National Research Council Canada
Conseil national de recherches Canada

Institute for Information Technology
Institut de Technologie de l'information

ERB-1067

NRC·CNRC

Estimating the Extent of Standards Use : The Case of ISO/IEC 15504

Khaled El-Emam and Inigo Garro

November 1999

Canada

NRC 43612

National Research
Council Canada

Conseil national
de recherches Canada

Institute for
Information Technology

Institut de Technologie
de l'information

Estimating the Extent of Standards Use : The Case of ISO/IEC 15504

Khaled El-Emam and Inigo Garro

November 1999

Estimating the Extent of Standards Use: The Case of ISO/IEC 15504

Khaled El Emam
National Research Council, Canada
Institute for Information Technology
Building M-50, Montreal Road
Ottawa, Ontario
Canada K1A OR6
Khaled.El-Emam@iit.nrc.ca

Iñigo Garro
European Software Institute
Parque Tecnológico de Zamudio # 204
48170 Zamudio, Vizcaya
Spain
garro@esi.es

Abstract

There has been a proliferation of software engineering standards in the last two decades. While the utility of standards in general is acknowledged, thus far little attempt has been made to evaluate the success of any of these standards. One suggested criterion of success is the extent of usage of a standard. In this paper we present a general method for estimating the extent to which a standard is used. The method uses a capture-recapture model that was originally proposed for estimating birth and death rates in human populations. We apply the method to estimate the number of software process assessments that were conducted world-wide between September 1996 and June 1998 using the emerging ISO/IEC 15504 international standard. Our results indicate that 1264 assessments were performed with a 90% confidence interval of 916 and 1895. The method used here can be applied to estimate the extent of usage of other software engineering standards, and also of other software engineering technologies. Such estimates can benefit standards (or technology) developers, funding agencies, and researchers by focusing their efforts on the most widely used standards (or technologies).

1 Introduction

It has been stated that there are more than 250 software engineering standards in existence today, issued by various professional, national, and international standards bodies (Fenton et al., 1994). Despite the proliferation of standards, it has also been claimed that there is a poor industrial take-up of standards (Fenton et al., 1993). If this were true, then this would be quite distressing for the software engineering standards community given that acceptance and usage represents one of two criteria suggested for evaluating the “success” of standards (Meek, 1996). Perhaps due to this perceived poor uptake there has been a recent effort by the IEEE Computer Society’s Software Engineering Standards Committee to survey users and potential users of IEEE standards to understand their needs and obtain the subjective evaluations of its standards by those who use them (Land, 1997; Land 1999).

In fact, the extent to which software engineering standards are actually used is unknown. A recent book on the topic does not indicate the extent of usage of the standards that it covers (Moore, 1998). To our knowledge, there have been no empirical studies thus far that have attempted to estimate the extent to which single or multiple software engineering standards are actually used by industry.

In addition to providing a means for evaluating the “success” of a standard, estimates of standards usage can benefit the developers of standards by prompting efforts to improve, update, or phase-out standards that are underused, and can provide standards developers with objective feedback on their efforts to make standards more usable (Land 1997; Land 1999). Furthermore, researchers can focus their efforts on empirically evaluating standards that are used frequently, as has been suggested (Fenton et al., 1994; Fenton et al., 1993).[1]

It is tempting to rely on the most obvious approaches for estimating the extent of usage of a software engineering standard, namely the number of sales and/or the number of 'certifications'. Both of these approaches have fundamental flaws. It is entirely plausible that many of the purchasers of software engineering standards do not actually use them. This may be because they find that the standards are not applicable (for example, the comments received in recent surveys bemoan the inapplicability of standards to small organizations (Land, 1997; Land, 1999)), or because the purchasers are educational institutions. Furthermore, frequently standards are used "indirectly" as reference material, training material, or used as general guidelines that are modified for organizational use (Land, 1997; Land, 1999). Under such conditions the definition of usage can be tricky, and the concordance between sales and usage flimsy at best. A good example of the certification argument is the number of software organizations whose ISO 9000 certification scope is directly related to software development (Weissfelner, 1999). However, most software engineering standards do not have a world-wide certification scheme behind them where such data can be collected. Also, it is entirely plausible that organizations would be using a standard for their own benefit and not seek a certification if their customers do not demand it. Therefore, it is necessary to devise an alternative method to determine the extent to which a standard is used.

A more direct approach for estimating the extent of usage of a software engineering standard is proposed in this paper. This approach utilizes a capture-recapture (CR) model. CR models have been used in the biological sciences to estimate the size of animal populations (Otis et al., 1978; White et al., 1982), and in epidemiology to estimate birth and death rates (Chandra Sekar and Deming, 1949; Greenfield, 1975), as well as the size of diseased populations (Hook and Regal, 1995). In software engineering, they have been applied to estimate the number of remaining defects in an inspected document (Briand et al., 1997; Briand et al., 2000), the number of defects in software (Duran and Wiorkowski, 1981; Ohba, 1982), and for deciding when to stop inspections (El Emam and Laitenberger, 1999).

We apply the CR approach to estimate the number of software process assessments that were performed world-wide using the emerging ISO/IEC 15504 international standard during the period of September 1996 to June 1998, the period of our study. Our results indicate that 1264 assessments were performed with a 90% confidence interval of 916 and 1895. In addition to its applicability to the estimation of the extent of standards usage, the general approach can be used to estimate usage of other software engineering technologies.

[1] Ideally the empirical evaluation should be performed before the standards are actually developed. However, given the plethora of standards that currently exist, there is a need to focus effort on those that have the largest user community. The results of empirical evaluation, especially studies that demonstrate the benefits of standards, are likely to improve their usability (Fenton and Neil, 1998), and increase their uptake (Land, 1997; Land, 1999).

In the following section we describe our data collection method in the case of estimating ISO/IEC 15504 assessments, and the estimation approach. Section 3 presents our results, and Section 4 concludes the paper with a summary and directions for future work.

2 Research Method

In this section we motivate our study, describe the data that was collected and our data collection method, then derive the estimator that we use and describe the technique for obtaining confidence intervals.

2.1 Motivation for the Study

ISO/IEC 15504 is an emerging international standard on software process assessment (El Emam et al., 1998). It defines a reference model for software process assessment, and a set of requirements on assessment models and methods.[2] During its development it goes through a series of ballots by national bodies (e.g., the British Standards Institute is a national body) whereby they provide technical comments and (dis)approve the documents. The whole process can take quite a few years, and at the time of writing it has been 6 years since the work on this standard had commenced.

The standardization route adopted for ISO/IEC 15504 allows the documents to be used in industry in between ballots. As the ISO/IEC 15504 documents have reached maturity in the recent past, there has been a rising interest in determining the extent to which ISO/IEC 15504 has been used thus far. This study was designed to address this specific need, although the general method can be applied to other standards and other technologies (such as tools or object-oriented development methods).

2.2 Data Collection

To estimate the number of assessments, we require at least two methods of ascertainment. Here we focus only on the case of exactly two methods. A method of ascertainment identifies assessments that were conducted. The two methods do not necessarily have to be independent, as will be discussed below.

In the context of ISO/IEC 15504 two such methods were available: the SPICE Trials and the ISO/IEC working group[3] that is developing ISO/IEC 15504.

The emerging ISO/IEC 15504 international standard is unique in software engineering standardization in that at the outset there has been an international effort to empirically evaluate its efficacy and usability in

[2] Assessments that satisfy the requirements are claimed to be compliant. Based on public statements that have been made thus far, it is expected that some of the more popular assessment models and methods will be consistent with the emerging ISO/IEC 15504 International Standard. For example, Bootstrap version 3.0 claims compliance with ISO/IEC 15504 (Bicego et al., 1998), and the future CMMI product suite is expected to be consistent and compatible (Software Engineering Institute, 1998a).

A mapping between the processes defined in ISO/IEC 15504 and the SW-CMM is available from (Software Engineering Institute, 1998b)

[3] Formally, this group is designated as ISO/IEC JTC1/SC7 WG10.

practice. Such empirical evaluations are informing the development of the international standard. This effort is known as the SPICE Trials (El Emam et al., 1998).

The SPICE Trials have been divided into three broad phases corresponding with the different stages that an ISO standard has to progress through during its development. We are interested in the second phase.[4] This phase started in September 1996 and data collection ceased in June 1998. A well defined infrastructure was set up during this period to collect data and to ensure its quality. This is described below.

During the trials, organizations contribute their assessment ratings data to an international trials database located in Australia, and also fill up a series of questionnaires after each assessment. The questionnaires collect information about the organization and about the assessment. From the SPICE Trials perspective, the world is divided into five regions: Europe and South Africa, South Asia Pacific, North Asia Pacific, Canada and Latin America, and USA. For each of these regions there is an organization that is responsible for managing the data collection and providing support. The organizations were: the European Software Institute, Griffith University, Nagoya Municipal Industrial Research Institute, Center de recherche informatique de Montreal, and the Software Engineering Institute respectively. These are termed “Regional Trials Co-ordinators”. Within each of these regions are a number of “Local Trials Co-ordinators” who operate at a more local level, such as within a country or state. There were 26 such co-ordinators world-wide during the second phase of the SPICE Trials. The co-ordinators (local or regional) interact directly with the assessors and the organizations conducting the assessments. This interaction involves ensuring that assessors are qualified, making questionnaires available, answering queries about the questionnaires, and following up to ensure the timely collection of data.

During phase 2 of the trials we collected data on 70 assessments world-wide. This constitutes the first method of ascertainment.

A separate entity from the SPICE Trials is the ISO/IEC working group that is developing ISO/IEC 15504 (known as WG10). This group consists of delegates from national bodies that are members of ISO. During the development of ISO/IEC 15504 members of WG10 also performed assessments using the emerging international standard within their organizations or otherwise within their home countries. During a meeting in Canada in November 1998 all members of WG10 attending the meeting were provided with a data collection form that requested them to enumerate all of the ISO/IEC 15504 assessments that they have been involved in directly or indirectly. After completion of the form the Regional Trials Co-ordinators verified each entry with the source and matched the entries with the assessments that were known about in the SPICE Trials.

Therefore, we have counts of assessments using two methods of ascertainment, and a matching of the assessments counted using both methods.

An important point to mention is that the definition of what constitutes an ISO/IEC 15504 assessment must be clear. For example, someone reading the ISO/IEC 15504 documents does not constitute an assessment. We defined an assessment in terms of two criteria:

- The ISO/IEC 15504 documents define requirements for a series of activities that must be performed during a conformant assessment. There are potentially a multitude of methods that can meet these requirements.[5] We considered an assessment as one that performed the stipulated activities.
- The ISO/IEC 15504 documents define requirements for the assessment model that is used. While the ISO/IEC 15504 documents provide an exemplar assessment model, an organization is free to use any model that meets the requirements. During phase 2 of the SPICE Trials there were six models that were claimed to meet these requirements. Therefore, if any of these models was used, then it was considered as a valid assessment.

2.3 Data Analysis

In describing the data analysis, we first present the method that was used for coming up with the point estimate of the number of assessments performed, and then this is followed by a description of the method that was used to construct 90% confidence intervals.

2.3.1 Point Estimates

To explain the method of estimation, we start off by casting the collected data in the form of a 2x2 contingency table as follows:

		Ascertained by Method 1		
		Yes	**No**	
Ascertained by Method 2	**Yes**	n_{11}	n_{12}	N_{1+}
	No	n_{21}	n_{22}	N_{2+}
		N_{+1}	N_{+2}	

Table 1: Notation for a 2x2 contingency table.

The rows represent the number of assessments ascertained by method 2, and the columns represent the number of assessments ascertained by method 1. In this table, the value n_{22} is unknown, and consequently so are the values N_{2+} and N_{+2}. The objective is to estimate n_{22}. Subsequently, it is possible to come up with an estimate of the whole population size.

It is known that the correlation coefficient between the two methods is given by:

[4] The version of the ISO/IEC 15504 documents that were studied during Phase 2 of the SPICE Trials is known as ISO/IEC PDTR 15504.

[5] For example, some methods are adaptations of full CBA IPI's, while others targeted at small organisations are set up as an interactive one or two day workshop where senior managers and staff go through the processes that are being assessed (of course, where there are reporting relationships amongst attendees, then the workshop is performed seperately for different groups).

$$r = \frac{n_{11}n_{22} - n_{12}n_{21}}{\sqrt{(n_{11}+n_{12})(n_{21}+n_{22})(n_{11}+n_{21})(n_{12}+n_{22})}} \qquad \textbf{Eqn. 1}$$

This is the phi coefficient (Sheskin, 1997), which is equivalent to the Pearson product moment correlation when the variables are binary.

We also know that, $n_{12} \geq 0$, $n_{21} \geq 0$, $n_{11} > 0$, and $n_{22} \geq 0$. This means that at least one assessment must be ascertained by both methods, which is the case in our study.

If the two methods are independent, then we would expect that $r = 0$. Under such a condition we have:

$$n_{11}n_{22} = n_{12}n_{21} \qquad \textbf{Eqn. 2}$$

and we can therefore estimate n_{22}:

$$\hat{n}_{22} = \frac{n_{12}n_{21}}{n_{11}} \qquad \textbf{Eqn. 3}$$

and the total population size:

$$\hat{N} = n_{11} + n_{12} + n_{21} + \frac{n_{12}n_{21}}{n_{11}} = \frac{N_{+1}N_{1+}}{n_{11}} \qquad \textbf{Eqn. 4}$$

This is the estimator suggested by Chandra Sekar and Deming (Chandra Sekar and Deming, 1949), and is known in the ecology community as the Lincoln-Petersen (LP) estimator. The LP estimate is an instance of the general maximum likelihood estimator (Otis et al., 1978), but limited to two methods of ascertainment, and is frequently used to estimate the size of animal populations.

The above CR approach for estimating the number of assessments (Eqn. 4) makes three assumptions:

1. *That the two ascertainment methods are independent.* This assumption is not tenable in our case since we expect that two ascertainment methods to be positively correlated. This means that if an assessment has been part of the trials then there is a nonnegligible chance that it will be ascertained by WG10. The reason being that, even though Phase 2 of the SPICE Trials was open to the whole of the software engineering community, members of WG10 acted as contact points in their regions and would therefore be expected to know about them. In general, if this dependence is not taken into account when estimating the number of assessments, then the resulting estimate will be too low (i.e., an underestimate).

2. *That the two methods could have different probabilities of ascertaining an assessment.* This is certainly plausible, since a priori we would expect that the probability of ascertainment using the WG10 method would be higher than the SPICE Trials methods given that it requires much more

effort to provide data to the trials. Although, if the probabilities are similar then this would also be accounted for by the LP estimator (see Eqn. 4).

3. *That assessments have equal probabilities of being ascertained.* A priori, we expect this to be a reasonable assumption to make in our case. However, a recent simulation study of this LP estimator under the independence assumption noted that if this assumption is violated considerably, then the population estimate will be too low (El Emam and Laitenberger, 1999). Therefore, if this assumption is violated, this means that our estimate would be considered a lower bound, and would be conservative.

From the above exposition we can conclude that the first assumption needs to be dealt with. The second assumption does not introduce difficulty since it is accounted for by the estimator. The third assumption, although we contend that it is reasonable, if it is violated considerably during our study it will lead to conservative results.

We therefore describe a method proposed and applied by Greenfield for the estimation of population size when the methods are not independent (Greenfield, 1975). We then modify Greenfield's estimator slightly to account for our specific context.

From Eqn. 1, we have:

$$r\frac{\sqrt{(n_{11}+n_{12})(n_{11}+n_{21})}}{n_{11}} = \frac{n_{11}n_{22}-n_{12}n_{21}}{n_{11}\sqrt{(n_{12}+n_{22})(n_{21}+n_{22})}}$$

$$= \frac{n_{11}n_{22}-n_{12}n_{21}}{n_{11}\sqrt{(n_{12}n_{21}+n_{12}n_{22}+n_{21}n_{22}+n_{22}^2)}} \quad \textbf{Eqn. 5}$$

it is clear from the above that:

$$n_{11}n_{22}-n_{12}n_{21} \le n_{11}\sqrt{(n_{12}n_{21}+n_{12}n_{22}+n_{21}n_{22}+n_{22}^2)} \quad \textbf{Eqn. 6}$$

Therefore, the left hand side of Eqn. 5 is ≤ 1. It follows that:

$$r \le \frac{n_{11}}{\sqrt{(n_{11}+n_{12})(n_{11}+n_{21})}} \quad \textbf{Eqn. 7}$$

and therefore the expression in Eqn. 7 represents the maximum value of the correlation coefficient:

$$r_{\max} = \frac{n_{11}}{\sqrt{(n_{11}+n_{12})(n_{11}+n_{21})}} \quad \textbf{Eqn. 8}$$

It is also clear from Eqn. 5 that r is a monotonically increasing function of n_{22} and that the minimal value of n_{22} is zero. Therefore, the minimal value of the correlation coefficient is obtained from Eqn. 1 by substituting the zero value of n_{22}, giving:

$$r_{\min} = -\sqrt{\frac{n_{12} n_{21}}{\left(n_{11} + n_{12}\right)\left(n_{11} + n_{21}\right)}} \qquad \textbf{Eqn. 9}$$

In our case it is highly improbable that the correlation is negative given the two ascertainment methods. Therefore, we can reasonably assume that the minimal value of the correlation is zero (i.e., independence).

According to the method of Greenfield (Greenfield, 1975), he then proposes the following mid-point estimate of the correlation coefficient:

$$\hat{r}_g = \frac{1}{2}\left(r_{\max} + r_{\min}\right) \qquad \textbf{Eqn. 10}$$

Which he then uses as the basis for estimating birth and death rates in Malawi.

Since our minimal correlation is zero, this can be modified to:

$$\hat{r} = \frac{1}{2} \cdot r_{\max} \qquad \textbf{Eqn. 11}$$

Now that we have an estimate of the correlation coefficient, we need to compute n_{22}. By simple algebraic manipulation of Eqn. 1, it can be shown that (Greenfield, 1975):

$$\hat{n}_{22} = -\frac{1}{2} B + \sqrt{\left(A + \frac{1}{4} B^2\right)} \qquad \textbf{Eqn. 12}$$

where:

$$A = \frac{n_{12} n_{21}\left(n_{12} n_{21} - \hat{r}^2\left(n_{11} + n_{12}\right)\left(n_{11} + n_{21}\right)\right)}{\hat{r}^2\left(n_{11} + n_{12}\right)\left(n_{11} + n_{21}\right) - n_{11}^2} \qquad \textbf{Eqn. 13}$$

and:

$$B = \frac{\hat{r}^2\left(n_{12} + n_{21}\right)\left(n_{11} + n_{12}\right)\left(n_{11} + n_{21}\right) + 2 n_{11} n_{12} n_{21}}{\hat{r}^2\left(n_{11} + n_{12}\right)\left(n_{11} + n_{21}\right) - n_{11}^2} \qquad \textbf{Eqn. 14}$$

With an estimate of n_{22} it is then easy to compute the total population estimate.

2.3.2 Confidence Intervals

Greenfield does not provide a method for computing the estimated variance for his estimator (Greenfield, 1975). We therefore construct a 90% nonparametric bootstrap confidence interval to gauge the uncertainty in the point estimate using the bias-corrected percentile method (Efron and Tibshirani, 1993).

Our estimator assumes that each assessment has the same probability of being ascertained by either method, but the methods may differ in their probability of ascertainment (i.e. assumption (2) in Section 2.3.1). Buckland and Garthwaite (Buckland and Garthwaite, 1991) describe how to construct bootstrap confidence intervals for this type of estimator. We first extend the data set with assessments that are not found by either method such that the total number of assessments is equal to $\hat{N}$. We sample assessments with replacement from this data set 1000 times such that each sample is of size $\hat{N}$. For each sample a new estimate $\hat{N}'$ is computed. The 1000 $\hat{N}'$ estimates provide us with the bootstrap distribution. The basic percentile method for constructing 90% confidence intervals uses the 5th and 95% quantiles as the end points of the interval. An improvement of this, the bias-corrected percentile method, corrects for possible biases in the initial estimate $\hat{N}$ (Efron and Tibshirani, 1993). The 90% confidence interval means that if we were to repeat the study a large number of times and each time calculate the confidence interval, this interval will contain the population size 90% of the time.

3 Results

The data that was actually collected can be represented in Table 2. This indicates that 70 assessments were ascertained by the trials[6] and 168 assessments were ascertained through WG10. There were 17 assessments that were matched and verified for both sources. This gives a total of 221 assessments that we can be certain about.

		Ascertained by WG10	
		Yes	No
Ascertained by the Trials	Yes	17	53
	No	151	

Table 2: Obtained counts in a 2x2 contingency table from our study.

[6] Of these 70 assessments, 1 was performed in Canada, 24 in Europe, 10 in the Northern Asia Pacific region, 34 in the Southern Asia Pacific region, and 1 in the USA. Based on this distribution, we could state that most assessments performed were concentrated in Europe and Southern Asia Pacific (mainly Australia). At the time of writing, the expressions of interest in the SPICE Trials from organizations around the world follows the following distribution: 52 in Canada and Latin America, 108 in Europe, 20 in the Northern Asia Pacific region, 34 in the Southern Asia Pacific region, and 105 in the USA. Although these are not necessarily good predictors of actual assessments that will be performed in the future, they do indicate a potential dramatic increase in participation from Europe, the USA, and Canada and Latin America.

The point estimate of the number of assessments using the method outlined earlier is 1264. The 90% bootstrap confidence intervals are 916 and 1895.

After obtaining this estimate we collected anecdotal evidence as to the plausibility of the point estimate. This involved presenting the results to eleven individuals involved in the development of ISO/IEC 15504 and involved in the SPICE Trials. With little exception, respondents felt that this was a plausible number given the assessments that they know about or that have been conducted in their region of the world *and* that were never reported by these two methods because they would have been difficult or costly to verify each one.

The limitation of our method is that it assumes that assessments have equal probabilities of being ascertained. If this assumption is extremely violated, we would expect our estimate to be conservative. Further improvements to the analytical method that we used is to estimate the number of potential assessments that could be performed, and compare that with the number of assessments that were actually performed.

In general, the approach that we have presented here can be used to estimate the extent of usage of other software engineering standards, and for that matter, any software engineering technology (for example, tools, object-oriented methods, formal methods, and other process assessment models and methods).

That there is concern in quantifying the extent of the usage of software engineering methods and tools is exemplified by the surveys reported in (Beck and Perkins, 1983; Necco et al., 1987). However, these studies are either not specific about response rate or have very low response rates, ignore item non-response, and therefore do not even account for missing information. In fact, the dearth of surveys and the inconsistency in the results of surveys that attempt to quantify method and tool usage has been bemoaned in (Wynekoop and Russo, 1993). A CR approach may alleviate some of these difficulties.

Potential primary beneficiaries of CR estimates of method and tools usage could be:

- Government agencies wishing to determine the success of technology dissemination initiatives. Also, non-governmental funding and sponsoring organizations that are interested in evaluating the extent of usage of software engineering technologies that they support.
- Developers of technology to determine the extent to which their technologies are in actual usage. This can help initiate or focus technology transition efforts.
- Researchers who wish to focus their efforts on evaluating and improving technologies that are in wide usage. This also applies to those who fund research work in that they can focus their support on technologies that are in wide usage by industry.

The prerequisites to operationalizing the approach that we have presented here are a clear definition of “usage” of the technology, and the availability of at least two methods of ascertainment. If there are more than two methods of ascertainment, then techniques such as log-linear models that can take into account dependence amongst the methods through interaction effects can be applied (Fienberg, 1972).

4 Conclusions

In this paper we have presented a method for estimating the extent of usage of a software engineering standard, namely ISO/IEC 15504. The current plan is to produce such estimates on the use of ISO/IEC 15504 on a biannual basis. The method is generally applicable and can be used for other standards, as well as other software engineering technologies.

As a starting point for improving on the general method that we have presented here, it would be of utility to compare the current approach with others proposed that can deal with two methods and dependence between them, such as (Ebrahimi, 1997). Ideally, such a comparison would be conducted through a Monte Carlo simulation so as to gain a general understanding of their relative advantages.

5 Acknowledgements

We wish to thank Anatol Kark for his comments on an earlier version of this paper, and the anonymous reviewers for their feedback and helpful suggestions for improving the paper.

6 References

Bicego, A., Khurana, M., Kuvaja, P., 1998. Bootstrap 3.0: Software Process Assessment Methodology. Proceedings of SQM'98.

Briand, L., El Emam, K., Freimut, B., Laitenberger, O., 1997. Quantitative Evaluation of Capture Recapture Models to Control Software Inspections. Proceedings of the Eighth International Symposium on Software Reliability Engineering, 234-244.

Briand, L., El Emam, K., Freimut, B., Laitenberger, O., 2000. A Comprehensive Evaluation of Capture-Recapture Models for Estimating Software Defect Content. IEEE Transactions on Software Engineering (to appear).

Beck, L., Perkins, T., 1983. A Survey of Software Engineering Practice: Tools, Methods, and Results. IEEE Transactions on Software Engineering, 8(5), 541-561.

Buckland, S., Garthwaite, P., 1991. Quantifying Precision of Mark-Recapture Estimates Using the Bootstrap and Related Methods. Biometrics, 47, 255-268.

Chandra Sekar, C., Edwards Deming, W., 1949. On a Method of Estimating Birth and Death Rates and the Extent of Registration. Journal of the American Statistical Association, 44(245-248), 101-115.

Duran, J., Wiorkowski, J., 1981. Capture-Recapture Sampling for Estimating Software Error Content. IEEE Transactions on Software Engineering, 7(1), 147-148.

Ebrahimi, N., 1997. On the Statistical Analysis of the Number of Errors Remaining in a Software Design Document after Inspection. IEEE Transactions on Software Engineering, 23(8), 529-532.

Efron, B., Tibshirani, R., 1993. An Introduction to the Bootstrap. Chapman & Hall, San Francisco.

El Emam, K., Drouin, J-N, Melo, W. (eds.), 1998. SPICE: The Theory and Practice of Software Process Improvement and Capability Determination. IEEE CS Press.

El Emam, K., Laitenberger, O., 1999. Evaluating Capture-Recapture Models with Two Inspectors. Technical Report of the International Software Engineering Research Network, ISERN-99-08 (available at http://www.iese.fhg.de/network/ISERN/pub/technical_reports/isern-99-08.pdf).

Fenton, N., Littlewood, B., Page, S., 1993. Evaluating Software Engineering Standards and Methods. In: Thayer, R., McGettrick, R. (eds.), Software Engineering: A European Perspective, 463-470, IEEE CS Press.

Fenton, N., Pfleeger, S-L, Page, S., Thornton, J., 1994. The SMARTIE Standards Evaluation Methodology. Technical Report, (available from the Center for Software Reliability, City University).

Fenton, N., Neil, M., 1998. A Strategy for Improving Safety Related Software Engineering Standards. IEEE Transactions on Software Engineering, 24(11), 1002-1013.

Fienberg, S., 1972. The Multiple RecaptureCensus for Closed Populations for Incomplete 2^k Contingency Tables. Biometrika, 59(3), 591-603.

Greenfield, C., 1975. On the Estimation of a Missing Cell in a 2x2 Contingency Table. Journal of the Royal Statistical Society, Series A, 138, 51-61.

Hook, E., Regal, R., 1995. Capture-Recapture Methods inEpidemiology: Methods and Limitations. Epidemiologic Reviews, 17(2), 243-264.

Land, K., 1997. Results of the IEEE Survey of Software Engineering Standards Users. Technical Report, BTG Inc.,. Presented at the IEEE International Symposium on Software Engineering Standards.

Land, K., 1999. Second Software Engineering Standards Users' Survey. Presentation at the IEEE International Symposium on Software Engineering Standards.

Meek, B., 1996. Too Soon, Too Late, Too Narrow, Too Wide, Too Shallow, Too Deep. StandardView, 4(2), 114-118.

Moore, J., 1998. Software Engineering Standards: A User's Road Map, IEEE CS Press.

Necco, C., Gordon, C., Tsai, N., 1987. Systems Analysis and Design: Current Practices. MIS Quarterly, December, 461-475.

Ohba, M., 1982. Software Quality = Test Accuracy X Test Coverage. Proceedings of the 6th International Conference on Software Engineering, 287-293.

Otis, D., Burnham, K., White, G., Anderson, D., 1978. Statistical Inference from Capture Data on Closed Animal Populations. Wildlife Monographs, 62, 1-135.

Pfleeger, S-L, Fenton, N., Page, S., 1994. Evaluating Software Engineering Standards. IEEE Computer, September, 71-79.

Sheskin, D., 1997. Handbook of Parametric and Nonparametric Statistical Procedures, CRC Press.

Software Engineering Institute 1998a. CMMI A Specification Version 1.1. Available at http://www.sei.cmu.edu/activities/cmm/cmmi/specs/aspec1.1.html, 23rd April.

Software Engineering Institute 1998b. Top Level Standards Map: ISO 12207, ISO 15504 (Jan 1998 TR), Software CMM v1.1 and v2 Draft C. Available at http://www.sei.cmu.edu/pub/cmm/Misc/standards-map.pdf, February.

Weissfelner, S., 1999. ISO 9001 for Software Organizations. In: El Emam, K., Madhavji, N. (eds.), Elements of Software Process Assessment and Improvement, IEEE CS Press.

White, G., Anderson, D., Burnham, K., Otis, D., 1982. Capture-Recapture and Removal Methods for Sampling Closed Populations. Technical Report LA-8787-NERP, Los Alamos National Laboratory.

Wynekoop, J., Russo, N., 1993. System Development Methodologies: Unanswered Questions and the Research-Practice Gap. Proceedings of the International Conference on Information Systems, 181-190.

Khaled El Emam is currently at the National Research Council in Ottawa. He is the current editor of the IEEE TCSE Software Process Newsletter, the current International Trials Coordinator for the SPICE Trials, which is empirically evaluating the emerging ISO/IEC 15504 International Standard world-wide, co-editor of ISO's project to develop an international standard defining the software measurement process, and Knowledge Area Specialist for the Software Engineering Process area of the IEEE's Software Engineering Body of Knowledge. Previously, he worked on both small and large research and development projects for organizations such as Toshiba International Company, Yokogawa Electric, and Honeywell Control Systems. Khaled El Emam obtained his Ph.D. from the Department of Electrical and Electronics Engineering, King's College, the University of London (UK) in 1994. He was previously the head of the Quantitative Methods Group at the Fraunhofer Institute for Experimental Software Engineering in Germany, a research scientist at the Centre de recherche informatique de Montreal (CRIM), and a research assistant in the Software Engineering Laboratory at McGill University.

Iñigo Garro has worked at the European Software Institute (ESI) since July 1997, acting as Senior Consultant responsible for the definition and implementation of software process improvement programs for SMEs and large companies, using models such as ISO/IEC 15504, SW CMM and ISO 9001. He is a SPICE lead assessor and instructor for SPICE assessors. He worked as SPICE European Trials Co-ordinator in 1995, and became the SPICE International Trials Co-ordinator for Data Management in 1997, validating the emerging ISO/IEC 15504 standard for software process assessment. Previously, he worked at Sema Group, a multinational European company, in consultancy and technology transfer services for project management, quality assurance, software engineering and customer-relationship relationships. He was member of Sema Group sae Quality Board for the definition and implementation of the ISO 9001 quality system. He is member of the IEEE and participates on the Software Engineering Technical Committee of AENOR (Asociacion Española de Normalizacion). He actively participates in conferences in Europe and Latin America.