Type-Checking CRDTs with Propel

GEORGE ZAKHOUR, PASCAL WEISENBURGER, and GUIDO SALVANESCHI,
University of St. Gallen, Switzerland

Conflict-free Replicated Data Types (CRDTs) [2, 3] are modern distributed data types that allow replicating data to different devices in a distributed system and enable local copies to diverge until they are merged with other replicas ensuring eventual consistency. CRDTs play a vital role in building *local-first applications* [1], i.e., applications where devices can always progress their local state independently while also enabling seamless collaboration among devices without being blocked by devices that are (temporarily) unreachable on the network. CRDTs enable keeping replicated data consistent while guaranteeing the absence of conflicts among replicas. CRDTs come in two flavors: state-based and operation-based (op-based). For correct operation, state-based CRDTs rely on a merge function for two states that is commutative, associative and idempotent, while operation-based CRDTs rely on an application function for operations on the state that commutes with itself.

However ensuring that such algebraic properties are satisfied by implementations is left to the programmer, resulting in a process that is complex and error-prone. While techniques based on testing, automatic verification of models, and mechanized or handwritten proofs are available, we lack an approach that is able to verify such properties on concrete CRDT implementations.

In this talk the first author will present the first type system that captures the algebraic properties required by a correct CRDT implementation. The type system is designed in Propel [4, 5], it can reason about programs and derive proofs of such properties with complex rules such as case analysis and induction: sum types guide the case analysis and algebraic properties in function types enable induction. Propel's key feature is its capacity to reason about algebraic properties (a) in terms of rewrite rules and (b) to derive the equality or inequality of expressions from the properties.

Propel's language is provided as a Scala embedding, in which several CRDTs were implemented and verified and compared with four state-of-the-art verification tools. The evaluation showed that Propel is able to automatically deduce the properties that are relevant for common state-based CRDT implementations found in open-source libraries even in cases in which competitors timeout.

We are in the process of extending this work also to op-based CRDTs that do not come with a single merge functions which needs to be checked for commutativity and associativity and idempotence, but instead we need to check the applying the operations on the CRDT commutes. Our initial investigation indicates that Propel's algebraic reasoning also helps in proving op-based CRDTs convergent, guaranteeing eventual consistency of local-first applications.

REFERENCES

[1] Martin Kleppmann, Adam Wiggins, Peter van Hardenberg, and Mark McGranaghan. 2019. Local-First Software: You Own Your Data, in Spite of the Cloud. In *Proceedings of the 2019 ACM SIGPLAN International Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software* (Athens, Greece) *(Onward! 2019)*. Association for Computing Machinery, New York, NY, USA, 154–178. https://doi.org/10.1145/3359591.3359737

[2] Marc Shapiro, Nuno Preguiça, Carlos Baquero, and Marek Zawirski. 2011. A Comprehensive Study of Convergent and Commutative Replicated Data Types. https://hal.inria.fr/inria-00555588

[3] Marc Shapiro, Nuno Preguiça, Carlos Baquero, and Marek Zawirski. 2011. Conflict-Free Replicated Data Types. In *Stabilization, Safety, and Security of Distributed Systems*, Xavier Défago, Franck Petit, and Vincent Villain (Eds.). Springer-Verlag, Berlin/Heidelberg, Germany, 386–400. https://doi.org/10.1007/978-3-642-24550-3_29

[4] George Zakhour, Pascal Weisenburger, and Guido Salvaneschi. 2023. Propel. https://propel-prover.github.io/. Last accessed on 19 Jul 2023.

[5] George Zakhour, Pascal Weisenburger, and Guido Salvaneschi. 2023. Type-Checking CRDT Convergence. *Proc. ACM Program. Lang.* 7, PLDI, Article 162 (jun 2023), 24 pages. https://doi.org/10.1145/3591276

Authors' address: George Zakhour, george.zakhour@unisg.ch; Pascal Weisenburger, pascal.weisenburger@unisg.ch; Guido Salvaneschi, guido.salvaneschi@unisg.ch, University of St. Gallen, Torstrasse 25, St. Gallen, SG, Switzerland, 9000.