Science of Computer Programming 77 (2012) 636–643

Contents lists available at SciVerse ScienceDirect

Science of Computer Programming

journal homepage: www.elsevier.com/locate/scico

Regulations and software evolution: An example from the military domain

K. Burak Codur [a,*], Ali H. Dogru [b,1]

[a] Turkish Military Academy, Defense Sciences Institute, 06654 Bakanliklar Ankara, Turkey

[b] Middle East Technical University, Computer Engineering Department, 06531 Ankara, Turkey

ARTICLE INFO

Article history:
Received 19 January 2011
Received in revised form 15 December 2011
Accepted 16 December 2011
Available online 23 December 2011

Keywords:
Software development standard
Software evolution
IEEE/EIA 12207
MIL-STD-498
DOD-STD-2167
MIL-STD-1679

ABSTRACT

In this article, the impact of regulatory changes on software development is assessed in the context of military standards. A previously conducted experiment incorporating three standards is further investigated for this purpose, outlining the characteristics of the evolution in standards and its effects. In addition to this experiment that was designed with projects conducted as graduate class work, a real project from the industry is utilized, to demonstrate the similar effects of the evolution as discovered in the earlier experiment. Finally, the results of the assessment are generalized and a forecast is presented for the next potential regulation change, the IEEE Std 12207-2008.

1. Introduction

The domestic and international markets of information and communication technologies are governed by specific bodies, complying with established rules and mechanisms. Changes in market governance can be explained by power, technology, ideas or domestic politics [3]. Although technology raises the possibility of change, it does not dictate a particular set of changes and it is not the sole enabler of change [3]. Regulations are among the instruments that are used to govern domestic and international markets. A recent example is provided by China, whose policies related to software and standards indicate protectionist tendencies [15].

It is possible to control and direct software development activities utilizing specific regulations, such as software development standards and certification directives. These regulations determine which management practices, which processes and which technologies can be or must be applied during software engineering activities. For example, a regulation may require the use of specific software development process models, such as waterfall, incremental or spiral models, or a regulation like RTCA DO-178B may require the performance of a set of tests on the executable software [14].

A typical example of such regulations is the IEEE/EIA 12207.0-1996 standard. Although this standard was defined by international professional and trade organizations (and thus does not have an obligation initially), government or private bodies can utilize it as a mandatory standard, which is the case with the US Department of Defense (DoD): the IEEE/EIA 12207.0-1996 standard has been the software development standard of the DoD since 1998.

Regulations evolve over time. For the military domain the first software development standard, MIL-STD-1679 (later DOD-STD-1679) was defined in 1978 [9]. It was followed by DOD-STD-2167 (1985) [4], MIL-STD-498 (1994) [10] and

* Corresponding author. Tel.: +90 312 417 51 90x500; fax: +90 3124280930.
E-mail addresses: kbcodur@yahoo.com (K.B. Codur), dogru@ceng.metu.edu.tr (A.H. Dogru).
1 Tel.: +90 312 210 55 90.

0167-6423/$ – see front matter
doi:10.1016/j.scico.2011.12.001

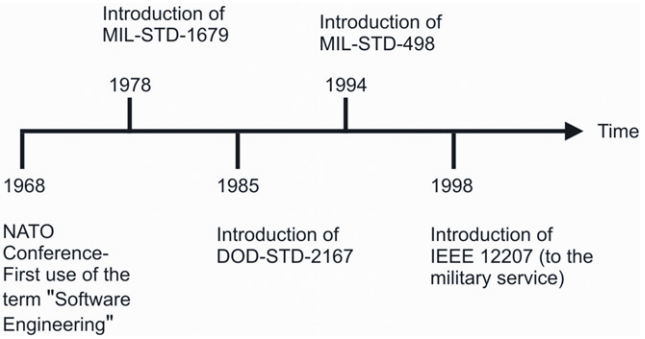

Fig. 1. Progress of software development standards in the military domain.

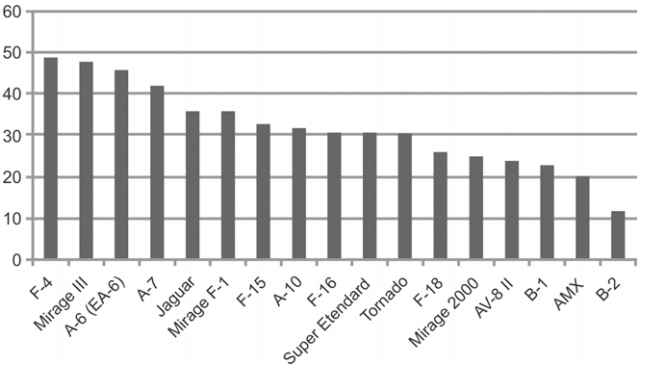

Fig. 2. In service history (in years) of jet aircrafts developed by NATO members.

IEEE/EIA 12207.0-1996 (1998) [5]. Fig. 1 depicts the chronology of the introduction of these standards. A review of these regulations' progress can be the starting point for understanding of their impact on software evolution.

However, although it may be possible to perform qualitative assessments on the progress of such regulations, finding empirical data and making quantitative analysis of the progress is not possible due to the nature of the subject: as regulations change over time, so do the requirements of the projects, the hardware and operating systems that run the software, the software engineering methodologies, the programming languages and the software engineers. Thus it is not possible to isolate the effects of regulatory changes from other factors. In a hypothetical world it might be possible to carry out two instances of a project with exactly the same requirements, software development environment and project team, but in accordance to different regulations; however such experimentation is impossible in the real world. Therefore, in order to perform quantitative analyses about the progress of regulations, special experiments and case studies have been conducted and are investigated in this article.

The military domain provides a field appropriate to research on the progress of regulations. This domain includes the whole spectrum of applications, but it is a monopsony, i.e., a market with a single acquirer: the armed forces. Thus the military domain is a closed and controlled domain. In this particular case, software development standards are the tools for regulation. Another relevant property of the military domain is the longer utilization time of the assets. For example, as depicted in Fig. 2, the average period of utilization of a military combat aircraft extends to thirty years. The software that operates on these aircrafts has a similar lifespan. Aircrafts in civil usage probably have similar utilization potential, but a ten-year old airliner is considered old and a twenty year-old ex-airliner, later adapted as a cargo aircraft, may be considered dangerous to fly. Consequently, a military aircraft may be developed in compliance with one standard and later may have to adapt to a newer standard. This long term utilization of military hardware increases the probability of one or more regulation changes in the life-time of the software. Therefore the connection between the lifespan of the software and the evolution of software development standards is strong in the military domain. Utilizing these unique properties of the military domain, discussions in the leading workshops related to this field of interest, such as the ERCIM Workshop on Software *Evolution* (*EVOL*) and the International Workshop on Principles of Software *Evolution* (IWPSE), can be investigated [2]. Such discussions demonstrate that the effects of regulations on software evolution have been neglected aspects of related research. This paper builds on these discussions, enhancing the research presented at IWPSE-EVOL 2009 with a case study and a projection onto the next standard.

Unlike previous research, this paper underlines the software engineering aspects of the standards' progress, which can be classified as an evolution. Previously, Moore and Rada covered the progress of standards, and stated that "if the 1970s and 1980s were a period of differentiation in life-cycle standards, the 1990s are a period of consolidation" [11]. McDonald