Chapter IV

Conceptual Modeling in Requirements Engineering: Weaknesses and Alternatives

Javier Andrade Garda, University of A Coruña, Spain
Juan Ares Casal, University of A Coruña, Spain
Rafael García Vázquez, University of A Coruña, Spain
Santiago Rodríguez Yáñez, University of A Coruña, Spain

Abstract

This chapter focuses on software engineering conceptual modeling, its current weaknesses, and the alternatives to overcome them. It is clear that software quality has its genesis in the conceptual model and depends on how well this model matches the problem in question. However, this chapter presents a representative study of the analysis approaches that highlights that (i) they have traditionally focused on implementation and have paid little or no attention to the problem domain and (ii) they have omitted the various stakeholders (viewpoints) generally involved in any problem. The proposed alternatives are based on those aspects that are related to a generic conceptualisation, independent of the implementation paradigms.
Introduction

The purpose of requirements engineering (RE) is to reach a thorough understanding of the needs of individuals involved in the problem to be solved. Because of this goal and its impact on subsequent phases, RE is an extremely relevant step in the software process. RE covers the requirements analysis activity, with conceptual modeling of the individuals' problem as one of its most remarkable tasks. In this activity, the problem to be solved is understood through conceptual models.

This chapter focuses on the conceptual modeling task within software engineering (SE), exposes its present weaknesses, and proposes a series of possible alternatives. The next section presents the basic and general aspects of conceptualisation regardless of SE. The following section tackles the relevance and orientation of conceptualisation in SE, and the two sections after that consider SE conceptualisation techniques and methods. The sections on alternatives and weaknesses list the weaknesses that result from this study and propose ways to avoid them based on the points of the section on understanding and conceptualising a problem. Finally the last section presents the most relevant conclusions.

Understanding and Conceptualising a Problem

Humans usually start solving non-trivial problems by gaining an understanding of these problems. This involves two basic activities:

1. Acquisition. All the possible information related to the problem is gathered from available sources. In general, initial acquisition is neither complete nor correct, because it is extremely difficult to gather all the information at once, and the gathered information is subject to inconsistencies. These inconveniences are gradually overcome by acquiring more information and by refining the information that is already available.

2. Conceptualisation. The gathered information is organised or modelled to form a meaningful whole: the conceptual model of the problem. If we define a concept as a mental structure that derives from the acquired information and is able to clarify or even solve a problem when applied to it, conceptualisation can be defined as the use of concepts and relationships to deal with and solve problems. Accordingly conceptual models are abstractions of the universe of discourse of the problem, as well as possible models of possible conceptual solutions to the problem.

Owing to the above-mentioned problem of acquisition, the timing of these activities is neither sequential nor clear; numerous overlaps and feedbacks occur that constitute an inherent evolutionary process, as befits any modeling process.
Conformance Analysis of Organizational Models: A New Enterprise Modeling Framework using Algebraic Graph Transformation
www.igi-global.com/article/conformance-analysis-organizational-models/75464?camid=4v1a

Implementing Internal Software Process Assessment: An Experience at a Mid-Size IT Company
www.igi-global.com/chapter/implementing-internal-software-process-assessment/61211?camid=4v1a