An Approach to Test-Driven Development of Conceptual Schemas[*]

Albert Tort, Antoni Olivé and Maria-Ribera Sancho

Department of Service and Information System Engineering
Universitat Politècnica de Catalunya - BarcelonaTech
Barcelona, Spain

{atort,olive,ribera}@essi.upc.edu

1 Summary

Conceptual modeling is an essential requirements engineering activity. Its objective is the development of the conceptual schema (CS) of an Information System (IS), which defines the general knowledge that an IS needs to know to perform its functions.

A CS consists of a structural (sub)schema and a behavioral (sub)schema. The structural schema consists of a taxonomy of entity types, a set of relationship types, and the constraints they must satisfy. The behavioral schema consists of a set of event types with their characteristics, constraints and effects.

A CS has semantic quality when it is valid and complete. Validity means that the schema is correct (the knowledge it defines is true for the domain) and relevant (the knowledge it defines is necessary for the system). Completeness means that the conceptual schema includes all relevant knowledge. Ensuring that a CS has semantic quality is a fundamental goal for its validation. This goal can be achieved by checking that the knowledge the system requires to know according to stakeholders' expectations is the same as the knowledge defined by the conceptual schema.

We report the development of a novel conceptual modeling method, named Test-Driven Conceptual Modeling (TDCM), which fosters the achievement of the above mentioned quality properties. TDCM is a test-driven method for the incremental development of conceptual schemas by continuous validation. As far as we know, this is the first work that explores the use of testing to drive conceptual modeling.

TDCM is based on our approach to test executable conceptual schemas written in formal modeling languages like UML/OCL. In this approach, automated conceptual test cases specify executable scenarios and include assertions that formalize expectations about them. In TDCM, the *Conceptual Schema Under Development* (CSUD) is developed in short iterations by applying the following tasks: (1) Write a test case that formalizes a story which is expected to be feasible with the knowledge defined in the schema; (2) change the schema to pass the test case; and (3) refactor the schema to improve the specification without changing the defined knowledge.

[*] This work has been partly supported by the Ministerio de Ciencia y Tecnología and FEDER under the project TIN2008-00444, Grupo Consolidado.

A. Ruíz, L. Iribarne (Eds.): JISBD'2012, pp. 407-408, ISBN:978-84-15487-28-9.
Jornadas SISTEDES'2012, Almería 17-19 sept. 2012, Universidad de Almería.

An iteration starts by adding a *new test case* to the passing test set of the previous iteration (*previous test set*). The objective of each iteration is to change the schema so that it includes the knowledge to correctly execute the *new test case*. The *previous test set* including the new test case is the *current test set* of the iteration. An iteration can only finish when the overall verdict of the *current test set* is *Pass*. At this point, the CSUD includes the knowledge to execute the tested cases as expected.

TDCM fosters refactoring of the CSUD in order to improve the quality of the conceptual schema specification without changing the knowledge specified in it. If the verdict of the *current test set* ceases to pass after refactoring, then we realize that the knowledge of the schema has not been preserved. If the verdict remains *Pass* and no more refactoring is felt to be needed, new iteration can be initiated.

Table 1 summarizes the interpretation of the verdict of the *current test set* (CTC) and the regression test set (RTS) at each iteration. This interpretation drives the application of TDCM in order to evolve the CSUD.

Verdict	**Current test case (CTC)**	**Regression test set (RTS)**	
Error	Relevant knowledge needs to be added to the CSUD	*Schema element removed*	The CSUD becomes incomplete and the deleted schema element needs to be restored
		Relevant constraint added	Modify inconsistent states to maintain consistent test cases
Fail	The knowledge defined in the CSUD needs to be corrected according to the asserted expectations	*Both CTC and RTS may Pass without changing them (only changing the CSUD)*	The knowledge defined in the CSUD needs to be corrected
		Neither CTC nor RTS can Pass without changing them	Inconsistent requirements
Pass	The CSUD has the necessary knowledge and satisfies the asserted expectations formalized in CTC	The CSUD still has the necessary knowledge and satisfies the asserted expectations formalized in RTS	

Table 1. Pass/Error/Failure interpretation

The proposed method is applicable to different kinds of projects and may be integrated into existing software development methods when they are based on iterative paradigms and they include artifacts to specify conceptual schemas. TDCM can also be used even if the conceptual schema to be developed is the main purpose of the project. Using TDCM, conceptual modelers have at any time fully tested schemas.

References

1. Tort, A., Olivé, A., Sancho, M. R.: An Approach to Test-Driven Development of Conceptual Schemas. Data & Knowledge Engineering, 69(6), 598-618 (2011)