The Measurement Service in Software Engineering Environments

Manuel Bollaín, Juan Garbajosa
Technical University of Madrid

mbollain@eui.upm.es, jgs@eui.upm.es

pp. 389-400.

Magdeburger Schriften zum Empirischen Software Engineering. Shaker Verlag.

ISBN 3-8322-4405-0
The Measurement Service in Software Engineering Environments

Manuel Bollaín  
mbollain@eui.upm.es  
Juan Garbajosa  
jgs@eui.upm.es  
Technical University of Madrid  
System and Software Technology Group

Abstract

It is commonly recognized that software engineering practice, to achieve a good performance, depends on measurement. While the automation of a number of software engineering activities are receiving a lot of attention, it is not the case for measurement. However a good automated support seems to be essential to consolidate the measurement process practice. At present a common understanding is that automation requires software engineering environments and properly integrated tools. This paper provides a first approach to provide a set of requirements for a software engineering environment measurement service, taking as a basis the on-going ISO work on software engineering environments services and INCOSE measurement tools survey criteria.

1 Introduction

Automation of software engineering lifecycle processes is still an open issue for the software engineering community. For many years the main obstacle to achieve automation was basically acknowledged as a technical problem; this technical problem was mainly focused on tool integration. That was the case in the early 90’s. By that time platforms such as PCTE [1] were seen as part of the solution to the problem, achieving integrated environments the goal and, for that, reference models a path. Some opinions highlighting that those commonly accepted approaches might be wrong can be found in [2]. A reference model jointly published by European Computer Manufacturers Association (ECMA) and the US national Institute for Standards and Technology (NIST) was released in 1993 [3]. This report became an essential cornerstone as it set up a reference architecture for software engineering environments. However services are mainly technical services and do not address many of the end-user services. This shows the extent technical requirements were considered as the main concern. Later on, end user services were addressed in [4].

In the meantime the definition of lifecycle processes progressed. ISO/IEC 12207 [5] was published in 1995. Reference [5] describes a comprehensive set of processes, activities and tasks to be performed when acquiring or developing software. It does not address their implementation nor their automation. It has been, most probably, one of the standards that with a highest impact in the software community, regardless ISO 9000 series. It covers areas such as the specification, development, re-engineering, acquisition, supply, or maintenance of software based systems. During the 90’s process got a more and more important position within the community; process maturity and improvement were also considered.

Software engineering environments (SEE) are intended to provide, at least, partial automated support to the software lifecycle processes. But the term SEE, in literature, is used to denote very different entities: from a juxtaposition of tools running under the same operating system, up to a fully integrated environment, able to control all data, processes and activities in the software lifecycle.
For a user interested in a specific process, there is no clear vision of what services an SEE might provide, what the relationships between those services might be, and how an SEE can relate to the overall software engineering life cycle. For this reason, it is difficult to adequately assess products that claim to be an SEE. Frequently it is difficult to understand the role a software engineering tool might play in a SEE without an overall view of what a SEE is. It is difficult for an organization to achieve the proper level of automation in its process improvement efforts without a set of defined software engineering environment services. This problem can be resolved by generating a comprehensive, objective description of the services that make up a SEE. This background, after a study period, led to start a standards-making project Software Engineering – Software Engineering Environment Services that, once finalised, will be ISO/IEC 15940, in order to produce a standard that would put together those services useful from a software lifecycle process point of view. Reference [3] was used as a key input, given its broad support.

One pre-eminent concern in software engineering is measurement. Thus, an important part of the efforts in developing software process tools is oriented to support measurements definition, collection and analysis. At this point a problem arises: the lack of standards in terms of measurement tools characterization. This makes tool evaluation somehow difficult. INCOSE’s Systems Engineering Measurement Primer [6] presents a introduction to system measurement concepts and use. Some measurements frameworks such as [9] and [10] are also available. The interesting thing is that taking INCOSE work as a basis a scheme for evaluating measurement tools has been produced, also by INCOSE, the Measurement Tool Survey available in [12]. Despite INCOSE and ISO/IEC work were developed separately, the INCOSE framework can be mapped onto ISO/IEC 15940. It must be noticed that INCOSE work was developed thinking in systems while ISO/IEC 15940 had software as a target; however this will not be a major problem as it can be concluded from this paper. Therefore it is possible to set up a basis for a scheme for evaluating measurement tools in the context of a software engineering environment.

In literature a number of works on supporting measurement by tools, and measurement deployment can be found such. Since the interest for measurement and metrics based tools comes since the rise of the automation and integrated environment times [13,14,15,16,17,18,19,20,21,22,23,24,25,26]. However a comprehensive description in terms of tools features and environment services has not been provided before. Tool characteristics are either not studied in depth or not in the context of a software engineering environment.

This paper is structured as follows: In section 2 a survey of the upcoming ISO/IEC 15940 Software engineering environment services is provided. In section 3, an introduction to INCOSE Measurement Process: Primer is presented. In section 4, a mapping between INCOSE Measurement Tools Survey and ISO/IEC 15940 is presented. Finally, in section 5, some next steps are identified.

2 A survey of the upcoming ISO/IEC 15940 Software engineering environment services

The effort oriented to provide a standard that describes software engineering environment services is expected to fructify in the coming ISO/IEC 15940. Part of the current effort and view of the upcoming ISO/IEC 15940 has been described [10]. SEEs refer to a collection of services, partially or fully automated by tools, tools that are used to support software engineering activities. A SEE provides automated services for the engineering of software systems and the management of the software process. It includes platform, system software, utilities, and CASE tools installed. A service is an abstract description of support for activities and tasks for the improvement of issues such as productivity, quality, or performance. A service can be assisted by CASE tools.

A service is self contained, coherent, discrete, and maybe composed other services. A CASE Tool is a software product that can assist software engineers by providing automated support for software life-cycle activities as defined in [5]. Finally, an automated process is a software process that its performed either fully or partially supported by CASE tools.

The term SEE may cover several situations; from the mere juxtaposition of a few tools running on the same operating system, up to the fully integrated environment, able to handle, monitor, and even control all the data, processes, and activities in the software life cycle. A SEE provides support to human activities through a series of services that describe the capabilities of the environment. The software process supported by a SEE becomes an assisted or automated software process. This standard describes
SEE services and relates them to [5] in a manner applicable to a range of organisations. In defining a lifecycle process an organisation needs to find the appropriate level of automation. This may result in establishing a new SEE or improving an existing one. Expressing the capabilities of the SEE via services emphasises the fact that an individual performs some activities with the help of the SEE. Services provide a link between a set of chosen software life cycle processes and their automation through tools. In most cases a tool’s functionality can be related to one or more services.

Through the automation of activities, either partially or fully, the SEE provides benefits to an organisation through reduced cost (higher productivity), improved management and from the higher product quality that can result. For example, the automation of repetitive activities such as the execution of test cases provides not only productivity gains, but can also help to ensure completeness and consistency in the testing activities.

Activities that are outside the scope of project 15940 are those specific to criteria or process used in the selection of one or more CASE tools, or recommendations to adopt tools within an organization. These criteria and a detailed CASE tool characteristics can be found in [7] and [8], currently under revision by ISO/IEC JTC1 SC7 Software and Systems Engineering Subcommittee.

Upcoming ISO/IEC 15940 shall provide a reference model for SEE services. As a reference model, ISO/IEC 15940 will make use of a set of conceptual descriptions to describe each service used in a project support environment. The “conceptual description” indicates that the description is from a reference viewpoint, and does not deal with any specific implementation. The description is therefore general and does assume any specific application domain, life cycle model, or tool in a project. In this way, ISO/IEC 15940 will be able to be applied to any defined organisational environment.

An actual environment is one that is realised from a reference model containing conceptual descriptions. Therefore, an actual description of a specific environment would reflect a particular activity with its tools and standards. In the current draft CD ISO/IEC 15940 services are grouped into six categories that reflect broad functional activities within a typical software engineering organisation. The six categories of services are:

- Technical engineering: Technical Engineering Services support activities related to the specification, design, implementation, testing, and maintenance of software.
- Technical management: The services in this section fall into a category that considers both Technical Engineering and Project Management. These services pertain to activities that are often shared by engineers and managers.
- Project management: The services in this section support the activities related to planning and executing a project. Following project initiation, detailed planning of the project activities will be necessary, together with ongoing monitoring and re-planning of the project to ensure its continued progress.
- Process management: The services in this section support projects in achieving discipline, control, and clear understanding in their life-cycle development processes as understood in ISO/IEC 12207 and individual process steps.
- Support: Support services include services that the rest of the services will require to become operational. They generally include those services associated with processing, formatting, and disseminating human-readable data.
- Framework: These services comprise the infrastructure of a SEE and will be required to support a SEE actually implemented.

### 3 INCOSE introduction to measurement process: Primer

process and some guidance for developing and implementing a measurement program. Primer is focused in two main aspects: the measurement process and its supporting infrastructure and the purposes of measurement.

The measurement process is divided into five activities:

- **Selection and Specification of Measures / Indicators**: this activity leads to the creation of a measurement plan. During the development of this plan, a set of measures are defined in order to provide the best indicators and insights for the least cost.

- **Data Collection Method**: this activity is oriented to define a repeatable method for collecting data. This method must identify the points in time when the data will be collected, what tools will be used to accomplish collection, the people responsible for collecting the data, how the data will be validated, and what is done with the data once it is collected.

- **Calculation Method**: this activity transforms the measures to indicators. In most cases, indicators are a mathematical combination of measures. The method for transforming measures to indicators must be defined and documented.

- **Analysis of the Measures or Indicators**: this activity defines and performs the tasks for measurement interpretation. This analysis is based on the stored historic measurement data, together with program attribute data. These form the basis for predictive models that should be used to estimate cost, schedule, and quality elements of the project or product.

- **Reporting and Using the Results**: the use of results is the most important part of the measurement process. The measurement analysis must be fed-back into the measurement process with a description of how results must be used.

The measurement program supporting infrastructure covers the following topics:

- **Management Commitment**: one of the most important aspects of a measurement plant is the level of support that is provided by management. This commitment must be both visible and well communicated.

- **Measurement Plan**: this is the document used to implement and guide the measurement program. The measurement plan covers the following topics: issues and measures selected, identification and definition of data elements to be collected, data collection details, data delivery details, communication process and analysis and reporting details.

- **Resources**: all tasks of measurement plan should be identified and scheduled. An estimation of effort and resource should be performed across the entire lifecycle.

- **Training**: all stakeholders must have the appropriate level of training. Training is an important part of the measurement plan. The stakeholders must understand the purpose of the measurement program and measures selected.

- **Tools**: a key for measurement program success is the usage of tools. Tools can automatically support part of the data collection, analysis and reporting.

- **Measurement Data Repository**: data storage and retrieval support must be part of the measurement plan. Storage and retrieval can be performed by a database management system or manual repository if records.

The purposes of measurement are the following:

- **Characterization**: gain understanding of products and processes. This understanding is achieved by means of measuring technical performance and measuring process performance.
- Improvement: identifying and evaluating process and product improvement opportunities. Opportunities for improvement are identified by analyzing actual measured process, product, or project attributes against target values and business objectives.

- Prediction: facilitating projections and planning. The availability of historical data is essential for improving projections and planning. Statistical and causal models of prediction can be developed using the historical data.

- Evaluation: providing feedback and status. Measures can be a good form of feedback for team and customer. Measures are an effective status reporting tool as well, particularly when presented in graphical form.

4 Mapping INCOSE Measurement Tools Survey into ISO/IEC 15940

INCOSE measurement concepts described in section 3 are used for developing a questionnaire for evaluating measurement tools: the INCOSE Measurement Tools Survey. In this questionnaire, the different tasks of the measurement process than can be automated are in the first column and the second column is for tools vendor response in terms of “full”, “partial” or “none”. The mapping has been performed from tool measurements required features into SEE services; that is, for each tool feature, presented as a question, the answer will be the SEE service or services that contain the corresponding task.

<table>
<thead>
<tr>
<th>Measurement Objectives (ref. Primer Section 4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1 Can the tool monitor and control ongoing product and services? (ref. Primer section 4.1.1)</td>
</tr>
<tr>
<td>- Project monitoring and scheduling service</td>
</tr>
<tr>
<td>- Project evaluation service</td>
</tr>
<tr>
<td>1.2 Can the tool quantify process improvements? (ref. Primer section 4.1.2)</td>
</tr>
<tr>
<td>- Process improvement support service</td>
</tr>
<tr>
<td>1.3 Does the tool support estimation and prediction for project planning? (ref. Primer section 4.1.3)</td>
</tr>
<tr>
<td>- Project Estimation Service;</td>
</tr>
<tr>
<td>- Project Risk management Service;</td>
</tr>
<tr>
<td>1.4 Does the tool support distribution of measurement information for organizational status and feedback? (ref. Primer section 4.1.4)</td>
</tr>
<tr>
<td>- SEE cooperative work support service</td>
</tr>
<tr>
<td>- Process documentation service</td>
</tr>
<tr>
<td>1.5 Does the tool allow measurement to be tailored for specific organizational units (OU)? [i.e. projects, departments, programs, etc...]</td>
</tr>
<tr>
<td>- Project process usage service</td>
</tr>
<tr>
<td>1.6 Does the tool provide a mechanism for establishing &quot;core&quot; measures?</td>
</tr>
<tr>
<td>- Project process usage service</td>
</tr>
</tbody>
</table>

SEE Services
## Measurement Process

### 2.1 Selection and Specification of Measures / Indicators (ref. Primer Section 3.1.1)

<table>
<thead>
<tr>
<th>Question</th>
<th>Service(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Does the tool support the definition of measures and indicators for each OU being measured? Provides a mechanism to specify attributes of measures?</td>
<td>Process improvement support service</td>
</tr>
<tr>
<td>Does the tool support the identification of project issues and measurement goals?</td>
<td>Project Estimation Service; Project monitoring and scheduling Service; Project Evaluation Service.</td>
</tr>
<tr>
<td>Does the tool allow the specification of measurement characteristics, including data source, data validity rules, collection method, temporal constraints and interpretive guidelines?</td>
<td>Measurement and analysis service; Project monitoring and scheduling Service; Project Evaluation Service.</td>
</tr>
</tbody>
</table>

### 2.2 Data Collection Method (ref. Primer Section 3.1.2)

<table>
<thead>
<tr>
<th>Question</th>
<th>Service(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Does the tool perform validation of collected measurement data?</td>
<td>Measurement and analysis service</td>
</tr>
<tr>
<td>Does the tool provide a means for manual data entry?</td>
<td>Measurement and analysis service</td>
</tr>
<tr>
<td>Does the tool provide a means for automated data collection?</td>
<td>Measurement and analysis service</td>
</tr>
<tr>
<td>Does the tool provide a mechanism for the user to tailor the tool for automated collection from user-defined data sources?</td>
<td>Measurement and analysis service</td>
</tr>
</tbody>
</table>

### 2.3 Calculation Method: Getting from Measures to Indicators (ref. Primer Section 3.1.3)

<table>
<thead>
<tr>
<th>Question</th>
<th>Service(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Does the tool allow measurement data to be combined into complex measures or indicators?</td>
<td>Measurement and analysis service</td>
</tr>
<tr>
<td>Does the tool allow specification of weights or normalization factors to be used in combining measures into indicators?</td>
<td>Measurement and analysis service</td>
</tr>
</tbody>
</table>
## 2.4 Analysis of the Measures or Indicators (ref. Primer Section 3.1.4)

<table>
<thead>
<tr>
<th>2.4.1</th>
<th>Does the tool support the review and analysis of measurement data?</th>
<th>Measurement and analysis service</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.4.2</td>
<td>Does the tool support statistical process control calculations including control charts and upper, lower and center line calculation?</td>
<td>Measurement and analysis service</td>
</tr>
<tr>
<td>2.4.3</td>
<td>Does the tool provide a means for comparing actual measured data to planned, target or historical values?</td>
<td>Measurement and analysis service</td>
</tr>
</tbody>
</table>

## 2.5 Reporting and Using the Results (ref. Primer Section 3.1.5)

<table>
<thead>
<tr>
<th>2.5.1</th>
<th>Does the tool present measurement data consistent with the needs of differing organizational or management levels?</th>
<th>Measurement and analysis service</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.5.2</td>
<td>Present measurement data appropriate for the role of the user accessing the tool, where roles are analyst, OU manager or customer?</td>
<td>Measurement and analysis service</td>
</tr>
<tr>
<td>2.5.3</td>
<td>Does the tool allow the user to customize the generation of reports, including adding, modifying and deleting reports?</td>
<td>Measurement and analysis service</td>
</tr>
<tr>
<td>2.5.4</td>
<td>Does the tool provide a mechanism for distributing measurement reports?</td>
<td>SEE cooperative work support service Process documentation service</td>
</tr>
<tr>
<td>2.5.5</td>
<td>Does the tool support the distribution of measurement reports via the Internet or Intranet?</td>
<td>SEE user communication support service SEE basic communication service</td>
</tr>
</tbody>
</table>

## 3 Measurement Program Infrastructure

<table>
<thead>
<tr>
<th>3.1</th>
<th>Does the tool allow identification and tracking of management commitment to the measurement process? (ref. Primer Section 3.2.1)</th>
<th>Measurement and analysis service</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.2</td>
<td>Does the tool provide automation for a measurement process plan? (ref. Primer Section 3.2.2)</td>
<td>Measurement and analysis service</td>
</tr>
<tr>
<td>3.3</td>
<td>Does the vendor identify the resources required to setup, operate and maintain the tool? (ref. Primer Section 3.2.3)</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>Section</td>
<td>Question</td>
<td>Services</td>
</tr>
<tr>
<td>---------</td>
<td>---------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| 3.4     | Does the vendor provide training for all aspects of tool? (ref. Primer Section 3.2.4) | - Measurement and analysis service  
- SEE policy enforcement service  
- Process initiation service  
- Project evaluation service |
| 3.5     | Does the tool provide a repository for capturing organizational measurement process best practices? (ref. Primer Section 3.2.6) | - SEE Repository management service  
- Measurement and analysis service  
- SEE information/data mining service  
- SEE repository service |
| 4       | **Computing Support**                                                    |                                                                         |
| 4.1     | **Measurement Database**                                                 |                                                                         |
| 4.1.1   | Does the tool provide import capabilities from popular file formats and tool interfaces? Which ones? | - SEE common support service |
| 4.1.2   | Does the tool provide export capabilities to popular file formats and tool interfaces? Which ones? | - SEE common support service |
| 4.1.3   | Does the tool support access to measurement repository by other commercial tools (i.e. for data mining, statistical analysis tools, reporting tools …) Which ones? | - SEE infrastructure management service  
- SEE basic communication service |
| 4.2     | The tool provides a security mechanism for restricting access to the repository data? | - Configuration management service  
- SEE repository management service  
- Documentation service  
- Process library service  
- SEE administration service  
- SEE policy enforcement service |
5 System Environment

5.1 Support a single user or multiple concurrent users?
- SEE cooperative work support service

5.2 Which platforms and operating systems does the tool run on?
- SEE Operating System service

5.3 Does the tool use a proprietary or commercially available database?
- SEE infrastructure management service
- SEE basic communication service

5.4 Please identify hardware/software configuration requirements: Not Applicable

This table can be used in two ways, at least. From left to right it provides information on which SEE services may be required to support each automated measurement feature. From right to left it provides a good description of what a SEE measurement service should make available to engineers, that is a set of requirements for a measurement service and the interaction of this service with others such as SEE policy enforcement service or SEE project evaluation service. A better understanding of this second issue can be achieved if the second column is order by service.

5 Summary and next steps

It is commonly recognized that software engineering practice, to achieve a good performance, depends on measurements. While the automation of a number of software engineering activities is receiving a lot of attention, it is not the case for measurement. This paper provides a first approach to provide a set of requirements for a software engineering environment measurement service, taking as a basis on-going ISO work on software engineering environments services and INCOSE measurement tools survey criteria. More specifically CD ISO·IEC 15940 Software Engineering Environment Services and the INCOSE tool survey have been mapped.

This study provides an approach to understand which SEE services may be required to support each automated measurement feature, and, as important, it provides a good description of what a SEE measurement service should make available to engineers, that is, a set of requirements for a SEE measurement service and the interaction of this service with others such as SEE policy enforcement service or SEE project evaluation service.

This paper also shows that mapping a set of tool measurement criteria devised in terms of systems onto a set of services defined in terms software needs has been easy.

The coming steps would be to define a measurement service according to this paper, to analyze the obtained results in terms of other existing publications, and to formally define the interaction of those services involved in measurement. A detailed description of interactions can be also useful to improve practical tool integration issues.
References


