**Operating System**

**Lab Report 6**

**Hafiz Ahmad**

**19l-1316**

**Section-6A2**

Inter Process Communication using Shared Memory

**INTRODUCTION:**

Inter Process Communication through shared memory is a concept where two or more process can access the common memory. And communication is done via this shared memory where changes made by one process can be viewed by another process. The problem with pipes, fifo and message queue – is that for two process to exchange information. The information has to go through the kernel. Server reads from the input file. The server writes this data in a message using either a pipe, fifo or message queue. The client reads the data from the IPC channel, again requiring the data to be copied from kernel’s IPC buffer to the client’s buffer. Finally, the data is copied from the client’s buffer. A total of four copies of data are required (2 read and 2 write). So, shared memory provides a way by letting two or more processes share a memory segment. With Shared Memory the data is only copied twice – from input file into shared memory and from shared memory to the output file.

Diagram

Description automatically generated

**OBJECTIVES:**

• Learn and Understand InterProcess Communication using implementation of Shared Memory

**Application:**

Shared memory is a memory shared between two or more processes. Each process has its own address space; if any process wants to communicate with some information from its own address space to other processes, then it is only possible with **IPC (inter-process communication) techniques**.

Remote procedure call interfaces.

Platform communication stack.

Operating system communication stack.

Distributed object models.

**Issues:**

No issue found regarding this lab.

**Conclusion:**

In this lab we learn inter process communication between two process using shared memory concepts.