Separating Agreement from Execution for Byzantine Fault-Tolerant Services

Rethinking Replicated State Machines

Jian Yin, Jean-Philippe Martin, Arun Venkataramani,

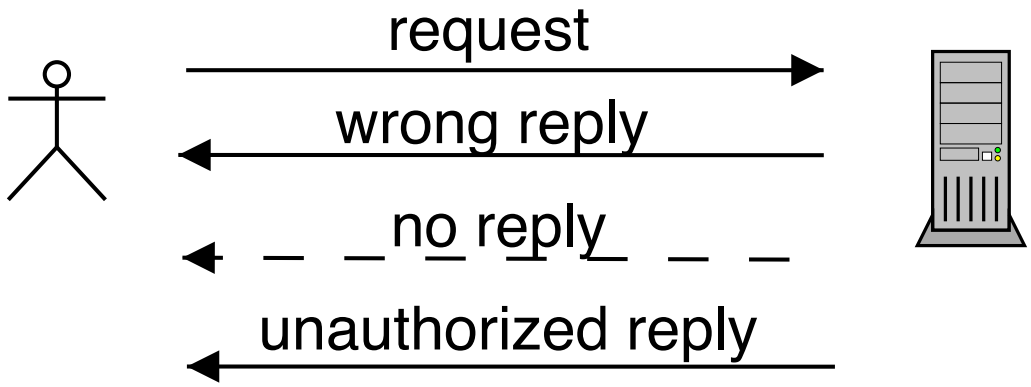
Lorenzo Alvisi and Mike Dahlin

jianyin@us.ibm.com, {jpmartin,arun,lorenzo,dahlin}@cs.utexas.edu

Laboratory for Advanced Systems Research (LASR)

The University of Texas at Austin

Problem: Tolerating Byzantine Faults

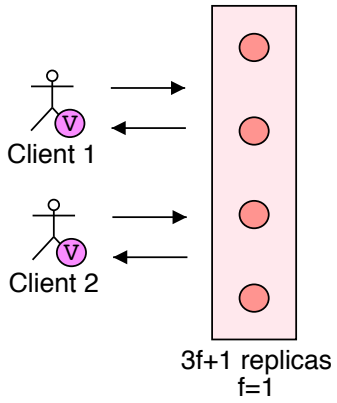


- Current solution: replicated state machine
  - ◊ $3f + 1$ versions of service
  - ◊ Hurts confidentiality
- Our solution: rethinking replicated state machine
  - ◊ Cheaper: $2f + 1$ versions of service
  - ◊ Helps confidentiality

Outline

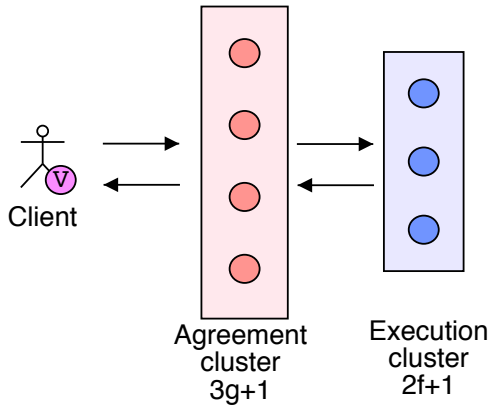
- Introduction
- Separating Agreement from Execution
- Enables
  - ◊ Fewer service replica
  - ◊ Confidentiality
- Prototype
- Conclusion

Current Solution



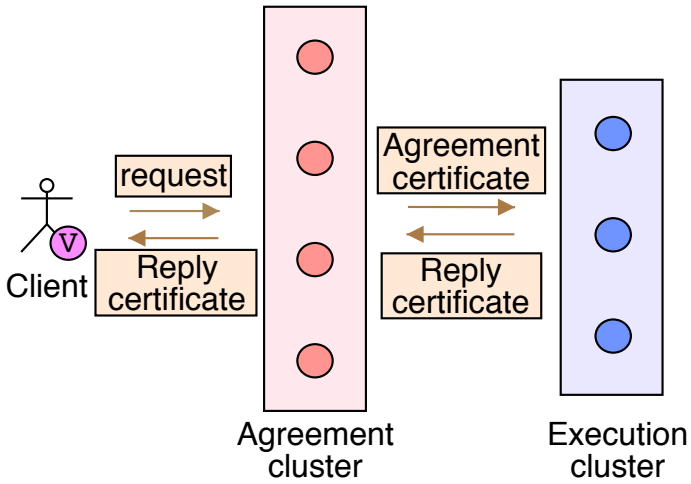
- Client
  - ◊ Send request and repeats
  - ◊ Pick majority reply
- Correct replica must return same reply
  - ◊ Start from same state
  - ◊ All replicas process the same requests in the same order (*replica coordination*)
- How
  - ◊ Replicated state machine protocol

Separating Agreement from Execution



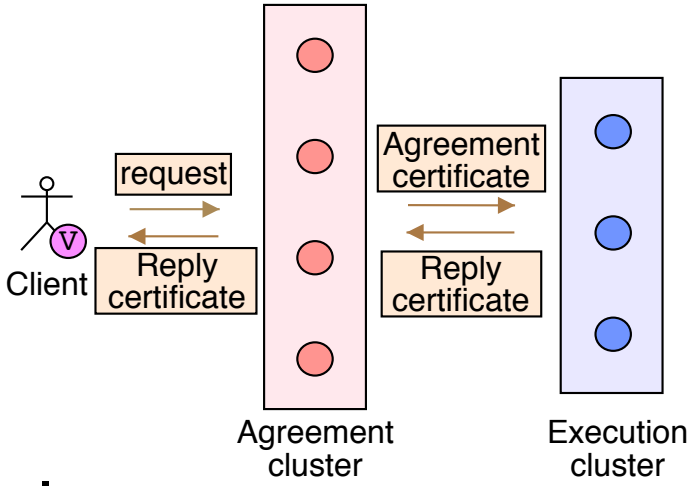
- Split problem into independent concerns
  - ◊ Agreement: All agree on sequence of requests
  - ◊ Execution: Requests executed in order
- Note different requirements
  - ◊ Agreement: $3g + 1$ servers, g faults
  - ◊ Execution: $2f + 1$ servers, f faults

Implementation



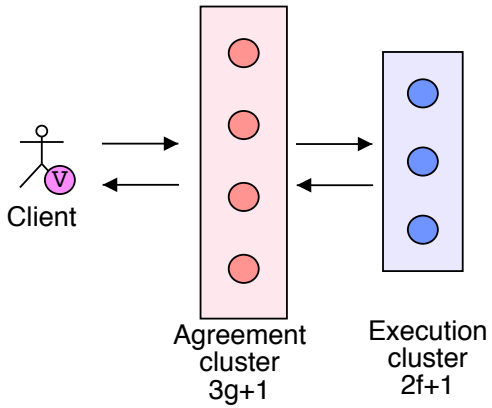
1. Assign unique sequence number to request
2. $\langle request, sequence\ number \rangle_A$: unique, certified
3. Execute in sequence order
4. $\langle reply, sequence\ number \rangle_E$: unique, certified

Cluster Implementation is Simple



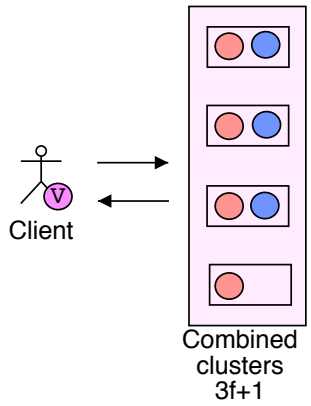
- Simple protocol
  - ◊ Agreement using traditional protocol
  - ◊ Send instead of executing
- Tricks in retransmission
  - ◊ Execution cluster internal retransmission
  - ◊ Confidential intercluster retransmission

Separation makes Replication Cheaper



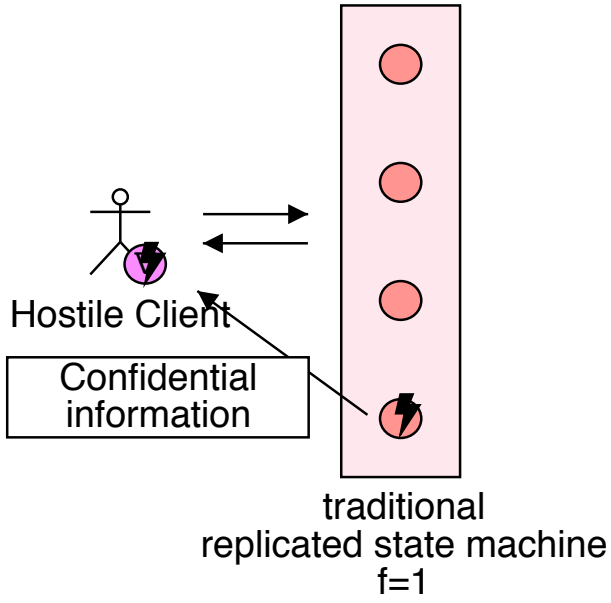
- Execution cluster
  - ◊ Fewer service replicas
  - ◊ Expensive because different
- Agreement cluster
  - ◊ Simple nodes, reusable
- Can merge

Separation makes Replication Cheaper



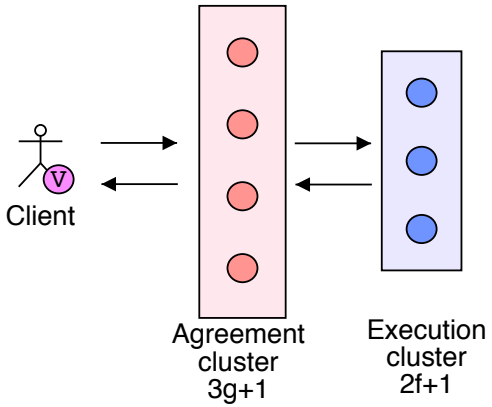
- Execution cluster
  - ◊ Fewer service replicas
  - ◊ Expensive because different
- Agreement cluster
  - ◊ Simple nodes, reusable
- Can merge

Confidentiality: The Problem



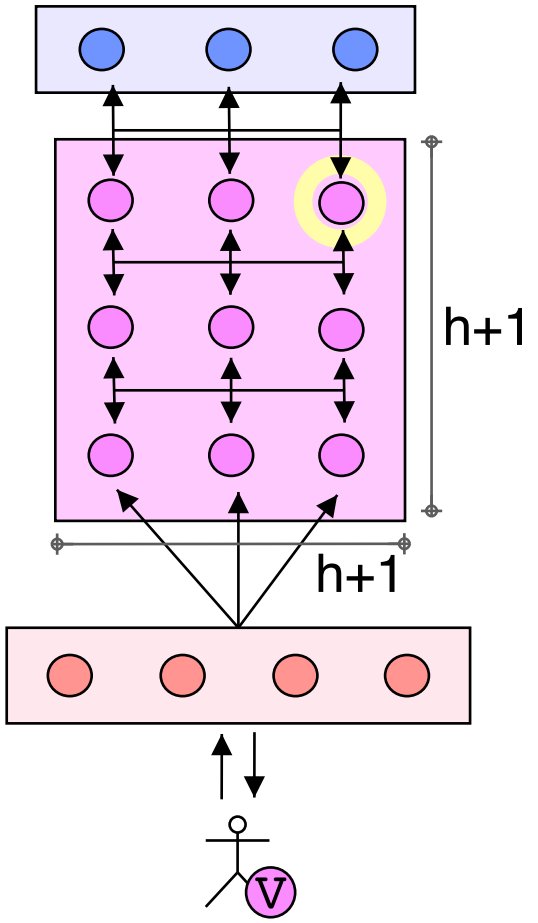
- Replication hurts confidentiality
- Privacy Firewall restores it

Separation Enables Confidentiality



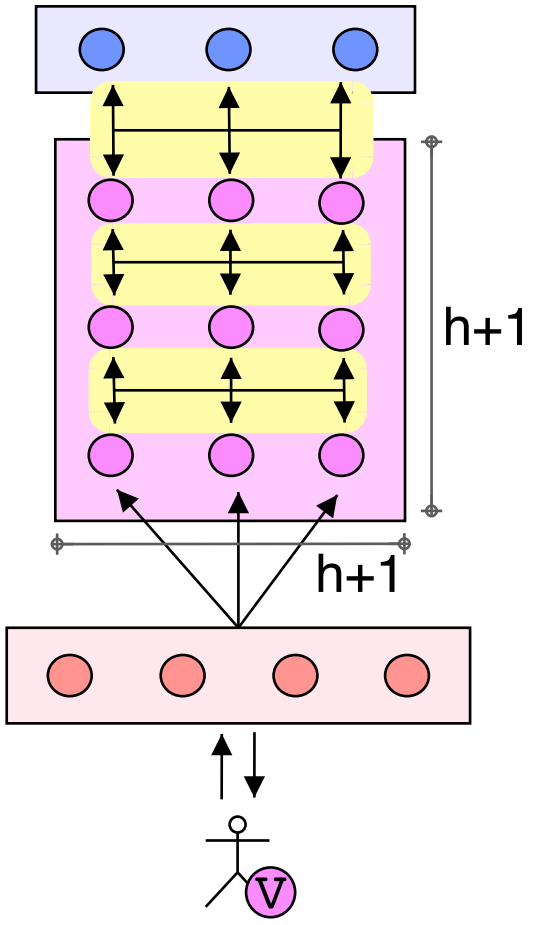
- Separation enables confidentiality
  - ◊ Agreement nodes as filters
- Key 1: Restrict communication
- Key 2: Separate choice from secrets
  - ◊ Choice in reply contents
  - ◊ Choice in who signs the reply certificate
  - ◊ Choice in retransmission
- One choice remains: speed

The Privacy Firewall



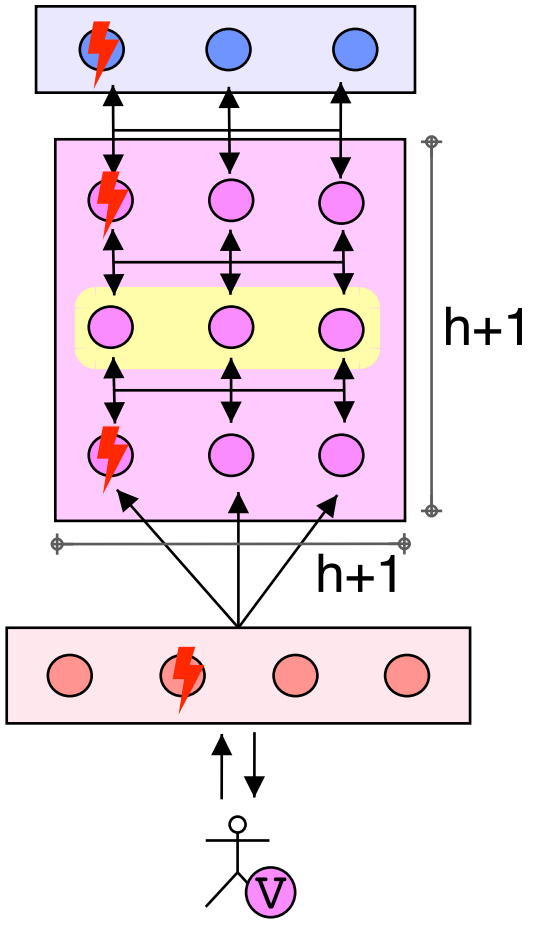
- Nodes check reply certificate
- Replicated for h Byzantine failures
- Restrict communication
- Only valid replies
  - ◊ $h+1$ rows $\Rightarrow$ one is correct
- Always reply
  - ◊ $h+1$ columns $\Rightarrow$ one is correct
- Minimal: $(h+1)^2$ servers

The Privacy Firewall



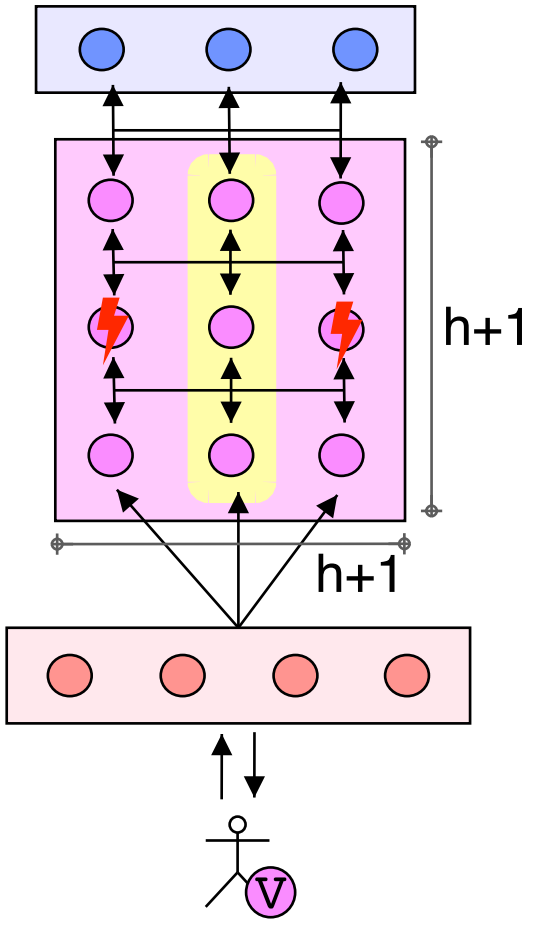
- Nodes check reply and order
- Replicated for h Byzantine failures
- Restrict communication
- Only valid replies
  - $h+1$ rows $\Rightarrow$ one is correct
- Always reply
  - $h+1$ columns $\Rightarrow$ one is correct
- Minimal: $(h+1)^2$ servers

The Privacy Firewall



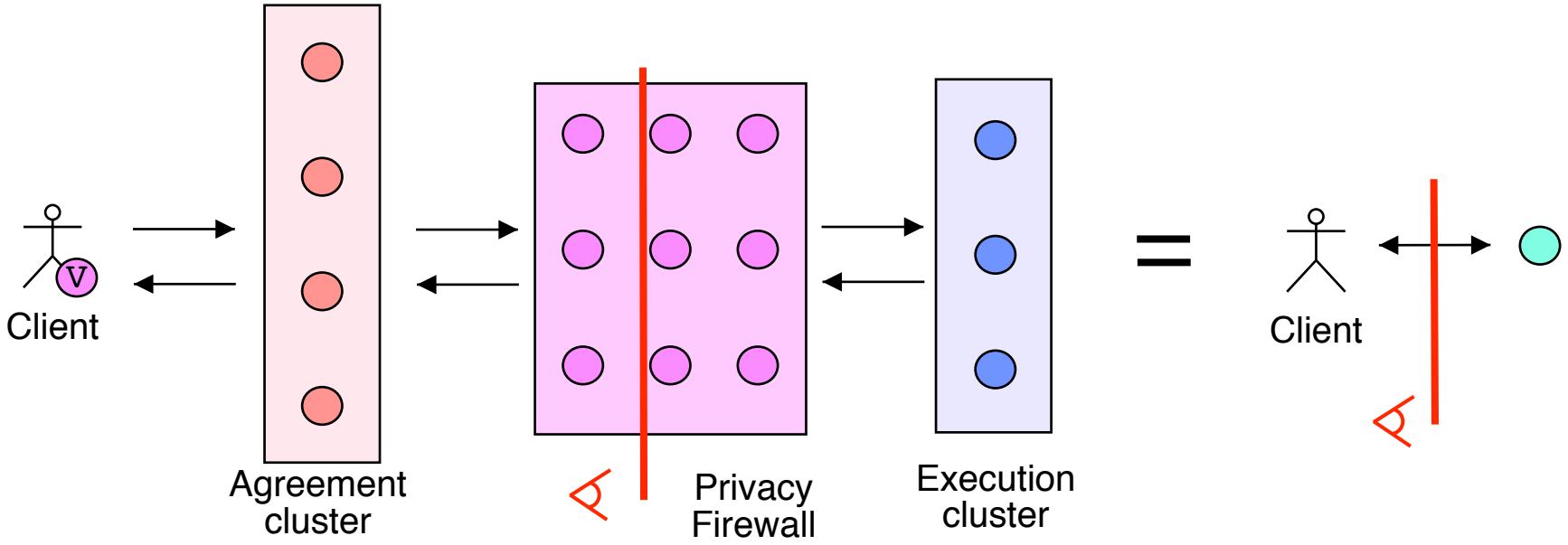
- Nodes check reply and order
- Replicated for h Byzantine failures
- Restrict communication
- Only valid replies
  - $h+1$ rows $\Rightarrow$ one is correct
- Always reply
  - $h+1$ columns $\Rightarrow$ one is correct
- Minimal: $(h+1)^2$ servers

The Privacy Firewall



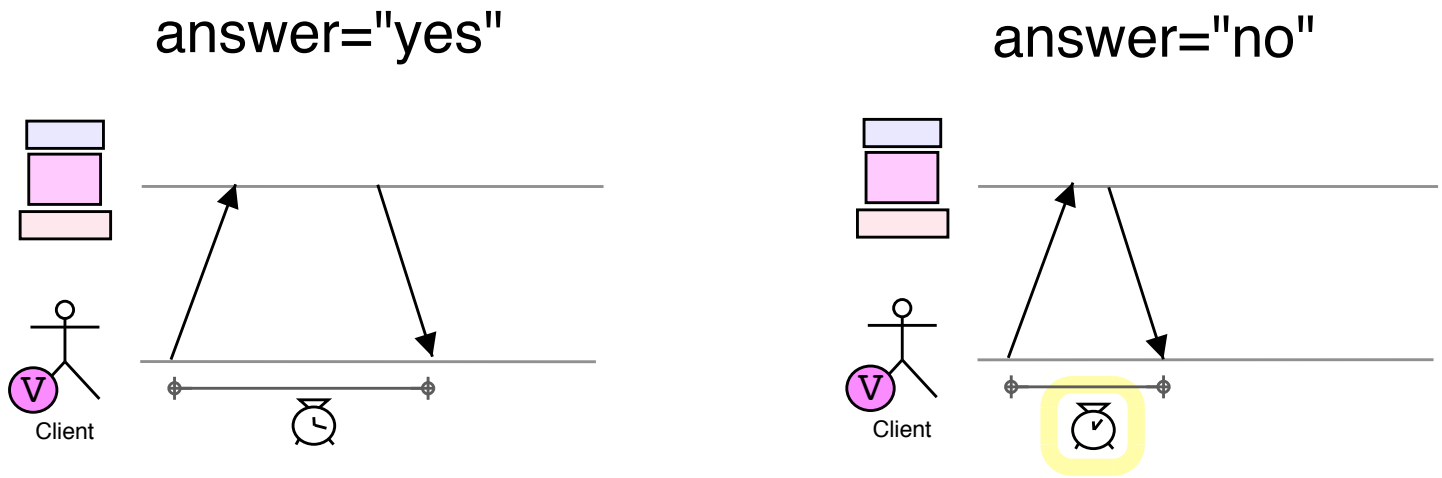
- Nodes check reply and order
- Replicated for h Byzantine failures
- Restrict communication
- Only valid replies
  - ◊ $h+1$ rows $\Rightarrow$ one is correct
- Always reply
  - ◊ $h+1$ columns $\Rightarrow$ one is correct
- Minimal: $(h+1)^2$ servers

Privacy Firewall Guarantees



- *Output set confidential*
  Output of correct cut is a valid output for a correct node through unreliable link
- Only *correct replies* get through
  - ◊ Replies that correct nodes send

Timing Attacks Remain

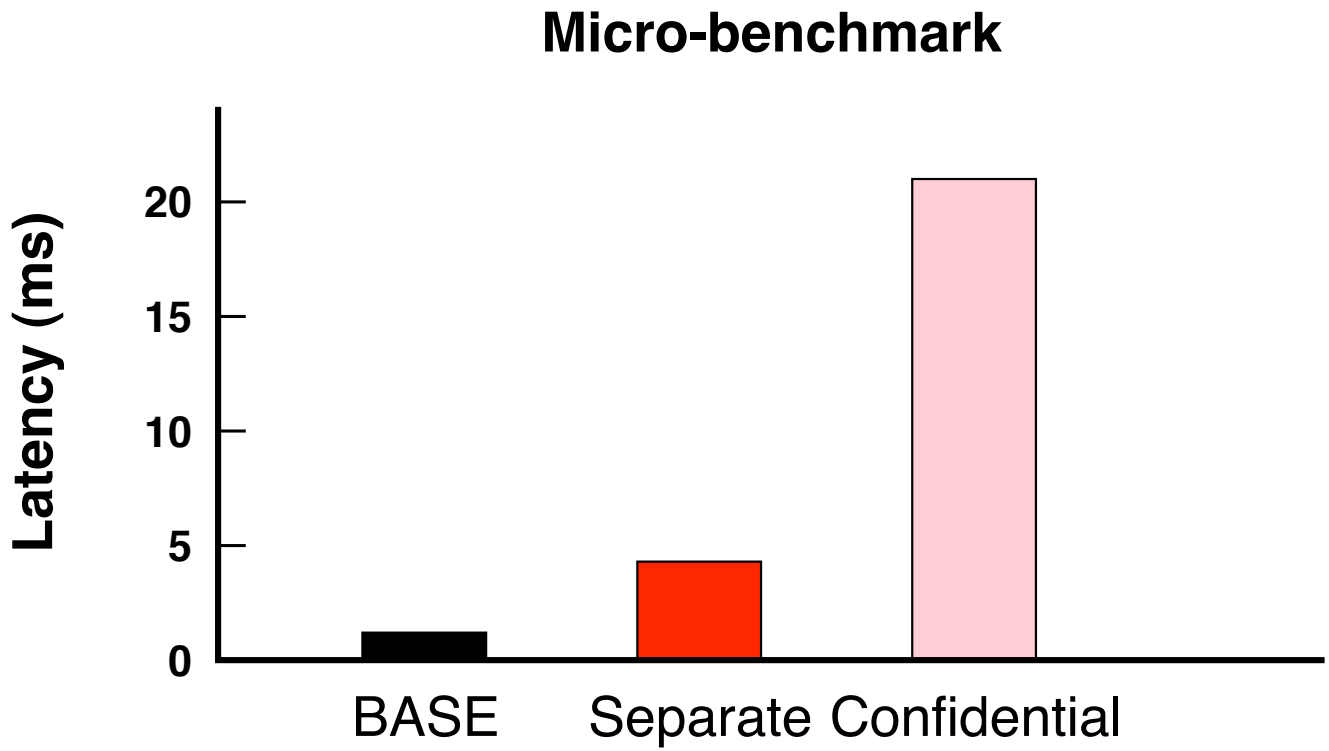


- One choice remains: execution speed
- Faulty execution server can influence when majority forms
- Information-theoretic confidentiality impossible without synchrony

Prototype

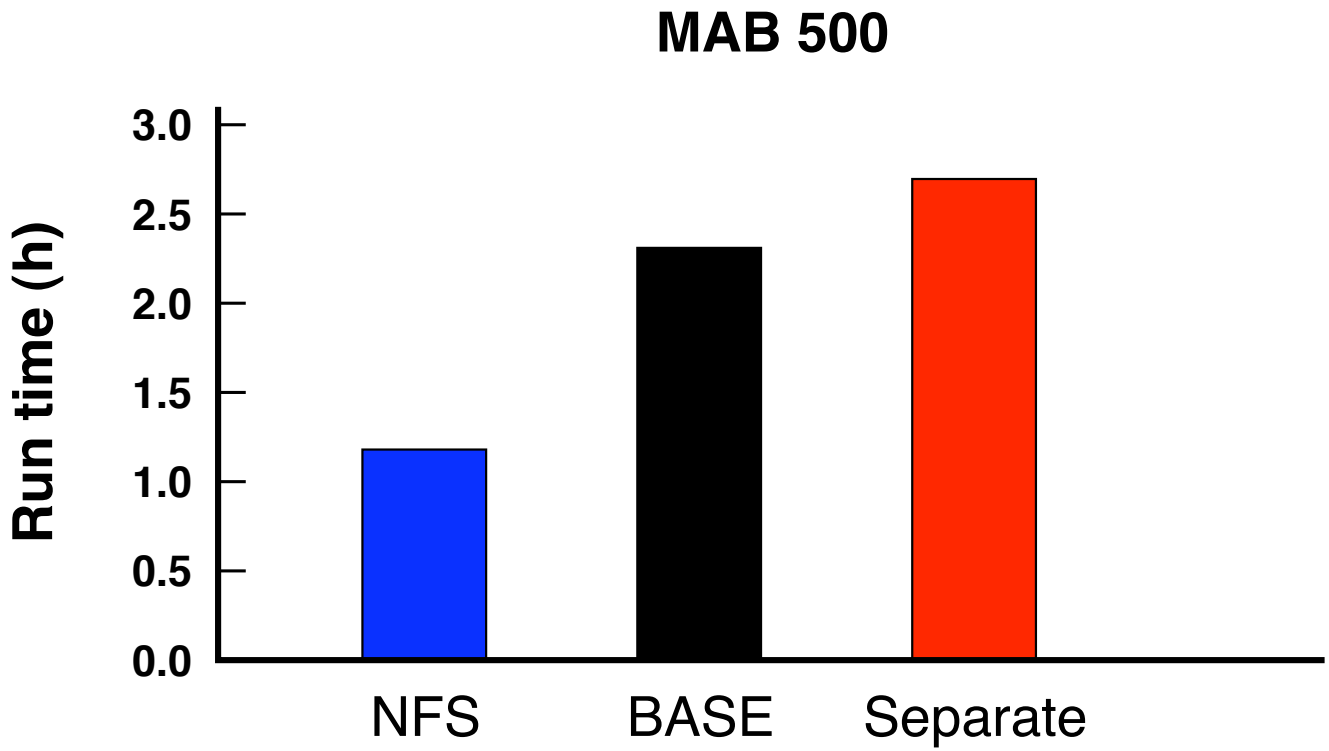
- Built prototype from BASE [Rodrigues01]
- Implements BFT confidential network file system
- 10 machines: 1 client, 4 ag+PF, 2 PF, 3 exec.
  - ◊ Tolerate 1 fault in each of agreement, PF, exec.
  - ◊ 128MB RAM, 100Mbps switch
- Limitations of prototype
  - ◊ No uninterruptible power supply
  - ◊ Same code
  - ◊ Communication not restricted

Latency Micro-Benchmarks



- Micro-benchmark latency
  - ◊ Removed some BASE optimizations
  - ◊ Only implemented one of six optimizations

Good Performance



- Separation and PF perform well in benchmarks
  - ◊ +16% for confidentiality

Conclusion

- Take home message:

  Separate agreement from execution!

- Benefits
  - ◊ Fewer service replicas
  - ◊ Privacy Firewall
  - ◊ Easy