

HQ Replication

Analisi e implementazione di un protocollo ibrido per
accordo in un sistema distribuito

I sistemi distribuiti

- Componenti distribuite nello spazio connessi ad una rete
- Scambio di messaggi asincroni
- Assenza di memoria condivisa
- Assenza di clock condiviso
- Assenza di failure detector

Fallimento di un sistema distribuito

- crash
- crash link
- omissione
- fallimento bizantino

Fallimento bizantino

- Nodo con comportamento arbitrario
- Per dichiararli falliti bisogna trovare accordo
- Per essere resistente ai fallimenti bizantini un sistema richiede $3f+1$ nodi

Protocolli previsti in HQ

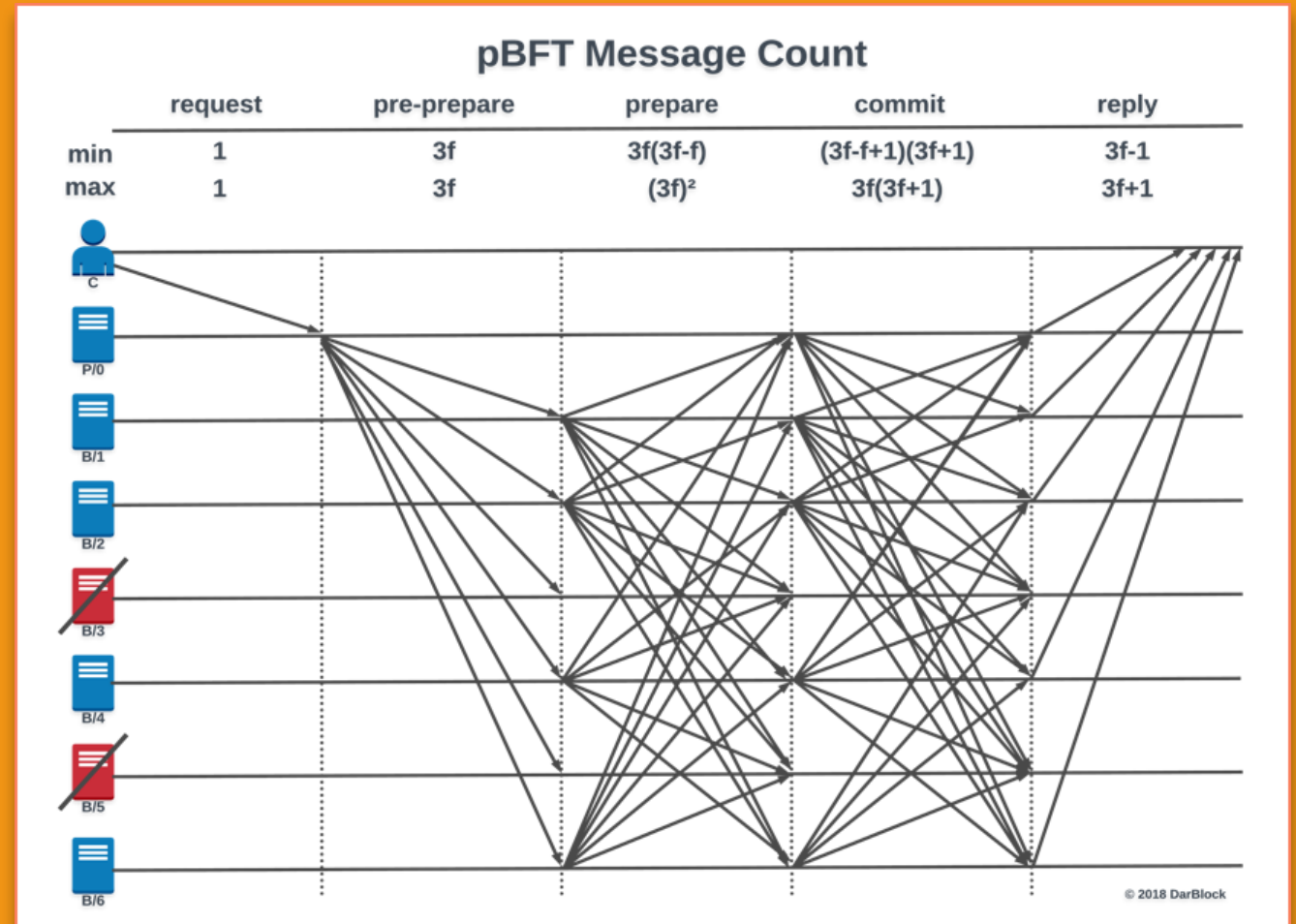
- Protocollo Query/Update
- Practical Byzantine Fault Tolerance (pBft)

Cosa prevede Query/Update

- Protocollo con quorum
- Quorum fissato a $4f+1$ dove f sono i fallimenti
- N fissato a $5f+1$
- 2 tipi di richieste: Query (lettura) e Update (scrittura) su oggetti in memoria
- Generalizzazione operazioni

pBFT

- Protocollo basato su accordo tra repliche
- Viene trovato l'accordo tra le repliche, sotto supervisione di una replica primary, in diverse fasi: pre-prepare, prepare e commit
- N fissato a $3f+1$



Pro/Contro dei protocolli

pBFT

- Pro:
 - $3f+1$ Repliche
 - Numero di fasi fisso e limitato
- Contro:
 - Lento
 - Intasa la rete

Q/U

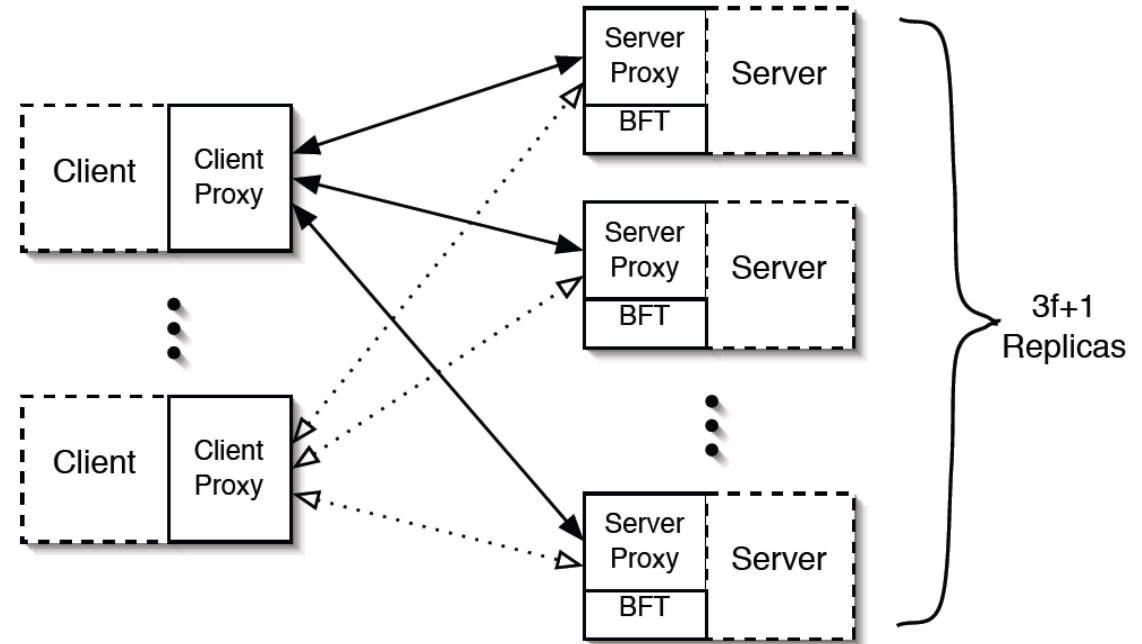
- Pro:
 - Nel caso ottimale è veloce e ha un basso carico sulle repliche
- Contro:
 - $5f+1$ repliche
 - Degrado delle performance in caso di conflitto

Protocollo HQ

- Comunicazione tra client e server richiede quorum
- 2 tipi di richieste Read (lettura) e Write (modifica) degli oggetti
- Usa metodologie di comunicazione di Q/U
- Usa metodologie di risoluzione del conflitto pBFT
- N fissato a $3f+1$ e quorum a $2f+1$
- Generalizzazione operazioni

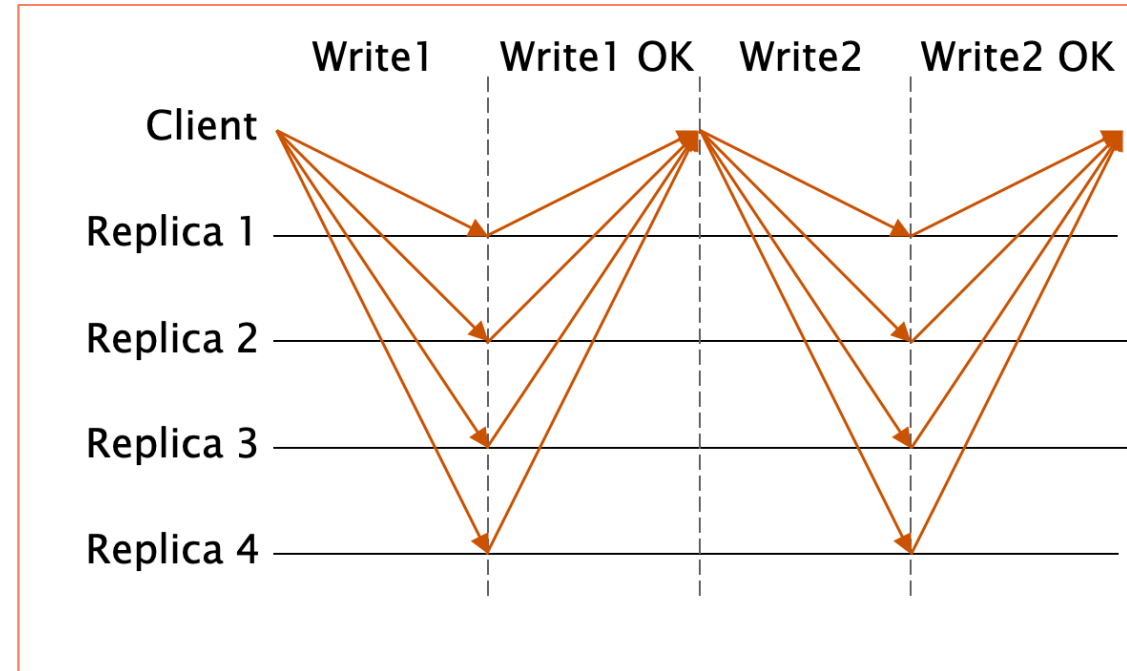
Architettura di HQ

- La comunicazione è affidata ai proxy
- BFT è una entità fisicamente presente nella replica e distinta dal proxy e dal server
- Server è l'entità che effettivamente richiama il codice che legge e scrive sugli oggetti



Comunicazione in HQ

- Lettura in una fase
- Scrittura in 2 fasi



In figura una normale scrittura

Scrittura in 2 fasi

- Fase uno:
 - Il client cerca di assicurarsi il grant sulla replica
 - La replica invia la risposta con il risultato dell'operazione
- Fase due:
 - Il client raccoglie i grant, e raggiunto il quorum di $2f+1$, compone il certificato per completare le scritture

Il Grant

Il grant è la concessione della replica e specifica determinati dati

1. Id del client che si è aggiudicato la concessione del time stamp
2. Id dell'oggetto su cui eseguire l'operazione
3. Time stamp
4. Firma della replica

Il Certificato

- Il certificato è una raccolta di $2f+1$ grant (concessioni)
- Certificano che l'operazione al time stamp t è assegnata al client c
- È uguale per tutte le repliche
- Fornisce una sorta di sincronicità sulle operazioni svolte su un determinato oggetto

Assegnazione delle concessioni

Tutte le repliche sono indipendenti una dall'altra

Le risposte dipendono dallo stato della replica in quel momento.

Possibili risposte in Fase 1

1. WRITE-1-OK se il grant non è assegnato al timestamp corrente
2. WRITE-1-REFUSED se il grant è già stato assegnato ad un altro client

Lato client si traduce in 3 diverse situazioni:

1. Richiesta accettata
2. Richiesta rifiutata
3. Individuato conflitto

Scrittura isolata

client 1

replica 1

State: **Quiescent**

Grant { Client: ?
Seq No: 0
Operation: ?

replica 2

State: **Quiescent**

Grant { Client: ?
Seq No: 0
Operation: ?

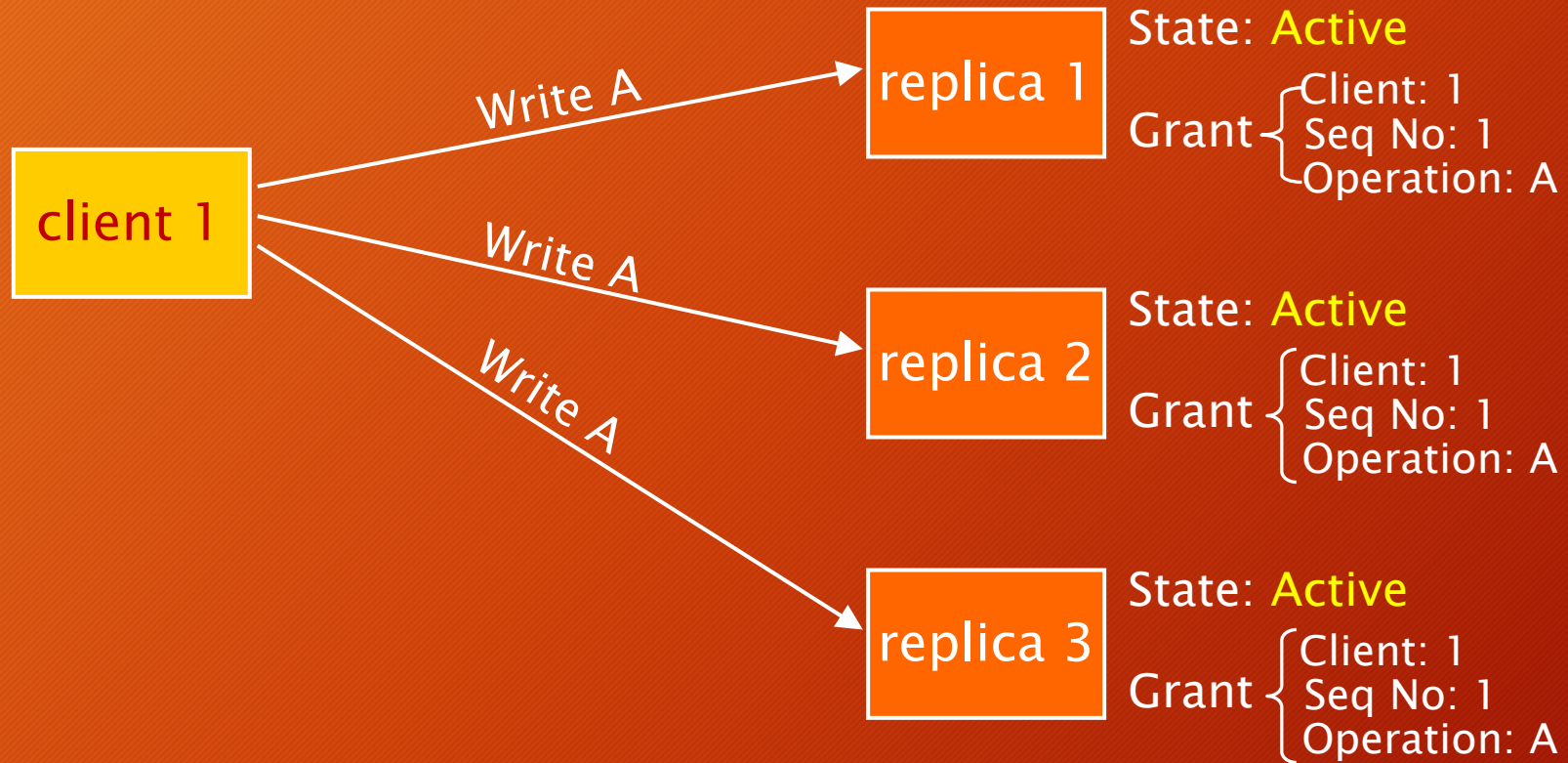
replica 3

State: **Quiescent**

Grant { Client: ?
Seq No: 0
Operation: ?

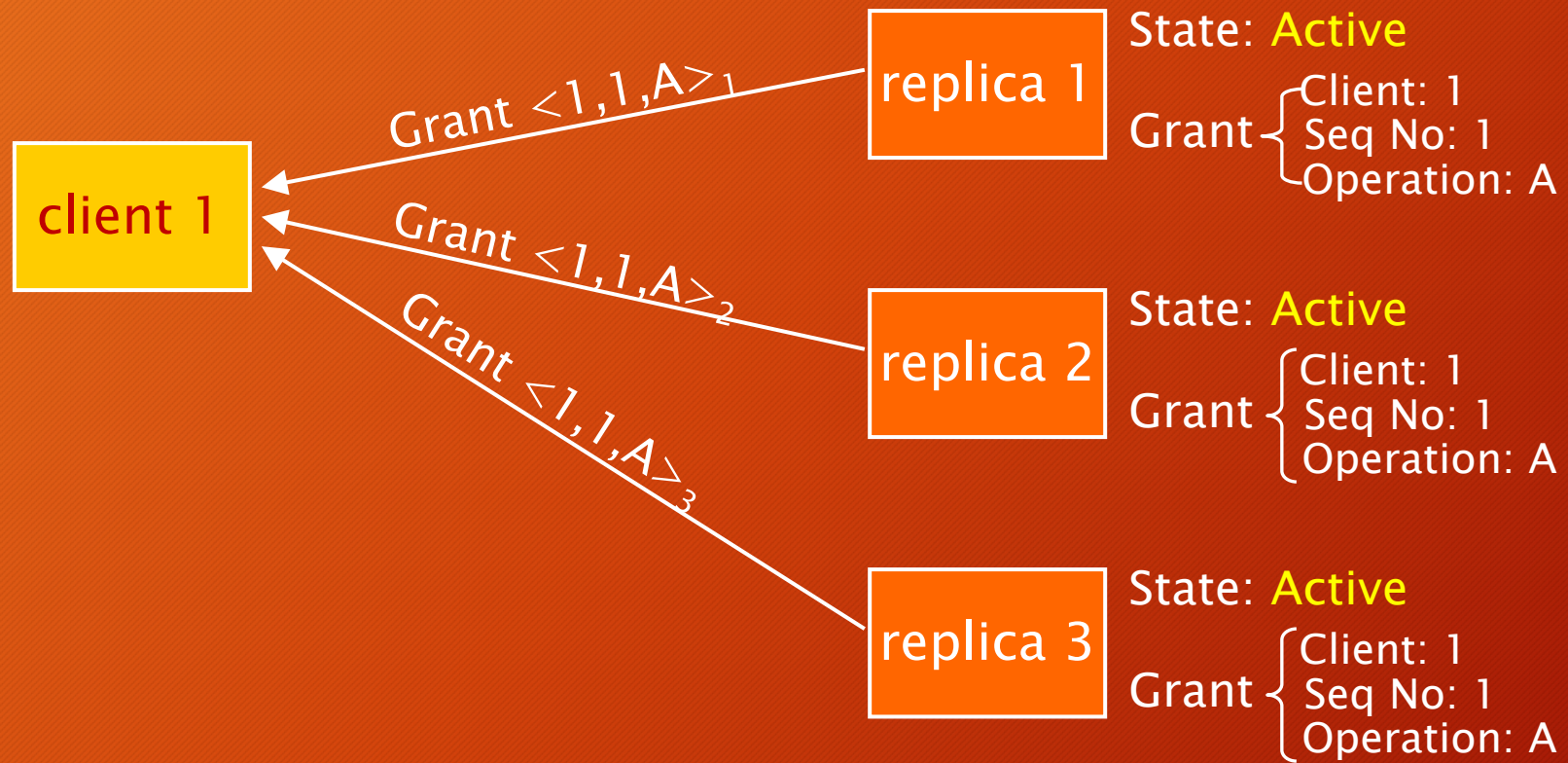
TS = Seq No = 0

Scrittura isolata



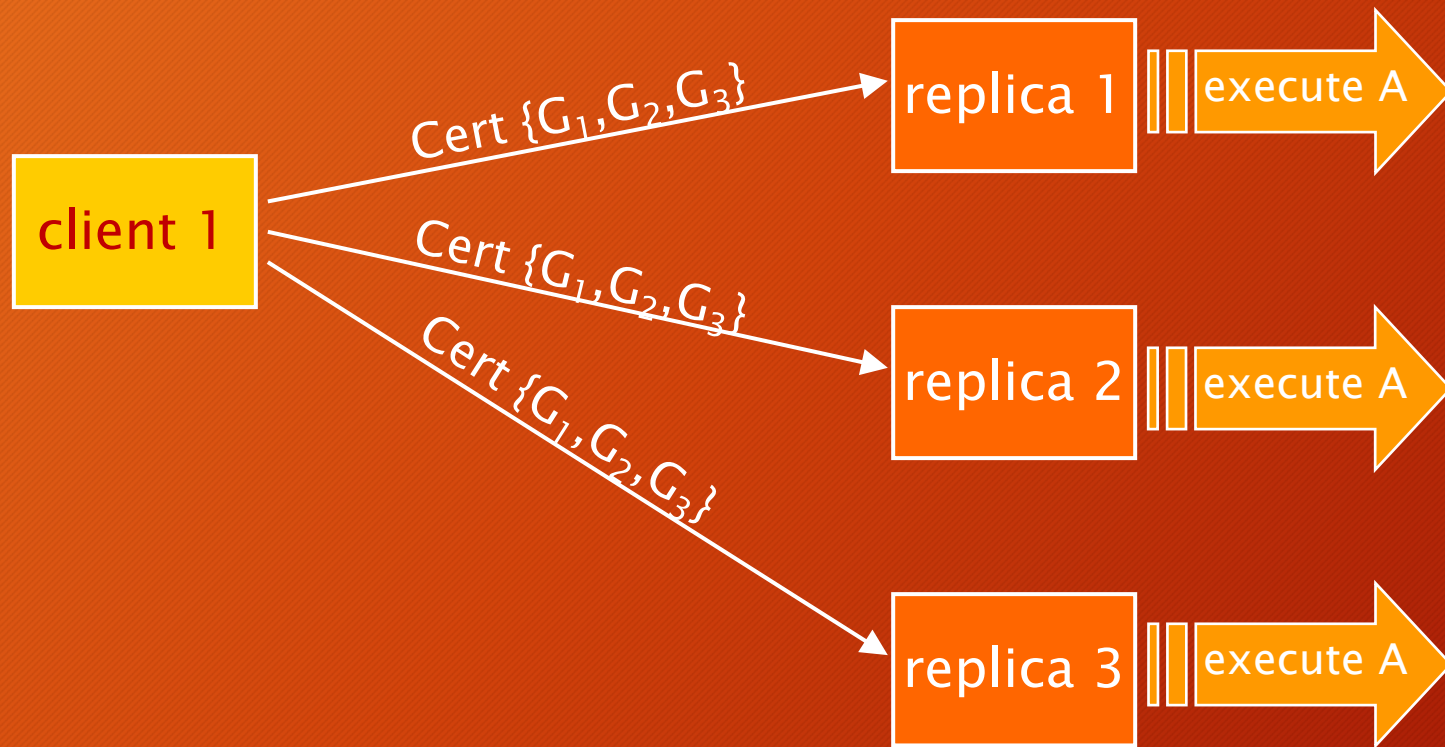
TS = Seq No = 1

Scrittura isolata



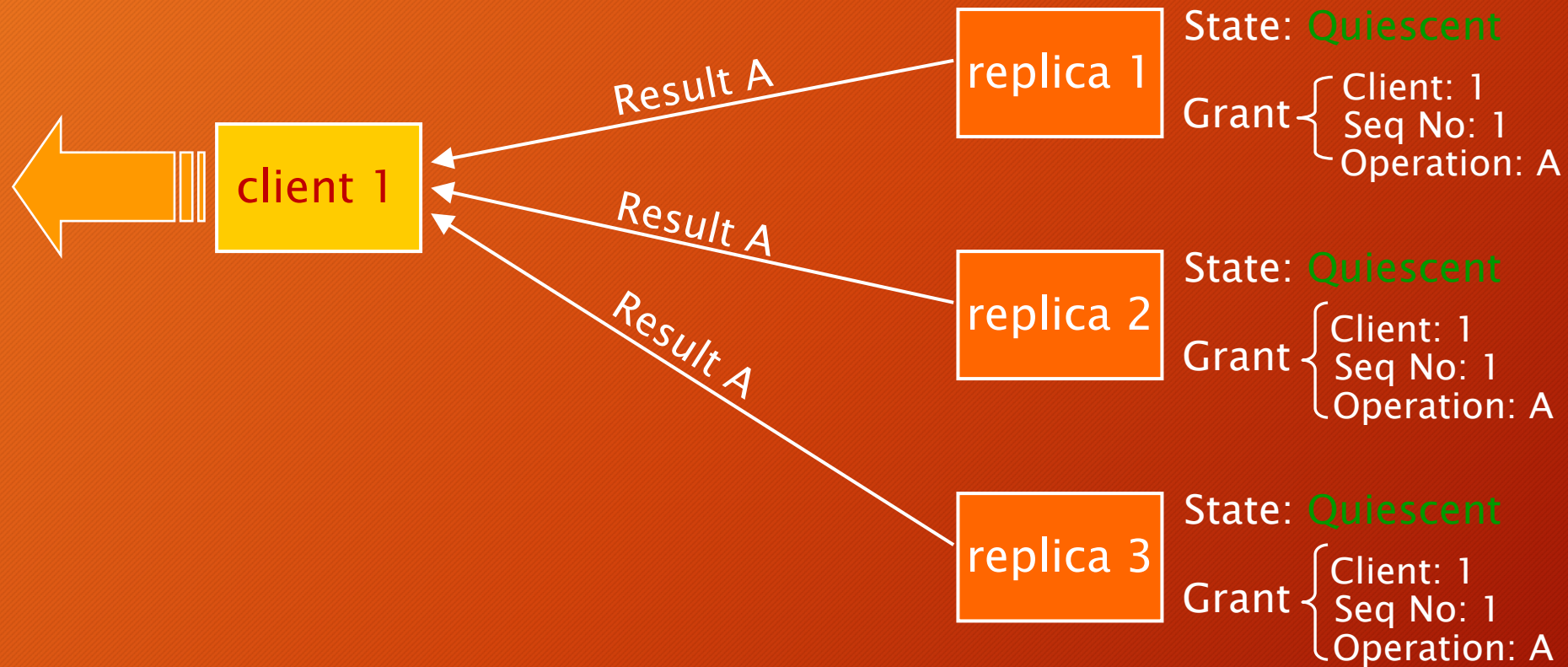
TS = Seq No = 1

Scrittura isolata



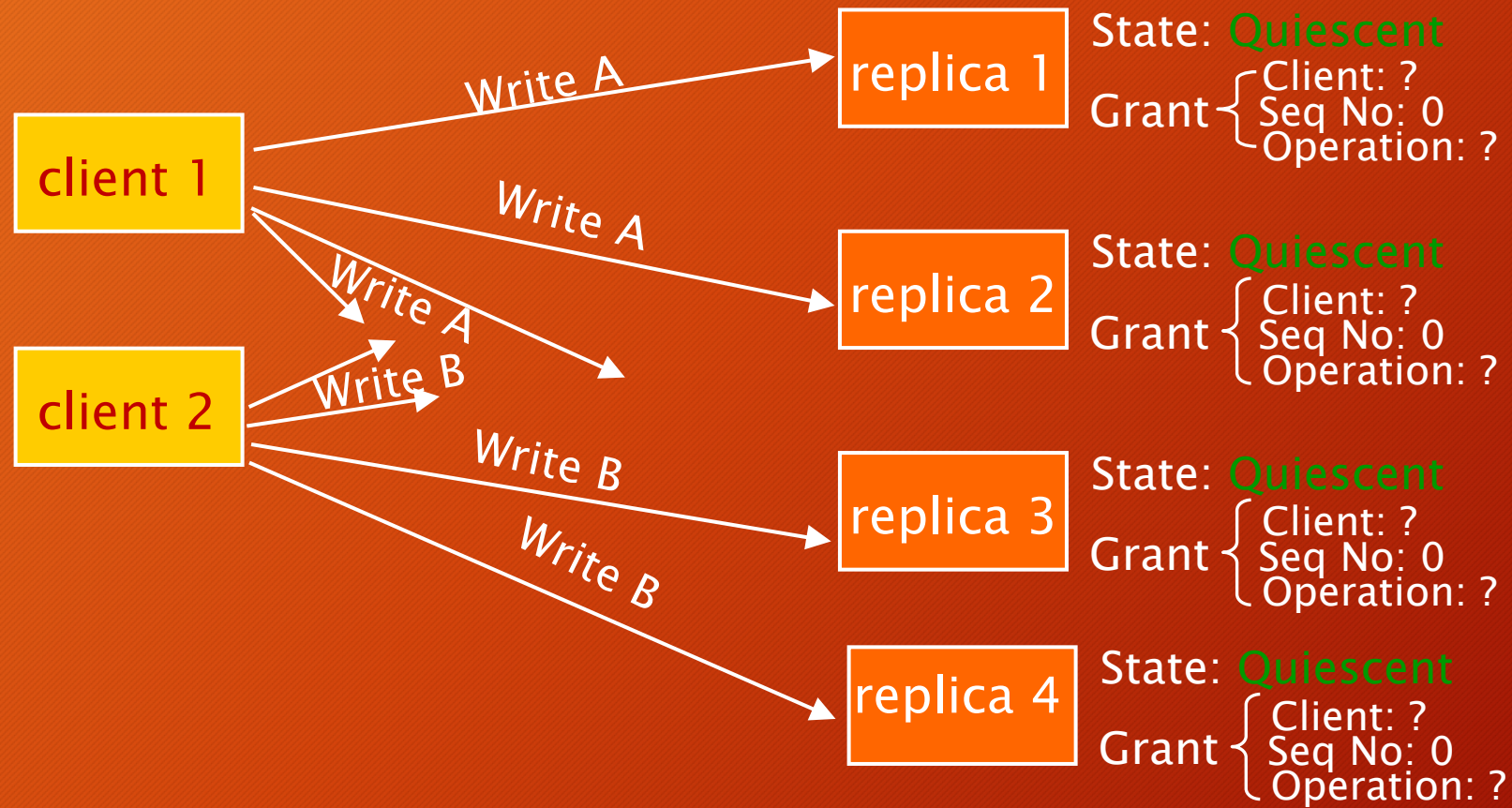
TS = Seq No = 1

Scrittura isolata



TS = Seq No = 1

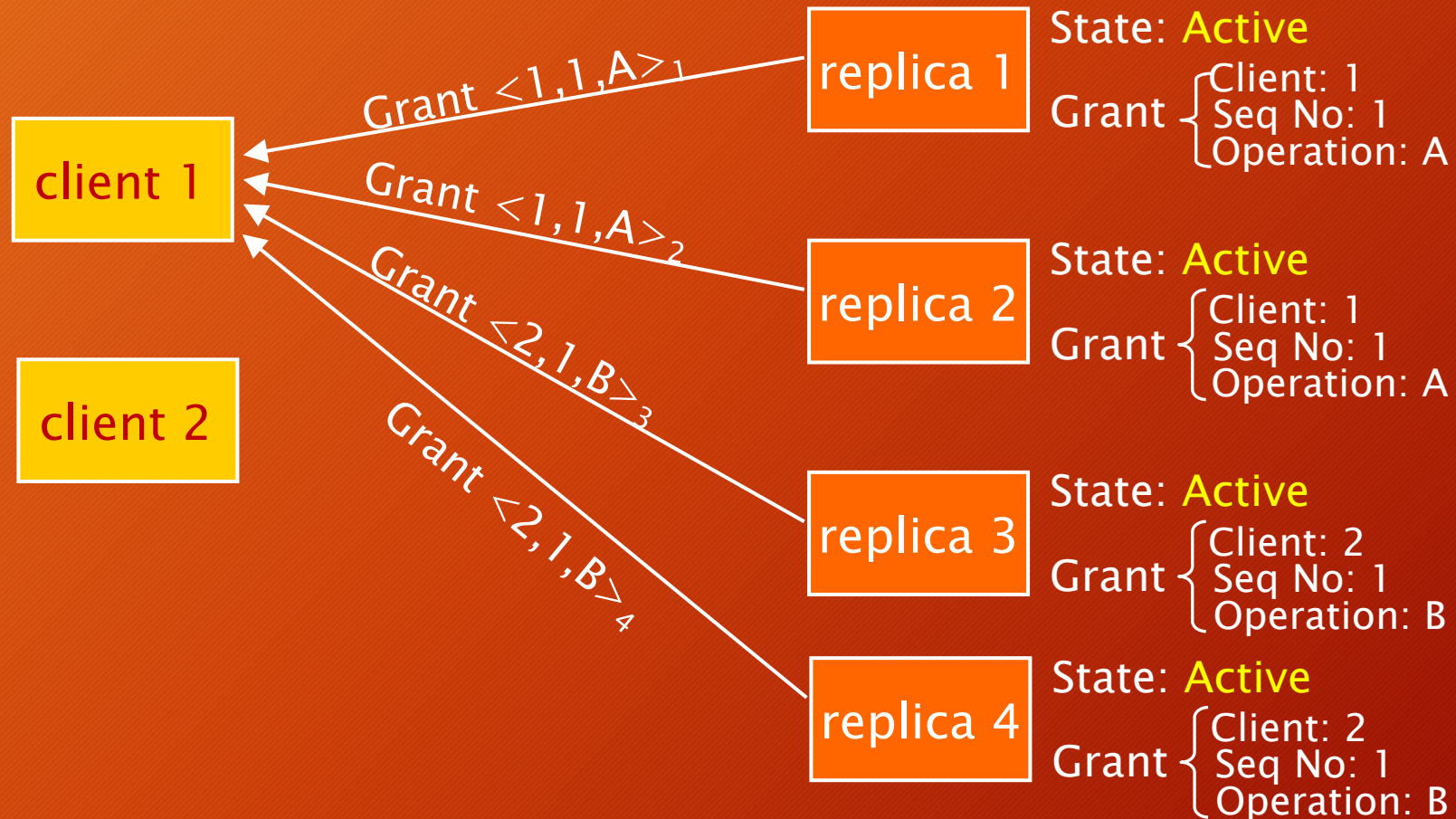
Scrittura con conflitto



TS = Seq No = 0

Scrittura con conflitto

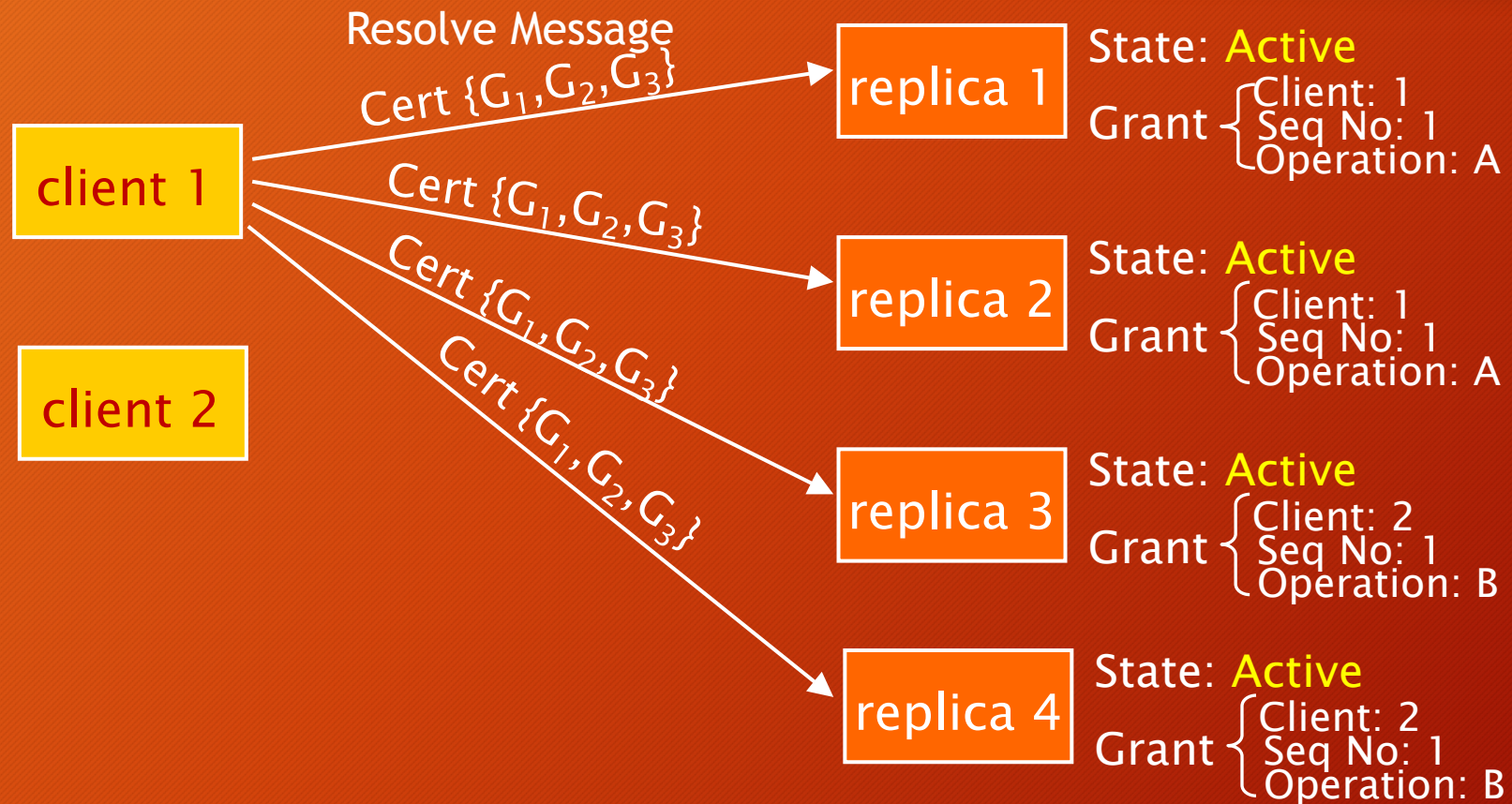
Il QUORUM non viene raggiunto



TS = Seq No = 1

Scrittura con conflitto

Viene inviato il
certificato di
conflitto



TS = Seq No = 1

Scrittura con conflitto

client 1

client 2

replica

replica

replica

replica

BFT

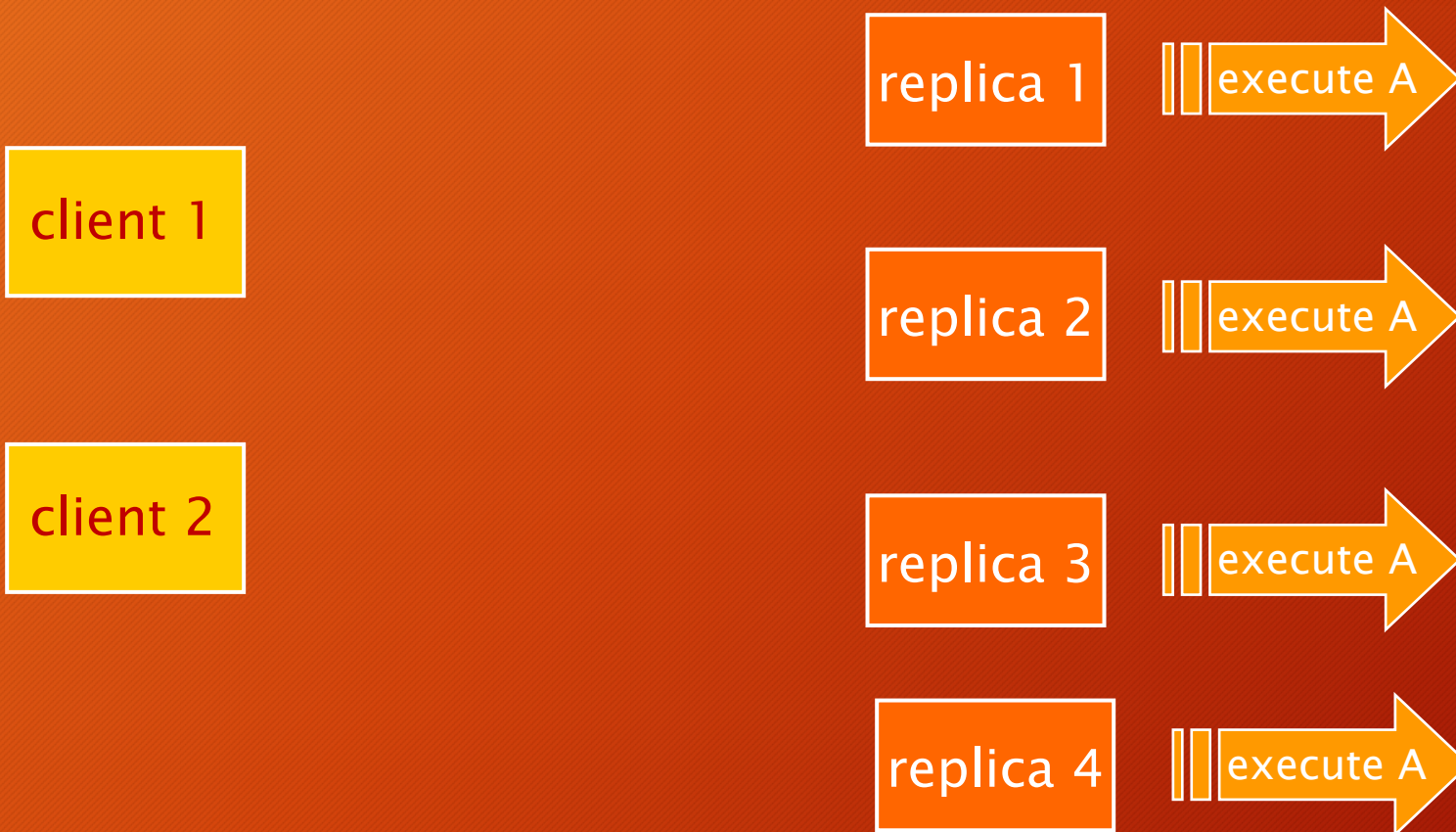
Risolve la
Contesa del TS
Fornendo la
Lista delle ops
Da eseguire

TS: B

TS: 1
Operation: B

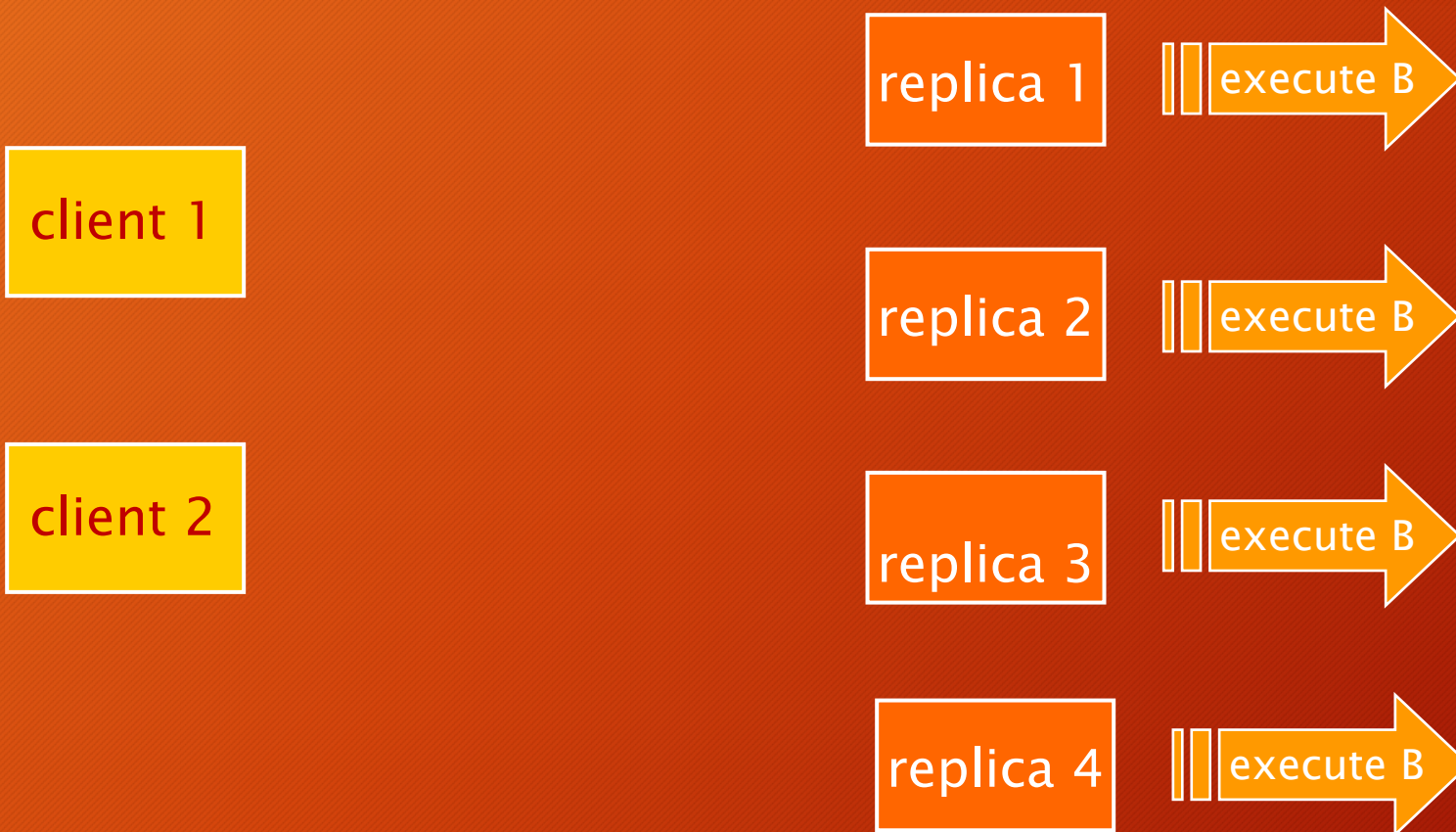
TS = Seq No = 1

Scrittura con conflitto



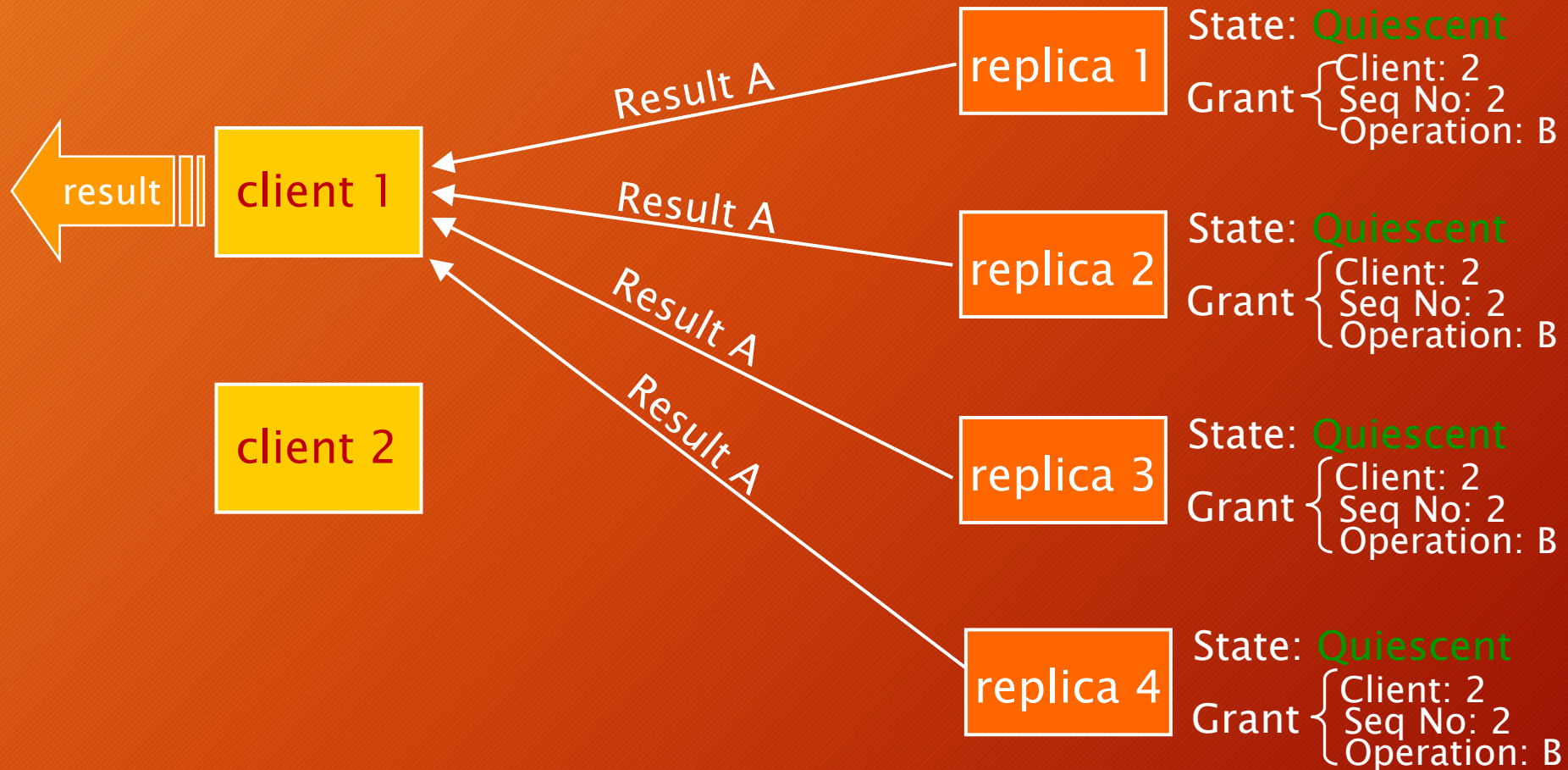
TS = Seq No = 1

Scrittura con conflitto



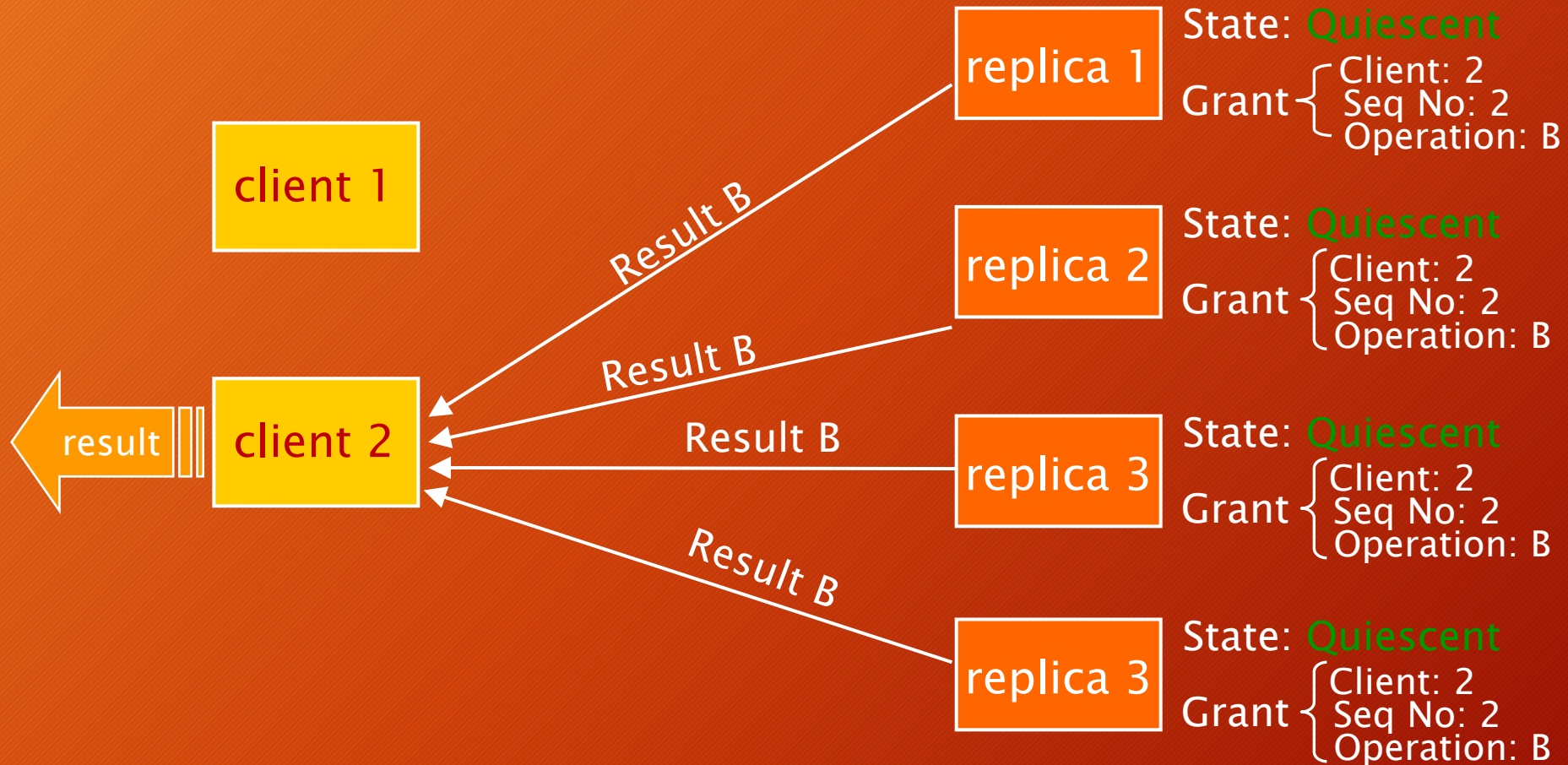
TS = Seq No = 2

Scrittura con conflitto



TS = Seq No = 2

Scrittura con conflitto



TS = Seq No = 2

Conclusioni

- Il protocollo non è particolarmente difficile da implementare
- Comportamento simile ad un data base
- Molte potenzialità e versatilità dovute soprattutto alle operazioni implementabili
- Prestazioni competitive

Sviluppi futuri

- Miglioramento metodo di distribuzione delle chiavi
- Protocollo per aggiunta repliche a run time
- Previsione di operazioni che coinvolgono più oggetti in una volta