

This material is adapted from Craig S. Mullins' [Database Administration: The Complete Guide to Practices and Procedures](#), Addison Wesley Professional; ISBN: 0201741296.

DBA Staffing Considerations

By *Craig S. Mullins* (<http://www.CraigSMullins.com>)

Staffing the DBA organization is not a simple matter. Several nontrivial considerations must be addressed, including the size of the DBA staff and the reporting structure for the DBAs.

How Many DBAs?

One of the most difficult things to determine is the optimal number of DBAs required to keep an organization's databases online and operating efficiently. Many organizations try to operate with the minimal number of DBAs on staff; the idea being that fewer staff members lowers cost. However, that assumption may not be true. An overworked DBA staff can make mistakes that cause downtime and operational problems far in excess of the salary requirements of an additional DBA.

Determining how many DBAs is optimal is not a precise science. It depends on many factors:

- Number of databases. The more databases that need to be supported, the more complex the job of database administration becomes. Each database needs to be designed, implemented, monitored for availability and performance, backed up, and administered. There is a limit to the number of databases that an individual DBA can control.
- Size of the databases. The larger the databases that need to be supported, the more difficult the job of database administration. A larger database takes longer to create, maintain, and tune. In addition, more potential for confusion arises when SQL takes longer to execute—causing the DBA to spend more time working with developers to tune SQL.
- Number of users. As additional users are brought online, optimal database performance becomes more difficult to ensure. Additionally, as the number of users increases, the potential for increase in the volume of problems and calls increases, further complicating the DBA's job.
- Number of applications. A single database can be utilized by numerous applications. Indeed, one of the primary benefits of the DBMS is that it enables the sharing of data across an organization. As more applications are brought online, additional pressure is exerted on the database in terms of performance, availability, and resources. As more applications are brought online, more DBAs may be required to support the same number of databases.

- Service-level agreements (SLAs). The more restrictive the SLA, the more difficult it becomes for the DBA to deliver the service. For example, a service-level agreement requiring subsecond response time for transactions is more difficult to support than an agreement requiring three-second response time.
- Availability requirements. Database administration becomes easier if databases have an allowable period of scheduled downtime. Some DBA tasks either require an outage, or are easier when an outage can be taken. Considerations such as supporting e-business transactions and the Web drive the need for 24/7 database availability. 24/7 availability is often incompatible with certain DBA tasks.
- Impact of downtime. The greater the financial impact of an unavailable database, the greater the pressure on the DBA to assure greater database availability.
- Performance requirements. As the requirements for database access become more performance oriented, database administration becomes more complicated.
- Type of Applications. The type of applications supported has a direct bearing on the number of DBAs required. The DBMS and database needs of a mission-critical application differ from those of a non-mission-critical application. Mission-critical applications are more likely to require constant monitoring to ensure availability. Likewise, an OLTP application has different characteristics and administration requirements than an OLAP application. OLTP transactions are likely to be of shorter duration than OLAP queries; OLTP applications perform both read and write operations whereas OLAP applications are predominantly read-only. Each has administration challenges that require different DBA procedures.
- Volatility. The frequency of database change requests is an important factor in the need for additional DBAs. A static database environment requiring few changes will not require the same level of DBA effort as a volatile, frequently changing database environment. Unfortunately, the level of volatility for most databases and applications tends to change dramatically over time. It's usually very difficult to ascertain how volatile an overall database environment will be over its lifetime.
- DBA staff experience. The skill of the existing DBA staff affects the need for additional DBAs. A highly skilled DBA staff will accomplish more than a novice team. Skills, more than experience, dictate DBA staffing requirements. A highly skilled DBA with two years of experience might easily outperform a ten-year veteran who is burned out and unmotivated.
- Programming staff experience. If the application developers are not highly skilled in database and SQL programming, the DBAs will need to be more involved in the development process. DBAs will be needed for tasks such as composing complex SQL, analyzing SQL and application code, debugging,

tuning, and ensuring connectivity. As the experience of the programming staff increases, the complexity of DBA requirements decreases.

- End user experience. When end users access databases directly with ad hoc SQL, their skill level has a direct impact on the complexity of DBA. If the end user has few SQL skills, the DBA will need to be initiate more performance monitoring and tuning.
- Variety of DBMSs. The more heterogeneous the environment, the more difficult it becomes to administer. For example, acquiring and maintaining expertise in both Oracle and DB2 is more difficult than gaining expertise in only one of them. Moreover, as multiple DBMSs of different types are installed, database administration becomes even more difficult. For example, a shop with DB2, IMS, and IDMS will have to possess relational (DB2), hierarchical (IMS), and network/CODASYL (IDMS) expertise.
- DBA tools. DBMS vendors and a number of ISVs offer tools that automate DBA tasks and make database administration easier. DBA tasks become less complex with the more tools available and the degree to which they are integrated. Lou Agosta, an industry analyst with Giga Group, states that “without [DBA] tools up to twice the number of DBAs might [be] required.”

This list of issues notwithstanding, creating a formula that will dictate the optimal number of DBAs to employ is difficult. Industry analysts at the META Group have established a loose formula for calculating DBA level of effort. The formula arrives at a level of effort by applying weights to six factors: system complexity, application immaturity, end-user sophistication, software functionality, system availability, and staff sophistication. After measuring each of these items, you plug in values to the formula to arrive at an estimate for the number of DBAs required. If you are interested in pursuing this metric further, I refer you to the META Group research (META Group, Open Computing & Server Strategies, File: 656, Date: 20-Mar-1998). META Group can be contacted at <http://www.metagroup.com> or by phone at 1-203-973-6700.

DBA Reporting Structures

To whom should the DBA group report? Different companies have taken different approaches to the DBA reporting structure, but a few reporting hierarchies are quite common. Some reporting structures work better than others, so let's review some of the possibilities.

One of the best structures is a data resource management (DRM) group that consists of all the data and information specialist of the organization—DA, DBA, data analysts, performance analysts, and so on. This group usually reports directly to the CIO, but might report through a systems programming unit, the data center, or technical support. Figure 1 depicts a typical reporting structure.

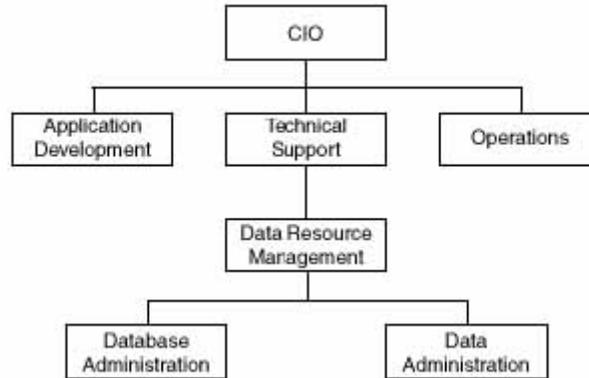


Figure 1: Typical DBA Reporting Structure.

When an organization staffs application DBAs, they will be spread out in application groups, typically with a direct line of report to the business programming managers. Each application development team has a dedicated application DBA resource as shown in Figure 2.

There are problems with both of these reporting structures, though. First, the DRM needs to be placed higher in the IT reporting hierarchy. It's a good idea to have the DRM group report directly to the CIO. When an organization understands the importance of data to the health of the organization, placing the DRM group at this level is encouraged.

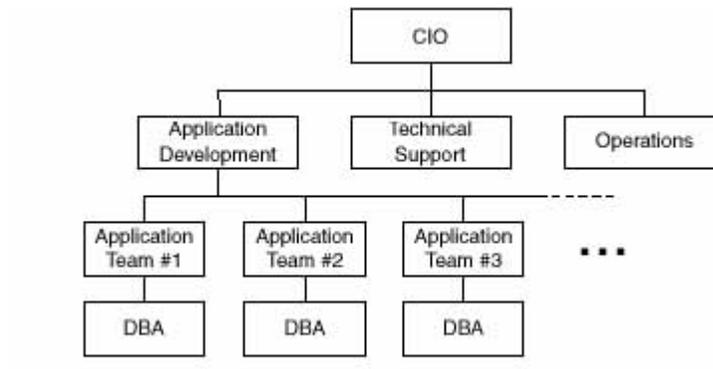


Figure 2: Application DBA Reporting Structure.

Furthermore, when application DBAs exist, they should not report to the application programming manager only. A secondary line of report to the DRM group will ensure that DBA skills are shared and communicated throughout the organization. Figure 3 delineates the recommended reporting structure for the DRM group.

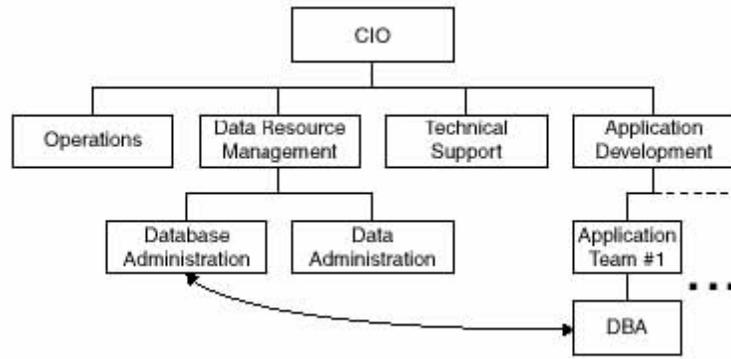


Figure 3: Recommended DRM Reporting Structure.