Chapter 1.19
An Overview of Software Quality Concepts and Management Issues

Alain April
École de technologie supérieure, Québec, Canada

Claude Y. Laporte
École de technologie supérieure, Québec, Canada

ABSTRACT

This chapter introduces the generally accepted knowledge on software quality that has been included in the (SWEBOK) Software Engineering Body of Knowledge (ISOTR 19759, 2005). One chapter of the SWEBOK is dedicated to software quality (April et al., 2005). It argues that ethics play an important role in applying the quality models and the notions of cost of quality for software engineers. It also describes the minimal content required in a software quality assurance plan. Finally an overview of what to expect in the upcoming international standards on software quality requirements, which transcend the life cycle activities of all IT processes, is presented.

INTRODUCTION

The business value of a software product results from its quality as perceived by both acquirers and end users. Therefore, quality is increasingly seen as a critical attribute of software, since its absence results in financial loss as well as dissatisfied users, and may even endanger lives. For example, Therac-25, a computer-driven radiation system, seriously injured and killed patients by massive overdosing (Levenson & Turner, 1993). Improving recognition of the importance of setting software quality requirements and of assessing quality causes a shift in the "center of gravity" of software engineering from creating technology-centered solutions to satisfying stakeholders.

Software acquisition, development, maintenance, and operations organizations confronted with such a shift are, in general, not adequately equipped to deal with it. Until recently, they did not have at their disposal the quality models or measurement instruments to allow (or facilitate) the engineering of quality throughout the entire software product life cycle. The objective of software product quality engineering is to achieve the required quality of the product through the definition of quality requirements and their implementation, measurement of appropriate quality attributes, and evaluation of the resulting quality. The objective is, in fact, software product quality.

This chapter is structured in accordance with the SWEBOK classification of the software quality body of knowledge (www.swebok.org) shown in Figure 1.
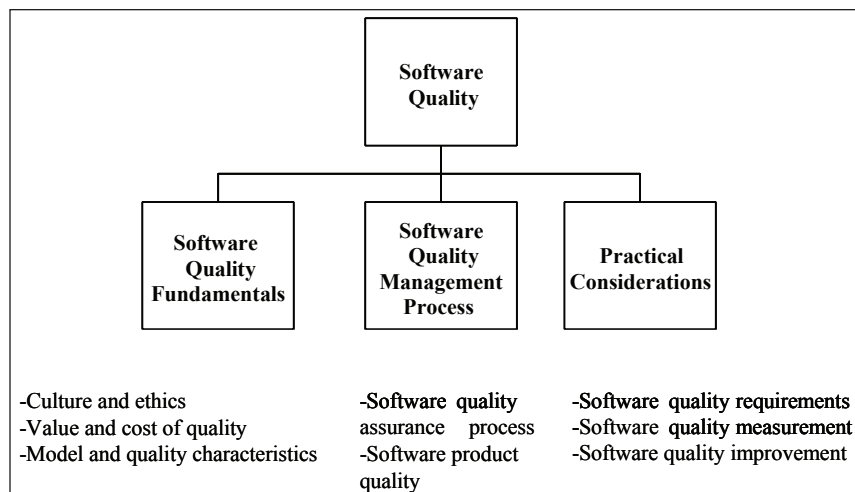
SOFTWARE QUALITY FUNDAMENTALS

Agreement on quality requirements, as well as clear communication on what constitutes qual-ity, require that the many aspects of quality be formally defined and discussed. Over the years, authors and organizations have defined the term "quality" differently. IBM used the phrase "market-driven quality", which is based on achieving total customer satisfaction. This definition was influenced by the total quality management approach of Phil Crosby (1979), who defined quality as "conformance to user requirements". Watts Humphrey (Humphrey, 1990), looking at quality in the software industry, defined it as "achieving excellent levels of fitness for use". More recently, quality has been defined in ISO 9001 (2000) as "the degree to which a set of inherent *characteristics* fulfills the *requirements*." The next section looks at how organizational culture and individual ethics play a role in quality in the organization.

Culture and Ethics

Edward B. Tylor (1871) defined human culture as "that complex whole which includes knowledge, belief, art, morals, law, custom, and any other capabilities and habits acquired by man as a member of society." Culture guides the behaviors, activities,

Figure 1. Adapted breakdown of software quality topics (ISOTR 19759, 2005)

-Culture and ethics
-Value and cost of quality
-Model and quality characteristics

-Software quality assurance process
-Software product quality

-Software quality requirements
-Software quality measurement
-Software quality improvement

Related Content

The Influences of Technology on Digital Economy Development in Vietnam
Vu Khanh Ngo Tan, Viet Phuong Truongand Truong-Xuan Do (2021). *International Journal of Software Innovation (pp. 10-18).*
www.irma-international.org/article/the-influences-of-technology-on-digital-economy-development-in-vietnam/289166

Component-Based ERP Design in a Distributed Object Environment
Bonn-Oh Kimand Ted Lee (2002). *Successful Software Reengineering (pp. 214-219).*
www.irma-international.org/chapter/component-based-erp-design-distributed/29979

Retrofitting Existing Web Applications with Effective Dynamic Protection Against SQL Injection Attacks
San-Tsai Sunand Konstantin Beznosov (2012). *Security-Aware Systems Applications and Software Development Methods (pp. 169-189).*
www.irma-international.org/chapter/retrofitting-existing-web-applications-effective/65848

A New Method for Writing Assurance Cases
Yutaka Matsunoand Shuichiro Yamamoto (2013). *International Journal of Secure Software Engineering (pp. 31-49).*
www.irma-international.org/article/new-method-writing-assurance-cases/76354

Multi-Object Tracking Using Gradient-Based Learning Model in Video Surveillance
 Mohana Priya D. (2021). *International Journal of Software Innovation (pp. 50-66).*
www.irma-international.org/article/multi-object-tracking-using-gradient-based-learning-model-in-video-surveillance/303332