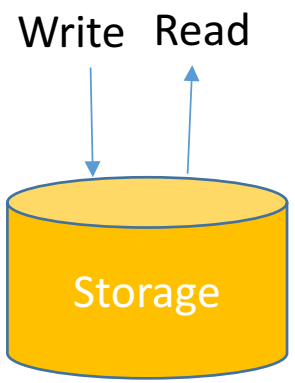
Transaction Management

Definition



Transaction = A list of writes and reads

For example: {Read, Read, Write}

Two Big Problems

1. Support multiple transaction at the same time

2. Make sure the data stored is reliable

Techniques

1. Concurrency Control

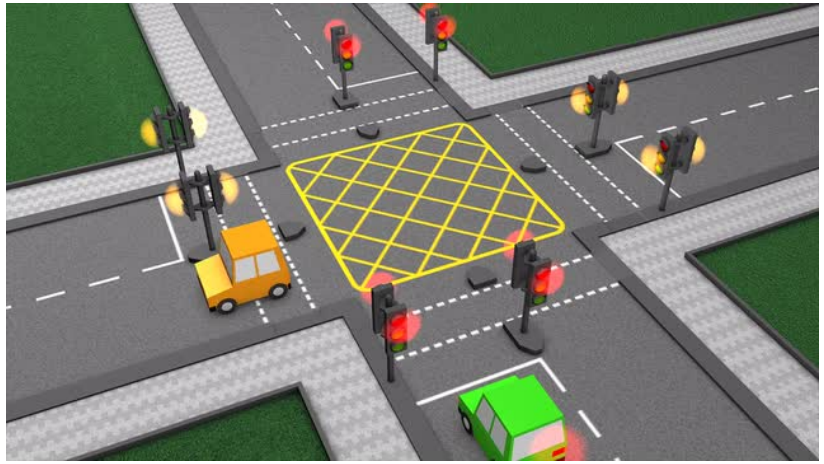
2. Database Recovery

Properties

- Atomicity
- Consistency
- Isolation
- Durability

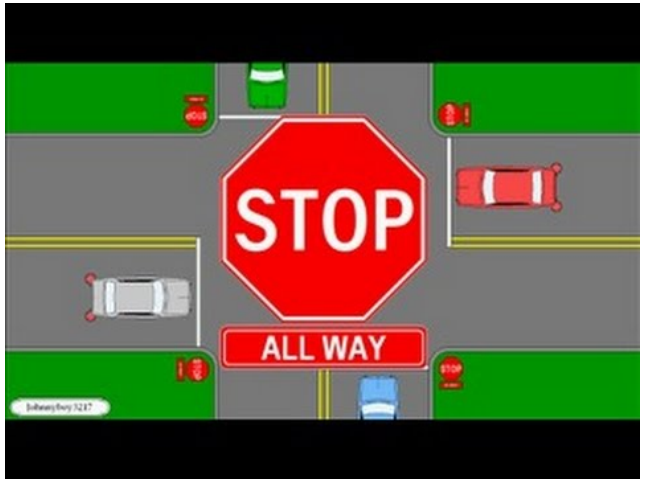
Concurrency Control

- Locking



2-phase locking

- Timestamp Ordering



Optimistic Concurrency Control

Multi-version concurrency control (MVCC)

Isolation Degrees

- **Motivation**
  - Serializability is expensive!!
- **Key Idea**
  - Holding locks for shorter time

Weaker

Stronger

Degrees	Read Lock	Write Lock
0	No Lock	Short
1	No Lock	Long
2	Short	Long
3	Long	Long

Isolation Degrees (Challenges)

- Specifications for weak isolation are often incomplete, ambiguous

Weaker

Stronger

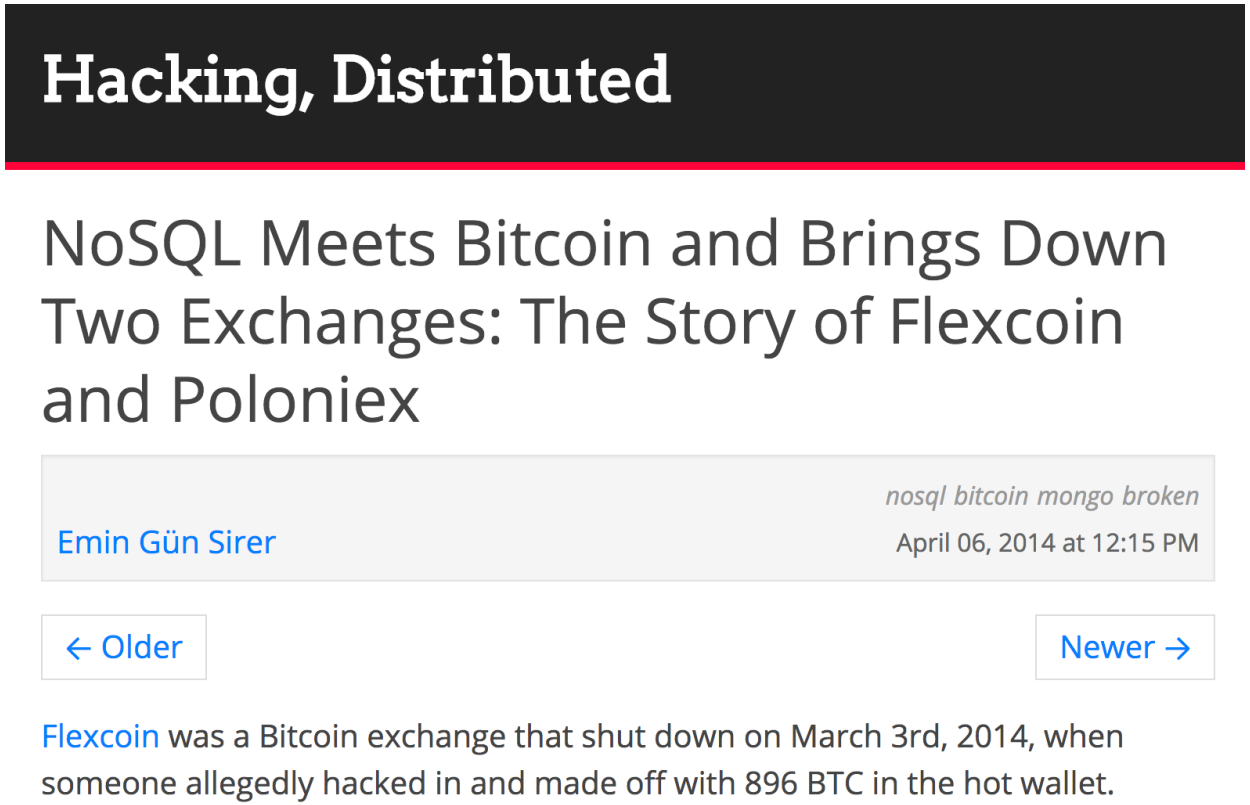
Degrees	Read Lock	Write Lock
0	No Lock	Short
1	No Lock	Long
2	Short	Long
3	Long	Long

- Hard to program

Spanner: Google's globally distributed database

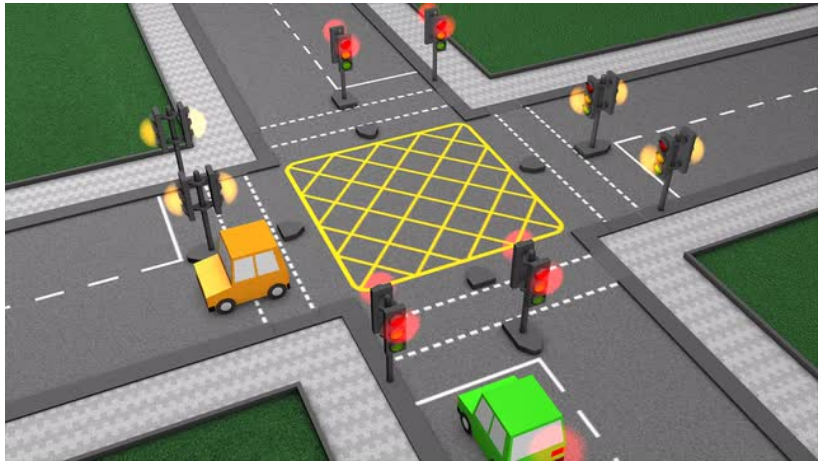
Ignore all issues

- At Facebook, only 0.0004% of results returned are inconsistent
- But,



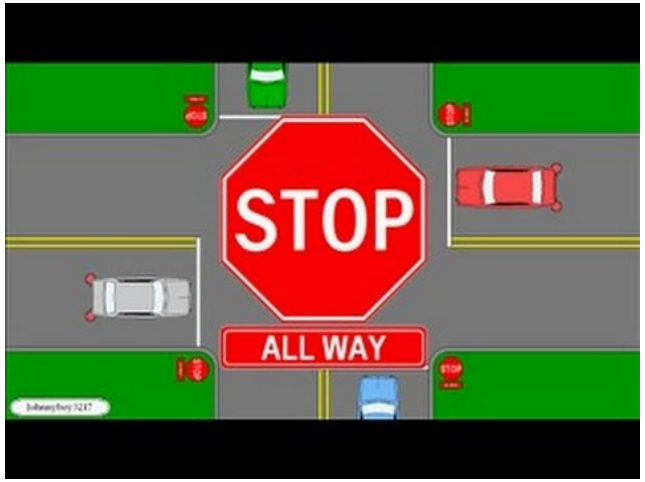
Concurrency Control

- Locking



2-phase locking

- Timestamp Ordering

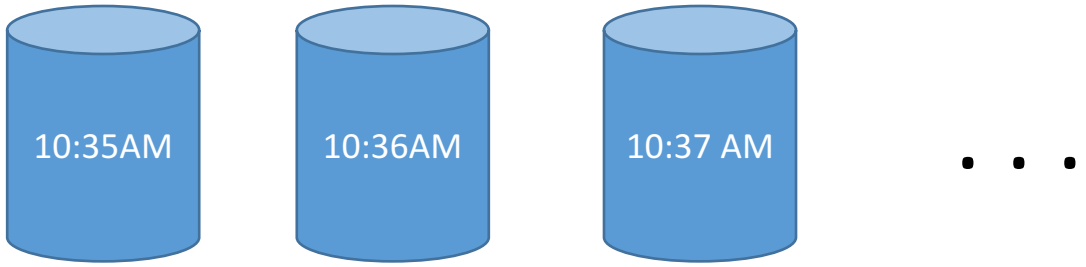


Optimistic Concurrency Control

Multi-version concurrency control (MVCC)

Snapshot Isolation

- **Key Idea:** Look at your database as a sequence of snapshots

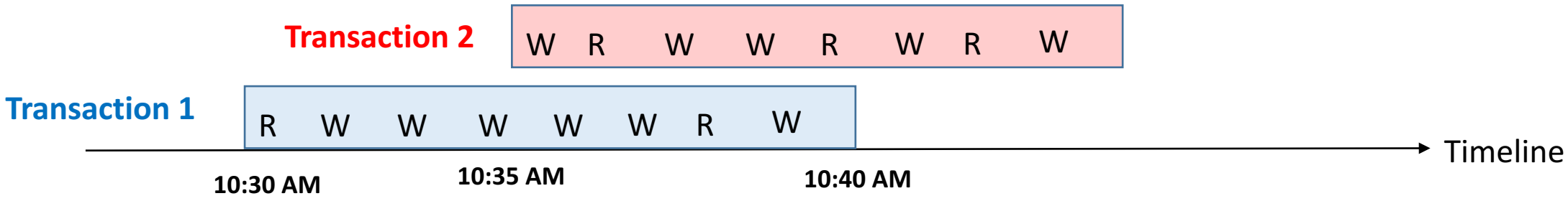


- **Oracle pioneered this idea**

- **Fact 1:** This is not equivalent to serializability
- **Fact 2.** Oracle calls this “serializable” mode

Multi-Version Concurrency Control (MVCC)

- **Conflict**

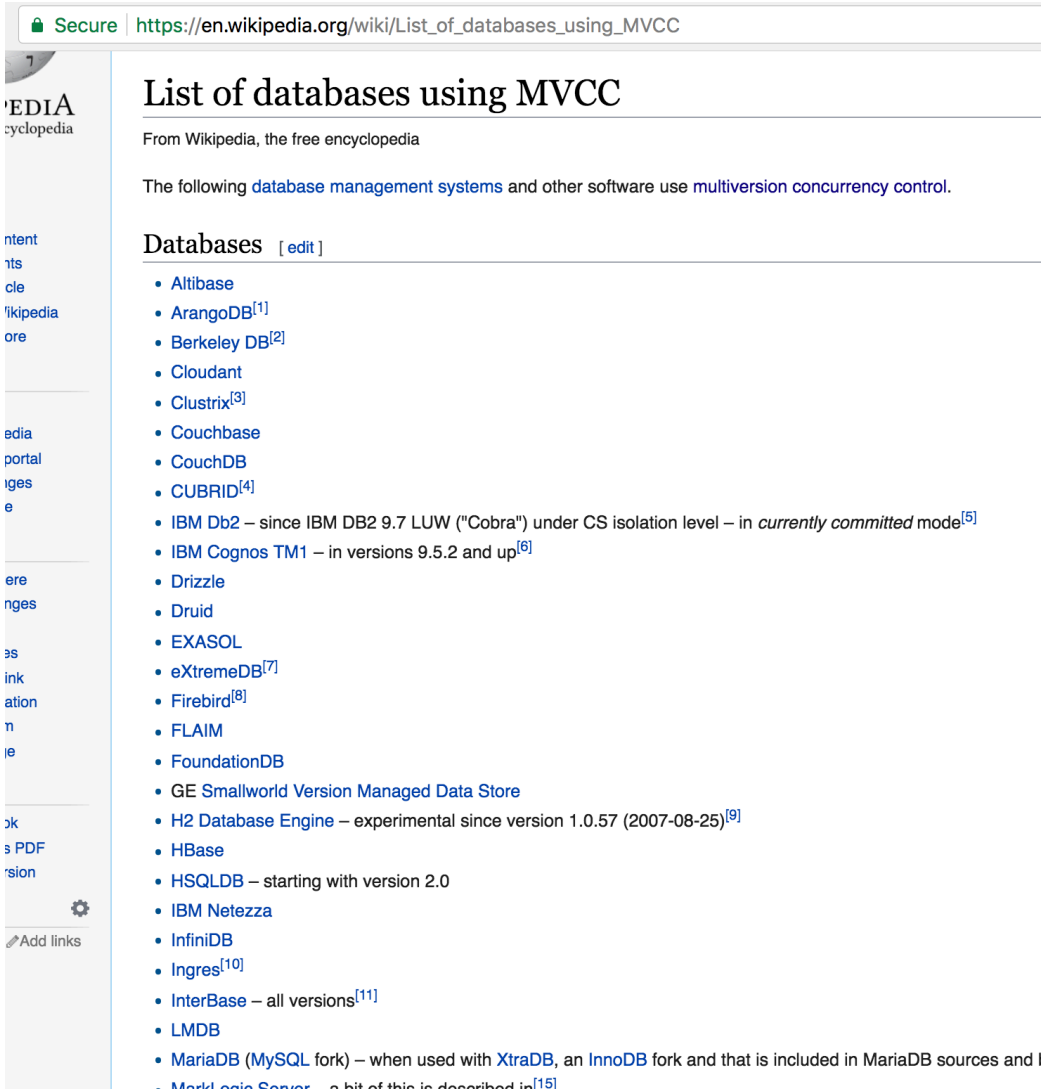


- **Key Idea:** Store multiple versions of a data value

Jiannan's Bank Balance: <$200, 10:31 AM>, <$100, 10:33 AM>, <$100, 10:35 AM>, …

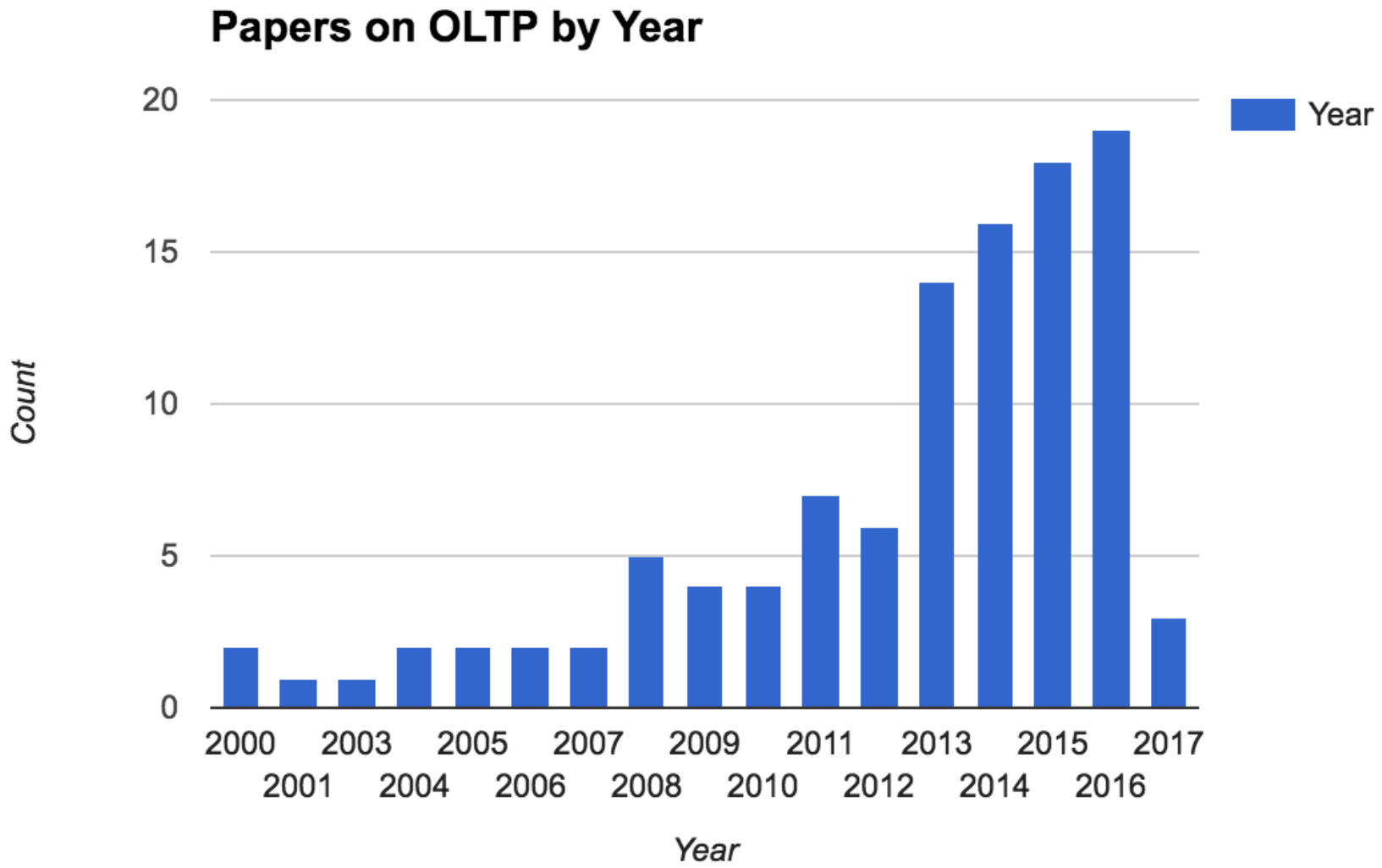
Locking vs. MVCC

- **In the Past**
  - More Conflicts → Locking is a winner
  - Less Conflicts → MVCC is a winner
- **Nowadays**
  - Flash/Memory makes data locality less important
  - CPU becomes the bottleneck
  - Cheap storage makes version history worth keeping



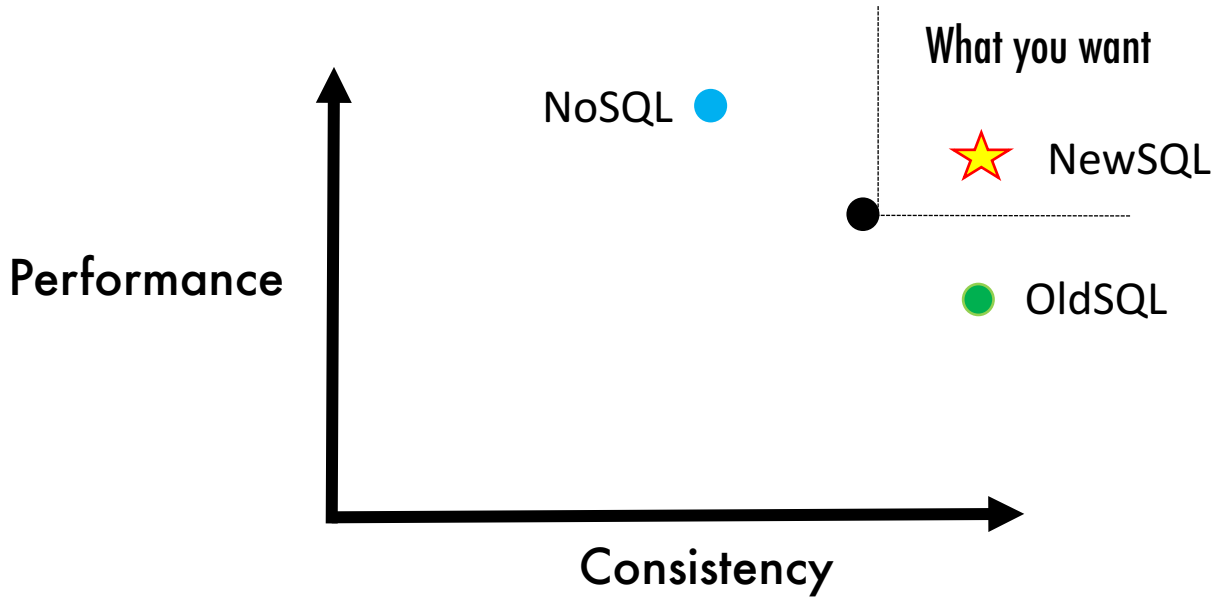
What happened in academia?

Papers in OSDI, SOSP, VLDB and SIGMOD



2000 ~ 2008

- **DB Community**
  - OLTP is a solved problem
- **Industry**
  - OLTP has bad performance



What happened afterwards (2008 – now)

- Technical Feasibility
  - Memory gets cheaper

- Market Feasibility
  - Google says that strong consistency is important (Spanner, 2012)
  - Cloud computing helps to lower the cost

No Concurrency Control!

Hogwild: A lock-free approach to parallelizing stochastic gradient descent

B Recht, C Re, S Wright, F Niu - Advances in neural information …, 2011 - papers.nips.cc

Abstract Stochastic Gradient Descent (SGD) is a popular algorithm that can achieve state-of-the-art performance on a variety of machine learning tasks. Several researchers have recently proposed schemes to parallelize SGD, but all require performance-destroying

☆ 99 Cited by 882 Related articles All 36 versions

Takeaways Messages

- What’s the difference between locking and timestamp ordering?
- Why Isolation degrees?
- Why MVCC dominates the market?
- Crazy Idea: No concurrency control!