

Software Modeling of S-Metrics Visualizer: Synergetic Interactive Metrics Visualization Tool

Sergiu M. Dascalu¹, Norm Brown², Derek A. Eiler¹, Herman W. Leong¹,
Nathan A. Penrod¹, Brian T. Westphal¹, and Yaakov L. Varol¹

¹ Department of Computer Science and Engineering
University of Nevada, Reno
Reno, NV, 89557, USA

² National Institute for Systems Test and Productivity
University of South Florida
Tampa, FL, 33620, USA

Abstract *S-Metrics Visualizer is a software development metrics acquisition and visualization tool for Windows. Throughout the software development process managers must be aware of the current status, timeline, risk, and resource demands of their projects. Software products commonly end up requiring more time and money than initially expected and, even with the amount of expended resources, projects often do not meet their initial requirements specifications. Looking back after the development is completed, one can often attribute the slippage in delivery expectations to a cascade of seemingly small problems that occur throughout the project's evolution. Software metrics help managers to better monitor the progress of their software products, making them more capable to actively respond to changing conditions and dynamic workplace environments. This paper presents aspects of the software modeling of S-Metrics Visualizer, a tool used to assist project managers to deliver their products on time and within budget.*

Keywords: software metrics, metrics visualization, requirements specification, use case modeling.

1 Introduction

Throughout the software development process it is important for a project manager to know how each component of the project he or she supervises is progressing. A project manager must know if each part of the project is on schedule, will likely be delivered on schedule and within cost, if it meets critical software functionality requirements, if individual components of the code are integrating properly, and if defect reduction, memory utilization, and

other factors related to the size and complexity of the software being built fit in desired ranges.

In order to help software managers monitor the development process, software metrics can be defined, updated and checked for each critical software item. Software metrics include data and statistics that meaningfully reflect the current state of development. Software metrics are valuable tools and the identification and definition of new and more effective metrics is an active area of research and development [1, 2, 3, 4, 5]. Current metrics may include information on the functional delivery date, test progress, integration process, milestone progress, defect resolution, defect aging, and many other project-related aspects [6, 7].

S-Metrics are a new class of software metrics focused on providing insight into the status and likelihood outcome of a software development project from a management perspective¹. S-Metrics provide insight into bottom-line issues of schedule, cost, and risk of large-scale software development projects, and utilize newly defined metrics (e.g., integration progress, functional availability, staffing) combined with more traditional software

¹ S-Metrics are a joint project of the Computer Science Departments of the University of South Florida and the University of Nevada at Reno under the auspices of the National Institute for Systems Test and Productivity with funding provided by the Naval Space and Warfare Systems Command.

development metrics (e.g., defect closure rate, memory utilization, earned-value cost and schedule performance) to provide enhanced management insight in a timely manner thus enabling management to anticipate problems and take timely corrective or remedial actions.²

Excel-based metrics collection and PowerPoint presentation to management are the typical currency of the software acquisition realm; and typically the display conversion from one to the other caused delay of information available to management since the conversion is manually intensive and time consuming (particularly with large projects having many critical software items). Thus, the S-Metrics Visualizer tool, introduced in this paper, was created to automate this transition.

The tool presented in this paper, S-Metrics Visualizer, is designed to support the daily work of software managers. Its goal is to provide a simple and easy to use interface for the management of the software development process. Using predefined and user defined visualization objects, users of the software are able to acquire data and enter it into regular Microsoft Excel spreadsheets [8]. Spreadsheet data can be processed through S-Metrics Visualizer to produce visual representations of the data underlying the associated metrics, typically in the form of graphs [9], which can be imported into Microsoft PowerPoint [8]. The S-Metrics Visualizer tool allows one to easily utilize software metrics data and create visually compelling presentations depicting the progress of the associated software development project.

S-Metrics Visualizer provides a graphical user interface that facilitates management metrics insight and supports the visualization process. The interface is made up of several components, each of which having a clearly defined, specific

function. The first such component, *Data Entry*, has a Microsoft Excel editor component embedded into it, providing easy access to metrics data contained in Excel spreadsheets. The second component, *Configuration*, allows the user to select and configure desired output graphs (visualizations). Finally, S-Metrics Visualizer contains a *Presentation Organizer* component that allows previewing of the PowerPoint presentation that will be generated, as well as simple modifications of the presentation.

S-Metrics Visualizer is extensible; the initially-defined S-metrics visualization objects are built as modules integrated in to the program. These metrics definitions can be dynamically changed, allowing different management metrics to be utilized as desired for any given development project. If a new metric has been defined, a new Visualizer template is created and inserted it into the Visualizer program. Accordingly, the Visualizer software is flexible enough to accommodate new user-defined metrics descriptions.

This paper, in its remaining part, is structured as follows: Section 2 provides details of the tool's requirements specification, Section 3 includes excerpts of S-Metrics use case modeling, Section 4 covers several aspects of high and low-level design, Section 5 reports on S-Metrics' current status, Section 6 points to several directions of future work, and Section 7 contains our conclusion.

2 Requirements Specification

Prior to beginning the implementation of S-Metrics Visualizer, we collected and prioritized a list of requirements [10] that our software management metrics tool should meet.

2.1 Primary Requirements

The following list represents the most significant requirements that we have identified for the S-Metrics visualization software tool.

S-Metrics Visualizer shall:

² S-Metrics are being currently applied at a major Department of Defense software acquisition activity. Further development and refinement of these metrics is ongoing. A complete description of S-Metrics will be published after development has completed its initial phase.

- R01 Accept a Microsoft Excel document as input.
- R02 Allow the user to generate Microsoft PowerPoint slides as output.
- R03 Generate meaningful visualizations of specified software metrics data.
- R04 Allow the user to generate JPEG images of visualizations.
- R05 Provide an interface for using third-party visualization modules.
- R06 Provide an interface for configuration of each visualization type.
- R07 Provide a mechanism for saving visualization settings.
- R08 Provide a mechanism for loading visualization settings.
- R09 Follow a Microsoft Excel format template document for accepting input.

2.2 Secondary Requirements

Additional requirements for features chosen to enhance the functionality of S-Metrics Visualizer are as follows:

- R10 Accept Comma Separated Value (CSV) documents as input.
- R11 Combine input from multiple Microsoft Excel documents.
- R12 Provide an interface for rearranging slides in the Microsoft PowerPoint output document.
- R13 Generate a Microsoft PowerPoint presentation consisting of multiple slides with no user intervention.
- R14 Allow the user to modify the input Microsoft Excel document while S-Metrics Visualizer is running.

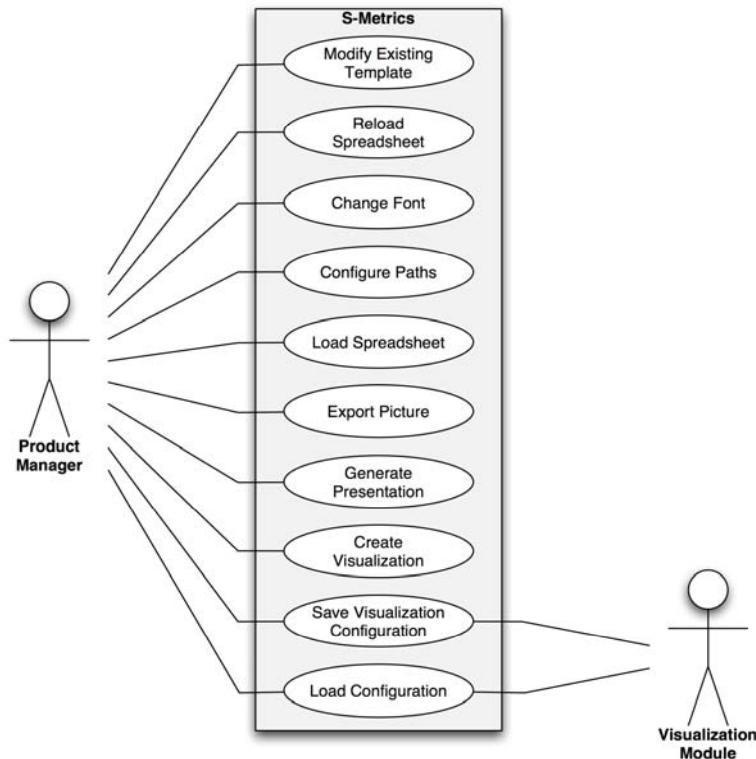


Fig. 1 The S-Metrics Visualizer Use Case Diagram

3 Use Case Modeling

In order to make the requirements for the S-Metrics Visualizer tool more concrete, several use cases and scenarios were developed. These helped manage the level of abstraction that we were dealing with and clarify some software requirements issues from a functional point-of-view. Figure 1 shows the use case diagram of the S-Metrics Visualizer tool.

Figures 2 and 3 present examples of the use cases that we elaborated in order to more completely and precisely describe the tool's functional requirements.

Use Case: Configure Paths
ID: UC4
Actors: Project Manager
Precondition: 1. Path previously set
Flow of events: 1. The use case starts when the user selects the configure paths option. 2. The user specifies a new path from a dialog. 3. The path is verified to exist and contain modules. a. If the path is not valid i. Ask for a new path 4. The path is accepted. 5. New modules are read and added to the visualizations.
Postcondition: 1. The new modules are available for use.

Fig. 2 Configure Paths Use Case

Use Case: Save Visualization Configuration
ID: UC4
Actors: Project Manager
Precondition: 1. Project manager has loaded an Excel document for input and has configured a visualization.
Flow of events: 1. The project manager selects "Save" visualization configuration" from the "Visualization" menu. 2. The operating system's standard file selection dialog opens. 3. The project manager selects a filename and location in which to save the configuration data. 4. If the project manager selects "OK" a. The configuration module provides a list of the configured configuration options and values. b. The configuration data is saved to the specified file. 5. If the project manager selects "Cancel" a. The file selection dialog closes and control is returned to the main window.
Postcondition: 1. The currently configured visualization settings are written to a file.
Alternative Flow 1: 1. The currently configured visualization settings are written to a file.

Fig. 3 Save Visualization Configuration Use Case

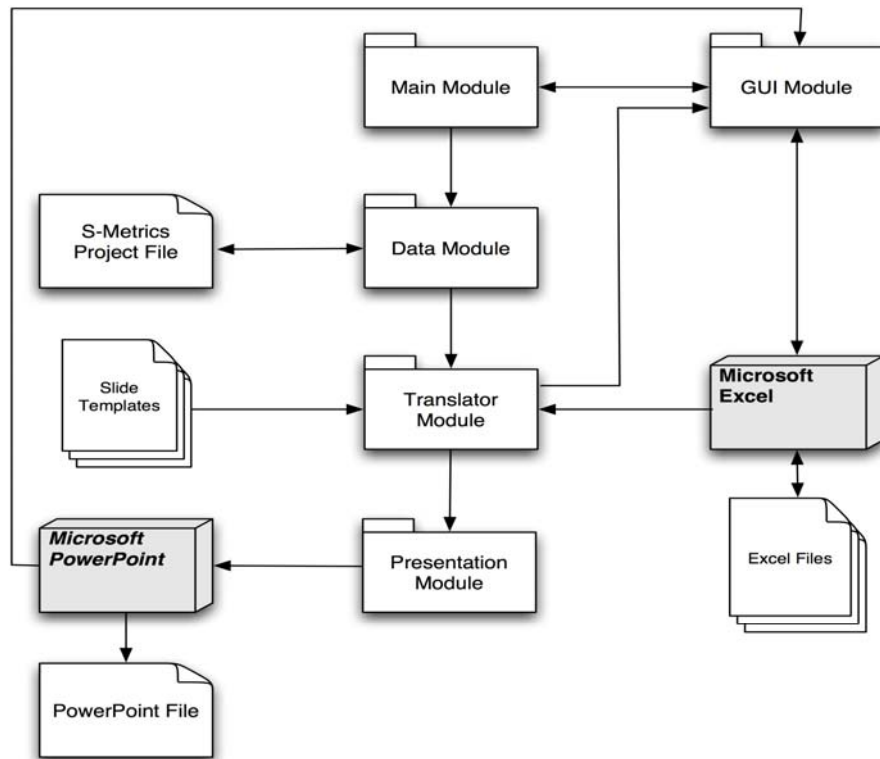


Fig. 4 System-Level Diagram for S-Metrics Visualizer

4 Design Aspects

The S-Metrics Visualizer tool is divided into five main components: GUI, Main Module, Data Module, Translation Module, and Presentation Module. The relationships between these modules are shown in Figure 4.

The S-Metrics Visualizer software has been designed in detail at lower level of abstraction using the UML notation [11] and guidelines presented in [12]. An example of lower-level design is provided by the activity diagram shown in Figure 5, which describes the initialization process of S-Metrics Visualizer.

More complete information on both the specification and the design of S-Metrics Visualizer can be obtained from the authors via the initial website of the project [13], soon to be modified and expanded (note that the project had a slightly different title in its initial stages).

5 Current Status

S-Metrics Visualizer has recently completed its initial implementation and testing phases. An operational version of the tool is currently undergoing additional testing through use by Dr. Norm Brown and his colleagues at the National Institute for Systems Test and Productivity at the University of Central Florida.

Figure 6 shows a screenshot of the main portion of the S-Metrics Visualizer program. The Main Window of S-Metrics Visualizer contains sections for configuring a presentation (top left); modifying Excel spreadsheets (bottom); and, previewing PowerPoint presentations (top right).

The configuration for a presentation consists of a tree-view that contains attributes for selected modules. Updates to attributes are shown in this area. The Spreadsheet Library contains open Excel documents for easy access. Choosing any spreadsheet within this area automatically brings

up the corresponding sheet. The Presentation Preview area displays a project in progress. Any presentation may be saved for later access.

6 Future Work

This project offers significant, challenging, exciting, and potentially very rewarding opportunities for future work. Our current goals for future work include packaging additional pre-built visualization modules with S-Metrics Visualizer, providing a graphical interface for defining new visualization modules, and supporting heterogeneous data sources for input, e.g. SQL databases. Besides generating PowerPoint files, a future version of S-Metrics Visualizer will be able to generate Macromedia Flash presentations and HTML web pages.

7 Conclusions

The S-Metrics Visualizer software development metrics acquisition and visualization tool

provides software project managers with a convenient means of handling software metrics.

The integration of Microsoft Excel and PowerPoint components allows users to make use of standard forms of data in order to track and present metrics of project progress and resource utilization, and facilitates more effective management of software development projects.

Although S-Metrics Visualizer tool currently comes with several built-in types of S-metrics and associated specified visualizations, the system was designed from the ground up for flexibility and adaptability. The templates that define the metrics and their visualizations are imported dynamically so that as new types of metrics and visualizations are developed, their definitions and visualization specifications can be readily and simply incorporated into the S-Metrics Visualizer.

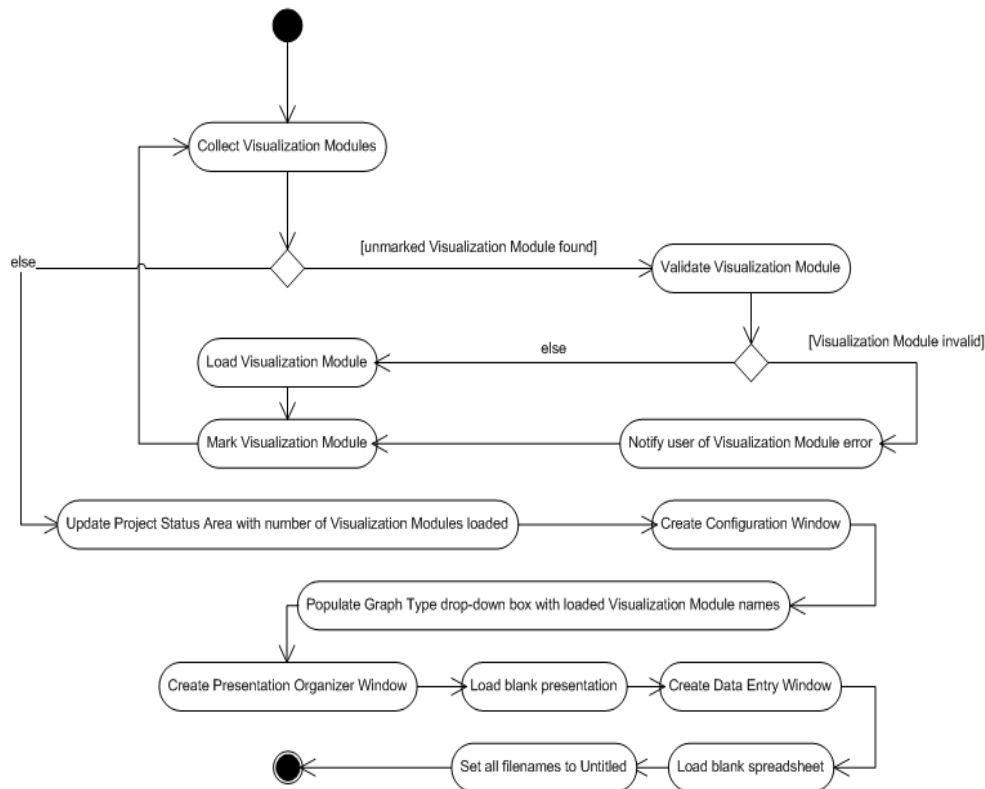


Fig. 5 S-Metrics Visualizer Initialization Process Activity Diagram

References

- [1] Kitchenham, B.A., Hughes, R.T., and S. G. Linkman, "Modeling Software Measurement Data," *IEEE Trans. on Software Engineering*, 27(9), 2001, pp. 708-884.
- [2] Gopal, A., Krishnan, M.S., Mukhopadhyay, T., and D.R. Goldenson, "Measurement Programs in Software Development: Determinants of Success," *IEEE Trans. on Software Engineering*, 28(9), 2002, pp. 863-875.
- [3] Irwin W. and N. Churcher, "Object-Oriented Metrics: Precision Tools and Configurable Visualizations," *Proceedings of the 9th IEEE International Software Metrics Symposium (METRICS'03)*, pp. 112-123.
- [4] Kilpi, T., "Implementing a Software Metrics Program at NOKIA," *IEEE Software*, November/December 2001, pp. 72-77.
- [5] Kulik, P., "A Practical Approach to Software Metrics," *IEEE IT Pro*, January/February 2000, pp. 38-42.
- [6] Menzies, T., Di Stefano, J.S., Chapman, M., and K. McGill, "Metrics That Matter," *Proceedings of the 27th Annual NASA Goddard/IEEE Software Engineering Workshop (SWE-02)*, pp. 51-57.
- [7] Haynes, S.R., "Institutional Metrics for the United States Marine Corps," *Proceedings of the 36th Hawaii International Conference on Systems Sciences (HICSS-03)*, January 2003, 10 pages (on CD).
- [8] Microsoft Corp. *How To Embed and Automate Office Documents with Visual Basic*. Accessed at <http://support.microsoft.com/?kbid=242243> (May 2, 2005).
- [9] Microsoft's MSDN Library. *Working with Shapes (Drawing Objects)*. Accessed May 2, 2005 via <http://msdn.microsoft.com/library/>
- [10] Sommerville, I., *Software Engineering*, 7th Edition. Addison-Wesley, 2004.
- [11] OMG's *UML Resource Page*, accessed May 2, 2005 at <http://www.omg.org/uml/>
- [12] Arlow, J., and I. Neustadt, *UML and the Unified Process: Practical Object-Oriented Analysis and Design*, Addison-Wesley, 2002.
- [13] Team 04 *Project Website*, available as of May 2, 2005 at: www.cse.unr.edu/~derek/sp2005/

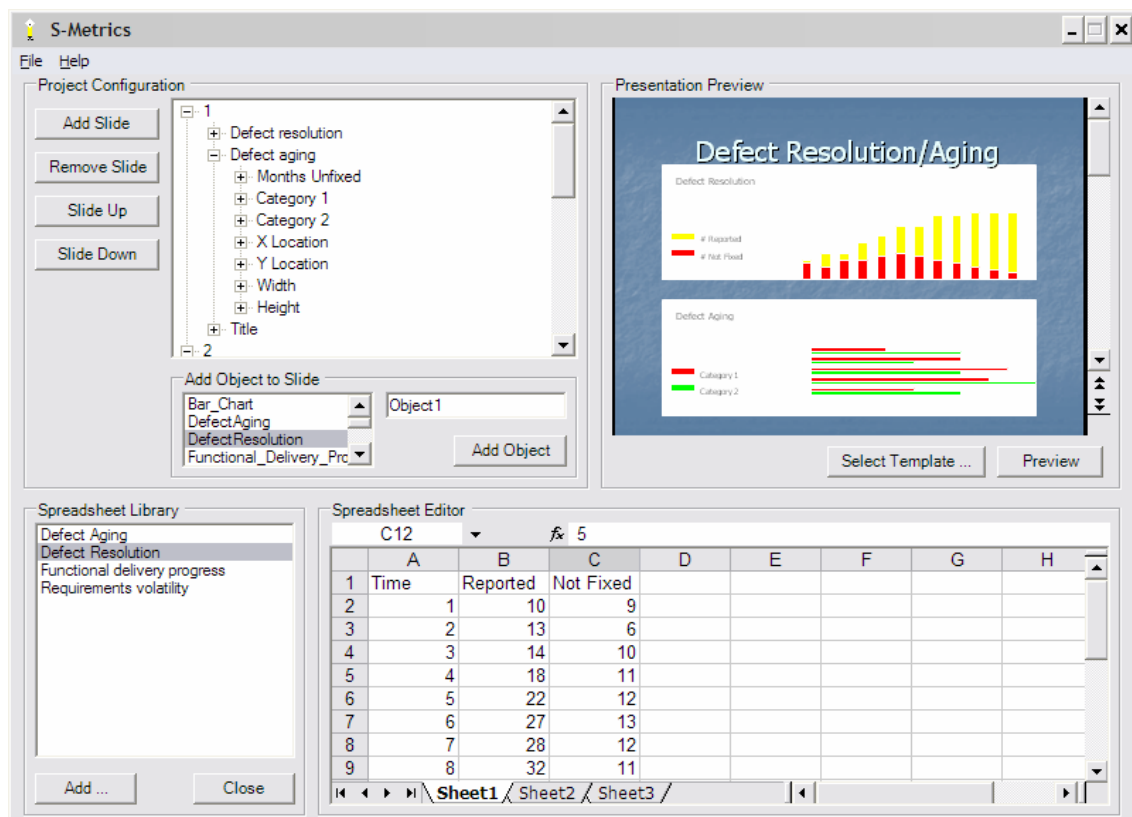


Fig. 6 The Main S-Metrics Visualizer Window