Annotated Bibliography on Transactions

Hengfeng Wei
hfwei@nju.edu.cn

1 Books

Weikum, G., and Vossen, G. *Transactional Information Systems: Theory, Algorithms, and the Practice of Concurrency Control and Recovery*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2001

It is researchers-oriented. Highly recommended.

2 Transactional Consistency Models

2 Frameworks

Berenson, H., Bernstein, P., Gray, J., Melton, J., O'Neil, E., and O'Neil, P. A critique of ansi sql isolation levels. *SIGMOD Rec. 24*, 2 (May 1995), 1–10

Defines Isolation Levels in terms of phenomena; Introduces new phenomena; Define Snapshot Isolation.

Atluri, V., Bertino, E., and Jajodia, S. A theoretical formulation for degrees of isolation in databases. *Information and Software Technology 39*, 1 (1997), 47 – 53

This paper formulates these different degrees of isolation in terms of histories, as in the case of the usual serialization theory and proposes timestamp-based protocols for different degrees of isolation.

Koskinen, E., and Parkinson, M. The push/pull model of transactions. In *Proceedings of the 36th ACM SIGPLAN Conference on Programming Language Design and Implementation*, PLDI '15, Association for Computing Machinery (New York, NY, USA, 2015), 186–195

We present a general theory of serializability, unifying a wide range of transactional algorithms, including some that are yet to come. To this end, we provide a compact semantics in which concurrent transactions PUSH their effects into the shared view (or UNPUSH to recall effects) and PULL the effects of potentially uncommitted concurrent transactions into their local view (or UNPULL to detangle).

Crooks, N., Pu, Y., Alvisi, L., and Clement, A. Seeing is believing: A client-centric specification of database isolation. In *Proceedings of the ACM Symposium on Principles of Distributed Computing*, PODC '17, Association for Computing Machinery (New York, NY, USA, 2017), 73–82

This paper introduces the first state-based formalization of isolation guarantees.

Crooks, N. *A Client-Centric Approach to Transactional Datastores*. PhD thesis, The University of Texas at Austin, 2019

The PhD Thesis version of [7].

2 Serializability

Eswaran, K. P., Gray, J. N., Lorie, R. A., and Traiger, I. L. The notions of consistency and predicate locks in a database system. *Commun. ACM 19*, 11 (Nov. 1976), 624–633

This is the first paper to formalize mathematically the concurrency control problem. It also defines "conflict serializability", which is termed DSR in [13].

Papadimitriou, C. H. The serializability of concurrent database updates. *J. ACM 26*, 4 (Oct. 1979), 631–653

It is shown that recognizing the transaction histories that are serializable is an NP-complete problem. Several efficiently recognizable subclasses are introduced.

Kanellakis, P. C., and Papadimitriou, C. H. Is distributed locking harder? In *Proceedings of the 1st ACM SIGACT-SIGMOD Symposium on Principles of Database Systems*, PODS '82, Association for Computing Machinery (New York, NY, USA, 1982), 98–107

We examine the problem of determining whether a set of locked transactions, accessing a distributed database, is guaranteed to produce only serializable schedules. For a pair of transactions we prove that this concurrency control problem (which is polynomially solvable for centralized databases) is in general coNP-complete.

Yannakakis, M. Serializability by locking. *J. ACM 31*, 2 (Mar. 1984), 227–244

It is shown that locking cannot achieve the full power of serializability. An exact characterization of the schedules that can be produced if locking is used to control concurrency is given for two versions of serializability: state serializability and view serializability.

See also its STOC conference version [19].

Attar, R., Bernstein, P. A., and Goodman, N. Site initialization, recovery, and backup in a distributed database system. *IEEE Trans. Softw. Eng. 10*, 6 (Nov. 1984), 645–650

Introduce One Copy Serializability (1SR), as a distributed/replicated counterpart of Serializability in a single-server system.

Kanellakis, P. C., and Papadimitriou, C. H. The complexity of distributed concurrency control. *SIAM J. Comput. 14*, 1 (Feb. 1985), 52–74

We present a formal framework for distributed databases, and we study the complexity of the concurrency control problem in this framework. Our transactions are partially ordered sets of actions, as opposed to the straight-line programs of the centralized case. The concurrency control algorithm, or scheduler, is itself a distributed program.

Ports, D. R. K., and Grittner, K. Serializable snapshot isolation in postgresql. *Proc. VLDB Endow. 5*, 12 (Aug. 2012), 1850–1861

This paper describes our experience implementing PostgreSQL's new serializable isolation level. It is based on the recently-developed Serializable Snapshot Isolation (SSI) technique. This is the first implementation of SSI in a production database release as well as the first in a database that did not previously have a lock-based serializable isolation level.

2 Snapshot Isolation

Yabandeh, M., and Gómez Ferro, D. A critique of snapshot isolation. In *Proceedings of the 7th ACM European Conference on Computer Systems*, EuroSys '12, Association for Computing Machinery (New York, NY, USA, 2012), 155–168

We introduce write-snapshot isolation, a novel isolation level that has a performance comparable with that of snapshot isolation, and yet provides serializability. The main insight in write-snapshot isolation is to prevent read-write conflicts in contrast to write-write conflicts that are prevented by snapshot isolation.

3 Theory

4 Robustness (and Dependency Graphs)

Fekete, A., Liarokapis, D., O'Neil, E., O'Neil, P., and Shasha, D. Making snapshot isolation serializable. *ACM Trans. Database Syst. 30*, 2 (June 2005), 492–528

This article develops a theory that characterizes when nonserializable executions of applications can occur under SI.

5 Concurrency Control Protocols

5 Overview

Bernstein, P. A., and Goodman, N. Concurrency control in distributed database systems. *ACM Comput. Surv. 13*, 2 (June 1981), 185–221

In this paper we survey, consolidate, and present the state of the art in distributed database concurrency control. The heart of our analysts is a decomposition of the concurrency control problem into two major subproblems: read-write and write-write synchronization. We describe a series of synchronization techniques for solving each subproblem and show how to combine these techniques into algorithms for solving the entire concurrency control problem. Such algorithms are called "concurrency control methods." We describe 48 principal methods, including all practical algorithms that have appeared m the literature plus several new ones. We concentrate on the structure and correctness of concurrency control algorithms. Issues of performance are given only secondary treatment.

5 Locking

Buckley, G. N., and Silberschatz, A. Beyond two-phase locking. *J. ACM 32*, 2 (Apr. 1985), 314–343

Graph protocols.

Thomasian, A. Concurrency control: Methods, performance, and analysis. *ACM Comput. Surv. 30*, 1 (Mar. 1998), 70–119

This tutorial reviews CC methods based on standard locking, restart-oriented locking methods, two-phase processing methods including optimistic CC, and hybrid methods (combining optimistic CC and locking) in centralized systems.

6 Formal Methods

Pollak, D. H. Reasoning about two-phase locking concurrency control. Technical report, June 2017

We present a program logic for serializable transactions that are able to manipulate a shared storage.

We show this by providing the first application of our logic in terms of the Two-phase locking (2pl) protocol which ensures serializability.

7 Systems