Harmonizing Quality Assurance Processes and Product Characteristics

César Pardo, *Kybele Consulting Colombia*
Francisco J. Pino, *University of Cauca, Colombia*
Félix García, *University of Castilla-La Mancha, Spain*
Mario Piattini, *Alarcos Quality Center, Spain*

Adapting software quality assurance processes to product requirements will result in greater product quality and compliance, and thus increased customer satisfaction.

Today, information technology is characterized by rapid innovation and intense competition. Consequently, software organizations must be prepared to develop products that meet quality requirements at a low cost and in a short time.

According to Alfonso Fuggetta, "the quality of a software product heavily depends on the people, organization, and procedures used to create and deliver it" ("Software Process: A Roadmap," *Proc. Conf. Future Software Eng.*, ACM Press, 2000, pp. 25-34). Numerous models have been developed to assure the quality of software processes including CMMI (Capability Maturity Model Integration), the ISO 12207 standard, COBIT (Control Objectives for Information and related Technology), and ITIL (Information Technology Infrastructure Library). Other models aim to guarantee the quality of software products including FURPS (Functionality, Usability, Reliability, Performance, Supportability) and the ISO 9126 and ISO 25000 standards.

Despite the wide range of available models to define processes and develop better products, companies typically implement a process without considering its effect on the quality characteristics of the product they're developing. However, it's impossible to select the proper reference model without understanding how it impacts the product.

THE PRODUCT-PROCESS DILEMMA

Researchers have attempted to ascertain the business value of adopting various CMMI-defined practices, with special emphasis on improving certain characteristics of software product quality. For example, correlating the maturity of the process a company uses with the number of errors in its products detected by customers has made it possible to determine the impact that the number of code lines (software size) has on the number of defects.

Most studies agree that institutionalizing process improvements improves software product quality. However, even companies that use the same development process at the same level of maturity don't achieve the same levels of quality.

What is a major quality requirement in one company may not be so in another. "An organization that produces mission-critical software considers reliability to be the most important factor," Noushin Ashafi observed, "while portability may be a necessity for an organization that produces a software product for a variety of platforms" ("The Impact of Software Process Improvement on Quality: In Theory and Practice," *Information & Management*, Aug. 2003, pp. 677-690). This makes it problematic to extend, say, ISO 9126 to specific software environments.

Because there's no agreed-upon way to validate the effect of process improvements on product quality characteristics, companies don't harmonize the models they use, exponentially increasing development time, cost, and resources required.

AN INTEGRATED APPROACH

The solution to this dilemma is to analyze the various process reference models available and select

 Published by the IEEE Computer Society

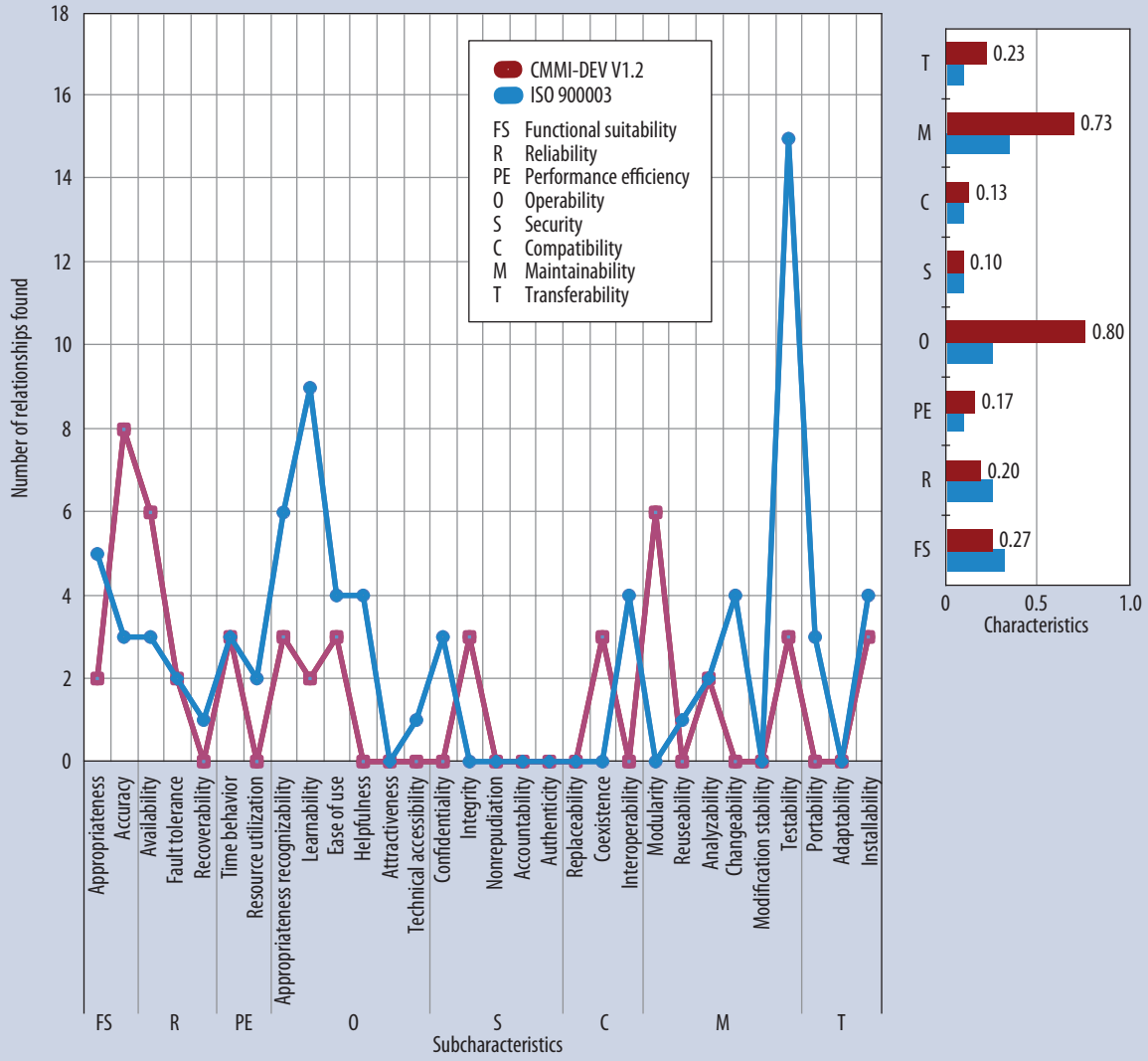

Figure 1. Relationship between ISO 25010 quality characteristics, CMMI-DEV V1.2 specific goals, and ISO 90003 clauses.

those processes most likely to support a product's particular quality requirements. Adapting processes from different models to product requirements will result in greater product quality and compliance, and thus increased customer satisfaction. It will also optimize the resources used for process improvement by linking those resources to the product's quality requirements. Although this integrated approach will likely increase short-term costs, it will ultimately cost less—and provide more value—than implementing separate process and product approaches.

No single process reference model is comprehensive enough to support all of a given product's quality characteristics. For example, specific quality requirements defined in ISO 25010 might best be supported by CMMI-DEV V1.2 or ISO 90003. As Figure 1 shows, CMMI-specific goals (SGs) provide more support for modularity, availability and accuracy, while ISO clauses offer greater support for installability, portability, testability, changeability, interoperability, helpfulness, ease of use, learnability, appropriateness recognizability, and appropriateness.

Special care must be taken when integrating the syntax and semantics of each model's descriptions, which can be at different levels of abstraction and detail. Consider, for example, the quality characteristic of time behavior, which ISO 25010 defines as "the degree to which the software product provides appropriate response and processing times, as well as throughput rates when performing its function, under stated conditions."

CMMI supports this requirement with three SGs. According to SG 3, "The verification methods, procedures, and criteria are used to verify the selected work products and any associated maintenance, train-

ing, and support services using the appropriate verification environment. Verification activities should be performed throughout the product life cycle. Practices related to peer reviews as a specific verification method are included in specific goal 2 …"

ISO 90003 has three clauses related to time behavior. According to clause 7.3.6.1, which deals with verification, "Before offering the product for customer acceptance, the organization should validate the operation of the product in accordance with its specified intended use, under conditions similar to the application environment, as specified in the contract. Any differences between the validation environment and the actual application environment, and the risks associated with such differences, should be identified and justified as early …"

CMMI provides greater implicit support for verifying time behavior than ISO, but an organization that carefully integrates CMMI and ISO processes could achieve even better support by tailoring the processes to the specific product requirements.

More work is needed to harmonize two software process engineering approaches that are widely used but rarely integrated in a coherent way: process quality assurance and product quality. To help guide organizations in applying processes from different models to specific quality requirements, researchers must create a formal framework that matches quality characteristics to process descriptions and outlines associated measures to ensure compliance.

César Pardo *is a member of Kybele Consulting Colombia, a consulting and research firm in Spinoff, Colombia, focusing on software process improvement, as well as a PhD student in computer science at the University of Castilla-La Mancha. Contact him at cpardo@unicauca.edu.co.*

Francisco J. Pino *is an associate professor at the University of Cauca, Colombia, as well as a PhD in computer science at the University of Castilla-La Mancha and founding partner of Kybele Consulting Colombia. Contact him at fjpino@unicauca.edu.co.*

Félix García *is an associate professor of computer science at the University of Castilla-La Mancha. Contact him at felix.garcia@uclm.es.*

Mario Piattini *is a director at Alarcos Quality Center, a spin-off from the Alarcos Research Group at the University of Castilla-La Mancha. Contact him at mario.piattini@uclm.es.*

Editor: Mike Hinchey, Lero—The Irish Software Engineering Research Centre; mike.hinchey@lero.ie