

P3 – Comunicação 1:n

INE5424-06208B (20242) – Sistemas Operacionais II | Curso de Ciências da Computação

Alexis Mendes Sequeira
(16100717)

Rafael Neves de Mello Oliveira
(17102816)

Luiz Maurício do Valle Pereira
(21104157)

Luis Henrique Goulart Stemmer
(18105165)



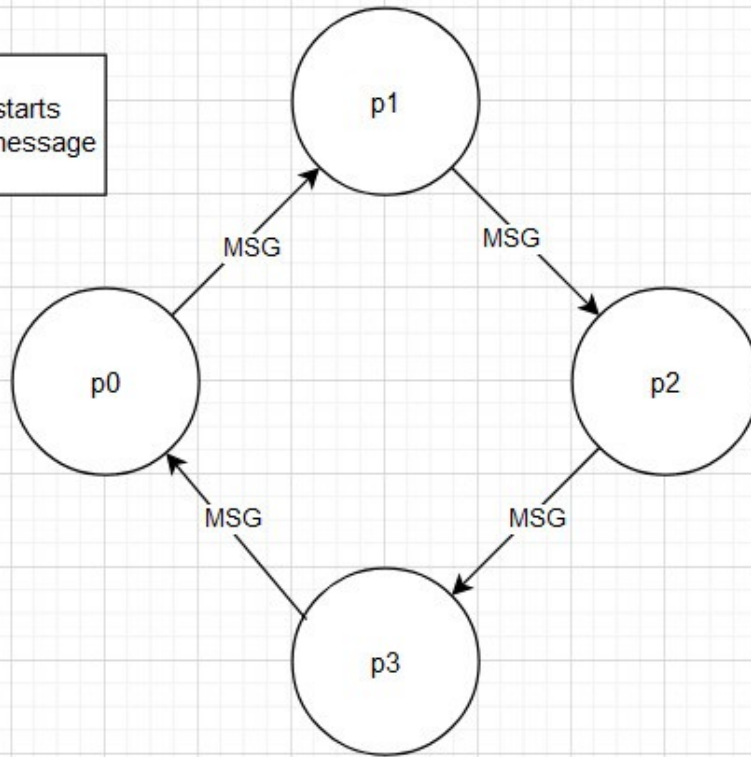
UNIVERSIDADE FEDERAL
DE SANTA CATARINA

Modificação da arquitetura

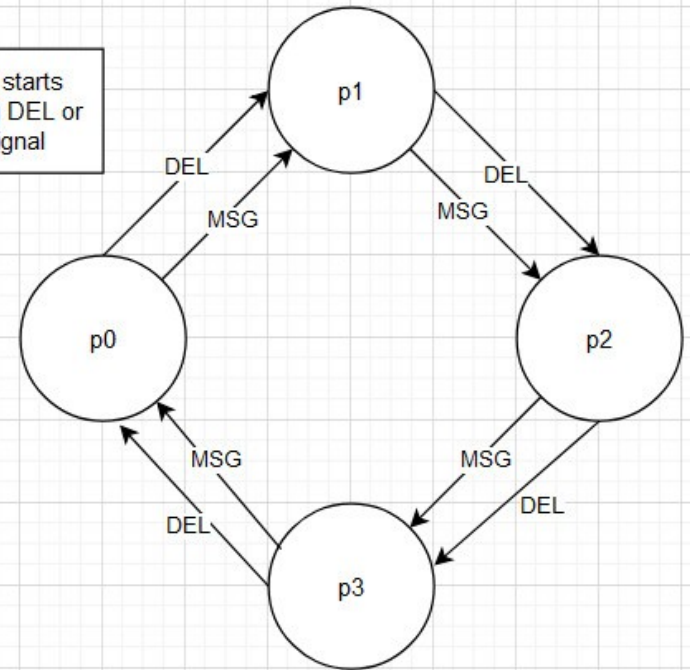
- Não foi implementado ARB nesta entrega devido a escolha do grupo de modificar a arquitetura da implementação, optando por uma implementação via ring, utilizada tanto para Uniform Reliable Broadcast quanto para Atomic Reliable Broadcast.
- No momento, o ordenamento de mensagens só é garantido entre mensagens enviadas pelo mesmo nodo.

Transmissão em anel

Step1: p0 starts transmitting message



Step2: p0 starts transmitting DEL or NDEL signal



Execução

```
luiz@Luiz-Notebook:~/repos/so2$ ./my_project 0 A
B
----- Executando Atomic Broadcast -----
Broadcasting message1 -----
[INFO] Tentativa de broadcast #1 para o nó 1
[INFO] next_node: 1
[INFO] send_message
[INFO] correct ACK
[DEBUG] Send SYN succesfull
[INFO] correct CLOSE
[DEBUG] Send CONTENTS succesfull
[INFO] SENT SUCCESSFULLY
[INFO] Mensagem enviada com sucesso para o nó 1
[INFO] waiting for ring to complete
[INFO] received new message
[DEBUG] checking if syn
[DEBUG] it is syn, loop entered, sending ack
[INFO] Send ack rcv msg
[INFO] Contents received
[DEBUG] Sending CLOSE
[DEBUG] exiting receive()
[INFO] Ring completed
[INFO] Tentativa de broadcast #1 para o nó 1
[INFO] next_node: 1
[INFO] send_message
[INFO] correct ACK
[DEBUG] Send SYN succesfull
[INFO] correct CLOSE
[DEBUG] Send CONTENTS succesfull
[INFO] SENT SUCCESSFULLY
[INFO] Mensagem enviada com sucesso para o nó 1
[INFO] waiting for ring to complete
[INFO] received new message
[DEBUG] checking if syn
[DEBUG] it is syn, loop entered, sending ack
[INFO] Send ack rcv msg
[INFO] Contents received
[DEBUG] Sending CLOSE
[DEBUG] exiting receive()
[INFO] Ring completed
Status do envio: 0
Broadcasting message2 -----
[INFO] Tentativa de broadcast #1 para o nó 1
```

```
luiz@Luiz-Notebook:~/repos/so2$ ./my_project 1 A
B
----- Executando Atomic Broadcast -----
[INFO] received new message
[DEBUG] checking if syn
[DEBUG] it is syn, loop entered, sending ack
[INFO] Send ack rcv msg
[INFO] Contents received
[DEBUG] Sending CLOSE
[DEBUG] exiting receive()
[INFO] AB: Delivering message from node 0
[INFO] send_message
[INFO] correct ACK
[DEBUG] Send SYN succesfull
[INFO] correct CLOSE
[DEBUG] Send CONTENTS succesfull
[INFO] SENT SUCCESSFULLY
[INFO] Waiting signal
[INFO] received new message
[DEBUG] checking if syn
[DEBUG] it is syn, loop entered, sending ack
[INFO] Send ack rcv msg
[INFO] Contents received
[DEBUG] Sending CLOSE
[DEBUG] exiting receive()
D
E
L
[INFO] RESEND Signal
[INFO] send_message
[INFO] correct ACK
[DEBUG] Send SYN succesfull
[INFO] correct CLOSE
[DEBUG] Send CONTENTS succesfull
[INFO] SENT SUCCESSFULLY
```

Received message from process 0:

F
i
r
s
s
t

```
luiz@Luiz-Notebook:~/repos/so2$ ./my_project 2 AB
----- Executando Atomic Broadcast -----
[INFO] received new message
[DEBUG] checking if syn
[DEBUG] it is syn, loop entered, sending ack
[INFO] Send ack rcv msg
[INFO] Contents received
[DEBUG] Sending CLOSE
[DEBUG] exiting receive()
[INFO] AB: Delivering message from node 1
[INFO] send_message
[INFO] correct ACK
[DEBUG] Send SYN succesfull
[INFO] correct CLOSE
[DEBUG] Send CONTENTS succesfull
[INFO] SENT SUCCESSFULLY
[INFO] Waiting signal
[INFO] received new message
[DEBUG] checking if syn
[DEBUG] it is syn, loop entered, sending ack
[INFO] Send ack rcv msg
[INFO] Contents received
[DEBUG] Sending CLOSE
[DEBUG] exiting receive()
D
E
L
[INFO] RESEND Signal
[INFO] send_message
[INFO] correct ACK
[DEBUG] Send SYN succesfull
[INFO] correct CLOSE
[DEBUG] Send CONTENTS succesfull
[INFO] SENT SUCCESSFULLY
```

Received message from process 1:

F
i
r
s
t

```
luiz@Luiz-Notebook:~/repos/so2$ ./my_project 3 AB
----- Executando Atomic Broadcast -----
[INFO] received new message
[DEBUG] checking if syn
[DEBUG] it is syn, loop entered, sending ack
[INFO] Send ack rcv msg
[INFO] Contents received
[DEBUG] Sending CLOSE
[DEBUG] exiting receive()
[INFO] AB: Delivering message from node 2
[INFO] send_message
[INFO] correct ACK
[DEBUG] Send SYN succesfull
[INFO] correct CLOSE
[DEBUG] Send CONTENTS succesfull
[INFO] SENT SUCCESSFULLY
[INFO] Waiting signal
[INFO] received new message
[DEBUG] checking if syn
[DEBUG] it is syn, loop entered, sending ack
[INFO] Send ack rcv msg
[INFO] Contents received
[DEBUG] Sending CLOSE
[DEBUG] exiting receive()
D
E
L
[INFO] RESEND Signal
[INFO] send_message
[INFO] correct ACK
[DEBUG] Send SYN succesfull
[INFO] correct CLOSE
[DEBUG] Send CONTENTS succesfull
[INFO] SENT SUCCESSFULLY
```

Received message from process 2:

F
i
r
s
t

Execução

```
Status do envio: 0
Broadcasting message2 -----
[INFO] Tentativa de broadcast #1 para o nó 1
[INFO] next_node: 1
[INFO] send_message
[INFO] correct ACK
[DEBUG] Send SYN successful
[INFO] correct CLOSE
[DEBUG] Send CONTENTS successful
[INFO] SENT SUCCESSFULLY
[INFO] Mensagem enviada com sucesso para o nó 1
[INFO] waiting for ring to complete
[INFO] received new message
[DEBUG] checking if syn
[DEBUG] it is syn, loop entered, sending ack
[INFO] Send ack rcv msg
[INFO] Contents received
[DEBUG] Sending CLOSE
[DEBUG] exiting receive()
[INFO] Ring completed
[INFO] Tentativa de broadcast #1 para o nó 1
[INFO] next_node: 1
[INFO] send_message
[INFO] correct ACK
[DEBUG] Send SYN successful
[INFO] correct CLOSE
[DEBUG] Send CONTENTS successful
[INFO] SENT SUCCESSFULLY
[INFO] Mensagem enviada com sucesso para o nó 1
[INFO] waiting for ring to complete
[INFO] received new message
[DEBUG] checking if syn
[DEBUG] it is syn, loop entered, sending ack
[INFO] Send ack rcv msg
[INFO] Contents received
[DEBUG] Sending CLOSE
[DEBUG] exiting receive()
[INFO] Ring completed
Status do envio: 0
luiz@luiz-Notebook:~/repos/so2$
luiz@luiz-Notebook:~/repos/so2$
luiz@luiz-Notebook:~/repos/so2$
luiz@luiz-Notebook:~/repos/so2$
luiz@luiz-Notebook:~/repos/so2$
luiz@luiz-Notebook:~/repos/so2$
luiz@luiz-Notebook:~/repos/so2$
luiz@luiz-Notebook:~/repos/so2$
luiz@luiz-Notebook:~/repos/so2$
```

```
Received message from process 0:
F
i
r
s
t

[INFO] received new message
[DEBUG] checking if syn
[DEBUG] it is syn, loop entered, sending ack
[INFO] Send ack rcv msg
[INFO] Contents received
[DEBUG] Sending CLOSE
[DEBUG] exiting receive()
[INFO] AB: Delivering message from node 0
[INFO] send_message
[INFO] correct ACK
[DEBUG] Send SYN successful
[INFO] correct CLOSE
[DEBUG] Send CONTENTS successful
[INFO] SENT SUCCESSFULLY
[INFO] Waiting signal
[INFO] received new message
[DEBUG] checking if syn
[DEBUG] it is syn, loop entered, sending ack
[INFO] Send ack rcv msg
[INFO] Contents received
[DEBUG] Sending CLOSE
[DEBUG] exiting receive()
D
E
L
[INFO] RESEND Signal
[INFO] send_message
[INFO] correct ACK
[DEBUG] Send SYN successful
[INFO] correct CLOSE
[DEBUG] Send CONTENTS successful
[INFO] SENT SUCCESSFULLY

Received message from process 0:
S
e
c
o
n
d
```

```
Received message from process 1:
F
i
r
s
t

[INFO] received new message
[DEBUG] checking if syn
[DEBUG] it is syn, loop entered, sending ack
[INFO] Send ack rcv msg
[INFO] Contents received
[DEBUG] Sending CLOSE
[DEBUG] exiting receive()
[INFO] AB: Delivering message from node 1
[INFO] send_message
[INFO] correct ACK
[DEBUG] Send SYN successful
[INFO] correct CLOSE
[DEBUG] Send CONTENTS successful
[INFO] SENT SUCCESSFULLY
[INFO] Waiting signal
[INFO] received new message
[DEBUG] checking if syn
[DEBUG] it is syn, loop entered, sending ack
[INFO] Send ack rcv msg
[INFO] Contents received
[DEBUG] Sending CLOSE
[DEBUG] exiting receive()
D
E
L
[INFO] RESEND Signal
[INFO] send_message
[INFO] correct ACK
[DEBUG] Send SYN successful
[INFO] correct CLOSE
[DEBUG] Send CONTENTS successful
[INFO] SENT SUCCESSFULLY

Received message from process 1:
S
e
c
o
n
d
```

```
Received message from process 2:
F
i
r
s
t

[INFO] received new message
[DEBUG] checking if syn
[DEBUG] it is syn, loop entered, sending ack
[INFO] Send ack rcv msg
[INFO] Contents received
[DEBUG] Sending CLOSE
[DEBUG] exiting receive()
[INFO] AB: Delivering message from node 2
[INFO] send_message
[INFO] correct ACK
[DEBUG] Send SYN successful
[INFO] correct CLOSE
[DEBUG] Send CONTENTS successful
[INFO] SENT SUCCESSFULLY
[INFO] Waiting signal
[INFO] received new message
[DEBUG] checking if syn
[DEBUG] it is syn, loop entered, sending ack
[INFO] Send ack rcv msg
[INFO] Contents received
[DEBUG] Sending CLOSE
[DEBUG] exiting receive()
D
E
L
[INFO] RESEND Signal
[INFO] send_message
[INFO] correct ACK
[DEBUG] Send SYN successful
[INFO] correct CLOSE
[DEBUG] Send CONTENTS successful
[INFO] SENT SUCCESSFULLY

Received message from process 2:
S
e
c
o
n
d
```

```

int AtomicBroadcastRing::broadcast(const std::vector<uint8_t>& message) {
    int attempt_count = 0;
    const int max_attempts = 3; // Máximo de tentativas

    while (attempt_count < max_attempts) {
        log("Tentativa de broadcast #" + std::to_string(attempt_count + 1) + " para o nó " + std::to_string(next_node_id));

        log("next_node: " + std::to_string(next_node_id));
        // Tenta enviar a mensagem para o próximo nó
        int result = send(next_node_id, message);

        if (result == 0) {
            log("Mensagem enviada com sucesso para o nó " + std::to_string(next_node_id));
            break;
        } else {
            log("Falha ao enviar mensagem na tentativa #" + std::to_string(attempt_count + 1), "WARNING");
            attempt_count++;
        }
    }

    if (attempt_count == max_attempts) {
        log("Exceeded retry amount", "WARNING");
        return -1;
    }

    attempt_count = 0;

    int status = 0;

    log("waiting for ring to complete", "INFO");
    while (attempt_count < max_attempts) {
        Message msg = receive();
        if (msg.sender_id != -1) {
            log("Ring completed", "INFO");
            break;
        }
        attempt_count++;
    }

    if (attempt_count == max_attempts) {
        status = -1;
        log("Falha ao entregar a mensagem após " + std::to_string(max_attempts) + " tentativas.", "ERROR");
    }

    return status;
}

```

```

Message AtomicBroadcastRing::deliver() {
    while (true) {
        Message msg = receive();

        // Deliver message to application
        log("AB: Delivering message from node " + std::to_string(msg.sender_id));

        // Forward the message to the next node in the ring unless it's the sender
        if (msg.sender_id != process_id) {
            send(next_node_id, msg.content);
        }

        return msg;
    }
}

```

Equipe

Alexis Mendes Sequeira (16100717)

Rafael Neves de Mello Oliveira (17102816)

Luiz Maurício do Valle Pereira (21104157)

Luis Henrique Goulart Stemmer (18105165)



UNIVERSIDADE FEDERAL
DE SANTA CATARINA