Practical 5

Distributed Operating System

Name - Ritesh Parkhi

Roll No. – 46

Batch – 3

AIM:

To implement CORBA mechanism by using C++ program at one end and Java Program on the other.

THEORY:

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.

PROGRAM:

Writer.cpp

```cpp
#include <iostream>
#include <sys/ipc.h>
#include <sys/shm.h>
#include <stdio.h>
#include<string.h>
using namespace std;

int main()
```

```
{
    // ftok to generate unique key
    key_t key = ftok("shmfile",65);

    // shmget returns an identifier in shmid
    int shmid = shmget(key,1024,0666|IPC_CREAT);

    // shmat to attach to shared memory
    char *str = (char*) shmat(shmid,(void*)0,0);

    cout<<endl<<endl<<"Write Data : ";
    fgets(str, 100, stdin);
    cout<<endl<<"Data written in memory: "<<str<<endl<<endl;


    //detach from shared memory
    shmdt(str);

    return 0;
}
```

Reader.cpp

```
#include <iostream>
#include <sys/ipc.h>
#include <sys/shm.h>
#include <stdio.h>
using namespace std;

int main()
{
    // ftok to generate unique key
    key_t key = ftok("shmfile",65);

    // shmget returns an identifier in shmid
    int shmid = shmget(key,1024,0666|IPC_CREAT);

    // shmat to attach to shared memory
    char *str = (char*) shmat(shmid,(void*)0,0);

    printf("Data read from memory: %s\n",str);

    //detach from shared memory
    shmdt(str);

    // destroy the shared memory
    shmctl(shmid,IPC_RMID,NULL);
```

```
    return 0;
}
```

OUTPUT:

CONCLUSION:

Hence we have successfully built a program to implement CORBA mechanism.