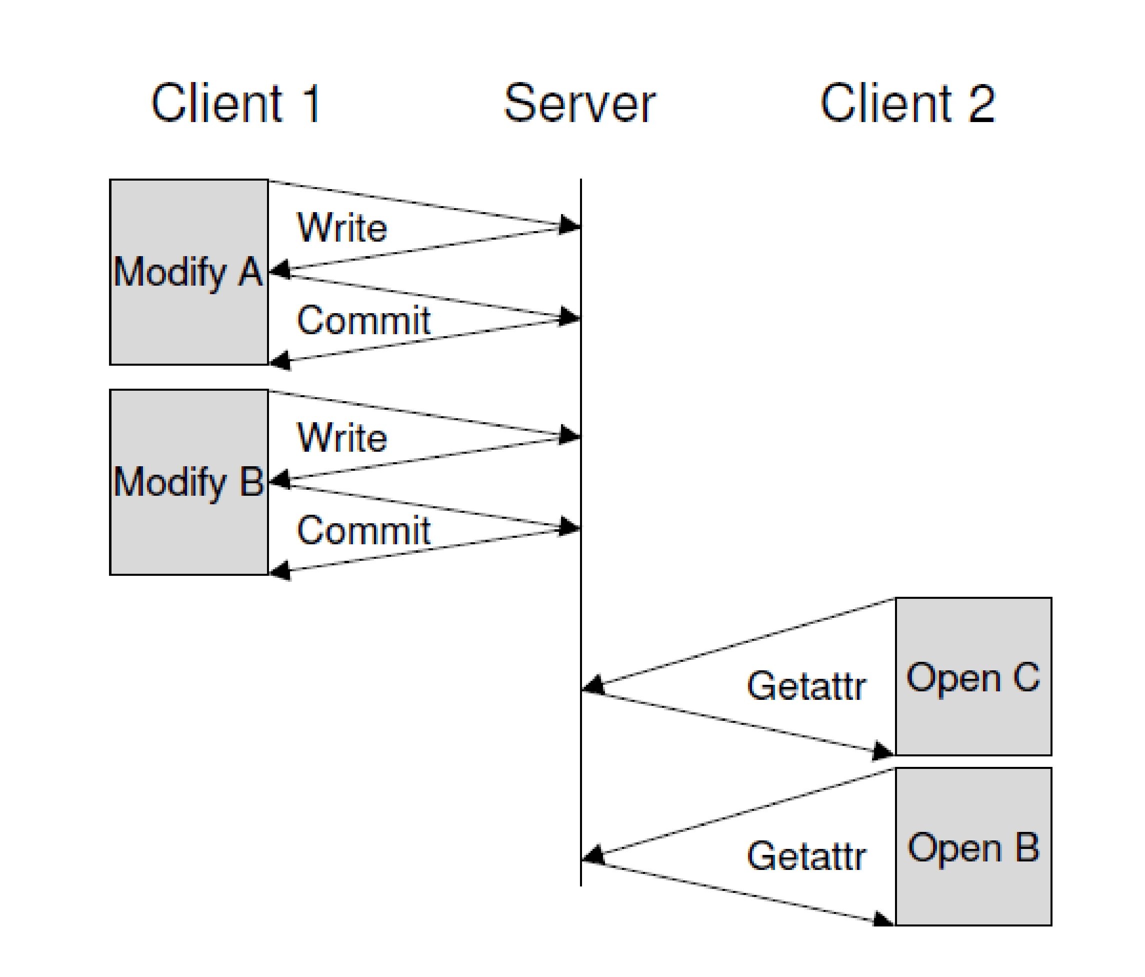
Speculative Execution

IN A DISTRIBUTED FILE SYSTEM

Consistency is Slow

- Synchronous client-server communication
  - Even slow over LAN
  - Performance vs. Consistency and Safety
  - Close-to-open consistency (NFS)



Synchronous Execution

Speculative Execution in a Distributed File System
Edmund B. Nightingale, Peter M. Chen, and Jason Flinn

What can we do?

What can we Take Advantage of?

#1: Filesystems know themselves.

#2: Rollback to a checkpoint > server round-trip time.

#3: Modern CPUs have resources to spare.

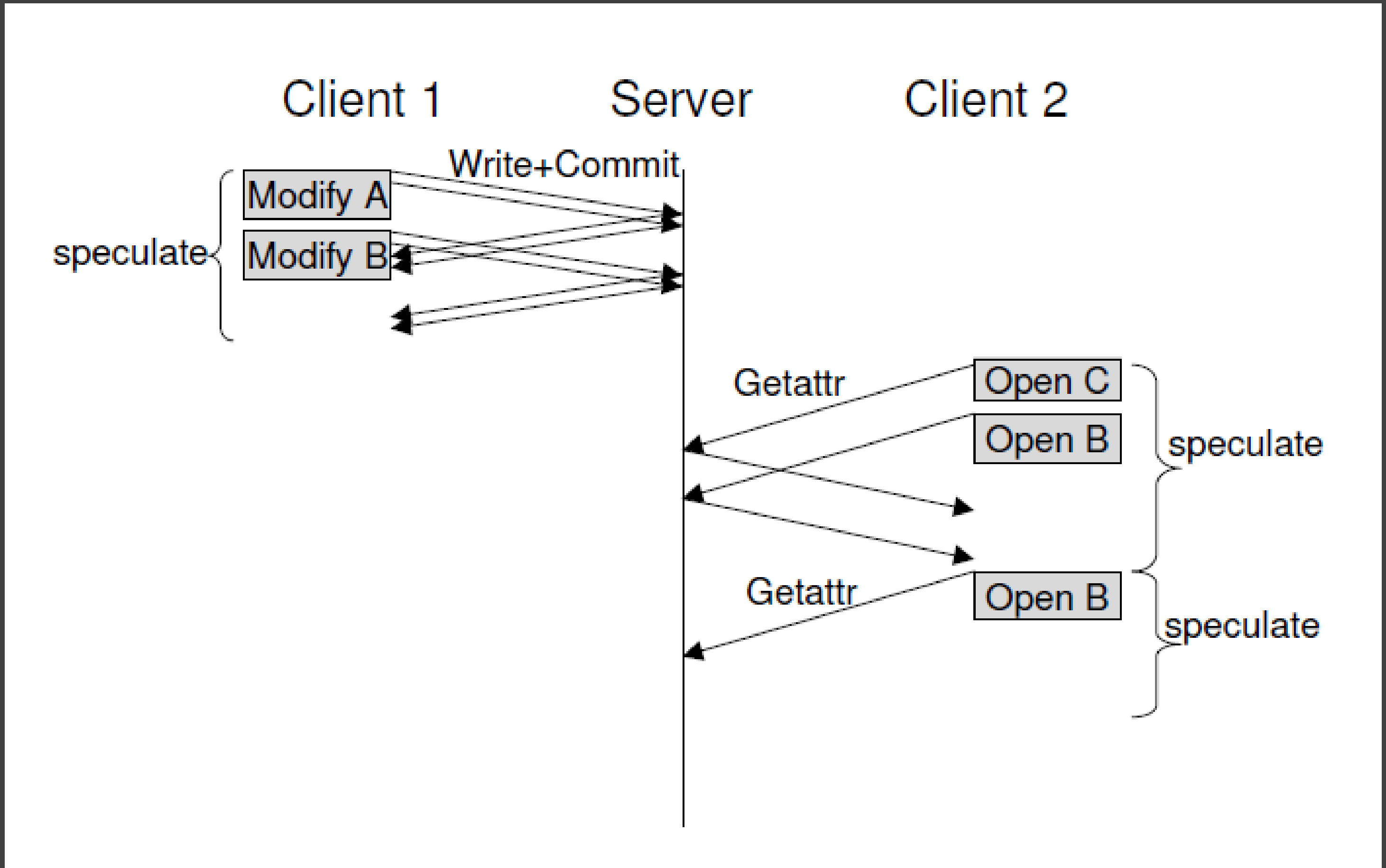
Speculator

Speculative Execution in a Distributed File System Edmund B. Nightingale, Peter M. Chen, and Jason Flinn

Speculator

- Speculative (asynchronous) execution.
- Lightweight check-pointing and rollback.

Speculative Execution in a Distributed File System Edmund B. Nightingale, Peter M. Chen, and Jason Flinn



Speculative Execution

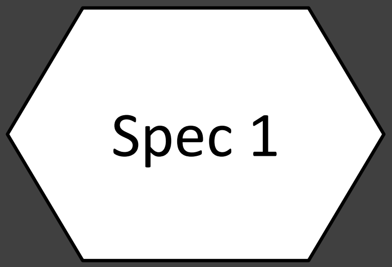
Speculative Execution in a Distributed File System Edmund B. Nightingale, Peter M. Chen, and Jason Flinn

Data Structures

Made for Speculator

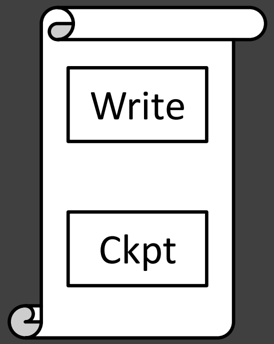
Speculation

- Tracks all objects dependent on a speculation.



Undo Log

- Tracks all modifications to an object.
- Each object that relates to a speculation gets one.
- Where checkpoints are stored.

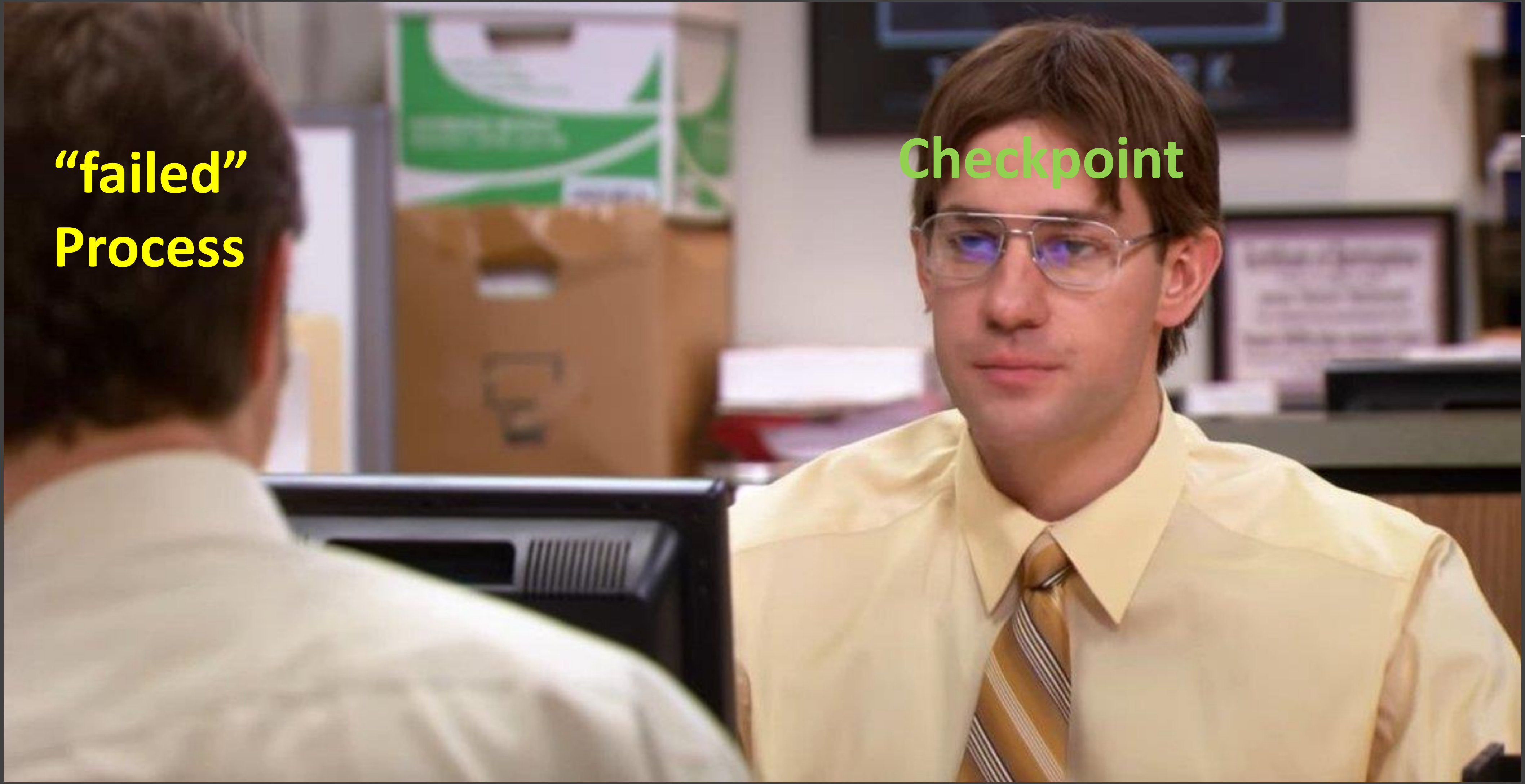


Checkpointing

- Fork of the current process stored in undo log.
- Additional state information (open files, signals, etc.).
- No more than 500ms apart.

Rollback

- When speculation is incorrect.
- Current process “failed”, ***replaced*** by checkpoint, all modifications gone.
  - Replaces process id, thread group id, files descriptors, signals, ...
  - … Identity theft?


“failed” Process
Checkpoint

Interface

- *create_speculation()*
- *commit_speculation()*
- *fail_speculation()*

Speculator

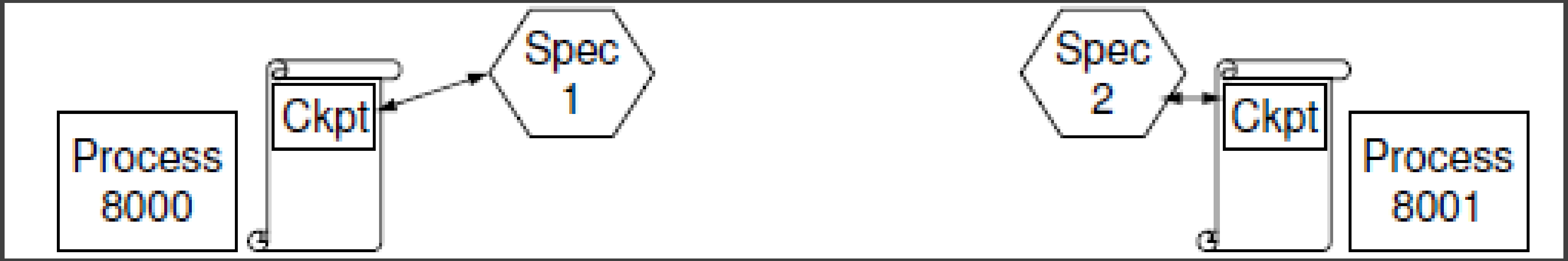
- Speculative (asynchronous) execution.
- Lightweight check-pointing and rollback.
- Propagating causal dependencies between processes.
  - Multi-process speculation.

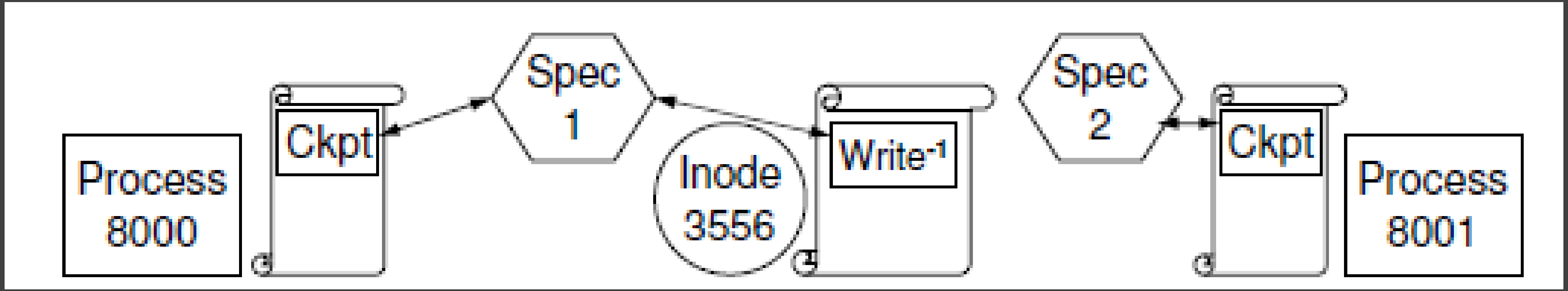
Speculative Execution in a Distributed File System Edmund B. Nightingale, Peter M. Chen, and Jason Flinn

When to Propagate?

When P modifies X (or same file as X) and P depends on speculations that X does not, propagate P's dependencies to X.

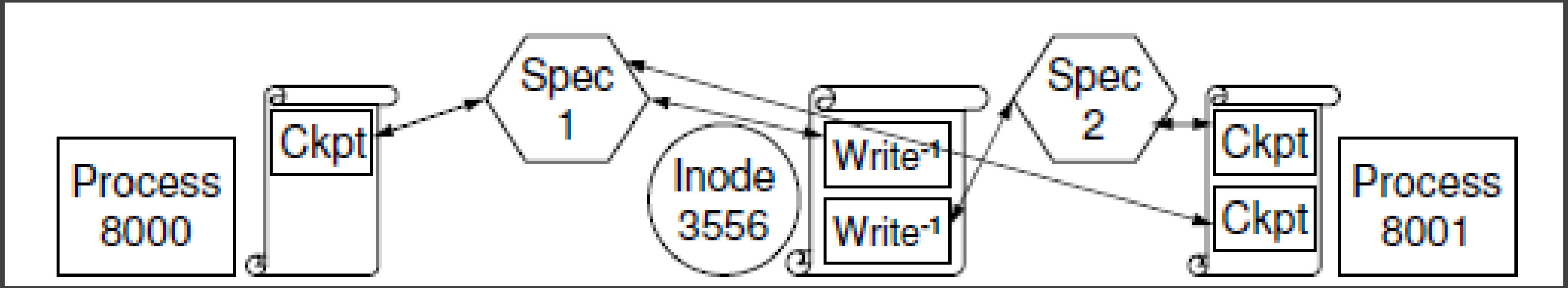
Propagating Causal Dependencies

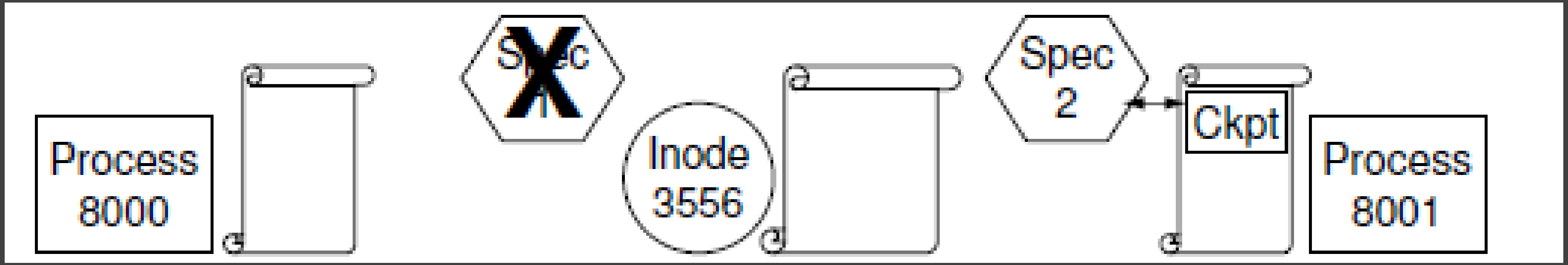




Speculative Execution in a Distributed File System Edmund B. Nightingale, Peter M. Chen, and Jason Flinn

Propagating Causal Dependencies





Speculative Execution in a Distributed File System Edmund B. Nightingale, Peter M. Chen, and Jason Flinn

More Speculator Design/Implementation

- Speculative processes can never output anything (screen, network, …).
- Never write Speculative state to Disk.
- Groups commits (increase throughput).
  - When network buffer is empty/100 ops processed → group commit.
- Handling mutating operations by trusting Server to know true state of file system.
- Managing shared meta-data structs: superblock, allocation bit-map, …
  - Shadow buffer.

What was Built

SpecNFS

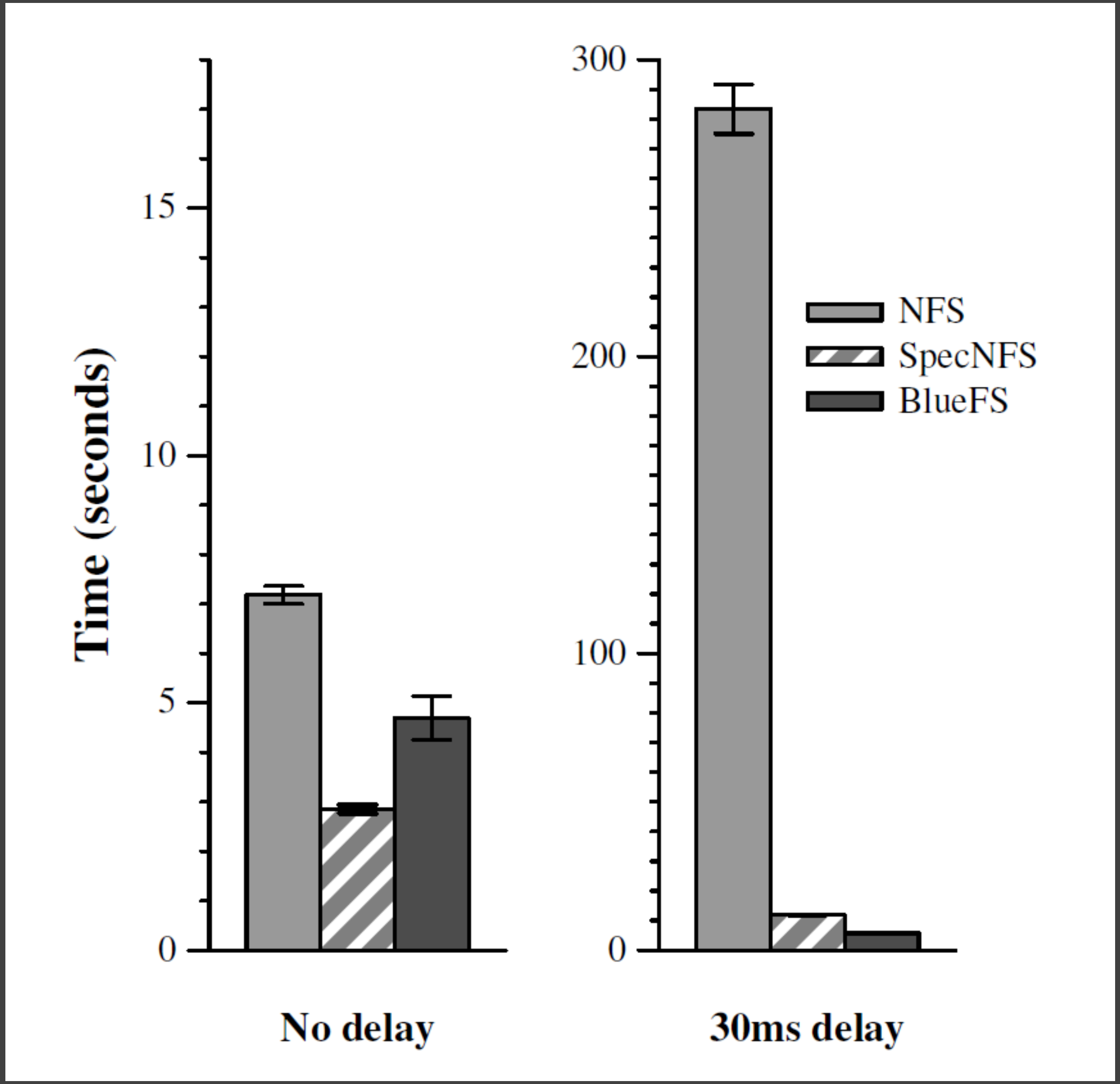
- NFS with Speculator.
- Non-deterministic RPC field values.
  - Aliased until resolved.
- Employs group commits.

BlueFS

- Single-copy consistency.
- Write-ahead log used for server to commit changes before reply (safety).
- Employs group commits.

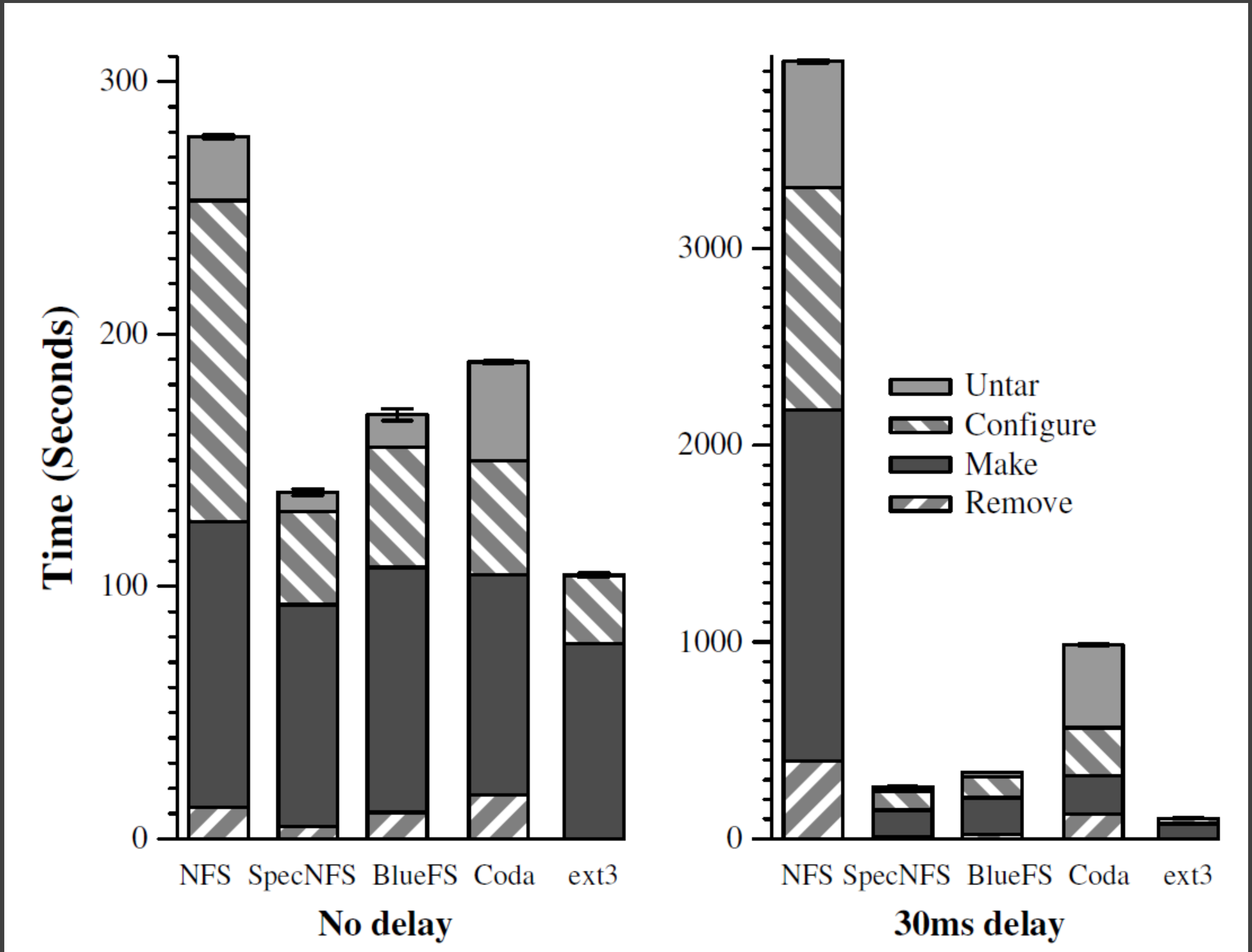
Testing Setup

- 2 Dell machines, one client, one server
  - RedHat Linux.
  - Routes packets through NISTnet emulator to introduce delay.
    - 0ms & 15ms
  - All connected through 100 Mb/s ethernet.



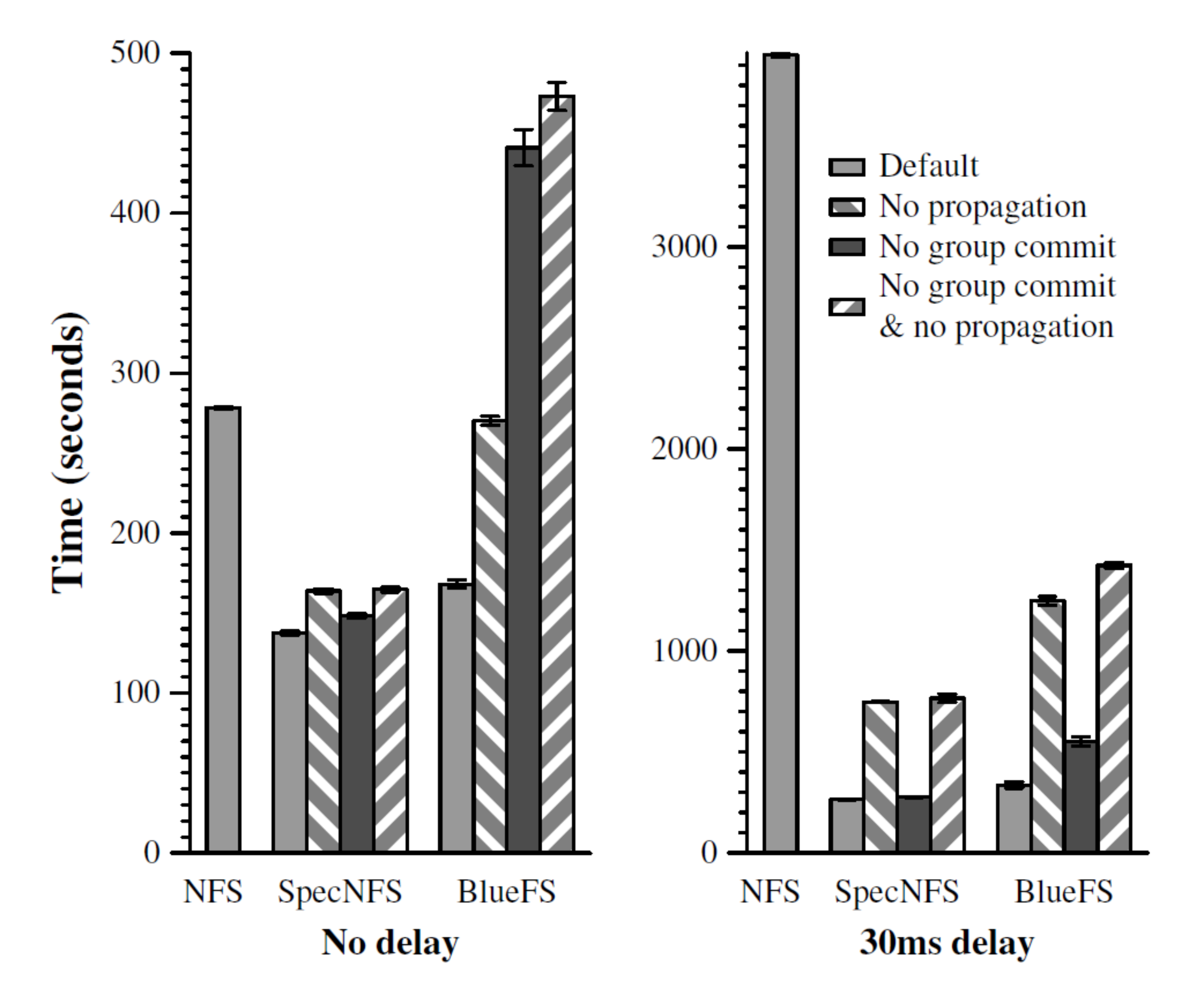
PostMark

Speculative Execution in a Distributed File System
Edmund B. Nightingale, Peter M. Chen, and Jason Flinn



Apache

Speculative Execution in a Distributed File System Edmund B. Nightingale, Peter M. Chen, and Jason Flinn

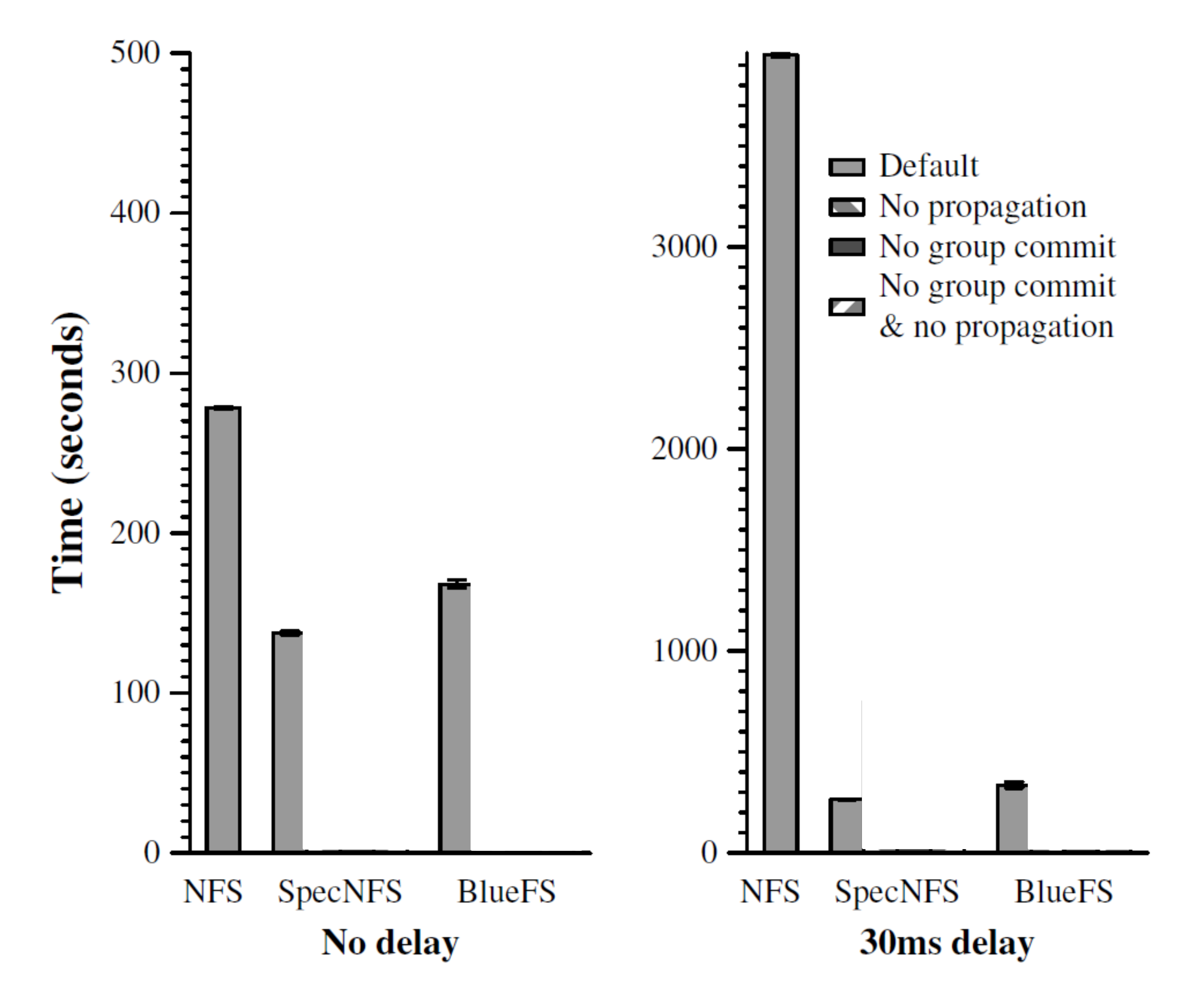


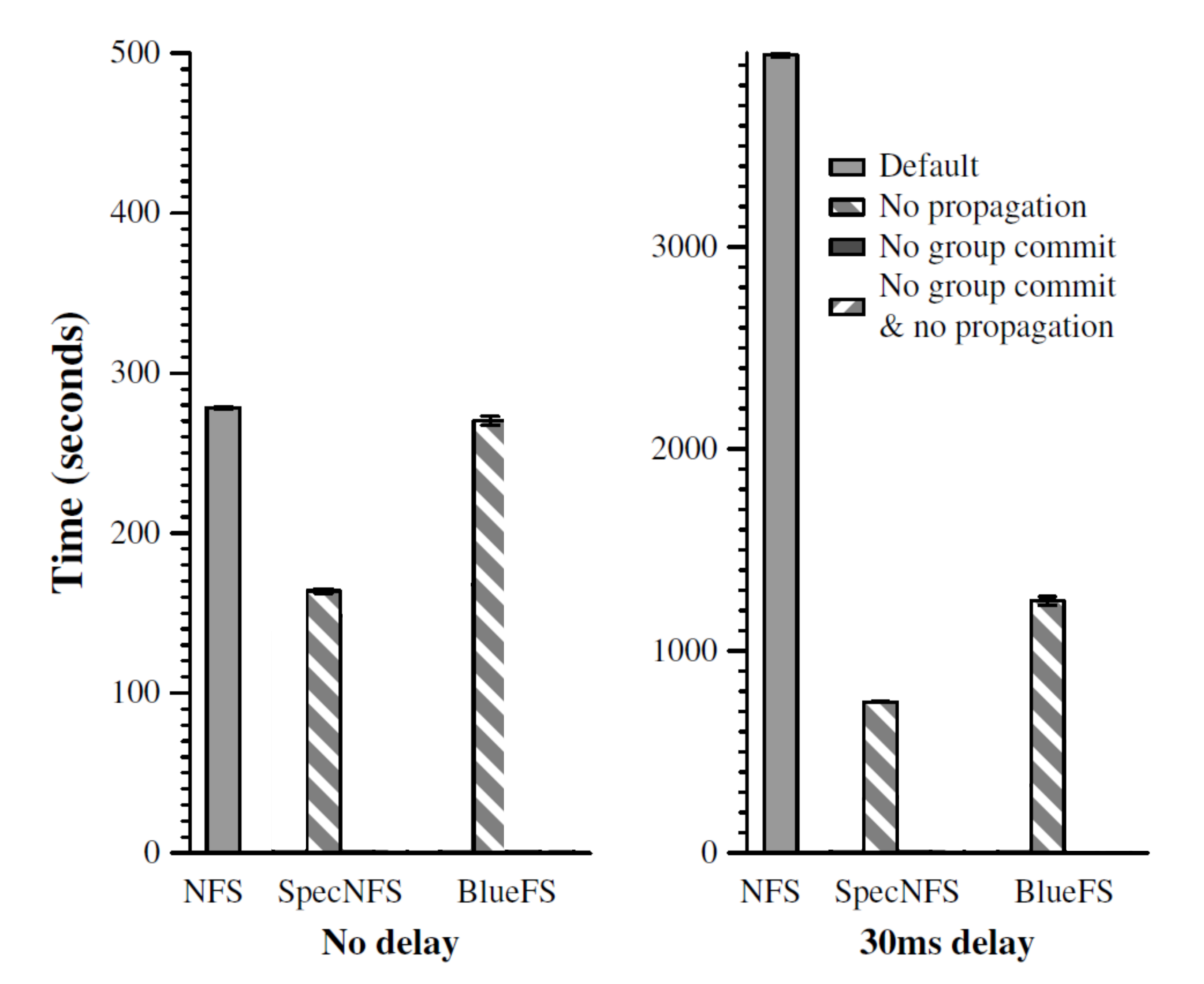
Group Commits and Propagation effect

Speculative Execution in a Distributed File System Edmund B. Nightingale, Peter M. Chen, and Jason Flinn

Group Commits and Propagation effect

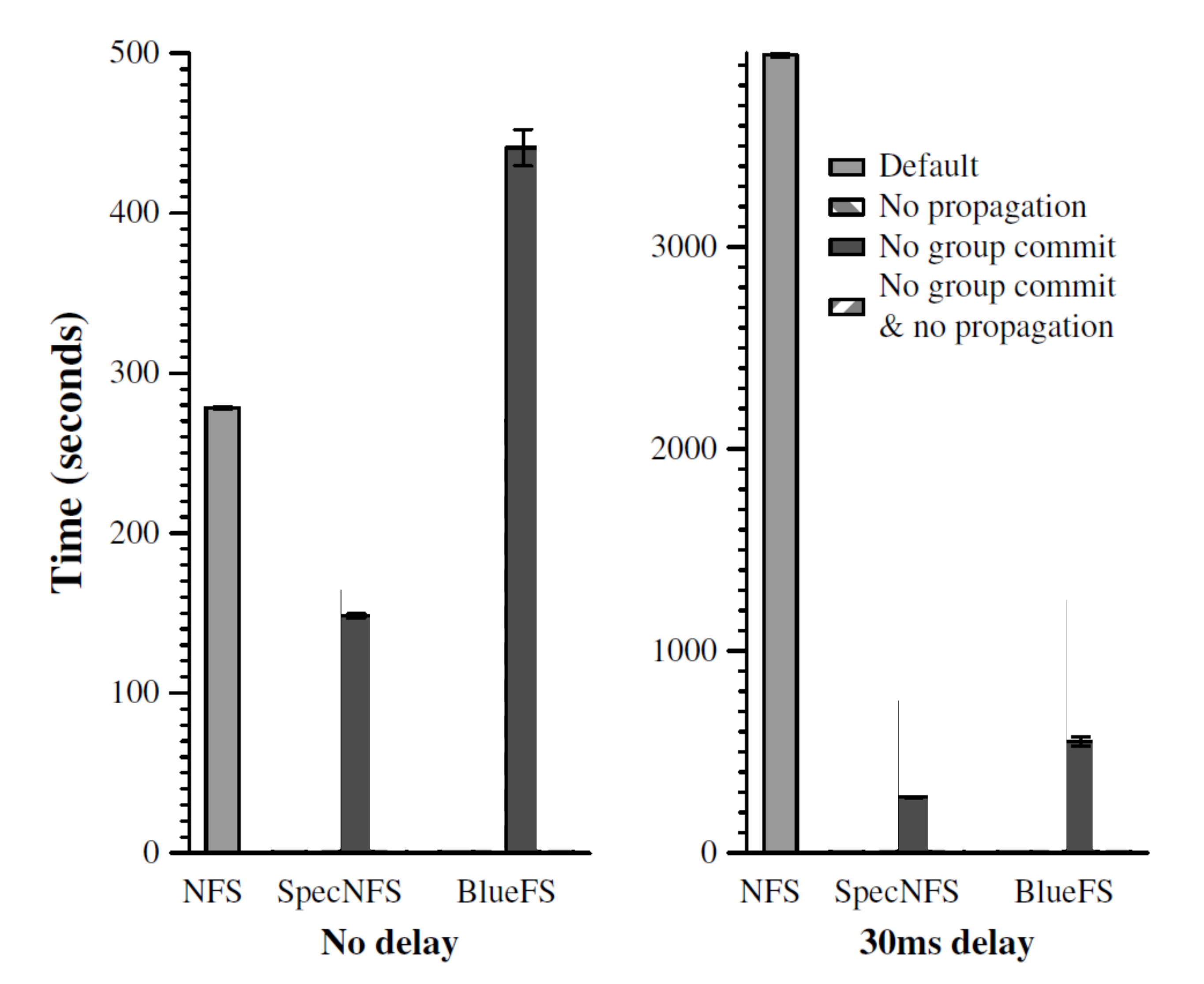
Speculative Execution in a Distributed File System Edmund B. Nightingale, Peter M. Chen, and Jason Flinn





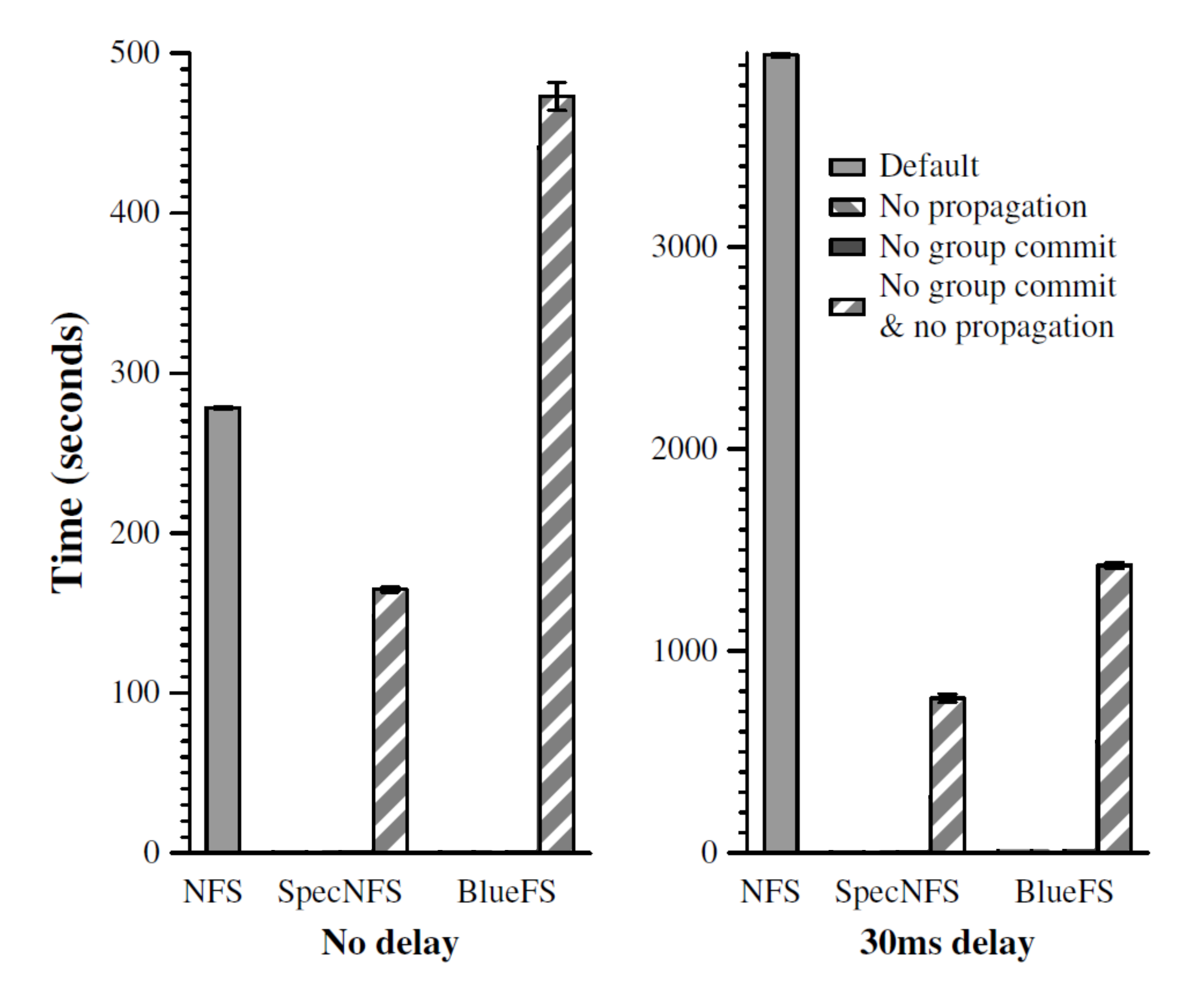
Group Commits and Propagation effect

Speculative Execution in a Distributed File System Edmund B. Nightingale, Peter M. Chen, and Jason Flinn



Group Commits and Propagation effect

Speculative Execution in a Distributed File System Edmund B. Nightingale, Peter M. Chen, and Jason Flinn



Group Commits and Propagation effect

Speculative Execution in a Distributed File System Edmund B. Nightingale, Peter M. Chen, and Jason Flinn

Discussion

- What other Apps can benefit from a Speculator?
  - Self-driving cars, text editor, …
- What security risks does Speculator introduce?
  - Malicious speculative actions could down your system?
- Uses today?
  - Computer architecture (branch prediction, pre-fetching, …)
  - Hadoop uses for slow tasks.