RPC

key piece

ease programming.

<u>server. add (3,5)</u>

transparency.

<u>proxy</u>. add (3,5)

Usually <u>as library</u>

(gRPC) JsonRPC XmlRPC

<u>NFS</u>

Client   Serve·

send (str)

recv (str)

send("hello")

send("add 3 5")

while (true)
  str=recv ()
  if str == "hello"
     send ("world")
  if str. starts-- ("add")
     <u>parse (str)</u>
     - - -

RPC Inside.

Register functions. handler

[protocol file] → [compiler] → [header]

Add
Minus-
Multiply
Divide

Network Connections.

byte arrays/str

(Un) Marshalling.
    (Serialization)
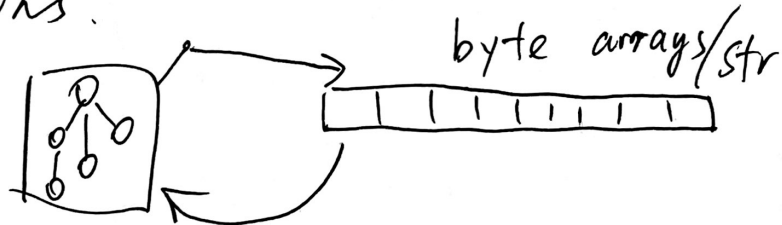
Thread — [data race]

RPC : Failures.

Client

no response.

retry . /timeout

report to App.

```
      C                    S
  ┌─────────┐         ┌────────┐
  │ Data    │ report  │  DB    │
  │ Base  ↑ │────────▶│        │
  ├─────────┤         ├────────┤
  │  rpc    │         │  rpc   │
  └─────────┘         └────────┘
```

Client   loss   server.

reorder

delay

netware...

(TCP)   discomen

At least once

```
    C              S

            exe
             O

             O

             O

             O
```
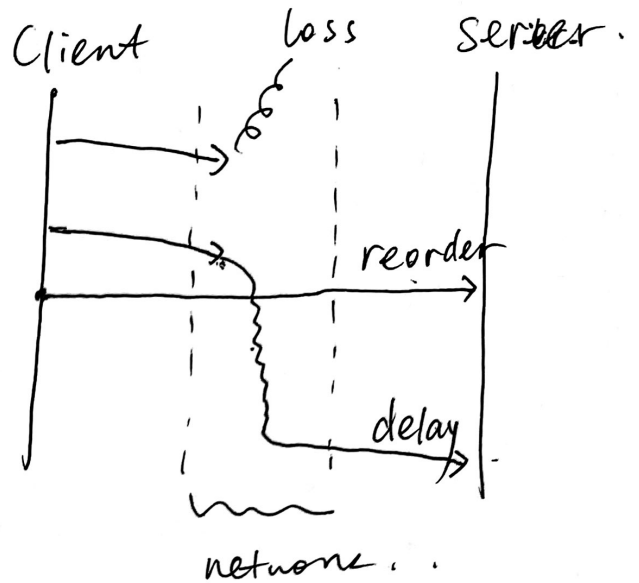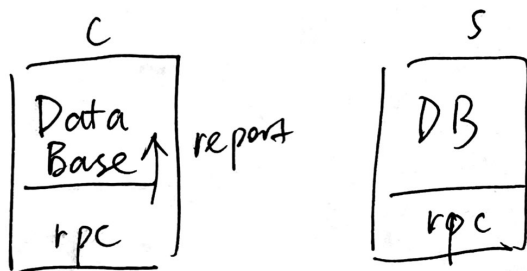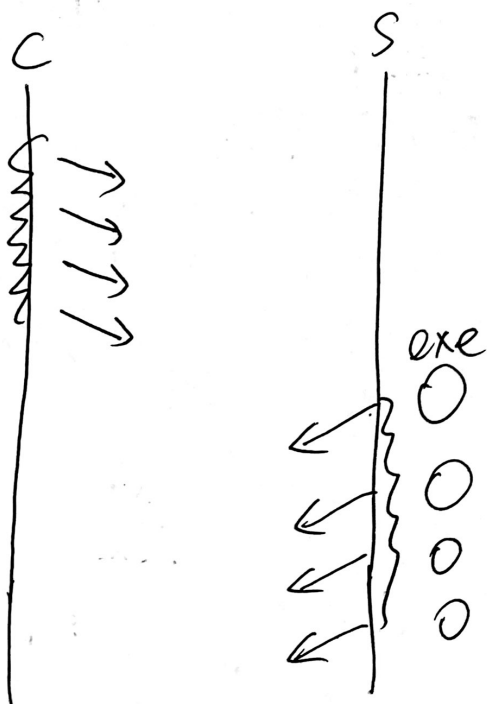
Multiple
Exe

Safe ?

buy computers

transfer money.

Safe When :

"read-only"

At most once.

$xid = cid \oplus counter$

$\underline{cid}$ 32-bit $\underline{counter}$ 32-bit

Client   Server.

$\rightarrow xid$

$r = saved\_resuts (xid)$

saved

handler if $(r == null)$

xid
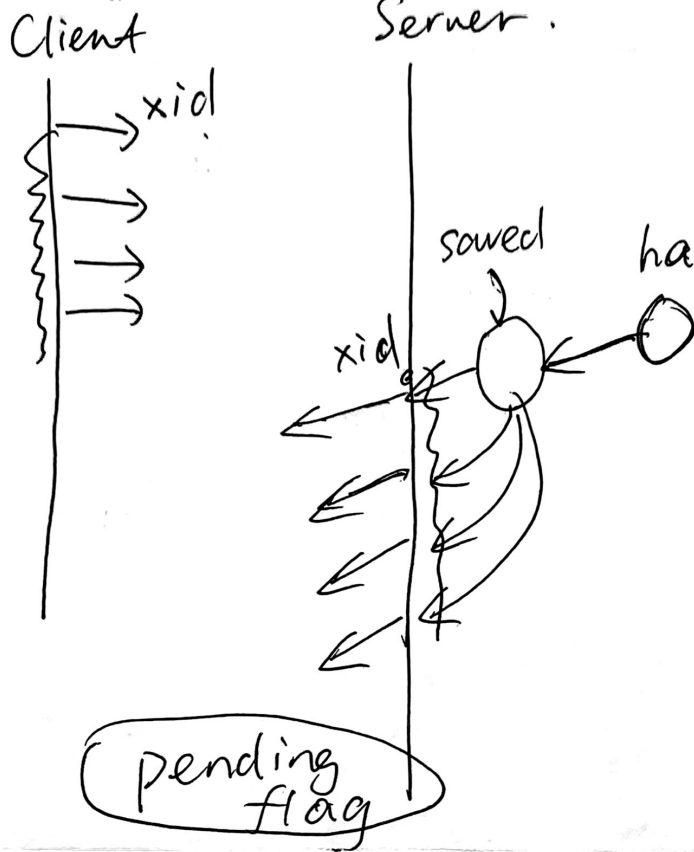
$r = handler (\quad)$

saved- resuts(xid)$\leftarrow r$

return r

(pending flag)

Problem   (Exactly once?)

saved_result

1 M ops/s.

Discard.

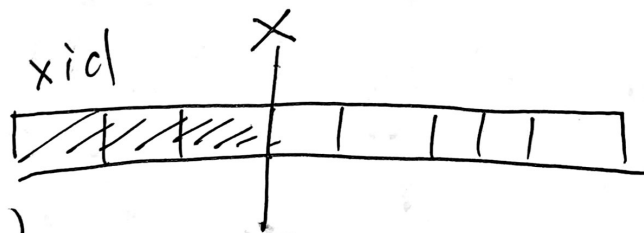xid   X

choose X

(TCP)

Failures. (Server crash)

write to disk   M.Q.

do not handle