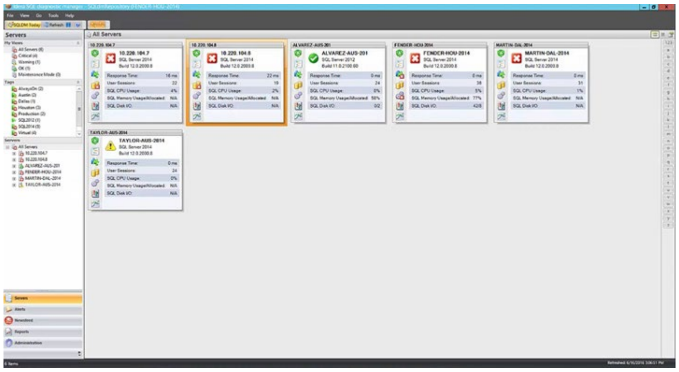






Desktop Console

Use the desktop console to view real-time status, configure alert notifications on specific metric thresholds at the server and database levels, view historical reports, and perform administrative functions. The desktop console retrieves historical information directly from the repository. All real-time requests use the product services to poll the monitored SQL Server.



Product Services

SQL Diagnostic Manager has three centralized services that reside on the same computer:

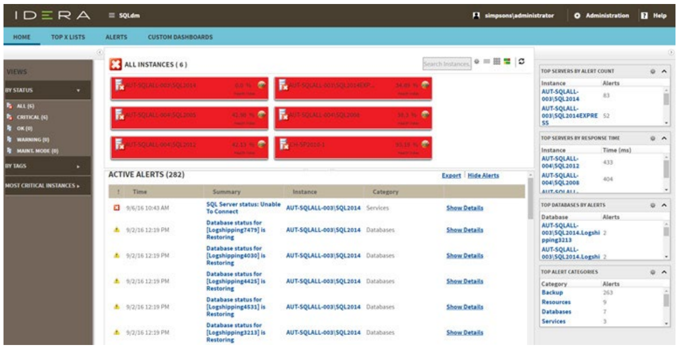
1. The management service provides real-time data to the desktop console, receives historical data from the collection service for storage in the repository, and raises alerts and sends alert notifications.
2. The collection service performs on-demand and scheduled collection from the monitored SQL Servers.
3. The predictive service calculates the alert forecast every hour and builds a forecasting model once per day.

Plug-In

When registering SQL Diagnostic Manager with the IDERA Dashboard, this deploys the product plug-in module. The plug-in consists of web views and widgets and a .NET Framework based add-in module. This plug-in deploys the web views and widgets in the web application service of the IDERA Dashboard, and the add-in in the core service of the same. The web application service dynamically loads in the views and widgets and makes them available to web console users. The views and widgets use the representational state transfer (REST) application programming interfaces (APIs) of the add-in to retrieve data. Likewise, the add-in retries data from the product services and repository. Refer also to the documentation.

Web Console in IDERA Dashboard

The IDERA Dashboard and the web console install automatically upon upgrade or during installation of version 9.0 higher of SQL Diagnostic Manager.



ACTIVE ALERTS (282)

!	Time	Summary	Instance	Category	
☒	9/6/16 10:43 AM	SQL Server status: Unable To Connect	AUT-SQLALL-003\SQL2014	Services	Show Details
⚠	9/2/16 12:19 PM	Database status for [Logshipping7479] is Restoring	AUT-SQLALL-003\SQL2014	Databases	Show Details
⚠	9/2/16 12:19 PM	Database status for [Logshipping4030] is Restoring	AUT-SQLALL-003\SQL2014	Databases	Show Details
⚠	9/2/16 12:19 PM	Database status for [Logshipping4425] is Restoring	AUT-SQLALL-003\SQL2014	Databases	Show Details
⚠	9/2/16 12:19 PM	Database status for [Logshipping4531] is Restoring	AUT-SQLALL-003\SQL2014	Databases	Show Details
⚠	9/2/16 12:19 PM	Database status for [Logshipping3213] is Restoring	AUT-SQLALL-003\SQL2014	Databases	Show Details

TOP SERVERS BY ALERT COUNT

Instance	Alerts
AUT-SQLALL-003\SQL2014	83
AUT-SQLALL-003\SQL2014EXPRESS	52

TOP SERVERS BY RESPONSE TIME

Instance	Time (ms)
AUT-SQLALL-004\SQL2012	433
AUT-SQLALL-004\SQL2008	404
AUT-SQLALL-	

TOP DATABASES BY ALERTS

Database	Alerts
AUT-SQLALL-003\SQL2014.Logshipping3213	2
AUT-SQLALL-003\SQL2014.Logshi	2

TOP ALERT CATEGORIES

Category	Alerts
Backup	263
Resources	9
Databases	7
Services	3

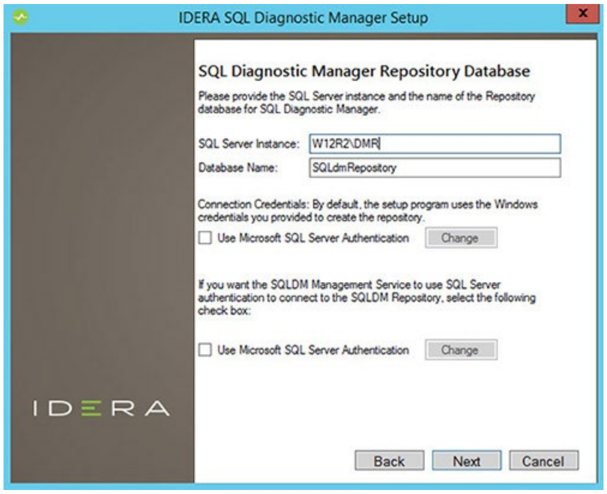
To access the web console:

- Open a web browser that is compatible with the web console requirements.
- Enter the product uniform resource locator (URL) http://<machinename>:<port>. Here, <machinename> is the name of the host or machine, and <port> is the port specified during installation. The default URL is http://<localhost>:9290.
- When the web console launches on the web browser, use the Windows user account <domain\user> with the corresponding password to log into the product.

The web application service of IDERA Dashboard comes with secure socket layer (SSL) already set up. Refer also to the documentation Run IDERA Dashboard over SSL (HTTPS).

Repository

The repository is a centralized SQL Server database that stores collected metrics on a scheduled basis, historical data, and alerts information. The repository also stores configuration information, such as the credentials used to monitor a registered SQL Server instance. Refer also to the documentation Connect to the SQL Diagnostic Manager Repository.



IDERA SQL Diagnostic Manager Setup

SQL Diagnostic Manager Repository Database

Please provide the SQL Server instance and the name of the Repository database for SQL Diagnostic Manager.

SQL Server Instance: W12R2\DMR

Database Name: SQLdmRepository

Connection Credentials: By default, the setup program uses the Windows credentials you provided to create the repository.

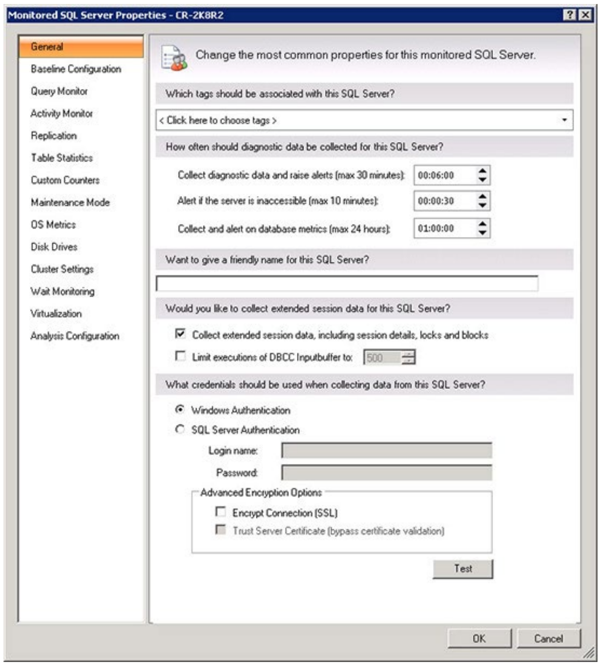
□ Use Microsoft SQL Server Authentication

If you want the SQLDM Management Service to use SQL Server authentication to connect to the SQLDM Repository, select the following check box:

□ Use Microsoft SQL Server Authentication

Authentication

SQL Diagnostic Manager uses the same types of authentication that are available in the SQL Server security model. Use Windows Authentication or SQL Server Authentication when specifying account credentials for the product services. SQL Server Authentication is required when no domain trust exists between the product services computer and the computers hosting the monitored SQL Server instances. For example, if the monitored SQL Server instances are located in an untrusted domain or behind a firewall, use SQL Server Authentication to deploy SQL Diagnostic Manager successfully. In this case, use the system administrator (SA) account or a SQL Server login that has system administrator permissions.



Monitored SQL Server Properties - CR-2K8R2

Change the most common properties for this monitored SQL Server.

Which tags should be associated with this SQL Server?

< Click here to choose tags >

How often should diagnostic data be collected for this SQL Server?

Collect diagnostic data and raise alerts (max 30 minutes): 00:06:00

Alert if the server is inaccessible (max 10 minutes): 00:00:30

Collect and alert on database metrics (max 24 hours): 01:00:00

Want to give a friendly name for this SQL Server?

Would you like to collect extended session data for this SQL Server?

☑ Collect extended session data, including session details, locks and blocks

□ Limit executions of DBCC Inputbuffer to: 500

What credentials should be used when collecting data from this SQL Server?

● Windows Authentication

○ SQL Server Authentication

Login name:

Password:

Advanced Encryption Options

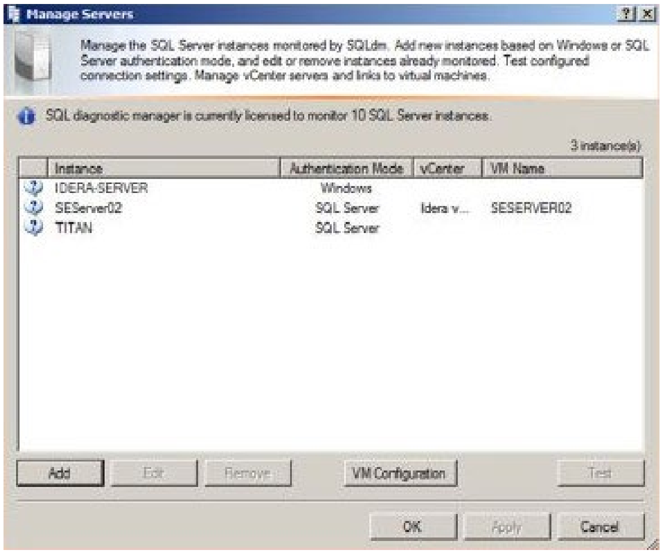
□ Encrypt Connection (SSL)

□ Trust Server Certificate (bypass certificate validation)

CAPABILITIES

Manage SQL Server Instances

To add an instance to SQL Diagnostic Manager, select File > Manage Servers and follow the instructions in the Manage Server window. This window displays all of the SQL Server instances monitored by SQL Diagnostic Manager. After adding an instance, SQL Diagnostic Manager polls key performance metrics to start identifying which areas of the SQL Server environment need attention the soonest. Within seconds, identify areas of the environment that cause pain. These areas include areas that were not known to have a connection to experienced issues. Refer also to the documentation Manage your SQL Server instances and Add your SQL Server instances.



Manage Servers

Manage the SQL Server instances monitored by SQLdm. Add new instances based on Windows or SQL Server authentication mode, and edit or remove instances already monitored. Test configured connection settings. Manage vCenter servers and links to virtual machines.

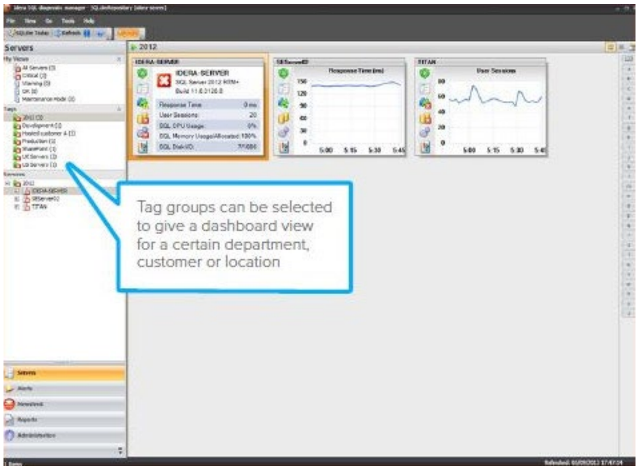
SQL diagnostic manager is currently licensed to monitor 10 SQL Server instances.

3 instance(s)

Instance	Authentication Mode	vCenter	VM Name
IDERA-SERVER	Windows		
SEServer02	SQL Server	Idera v...	SESERVER02
TITAN	SQL Server		

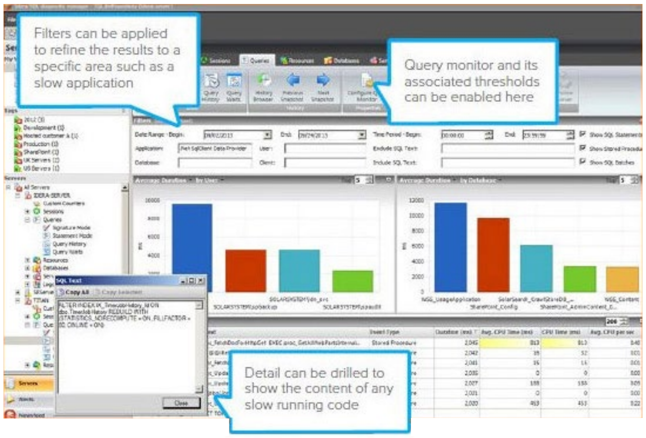
Organize Instances

Apply one or more text-based labels to instances to ensure ease of management for reporting and user management, and creating an additional dashboard for each department, geographical location, and hosted customer. After tagging an instance, the instance belongs to a group. Selecting that group displays all members in a dashboard style view used to monitor and segment departments, locations, and versions for each thumbnail in the list. Change the display to show an overall view or details specific to any of the key performance indicators. Refer also to the documentation Work with tags.



Tag groups can be selected to give a dashboard view for a certain department, customer or location

Identify Impact of Slow-Running Queries



Filters can be applied to refine the results to a specific area such as a slow application

Query monitor and its associated thresholds can be enabled here

Detail can be drilled to show the content of any slow running code

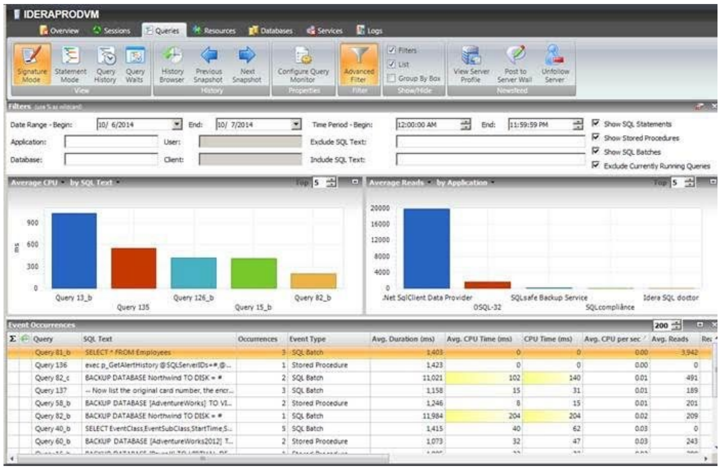
Identify slow-running code that is one of the main reasons for performance bottlenecks in SQL Server. The Query Monitor is a standard SQL Server trace that collects all of the events that occur on SQL Server instances over a period. Enable this option when experiencing query timeouts or other performance issues. Manually enable and disable the Query Monitor. Enable the Query Monitor automatically by using an alert action so that the Query Monitor does not have to be utilized all of the time but only for the periods that the Query Monitor is needed. Refer also to the documentation Set query monitor options.

Analyze Query Performance

Analyze all queries or selected queries several different ways.

Signature Mode View (General Performance)

View individual SQL statements or to view query signatures. Query signatures are groupings of SQL Statements that match after stripping their literals. A query signature broadly defines queries and trends with a less overwhelming amount of data to diagnose a query in a general sense. After identifying a potential problem signature, drill into individual queries that make up the signature. Monitor average CPU, average reads, average writes, average duration, and number of occurrences. Refer also to the documentation View the query monitor signature mode and Advanced query signature view.



☑ Filters
☑ List
☐ Group By Box

Filters

Date Range - Begin: 10/ 6/2014 End: 10/ 7/2014 Time Period - Begin: 12:00:00 AM End: 11:59:59 PM

Application: User: Exclude SQL Text:

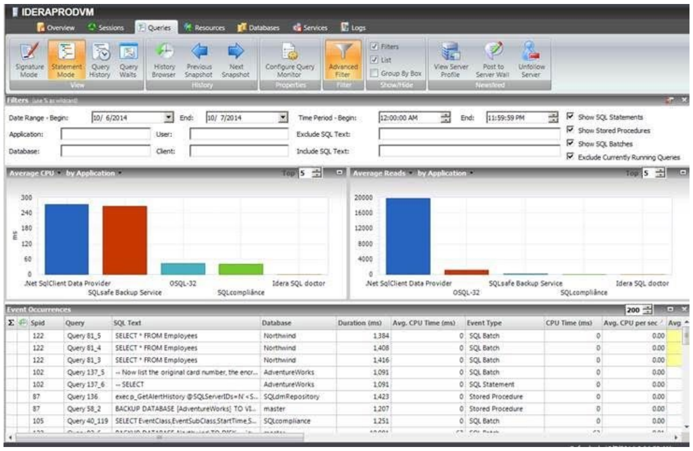
Database: Client: Include SQL Text:

☑ Show SQL Statements
☑ Show Stored Procedures
☑ Show SQL Batches
☑ Exclude Currently Running Queries

Query	SQL Text	Occurrences	Event Type	Avg. Duration (ms)	Avg. CPU Time (ms)	CPU Time (ms)	Avg. CPU per sec	Avg. Reads
Query 81_b	SELECT * FROM Employees	3	SQL Batch	1,403	0	0	0.00	3,942
Query 136	exec p_GetAlertHistory @SQLServerIDs=#,@...	1	Stored Procedure	1,423	0	0	0.00	0
Query 82_c	BACKUP DATABASE Northwind TO DISK = #	2	SQL Batch	11,021	102	140	0.01	491
Query 137	-- Now list the original card number, the encr...	3	SQL Batch	1,158	15	31	0.01	189
Query 58_b	BACKUP DATABASE [AdventureWorks] TO VI...	2	Stored Procedure	1,246	8	15	0.01	201
Query 82_b	BACKUP DATABASE Northwind TO DISK = #	1	SQL Batch	11,984	204	204	0.02	209
Query 40_b	SELECT EventClass,EventSubClass,StartTime,S...	5	SQL Batch	1,415	40	62	0.03	0
Query 60_b	BACKUP DATABASE [AdventureWorks2012] T...	2	Stored Procedure	1,073	32	47	0.03	243

Statement Mode View (Specific Performance)

View individual SQL statements or view query signatures. Query statements are presented exactly as the query monitor trace collects them. Query statements provide all of the detail needed to diagnose a specific problem with a query. After identifying a potential problem statement, drill into the Query Details view. Monitor individual CPU time, individual reads, individual writes, and individual execution duration, as well as user database and application. Refer also to the documentation View the query monitor statement mode and Advanced query statement view.



Filters (use % as wildcard)

Date Range - Begin: 10/ 6/2014 End: 10/ 7/2014 Time Period - Begin: 12:00:00 AM End: 11:59:59 PM
Application: User: Exclude SQL Text:
Database: Client: Include SQL Text:

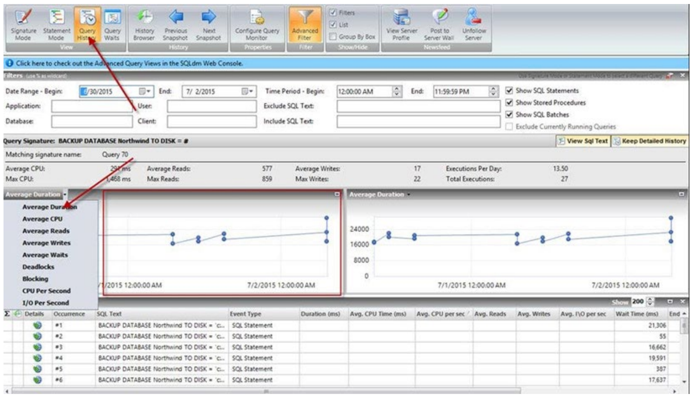
☑ Filters
☑ List
☐ Group By Box

☑ Show SQL Statements
☑ Show Stored Procedures
☑ Show SQL Batches
☑ Exclude Currently Running Queries

Spid	Query	SQL Text	Database	Duration (ms)	Avg. CPU Time (ms)	Event Type	CPU Time (ms)	Avg. CPU per sec	Avg
122	Query 81_5	SELECT * FROM Employees	Northwind	1,384	0	SQL Batch	0	0.00	
122	Query 81_4	SELECT * FROM Employees	Northwind	1,408	0	SQL Batch	0	0.00	
122	Query 81_3	SELECT * FROM Employees	Northwind	1,416	0	SQL Batch	0	0.00	
102	Query 137_5	-- Now list the original card number, the encr...	AdventureWorks	1,091	0	SQL Batch	0	0.00	
102	Query 137_6	-- SELECT	AdventureWorks	1,091	0	SQL Statement	0	0.00	
87	Query 136	exec p_GetAlertHistory @SQLServerIDs=N'<S...	SQLdmRepository	1,423	0	Stored Procedure	0	0.00	
87	Query 58_2	BACKUP DATABASE [AdventureWorks] TO VI...	master	1,207	0	Stored Procedure	0	0.00	
105	Query 40_119	SELECT EventClass,EventSubClass,StartTime,S...	SQLcompliance	1,251	0	SQL Batch	0	0.00	

Query History View

Measure the daily historical performance impact based on number of occurrences throughout the day for every day. The history includes the durations, amount of CPU time each day, and the level of reads and writes. Understand historical trends for the selected query performance and how code changes may have improved performance into the future. Refer also to the documentation View the query history.



Click here to check out the Advanced Query Views in the SQLdm Web Console.

Filters (use % as wildcard)

Date Range - Begin: /30/2015 End: 7/ 2/2015 Time Period - Begin: 12:00:00 AM End: 11:59:59 PM
Application: User: Exclude SQL Text:
Database: Client: Include SQL Text:

☑ Filters
☑ List
☐ Group By Box

☑ Show SQL Statements
☑ Show Stored Procedures
☑ Show SQL Batches
☐ Exclude Currently Running Queries

Query Signature: BACKUP DATABASE Northwind TO DISK = #

View Sql Text | Keep Detailed History

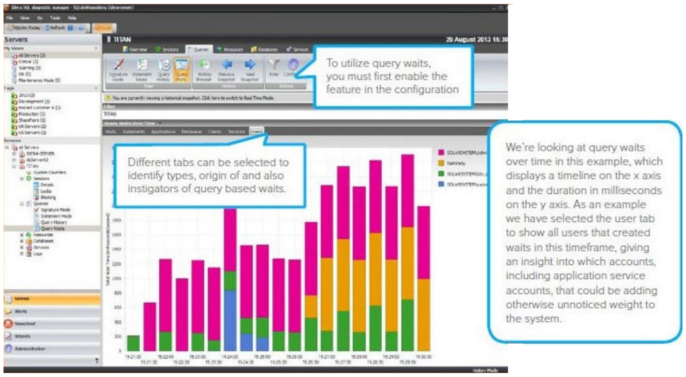
Matching signature name: Query 70

Average CPU: 29 ms Average Reads: 577 Average Writes: 17 Executions Per Day: 13.50
Max CPU: 1,468 ms Max Reads: 859 Max Writes: 22 Total Executions: 27

Details	Occurrence	SQL Text	Event Type	Duration (ms)	Avg. CPU Time (ms)	Avg. CPU per sec	Avg. Reads	Avg. Writes	Avg. I\O per sec	Wait Time (ms)	End
	#1	BACKUP DATABASE Northwind TO DISK = 'c...	SQL Statement							21,306	
	#2	BACKUP DATABASE Northwind TO DISK = 'c...	SQL Statement							55	
	#3	BACKUP DATABASE Northwind TO DISK = 'c...	SQL Statement							16,662	
	#4	BACKUP DATABASE Northwind TO DISK = 'c...	SQL Statement							19,591	
	#5	BACKUP DATABASE Northwind TO DISK = 'c...	SQL Statement							387	
	#6	BACKUP DATABASE Northwind TO DISK = 'c...	SQL Statement							17,637	

Query Waits View

Analyze waits over time and by duration to locate the top bottlenecks and what changes may potentially have the biggest performance boost on the SQL Server instance. Display a graphical dual view of query wait statistics to see an impact analysis of waits historically and to perform a real-time assessment of existing query activity and associated waits. Use the history browser in



You are currently viewing a historical snapshot. Click here to switch to Real Time Mode.

To utilize query waits, you must first enable the feature in the configuration

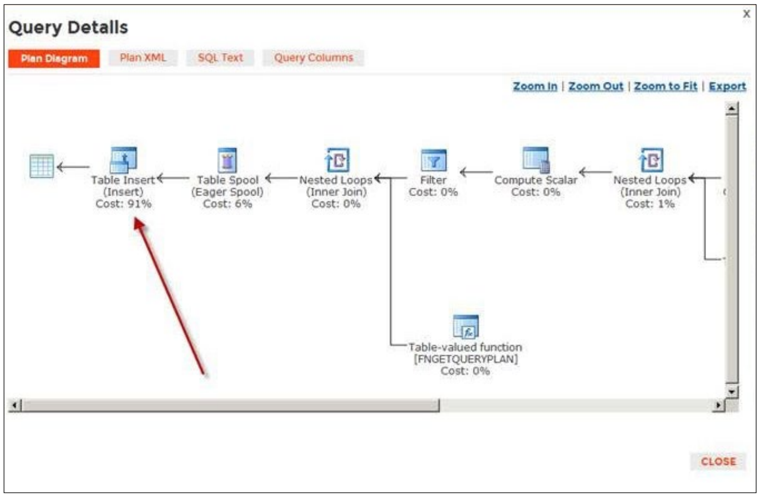
Different tabs can be selected to identify types, origin of and also instigators of query based waits.

We're looking at query waits over time in this example, which displays a timeline on the x axis and the duration in milliseconds on the y axis. As an example we have selected the user tab to show all users that created waits in this timeframe, giving an insight into which accounts, including application service accounts, that could be adding otherwise unnoticed weight to the system.

conjunction with wait stats for a very granular level of root-cause analysis when identifying performance bottlenecks in the past. Refer also to the documentation View query waits and View your SQL Server query waits information.

Execution Plan

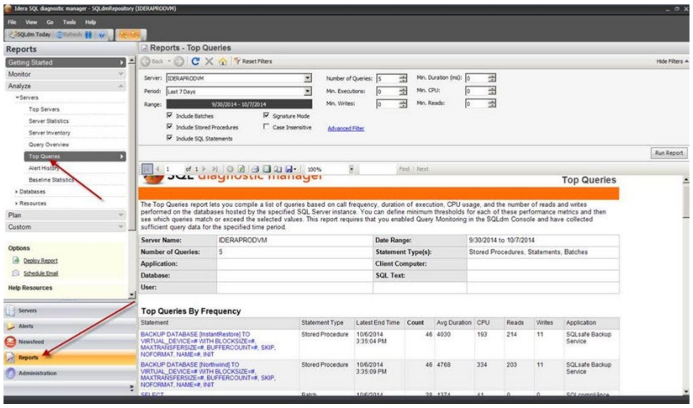
The execution plan assesses how queries perform and where to improve the code. The execution plan diagram displays the query execution plan (actual or estimated). The diagram shows the tree of operations that make up a query. This tree shows individual operation nodes and the pertaining graphical execution plan icon, along with basic information such as operator name and operation percentage of total cost. The execution plan also shows the referenced tables and columns to understand where a potential index adjustment may improve performance. Refer also to the documentation Query Details view.



Query Details
x
Plan Diagram
Plan XML
SQL Text
Query Columns
Zoom In | Zoom Out | Zoom to Fit | Export
Table Insert (Insert) Cost: 91%
Table Spool (Eager Spool) Cost: 6%
Nested Loops (Inner Join) Cost: 0%
Filter Cost: 0%
Compute Scalar Cost: 0%
Nested Loops (Inner Join) Cost: 1%
Table-valued function [FNGETQUERYPLAN] Cost: 0%
CLOSE

Top Queries Report

The Top Queries report compiles a list of queries based on call frequency, duration of execution, CPU usage, and the number of reads and writes performed on the databases hosted by the specified SQL Server instance. Define minimum thresholds for each of these performance metrics and then see which queries match or exceed the selected values. Report on the worst queries within the report interface. Apply some report filters such as filtering by user, database, application, host name, and by different performance criteria (such as CPU, memory, reads, and writes). Refer also to the documentation Top Queries server analysis report.



Server: IDERAPRODVM
Period: Last 7 Days
Range: 9/30/2014 - 10/7/2014
☑ Include Batches
☑ Include Stored Procedures
☑ Include SQL Statements
☑ Signature Mode
☐ Case Insensitive
Advanced Filter
Number of Queries: 5
Min. Executions: 0
Min. Writes: 0
Min. Duration (ms): 0
Min. CPU: 0
Min. Reads: 0

SQL diagnostic manager

Top Queries

The Top Queries report lets you compile a list of queries based on call frequency, duration of execution, CPU usage, and the number of reads and writes performed on the databases hosted by the specified SQL Server instance. You can define minimum thresholds for each of these performance metrics and then see which queries match or exceed the selected values. This report requires that you enabled Query Monitoring in the SQLdm Console and have collected sufficient query data for the specified time period.

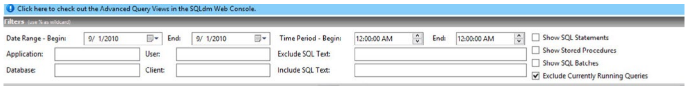
Server Name:	IDERAPRODVM	Date Range:	9/30/2014 to 10/7/2014
Number of Queries:	5	Statement Type(s):	Stored Procedures, Statements, Batches
Application:		Client Computer:	
Database:		SQL Text:	
User:			

Top Queries By Frequency

Statement	Statement Type	Latest End Time	Count	Avg Duration	CPU	Reads	Writes	Application
BACKUP DATABASE [InstantRestore] TO VIRTUAL_DEVICE=# WITH BLOCKSIZE=#, MAXTRANSFERSIZE=#, BUFFERCOUNT=#, SKIP, NOFORMAT, NAME=#, INIT	Stored Procedure	10/6/2014 3:35:04 PM	46	4030	193	214	11	SQLsafe Backup Service
BACKUP DATABASE [Northwind] TO VIRTUAL_DEVICE=# WITH BLOCKSIZE=#, MAXTRANSFERSIZE=#, BUFFERCOUNT=#, SKIP, NOFORMAT, NAME=#, INIT	Stored Procedure	10/6/2014 3:35:09 PM	46	4768	334	203	11	SQLsafe Backup Service
SELECT	Batch	10/6/2014	28	1374	41	0	0	SQLcompliance

Queries Tab Filtering

On each Query tab, the filtering capabilities provide an option for focus on specific queries relevant to the performance. Include and exclude specific applications, databases, users, clients, SQL text, and more via advanced filters. Refer also to the documentation View your SQL Server queries information.



Click here to check out the Advanced Query Views in the SQLdm Web Console.

Filters (use % as wildcard)

Date Range - Begin: 9/ 1/2010 End: 9/ 1/2010 Time Period - Begin: 12:00:00 AM End: 12:00:00 AM

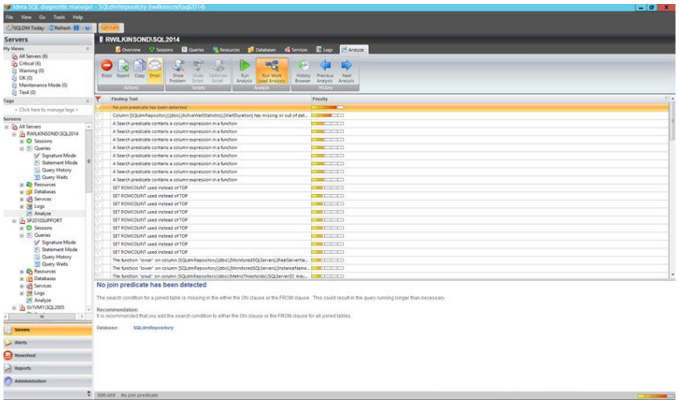
Application: User: Exclude SQL Text:

Database: Client: Include SQL Text:

□ Show SQL Statements
□ Show Stored Procedures
□ Show SQL Batches
☑ Exclude Currently Running Queries

Prescriptive Workload Analysis

Run a prescriptive analysis on a specific SQL Server instance to identify and resolve SQL Server performance problems. The analysis engine scans the SQL Server configuration for potential problems and the health of the databases, resulting in a useful set of recommendations for improving performance. Prescriptive analysis targets some of the most common areas of SQL Server performance problems. Instead, workload analysis targets the performance categories for index optimization and query optimization. Workload analysis provides recommendations for these two categories that use a high amount of performance resources when running. Refer also to the documentation Run a workload analysis on your SQL Server.



Finding Text	Priority
No join predicate has been detected	
Column [SQLdmRepository].[dbo].[ActiveWaitStatistics].[WaitDuration] has missing or out of dat...	
A Search predicate contains a column expression in a function	
A Search predicate contains a column expression in a function	
A Search predicate contains a column expression in a function	
A Search predicate contains a column expression in a function	
A Search predicate contains a column expression in a function	
A Search predicate contains a column expression in a function	
A Search predicate contains a column expression in a function	
A Search predicate contains a column expression in a function	
SET ROWCOUNT used instead of TOP	
SET ROWCOUNT used instead of TOP	
SET ROWCOUNT used instead of TOP	
SET ROWCOUNT used instead of TOP	
SET ROWCOUNT used instead of TOP	
SET ROWCOUNT used instead of TOP	
SET ROWCOUNT used instead of TOP	
SET ROWCOUNT used instead of TOP	
SET ROWCOUNT used instead of TOP	
The function "lower" on column [SQLdmRepository].[dbo].[MonitoredSQLServers].[RealServerNa...	
The function "lower" on column [SQLdmRepository].[dbo].[MonitoredSQLServers].[InstanceName...	
The function "isnull" on column [SQLdmRepository].[dbo].[MetricThresholds].[SQLServerID] mau...	

No join predicate has been detected

The search condition for a joined table is missing in the either the ON clause or the FROM clause. This could result in the query running longer than necessary.

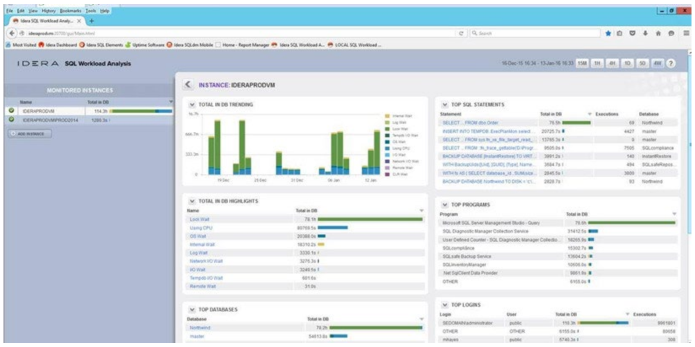
Recommendation:
It is recommended that you add the search condition to either the ON clause or the FROM clause for all joined tables.

Database: SQLdmRepository

SDR-Q19 No join predicate

SQL Workload Analysis Add-On

The SQL Workload Analysis add-on provides granular wait state monitoring, continuous SQL sampling, intuitive drill down to view top activity, query plan tuning and recommendations, lock and latch resolutions, and storage visibility and contention resolution. Identify, isolate, and resolve tough performance issues with specific SQL transactions or workloads in just a few mouse clicks. Refer also to the documentation Launch SQL Workload Analysis and Welcome to SQL Workload Analysis.



TOP SQL STATEMENTS

Statement	Total in DB	Executions	Database
SELECT ... FROM dbo.Order	76.5h	69	Northwind
INSERT INTO TEMPDB..ExecPlanMon select ...	20725.7s	4427	master
SELECT ... FROM sys.fn_xe_file_target_read_...	13765.3s	0	master
SELECT ... FROM ::fn_trace_gettable(D:\Progr...	9505.0s	7505	SQLcompliance
BACKUP DATABASE [InstantRestore] TO VIRT...	3991.2s	540	InstantRestore
WITH BackupUids([Uid], [GUID], [Type], Name...	3684.7s	494	SQLsafeRepos...
WITH fs AS (SELECT database_id, SUM(size...	2845.5s	3800	master
BACKUP DATABASE Northwind TO DISK = 'c:\...	2828.7s	93	Northwind

TOTAL IN DB HIGHLIGHTS

Name	Total in DB
Lock Wait	78.1h
Using CPU	80769.5s
OS Wait	20388.0s
Internal Wait	18310.2s
Log Wait	3330.1s
Network I/O Wait	3275.3s
I/O Wait	3249.5s
Tempdb I/O Wait	601.6s
Remote Wait	31.0s

TOP PROGRAMS

Program	Total in DB
Microsoft SQL Server Management Studio - Query	78.6h
SQL Diagnostic Manager Collection Service	31412.5s
User Defined Counter - SQL Diagnostic Manager Collectio...	18265.9s
SQLcompliance	15302.7s
SQLsafe Backup Service	13604.2s
SQLInventoryManager	10606.0s
.Net SqlClient Data Provider	9861.9s
OTHER	6155.0s

TOP DATABASES

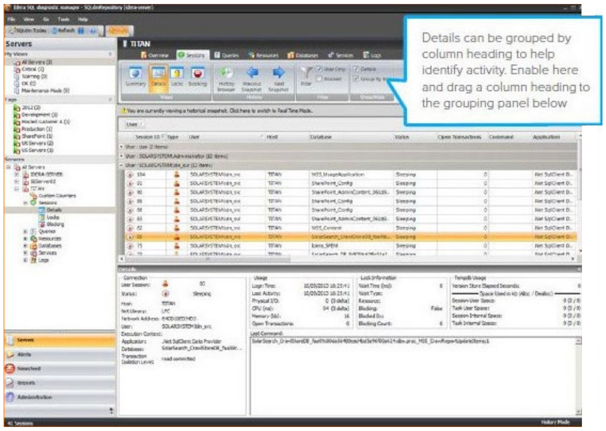
Database	Total in DB
Northwind	78.2h
master	54613.8s

TOP LOGINS

Login	User	Total in DB	Executions
SEDOMAIN\administrator	public	110.3h	9961801
OTHER	OTHER	6155.0s	80658
mhayes	public	5740.3s	308

View Session Detail Analytics

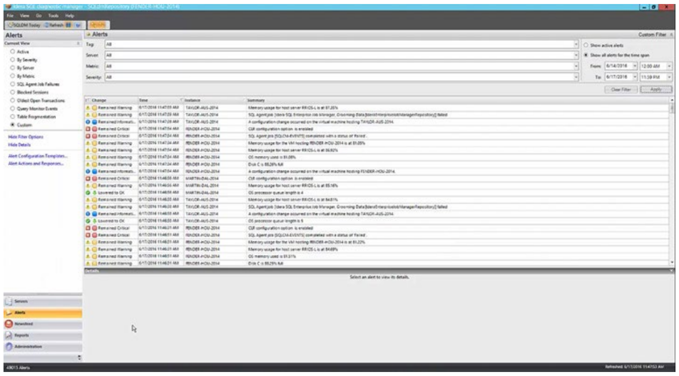
The Sessions Details view provides an in-depth analysis of sessions running on monitored SQL Server instances. View a broad range of information from performance details to open transactions and configured options. Track process activity at the statement level. Individual sessions appear in the top portion of the window. Right-click any session in the list to view locks, show the query history, trace the session, kill the session, print the associated statistics, and export statistics to Excel. Cross-reference information gathered from the Query Waits view to build a full and forensic picture of root cause analysis. Refer also to the documentation Get sessions performance details.



Details can be grouped by column heading to help identify activity. Enable here and drag a column heading to the grouping panel below

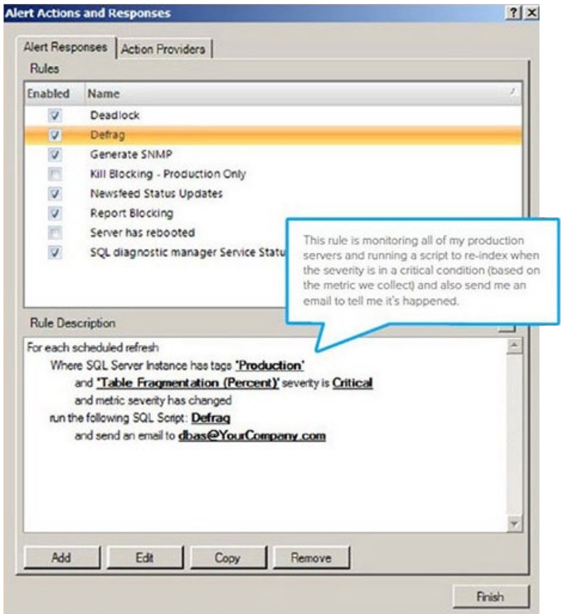
Alert Predictively

Configure alerts to inform and warn about approaching issues with SQL Server instances. See the alerts using the desktop and web consoles, the IDERA Newsfeed, and the mobile console. After correcting the situation triggering the alert, send alerts again if the situation recurs. Highlight columns containing associated alerts with their status color based on configurable thresholds. Identify trends in past alert activity and recognize the likelihood of the events happening going forward. Discover past trends in alert behavior and determine the percentage of likelihood that these same events may pop up at particular times later in time. Be better prepared to respond instead of reacting with surprise to alerts. Refer also to the documentation Alert on SQL Server Metrics.



Automate Alert Actions and Responses

When reaching an alert threshold, automatically send an email notification, display an alert message in the Windows taskbar, write an event to the Windows Event Log, generate an event on the timeline, and execute a SQL or PowerShell script. Be informed very quickly to issues of health and performance, and implement steps to minimize and resolve issues and problems SQL Diagnostic Manager finds automatically. View a list of current rules used to assist in managing the SQL Server environment. Add new rules and modify existing rules. Generate multiple alerts for different levels of severity, metric, and time frame. Alert and report on custom counters. The breadth of alerting and management capabilities is vast. Refer also to the documentation Configure how SQL Diagnostic Manager responds to alerts.



Alert Actions and Responses

Rules

Enabled	Name
☑	Deadlock
☑	Defrag
☑	Generate SNMP
□	Kill Blocking - Production Only
☑	Newsfeed Status Updates
☑	Report Blocking
□	Server has rebooted
☑	SQL diagnostic manager Service Statu

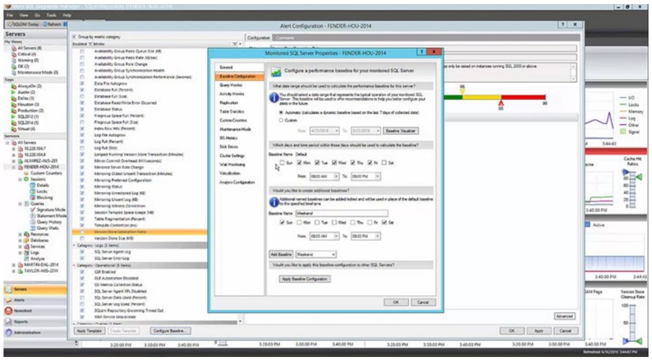
This rule is monitoring all of my production servers and running a script to re-index when the severity is in a critical condition (based on the metric we collect) and also send me an email to tell me it's happened.

Rule Description

For each scheduled refresh
Where SQL Server Instance has tags 'Production'
and 'Table Fragmentation (Percent)' severity is Critical
and metric severity has changed
run the following SQL Script: Defrag
and send an email to dbas@YourCompany.com

Set Up Server Baselines

Adjust metric thresholds based on the baseline fluctuations throughout the day. Calculate each baseline out of a pool of collected data based on the selected period and collection interval. Use each baseline to provide alert recommendations to set effective alert thresholds. Visualize times of high and low activity to establish effective baseline periods. Enable alerting and recommendations based on baseline violations. Define and schedule multiple baselines per server. Quickly apply baselines to more than one instance. View the relevant baseline in effect in various metric graphs.



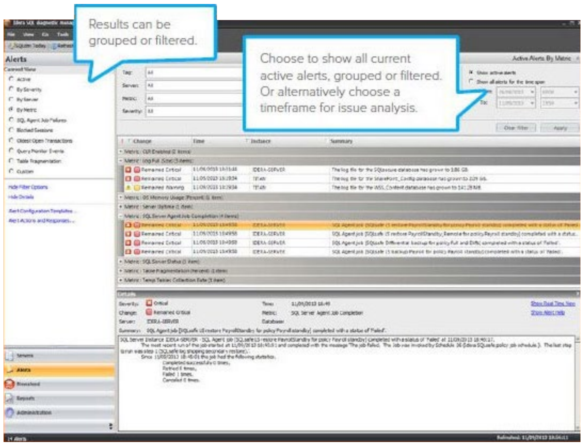
Alert Configuration - FENDER-HOU-2014
Monitored SQL Server Properties - FENDER-HOU-2014
Configure a performance baseline for your monitored SQL Server.
OK
Cancel

Create multiple baselines for multiple activity windows (such as logins, SSRS report executions, and job activities) for the best baseline accuracy. Metric thresholds change dynamically based on the baseline fluctuations throughout the day, to alert only when current performance jumps excessively outside of normal. The baseline visualizer displays metrics to determine the hot spots during the day, to understand what may be the appropriate time windows to create within the multiple baseline functionality.

Refer also to the documentation Configure server baseline options and Baseline Statistics server analysis report.

Aalyze Alert Patterns

Locate alerts generated over a period. Group and filter the results. Quickly identify and triage alerts and issues before they escalate. Pinpoint groups of alerts that happen over a period. Refer also to the documentation Organize alerts.

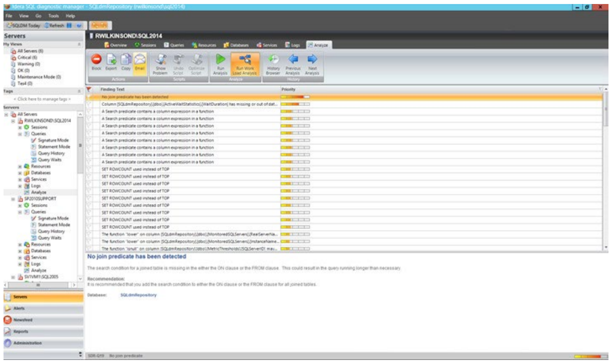


Results can be grouped or filtered.

Choose to show all current active alerts, grouped or filtered. Or alternatively choose a timeframe for issue analysis.

Analyze Prescriptively with Expert Recommendations

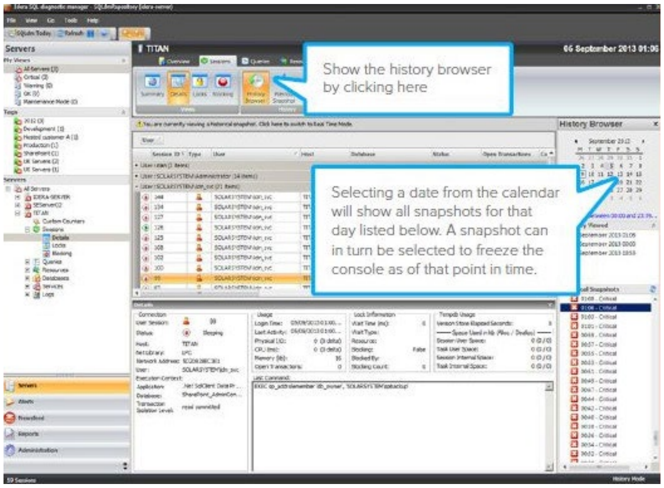
Run a prescriptive analysis on a particular instance to identify and resolve potential SQL Server performance problems. The prescriptive analysis provides quick and minimally invasive analytics of SQL Server instances. The analysis results in a useful set of expert recommendations for improving performance including executable scripts. Prescriptive analysis targets some of the most common areas of SQL Server performance problems, such as workloads, missing indexes, redundant indexes, poor queries, wait states, server configuration, security, database objects, memory, and



more. Execute the prescriptive analysis manually, scheduled to run, or launched in response to an alert. Refer also to the documentation Run a prescriptive analysis on your SQL Server.

View Past Performance

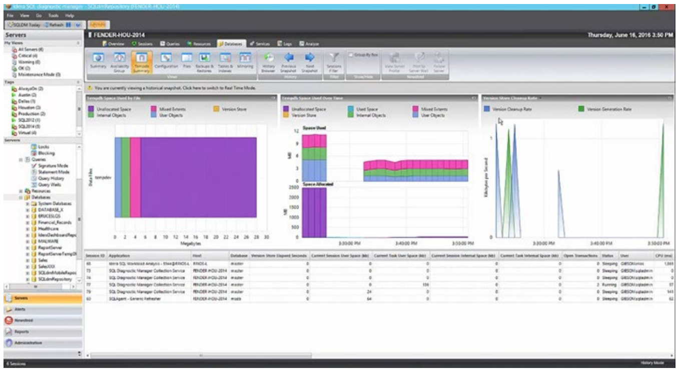
The history browser provides information about the state of SQL Server instances at the time of a standard snapshot. Use this information to diagnose and resolve issues to keep the issue from happening again. Select a historical snapshot collected by the standard refresh and view the collected data using almost all of the real-time views. After choosing a snapshot, replay the data as of that time in the console to select other tabs and view data from all over the tool to correlate issues and fully diagnose problems that have passed. Refer also to the documentation View past performance.


Show the history browser by clicking here

Selecting a date from the calendar will show all snapshots for that day listed below. A snapshot can in turn be selected to freeze the console as of that point in time.

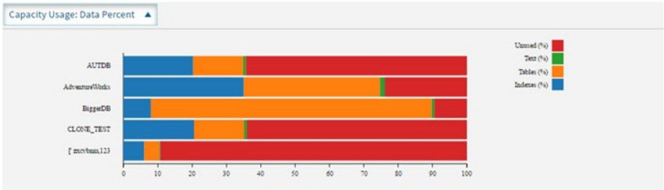
Monitor Tempdb

Identify and resolve contention and performance issues with the tempdb system database. Monitor and view the tempdb space and performance related information, including tempdb file space, tempdb version store, tempdb sessions space usage, and waits related to tempdb. See the current capacity usage and recent trends of the files over time. Display a list of sessions currently using tempdb along with their cumulative usage and tempdb space. Refer also to the documentation Tempdb panel, Get the tempdb status summary, and Tempdb Statistics database analysis report.

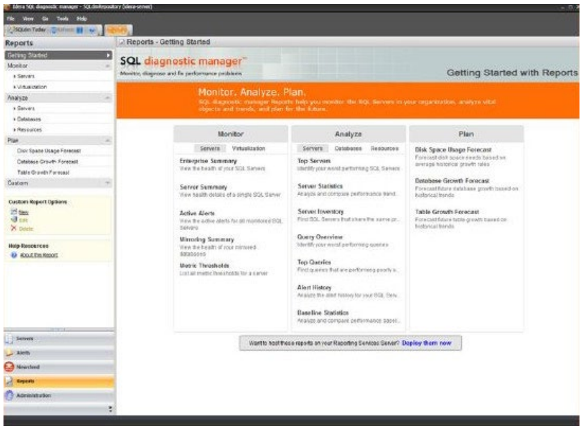


Plan Future Capacity

The Capacity Usage chart in the Summary view of the Database tab displays the usage of data and logs in the databases. This chart provides information on the text, tables, and indexes space, as well as the unused space of the databases. Refer also to the documentation View your SQL Server databases information.



The Baseline Statistics report analyzes and compares baselines within a single SQL Server instance and across two instances. When viewing baseline statistics for a monitored SQL Server instance, compare the baseline metric values at two different times or two different metrics at the same time. Include another instance and compare baselines values occurring at the same time or different times. View trends in the average value of a metric for a SQL Server instance and how this value changes over time for capacity planning.



Reports - Getting Started

SQL diagnostic manager™
Monitor, diagnose and fix performance problems

Getting Started with Reports

Monitor. Analyze. Plan.
SQL diagnostic manager Reports help you monitor the SQL Servers in your organization, analyze vital objects and trends, and plan for the future.

Monitor — Servers | Virtualization
- Enterprise Summary: View the health of your SQL Servers
- Server Summary: View health details of a single SQL Server
- Active Alerts: View the active alerts for all monitored SQL Servers
- Mirroring Summary: View the health of your mirrored databases
- Metric Thresholds: List all metric thresholds for a server

Analyze — Servers | Databases | Resources
- Top Servers: Identify your worst performing SQL Servers
- Server Statistics: Analyze and compare performance trend...
- Server Inventory: Find SQL Servers that share the same pr...
- Query Overview: Identify your worst performing queries
- Top Queries: Find queries that are performing poorly a...
- Alert History: Analyze the alert history for your SQL Serv...
- Baseline Statistics: Analyze and compare performance basel...

Plan
- Disk Space Usage Forecast: Forecast disk space needs based on average historical growth rates
- Database Growth Forecast: Forecast future database growth based on historical trends
- Table Growth Forecast: Forecast future table growth based on historical trends

Want to host these reports on your Reporting Services Server? Deploy them now

Plan reports to forecast for future needs based on historical growth trends. The Disk Space Usage Forecast report forecasts disk space needs. The Database Growth Forecast report forecasts future database growth. The Table Growth Forecast report forecasts future table growth.

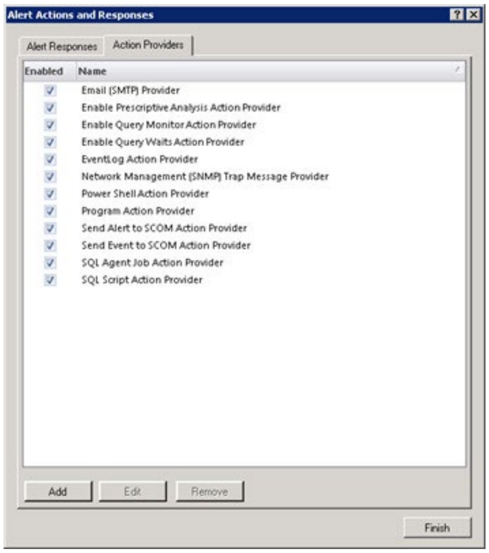
Refer also to the documentation Baseline Statistics server analysis report and Plan reports.

Connect to Other Tools

Connect SQL Diagnostic Manager to other tools. For example, provide in-depth granular diagnostic information to generic management tools currently on the market.

Send Alerts to Other Systems

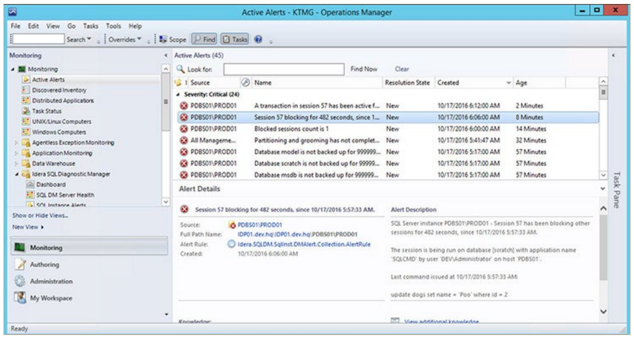
Set up simple network management protocol (SNMP) to send out alert events to other network management systems. SQL Diagnostic Manager includes a management information base (MIB) to use in another network management system to format the events sent by SQL Diagnostic Manager. Add, modify, import, export, and configure alert responses via the Alert Actions and Responses window to send out alert notifications using the email server available on the network with simple mail transfer protocol (SMTP) to issue tracking applications and workflow engines for intelligent routing and automated interaction. Refer also to the documentation Configure network management settings, Understanding the IDERA MIB, and Configure how SQL Diagnostic Manager responds to alerts.



Enabled	Name
☑	Email (SMTP) Provider
☑	Enable Prescriptive Analysis Action Provider
☑	Enable Query Monitor Action Provider
☑	Enable Query Waits Action Provider
☑	EventLog Action Provider
☑	Network Management (SNMP) Trap Message Provider
☑	Power Shell Action Provider
☑	Program Action Provider
☑	Send Alert to SCOM Action Provider
☑	Send Event to SCOM Action Provider
☑	SQL Agent Job Action Provider
☑	SQL Script Action Provider

Integrate with SCOM

Integrate the detailed alert and status information gathered by SQL Diagnostic Manager into the Microsoft System Center Operations Manager (SCOM) environment. Provide a stream of live updates to IT operators on the status of SQL Server instances being monitored by SQL Diagnostic Manager. Extend management information from SQL Diagnostic Manager to a new custom node within SCOM. Automatic discovery of installations of SQL Diagnostic Manager and of the managed instances from



Source	Name	Resolution State	Created	Age
Severity: Critical (24)				
PDBS01\PROD01	A transaction in session 57 has been active f...	New	10/17/2016 6:12:00 AM	2 Minutes
PDBS01\PROD01	Session 57 blocking for 482 seconds, since 1...	New	10/17/2016 6:06:00 AM	8 Minutes
PDBS01\PROD01	Blocked sessions count is 1	New	10/17/2016 6:00:00 AM	14 Minutes
All Manageme...	Partitioning and grooming has not complet...	New	10/17/2016 5:41:47 AM	32 Minutes
PDBS01\PROD01	Database model is not backed up for 999999...	New	10/17/2016 5:17:00 AM	57 Minutes
PDBS01\PROD01	Database scratch is not backed up for 99999...	New	10/17/2016 5:17:00 AM	57 Minutes
PDBS01\PROD01	Database msdb is not backed up for 999999...	New	10/17/2016 5:17:00 AM	57 Minutes

Session 57 blocking for 482 seconds, since 10/17/2016 5:57:33 AM.

Source: PDBS01\PROD01
Full Path Name: IDP01.dev.hq\IDP01.dev.hq\PDBS01\PROD01
Alert Rule: Idera.SQLDM.SqlInst.DMAlert.Collection.AlertRule
Created: 10/17/2016 6:06:00 AM

Alert Description

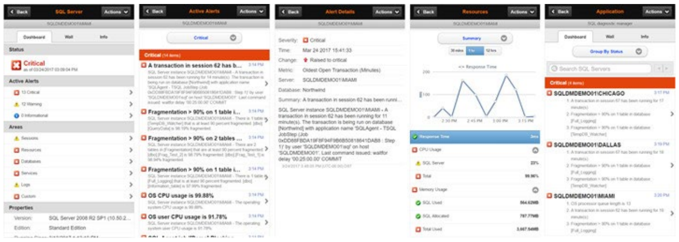
SQL Server instance PDBS01\PROD01 - Session 57 has been blocking other sessions for 482 seconds, since 10/17/2016 5:57:33 AM.

The session is being run on database [scratch] with application name 'SQLCMD' by user 'DEV\Administrator' on host 'PDBS01'.

Last command issued at 10/17/2016 5:57:33 AM:

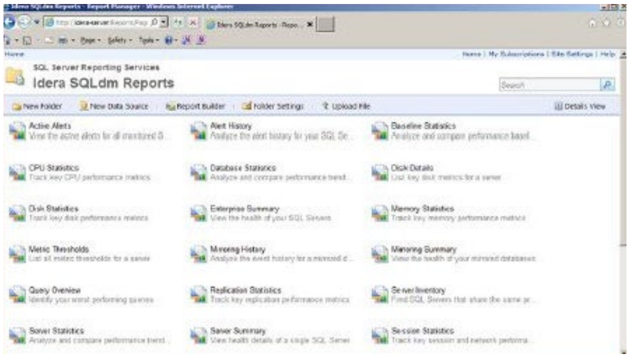
update dogs set name = 'Poo' where id = 2 those installations. Once discovered, propagate the status, health, and events of these instances up to SCOM. Refer also to the documentation Integrate SQL Diagnostic Manager with SCOM.

Access Mobile Console



The mobile console is a web-based application that displays SQL Server performance on a variety of mobile devices (including iPhone, iPad, Android, and Blackberry). The mobile console includes the IDERA Newsfeed that is an effective way for database administrators and managers to collaborate, share knowledge, and keep close tabs on the most critical SQL Server issues as they unfold. View real-time dashboards to identify problem areas and get current server health. Drill down into the details of the main performance metrics and alerts. See the entire SQL Server environment, and stay productive and connected from anywhere. Refer also to the documentation Navigate the IDERA Newsfeed and SQLDM Mobile.

Deploy Web-Based Reports via SSRS



SQL Server Reporting Services

Idera SQLdm Reports

Active Alerts
View the active alerts for all monitored S...

Alert History
Analyze the alert history for your SQL Se...

Baseline Statistics
Analyze and compare performance baseli...

CPU Statistics
Track key CPU performance metrics

Database Statistics
Analyze and compare performance trend...

Disk Details
List key disk metrics for a server

Disk Statistics
Track key disk performance metrics

Enterprise Summary
View the health of your SQL Servers

Memory Statistics
Track key memory performance metrics

Metric Thresholds
List all metric thresholds for a server

Mirroring History
Analyze the event history for a mirrored d...

Mirroring Summary
View the health of your mirrored databases

Query Overview
Identify your worst performing queries

Replication Statistics
Track key replication performance metrics

Server Inventory
Find SQL Servers that share the same pr...

Server Statistics
Analyze and compare performance trend...

Server Summary
View health details of a single SQL Server

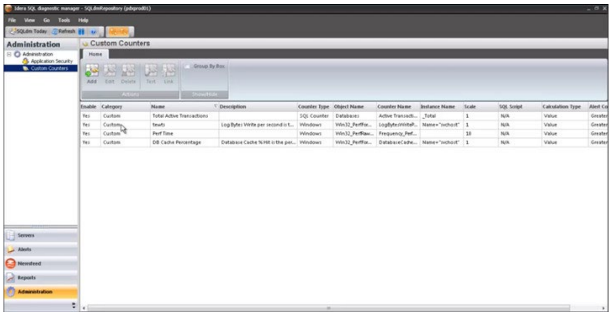
Session Statistics
Track key session and network performa...

In addition to the included reports, deploy web-based custom reports using SQL Server Reporting Services (SSRS) for a comprehensive auditing solution. Access these web-based reports to analyze current and historical performance and statistical data. Refer also to the documentation Deploy SQL Diagnostic Manager reports to Reporting Services.

Adapt to Specific Needs

Setup Custom Counters

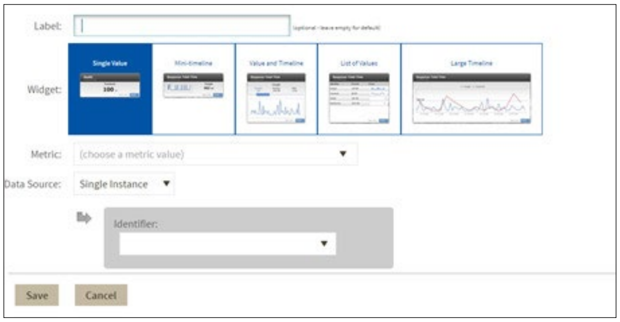
In addition to monitoring a wide variety of the most common SQL Server and operating system performance metrics, add additional performance metrics via custom counters. Add any Windows system counters including any Performance Monitor and Windows Management Instrumentation (WMI) counters. Also, add any SQL Server system counter stored in the sysperfinfo system table, and any numerical value that custom SQL scripts return. Also, add and any performance counters accessible through the virtual machine and host server. Refer also to the documentation Use custom counters to track metrics.



Enable	Category	Name	Description	Counter Type	Object Name	Counter Name	Instance Name	Scale	SQL Script	Calculation Type	Alert Co
Yes	Custom	Total Active Transactions		SQL Counter	Databases	Active Transacti...	_Total	1	N/A	Value	Greater
Yes	Custom	tewts	Log Bytes Write per second is t...	Windows	Win32_PerfFor...	LogBytesWriteP...	Name="svchost"	1	N/A	Value	Greater
Yes	Custom	Perf Time		Windows	Win32_PerfRaw...	Frequency_Perf...		10	N/A	Value	Greater
Yes	Custom	DB Cache Percentage	Database Cache % Hit is the per...	Windows	Win32_PerfFor...	DatabaseCache...	Name="svchost"	1	N/A	Value	Greater

Create Custom Dashboards

Customize the dashboard per monitored SQL Server instance by selecting the panels that are the most important to display in each instance. Define and save multiple dashboards to compare metrics across monitored SQL Server instances. Customize the dashboard for the selected SQL Server instance or as the default for all added SQL Server instances. Refer also to the documentation Create custom dashboards.



Label: (optional - leave empty for default)

Widget: Single Value

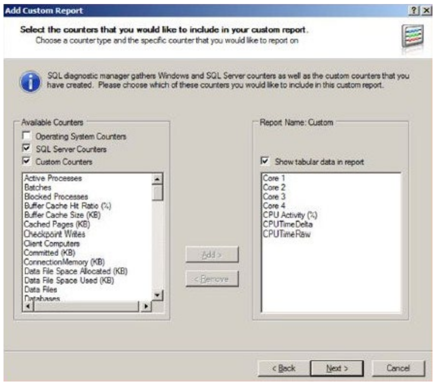
Metric: (choose a metric value)

Data Source: Single Instance

Identifier:

Build Custom Reports

Create or edit custom reports. Include in custom reports any collected metric, including custom counters. Choose the counters to include in a report, order the way the metrics appear, and specify the aggregation method used on each of the metrics. Refer also to the documentation Custom reports.



Select the counters that you would like to include in your custom report.
Choose a counter type and the specific counter that you would like to report on

SQL diagnostic manager gathers Windows and SQL Server counters as well as the custom counters that you have created. Please choose which of these counters you would like to include in this custom report.

Available Counters

☐ Operating System Counters
☑ SQL Server Counters
☑ Custom Counters

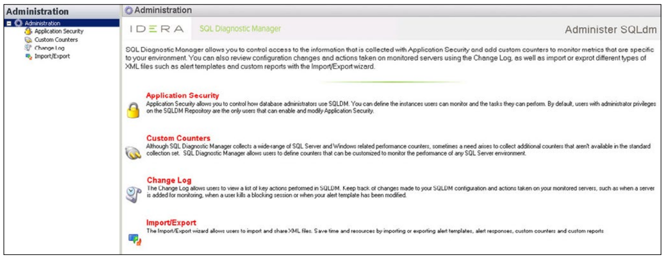
Active Processes
Batches
Blocked Processes
Buffer Cache Hit Ratio (%)
Buffer Cache Size (KB)
Cached Pages (KB)
Checkpoint Writes
Client Computers
Committed (KB)
ConnectionMemory (KB)
Data File Space Allocated (KB)
Data File Space Used (KB)
Data Files
Databases

Report Name: Custom

☑ Show tabular data in report

Core 1
Core 2
Core 3
Core 4
CPU Activity (%)
CPUTimeDelta
CPUTimeRaw

Audit Activity



Administration

- Administration
  - Application Security
  - Custom Counters
  - Change Log
  - Import/Export

Administration

IDERA SQL Diagnostic Manager Administer SQLdm

SQL Diagnostic Manager allows you to control access to the information that is collected with Application Security and add custom counters to monitor metrics that are specific to your environment. You can also review configuration changes and actions taken on monitored servers using the Change Log, as well as import or exprot different types of XML files such as alert templates and custom reports with the Import/Export wizard.

Application Security
Application Security allows you to control how database administrators use SQLDM. You can define the instances users can monitor and the tasks they can perform. By default, users with administrator privileges on the SQLDM Repository are the only users that can enable and modify Application Security.

Custom Counters
Although SQL Diagnostic Manager collects a wide-range of SQL Server and Windows related performance counters, sometimes a need arises to collect additional counters that aren't available in the standard collection set. SQL Diagnostic Manager allows users to define counters that can be customized to monitor the performance of any SQL Server environment.

Change Log
The Change Log allows users to view a list of key actions performed in SQLDM. Keep track of changes made to your SQLDM configuration and actions taken on your monitored servers, such as when a server is added for monitoring, when a user kills a blocking session or when your alert template has been modified.

Import/Export
The Import/Export wizard allows users to import and share XML files. Save time and resources by importing or exporting alert templates, alert responses, custom counters and custom reports

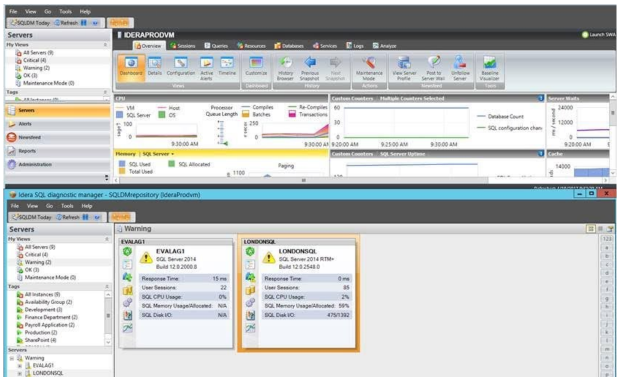
Given the number and different types of users in SQL Server production environments, keeping track of changes to the configuration of SQL Diagnostic Manager is critical. View a list of the main actions performed in the SQL Diagnostic Manager environment (such as when adding a server for monitoring, and changing an alert configuration or grooming configuration). View a thorough and instant list of information on modifications. Refer also to the documentation Use the Change Log to review changes in your SQL Diagnostic Manager configuration.

Manage Large Environments

Scale per each deployment whether monitoring 50, 150, 250, or 350 instances.

Access Multiple Repositories with Desktop Console

The desktop console is also flexible in its connectivity options. Open the desktop console more than a single time to view data from multiple repository deployments. For example, view two different consoles open together. Refer also to the documentation Monitor instances.



EVALAG1

EVALAG1
SQL Server 2014
Build 12.0.2000.8

Response Time:	15 ms
User Sessions:	22
SQL CPU Usage:	0%
SQL Memory Usage/Allocated:	N/A
SQL Disk I/O:	N/A

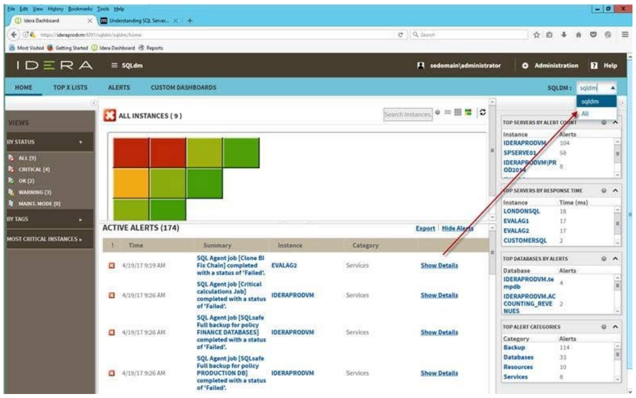
LONDONSQL

LONDONSQL
SQL Server 2014 RTM+
Build 12.0.2548.0

Response Time:	0 ms
User Sessions:	85
SQL CPU Usage:	2%
SQL Memory Usage/Allocated:	59%
SQL Disk I/O:	475/1392

Access Multiple Repositories with Web Console

The web console provides additional product scalability and federation of many IDERA products, viewable in a single web browser interface. Attach to multiple deployments to view them as a single enterprise unit. The web console provides much of the same useful data as the desktop console including alerts, heat map, sessions, query activity, query waits over time, query waits by duration, database health, alert timelines, top instances, and custom dashboards. The web console manages selected individual deployments or all instances. Refer also to the documentation Navigate the web console dashboard.



ACTIVE ALERTS (174)

!	Time	Summary	Instance	Category	
☒	4/19/17 9:19 AM	SQL Agent job [Clone BI Fix Chain] completed with a status of 'Failed'.	EVALAG2	Services	Show Details
☒	4/19/17 9:26 AM	SQL Agent job [Critical calculations Job] completed with a status of 'Failed'.	IDERAPRODVM	Services	Show Details
☒	4/19/17 9:26 AM	SQL Agent job [SQLsafe Full backup for policy FINANCE DATABASES] completed with a status of 'Failed'.	IDERAPRODVM	Services	Show Details
☒	4/19/17 9:26 AM	SQL Agent job [SQLsafe Full backup for policy PRODUCTION DB] completed with a status of 'Failed'.	IDERAPRODVM	Services	Show Details

TOP SERVERS BY ALERT COUNT

Instance	Alerts
IDERAPRODVM	104
SPSERVE01	58
IDERAPRODVM\PROD2014	8

TOP SERVERS BY RESPONSE TIME

Instance	Time (ms)
LONDONSQL	18
EVALAG1	17
EVALAG2	17
CUSTOMERSQL	2

TOP DATABASES BY ALERTS

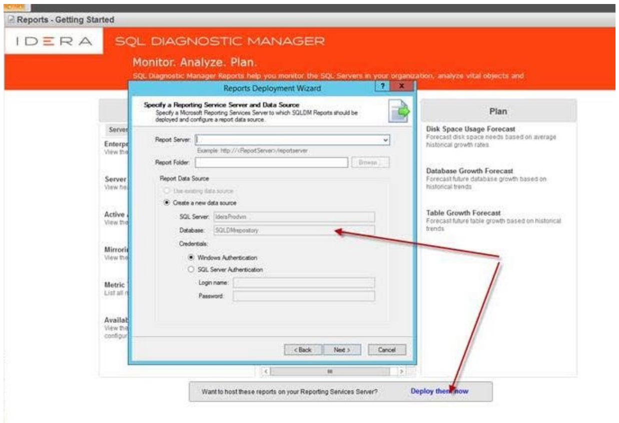
Database	Alerts
IDERAPRODVM.tempdb	4
IDERAPRODVM.ACCOUNTING_REVENUES	2

TOP ALERT CATEGORIES

Category	Alerts
Backup	114
Databases	33
Resources	10
Services	8

Publish Reports from Multiple Data Sources with SSRS

Deploy and publish the provided reports via the publishing wizard. The publishing wizard creates a data source automatically for whichever repository the console is connected. Through SQL Server Reporting Services (SSRS) folder-based management, generate reports from a single browser interface to span multiple repository data sources. Refer also to the documentation Report on SQL Server Performance.



SQL Diagnostic Manager Reports help you monitor the SQL Servers in your organization, analyze vital objects and

Specify a Reporting Service Server and Data Source
Specify a Microsoft Reporting Services Server to which SQLDM Reports should be deployed and configure a report data source.

Report Server:
Example: http://<ReportServer>/reportserver
Report Folder:
Report Data Source
○ Use existing data source
● Create a new data source
SQL Server: IderaProdvm
Database: SQLDMrepository
Credentials:
● Windows Authentication
○ SQL Server Authentication
Login name:
Password:

Disk Space Usage Forecast
Forecast disk space needs based on average historical growth rates

Database Growth Forecast
Forecast future database growth based on historical trends

Table Growth Forecast
Forecast future table growth based on historical trends

Want to host these reports on your Reporting Services Server?