



SOFTWARE
PRODUCTIVITY
RESEARCH

Ensuring High-Quality Software Metrics

“Truth is truth to the end of reckoning.”

William Shakespeare

Measure for Measure, Act V, Scene i



Introduction

In 1604, William Shakespeare wrote *Measure for Measure*, his penetrating meditation on the relationship between justice and mercy. It's a complicated tale involving:

- pompous authority figures of unbending moral certainty,
- delegation of power to the ill-prepared,
- concealed and mistaken identities,
- abuse of the innocent,
- virtue corrupted,
- reputations ruined,
- and, of course, the ever-popular lust and greed.



Introduction

Now, we all know Shakespeare was a genius, but how could he so perfectly describe the modern software development environment ***four hundred years in advance?***

Well, just as Shakespeare believed that there must be a balanced, measured link between justice and mercy, we know that our IT endeavors must also seek balance among numerous observable, measurable, and often conflicting motives and influences.



Introduction

Software metrics are critical to:

- good project estimation,
- project management,
- product implementation,
- quality assurance,
- customer care,
- service level agreements,
- balanced scorecards.



Introduction

The focus today will be on:

- identifying the software metrics that Best in Class organizations collect,
- how even well-intentioned metrics programs can become dysfunctional and incent unanticipated behaviors,
- a straightforward technique for ensuring that your organization both collects and *uses* high-quality software metrics,
- And perhaps a few more references to Shakespeare!



Introduction

The agenda for this program includes the following major elements:

- The case for software measurement
- An overview of the metrics most commonly collected and analyzed by industry-leading companies
- An introduction of the evil-twin concepts of “Measurement Noise” and “Dysfunctional Metrics”
- An example from real life of how good metrics go bad (kind of like an episode of “Cops” where shirtless metrics hide from the authorities under an old pickup truck)
- And the G/Q/M technique for ensuring high-quality metrics—metrics you would be proud to take home to meet your parents.





The Case for Software Measurement



S P R

Ensuring High-Quality Software Metrics



September 2004

*“The miserable have no other medicine,
but only hope.”*

William Shakespeare
Measure for Measure, Act III, Scene i



The Case for Software Measurement

- Given the unhappy combination of measurement ambivalence and performance anxiety, software metrics programs are frequently—very frequently—torpedoed by the people and organizations that stand to gain the most from them.
- Software development has become more, rather than less, people-intensive over time—this in spite of the perceived advance of “automated” tools and techniques. People are hard to measure, and they don’t like it when they are.
- Breaking through the instinctive resistance to thoughtful measurement and enlightened, analytical management has become the Holy Grail of software metrics proponents.



The Case for Software Measurement

- Best in class software companies know their quality and productivity levels because they measure them.
- They can identify, isolate, and invest in problem areas.
- Industry stragglers don't measure and therefore have no idea how good or bad they are, making it difficult to change and far less likely that they will improve.



The Case for Software Measurement

SPR's founder, Capers Jones, is fond of asking his CEO clients three very basic questions:

- Is your company's software quality better than your competitors'?
- Is your company's software productivity better than your competitors'? And...
- Is your company's time to market better than your competitors'?

Jones likes to say, measurement is a “root technology” that is both a cause and an effect of excellence in the other factors.





The Metrics Used by Best in Class Companies



S P R

Ensuring High-Quality Software Metrics



September 2004

*“Virtue is bold, and goodness never
fearful.”*

William Shakespeare
Measure for Measure, Act III, Scene i



The Metrics Used By B.I.C. Companies

The following metrics are closely associated with industry-leading companies, and they provide an ambitious target for all others to pursue:

- Some are called “primitive metrics”, because they are basic, stand-alone measures that are not expressed as a ratio of two different metrics.
- Alternatively, metrics may be called “relative” or “derived”, when they represent the relationship of two or more metrics.



The Metrics Used By B.I.C. Companies

Best in Class companies generally capture primitive and derived metrics in four broad categories:

- Quality Measures
- Productivity and Schedule Measures
- Assessment Measures, and
- Business and Corporate Measures



The Metrics Used By B.I.C. Companies

Prominent Quality measures include:

- Customer satisfaction
- Defect Quantities
- Defect Removal
- Delivered Defects
- Defect Severities
- Software Reliability and Availability
- Service Time Response
- Complexity
- Test Coverage
- Cost of Quality



The Metrics Used By B.I.C. Companies

Prominent Productivity and Schedule measures commonly collected by Best in Class organizations include:

- Size Measures
- Activity-based Schedule Measures
- Activity-based Cost Measures
- Indirect Cost Measures
- Monthly Milestones



The Metrics Used By B.I.C. Companies

Regardless of which kind of assessment methodology is employed, Best in Class companies are accounting for the following capabilities and constraints:

- Software Processes
- Software Tools
- Software Organization Infrastructure
- Software Skills
- Staff and Management Training, and
- The Physical Work Environment




The Metrics Used By B.I.C. Companies

The fourth category of measurement that distinguishes Best in Class companies comprises a broader set of high-level intelligence data. These are a few measures that can be tied back to the detailed metrics previously mentioned:

- Portfolio Measures
- Salary and Benefit Measures
- Market Share Measures
- Competitive Measures





The Evil Twins “Measurement Noise” & “Dysfunctional Metrics”



“The cunning livery of hell.”

William Shakespeare
Measure for Measure, Act III, Scene i



Measurement Noise & Dysfunctional Metrics

- They are “evil twins”, because they are often equally subtle and similarly destructive to the objective of measuring and using meaningful metrics—and as “twins”, they frequently occur at the same time.
- Moreover, they can occur without any conscious intent and despite reasonable diligence.



Measurement Noise & Dysfunctional Metrics

What is “Measurement Noise”?

- Imprecise, careless, or mismatched measurement techniques introduce doubt, suspicion, and error into otherwise sensible metrics studies, and this uncertainty is a prime example of “measurement noise”.
- At SPR, we also speak of this as the “fuzz” factor around software metrics.
- The fuzzier, or the noisier, the basic measurements are, the more doubtful any analysis or decisions based upon them will be.
- The chief cause of measurement noise is the unfortunate tendency to encourage “quick and dirty” metrics collection out of fear that measurement will cost too much.



Measurement Noise & Dysfunctional Metrics

What are “dysfunctional metrics”?

- A dysfunctional metric is one that incents or otherwise enables unintended and often destructive behavior.
- This is where good metrics go bad, sometimes bringing on behavior in IT organizations that is the exact opposite of that which was intended.





A Classic Case of Dysfunctional Metrics



Lucio: “If you will hang me for it, you may; but I had rather it would please you, I might be whipped”

Duke: “Whipped first, sir, and hanged after.”

William Shakespeare
Measure for Measure, Act V, Scene i



Famously Dysfunctional Metrics

- In 1963, Peter Blau of the University of Chicago published a study of government employment offices.
- A government employment office had a noble mission statement: “To serve workers seeking employment and employers seeking workers”.
- To achieve this lofty goal, interviewers conducted interviews with prospective workers, and if they felt there was a match with a particular employer, they referred the worker to the employer.
- Initially, employment interviewers who worked in the employment office were evaluated by the number of interviews they conducted.



Famously Dysfunctional Metrics

- It was quickly apparent to the interviewers that their interests were served by doing everything they could to maximize the number of interviews per day.
- The employment office certainly got more interviews under this arrangement, but they soon noticed that there were fewer and fewer referrals to employers.
- The quality of interviews deteriorated as well, and job placements continued to fall, as interviewers motored through as many interviews as possible.
- They replaced the single metric with eight separate indicators, which they believed would present a more balanced picture of office activity.



Famously Dysfunctional Metrics

- Miraculously, improvement in the number of interviews *and* in the number of referrals followed!
- What looked like improvement turned out to be more bad, unintended behavior.
- Confronted with more complicated metrics, the interviewers developed more subversive, more sophisticated dysfunctional behaviors, including falsifying records and destroying interview slips that failed to result in referrals.
- Stan Rifkin has described this phenomenon as “penalizing the truth”.





The G/Q/M Paradigm



S P R

Ensuring High-Quality Software Metrics



September 2004

*“The hand that hath made you fair hath
made you good.”*

William Shakespeare
Measure for Measure, Act III, Scene i



The Goal/Question/Metric Paradigm

The “G/Q/M” method was developed by Victor Basili and David Weiss and first published in 1984 in the IEEE journal, “Transactions on Software Engineering”.

It is based on two assumptions:

- Measurement programs should be “goal-based” not “metrics-based”, and
- The way both goals and metrics are defined should be customized to an individual organization’s needs.



The Goal/Question/Metric Paradigm

- By “Goal”, Basili and Weiss meant a specific, measurable business goal for an organization.
- By “Question”, they suggest that whether and how well a goal has been attained must be knowable by answering one or more key questions that can be defined ahead of time.
- By “Metric”, they include only quantitative entities that answer a specific question--and the question must address directly a stated goal or at least part of a goal.



The Goal/Question/Metric Paradigm

In execution, there are four phases to the GQM method:

- The planning phase
- The definition phase
- The data collection phase
- The interpretation phase



The Goal/Question/Metric Paradigm

- Sir Francis Bacon—who incidentally is one of the men some believe actually wrote the plays attributed to Shakespeare—Bacon said this, “A prudent question is one-half of wisdom.”
- Another, more recent philosopher—Captain Kangaroo—liked to say that knowing that you don’t know something is the beginning of knowing it.
- A good metric must provide the answer to a question that in turn helps to determine if a goal has been met.



The Goal/Question/Metric Paradigm

- The GQM Method imposes discipline on a measurement program.
- It ensures that the objectives and deliverables of the program are directly related to and illustrative of goals that the leaders of the business have articulated themselves.
- Time is not wasted capturing metrics simply because they were listed in a book or because a friendly metrics consultant gave a charming presentation about them.
- GQM is a proven technique that fights the dysfunction and ensures high-quality software metrics.

