

Non-Monotonic Snapshot Isolation: scalable and strong consistency for geo-replicated transactional systems

Masoud Saeida Ardekani, *Université Pierre-et-Marie-Curie*
Pierre Sutra, *Université de Neuchâtel*
Marc Shapiro, *INRIA & Université Pierre-et-Marie-Curie*

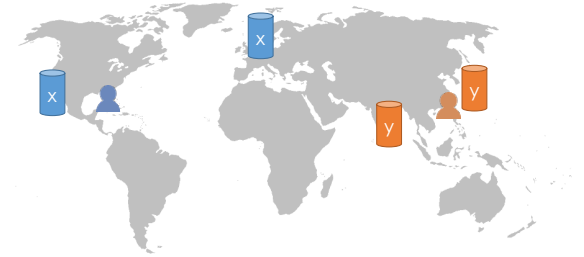
01/10/13

Non-Monotonic Snapshot Isolation
Masoud Saeida Ardekani

1

Motivation

- Geo-replication for
 - Low latency
 - Availability
 - Disaster tolerance



Requirements

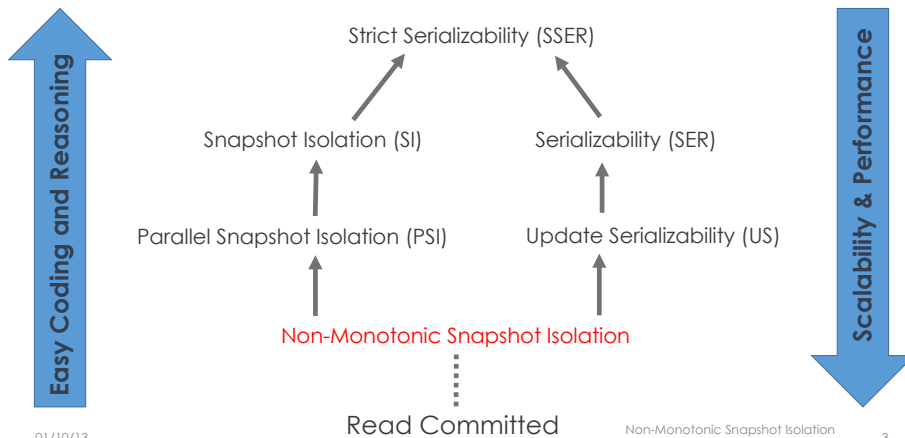
- Transaction
 - Strong consistency: No concurrent writes to the same object
 - Progress: if no conflict then commit

01/10/13

Non-Monotonic Snapshot Isolation
Masoud Saeida Ardekani

2

Consistency Hierarchy



01/10/13

Read Committed

Non-Monotonic Snapshot Isolation
Masoud Saeida Ardekani

3

Desired Scalability Properties

1. *Wait-Free Queries*
 - Queries do not wait or abort
2. *Minimal Commitment Synchronization*
 - Synchronize only if necessary
 - Transaction T_i waits for T_j only if they write-write conflict
3. *Genuine Partial Replication* (next slides)
4. *Forward Freshness* (next slides)

01/10/13

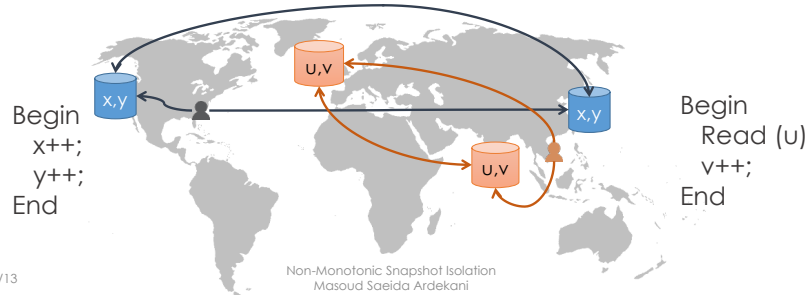
Non-Monotonic Snapshot Isolation
Masoud Saeida Ardekani

4

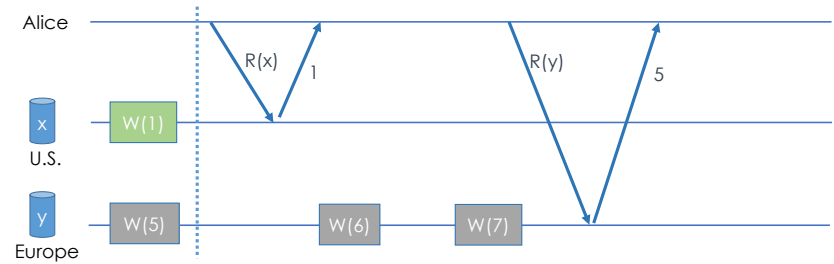
Desired Scalability Properties

3. Genuine Partial Replication [Schiper'10]

- Only replicas of objects read or written inside the transaction communicate
- Non-conflicting transactions do not interfere with each other
- Intrinsic parallelism of workloads can be exploited



Base Freshness Snapshots



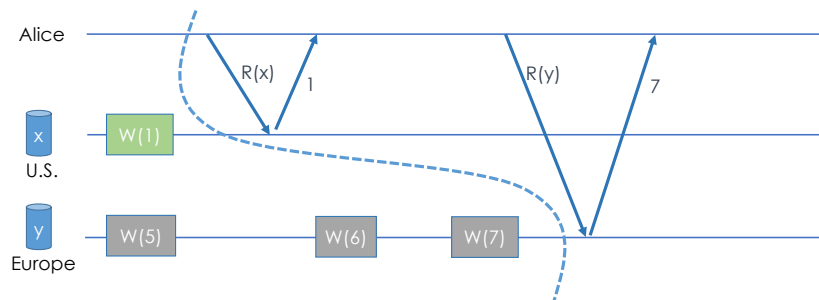
- Stale Data Reads
- Increased Abort Ratio
- More Global Communication [Saeida'13]

Snapshot Isolation
Parallel Snapshot Isolation

Desired Scalability Properties

4. Forward Freshness Snapshots

- try to read as recent as possible



Objective:
ensuring all four properties

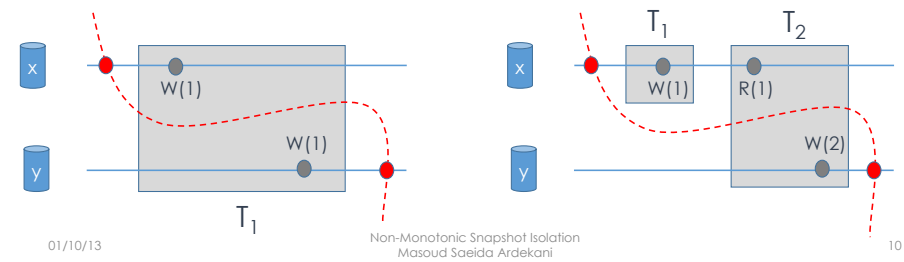
	Strict Serializability	Serializability	Update Serializability	Snapshot Isolation	Parallel Snapshot Isolation
Wait-free Queries	✓	✓	✓	✓	✓
Genuine Partial Rep.			✓		
Forw. Freshness Snap.	✓	✓	✓		
Minimum Comm. Sync.				✓	✓

Our solution: Non-Monotonic Snapshot Isolation (NMSI)

	Strict Serializability	Serializability	Update Serializability	Snapshot Isolation	Parallel Snapshot Isolation	NMSI
Wait-free Queries	✓	✓	✓	✓	✓	✓
Genuine Partial Rep.			✓			✓
Forw. Freshness Snap.	✓	✓	✓			✓
Minimum Comm. Sync.				✓	✓	✓

Non-Monotonic Snapshot Isolation

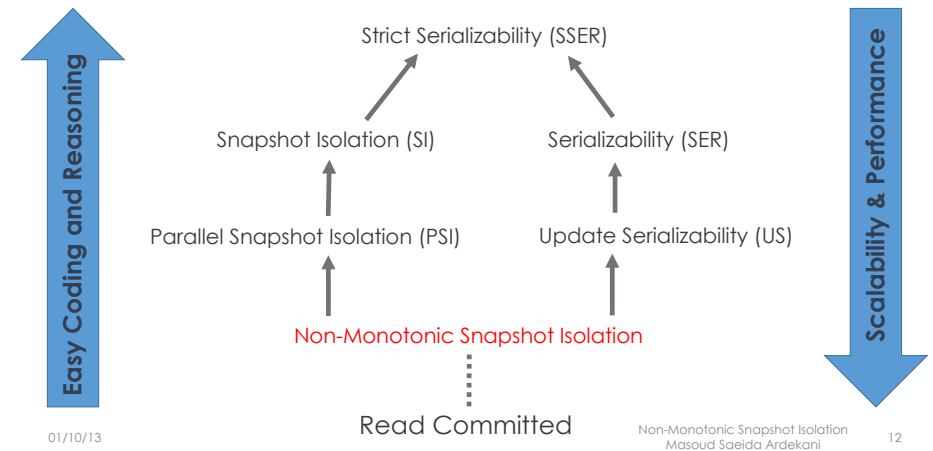
- Read committed values
- Take consistent snapshots



Non-Monotonic Snapshot Isolation

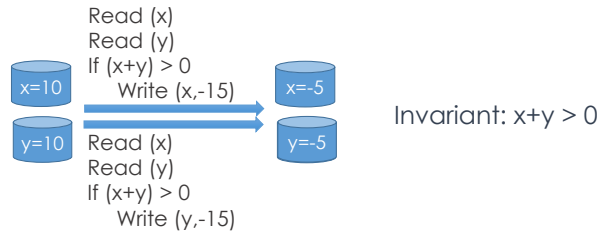
- Read committed values
- Take consistent snapshots → Forward Freshness
- During commit time
 - Always commit queries → Wait-Free Queries
 - Commit update transaction if no concurrent write-conflicting transaction → Minimal Commitment Sync & Genuine Partial Replication

Consistency Hierarchy



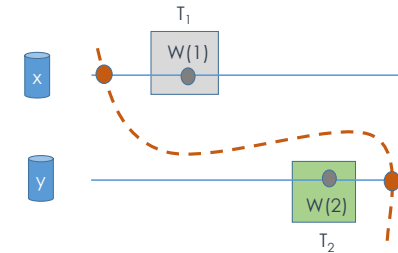
Anomalies

- Write Skew
 - Two concurrent transactions do not observe update of each other
 - Easy to solve [Cahill'08]
 - Pack each invariant into an object



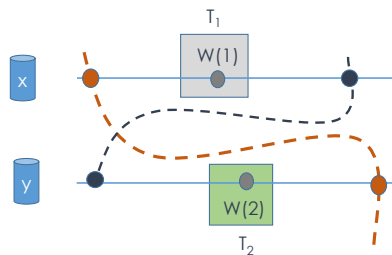
Anomalies

- Real-time Violation
 - Observe the effect of a transaction, but not all transactions preceding it in real time
 - Observable in Parallel Snapshot Isolation and Serializability



Anomalies

- Non-Monotonic Snapshots
- Avoidable in case of one client
- Observable in Parallel Snapshot Isolation



Anomalies

	Strict Serializability	Serializability	Update Serializability	Snapshot Isolation	Parallel Snapshot Isolation	NMSI
Write Skew				Observable	Observable	Observable
Real-Time Violation		Observable	Observable		Observable	Observable
Non-Monotonic Snapshots			Observable		Observable	Observable

Our Protocol

- Execution Phase
 - Execute reads using *Dependence Vector*
 - Compute updates
- Termination Phase
 - Propagate update transactions using Atomic Multicast
 - Certify update transactions

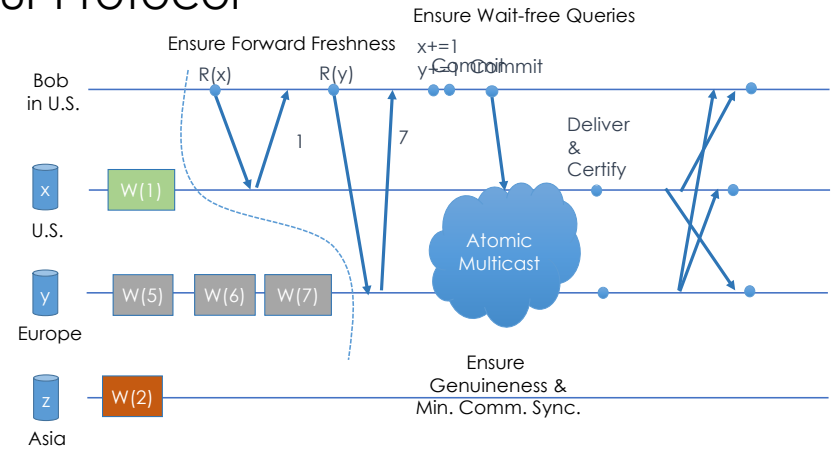


01/10/13

Non-Monotonic Snapshot Isolation
Masoud Saeida Ardekani

17

Our Protocol



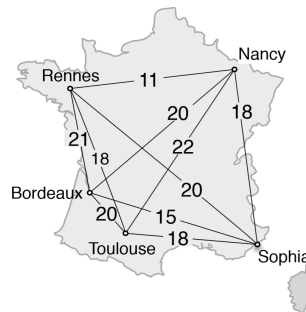
01/10/13

Non-Monotonic Snapshot Isolation
Masoud Saeida Ardekani

18

Evaluation Setup

- 5 Sites in Grid'5000
- A group of 3 replicas in each site
- Clients distributed uniformly among sites
- Modified YCSB benchmark [Cooper'10]

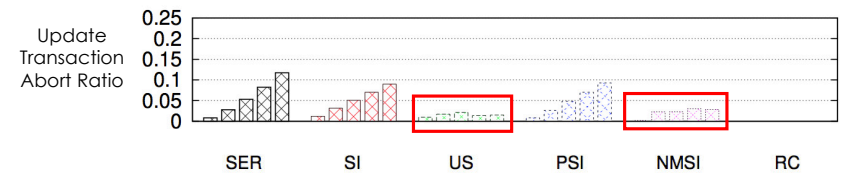


01/10/13

Non-Monotonic Snapshot Isolation
Masoud Saeida Ardekani

19

Forward Freshness & Min. Comm. Sync.



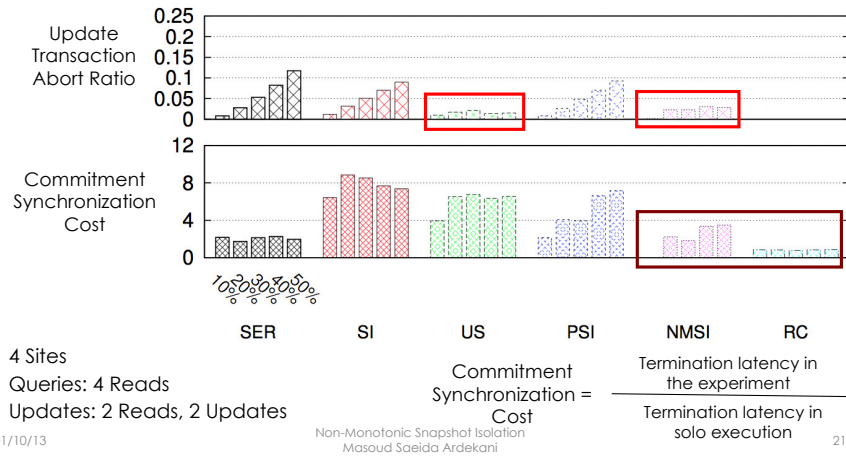
- 4 Sites
- Queries: 4 Reads
- Updates: 2 Reads, 2 Updates

01/10/13

Non-Monotonic Snapshot Isolation
Masoud Saeida Ardekani

20

Forward Freshness & Min. Comm. Sync.

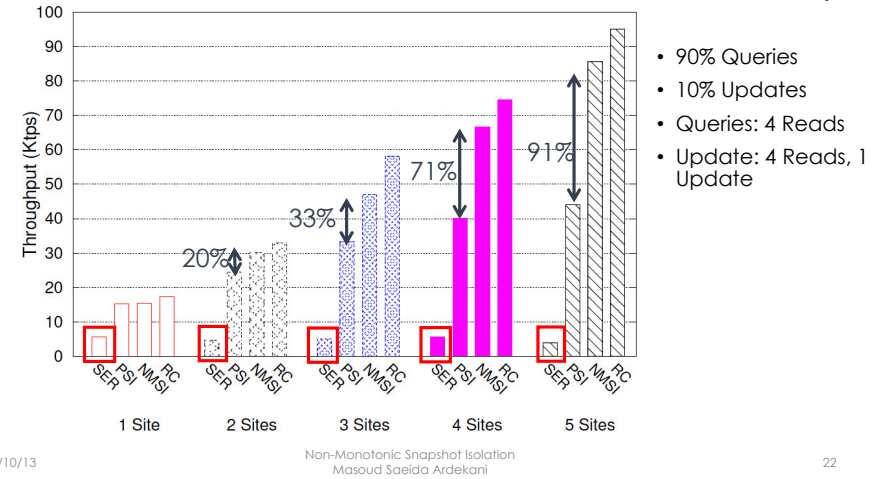


01/10/13

Non-Monotonic Snapshot Isolation
Masoud Saeida Ardekani

21

Wait-free Queries & Genuine Partial Rep.



01/10/13

Non-Monotonic Snapshot Isolation
Masoud Saeida Ardekani

22

Conclusion

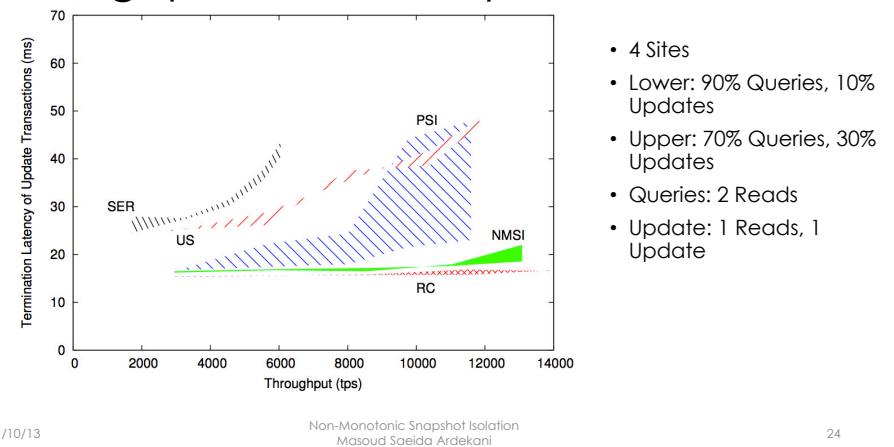
- Four scalability properties:
 - Wait-free queries
 - Forward freshness
 - Genuine partial replication
 - Minimal commitment synchronization
- Non-Monotonic Snapshot Isolation
 - Satisfying four properties
 - Between 2 to 14 times faster than other strong consistencies
 - No new anomaly compared to Parallel Snapshot Isolation

01/10/13

Non-Monotonic Snapshot Isolation
Masoud Saeida Ardekani

23

Throughput vs. Latency



01/10/13

Non-Monotonic Snapshot Isolation
Masoud Saeida Ardekani

24

Genuine Partial Replication (GPR)

[Schiper'10]

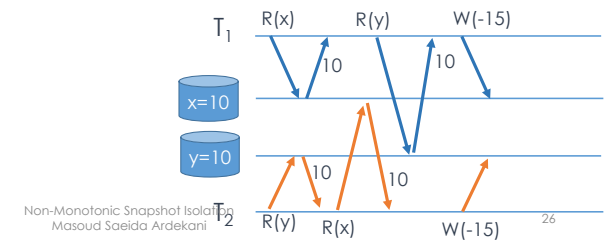
- Full Replication
 - Every replica must perform all updates
 - Not scalable
- Partial Replication
 - Addresses full replication issue
 - Protocols still rely on system-wide consensus/communication
- Genuine Partial Replication
 - Only replicas of objects accessed or modified inside the transaction communicate with each other
 - Non-conflicting transactions do not interfere with each other
 - Intrinsic parallelism of workloads can be exploited

01/10/13

Non-Monotonic Snapshot Isolation
Masoud Saeida Ardekani

25

01/10/13



Non-Monotonic Snapshot Isolation
Masoud Saeida Ardekani

26