

60 years of Software Maintenance: Lessons Learned

Nicholas Zvegintzov
Software Management Network
zvegint@softwaremanagement.com

Girish Parikh
*Computer SOFTWARE MAINTENANCE
Network*
girish116@yahoo.com

Abstract

Ever since there has been software there has been software maintenance. Nothing in theory or practice indicates that this will change. This Tutorial presents the historical realities of software maintenance – what it is, what it does, how it is done, and what it will be.

1: The history of software: 1945-2005

The very earliest observations of software envisioned its continual growth and change – via development, enhancement, correction, integration, and redevelopment. For 60 years these dynamics have remained in force, and software continues to grow under their influence.

A time-line of software

We look at the history of software, building a perspective of how and why software has grown.

- The functions of software – applications, domains, capabilities. Snapshots of software functions over time.
- The platforms of software – devices, delivery vehicles, networks, embedding environments. Examples of software platforms over time.
- The technology of software – languages, operating systems, programming environments, support technology. Benchmarks of software technology over time.
- The amount of software – per program, per device, per system, per organization. Benchmarks of software size over time.
- Software professionals – programmers, analysts, documenters, testers, support personnel, operations personnel, managers. Snapshots of software professionals over time.
- Software effort – hours, roles, teams, interactions. Observations of software effort over time.

2: The nature of software change

We examine why software needs “software maintenance”. What this phrase means. Why it is misleading.

The demands that cause software change are:

- Requirements for functional change (“enhancement”)
- Failures and errors (“correction”)
- Technology changes (“adaptation”)
- User help and support (“support”)
- New developments.

We examine the respective contribution of these demands to the total software work.

3: The history of software maintenance

We build up a historical time-line of software maintenance in the context of the history of software by tracing the people, events, controversies, publications, organizations, research, and standards that have marked software maintenance.

The early pioneers of software maintenance

This session includes glimpses of the lives and works of the following pioneers: Ned Chapin, Robert Glass, Carma McClure, J Cris Miller, Girish Parikh, Tom Pigoski, Paul C. Tinnirello, Jean Dominique Warnier, Gerald M. Weinberg, Nicholas Zvegintzov, and others.

Books (a selection)

This partial list of books relating to software maintenance is given in publication order to indicate landmarks in the understanding and description of software maintenance. More books and articles appear in the bibliography of the Tutorial materials.

Jean-Dominique Warnier. *La transformation des programmes*, 1975. First book on software maintenance. *Program modification*, 1978. English edition.

Richard C. Linger, Harlan D. Mills, Bernard I. Witt. *Structured programming: Theory and practice*, 1979. The classic on reading, writing, and establishing the function of structured programs.

Girish Parikh, editor. *Techniques of program and system maintenance*, 1980, 1st edition, 1982. 1st edition reprint. First American book on software maintenance. 1988. 2nd edition, revised and enlarged.

Bennet P. Lientz, E. Burton Swanson. *Software maintenance management - a study of the maintenance of computer application software in 487 data processing organizations*, 1980.

The original data on maintenance practices and management attitudes, validated by many later studies.

Girish Parikh and Nicholas Zvegintzov, editors. *Tutorial on software maintenance*, 1982. Readings and reprints, with historical context and biographies.

Nachum Dershowitz. *The evolution of programs*, 1983. The first book to face the central problem of software maintenance: What does it *mean* to modify a software system?

James Martin, Carma L. McClure. *Software maintenance: The problem and its solutions*, 1983. A book co-authored by industry guru James Martin with “maintenance” in the title adds legitimacy to the topic.

M. M. Lehman, Laszlo A. Belady. *Program evolution - Processes of software change*, 1985. Papers collected by the two originators of the observable laws of program evolution dynamics.

Thomas M. Pigoski. *Practical software maintenance: Best practices for managing your software investment*, 1997. The support and enhancement of software systems from their initial design and development through their useful lifetimes.

Michael C. Feathers. *Working effectively with legacy code*, 2005. Little by little, “how to” books begin to appear.

Conferences

IEEE International Conference on Software Maintenance. Peer-reviewed papers and industrial presentations.

SMA Conference.

Organizations

Software Maintenance Association.

Periodicals

Software Maintenance News.

J. Software Maintenance: Research and Practice.

Research

Research in software maintenance is performed in Universities and presented as papers at peer-reviewed conferences such as ICSM and at workshops on software understanding and modification, and has been supported by several multinational academic-industrial projects. Topics of research include:

- Observation of actual software and actual software change.
- Frameworks and architectures for modification and evolution.
- Empirical study of methods used in analysis and change.
- The basis and prerequisites for modification.
- Testing as a basis for change, as a component of change, and as validation of change.

Standards

Military Handbook MIL-HDBK-347. *Mission-Critical Computer Resources Software Support*, 1990.

IEEE Standard 1219-1993. *IEEE Standard for Software Maintenance*, 1993.

4: Crises and controversies in software maintenance

Software maintenance is often unseen and unreported, but some issues and events have become notorious:

- Software disasters and how they happened
- The “Year 2000”
- The controversy over testing and correctness
- The growth of support and service management.

5: The future of software maintenance

The “end” of maintenance is often announced and promised as new technologies and new developments are promoted. But they offer ever greater problems of complexity and fallibility, and ever wider opportunities for incremental enhancement. The future of software is still the future of software maintenance.

The Software Maintenance History Foundation

The foundation can be named “SSME Heritage”, and its motto can be: “For Now and the Future!” SSME is the acronym for Software Support, Management (or Maintenance), and Evolution.

Materials

Tutorial materials include lecture notes, photographs, diagrams, and facsimiles in PowerPoint form, and additional printed material in PDF/DOC form including in depth interviews: with Ned Chapin, spanning 50 years of maintenance performance and management; with Harry M. Sneed, the General-Chair of ICSM 2005, on ICSM; with J. Cris Miller on Software Re-Engineering, Reverse Engineering, and Software Maintenance; and other selected “Journals of Girish Parikh” (JOGP) columns on software maintenance history posted on the web site of the Yahoogroup CSMN (Computer SOFTWARE MAINTENANCE Network). The materials are also available outside the Tutorial.

The Exhibition

Accompanying the Tutorial are exhibits of the rich history of maintenance, including first and autographed editions, original magazines, conference brochures, tool descriptions, manuals, photos, and other artifacts.