WS-Replication: A Framework for Highly Available Web Services

J. Salas, F. Pérez-Sorrosal, M. Patiño-Martínez, R. Jiménez-Peris

Lsd Distributed Systems Laboratory

Universidad Politécnica de Madrid (UPM)

Index

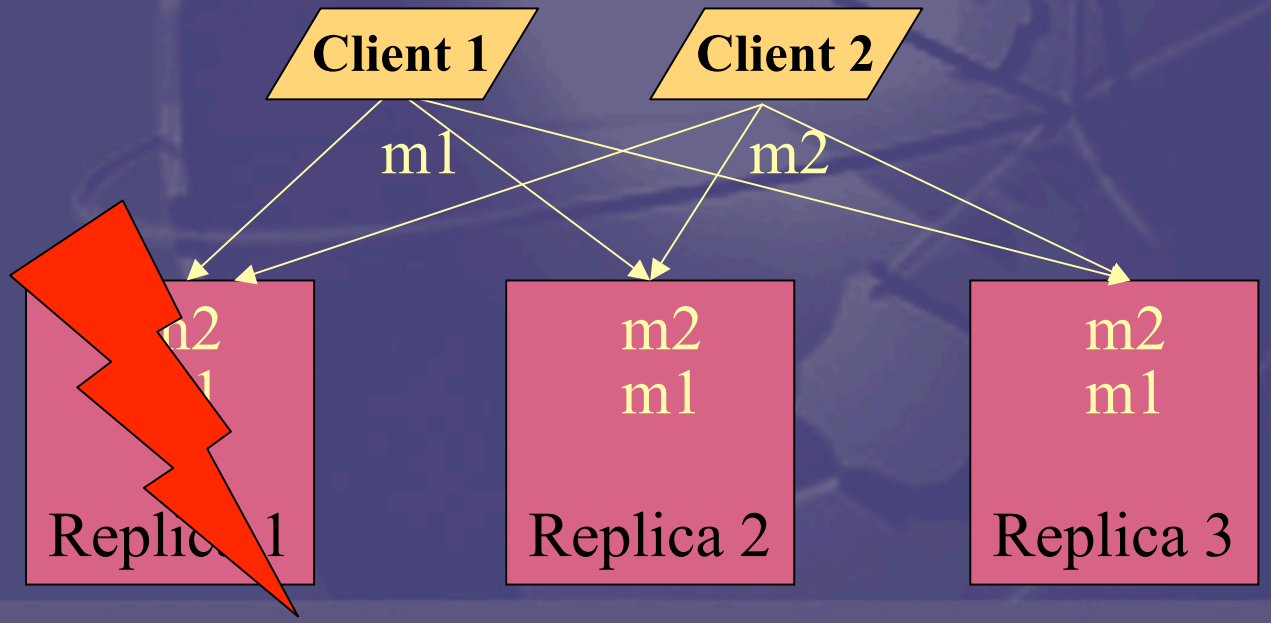
1. Motivation
2. Introduction
   - Replication
   - Group Communication
3. WS-Replication Framework
4. WS-Replication Service
5. Engineering WS-Replication service
6. Evaluation
   - Setup
   - WS-Replication Micro-benchmark
   - Evaluation of a critical service: WS-CAF replication
7. Conclusions

Motivation

- Web service technology is maturing very fast.
- During the next few years an increasing number of mission critical web services will start to be deployed.
- Many of them will require strong availability.
- Current clustered solutions will be insufficient since they are not resilient to network outages.
- We propose an infrastructure to provide high availability of web services relying exclusively on web service technology.

Introduction: Replication

- Replication is the technique used to provide high availability.
- We have opted for **active replication** due to it provides fast failover and strong consistency.



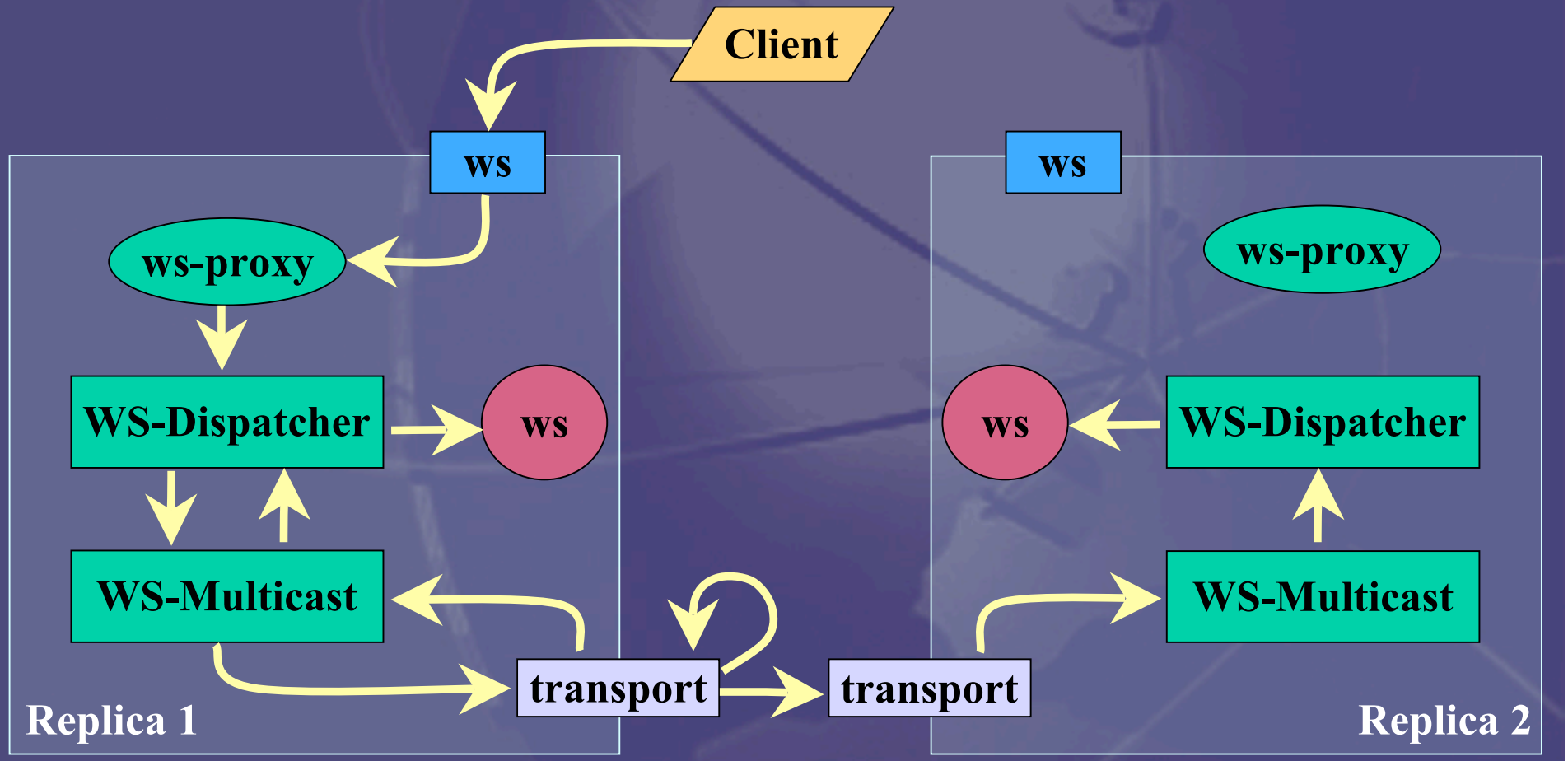
Introduction: Group Communication

- Group communication provides a membership service.
- Group communication also provides reliable multicast to groups of replicas with different ordering and reliability guarantees.
- In here, we are interested in total ordering and reliability.

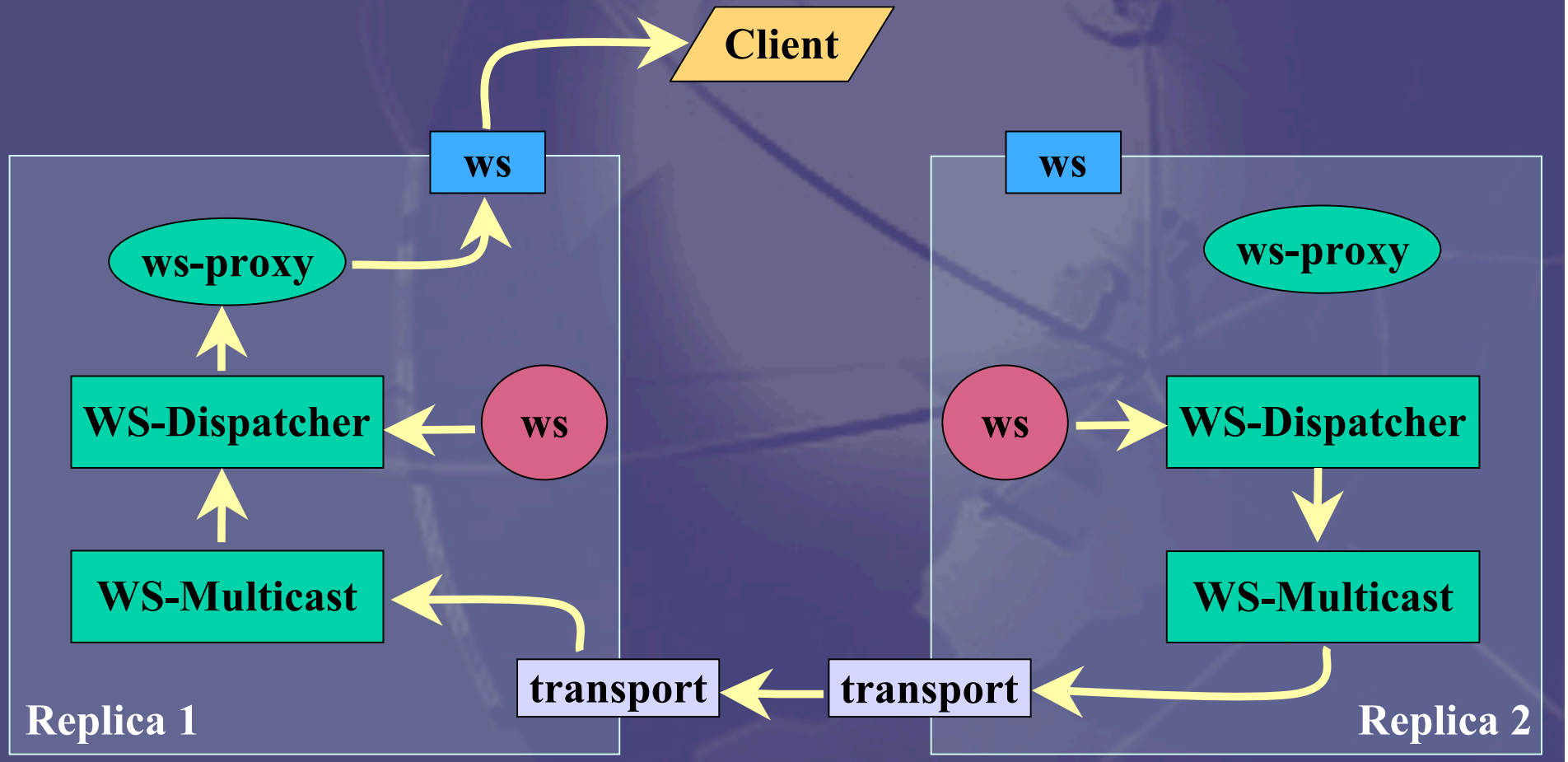
WS-Replication Framework

- WS-Replication is a framework that provides active replication of WSs.
- Components:
  - Deployer tool.
  - WS-Multicast service.
  - WS-Dispatcher.
- Properties:
  - Respects WS autonomy.
  - Provides transparent fault-tolerance.

WS-Replication: Invoking a Replicated Service I

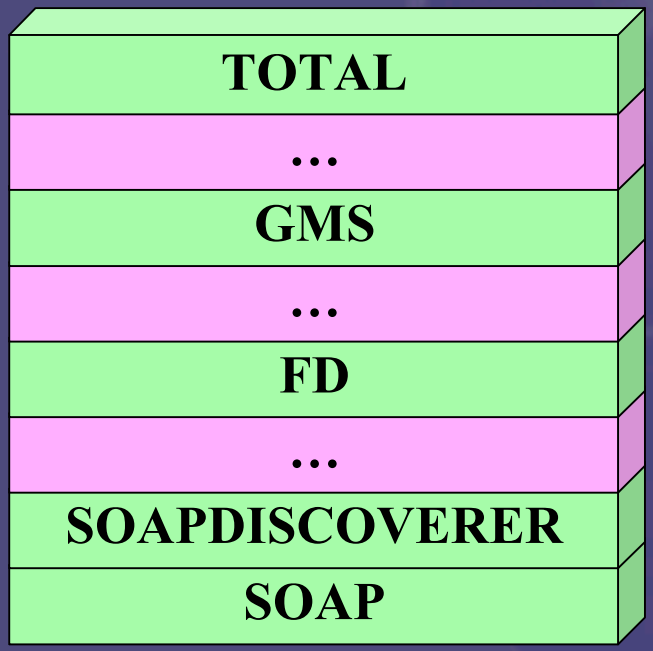


WS-Replication: Invoking a Replicated Service II



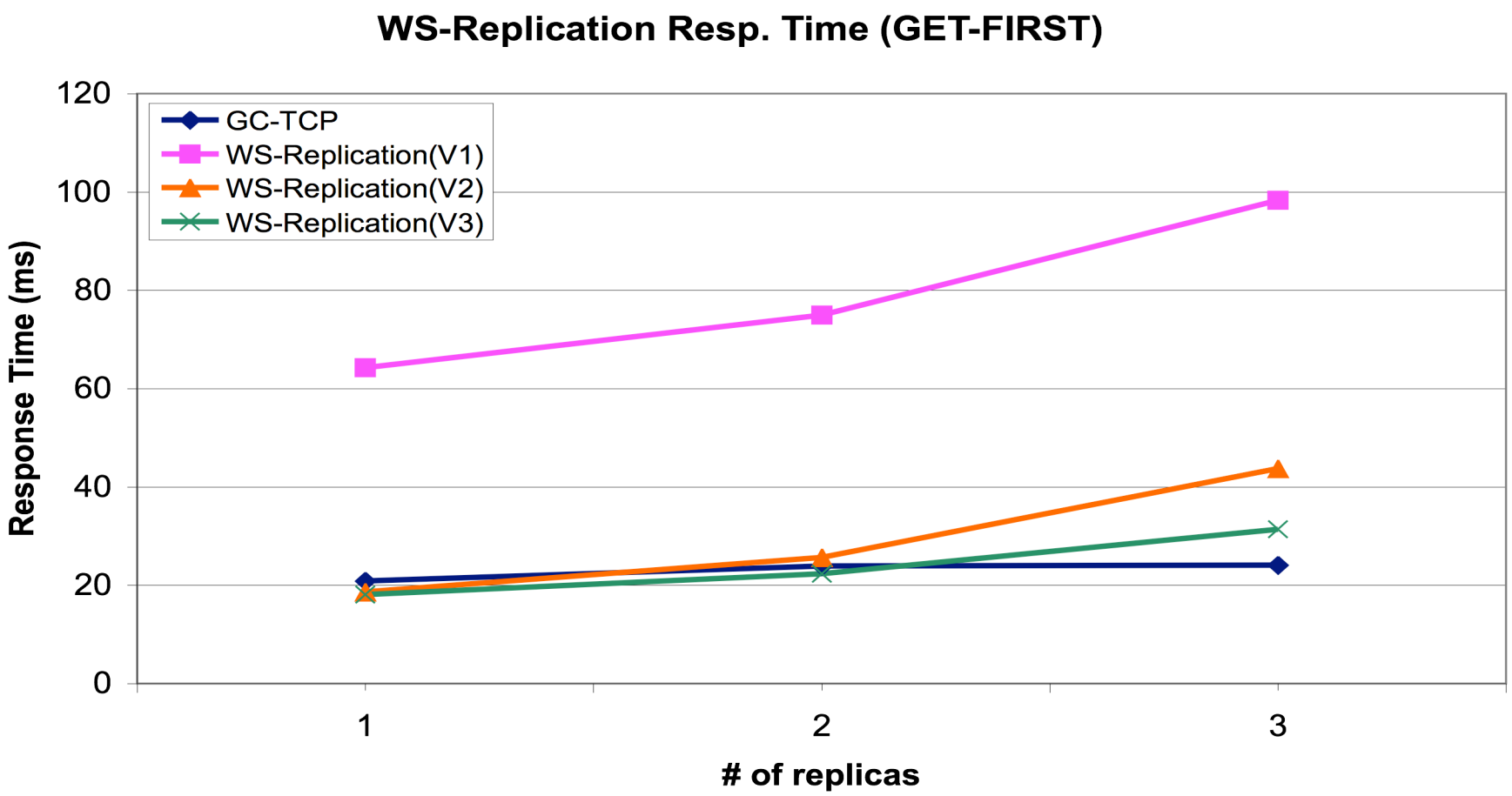
WS-Multicast

- WS-Multicast extends a group communication stack: JGroups.
- The main challenge is to attain a performant SOAP-based multicast layer.



- **TOTAL: Total ordering.**
- **GMS: Group membership.**
- **FD: Failure detection.**
- **SOAPDISCOVERER: Finds new group members.**
- **SOAP: SOAP transport.**

Engineering WS-Replication I



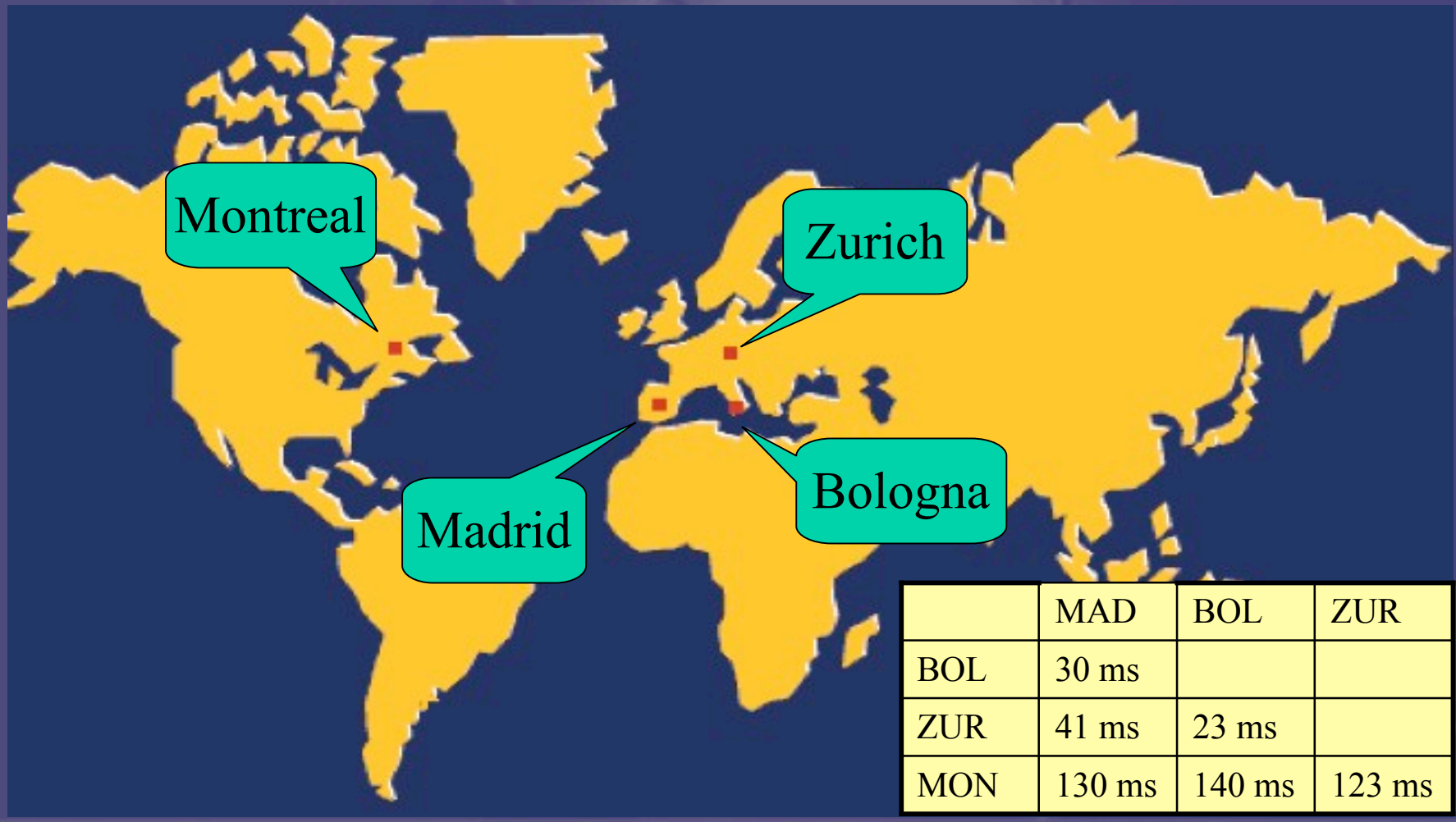
Engineering WS-Replication II

- Improvements on standard Java object serialization mechanism.
- Simplified XML tags and parameters to reduce the amount of information exchanged between replicas.
- Local invocations within components of the framework were transformed to Java invocations.

Evaluation

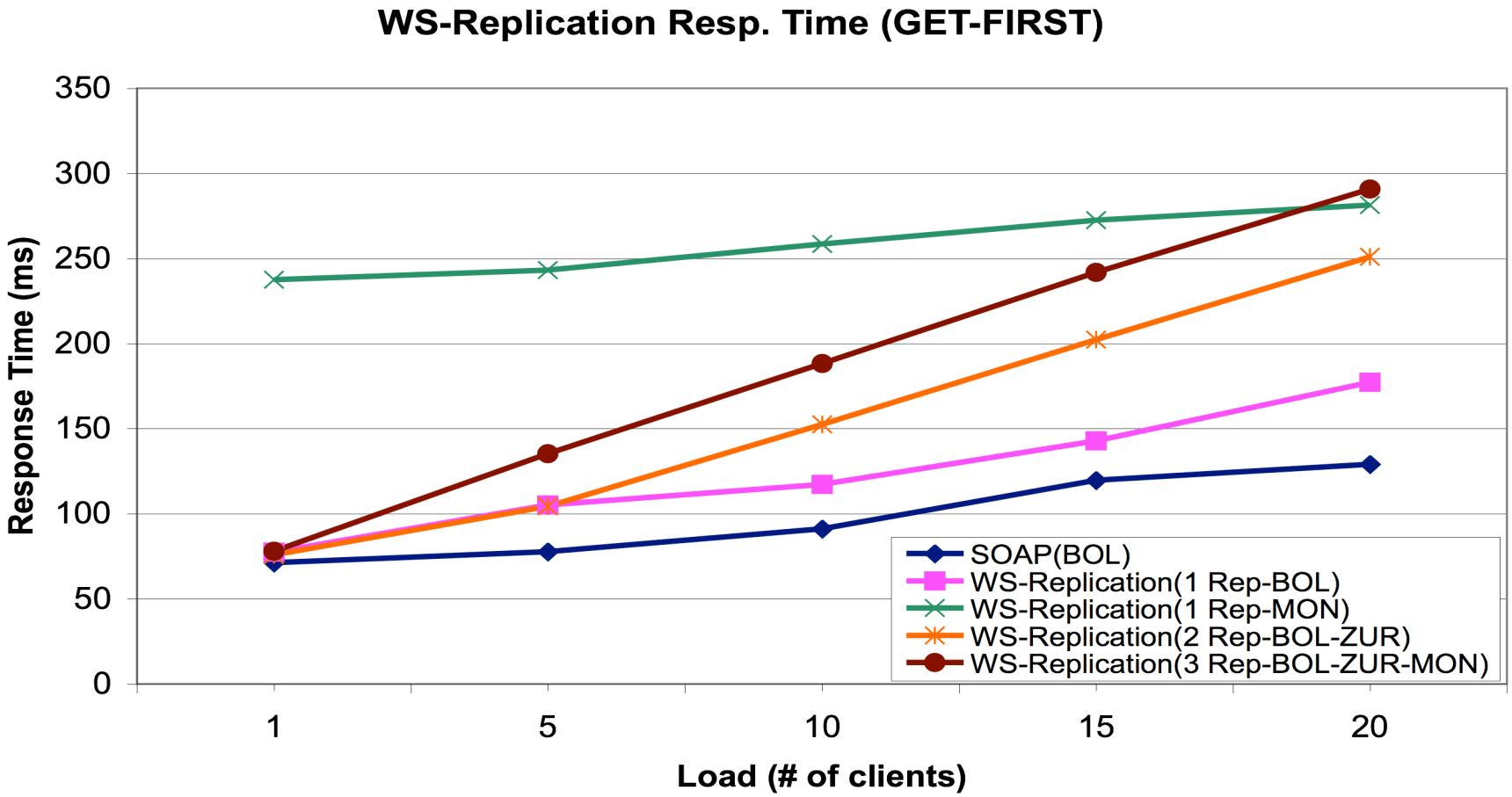
- The goal is to measure the overhead of the replication framework and its performance in a WAN environment.
- Two evaluations:
  - Micro-benchmark for measuring the overhead of the implementation.
  - A benchmark for a realistic use of the framework based on a WS-I scenario.

Evaluation: Setup



	MAD	BOL	ZUR
BOL	30 ms		
ZUR	41 ms	23 ms	
MON	130 ms	140 ms	123 ms

Evaluation: Micro-benchmark

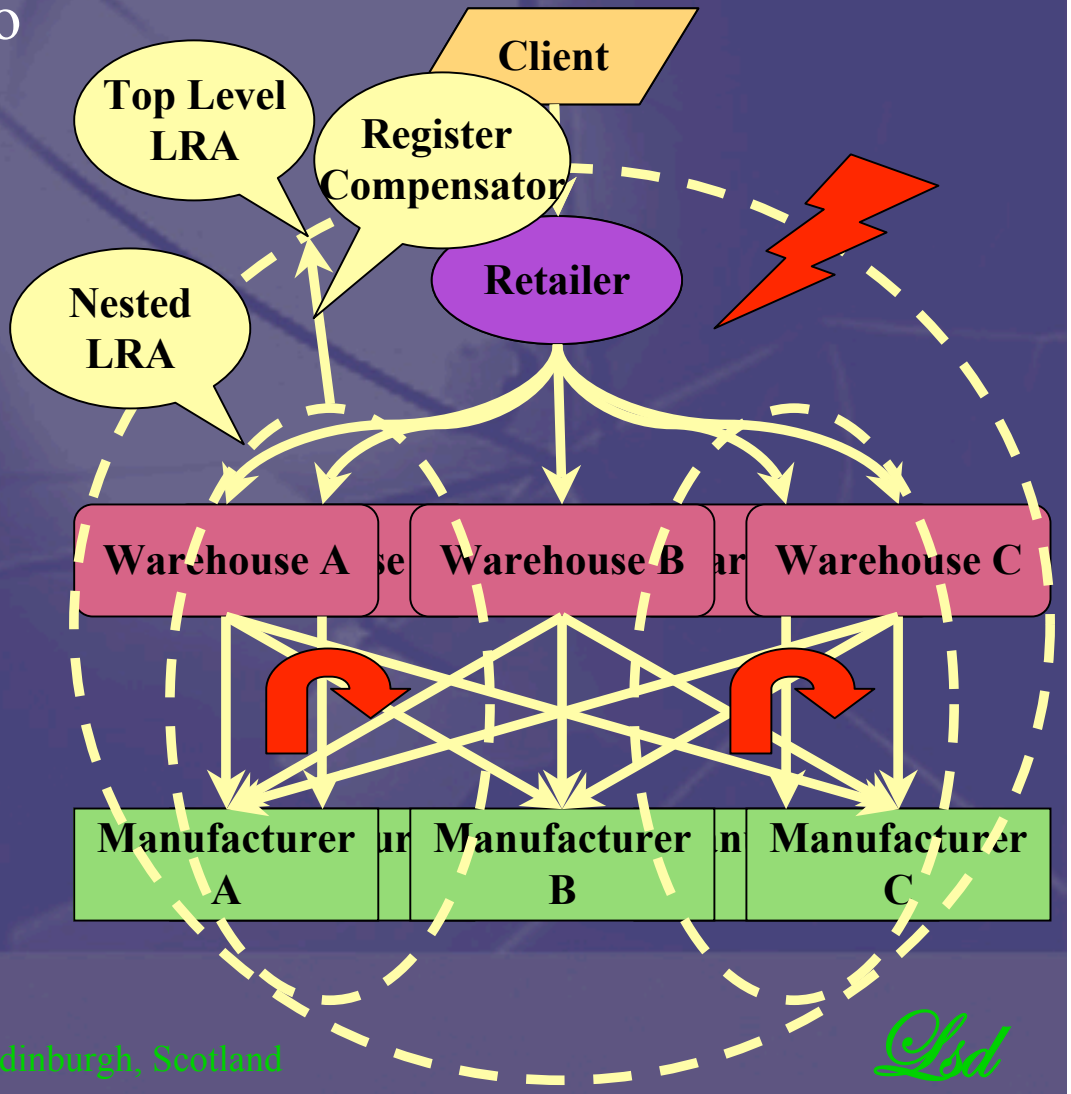


Evaluation: WS-Composite Application Framework (WS-CAF)

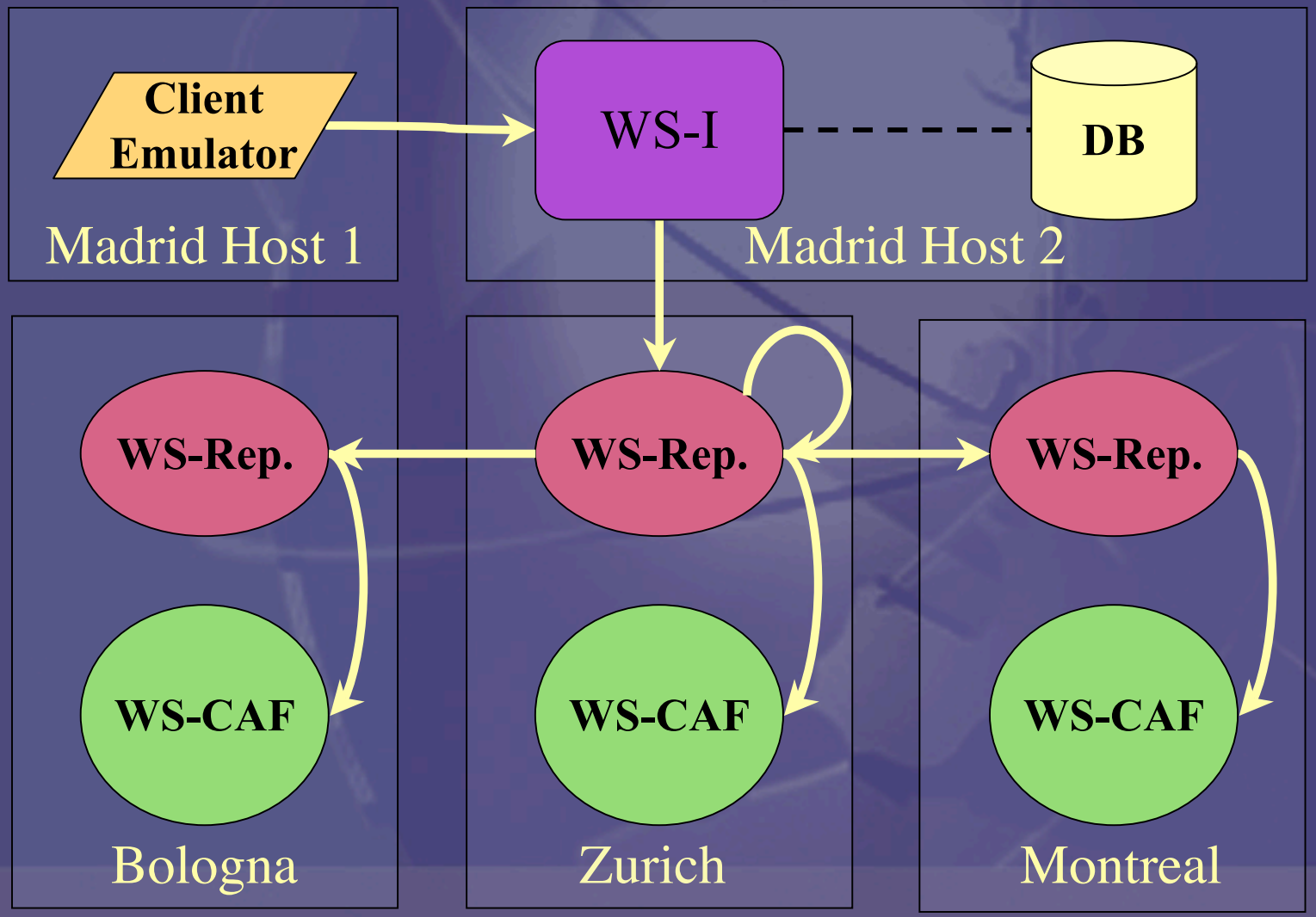
- WS-CAF is a protocol stack that provides transactional facilities to WSs.
- WS-CTX allows to manage a common context structure.
- WS-CF defines a coordinator to guarantee the notification of messages to the WSs participating in a particular context.
- WS-TXM provides specific coordination:
  - ACID transactions
  - Long running actions
  - Business process transactions

Evaluation: WS-I & WS-CAF Integration (I)

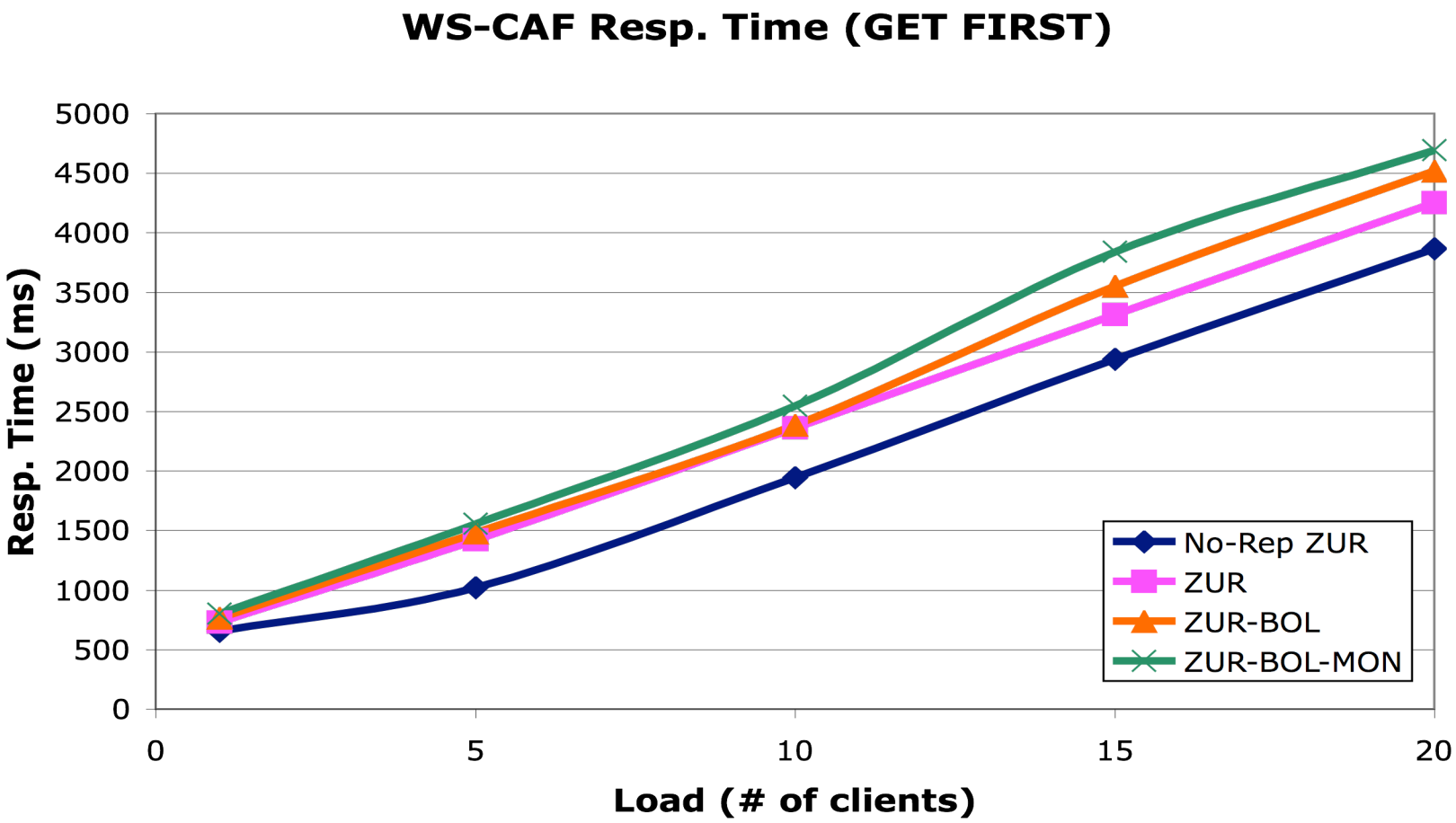
- We used a WS-I scenario to evaluate WS-Replication.
  - Supply-chain management scenario
- WS-I was enhanced with long running activities provided by WS-CAF.
- WS-Replication was used to provide high availability for WS-CAF services.



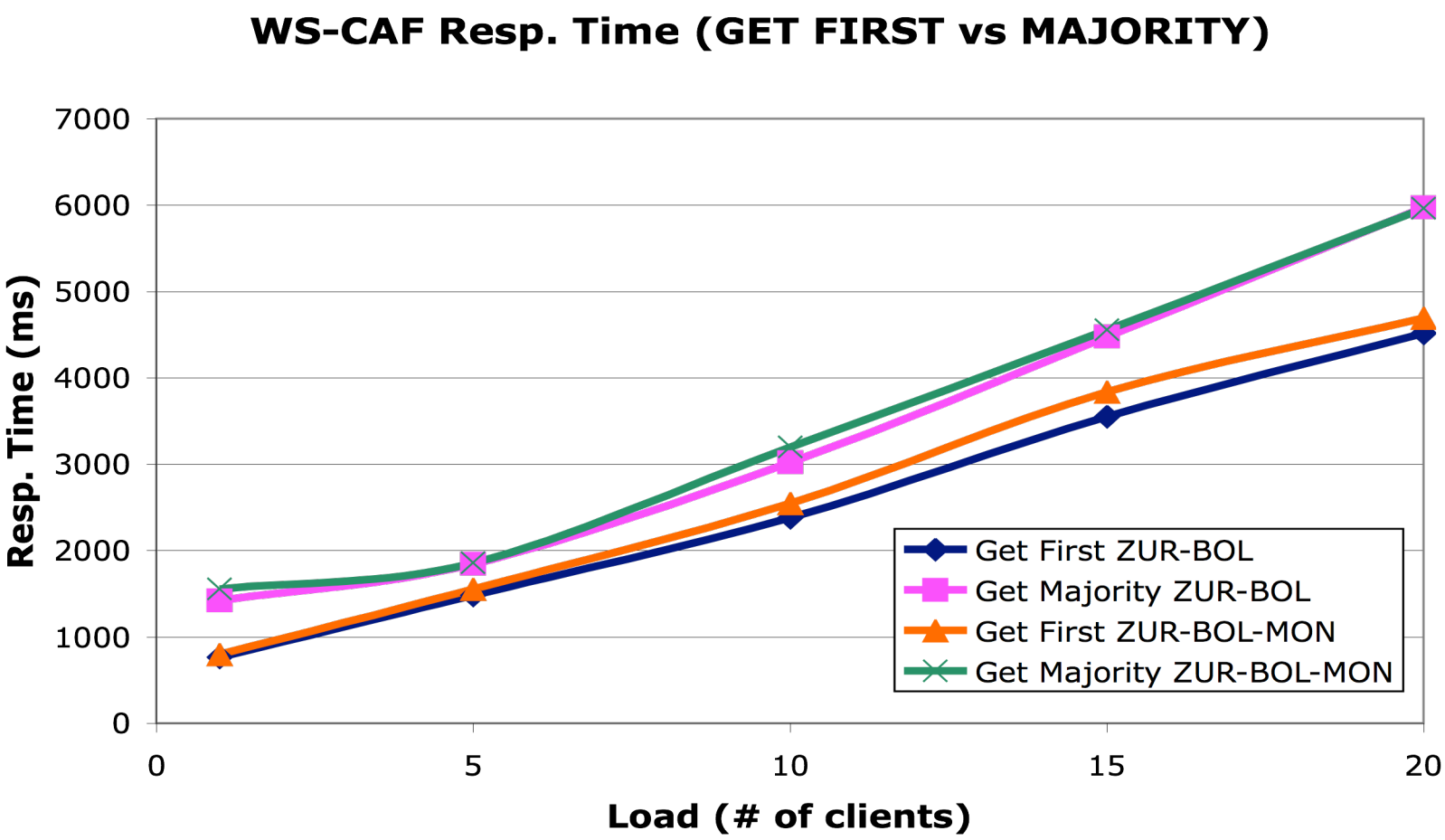
Evaluation: WS-I & WS-CAF Integration (II)



Evaluation: WS-CAF Replication



Evaluation: WS-CAF Replication



Conclusions

- The spreading of WS technology will require support for high availability solutions.
- WS-Replication is a replication framework that aims to provide seamless replication of WSs.
- Adequate engineering proved to provide affordable performance.
- Evaluation of a realistic application has shown that the overhead is quite reasonable.

Questions?