Midterm Questions and Solution

Department of Computer Science and Engineering

Sub: Distributed Systems (8CS3A)

VIII Semester **Midterm –II** **Midterm Date: 16/02/2018**

Subject In Charge: Rukhsar Sultana

Q.1. Write Chandy-Lamport algorithm for consistent state recording in distributed systems 2.5 Marks

Q.2. Write synchronization rules for logical clocks. 2.5 Marks

Q.3. What is DCE and DCE cell? Explain its architecture and services with diagram. 5 Marks

OR

Write short notes on: (i).Distributed Systems (ii).Interaction Models (iii).Client-Server paradigm for distributed computing.

Solution 1: Chandy-Lamport Algorithm for Consistent State Recording

The algorithm makes the following assumptions:

1. Channels are unidirectional.
2. Channels have unbounded capacities to hold messages.
3. Channels are FIFO.

Following two notations are used to determine the state of a channel:

$Received_{ij}$: The set of messages received by process P_j on channel Ch_{ij} before it received the marker on channel Ch_{ij}.
$Recorded\_recd_{ij}$: The set of messages recorded as received over channel Ch_{ij} in the state of process P_j.

Algorithm:

1. When a process P_i initiates the state recording: P_i records its own state and sends a marker on each outgoing channel connected to it.

2. When process P_j receives a marker over an incoming channel Ch_{ij}: Process P_j performs the following actions:
   a. If P_j had not received any marker earlier, then
      i. Record its own state.
      ii. Record the state of channel Ch_{ij} as empty.
      iii. Send a marker on each outgoing channel connected to it.
   b. Otherwise, record the state of channel Ch_{ij} as the set of messages $Received_{ij} - Recorded\_recd_{ij}$.

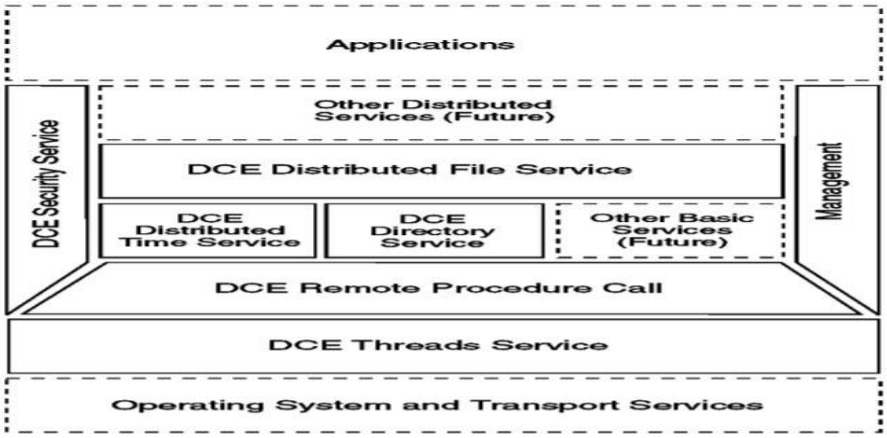
Solution 2: A process may increment its local clock by 1 only when an event occurs in it, and synchronize the local clock, if necessary, when it receives a message. Such clocks are called logical clocks. We denote the logical clock of process P_k by LC_k. Logical clocks are implemented according to the following rules:
R1: A process P_k increments LC_k by 1 whenever an event occurs in it.
R2: When process P_k receives a message m containing $t_s(send(m))$, P_k sets its clock by the rule $LC_k = \max(LC_k, ts(send(m))+1)$.

Solution 3: DCE (Distributed Computing Environment)
DCE was developed by the Open Software Foundation (OSF) to develop standard (cross-platform) computing solutions and to provide framework for client server applications.

Architecture: The largest unit of management in DCE is a cell. The highest privileges within a cell are assigned to a role called cell administrator , who is a real OS – Level user.
Three privileges categories of users are :

- user_obj : Owner
- group_obj : Group member
- other_obj : Any other DCE/non-DCE principal.

Major Components of DCE Cell
1) Security Server: that is responsible for authentication.
2) Cell Directory Server: that is the repository of resources and ACLs.
3) DCE Time Server: that provides an accurate clock for proper functioning of the entire cell.



Architecture Of DCE

Services of DCE

1. Remote Procedure Call (RPC): A procedure call is a method of implementing the Client/Server Communication. The procedure call is translated into network communications by the underlying RPC mechanism.
2. Directory Service: The DCE Directory Service advertises that the server supports the new interface defined using the IDL. DCE Security Service also ensures that only authorized client end users can access the newly defined server function.
3. Security Service: There are three aspects to DCE security:

- Authentication: This identifies that a DCE user or service is allowed to use the service.
- Secure communications: Communication over the network can be checked for tampering or encrypted for privacy.
- Authorization: This issues the permission to access the service. These are implemented by several services and facilities which include the Registry Service, Privilege Service, Access Control List (ACL) Facility, and Login Facility.

4. Time Service: The DCE Time Service (DTS) provides synchronized time on the computers participating in a Distributed Computing Environment. DTS synchronizes a DCE host's time with Coordinated Universal Time (UTC), an international time standard.
5. File Service: The DCE File Service (DFS) allows users to access and share files stored on a File Server anywhere on the network, without having to know the physical location of the file.
6. Threads: DCE Threads supports the creation, management, and synchronization of multiple threads of control within a single process. This component is conceptually a part of the operating system layer, the layer below DCE.

OR

(i) Distributed Systems: A distributed system is one in which components located at networked computers communicate and coordinate their actions only by passing messages. A distributed system is a collection of independent computers that appear to the users of the system as a single computer.

Example DS: Web (and many of its applications like Online bookshop), Data Centers and Clouds

Features: Features of DS are parallel activities, communication via message passing, resource sharing, no global state, no global clock.

Challenges: Challenges are heterogeneity, distribution transparency, fault tolerance, scalability concurrency, openness, security.

(ii) Interaction Models: Interaction Models deals with performance and with difficulty of setting time limits in a distributed system. There are two types of distributed systems

1. Synchronous DS – In this type

- The time taken to execute a step of a process has known lower and upper bounds.
- Each message transmitted over a channel is received within a known bounded time.
- Each process has a local clock whose drift rate from real time has known bound.

2. Asynchronous DS: There is No bounds on:

- Process execution speeds
- Message transmission delays
- Clock drift rates.

(iii) Client-Server paradigm for distributed computing: In this paradigm processes take on the roles of being clients or servers. In particular, client processes interact with individual server processes to request for services in potentially separate host computers and then server provides the response/result for request.