RPC

How to find service?

How to bind?

What are some possibilities?

main()

{

add(x,y) //1 call add function

}

int add(x,y) //2 input some data, go back to function

}

return x+y;// return result

}

what we can do if the server fail during the receive part: waiting, time out, retry

for retry: we don’t know if the operation been done before the crush or not, otherwise the operation may happened twice like withdraw twice

RPC call semantics:

At lease once, at most once, exactly once

# Idempotence RPC

# RPC APIs — Services

Security: Authentication, encryption

Do pass by value, not by reference to improve the speed

Why synchronization

Ordering of events by concurrent processes

Coordinating activities amongst processes

One instruction

Clocks

Physical clocks vs logical clocks

Physical/real

Time of day

Distributed system goal – keep consistent

Clock sources

Atomic clocks, quartz crystal clocks, CMOS, etc

RTC: CMOS clock circuit driven by quartz oscillator

Impact of imperfections?

Logical clocks

Track event ordering

Synchronous networks almost never use, ex: network

Cristian’s Algorithm

Berkeley Algorithms

Master, and some low level servers who contains times

Master send request to the slaves, including itself

Delete those far wrong, and calculate the average

If we have a clock drift so far, we need to do large synchronization, since we cannot bring time back, so just to slow down the clock, eventually they will converge.