



GET REAL:

An Executive Dummy's Guide

to

IT Quality and Security

Version 1.1

Barbara Lockwood, csqa, cisa, cism, cissp

QualityIT

September 5, 2005



An Executive Dummy's Guide to IT quality	3
1. Quality isn't a catchword, its THE KEY	3
2. Quality Systems are Secure Systems	5
3. Put Information Security Assurance in perspective.....	5
4. Treat Internal IT Projects as Commercial Transactions	8
5. Encourage Team shared vision	9
6. Test the Requirements.....	9
7. Test security first, not last.....	11
8. Apply "Sufficient" standards	11
9. Reduce documentation redundancy	12
10. Remove impediments to productivity	14
11. Empower QA as an Independent Management Resource.....	14
12. Align QA with Internal Audit and Corp Risk.....	15
13. Engage a qualified Information Technology Monitor.	15
14. IT Managers	16
15. Project Managers	17
16. Pull, don't push projects to success	18
17. Software Developers.....	19
18. Get HR out of the IT hiring loop	20
19. Vendor Software: Up your BS meter.....	20
20. Align Networking, Data and Communication Divisions with Enterprise Development	20
21. Apply QA to Networking/Communications/Data infrastructure.....	21
22. Focus on the core essentials	22
23. It's the culture, stupid!	23
24. Hire Career Professionals for Key IT positions.....	23
25. Stop Costly Tool Mania.....	24
26. Control Training Costs...and get the most out of them.....	25
27. Begin Grooming a Chief Monitoring Officer	25
28. Demonstrate Quality Values and Virtue	26
29. Back quality words with quality actions	26
30. Why listen to me?	26



An Executive Dummy's Guide to IT quality

Is your IT on track or just spinning it's wheels and costing you money? How do you know? Mainstream corporate business is by far the overwhelming market sector engaged in software purchasing, customization and development. Yet to even those of us way down here in the trenches, the quality and efficiency of mainstream American corporate business IT looks pretty abysmal.

Why is it so hard to get anything done? Why do projects get launched, funded, overrun their costs or never get completed? Why are people building to Byzantine legacy applications that no one any longer understands? Why isn't the user—you know, the guy who will actually use the product—consulted throughout the project? Why are PMs allowed to walk away after completion of a project, leaving the project team holding the bag for their management mistakes? Inundated with the variety of technological means we have at our immediate disposal to communicate, why don't we? Why can't anyone fathom how much IT downtime costs his company when a virus slips through? Wow, what a mess.

IT is a complicated business, but not so complicated that you can't understand it. IT is also not so complicated it can't be fixed with a little brain power and common sense. If you feel in the dark about your IT...don't know where it's going, what it's up to and up against, then this article is for you. Here's a checklist of common sense approaches meant for the Business Executive that is savvy about their industry but largely clueless about the technology on which it depends.

1. Quality isn't a catchword, its **THE KEY**

If you want to save money now and still be in business by the end of the decade, read on. Nowhere is the notion of Quality more timely and critical than for the IT that supports your company. Your Quality Assurance division is not the purveyor of Quality in your company. If a Quality attitude does not permeate all aspects of your corporation right up to the top of the chain—then there is essential work to be done. This is no longer opinion. This is historical fact.

During the 1940s and 50s, American industry failed to recognize the genius of W. Edwards Deming, now universally regarded as the founding father of Quality Assurance. This oversight nearly lost the American automotive industry to the Japanese during the 1970s. Congress quickly set up the Malcolm Baldrige Award program to encourage attention to quality services and products in American industry. Over the following decades America rose to compete admirably with external markets that could often provide comparable products at a lower cost.



During the 1990s, however, America once again set aside Deming's common sense principles aside in a rush to cash in on the Internet craze. We're now feeling the effects of that folly, the stock market still reeling from the demise of countless dot-coms whose investors never stopped to ask "Just how will this make me money?" What sealed their demise was not only the lack of a viable profit model but poor management, sloppy application design, and insecure coding that ultimately made their business applications too costly to fix and support.

The technological advances that came out of that period can't be denied and global business is richer for them. But what they also did is undermine and obscure the rational tenets of sound business and software development practice. What was technologically plausible and challenging superseded what the user really needed and what the company could support and profit by. Common sense was cast by the wayside, along with verifying user Requirements, protecting against security threats, considering maintenance costs, quantifying ROI, measuring worker productivity and tracking software success in production. Instead, the cowboys ruled.

But the fog is now lifting. It is clear that only those companies built on solid business models that emphasize quality applications that support quality customer service and products—companies like Amazon and Expedia--will survive the dot-com crisis. The same is surely true for mainstream business.

With the onset of the global Internet comes availability to universal professional knowledge that threatens to undercut American commercial goods and services once again. Corporate executives must recognize the strategic balance between the bottom line and that of competitive quality if they intend to survive in the 21st century global marketplace. In short, if quality goes in, the name goes on.

The computer industry has become the backbone of corporate business. Few companies can exist, let alone thrive, without it. But it is still a young industry—maverick and uncontrolled. Corporations must choose between academic graduates certified in decades old technology and ambitious hobbyists and hackers—often lacking in discipline and common business sense--that have become experts in skill areas immediately needed.

The choice is obvious, but dangerous. Without an IT infrastructure that continually enforces and measures skills training—in communication, research, efficiency, quantitative analysis, motivational management, and best practice principles and methodology--corporate executives cannot judge or accurately reward workers for the quality of their work in ways that will engender loyalty and assure optimal return on human resource investment.



The good news is that those people who have acquired these skills are getting tired of working with those who have not. The industry is beginning to sift out to reveal those employees just in it for the money and those who are truly dedicated to increasing your market value. Their passion will support and fuel your business in the long term. The rest you can live without.

Here's what every high level corporate executive needs to know about IT as well as what they can do about it.

2. Quality Systems are Secure Systems

Dave LeBlanc, Microsoft's chief threat modeler, asserts this in the opening pages of his and Michael Howard's book Writing Secure Code (Microsoft Press, 2002). Microsoft's recent public commitment to "trustworthy" computing placed this book in the hands of every Microsoft employee, and placed security as the number one priority, over new feature development and enhancements going forward. The book introduces the STRIDE threat model, naming Spoofing Identity, Tampering with data, Repudiation, Information disclosure, and Denial of Service as the current threats to be avoided. Whether these can affect your company depends on how your IT is structured and what type of application or system you need built. However, including threat modeling as a key activity in development projects makes good business sense for other reasons. You can't seriously think through these issues without completely thinking through your application at design time. Therefore, making Security a top priority for your firm has the added benefit of reducing ambiguity in your application and system designs. This further clarifies what will be built before coding begins, thus saving costly rework after coding begins.

3. Put Information Security Assurance in perspective

A word about security: the more the better. But how much security is enough?

First off, you will never be 100% secure, in the same way that no amount of locks, and bars and gates on your house, will every keep a determined thief from gaining entry. Therefore, investment in security should be based on your Security Risk threshold. Your security risk threshold is exactly the point where the cost of implementing and supporting preventive controls exceeds the loss you would experience from a security breach. OK...sounds good...where's that?

Well, it moves around. To put it in perspective, calculating Security Risk is probably 10 times as hard as calculating Business Continuity Risk due to Disaster. Unlike the handful of disasters a company might face—tornado, earthquake, fire, flood—where there is ample available industry data, few



companies are willing to share the actual cost of their security breaches, which might lead to a loss of public confidence in their company or, worse yet, litigation. Without solid industry benchmarks out there, it's hard to pin down what your reasonable risk threshold might be. So companies either ignore the problem or overcompensate.

There is danger in either course--serious danger. A security breach that compromises private customer data—such as health, identity or financial information--can be used to trigger the Sarbanes-Oxley Act of 2002. Born out of the Enron debacle and insider trading scandals, in the hands of able counsel this Act can subject Executive Management to stiff penalties—even jail time—if the investigation reveals malfeasance.

On the other hand, you can go the other extreme and turn your company into a virtual security fortress of firewalls and DMZs, filtering every packet that goes in and out of your company using costly intrusion detection and packet filtering programs that slow your business to a crawl, and uncomfortable policies and procedures your workforce actively works to circumvent. So where's your threshold now?

Unless you have a good solid sense of the value of what you are protecting, you won't know how much money to apply to keep it secure. "Code Red" meant a whole different thing to a Healthcare company, whose MRI's results are conveyed to a SQL server in real time, than to a manufacturing company that could live without its inventory database for a day or two while it was being restored.

See what I mean? This stuff is hard. You can't ever be truly secure. You can't ever anticipate and plug all the holes. But you can be realistic in your approach: Here are a couple of common sense tips that will help you stay safe and maybe save you some money:

1. Don't shoot the messenger. An employee that immediately reports that his machine was probably the point of entry of an inadvertent security breach shows courage and integrity. He's just saved you a ton of money in analysis. Don't penalize him. Thank him.
2. On the other hand, do make employees personally responsible for the security of the company equipment under their care. When a clearly careless employee circumvents security policy, exposes his corporate laptop to un-trusted internet sites without good reason, or opens mail with attachments from unknown sources and releases a bug into your environment, then clamp down hard. This goes for your Senior Executives and VPs too—among the worst offenders, and the people you should expect to set the example. Make it hurt. Publish their names on a



corporate “Doghouse” list. Dock their bonuses. Trust me--people up and down the chain will begin to understand that your company doesn’t mess around when it comes to security, and that they are as responsible as the Security Operations team or Vulnerability Response group when it comes to securing company assets.

3. Don’t reward your network operations team for working overtime to contain a virus—until you prove they were not the ones supposed to be responsible for maintaining the controls that would have prevented it in the first place. For example, they should have a really, really good reason why a patch for a known technology vulnerability wasn’t applied. Distinguish between whether their claims of lack of time and resources don’t simply mask the inability of leadership to properly fund and prioritize work.
4. Don’t fire your security technicians when a critical security task is overlooked—not yet, anyway. Their manager is the one that’s responsible for assuring the quality of their work. If policies are in place to assure separation of duties and check the quality of line work, then why aren’t the policies being enforced? Who knew Eli Lilly would pay more than a hundred thousand dollars to settle a class action suit over a software program flaw that placed the names of private consumers in the “TO” field rather than the “BCC” field, where they would be hidden from each other. To their credit, the company produced records that showed they had appropriate security policies in place. Unfortunately, this was effectively countered when it was shown the policies in place were not being enforced. You might fire the technician, but you still have a management problem that sooner or later is going to bite you.
5. Don’t put all your money into security perimeter controls—firewalls, IDS, etc. Only 5-8% of all security issues result from external hacking. The overwhelming cause of security problems occur internal to your organization, with 65% the result of errors and omissions. 65%!!!! These are software design flaws, system mis-configurations, and errors in support and workflow process and procedures. 13% are about mischievous employees possibly in it for personal gain, and another 10% are disgruntled employees in a position to do you real harm. Fix those!!! Investing all your energy protecting against external breaches is like putting all your cannons on the ramparts, and nobody’s watching the keep! Take a “Defense in Depth” approach—multiple layers of security protections that include controls at the perimeter, system, and application levels supported by solid security policies and procedures. In other words, put reasonable investment into perimeter protections to keep the hackers and script kiddies at bay, but put the bulk of it on cleaning up your own house.
6. Nor should you jump on the “flavor of the month” security solution (intrusion response teams, security awareness, secure coding education, etc. etc.) Security is an enterprise quality attribute and a familiar business



- risk. A responsible solution addresses the constellation of security efforts that should take place across the enterprise, balancing funding for each approach to achieve enterprise security objectives based on assessed business risk. These efforts touch every facet of your organization, including hiring practices, governance policy, standards enforcement, skills training and many others. Decide where your worst risk is (people? process? technology?) and address the solution as a holistic enterprise effort, not a piecemeal stopgap. Root cause analysis on your breach history will likely reveal where your most significant vulnerabilities are.
7. Don't freak out the next time you get hit with a virus that interrupts your business and damages your systems. Freak out when your damage is greater than that of your competitor.

So what is your Security Risk threshold? It changes in response to the changing threats, changing protection technologies, changing business goals, and the changing value of your assets. That makes it hard to pin down. Probably the closest you can come to quantifying it is a gross interpretation of tip # 6. Your security risk threshold is the level at which the potential damage to your company is the same or less than that of your competitor. In other words, when everyone is hit, it's written off in the trades as an act of God. When you are among only a few companies hit, you can be sure you will be unfavorably compared in the trades to your more proactive competitors. The cost to you will be way more than the loss of productivity or data. It will be the loss of public profile and consumer confidence in your infrastructure and services. Can you afford that?

4. Treat Internal IT Projects as Commercial Transactions

The key reason internal IT projects go off track is because the role between the players becomes fuzzy. If a business unit wants, and is willing to fund, an IT solution, the IT division feel obliged to respond—even when the solution doesn't seem sensible or is not technologically cost effective. That's a recipe for disaster.

Make all IT Projects internal commercial transactions subject to the same constraints as would be applied if you were selling the product to an outside company. In such case, your solution unit wouldn't accept the contract until the buyer supplies a minimum level of detail of the problem sufficient for them to be fairly sure they can deliver a successful product. They would decline an unresponsive customer and cut their losses. Your IT needs to be empowered to supply a checklist of required information be fully addressed within a specified period of time before accepting and committing to a solution deliverable. They should have the power to decline an internal customer and move on to the next without political consequence.



By the same token, the customer has rights. They should expect the team to sell them on funding each of the three key phases of any Software Development project. They should fund the Requirements Phase when they are convinced the Project team understands how to go about analyzing and quantifying the problem. Phase 2 funding should be provided when they are convinced the team fully understands the business problem and has a viable approach to a solution. And they should fund the remainder of the project based on a comprehensive Functional Spec review that convinces them that not only do they know what specific tools and technologies will be used, but how they will assure the product's quality when built. Your Software Development efforts will begin to make sense to you—and to everyone involved—when you start treating your internal IT projects as commercial transactions between a customer and a vendor.

5. Encourage Team shared vision

To create a viable product, the team must maintain a shared vision of the outcome. Review sessions are helpful in realigning team vision. Formal Business Requirements Reviews, for instance, include all affected parties—the user, business management sponsor, the Business Analyst, Developers, Quality Assurance Analysts, Infrastructure, Training and Logistics personnel. The consolidated brain force of this body can root out costly, unproductive, and absurd approaches that miss the point.

Unfortunately, these are the first to go when projects begins to slip. This is a PM call, and a bad one. Worse yet, as vision begins to unravel and chaos begins to set in, inexperienced PMs impose restrictive communications controls that can prohibit members from direct access to the people who hold the information they need to do their jobs.

Defined communications paths are needed to control unproductive chaos. But restrictive, hierarchical models of communication are thoroughly unproductive in IT projects. No member should feel they cannot pick up the phone and speak to anyone on the project—from the corporate financial sponsor to the end user, if they aren't convinced all the bases are covered, or if they can't obtain the information they need to do their jobs through the usual channels. Good projects happen when everyone knows what's going on to *their* satisfaction level. Stop playing telephone! Good IT is about sharing one brain.

6. Test the Requirements

Standish Group studies have revealed what is often referred to as the “1-10-100 Rule.” Problems rooted out during the Planning Phase that cost \$1 to fix, will cost



\$10 to fix during the Construction phase, and \$100 or more to fix after the application is deployed. When you're talking about a foundation issue that affects fundamental system layers like that of the core architecture or key technologies used, you might be talking hundreds of thousands of dollars to fix, not to mention the loss in business productivity. Why take that chance? The most powerful means of insuring that what gets proposed is feasible, efficient and cost effective is to "test" the Requirements before Design. The idea here is, if a Requirement can be quantified, then it can be tested. If it can be tested, then not only can it probably be built, but its quality can be assured.

Your Business Analysts are probably members of the Project Team whose job it is to root out IT opportunity, and sell the vision of automated solutions to your business lines. These people are usually experts at understanding how your business works and how to express solutions in Business prose the customers can understand. Often they are not sufficiently skilled or experienced in then translating their prose Business documents into precise technical terms the Developer can understand. And this is exactly what is needed if the Developer is to build what their customers want.

Your Quality Assurance Analysts can act as translators in this process. They speak both languages and can apply their trouble shooting skills to translate the prose Business Requirements into precise technical form. They then extrapolate inferred requirements and apply checklists and standard system categories based in Function Point complexity analysis or other similar means to expose hidden Requirements and risks that might have been overlooked by the Business Analyst. No Requirement is accepted until it is concise and specific enough to be tested. Thus, a body of precise technical statements verified by the Business Analyst emerges from this analysis, which then gets handed over to the Devs. This insures that what is built is built correctly and reflects exactly what your user needs. For every ambiguity or missing Requirement they find, you just saved some substantial wad of cash.

This has the added benefit of engaging QAs early in the game, giving them an in depth understanding of the product they will need to test at the end of the process, when the schedule is usually crunched. It gives them a head start on test planning, and a core set of precise expected results against which they will measure progress and success to provide you precise metrics that provides concise status and help control costs.

If you feel your QAs cost you too much, it's probably because they can't quantify what they are testing against and therefore never feel done. Ad hoc testing is a good way to explore an application, but if your testers aren't measuring against a precise, documented set of specific expectations, they will keep on doing redundant testing until deployment day. Their testing must be systematic and



goal driven. They must have something to measure against, if they are going to measure at all. With a pre-defined set of expectations, they can measure progress and prove success against your pre-defined success level, thereby assuring optimal cost efficiency and product quality.

7. Test security first, not last.

Operating in the internet world requires a shift in priority and emphasis in the testing model. When applications and systems were free standing and internal only to the company network, security and performance, testing could be left to last. Now that your networks are equipped with outward facing portals, allowing your employees to access data from beyond your corporate walls, the priority must put the highest priority on Security testing. Putting anything before your customer's wants and your business needs is a pretty radical idea, but consider this: however cool it might be, would you sanction any application that could potentially stop your business even for five minutes? Would you authorize any action that could potentially leak confidential information about your business? Would you feel comfortable if your customer's private data could be exposed to the outside world? Where will your company be after it becomes a disparaging security headline? Think long term. Be smart.

8. Apply "Sufficient" standards

Standards have become a dirty word in IT. This is a legacy of the cowboy IT of the 90s when to get qualified IT people to work for you, you had to bend--if not do away with--common sense business and management principles. Them days is over, folks. If your managers don't have the courage to set reasonable standards, divine realistic policies, and determine and monitor business sensible IT procedures, get them out of management. Management *is* measurement. It's not cow towing to maverick developers or under-qualified Business Analysts. It's not bending to please your business units at the expense of the integrity of your IT infrastructure.

Standards need not be an impediment to productivity, if applied correctly. However, you can expect that the more fast paced your IT environment is, the more—not less—controls are needed. The more chaotic it is, the more checkpoints and oversight should be applied. People will tell you they needed to cut corners, or remove some activities from the process. Nonsense. All the basic cycle milestones can be hit "in-process" if not formally—and most *must* be hit if a product is to deploy at all. How well your system or application performs in production is not a measure of *what* was or was not done on a project. It is a measure of *how well* these essential activities were done.



The key to process efficiency is only applying standards where they are needed. If your people are measuring project and fault trends consistently, you should be able to compare the workflow and deliverables attributes of successful projects against those were not, and determine the “sufficiency” tolerance you should establish for your IT projects. Rather than apply a standard across the whole of your IT by corporate policy, consider applying standards group by group. One team that is keenly in synch might be able to produce excellent software quickly with a minimum of documentation during the process, while another needs to document everything in detail, then apply formal inspection to every documented deliverable. If a team consistently does high quality, highly successful work, with a minimum of interpersonal grief and a maximum of return on investment, why slow them down with encumbering activities that don’t add to the project success?

It’s rare you’ll come across such a team. And if you think you do, you must prove it with the consistency of their metrics—both during and after deployment. An application that is built with insufficient documentation is probably unsupported in the long run. Don’t rely on key Project member opinion. The best way to find out how effective the team is, is to ask the last person on the assembly line what they think. Your lowly testers are likely to give you a different story. They are least empowered and the most burdened by team problems resulting from non-standard process and deficient team deliverables. Simply ask them if they are comfortable that they obtain all the information they need to do their jobs, and are given what they believe is the appropriate amount of time to do it in. If the metrics don’t correlate with your Tester’s opinions, then one or the other is not telling the truth and controls need to be applied.

9. Reduce documentation redundancy

What’s the difference between a Bug, a Test, a Feature, and a Requirement? The answer is...nothing. A bug is a failed test. A failed test indicates a failed feature. A failed feature indicates a failed Requirement, and a failed Requirement contributes to a failed product.

Then why is project documentation scattered everywhere? Bugs are in a Bug database, the information about your production problems in a Help Desk system, your system change requests are in Outlook emails, your tests in Excel or some other Test management tool, your Requirements are in Word docs, your Design docs in MS Source Safe, and your status reports on someone’s desktop?

These Software development deliverables are all part of the same continuum. They are different sides to the same box. Few people in IT get that. If they did, multi-million dollar companies like Rational Software who offer integrated



software development suites wouldn't be in business at all. You can reduce documentation redundancy by keeping it all in one place. If your people aren't comfortable keeping it all in one form, then demand information migrate from one tool to another, and the information in the former tool retired. You want your teams using, one, single verified interpretation of the business problem and solution, rather than having 6 people wasting time changing 6 repositories every time a change happens in the system.

To insure this, you can buy those vendor tools can cost you hundreds of thousands of dollars a year in licensing. Or you can make your people understand this core concept and use the power of your desktop office suite, common sense and workflow management to insure that your team doesn't waste your time and money.

You can get some clues as to how clueless your teams are by simply asking them to give you the single location on your LAN where you can find all the current documentation—the test plans, design docs, change control requests, problem resolutions, production performance reports, and business continuity plan for any one of your Enterprise application systems. Then compare dates. If you don't find all docs bearing the same approximate last modified date, then not only is nobody really managing this application, everyone involved has a different interpretation of it. The smallest change in the system is going to affect all of these areas, and the documents must be updated to reflect the current state of the system. If they're not diligent about this, before long nobody remembers how the system is built, works or should be tested, so you're building to a black box going forward.

An exhaustive test plan, validated by Requirements trace-ability is probably the best, most comprehensive interpretation of what your system actually is and does. If your team can consider it the vehicle for change control, than other forms of interpretation can be retired after deployment. There is no reason why your teams should be supporting redundant documentation for your systems. If multiple interpretations are being used, all documentation should map back to a single definitive source, whether that be a master Word document, a database or some spreadsheet. Problems in production can't possibly be analyzed or fixed quickly if the reference material is out of date and scattered around on people's desktops. Make sure that documentation inspection and consolidation is being done to complete your projects.



10. Remove impediments to productivity

The chief reason IT projects take longer than they should has something to do with insufficient planning and troubleshooting. But that's not the biggest obstacle. The biggest obstacle to productivity is timely access to critical human resources.

Timely information is key to productivity. Offering your people flex time and remote work access can be beneficial to a corporation. Some creative people work better and more effectively away from the office or from 10pm til dawn. There's nothing wrong with that. But too often work gets handed off for the next stage and people disappear. There are vacations, conferences, training, braces, for the kids, maternity leave, etc. When the person now doing the work has to wait until tomorrow or next week to get the answer they need, schedules slip.

In this day and age of wireless communications, with all the pagers, cell phones and mobile computing devices, there is no reason whatsoever for your employees not to be reachable when they are needed. If you are going to allow amenities like flex time, floating holidays and remote computing, then you have a right to demand something in return. Planned work should happen when scheduled. When it doesn't, people waste their time and your money. Such slips don't just affect these people, but all the others on the project whose schedules depend on their work being done on time.

Insure that your employees are equipped with mobile contact devices and understand that, whatever their personal schedules or commitments, they are on call during regular business hours, five days a week. Tell them their first communication method should be the phone—not email—and that it should be used anytime they can't solve the problem that is impeding their work.

11. Empower QA as an Independent Management Resource

Your Quality Control analysts—your testers and test managers—are engaged in day to day risk analysis and success measurement for your company. Although they are often the lowest rung on the project totem pole, these are the people on whose objective assurances your company depends to avoid information access crisis that could potentially paralyze parts of your company for hours or days.

To obtain your best benefit, these people need to be removed from the sphere of business line and Project Management influence, and applied to IT projects as an independent, objective management resource. Testers will still measure and report Quality Control issues (product quality) to their PMs. But they need unencumbered communication paths to report larger quality risks—like workflow problems that indicate costly inefficiency, and risky or redundant management



decisions that could burden your company for years--to an independent Quality Assurance manager who in turn reports directly to you or your chief lieutenants.

12. Align QA with Internal Audit and Corp Risk

Close the circle of problem risk evaluation, fault factor identification and problem solution. When a problem occurs, you apply Corp Risk to evaluate the impact of the problem. If it exceeds your risk tolerance you then apply Internal Audit to identify fault factors by comparing the attributes of the problem against your corporate policies and standards. Then what? Too often the responsibility for resolution gets handed back to the very people responsible for the problem in the first place. Consider this: If you don't already have confidence that the solutions now in place are not the very best your people could come up with, why are they still working for your company? What makes you think they can come up with any better solution on their own? The people responsible for the problem are exactly the wrong people to fix it. They need independent assistance to discover a new way of doing things that will fix the problem without breaking something else or costing vastly more.

Unlike Internal Audit, certified QAs are both ethically and professionally bound to explore and recommend industry standards and best practices that will assist your divisions in implementing quality solutions. Your problems will stop the day complete the circle that is a rational approach to corporate quality improvement.

13. Engage a qualified Information Technology Monitor.

IT CIOs are the people that must justify corporate infrastructure investment—the money pit of any corporation. Too often, corporate IT CIOs are so burdened with politics and public relations they lose their technical and managerial edge on handling the IT juggernaut. When this happens, they cease to be technical assets to the company, and instead drop back to a supervisory role. Given their overwhelming burden, they have no choice but to rely on their Senior Managers to make critical technical decisions--often for expediency's sake, trusting—rather than proving—that these are always made in the best interest of the company. Worse yet, many CIOs rise to the rank because of their knowledge of the corporate business and management, rather than any hands on knowledge of IT infrastructure and technology. In either case, the value of any CIO is only proven if they know the right questions, ask them often, and demand proof of the answers in a form they completely understand. This is the only way CIOs can be confident their decisions are based in fact, not opinion. Any CIO that feels in the dark about his IT hasn't done his due diligence by making his lieutenants do theirs.



Given the overwhelming burden of the job, your CIO can't do it alone. Provide your CIO the independent technical assistance they need to keep their finger on the pulse of your IT. Engage an Information Technology Monitor, someone qualified to help your CIO tame, tune and monitor your IT to achieve maximum efficiency and return on investment. At least certified in Quality Assurance, this person would be applied directly to the CIO or CTO and would conduct all IT post mortems, following up to insure that lessons learned are not lessons lost. They would work directly with your PMO or Senior Managers to optimize the software development and vendor integration process, seek hidden risk in project proposals, measure implementations against industry benchmarks, investigate efficiency and productivity issues, assist line managers with audit remediation, and insure security is built in—rather than patched out—of information systems. They would work in the trenches with the people that build, protect and administer your IT. They would hear their concerns, evaluate their needs, assess their challenges, and report back to your CIO on the overall health and stability of your IT organization. This person's focused mission would be to help your CIO quantify and control security liability, avoid costly tool expenditures, reduce the need for consultants or temp workers, find practical alternatives to costly, risky business requests, and motivate your IT work force towards excellence.

A CIO must exhibit balance and perspective in order to provide sound IT direction. He must be in tune with his organization's culture and challenges and on top of changing technology and perception at all times to be effective. He can't do this if he doesn't know what's going on. Those that aren't sure either tend to be complacent and timid, or reckless. The breaking world of global IT is no place for timid or reckless CIOs.

14. IT Managers

Managing creative people well is a nearly impossible task. The target is constantly moving—people's attention, interests, ambitions and concerns change at different rates. On the soft skills side, there are many things a manager should have—including leadership skills, comprehensive business understanding, flexibility, etc. But there is one thing every manager must have: *ethics*. A manager whom his workers cannot depend on being straightforward, decisive, consistent and fair is worthless, regardless of whatever other admirable skills he/she may possess.

Soft skills are important but how many of your managers think that's all there is? How many of your IT managers are just glorified HR people? Just when did management stop being a science and become a liberal art?



At its core, a manager's job is to prioritize, measure and enforce. IT managers should communicate corporate directives, prioritize projects, measure productivity, and define and enforce corporate policies, standards and procedures.

These skills are sorely lacking among contemporary IT managers—in part, because the specifics of how to manage in this relatively young industry are still being formulated. Perhaps the best way to judge the quality of your IT managers is to inspect the performance reviews they file on their direct reports. If the manager does not describe, in estimated dollars and cents, what business value this employee brings to the company, then they are probably not measuring the same for their team. By the same token, if the manager does not describe how the employee measures up against team quality standards, they have probably not defined any, or are not enforcing them.

15. Project Managers

A Project Manager motivates, mediates and facilitates their teams. Most Corporations not themselves engaged in software development as a product cannot afford career Project Management professionals. Instead, people rise from the ranks of Business, Development and Quality Assurance to organize and lead software development projects to conclusion. However, burdening your most valuable technical resources with Project Management duties is about the dumbest thing any IT division can do. It takes your most valuable technical talent away from doing their best technical work, and saddles them instead with activities they are usually ill-equipped and unmotivated to do.

The obvious solution is to encourage career Project Management Professionals in your organization. But it is not the right one. This approach usually means that the people with the least understanding of the business problem and proposed technical solution lead the project. Often they don't understand how software and systems development works. Worse yet, many don't even think they should. Traditional Business Management training is appallingly lacking where IT Software development is concerned. Some highly respected management methodologies seem to suggest a manager need know nothing about what will be managed—only how to manage. Does *that* make sense to *you*?

What you really have is a wheel in a wheel—the project lifecycle and the software development lifecycle. And if the two cogs aren't in perfect alignment throughout the process, your technology solutions will be faulty, late and consistently over budget.

To find out what your PMs know and don't know, collar a Project Manager and ask them to describe what happens during each of the four main phases of a



development cycle (Requirements, Design, Construction and Testing). Ask them how they insure against redundant documentation. Dig down into the dirt and require proof. Ask to see their Business Continuity Plan. Ask them how each member of the team assures the quality of inputs to their workbench, and what steps are taken when these are substandard. Ask to see the bug trend and production performance reports on applications from time to time. Ask them to compare their best and worst IT project this year, and how they determined the root cause of inefficiency and addressed it. You'll know in a heartbeat who understands what they are managing and who does not.

Career Project Managers move from project to project, leaving the Project Team holding the bag for their management mistakes after deployment. The right idea for a lean, efficient IT, is assessing the skill sets of all the team members—business analysts, developers and QA—and spreading the burden of mundane project management activities like cost tracking, customer communications, project schedule management and risk assessment—over the whole team. This both educates all members on the managed project process as well as encourages team ownership of, and accountability for, the quality and success of the project as a whole. Someone will still be in charge—probably the same highly skilled person that would have been chosen anyway—but enough of the Project Management duties are removed from their world that they can also apply their critical expertise to the technical problem effectively.

16. Pull, don't push projects to success

Successful projects are the ones that anticipate up front, all the obstacles right up to the day of deployment and beyond. This reveals any schedule absurdities inherent in the project at the outset, which can then be negotiated before the project is under way, not halfway through the project (and money) when tensions are already running high. A bad project is approached like an off Broadway show that expects to go into extended run. A good project is approached like a Carnegie Hall debut. Promotion and logistics planning for the day the curtain goes up happens as soon as the project is under way.

Here's how to tell the difference between a "push" and a "pull" project: On a "push" project, the PM boots up MS Project and starts mapping the team's activities through the first phase, figuring there are too many unknowns to anticipate a full project schedule at the top of the project. On a "pull" project, the PM starts with a vision of success based on knowns, fills in the blanks with assumptions, whips out a calendar, locates the deadline date and starts walking backward through the project workflow, mapping dependencies.



This does two things: it focuses the team on a single vision of a successful result (whether feasible or not) and anticipates most of the tasks that would have to be accomplished to achieve that result. This does not mean that the tactical plan won't change radically upon reaching Phase 1 milestone. But it ensures that the whole team understands what a successful result looks like and is working toward it through Phase 1.

It's as simple as that. A "push" oriented Project Manager tries to push the team to the next milestone. A "pull" oriented Project Manager's has already walked backward from the finish line, knows the track, and concentrates on getting the project team members what they need, before they need it.

17. Software Developers

Software Developers know how to research, design and create your solutions. What little most developers know about quality software development process is usually wrong. If they avoided academic learning entirely, dropped in from hacking or transitioned from other professions they probably know nothing at all. If they have been in programming for more than 10 years then the models they used are probably no longer relevant. If they have been in the industry for less than 10 years and emerged from academia, then they have seen the models they studied bent and bloodied by the Object Oriented programming and the Web development craze. Baseline your developer's knowledge of industry standard software development process and terminology is essential to assuring IT efficiency.

Beware of developers that don't put security as a top priority on their design list. Secure code is quality code.

Beware of trendy styles of software development. Approaches like Rapid Application Development (RAD) compress development process activities to effect fast release to market. Things happen fast, but are they good? And just because they happen fast doesn't mean they're not expensive. RAD process is a natural preference for most developers—and understandably so. Taken in the wrong context it offers them unbounded freedom to experiment until they come across something that might work. This is far more interesting than thinking through a problem fully, then mechanically writing out what is already solved in your head.

The faster the pace, the greater the chaos, the more rigorous controls must be applied. If your developers require working in these chaotic styles, then glue a tester to their armpit and monitor and measure their productivity and its cost.



18. Get HR out of the IT hiring loop

If you have reason to question whether the right people are placed in your IT, maybe it's because the people hiring your technical workforce aren't qualified to do so. Most large corporations have mature but ineffective HR departments where IT is concerned. Their hiring practices are slow and the procedures are downright Byzantine. It can be weeks before a highly qualified technical candidate gets a call back, and weeks more before they get to talk to anyone in the ranks who can understand the value they might bring to your corporation. Get your non-technical people out of the hiring loop where IT is concerned. Demand fast response to interesting candidates by people qualified to speak their language and excite them to work for your company. You can't afford to let brilliant technical minds walk away.

19. Vendor Software: Up your BS meter

Metaphor is a powerful thing. So your CIO comes to you and says this: "It's like... Ok. Our company network is like a medieval knight, clothed in a stainless steel set of armor from head to toe. When we came back from battle yesterday we noticed a chink. We discovered there's now a gaping hole that exposes....." Having some part of your virtual anatomy lopped off might be more than enough to get you to sign the software security purchase order. But before you do, *do the due diligence*. Studies show that most large corporations have all the security software they need but don't use it effectively. The pace is so fast, administrators are often inclined to trust their instincts and experience rather than take the time to actually read the documentation during install. These studies showed that a high percentage of corporate anti-virus software is mis-configured or improperly used. The same is probably true for other third party packages you've licensed.

So before you sign on the dotted line, demand proof that what you already have won't fix the problem. A simple email direct from a vendor will do. That your employees are dedicated, loyal and hard working people should give you no confidence whatsoever that the product you already have has been set up, configured correctly and is being used properly.

20. Align Networking, Data and Communication Divisions with Enterprise Development

The tug of war between infrastructure and development divisions is legendary in the software world. Devs necessarily require freedom from standards and network restraints to experiment with new technologies. Network Administrators are specifically in place to protect your production network against such renegades. So innovative development necessarily takes place in isolated, insufficient environments that rarely reflect production.



Consequently, corporate development solutions are never fully tested until they are deployed to the field. Unanticipated problems after launch tend to be the rule, rather than the exception. Testers sign off, devs hand over their code and the Net Sys Admin dutifully installs and distributes it. Users report faults to the developers. The devs diligently research the problem but can't replicate it in their insufficient test environments so they escalate it to the server administrators. These folks can't see anything wrong with their systems so they escalate to the Networking people who determine the same. They then they pass the buck back to the developers.

Thus begins the mincing. As customer complaints mount, a relentless, thoroughly unproductive exchange ensues between the devs and infrastructure people. The more heated it gets, the more territorial and political it becomes. In the end, it might have been something as simple as a system setting that was different in the Test lab than it is in Production. But more often than not the root of the problem is some obscure system or network interaction that will never be found.

That's really not the problem. The bad feelings this exchange engenders is. It adds yet another layer of brick and mortar to the thickening wall between the two essential arms of your IT organization that must, under every circumstance, work cooperatively to give your company the essential services and needs. Break down the barriers between these warring factions. Align them and co-locate them to encourage the free flow of knowledge that leads to the creative insight and interdepartmental cooperation needed for your corporate solutions.

21. Apply QA to Networking/Communications/Data infrastructure

Test results are the proofs used to determine the soundness, security and fitness of development solutions across your company. Inter and Intranet applications are fast becoming the core of mainstream corporate communications and marketing. These applications can no longer be tested out of context. Web applications require a shift from Requirements to Corporate Risk based testing. This type of testing requires evaluation not only of the software sitting on the host server and desktop, but of the data systems, networking and communications systems that deliver service to your customers.

QA is all but unheard of in the Network Systems world. Testing happens, but in the absence of quality testing principles. The growing emphasis on corporate web communications and e-commerce demands formal QA on the infrastructure side. Without it, applications are deployed only partially tested, at expense of your corporate security, data integrity, and system performance—the very factors most responsible for customer confidence and satisfaction. So mandate that any



new application deployed MUST be tested, end to end, under the direction of a qualified, certified QA or a network administrator schooled in formal quality testing principles. Aside from assuring the application, this also serves to spot check your IT infrastructure from time to time. Hidden problems will probably be revealed that your Net Sys people don't have the time, skills or inclination to thoroughly explore themselves.

22. Focus on the core essentials

Communication is the root of all issues. Change is the root of all confusion. Risk is the root of all failure. Master these, and you've mastered IT. On the Business side, managing these three variables requires consistent method, the application of core analysis skills and the application of process controls. Certified Quality Assurance Analysts understand such methodologies for IT-- how to analyze IT problems, and how to apply appropriate IT controls. Their profession is all about identifying issues, quantifying risk and recommending proven solutions. Chief among their tools is Root Cause Analysis. Unfortunately, the root cause for many project issues are often external to the project. They may point to arcane standards, or missing/faulty policies and procedures that should be addressed. You should consider hearing them, even when your top managers want to shoot the messenger. They come bearing proof.

Implicit in this directive is the need to manage change. At the Project level we call this change management. At the system/application level it is called Configuration Management. Strategies for managing both can't be dictated by a single individual, unless a single comprehensive, integrated software project management tool is applied and all project members are equipped with licenses for it. The approaches and tools that will be used by business analysts, developers and testers for communication, documentation and status reporting all need to be considered when making decisions about how manage the flow of communication and authorize changes the project and system—both during and after development.

Authorizing change, and controlling the subsequent activities that flow from it are probably the single most important responsibility for a project manager, and the very first thing that should be planned. Without a clear, comprehensive, regimented change management strategy that stresses change impact analysis, risk evaluation, short interval planning, design review, testing and signoff, projects quickly go out of control, and spiral down into incomprehensibility. If you want to waste lots of money, encourage strife and turmoil in your project groups, and create unsupportable products, then allow your projects to go forward without a documented change control process.



23. It's the culture, stupid!

The root cause for most problems in your IT have nothing at all to do with technology or money. They have to do with people. Contemporary software development, management and analysis theories are beginning to account for this. They provide strategies and recommendations on how to insure you get the most from your creative groups.

Brian Foote and Joseph Yoder's Big Ball of Mud theory of software evolution patterning suggests that you can't evaluate your IT projects as strictly technical problems, but must consider the human factors—the soft inputs—that are impacted by the process and outcome. When a system begins to fail or a project goes awry, at least half the attention should be on the human factors. Too often great teams are given the wrong task at the wrong time with the wrong resources. Projects get launched before enough attention is paid to the question of information service redundancy, technological feasibility and cost vs. benefits. Those are human faults, not software faults.

Jim McCarthy, author of The Dynamics of Software Development, has expanded his ideas and embodied them in a methodology called the CORE. He suggests that almost all attention should be placed on testing the human inputs to a project before it begins. Creative people must get to know one another and themselves—intimately-- to be effective together as a creative team. Individuals must be allowed to explore and understand their personal motivations for being on the project, and management must understand what fundamental wants, needs and aspirations the project will fulfill for them to know how to drive them to success. The emphasis is on self-discovery, team dynamics, and on the relationship between the team and management. When expectations are known and understood by all affected parties on all sides, the creative juices are ready to release. Then, says McCarthy, the project will pretty much run itself and will produce a high quality product.

Make sure your keenest minds have the time to explore such contemporary approaches. If you want to get the best out of your people, you need to challenge and stretch them and give them the latitude to pilot new ideas they are convinced could bring greater efficiency, effectiveness and business value to your company.

24. Hire Career Professionals for Key IT positions.

If you expect your IT to service you 24 x 7, then you need to put people in key IT positions that can handle 24 x 7 responsibility. Most IT jobs can be managed well by normal people, who expect a normal workload, who get paid a normal, industry competitive salary, and leave their jobs behind when they go home. But



for some key IT positions—like Network Systems Administrator Lead, Chief Network Security Administrator etc. as well as perhaps some of the higher Director/Manager positions in IT, you should look for IT career—rather than casual--professionals. They are easy to recognize. I'm talking about "geeks without a life"—usually single, without family or other encumbering personal commitments, whose life *is* their jobs. These people have little that will get in the way of insuring your company works well, all the time. They are the ones that already sleep with a pager, and think nothing of wandering by on a Sunday to make sure things are working perfectly well and everyone is doing what they should. The mystique of the geek as some mal-adjusted misanthrop is long gone. Lifestyles in dynamic fields like IT run the gamut, and you should recognize that for some of your key IT positions, your best bet won't necessarily align with the rest of your culture. It's a trade off you should be willing to take.

25. Stop Costly Tool Mania

IT Project Management, Software and QA tools are nothing more than packaged principles. Before you invest in costly integrated tool suites that claim to manage your IT projects end to end, first find out whether your IT folks understand the principles behind them. If they don't, you're investing in a costly fantasy that is not likely to show return.

Start at the top. If anyone in your IT management can't describe which Software Development Life Cycle model is used in their groups, what industry standards they apply, how they manage change control and configuration management, how they prioritize faults, manage scope, define versions and prove the efficiency of their process and the quality of your products, your people aren't disciplined or knowledgeable enough to use a comprehensive tool suite effectively.

There isn't anything in a Software Project Management suite that can't be achieved using the tools provided by the standard Microsoft Office suite. If team's can't show you how a change request conveyed in outlook mail, doesn't get detailed Business Requirement Word doc, then migrate to an Excel Test plan, then make into the Design document, and into code, your team doesn't understand the core concept of trace-ability. If someone can't build a simple chart using Excel that shows your bug trends over development cycles, cost rate over calendar time for project development, percentage of Project completion or application coverage for testing, no costly tool suite is going to teach them what they need to know. A fool with a tool is still a fool.



26. Control Training Costs...and get the most out of them

Most corporations understand that to be effective, their IT employees must stay informed. Your people must keep abreast of fast paced changes happening in the technology world if you expect them to give you the technology efficiency edge your business needs to remain competitive. Too often that means carte blanche where IT training is concerned. How do you know those dollars are being used effectively?

There are a few key ways you can find out. Be free with training dollars, but demand pay back. Collar any line level worker and ask them to show you how they applied the last training they took. This is especially important for your Project Managers. Most green project managers are quickly overwhelmed by the human factors of the project and abandon what they learned in the classroom, as well as sense of rational control as soon as the project is under way. And there goes your training dollars.

Sitting through a week long Project Management class is only cost effective if the first time PM has hands on mentoring through their first few projects. Skilled and qualified Quality Assurance Analysts can support them with workflow and process analysis, and provide recommendations on tools, templates and standards that should be applied. Veteran, quality oriented Project Managers can guide them and provide advice concerning the many interpersonal issues that arise, give them the courage they need to make hard, unpopular team decisions, help them analyze and scope the goal, and prioritize risk and work. Together, these resources provide the support structure that insures your training dollars will be productively applied.

27. Begin Grooming a Chief Monitoring Officer

In a projection of the Top 10 most powerful enterprise positions over the next decade, a Gartner Interactive Insight Report (November 2000) rated the Chief Monitoring Officer role at the top of the heap. It described this role as "...monitoring business processes and metrics in real time, this person will have their finger on the pulse of the enterprise. These are the people that will find the problem and solve the problem before they ever arise."

Without explicitly naming it, Gartner has described Quality Assurance as the key to business success. Certified QAs are trained and certified in process evaluation, metric evaluation, and in applying preventive controls. These people can be groomed with the critical skills they will need to supply you with real time proof of the status of your corporation as a whole, at any moment.



In fact three of the 10 positions described are dedicated to quality assurance activities. At number 6 is the Transaction Cop. To support web commerce, these individuals would "...make sure there is integrity in the transactions." And finally at number 10, the Anthropologist, who will "...do the proper diagnosis and the professionals who can do the best descriptions." So if you ever thought Quality Assurance was just some costly, non-essential drag on your company's productivity, think again. Because according to the experts, it's going to account for almost 30% of your strategic effort over the next decade.

28. Demonstrate Quality Values and Virtue

In your leadership role, stress and use the word "quality" in every corporate communication. Require proof of quality oversight that meets standard business and audit practice guidelines for each level of your corporation, from your direct reports, right on down to the maintenance man. After a while, people will "get" the notion that they too fully own the responsibility for assuring the quality of their own work and should be actively participating in monitoring the quality of products and services turned out by their units.

29. Back quality words with quality actions

Don't just talk it. Demonstrate your commitment to Quality in your organization. Encourage (if not require) that your Directors and key operations people, your project managers and your business managers and analysts—both internal and external to IT-- be trained (if not certified) in contemporary Total Quality Management methods as a pre-requisite to upper job mobility. Encourage professional certification and promote IT QA as a professional career path in your organization, rather than a stepping stone to development, project management or business analysis. Heap accolades on people, both large and small, who achieve quality in their work and your products. And do it in dollars and cents too, because very likely, out of these ranks is likely the to rise the quality Officers you'll be looking to hire in the next few years. So unless you want to be paying big ticket prices on the outside for this kind of talent, you need to start engendering fierce company loyalty among those likely candidates who demonstrate a strong commitment to quality.

30. Why listen to me?

Because I'm that passionate geek actually working in your trenches. Because I've had the opportunity to see the best and worst at work in IT ranging from software development to telecom to the insurance industry.

If I could identify one factor that makes for good IT, it's passion. Bright, creative people are passionate. Most of them don't let politics or culture get in the way of knowing what they think, saying what they mean, and fighting for it. Everything is



always on the table. Everything is always open to debate. And if you demonstrate a commitment to excellence by instilling the same in your lieutenants, common sense, logic and rationale tend to win out in the end.

Do you have bright, creative people working in your IT? Do some of their concerns ever filter up to you? Do any of them have the courage to break ranks and tell you what's really going on? If not, then you have a top heavy, bureaucratic IT structure that won't attract the passionate, creative people you need willing to go out on a limb to make your company a success. Don't you *want* me working for you?

Do these things and you'll find out how sick or healthy your IT is, and what needs to be done about it. Demanding proof of quality is your ticket to long term success. The longer you delay, the greater jeopardy you and your company will face. IT is no longer the bastard child of your company. It's the heartbeat the supplies the blood to your business. If you don't pay attention, apply common sense, bring it back to center, and smooth out the rough edges, the coronary it gets may also be your own.

Bar Biszick-Lockwood is a Certified Information Systems Auditor (CISA), a Certified Information Systems Security Professional (CISSP), and a Certified Software Quality Analyst (CSQA). She is an expert in Security Life Cycle Process standards and specializes in IT regulatory compliance audit, IS assessment and IT process re-engineering to optimize organizational security and meet regulatory mandates.

Ms. Biszick-Lockwood authored the security activities in the pending revision of IEEE P1074 Standard for Developing Project Life Cycle Processes that provides practical guidance in applying optimal security controls to software projects and building adequate security controls into products. She is a member of IEEE, ISSA, ISACA, and SIM and is also on the adjunct curriculum writing staff of Logical Security, a security education company led by Shon Harris, author of McGraw-Hill's best selling CISSP All-In-One-Guide.

She has worked as a SOX auditor, Quality Improvement Analyst, Trainer, Tester, Security Test Lead, Test Manager and Special Projects Manager in the software development, telecom, insurance, and healthcare industries. She is published in Software Testing and Quality Engineering Magazine (STQE) for recommendations on Software Patterning, and has been a featured speaker at national and international IT Quality and Security Conferences, and at Microsoft.