

# Current Issues

In Software Development & Economics

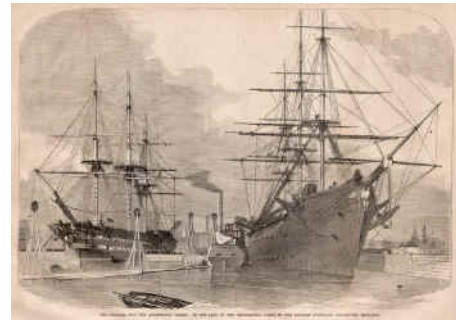
[www.softwaremetrics.com](http://www.softwaremetrics.com)

## Forces and Behaviors Inhibiting Quality

David Longstreet

*One of the most thoroughly accepted notions in psychology is the principle that behavior eventually extinguishes if not followed by reward. Many quality initiatives fail because the focus is on the technical issues of quality instead of behavioral aspects. The author identifies 5 key behaviors when addressed will improve the likelihood of a quality programs success. When these 5 key behaviors are ignored it is nearly impossible to implement a quality program.*

The early explorers would travel the globe looking for treasures. While they searched, they would stumble across products such as like tea, pasta, and cocoa, and bring them back to Europe (the old world). I have worked with organizations around the world and in my work I too have stumbled upon ideas and practices which I want to share.



I have consulted and presented papers on every continent on Earth (with the exception of Antarctica). I have advised organizations with IT budgets in excess of a billion dollars and those with only a few employees. I have worked for government agencies, companies that make dog food, organizations that launch rockets and every company in between. The bottom line, I have worked with every type of organization that relies on software.

There is a Johnny Cash song I relate to where he sings, “I’ve been everywhere, man” and he lists a series of towns, countries and places he has been. I have over 2 million frequent flyer miles and this equates to 80 times around the earth. There have been a few times I think I passed myself in an airport.

## Introduction

In this paper I address some common behaviors which prevent quality. Often a quality assurance group within an organization make attempts to better train and educate the organization on quality principles and technique. Training and education are important, but it is more important to understand the organization as a whole. In this paper, I identify 5 key behaviors which inhibit quality.

1. Negotiable Estimates
2. Measurements without consequences

3. Managing Up v. Managing (who gets promoted)
4. Costing of Software
5. Thinking you know more than the business

We could all start listing quality practices – right. I think everyone knows what those practices are. This paper covers why people don't do those quality practices in the first place.

### **Background**

Over the past 20 years or so, I have collected both quantitative and qualitative data for IT organizations. I have observed a variety of organizational patterns of behaviors, customs, rituals, myths and ways of life. I have tried to look at what people do (behaviors), what they say (choice of words or language), and the tension between what they do and what they ought to do as well as what they make (artifacts). Therefore, I gather artifacts, physical evidence, listen to stories, observe organizational rituals and myths, all in hopes of trying to understand organizational cultural patterns and consistent themes between organizations.

You may find it a bit unusual to use the terms customs, rituals, myths and ways of life in conjunction with an organization, but let me assure you organizations do have cultures, ways of life, rituals and even myths. Now I do not arrive at an organization and ask, “tell me about your rituals and myths” because most people do not see these things within their own organization. The longer you stay at an organization the less likely you will recognize organizational cultures, rituals and myths.

### **Approach (Quantitative v. Qualitative Research)**

I do not want to digress into a statistics discussion, but it is important to define the difference between quantitative and qualitative research and data. A quantitative approach is gathering hard data, “facts.” Gathering data such as hours, number of staff, defects, so on and so forth. Quantitative research or data gathering is the most common type of research. On the other hand qualitative research deals with the inquiry and exploration of social interaction and human behaviors. Qualitative data is gathered by surveys, interviews and observations.

Let me give you an example, I was working on an acquisition/merger between two French companies in two different cities. One was in the south of France and the other in Paris. The firm to be acquired was in Paris and quantitatively they were outstanding. During casual conversations over lunches and dinner, I gathered qualitative data regarding the staff. I learned some employees were life long Parisians, most of their entire families lived in Paris and their children were involved in several of activities. I reported to my client I did not believe some of the key personnel would relocate their families to the south of France and in the end I was correct.

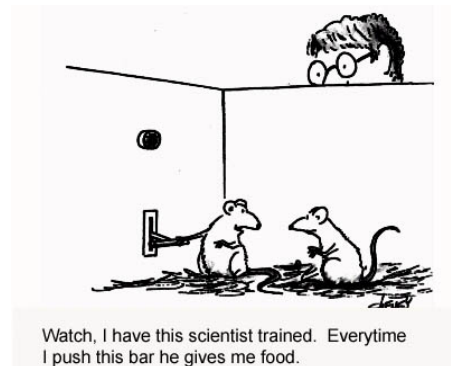
Ultimately I try to combine both quantitative and qualitative research to formulate a holistic picture of an organization.

I have learned the most by working with Venture Capitalist who are investing or considering investing in an IT organization or where one firm is in the process of acquiring another firm. Often a manager's intuition is correct, but investors want much more than intuition. They like, they demand, facts. Perhaps even more important they want correct assessments. As one investor told me, I am not paying you to be wrong.

Besides consulting I teach economics, statistics and psychology. I have learned it is very seldom technical issues cause projects to fail. In fact, most if not all problems are related to behaviors.

## Behaviors

There is a basic psychological principle stating behaviors which are rewarded are repeated, and when a behavior is no longer rewarded it will eventually be extinguished. The adage what gets measured gets done is only partially correct. What gets rewarded gets done. When there are measurements without consequences (or rewards) and/or what is rewarded is different from what is measured, then the wrong behavior will be exhibited. To really understand an organization one must understand what behaviors are rewarded.



All this being said, if an organization rewards the wrong behaviors and those behaviors do not support quality efforts, then quality efforts will never be implemented.

## Manipulating Behaviors

I was consulting with a Senior Vice President who said controlling the reward/punishment or the concept of such was the same as manipulating people. My response, absolutely it is! By understanding what makes people tick, you can control their behavior. I explained to her she is controlling behaviors already but just not aware of how.

## Don't Reward That Behavior!

I was on an airplane and in front of me was a father and his 3-year-old daughter. When it came time to take off the father worked and forced his daughter to sit down in her seat with her safety belt fastened. The child screamed and yelled. As an outside observer I understood if he let this child out of her seat, the child would learn the way to get out of her seat is by yelling and screaming. The father gave in and let the child out of her seat, but the flight attendant had other ideas and told the father his child had to be seated.

Let's look at the behavior and the reward. The father's behavior of allowing his child to stand in her seat was immediately squelched and he acquiesced to the rules. Can you imagine what happened the second time the father tried to fasten his child in the seat. The child was more resistant and screamed. After the plane took off, I put on my

headphones and listened to some music. The child eventually gave up and quieted down. The point is the father made the situation worse by giving into the behavior the first time. In other words, the father rewarded the behavior of screaming and yelling.

This brings me to the first behavior which inhibits quality.

## Negotiable Estimates

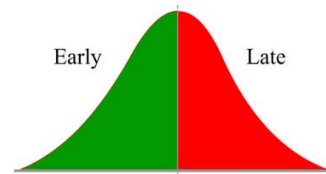
Estimates need to be non-negotiable. I mean it. An estimate should be created using a quantified method. That means there is some method to creating your estimates. You put some inputs into a formula and derive the result. The only thing you should be willing to budge on is the inputs. There are several inputs into an estimate including size (or scope) of the project, the deadlines, the staff, so on and so forth. Hence if the estimate is too high one of the inputs needs to be changed.

Unfortunately, what traditionally happens is an estimate is nothing more than a guess. The estimate has no substance at all. In other words, it is not based upon historical performance or statistical modeling. Often I am working on a contract and I ask the question, “How did you come up with your estimate?” more often than not the person actually admits it was a guess. Another answer based upon my “*vast experience*” as a software professional. In other words, questioning the estimate is the same as questioning their integrity.

One of my favorite movie scenes is in *Star Trek III: The Search for Spock*. Of course the Enterprise has some mechanical problems. Kirk asks Engineer Scott, “Scotty, how long will it take to make the repairs.” Scotty replies, “8 weeks Captain, but you don’t have 8 weeks, so I will do it in 2 weeks.” Kirk says, “Have you always multiplied your repair estimates by a factor of 4.” Scotty’s response was, “Of course. How do you think I keep my reputation as a miracle worker?”



Often I ask an estimator, what is the probability of being 20% over budget and late? Normally they have an answer. Then I ask what is the probability of being 20% under budget and early? This generally causes confusion. Don’t we want an estimate which has the highest possibility or probability of being correct? If an estimate has an equal amount of possibility to be early as late, then this is the highest probability. If on the other hand an estimate has no probability of being early, then it has just about zero percent probability of actually happening and about 100% probability of being late. A good estimate should have an equal probability of being early or late.



I don’t want to do it here, but when you derived estimates based upon past historical performance (historical data) you also should derive a margin of error. The margin of

error allows you to calculate your confidence intervals (upper and lower boundaries). By the way, the wider the confidence interval the higher the risk of the project.

Let me tell you two stories. I was working with a manager and I watched him interact with his client (it was an internal client). The manager estimated it would take 600 hours to complete a project. His client pushed and challenged him, so the manager reduced the project by 100 hours. This is a 20% reduction in time. Then his client pushes a bit more and gets the estimate reduced another 50 hours to 450 hours. This is a 25% reduction! Whoa! What behavior did the manager exhibit and reward?

I was working at Sprint for Bob who was a retired Marine Corps Colonel. I completed an estimate and when I presented it to Bob, he looked at it told me and said, "I don't believe this and go back and rework the numbers." I went back to my desk and re-checked the numbers and I considered reducing the estimate. I went back to his office and said, "I am pretty sure these number are right." Bob barked back at me, "pretty sure." Feeling a bit insecure I left his office again with my tail between my legs. I sat for a few minutes at my desk thinking. I returned to his office. Again, he said, "Longstreet I told you not to come back until you re-worked those numbers." I stopped him and said, "with all due respect, sir, I have spent a lot of time on this, I have checked and I have doubled checked and I am positive this is right." He looked up, smiled and said, "Good" He paused and said, "If you are not willing to stand up to your boss with your numbers, then why should I be willing to stand up to my boss with your numbers."

When you come up with an estimate based upon some quantitative process, then you are more likely to stand behind that estimate.

By the way, one of the benefits of good estimating is making money and utilizing resources the most efficient way possible. There is an inherent relationship between making money, utilizing resources and waste. There is not much future in losing money. I guess the largest benefit to estimating well is having less waste.

### **Not getting business**

Some of my clients are worried about estimates, which are too high. They worry about not getting the business. There is a big difference in understanding a project and cost and bidding the appropriate amount vs. bidding on a project where you are not sure. My experience has been when companies bid on projects where they are not sure they are probably going to lose money.

### **Working Overtime**

One of the primary reasons for working overtime is the project was not estimated correctly to begin with. I was working with a CIO and he told me an interesting story. First of all he came from the business side and had not IT experience. Almost every project required a lot of overtime at the end of the project. At the conclusion of each project he would ask the IT managers, "instead of working a lot of overtime at the end of the project why not work a little overtime throughout the entire life of the project." The

answer is because they did not understand what needed to be done until the end of the project.

Here is another angle on working overtime. Lets assume you are really busy and there is a need to work 10 hours overtime. There is no slack time at all. In other words you spend all your time working on today's problems, who exactly is worried about the future. If you spend all your time thinking about today's problem, then your situation only gets worse.

Working overtime will decrease morale of an organization and this does hurt productivity. There is empirical evidence to suggest individuals work output will remain constant regardless plus or minus about 20%. In other words a person will create just about the same amount of work product if they work 32 hours as they will if they work 48 hours. The amount of actual product produced will fall if a person works more than 48 hours per week and it will fall if the person works less than 32 hours per week. It is safe bet to conclude the optimal output is in the range of 32 to 48 hours.



Working overtime is a symptom of poor planning. World-class companies do not work overtime because overtime is due to poor planning. It should make perfect sense to all of you that World-class companies are very good at planning. A big part of success is planning what you are going to do, but being flexible enough to adapt to new situations. Granted there are times when overtime is required, but it should not be a company policy.

Working overtime and layoffs are not the nature of IT, but they are the nature of poor planning. Poor planning is the nature of IT.

### Productivity

Productivity is defined as output divided by resources (or inputs). This formula applies to everything regardless of what you make. I (and many others) define productivity for software development as function points / \$. Often I use hours instead of dollars, but the concept is exactly the same.

$$\text{Productivity} = \text{FP} / \$$$

Mathematically you can increase productivity by reducing hours or increasing output (function points). In practice, it is not possible to increase productivity by reducing costs. In fact there is empirical evidence to support, for every \$1 reduced in cost, you should expect a \$1.18 reduction in productivity. Henceforth, there is a negative impact to reducing costs especially by having layoffs. Anyone who has worked for an organization and survived a layoff will tell you a tremendous amount of time was spent worrying, talking and fretting about the layoffs.

## Measurements without consequences

They put one of those speeding monitors out in front of my house. Actually they are called “speed monitoring trailers.” I thought how cool is this, so I hid in the bushes and watched cars go by. I could tell if a person reacted by observing if they hit their breaks or not. About 85% of the drivers who were going at least 10 miles over the speed limit slowed down or hit their breaks. This machine sat outside my house for about 5 days. Each morning I went out and monitored (collected quantitative data) cars from about 6:30 am – 7:30 am. Each day fewer and fewer drivers hit their breaks. By the fifth day the drivers had learned there was no consequence to speeding and of those driving 10 miles per hour over the speed limit only about 20% hit their breaks. The drivers learned nothing was going to happen to them if they continued to speed.



I was channel surfing and came across the show *Nanny 911*. The premise of the show is there are families who cannot control their children, so they call *Nanny 911*. The Nanny focuses on the behaviors of the parents rather than the behavior of the children. A Nanny

trying to control the behavior of children is analogous to putting your hand in a bucket of water in hopes of displacing water. Your hand displaces the water, but as soon as your hand is removed the water goes back to its original position. The same analogy holds true for the quality consultant. It is impossible to enforce quality standards within an organization without holding management accountable.

The Nanny consults most with the parents. The Nanny teaches parents that children need to have boundaries and children need to understand there are consequences to their behaviors. Generally, the consequence for the children is timeout. Since there is no adult time out, organizations need to have other consequences. An executive manager can't put an employee in time out, so other types of consequences must be adopted.

I often ask what are the consequences of not following established procedures and policies. What are the rewards for following procedures and policies? Do performance reviews actually include activities which support the quality program?

Too often those “heroes” get rewarded who do not follow procedures and policies. They were able to overcome the “obstacles,” “the road blocks,” so on and so forth. Everyone knows “road blocks” are code words for the process. It is great they delivered a project in a short time frame or helped a customer, but we need to look at the end result. Did they create software thrown into production, undocumented, making the production

software of lower quality. In essence they are actually making the situation worse, but rewarded for it.

I wonder, do you think anyone would watch a Quality 911 reality show?

The bottom line, quality folks can work, talk, train, cajole, sell and communicate until they are blue in the face, but if executive management does not enforce quality policies then quality is not going to happen. By enforce, I mean to have consequences. A consequence does include loss of job. Those successful executives who I have worked with do fire people for not following the process.

Ideally an organization will reward the appropriate behaviors instead of punishing behaviors.

If there are measurements put in place without any consequences, then behaviors will temporarily improve. BUT behaviors will return back to “normal” as soon as it is realized there is no consequences.

The reason people refuse to follow a process or procedures is because they have learned this is not what is rewarded. This brings me to the next behavior.

## Managing Up v. Managing (who gets promoted)

The number of managers in IT which are promoted to their levels of incompetence (actually beyond them) is pretty high. Most managers have reached their level without ever using measurements or using quality principles so these managers default to what has worked their entire careers.

I believe humans always try to maximize their own situation and their own careers. I was asked to participate in a very large merger between two telecommunications companies (not Sprint). My assessment would be used as part of the criteria to determine which applications would be kept and which would be thrown out. One of the senior executive VP's for one of the telecommunications companies asked me a direct question and I am certain my response prevented me from doing the assessments. She said, “You are going to assemble a pretty high powered team of individuals and they are going to do some good analysis and write a report. How do I know you will not make my applications obsolete and me along with it.” This executive was very charismatic and this is what got her promoted to her current level. She felt there was less risk in using what worked in the past instead of trying something new. She was totally secure in her approach.

On the other hand, there is a lot of IT management, which are very insecure and defensive. The way I consult and approach them is totally different than I would a secure person. Now, there are some managers and developers who are complete idiots. I might even go as far as calling them the village idiot or the company idiot. By the way, most people do not like being called an idiot and I would recommend you refrain from calling management or your fellow workers idiots.

Several years ago before I learned to be more diplomatic, I was in a meeting with a very senior executive and I thought, well for a lack of better words, he was an idiot. In one of our many meetings he said to me, “Every time I talk I get the impression you think I am an idiot.” I sat quietly and did not respond. Then he said, “what, you have no response.” So I said, “your observant.” Trust me, that one little comment did not help my career.

By the way, there is a difference between ignorance and being an idiot. Once an ignorant person is informed and educated they do not let their ego get in their way. They learn and they correct the problem. On the other hand, an idiot is a person who is not persuaded with facts and figures. They let their egos prevent them from changing direction and adopting new policies. Generally they are very insecure and surround themselves with people with less competency than they have.

In many organizations managing up is rewarded instead of actually managing. Often there is too much focus on what meetings one attends, how one speaks in the meetings instead of actual work delivered. If quality principles are part of the review process, then it is basically impossible to get promoted without following them. When quality principles are not part of the review process, then individuals are promoted and rewarded for different reasons.

Another type of manager inhibiting quality even more is the “Dorothy” management style. At the end of the movie “The Wizard of Oz,” Dorothy holds Toto and clicks her heels together and repeats over and over again, “There is no place like home.” The Dorothy style of manager clicks their heels together and wishes the project would go well. These types of managers are the folks who see a project plan as overhead and would rather manage the project by the seat of their pants. In fact, they see most everything as overhead such as metrics, project plans, documentation, actually the entire process is seen as overhead. It makes me wonder how are they actually managing the project?

## Costing of Software

This is my number one recommendation to improve quality and productivity. There is a lot to this idea, but basically the way you charge for software inhibits quality. Even though we all know the unit cost for software is not the same throughout the software lifecycle often the unit price and/or price is constant. Assuming of course you know your unit cost! Unit cost rises as the project progresses down the software lifecycle. A rising unit cost is consistent across industries, but software is the only industry where it is common to charge a consistent unit cost (or total cost).

It is not uncommon that software is delivered at the same cost and close to the same day. How cool is this! What incentive does your client have to come to the table (the initial requirements process) with everything defined and completed? What if your organization advertised this.... *We charge the same for changes made in requirements as in testing* this is what many IT organizations do. How long would someone stay in the building & construction business adopting this same practice?

There is a sharp change when an organization starts charging \$1,000/fp in requirements and \$10,000/fp in test. There is a huge incentive to get things done and defined early in the process. If it is clear the cost is 10 times more during testing than in requirements, then there is an incentive for clients to have all the requirements done prior to moving onto the next phase; if and only if, they are charged accordingly. Again, this rewards the right behavior and that is getting things defined before you start. Sometimes the IT world is just crazy. It reminds me of the old cartoon where the manager says, “you start coding and I will ask what they want.” For a company to move to world class, they must start defining their costs like other industries.

### Communicating Hourly Rates v. Unit Costs

IT needs to communicate in unit cost instead of hourly rates. What do you care about when having a home built? Do you care about how much the workers make per hour or do you care about the \$ /per square foot and total cost. Of course, there is a relationship.

When IT only communicates hourly rates, this is what clients (internal & external) will focus on. This is one reason why some IT organizations are outsourced. They can’t explain nor they do know their unit costs. The only thing which can be compared is the hourly rate. The client (or customer) sees really low rates and just can’t believe it. This is the driving force for outsourcing. There are many IT organizations that do not know how much it costs to develop software.

Not knowing how much something costs to develop and only knowing the hourly rate is dangerous. The unknown is the length of time it takes to get the job done. I often ask people if they have ever had a house built. Then I ask them would they be OK if their builder adopted the same rigor as their software projects.

I try to encourage my clients to adopt a firm fixed bid approach instead of an hourly rate.

Let me tell you a secret. Those organizations who fix their requirements process, see a leap in productivity. I have had clients move from 40 hours per function point to around 8 hours per function point. In other words, they reduced their costs. In other words they reduced their unit costs 5 times.

No matter how hard you try, some clients will come late to the game with new requirements and changes. They will cry, scream, whine, and some may even hold their breath. They will accuse you of not being a team player. Unfortunately this behavior of coming late with requirements has been rewarded time and time again. They will want to see you terminated and they will escalate this to the highest levels of management – this is a guarantee!

## Thinking more you know than the business

I hear folks in IT say things like, “I know more about the business than the business person”. Now I am not suggesting it is not

*He who knows best knows how little he knows." Thomas Jefferson*

true, but it is an indicator of a bigger problem when your technical staff with a higher level of business knowledge than your business staff.

The credit card machine at my local grocery store has a taped on “OK” over the enter button. I am certain there is some IT person still insisting there does not need to be an OK button.

## Unskilled and Unaware - *Ignorance is bliss, but not in this case.*

When people are incompetent in the strategies they adopt to achieve success and satisfaction, they suffer a dual burden. Not only do they reach erroneous conclusions and make unfortunate choices, but their incompetence robs them of their ability to realize it. They are left with the mistaken belief they are doing fine. Much worse is the fact they will never learn from their mistakes.

*“Ignorance more frequently begets confidence than does knowledge,”*  
Charles Darwin.

For example, consider the ability to write grammatical English. The skills enabling one to construct a grammatical sentence are the same skills needed to recognize if a grammatical mistake has been made. Hence, if a person does not have grammatical skills they may actually think their grammatical skills are just fine.

In essence, those skills making one competent in a particular domain are often the very same skills necessary to evaluate competence in the domain. Because of this, incompetent individuals lack what cognitive psychologists call “*metacognition*” or self monitoring skills. In other words they are just too ignorant to realize they have a problem.

Many IT companies are left with the mistaken belief they are doing fine when in actuality they are not. These organizations actually believe they are doing a “*quality*” job because they cannot define quality. Since they do not have a handle on quality, they do not have the fundamental skills necessary to realize they are actually doing a poor job.

No one in the company is actively involved in trying to understand problems and formulate solutions. Often companies will collect a wealth of data and sometimes they actually write post mortem reports. The problem is they never go back and review the data and read the reports. No one tries to understand what went wrong, look for patterns and then develop plans to solve those specific problems. In fact, a lot of companies problems and potential solutions are right there in front of them, but they lack the fundamental competency to recognize this and then formulate a strategy. In other words, they are too ignorant to realize they don’t know.

### Fixing the problem

It is often difficult to implement quality programs because senior executives and other staff members within the organization actually hold the mistaken believe they are doing a good job. Typically quality assurance staff try to demonstrate the truth by producing

charts and graphs, but to their surprise these charts and graphs are often ignored. A better strategy is to attempt to provide enough education so they can recognize there is a problem.

### **Improving productivity and quality**

I have never seen an organization improve productivity by reducing costs. Productivity is defined as the ratio of Functionality/Effort. You can improve this ratio by either reducing cost or by increasing functionality. For every dollar cut it reduces productivity about \$1.18. Trying to improve productivity by reducing costs actually has the opposite effect.

### **Quality when it is too late**

There are organizations that try to implement quality programs when it is too late. It is like trying to establish a fire evacuation plan when your building is already on fire. There are just so many things wrong there is no way to fix it. It is best just to tear it down and build it back up from scratch. This is what happens to many software applications they reach a point of critical mass. In other words, it has become so expensive to maintain and fix the application it is best just to trash it and start over.

### **Summary**

Over the years I have learned it is seldom-technical issues, which inhibit project success. Most of the time it is people issues. There are fundamental behaviors that need to change, but changing behaviors is not easy to do. It is difficult to lose weight. For those addicted to gambling it is nearly impossible for them to kick the habit.

### **References**

- Buehler, R., Griffin D., & Ross M. (1994). Exploring the “planning fallacy”: Why people underestimate their task completion times. *Journal of Personality and Social Psychology*, 67, 366-381.
- Creswell, John. [Qualitative Inquiry and Research Design Choosing Among Five Traditions](#). Sage Publications. London, 1997.
- Gill, John & Johnson, Phil. [Research Methods for Managers](#). Sage Publications, 2002.
- Herman C. P. & Polivy J. (2002) If at First You Don’t Succeed, False Hopes of Self-Change. *American Psychologist*, Vol. 57, No 9, 677-689.
- Kruger, J. & Dunning, D. (1999). Unskilled and unaware of it: How difficulties in recognizing one’s own incompetence lead to inflated self-assessments. *Journal of Personality and Social Psychology*, 77 1121-1134.
- Lecci, L. Okun, M.A. & Karoly P. (1994). Life regrets and current goals as predictors of psychological adjustments. *Journal of Personality and Social Psychology*, 66, 731-741.