

Making Sense of Usability Metrics: Usability and Six Sigma

Jeff Sauro

Oracle, Inc.
Denver, Colorado USA
Jeff.Sauro@oracle.com

Erika Kindlund

Intuit, Inc.
Mountain View, California USA
Erika_Kindlund@intuit.com

ABSTRACT

This paper identifies the limitations of traditional usability metrics and presents a process to increase their meaning by adapting Six Sigma methods. We define how common usability metrics can be evaluated in terms of a standardized *defective rate* or *quality level* and explore the benefits of this data transformation.

INTRODUCTION

Traditional usability metrics are difficult to use. Current methods of analyzing and reporting the most common metrics do not effectively communicate a comprehensive assessment of usability.

The metrics that usability analysts commonly collect to measure usability across its multiple aspects of effectiveness, efficiency and satisfaction are generally measured on different scales. Examples of these metrics are: task completion rates, average time to task completion, average task error counts and average task satisfaction scores (see [1] and [6] for more examples). These metrics are typically reported individually on a task-by-task basis, with little context for interpreting the relationship among the metrics of a particular task or across a series of tasks. Analysts often rely on their experience or resort to “eyeballing” the various metrics to arrive at an intuited assessment of task or product usability. Conversely, stakeholders often aren’t able to find meaning across disparate metrics and will gravitate towards one, typically the task completion rate, and use that as the single dependent variable for “Product Usability” in tandem with other business metrics. In either approach, what results is a decidedly non-robust analysis of overall task or product usability.

Usability analysts need a method to effectively compare and derive meaning from traditional usability metrics and a way to communicate better with stakeholders. We propose a process that supports more effective analysis of usability data by standardizing traditional usability metrics on a uniform scale. The method is based on Six Sigma and retains the completeness of usability metrics while reducing the confusion.

Six Sigma is a methodology that promotes product or system quality. At its heart are statistical techniques used to quantitatively measure process defects that are defined by customers or users. While Six Sigma has been rigorously applied to manufacturing and organizational processes in many Fortune 500 companies, the state of Six Sigma in software development has almost exclusively focused on call center improvements and decreasing coding errors in software development processes.¹ To date, while very little has been done in the area of user interface analysis and usability testing with Six Sigma, there is interest in its application [12].

¹ See <http://software.isixsigma.com/> for more information on Six Sigma software deployments.

UPA Conference 2005

A primary benefit of Six Sigma is that it identifies meaningful goals for process measurements and provides a uniform scale on which the process can be measured against those goals. This uniform scale provides a context for meaningful comparison between the different aspects of a process. And perhaps more importantly, it provides a relative framework on which similar aspects can be compared across different processes. This framework is currently lacking in the traditional analysis of usability metrics.

Our process improves the analysis and communication of usability metrics so they can effectively drive decision-making processes within organizations. Metrics can be expressed in terms of a standardized "Quality Level" percentage that is derived from the commonly used Process Sigma metric—a measure of a process's capability.

DESCRIPTION OF METHOD

Before we detail our process, we will first briefly describe the major components of Six Sigma.

Introduction to Six Sigma

Six Sigma is statistical-based quality improvement methodology. Using statistics to improve business processes and products isn't new, and many of the methods used in Six Sigma have been perfected for the past several decades [15]. Six Sigma provides an emphasis on quality improvement that makes it a natural fit for usability: it makes the user an integral part of its methods. Specifically, customers or process users define what is an acceptable level of quality for any measure of a process. Acceptable levels of quality are identified as the "target", "goal" or "specification limit" and unacceptable levels of a measure are identified as "defects." The conditions under which a defect might occur are identified as "opportunities." The probability that a defect will occur in a sample, also referred to as the "defective rate," corresponds to a value called "process sigma."

It is the process sigma metric that provides Six Sigma with its moniker. Being 99% defect-free (roughly 3-sigma) isn't good enough for critical functions—99% effective means 200,000 wrong drug prescriptions occurring each year [5]. Advanced statistical analysis can push efficiencies closer to only 3.4 defects per million opportunities or 6-sigma.

Although many applications don't require a 6-sigma quality level to be considered best in class, the Six Sigma methodology provides a wealth of systematic approaches to an industry needing refinement. While most human factors engineers won't be testing airplane cockpits or nuclear power plants, many are involved with applications that manage financial, healthcare or other high-consequence activities. Any application that has significant consequences if human errors and inefficiencies aren't addressed can benefit from applying Six Sigma.

Using the Process Sigma Metric for Usability Data

Choosing the right metric to assess the quality of a process is crucial to driving improvements. Improperly chosen metrics can lead to sub-optimal behavior and lead people away from the right goals [15]. Choosing the right metric for assessing usability has had much discussion and we do not propose to end that discussion here. We instead show how to apply Six Sigma to usability through the most commonly used usability metrics.

The most commonly used usability metrics are task completion, time on task, error counts and satisfaction scores. Making these metrics more meaningful to stakeholders and providing easier interpretation to the usability analyst requires three steps:

1. A specification limit or maximum acceptable point needs to be identified for each measure based on user behavior or attitudes.
2. Conditions not meeting the user-defined goals will be defined as "defects." Conditions under which a defect might occur will be defined as "opportunities" for a defect.
3. Each measure will be converted into a standardized form allowing the stakeholders and analysts to know how far a metric is from the user-defined goal. The standardized form is the process sigma.

Process sigma describes how well a process meets its goals in terms of standardized sigma (σ) units. Sigma units can also be expressed as the *defective rate* or *quality level* of a measure.² For example, a process at a quality level 1 standard deviation above the specification limit would be said to be 2.5 sigma³ or to have an "85% quality level" or to be "15% defective" – all three express the same concept of quality.

Calculating Process Sigma

The process sigma for any measure, or its corresponding defective rate or quality level, is calculated with one of two methods depending upon the type of data that is collected - discrete or continuous. The four common usability metrics fall into both data types. Discrete data (errors and task completion rates) cannot be sub-divided into smaller meaningful units—you cannot have half an error. Continuous data (time and satisfaction) can be subdivided into infinitely smaller units that still have meaning.⁴

Method One: For Discrete Data

To calculate process sigma for a measure that is comprised of discrete data, one identifies all the defects in the sample (or measures that are not within the specification limits). The total defects are then divided by all the opportunities for defects (or total number of conditions under which a defect might occur) to arrive at the defective rate:

- Total Defects / Total Opportunities = defective rate (or probability of a defect)

The quality level is then calculated by subtracting the defective rate from 1:

- 1 - Defect Rate = Quality Level

² For more information on creating standardized metrics, z-scores and process sigma see "What's a Z-Score and Why Use it in Usability Testing?" <http://www.measuringusability.com/z.htm>

³ 2.5 = 1 plus a 1.5 sigma shift. See note 4 below for an explanation of the 1.5 sigma shift.

⁴ While the data may be theoretically infinitely subdivided, we are always limited by the sensitivity of the instruments in recording data.

UPA Conference 2005

The quality level corresponds to the area under the normal curve that represents the part of the process that is non-defective. A normal deviate (or z-score) for the quality level can be looked up on a standardized z-table.⁵ Process sigma is then calculated by adding a 1.5 "sigma shift" to the z-score. This sigma shift is added to reflect changes in a process over time.⁶

- Quality Level = Desired Area in the Normal Curve (z-score)
- Process Sigma = Z-score + 1.5 sigma shift

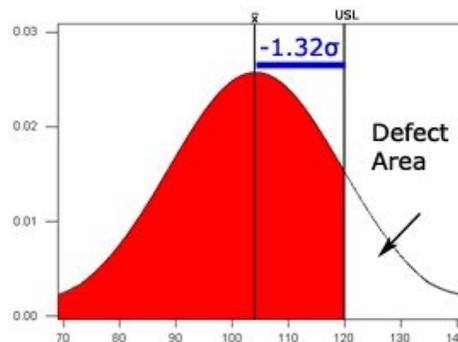
Calculating process sigma using Method 1 does not require any assessment of normality since the data is discrete.

Method 2: Continuous Data

To calculate process sigma for a measure that is comprised of continuous data requires the following computation:

- Z-score = (Sample Mean - Spec) / St. Dev
- Defect Rate = The area under the standard normal curve to the RIGHT of the Z-score (see figure 1 below)
- Quality Level = 1 - Defective rate, or the area under the standard normal curve to the LEFT of the Z-score
- Process sigma = Z-score + 1.5 sigma shift

Figure 1: A normal curve showing the defective area above the upper spec limit (USL) of 120 seconds. The mean time of the sample is 104 seconds, standard deviation of 12 seconds providing a z-score of $(110-104)/12 = -1.32\sigma$ and quality level of 91%.



Calculating process sigma using Method 2 does require that the data be roughly normally distributed. If the data are not normal, then Method 2 will be an inaccurate representation

⁵ Most statistics texts have standard normal (z) tables in an appendix and there are many online. In MS Excel, the formula `NORMSINV()` will also return a z-score from a percentage, and the formula `NORMSDIST` will return a percentage from a z-score.

⁶ The 1.5 sigma shift represents how a process may drift over time. When the actual shift isn't known, 1.5 is used. The calculation involves several lengthy and somewhat controversial calculations. For more information and debate see <http://www.isixsigma.com/library/content/c010701a.asp>. We currently do not include the shift in calculations that appear in human factors discussions but include it when they appear in six sigma discussions. It is always best to specify whether a shift was added when reporting process sigma.

of the true quality level. If the data cannot be transformed into a normal distribution, then it is recommended to use Method 1.⁷

A Process for Standardizing Usability Metrics

Having covered the two methods for converting raw data into standardized metrics, we will now define our process for using them to transform the four usability metrics.

Standardizing Task Completion

Task completion is the easiest usability metric to standardize. Task completion usually takes the form of binary data (complete, didn't complete), which is a special type of discrete data⁸. We can assume that all users want to successfully complete tasks, so a defect in task completion can be identified as an instance of a user failing a task. As such, every instance that participants in the sample attempted a task can be defined as an opportunity for a defect to occur. Therefore, if 16 out of 20 users completed a task then the defective rate is calculated with the discrete method as follows:

- Total defects / Total Opportunities = Defective Rate
- 4 task failures / 20 task attempts = 20% defective

The following steps then calculate the process sigma:

- $1 - .20 = .80 = 80\%$ Quality Level
- The corresponding z-score to 80% on a standardized z-table is .841
- $.841 + 1.5$ sigma shift = 2.34 sigma

Task completion for that task would then be described as having a quality level of 80% or 2.34 sigma.

Standardizing Error Rates

Since error counts are discrete data, we can apply the discrete calculation again. But calculating the process sigma is a bit more challenging because the discrete method requires us to identify the total number of opportunities there are for any user to make a single error. Unlike the calculation for task completion, it is insufficient to define "opportunities" as simply each instance of a user in the sample attempting a task. This is because not all tasks are equal when it comes to error potential. Complex tasks with many required components for task success have a greater potential for error than less complex tasks [14 esp. p154].

Our standardization process needs to account for this variation in error potential when trying to calculate the error probability. Therefore, we define the "total opportunities" under which an error might occur as the number of sub-tasks that a user must conform to in order to complete a task error-free. This method is similar to calculating the Human Error Probability (HEP) as described in [14]:

⁷ The inaccuracy will be most apparent for samples that are well above or below the spec limit (in the tails of the distribution). Data that are slightly non-normal and close (within 1 standard deviation) of the specification limit will not be as inaccurate. Displaying the data in a normal-probability plot instead of just a histogram will often prevent many false identifications of non-normality.

⁸ Some analysts might code task completion as categorical such as 1=complete, 2 = complete w/ assistance, 3 =ran out of time 4= failed. [Jurek Kirakowski Personal communication May 2004]

UPA Conference 2005

The general approach for [determining HEP] is to divide human behavior in a system into small behavioral units, find data for these subdivisions and then recombine them to estimate the error probabilities for the task.

Here is an example of how we define a task's opportunities in terms of its sub-tasks:

Example Task: Add a new customer record to the Customer List

- Opportunity 1: Locate access point for adding a new customer record and launch new customer record form
- Opportunity 2: Enter new customer record ID information
- Opportunity 3: Enter account opening balance information correctly
- Opportunity 4: Enter customer address information
- Opportunity 5: Enter customer contact information
- Opportunity 6: Submit record successfully

While there are 6 opportunities for the user to make errors, there can be multiple ways an error can be committed. It's important to note that identifying opportunities does not mean identifying ideal paths through the software. Users may take many paths or choose many directions to accomplish a task. If certain required operations are not completed, it's an error regardless of how the user arrived at the screen. For example, Opportunity #1 can have the following error instances associated with it:

- User can't find access point;
- User launches an existing customer record instead of adding a new one;
- User launches a new vendor record instead of a new customer record.

Each error instance is unique, yet all are associated with the same "opportunity" for making an error in this component of the task.

Additionally, each time the task above is attempted it has 6 opportunities for an error to occur, but the "total opportunities" is dependent on the number of times the task was attempted in the sample. For instance, 6 opportunities x 20 participants = 120 Total Opportunities in the sample.

The defective rate can be calculated using the discrete data equation in the following example:

Say 20 participants completed a task that has 6 opportunities for error. A total of 36 errors were identified across all participants.

- Total Defects / Total Opportunities = Defective Rate
- $36 / (6 \times 20) = .3 = 30\%$ Defective

The following steps then calculate the process sigma:

- $1 - .30 = .70 = 70\%$ Quality Level
- The corresponding z-score to 70% on a standardized z-table is .524
- $.524 + 1.5$ sigma shift = 2.02 sigma

The Error rate for that task would then be described as having a quality level of 70% or 2.02 sigma.

Standardizing Satisfaction Scores

The ordinal data from most questionnaires can be treated as pseudo continuous data—especially if the scale has more steps (at least 5). Post-task satisfaction can be measured across multiple dimensions using semantic distance scales including the After Scenario Questionnaire [9].

Prior research across numerous usability studies suggests that systems with “good-usability” typically have a mean rating of 4 on a 1-5 scale [13]. Therefore we set the specification limit to 4 for our post-task questions using 5-point semantic distance scales⁹.

Having identified the target specification level for task satisfaction, we can now apply the Six Sigma *Continuous Method* when standardizing 5-point satisfaction scale data. For example, assuming that the average post-task satisfaction for a task attempted by 20 participants is 3.6 and the standard deviation is 1.1, we can calculate the defective rate for task satisfaction as follows:

- $(\text{Sample Mean} - \text{Spec}) / \text{St. Dev} = \text{z-score}$
- $(3.6 - 4) / 1.1 = -.364$
- $-.364 = 36\%$ on a standardized z-table = 36% Quality Level (64% Defective Rate)

Since the average of the sample was below the goal, the z-score is negative. The process sigma is then simply:

- $-.364 + 1.5 = 1.14$ sigma

Post-task satisfaction for that task would then be described as having a quality level of 36% or 1.14 sigma.

Standardizing Task Times

We use the continuous data method to calculate process sigma for task time. To do so, however, the specification limit or the maximum acceptable task time needs to be identified. There is no hard and fast rule for coming up with how long a task should take. The only requirement is that in some way the time should be based on user data. See [16] for a discussion of deriving a “bootstrapped” specification limit from user behavior.

Once a specification limit is identified, it is used again in the continuous calculation. Let’s assume a sample of 20 users took an average of 120 seconds (*SD* 35) to complete a task and the goal was identified as 150 seconds.

- $(\text{Sample Mean} - \text{Spec}) / \text{Standard Deviation} = \text{normal deviate (or z-score)}$.
- $(120 - 150) / 35 = -.804$

Since lower task times are the goal, we reverse the sign of the z-score when calculating process sigma and $-.804$ becomes $.804$.

- $.804 = 79\%$ on standardized z-table = 79% Quality Level (21% Defective Rate)
- $.804 + 1.5 = 2.30$ sigma

⁹ For 7 point scales, the same study recommends using 5.6 as the specification limit. This value may change depending on the goals of the study and the analyst may consider alternative specification limits.

Task time for that task would then be described as having a quality level of 79% or 2.30 sigma.

Standardizing Other Usability Metrics

The processes described above can be applied to other usability metrics in accordance to the guidelines specified by their data types. For instance, the continuous data method can be applied to time spent recovering from errors and the discrete method can be applied to unsuccessful help use attempts.

Reporting the New Standardized Metrics

The four standardized usability metrics can now be reported using the quality level and the process sigma. Table 1 shows the standardized form with the original raw values of the four metrics.

Table 1: Process Sigma and Quality Level for four usability metrics

Metric	Process Sigma	Quality Level	Raw Value
Completion	2.34 sigma	80%	80%
Errors	2.02 sigma	70%	36 Errors
Satisfaction	1.14 sigma	36%	3.6
Times	2.30 sigma	79%	120 seconds

DISCUSSION

Now that disparate usability metrics can be expressed in standardized terms of sigma values or quality levels, there are two major benefits. First, since the standardized metrics were derived from the user-defined goals, the analyst can see which metrics are falling short and which are exceeding these goals. In the example in Table 1 above, satisfaction is falling well short of the goal whereas completion and time are exceeding the specification limit—although there is still much room for improvement.¹⁰ Second, the common scale makes reporting and ranking much easier than with the raw data. This benefit becomes immediately clear when multiple tasks are juxtaposed in a report (see Table 2 below).

Table 2: Quality Level for four usability metrics for three tasks.

Metric	Task 1	Task 2	Task 3
Completion	95%	80%	55%
Errors	80%	70%	72%
Satisfaction	79%	36%	62%
Times	75%	79%	20%

Consolidating the Metrics

Four numbers on the same scale are easier to interpret than four numbers on different scales. It’s still not the easiest task discerning which task needs the most improvement. The next obvious benefit to this standardization method would be to present a single, consolidated usability metric that maintains the information contained within the component aspects.

¹⁰ If the quality level is at 50% (1.5 sigma) this means only half the users are meeting the specification limit, certainly not an ideal situation.

UPA Conference 2005

An earlier analysis of four summative evaluations with 1860 task observations has found that the four aspects of usability correlate and the simple average of their standardized values can summarize the majority of variance in the four measures into a single usability metric (SUM) [17].

Maintaining the completeness of four measures while removing the complexity provides the best of both worlds. The standardized and consolidated model provides one number to make high-level decisions and comparisons with different versions of the same product or with competing products [18]. Looking again at the three tasks with the single usability score, it becomes clearer which tasks have worse usability (see Table 3).

Table 3: Summative Usability Metric for four usability metrics for three tasks

Metric	Task 1	Task 2	Task 3
SUM (Single Usability Metric)	82%	66%	52%
Completion	95%	80%	55%
Errors	80%	70%	72%
Satisfaction	79%	36%	62%
Times	75%	79%	20%

There has been a need for a single metric in measuring and reporting usability, as proposals in the usability literature make clear. Prior attempts to derive a single measure for the construct of usability have relied solely on subjective assessments of the system through standardized questionnaires: SUMI [7], QUIS [4] and SUS [3], or through a method of magnitude estimation [11]. Other methods use a rank based system when assessing competing products [8] and one method used performance measures only [2].

While these methods provide helpful information to the analyst in making decisions about usability, it is difficult to think these methods are taking into account all aspects of usability in light of the guidance set by ISO 9241[1] and ANSI 354-2001[6]. The reliance on solely objective or subjective measures of usability leaves one to question the ability of these models to effectively describe the entire construct of usability.

As stated by Molich, et al [10], the effectiveness of a usability test is dependent on the chosen tasks, the methodology, and the persons in charge of the test. We acknowledge that the reliability of any metrics procured from a summative evaluation can be equally dependant on these factors. However, having a model for deriving a standard measure is also a powerful tool to evaluate differences in testing procedures.

Conclusion

Our model of standardizing the various aspects of usability provides the analyst with a clearer quantitative model of usability, and in turn provides business leaders with a more succinct and accurate description of usability at the task, feature and product level. While all activities in usability evaluations can never be summarized with a few numbers, having a method to better communicate salient aspects will go a long way in making usability an integral part of the product development lifecycle.

ACKNOWLEDGEMENTS

We would like to thank Lynda Finn from Statistical Insights for her thorough knowledge of Six Sigma.

REFERENCES

1. ANSI (2001). *Common industry format for usability test reports* (ANSI-NCITS 354-2001). Washington, DC: American National Standards Institute
2. Babiker, E.M., Fujihara, H., Boyle, Craig. D. B. (1991). A metric for hypertext usability. In *Proc. 11th Annual International Conference on Systems documentation*, (pp.95-104). ACM Press.
3. Brooke, J. (1996). SUS: A "quick and dirty" usability scale. In P. Jordan, B. Thomas, and B. Weerdmeester (Eds.), *Usability Evaluation in Industry* (pp.189-194). London: Taylor and Francis.
4. Chin, J. P., Diehl, V. A., and Norman, K. L. (1988). Development of an instrument measuring user satisfaction of the human-computer interface. *Proc. CHI '88*. (pp. 213-218). Washington, D.C.: ACM Press.
5. Harry, M. J (1987). The Nature of Six Sigma Quality. Technical Report, Government Electronics Group, Motorola Inc. Scottsdale, AZ.
6. ISO. (1998). *Ergonomic requirements for office work with visual display terminals (VDTs) – Part 11: Guidance on usability* (ISO 9241-11:1998(E)). Geneva, Switzerland: Author.
7. Kirakowski, J. (1996). The Software Usability Measurement Inventory: Background and usage. In P. Jordan, B. Thomas, and B. Weerdmeester (Eds.), *Usability Evaluation in Industry* (pp. 169-178). London, UK: Taylor and Francis. (Also, see <http://www.ucc.ie/hfrg/questionnaires/sumi/index.html>)
8. Lewis, J (1991) A Rank-Based Method for the Usability Comparison of Competing Products from the Proceedings of the Human Factors and Ergonomics Society 35th Annual Meeting San Francisco California (pp. 1312-1316).
9. Lewis, J. R. (1992). Psychometric evaluation of the Post-Study System Usability Questionnaire: The PSSUQ. in *Proceedings of the Human Factors Society 36th Annual Meeting* (pp. 1259-1263). Atlanta, GA: Human Factors Society.
10. Molich, R., Ede, M., Kaasgaard, K., and Karyukin, B (2004). Comparative Usability Evaluation. *Behaviour & Information Technology*, 23(1), 65-74.
11. McGee, M (2004). Master usability scaling: magnitude estimation and master scaling applied to usability measurement. In *Proc. CHI 2004*, (pp 335 - 342). Washington, D.C.: ACM Press.
12. Nielsen, J (2003) Two Sigma: Usability and Six Sigma Quality Assurance from Alertbox <http://www.useit.com/alertbox/20031124.html> article retrieved June 2004
13. Nielsen, J. and Levy, J. (1994) Measuring Usability: Preference vs. Performance. *Communications of the ACM*, 37, p. 66-76
14. Park, Kyung S. (1997). Human Error. In Gavriel Salvendy (Ed.), *The Handbook of Human Factors and Ergonomics*, (2nd Edition). John Wiley & Sons.
15. Pyzdek, Thomas (2003). *The Six Sigma Handbook*. McGraw-Hill Publishing
16. Sauro, J. & Kindlund E. (2005) "How long should a Task Take? Identifying Specification Limits for Task Times in Usability Tests" in *Proceeding of the Human Computer Interaction International Conference (HCII 2005), Las Vegas, USA*
17. Sauro, J. & Kindlund E. (2005) "A Method to Standardize Usability Metrics into a Single Score." in *Proceedings of the Conference in Human Factors in Computing Systems (CHI 2005) Portland, OR (p 401 – 409)*
18. Sauro, J. & Kindlund E. (2005) "Using a Single Usability Metric (SUM) to Compare the Usability of Competing Products" in *Proceeding of the Human Computer Interaction International Conference (HCII 2005), Las Vegas, USA*