

Data Management Algorithms

Consistency, Concurrency control, and Recovery

Boris Novikov

Contents

- Introduction: what is the problem
- Consistency: correctness criteria
- Concurrency control protocols
- Concurrency control for object model
- Recovery
- Concurrency control and recovery for distributed systems
- Extending consistency models: delivery

Data Management Algorithms
Consistency, Concurrency control, and Recovery
Introduction

Boris Novikov

Introduction: The Problem and Context

Global View: Data Centric vs. Application Centric

Data Centric View

- Data persist
- Applications are coming in and out
- The value of data is much higher than a single run of any application
- Relatively large amount of data

Application Centric View

- Data are imported when application starts and exported before it terminates
- Nothing happens between application executions
- Complex computations with small amount of data
-

Defining Transactions

- Transform the database from one consistent state to another if
 - 1.All statements were executed (no failures)
 - 2.No interference with other transactions
- The concept of consistent state depends on application semantics

Consistency is Essential for

Several different application scenarios:

- A stream of short transactions (OLTP)
- Distributed heterogeneous environments (e.g e-commerce)
- Complex workflows

Concurrent Execution: Interference

Time	Transaction 1		Transaction 2	
	Operation	Value	Operation	Value
1	read(x) into v	100		
2	v=v+50	150	read(x) into v	100
3	Write (x) from v	150	v=v+50	150
4			Write (x) from v	150

Finally $x=150$, should be 200

Commutative Operations

Time	Transaction 1		Transaction 2	
	Operation	Value	Operation	Value
1	Increment (x, 50)	150		
2			decrement(x, 20)	130
3	decrement(y, 50)	250		
4			increment(y, 20)	270

All results are correct

Distributed Coordination

- Seller sends card identification and amount to seller's bank
- Seller's bank sends request to authorization center
- Authorization center sends request to customer's bank
- Customer's bank sends response
- Authorization center sends response to seller's bank
- Seller's bank books the (financial) transaction
- Seller prints receipt

Workflows

- Long-running activities, probably including human actions
- An author submits an article
- The journal editor assigns reviewers
- Reviews returned to author(s), depending on evaluation
- Paper is rejected, revision requested, or paper is accepted
- The author submits a revised version,
- etc.
-

Desirable Properties of Concurrency Control and Transactions: ACID

- Atomicity: all or nothing
- Consistency: ensure correctness
- Isolation: avoid interference with other transactions
- Durability: never lose results of committed transactions

Computational Models

What is a computational model?

- A database model: data units and structures
- Operations
- Definition of transactions
- Models of concurrent execution
- Correctness criteria

Page Computational Model

- A database consists of independent elements (pages), denoted x, y, z, \dots
- Operations: read $r(x)$ and write $w(x)$, both operations are atomic
- Definition of transactions: to be formally defined
- Models of concurrent execution: to be formally defined
- Correctness criteria: to be defined

Page Model Semantics

Operation $r(x)$ reads the value of x written by the preceding write operation $w(x)$

Object Computational Model

- Database: same as for page model
- Operations: page model operations (r , w) and high-level operations organized into a tree
- Transactions, concurrency, and correctness: to be defined

Object Transactions: Example

