“Good quality” software requirements are prerequisite for “good quality” software products. Results of the research by Standish Group [23] verify our theory. The Standish Group’s CHAOS report that covers the findings from study of 8380 IT projects illustrates that 31.1% of projects are cancelled before they are completed. The results indicate 52.7% of projects cost 189% of their original estimates, and still deliver fewer features and functionalities than originally specified. Only 16.2% of software projects are completed on time and on budget. Among the projects completed by the large companies, only 42% of them comprise the originally proposed features and functions. The top three factors on challenged projects are lack of user input (12.8%), incomplete requirements and specifications (12.3%), and changing requirements and specifications (11.8%). Finally, the major reason for projects cancellation is reported as incomplete requirements (13.1%).
Cost of “bad quality” requirements have been studied since early 1970s. Boehm’s study of 63 software projects from three companies, namely GTE, TRW, and IBM, illustrated that the cost of change grows exponentially as the project progresses [2]. [4] reiterates this result by stating that the relative cost of repair is two hundred times greater in the maintenance phase than if it is detected in the requirements phase. Further, it bases the escalation in cost on two factors: (i) the delay from when the defect was introduced until it was detected, (ii) the amount of rework needed to correct both the original defect as well as the consequent defects in the later stages. As referred to by [4], DeMarco states that 56% of the bugs detected during testing can be traced to the requirements errors.

Iterative nature of RUP assists in eliminating above mentioned risks by integrating a software product progressively throughout its development life cycle, by managing requirements change and “creep” in a controlled manner, by learning early and improving incrementally, and by detecting flaws early thus, building higher quality over several iterations. Yet, RUP is a generic process and it is inevitable to tailor it according to the needs of a particular project or the projects of a specific department for better efficiency and effectiveness. In an attempt to establish a balance between delivering good quality software products and delivering them on time, ABB’s Stressometer product line adapted RUP in an agile fashion while adhering to the RUP essentials.

The main aim of this chapter is to evaluate a use-case driven, iterative software development process during which modeling is done via UML², within the context of requirements development and management, against the quality of the requirements established during such a process. To this end, Sect. 17.2 provides background information about ABB and the Stressometer product line. Section 17.3 presents the requirements management and engineering activities involved in ABB’s RUP-based software development process, SUP. Section 17.4 describes the characteristics of “good quality” requirements, elaborates on the relations among the characteristics, and further discusses how ABB Stressometer projects managed to achieve “good quality” requirements, supplying the discussions with experiences from the three major projects at ABB. Finally, Sect. 17.5 concludes the chapter.

17.2 Background

ABB (Asea Brown Boveri Ltd.) began operations in 1988 following a merger of two parent companies namely, ASEA AB and BBC Brown Boveri Ltd, each of which has been in business for more than a century (www.abb.com). Today, with about 105000 employees in around 100 countries, the ABB Group of companies functions in two core business areas, automation and power technologies that enable utility and industry customers to improve performance while lowering environmental impact.

---

² Unified Modelling Language