

Comunicação em Sistemas Distribuídos

- Modelo Cliente-Servidor
 - Caracterização
 - Endereçamento
 - Primitivas
 - Confiabilidade

Comunicação em Sistemas Distribuídos

Modelo Cliente/Servidor

- A construção básica em comunicação entre processos é a passagem de mensagens. Esta abstração permite um processo transmitir uma mensagem para outro.
- Ainda que protocolos da linha OSI permitam a transmissão de bits do enviante para o recebedor, existem um quantidade significativa de *headers*.
- Além disto, o modelo OSI não aborda sobre como o sistema distribuido deveria ser estruturado.

Comunicação em Sistemas Distribuídos

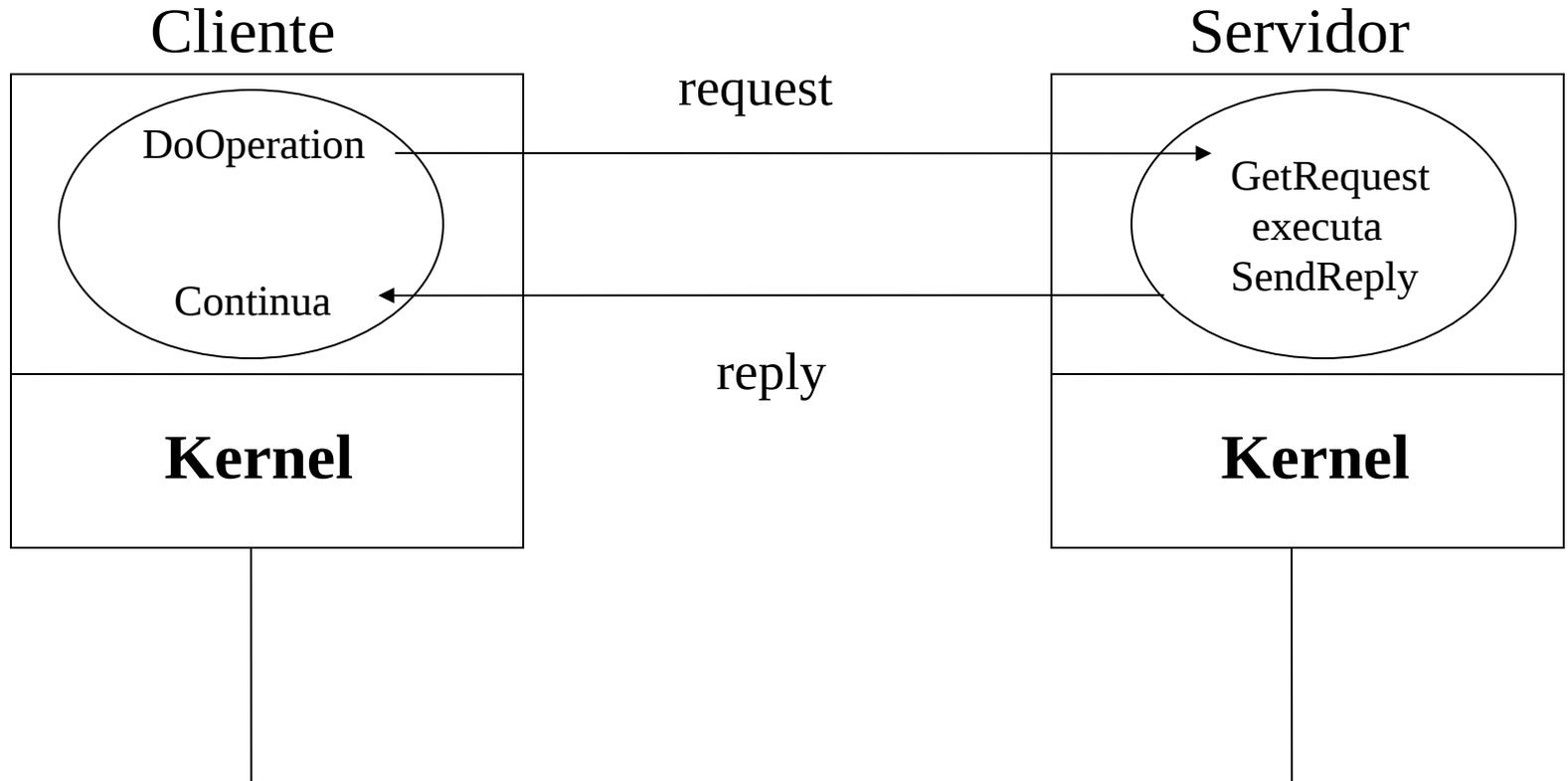
O Modelo Cliente / Servidor

Definição :

Implica um processamento cooperativo de requisições submetidas por um cliente para o servidor que as processa e retorna os resultados para o cliente. Neste modelo o processamento da aplicação é dividido entre o cliente e o servidor. O processamento é iniciado e parcialmente controlado pelo cliente e tanto cliente quanto servidor cooperam para executar com sucesso uma aplicação.

Comunicação em Sistemas Distribuídos

O Modelo Cliente / Servidor



Comunicação em Sistemas Distribuídos

O Modelo Cliente / Servidor

PROCEDURE DoOperation(serverPort:PortId; request:Mens; VAR reply:Mens); -> envia mensagem de requisição para o servidor na porta *serverPort* e recebe resposta *reply*.

PROCEDURE GetRequest(serverPort:PortId; VAR request:Mess); -> recebe uma requisição a partir da porta *serverPort*.

PROCEDURE SendReply(clientPort:PortId; reply:Mess); -> envia uma resposta para a porta *clientPort*.

Comunicação em Sistemas Distribuídos

O Modelo Cliente / Servidor



Comunicação em Sistemas Distribuídos

O Modelo Cliente / Servidor

```
#define MAX_PATH 255
#define BUF_SIZE 1024
#define FILE_SERVER      243

#define CREATE          1
#define READ            2
#define WRITE          3
#define DELETE         4

#define OK              0
#define E_BAD_OPCODE   -1
#define E_BAD_PARAM    -2
#define E_IO            -3

struct mensagem{
    long source;
    long dest;
    long opcode;
    long count;
    long offset;
    long extra1;
    long extra2;
    long result;
    char name[MAX_PATH];
    char data[BUF_SIZE];
};
```

Comunicação em Sistemas Distribuídos

O Modelo Cliente / Servidor

```
#include "fs.h"
void main(void)
{
    struct mensagem m1,m2;
    int    r;

    while (1) {
        receive(FILE_SERVER,&m1);
        switch(m1.opcode) {
            case CREATE:    r=do_create(&m1,&m2); break;
            case READ:      r=do_read(&m1,&m2); break;
            case WRITE:     r=do_write(&m1,&m2); break;
            case DELETE:    r=do_delete(&m1,&m2); break;
            default:        r=E_BAD_OPCODE;
        }
        m2.result = r;
        send(m1.source, &m2);
    }
}
```

Comunicação em Sistemas Distribuídos

O Modelo Cliente / Servidor

```
#include "fs.h"
int copy(char *src, char *dst)
{
    struct mensagem m1;
    long position;
    long client = 110;

    initialize();
    position = 0;
    do {
        m1.opcode = READ;
        m1.offset = position;
        m1.count = BUF_SIZE;
        strcpy(&m1.name, src);
        send(FILE_SERVER, &m1);
        receive(client, &m1);

        m1.opcode = WRITE;
        m1.offset = position;
        m1.count = m1.result;
        strcpy(&m1.name, dst);
        send(FILE_SERVER, &m1);
        receive(client, &m1);
        position += m1.result;
    } while (m1.result > 0);
    return (m1.result >= 0 ? OK : m1.result);
}
```

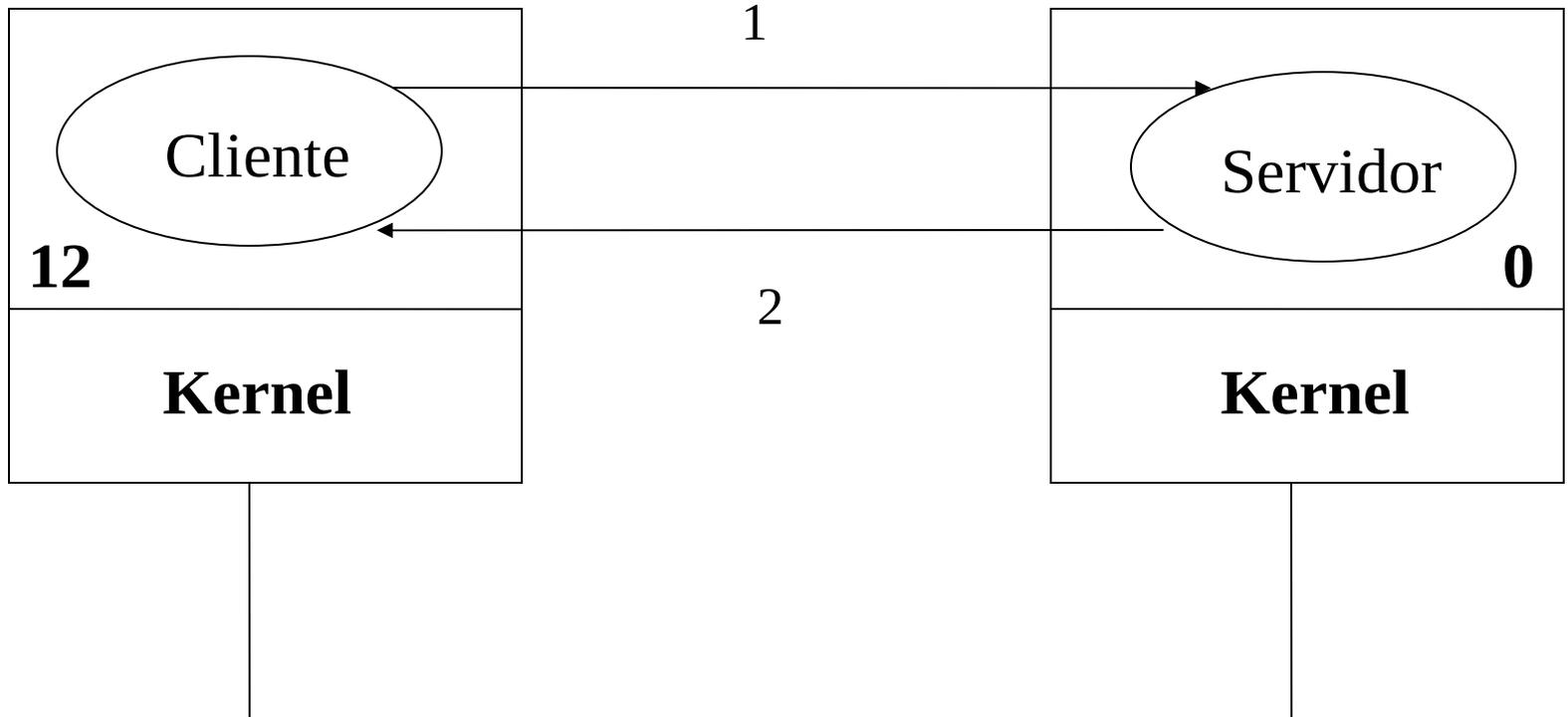
Comunicação em Sistemas Distribuídos

O Modelo Cliente / Servidor

Endereçamento – Opção 1

199

243



1: Request para 243.0

2: Reply para 199.12

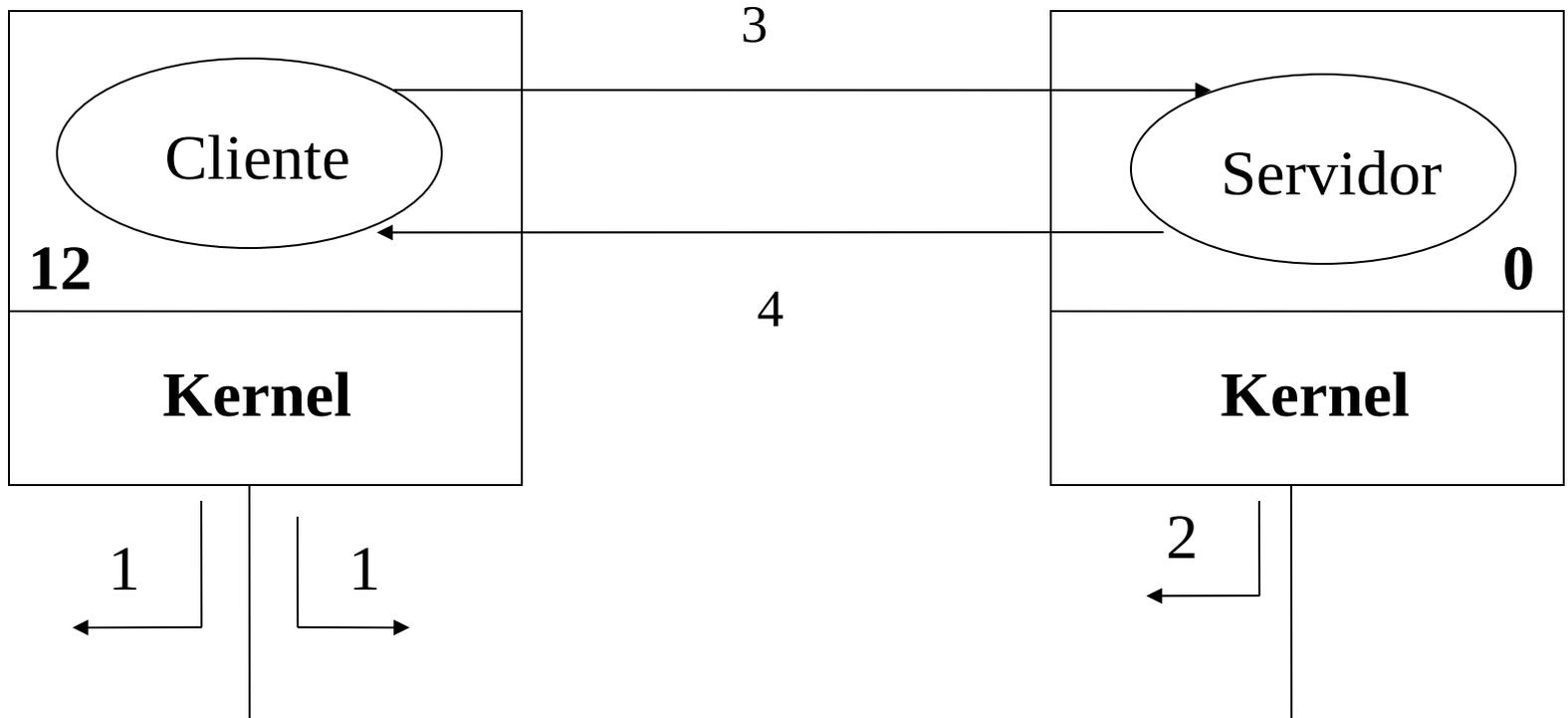
Comunicação em Sistemas Distribuídos

0 Modelo Cliente / Servidor

Endereçamento – Opção 2

199

243

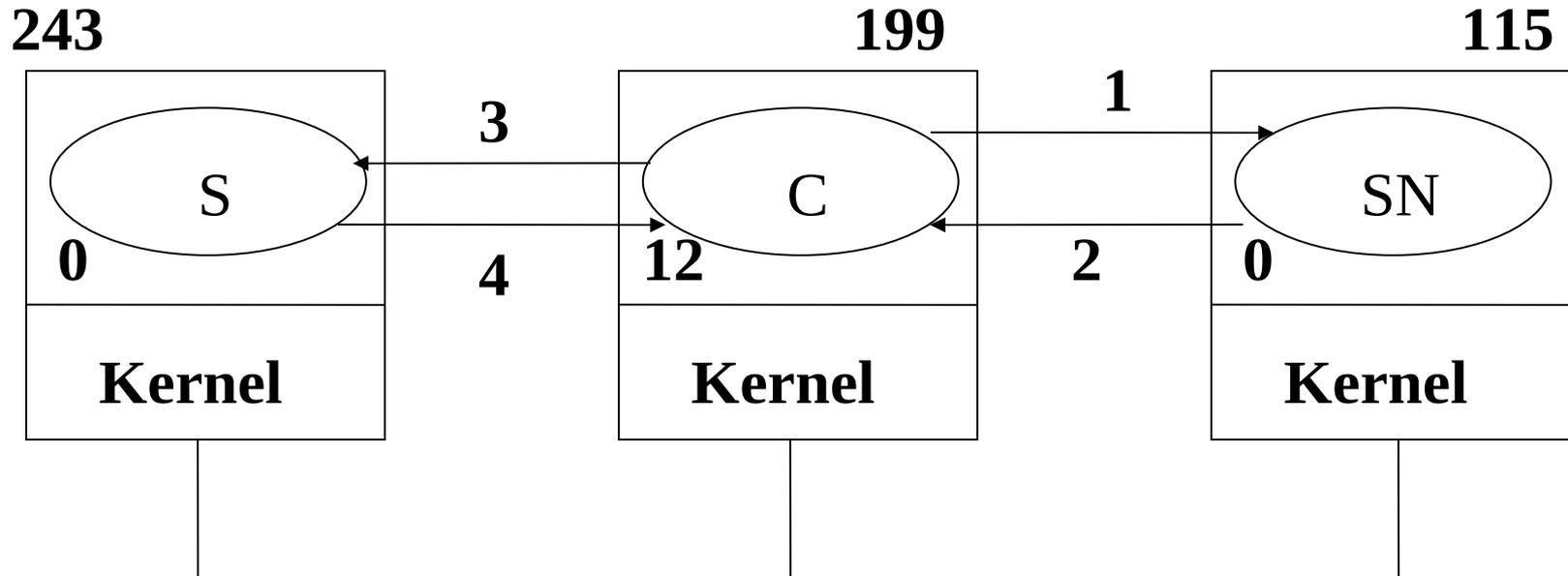


1: Broadcast “onde esta o Servidor X?” 3: Request para 243.0
2: endereço de X é 243.0 4: Reply para 199.12

Comunicação em Sistemas Distribuídos

O Modelo Cliente / Servidor

Endereçamento – Opção 3



1: Request NS “onde esta o Servidor X?” 3: Request para 243.0
2: endereço de X é 243.0 4: Reply para 199.12

Comunicação em Sistemas Distribuídos

O Modelo Cliente / Servidor

Confiabilidade das Primitivas

Prováveis causas de falha na entrega:

- Descarte de mensagens
- Particionamento da rede, alguns nodos podem ficar de fora do restante da rede
- Processos podem falhar, como diferenciar falha de processo e falha de comunicação?
- Dados corrompidos, mensagens não são recebidas corretamente
- Como uma operação do tipo DoOperation pode detectar uma falha de entrega?

Comunicação em Sistemas Distribuídos

O Modelo Cliente / Servidor

Confiabilidade das Primitivas

Abordagens para tratar com o problema das perdas:

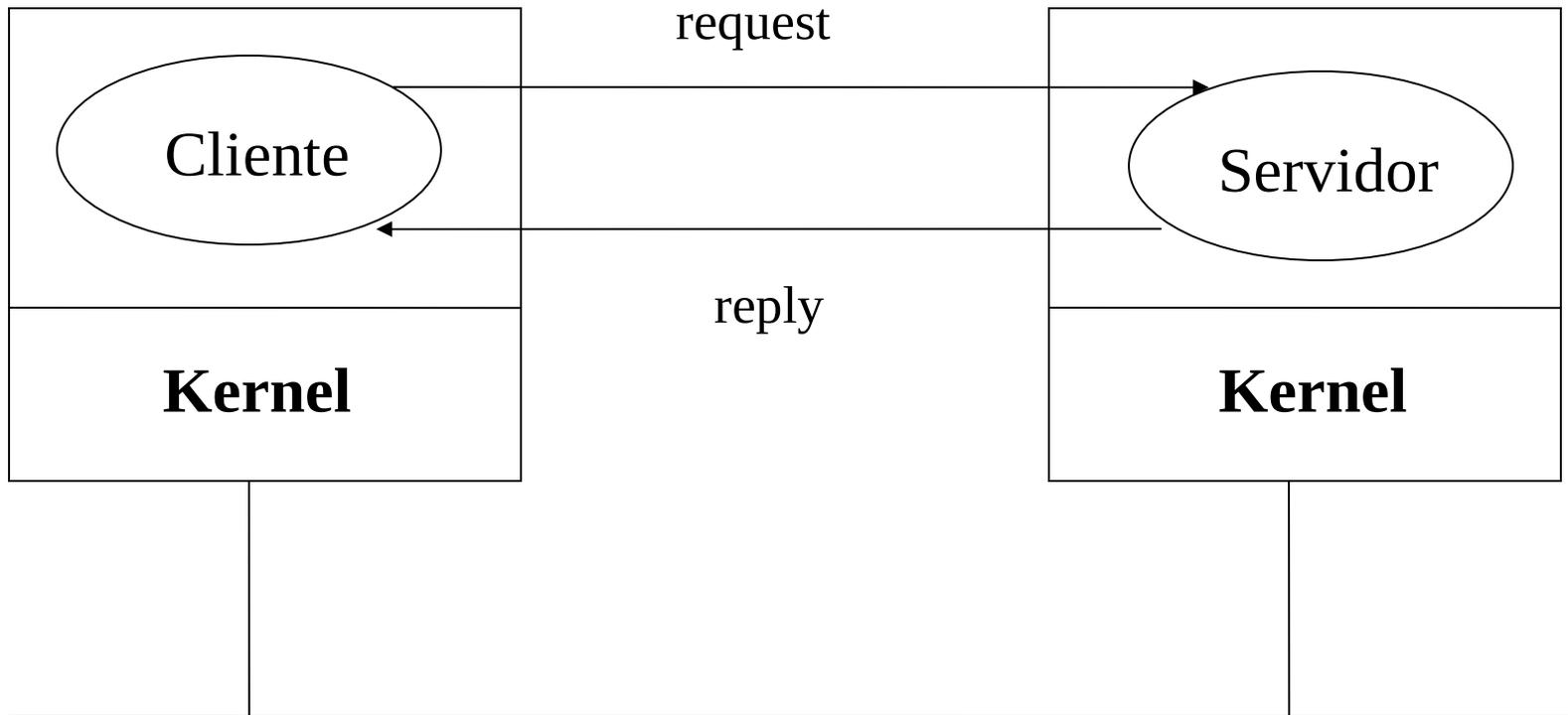
Definir a semântica do *Send* para ser não confiável, o sistema não dá garantias (correios).

Requisitar que o kernel da máquina recebedora envie um ack de volta para o enviante.

Utilizar a mensagem reply como sendo ack do request e enviar ack para o reply.

Modelo Cliente/Servidor

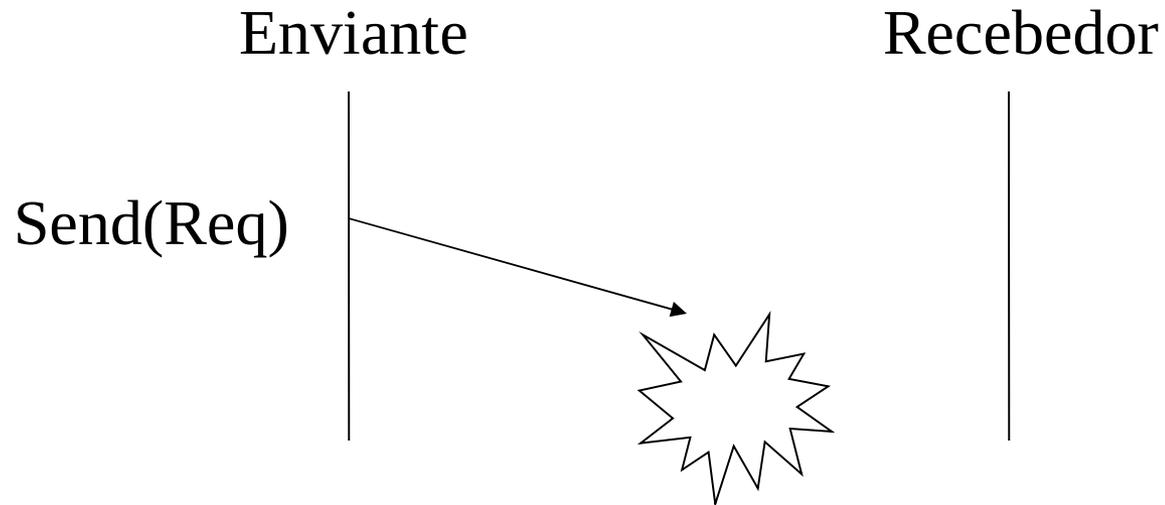
Confiabilidade das Primitivas (RR)



Request ou Reply perdidos. Recuperação tem que ser feita no nível C/S

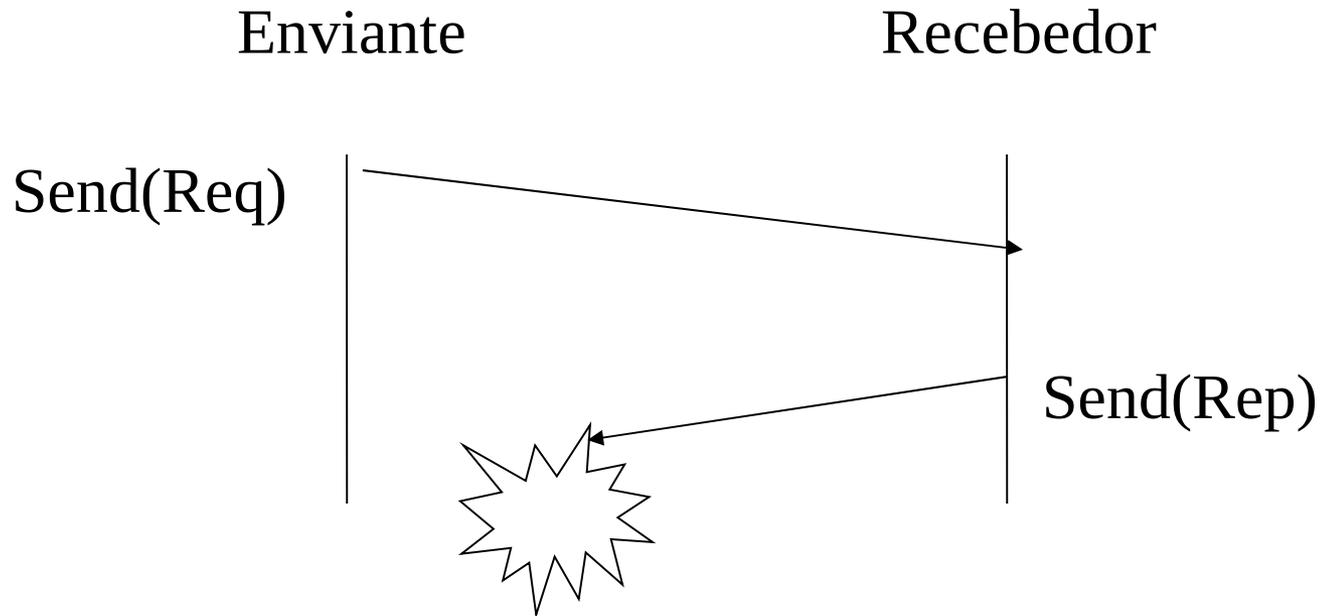
Modelo Cliente/Servidor

Confiabilidade das Primitivas



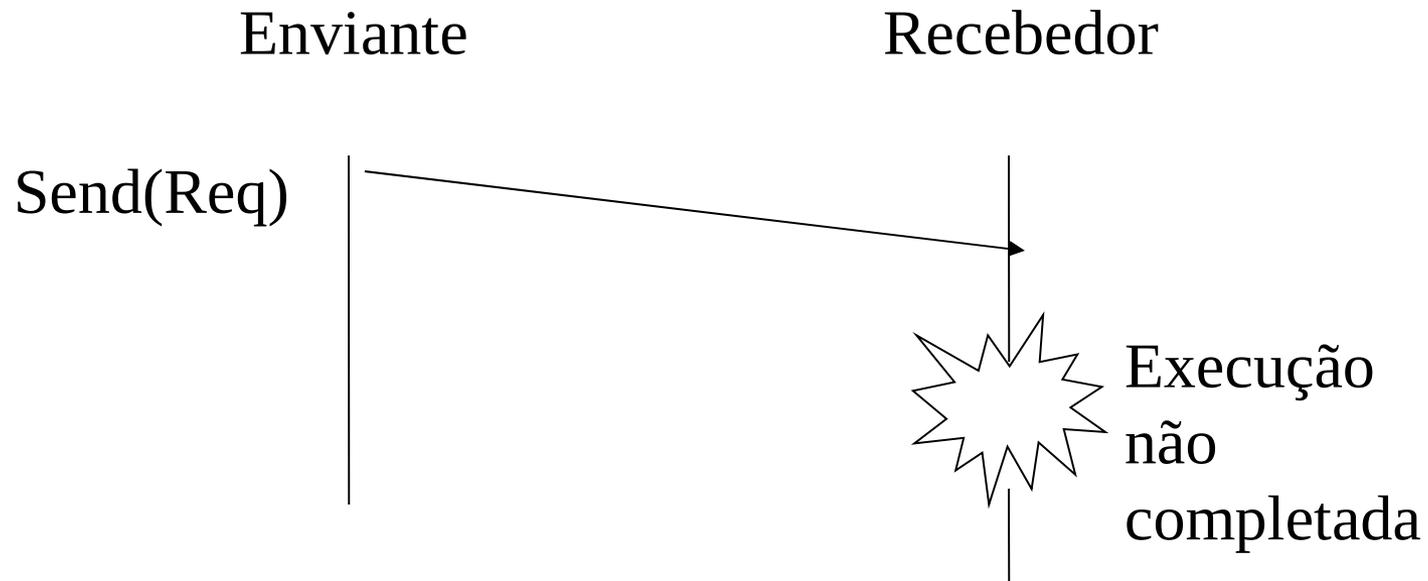
Modelo Cliente/Servidor

Confiabilidade das Primitivas



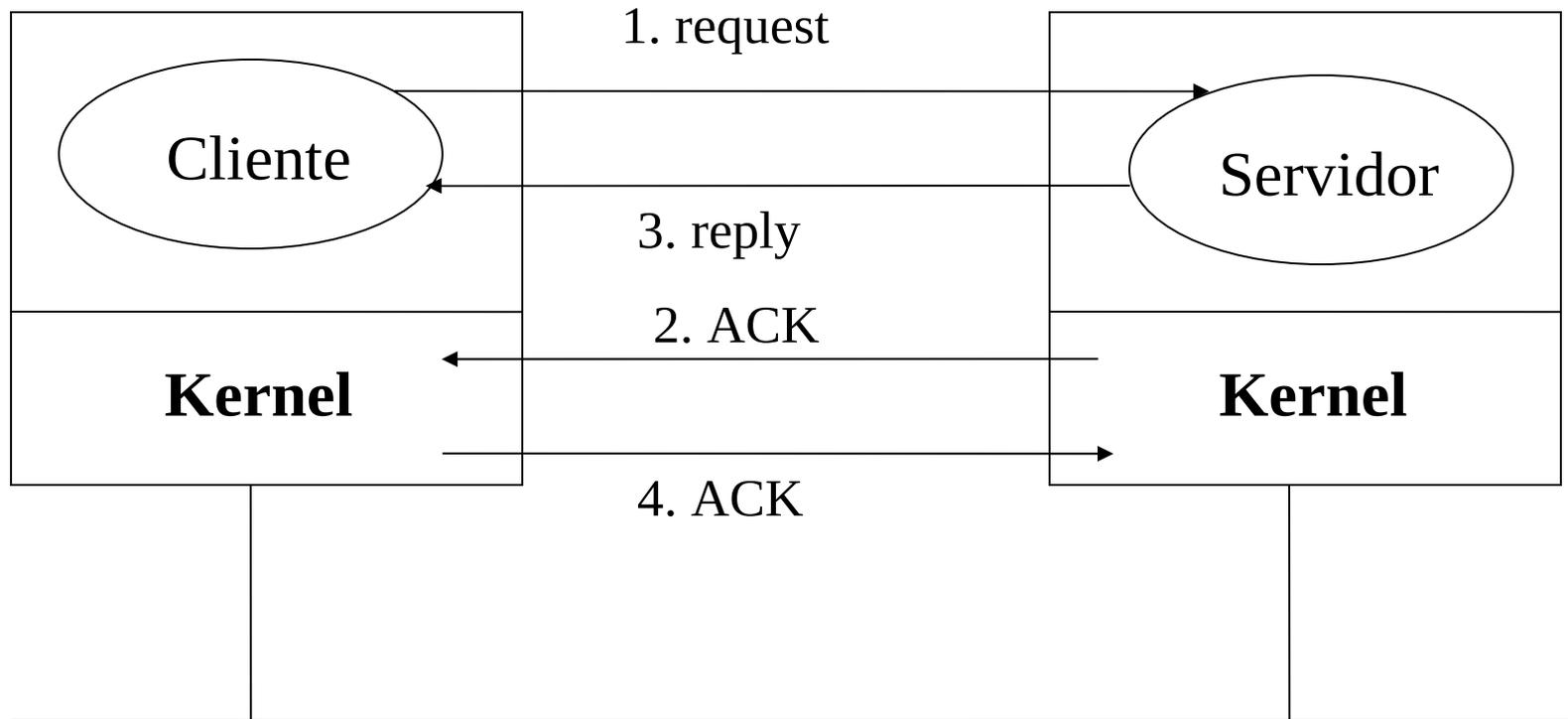
Modelo Cliente/Servidor

Confiabilidade das Primitivas



Modelo Cliente/Servidor

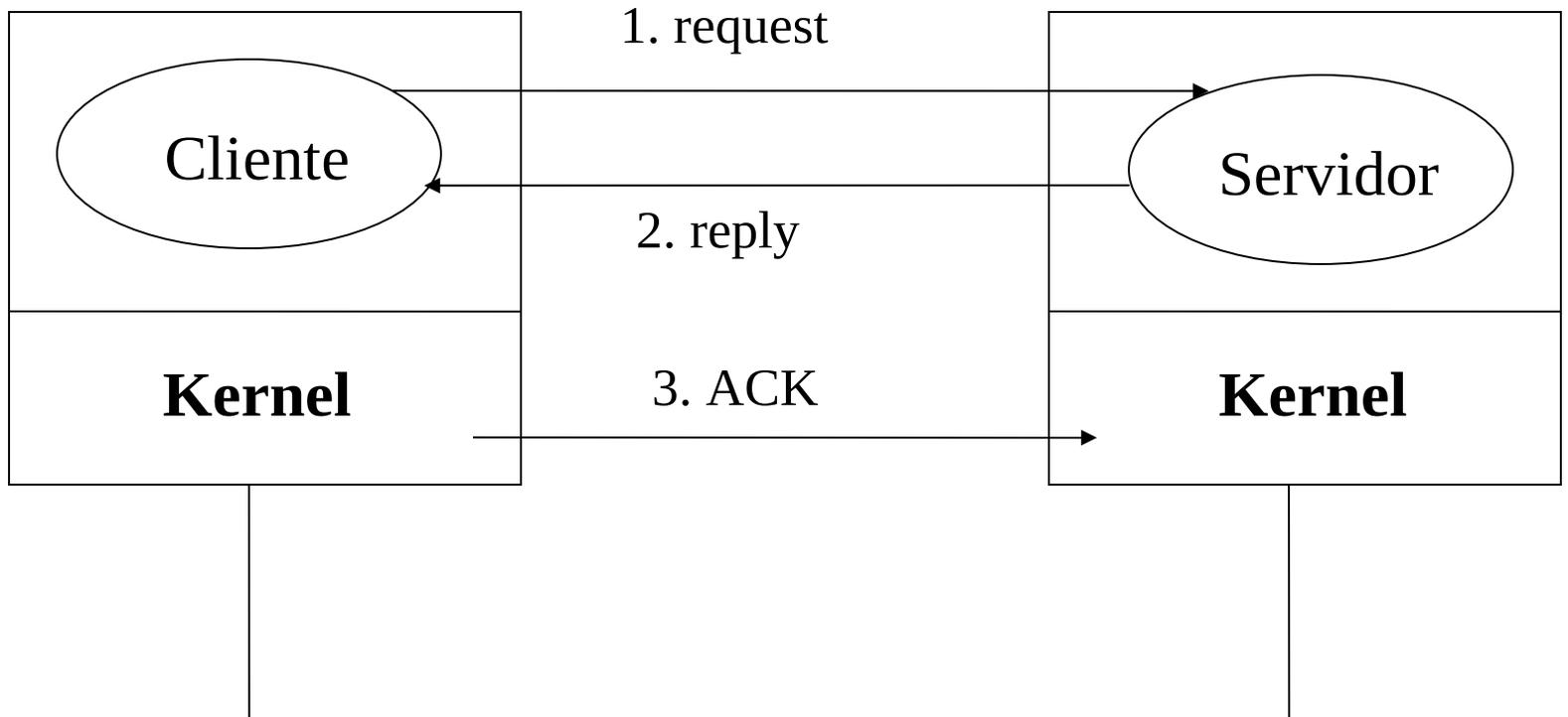
Confiabilidade das Primitivas



Request e Reply são ACK : timeout e retransmissão são usados para recuperar mensagens

Modelo Cliente/Servidor

Confiabilidade das Primitivas (RRA)



Reply serve como ACK para o request

Comunicação em Sistemas Distribuídos

O Modelo Cliente / Servidor

Código	Tipo	Origem	Destino	Descrição
REQ	Requisição	Cliente	Servidor	Cliente serviço
REP	Resposta	Servidor	cliente	Resposta serv.
ACK	Ack	Qualquer	Outro	Confirma
AYA	Voce esta vivo?	Cliente	Servidor	Verifica condição do servidor.
IAA	Estou vivo	Servidor	Cliente	Servidor ativo
TA	Tente de novo	Servidor	Cliente	Servidor não tem espaço
AU	Endereço desconhecido	Servidor	Cliente	Endereço não esta sendo usado

Modelo Cliente/Servidor

Interação Cliente/Servidor

