



Research Report #17
Author: James M. Spitze
Date: June 2006

The Joy of Management

Table of Contents

ABOUT THIS REPORT	1
PRELIMINARY THOUGHTS, PRELIMINARY STEPS	2
PHILOSOPHY OF MANAGEMENT	4
INSTITUTIONALIZING COMPETENCE	5
Teaching Pigs To Sing.....	6
Bench Strength	6
Fundamental Assumption.....	7
Context.....	7
Balance	8
Druthers	10
One Good Programmer	12

ABOUT THIS REPORT

The title of this SCC Research Report is not a contradiction in terms. Rather, it is intended to make the point that *Management*, at its best, can be more fun and much less methodical than many management thought-leaders would have us believe. This is not exactly a Research Report since no structured research has been performed in its preparation. Instead, it is an Experience Report based on my personal experience.

James Gallier, the famous 19th century architect of many of New Orleans' most well-regarded buildings, once said "*I hold it to be a duty incumbent on everyone connected with the Art to attempt an addition to the general stock of knowledge and be that addition ever so little, it will tend to advance the profession, and ultimately prove beneficial to the public.*"

It is my goal here to hopefully advance the art of IT management, even if my effort advance that state "ever so little." In this paper, I will outline a philosophy of management that I think is important, and attempt to counter the too numerous attempts to *institutionalize competence* by excessive reliance on formal processes, regulations, and various management concepts.

PRELIMINARY THOUGHTS, PRELIMINARY STEPS

The history of the management of computers and the inherent formalism of their programming and physical structure have combined to fuel a multi-decade over-emphasis on process over people in virtually all areas of modern private and public sector enterprises.

Fifty years ago, before the days of copiers and the widespread use of computers in business enterprises, most medium to large companies had a Systems and Procedures department. These departments designed with great care five, six, and seven-ply pre-printed forms to expedite many mission critical business processes. Using clipboards and stopwatches, these departments designed and implemented well crafted processes that "got things done" in the pre-computer world.

In a 1991 interview, Dr. J. Presper Eckert, the co-inventor of the pioneering 1950s era ENIAC and Univac computers, was asked "What do you think computing has accomplished over the past 40 years?" His insightful answer was:

What we were beginning to see 30 years ago was that we were starting to have more people doing paperwork than doing production work in the United States. I think that all the computer has done so far is stem the tide to prevent it from getting worse, preventing a paperwork explosion, preventing the paperwork explosion that never occurred, which is the result of the computer arriving in time.

Yes, the computer arrived in time - and it was quite often made the responsibility of the already in place Systems and Procedures departments. The inherent focus on formal procedures of those departments and the inherent formalism of the programs that were used to control the then new computers reinforced each other to support - with little contrary discussion at the time - the assumption that the best way to manage the computer and the programs that controlled it was via an array of formal processes that have been growing and blossoming ever since.

The time has come to rethink that assumption.

In March 2006, I distributed a draft of this paper to my firm's core team members for their comments and suggestions. Terry Curtis replied swiftly that I must read Professor Claudio Ciborra's 2002 book (published in paperback in 2004) entitled *The Labyrinths of Information*. I did as suggested and can only say that Ciborra's seminal book is one of the very, very few *Must Reads* that I have

encountered in over forty years of reading management tomes. The following paragraphs are a condensation and re-phrasing of the book's first fifty or so pages. The condensation process necessarily adds a degree of terseness that Professor Ciborra avoids by his generally understated and modest choice of words. I encourage you to read the full book.

The highly formalized nature of programming and software has led most IT professionals to the conclusion that the development, maintenance and documentation of software should also be highly formalized. However, the same IT professionals will agree that in practice, the everyday life of software development is punctuated ubiquitously by improvising, tinkering, applying patches, and (if the truth be told) cutting corners. Fundamental systems practices are, in the real world, quite untidy and irregular. Whereas we genuflect to our profession's formalisms, it is the untidy and irregular practices that are often the source of competitive advantage.

The high functional power of improvisation - anathema to prevailing software engineering models and methodologies - is clear from observation of real people developing real systems. Everyday experience shows us the ubiquitous nature of muddling through to get new systems developed, implemented and used. Precisely because this kind of experience is well known and widely shared, many of us don't want to hear about it.

The formalisms we have developed over the last forty years are, in practice, an inauthentic way to deal with the challenges we face. We must dare to enquire publicly into them - and to start the process of replacement. At their root, our numerous method-driven approaches are too often misleading, deceptively abstract, and removed from real-world practice. Our methods are based on a too narrow model of rational, ideal actors. We must broaden our model to embrace the real world of tinkering, improvising, and ubiquitous untidiness.

We err when we use the same analytical approaches for both physical systems (the computer) and for social systems (business organizations and their human-based processes). We delude ourselves when we think we are modern when we trust data flows, entity-relationship graphs, and structured methodologies. We have overdosed on methodologies and must envisage an alternative approach.

We must suspend our belief that behind the messy everyday reality there is - if we could but perceive and document it - a geometrically pure process that will eliminate risk and insure consistent success. Our striving for excellence is too often an effort to do better what we already know how to do. Such paves the way to routinized, though efficient, systems and

procedures - but it does not pave the way to competitive advantage through strategic innovation. To innovate, we must move beyond rigid controls aimed at uncertainty reduction.

PHILOSOPHY OF MANAGEMENT

Ciborra makes the point that many of our carefully crafted organizational structures and formal processes have become obstructions rather than assets. With numerous anecdotes and an impressive case study (Olivetti), he asks us to re-think our rejection of tinkering, untidyness, and *muddling through*.

Too often we lose sight of what management is supposed to be all about. That is, to control and direct the affairs of an organizational unit (household, business or government unit, etc.) towards a desired end. Too often we over-emphasize the *means* (its process, structure, integrity, etc.) and lose sight of what we are trying to achieve.

Too often we encounter excessive adherence to budget, schedule, process, organizational protocol or another example of the seemingly infinite array of bureaucratic hurdles. With "success" clearly defined as "achievement of x," it is amazing what the human mind can come up with to insure not getting there. At the very least, we are wonderfully creative at greatly complicating the task.

The idea of "keeping your eye on the ball" seems pretty obvious. However, in my 40+ years in this field, I have encountered many about-to-fail projects that I have swiftly turned around, often with very little effort. I say this not to brag but to make a point - If you have a project that is about to fail, take a careful, objective look at what is causing this to happen (people? process? organization?). My experience indicates that the cause will be pretty obvious and often embarrassingly easy to correct. It will most likely take some courage along with a willingness to temporarily muddy some relationships but, remember, the objective is success, not excuses and well-worded CYA reports.

It surprises me how often the difference between success and failure is simply a bit of positive energy, a willingness to ask some generally pretty obvious questions, and then a "dig your heels in" reluctance to accept anything less than success. "Success" here means happy users/customers. The Program Management Office might have its nose out of joint because its processes weren't followed. The CFO might be upset that you spent 6.2367% more than your budget. Charley in the Procurement Department might threaten you with dire consequences because you didn't use his low price bidder (and their nightmare interface). None of these count in the greater scheme of things.

Speaking as one who has spent about half his career within very large corporations, I have encountered my full share of tyrannical bullies and power-

grabbing, holier-than-thou, obstructionist bureaucrats. On the plus side, such people offer the rest of us "interesting" intellectual challenges regarding how to disempower them.

The obvious job of management is to succeed, not to generate excuses. Sadly, the evidence often suggests the opposite. Just think of all the things you've seen fail for really dumb reasons - scope creep, incomplete spec, trivial budget issue, inadequate Q/C, poorly attended training, databases full of garbage, etc. etc. In too many organizations, the operative rule seems to be "Go along to get along" which means that rather than putting our foot down and saying, however gently, "No!" to something that we know is well off the mark, we tolerate it ... with someone else (we hope) left to pay the price later.

Perhaps a significant part of the problem is to be found in how we say (or don't say) "No!" As those of you who know me will attest, I try to do this by telling a hopefully relevant story that gets an individual to see the error of his/her action without a nasty, relationship-damaging argument. There's an old saying about disagreeing without being disagreeable - and I really think that's a pretty good thing to try to do.

People are people. Even the *Best and Brightest* screw up sometimes. Work with them to get them to focus on success. Get them to realize that success is, most often, a lot easier and far more fun than failure. Although some might say this is self-delusional, I disagree. Vince Lombardi once said "The difference between a successful person and others is not a lack of strength, not a lack of knowledge, but rather a lack of will." He also said "The price of success is hard work, dedication to the job at hand, and the determination to apply the best of ourselves to the task at hand." I couldn't agree more.

INSTITUTIONALIZING COMPETENCE

Management is constantly bombarded by various well intentioned "things" that in some way or other are supposed to *institutionalize competence*. These can be regulatory such as SOX or semi-regulatory such as ISO or conceptual such as Six Sigma, Workout, Empowerment, TQM, Zero Defects, ROI, CMM, and so forth. Whatever their nature, virtually all of these things tend to drain the joy from well performed management while too often negatively impacting the success they are supposed to foster.

Why is this? Why, as recently observed by a respected management guru, have something like half of the Baldrige Quality Award winners gone out of business? Why, after forty years of tool and process development, do over 50% of IT development projects still fail? Why according to the latest statistics do over two-thirds of all major, comprehensive outsourcing projects fail? Why does Six Sigma

and Workout work well within GE but - even when managed by ex GE executives - often fail elsewhere?

Teaching Pigs To Sing - There is an old saying "Never try to teach a pig to sing; it wastes your time and annoys the pig." The problem is the receptivity of the student - and the key to success in all efforts to institutionalize competence is exactly the same - receptive students, students who will understand the nuances, when to use and when not to use, and the required context for success.

If you talk to virtually any professor of a leading university, they will tell you of the joy they get out of teaching receptive students and of their efforts to attract such students. The key to success of great universities is both great professors and great students. This has a direct parallel in the business world yet too often we see efforts to institutionalize competence structured as one-time-only projects or as management decrees such as "Thou shalt do x or suffer onerous consequence z." The quality of the employee-student is being ignored which any honest professor will tell you is well off base.

Several years ago, I guided the selection of a new corporate-wide order entry/order management system for Hewlett-Packard. We were down to two finalists and were using a Kepner-Tregoe weighted evaluation matrix with six or seven criteria - including cost. All the right HP executives were in the room and we were making slow but good progress when the HP financial executive said something like "I think we should remove cost from the list." Although one alternative cost far more than the other, he rose above the evaluation mechanism to consider the broader context. Talk resumed and in perhaps another half-hour or so, we had made a selection - as luck would have, the less costly but not because of that fact. Everyone seemed to have enjoyed the meeting. As a group, they seemed to think they had come to the right decision and had had a certain amount of fun while doing it.

It is interesting (and relevant) to note that HP, along with GE, Motorola, Xerox, and a few other large US companies, have had well entrenched employee selection and nurturing programs. Motorola's extensive employee training programs (much reduced in recent years) gave that company one of the most stable and well trained workforces of any major US firm. Both HP and Xerox had slow, arduous management hiring programs that weeded out all except those with clearly high potential. The early successes of both of these companies can be attributed to these hiring programs. They "got it right" in their efforts to *Institutionalize Competence* by starting with receptive students.

Bench Strength - The institutionalization of competence assumes something more beyond receptive students. It also assumes a recognition of the

importance of *Bench Strength* and a willingness/desire to have such a bench. In recent years many US firms have pruned their management ranks, eliminating their *Bench Strength* and justifying that by touting the benefits (often brief and/or non-existent) of various new and mechanistic management tools. While most companies these days require *scaleability* in their IT systems and in their business processes, many of these same companies fail to perceive the need for the same thing in their management teams.

Just last year an associate and I had the pleasure of advising one sizeable hi-tech company with expectations of quite rapid growth. They were concerned that their ERP system wouldn't be "up to it" and might need to be replaced with, perhaps, either Oracle or SAP. Since what they had seemed both scaleable, well supported, and in keeping with their current and expected 3-5 year needs, we recommended they stick with it, implying a likely saving of \$6-8 million. However, while their systems and processes seemed appropriately scaleable, their management *Bench Strength* was virtually nonexistent. Thus, we recommended significant increases in that area which the company is now responding to.

Fundamental Assumption - There is a fundamental assumption in all *Institutionalization of Competence* efforts - and that is that such a thing is feasible. Many years ago I took six units of "Leadership Training" as part of the NROTC program at UC-Berkeley. I went in thinking that *Leadership* was a skill you had to be born with. I came out believing that *Leadership* could be taught (and learned) but only up to a certain point. I continue to believe this and think about the same with regard to *Competence*. The key with regard to institutionalizing the latter is doing it carefully - with premeditation, with clear objectives in mind, with a real understanding of the what, why, and how.

"Workout" without proper budget and "released time" support can be a real *downer*. "Six Sigma" programs done on a departmental-centric basis can be hugely sub-optimal (read "counter-productive"). Happily the days of mechanistic subservience to ROI "hurdle rates" seem to have passed ... at least I thought so till a few weeks ago when I attended a talk on IT Governance and saw an in use project selection equation that gave a 40% weight to ROI.

Context - Many *Institutionalization of Competence* tools-concepts-regulations have real value if used within their intended context. However, too often they are seen as *Silver Bullets* and a generalized "use it everywhere" approach is followed - with disastrous results. ITIL is strong in some areas, weak in others. CMM is useful to some level but only in some (not all) environments. One highly regarded IT consultant, far more knowledgeable on many of these topics than this writer, recently expressed his concern that most of the IT-related *Institutionalization*

of *Competence* tools had an increasingly academic aura to them that was causing him to approach them with growing skepticism.

Many well conceived management tools and concepts that seem "on the mark" in theory prove to be "wide of the mark" in practice because of the retardant nature of the entrenched culture; what might be called the "ignore it long enough and it will go away" syndrome. The best counter to this is the "do it once and do it right" concept - involving careful planning and a thorough appreciation of the fact that you are, to use another old expression, trying to teach an old dog some new tricks. It isn't easy and it almost always takes much longer than expected.

Balance - Central to all of this is the issue of *Balance*; where does the need for a quality concept leave off and the need for a quality person begin? In recent years, senior management at several large California private sector firms and public sector government agencies have erred seriously in focusing far too much on tools (often of questionable value) and far too little on people. This is a potentially very expensive form of avoidance. It is simply easier to focus on the abstract world of tools, concepts, processes, policies, and the like, than it is to focus on the very real (read "human") world of hiring, nurturing, and retaining excellent people. No amount of Workout, Empowerment, or whatever, programs will have much in the way of enduring positive impact in the absence of excellent people.

Happily, there are lots more excellent people than many executives seem to think! In my experience I have found that most people have a lot more "excellence" in them than one might expect. It is management's challenge to discover this excellence, to release it and channel its expression to the greater good of the organization.

I have found that each and every person that works for me is important and has value as a human being. I see it as a major element of my job to understand their skills and to enable them to use those skills for the greater good of the organization for which they (and I) work. Multiple times in my career I have found myself responsible for large but under-performing organizations. In the majority of these cases, I have been told by management that it is likely that I will need to make *big changes*, put in place entirely new processes and policies, etc. while in virtually all of these cases none (or only a few) of these were required.

What I have been doing for thirty or so years is something Professor Ciborra refers to as *small interventions* or, elsewhere, as *tinkering*. He makes the point that when we overdose (his word) on processes and methods, when we come to rely on procedural abstractions and distance ourselves from the real world, we often suffer the dire consequences that have been the norm for IT projects (software development, outsourcing, business process reengineering, etc.) since their

inception a half-century ago. Something must be wrong and it is our over reliance on the abstractions of process over the realities of people.

People are not machines and when we treat them as such via misguided mechanistic attempts to institutionalize competence, we err hugely. As mentioned in passing earlier, the challenge is one of *balance* yet far too many highly placed business executives seem to become enamored with process-based *Silver Bullets* while forgetting that the real *means* is Excellent People and the real *end* is Business Success.

A good *Sense of Balance* is often the personality trait that sets an excellent executive apart from the rest of the field. While lesser executives overdose on process or metrics or outsourcing (via unthinking conformance to the "core versus context" concept) or on any one of the latest management *Silver Bullets*, the superior executive seeks out - and finds - the needle of value within the haystack of passing fads and fancies. Narrowing our focus to just the IT arena, there is a highly apropos and marvelously "ahead of her time" 1842 statement by Ada Augusta, the Countess of Lovelace, writing about Charles Babbage's invention of the stored program computer - "In considering any new subject, there is frequently a tendency, first, to *overrate* what we find to be already interesting or remarkable; and, secondly, by a sort of natural reaction, to *undervalue* the true state of the case, when we do discover that our notions have surpassed those that were really tenable."

Besides our natural tendency to *overrate* and then to *undervalue*, there is also our ongoing desire for a *Silver Bullet* - to solve our IT management problems. However, as Fred Brooks (the Project Manager for the IBM System/360 computer family and later for OS/360) told us in his insightful 1986 essay "No Silver Bullet," there aren't any. To reduce the pessimism of this, he states:

The central question of how to improve the software art centers, as it always has, on people ... Those software systems that have excited passionate fans are those that are the products of one or a few designing minds, great designers ... Although I strongly support the technology transfer and curriculum development efforts now underway, I think the most important single effort we can mount is to develop ways to grow great designers.

He also mentions - and supports - the efforts of such organizations as the Software Engineering Institute (SEI) and its Capability Maturity Model (CMM) but stresses his central point, the primacy of people (excellent people) over process.

In a similar vein, Paul Strassman (ex "CIO" of Xerox, and the US Dept. of Defense) in his important 1990 book *The Business Value of Computers* states:

The business value of computers comes from the people who manage and use them. This is the principal rationale for insisting that managing the benefits takes precedence over managing the costs of information technologies.

In case of cost overruns or schedule slips, everyone finds it more acceptable to blame technical mistakes. This depersonalizes the real causes of failure and devalues the importance of the human element for all systems successes.

Druthers - It is really Fun (with a capital "F") and a lot simpler than one might expect, to take over a deeply troubled organization or project, define with some care what a "successful" turnaround might be, figure out what I need if I were given "my druthers," and then go get those "druthers." Rarely have those "druthers" amounted to much - maybe a few organizational changes, sometimes "encouraging" someone to stop being an obstacle, and almost never a request for significant incremental resources.

It continues to amaze me that so many people seem to create their own obstacles to success whereas if they worried less about obstacles and more about success, they'd have a much better chance of getting there. In case after case, I have discovered that my predecessor had all the resources he or she needed for success and could have easily achieved that success if they had simply focused intently on success and on the highest and best use of the resources already "on hand" to achieve that success.

In the early 1990s I abruptly took over a large IT organization with a recently measured "employee satisfaction" level that was one-third the average of the entire rest of the corporation. Morale was awful and staff turnover was about 30% per year. Luckily only one direct report to the CIO had resigned and I was able to get that person to return with not overly much effort. I interviewed every single IT employee in one-on-one meetings and did the same with every EVP, SVP, VP, Sr. Director, and Director in the entire company. Then, I fired one IT person, demoted another, got rid of virtually all of the contractors, worked with the user community to cancel a long list of small, inconsequential projects and repositioned the thereby released developmental staff onto two truly important projects. Resignations stopped and morale shot upward.

IMPORTANT PROJECT #1 - Within six months we had reduced order to ship time from 21 days to 12 days and by the end of eighteen months (with an entirely new Order Management system up and running) we had reduced that even further to routine delivery within 7 days, the best in the industry. The project came in on-time and a smidge below budget. This was success by any measure and we

all had a grand time while doing it.

Yes, in accomplishing the above, I did need to put in place some *Institutionalization of Competence* tools in the form of a tailored Systems Development Life Cycle and some not too structured project status reporting mechanisms. My goal with the latter was candor. I simply wanted to be made aware of any project problems before they got out of hand so that I could re-grease the skids and keep things moving towards success. I saw my job as one of obstacle removal plus I had to provide just the right level of tools and techniques to keep things moving without them being needlessly burdensome. Too often project status reporting becomes an end in itself.

IMPORTANT PROJECT #2 - The second of the two truly important projects involved the replacement of a large and complex system that had for over a decade required a team of ten or so IT people to maintain. I received a thick project status report from the contractor serving as the project manager. More resources were requested with particular emphasis on the complexity of the proposed new system's data architecture, which I found totally confusing. A company employee and I spent a few hours one afternoon trying to simplify it. The next day we declared success. Then, we got rid of the contractor project manager and her ten or so contract personnel, assembled from within the IT department's employees a "best & brightest" team, gave the challenge of running the project to a young woman with many years with the company and who thus understood the project's importance - and we were off and running. The keys to success were multiple: the much simplified data architecture, the meticulous (and well respected) personality of the new project manager, a top-notch developmental team, superb user support - right from the top, and the understanding for the outset that this thing was going to be time consuming and not at all easy. It took us slightly over two years. When we got into "Final Test" mode, things did look pretty darn good. However, we decided to hold off till fiscal year end to simplify the transitional bookkeeping. To my delight, the company's EVP/CFO - being of a somewhat impatient nature - decided to host a pre-implementation celebration at San Francisco's upscale Waterfront Restaurant. It went well with virtually the entire project team and a large portion of the user community happily present. The actual implementation a few months later also went well and the expected result - a large reduction in ongoing maintenance - occurred fully as predicted. The project was truly Great Fun for everyone involved, all of the programmers and analysts, the Project Manager, her boss (me, the SVP/CIO), the users and their tier of bosses (Controller, VP-Finance, EVP/CFO).

If there is anything that I learned from these two projects it was the extreme importance of a proper balance of tools and people - the right tools and the right people. Perhaps I was lucky (I did report to the Chairman/CEO and had easy access to him) yet I can think back to a few key things that I think we did right that it

seems lots of IT projects don't do as well. First, we made a huge effort to figure out what we were trying to construct and how we might best go about doing that. As part of this, we really, really, really listened to the users. We took notes and fed them back till some people probably thought we were slightly nuts. When we finally started construction, we used a prototyping approach that kept the users involved and provided ongoing confirmation that what we were building was what the users wanted. I made it clear that I didn't want one moment wasted programming something that really wasn't needed. It seemed to me that my critical resource was the time of my skilled "best of breed" programmers and I didn't want any of them to think that his or her hard work was wasted because "Management" had done a less than excellent job of figuring out what to build. Second, we selected the best darn team we could get our grubby hands on. Elitist? Yes. Did it work? Yes. Fun? You bet!

One Good Programmer - Early on in my career, I learned that one good programmer will accomplish in a day what ten not-so-good programmers will *never* accomplish. I have learned over the intervening 40+ years that this is truly generalizable to people management at all levels. Yet, I still, in the year 2006, encounter companies that will hire two not-so-good people at \$110,000/yr rather than hire one really excellent person at \$140,000/yr. I have tried several times to make the point that a "cheap" CIO can be the most expensive mistake the hiring executive might ever make - if one considers the high frequency and cost of IT project failures.

There are two major aspects to the people "task" - people selection and people management. Too often the selection process is done under some duress and management finally hires someone it sees as the "least bad" of the candidates. Whenever a hiring executive says "Well, if he doesn't work out, we'll simply fire him," that executive should take a deep breath and think twice about the consequence of what he/she is doing - and the adverse consequences of selecting less than the best.

No matter how seemingly inconsequential the job, people should take their time and hire the best darn person for the role as possible. I have heard all sorts of variations on the words "You don't need a rocket scientist for [fill in the blank]" yet having an over qualified person fill some role has, in my experience, led to far more positive results than negative. One person I hired took a few weeks to come up with a way for doing his job in one-tenth the time. Another person was "gifted" to me by a manager who found him impossible to work with. In short order, my "impossible" employee asked to work on a particularly difficult project and in just a few months came up with an astonishing result that almost doubled our product's MTBF (mean time between failure).

Most honest executives have similar stories to tell – of extraordinary *excellence* displayed by seemingly un-extraordinary people.

In short, GE got it right with its central focus on developing its high potential people while relentlessly weeding out its poor performers. While other companies fixate on GE's *secondary* Six Sigma programs and Workout meetings but deny the foundational need for truly superior people, GE has institutionalized that central focus - with all the rest, the Six Sigma quality process, the Workout approach to team problem solving, and so forth simply being "gilding the lily."

When you start without the lily, you are - to use another old saying - "trying to make a silk purse out of a sow's ear." It is for this reason that so many well-intentioned Six Sigma, Workout, and similar efforts fail.

Thus, the key is finding your lilies and moving forward with the gilding - with the latter being *far* less important than the former. The surprising thing is that there are lots of lilies to be found if we but make a real effort to find them. People like to be liked; they like to do well; they like to be trusted (and deserve that trust); they like to be challenged; they like to be thanked and rewarded for honest, successful effort. If these statements are true (as virtually all of my experience tells me) why do so many companies mistrust their employees, micro-manage them to absurd degrees, and fail miserably to identify those with high potential and develop them to achieve that potential?

The answer seems to be that too many executives find truly meaningful people management as something to be avoided - while concurrently continuously searching for a *Silver Bullet* (one or more of the aforementioned tools, concepts, etc.) that will give them what they want in the absence of the excellent people that are the only real solution. For all we've heard about Jack Welch, Six Sigma, and "workout" meetings, the foundation element of GE's excellence is its consistent, enduring, ongoing emphasis on developing its people - painfully careful selection, well planned training, meaningful evaluation, careful mentoring.

It is the whole "people thing" that is so hard to replicate. When a GE executive becomes a CEO of another company, he almost always brings along some of his GE tools - Six Sigma, workout, etc. However, hardly ever can he successfully bring along the intense emphasis on people that is the foundation of all the rest.

No amount of superior processes, policies, quality programs, or team problem solving meetings will fill the void created by less-than-the-best people and it is finding and nurturing such people that is central to achieving what I call *The Joy of Management*.

Bibliography

- 1 Brooks, Frederick P, Jr., *The Mythical Man-Month: Essays on Software Engineering - Anniversary Edition*, Addison-Wesley Publishing Company, Inc. 1995
- 2 Ciborra, Claudio, *The Labyrinths of Information: Challenging the Wisdom of Systems*, Oxford University Press, 2002 (paperback 2004)
- 3 Strassmann, Paul A., *The Business Value of Computers: An Executive's Guide*, The Information Economics Press, 1990

Also of interest by Paul Strassman:

- 4 Strassmann, Paul A., *The Politics of Information Management: Policy Guidelines*, The Information Economics Press, 1995
- 5 Strassmann, Paul A., *The Squandered Computer: Evaluating the Business Alignment of Information Technologies*, The Information Economics Press, 1997

The Ada Augusta quotation is from the book *Charles Babbage and his Calculating Engines: Selected Writings by Charles Babbage and Others*, Edited and with an Introduction by Philip Morrison and Emily Morrison, Dover Publications, 1961. The specific article is "Sketch Of The Analytical Engine Invented By Charles Babbage," by L. F. Menabrea, October, 1842, No. 82, Note G, With notes upon the Memoir by the Translator, Ada Augusta, Countess of Lovelace.

This report was prepared by the Systems Consulting Consortium, Inc. (SCC) as a service to our past, present, and future clients. The author, James M. Spitze, has served as the CIO for Xerox Data Systems, American President Lines, and other prominent private sector firms, and - as a consultant - as an Interim CIO for QANTAS, Tri Valley Growers, Lam Research, Tencor Instruments, and others. Mr. Spitze has served as SCC's Managing Partner since 1988. For more information about SCC and our services, please visit our website at www.scc.cc or contact us directly at:



The SYSTEMS CONSULTING CONSORTIUM, Inc.
P.O. Box 519, Orinda, CA 94563-0519
925-254-0760 ofc, 925-254-8524 fx, www.scc.cc