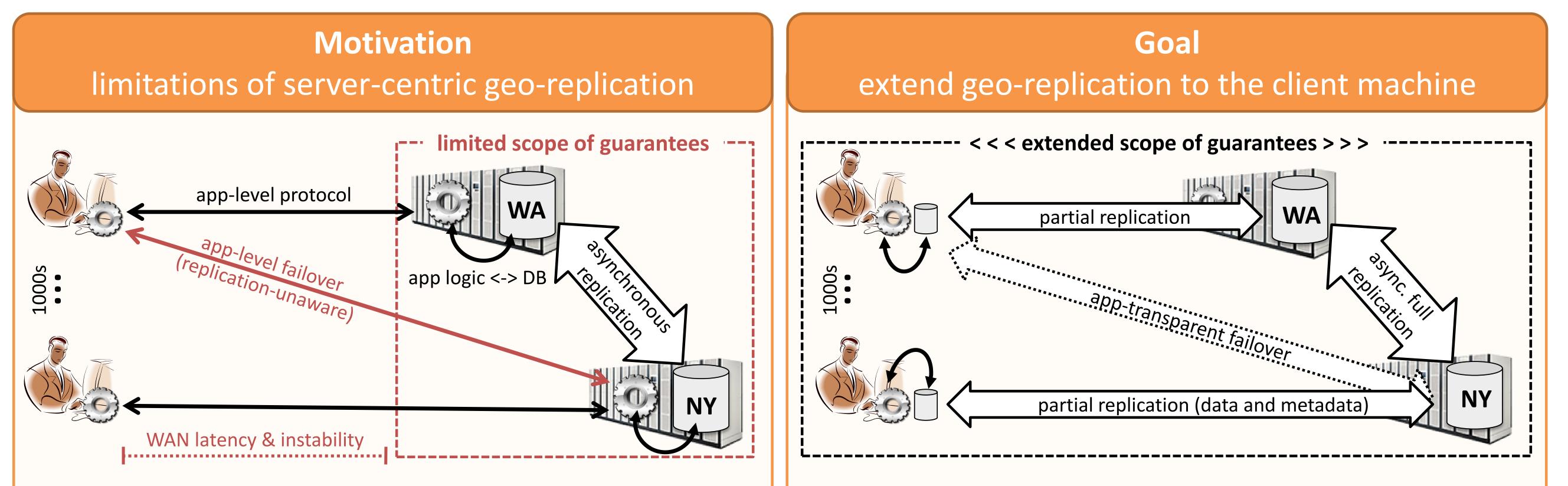
Geo-replication integrated all the way to the client machine

Marek Zawirski¹ Annette Bieniusa² Valter Balegas³ Sérgio Duarte³ Carlos Baquero⁴ Marc Shapiro¹ Nuno Preguiça³ ¹ Inria & UPMC-LIP6 ² U. Kaiserslautern ³ U. Nova de Lisboa ⁴ INESC Tec. & U. Minho



Ad-hoc client-side caching – today's solution plagued with issues:

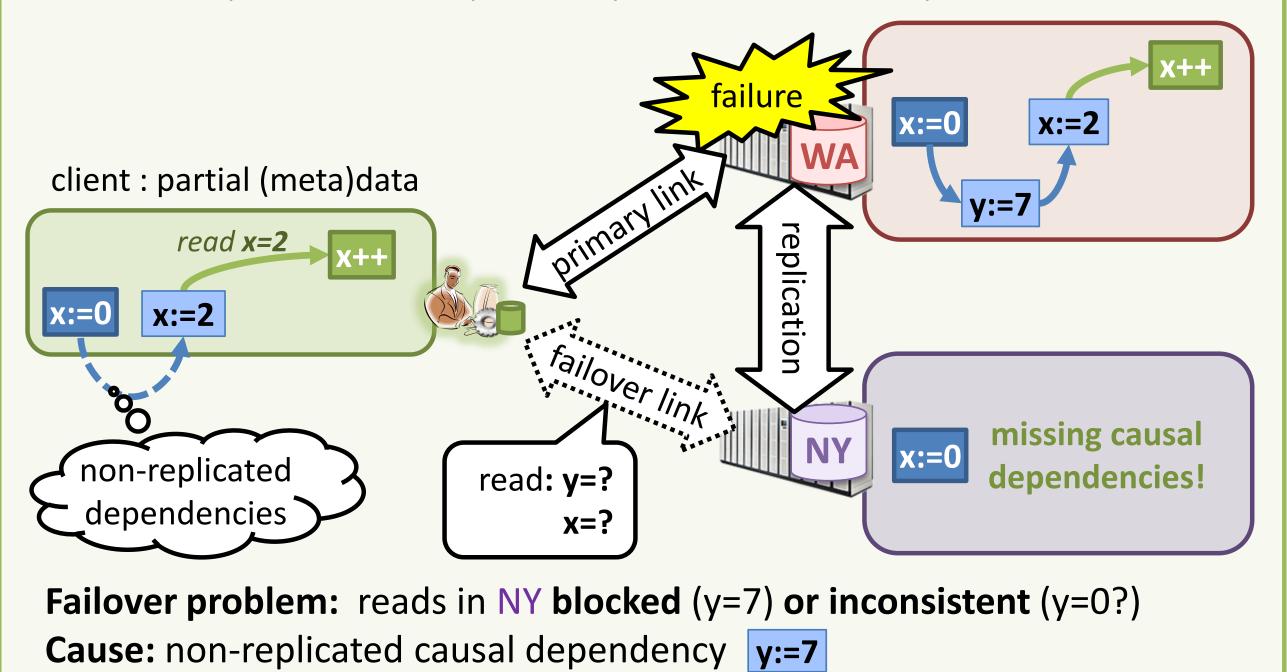
- => Error-prone application-level logic
- => Inconsistent on partial cache misses or failures (no metadata/updates)

Integrated solution – expectations:

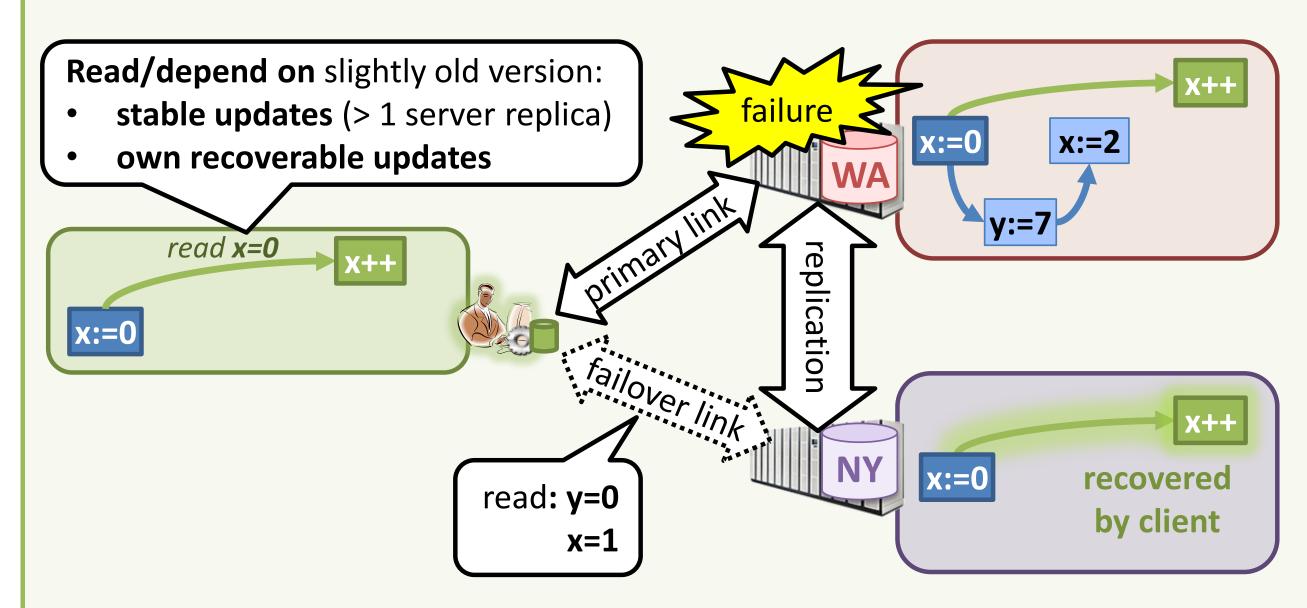
- Lower latency and improved availability for some operations
- (Causally) consistent access to partial replicas despite faults

Problem (1) with naïve approach liveness of causal consistency w/o full (meta)data

example execution: replicated updates and causal dependencies



Multi-versioned approach read stable updates of all clients + own updates

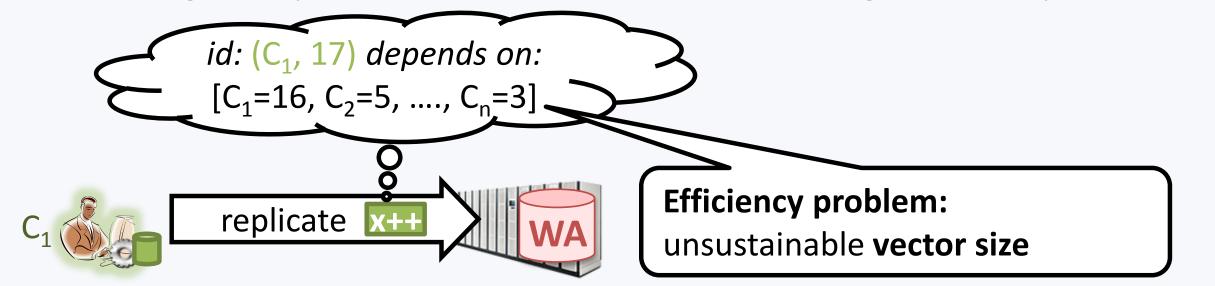


- Consistent access on failover vs. inconsistent in asynchronous systems
- No added WAN latency vs. high latency in quorum-synchronous systems
- Low staleness increase: ≤ 1% more stale reads under contention

Problem (2) with naïve approach inefficient or insufficient metadata

Approach A

client-assigned update id + Version Vectors encoding causal dependencies



Approach B server-assigned update id + any efficient encoding depends on: WA=77, NY=58] **VA,79** replicate **Safety problem:** >= 1 update id eplicatio multiple execution of non-idempotent updates failover e.g. transient WA fault X++ x+=2? \mathbf{OO} X++

Hybrid approach separated concerns: update identity and summary

update lifecycle

