Gregor Engels, Regina Hebig, Matthias Tichy (Hrsg.): SE 2023,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2023 43

BXtendDSL: A layered framework for bidirectional model transformations combining a declarative and an imperative language (Summary)

Thomas Buchmann[1], Matthias Bank[2], Bernhard Westfechtel[3]

Abstract: This summary is based on an article which appeared in 2022 in The Journal of Systems & Software [BBW22].
Bidirectional transformations have been studied in a wide range of application domains. In model-driven software engineering, they are required for roundtrip engineering processes. We present a pragmatic approach to engineering bidirectional model transformations that assists transformation developers by domain-specific languages, frameworks, and code generators. A thorough evaluation demonstrates conciseness, expressiveness, and scalability of our approach.

Keywords: model-driven software engineering; round-trip engineering; bidirectional transformation

1 Summary

Bidirectional transformations (bx) occur in different application domains, including e.g. databases, programming languages, and software engineering [Cz09]. Programming bidirectional transformations in a conventional programming language is both laborious and error-prone: Both transformation directions have to be programmed separately, and consistency of forward and backward transformations has to be checked by testing.

In response to these problems, a wide variety of bx approaches (functional, relational, or grammar-based) have been developed in research [Hi16]. A recurring theme driving bx research are *roundtrip properties*, also referred to as *bx laws*. Roundtrip properties are constraints on the interplay of forward and backward transformations. The goal of many bx approaches consists in the construction of bidirectional transformations that are *provably correct* with respect to roundtrip properties.

However, *empirical evaluations* [An20] demonstrate limitations of bx approaches with respect to *expressiveness*, i.e., the capability to solve a given bx problem. These limitations follow from the conditions that transformations have to satisfy in order to guarantee roundtrip

[1] Faculty of Computer Science, Deggendorf Institute of Technology, Dieter-Görlitz-Platz 1, 94469 Deggendorf, Germany thomas.buchmann@th-deg.de
[2] Chair of Applied Computer Science I, University of Bayreuth, Universitätsstraße 30, 95440 Bayreuth, Germany matthias.bank@uni-bayreuth.de
[3] Chair of Applied Computer Science I, University of Bayreuth, Universitätsstraße 30, 95440 Bayreuth, Germany bernhard.westfechtel@uni-bayreuth.de

CC BY SA

properties. Additional shortcomings were observed with respect to *conciseness* — the ability to provide for short solutions with respect to size metrics — and *scalability* — the ability to perform transformations efficiently on large data sets.

These observations motivated the development of *BXtendDSL* [BBW22], a framework for engineering bidirectional model transformations. BXtendDSL combines domain-specific languages (DSL) for bidirectional model transformations that are located at different levels of abstractions. A declarative language serves to specify a bidirectional transformation concisely. Round-trip properties are guaranteed as long as the transformation specification conforms to well-behavedness conditions. Intentionally, the declarative language is computationally incomplete, i.e., it is usually not possible to completely specify a bidirectional transformation at the declarative level. Therefore, the declarative language provides extension points for adding imperative code, which is written in an internal DSL.

We evaluated BXtendDSL with respect to expressiveness, conciseness, and scalability and compared it against competing bx approaches with a number of benchmarks, using the Benchmarx framework [An20]. Our evaluations demonstrate expressiveness (all test cases were passed), conciseness (short solutions were provided), and scalability (the implementation is scalable to large model sizes).

Data Availability

The software and the benchmarks are publicly available; see the instructions given at the end of the journal paper [BBW22].

Bibliography

[An20] Anjorin, Anthony; Buchmann, Thomas; Westfechtel, Bernhard; Diskin, Zinovy; Ko, Hsiang-Shang; Eramo, Romina; Hinkel, Georg; Samimi-Dehkordi, Leila; Zündorf, Albert: Benchmarking bidirectional transformations: theory, implementation, application, and assessment. Software and Systems Modeling, 19(3):647–691, 2020.

[BBW22] Buchmann, Thomas; Bank, Matthias; Westfechtel, Bernhard: BXtendDSL: A layered framework for bidirectional model transformations combining a declarative and an imperative language. J. Syst. Softw., 189:111288, 2022.

[Cz09] Czarnecki, Krzysztof; Foster, J. Nathan; Hu, Zhenjiang; Lämmel, Ralf; Schürr, Andy; Terwilliger, James F.: Bidirectional Transformations: A Cross-Discipline Perspective. In (Paige, Richard F., ed.): Proceedings of the Second International Conference on Theory and Practice of Model Transformations (ICMT 2009). volume 5563 of Lecture Notes in Computer Science, Springer-Verlag, Zurich, Switzerland, pp. 260–283, June 2009.

[Hi16] Hidaka, Soichoro; Tisi, Massimo; Cabot, Jordi; Hu, Zhenjiang: Feature-Based Classification of Bidirectional Transformation Approaches. Software and Systems Modeling, 15(3):907–928, July 2016.