

How to Get Budget for the Tools You Need

Steve Lemme, CA Inc.

steve.lemme@ca.com

Overview

One day I called the help desk technician to repair my PC after I heard the hard drive start to whine. The tech grabbed an old rusty bent screwdriver and tried unsuccessfully to turn a few screws. He then finally used a pair of pliers to break off the screws to finally get the cover removed. He was successful in getting the cover off, but what has to be done to reinstall the cover? And now he is headed for your hard drive, do you really want him to work on it? Is the tech really skilled to fix the problem or will he just make it worse?

The moral of this story is that there may be many ways to get something done, but not all maybe beneficial in the end. It can be just as damaging for an organization to provide its employees with inadequate technology or no technology at all.

As companies' move towards embracing service oriented architecture application, they must be able to quickly respond to changing business conditions, staying competitive, as well as customer service and satisfaction.

To do so, most companies are not going to completely overhaul their systems, but instead look at better ways to optimize and exploit what they own today. Unfortunately, many organizations are so resource constrained they typically don't have a clear understanding of what they have today. Or even that somebody else, somewhere else in the organization has already is working on, has solved, or has a solution.

The key is to continue to innovate with new technologies, while managing with what's already in place. How can that be accomplished? Some may say more money. Why? Because many believe that money solves everything. Aside from that, it comes down to a business decision:

- What are the issues?
- What are the options to solve the issues?
- Do the cost of the options out weigh the cost of the issue?

This method uncovers the real business requirements, helps assign business value to not meeting them, and prepares a basis for making a business case down the road for future recommendations.

The Dependence on Data

Whether at one location or spread across the globe, distributed relational databases continue to increase in their importance to the enterprise. With this dependency, as well as data growth, comes a significant rise in the complexity of database maintenance and its related dependencies. Database administrators (DBAs) are faced with new challenge of keeping databases constantly online and operational to meet business needs. Increased application usage, coupled with new functionality like web enablement, increases the number of resources required for maintenance tasks as well as the time it takes to do them. In the ever-changing business environment, database administrators can easily be consumed with database installations, followed by another and another and another.

- But what about database health and performance issues after the initial installation?
- Will databases continue to run without any other interaction?
- When can a DBA do maintenance and monitoring work?
- Won't superiors want data to substantiate budgeting and capacity planning activities?

It is too easy for a DBA to become so occupied reacting to current problems that eventually minor problems will get out of control. As a result, small smoldering problems in the database become potentially large raging fires before they can be identified. Daily processes and procedures should be put in place to ensure stability whether manual or automated.

Managing the Data

Today, developers and administrators are wrestling with scattered servers and database systems that have become mission critical, as companies are becoming global businesses. To keep up, organizations have to automate and optimize tasks.

Not long ago, a DBA may have had a single database, platform or vendor. That environment could be easily manageable under manual control and or simple scripts. However today, a DBA faces additional challenges to design, deploy and maintain multi-platform, multi-vendor databases across various networks across the world. Coupled with growth of databases, and a shortage or cost of individuals with these skillsets, automation or tools have become necessary to leverage and retain existing resources and to prevent strapping a DBA to a console ten hours a day, seven days a week. So how do you go about determining what is appropriate for your company? The next sections will review what information needs to be collected to and how to make and support that decision.

Realizing the Business Requirements

Use of technology helps combat the rising complexity and helps reduce the need for hiring additional resources. While total automation may be considered optimal, there will always be a need for human expertise for some things.

In the past, DBAs primarily had to work in a defensive posture, as problems were discovered or only after major issues had already occurred. In fact, some DBAs created additional problems altering database parameters with no real way of monitoring, or tracking the outcome. With automation, DBAs can take a proactive approach in managing their database, reducing risk, as well as increasing consistency and reliability. Instead of having to constantly firefight, administrators can now prevent problems as opposed to being surprised and unprepared. By working with basic automation, DBAs are better prepared for database issues and are able to quickly respond before they become worse. With more advanced technology and tools, DBAs can utilize features like event management and correlation to trigger preventive or corrective actions without the need for manual interaction by the DBA.

To be truly effective, a DBA must understand the business requirements, determine the necessity, communicate, and configure the required database management architecture. A non-mission critical system probably does not require the same architecture as a 24x7 Mission Critical OLTP application.

There are several areas that should to be reviewed:

- Do superiors want data to substantiate budgeting and capacity planning activities?
- What type of items does a DBA need to maintain, monitor, and manage? How often?
- Can the DBA do work across domains or organization boundaries?
- How many database environments need to be maintained?
- What sizes, versions, and, types of databases?
- Are there any monitoring tradeoffs? - Performance impact from SQLNET, ODBC?
- Does the DBA have any other job responsibilities?
- Does the DBA have a backup or additional help from other resources?
- What are the windows for maintenance like database restructuring, space recovery?
- And others. . .

Available Options

To make a decision, the DBA needs to be aware of differences in the strategies: Manual Monitoring, Automated Scripts, Host Operating System Tools, Host Based Monitoring Tools, Client Server Based Monitoring Tools, and Event or Sampling Monitoring Tools.

Manual Monitoring - In manual monitoring, a DBA executes manual queries in SQLPLUS, SQLDBA and/or uses ORACLE Server Manager to monitor the database.

Automated Scripts - At the next level, a DBA automates these queries into daily or weekly scheduled jobs via CRON, batch queues, or other methods.

Host Operating System Tools - Most host systems have a set of operating system tools. For VMS this could be Monitor or Polycenter. For UNIX this could be IOSTAT, SAR, VMSTAT or GLANCE. These tools help identify problems at a system level, but do not address internal database problems.

Host Based Tools - At this point, a DBA may decide upon purchasing a 3rd party tool instead of reinventing the wheel. Host based tools reside on the machine running the database, and connect to the instances on that machine through SQLNET or ODBC. Ok for smaller environments, scalability issues for companies with several DBAs or many databases. They are reliant on a network but also can create network congestion is over utilized.

Browser Based Tools - Browser based tools allows a DBA to maintain a virtual console that can maintain multiple databases on many machines across a network. Browser based tools are reliant and scalable because of their agent architecture. They are dependent upon the network but agent design enables scalability by minimizing and optimizing network and server resource usage. There are a number of 3rd party monitoring tools currently available: event-based, client server, or all browser based.

- Event monitoring tools send messages to counters when events occur. SNMP could be considered an event monitoring tool.
- Client server tools typically involved licensing and installation of client software to a specific users PC
- Browser based tools have no designated PC and provide accessibility as well as collaboration from a web browser from virtually anywhere.

Scripting

Most DBA's have a set of scripts they use to help them in their day-to-day operations. Many DBA's have a number of scripts that kick off periodically to automate certain management tasks – nightly exports, purging, rebuilds etc. Certainly a DBA can even whip up a script on demand or they can be easily downloaded from the Internet. The popularity of scripts is not surprising. They are mostly platform independent and can be quick and easy to use in a specific situation. Most often, scripts achieve the same results as a tool would. Further, tools typically have certain hardware and operating system parameters. For example, Windows NT with at least 200MB free, 100M on the database and 128MB RAM. Scripts often do not require a lot of space or hard-disk and are generic enough to be deployed on most operating systems. Most scripts just need a UNIX shell, SQL*Plus, Windows and can provide some diagnostics about the RDBMS. If there's a problem to be solved, you can probably write a script for it. However, is that the best use of time and resources?

Budget Concerns

Database tools are typically not free; neither are the license and maintenance fees. Purchasers should make sure to get the most out of their investment. And what about training and or consultancy to install and configure the tool? All this can add up to an investment of money; money that some companies say they don't have. However, how much do you pay a consultant to set up their scripts so that they can tell you what's wrong with your system and then charge you to fix them? It is all a matter of perspective.

Learning Curve

Time should be spent getting to know the tool. Some tools may require formal education in a classroom away from work as well as extra configuration for each environment. Additional resources may be needed and scheduled to come and assist with the installation and configuration of the tool. If a tool breaks down, what do you do? Consider buying maintenance so the vendor's support staff can call you back and provide additional assistance should you need it.

Consultant Concerns

Consultants move around companies a lot. Having just a set of tools or scripts does not take care of every single business need. Knowledge of the basics and how to solve problems should be understood because you may not be able to use or rely on tools all the time. There are many people who call themselves DBA's, but just click on button in a GUI to do certain things. Without that GUI they are completely lost.

There can also be installation and software guidelines. Some companies have restrictions on what can be done or not done in their environment. When visiting a client for just a couple of days to diagnose their system a set of scripts can be installed and removed very easily. Not typically as easy with tools.

Fear of the Unknown

What does this button do behind the scenes? Was it written with performance in mind or is it a performance hog? It is easy to configure or can I even configure it at all? With scripts, you see the commands used, and can modify them if required. Consider this functionality when evaluating tools.

Upgrade/Customization

A tool can sometimes complicate software upgrades. If the tool does not support the latest and greatest version of the software available, then you may not be able to move forward with your upgrade. Sometimes you will have to wait for the vendor to get their tool up-to-date prior to any implementation or any other upgrades. Likewise, if you wish to stay on an old version of the software, but the tool vendor decides that the version is no longer supported, you will have to either stick with the current version and risk falling out of support, or be forced to upgrade.

Some tools do not allow any user customizations any you may have to wait for the tool vendor to fix or enhance the product in the next release. For tools that do allow, customization, you need to be careful of what has been changed and ensure that you put the changes back once you have upgraded. If you have written your queries based on some tables, and the vendor decides to change its data model, you may need to re-write!

Job Security

There are some who say that many DBA's don't like tools because it threatens their job. By hiding behind their custom scripts the perception is that the company is dependent upon the DBA to maintain and modify the scripts, and cannot afford to let them go.

Many years ago when the Oracle Enterprise Manager was introduced within the user community, there were discussions of the DBA being replaced by tools. OEM was supposed assist you with the database administration side of things. In other words, be a DBA. Many years later, DBA's are still in high demand. Why? Because a typically DBA's job cannot be defined by a tool, or set of tools. They have many roles and responsibilities that make them an invaluable asset to any company. Tools are there to assist and relieve you of tedious or complicated tasks. They should not be seen not as a threat, but as a way to do more as long as you don't lose those basic underlying technical skill sets.

Many Available Database Tools – Apples, Oranges, and Fruit Baskets

By utilizing database tools, a DBA can be more proactive, and productive. Instead of having to firefight, the DBA can now map out architecture ahead of time and deal with problems before they erupt. Tools are available as point solutions, suites, or in integrated frameworks. The key is to having the right tools to address the needs of the business.

A database administration tool should assist in leading the DBA straight to a problem across heterogeneous, highly distributed environments. Database administration tools help enable users to collaborate on problem solving, manage distributed resources, automate critical management processes, proactively monitor system health, and isolate and resolve problems before they turn critical and impact system availability.

A majority of companies oversee diverse platforms, environments, databases, and networks consisting of Windows NT, UNIX, MVS, Oracle, Sybase, SQL Server, DB2, TCP/IP, LANS, and WANS. Databases are reliant upon operating systems and networks, with availability and performance issues across all three. As well as the majority of companies wanted to increase usage and functionality to

their users while maintaining or increasing response times and availability. To make this possible, the tools must be able to easily work across distributed heterogeneous environments.

Architectures

Primarily there are two types of architecture are used for database monitoring: zero footprint and agent based. There are even a few now that offer both. It should not be assumed that if a tool has either architecture that it is directly comparable to another. In contrast, you can buy a vehicle that provides transportation but a compact car and a pickup truck definitely have different capabilities that are taken into consideration at the time of purchase.

Not all tools are the same, although they may appear to be from the first overview. Some vendor products are multi-platforms, enterprises, workgroups, and a majority are simple client applications that have a pretty GUI, yet execute a lot of backend queries on the database. Many monitoring tools have the ability to standalone, yet they should also integrate easily into a central monitoring architecture or integrate to provide end to end Enterprise Performance Management.

Each vendor has a list of reasons why their tool is better. A whole paper could be written on how each specific tool offering. The DBA must look beyond the vendor claims, and make their own determinations based upon overall business requirements versus one specific workgroups.

Selecting a Tool - Avoiding “Analysis Paralysis”

For some companies, selection of a tool can be as simple as a credit card purchase, or as complex as an RFP and yearly evaluation. Many, companies get locked into an endless cycle of tool evaluations - “Analysis Paralysis” and in most cases; the cost of any tool could have been reclaimed within that initial evaluation period.

If a new product can be used and integrates performing similar functions of many existing individual products, savings on maintenance alone may justify the Return on Investment (ROI) not to mention the dedicated PCs that can be recovered and reused. Other financial benefits can provide the core of justification. Examples of these benefits include:

- Reduction in software/hardware license costs
- Reduction in software/hardware maintenance costs
- Reduction in staff training costs
- Ability to use the same staff on different problems because they can use the same tool with no re-training
- Reduction in costs of supporting the management technology itself

Many tools assist in improving the levels of efficiency of the organization. However, the administrative overhead of some may introduce additional work in the form of updates, infrastructure changes, and configuration management. In some cases that cost is greater than the personnel savings originally forecasted. When making a tool decision, it is often best to look at Return on Investment.

Reduced downtime, increased performance, labor savings, and elimination of time spent on manual methods are a few of the major points to consider when determining ROI on tools. There can be several specific sources of savings:

- Proactive monitoring and administration reduces the amount of time a DBA must spend at a console

- Multi-server monitoring from a central console eliminates additional hardware and personnel dedicated to monitoring individual databases
- Automated reporting eliminates the need to manually check and generate database health and statistic reports
- Centralized GUI analytical tools reduce time spent hunting for scripts and troubleshooting the problems

In-house effort invested in scripts often duplicates existing functionality already in the market. Additionally, the savings in dollars are likely to be offset by the actual cost of labor used to create and maintain the custom tool solution. For a small company this may be fine, but for a large 24x7 operation, configuration management, annual maintenance and upgrades, training, and 24 x 7 support are important factors in deciding what tools to purchase. Database tools help insure against downtime with proactive automated alarming capabilities that trigger before a problem grows fatal. Consequently, DBAs can focus on tasks that contribute positive benefit to the organization, such as proactively working with developers to plan and design database applications more effectively.

Simple Method of Estimating Your Downtime Cost

Assumptions/Calculations Required:

1. Aggregate cost of downtime estimated for your company's business.
2. Estimate the time it typically takes to repair problem and/or reboot system.
3. Estimate cost of personnel dedicated to solving problem.
4. Business interruption costs such as lost business revenue, idle workers and/or consultants, time loss, cost of overtime to recover business process, and cost of outside consultants to solve downtime problems, as well as recovery operations.
5. Performance tool investment cost + maintenance.

What does one hour of downtime cost your company?

Compare the cost of downtime, other costs + personnel costs (database, server, network, Internet personnel) over a one year period to the software tool investment to determine payback period and ROI. Consider all factors which might affect the company including:

- a. Defaults on contracts or penalties based on not meeting service levels
- b. Delayed transaction costs or missed delivery date costs
- c. Lost business or loss of customer contracts due to delays
- d. Other business costs (trucking, shipments) for contracted work not completed on within the required time.

A simple downtime cost estimate can be calculated with this general formula:

Costs

- 1. Hours of operation (24hr operation enter 8736, 8hr enter 2080) _____
- 2. Number of employee's _____
- 3. Average employee hourly wage _____
- 4. Annual company gross income _____

Loss Variables

- 6. Cost per hour (Line 1/ Line 4) _____
- 7. Lost wages (Line 6 * Line 2 * Line 3) _____
- 8. Enter total number of hours the systems are down _____
- Total Downtime Costs = (Line 8 * Line 6) + (Line 8 * Line 7) _____

Coupled with the primary financial impact of downtime are other attributed losses some company's experience¹:

- Lost Productivity - 88.9%
- End-user/management dissatisfaction - 87.1 %
- Customer dissatisfaction - 66.9%
- Overtime - 59.3%
- Lost revenues - 41.8%
- Lost transactions - 34.4%
- Lost customers - 23.1%
- Penalties or fines - 7.6%

¹Contingency Planning Research

Areas of ROI

In calculating ROI, there are investment alternatives based on ROI analysis and cost justifications prior to making investment recommendations and decisions. There are a number of traditional methods that can be used to evaluate and analyze IT investments including:

Cost Savings

The amount of money saved when comparing the cost of investment in tools less the total amount of estimated costs if these tools were not purchased.

Payback Period

The time in weeks, months or years required to return the value of the investment made in purchase of performance tools.

Opportunity Cost Saved

The costs avoided or deferred as a result of investment in performance tools.

Return On Investment

The percent of effective interest rate return on the software investment over a given period, typically one year. In other words, for every dollar spent how much more of a dollar is returned. This comparison is typically done when a company has limited capital to invest and wants to select in the most attractive business solution.

Value-added Savings

Value-added savings are typically defined as “soft” or less quantifiable savings or costs avoided as the result of purchasing tools.

One example of database management tool ROI is the benefits automation. Through automation, managing the enterprise is accomplished more productively. The ability to correlate important system events across disciplines, and execute response rules for corrective action, means that many system problems can be proactively corrected before IT personnel are even involved. Administrators can automate processes that span multiple tools because the tools are able to work together to proactively monitor, manage, alert and correct problems before they become serious. This benefit can be further enhanced in that an operator behind a console may find customer service work too stressful, and as a result sees all events as just problems. The operator doesn't get or care about the “Big Picture” and just logs the problems. This type of scenario would be best automated with tools to add value by becoming reactive instead of proactive.

Additional ROI points of consideration:

- Analysis, Forecasting, Reporting – always anticipate the need to perform different, and often more sophisticated, analyses, forecasts, and reports
- Understand and develop requirements, required functionality, alternatives, anticipated benefits and costs, shortcomings of existing systems and current anticipated timelines
- Does the current solution comply with organization policies, requirements, and procedures?
- Does the current solution meet end-user requirements?
- Evaluate risks, benefits, and costs - comparing the in-house option to commercial alternatives and which makes the most sense for your organization

- Is the current solution consistent with overall strategic direction and business objectives?
- Is the current solution cost-effective?
- Objectively evaluate the build vs. buy alternatives
- Perform the analysis - refine the requirements, thoroughly evaluate the commercial products, and estimate comparable in-house development requirements.
- Timing - how quickly does the solution need to be in full production

Outlining ROI Assumptions

One of the most important steps in calculating ROI, is knowing and determining the proper business assumptions. Below is a list of typical database administration tool assumptions that can be used:

- Allows IT personnel to be more proactive
- Automates repetitive tasks
- Reduces human error and overtime costs
- Saves time due to automation of repetitive tasks
- Reduces cost of system downtime
- Creates faster problem resolution
- Centralizes event management and correlation to provide faster problem determination and correction
- Increases potential hours saved due to reduced failures
- Decreases hours not spent re-running jobs
- Delays upgrade costs due to improved performance
- Determines when and how often a database needs to be reorganized
- Saves disk space due to optimization
- Identifies root cause of failure
- Improves end user productivity due to faster response times
- Increases user confidence and satisfaction
- Creates labor savings
- Decreases man-hours per week writing scripts
- Decreases manually initiation of custom jobs
- Creates proactive management to improve performance and reduce downtime
- Decreases productivity loss due to application availability
- Decreases productivity loss due to poor application response times
- Reduces training costs
- Reduces manual efforts and CPU costs due to improved performance
- Reduces manual monitoring requirements
- Provides single vendor solution combined with standards, policies and best practices to improve serviceability software or hardware upgrades, etc.

An ROI Example

Requirement: robust production scheduling - issues and problems when utilizing CRON

The following are common issues and problems that companies may face when utilizing CRON, the native UNIX date and time driven utility, to perform processing:

- Shell Scripts must be developed In-House - Shell scripts are not easily maintained, and often do not provide for straight forward error-detection when processing problems occur.

- No Inherent Fault Tolerance - Processing is tied to specific hardware devices.
- CRON Lacks Reliability and Centralization - CRON must be set up on each machine specifically for that machine, therefore hard-coding processing to a specific resource.
- Lack of Easily Reviewed Processing Audit Trail - Often difficult to determine “where things actually broke” when processing errors occur.
- Problems in Processing Typically Require Manual Intervention - There is no automatic restart, rerouting, or recovery in error conditions.

Evaluate the cost of IT Activities

- Time required to use CRON to manually perform job scheduling
- Time savings resulting from reduced manual effort
- Time savings from reduced errors occurring during production
- Reduced human errors resulting in costs
- Cost when production jobs fail to complete

Evaluate the cost of lost CPU cycles

- Cost of production downtime due to failures in the existing job management system
- Cost associated with job management system shutdown
- Jobs run out of sequence
- Jobs run with wrong or incomplete data
- Jobs re-run due to operator error
- Unutilized CPU time between manually submitted jobs

Recommended ROI Calculations

- Elimination of production downtime
- Increase of productivity, job optimization and workload balancing
- Automation-exception only processing
- Increase of CPU processing time savings
- Decrease of Lost business opportunity when production jobs fail to complete

Additional Benefits of Consulting and Training Resources

Training is as important as installation and configuration. Leveraging the resources offered by the vendor is a valuable advantage over custom scripts and code. While custom code might first appear to be more cost effective than a purchased product, vendors usually add value with support and training solutions. In most cases, custom code is rarely documented, maintained, and regression tested as rigorously as purchased products. When the resources that developed the code have left, most often has the knowledgebase. However, in the case of a purchased product, other personnel can quickly brought upto speed by way of vendor training and or consulting services.

Conclusion

Almost every organization monitors and intervenes to correct conditions that impact performance. This becomes more difficult as databases increase in numbers, complexity and size. Managing without tools is often difficult and costly with added cost hidden in a DBA's time. Database management, performance management has become quite sophisticated.

With current tool offerings a DBA can now be proactive instead of reactive allowing time for root causes analysis and prevention planning. Tools more economically assist in the identification of problems, alerting of thresholds, corrective action, and as an enabler of event correlation. Correlation of data can

not only give a DBA a picture of what is currently happening, but also what has already happened. With historical information, a DBA is able to create known operating ranges or baselines in which the database should be operating. Deviation from this baseline is an immediate Red Flag for a DBA to investigate.

Productivity can be increased by monitoring and trending database network and system usage patterns. Data collection is the key to capacity planning, forecasting and budgeting. Armed with the information, a DBA maybe able to pinpoint under-utilized systems, possibly postponing acquisition of unneeded costly new hardware.

For mission critical applications, off the shelf tools can be cheap insurance against money lost during a production down situation. Once the business needs have been identified, then the budgeting aspect usually falls in line. You must always be planning for the future. Applications will be changing with the influx of data and business needs.