Summary of DeMarco and Lister "Peopleware. Productive Projects and Teams"

by

David S Burris

Ph.D. in Computing Science

Certified Computer Programmer

Certified Systems Professional

Certified Computing Professional

csc_dsb@shsu.edu

Historical Theories of Value

Spanish:

There is only a fixed amount of value on the earth. The path to accumulating wealth is to learn to extract it more efficiently from the earth or peoples backs.

The Spanish Conquistadors enslaved the Aztec and other civilization in South America to accomplish this goal.

English:

Value can be created through technology and ingenuity (the English had the misfortune of landing in North America). The result was inventions such as the Cotton Gin, Spinning Ginny, automated weaving looms, and the Industrial Revolution.

The industrial revolution led to the creation of cities. We are asked to overlook the unsafe work environments known as sweat shops, child labor, and economic enslavement of their lower classes.

The Spanish Theory for Accumulating Wealth is Alive and Well in many Organizations!

It's called Unpaid Overtime or occasionally as Salary!

Increasing productivity should mean achieving more in an hour of work!

Too often it means extracting more form an hour of pay.

Productivity through unpaid overtime is a fraud. Long term, there is no such thing as "Unpaid/Free Overtime."

Frequent overtime results in periods of compensatory "Under Time." <u>An employee may be bodily present but they are mentally absent</u>.

Frequent overtime results in poor work attitudes. Your best employees will eventually leave.

Managers frequently brag about the benefits of "overtime" on a project in terms of cost reduction.

Managers frequently fail to factor in the very high cost of employee turnover including advertising, interviewing, training, and the time it takes to bring new employees up to speed after they are hired. In addition, a new employee will typically require large chunks of time from existing employees to become productive. Hence existing employee productivity drops.

Manager:

Consider a manager who frequently bullies or cajole their professionals into long hours, trick them into accepting hopelessly tight delivery schedules, shame them into sacrificing any and all to meet a cruel deadline. This process leads to job frustration! (DEC Tops 20 Operating System)

Message from the Home Front:

1) Life is passing you bye.

2) Your spouse is starting to look elsewhere!

3) Your children do not recognize you!

Even the most dedicated workaholic will eventually realize you only get one shot at life and there are things more important then this silly project.

Your best employees will leave. The cowards, those who cannot hold a job elsewhere, and those that know how to cheat the system will remain!

"Overtime is like sprinting: It makes sense for the last 100 yards of a marathon for those with any energy left. If you start sprinting at the first mile, you're just wasting time."

Tom DeMarco and Timothy Lister – "Peopleware, Productive Projects and Teams"

DeMarco and Lister "Coding War Games"

Measurement of Productivity

Teams of software implementers from different organizations compete against each other to complete a series of benchmark coding and testing task in minimal time with minimal defects!

Rules:

1) **The basic unit is a pair of implementers from the same organization. The pair members do not work together, but rather in competition against each other to represent their organization. They also compete against all other teams. Hence an organization is represented by at least two people.**

2) **Both pair members perform exactly the same work designing, coding, and testing a medium sized program to a fixed specification.**

3) **Participants must record how time is spent on a log. They are guaranteed that management will never see the log. It is anticipated that participants will have meetings, hot spots that must be taken care of, and other activates that will keep them from devoting 100% of time to the project.**

4) **After all participant testing is completed; <u>the products are subjected to a standard acceptance test</u>.**

5) **Participants work in their normal work areas using the same hardware, software, and other tools they would normally use.**

6) **All individual results are kept confidential from management. Management gets a summary. Individuals are allowed to see how they stacked up against the competition.**

Individual benefit by seeing how they stack up against other professionals.

Companies hope to determine their performance relative to other companies.

Other questions include:

1) How wide is the range in performance?

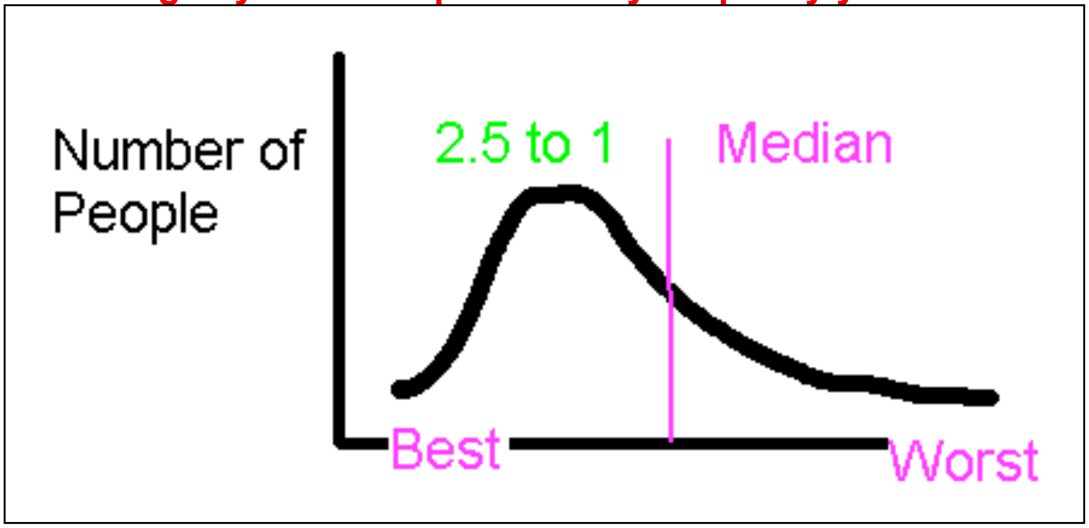
2) Do the best people tend to gravitate to a few companies or are high performers uniformly distributed?

3) Are there consistent factors that tend to lead to higher productivity?

4) Should more resources be expended on software development or are we ahead of other companies?

From 1984 to 1986 over 600 developers from 92 companies participated.

Using any metric of productivity or quality you desire!



General Results

Though asked to utilize typical (average) personnel, it might be anticipated that companies would use their best people to represent them. You would then expect a very small spread in observed performance.

Individual Differences: Rule of Thumb

1) **Count on the best performer outperforming the worst by about ratio of 10:1 or more.**

2) **Count on the best performer being about 2.5 times more productive than the median performer.**

3) **Count on the half that is average out producing the lower half by more than 2:1!**

This is consistent with result in studies by Sackman, Erickson, and Grant done in the late 60's and early 70's.

A few good employees are better than an army of average employees. The best performers do tend to gravitate to the same companies.

What does affect quality and productivity?

The hope is that the answer to this question will allow individuals and companies to improve their performance.

Traditional Ideas:

1) **Choice of programming language.**

2) **Experience is an asset.**

3) **Increasing development time helps reduce the number of defects in the final product.**

4) **Salary.**

(Available hardware and operating systems.)

Choice of programming language.

Programming language is the traditional choice of the propeller beanie heads in the organization. Reality: language has very little effect on either productivity or quality. Those who used older languages in the study such as Fortran or COBOL did essentially as well as those who used new languages such as Pascal or "C."

The productivity and quality spread within each language group was much like the overall spread of performance.

The only exception was assembly language. Individuals using assembly language were badly left behind all other groups (but they are used to this phenomenon).

Higher reported productivity rates in languages such as C++, Ada, and other new languages is apparently not a result of the language but rather the use of a development paradigm such as Jackson's Technique or Object Oriented Design and Programming. <u>The actual language used to implement the paradigm is not important</u>.

Years of Experience.

It is widely believed that experience is an important determinant of productivity and quality.

People who had ten years of experience in the study did not out perform those with two years of experience.

There was no correlation between experience and performance except those with less than 6 months experience with the language used in the exercise did not perform as well as the rest of the sample.

A person right out of college is likely to perform better than a person with 10 or more years of experience.

Individuals do not perform to their maximum capacity until they have learned and adjusted too the corporate culture!

These results compare favorably to other studies.

More development time helps reduce the number of defects in the final product.

It is widely believed that if you wish to improve quality by reducing the error rate, you must accept longer delivery times. That is there is a time versus quality trade off.

Nearly a third of the participants in the war games completed the exercise with zero defects. As a group, the zero defect workers paid no performance penalty for doing more precise work. In fact, they took slightly less time on the average, to complete the exercise than those who had one or more defects!

Salary

The most productive workers are remunerated accordingly.

Salary levels varied widely over the sample. There was a weak relationship between salary and performance.

The half above the median made less than 10% more than the half below the median.

Mediocrity pays!

Remember, the half above the median is approximately twice as productive and a third more likely to produce zero defect work.

The performance spread at any given salary level was nearly as wide as the whole sample.

Stated another way: In most organizations, people are not paid according to the quantity and quality of work they do. In most organizations, mediocrity pays! Apparently management does not know how to separate the wheat from the chaff.

What does affect productivity and quality?

Environment of the Best and the Worst Performers in the Coding War Games

Environmental Factor	1st Quartile	4th Quartile
How much dedicated work space do you have?	78 sq. ft.	46 sq. ft.
Is it acceptably quiet?	57% yes	29% yes
Is it acceptably private?	62%	19%
Can you silence your phone?	52%	10%
Can you divert your calls when needed?	76%	19%
Do people interrupt you needlessly and often?	38%	76%

This table was based on participant surveys.

** The best performers tend to gravitate to organizations that provide a better working environment.

** Remember those employees above the median are about twice as productive as those below the median. They are one third more likely to produce zero defect work and do it in less time.

Cutting cost on office equipment such as desks, chairs, computers, large monitors, software products, documentation, and office space has a high hidden cost!

"I get my best work done in the early morning before everybody else arrives!"

"In one evening, I can accomplish two or three days worth of work!"

"The office is a zoo all day, but after 5:30 P.M., things settle down and you can really accomplish something.

Do you or your employees are making these statements, arrive early or stay late to accomplish their jobs?

Are employees hiding in conference rooms, storerooms or elsewhere?

Do employees feign sickness just prior to crucial deadlines to stay home and catch up?

If so it is quite possibly an indictment against your leadership and management abilities!

It is not unusual for these conditions to exist. What is unusual and hard to believe is that everyone knows you cannot accomplish anything in the office no effort is made to correct the problem!

Even if you can prove that two workers can normally function adequately in 100 square feet, the noise level will be substantially higher. You risk losing the more valuable creative leap when it is needed!

DeMarco and Lister

Study for Building the

IBM Santa Teresa Research Facility

in California, (IBM SYST J, VOL17, NO1, 1978), by an Outside Contractor

Before building the IBM Santa Teresa Research Facility in California, IBM employed an outside contractor to study the work environment required by professional such as engineers, software specialist, and other technical personnel. I have summarized some of these recommendations as well as the recommendations of similar studies.

1) **100 square feet of dedicated workspace minimum.**

2) **At least 30 square feet of horizontal work surface per person. If a person works on more than one project at the same time, the work area should be sufficiently large for each project to be completely laid out independently without any overlap.**

3) **Noise protection in the form of enclosed offices or minimum six-foot partitions.**

4) **A combination of one and two person offices with group meeting rooms.**

Three person or larger offices should be avoided. Some people have to be able to hear a heart beat even if they do no talk to them. Some people work best alone. Frequently two people on highly interrelated portions of a project must be in the same room to reach maximum productivity due to the need for communication. If an office contains three or more people, there is a good chance that one person just has to talk about a non-work related activity interrupting the others. When this person is ready to start working, another employee has reached a frustration level where they now insist on being the next person to discuss non-work related topics. Exceptions: SHSU networking group Windows, Linux, and administrative VAX VMS.

Other Considerations

1) Workers in acceptably quiet, private, non-interrupted environments are one third more likely to produce zero defect work. One employee wants to be able to hear a pin drop, another required country-western in the background while another prefers heavy metal or Mozart.

2) Employees should be able to control their environments individually (heating and cooling).

3) Indirect lighting that is evenly diffuse with no shadows is best. It is hard to concentrate when a tungsten lamp is heating your shoulder and forming shadows on your work. The 60 cycle per second hum of poorly maintained fluorescent lighting reduces productivity.

4) Outside noise sources reduce productivity (jack hammer, student center speakers in quad).

5) Open space such as patios, picnic tables, comfortable break areas, and other amenities help employees to relax and perform at their highest levels.

6) A white board for scheduling activities, notes, or problem solving increases productivity for many employees.

7) Do not cut corners on the quality of chairs, desks, workstations and other equipment and physical amenities.

8) Do not save money by purchasing one copy of the documentation or professional magazines to be shared. Supply every employee with their own copy of all tools indigenous to their job.

9) Bookshelves, plants, windows, and other amenities increase job satisfaction and productivity.

10) Do not allow ‘furniture police” in your organization!

Office Environment Requirements

for Information Technology Workers

IBM – Santa Teresa pre-construction Think Tank Study:

How Information Workers spent their time with others;

Work Mode	**%**
Working Alone	**30%**
Working with one other person	**50%**
Working with two or more people	**20%**

Recommendation:

A) Mostly one and two person offices, avoid 3 or more people in same office. Partitions frequently have a negative effect on productivity and quality.

B) Must supply group areas: Conference rooms with white boards, etceteras.

C) **Corporate culture:** You do not disturb people in the conference rooms. Glass window or doors allow others to see the facility is in use so they will not disturb the occupants. Do not disturb an individual with their cowboy bandanna draped over the front of their cubicle.

Flow

Psychologists define flow as a deep, nearly meditative involvement with work attained during single-minded work with no interruptions. This is an individual's most creative period.

Flow takes approximately 15 minutes to attain. A five minute phone call cost 20 minutes, 5 for the call and 15 to re-attain flow. A dozen phone calls results in loss of a half days work.

Most Creative Period

Recommendation:

A) Allow phone to be silenced and re-routed.

B) Do not use four-foot partitions to form work spaces. If partitions must be used, they should reach all the way to the ceiling.

C) Do not locate information workers in high traffic areas such as around water coolers or break rooms. Avoid "broadcast" intercoms and phones that everyone hears ringing.

Environmental Factor

Uninterrupted hours
E-Factor = ----------------------------------
Body present hours

Work should not be measured in *"body present hours,"* rather in uninterrupted hours. Remember the cost of flow. A phone call every 20 minutes, the intercom page heard by everyone, or someone stopping to say hello will guarantee that no work is accomplished in an 8 hour day.

E-factors below 40% result in frustrated employees. The best employees will quit and look for a better work environment.

Technical workers like their jobs. *The ability to seek gratification from their work is more important than money.* The best employees migrate to companies that provide the best work environments.

Evidence suggests that top performers in a company typically out perform the low end by a ratio of 10:1 to as much as 20:1.

The group of employees above the median generally produces at least twice as much as the group below the median!

Money saved short term by skimping on space, creature comforts, or decisions that increase distractions in this group is lost everyday from that point on in the form of productivity and quality!

Cheap is extrodinairly expensive!

(DeMarco & Lister)

Consider holding important meetings off-site where there are no phones and the rest of the company does not know how to contact the desired individual directly!

Subliminal Suggestion

"Subliminal Suggestion" is the use of background music, color schemes, etceteras, too influence productivity.

Production Line versus Professionals

A) Music
B) Wall Colors (dark versus warm, pastel versus harsh)
C) Films (Jaws) {fall off seats, record popcorn & drink sales}

1960's Cornell University research indicates that workers in noisy – interrupted environments are measurably less creative.

The use of white noise/mood music can significantly influence productivity in production line environments. It frequently reduces productivity in white-collar professional environments.

Much of the repetitive daily work activity of the professional is done as serial processing using the symbolic manipulative abilities of the left brain.

The creative leap requires right-brain function. If it is listening to 1001 strings of Muzak or noise, the creative leap is lost.

Information Technology Workers (IT) have been shown to be 30% more likely to produce zero defect work when provided with a quality environment.

Typical Quality Measures

1) Mean-Time-Between Failure (MTBF). How long on the average does a product function properly before breaking? This should probably be weighted by the number of units of the product in use, product size and cumulative actual time of use.

2) Mean-Time to Repair (MTTR). When a product breaks, how long is required on average to fix it.

3) Systems Availability = MTBF / (MTBF + MTTR)

4) User Change Request Rate measured per volume of code or delivered functionality.

   A) New functionality (missed specification, skipped in design or implementation: looking for patterns).

   B) Modification of existing functionality (missed in specification, not understood correctly at the time of specification, changing environment, poor communication).

   C) Functionality not utilized.

Quality versus Excellence

As designers and implementers, we tend to tie our self-esteem strongly to the quality of the product we produce – not the quantity of the product.

For some reason there is little personnel satisfaction in turning out large amounts of mediocre products even though that may be exactly what is required for a given situation.

For example, producing the best user interface at higher cost at a latter date for a select market of connoisseurs may not be nearly as important economically as producing the first user interface cheaply and mass marketing it to a less demanding group of connoisseurs. But it will be more satisfying?

If you suggest constructive criticism with respect to the quality of the product you risk the developer interpreting your comments as a vicious personnel attack.

A major arouser of emotion in the work place is a perceived attack on self-esteem.

Any action by management that is perceived by the development staff as affecting quality is likely to invoke a negative emotional response!

What is acceptable quality?

Builder: Since their self-esteem is tied to the product, the minimum standard is typically the highest quality perceived to have ever been attained in the past.

Managers: Managers tend to think of quality as just something that may be supplied in varying degrees according to the pocketbook of the market place. It's like toppings on a baked potato. The more you are willing to spend the more topping that will be supplied.

The Market: "The market does not give a damn about that much quality. If the features you depend on work, who cares about the rest." People may talk in glowing terms about quality or complain bitterly about its absence. The acid test however is what they are willing to pay for quality.

Fact of life for developers: Quality is accomplished during specification, design, and implementation. It may not be economically added as an afterthought at a later date, if at all.

Standard Western Quality Quandary: Managers / development staff must determine quality standards based on what the markets wants balanced by what they are willing to pay (or our impression of this tolerance).

The Flight from Excellence

Allowing the standard for quality to be set by our perception of what the buyer/market place is willing to live with and what they are willing to spend as opposed to higher quality goals represents a flight from excellence.

Managers typically associate high quality with excessive time spent in development and testing.

A typical manager might be quoted as saying "Some of my folks would tinker forever with a task all in the name of quality!"

For example, assume we can extrapolate from past empirical data that the mean time between failures of 3.4 weeks could be extended to 2 months if we spend 3 more weeks in development.

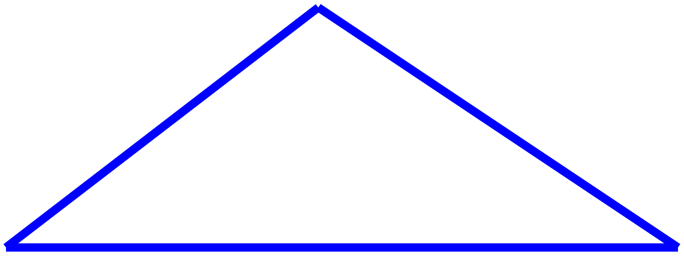
Expect to see some squirming (Olympic class hemming and hawing) from potential customers who want the product now as well as management desiring to sell the product. Typically they will state they are as quality conscious as the next person but three weeks represents real money to them.

Note they seldom consider the lost revenues that are required in the future to correct mistakes caused by using software not yet ready for the market (,e.g. DBase II, III, and IV - same control key saved file in version II deleted file in IV)!

Development Trade Offs

Frederick Books “The Mythical Man-Month”

Schedule
(delivery date)



Functionality
(feature richness)

Quality
(absence of defects)

To minimize one goal you must sacrifice one or both of the others!

Larry Putman in “Measures of Excellence” demonstrated the tradeoffs between schedule, functionality, and quality is not linearly related. If you double the number of required features and hold quality (including cost and manpower) constant then the schedule is likely to considerably more than double.

Quality is not free. A policy of “quality as time permits” almost assures that no quality at all will sneak into the product!

Affect of Deadlines!

Assume that we are not going to be able to meet a hard deadline under current circumstances. What can be done to improve our situation?

1) **Increase staff and or/other resources. This is normally not an option.**

   Even if it is an option, large short-term increases in staff are seldom productive. They actually slow development. New people are not immediately productive as they must determine what has been done and what needs to be done. Unfortunately this requires the personnel on the project to stop working directly on the project to help. The result is at least a short-term drop in productivity.

2) **Reduce the software functionality.**

3) **Reduce Quality!**

Quality reduction resulting from efforts to save time is not always immediately obvious. It creeps in from such activities as:

a) **Reduced time in the specification phase prior to the commencement of development.**
b) **Elimination of prototyping studies.**
c) **Reduced customer and technical inspections.**
d) **Reduced time for testing including creating the test plan, number of tests, and types of test.**
e) **Pushing some aspects of the project from development to maintenance and modification (enhancement - close current year and start next).**
f) **Not closely investigating the ramification of changes in requirements.**

"Just Good – Enough Quality"

as reported by Edward Yourdon in the "Rise and Resurrection of the American Programmer" (1996, Prentice Hall, ISBN 0-13-121831-x) (previous book "Decline and Fall of the American Programmer).

When queried about MS-Word, a spokesman for Microsoft responded as follows:

Version 1 was approximately 27,000 lines.

Version 2 was approximately 2,000,000 lines.

When asked:

- **a) How many people where on the project?**
- **b) How many lines of code were generated per person-day?**
- **c) How much of the code written was deliverable and how much was discarded?**
- **d) How efficient was the project?**
- **e) How many bugs were shipped and how could it have been done better?**

MS Executive Response: Who cares if we used 100 or 200 programmers? What difference does it make if the project was efficient?

We sold 17 million copies!

If you sell enough copies, efficiency is not an issue!

"Good Enough" is the Enemy of Quality!

Traditional Western Business Attitude

Managers must balance the desire of the builder to produce a high quality product free from errors with the realities of the market place.

These realities include the level of quality with which the customer is willing to live in conjunction with the price they are willing to pay!

We must also consider the influence of the competition when establishing software delivery deadlines.

It does no good to deliver a perfect product six months behind an acceptable product from the competition. ***The consumer will not wait, they will not change products after the startup cost incurred to use the competitor's product, and they are usually not willing to pay extra for a perfect product*****.**

Parkinson's Law: ***Work expands to fill the allocated time*****. Without deadlines the worker bees will not work at maximum capacity.**

Deadlines are an effective motivational tool when properly utilized by managers

Survey

Based on “Peopleware: Productive Projects and Teams” by Demarco and Lister

Ask 100 people on the street:

1) What organization or culture or nation is associated with high quality?

2) What organization or culture or nation is associated with high productivity?

The response to both questions is frequently Japan, Germany, and Switzerland.

If the same countries are selected for both questions, it flies in the face of the assumption that there is a tradeoff between high productivity and high quality!

In fact many of the most productive companies and nations apparently subscribe to the notion:

“Quality far beyond that required by the end user is a means to higher productivity and reduced long term costs!”

“The tradeoff between price and quality does not exist in Japan. Rather the idea that high quality brings on cost reduction is widely accepted.” Tajima and Matsubara, respected commentators on the Japanese phenomenon.

Hitachi Software and part of Fujitsu have the following policy: The project team has the power to veto delivery of software they believe to be not-yet-ready even if the client insists on delivery and is willing to make immediate payment including a bonus. (DBASE)

Hewlett-Packard has a reputation as a company that reaps benefits from increased productivity due to high builder-set quality standards (calculators/instrumentation)**. Builders know they are part of a corporate culture that delivers quality beyond what the market/customer requires. Their sense of quality identification works for increased job satisfaction and some of the lowest turnover statistics in the industry. 3M, Monsanto, Compaq** (suit case)**, and other companies have experienced similar advantages at points in their corporate life.**

Unrealistic project goals do not act as an incentive or challenge. Unreasonable or unrealistic, no win projects only lead to frustration and sacrifice of quality.

People under pressure (especially time) seldom work better or smarter; they just cut more corners faster!

If you know you have to work this weekend anyway, why work harder today!

Parkinson's Law

"Work expands to fill the time allotted for the task."

Northcote Parkinson (1954) was a British humorist like Will Rogers in the United States. He contrived the law attempting to describe the British Post Office System. In his view it was a bureaucracy where you could not get ahead by merit or fired for lack of merit. The only way to the top was via death and whom you know. There is nothing scientific about the law. (unionized universities)

Parkinson's law may be true on an assembly line or in a bureaucracy. It is seldom true for computer professional who derive satisfaction form their jobs.

A variation of Parkinson's law is likely true.

Organizational busy work will expand to fill the working day! (public school teachers)

A good manager attempts to shield their people from as much of this "busy" work as possible.

Your people could have found jobs that pay just as much for less effort in college. In fact they may not have even had to go to college. They have sacrificed a portion of their lives and monetary commitment to reach their current level of performance and job position. Usually because they genuinely enjoy what they are doing! <u>A manager's job is to allow them their enjoyment</u>.

Results of a 1985 Survey by Michael Lawrence and Ross Jeffery, University of New South Wales, Australia.

The goal was to determine the effect on productivity by various influences on estimating. These include:

1) Managers are the best estimators for projects. Developers should not be given more time than is actually required or the additional time will be squandered. In fact many believe the deadline should be a little shorter than what is actually required to encourage high productivity and some voluntary overtime to meet the attainable goal.

2) Development staff is the best estimators of time to complete a project. They know what they can accomplish better than anyone else. Hence, deadlines for project deliverables should be established by development staff (<u>an almost folkloric belief in our industry</u>).

3) Deadlines should be set by a systems analyst who will have absolutely no involvement in the project other than to estimate the most realistic delivery schedule.

All measurements of productivity were made using Barry Boehm's CoCoMo methodology. CoCoMo is a unit less number. The higher the CoCoMo number the higher the productivity.

Actual industrial projects were used to collect the data.

Results of 1985 Survey by Lawrence and Jeffrey

Effort Estimate Prepared by:	Average Measured Productivity	Number of Projects
Developer Alone	8.0	19
Supervisor alone	6.6	23
Developer and Supervisor Compromise	7.8	16
Systems Analyst (not involved with project).	9.5	21
Control Group (No deadline, wake me when through)	12.0	24

Projects where time pressure was not used as an inducement to increase productivity experienced the highest productivity.

The best management strategy might be "Just wake me up when you are done."

The best estimate of actual performance was by a systems analyst that did not participate in the project except to project the most likely time to completion with a fixed set of resources.

Developers tend to under estimate the time they require to complete a project. Apparently their egos get involved.

Managers tend to be more interested in what is desired, or must occur, as opposed to what is possible with a fixed set of resources.

The proper function of management is not to make people work, but to make it possible for them to work!

Tom DeMarco "Peopleware: Productive Projects and Teams"

Weinberg-Schulman Experiment

“Human Factors,” 1974, Vol. 16, pp 70-77

Metric Premise

Rational, competent men and women can work to effectively maximize any single observed indication of success.

Assign five project teams to maximize one of five project goals.

Goal	Shortest Completion Time	Minimize Program Size	Minimize Data Space	Maximize Program Clarity	User Friendly Interface
Shortest Completion Time	1	4	4	5	3
Minimize Program Size	2 – 3	1	2	3	5
Minimize Data Space Used	5	2	1	4	4
Maximize Program Clarity	4	3	3	1 – 2	2
User Friendly Interface	2 – 3	5	5	1 – 2	1

Highest rating is 1, the lowest rating is 5.

Teams generally achieved their assigned goal.

To achieve the assigned goal they typically sacrificed other goals.

The team assigned to “shortest competition time” sacrificed the most with respect to other goals.

Desirable Goal for Most Projects

Maximize the amount of delivered function (weighted by years of useful system life) per dollar of total system lifetime cost.

The default goal for projects that are not monitored for all development and quality aspects is "Completion in the shortest possible time." Time is the only variable directly observable by management that does not require special effort!

Control of a software project requires measurement of project activities and deliverables! That which is not measured tends to drift to non-measured portions of the projects. For example if development is measured but not maintenance, features not immediately required by the user tends to migrate from development to maintenance / enhancement!

The cost of software is a polynomial: cost = specification + design + development + maintenance + enhancement + ease of use + •••.

The major goal for most software managers should not be goal attainment, but rather "Goal Alignment."

Tom DeMarco: "Controlling Software Projects," Yourdon Press