Quality Assurance in Technical Management

DAVID BADEN
Radio Free Asia
Washington DC, USA

ABSTRACT

"It is not the strongest of the species that survive, nor the most intelligent, it is the one that is most adaptable to change" -Charles Darwin

This paper examines how a Technical Support Group, whether an entire division or a small engineering section, can adapt a Quality Assurance (QA) Program for use in a broadcast organization. Quality Assurance is a management philosophy that motivates the people involved in all phases of the work process to understand and to be actively involved in working to achieve an organization's goals and objectives.

To institute a QA Program is to make a commitment (Assurance) that technical management's primary considerations and efforts will be geared towards positive improvements only (Quality) - improvements that are designed to provide outstanding service on all levels. A QA Program embodies this goal in a comprehensive approach that can be easily documented, is logically structured and most importantly can be audited.

A QA PROGRAM / WHY DO IT?

"Progress is impossible without change; and those who cannot change their minds, cannot change anything" -George Bernhard Shaw

While there is a plethora of ever changing management philosophies, buzzwords, acronyms, seminars and motivational programs, the real question in any manager's minds should be "why change at all"? All organizations by virtue of the fact that they are currently in operation, at least superficially appear to be successful operations. Implementing a QA Program starts with asking the question, "Are we as good as we can be"?

From a complacent manager's point of view, managing to get out of the office at a reasonable time each day is the pinnacle of success.

"By working faithfully eight hours a day, you may get to be a boss and work twelve hours a day." -Frost Robert

Complacency has no place when you establish a QA Program. So why institutionalize a QA Program at all? There are two primary reasons to launch a QA Program:

1- There is always room for improvement whether you are willing to admit it or not. If you sincerely believe that the management processes that you now have in place are the most constructive and as fine-tuned as they can get, then ask around. Ask the technical staff under you if they agree. Ask the user community that depends on the support you provide if it could be better.

Make sure to ask in a way that an honest answer can be given. A simple verbal, "How am I doing?" conversation will not yield honest answers. Conduct a blind survey of your own technical staff first; they are closest to the daily operations. By asking, you have already started down the path towards your own QA Program with two improvements already been made. One – you have changed your mindset to acknowledge that the possibility for improvement may exist. Two - a channel of communications that may not have previously existed has been opened.

2- CYA. A well defined QA Program will not only enhance the quality of the working environment for technical staff members, the user community and improve the performance of all technical systems; it will protect your job as a manager. As the agent of change for a work environment that is committed to and constantly improving, your job security can only improve.

Problems will no longer sneak up on you as the performance of all systems and processes are monitored. Deviation from the established standard performance can be analyzed, identified and corrected before it becomes an overall problem. Decisions will never be questioned or

second guessed as the decision making process has been expanded to include wider input and itself is quality checked against other options and the current systems or processes they are replacing.

Customer studies have repeatedly shown that problems fixed quickly promote greater customer confidence and loyalty than found in customers not having problems. Rather than opening yourself up to greater criticism, the QA Program provides the opportunity to ratchet up the importance and appreciation of your work. Provided you carry it out well.

For whatever reason you embark on reinventing your management process or whatever changes you ultimately decide to implement, the first step towards change and improvement has already been taken by acknowledging the need for a change and showing a willingness to implement that change. However, just because you are finally convinced does not mean others are convinced as well. You will have to pull them along through motivating, convincing, cajoling, and other methods. The hardest part will always be breaking the status quo, or inducing some movement. According to Pettigrew and Whipp (1991) there are three major features of managing change:

1- Legitimizing Change

2- Creating the Capability for Change

3- Constructing the Content and Direction of Change

The QA Program is one way of packaging this effort in an easy-to-sell fashion. A properly executed QA Program will give you the tools needed to customize the work processes within a framework which will drive positive benefits for your organization.

Numerous 'off-the-shelf' management software, packages, standards and plans are available. But simply because one management restructuring initiative was successful does not guarantee that the same initiative will be successful for the needs of your organization. This is why it is essential to tailor any QA Program to your specific set of circumstances before proceeding to implementation. Change must be for the good or there is no point of change. The secondary mantra for any QA Program should be “First, do no harm”.

Once you as a manager have conceded the need for change and grasped the concepts of the process of change, you must assume the leadership role and become the instrument of change.

"I am certainly not one of those who need to be prodded. In fact, if anything, I am the prod." -Sir Winston Churchill

AREAS OF RESPONSIBILITIES

"Everything should be made as simple as possible, but not simpler." -Albert Einstein

Quality Assurance is an approach towards management and work processes whose goals are self-explanatory to its name; the assurance of quality in all aspects of the operation. For any Technical Support Group, there are three areas of the operation that a QA Program should encompass:

1- All Technical Systems that you plan, design, install maintain and upgrade. From transmitters to the wide area network to the local area network to network servers and the applications they run, on to individual workstations and the applications found there.

2- The User Community you support. This includes the broadcast staff and administrative staff of any broadcast organization.

3- The Technical Staff that make it all possible. The people that actually plan, design, install and maintain the technical systems and provide technical support to the user community on a daily basis.

Each area is usually thought of as separate, but changes in the performance level for any one area has a direct and sometimes unconsidered impact on the other two. The goal of a QA Program for technical management is to ensure that any positive improvement in one area does not result in a negative impact on either of the other two areas. The best that can be hoped for is positive change in all three areas simultaneously. The worst acceptable byproduct of any positive change for one area should be no impact or “no harm done” to the other two areas.

As an example, let's consider the purchase of new digital production software, called X-Edit Plus for this illustration. This software might enhance the technical systems' overall performance (faster, better automation, more flexible interface with other systems, etc.), but if not properly implemented it could adversely affect the other two areas. X-Edit Plus may place an extra support burden on an already overburdened technical staff. X-Edit Plus might be more involved to use and cause more work or confusion for the users. Both the users and support staff would view this upgrade as having an overall negative impact from the bias of their own perceptions. In that case, both the users and the support staff will be making their views generally known, usually loudly, though not necessarily to your ears. X-Edit Plus is now becoming known as your bad decision.

Under a QA Program, X-Edit Plus would not have been purchased without the input from both the support staff and the user community. By being part of the discussion and decision-making process, it is possible that their views on X-Edit Plus would have changed to the positive before installation. While X-Edit Plus may still be a support burden for the technical people, they may now view it as new technology and therefore something that will augment their personal professional development skills. For the users, the few extra work steps may seem insignificant to the new features that will benefit their overall productivity- that they would have been previously unaware of.

Unexpected problems in a well-planned migration invite sympathy, while systems "thrown at us" invite scorn. Where lack of information is available, the system as a whole is painted by its problems, no matter how peripheral these are to its main functions.

Kotter, Schlesinger and Sathe (1986) note that there are four common reasons for resistance to change from individuals:

1- Self-Interest

2- A Lack of Trust Coupled with Misunderstanding

3- Different Viewpoints or Assessments of the Benefits of Change

4- A Low tolerance for change

While points two and three can be handled through a methodical, consultative change process, the other two points require enlisting others in the higher-order goals of steady improvement. It is easier to include all parties that have a vested interest in the decision-making process then it is to sell a project after the fact.

With a QA Program, Technical Support management acknowledges and accounts for the interconnectedness of the three areas, systems, staff and users. In recognizing this interconnectivity, the goal of a QA Program, "steady constant improvement", also become their goal.

QA IN ACTION

A QA Program is a self-perpetuating process driven by three independent but interrelated sub-processes:

1- Feedback, Evaluate and Diagnose (Potential for Improvement)

2- Planning and Develop a Vision (Strategy for Improvement)

3- Implementation

Feedback, Evaluate and Diagnose (Potential for Improvement)

Performance processes, systems and services should be constantly monitored, results compared and, when they can be, processes improved upon. Where improvement cannot be made, then a minimum "do no harm" mode is established. A QA Program by nature prohibits things from getting worse. Therefore, one of the key elements to inaugurate the Feedback, Evaluate and Diagnose (Potential for Improvement) sub-process is to know exactly where you are starting from. This requires the establishment of a baseline to measure all future efforts against. It is impossible to plan where you are going unless you know where you are at.

"No matter where you go, there you are"- Buckaroo Banzai

Baseline measurements must be established for all three areas of responsibility. These baseline measurements should be for all aspects of both process and product and must be objectively

quantified and documented by the Technical Support Group. "Process" being defined as the way one goes about performing his/her daily tasks and "Product" being defined as the end results that these efforts actually produced. "Product" can be either a tangible physical "thing" or a non-tangible "service".

Technical Systems

There are several performance measurement tools to baseline technical systems and equipment that a Technical Support Group may be responsible for. Where applicable, these tools should be dynamic monitors that establish the baseline and then constantly monitor the systems. Dynamic monitoring is the real-time comparison of current performance against the initial baseline as well as previous time-periods. Real-time alert notifications can then be received when a system's performance starts to degrade. As a result these problem systems can be identified and corrective measures taken with near real-time reaction. A good example of where the dynamic monitoring model would apply would be with the performance monitoring of a Local Area Network (LAN) or a transmitter.

With dynamic monitoring within the QA Program's Feedback, Evaluate and Diagnose sub-process, small problems can be identified in systems or equipment before they becomes "known" problems. "Known" problems tend to be noticeable when they are systems stoppers or adversely affect other systems and therefore become harder to trace.

For example: Radio Free Asia uses a telephone codec to connect a remote location for a weekly live remote program. Dynamic monitoring of the codec's performance, with tools supplied by the manufacturer, showed that it was experiencing continually longer make connection times and an occasional temporary line loss. Prior to the QA Program and dynamic monitoring, this would have passed unnoticed. There is a hot redundant backup line of lesser quality, but an occasional 5 second drop of the main line would not have been commented on. Since the dynamic monitoring indicated the potential for a problem, the system was run through a full factory diagnostics. These tests showed that the codec did in fact need repair. After repairs, the unit functioned to its original baseline.

As a result of dynamic monitoring the unit was repaired before catastrophic failure occurred. Catastrophic failure would have caused an on-air switch to the lesser quality backup feed line for an entire live program. Under the QA Program the problem was identified and corrected before it became a "real problem". A real problem that would have made the Technical Support Group looked bad in the eyes of the user community.

Other technical systems performance monitoring will be more static by nature, with periodic scheduled tests. Static monitoring would include such systems as PCs and workstations. Baselines are established more for a class of equipment and this baseline is used as the standard performance criteria for the installation and maintenance of individual pieces of equipment within the same series.

For example, if the performance level is baselined for a whole series of PCs (i.e. P-4 2G Hz with 512 K Ram Running Windows XP), individual PCs assigned to the users will only be installed if they meet or exceed the expected baseline performance level for that series of PCs. Once an individual PC meets the expected level of performance a baseline of its performance is documented and stored locally for that particular PC. Performance measurements are checked on a scheduled basis to ensure that the PC is up to par. Performance measurements are taken whenever either a system upgrade is required (software or hardware) or a service trouble ticket is logged for that PC. After any service changes to the PC, its performance is measured once again and compared to the initial unit's baseline. If the PC does not or cannot be made to meet the expected performance level for that particular series (or its former baseline) it will be replaced.

This QA methodology ensures that any changes to the machine either improved performance or left it the same (no harm done).

An example of how the QA Program Feedback, Evaluate and Diagnose sub-process benefits with static performance monitoring happened recently with a workstation upgrade at Radio Free Asia during the upgrade of an upscale CAD workstation. The upgrade on the surface should have been simple, an OS upgrade from Windows 2000 to Windows XP. The upgrade should have been as easy as running the Windows XP upgrade disk. Before the QA Program that is all that would have been done, the disk would have

been run and the workstation reinstalled. With QA Program the performance of the PC was already baselined under Windows 2000 and stored on the machine. When the XP installation was complete the new benchmark showed a 21% reduction in performance to the old. After upgrades to the BIOS and multiple system level drivers, the workstation ended up with a 5% edge over the initial Windows 2000 baseline.

In the past, before the QA Program, this would have caused consternation between the Technical Group and the user as the machine would have been returned with noticeably reduced performance. Without the proper objective measurement tools any complaints would have been dismissed as purely "subjective" and the user categorized as a "whiner".

A list of performance monitoring tools in use at Radio Free Asia can be found at the end of this document.

The User Community

The Feedback, Evaluate and Diagnose sub-process must be inclusive of the user community also. Generally Technical Support Groups provide more to the user community than simple access to computer, studio, transmission, telephone systems and support. In most cases they also provide training, production support as well as a full range of other services that are essentially based on human-to-human interaction. Any human-to-human interaction requires collaboration and coordination.

Collaboration is defined as "a process through which parties who see different aspects of a problem [or issue] can constructively explore their differences and search for solutions that go beyond their own limited vision of what is possible" (Taylor-Powell, 1998).

Coordination is defined as "a process of communication, planning, sharing or resources, risks and rewards for purposes of efficiency and effectiveness in achieving the complementary goals of the parties involved." (Taylor-Powell)

In order for any Technical Support Group to work collaboratively with coordination an effective communications and trust are critical. While user feedback tends to be more "qualitative / subjective" then "quantitative / objective", deriving a sense of the user community's general feelings towards the Technical Support Group is critical.

The corner stone for all successful communications is processing feedback (or listening). The Feedback, Evaluate and Diagnose sub-process of a QA Program is critical for the Technical Support Group as a service provider in order to establish the current satisfaction level for the job it is doing. Without a method to receive feedback and objectively evaluate it, it is impossible to identify ways to improve on relationships or improve on or expand the services being offered.

To baseline the user community's perception as to the job a Technical Support Group is doing, a user community survey must be carried out. While it is easy to baseline and document the performance of a PC, it is difficult to measure the feelings of its end user with regards to his or her experience with the service provided. Was the Technical staff member professional, friendly and polite? Were all the new features on the PC explained or was the user left confused? A user survey allows for a sample representation of how the user community feels in general towards the Technical Support Group. The resulting data from a user survey is to be used as the baseline point to measure future impact on the attitudes of the user community that any changes may bring.

To this end Radio Free Asia conducted two on-line User Surveys during the implementation phase of its own QA Program. There was one survey for each of the two technical sub-groups that have the highest level of daily interaction with the user community. The surveys were HTML Forms that were located on the Internal Home Page (Intranet). The Intranet is maintained by the Technical Division at Radio Free Asia. The two surveys were posted for twenty-one days.

The majority of the questions on the surveys were multiple-choice. There were at least two essay questions on each survey where the users were solicited for their specific opinions on a topic. The questions for the surveys were written respectively by the lead managers of the technical sections they were posted for. Public e-mail notice was sent out on several occasions to the user community explaining the purpose of the survey process and requesting them to please participate by taking the time and filling out the surveys. As an added incentive, two "Free

Lunches" were randomly given away each week as a prize while the survey was posted. Completing either of the surveys allowed a user to be eligible for a random drawing to win a free lunch.

Confidentiality was assured during the survey process. Once the "Submit" button was hit on the survey page, the results were sent as an anonymous e-mail to the staff member collating the data. The e-mail that contained the data could not be attributed to the person who completed the survey. Since the surveys were anonyms to be eligible for the lunch drawing a separate area of the survey needed to be completed and submitted.

There are three important factors to consider when composing your own user community survey.

1- Anonymity is an important. The people must feel that they are free to speak their mind without the fear of reprisal. Most people only feel free to criticize when that criticism is not attributable to them. Additionally, people are social animals – in public, they might provide answers to please or parrot group beliefs, rather than providing valuable individual data.

2- Crosscheck. Place similar questions in different places in the survey. If the answers provided are in sync with each other, then it is a good indication that honest and consistent answers are being provided.

3- Value Return. Most people will not perform what they perceive to be an extra work duty unless they see a return value in it for them. The free lunch worked much better as an incentive to complete the survey than the abstract promise of "helping us to help you".

Once the results are in and tabulated, the data should be shared with the user community as a whole. Accompanying this data should be a list of any short term fixes that can be applied to problems may have been identified. The immediacy of a corrective response shows a good faith and establishes integrity in the communications process from the start, "we asked, we listened, we will fix it". If the user community can see direct and timely results from their input, increased participation will be guaranteed when the process is repeated. Once the user community survey is complete, the data is in place to establish a baseline for the current level of user satisfaction.

It is important to keep the feedback coming. While it is impossible to constantly survey the users, other tools that keep the communications lines open should be considered. After the initial user surveys at Radio Free Asia were completed, they were replaced by on-line "Suggestion" and "Complaint" boxes on the company's Intranet. These were put up to provide real-time feedback from the user community and establish an instant venue for either complaints, general gripes or to simply offer suggestions.

As a result of user surveys, technical managers at Radio Free Asia are now mandated to do more follow-up. A reoccurring complaint on the survey had to do with the users feeling that they were not informed enough as to what problems were fixed after a trouble ticket was turned in. In the past a problem would have been fixed and forgotten by a technical staff member. Today the user is revisited to ensure the problem was fixed to the user's satisfaction and to explain what corrective actions were taken.

Another new process resulted from the user survey. For any outage or interruption of service, an Incident Response Report is issued. These reports follow a template and document all the relevant facts of the incident. More importantly, Incident Response Reports include a forensic analysis, a summary of corrective actions and recommendations on how to avoid the same problem in the future. These reports are distributed to all parties that had a vested interest in the incident. The Incident Response Reports shows that the Technical Support Group is professional and willing not only to communicate and explain events but also inform how we plan to prevent similar events in the future.

The Technical Staff

As stated before, for the Feedback, Evaluate and Diagnose sub-process to be successful within a QA Program it must be all encompassing. It cannot be all encompassing unless it takes into consideration the morale, productivity and efficiency of the staff that is implementing the QA Program (meaning, *your staff*). The positive result of any change is negated if the byproduct of that positive change has an adverse impact on the efficiency, productivity or morale of the

technical staff tasked with implementing the change. Therefore the Technical Support Group itself also needs to be baselined and monitored.

To this end, the Technical Division at Radio Free Asia conducted an Employee Satisfaction Survey for its staff to baseline the culture and morale of its own staff. While both can be measured, it is important to understand the difference between "culture" and "morale".

Culture - encompasses the concepts, skills, habits, art, instruments, institutions, intuition, etc. of a given people.

Morale - is the mental condition as regards courage, zeal, confidence, discipline, enthusiasm, willingness to endure hardship, etc.

Culture represents the atmosphere that allows for innovations to be conceived, while morale represents the will to make the conceptions a reality. Culture is harder to "fix" than morale, as it is more in-grained and subliminal. It is also more difficult to motivate people to think than it is to get them to work. While atrophy affects all muscles, the brain is the hardest muscle to exercise back into shape. When tired or stressed, people will willingly do menial tasks, while tasks involving concentration or consideration are either rejected or done poorly.

The Radio Free Asia Employee Satisfaction Survey was in the form of a word processor file. This file was e-mailed to the entire Technical Division staff. There were a total of thirty-five questions of which eighteen required a numeric rating (scale of 1-10) and the remainder required a written answer. The surveys were completed by the staff and then submitted anonymously to a blind FTP site. In no way could attribution to the respondent be traced back to the individual. In practice, there might be Machiavellian ways to identify respondents, but this would be absolutely counter to the goals of encouraging honest, valuable feedback.

The three above-mentioned survey factors were considered:

1- Anonymity - blind FTP made attribution impossible.

2- Crosscheck - similar questions were in different places on the survey to indicate consistent answers.

3- Value Return - no free lunch this time, but the promise of improvements in the working conditions as an inducement to participate.

The following methodology was used for numerically rated survey questions to gauge if a question was answered overwhelmingly either pro or con. If better than 50% of the respondents rated the question at 5 or below, it was considered an overall negative response. Conversely, if over 50% of the respondents rated the question a 6 or above, it was considered an overall positive response. Using this methodology allowed for a quick overview if any specific area of questions was receiving a majority negative or positive response.

Questions that required a written essay type answer were given an overall rating based on whether a majority of the responses were positive or negative. However, for some given answers, it was necessary to make a subjective judgment call as to whether the tone was negative or positive. Any questions that were not answered at all were weighted as neutral, neither negative nor positive. In general, the majority of questions that were not answered in the Radio Free Asia Employee Satisfaction Survey were those that solicited thoughtful input (e.g. how would you improve a given situation). Unsurprisingly, when given the chance to complain people showed no hesitation write in a complaint.

The questions on an Employee Satisfaction Survey should not only focus on the Technical Support Group itself, but also contain a series of questions on how the employee feels about the organization in general. Often bad feelings towards an organization does not necessarily translate into bad feelings towards the specific department or division an employee works for. Establishing whether such a rift exists also allows the technical manager to sort possible morale problems into two distinctive categories. The first category being problems that are internal to the Technical Support Group. These are problems that you can directly affect as a technical manager. The second category is problems external to the Technical Support Group or organizational problems within the company as a whole. These sorts of problems generally involve issues that a technical manager can only influence by bringing to the attention of upper corporate management.

With the Employee Satisfaction Survey completed, you now possess a baseline against which to measure any future changes in assessing their impact on the staff that is implementing the changes. Having a baseline is good but the QA Program Feedback, Evaluate and Diagnose sub-process also requires a method of dynamic feedback for this area. To that end, some sort of productivity tool is required. A productivity tool tallies the staff's effort to determine how much time is being expended on what projects or work processes.

"Anyone can do any amount of work provided it isn't the work he is supposed to be doing at that moment."– Robert Benchley

There are several off-the-shelf Productivity Monitoring Tools available. Some can be quite intrusive and highly granular, measuring right down to the number of keystrokes a worker enters. There is a fine balance that needs to be struck when implementing a productivity monitoring tool for your staff. The staff being monitored must see the benefits in complying. This tool should not be used to monitor the productivity of one individual to be compared against the productivity of another; that is the job of a good manager. A Productivity Monitoring Tool should not be used to abdicate a manager's responsibility. A Productivity Monitoring Tool should not be associated with the possibility of punitive action or for that matter a reward. The subject of rewards is in itself highly involved, but many studies show that reward systems focused on the individual can effectively destroy group morale and performance. Here the focus is to give staff better tools to evaluate their workload and achievement as an integral part of the work process.

A Productivity Monitoring Tool's purpose is to get a real time indication of the amount of staff resources any one particular project or work process is absorbing. This indication is needed in a QA Program as feedback for constant self-evaluation. Diagnosis of potential for improvement is then simplified. Is a process such as wiring a patch bay taking more resources in man-hours than the cost to buy a patch bay prewired? Next time a new patch bay needs to be installed, you will have concrete data to base this decision on. More importantly, facts laid out in black-and-white make your job as a manager easier. Guiding the staff member through maintenance figures lets the employee reach the obvious conclusions and take responsibility for them, whereas a managerial directive based on observation can often produce rebuttals and disagreement.

A Productivity Monitoring Tool can start out simple. At Radio Free Asia, a spreadsheet was used to link and track man-hours to work done, while a more robust on-line XML database is being developed for this purpose. The staff simply entered the time they spent on a list of projects and work processes in the spreadsheet at the end of each day. This data alone was sufficient to fine-tune several work processes.

Planning and Developing a Vision (Strategy for Improvement)

"If you feel you are in control, you are not driving fast enough" - Emerson Fittipaldi

With the QA Program sub-process for Feedback, Evaluate and Diagnose (Potential for Improvement) in place, the next sub-process of Planning and Developing a Vision (Strategy for Improvement) follows naturally. Planning can now be driven by an intelligent and accessible management process with well defined strategic goals. Instead of applying resources to correct perceived needs, real needs will be identified by hard substantiated data and get priority attention.

For example: a strategic commitment is made that studio digital production workstations will never fall below a 30% performance rating compared to the newest generation of computers available. The baseline performance for the series of computers that are currently used as digital production workstations is already established. The Technical Support Group under its QA Program has already defined and works to ensure the minimum performance level for these computers. As these levels remain static, real world variables will constantly be scaling upward as overall computer technology improves. New generation technology performance levels are public, widely known and predictable. Your vision and strategic plan has already made a commitment to users that studio digital production workstations they work on will never fall 30% below the performance level of current technology. With the predictability of technology, corporate management has long term planning information available. It knows not only when resources will be needed to be allocated, but why.

These same QA sub-processes can be applied across the board in a broadcast organization to develop a vision and strategy for improvement. The processes can be applied not only to the tangibles (technical systems such as network equipment, studio equipment, servers, transmitters, etc.), but also to the intangibles such as work processes and morale.

At Radio Free Asia, by using the feed back from the technical staff and the user community two areas of weakness became apparent. One, the users were unsatisfied with the level of Help Desk support. In evaluating the complaints, their dissatisfaction was with support in after-hour periods when no Help Desk staff on site. Two, Master Control staff felt as if they were not getting enough cross training. More was going on in the organization then they knew about because it was outside their normal job function. The Master Control staff felt dead-ended.

These were two issues technical management was previously unaware of in what might seem unrelated areas. But the solution became self-apparent. Master Control technicians received cross training. They willingly learned basic network and PC trouble shooting skills. They did this eagerly as they saw the process as improving their skill sets and therefore their marketability or chances for promotion. At the same time, this fixed the after-hour support issues for the user community. Master Control is staffed 24/7. The small problems users had in the past, such as a printer not responding, are now fixed by a Master Control technician. Before the QA Program these problems would have never been discovered, much less fixed. Technical managers would have been too busy doing what they perceived as "their jobs" to think to ask. The QA Program has incorporated feedback into the system.

While there will not always be two birds to kill with one stone, the QA Program itself facilitates "Planning and Developing a Vision (Strategy for Improvement)" by recognizing the interconnectedness of all areas. As more and more facts actively are delved out and cross-correlated, planning is no longer based off biased perceptions and assumptions.

In addition, by having standards for maintenance and upgrades based on concrete performance data, you have more time to bring all vested parties to the table in order to participate in the decision. Corporate management will know when a system should be replaced and a budget needs to be allocated. With communication tools in place under your QA Program, the technical staff as a whole can be brought in to propose viable options. These options can be narrowed down and discussed with the user community. While technical managers can filter the choices at every step, based on available resources and "judgment calls", an inclusive process can still be created that allows everybody to feel as if they have ownership in the final decision.

The systems and processes slated for improvement in a QA Program with the Planning and Developing a Vision (Strategy for Improvement) sub-process are now aligned with real user and/or system needs. Available resources (both in human and financial terms) can now be more properly allocated to affect the priority projects. These will no longer be intuitively perceived needs but needs that can be documented and justified based on hard data in an overall process that follows a strategic plan.

Implementation

Implementation is the act of doing. Under a QA Program what changes when Implementation becomes a sub-process is "Communications". All vested parties are aware of whatever changes are being implemented under the QA Program, as they played a role in the decision making process. The added benefit is that now new communications channels are open, they should be utilized. Feedback from your staff as to the current status of any project should now be free flowing. As the project progresses the user community will become a new ally as these new channels of communications you have created notify you to any shortcomings either real or perceived.

Before any implementation is considered complete the in a QA Program performance must be baselined. Is it up to performance expectations? If it is up to expected performance levels then can it be improved? In the self-perpetuating QA Program you are now back to sub-process one- Feedback, Evaluate and Diagnose (Potential for Improvement).

CONCLUSION

"One of the symptoms of an approaching nervous breakdown is the belief that one's work is terribly important." -Bertrand Russell

Under a QA Program, all baselines are used when planning any changes to processes, systems and equipment. By testing and benchmarking any changes under consideration and referring the results back to the original baselines, more intelligent choices can be made. Improvements to technical systems can be quantified. Relevant cost/benefit models can be derived. Decisions will be backed with more comprehensive justifications.

A QA Program allows the technical staff to be more committed now that every step of the technical support process is backed by conformance checks which promote an environment where hard work and excellence is expected. Additionally, the staff "owns" the process by having the tools, data and background information to properly solve problems.

The user community is assured that as a Technical Support Group you are committed to not only maintaining the current level of support but constantly improving the users' work experience. With open communications and clear justifications of any changes, users can better perceive needs and benefits, and competition for limited resources will be reduced. The users see that your interests are their interests.

As a manager, your position is better secured, as you are able at any give moment to pass up the food chain an accurate snapshot of where the technical facility is now, a projection of where it will be in the future, and the vision of how it will get there.

APPENDIX MONITORING TOOLS AND METHODOLOGY IN USE AT RADIO FREE ASIA

Digital Audio Servers and Workstations

Winstone Ziff-Davis benchmarking software containing system-level, application-based benchmarks that measure a PC's overall performance.
http://www.ziffdavis.com/

PC Workstations

Passmark Performance Test is used, which allows for the objective benchmarking of Windows based PCs using a variety of different speed tests. Passmark Performance Test provides twenty-seven standard benchmark tests. Performance Test by Passmark is used as a baseline for each series of PC and with separate benchmarks for each type of OS in use. This baseline is used to pass-check each new PC assembled before installation.
http://www.passmark.com/

Networked Applications Servers
Benchmarked by timing network transfers of a 1 MB file 50 times and a 100 MB file once.

Server Status and Services

All servers are ping-tested once every 5 minutes to check if they are "alive", and all major services (http, windows file sharing, ftp, ssh, etc.) are checked at the same interval. This is done with Nagios, an open-source monitoring package. Other networked services such as printers, hubs, routers and switches are being added. The ability exists to create different notification hierarchies so this can be extended to include other support groups.
http://www.nagios.org

Linux Servers Disk Input/Output (I/O)
The Linux hdparm utility is used with the -t flag pointed at the complete disk. The test is run multiple times on each disk to arrive at an average performance rate.

Linux Servers Uptime
System startup time is recorded in /root/changelog anytime a Linux server is rebooted.

Linux Servers CPU Performance
Kernel compilation is run and timed multiple times. To return a kernel source directory to its pristine state, the "make mrproper" command is used amongst other things; this will remove the .config file used to configure the kernel, so the next compile will return to the default options which the kernel maintainers set. This will ensure that a consistent kernel configuration is used when benchmarking and thus measure the same test (since different configurations of the

same kernel can differ dramatically in compile time).

LAN Status

The network is monitored using Inmon, an SNMP monitoring interface to the Network Switches. This gives real-time utilization, status and configuration information as well as historic reports.
http://www.inmon.com/

WAN ISP Performance
Level-1 e-mails weekly reports of the following: a Router Summary report spreadsheet, a Bit-Rate/Utilization report-text report with timeline graph, and a Utilization Detail spreadsheet report of 15-minute samples for the week.

Web Performance
Web statistics are collected weekly from all web servers using Webalizer, an open-source web statistics collection and reporting tool.
http://www.mrunix.net/webalizer/

Web Audio Archives
All programs are checked for silence and to verify that MP3 has been correctly encoded as written to the archive. A mail is sent to the support staff if a silent period of more than eight seconds is detected.

Remote Access Server (RAS) Usage
All RAS logons are logged and total times tallied and mailed to administrators daily.

IS Port Security
All packet rejects on all servers are logged to a central server and analyzed for patterns. These are tallied monthly by counting the number of complaint mails sent to ISPs.

Intrusion Detection
A third party vendor provides additional monitoring of intrusion attempts on the LAN, while e-mailing to a special account. This number is tallied monthly.

Mail Server Virus Statistics
Anytime a virus is detected coming in to the mailserver, an e-mail is sent to a special account. These are tallied monthly.

Telephone Usage
Call Accounting PBX utility tracks all inbound and outbound calls to the PBX. Outbound International is tracked using AT&T's Business Direct billing CDs and an in-house developed Access database.