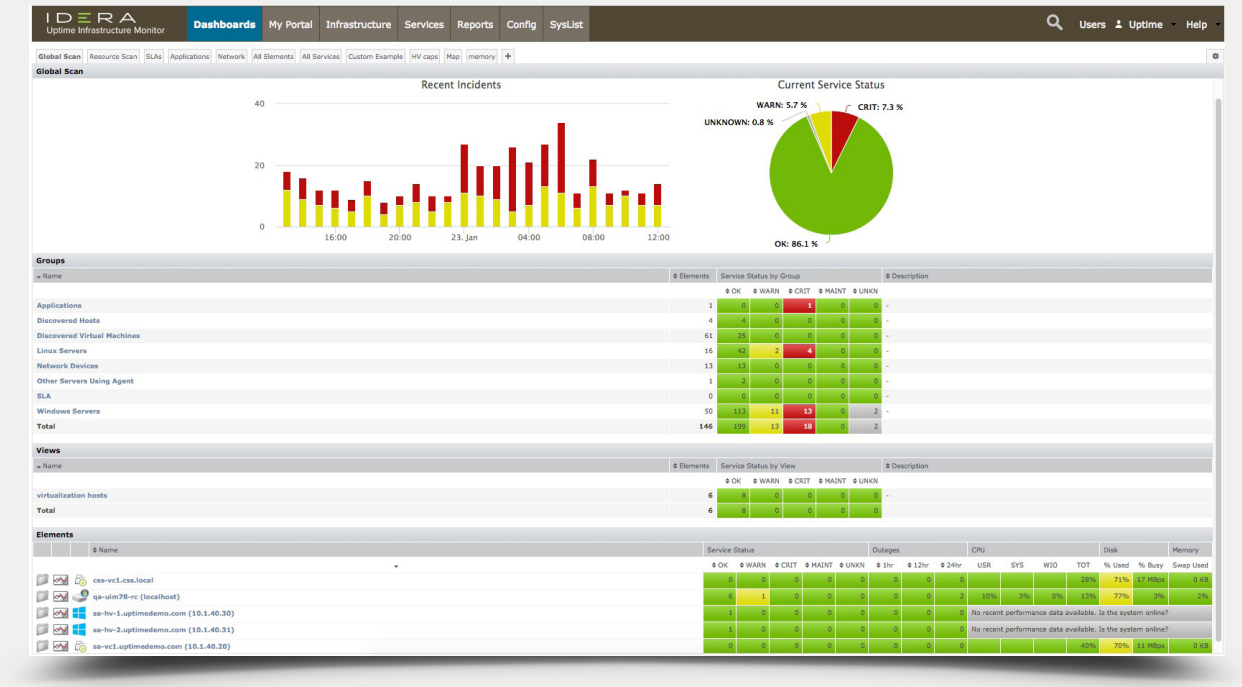
IDERA® | Database Administration

KEEP TRACK OF DATABASE PERFORMANCE WITH UPTIME INFRASTRUCTURE MONITOR

UPTIME INFRASTRUCTURE MONITOR

Uptime Infrastructure Monitor (UIM) is an enterprise tool that provides a unified view for Information Technology systems monitoring. Although comprehensive and scalable, UIM is also easy to use and deploys in minutes. Through proactive, comprehensive, and integrated monitoring, reporting, and alerting, UIM gives complete visibility and control over the availability, performance, and capacity of Information Technology environments — including applications, physical and virtual machines, network devices, and storage.



UIM also monitors the performance of the following relational database management systems:

- Microsoft SQL Server
- Oracle Database
- MySQL and MariaDB
- IBM Db2
- PostgreSQL
- SAP (Sybase) ASE (Adaptive Server Enterprise)

This solution brief discusses the service monitors and the advanced monitor plug-ins for database performance monitoring.

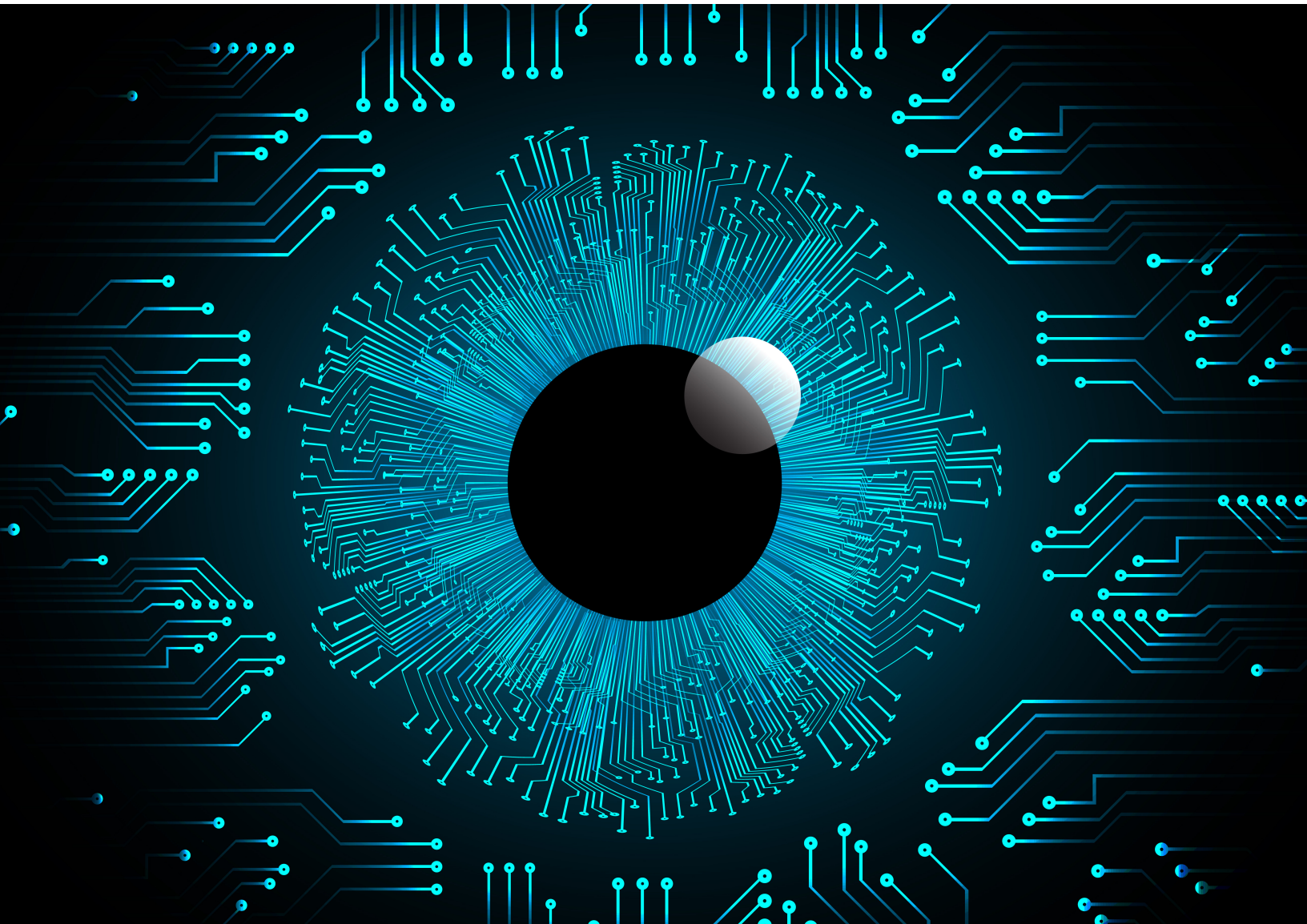
Service monitors

The **service monitors** are processes within UIM that regularly check the performance and availability of services in your environment. If the monitor detects a problem, UIM issues an alert. The different service monitors that are available in UIM are agent monitors, Microsoft Windows monitors, virtual machine (VM) monitors, database monitors, application monitors, network service monitors, and advanced monitors.

The **advanced monitors** are custom scripts and programs. Examples of advanced monitors are available as downloadable plug-ins. The advanced monitors do not require an installed agent on monitored systems.

Relational database management systems

UIM monitors the performance of the relational database management systems Microsoft SQL Server, Oracle Database, MySQL and MariaDB, IBM Db2, PostgreSQL, and SAP (Sybase) ASE (Adaptive Server Enterprise).



Microsoft Sql Server

For Microsoft SQL Server, monitor basic and advanced metrics, queries, tablespaces, clusters, Always On availability groups, and job status.

The **basic checks monitor** determines whether an instance is listening on a server's ports, determines whether an instance can process queries, and checks for values in any base and computed tables. The monitor runs queries against an instance, matches for user-specified regular expressions, and reports and alerts on the response and response time.

The **advanced metrics monitor** collects information on the availability and performance of individual databases. Collect performance metrics per instance or capture separately different performance metrics for each database. The monitor relies on agent or Windows Management Instrumentation (WMI) connections to access the Windows counters that provide the metrics. To monitor a virtual machine (VM) element, it needs to use agent or WMI collection. The monitor captures these metrics: Lock Wait / Sec., Lock Requests / Sec., Average Lock Wait Time, User Connections, Transactions / Sec., Data File Size / KB, Total Latch Wait Time, Latch Waits / Sec., Average Latch Wait Time, Maximum Workspace Memory, Connection Memory, SQL Cache Memory, Total Server Memory, and Response Time.

The **query monitor** is like the basic checks monitor. The monitor runs queries against an instance, and checks the response and the response time. For a response that is numeric or the number of returned rows, it uses an operator such 'greater than' and 'less than' to check the output. For a response that is a text string, it uses an operator such as 'contains' and 'does not contain' to check the output. If the response does not match the expected outcome, then it sets a warning or critical state for reporting and alerting.

The **tablespace check monitor** evaluates the size of data files within databases. The monitor gathers information from all databases across all instances on a system and aggregates this information in its metrics. The monitor also reports and alerts whether any of the data files in a filegroup or any log file in any database on an instance exceeds user-specified warning and critical thresholds.

Oracle Database

For Oracle Database, monitor basic and advanced metrics, queries, and tablespaces.

The **basic checks monitor** determines whether a host server is available, determines whether an instance is listening on a server's ports, determines whether it can log into an instance, and evaluates a response based on a SQL script that runs against an instance or a database. For the executed SQL script, the monitor measures the response time, and matches for the response text using user-specified regular expressions for reporting and alerting.

The **advanced metrics monitor** collects metrics for database performance. Some metrics are intended for tuning databases for long-term performance gains, rather than avoiding outages. The long-term probes are: Buffer Cache, Data Dictionary Cache, Disk Sort Ratio, Library Cache, and Redo Log. The monitor collects these metrics: Buffer Cache Hits Ratio, Data Dictionary Cache Hits Ratio, Library Cache Hits Ratio, Redo Log Space Request Ratio, Disk Sort Rate, Active Sessions, Oracle Blocking Sessions, Oracle Idle Sessions, and Response Time.

The **query monitor** is like the Basic Checks monitor. The monitor runs queries against an instance, and checks the response and the response time. For a response that is numeric or the number of returned rows, it uses an operator such as 'greater than' and 'less than' to check the output. For a response that is a text string, it uses operators such as 'contains' and 'does not contain' to check the output. If the output does not match the expected outcome, then it creates a warning or critical state for reporting and alerting.

The **tablespace check monitor** checks the relative size of individual tablespaces within database instances. The monitor reports and alerts when a tablespace in an instance exceeds user-specified thresholds.

The **extendable tablespace check monitor** collects metrics for tablespace capacity. The monitor collects these metrics: Available Space, Used Space, Free Space, Percent Free, and Response Time. The monitor captures these metrics for each tablespace within a database, and uses these metrics for alerting and keeps them for graphing within UIM.

MySQL and MariaDB

For MySQL and MariaDB, monitor basic and advanced metrics, replication, and status.

The **basic checks monitor** determines whether a host that is running a database is available, determines whether it can log into a database, and evaluates a response based on the execution of a user-specified SQL script against an instance or a database. The monitor runs queries, and matches for user-specified regular expressions for reporting and alerting.

The **advanced metrics monitor** determines whether an instance is listening on a server's ports, and checks performance values to establish the efficiency of an instance. The monitor collects these metrics: Uptime, Questions, Slow Queries, Open Tables, QPSA, Bytes Received, Bytes Sent, Delayed Insert Threads, Delayed Errors, Max Used Connections, Open Files, Open Streams, Table Locks Immediate, Table Locks Waited, Threads Cached, Threads Connected, Threads Running, QCache Queries In Cache, QCache Inserts, QCache Hits, QCache Lowmem Prunes, QCache Not Cached, QCache Free Memory, QCache Free Blocks, QCache Total Blocks, and Response Time. The monitor checks the performance metrics of databases and instances that are running on a system against user-specified thresholds. If an instance or a database is not responding, then a database can process queries, but the results show behavior that alerts to a problem.

The **replication monitor** collects these metrics: Seconds Behind Master, Slave IO State, Slave IO Running, Slave SQL Running, Last IO Error Number, Last IO Error, Last SQL Error Number, Last SQL Error, and Response Time.

The **status monitor plug-in** gathers from a database using the 'Status' command Connections / Threads Connected, Open Tables, and Average Queries per Second.

IBM Db2

For PostgreSQL, monitor basic and advanced metrics, replication, and status.

IBM **Db2**

For IBM Db2, monitor basic metrics, tablespaces, and buffer pools.

The **basic monitor plug-in** runs queries against an instance, matches for user-specified regular expressions, and reports and alerts on the response and the response time.

The **tablespace monitor plug-in** collects metrics concerning tablespace performance for reporting and alerting. The monitor captures these metrics: Tablespace Used, Tablespace Available, Tablespace Used, Used Pages In All Tablespaces, and Free Pages in all Tablespaces. The monitor collects the data for all of the tablespaces that exist on an instance.

The **buffer pool monitor plug-in** collects metrics concerning buffer performance for reporting and alerting. The monitor captures these metrics: Non-Accessed Asynchronous Reads, Average Asynchronous Read Time, Average Synchronous Read Time, Average Read Time, Non-Prefetched Synchronous Reads, Average Asynchronous Write Time, Average Synchronous Write Time, Average Write Time, Asynchronous Write Percentage, Data Hit Ratio, Index Hit Ratio, Auxiliary Storage Objects Hit Ratio, Total Hit Ratio (Index, Data, and XDA), and Response Time.

PostgreSQL

For PostgreSQL, monitor basic and advanced metrics, replication, and status.

For PostgreSQL, monitor basic metrics and availability.

The **basic monitor plug-in** captures the availability of a database for reporting and alerting.

The **availability monitor plug-in** determines whether it can connect to an instance, runs queries against an instance or database, and compares the response to user-specified text strings for reporting and alerting.

SAP (Sybase) ASE (Adaptive Enterprise Server)

For SAP (Sybase) ASE (Adaptive Enterprise Server), monitor basic metrics.

The **Sybase monitor** determines whether a database is listening to a server's ports, runs queries against an instance or database, matches for user-specified regular expressions, and reports and alerts on the response and the response time.

EXAMPLES

UIM triages large environments so that you can quickly determine what needs your immediate attention. For example, databases down, instances that are stopped, instances that are about to run out of space, and instances that are imminently unavailable are highlighted in red. The availability is tested via ping, process or service running, and successful sign-in and query.

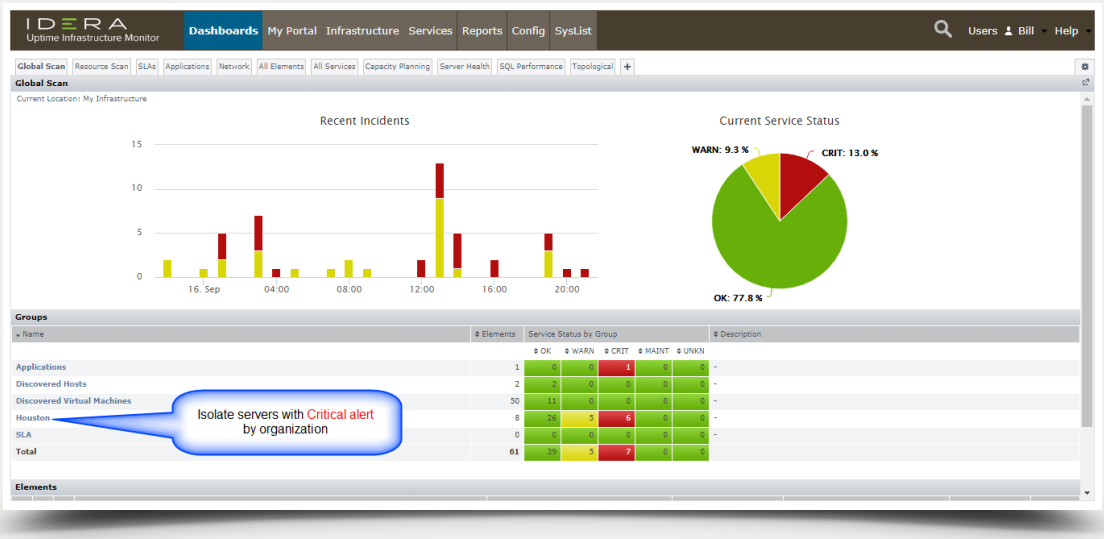
Many monitors create a fog of numerous alerts. This wastes time as a database administrator has to figure out what is truly important and what is not. UIM focuses on what you need to know first. Fixing the problem can be automated. UIM can execute a script to restart a service, for example. UIM will then retest to make sure that the fix succeeded. Automation that makes your life easier is critical to managing a large environment.

The following three screen capture images show how to quickly assess a large environment:

1. Isolate an organizational grouping (Houston) that needs your attention
2. Focus on the SQL Server instances grouped in QA, trend the crucial metrics
3. Pinpoint a particular server & instance that is down / performance problems.

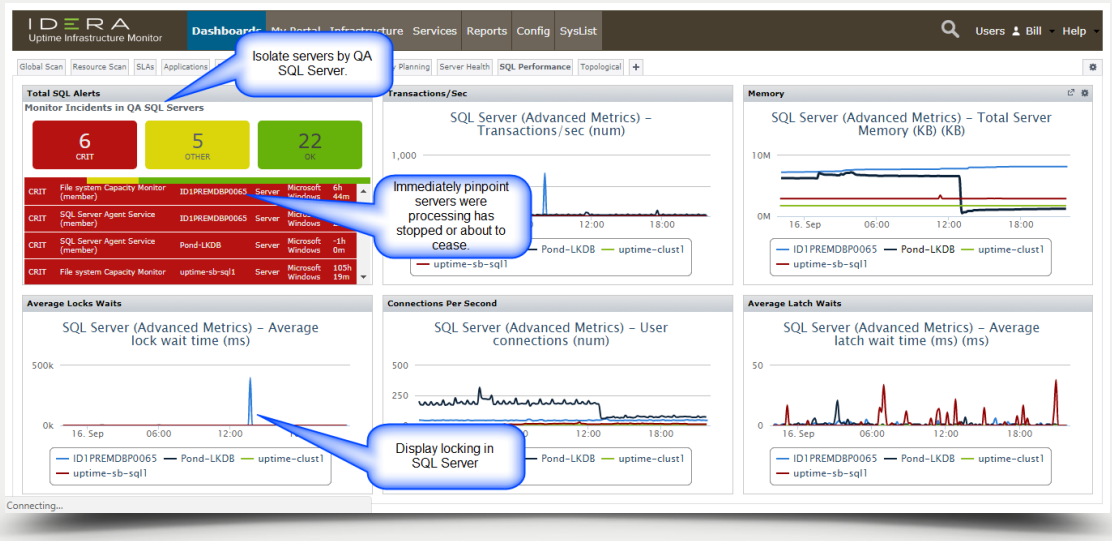
Operations dashboard Enterprise Server)

View critical alerts at the data center level:



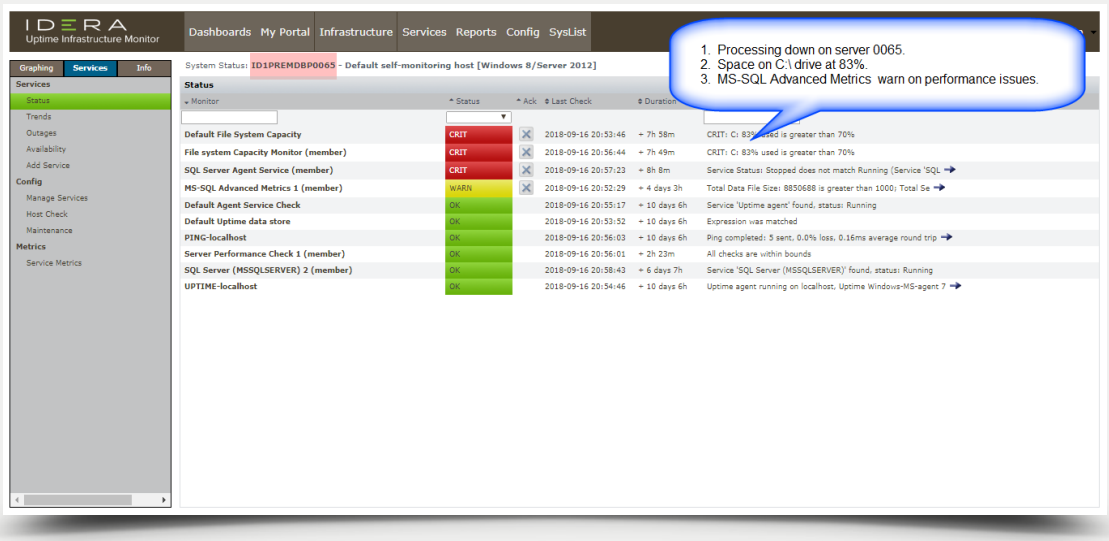
Focus on particular SQL Server instances

Monitor SQL Server instances:



Isolate individual servers

Alert for service down and system full for SQL Server instances:



IDERA'S SOLUTION

IDERA's Uptime Infrastructure Monitor (UIM)

Uptime Infrastructure Monitor (UIM) provides comprehensive and unified monitoring and optimization of physical and virtual servers, applications, networks, and databases from a central dashboard. Also, use historical data to plan for future server capacity needs, and track service-level performance trends for compliance and reporting concerning service level agreements.

Learn more by downloading a free, fully-functional, 30-day trial.

See UIM in action by requesting a personalized demo.

Start for FREE

FREE DEMO