DATABASE MANAGEMENT
FROM CRISIS TO CONFIDENCE

David R. Cook
Senior Consultant - System Performance Group
Oracle Services, Salt Lake City, Utah USA

SUMMARY

Database Administrators in high stress environments often operate in crisis management mode, which escalates to poor database performance. This paper presents ideas for creating checklists that include proven performance tuning and administrative values. Consistent review of key areas of the database as outlined in the checklists will lead to an optimal database environment.

1. INTRODUCTION

As Peter F. Drucker states: "A well-managed factory is boring. Nothing exciting happens in it because the crises have been anticipated and have been converted into routine." A database management system is a virtual factory taking raw data and producing organized information. Maintaining stable information systems is essential to the success of many businesses. The stability of the database environment indicates the success of the database administrator (DBA).

All DBAs go into the job hoping to be successful in maintaining the perfect database environment. They often believe this means reacting to every user request. While in this noble mind-set, it is easy for the DBA to slip into reactive mode, running from one escalated issue to another. The problem is computers can't talk, but users can. DBAs often believe maintaining a stable environment means they must react to all user requests. They find themselves trying to please the user requests by day

and sacrificing personal time resolving growing problems in the system after hours. They become so busy reacting to problems, eventually the problems are in control. This allows small problems festering in the database to become large problems before being identified. The DBA eventually winds up slumped over the keyboard consoling the system on its first birthday instead of home with the spouse on their child's first birthday.

If we have learned anything from the old saying "an ounce of prevention is worth a pound of cure," we know large problems are more costly to resolve than small problems. Creating a solid database environment doesn't occur automatically and it doesn't occur by reacting to the problems after they have surfaced in the eyes of the users. It requires sound organization and execution of proven procedures, which monitor key points of the system to ensure stability on a daily basis. The DBA must be proactive. They must allocate time in their day to review key areas of the system and resolve problems before a crisis evolves.

2. PROACTIVE ADMINISTRATION

While acting as the DBA of a fast growing Oracle database, I found the new tasks of the DBA role to be overwhelming. In an effort to define the role of a DBA for my company, I began to list tasks required in order to maintain a stable database environment. Over time, I developed procedures that I ran on a daily, weekly, semi-monthly and monthly basis. I set aside time each day to go through at least my daily checklist. I convinced my co-workers of the importance of this daily system check in maintaining a stable system.

2.1. Evolving from Reactive to Proactive Administration

Each database environment is different and therefore requires a different approach, but sound checklists are essential to maintaining a consistent environment. A basic understanding of the Oracle RDBMS and user requirements are essential to developing sound checklists. The more you

know about the user requirements and available system resources, the better you will be able to match the two. System stability is created when consistent checks on the system ensure a constant match of system resources and user requirements. When you see a mismatch evolving you can react to the problem by providing additional system resources or by changing the demands of the users.

2.1.1. Improving Your Knowledge

There are a number of sources for improving one's knowledge on the administration of the Oracle environment. Publications from user conferences (such as white papers), the Oracle manuals and formal training are all excellent ways to improve knowledge and add value to any Oracle administrator. Unfortunately you can't assume everything you read is A) true, and B) applies directly to your environment. You need to test new ideas to determine if they improve stability in your situation. The more you read, the more ideas you'll add to your bag of tricks. The more you test these ideas in your environment, the more you will improve your understanding of both Oracle and your environment. Having said that, I've included a number of references at the end of this paper that contain many good ideas for database administration. I highly recommend you include these publications in your library and refer to them as you build sound checklists.

2.1.2. User Requirements

There is no better way to gain an understanding of what the users require of the Oracle environment than to ask them. This is a simple enough concept, but I am continually surprised at the number of sites where communication between the users and the administrative staff is limited. Part of the reason for this miscommunication is caused by the technical disconnect which often exists between the administrative staff and the users. As a DBA, it is your job not only to understand how Oracle works, but to understand how your business works and what is required of the system to be successful.

There is no better way to accelerate an understanding of the needs of the users (and business) than to put it in writing. Words are said and seconds later they float away, but when you put something in writing, it forces both sides to focus on the issues at hand. *Service Level Agreements* should be created with the users of the database and the administrative staff to define availability and performance requirements of the system to meet the needs of the business. These agreements should document how long the database can be down without negatively impacting the business and should specify the best time to conduct scheduled maintenance. The *Service Level Agreements* should document the key processes required to run the business with the expected throughput and response time for these key processes. An understanding of the key processes and required response time will assist you in setting system tuning goals. This is something that should be done before tuning takes place.

2.1.3. Defining Checklists

After you have a basic understanding of how Oracle functions, and you understand what the users need to run the business, you can begin listing the tasks required to meet the needs of the users and keep the system stable. When defining points in the checklists, keep the following thoughts in mind:

- **Recovery.** Ensuring the integrity of the database is the primary concern. You must ensure the data can be recovered with in a time-frame acceptable to the business.
- **Space Management**. Proper space management is essential not only to the stability of the database but to performance and ultimately cost.
- **Performance.** Maintaining optimal performance keeps the users happy and ensures the most cost effective use of system resources.

The checklists should include enough detail that someone with little knowledge of your environment could perform the system check. The checklists will become a valuable resource for those filling in while others are on leave. *Figure 1* gives an idea of the level of detail a checklist should include.

Figure 1: Sample Checklist

```
DBA Daily Checklist
Revised:    04-JAN-96
Author:     Dave Cook
Purpose:    This script gives detail for performing a daily
            check of the status of the database.
Change Notes:

      Clint Woodcox 04-JAN-96 :     Added the section to deal
                                    with archive file review.

Support Information:

Backup DBA Phone Information              999-9999
      Pager                               999-9999

Support Phone Information                 999-999-9999
      CSI#                                #########

Daily Procedures:

10)   Check e-mail account for error messages from the backup
      utility.
      20)   Logon to UNIX system as user cwilson.
      30)   Run e-mail command xmail cwilson at the
            UNIX prompt.
      40)   Look for messages with Backup Failed in
            subject line.
      50)   If the backup's have failed, discuss with
            the system administrator.
```

You may notice the checklist looks much like a pseudo program. In a sense it is a program; instead of a computer executing the script, it is executed manually. After the script is well defined, you may find a number of the tasks can be automated. The checklists provide direction for a smooth and effective path to the automated process.

The intervals at which the checks are run must be set in order to maintain the confidence level required. The more dynamic the environment, the more frequent checks on the environment should occur. The checkpoints are organized into checklists based on the frequency at which they must occur. Checkpoints that must occur on a daily basis are organized into a daily checklist, checkpoints that must occur on a weekly basis are organized into a weekly checklist, and so on. I've included ideas for daily, weekly, and semi-monthly checklists below. You will need to adjust the contents and intervals of your checklists based on the requirements of your environment.

2.1.3.1. Defining a Daily Checklist

A daily checklist should quickly determine the current state and stability of the system. Consider the following tasks for inclusion in a daily checklist:

- Make sure the database is available; log into the applications and perform basic functions.
- Make sure the database backup was successful.
- Make sure database archiving is functioning properly and the archive log files are being moved to tape properly.
- Review the *alert.log* file of the Oracle database instance to determine if there are any errors indicating problems with the database. It's a good idea to review the output of these critical log files more than once a day. You may find it useful to direct the output to these log files to your screen, so you can monitor the output on a regular basis.
- Make sure resources on the system are sufficient to provide acceptable performance. This may involve running processes to review contention for CPU, memory, network, or disk resources.
- Look for segments in the database that are running out of resources (e.g., extents) or growing at an excessive rate. You may need to adjust the storage parameters of these segments.

Capture the output from these daily reports and store this for future reference. The output from these reports will assist in determining future space and performance requirements.

2.1.3.2. Defining a Weekly Checklist

The weekly checklist should focus primarily on database maintenance. You may want to consider the following ideas for inclusion in the weekly checklist:

- Clean out or rebuild interim tables.
- Purge log files.
- Adjust storage parameters on database objects.
- Review user settings to ensure proper defaults and grants.

2.1.3.3. Defining a Semi-Monthly Checklist

The semi-monthly checklist should focus on maintenance and tuning. These checkpoints may need to be run on a weekly, or even a bi-weekly basis if the system is relatively new or is undergoing significant tuning.
You may want to consider the following tasks for inclusion in the semi-monthly checklist:

- Review changes in segment growth when compared to previous reports to identify segments with a harmful growth rate.
- Review common Oracle tuning points such as cache hit ratio, latch contention, and other points dealing with memory management; compare with past reports to identify harmful trends or determine impact of recent tuning adjustments.
- Review database file activity; compare to past output to identify trends that could lead to possible contention.
- Include checkpoints to investigate fragmentation (e.g., row chaining).

- Compare reports on CPU, memory, network and disk utilization both from Oracle and the operating system (work with the system administrator) to identify trends that could lead to contention for any one of these resources in the near future.
- Make the adjustment necessary to avoid the contention for system resources. This may include scheduled down time or request for additional resources.

3. CONCLUSION

Without routine review of proven checklists, it is easy to slip into reactive mode and ignore warning signs being given of an upcoming crisis. Developing checklists with sound management techniques will provide a road-map for experienced or inexperienced DBAs alike to identify and resolve problems before they reach the crisis stage.

4. ABOUT THE AUTHOR

David R. Cook is a Senior Consultant for the Oracle Services, System Performance Group. During his five years of experience with Oracle products, he has served as a developer, database administrator, performance specialist and technical architect. His specialization is performance management and capacity planning. David is based in Salt Lake City, Utah and can be reached at **(801)595-5674** or via e-mail at **dcook@us.oracle.com**. David's papers are available via the World-Wide-Web at **http://www.europa.com/~orapub**.

5. ACKNOWLEDGMENTS

This work was supported by Oracle Corporation without whom none of this would be possible. A special thanks to my wife Karen for being my

backbone and to Cary Millsap, Craig Shallahamer, Caryl Seastrand and Clint Woodcox for their technical and editorial support and expertise.

BIBLIOGRAPHY

1. Millsap, C. *Oracle7 Server Space Management*. October 1995. http://www.europa.com/~orapub.
2. Millsap, C. *Designing your system to meet your requirements*. Oracle corporation white paper. July 24, 1995. http://www.europa.com/~orapub.
3. Millsap, C. *The OFA Standard, Oracle7 for Open Systems*. Oracle Part No. A19308-1 May 1994. http://www.europa.com/~orapub.
4. Shallahamer, C. *Total Performance Management*. February 3, 1995. http://www.europa.com/~orapub.
5. Shallahamer, C. *Avoiding A Database Reorganization*. November 2, 1994. http://www.europa.com/~orapub.
6. Loukides, M. *System Performance Tuning*. O'Reilly & Associates, Inc. December, 1992.
7. Drucker, P. *Effective Executive*. Harper Business. 1967.