The Nine-Step Metrics Program

Timothy K. Perkins
Software Technology Support Center

Metrics are essential in evaluating program or project performance. However, several organizations remain confused regarding what measurements to collect, and how to use the measurements after they are collected. Before addressing these questions, some terms need to be defined. For the purpose of this article, a metric is defined as a combination of two or more measurements. Measurements are the raw data gathered from comparing an entity to a standard. After reading the article, feel free to use your own definition of the above terms.

While evaluating and commenting on the Measurement and Analysis Process Area of the Software Engineering Institute's (SEI) Capability Maturity Model Integration (CMMI[SM]) version 0.2, members of the Software Technology Support Center at Hill Air Force Base, Utah, developed a nine-step measurement process with the steps logically grouped by activity type. The three activity groups are measurement planning, measurement implementation, and measurement program evaluation. Following is a presentation of these steps.

Activity Group 1

Measurement Planning

There are four measurement planning activities, the results of which are documented in the measurement plan. The following are the four activities:

1. Define Information Needs.

All measurements should adhere to the following criteria:
- Criterion 1 – Measurements should induce the parts to do what is good for the system as a whole.
- Criterion 2 – Measurements should direct managers to the point that needs their attention [1].

These criteria support the goal–question–metric (GQM) paradigm developed by Victor Basili. The key concepts of this paradigm are:
- Processes (software development, program management, acquisition management, etc.) have associated goals.
- Each goal leads to one or more questions regarding the accomplishment of the goal.
- Each question leads to one or more metrics that will answer the question.
- Each metric requires two or more measurements needed to create the metric.
- Measurements should be selected that provide the data needed to create the metrics necessary to answer the questions that determine goal accomplishment.

Eliyahu Goldratt supports the GQM paradigm in his statement "Measurements are a direct result of the chosen goal. There is no way that we can select a set of measurements before the goal is defined [2]."

Another point to remember, according to Joseph Juran, is that different organizational levels require different metrics. At the worker level, measurements are usually taken in terms of deeds performed or in things produced, e.g., how many, how much, or physical things (time, mass, space). Top level managers usually speak in terms of dollars—the impact on the bottom line. Those in the middle must be capable of communicating using both frames of reference. For example, the company financial statement is in the language of dollars. The sales forecasts and the results are in dollars and units. The production schedules, order points, and material requisitions are all in units [3].

2. Define Metrics and Analysis Methods.

This step is a continuation of the GQM paradigm described above, with the additional task of defining the analysis methods that will be used to create information from the data collected. The topic of proper data analysis is not trivial, and is well beyond what can be covered in a short article. The handbook [4] is an excellent starting point regarding measurement in general and includes several chapters that discuss analyzing data. Best of all, it can be downloaded free from the SEI Web site [www.sei.cmu.edu].

3. Define the Selected Measures.

This is the final step of the GQM paradigm. The selected measures are chosen not only to provide the information needed to answer the questions, but also to allow analysis using the methods determined in Step 2 above. Measurements used to characterize process performance should:
- Relate closely to the issue under study.
- Have high information content.
- Pass a reality test.
- Permit easy and economical collection of data.
- Permit consistently collected, well-defined data.
- Show measurable variation.
- Have diagnostic value as a set [4].

Let's look at each of the above points in some depth.

Relate closely to the issue under study. As mentioned in the paragraph discussing GQM, measurements must enable us to answer the questions related to individual goals.

Have high information content. A single measurement that provides a significant amount of information is more valuable than a set of three, four, or more measurements required providing the same information.

Pass a reality test. Does the proposed measurement really provide information necessary to answer a question regarding a goal? Or is it just a *feel-good* measurement that has been collected traditionally, but offers no real value?

Permit easy and economical collection of data. This is one goal of a measurement program. Data that are readily available and answer the questions regarding a goal are more desirable than similar data that are difficult or expensive to gather. Do not hesitate to perform a cost benefit analysis regarding data that appear to be difficult or expensive to collect.

Permit consistently collected, well-defined data. Once again, repeatability of data is the reason for strict identification of collection points.

Show measurable variation. Data that does not exhibit variation is useless in determining how to improve a process. It is the range of the variation that determines whether or not a process is under statistical control and indicates whether or not process changes are achieving desired results.

As a set, have diagnostic value. As stated, measurements combine to form metrics that are used to answer questions regarding goals. The set of measurements selected must provide the information needed to determine goal accomplishment, otherwise the measurement set is insufficient. According to Florac et al., "They should be able to help you identify not only that something unusual has happened, but what might be causing it [4]."

In the process of selecting measurements, do not forget to spend some time determining how collected data will be analyzed. Some analysis techniques require a certain volume of data collected at regular frequencies with a minimum level of accuracy in order to provide meaningful results. Make sure you understand how the data will be analyzed and plan accordingly.

4. Define the Collection Process of Measurement Data.
The points in the process where measurements are to be collected should be identified. Are measurements to be taken before or after a certain procedure has been performed, prior to or subsequent to certain integration efforts, etc.? Additionally, the manner whereby data is to be collected and the individual responsible for collecting the data should be specified by job title. If at all possible, the data generation should be a normal part of or step in the process.

Measurements must be clearly defined. This definition should explicitly state what is included in and excluded from the measurement. This allows those who use the data to thoroughly understand what the data represent, to permit the repetition of data collection, and to compare data samples. A good example of the necessity of clearly defining measurements is to ask a group of individuals to determine the number of lines of code in a short program listing. Depending on the language, arguments can be made regarding control code, comment lines, multiple executable statements on a single line, etc.

The frequency of data collection also needs to be specified. Measurements should be taken frequently enough to identify problems, and allow their correction prior to generating substantial scrap, creating substantial rework, or missing critical milestones. For example, if an organization cannot afford to lose the month-long effort of five individuals working on a project prior to identifying a problem in product production, measurements frequency must be substantially more than monthly. In determining the collection frequency, do not forget to include the time required to process the data into measurements and metrics, and to get the metrics into decision-makers' hands.

The Nine Steps in Action – An Example
The following example of how the steps are used in a measurement program is based on the idea that an organization has been tasked to deliver a new software release within 120 days.

See [5] for an example of how the desired metric is calculated.
Step 1: Define information needs. Suppose one of the goals of the organization was to deliver the product on time. Questions regarding this goal are:
- Does the schedule estimate allow sufficient time to produce the product, or is the schedule artificially constrained?
- How much time is allocated for product development?
- Is there sufficient staff to provide the estimated needed hours within the required schedule?
- How much work has been accomplished on the critical path?
- How much work should have been accomplished on the critical path?
- How much time is remaining?

Step 2: Define metrics and analysis methods to address information needs. For the sake of this example, let's look at the third bullet from Step 1: Is there sufficient staff to provide the estimated needed hours within the required schedule? The metric used may be staff hours available per day. The analysis method chosen could be the use of X-Bar and R charts to determine if the number of hours delivered per day is within statistical control. The term "staff hours available per day" should be explicitly defined so that everyone on the project understands what is meant by an available staff hour.

Step 3: Define the selected measures. The measure to be collected would be the number of productive hours per person assigned to the project per day. For example, time spent in a team meeting discussing the project may be included while time spent answering e-mail on an unrelated project may not be included.

Step 4: Define the collection process of the measurement data. The collection process is to record the time reported against the project per person per day.

Activity Group 2
Measurement Implementation
The next activity group is measurement implementation. Continue with the following procedures:

5. Collect the Measurement Data.
This activity is simply the execution of the measurement data collection process methods.

6. Analyze the Measurement Data to Derive Metrics.
Metrics are derived from the analysis performed on the measurement data. The quality of the metric is tied to the rigor of the analysis process and the quality of the data collected.

7. Manage the Measurement Data and Metrics.
The measurement data and metrics must be managed properly according to the requirements defined in the measurement plan.

8. Report the Metrics.
Once the metrics are derived from the analysis of the measurements, they are made available to all those either affected by the metrics or by the decisions made because of the metrics.

Continuing the Example
Step 5: Collect the measurement data. The hours per day accomplished on the project per person is collected from the time cards the workers complete daily.

Step 6: Analyze the measurement data to derive metrics. Daily measurements are used to calculate additional points on X-Bar

and R charts, which are then analyzed to determine if the process is in statistical control and to determine if the average number of hours delivered are equal to or greater than the average number expected. Remember that the process can be within statistical control without meeting the average number of hours needed.

Step 7: Manage the measurement data and metrics. Archive the data in a manner that it can be readily retrieved, if needed.

Step 8: Report the metrics. Report whether or not the process is in statistical control and whether or not the necessary number of hours is being delivered.

Activity Group 3

Measurement Program Evaluation

9. Review the usability of the selected metrics.

Initially, the selection of metrics, analysis methods, and specific measurement data may be a *best guess*. Whether or not they meet specified information needs must be determined by experience. Over time, through a review of the usefulness of the metrics, the selection can be refined to a high correlation between the metrics selected and the information needs. This will be an iterative process.

The old adage "keep it simple" is a good rule to follow when establishing a metrics program. Remember to focus the metrics on the organization's goals. Because an organization will have a limited number of goals, there should be a limited number of necessary metrics. In other words, do not go overboard in collecting all potential measures. Collect only those necessary to determine goal achievement.

In a study performed in the early 1990s, Rifkin and Cox sampled organizations that had excellent software measurement practices. Their finding was that none of the organizations sampled had more than a dozen metrics [5]. Start small, collect and evaluate the data, and make changes as necessary. It is better to implement the 70 percent solution and evolve to the 100 percent solution rather than allow the analysis paralysis of trying to hit 100 percent on the first try to keep the organization from implementing anything.

Continuing the Example

Step 9: Review the usability of the selected metrics. After analyzing and reporting on staff hours per day, you may realize those hours alone are not a sufficient metric. Maybe productivity should be included, e.g., how much work on the project is being accomplished for each hour delivered. This would involve measuring task accomplishment per hour delivered.

A Caution in Selecting Metrics

Goldratt offers the following caution regarding measurements, "Tell me how you measure me, and I will tell you how I will behave. If you measure me in an illogical way… do not complain about illogical behavior [6]." This implies that the measurements you take may cause individuals within an organization, or an organization as a whole, to behave in a given manner.

To illustrate, Goldratt uses the example of a steel mill that was losing money where the primary performance measurement was tons per hour. At the end of the month, when the monthly measurement of tons per hour was coming due, the organiza-

tion concentrated on producing *tons* of steel without regard to customer orders. In the rolling operation, thick steel plate took less time to produce than thin steel plate. Can you guess the thickness of plate produced? Customer orders went unfulfilled, and customers complained, but to no avail. Only when the measurement was changed from tons per hour to orders satisfied did the steel mill begin to use black ink rather than red ink [7]. Make sure the measurements you select cause the organization to behave in the desired manner.

Goldratt further warns, "Change my measurements to new ones that I don't fully comprehend and nobody knows how I will behave, not even me [8]." Make sure the organization's members understand the reason for the measurements and how they will be used before attempting to institute the collection of a set of new measurements.◆

References

1. Goldratt, Eliyahu M., *Critical Chain*, The North River Press Publishing Corp., Great Barrington Mass, 1997, pp. 81-82.
2. Goldratt, Eliyahu M., *The Haystack Syndrome*, North River Press, Croton-on-Hudson, N.Y., 1990, p. 14.
3. Juran, Joseph M., *Managerial Breakthorugh*, 30th Anniversary Edition, McGraw-Hill, New York N.Y., 1995, pp. 240-241.
4. Florac, William A., Park, Robert E., Carlton, Anita D., *Practical Software Measurement: Measuring for Process Management and Improvement* (CMU/SEI-97-HB-003), Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pa, 1997, p. 19
5. Rifkin, S. and Cox C., *Measurement in Practice* (CMU/SEI-91-TR-016), Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pa, 1991.
6. Goldratt, *The Haystack Syndrome*, p. 28.
7. Goldratt, *Critical Chain*, pp. 104-112.
8. Goldratt, *The Haystack Syndrome*, p. 88.

About the Author

Timothy K. Perkins has been involved in software process improvement for the past 11 years, since he led the effort to initiate the software process improvement effort at the (then) five Air Force Air Logistics Centers. As the Software Engineering Process Group leader at the Software Engineering Division at Hill Air Force Base, Utah, he led the division in reaching CMM Level 3. The division has gone on to achieve CMM Level 5. Since retiring from the Air Force, he has been employed by Science Applications International Corp. as a process improvement consultant currently under contract with the Software Technology Support Center, which provides consulting services to Air Force and other DoD and government agencies. Perkins holds a bachelor's degree in electrical engineering from Brigham Young University and a master's degree in business administration from the University of Phoenix.

OO-ALC/TISE
7278 4th Street
Hill AFB, Utah 84056
Voice: 801-775-5736
Fax: 801-777-8069
E-mail: tim.perkins@hill.af.mil