EECS 491
Introduction to Distributed Systems

Fall 2019

Harsha V. Madhyastha
Primary Backup Replication

View service

(1,S1,S2)
(2,S1,S3)
(3,S3,S4)

Client

S1

(1,S1,S2)
(2,S1,S3)

S2

(1,S1,S2)
RSM with Primary Backup

Server1

Application

Operating System

Virtual Machine Monitor

Hardware

Server2

Application

Operating System

Virtual Machine Monitor

Hardware
Example: Bank Server

```c
last_month uint
Deposit(user, amount) {
    balance[user] += amount
    if (curr_month > last_month) {
        add 1% monthly interest
        last_month = curr_month
    }
    return balance[user]
}
```
Example Execution

Primary
- Receive deposit request
- Update balance
- Read current time T1
- Add interest
- Return new balance

Backup
- Receive deposit request
- Update balance
- Read current time T2
- Return new balance

VMM can capture all sources of non-determinism
Log-based VM replication

- VMM at primary logs external inputs and causes of non-determinism

- Examples:
  - Results of non-deterministic instructions
    » e.g., timestamp counter read (RDTSC)
  - Disk reads
  - Incoming network packets
  - Keyboard and mouse events
  - What else?
Example: Bank Server

Deposit(user, amount)  {  // thread 1
    balance[user] += amount
    return balance[user]
}

AddInterest()  {  // thread 2
    while(currtime != 12am)  {
        sleep(1 hour)
    }
    add 0.1% interest for all users
}
Problematic Execution

**Primary**
- Receive deposit request
- Add deposit to balance
- Timer interrupt fires
- Wake up AddInterest
- Read current time T1
- Add interest
- Go to sleep
- Return new balance

**Backup**
- Receive deposit request
- Timer interrupt fires
- Wake up AddInterest
- Read current time T1
- Add interest
- Go to sleep
- Add deposit to balance
- Return new balance
Logging Interrupts

Primary
- Receive deposit request
- Add deposit to balance
- Timer interrupt fires
- Wake up AddInterest
- Read current time T1
- Add interest
- Go to sleep
- Return new balance

Backup
- Receive deposit request
- Add deposit to balance
- Fire timer interrupt
- Wake up AddInterest
- Read current time T1
- Add interest
- Go to sleep
- Return new balance

Can primary emit output without waiting for backup to execute request?
Handling Outputs

Primary
- Receive deposit request
- Add deposit to balance
- Timer interrupt fires
- Wake up AddInterest
- Read current time T1
- Add interest
- Go to sleep
- Return new balance
- *** Failure ***

Backup
- Receive deposit request
- Add deposit to balance
- Fire timer interrupt
- Wake up AddInterest
- Read current time T1
- Add interest
- Go to sleep
- *** Becomes primary ***
- Timer interrupt fires …
- Return new balance
Logging Interrupts and Outputs

Primary
- Receive deposit request
- Add deposit to balance
- Timer interrupt fires
- Wake up AddInterest
- Read current time T1
- Add interest
- Go to sleep
- Return new balance
- *** Failure ***

Backup
- Receive deposit request
- Add deposit to balance
- Fire timer interrupt
- Wake up AddInterest
- Read current time T1
- Add interest
- Go to sleep
- *** Becomes primary ***
- Return new balance
- Timer interrupt fires …
Log-based VM replication

- VMM at primary logs
  - External inputs and results of non-determinism instructions
  - Interrupts
  - Outputs

- VMM at backup replays log entries
  - Stops backup VM at input events and non-deterministic instructions
  - Delivers same inputs as primary
  - Fires interrupts based on log, not hardware
  - Suppress outputs
Project 2

- Primary-backup based key-value store

- Part 1: View service
  - Not fault tolerant

- Part 2: Primary-backup replication
  - Support GET, PUT, and Append operations

- Due October 10th

- Do questions for tomorrow’s discussion
Replicating Bank Database

- One copy in SF (primary), one in NY (backup)

“Deposit $100”

$1,000

“Pay 1% interest”

$1,000
“Deposit $100”

“Pay 1% interest”

$1,000

$1,110

$1,111
Ordering of Updates

- All updates must be applied in the same order at all replicas

- External view: Total ordering of writes

- Primary effectively serializes all writes
Serving Reads

- Can backups serve reads?
  - Assume no split brain

- What if primary’s state is ahead of backup?
  - Updates to primary not yet externally visible
  - Effect of read equivalent to if primary fails at this point

- What if backup’s state is ahead of primary?
  - Different backups may not be in sync
  - Primary may get replaced before it applies update
Reads: Primary vs. Backup

C1

“Deposit $100”

P

×

B1

B2

C2

$1100

$1000

$1000
Desired Properties

- All writes are totally ordered
- Once read returns particular value, all later reads should return that value or value of later write
- Once a write completes, all later reads should return value of that write or value of later write
Reads relative to Writes

C1

“Deposit $100”

P

“Pay 1% interest”

B

$1100

$1111

C2
Linearizability

- Total ordering of writes
- Read returns last completed write

- Single copy semantics
  - Externally visible effects of writes and reads are equivalent to if there existed a single copy

- Users oblivious to replication
Consistency Spectrum

- Consistency: What are the properties of externally visible effects?
Why weaken consistency?

- Shouldn’t we always strive for single copy semantics?
  - Comes at the expense of lower performance

- Latency vs. consistency tradeoff
Consistency Spectrum

- Eventual Consistency
- Read-after-write Consistency
- Causal Consistency
- Sequential Consistency
- Linearizability Consistency

- Consistency
- Latency
- Ease of programming
Causal Consistency

- Order of causally related writes must be preserved in values returned to reads
  - If $W_1 \rightarrow W_2$, then if a read sees effect of $W_2$, it must see effect of $W_1$

- **Example:** Facebook News Feed
  - Okay to not see all completed posts
  - But, if you see a comment, you must see the post on which the comment is made

- Main utility: Lazy sync between replicas