

Resilient Cloud-based Replication with Low Latency

December 9, 2020

Michael Eischer and Tobias Distler

Friedrich-Alexander University Erlangen-Nürnberg (FAU)



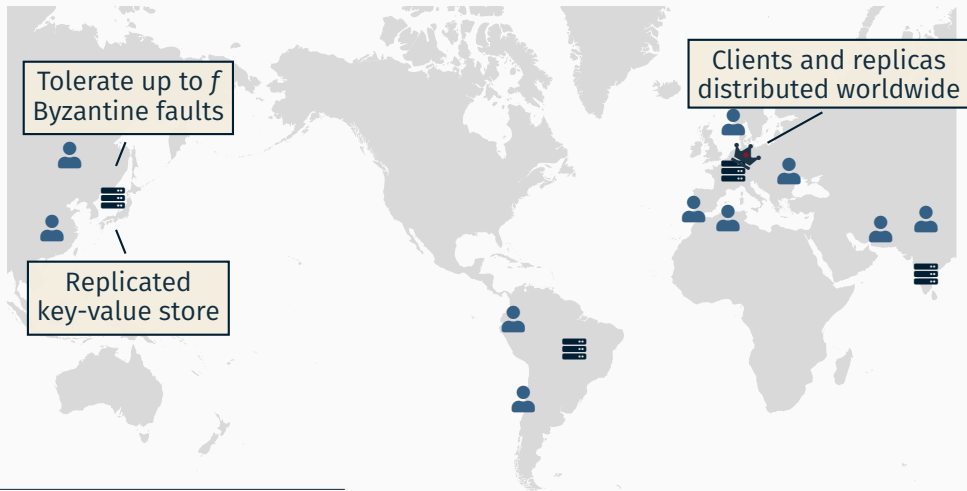
FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG

FACULTY OF ENGINEERING

Supported by



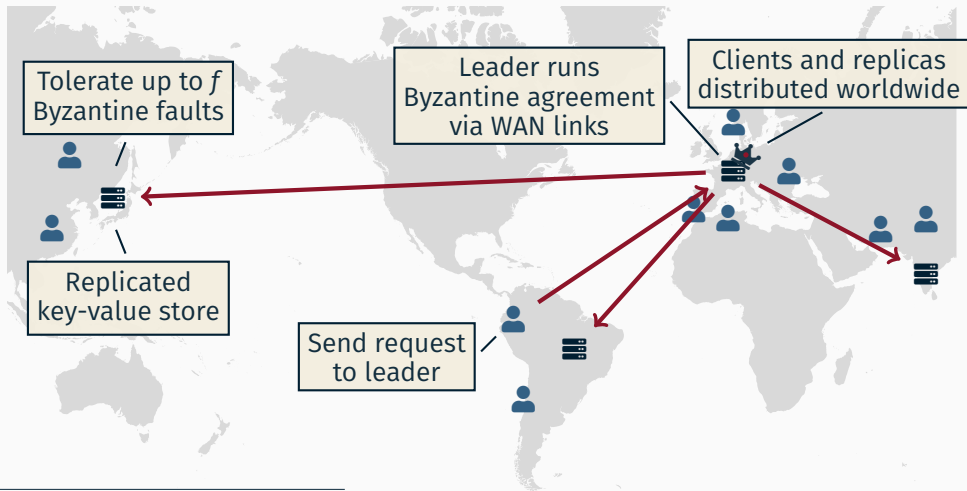
grant no. DI 2097/1-2 ("REFIT")



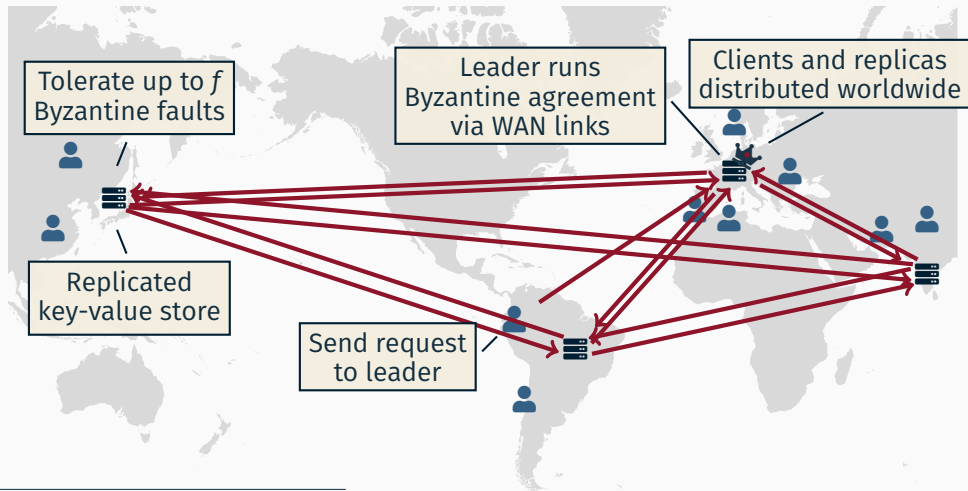
¹Miguel Castro and Barbara Liskov. "Practical Byzantine Fault Tolerance." In: *Proceedings of the 3rd Symposium on Operating Systems Design and Implementation (OSDI '99)*. 1999, pp. 173–186.



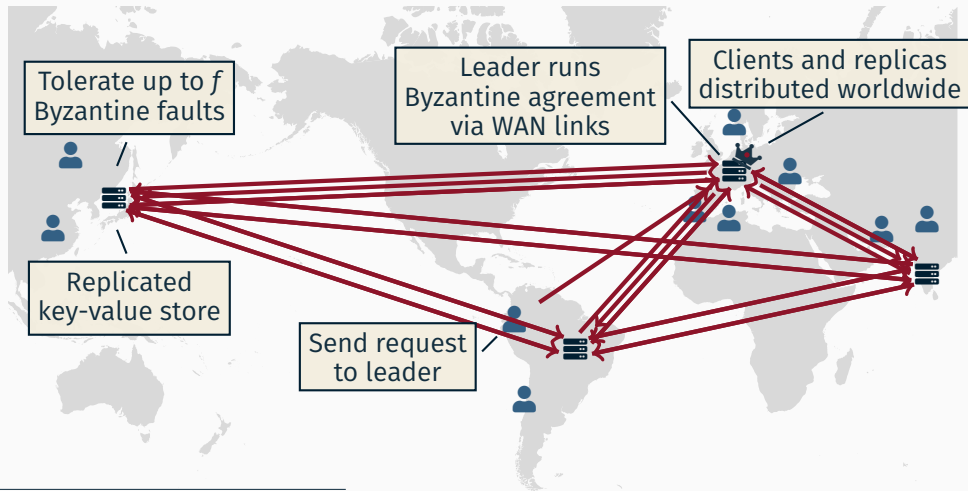
¹Miguel Castro and Barbara Liskov. "Practical Byzantine Fault Tolerance." In: *Proceedings of the 3rd Symposium on Operating Systems Design and Implementation (OSDI '99)*. 1999, pp. 173–186.



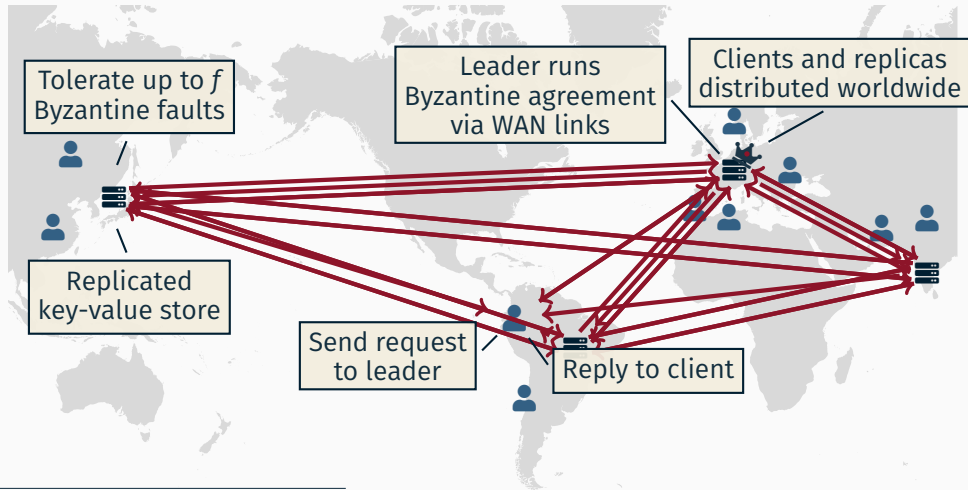
¹Miguel Castro and Barbara Liskov. "Practical Byzantine Fault Tolerance." In: *Proceedings of the 3rd Symposium on Operating Systems Design and Implementation (OSDI '99)*. 1999, pp. 173–186.



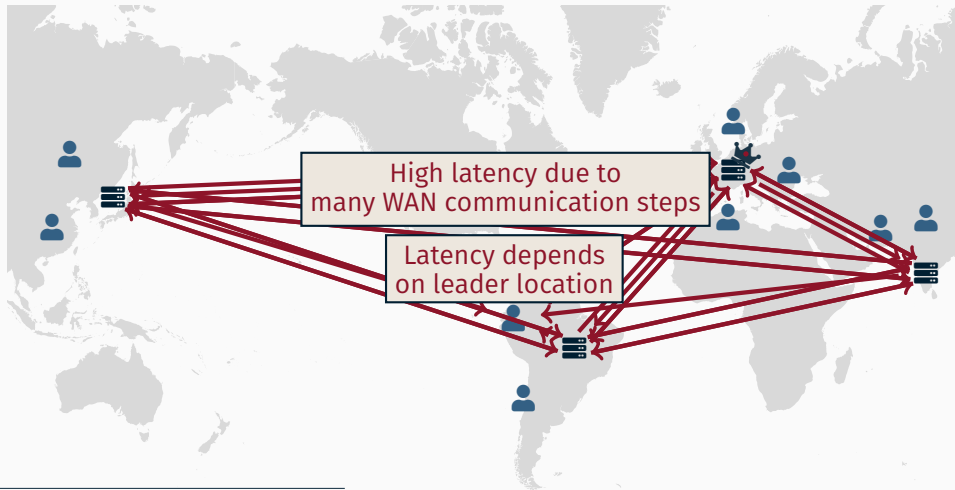
¹Miguel Castro and Barbara Liskov. "Practical Byzantine Fault Tolerance." In: *Proceedings of the 3rd Symposium on Operating Systems Design and Implementation (OSDI '99)*. 1999, pp. 173–186.



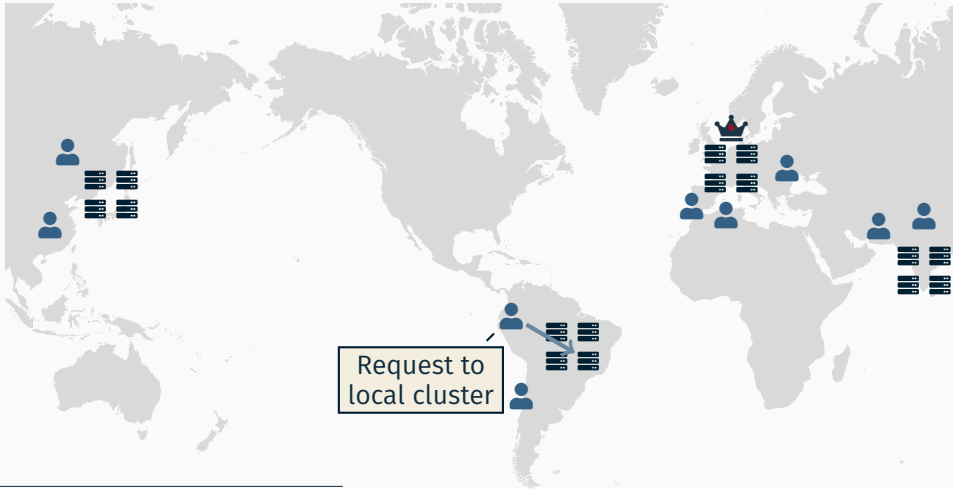
¹Miguel Castro and Barbara Liskov. "Practical Byzantine Fault Tolerance." In: *Proceedings of the 3rd Symposium on Operating Systems Design and Implementation (OSDI '99)*. 1999, pp. 173–186.



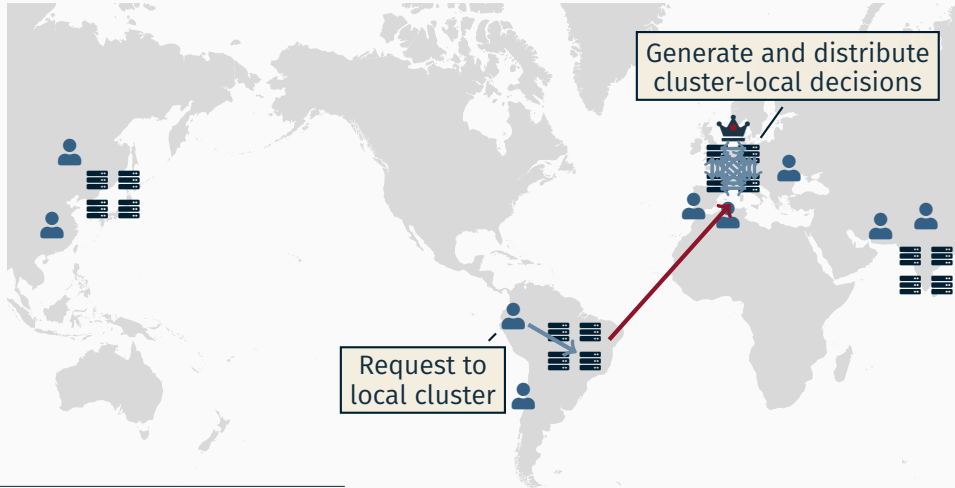
¹Miguel Castro and Barbara Liskov. "Practical Byzantine Fault Tolerance." In: *Proceedings of the 3rd Symposium on Operating Systems Design and Implementation (OSDI '99)*. 1999, pp. 173–186.



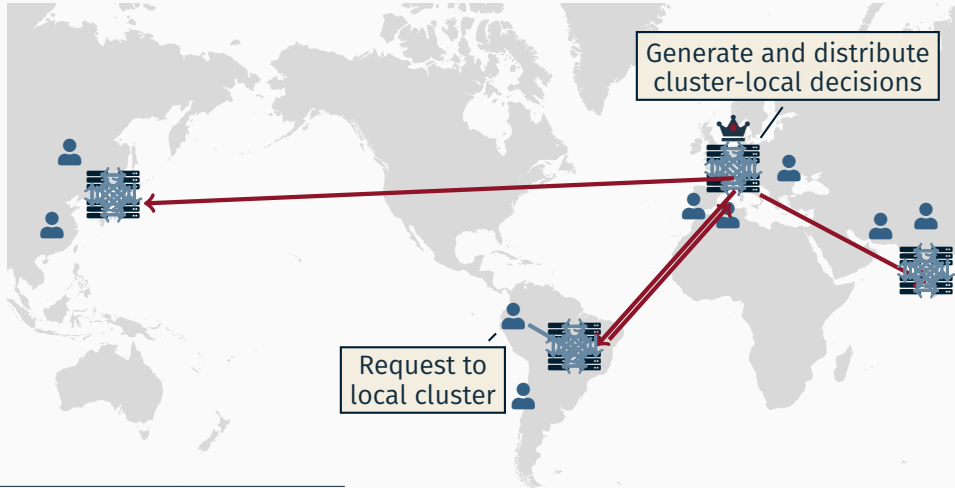
¹Miguel Castro and Barbara Liskov. "Practical Byzantine Fault Tolerance." In: *Proceedings of the 3rd Symposium on Operating Systems Design and Implementation (OSDI '99)*. 1999, pp. 173–186.



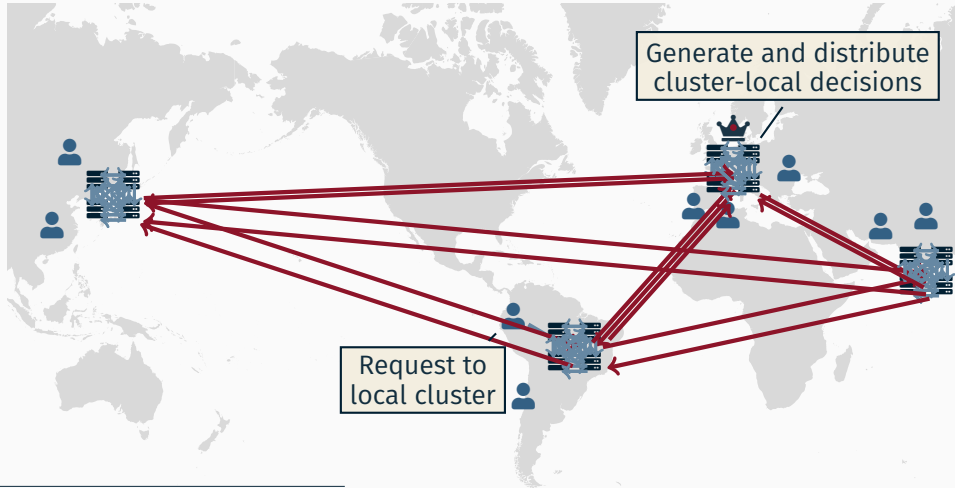
²Yair Amir et al. "Steward: Scaling Byzantine Fault-Tolerant Replication to Wide Area Networks." In: *IEEE Transactions on Dependable and Secure Computing* 7.1 (2010), pp. 80–93.



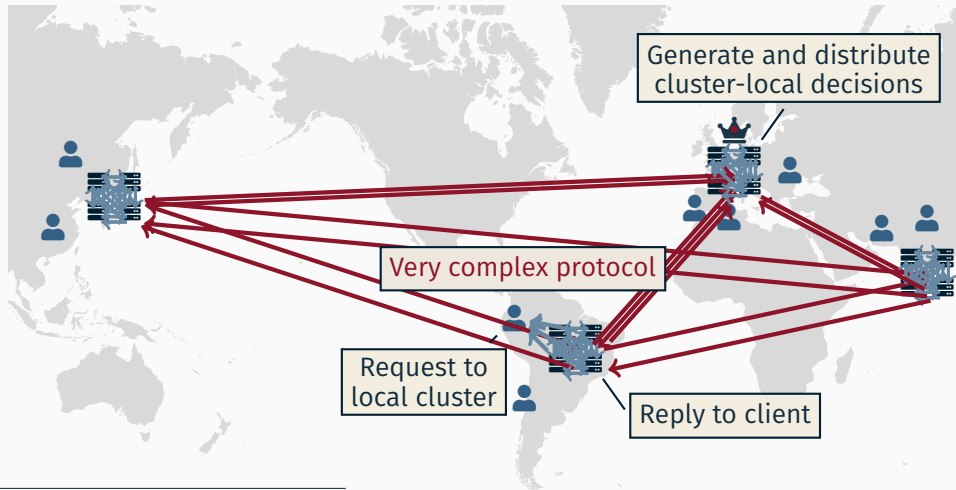
²Yair Amir et al. "Steward: Scaling Byzantine Fault-Tolerant Replication to Wide Area Networks." In: *IEEE Transactions on Dependable and Secure Computing* 7.1 (2010), pp. 80–93.



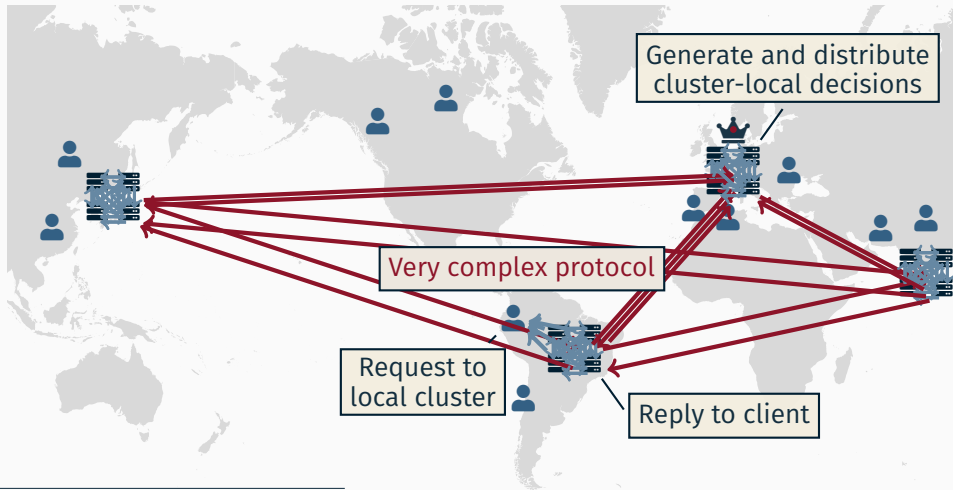
²Yair Amir et al. "Steward: Scaling Byzantine Fault-Tolerant Replication to Wide Area Networks." In: *IEEE Transactions on Dependable and Secure Computing* 7.1 (2010), pp. 80–93.



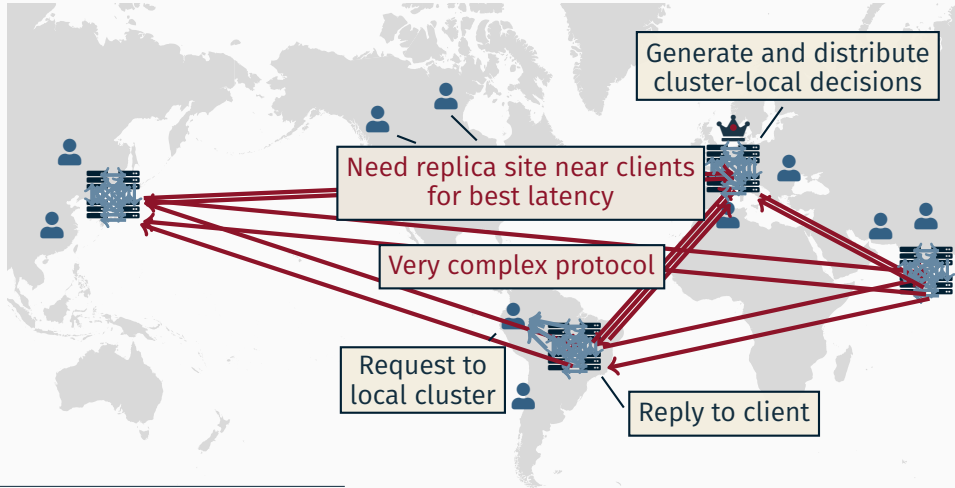
²Yair Amir et al. "Steward: Scaling Byzantine Fault-Tolerant Replication to Wide Area Networks." In: *IEEE Transactions on Dependable and Secure Computing* 7.1 (2010), pp. 80–93.



²Yair Amir et al. "Steward: Scaling Byzantine Fault-Tolerant Replication to Wide Area Networks." In: *IEEE Transactions on Dependable and Secure Computing* 7.1 (2010), pp. 80–93.



²Yair Amir et al. "Steward: Scaling Byzantine Fault-Tolerant Replication to Wide Area Networks." In: *IEEE Transactions on Dependable and Secure Computing* 7.1 (2010), pp. 80–93.



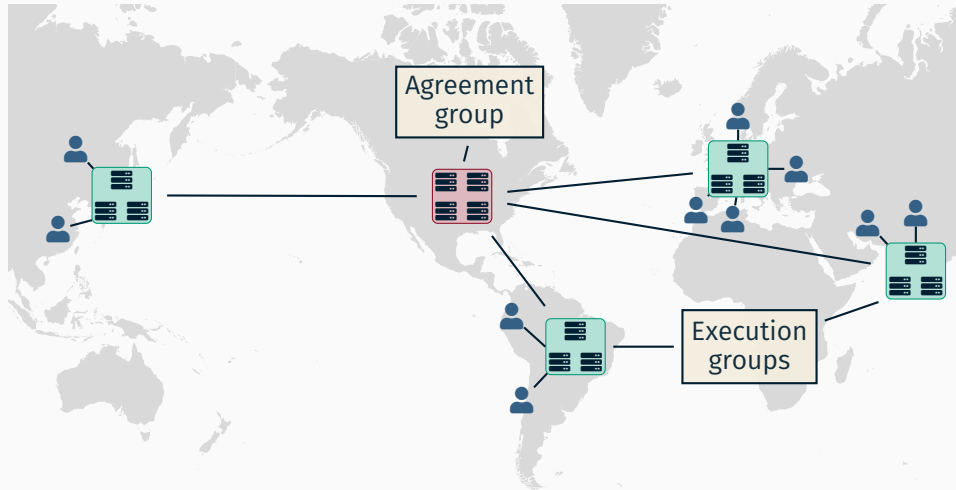
²Yair Amir et al. "Steward: Scaling Byzantine Fault-Tolerant Replication to Wide Area Networks." In: *IEEE Transactions on Dependable and Secure Computing* 7.1 (2010), pp. 80–93.

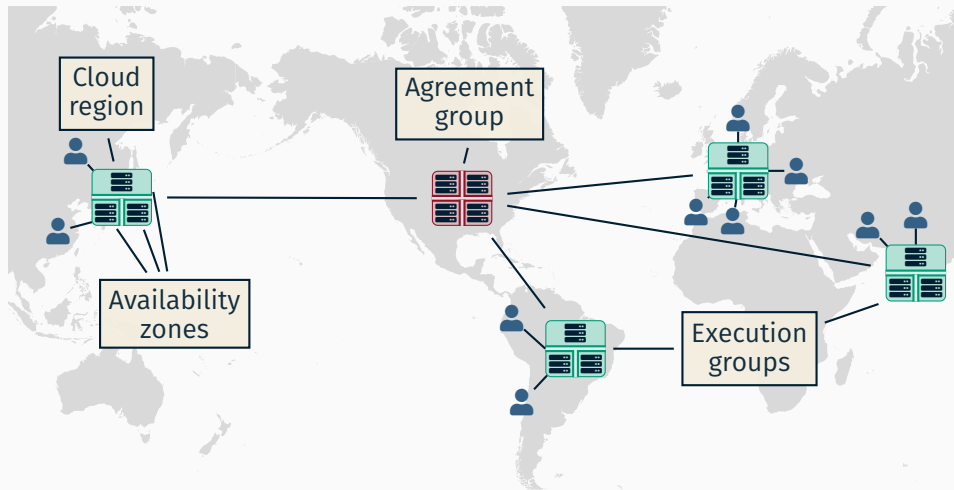
Challenges

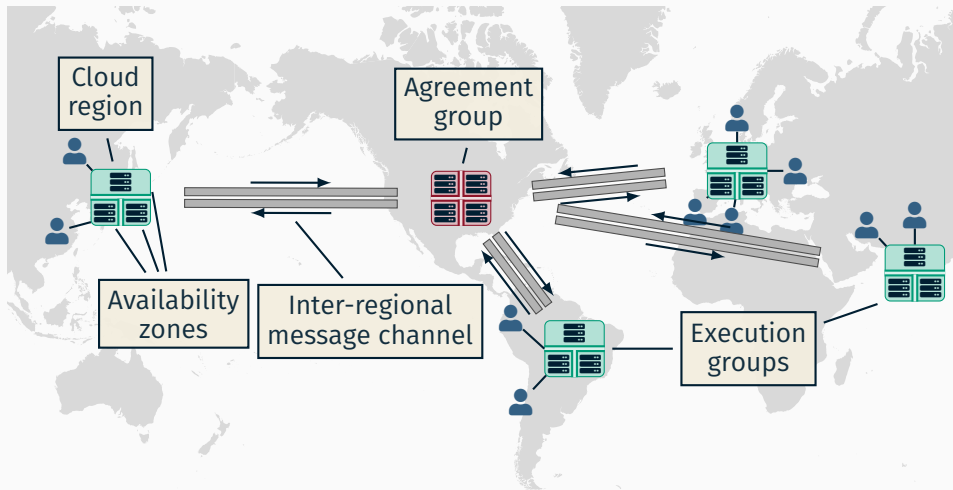
Need for a replication protocol that provides

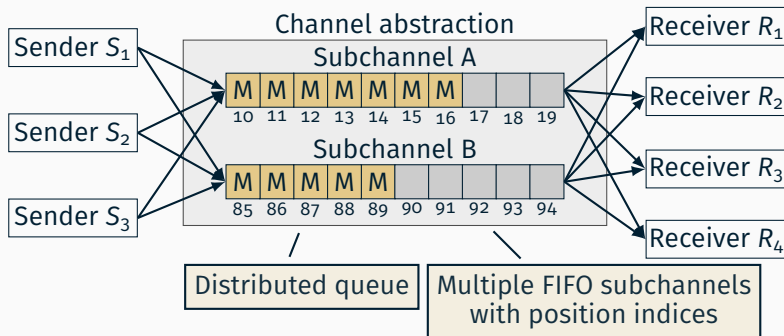
- **Efficiency:** No complex protocols over wide-area links
- **Modularity:** Allow integrating with different consensus protocols
- **Adaptability:** Add and remove new locations

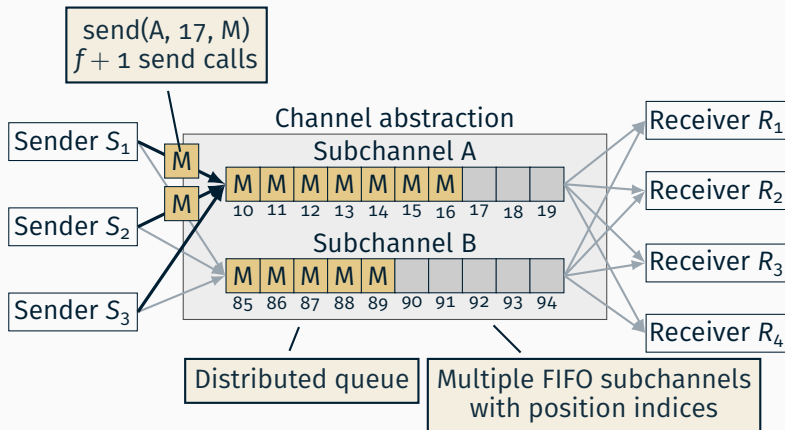
Our Approach: SPIDER

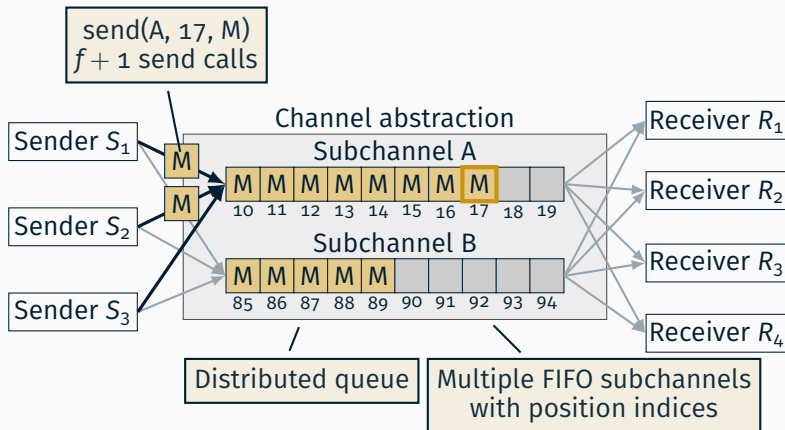


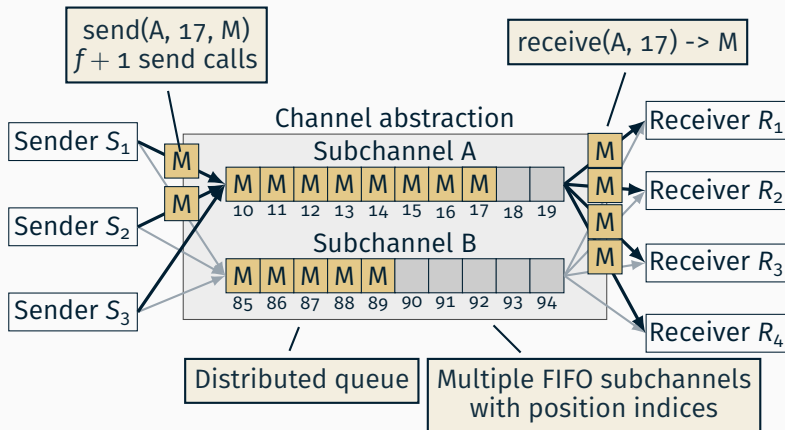


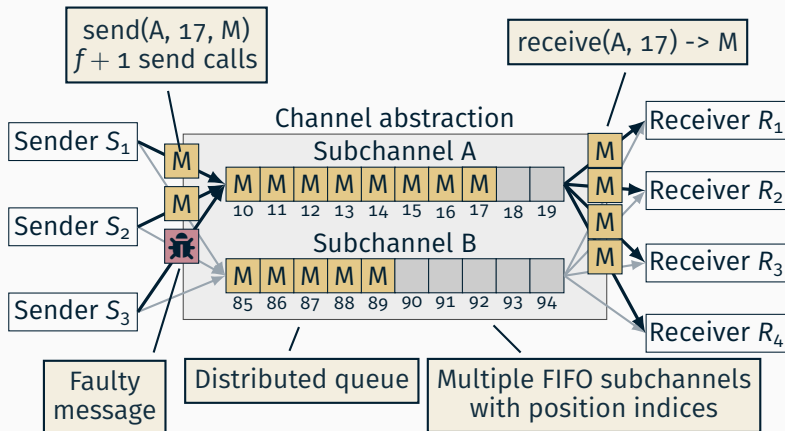


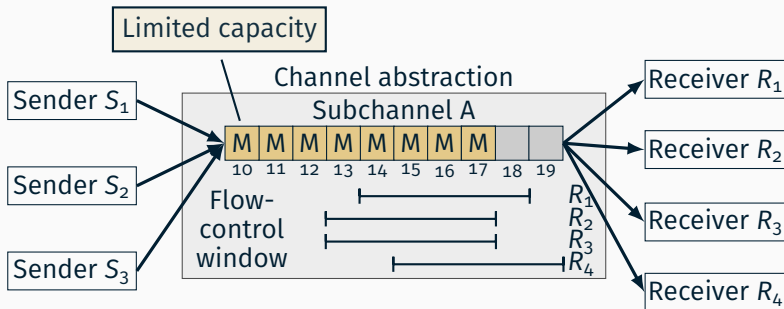


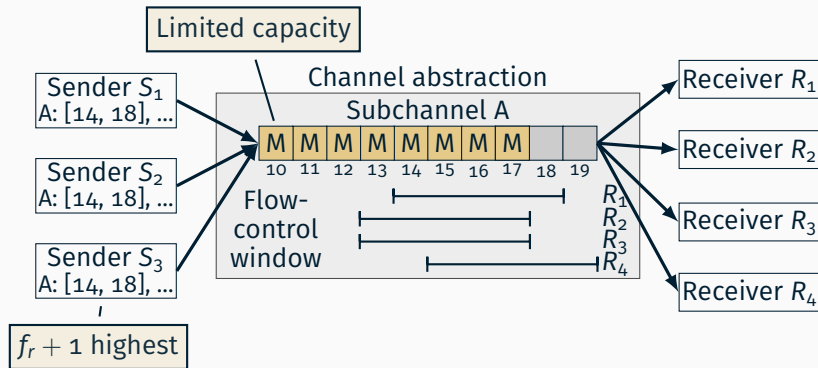








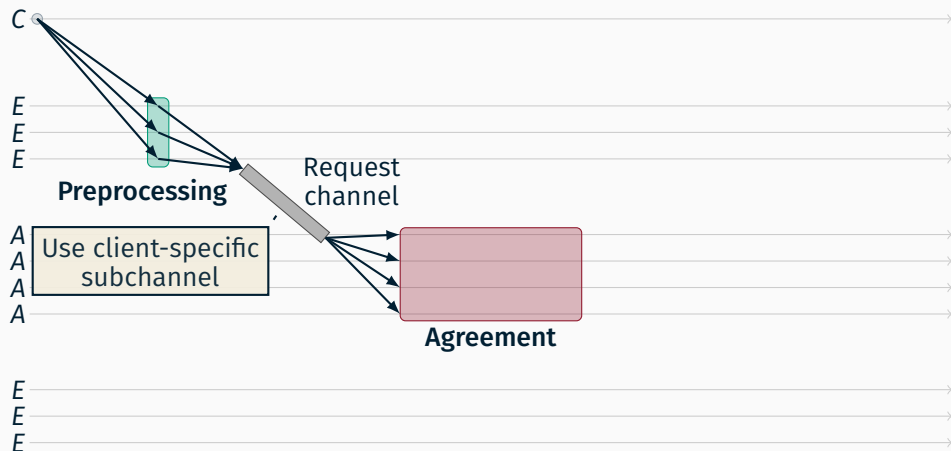


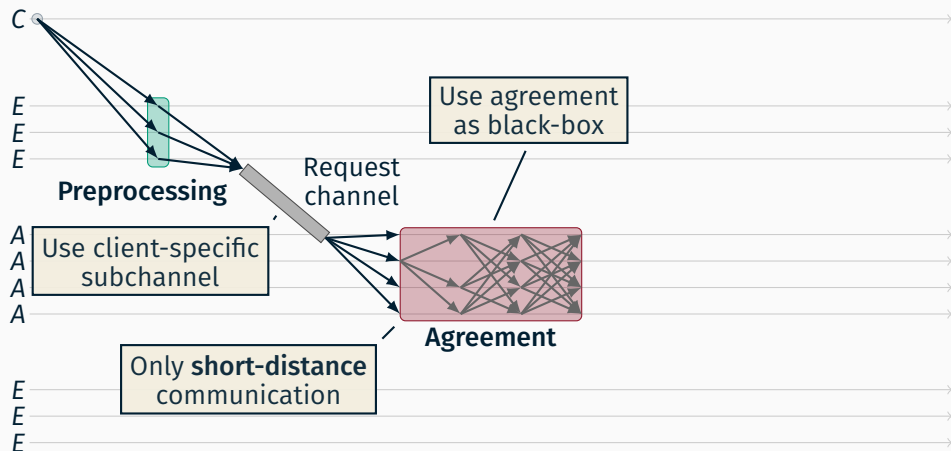


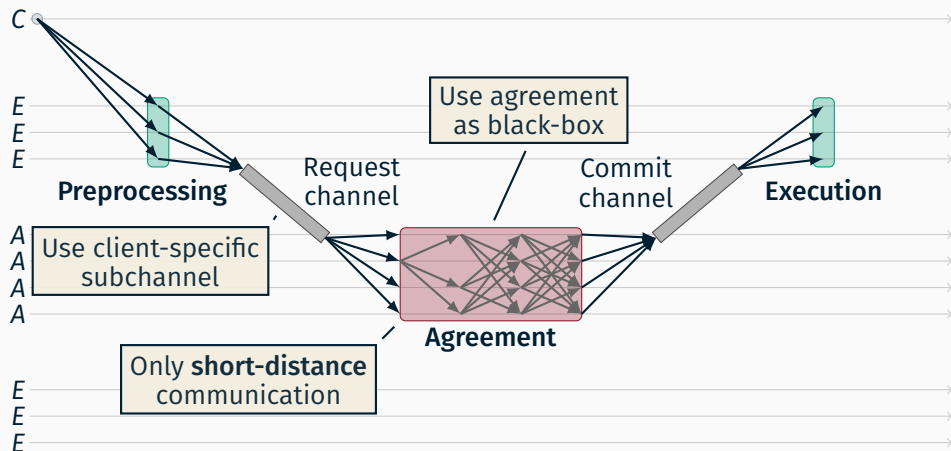


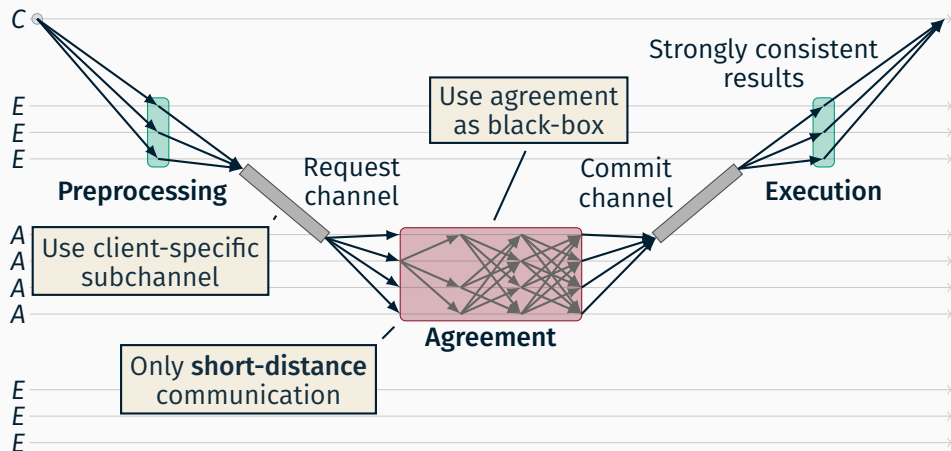


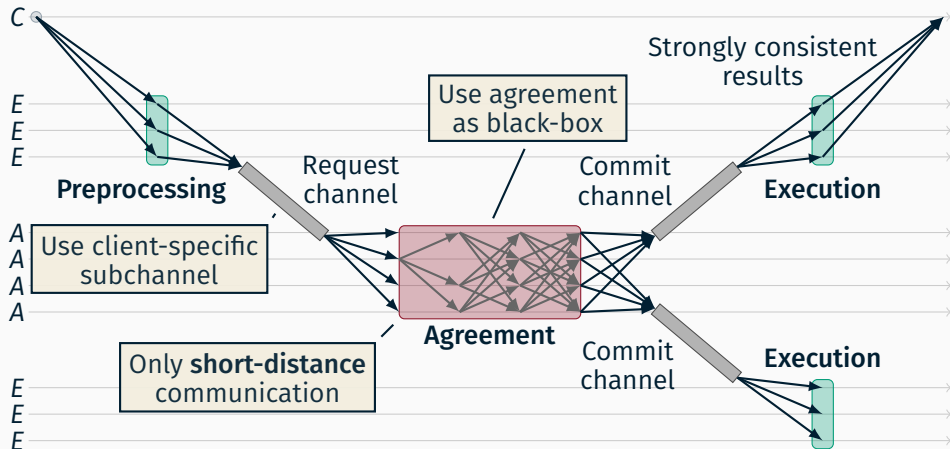


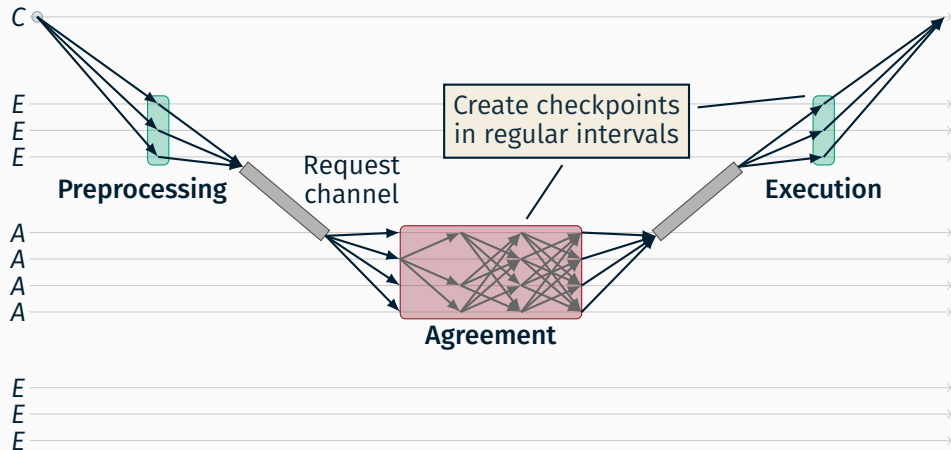


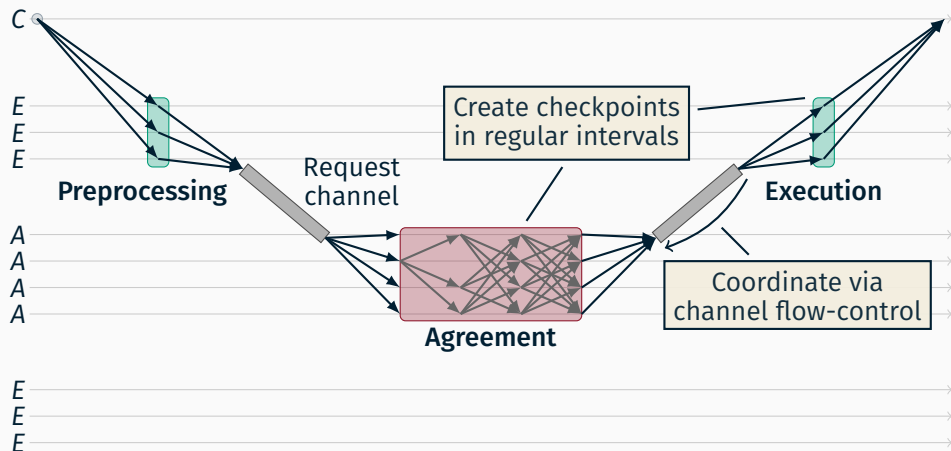


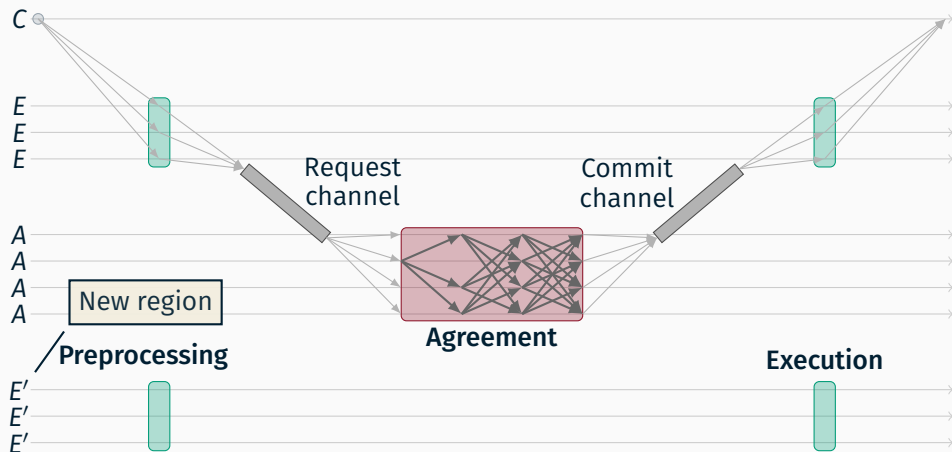


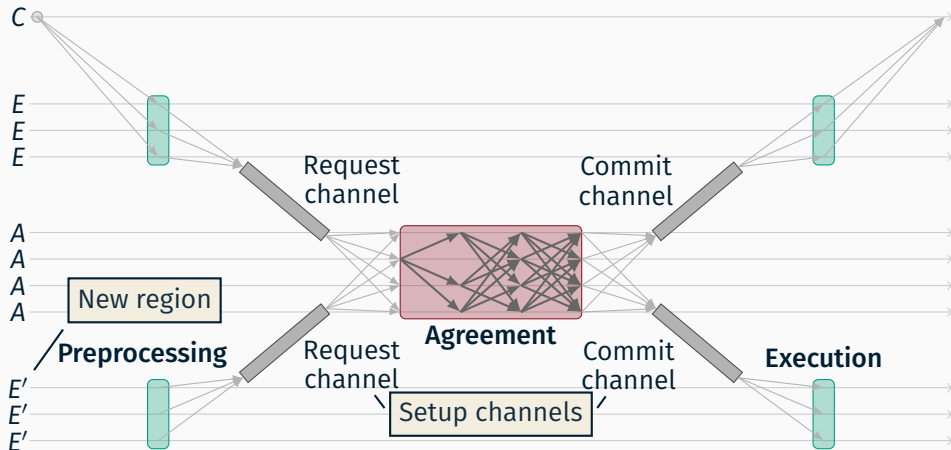


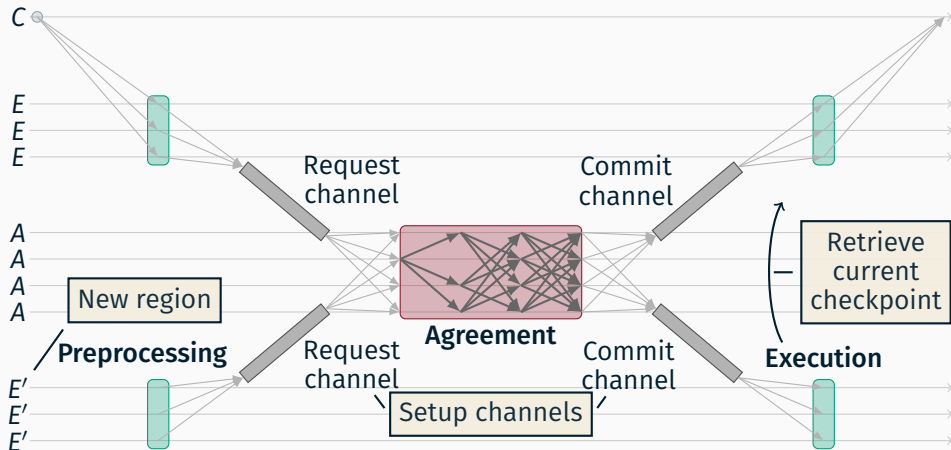


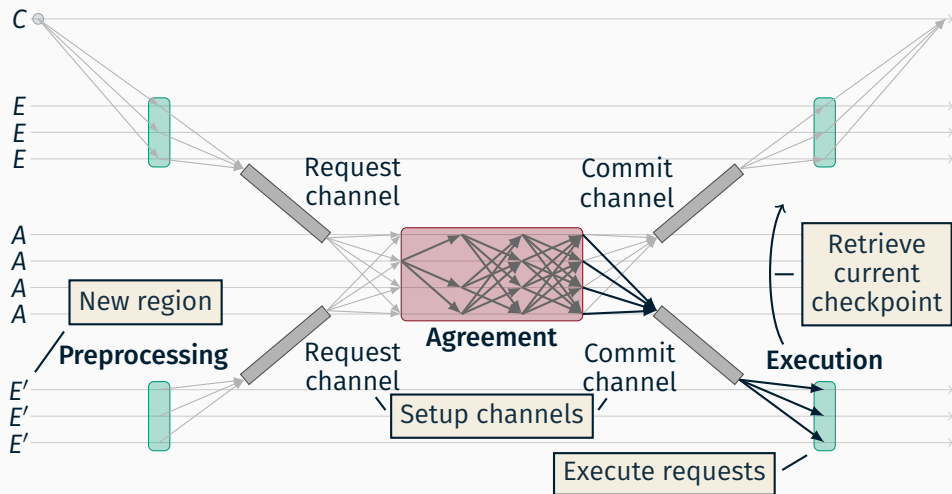


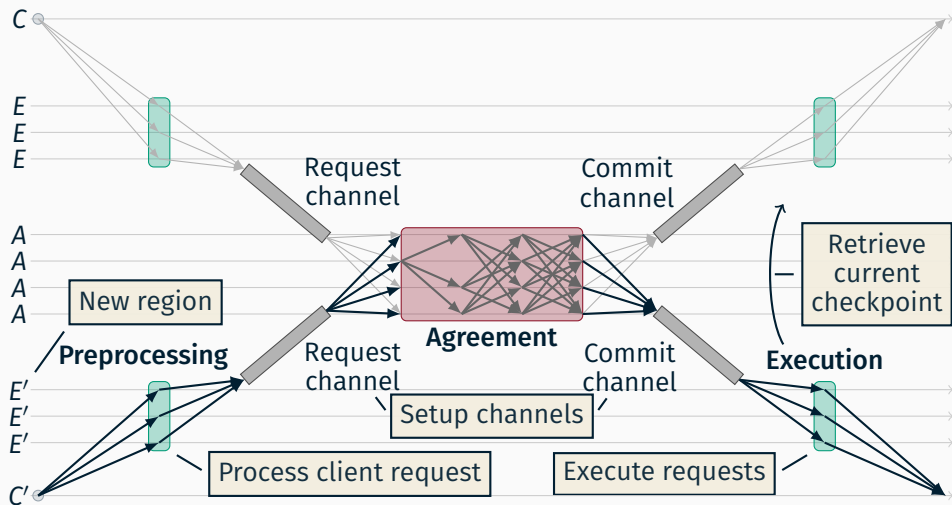








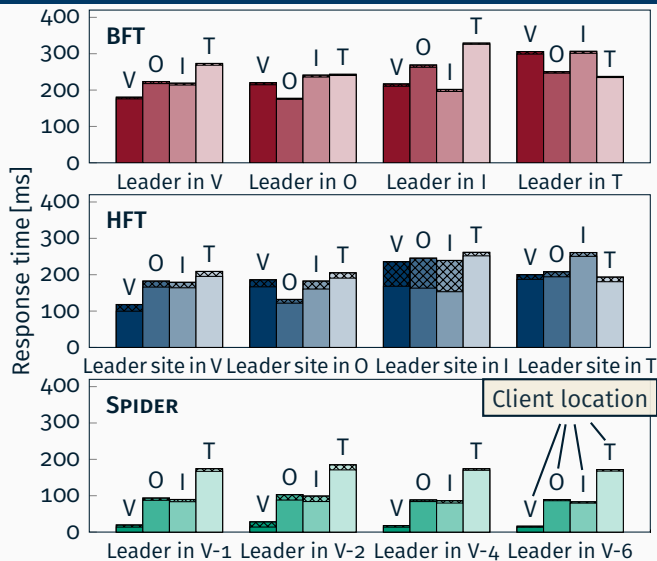




Evaluation

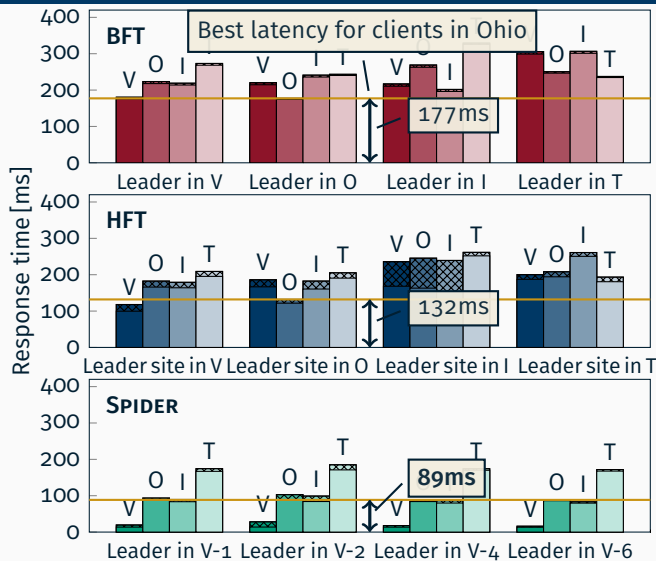
Evaluation: Write Requests (200 Bytes)

- Replicas in 4 AWS EC2 regions: **Virginia, Oregon, Ireland, Tokyo**
- 50 clients per region
- BFT**: PBFT with 1 replica per region
- HFT**: Steward with 4 replicas as cluster in each region
- SPIDER**: 4 agreement replicas in Virginia, 3 replicas per execution group per region



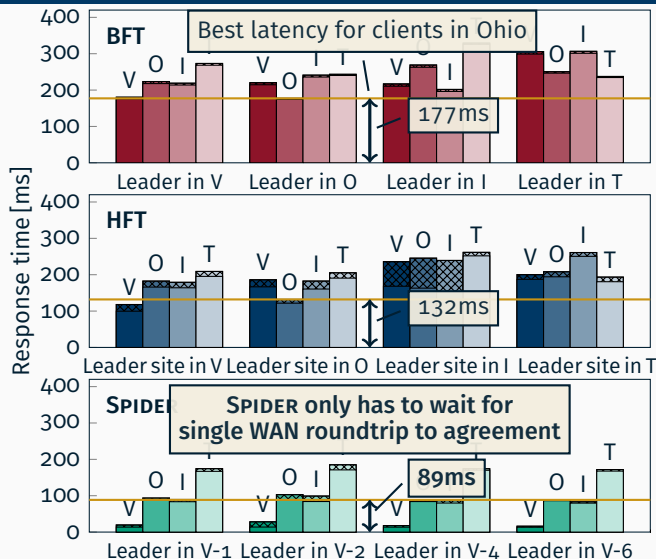
Evaluation: Write Requests (200 Bytes)

- Replicas in 4 AWS EC2 regions: **Virginia, Oregon, Ireland, Tokyo**
- 50 clients per region
- **BFT**: PBFT with 1 replica per region
- **HFT**: Steward with 4 replicas as cluster in each region
- **SPIDER**: 4 agreement replicas in Virginia, 3 replicas per execution group per region



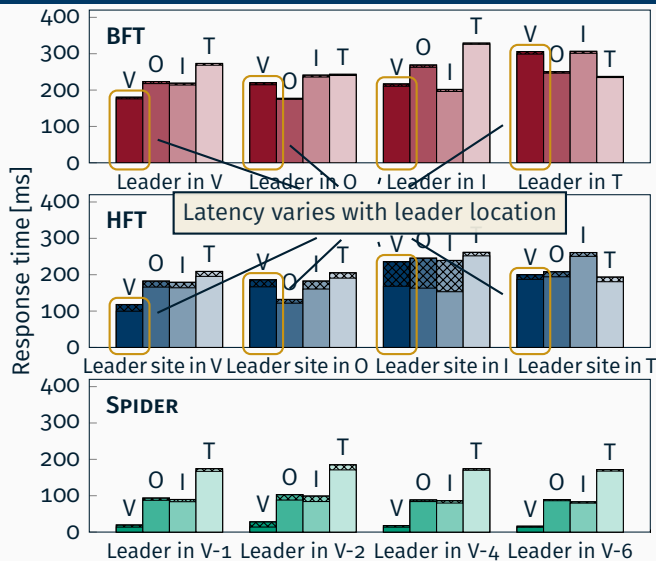
Evaluation: Write Requests (200 Bytes)

- Replicas in 4 AWS EC2 regions: **Virginia, Oregon, Ireland, Tokyo**
- 50 clients per region
- **BFT**: PBFT with 1 replica per region
- **HFT**: Steward with 4 replicas as cluster in each region
- **SPIDER**: 4 agreement replicas in Virginia, 3 replicas per execution group per region



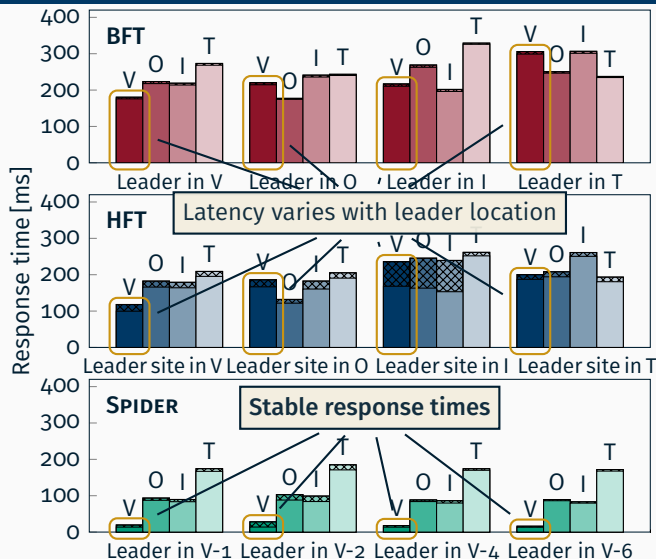
Evaluation: Write Requests (200 Bytes)

- Replicas in 4 AWS EC2 regions: **Virginia, Oregon, Ireland, Tokyo**
- 50 clients per region
- BFT**: PBFT with 1 replica per region
- HFT**: Steward with 4 replicas as cluster in each region
- SPIDER**: 4 agreement replicas in Virginia, 3 replicas per execution group per region



Evaluation: Write Requests (200 Bytes)

- Replicas in 4 AWS EC2 regions: **Virginia, Oregon, Ireland, Tokyo**
- 50 clients per region
- BFT**: PBFT with 1 replica per region
- HFT**: Steward with 4 replicas as cluster in each region
- SPIDER**: 4 agreement replicas in Virginia, 3 replicas per execution group per region



Summary

Problem

- Performance depends on leader location
- Either high latency or high complexity
- Best replica locations depend on client locations

SPIDER

- Efficient: IRMCs forward group decisions
- Modular: Decoupled agreement and execution groups
- Adaptable: Add or remove execution groups at runtime

More details in the paper

- Different possible Inter-Regional Message Channel (IRMC) implementations
- Handling malicious clients and other attacks