CASE STUDY NO : 15

Date	/ /20
Page No.	71

Case Study On : Study of DCE Thread and various services.

Scheduling -

Thread scheduling is similar to process scheduling, except that it is visible to the application. The scheduling algorithm determines how long a thread may run, and which thread runs next. Just as with process scheduling, many thread scheduling algorithms are possible.
Threads in DCE have priorities and these are respected by the scheduling algorithm. High-priority threads are assumed to be more important than low priority threads, and therefore should get better treatment, meaning they run first and get better treatment, meaning they run first and get a larger portion of the CPU. DCE supports the three threads scheduling algorithms. The first, FIFO searches the priority queses from highest to lowest.

Synchronization -

DCE provides two ways for threads to synchronize : mutexes and condition variables. Mutexes are used when it is essential to prevent multiple threads from accessing the same

resource at the same time. For example, when moving items around on a linked list, partway through the move, the list will be in an incosistend state. To prevend disasdes, when one thread is manipulating the list, all other threads must be kept away. By requiring a thread to first successfully lock the muter associated with the list before touching the list, correct operation can be ensured.

Thread ~~C~~ Calls -

The DCE threads package has a total of 54 primitives. Many of these are not strictly necessary but are provided for convienience only. This approach is same what analogous to a four-function packed calculator that has keys not only for +, -, x and /, but also has keys +1, -1, x2, x10, x π, 12 and 1/10 on the grounds that these save the user time and effort. Due to the large number of calls, we will discuss only most importand ones. Nevertheless, our treatmend should give a reasonable impression of the available functionality. A parend thread can wait for a child using join, which is similar to the ~~wand~~ WAIT system call in UNIX. The DCE package allows the user to create, destory and manage templates for threads, muter and condition variables. The templates can

be set up to have appropriate initial value.

DCE services -

1) Remote Procedure Call (RPCs) -
A distributed application based on the client/server model consists of two parts: The client side of the application, which runs on one machine and makes a request for service on behalf of a user, and the server side of the application, which runs on another machine on the network and fulfills the services request. The two pieces of code on two different machines need to be able to communicate across the network. One model for implementing communications between the client and server of an application is the RPC facility.

2) Time Service -
Time is an important concept in most distributed systems. To see why, consider a research program in radio astronomy. A number of radio telescopes spread all over the world observe the same celestial radio source simultaneously, accurately recording the data and the observation time. The data are send over a network to a central computer for processing.
To try to prevent problems like these, DCE has a service called DTS (Distributed time

service). The goal of DTS is to keep clocks on separate machines synchronized. Getting them synchronized once is not enough, because the crystals in different clocks tick at slightly different rates.

3) Directory Service -

A major goal of DCE is to make all resources accessible to any process in the system, without regard to the relative location of the resource user and the resource provider. These resources include users, machines, cells, services, files, security data, and many others. To accomplish this goal, it is necessary for DCE to maintain a directory service that keeps track of where all resources are located and provide people-friendly names for them.

The DCE directory service is organized per cell. Each cell has a cell Directory Service (CDS), which stores the names and properties of the cell's resources.

Date	/ /20
Page No.	75

4) Security Service -

In most distributed system, security is a major screen concern. The system administrator may have definite ideas about who can use which resource and many users may want their files and mailboxes protected from prying eyes. These issues arises in traditional timesharing system too, but there they are solved simply by having the kernel manage all the resources.
In a distributed system consisting of potentially untrustworthy machines communicating over an insecure network this solution.