

# OOPSLA2 2024

## Proceedings of the ACM on Programming Languages, Volume 8, Number OOPSLA2

Powered by  
Conference Publishing Consulting

Proceedings of the ACM on Programming Languages, Volume 8, Number OOPSLA2, October 20–25, 2024, Pasadena, CA, USA

### Plume: Efficient and Complete Black-Box Checking of Weak Isolation Levels

Si Liu, Long Gu, Hengfeng Wei, and David Basin

(ETH Zurich, Switzerland; Nanjing University, China)

Modern databases embrace weak isolation levels to cater for highly available transactions. However, weak isolation bugs have recently manifested in many production databases. This raises the concern of whether database implementations actually deliver their promised isolation guarantees in practice. In this paper we present Plume, the first efficient, complete, black-box checker for weak isolation levels. Plume builds on modular, fine-grained, transactional anomalous patterns, with which we establish sound and complete characterizations of representative weak isolation levels, including read committed, read atomicity, and transactional causal consistency. Plume leverages a novel combination of two techniques, vectors and tree clocks, to accelerate isolation checking. Our extensive assessment shows that Plume can reproduce all known violations in a large collection of anomalous database execution histories, detect new isolation bugs in three production databases along with informative counterexamples, find more weak isolation anomalies than the state-of-the-art checkers, and efficiently validate isolation guarantees under a wide variety of workloads.

 Published Artifact  Artifacts Available  Artifacts Reusable  Results Reproduced

### References

1. Atul Adya. 1999. Weak Consistency: A Generalized Theory and Optimistic Implementations for Distributed Transactions. Ph. D. Dissertation. USA. 
2. Mustaque Ahmad, Gil Neiger, James E. Burns, Prince Kohli, and Phillip W. Hutto. 1995. Causal Memory: Definitions, Implementation, and Programming. *Distributed Comput.*, 9, 1 (1995), 37–49. 
3. Deepthi Devaki Akkooorath, Alejandro Z. Tomsic, Manuel Bravo, Zhongmiao Li, Tyler Crain, Annette Bieniusa, Nuno M. Preguiça, and Marc Shapiro. 2016. Cure: Strong Semantics Meets High Availability and Low Latency. In *ICDCS 2016*. IEEE Computer Society, 405–414. 
4. AntidoteDB. Accessed in October, 2023. <https://www.antidotedb.eu/> 
5. Hagit Attiya, Faith Ellen, and Adam Morrison. 2015. Limitations of Highly-Available Eventually-Consistent Data Stores. In *PODC 2015*. ACM, 385–394. 
6. Peter Bailis, Aaron Davidson, Alan Fekete, Ali Ghodsi, Joseph M. Hellerstein, and Ion Stoica. 2013. Highly Available Transactions: Virtues and Limitations. *Proc. VLDB Endow.*, 7, 3 (2013), nov, 181–192. 
7. Peter Bailis, Alan D. Fekete, Ali Ghodsi, Joseph M. Hellerstein, and Ion Stoica. 2013. HAT, Not CAP: Towards Highly Available Transactions. In *HotOS XIV*. USENIX Association. 
8. Peter Bailis, Alan D. Fekete, Ali Ghodsi, Joseph M. Hellerstein, and Ion Stoica. 2016. Scalable Atomic Visibility with RAMP Transactions. *ACM Trans. Database Syst.*, 41, 3 (2016), 15:1–15:45. 
9. Sam Bayless, Noah Bayless, Holger H. Hoos, and Alan J. Hu. 2015. SAT Modulo Monotonic Theories. In *AAAI 2015*. AAAI Press, 3702–3709. 
10. Hal Berenson, Phil Bernstein, Jim Gray, Jim Melton, Elizabeth O’Neil, and Patrick O’Neil. 1995. A Critique of ANSI SQL Isolation Levels. In *SIGMOD ’95*. ACM, 1–10. isbn:0897917316 
11. Philip A. Bernstein, David W. Shipman, and Wing S. Wong. 1979. Formal Aspects of Serializability in Database Concurrency Control. *IEEE Trans. Software Eng.*, 5, 3 (1979), 203–216. 
12. Ranadeep Biswas and Constantin Enea. 2019. On the complexity of checking transactional consistency. *Proc. ACM Program. Lang.*, 3, OOPSLA (2019), 165:1–165:28. 
13. Ahmed Bouajjani, Constantin Enea, Rachid Guerraoui, and Jad Hamza. 2017. On verifying causal consistency. In *POPL 2017*. ACM, 626–638. 
14. Sebastian Burckhardt, Alexey Gotsman, Hongseok Yang, and Marek Zawirski. 2014. Replicated data types: specification, verification, optimality. In *Proceedings of the 41st ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL ’14)*. ACM, 271–284. 
15. Sebastian Burckhardt, Daan Leijen, Jonathan Protzenko, and Manuel Fähndrich. 2015. Global Sequence Protocol: A Robust Abstraction for Replicated Shared State. In *ECOOP 2015 (LIPIcs, Vol. 37)*. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 568–590. 
16. Andrea Cerone, Giovanni Bernardi, and Alexey Gotsman. 2015. A Framework for Transactional Consistency Models with Atomic Visibility. In *CONCUR’15 (LIPIcs, Vol. 42)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 58–71. isbn:978-3-939897-91-0 issn:1868-8969 

17. Andrea Cerone and Alexey Gotsman. 2018. Analysing Snapshot Isolation. *J. ACM*, 65, 2 (2018), Article 11, Jan.. 
18. Audrey Cheng, Xiao Shi, Lu Pan, Anthony Simpson, Neil Wheaton, Shilpa Lawande, Nathan Bronson, Peter Bailis, Natacha Crooks, and Ion Stoica. 2021. RAMP-TAO: Layering Atomic Transactions on Facebook's Online TAO Data Store. *Proc. VLDB Endow.*, 14, 12 (2021), 3014–3027. 
19. Jack Clark, Alastair F. Donaldson, John Wickerson, and Manuel Rigger. 2024. Validating Database System Isolation Level Implementations with Version Certificate Recovery. In *Proceedings of the Nineteenth European Conference on Computer Systems (EuroSys '24)*. ACM, 754–768. isbn:9798400704376 
20. MariaDB Galera Cluster. Accessed in October, 2023. <https://mariadb.com/kb/en/what-is-mariadb-galera-cluster/> 
21. Natacha Crooks, Youer Pu, Lorenzo Alvisi, and Allen Clement. 2017. Seeing is Believing: A Client-Centric Specification of Database Isolation. In *PODC'17*. ACM, 73–82. 
22. Diego Didona, Rachid Guerraoui, Jingjing Wang, and Willy Zwaenepoel. 2018. Causal Consistency and Latency Optimality: Friend or Foe? *Proc. VLDB Endow.*, 11, 11 (2018), 1618–1632. 
23. Jiaqing Du, Calin Iorgulescu, Amitabha Roy, and Willy Zwaenepoel. 2014. GentleRain: Cheap and Scalable Causal Consistency with Physical Clocks. In *SoCC 2014*. ACM, 4:1–4:13. 
24. ElectricSQL. Accessed in December, 2023. <https://electric-sql.com/> 
25. Elle. Accessed in November, 2023. Consistency Models in Elle. [https://github.com/jepsen-io/elle/blob/main/src/elle/consistency\\_model.clj](https://github.com/jepsen-io/elle/blob/main/src/elle/consistency_model.clj) 
26. João Ferreira Loff, Daniel Porto, João Garcia, Jonathan Mace, and Rodrigo Rodrigues. 2023. Antipode: Enforcing Cross-Service Causal Consistency in Distributed Applications. In *SOSP '23*. ACM, 298–313. 
27. C. J. Fidge. 1988. Timestamps in message-passing systems that preserve the partial ordering. *Proceedings of the 11th Australian Computer Science Conference*, 10, 1 (1988), 56–66. 
28. Yifan Gan, Xueyuan Ren, Drew Ripberger, Spyros Blanas, and Yang Wang. 2020. IsoDiff: Debugging Anomalies Caused by Weak Isolation. *Proc. VLDB Endow.*, 13, 12 (2020), 2773–2786. 
29. Martin Gebser, Roland Kaminski, Benjamin Kaufmann, and Torsten Schaub. 2012. *Answer Set Solving in Practice*. Morgan & Claypool Publishers. 
30. Graphviz. Accessed in December, 2023. Open source graph visualization software. <https://graphviz.org/> 
31. Long Gu, Si Liu, Tiancheng Xing, Hengfeng Wei, Yuxing Chen, and David Basin. 2024. IsoVista: Black-box Checking Database Isolation Guarantees. *Proc. VLDB Endow.*, 17, 12 (2024). 
32. Kaile Huang, Si Liu, Zhenge Chen, Hengfeng Wei, David A. Basin, Haixiang Li, and Anqun Pan. 2023. Efficient Black-box Checking of Snapshot Isolation in Databases. *Proc. VLDB Endow.*, 16, 6 (2023), 1264–1276. 
33. Jepsen. Accessed in March, 2024. <https://jepsen.io> 
34. Jepsen. Accessed in March, 2024. Jepsen Analyses. <https://jepsen.io/analyses> 
35. Zu-Ming Jiang, Si Liu, Manuel Rigger, and Zhendong Su. 2023. Detecting Transactional Bugs in Database Engines via Graph-Based Oracle Construction. In *OSDI '23*. USENIX Association, 397–417. 
36. Nick Kallen. Accessed in December, 2023. Big Data in Real Time at Twitter. <https://www.infoq.com/presentations/Big-Data-in-Real-Time-at-Twitter/> 
37. Kyle Kingsbury. 2023. MySQL 8.0.34. <https://jepsen.io/analyses/mysql-8.0.34> 
38. Kyle Kingsbury and Peter Alvaro. 2020. Elle: Inferring Isolation Anomalies from Experimental Observations. *Proc. VLDB Endow.*, 14, 3 (2020), 268–280. 
39. Leslie Lamport. 1978. Time, Clocks, and the Ordering of Events in a Distributed System. *Commun. ACM*, 21, 7 (1978), 558–565. 
40. Si Liu. 2022. All in One: Design, Verification, and Implementation of SNOW-Optimal Read Atomic Transactions. *ACM Trans. Softw. Eng. Methodol.*, 31, 3 (2022). 
41. Si Liu, Long Gu, Hengfeng Wei, and David Basin. 2024. Plume: Efficient and Complete Black-Box Checking of Weak Isolation Levels. <https://github.com/draco00000/Plume> 
42. Si Liu, Luca Multazzu, Hengfeng Wei, and David Basin. 2024. NOC-NOC: Towards Performance-optimal Distributed Transactions. *Proc. ACM Manag. Data*, 2, 1 (2024). 
43. Si Liu, Peter Csaba Ölveczky, Qi Wang, Indranil Gupta, and José Meseguer. 2019. Read atomic transactions with prevention of lost updates: ROLA and its formal analysis. *Formal Aspects Comput.*, 31, 5 (2019), 503–540. 
44. Si Liu, Peter Csaba Ölveczky, Min Zhang, Qi Wang, and José Meseguer. 2019. Automatic Analysis of Consistency Properties of Distributed Transaction Systems in Maude. In *TACAS 2019 (LNCS, Vol. 11428)*. Springer, 40–57. 
45. Wyatt Lloyd, Michael J. Freedman, Michael Kaminsky, and David G. Andersen. 2011. Don't settle for eventual: scalable causal consistency for wide-area storage with COPS. In *SOSP 2011*. ACM, 401–416. 
46. Wyatt Lloyd, Michael J. Freedman, Michael Kaminsky, and David G. Andersen. 2013. Stronger Semantics for Low-Latency Geo-Replicated Storage. In *NSDI 2013*. USENIX Association, 313–328. 
47. Haonan Lu, Siddhartha Sen, and Wyatt Lloyd. 2020. Performance-Optimal Read-Only Transactions. In *OSDI 2020*. USENIX Association, 333–349. 
48. Taras Lykhenko, Rafael Soares, and Luis Rodrigues. 2021. FaaSSTCC: Efficient Transactional Causal Consistency for Serverless Computing. In *Middleware '21*. ACM, 159–171. isbn:9781450385343 
49. Umang Mathur, Andreas Pavlogiannis, Hünkar Can Tunç, and Mahesh Viswanathan. 2022. A Tree Clock Data Structure for Causal Orderings in Concurrent Executions. In *ASPLOS'22*. ACM, 710–725. 
50. Syed Akbar Mehdi, Cody Littley, Natacha Crooks, Lorenzo Alvisi, Nathan Bronson, and Wyatt Lloyd. 2017. I Can't Believe It's Not Causal! Scalable Causal Consistency with No Slowdown Cascades. In *NSDI 2017*. USENIX Association, 453–468. 
51. Microsoft. Accessed in December, 2023. Azure CosmosDB DB. <https://learn.microsoft.com/en-us/azure/cosmos-db/consistency-levels> 
52. Neo4j. Accessed in December, 2023. <https://neo4j.com/> 
53. Christos H. Papadimitriou. 1979. The Serializability of Concurrent Database Updates. *J. ACM*, 26, 4 (1979), 631–653. 
54. Pedro Pereira and António Rito Silva. 2023. Transactional Causal Consistent Microservices Simulator. In *Distributed Applications and Interoperable Systems - 23rd IFIP WG 6.1 International Conference, DAIS 2023 (LNCS, Vol. 13909)*. Springer, 57–73. 
55. Matthieu Perrin, Achour Mostefaoui, and Claude Jard. 2016. Causal consistency: beyond memory. In *Proceedings of the 21st ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (PPoPP '16)*. ACM, Article 26, 12 pages. 

56. PostgreSQL. Accessed in December, 2023. Transaction Isolation. <https://www.postgresql.org/docs/current/transaction-iso.html> 
57. Potassco. Accessed in December, 2023. Clingo: A grounder and solver for logic programs. <https://github.com/potassco/clingo> 
58. RUBiS. Accessed in December, 2023. Auction Site for e-Commerce Technologies Benchmarking. <https://projects.ow2.org/view/rubis/> 
59. Cheng Tan, Changgeng Zhao, Shuai Mu, and Michael Walfish. 2020. COBRA: Making Transactional Key-Value Stores Verifiably Serializable. OSDI'20. USENIX Association. Article 4. 
60. TPC. Accessed in December, 2023. TPC-C: On-Line Transaction Processing Benchmark. <https://www.tpc.org/tpcc/> 
61. Viper. Accessed in December, 2023. Configurations in Viper. <https://github.com/Khoury-srg/Viper/blob/main/src/config.yaml> 
62. Todd Warszawski and Peter Bailis. 2017. ACIDRain: Concurrency-Related Attacks on Database-Backed Web Applications. In SIGMOD'17. ACM, 5–20. 
63. Chenggang Wu, Vikram Sreekanti, and Joseph M. Hellerstein. 2020. Transactional Causal Consistency for Serverless Computing. In SIGMOD '20. ACM, 83–97. isbn:9781450367356 
64. Shale Xiong, Andrea Cerone, Azalea Raad, and Philippa Gardner. 2020. Data Consistency in Transactional Storage Systems: A Centralised Semantics. In ECOOP 2020 (LIPIcs, Vol. 166). Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 21:1–21:31. 
65. YugabyteDB. Accessed in December, 2023. <https://www.yugabyte.com/> 
66. Rachid Zennou, Ranadeep Biswas, Ahmed Bouajjani, Constantin Enea, and Mohammed Erradi. 2022. Checking causal consistency of distributed databases. Computing, 104, 10 (2022), 2181–2201. 
67. Jian Zhang, Ye Ji, Shuai Mu, and Cheng Tan. 2023. Viper: A Fast Snapshot Isolation Checker. In Proceedings of the Eighteenth European Conference on Computer Systems (EuroSys '23). ACM, 654–671. 

Extracted 2024-10-02 06:18 by BBLTEX

proc time: 4.95