

Middleware for High Availability and Scalability in Multi-Tier and Service-Oriented Architectures

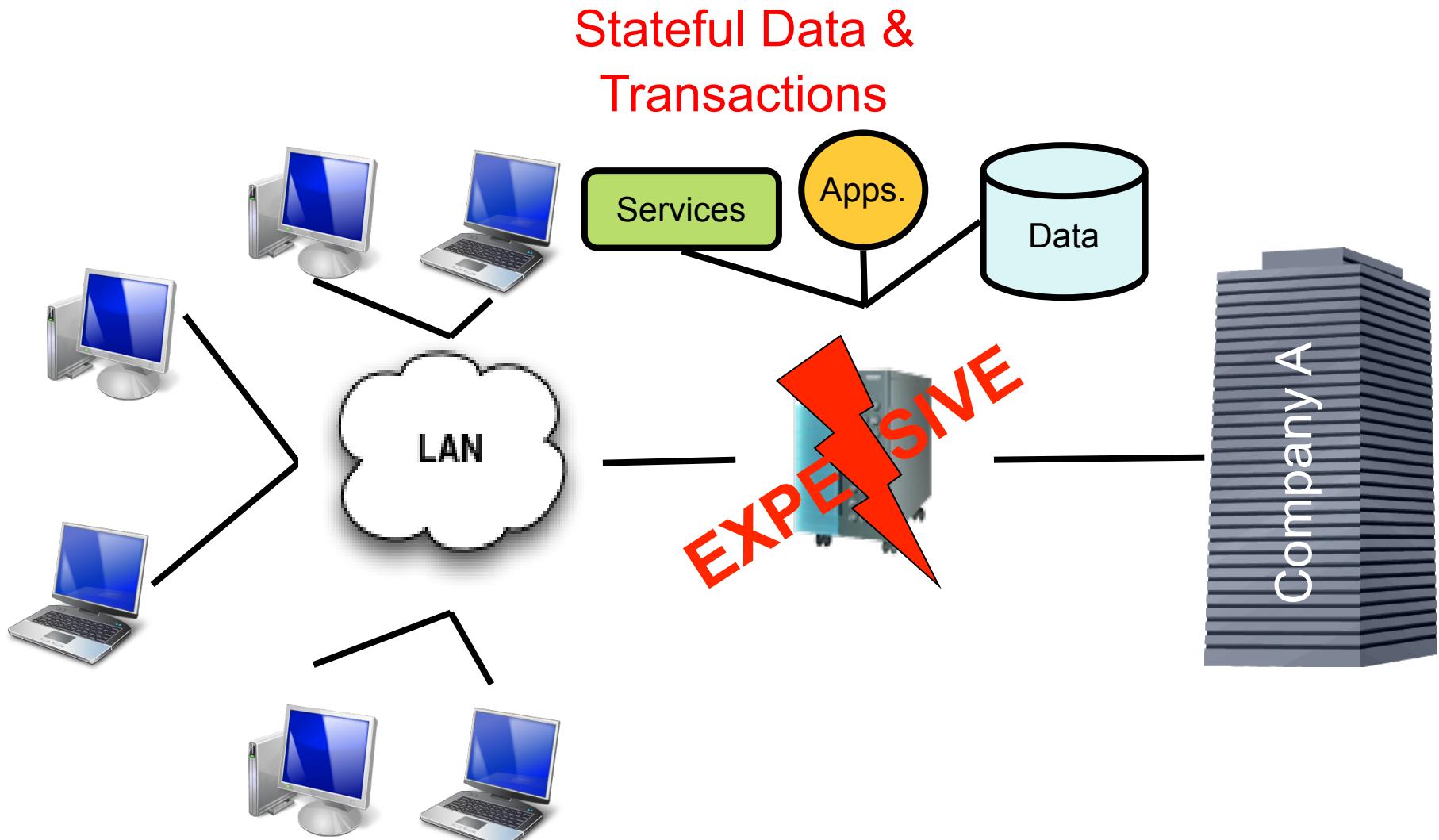
© Francisco Pérez-Sorrosal

Advisor: Marta Patiño-Martínez

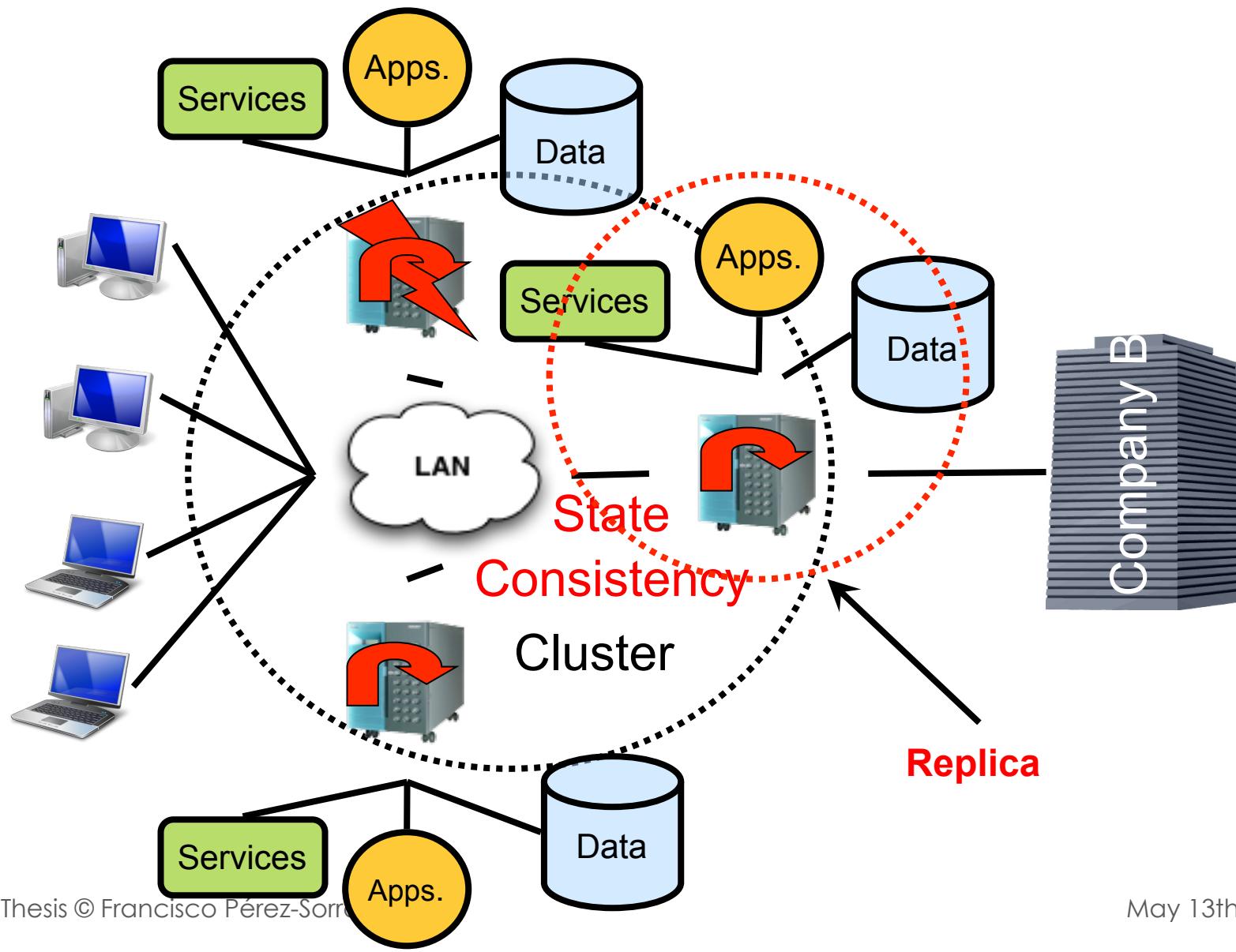
Distributed Systems Laboratory
(DSL/LSD)

Universidad Politécnica de Madrid
Madrid, Spain

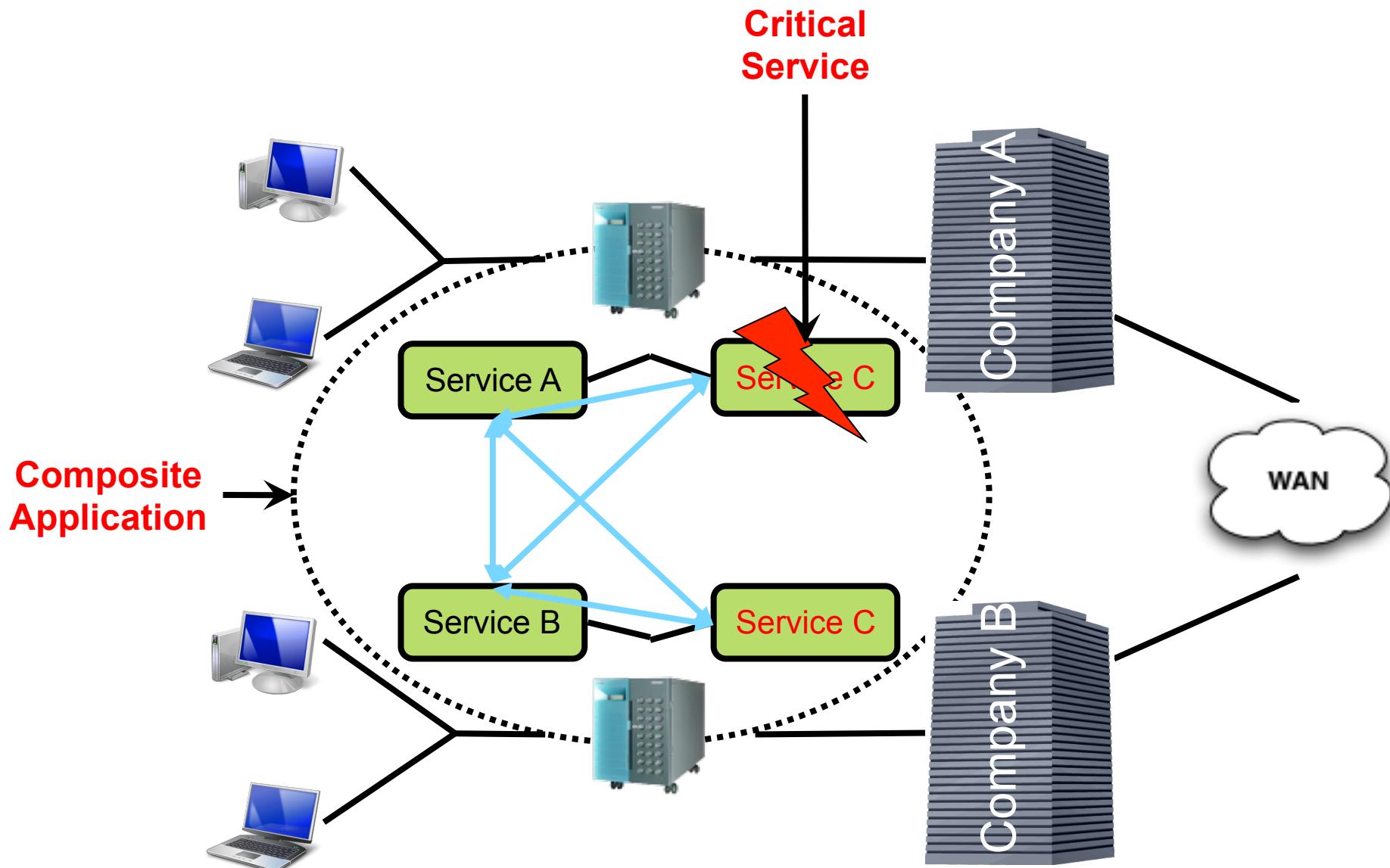
Motivation



Motivation



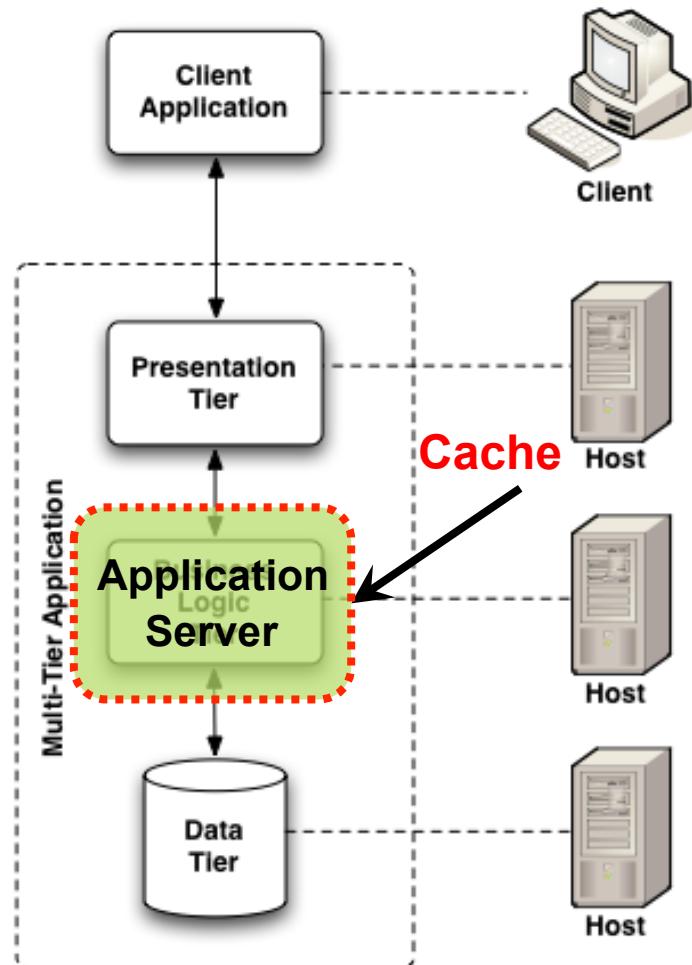
Motivation



Outline

- **High Availability (HA) and Scalability in Multi-Tier Architectures**
 - Protocols for High Availability in MTAs
 - A Protocol for HA and Scalability in MTAs
- High Availability in Service-Oriented Architectures
 - WS-Replication Framework
- Conclusion
- Publications

Multi-tier Architectures: Motivation

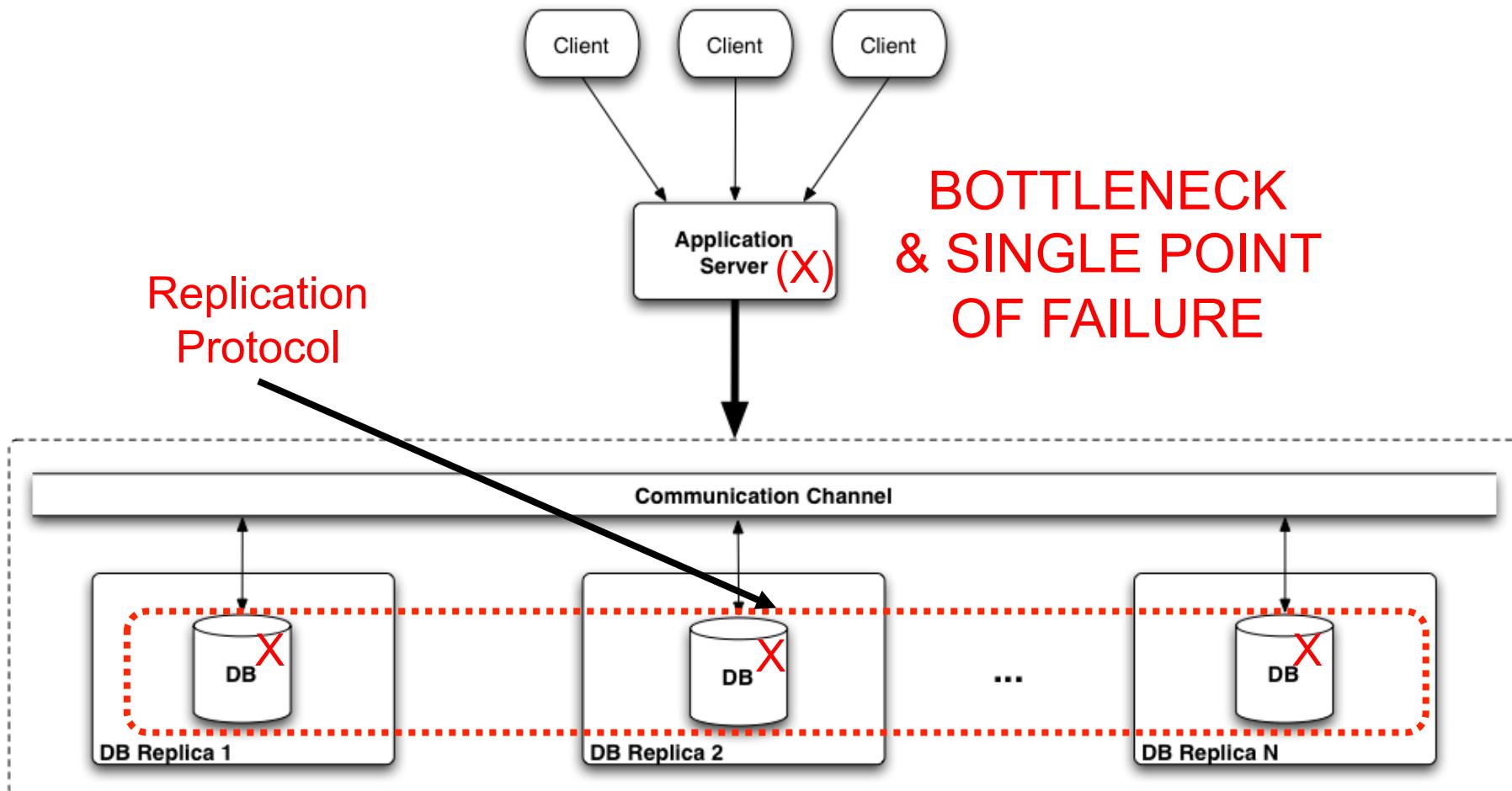


- **Great success of MTAs**
 - CORBA, .NET & J(2)EE
- **Cache requires concurrency control**
 - Serializability
 - Synchronization with the underlying database
- **Many databases provide**
 - Classical isolation levels + Snapshot Isolation

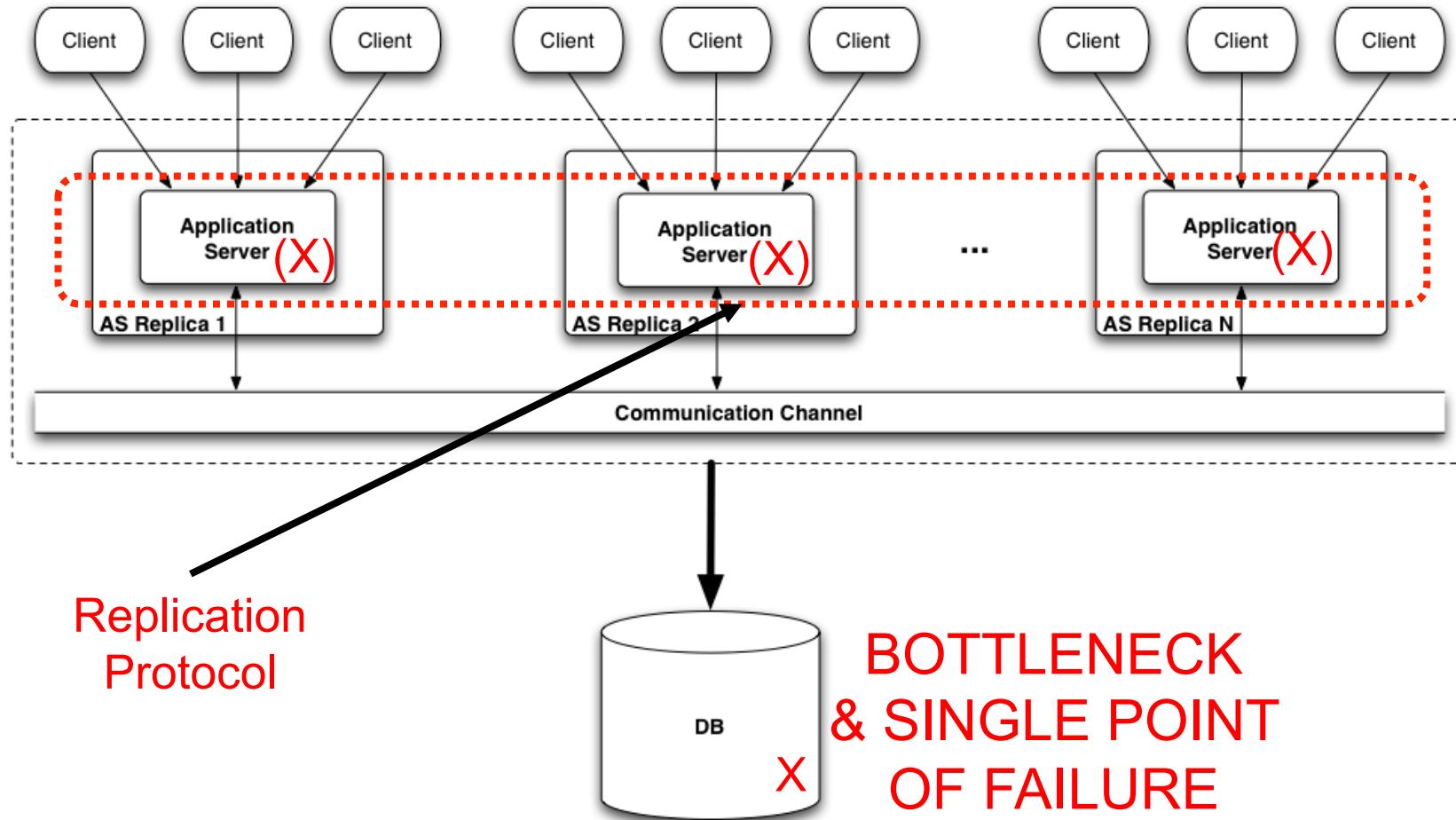
HA and Scalability in MTAs: Context

- **J2EE application servers**
 - **Transactional Services:**
 - ACID Transactions ([JTA](#))
 - Advanced Transactions ([Activity Service](#))
 - Our implementation available at <http://jass.objectweb.org>
 - **Component Model:** Enterprise Java Beans ([EJBs](#))
 - Stateless (SLSB) and Stateful (SFSB) Session Beans, Entity Beans (EB) & Message-Driven (MDB)
- **When replicating EJBs:**
 - SLSBs & MDBs don't keep state => NOT Replicated
 - SFSB beans **keep client-related state** across requests
 - EBs represent persistent data in a datasource

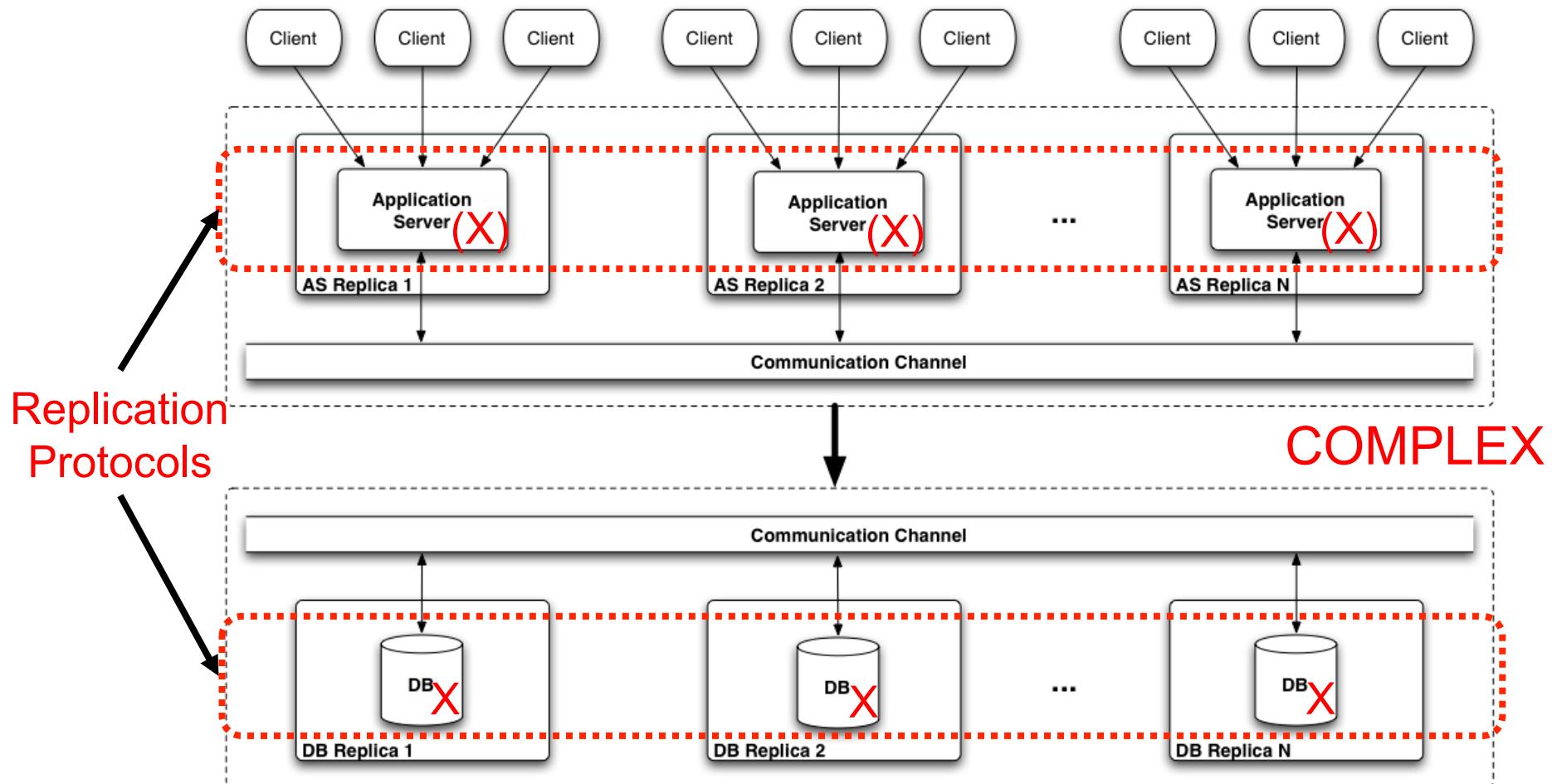
Horizontal Replication (DB Replication)



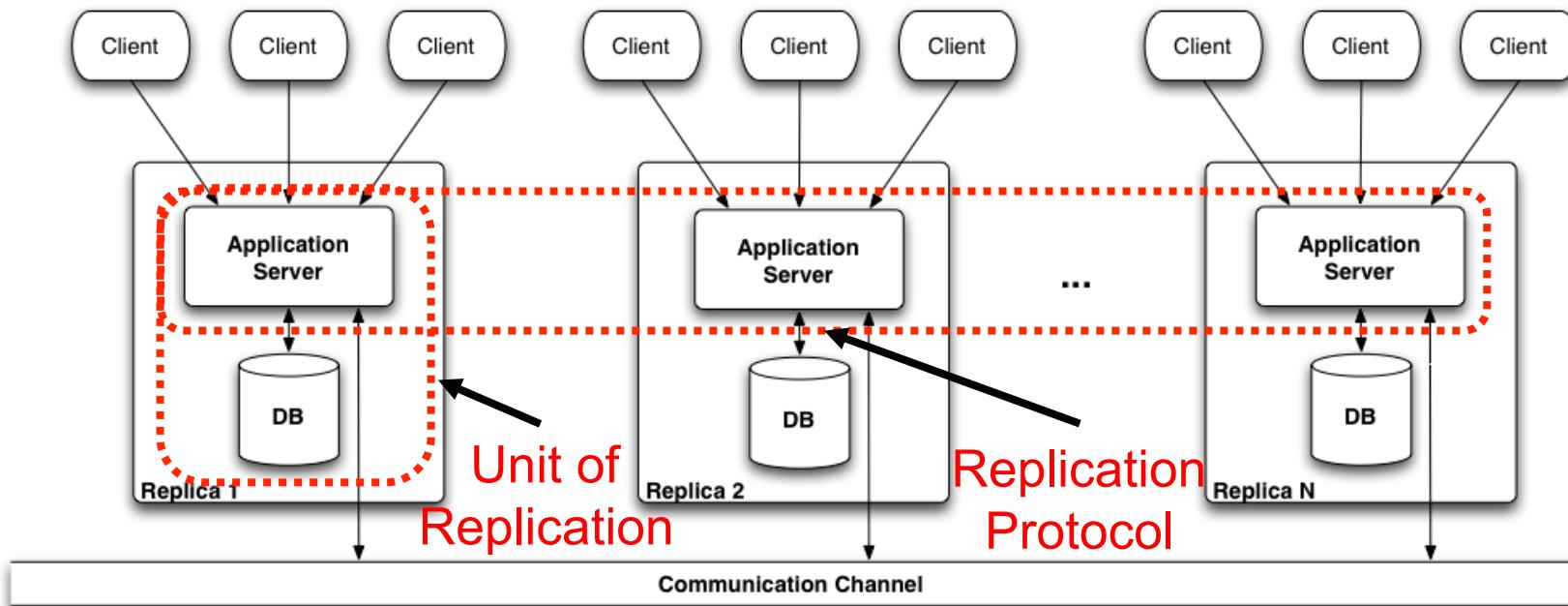
Horizontal Replication (App. Server Replication)



Horizontal Replication (AS and DB Replication)



Our Solution: Vertical Replication



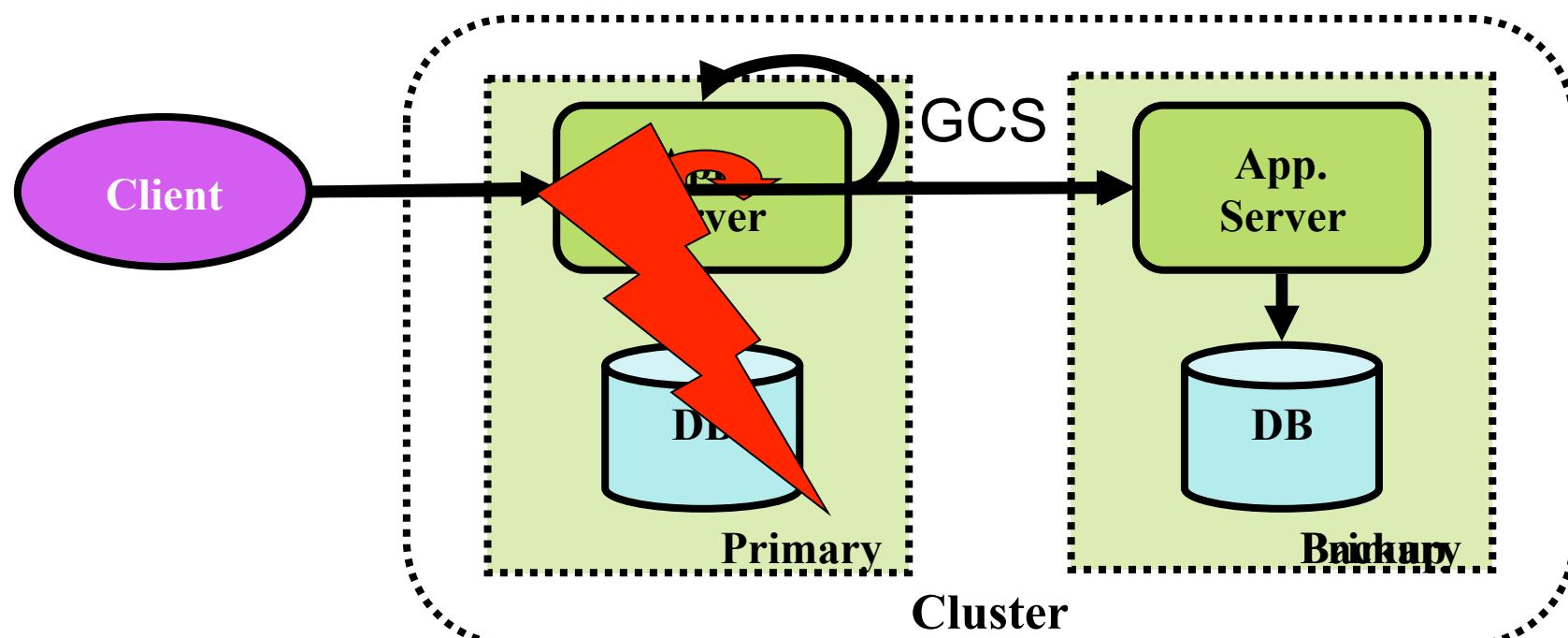
- **No single** bottleneck
- **No single** point of failure
- **Only one** replication protocol

Outline

- **High Availability (HA) and Scalability in Multi-Tier Architectures**
 - **Protocols for High Availability in MTAs**
 - A Protocol for HA and Scalability in MTAs
- High Availability in Service-Oriented Architectures
 - WS-Replication Framework
- Conclusions
- Publications

Protocols for HA in MTAs

- **Consider** session data (**SFSBs**) **and** persistent data (**EBs**)
- **Are** transaction aware & mask failures transparently
- **Approach:** Vertical Replication + Primary-Backup



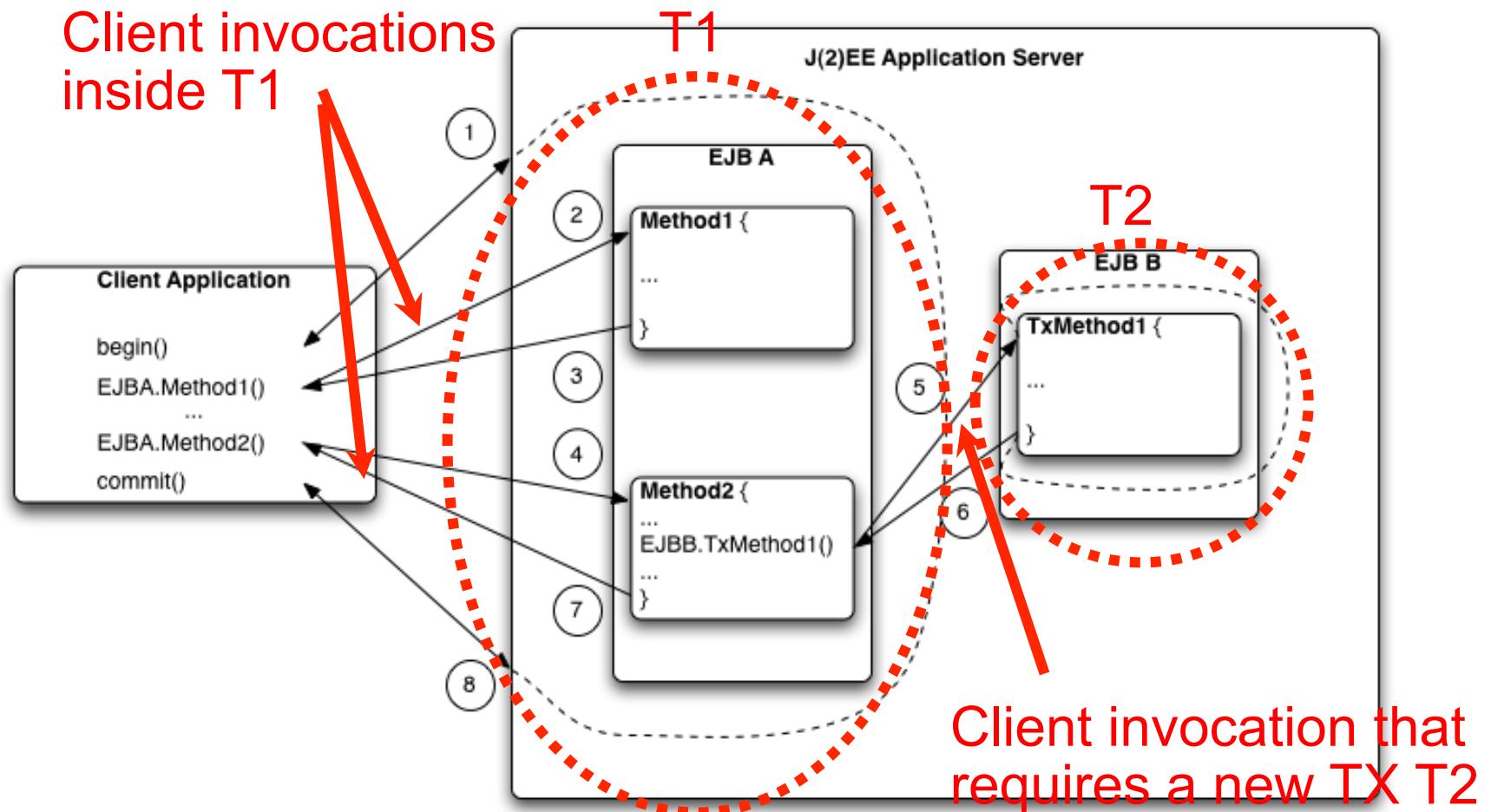
Our protocols offer...

- **Data consistency** in all the replicas
 - Vertical replication + transaction management
 - 1-copy correctness
- **Exactly-once execution**
 - The client performs a request only once and gets the results also only once
- **High available transactions**
 - The replication protocols are transaction-aware
 - Transactions are not aborted if the primary fails
- **Different interaction patterns**
 - 1 Req/1 Tx, N Req / 1 Tx, 1 Req / N Txs and N Req / M Txs

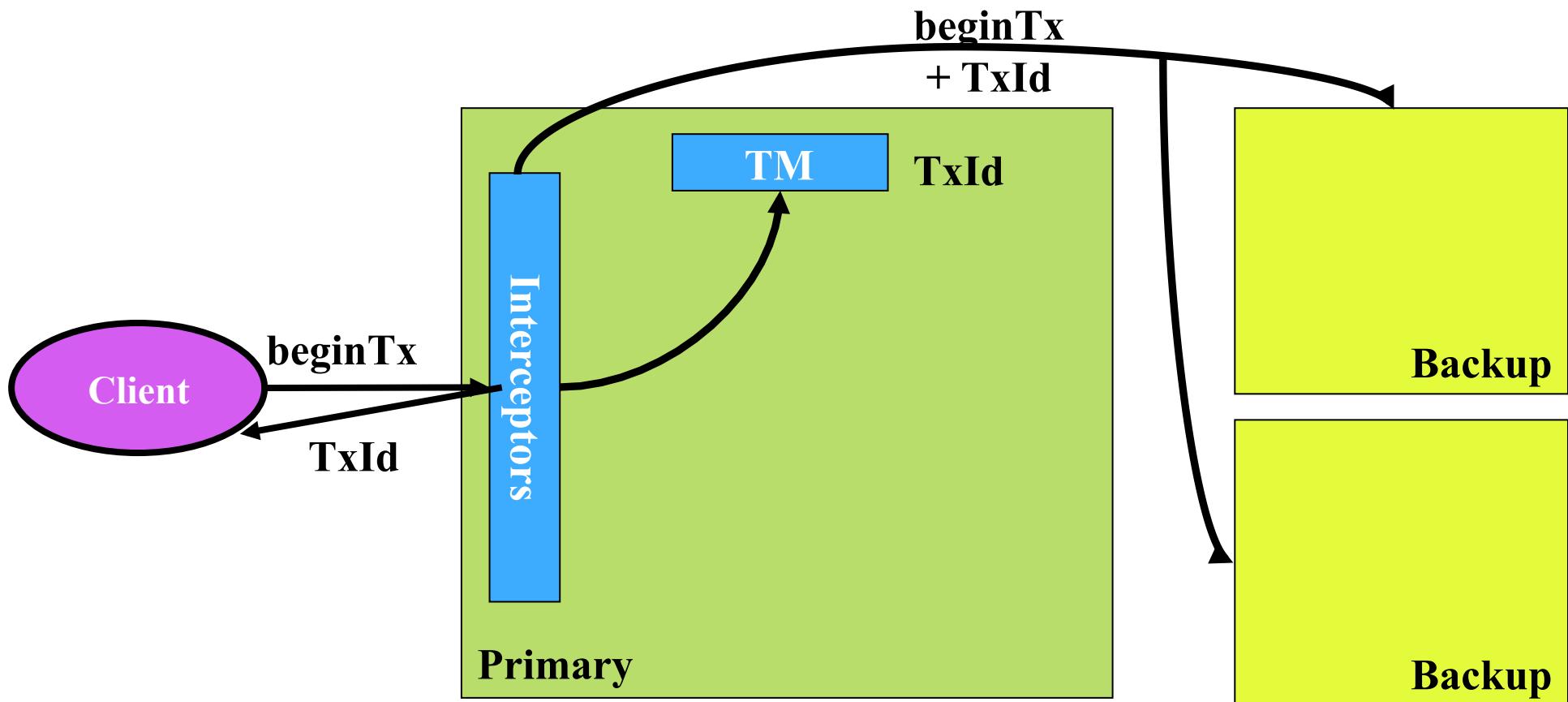
N requests/1 transaction: Goals

- **Support** **transactional conversations**
 - Several client requests inside a single transaction
- **Upon failover**, resume the conversation from the last interaction
 - Do not abort ongoing transactions

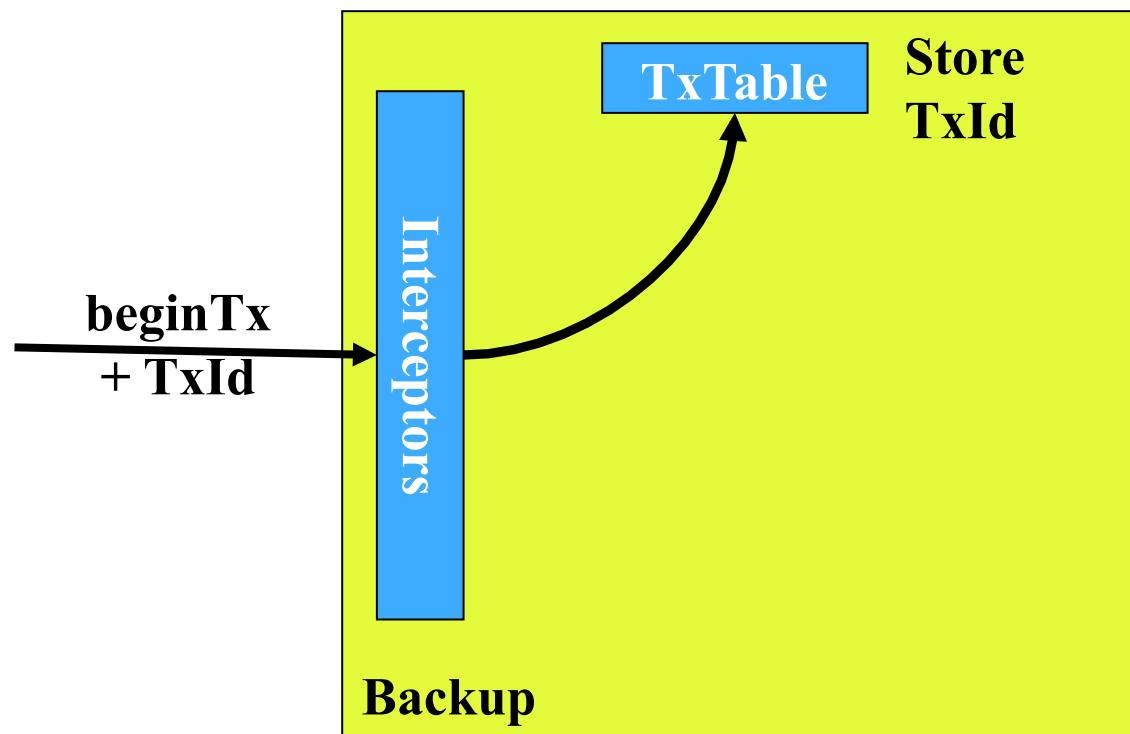
N requests/1 transaction



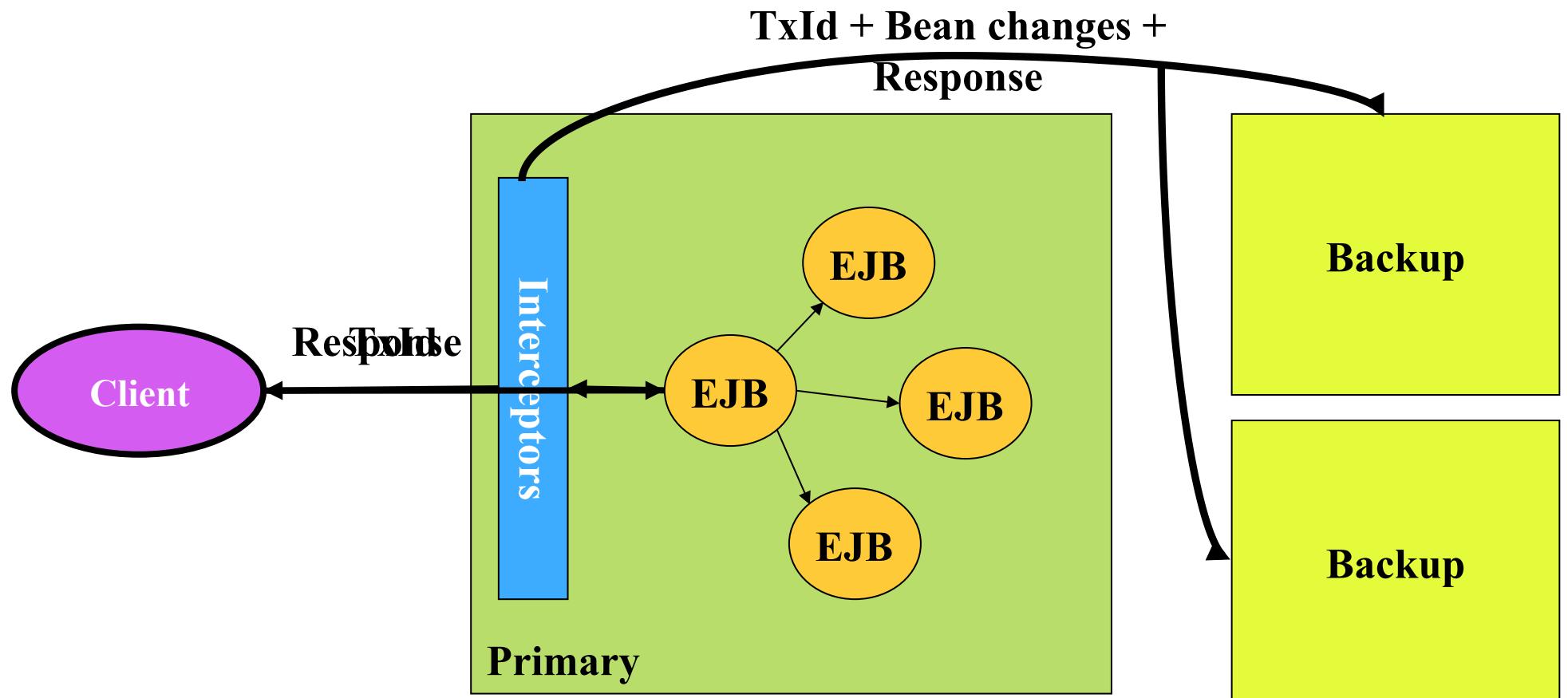
N Req / 1 TX Replication Protocol: Primary (Begin)



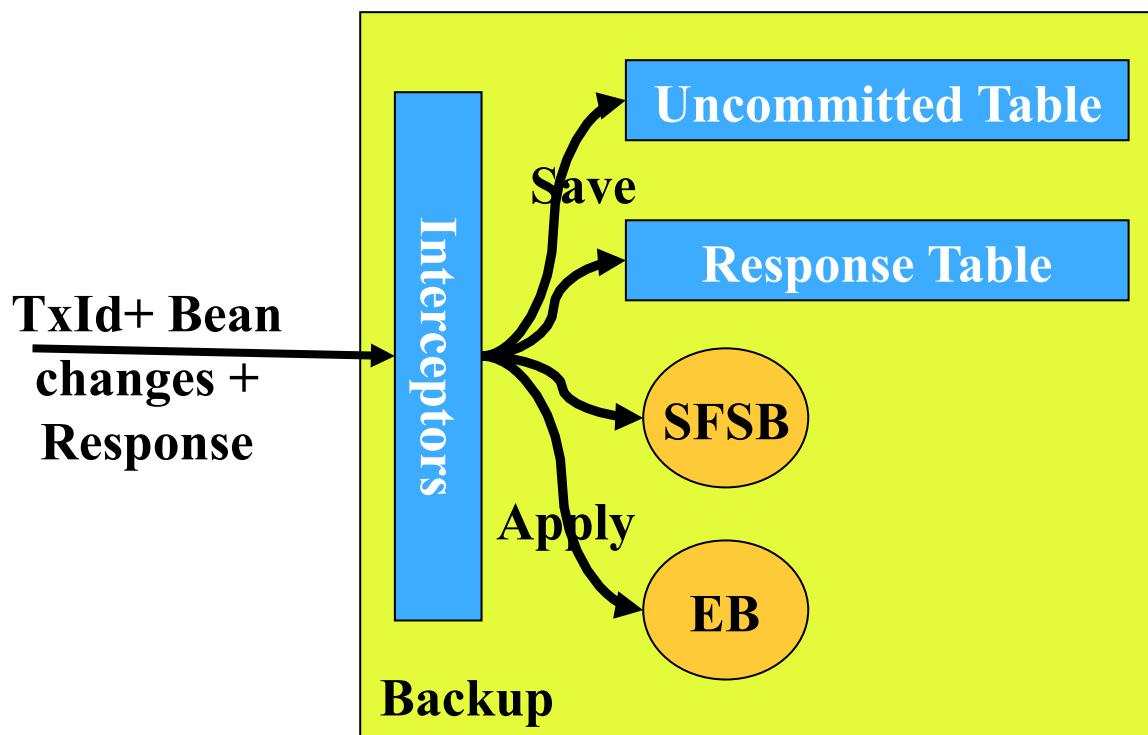
Replication Protocol: Backup (Begin)



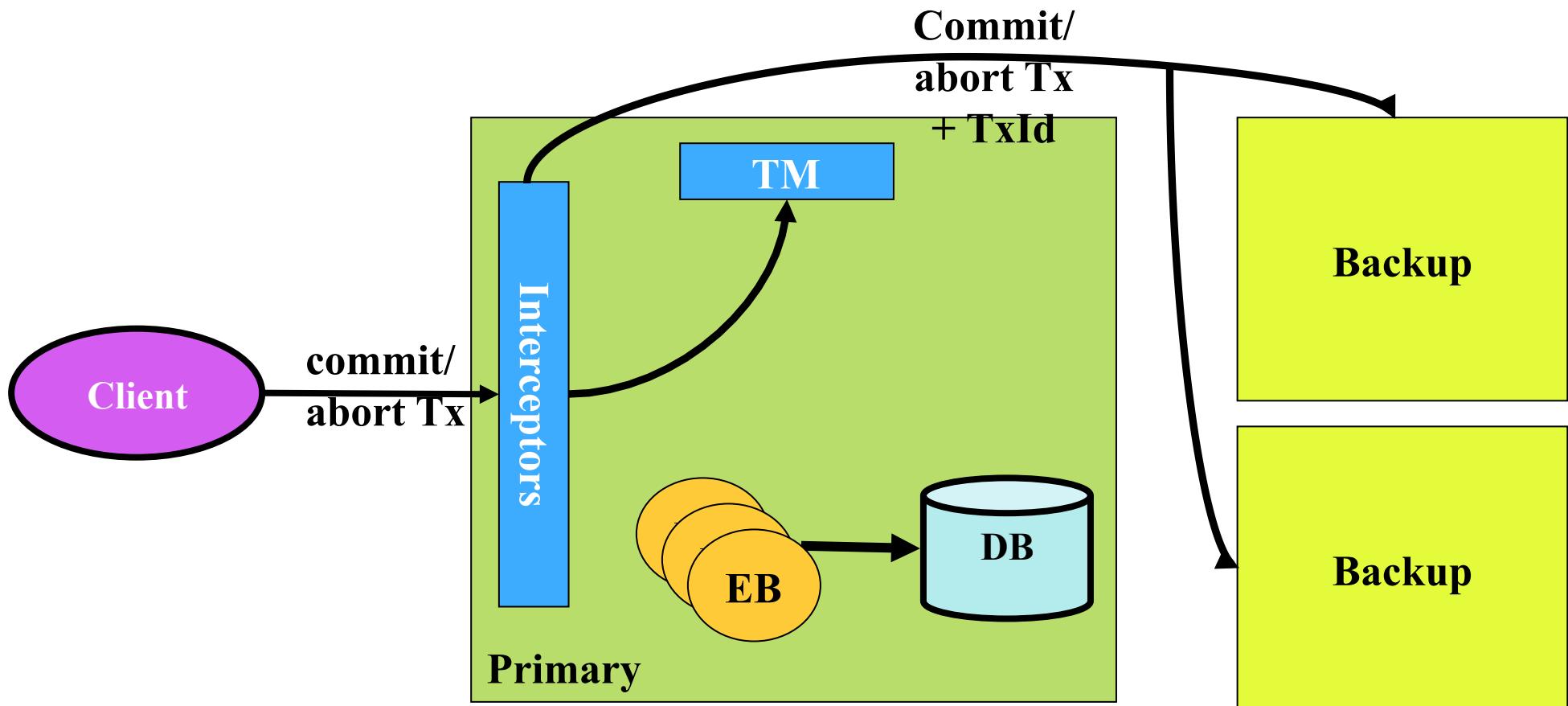
Replication Protocol: Primary (Invocation)



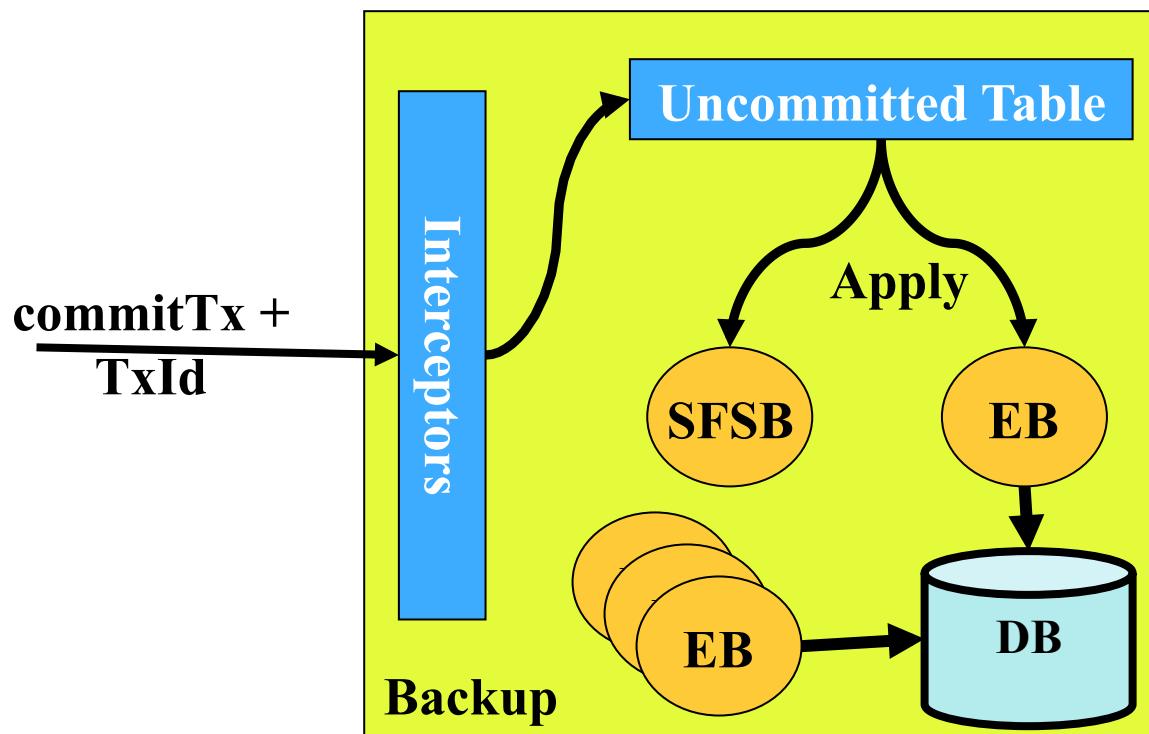
Replication Protocol: Backup (Invocation)



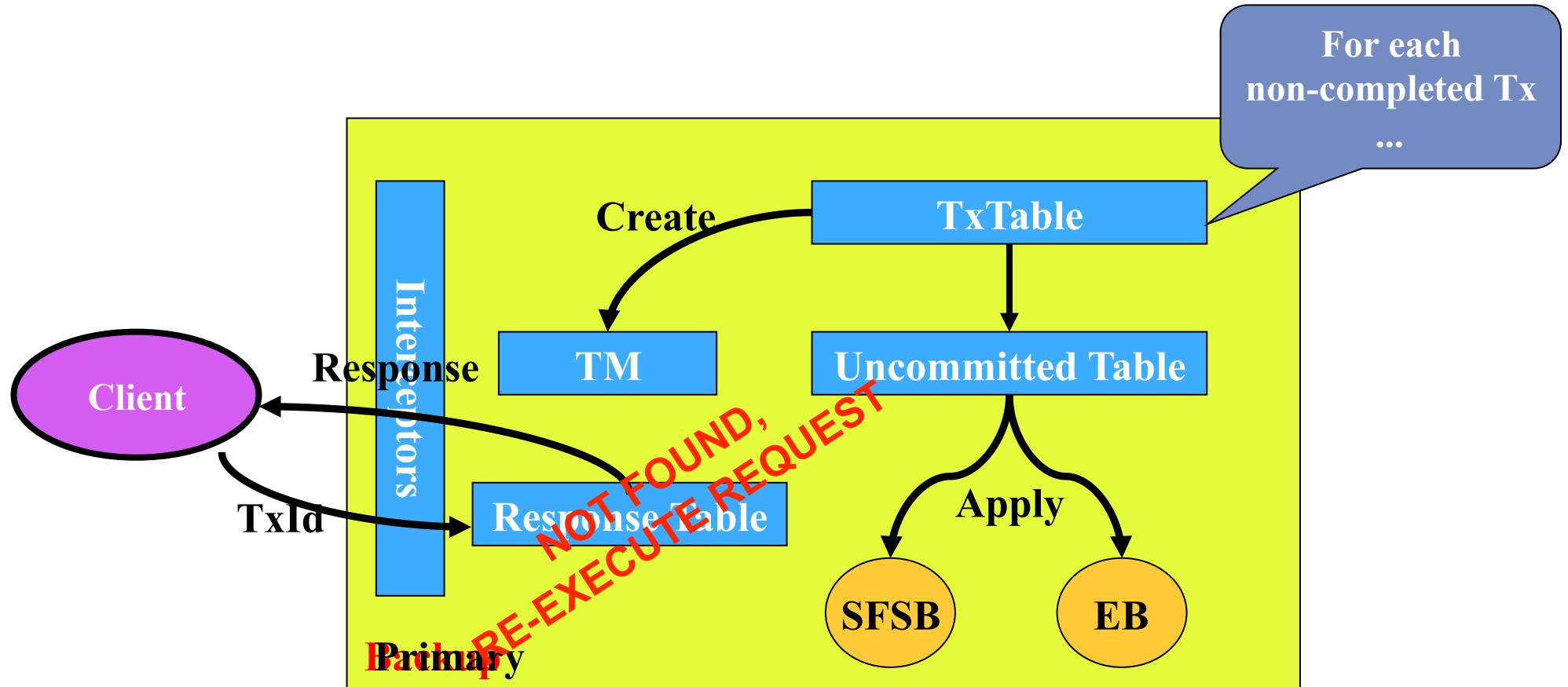
Replication Protocol: Primary (Commit/Abort)



Replication Protocol: Backup (Commit/Abort)



Replication Protocol: Failover

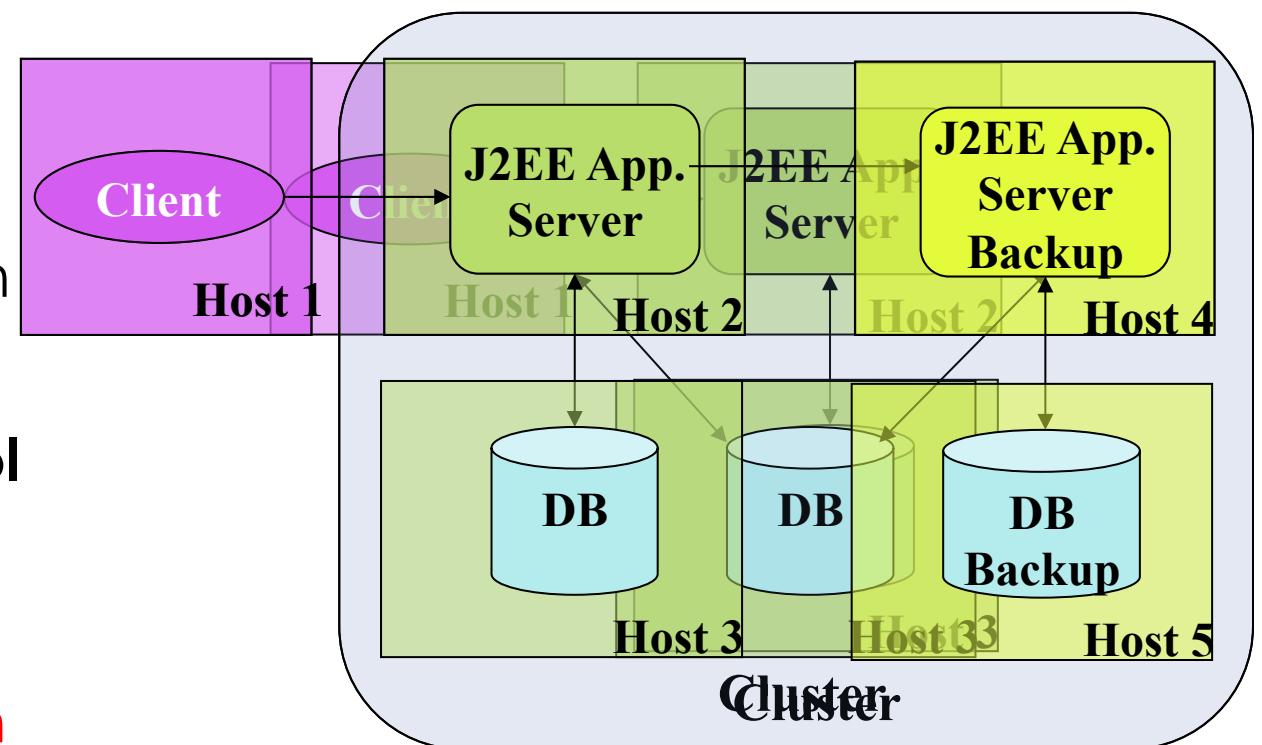


Evaluation: ECPerf

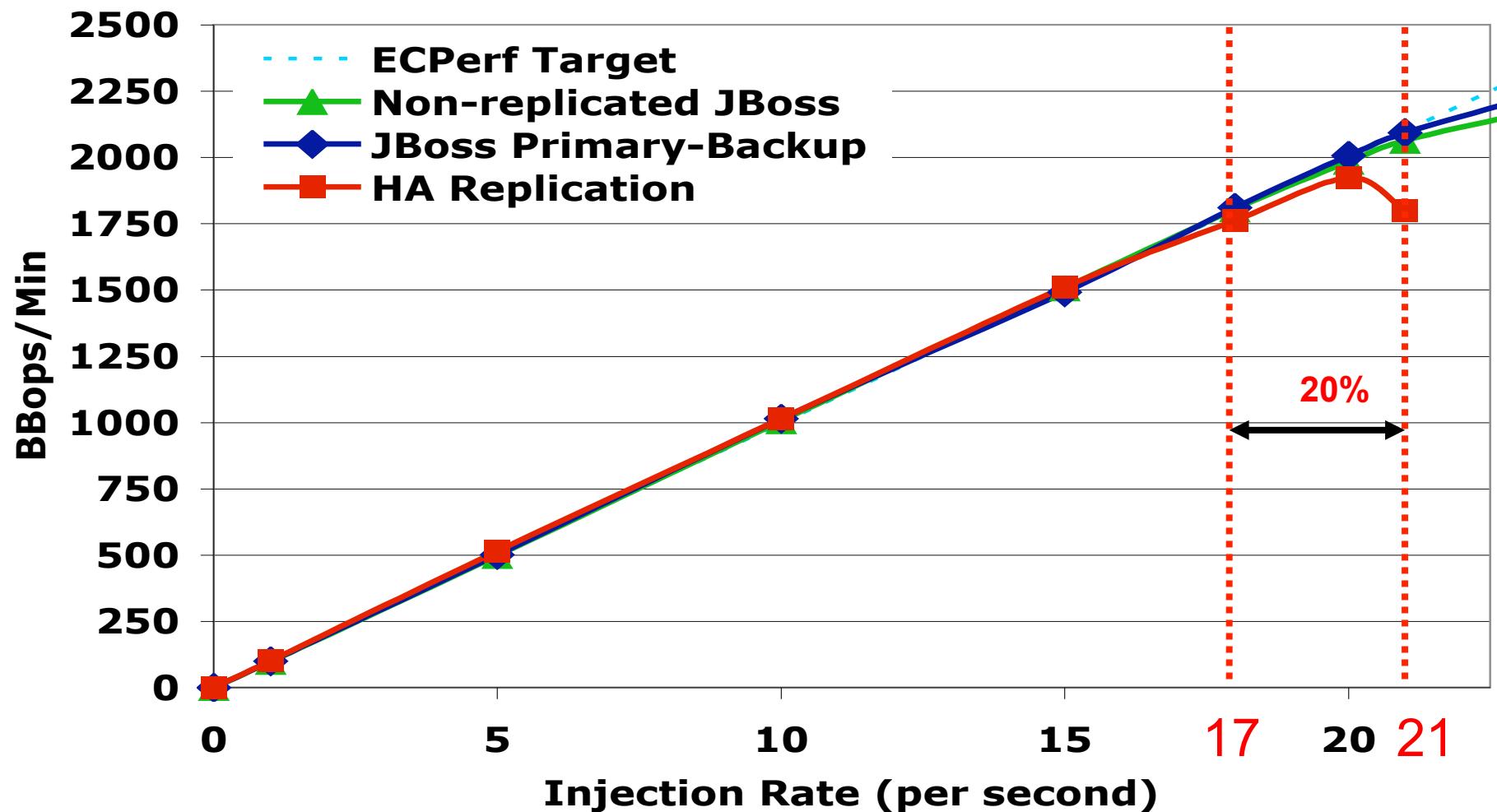
- **Benchmark to evaluate** the throughput and scalability of J2EE Application Servers
- **Emulates** the processes involved in a supply-chain management scenario
- **The load** is measured as the Injection Rate (**IR**)
 - # of clients = IR * 5
- **Throughput** is given in Benchmark Business Operations per Minute (**BBOps/Min**)

Experiment Setup

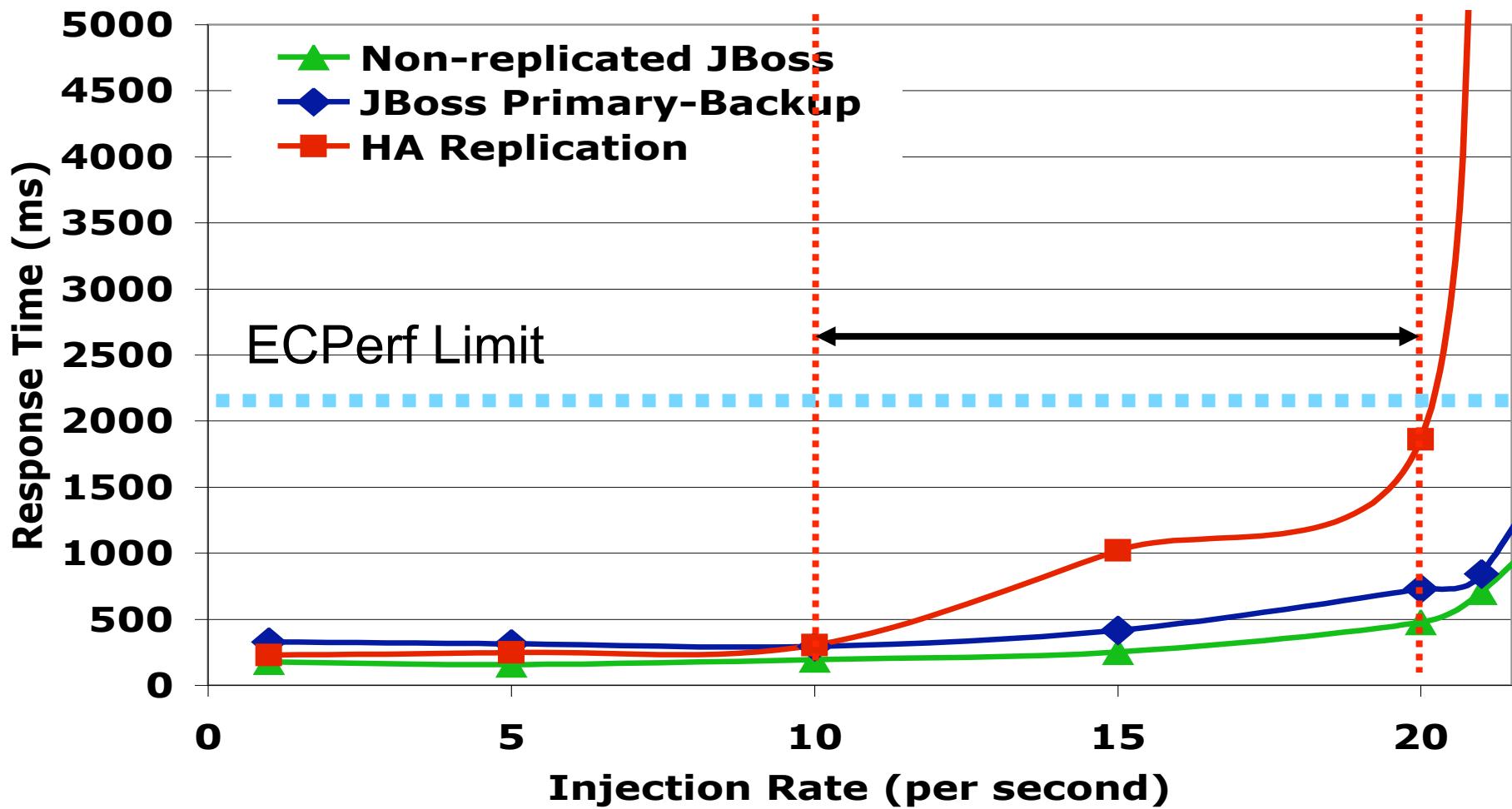
- **JBoss**
 - Non-replicated
- **JBoss Primary-Backup**
 - Only SFSB replication
 - Shared DB
- **Our replication protocol**
 - Primary-Backup + Vertical replication
 - SFSB & EB replication
 - Transaction aware



ECPerf: Throughput



ECPerf: Response Time



Outline

- **High Availability (HA) and Scalability in Multi-Tier Architectures**
 - Protocols for High Availability in MTAs
 - **A Protocol for HA and Scalability in MTAs**
- High Availability in Service-Oriented Architectures
 - WS-Replication Framework
- Conclusions
- Publications

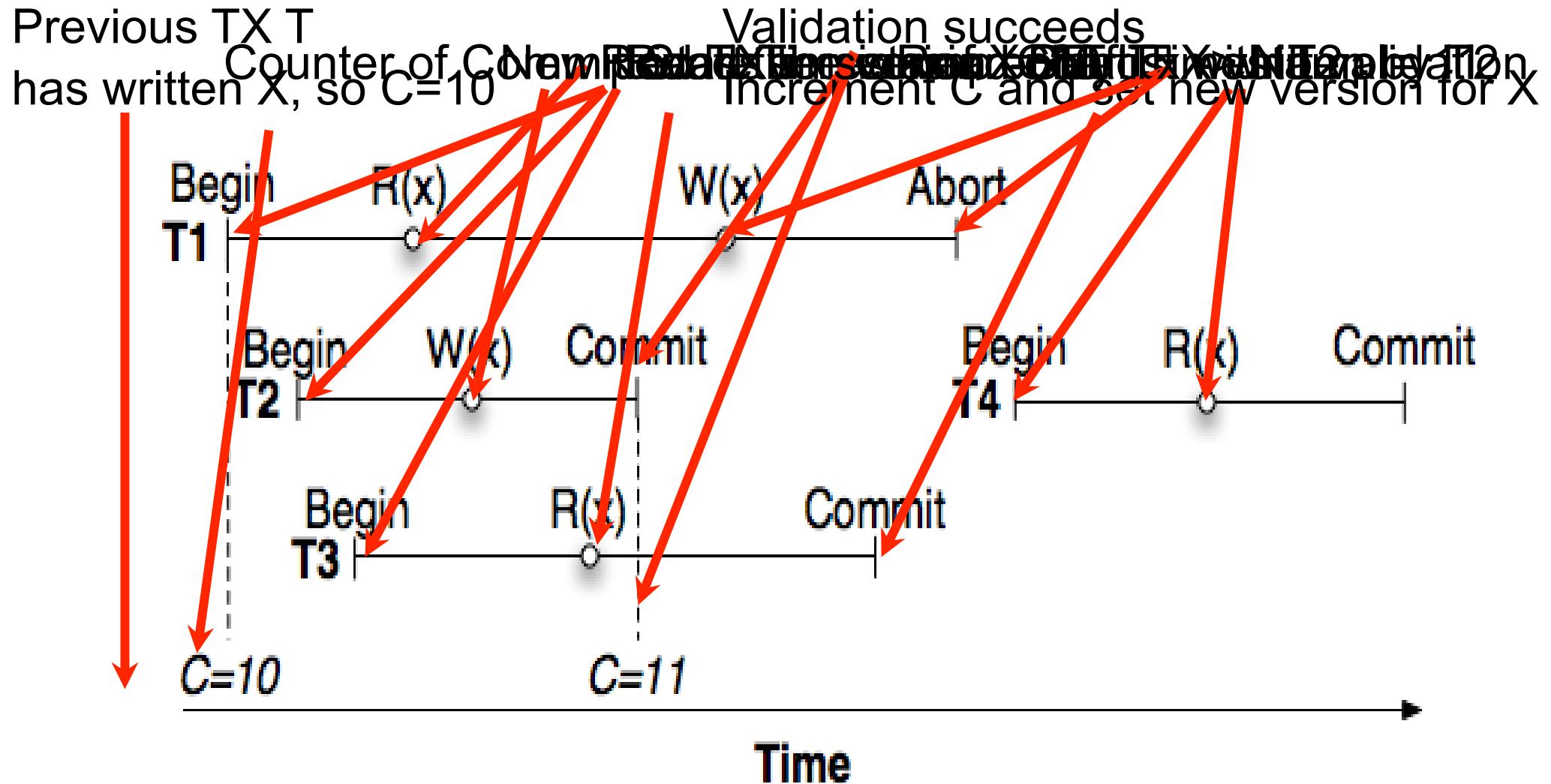
Limitations of Current Middleware for HA in MTAs

- **Mismatch** between isolation at Application Server and DBMS
 - Current application servers do not work correctly with SI databases
- **Snapshot Isolation (SI)** has become the “de-facto” standard isolation level
- **Current middleware** does not scale-out stateful applications consistently

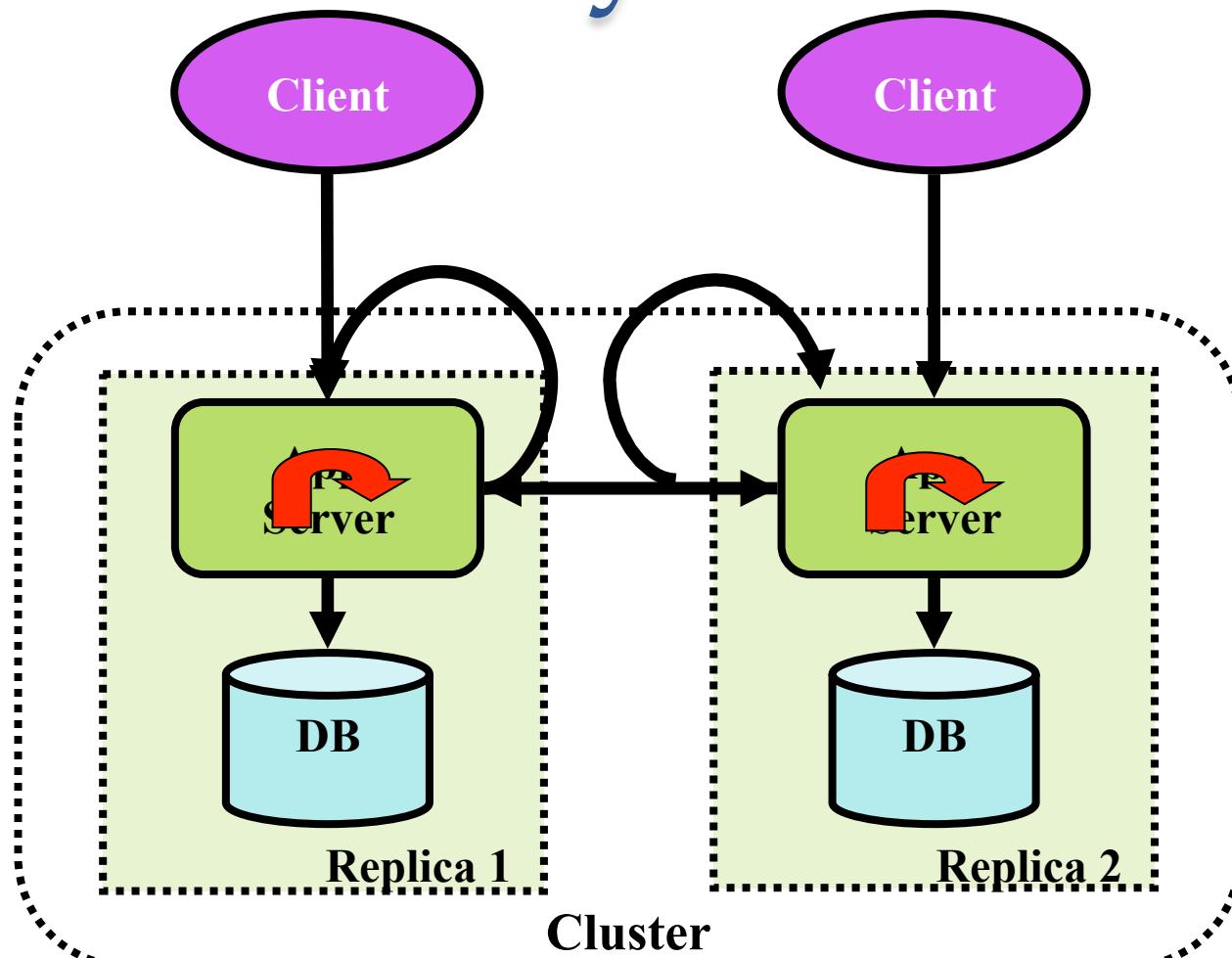
Our Protocol for HA and Scalability in MTAs...

- Is consistent, high available and scalable
- Includes a SI cache at the middleware level for correctness and performance in a single replica
- SI cache is combined with replication for scalability and fault-tolerance in a cluster
- Vertical replication
 - Only-one replication protocol coordinates the execution of transactions and the propagation of changes in a cluster

Snapshot Isolation



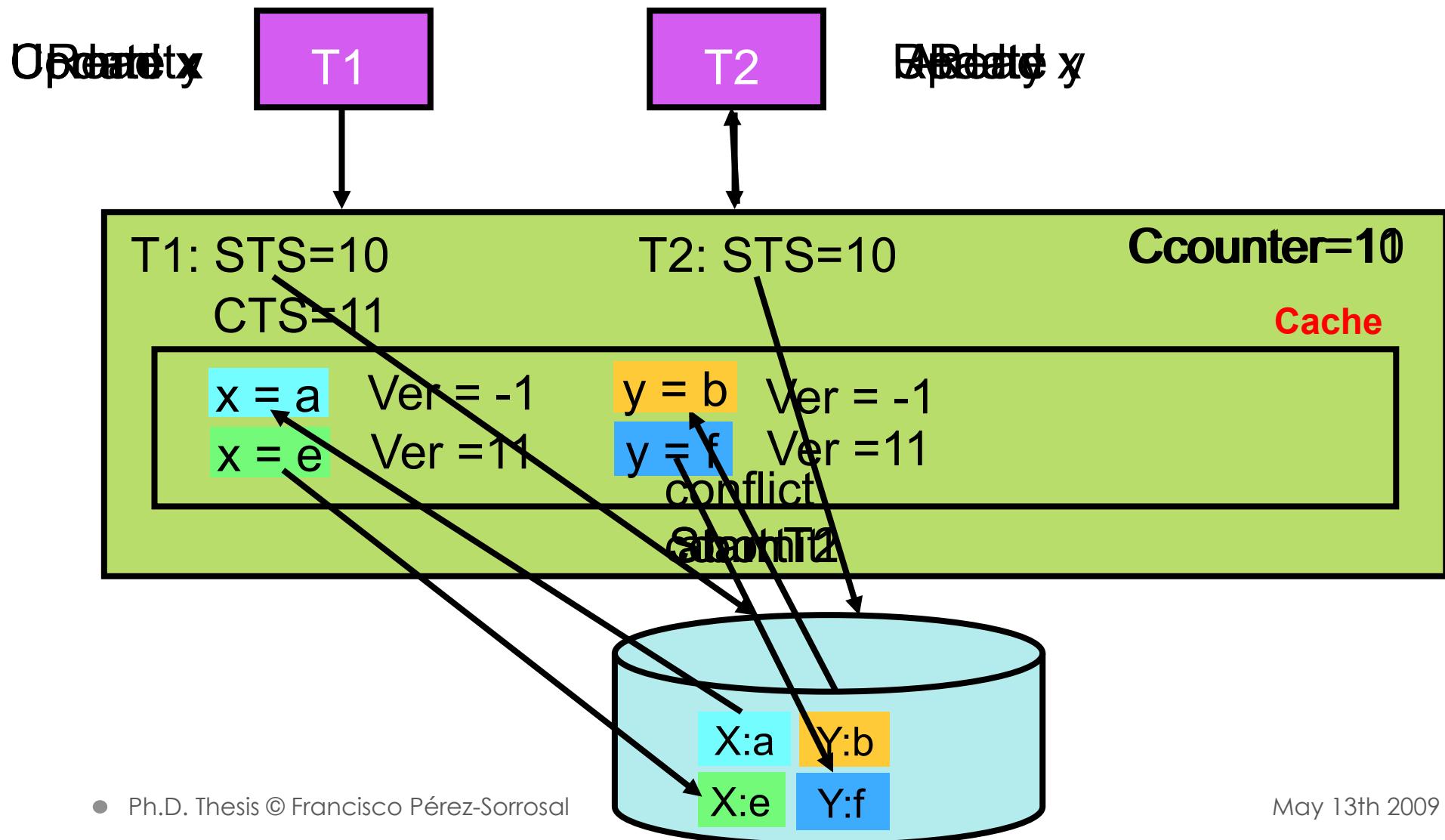
A Protocol for HA and Scalability in MTAs



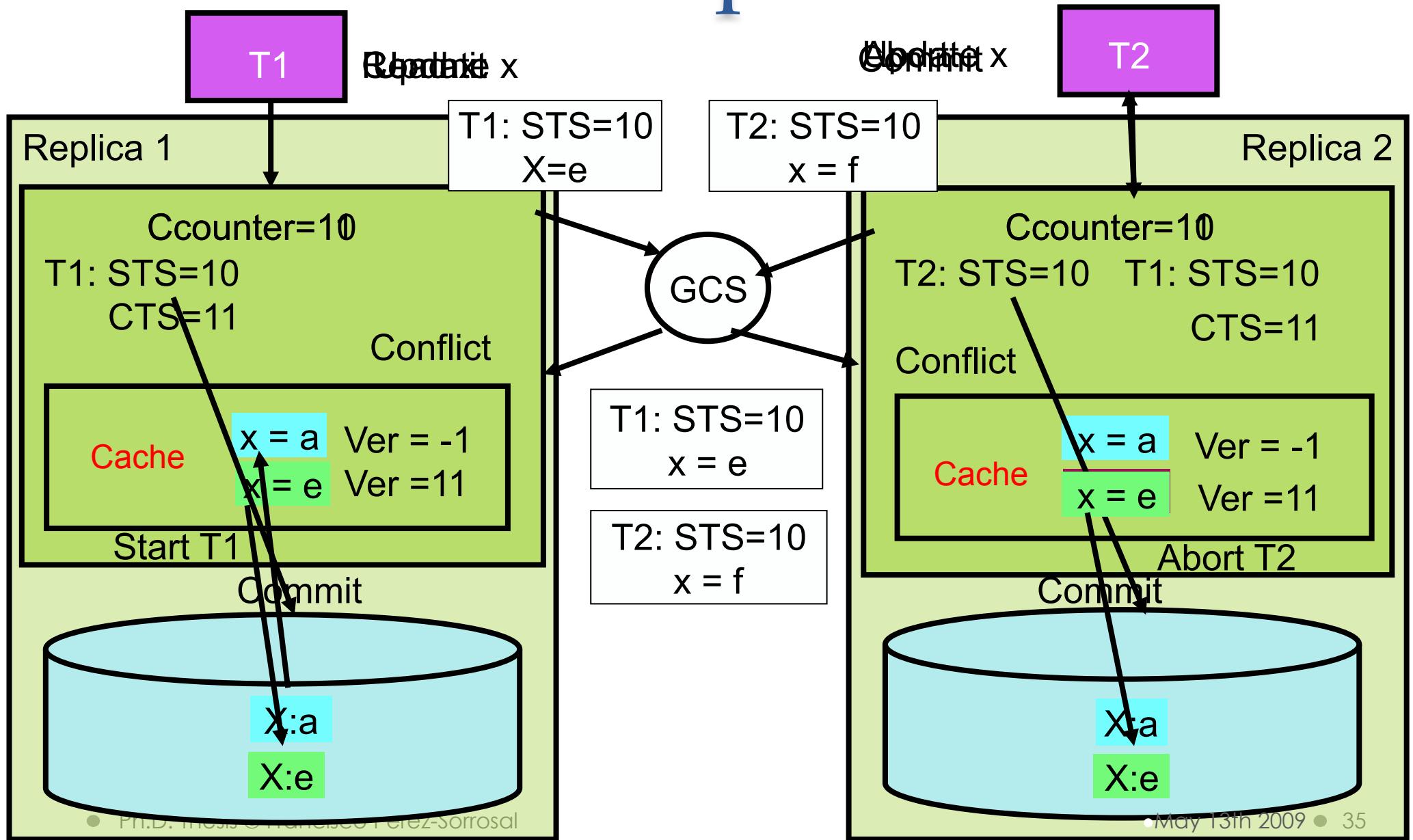
Protocol Features

- **Transactions:** Started at the same time in AS and DBS
- **SI Cache:** Maintains a certain number of versions to
 - Avoid accesses to the DB
 - Guarantee conflict detection
- **Conflicts:**
 - Locally: Detected on-the-fly (Pessimistic)
 - Remotely: Detected on a validation phase
- **Other Issues:**
 - Creation and Deletion of Components (CRUD Ops.)
 - Garbage Collection
 - Session Replication
 - Failure Handling (Transparent failover of clients)
 - Replica Recovery

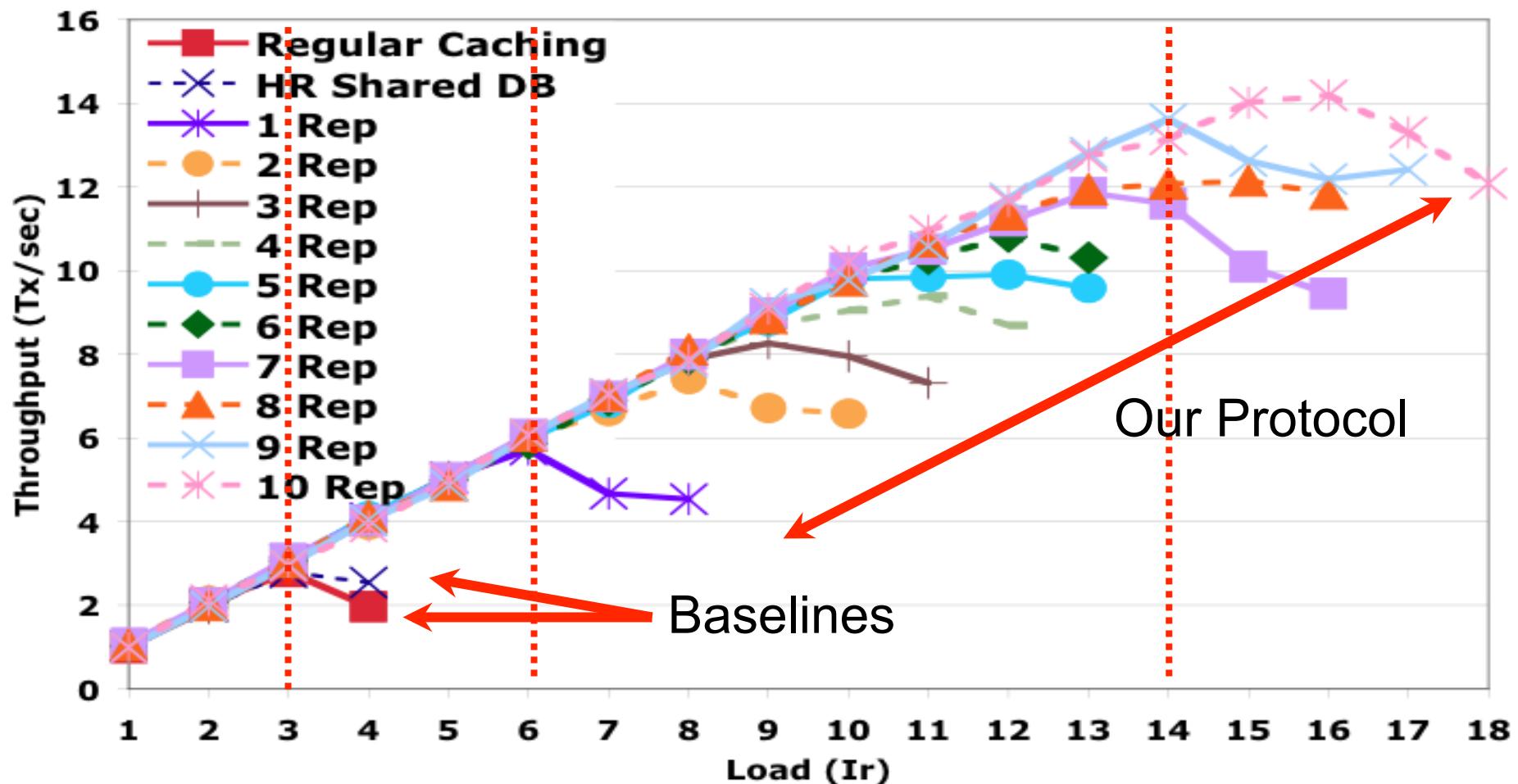
How the Multi-version Cache Works



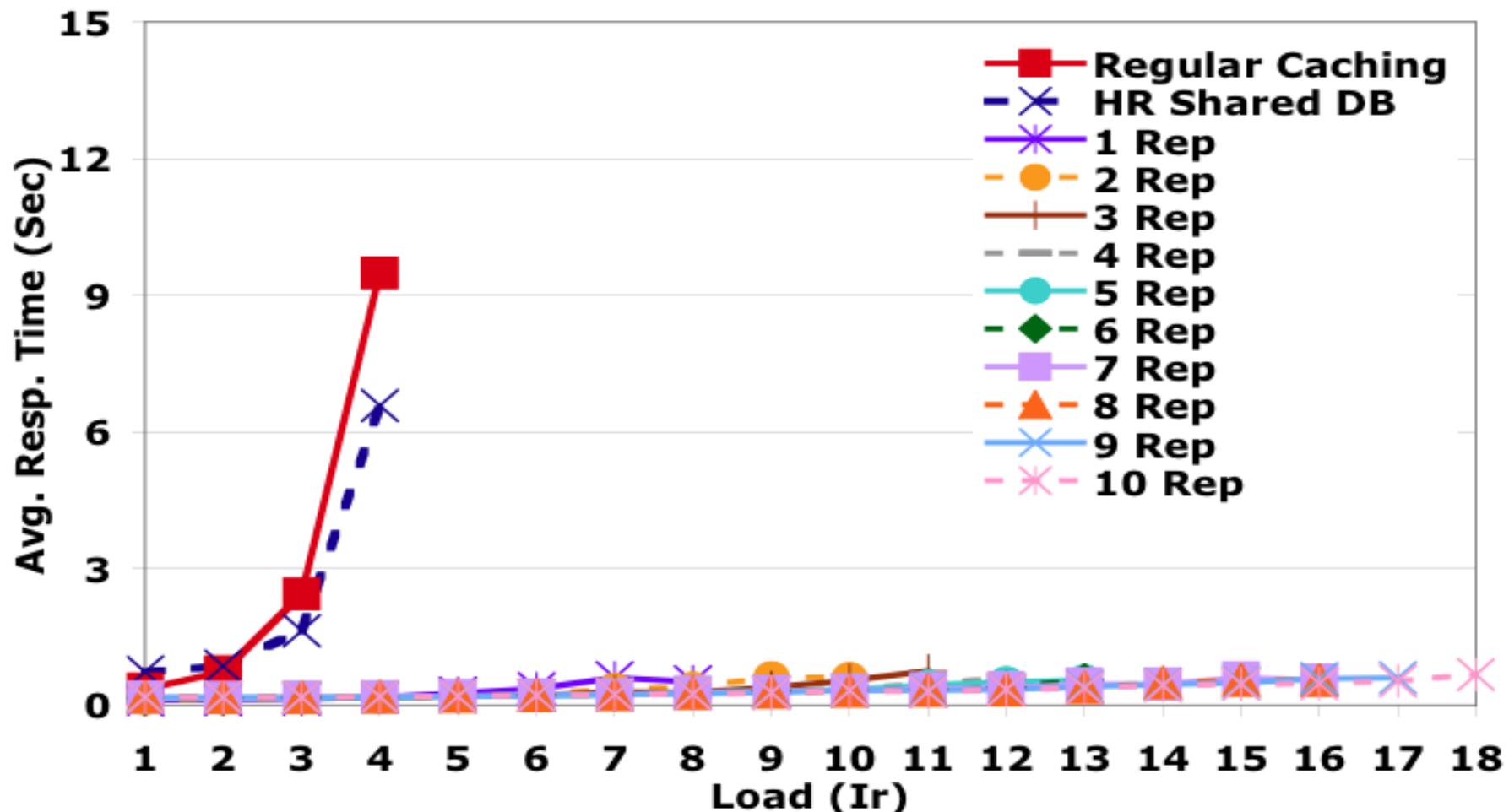
Cache Replication



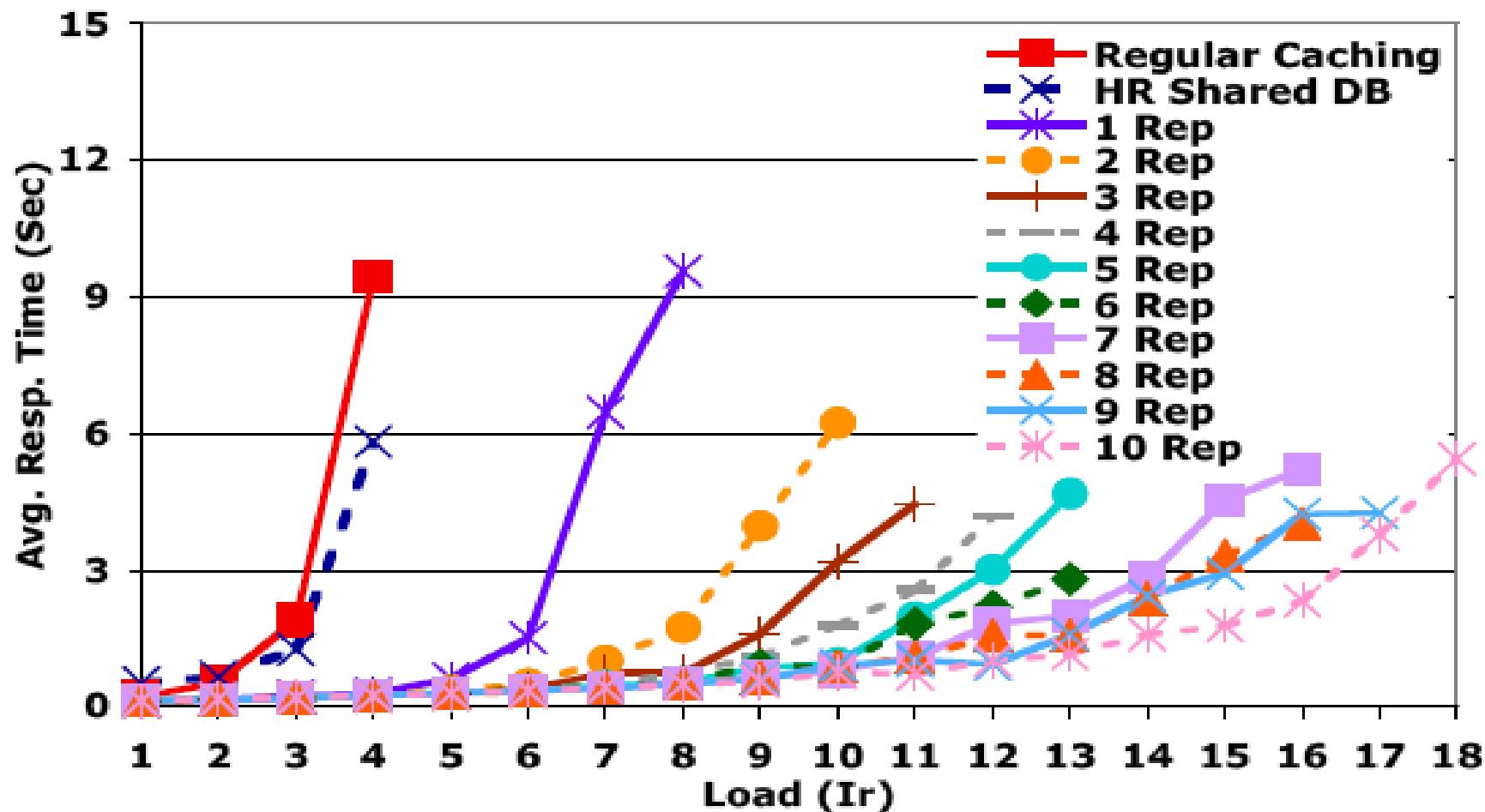
Throughput (SPECjAppServer)



Response Time: Read-only Txn



Response Time: Update Txn



Outline

- High Availability (HA) and Scalability in Multi-Tier Architectures
 - Protocols for High Availability in MTAs
 - A Protocol for HA and Scalability in MTAs
- **High Availability in Service-Oriented Architectures**
 - **WS-Replication Framework**
- Conclusions
- Publications

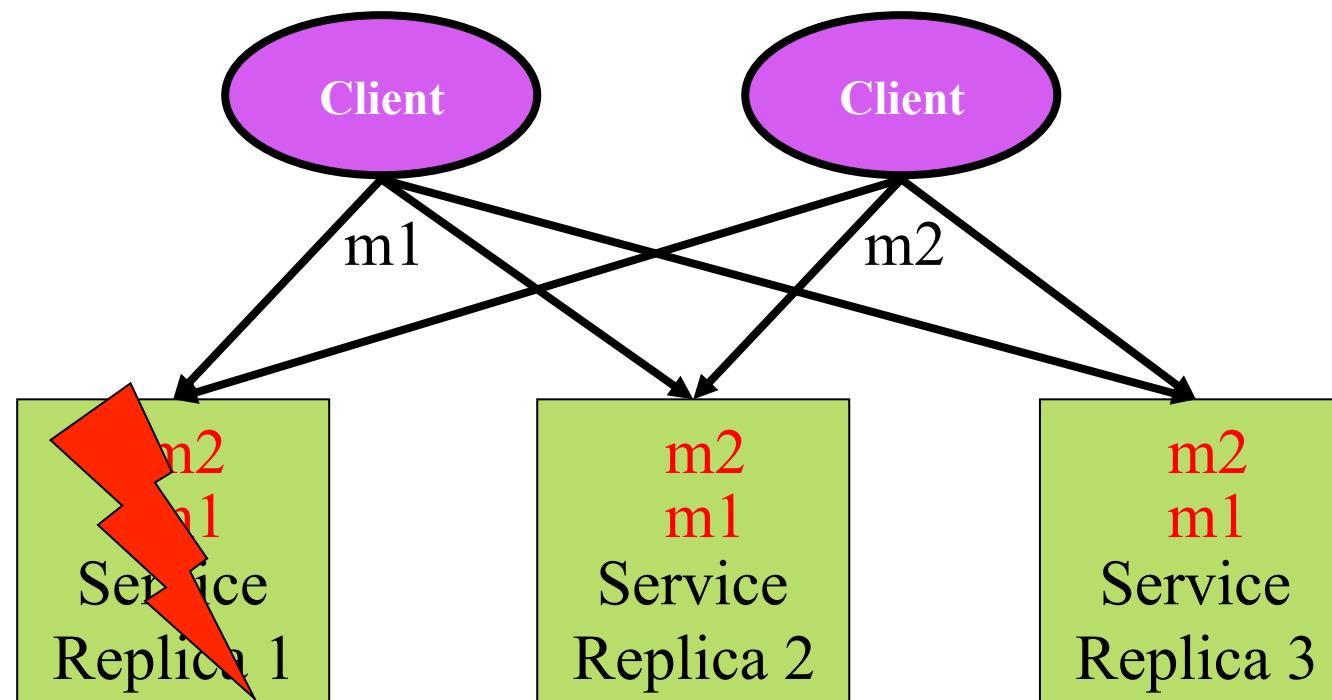
HA in SOA: Motivation

- **Some Web Services are critical** for the interaction among organizations and should remain available despite failures
- **WS-Replication** Framework helps on replicating these **critical Web Services**

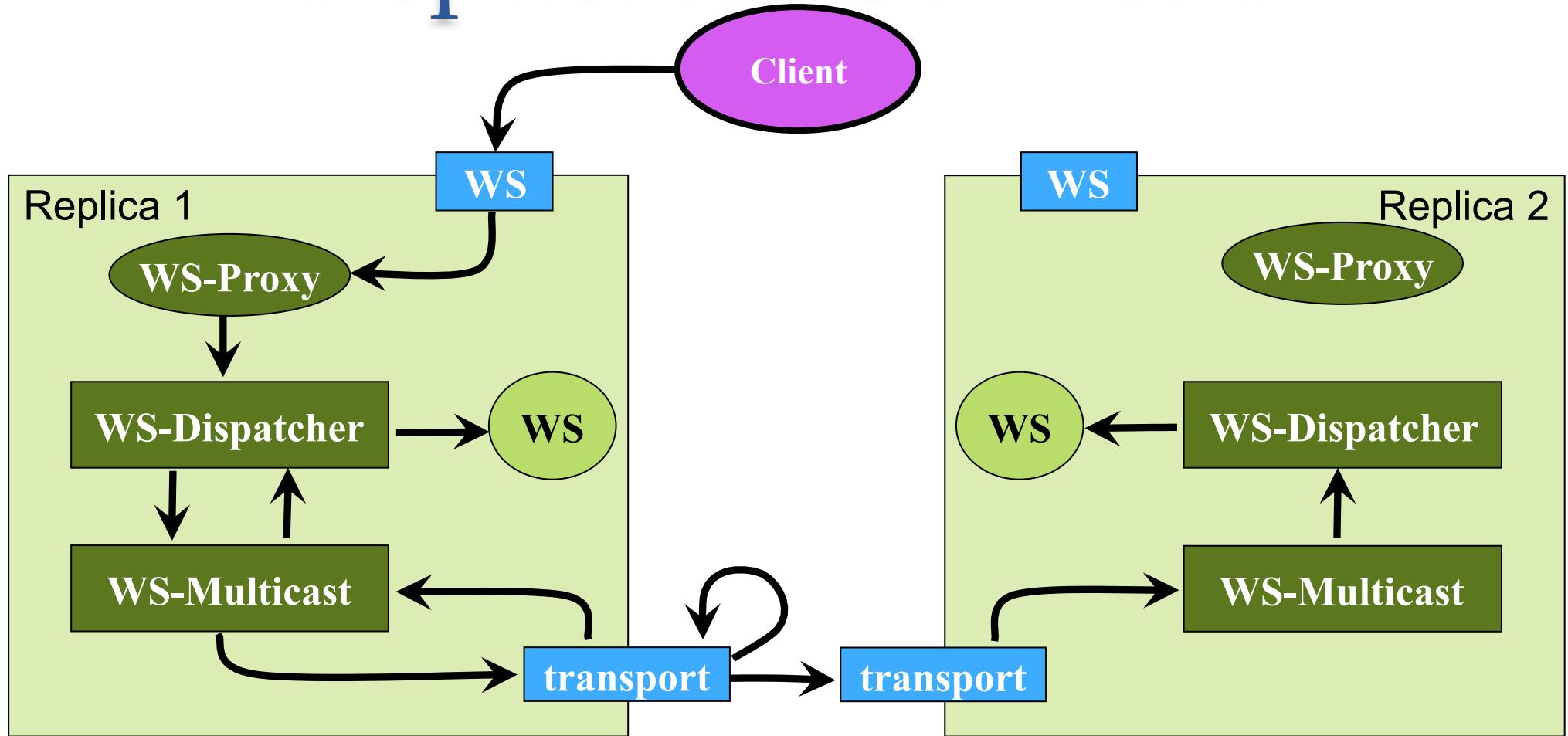
HA in SOA: The WS-Replication Framework

- **WS-Replication** is a framework that eases the replication of WSs
 - SOAP-based web services
- **Properties:**
 - Respects WS autonomy
 - Provides transparent fault-tolerance
- **Components:**
 - Deployer tool
 - WS-Multicast service
 - WS-Dispatcher

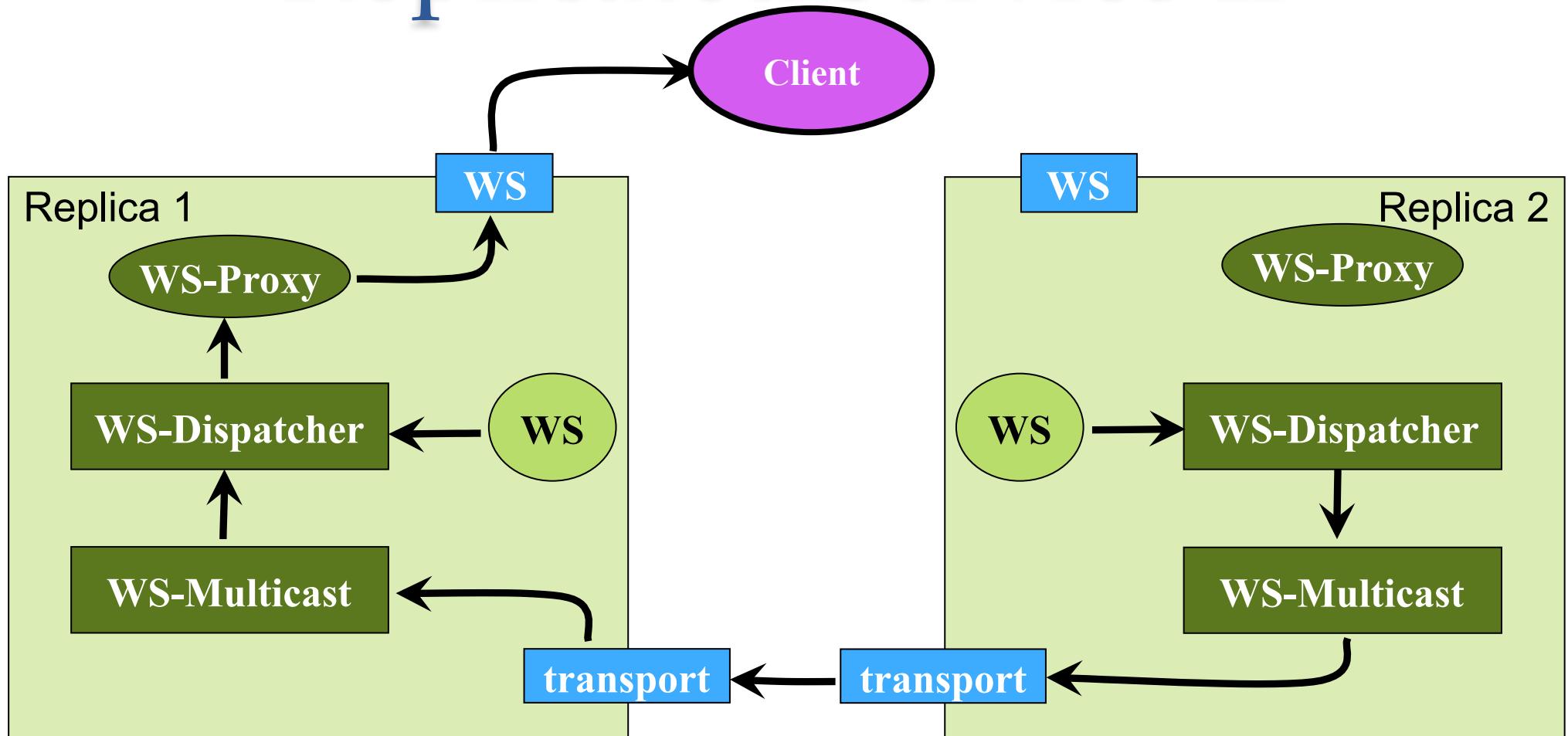
Background: Active Replication



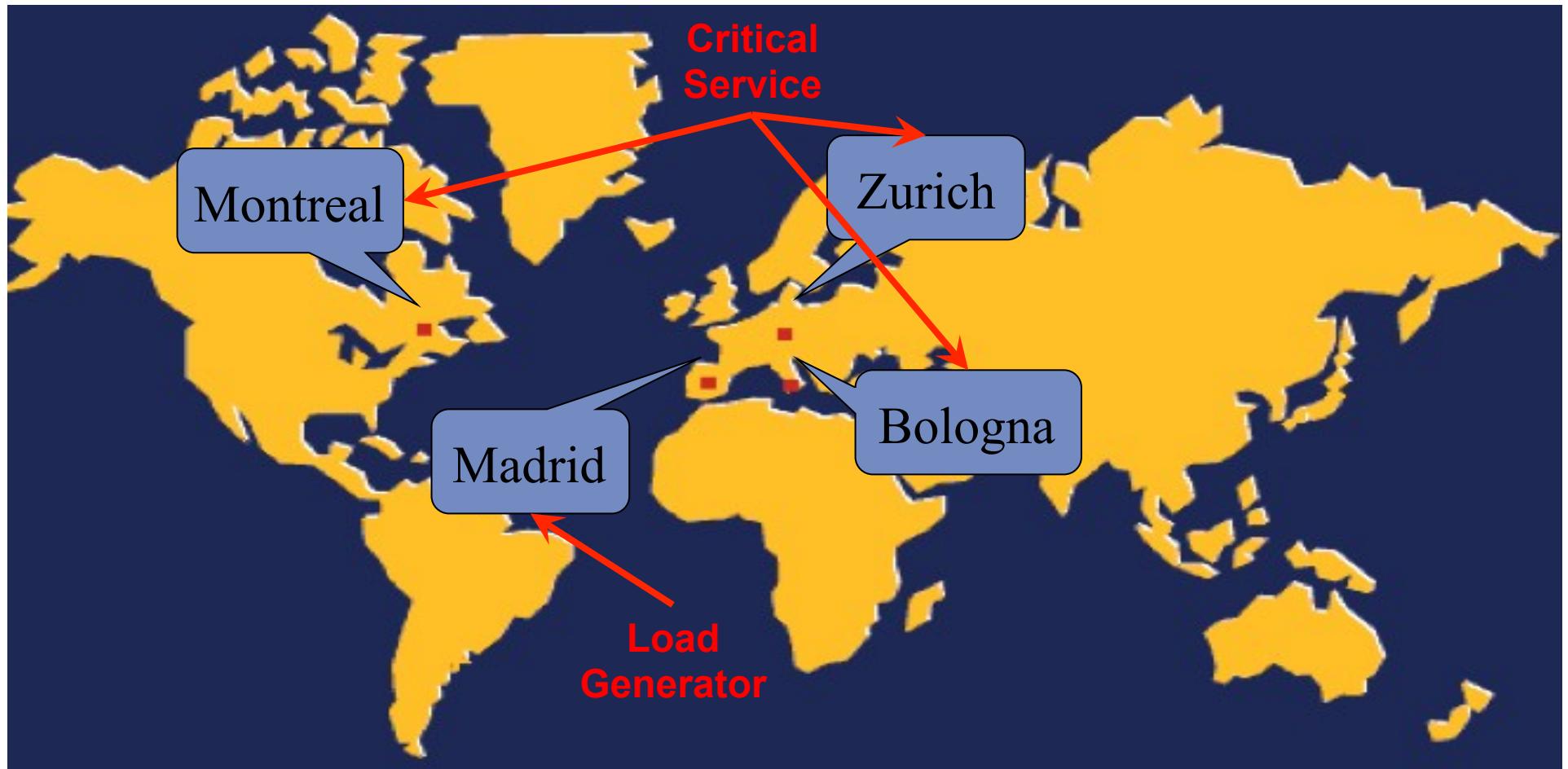
WS-Replication: Invoking a Replicated Service I



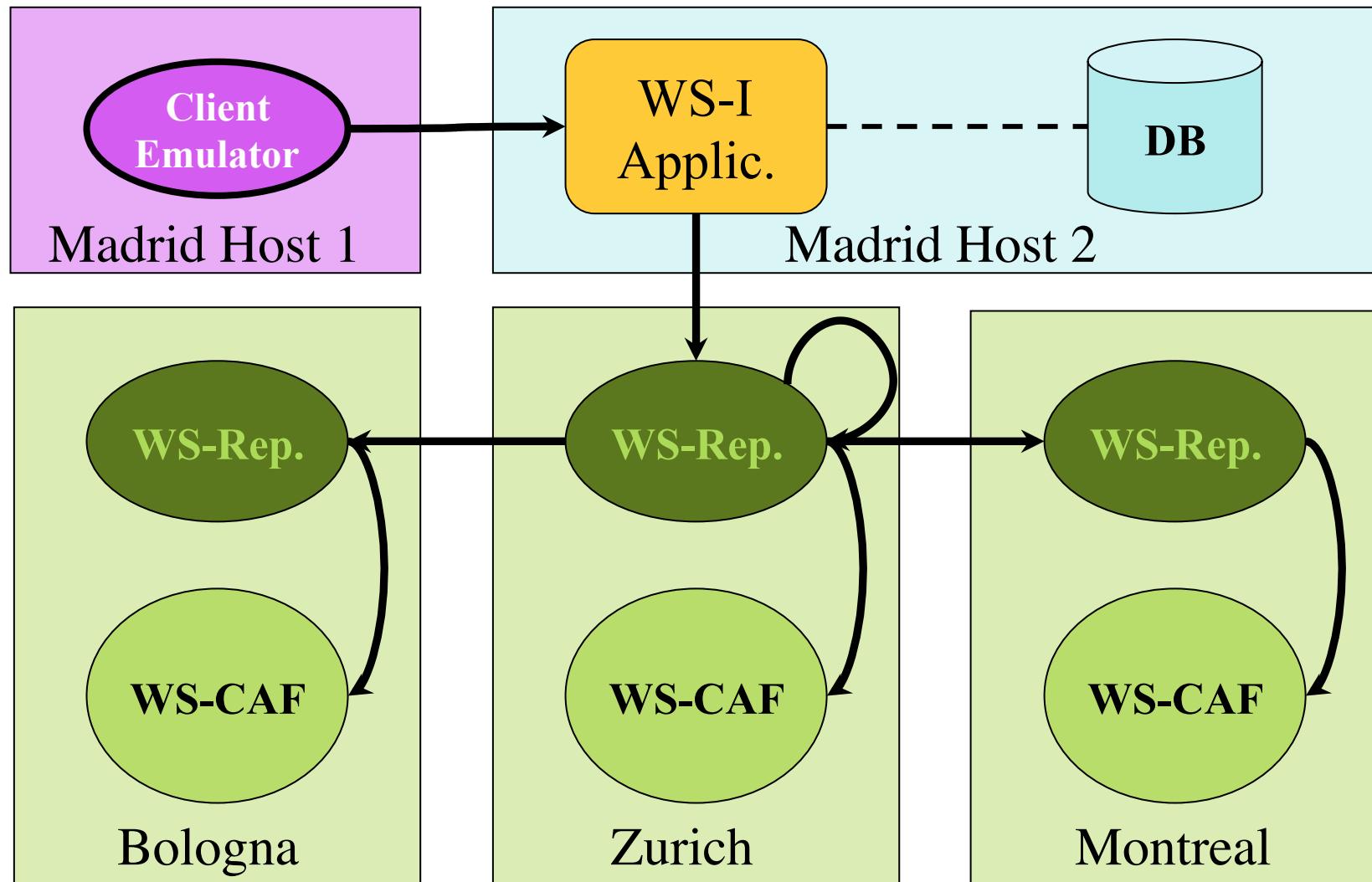
WS-Replication: Invoking a Replicated Service II



WS-Replication Evaluation: Setup

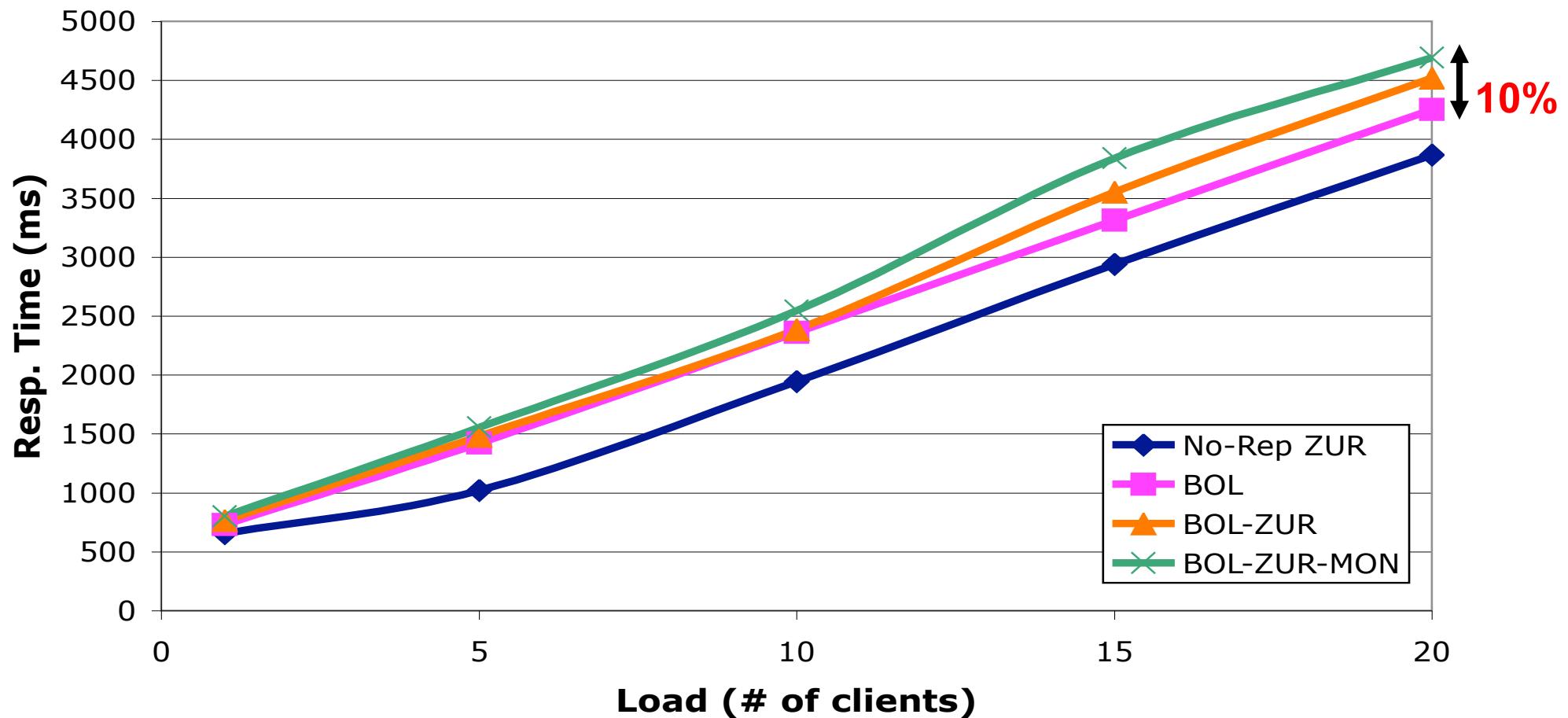


WS-I & WS-CAF Integration



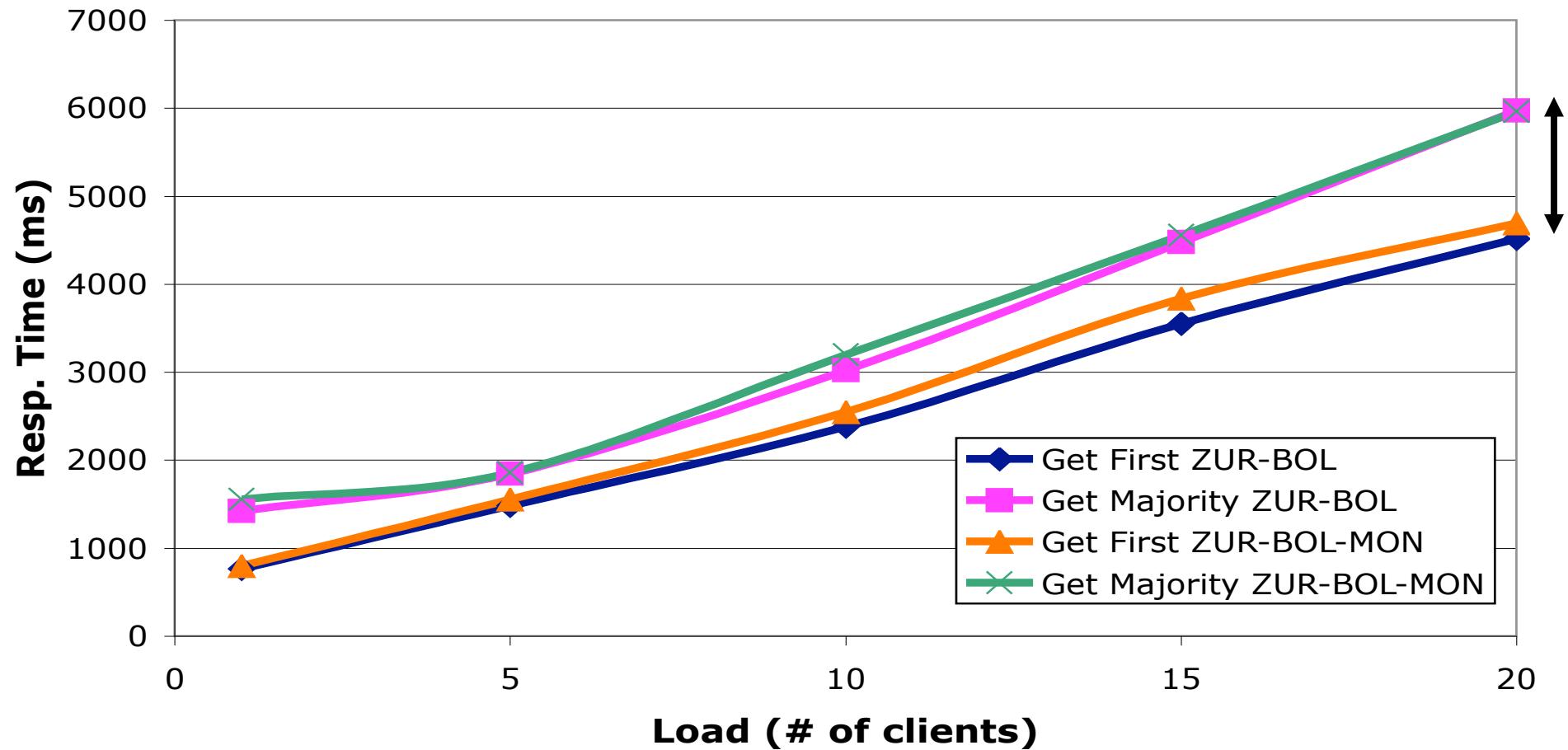
WS-CAF Replication

WS-CAF Resp. Time (GET FIRST)



WS-CAF Replication

WS-CAF Resp. Time (GET FIRST vs MAJORITY)



Outline

- High Availability (HA) and Scalability in Multi-Tier Architectures
 - Protocols for High Availability in MTAs
 - A Protocol for HA and Scalability in MTAs
- High Availability in Service-Oriented Architectures
 - WS-Replication Framework
- **Conclusions**
- Publications

Conclusions

- **We have developed** a set of replication and recovery protocols for providing consistent high availability and scalability to multi-tier applications
- **Main contributions:**
 - Transaction-aware replication
 - Exactly-once execution of client requests
 - Deal with several interaction patterns
 - Scalability through a replicated SI cache in the app. server
 - Online recovery (Not presented because the lack of time)
- **Results show** that the proposed protocols are affordable

Conclusions

- **We have also developed** a framework to provide high availability to SOAs
- **WS-Replication provides** seamless replication to critical WSs
- **Adequate engineering** proved to provide affordable performance
- **Evaluation** of a realistic application in WANs **has shown** a quite reasonable overhead

Outline

- High Availability (HA) and Scalability in Multi-Tier Architectures
 - Protocols for High Availability in MTAs
 - A Protocol for HA and Scalability in MTAs
- High Availability in Service-Oriented Architectures
 - WS-Replication Framework
- Conclusions
- **Publications**

Publications

- Jorge Salas, *Francisco Pérez-Sorrosal*, Marta Patiño-Martínez and Ricardo Jiménez-Peris. **WS-Replication: a Framework for Highly Available Web Services**. WWW, 2006.
 - Acceptance rate: **11 %**
 - Percentile top **0 %** in Microsoft's Libra (WWW category)
- *Francisco Pérez-Sorrosal*, Marta Patiño-Martínez, Ricardo Jiménez-Peris and Bettina Kemme. **Consistent and Scalable Cache Replication for Multi-tier J2EE Applications**. Middleware, 2007.
 - Acceptance rate: **20 %**
 - Percentile top **12 %** in Microsoft's Libra (Dist. And Parall. Computing category)
- *Francisco Pérez-Sorrosal*, Marta Patiño-Martínez, Ricardo Jiménez-Peris and Jaksa Vuckovic. **Highly Available Long Running Transactions and Activities for J2EE Applications**. ICDCS, 2006.
 - Acceptance rate: **13 %**
 - Percentile top **3 %** in Microsoft's Libra (Dist. And Parall. Computing category)

Thank You!

QUESTIONS?