

# BLOXY: Providing Transparent and Generic BFT-Based Ordering Services for Blockchains

Symposium on Reliable Distributed Systems 2019

Signe Rüsçh, Kai Bleeke, Rüdiger Kapitza. TU Braunschweig, Germany.

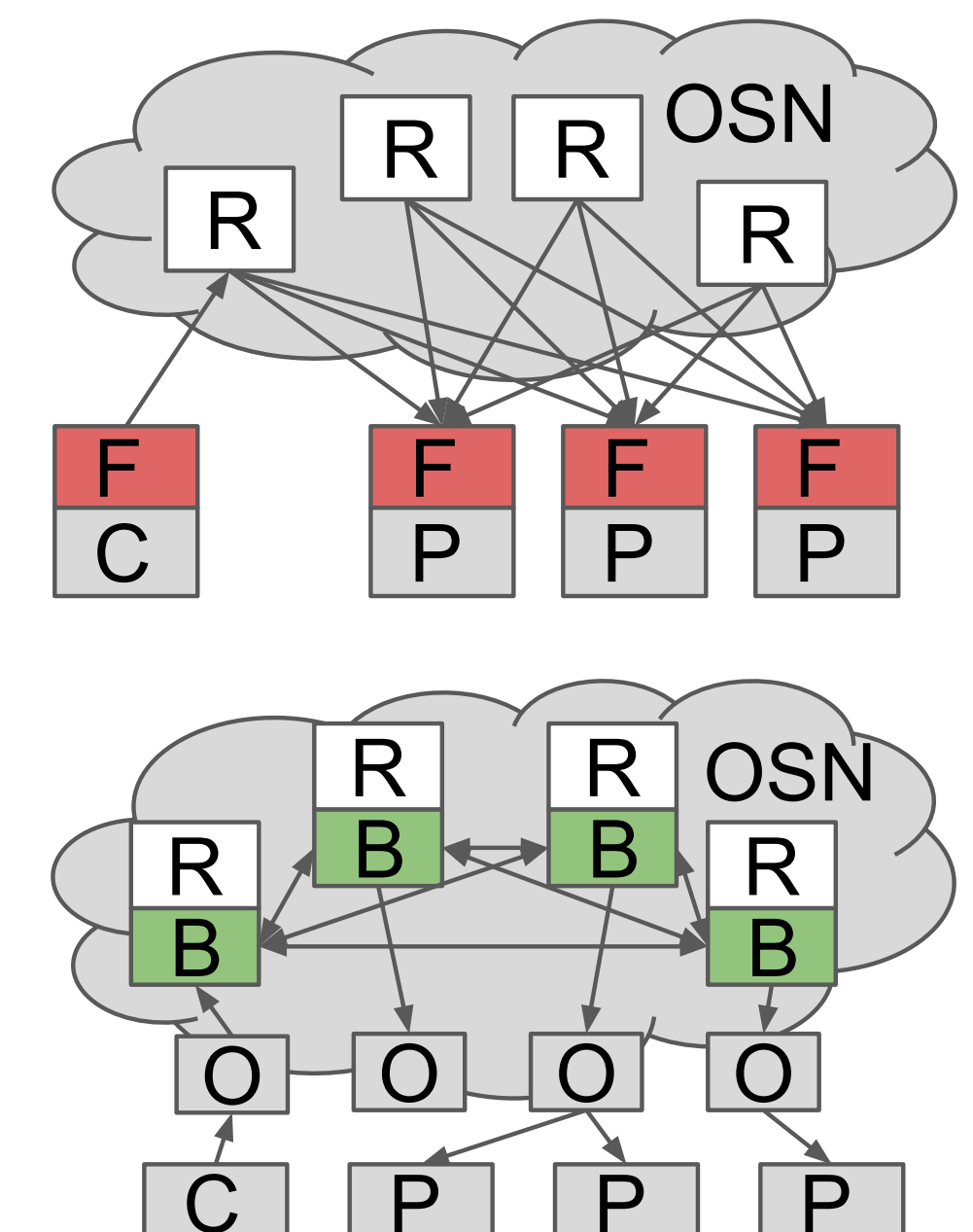
{ruesch|bleeke|kapitza}@ibr.cs.tu-bs.de

## Motivation

- Byzantine fault tolerant protocols (BFT):
  - Consensus despite arbitrary behaviour
  - Block creation for blockchains
  - Optimized** for different aspects
- Permissioned** blockchains: several use cases
- Different use cases have different requirements
- No **one-size-fits-all** BFT solution!
- **Generic consensus protocols for blockchains!**

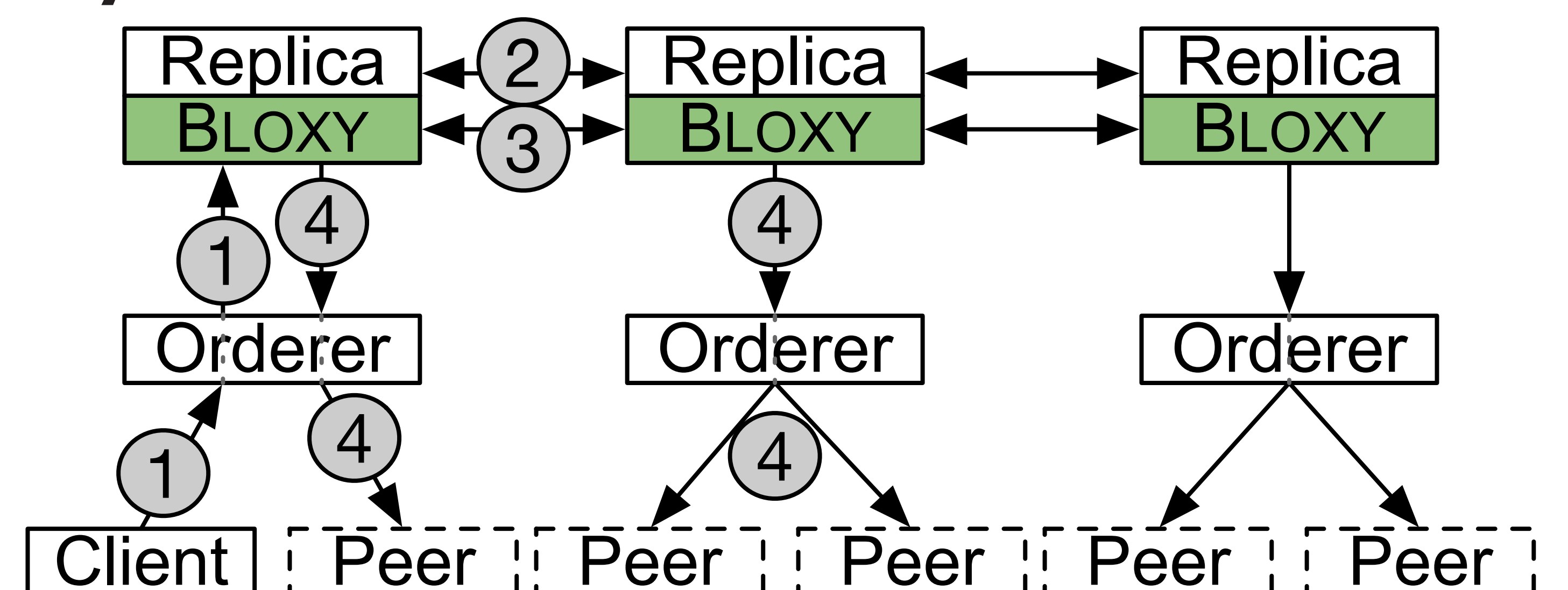
## Hyperledger Fabric [Androulaki et al., EuroSys'18]

- Permissioned blockchain platform
- Modular design with **pluggable consensus**
- BFT-SMART** integration [Sousa et al., DSN'18]
  - **"Frontend"**: Fabric orderer now part of peers
- BFT client functions compromise modularity
- Re-implementation for all protocols needed
- Shift **BFT client** functionality onto **replicas** to maintain Fabric's modularity!



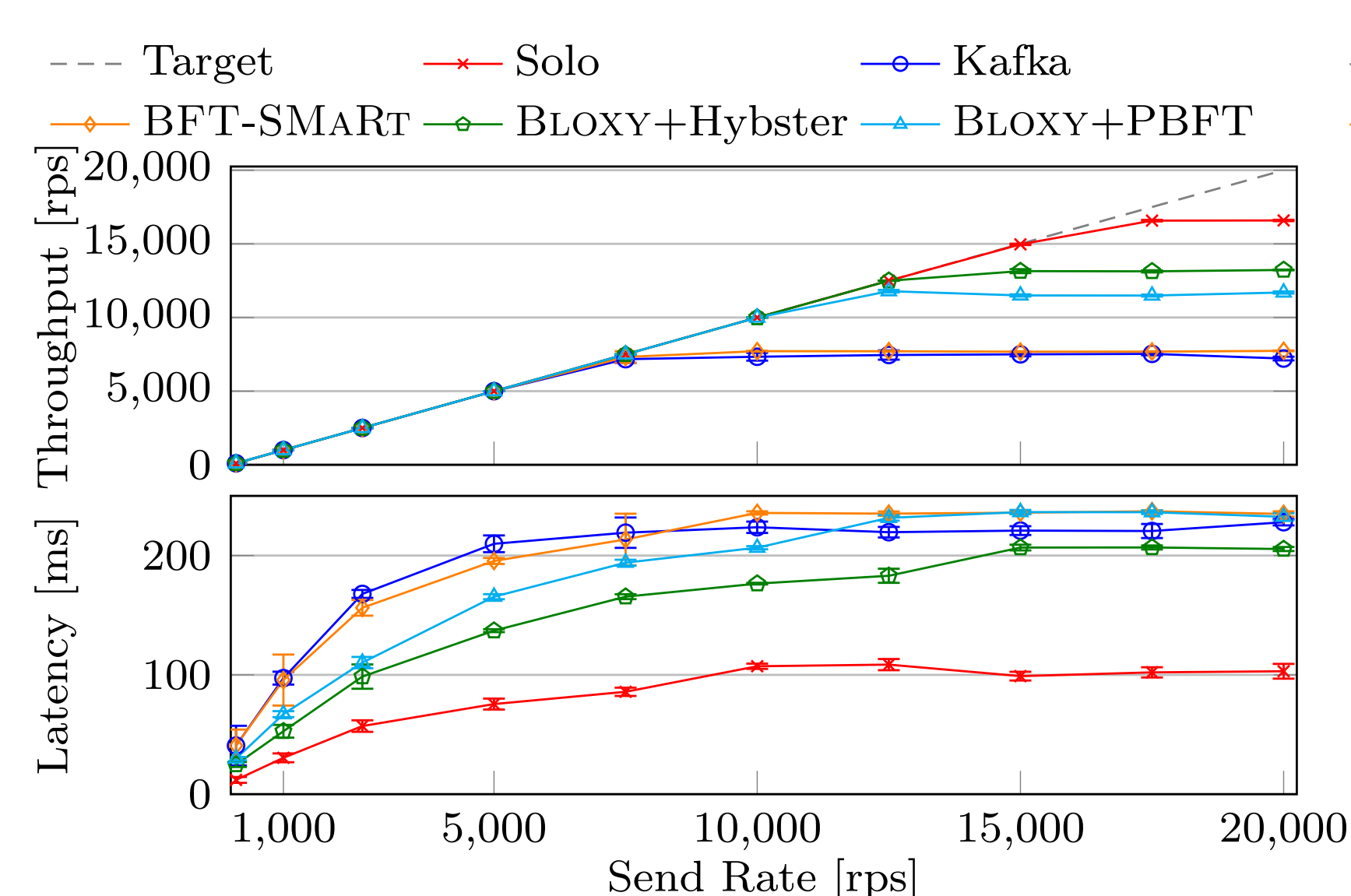
## BLOXY: A Blockchain-Aware Trusted Proxy

- Consensus proxy** for Fabric encapsulates BFT client functionality
- Transparent** access to a **generic** ordering service
- Leverage **trusted execution** for trusted operations on replicas
  - Hybrid fault model: subsystem can only fail by crashing
- Proof-of-concept integration of BFT protocols:
  - Hybrid protocol **Hybster** [Behl et al., EuroSys'17]
  - Conventional protocol **PBFT** [Castro et al., OSDI'99]
- BLOXY functionality:
  - Establishing secure connections
  - Forwarding transactions as BFT requests
  - Majority voting
  - Block dissemination



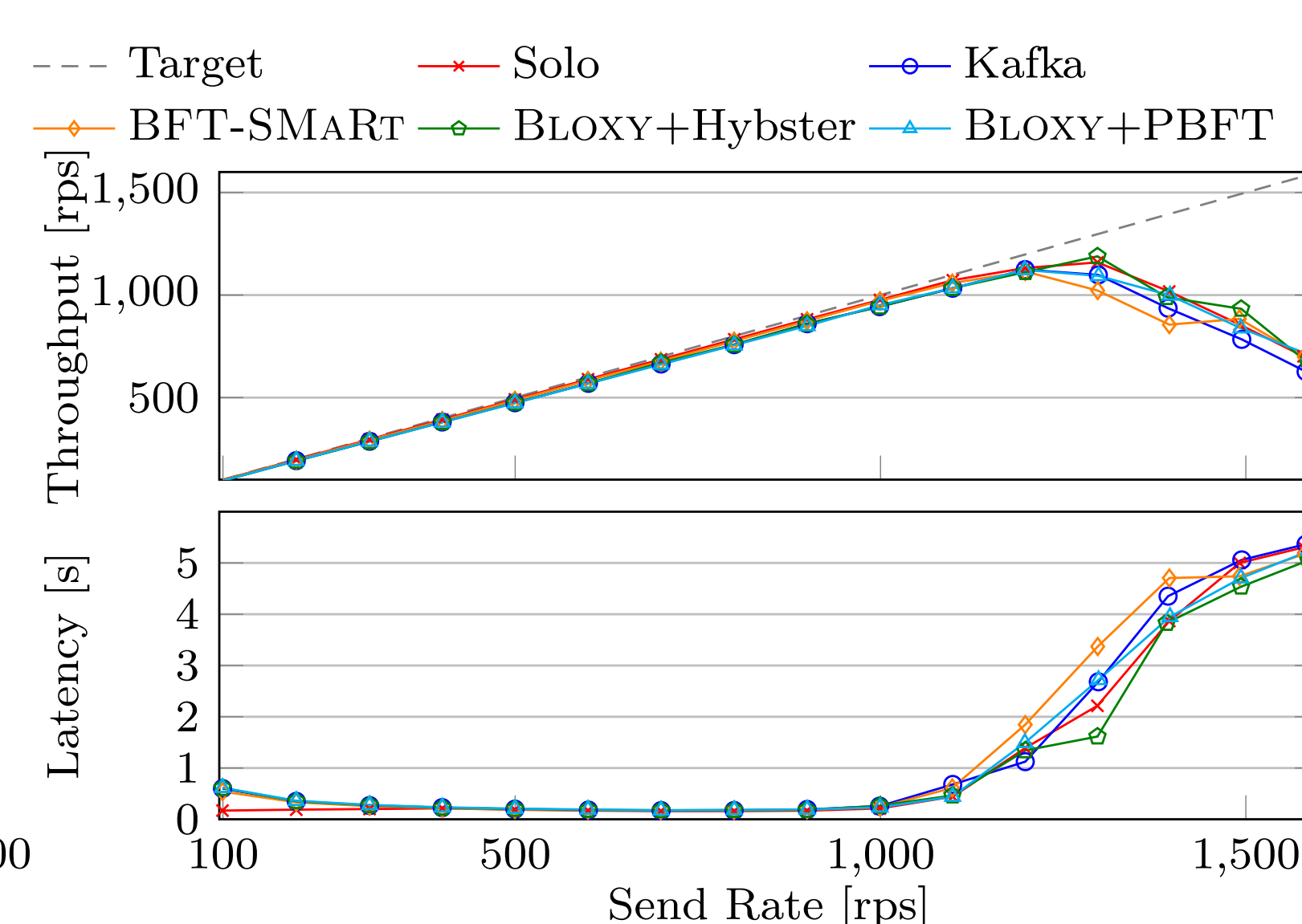
- Transaction forwarded to BLOXY via orderer
- Replicas order BFT request from BLOXY
- BLOXYs get block, exchange hashes, perform majority voting
- Blocks are disseminated to all connected nodes

## Evaluation



Microbenchmarks of ordering service:

- Lower communication overhead** of BLOXY-integrated BFT protocols
- **Improved** latency and throughput



Macrobenchmarks of Fabric network:

- Evaluated using Hyperledger Caliper
- Ordering Service has **no noticeable effect** on Fabric's performance

➤ **Generic consensus at no performance cost!**

## Block Storage

- Ensure all peers have all blocks in correct order
- Short-Term Block Storage** on replicas
- Store a block until  $f + 1$  orderers received it
- Orderers can query for missing blocks
- Tolerate **Byzantine orderer** nodes

## Conclusion

- Consensus proxy** for Hyperledger Fabric
- Encapsulation of **BFT client functionality**
- Easy integration** of new consensus protocols
- BLOXY maintains Fabric's **modularity**
- Good performance** compared to directly integrated BFT protocols