



॥ त्वं ज्ञानमयो विद्वानमयोऽसि ॥

Distributed Data Storage and Management Part III

Saptarshi Pyne

Assistant Professor

Department of Computer Science and Engineering
Indian Institute of Technology Jodhpur, Rajasthan, India 342030

CSL4030 Data Engineering Lecture 13
August 30th, 2023

What we discussed in the last class

- Distributed data storage
 - Homogeneous and heterogeneous systems
 - Fragmentation and replication
 - Query processing with fragments

Data transparency

- Fragmentation, replication, and location transparencies
- Name server (network issues, server crash)
- Site identifiers
- Hybrid: Aliases stored at each site

Transactions

- A transaction is a logical unit of work.
E.g., a set of queries that performs a single task.
- When executing transactions on distributed databases, we must honour the ACID properties.

The ACID properties of transactions

- **Atomicity:** All-or-none.
- **Consistency**
- **Isolation:** Concurrent transactions produce the same outcome as in they were executed sequentially.
- **Durability:** The effect of a committed transaction must endure system failures.

An example of a transaction

Transaction T_i : John is sending money from account A in SBI to Kendriya Bhandar's account B in ICICI.

read(A);

$A := A - 50$;

write(A);

read(B);

$B := B + 50$;

write(B).

An example of a transaction

Transaction T_i : John Smith is sending money from account A in SBI to Kendriya Bhandar's account B in ICICI.

read(A);

$A := A - 50$;

write(A);

read(B);

$B := B + 50$;

write(B).

Local vs. global transactions

Local = Reads or writes the database at one particular site.

Global = Reads or writes databases at multiple sites.

Q. Was the exemplary transaction a local transaction or a global transaction?

Execution of a global/distributed transaction

Each site has a log file and two computer programmes – a transaction manager (TM) and a transaction coordinator (TC).

SBI

SBI initiates transaction T_i .

TC_{SBI} starts the execution.

TC_{SBI} breaks the transaction into two sub-transactions and distributes them to appropriate sites.

TM_{SBI} executes the following sub-transaction:

lock(A); read(A);

$A = A - 50$;

write(A); unlock(A);

TM_{SBI} maintains a log for recovery purposes.

TM_{SBI} informs TC_{SBI} that it has completed its task.

TC_{SBI} sends a “commit T_i ” message to all TMs.

TM_{SBI} adds <commit T_i > to its log.

ICICI

TM_{ICICI} executes the following sub-transaction:

lock(B); read(B);

$B = B + 50$;

write(B); unlock(B);

TM_{ICICI} maintains a log for recovery purposes.

TM_{ICICI} informs TC_{SBI} that it has completed its task.

TM_{ICICI} adds <commit T_i > to its log.

What could go wrong?

- Site failures
- Loss of messages
- Link failures and 'network partitions'

Resolution: The two-phase commit protocol (2PC)

Today we discussed

- Data transparencies
- Distributed/global transactions
 - The ACID properties

References

- A. SILBERSCHATZ, H.F. KORTH, S. SUDARSHAN (2011), Database System Concepts, McGraw Hill Publications, 6th Edition.
 - Chapter 19. Distributed Databases
- Paper: Bronson et al., “TAO: Facebook’s Distributed Data Store for the Social Graph”, 2013 USENIX Annual Technical Conference (USENIX ATC ‘13).
 - Video:
<https://www.usenix.org/conference/atc13/technical-sessions/presentation/bronson>

Thank you