

Abstract.

In Software Engineering (and also in Computer Science in general), it has been frequently to dedicate much more attention to the problems related with development of new products than to those related with its later maintenance. In the last years, this lack of general attention toward the maintenance has changed lightly due to events so significant as the Y2K or the Euro effects. The change of tendency has not been a fleeting fashion (like it happened with the mentioned events), but rather it has been confirmed the necessity to approach the specific problems that happen when maintenance is carry out and that don't happen when development is carry out, or that have different characteristics and, in consequence, the solutions cannot be the same ones. This new importance assigned to the maintenance is in order to consider better the reality: all studies confirm that the maintenance costs are more than two thirds (and in any cases more than 80%) of the total costs of a software product life cycle. It is also well-known that the cost of correcting an error during the maintenance stage is, approximately, one hundred times the cost that during the requirements initial stage.

In this thesis, a proposal is presented that seeks to help organizations to improve the way they carry out the software maintenance impacting in the project management improvement. The main point of view of this proposal is to approach the maintenance problem from a “business process” perspective, that is to say, integrating software engineering with organizational and management aspects. The idea is to advance toward the consideration of the software maintenance like a key factor for the competitiveness of the organizations because, like several researchers have pointed out, this work is the core part of the current information systems change.

The proposal is based on a new concept, named "extended Software Engineering Environment" (extended SEE), that integrates and extends the traditional concepts of methodology (shortly, a collection of related methods) and software engineering environment (a collection of technically integrated tools to automate the software engineering processes). In this work, it has been considered that an “extended EIS” is a collection of conceptual, methodological and technical (software) tools whose purpose is to be able to approach the software processes (development or maintenance) from a global perspective of business process.

The elaborated proposal considers that the software projects management (in this case, of maintenance) is a multidimensional problem with multiple faces. For this reason, the proposed solution is based on several different topics (according to the ACM taxonomy for the Computer Science): software engineering (software process technology: SEE, process system, CASE tools,...), information technologies and systems (conceptual architectures, meta-modeling,...), computing methodologies (ontologies, knowledge management,...) and, of course, project management.

Based on the previous ideas, the MANTIS Environment has been defined in this thesis with the following main characteristics: tool-based integration, process driven, maintenance specialization, and scalability and adaptability. The three classes of elements of the MANTIS Environment are:

- 1) A conceptual framework for the maintenance projects management that includes an multi-level conceptual architecture (based on the OMG MOF standard), a processes system based in the ISO standards, a collection of ontologies (of the maintenance, of the workflows, and of the measurement) that can be shared for the software tools and for

the human agents, and a meta-models set represented in form of XML, DTD or XMI documents.

- 2) A collection of procedures for the managing and organizational processes (as they are defined in the ISO 12207 life-cycle norm), that complement and improve the MANTEMA methodology, a complete and rigorous model in order to carry out the maintenance process.
- 3) A suite of prototypes which are defined as software components of the MANTIS Environment. The software architecture of the MANTIS system establishes three types of components: vertical (to automate a concrete activity type), horizontal (to automate a global service of the Environment), and external (not included into the MANTIS Environment, but which can be invoked by the internal tools).

The main horizontal developed prototypes have been the following: a data and metadata (in XML and XMI formats) repository manager, a meta-modeling tool, a knowledge base manager, and an interface of integration. The vertical prototypes include, among other, a modification requests manager, a tool for the human resources management among a maintenance projects portfolio, or a questionnaires manager for the maturity evaluation of a maintenance services. Lastly, the main external tools considered in the MANTIS Environment are the "Workflow Management Systems" which are proposed as "process engine" for the processes enacting, that is to say, to automate the projects monitoring and control.

Códigos UNESCO.

120317: Informática / Computer Science.

120318: Sistemas de Información, Diseño y Componentes / Information Systems, Design and Components.

Palabras Clave / Keywords.

Entorno de Ingeniería del Software.

Gestión de Proyectos Software.

Tecnología de Proceso Software.

Mantenimiento de Software.

Software Engineering Environment.

Software Project Management.

Software Process Technology.

Software Maintenance.