

Quality Software Management – Listing of Laws, Rules and Principles

Gerald M. Weinberg

1 Volume 1 – Systems Thinking

Crosby's Definition of Quality Quality is 'conformance to requirements' (page 5)

The Quality Statement Every statement about quality is a statement about some person(s).
(page 5)

The Political Dilemma More quality for one person may mean less quality for another.
(page 6)

The Quality Decision Whose opinion of quality is to count when making decisions? (page 7)

The Political/Emotional Dimension of Quality Quality is value to some person. (page 7)

The Inadequate Definition of Quality Quality is the absence of error. (page 9)

Crosby's Economics of Quality "It is always cheaper to do the job right the first time."
(page 19)

The Quest for Perfection The quest for unjustified perfection is not mature, but infantile.
(page 21)

Building's Backward Basis Things are the way they are because they got that way. (page 22)

The Superprogrammer Image There is no knowledge of management as a development tool. (page 25)

Using Models to Change Thinking Patterns When the thinking changes, the organization changes, and vice versa. (page 35)

The Formular for a System's Behavior Behavior depends on both state and input. (page 59)

The First Law of Bad Management When something isn't working, do more of it. (page 62)

Brooks's Model (Rephrased) Lack of calendar time has forced more failing software projects to face the reality of their failure than all other reasons combined. (page 74)

Why Software Projects Go Wrong More software projects have gone awry for lack of quality, which is part of many destructive dynamics, than for all other causes combined. (page 76)

The Scaling Fallacy Large systems are like small systems, just bigger. (page 77)

The Reversible Fallacy What is done can always be undone. (page 89)

The Causation Fallacy Every effect has a cause... and we can tell which is which. (page 90)

Decisions by People Whenever there's a human decision point in the system, it's not the event that determines the next event, but someone's reaction to that event. (page 111)

The Square Law of Computation Unless some simplification can be made, the amount of computation to solve a set of equations increases at least as fast as the square of the number of equations. (page 130)

The Natural Software Dynamic Human brain capacity is more or less fixed, but software complexity grows at least as fast as the square of the size of the program. (page 135)

The Size/Complexity Dynamic Ambitious requirements can easily outstrip even the brightest developer's mental capacity. (page 144)

The Log-Log Law Any set of data points forms a straight line if plotted on log-log paper. (page 146)

The Helpful Model No matter how it looks, everyone is trying to be helpful. (page 154)

The Principle of Addition The best way to reduce ineffective behavior is by adding more effective behavior. (page 155)

An Additional Model The way people behave is not based on reality, but on their models of reality. (page 156)

The First Principle of Programming The best way to deal with errors is not to make them in the first place. (page 184)

The Absence of Error Fallacy Thought copious error guarantees worthlessness, having zero errors guarantees nothing at all about the value of software. (page 185)

The Controller Dilemma The controller of a well-regulated system may not seem to be working hard. (page 197)

The Controller Fallacy If the controller isn't busy, it's not doing a good job. If the controller is very busy, it must be a good controller. (page 197)

The Difference Detection Dynamic First, the smallest amount of the test time is spent on a few easy problems; and second, most of the easy problems are found early in the test cycle. (page 202)

The Failure Detection Curve (the Bad News) There is no testing technology that detects failures in a linear manner. (page 205)

The Failure Detection Curve (the Good News) Combining different detection technologies creates an improved technology. (page 205)

The Army Principle There are no bad soldiers; there are only bad officers. (page 212)

The Army Principle (Modified) There are no bad programmers; there are only bad managers who don't understand the dynamics of failure. (page 213)

The Self-Invalidating Model The belief that a change will be easy to do correctly makes it less likely that the change will be done correctly. (page 236)

The Ripple Effect This effect involved the number of separate code areas that have to be changed to effect a single fault resolution. (page 237)

The Modular Dynamic The more modular you make the system, the fewer side effects you need to consider. (page 238)

The Titanic Effect The thought that disaster is impossible often leads to an unthinkable disaster. (page 241)

The Pressure/Judgment Dynamic Pressure leads to conformity leads to misestimating leads to lack of control leads to more pressure. (page 255)

The Law of Diminishing Response The more pressure you add, the less you get for it. (page 262)

Weinberg's Zeroth Law of Software If the software doesn't have to work, you can always meet any other requirement. (page 275)

Managers Not Available Busy managers mean bad management. (page 276)

No Time to Do It Right Why is it we never have time to do it right but always have time to do it over? (page 278)

The Boomerang Effect Attempts to shortcut quality always make the problems worse. (page 279)

2 Volume 2 – First-Order Measurement

The Observation Lesson, Part 1 Every situation offers possibilities for observation, but you must choose which possibilities to accept. (page 28)

DeMarco's Principle Effort moves toward whatever is measured. (page 29)

The Rat Hair Rule What you observe may be indirect, and may not even be explained rationally, but the observation is okay if it leads to the right control action. (page 32)

The Brain-Eye Law To a certain extent, observational power can compensate for mental weakness. (page 35)

The Eye-Brain Law To a certain extent, mental power can compensate for observational weakness. (page 35)

The Rationalization Principle You can design a measurement system for any conclusion you wish to draw. (page 35)

The Cybernetic Rule Only when you know the state of the system can you make sensible interventions to control it's behavior. (page 31) Or, Without visibility, control is not possible. (If you can't see, you can't steer.) (page 48)

The Swedish Army Dictum When the map and the territory don't agree, always believe the territory. (page 57)

Counting Lines of Code As a Metric Any technique that visualizes something meaningless leads to distortion. (page 58)

The Information Rule No matter what else it is, everything is information. (page 58)

The Physician Principle Every process is created by people, and thus can be changed by people. (page 59)

The 80-20 Rule Roughly 80 percent of the variance is contributed by 20 percent of the cases, in many real-world situations. (page 64)

The Rule of Three Interpretations If I can't think of at least three different interpretations of what I received, I haven't thought enough about what it might mean. (page 90)

The Military Measurement Rule Measure it with a micrometer, mark it with a piece of chalk, and cut it with an ax. (page 101)

Definition of Quality Quality is whatever I like. (page 108)

The Quality Problem Every software problem is a quality problem. (page 111)

The Zeroth Law of Software If the software doesn't have to work, you can always meet any other requirement. (page 111)

- The Zeroth Law of Quality** If you don't care about quality, you can meet any other requirement. (page 112)
- The Second Law of Thermodynamics** To decrease entropy (increase information), you need to add energy. Or, There's no such thing as a free lunch. (page 121)
- The First Law of Human Nature** People never want to believe the Second Law of Thermodynamics applies to them. (page 121)
- The First Rule of Failure Prevention** Nothing is too small to be worth observing. (page 158)
- The First Principle of Financial Management** A loss of X dollars is always the responsibility of an executive whose financial responsibility exceeds X dollars. (page 159)
- The Second Rule of Failure Prevention** A loss of X dollars is always the responsibility of an executive whose financial responsibility exceeds X dollars. (page 159)
- The Army Principle** There are no bad soldiers; there are only bad officers. (page 165)
- The Universal Pattern of Huge Losses** A quick, "trivial" change is made to an operational system, without any of the usual software engineering safeguards. The change is put directly into the normal operations. A small failure is multiplied by many uses, producing a large consequence. (page 166)
- The Observation Lesson, Part 2** Survival responses lead to imprecision in thought; imprecision in thought destroys quality. (page 209)
- Satir's Dictum** The problem is nothing. The coping with the problem is everything. (page 211)
- The Acting Crazy Principle** When somebody is acting crazy, go to the empathic position and find a logical reason for the craziness. (page 230)
- The Zeroth Law of Software Engineering** If you don't care about quality, you can achieve any other objective. (page 255)
- The Measure of Bureaucracy** The percentage of what you're doing that you don't understand why you're doing it. (page 269)
- The Candidate Product Rule** Actually, it ain't nothin' 'til it's reviewed. (page 290)
- The Zeroth Law of Unreliability** If a system doesn't have to be reliable, it can meet any other objective. (page 295)
- The System Behavior Rule** Behavior depends on both state and input. (page 321)

3 Volume 3 – Congruent Action

Brooks's Aphorism Modified There is no silver bullet, but sometimes there is a Lone Ranger. (page 1)

Ashby's Law of Requisite Variety The action taken by the controller must be congruent with the situation, in that there is at least one controller action to deal with each possible system action. (page 6)

The High-Quality Software Delivery Principle Management is the number one random process element. (page 8) The number one random process element stands in the way of improving all the other random process elements. (page 9)

The Personal Effectiveness Principle If you cannot manage yourself, you have no business managing others. (page 9)

The One-Dimensional Selection Model of Management A common management model that is based on three faulty assumptions: Managers are born, not made; people can be ranked on a one-dimensional scale; and the scale for programming is the same as the scale for management. (page 15)

The Mana Model Certain people have big magic (mana) and certain people don't. (page 16)

The Principle of Choosing to Be a Manager If you don't really want the job of manager in the first place, everything you do as manager will be incongruent. (page 20)

The Blaming Style of Coping "I am everything; you are nothing." (page 28)

The Placating Style of Coping "I am nothing; you are everything." (page 29)

The Superreasonable Style of Coping "It is everything; you and I are nothing." (page 33)

The Loving/Hating Style of Coping "It is nothing; you and I are everything." (page 35)

The Irrelevant Style of Coping "Nothing counts for anything." (page 36)

The Houdini Approach to Management You mystify them with complicated formulas and transformations, so they don't see what you're really doing. (page 53)

The Rip van Winkle Approach to Management You wake up after two years and demand to know, "Why is this project two years late?" (page 53)

Brooks's Law Adding manpower to a late software project makes it later. (page 78)

Brooks's Law Refined Assigning new people late in a software project to the tasks other people are already trying to do makes the project later. (page 78)

The Congruent Manager Model Leadership is the ability to create an environment in which everyone is empowered to contribute creatively to solving the problems. (page 80)

The Controller Fallacy If the controller isn't busy, it's not doing a good job; and if the controller is very busy, it must be a good controller. (page 128)

The Principle of Addiction To prevent an addiction (X), prohibit X; provide an alternative solution (Z) that really works; and soften the short-term pain if necessary, but not with X. (page 161)

Deming's Point Eight, "Drive Out Fear" Most people on a job, especially people in management positions, do not understand what the job is, nor what is right or wrong. Moreover, it is not clear to them how to find out. (page 167-168)

The Basic Principle of American Capitalism To prohibit placating, give customers alternative sources of services. (page 172)

The Congruent Commenting Standard If there is one standard, there must be at least two standards. (page 195)

The Helpful Model No matter how it looks, everyone is trying to be helpful. (page 208)

The Paranoid Model Things are going wrong because somebody is trying to hurt me. (page 209)

The Stupid Model Never attribute to maliciousness that which can otherwise be attributed to stupidity. (page 211)

The Giver's Fact No matter what it appears to be, feedback information is almost totally about the giver, not the receiver. (page 220)

The Big Secret All people like to have their work appreciated. (page 224)

Wain's Fifth Conclusion "Nothing motivates a man more than to see his boss put in an honest day's work." (page 224)

The Team Formation Principle Everyone on the team has *some* unique contribution. (page 265)

4 Volume 4 – Anticipating Change

The Newtonian Model Principles The bigger the system you want to change, the harder you must push. The faster the change you want, the harder you must push. To change in a certain direction, you must push in that direction. Push works both ways. (page 10)

The Newtonian Model Lessons, Applied to People Management When you push in one direction, people may move in the opposite direction. When you push harder, people may move less easily. When you push in one direction, people may move in a totally unexpected direction. When you push less, people may move more easily. When you push too fast, they may shatter – like glass when it is struck, rather than pushed. (page 12)

Satir on Resistance to a Foreign Element Familiarity is always more powerful than comfort. (page 22)

Rule of Decision Making in the Chaos Stage Chaos is definitely not the time to make long-term decisions. (page 24)

Copeland's Law of Discontinuity A discontinuity is an opportunity to stop doing old things and start doing new things. (page 66)

Cannon's Principle of Structure and Function "structure and function are inseparably related." (page 71)

Simple Rules for Change Artistry Management Don't blame. Give and receive information. Don't placate. Take no job that you don't believe in. Cut out the superreasonable slogans and exhortations. No tricks. Means *are* ends. Trust, and merit trust. Never stop training yourself in change skills. Never stop seeking improvements right around you. Remember that you were born little, just like everybody else. Just because you have a title, you haven't ceased to be a human being. Be an example of what you want others to be. (page 81)

The Affirmation Challenge for Becoming a Change Artist Each and every day, give on affirmation to one person. (page 99)

Deming's Fifth Deadly Disease "running a company on visible figures alone." (page 117)

The Law of Limiting Factors When a number of conditions are necessary to a process, its rate is controlled by the least favorable of these conditions. (page 120)

Minot's Law, Extended to Organizational Growth Management's efforts to raise quality by successful organization may succeed for a while, but may also produce a more complex organization, which becomes harder to organize for additional improvements. Thus, the current changes eventually become growth-rate-limiting structures for future changes. (page 132)

The Theory of Critical Periods of Development Early small decisions about the organization may have an enormous impact on the ultimate success of the organization. (page 133)

The Principle of Similitude, Applied to the Growing Organization As the organization grows, its relationship with the outside is strained as it tried to maintain its internal viability. (page 135)

The PLASTIC Model Plan to the Level of Acceptable Stable Talent In Completing Projects. (page 165)

The MOI Model Evaluate the Motivation, Organization, and Information resources required for implementation of a plan. (page 167)

The Swiss-Style Rule of Thumb for Assigning Control Responsibilities a) Push every decision down to the lowest level that has the information and tools to make that decision; b) Push all tools and information to the lowest level that will take them. (page 186)

The \$50 Million Rule of Management Error If it's a \$50 million error, then the management at the \$50 million level must be responsible. (page 189)

The Stability Principle Every part of a process must be a controlled system. (page 216)

The Visibility Principle Everything in the project must be visible at all times. (page 219)

The Reality Principle Nothing is real until it has passed independent review. (page 220)

The Measurability Principle Anything you don't measure will be out of your control. (page 221)

The Zeroth Law of Software Engineering If you don't have to meet requirements, management is no problem. (page 223)

The Product Principle Products may be programs, but programs are not products. (page 224)

The Culture/Process Principle Whatever you can safely assume in the culture, you don't have to specify in your process description. (page 230)

Process Improvement Lessons Process improvement must involve all levels of the organization. (page 252) Individual issues often underlie the toughest improvement situations. Cultural changes will involve upper managers. You can change the logical process first, but consider this a test to see if the problem is entirely logical. To address emotional problems, you'll need to get under the surface to the layers of information protected by the cultural rules governing what's not okay to talk about. Be careful that changes are not made in a blaming way. A policy of not blaming does not mean a policy of placating. (page 253)

The Zeroth Law of Quality If you don't care about quality, you can meet any other requirement. (page 264)

The Zeroth Law of Software If the software doesn't have to work, you can meet any other requirement. (page 264)

The Zeroth Law Corollary The less closely you have to meet requirements and the more your requirements approximate the Assumption of Fixed Requirements, the easier it will be for you to manage. (page 264)

The Assumption of Fixed Requirements The assumption that developers should have unchanging requirements before starting any project. (page 262)

The Right Product Principle Anything not worth doing is not worth doing right. (page 272)

Jones's Law Those parts of the product that don't go through the entire development process will cause 90 percent of your problems. (page 302)

Brook's Law Adding X percent to the staff will not generally speed the schedule by X percent. (page 302)

The Square Law of Computation Adding X percent to the schedule will not accommodate X percent increase in functionality. (page 302)

The Basic Principles of Cybernetic Management Act early, act small, using more-or-less continuous feedback. (page 307)

The Software Rule of Two When estimating your labor for a software job, make your best guess and multiply it by two. (page 310)

The Waterfall Model Rule Always use the Waterfall Model *when it applies*. (page 316)

The Loop Rule for Process Descriptions Never use loops in a process description. (page 343)

The Loop Rule for Plans Never use loops in a plan. (page 344)

Augustine's Ratio of Test Failures "... the incidence of test failures is directly proportional to the square of the size of the crowd multiplied by the rank of the senior observing official." (page 346)

The Helpful Model Most of the time, in spite of appearances, everybody is trying to be helpful. (page 368)

Mills's Principle of Design The goal is not to prove programs correct; the goal is to write correct programs. (page 395)

Augustine's Law of Insatiable Appetite The last ten percent of performance generates one-third of the cost and two-thirds of the problems. (page 399)

The Rule of Three If you can't think of three circumstances in which the design might fail, you haven't thought enough about it. (page 402)

The Paradox Rule If there is no paradox that the design must resolve, you don't understand the problem. (page 402)

The First Law of Technology Transfer Long-range good tends to be sacrificed to short-range good. (page 414)

The Second Law of Technology Transfer Short-range feasibility tends to be sacrificed to long-range perfection. (page 414)

The Ten Commandments of Technology Transfer Thou shalt have a plan to lead thee out of the wilderness. Thou shalt not worship thy plan. Thou shalt ask for no person in vain. Thou shalt not work seven days a week. Thou shalt honor thy users and listen to them. Thou shalt not kill support for change. Thou shalt not adulterate the work. Thou shalt not steal resources from the work. Thou shalt not bear false witness against thy plan. Thou shalt not covet thy neighbor's optimal technology. (page 418)

The Principle of Attraction People will volunteer their services when they see people doing something that seems worthwhile. (page 419)

The Eleventh Commandment of Technology Transfer In times of trial, always remember the Helpful Model. (page 423)

The System Behavior Principle Behavior depends on both state and input. (page 445)