

A Comparison of Proactive Monitoring Methods for Oracle Databases

Executive Summary

The lowest common denominator when it comes to basic database monitoring for availability and performance metrics is the familiar script-based solution. It is a relatively straightforward option to build and can be done piecemeal, between other projects. With fewer moving parts than some other alternatives, this may be the path of least resistance, but it can still be fraught with pitfalls. These systems are all too often cobbled together, poorly documented, and difficult to maintain. Even the best ones are not as flexible or scalable as the average commercial solution. The script-based monitoring solution is likely adequate only for low-end, relatively static environments.

Oracle's out-of-the-box management and monitoring architecture is known as *Enterprise Manager* (OEM), and the accompanying GUI is the *Database Control*, a browser-based tool that is new with Oracle 10g. With Enterprise Edition and the separately licensed, performance-related extensions, Diagnostics and Tuning Packs, this makes for a highly capable application. In the right hands and with development and enforcement of the right processes, Database Control presents a big improvement over the script-based solution in proactive monitoring and general monitoring effectiveness.

As capable as it is, however, Database Control gets you only so far. Getting through the implementation is no small chore. Proactive monitoring is as much about process as it is about technology. In other words, who is looking at notification emails, acting on them, and communicating status to other staff? If there is no procedure in place to dictate a proactive stance or available staff to carry it out, then the best tools on the market will not help.

A service like the one Database Specialists offers provides the full package—the monitoring tool, reporting portal, time-tested procedures, and experienced personnel to pull it all off. The primary focus is on proactive monitoring and maintenance, with the goal of preventing problems before they happen. Collaboration with internal staff should yield a partner role that is best for the organization, whether that is performance consulting and providing general Oracle resources, being the complete Oracle database production support capability, or something in between. In addition, Database Specialists' Daily Review service provides you with a daily email reminder of current health, along with any new events over the past 24 hours, written by one of your "primary" DBAs and including relevant advice and recommendations. This all makes for a formidable combination of people and process that neither script-based solutions nor Oracle's out-of-the-box tools can match.

Script-based Monitoring Solutions

Many DBAs, when faced with the question of how to do basic monitoring for availability and performance metrics, will rely on familiar and time-tested Unix shell scripts. This solution is relatively straightforward to build and can be done in stages, between other projects. Scripts can be scheduled to run with the help of the standard Unix scheduler, cron. Notification can be handled through a local email server and a set of email aliases. The following examples illustrate the possibilities.

Availability Monitoring. The most fundamental form of monitoring is a simple "ping" of the database to see if it is responsive. A script will generally initiate a database session via SQL*Plus and possibly perform a simple transaction. If the connection is established via the Oracle listener, it serves as a legitimate test of whether the database is available to applications and users.

Oracle Alert Log Monitoring. The next logical place to keep an eye on is the Oracle alert log, where server-based and some session-based errors are reported. This is the central logging point for the server, so it is important to watch for errors and warnings. A monitoring script will typically run every 5 to 15 minutes, keeping track of where it left off in the log following the last run, or tailing off the log should it find any errors.

Oracle will log many different types of errors and warnings, and raising a red flag for each of them can become an annoyance to the recipient of frequent emails or pages. It is therefore necessary to account for exceptions—messages that might fit the basic alerting

criteria but do not indicate a problem that requires attention. The standby databases, for example, are notorious for sending many warnings to the alert log. The key failure of a script-based solution here is that exceptions are typically hard-coded in the script. To inspect and maintain these exceptions, the DBA must access and read the script code. That's not the end of the world, of course, but a dynamic, data-driven solution for managing exceptions and alert severity based on the error message would be an improvement.

Tablespace Free Space Reporting. The next-most-attainable area of monitoring one might expect from a script-based solution would be in the area of free space. A tablespace is the logical extension to a file system data file, an association of physical files that hold existing data and reserve free space for future consumption. When a tablespace becomes full, tables and indexes contained in the tablespace can no longer expand. A script can realistically define an alerting threshold that is good for most cases, and can send notifications when a tablespace exceeds that threshold.

File System Free Space Monitoring. The other side of the free-space coin, the physical disk space available for expanding tablespaces, is also a realistic monitoring target for script-based solutions, although there are challenges that go beyond monitoring tablespace free space. It is important, for example, to forewarn of potential space issues based on growth rates, particularly when it comes to archived transaction logs. When a big batch update runs, it could generate gigabytes of transaction logs. The better monitoring solutions will gauge that rate of growth and predict how long the available free space will last if the growth rate is sustained.

Another challenging area in monitoring free disk space is related to Oracle's newer file-storage models, such as the *Flash Recovery Area* and *Automatic Storage Management (ASM)*. The flash recovery area holds backups and archived redo logs, and potentially other types of database files. It utilizes the concept of reusable space and tracks the necessity of particular files to a database recovery. ASM pools available disks into logical groups and manages storage through a separate, dedicated Oracle instance. A file system free-space monitoring solution that does not account for both of these possibilities is not up to snuff.

SQL Statement Captures. Given what we know about performance bottlenecks, monitoring resource-intensive SQL statements is probably the most important thing we can do. Unfortunately, detecting offending SQL statements proactively is challenging for any monitoring system, let alone one that is script-based. The typical shared memory segment that contains currently and formerly executed SQL statements will contain thousands of them, so just reporting everything that's running would result in a flood of useless information. We must therefore identify metrics, such as logical and physical reads, and set up alerting thresholds to limit the red flags raised. This most certainly requires a data-driven monitoring architecture, which likely holds a history of "interesting" SQL statements. We must also determine whether to alert on SQL statements violating the alerting threshold based on total impact, per execution numbers, or some other metric. This is no small feat, which is why a script-based solution is not likely to succeed, at least not in terms of our taking a proactive stance.

Statspack and Automatic Workload Repository (AWR) Report Captures. The database workload repository creates snapshots every hour. The same goes for the optional legacy product Statspack. Both of these repositories are simply recording mechanisms, but they are effective for forensic analysis. When presented with the question of what might have been happening in the database at any given hour, Statspack and AWR reports will likely shed enough light to allow for the formation of a diagnosis. Both repositories are kept in the same database that they are monitoring. The Statspack data is compact enough that a long history can be maintained. AWR data is much more broad and therefore cannot practically be housed as long without increasing management overhead. A script-based monitoring solution might therefore come into play with capturing reports or data for longer-term storage.

Licenses for Oracle Enterprise Edition, along with the Tuning and Diagnostics Packs, are necessary to run AWR reports. There are no licensing requirements to use Statspack.

Advantages of script-based solutions are few, but some do exist nevertheless. Implementation is typically straightforward, due to the limited number of moving parts. Script-based solutions are self-contained and don't typically involve data repositories. They are often homegrown solutions, so customization is more accessible. Impact on the production database and host is usually very slight.

Disadvantages of script-based solutions are considerably easier to list, because there are many. The maintenance burden of code and alerting conditions is inherent in the solutions' homegrown nature. As with any source code base, documentation is likely limited. Knowledge of its inner workings may depart with the DBA most responsible for implementation and/or maintenance. The lack of sophistication does not lend itself to a more dynamic, data-driven management interface for alerting threshold adjustment and notification settings, among other things. Major database upgrades that often impact the database data dictionary can mean losing some monitoring points until changes are made to the source code. And all of these issues only lead up to the biggest drawback of all: The monitoring points are never as comprehensive as what you would find with a commercial monitoring solution.

Oracle Enterprise Manager and the Database Control

Overview

Oracle's built-in management and monitoring infrastructure is identified by the product name Enterprise Manager. You will also see it referred to by either of the two acronyms EM or OEM. It will handle management of many Oracle components that make up the enterprise, one of which is the database. The database documentation set contains no less than 43 manuals covering various aspects of the Enterprise Manager environment. This is a testament to the scope involved and the level of challenge presented in the implementation of these tools.

Database-specific Products

The first challenge in getting your arms around Enterprise Manager is to figure out the ever-changing product names and translate Oracle's pervasive marketing-speak. For the sake of brevity, we will cover the Database Control product and the applicability of the Tuning and Diagnostic Packs to each.

Database Control. This is the built-in, out-of-the-box tool Oracle provides with all databases; it is intended as the default option for limited management and monitoring. Configuration is typically straightforward and provided by the Database Configuration Assistant (DBCA), although if something doesn't go quite right, a lot of time can be invested in figuring out the problem. One common problem is that even though this product is applicable to local databases only, server names and IP addresses are embedded into configuration files, so if something changes in that regard, the Database Control will cease to function.

The Database Control provides management and monitoring of the following:

- General database status (up/down/quiesced/etc.)
- Host CPU usage
- Bottlenecks
- Storage-related issues
- Backup configuration and maintenance
- Job scheduling

The Management Packs. Performance management is provided through the Database Control, provided you have Enterprise Edition and have sufficiently licensed it with the Diagnostics and Tuning Packs add-ons. The tool does a fine job of presenting information in colorful displays of various performance metrics. The graphs often allow drilldown capability, giving the knowledgeable DBA the opportunity to pinpoint exactly what the charts are expressing visually.

Diagnostics Pack. This add-on provides for the following features and functionality:

- Access to AWR data
- Command-line APIs and PL/SQL packages
- Performance monitoring and diagnostics
- ADDM Reports
- Limited operating system monitoring on the database host
- Adaptive metric thresholds
- Advanced alert management
- Maintenance of diagnostic data history
- Notification methods, rules, and schedules

Tuning Pack. This second add-on requires the Diagnostics Pack and is aimed at tuning applications through better SQL execution plans and data access paths. It contains the following components:

- SQL Tuning Advisor
- SQL Access Advisor

- Online object reorganization
- SQL Tuning Sets

The specific feature sets provided by each of these applets are beyond the scope of this document but can be obtained from Oracle's product documentation set.

Implementation

Oracle Enterprise Manager with Database Control and the Diagnostic and Tuning Packs has the potential to provide a solution to the proactive monitoring challenge, more so than a script-based solution. There is no shortage of available information. In fact, it could be overwhelming at times. While the graphs are appealing and the drilldown capability very handy, there will always be times when the interface is lacking. The DBA must invest the time in OEM configuration and troubleshooting to get it going, gaining a comfort level with the GUI, then have patience while thresholds (adaptive or not) are tuned so that the meaningless alerts are constrained. So the question remains: Does the tool save time and allow a busy DBA to get the job of monitoring accomplished? There is certainly some potential for that if you have the time to invest in the tool in the first place and acquire a separate skill set associated with OEM and Database Control.

Many GUIs meant to make database administration easier tend to give pause to DBAs, because all too frequently, they take unintended actions. The Database Control user interface is not so sophisticated that it will warn you before stopping, or "quiescing," the database at times. The user interface does not make clear exactly what it will be doing in response to a button or link click. The DBA might be doing something that appears relatively harmless but unintentionally locks a table. The truth is that while the GUI simplifies some tasks, such as job scheduling, it does not preempt the necessity that the DBA still have the fundamental skills to manage the database using the basic tool set—command-line tools like SQL*Plus and Recovery Manager, along with shell scripting.

OEM is not configured on the production database server at the present time, so the first step in getting these tools set up would be to create a repository and the EM agent and server configuration. This is usually a simple step, and Database Control just works. When it doesn't, however, a solution can become elusive, and often Oracle Support must become involved. The command-line configuration tool has been known to inexplicably kill database sessions.

Database Specialists DBA Pro Monitoring

Overview

Our Daily Review service is not *tool*-centric—rather, it is *team*-centric. We have our tools—the primary one of which is the Database Rx® monitor—but what sets us apart is our team of highly experienced DBAs and our method of collaboration with in-house staff. Our goal is to complement your operation to the extent that you feel is necessary, whether that is to serve as a performance-tuning resource to an in-house DBA, as an Oracle resource to your system administrator, or as your complete production support capability. These are just a few examples; we can also do training, mentoring, advisement, and special projects. It is up to you, but monitoring and proactive maintenance are always a part of any Daily Review plan.

Database Rx®

Central to our method is the implementation of our monitoring agent, our analysis and notification engines, and the Performance Portal. This combination allows us to maintain constant contact with your database and stay informed about database health.

Monitoring Agent. This is a lightweight shell script that runs every few minutes, to do a quick database-health check, and then hourly, to gather detailed statistics. The agent does not query the AWR data, so special Oracle licensing is *not* required. Monitoring data is transmitted to our servers via encrypted email messages. The agent can run on the database server or on a dedicated appliance provided by Database Specialists.

Analysis Engine. Incoming monitoring data is loaded into our database and tested immediately for the possibility of new alerts. Over 100 tests are applied to the newly arrived metric data. Alerting categories and thresholds are dynamically tunable at a high level of granularity so that needless alerting is avoided. Whether or not any alerting criteria are met, the statistical data is stored and used in order to accumulate a six-month history of database metrics that becomes invaluable in the analysis of future performance problems and many other unforeseen issues.

Notification Engine. Alerts that do occur are categorized by severity level, from 1 to 5. Severity 1 and 2 alerts are those that often call for immediate action; in these cases, our team of DBAs may be paged, depending on whether your specific plan calls for 24/7 or business-hours coverage. Other alerts will use email as the notification method. Your staff may also choose to be paged and/or emailed, contingent upon alert severity easily adjusted via the Database Rx® Performance Portal.

Performance Portal. Your database's current health and entire history of metric data are available from any web browser with your username and password. A variety of reports are available to you, ranging from the Performance Summary to Database Growth and Daily Review history. You won't find a lot of colorful graphs—what you will find is useful information that is easily digestible and relevant to your day-to-day operations.

Proactive Maintenance

Our proactive approach to monitoring and maintenance is key to staying ahead of the curve. We monitor for areas with potential to deter or completely interrupt normal processing, and seek to take action before database users are impacted. That is the whole approach to our Daily Review service. Every weekday morning, you will receive an email with a review of the past 24 hours (or 72 hours on Mondays), along with recommendations, general information, and a summation of ongoing issues. Depending on the particular style of our collaboration with your staff, we may take corrective action on our own initiative, ask your permission before doing anything, or simply recommend what action your staff should take.

Included Monthly Consulting Hours

Every DBA Pro plan includes an appropriate number of consulting hours at your disposal. We recommend using these hours for proactive maintenance, but when things are quiet and you have a query that needs to be tuned, a QA database that needs to be refreshed, a server migration in the works, or any other Oracle database-related task, you are free to use our consultants to augment your staff. Hours used beyond the monthly included hours are billed at the contractual rate, but we try to issue a warning before you reach that point.

Fully Customizable

Every environment is different. Expecting one out-of-the-box tool to accommodate every monitoring point is somewhat unrealistic. The Database Rx® suite is designed to be customized with up to five monitoring points that are tightly integrated into the analysis and notification engines. For example, a logical standby database, not something you see every day, might be the target of a custom metric metering the transaction apply process to be sure it was staying reasonably current with production.

Dedicated Primary and Secondary DBAs

As part of the DBA Pro plan, you are assigned two of our DBAs, who act as your primary points of contact. Both will be very familiar with your staff and environment and will share daily review duties. They will also provide backup/vacation support if your in-house DBA becomes unavailable. You will be guaranteed two Oracle-certified professionals with at least ten years of experience, a basic prerequisite for joining the Database Specialists team. Your staff will also have access to other DBAs on our team, guaranteeing that you will never be left in the lurch because you cannot reach the DBA.

Comparative Analysis

The following is a discussion of how each of the three previously mentioned monitoring strategies stacks up against a variety of measuring criteria.

Technical Requirements

Script-based Solution. A shell script-based monitoring solution generally has few technical requirements, with the exception of email, although the AWR and ADDM data they may access requires Enterprise Edition and add-on licensing of the Diagnostics and Tuning Packs. The load placed on the database and host by these monitoring scripts is usually minimal. When historical data is stored in the file system, the production database is not impacted; however, it may also be stored in the same or another support database.

Database Control. This Oracle tool requires configuring Enterprise Manager on the database server, a procedure that is typically straightforward but occasionally problematic enough to require support from Oracle. Enterprise Edition, plus add-on licensing of the Diagnostics and Tuning Packs, is required for performance monitoring. The load on the database caused by monitoring queries is not typically a problem. Historical data is stored in the production database and can become unwieldy in rare circumstances. The retention cycle is seven days by default but can be adjusted as appropriate.

Database Specialists. Similarly, our solution has few requirements, aside from email, and does not require Enterprise Edition or licensing the Diagnostics and Tuning Packs. Installation is simple and can be performed by you or our staff. The load placed on the database and host by the monitoring agent is fairly insignificant and will not impact users. Historical metric data is stored on our servers.

Staff Requirements

Script-based Solution. The demand placed on your DBA in this scenario largely depends on the complexity of the code and the extent to which monitoring is implemented. A large code base means more time with development and maintenance, enhancements, version control, alerting threshold adjustments, and so on. A simpler solution and less dynamic environment would probably mean little demand on staff, however.

Database Control. This tool has the potential to perform proactive monitoring while limiting staff requirements. The proverbial devil may be in the details, however. A considerable investment in DBA time would be required in order to implement the tool, see it through a shakeout period, fine-tune alerting thresholds, adjust notification methods and frequency, and so on. The staff requirements for getting through that process would likely be substantial.

Database Specialists. Our solution is designed to augment your staff and minimize unnecessary impact on your available resources. That's what we are about, so we would hope that you would establish staff requirements to meet your needs, depending on the style of collaboration we put into effect. In other words, you set the staff requirements, rather than having the monitoring solution dictate them to you.

Health Check Capability

Script-based Solution. The extent to which monitoring scripts perform health checks is all over the map. Some do a good job of recording overall server health by archiving AWR and ADDM reports. This data is not easily aggregated or presented, however. Report intervals are usually preset; therefore, problem periods must be anticipated. Because they are constant, however, it would be straightforward to compare periods of normal activity with periods of abnormal activity. Often, a snapshot of database health based solely on the current state is attainable, but more complex, trend-based metrics might be more elusive with this approach.

Database Control. This tool is fully capable of monitoring a number of database-health metrics; it is also capable of communicating issues via email. It would be up to staff to digest those emails, consult the console reports, and record/communicate current health to other concerned staff and management. It is certainly possible with this tool's notification layer; however, metric data is not kept for very long, because it is stored in the same database that it is monitoring. The default retention period is seven days.

Database Specialists. The Database Rx monitoring tool pings the database every few minutes and records a wide range of health-related metrics on an hourly basis. Because we maintain a much longer history of metric data than the other two options, this option grants us more flexibility in analyzing past versus present status: We can see how things looked months prior and compare that impression to the present day, something that would not be possible with the other two options. Furthermore, a dedicated DBA is issuing daily written reports on general health. Those reports themselves provide a useful record of health history.

Proactive Monitoring Capability

Script-based Solution. Much as is true of the health-check area, the extent to which proactive monitoring can be performed with scripts is highly variable. It is trickier to get proactive monitoring right than it is to perform health checks. The notification layer and its flexibility are key to success, as is the ability to fine-tune alerting thresholds so that alerting conditions requiring immediate action can be highlighted and not lost in a sea of alert noise. Like health checks, some monitoring areas are trend based, so those would be a challenge to implement successfully with scripts.

Database Control. This tool is fully capable of extending its health check–monitoring capability with an advanced notification model, providing the basic building blocks of rudimentary proactive monitoring. But proactive monitoring is as much about process as it is about technology. In other words, who is looking at notification emails, acting on them, and communicating status to other staff? If there is no procedure in place to dictate a proactive stance or available staff to carry it out, the best tools available will not help.

Database Specialists. Database Rx and the DBA Pro service plan are designed to provide proactive monitoring. We are constantly tasked with raising red flags and acting on issues before they reach the crisis stage, or even before the users become aware of a problem. Our alert notification model is best-of-breed, with email and pager methods backed up by an internal alert–escalation procedure. There will be eyes surveying and communicating database health every day. Of course, there will always be those rare times when database performance unexpectedly hits the skids, but we have a track record of successfully heading off problems before they happen.

Availability of Metric Data

Script-based Solution. This strategy does not typically lend itself to extensive data repositories or any data repositories other than those kept in the AWR or Statspack, so this area is not particularly relevant. What persistent data is available depends upon the design and complexity of the solution.

Database Control. This tool does a nice job of presenting information to the user, provided you know where to find it. The user interface can be a bit overwhelming to the casual user, because of the wide array of management capabilities available through the same tool, combined with reporting, advisors, etc. Nevertheless, once you find the screens you like and learn how to click through the graphs, the information that is available could be quite useful to the trained eye.

Database Specialists. The Database Rx Performance Portal shows strength in this area by making metric data available to anyone with Internet access and an appropriate username and password. Reports are varied and useful, although they are aimed mostly at DBAs and developers. Management and other staff may benefit from portal access, however, because this is where notification levels are defined and alert history is available.

General Effectiveness

Script-based Solution. Effectiveness of a script-based solution ranges from fairly adequate for a low-end database to totally insufficient. In a fairly static environment monitoring databases that don't experience high volume or growth rates, a carefully implemented script-based solution might be all that is required. Because script-based systems often do not scale or adapt well to change, they may not represent a long-term solution.

Database Control. Implementation of Database Control would typically be an improvement over the script-based approach, in terms of proactive monitoring. As far as monitoring tools go, this one is exemplary when used to its full potential. It is capable of monitoring one database and will store a relatively brief history of metric data in the production database. Licensing of the Tuning and Diagnostic Packs is required for monitoring of performance. The primary concern is less about this tool's capability and more about the ability of this tool alone to provide an end-to-end solution. It's just a tool, albeit a good one.

Database Specialists. We are focused on proactive monitoring and maintenance, collaboration with staff and management, communication of current and emerging issues, and filling in any gaps in staffing or coverage that may exist. We go beyond our capable monitoring tool to provide clients with the benefit of our experience, which is considerable. Monitoring covers a wide range of data points, including general availability, performance, recoverability, free space in the database and file system (both current and projected), high-resource SQL, and many other areas.

About Database Specialists, Inc.

Since 1995, Database Specialists has been providing Oracle database consulting and remote DBA support in Solaris, HP-UX, Linux, AIX, and Windows environments through our DBA Pro service offering. We are committed to educating Oracle users and the broader IT community on how to best manage their Oracle production environments. In addition to this document, we have numerous presentations, tutorials, and scripts available without obligation at www.dbspecialists.com.

Our Oracle DBAs have 10+ years of hands-on experience. Many of them are featured speakers at Oracle OpenWorld and the Independent Oracle Users Group (IOUG). They frequently publish white papers and technical articles in publications such as *SELECT Magazine*, *Oracle Magazine*, the *NoCOUG Journal* as well as our own Monthly Publication—*The Specialist* (<http://www.dbspecialists.com/specialist.html>). Our dedication to knowledge sharing has been acclaimed by IOUG through *SELECT Magazine*.

Visit our website or contact David Wolff (888-648-0500 ext. 525) today to learn more about how partnering with us can make a substantial business and technical impact on your organization.

