DISTRIBUTED SYSTEMS

B.Tech (C.S.E), 7th Semester (A.Y. 2022-23) NIT Raipur

Assignment-1&2

SET-1: (Roll No. 19115001 to 19115017 & 18115016, 18115055)

1. A system consists of three processors P_1, P_2 and P_3 and a process having four tasks t_1, t_2, t_3 and t_4 is to be executed on this system. Suppose E_{ij} is the cost of executing task t_i, on processor p_i and C_{ij} is the communication cost between tasks t_i, and t_j when the two tasks are assigned to different processors. Let E_{11} = 31, E_{12} = 4, E_{13} = 14, E_{21}= I, E_{22} = 5, E_{23} = 6, E_{31}= 2, E_{32}= 4, E_{33} = 24, E_{41}=3, E_{42} = 28 , E_{43} = 10, C_{12} = 35 , C_{13} = 3 , C_{14} = 8, C_{23} = 6 , C_{24} =4, and C_{34} =23. Find an optimal assignment of the tasks to the processors and calculate the cost of optimal assignment. Now compare this cost with the assignment cost of the case in which t_1 and t_2 are assigned to p_1, t_3 is assigned to P_2 and t_4 is assigned to P_3.

2. Suppose you have decided to use the high-low policy as the process transfer policy of a load balancing algorithm for a distributed system. Suggest a suitable method that you will use in your implementation for choosing the high-mark and low-mark values. Do these threshold values have to be the same for all processors in the system? Give reasons for your answer.

3. What are some of the main issues involved in freezing a migrant process on its source node and restarting it on its destination node? Give a method for handling each of these issues.

4. Give an example to show how a server process can be designed to benefit from the concurrency made possible by threads. Now give an example to show how a client process can be designed to benefit from the concurrency made possible by threads.

5. Discuss the relative advantages and disadvantages of implementing a threads package in user space and in the kernel.

6. A stateful file server records state information for its clients. What problems are associated with this type of file server? Give two examples where it might be necessary to use stateful file servers.

7. An application consists of three transactions T}, T2, and T3 that are defined below:

T_1: begin_transaction
    read(x); read(z); write(x–5); write(z+5);
end_transaction
T_2: begin_transaction
    read(z); write(z–8); read(y); write(y+8);
end_transaction
T_3: begin_transaction
    read(x); write(x+4); read(y); write(y–4);
end_transaction

Describe how the concurrency of these three transactions can be controlled by using the following: (a) Same type of locks for both read and write operations (b) Type-specific locks (c) Intention-to-write locks (d) Optimistic concurrency control scheme (e) Timestamp-based concurrency control scheme.

8. In what manner is the timestamp-based concurrency control scheme better than the optimistic concurrency control scheme? For the example given by you for case (b) of the previous exercise, use the timestamp-based concurrency control scheme and show that the transaction that was aborted cannot complete its first phase in this case.

9. What are the commonly used approaches for user authentication in computer systems? Explain how a user is authenticated in each of these approaches.

10. What are the most important scalability problems in JINI?

1. A system has two processors P1 and P2with P1 having limited memory capacity and P2 having unlimited memory. A process having multiple tasks is to be executed on this system. The execution costs for each task on each processor, the intertask communication costs, the memory requirement of each task, and the total memory capacity of processor P1 is given. Suppose a1 is the assignment that minimizes the total execution and communication costs without the memory size constraint, and a2 is the assignment that minimizes the total execution and communication costs with the memory size constraint. Prove that the tasks assigned to P1 for assignment a2 is a subset of the tasks assigned to P1 for assignment a1.

2. What are the main differences between the load-balancing and load-sharing approaches for process scheduling in distributed systems? Which of the various policies to be used in the implementation of the two approaches are different and which of them are same?

3. From the point of view of supporting pre-emptive process migration facility, is a stateless or stateful file server preferable? Give reasons for your answer.

4. In a distributed system, parallelism improves performance and blocking system calls make programming easier. Explain how the concept of threads can be used to combine both advantages.

5. The operating system of a computer uses processes as the basic unit of CPU utilization. That is, it does not support threads. Can threads facility be provided in this computer system without modifying the operating system kernel? If no, explain why. If yes, explain how.

6. Suppose a file system uses the client-initiated approach for validating the contents of client caches. Also suppose that the validity check is performed by comparing the time of last modification of the cached version of the data with the server's master copy version. This file system will not function correctly in a system in which the clocks of various nodes are not synchronized. Suggest a scheme that can be used with the client-initiated approach for cache validation in systems in which clocks of various nodes are not synchronized.

7. Let the initial balance in all the three accounts of the two banking transactions of Figure 9.9 be $100. For these two transactions, enumerate all the schedules that produce different final values of z. Which of these are legal schedules?

a_1: read balance (x) of account X
a_2: read balance (z) of account Z
a_3: write $(x - 5)$ to account X
a_4: write $(z + 5)$ to account Z

T_1 : Transfer $5 from account X to account Z.

b_1: read balance (y) of account Y
b_2: read balance (z) of account Z
b_3: write $(y - 7)$ to account Y
b_4: write $(z + 7)$ to account Z

T_2 : Transfer $7 from account Y to account Z.   **Fig. 9.9**   Two banking transactions.

8. What are serially equivalent schedules? For the three transactions of the previous exercise give at least six schedules that are serially equivalent.

9. Explain the password-based approach for user logins authentication. What are the problems associated with this approach? Suggest solutions to overcome these problems.

10. A lease period in JINI is always specified as a duration and not as an absolute time at which the lease expires. Why is this done?

1. Comment on the practical applicability of the load-balancing approach as a scheduling scheme for the following types of distributed systems: (a) A LAN-based distributed system (b) A WAN-based distributed system (c) A distributed system based on the processor-pool model (d) A distributed system based on the workstation-server model.

2. Suggest some policies that may be used for load estimation in load-balancing algorithms. Discuss the relative advantages and disadvantages of the policies suggested by you. Which one of the policies suggested by you can also be used for load-sharing algorithms? If none, suggest a suitable load estimation policy for a load-sharing algorithm.

3. Differentiate between pre-emptive and non-pre-emptive process migration. What are their relative advantages and disadvantages? Suppose you have to design a process migration facility for a distributed system. What factors will influence your decision to design a pre-emptive or a non-pre-emptive process migration facility?

4. List some of the potential advantages and disadvantages of process migration.

5. A distributed system has three types of processors A, B, and C. The numbers of bits used for the exponent of a floating-point number by processors of types A, B, and C are 8, 12, and 16, respectively; the numbers of bits used for the mantissa of a floating-point number by processors of types A B, and C are 16, 32, and 64, respectively. In this system, from which processor type to which processor type should process migration be allowed and from which processor type to which processor type should processor migration not be allowed? Give reasons for your answer.

6. In the design of a distributed operating system, the data-caching mechanism may be used for caching many different types of data. A separate cache can be maintained for each type of data. In your opinion, is it necessary to always keep a cached data up to date? If yes, explain why. If no, give an example in which a system can function correctly even when processes access cached data that are not always up to date.

7. Give two methods that may be used in the design of a file system to record updates to a file in a reversible manner. That is, the file system provides the flexibility to the users to cancel the updates made to a file within an open-close session and revert the file back to the state that it was in before the start of the session.

8. Prove that if all transactions of an application use the two-phase locking protocol, all schedules formed by interleaving their operations are serializable.

9. In a distributed system it is desired that a server process should serve any client that needs its service only after verifying the identity of the client. Describe how to implement this authentication requirement.

10. Some implementations of distributed-object middleware systems are entirely based on dynamic method invocations. Even static invocations are compiled to dynamic ones. What is the benefit of this approach?

1. A distributed operating system designer is of the opinion that state information in a distributed system is typically gathered at a high cost. Therefore, for a distributed system based on the processor-pool model, he or she decides to use a load-balancing policy that uses the following simple process placement rule: Execute all processes originating from terminal i on processor j (j << i). The value of j is defined for all values of i, and for several values of i, the value of j may be the same. In your opinion, is the designer's choice of the scheduling algorithm appropriate for this system? Give reasons for your answer. What drawbacks, if any, does this scheduling algorithm have? If you feel that this algorithm is not appropriate for this system, suggest a suitable global scheduling algorithm for this system.

2. Suppose you have to design a load-sharing algorithm for a distributed system. Will you prefer to use a sender-initiated or a receiver-initiated location policy in your algorithm? Give reasons for your answer.

3. When a migrant process is restarted on its destination node after migration, it is given the same process identifier that it had on its source node. Is this necessary? Give reasons for your answer.

4. When should the external data representation mechanism be used in the design of a process migration facility? Suppose you have to design the external data representation format for a process migration facility. What important factors will influence your design decisions?

5. Differentiate between handoff scheduling and affinity scheduling of threads. In your opinion, which of the two is a more desirable feature for a threads package and why?

6. Of the four data transfer models that may be used in a distributed file system that uses the data caching model for file accessing, which models are suitable for each of the following types of distributed systems: (a) A distributed system that supports diskless workstations (b) A distributed system in which each node has large disk storage space (c) A distributed system that uses the structured file model in which each file is a group of records.

   If more than one model is suitable for a particular case, which model will you prefer to use and why?

7. Discuss the need for serializability property in transactions. What is the main goal in devising a mechanism for satisfying this property? Describe at least three mechanisms that may be used in the implementation of a transaction facility for satisfying this property. Now compare the mechanisms described by you to show how close each of them is to the main goal.

8. What is false sharing? Discuss the importance of granularity of locks in combating the false sharing problem.

9. In the preceding exercise, suppose that both the client and the server should verify each other's authenticity before a communication session can be started between them. Describe how to implement this authentication requirement.

10. Suppose that a JINI client caches the tuple it obtained from a JavaSpace so that it can avoid having to go to the JavaSpace the next time. Does this caching make any sense?

1. Suppose you have to design a centralized load-balancing algorithm for global scheduling of processes in a distributed system. What issues must you handle? Suggest a suitable method for handling each of the issues mentioned by you.

2. A distributed system does not support pre-emptive process migration facility. You are to design a load-sharing algorithm for scheduling of processes in this system. Will you use a sender-initiated or a receiver-initiated location policy for your algorithm? Give reasons for your answer.

3. Which one or more of the address space transfer mechanisms described in this chapter are suitable for a process migration facility with the following goals? (a) High performance is the main goal. (b) High reliability is the main goal. (c) Effectiveness of process migration policy is the main goal. (d) Simple implementation is the main goal. (e) Both reliability and effectiveness of process migration policy are important goals. If more than one mechanism is suitable for a particular case, which one will you prefer to use and why?

4. Which of the mechanisms described in this chapter to handle communication among co-processes are suitable for a process migration facility with the following goals? (a) Performance is the main goal. (b) Reliability is the main goal. (c) Simple implementation is the main goal. If more than one mechanisms are suitable for a particular case, which one will you prefer to use and why?

5. What is an immutable file? Can a file system be designed to function correctly by using only immutable files? If no, explain why. If yes, explain how the basic file operations (create, read, write, delete) can be performed in this file system for shared files.

6. A distributed operating system designer is of the opinion that since both replication and caching of objects provide more or less similar advantages to a distributed system, both concepts need not be implemented in the same distributed system. Is he or she correct? Give reasons for your answer. Now differentiate among the following types of distributed operating systems by listing their relative advantages and disadvantages: (a) One that implements object caching but no object replication (b) One that implements object replication but no object caching (c) One that implements both object caching and object replication

7. Why are transactions needed in a file service? Give suitable examples to illustrate how transactions help in doing the following: (a) Improving the recoverability of files in the event of failures (b) Allowing the concurrent sharing of mutable files by multiple clients in a consistent manner.

8. What is a transaction deadlock? Give an example to illustrate how a transaction deadlock occurs when: (a) The same type of locks are used for both read and write operations. (b) Type-specific locks are used. Now give a method that may be used to avoid transaction deadlocks and apply the method to your examples to show how it prevents the deadlocks that occurred in cases (a) and (b).

9. The password mechanism is used in a distributed system to authenticate users at login time. State the most suitable locations (according to you) for storing the login program and the password file in the following cases: (a) The distributed system is based on the workstation-server model with each workstation having a small hard disk of about 20 megabytes capacity. (b) The distributed system is based on the workstation-server model. Some of the workstations are diskless and others have a small hard disk of about 20 megabytes capacity. (c) The distributed system is based on the processor-pool model. Assume that any user is free to use any of the user terminals or workstations.

10. An implementation of CORBA's asynchronous method invocation does not affect the server-side implementation of an object. Explain why this is the case.

1. Suppose you have to design a load-balancing algorithm for a distributed system. Which of the selfish, altruistic, and intermediate priority assignment policies will you use in your algorithm if the distributed system is based on the following: (a) Workstation-server model (b) Processor-pool model Give reasons for your answer?

2. A distributed system uses the all-or-nothing strategy as the process transfer policy for its load sharing algorithm. Explain why the processing capabilities of the processors of this system may not be properly utilized. Suggest a suitable method to overcome this problem.

3. Which one or more of the message-forwarding mechanisms described in this chapter are suitable for a process migration facility with the following goals? (a) Transparency is the main goal. (b) Reliability is the main goal. (c) Performance is the main goal. (d) Simple implementation is the main goal. If more than one mechanisms are suitable for a particular case, which one will you prefer to use and why?

4. The process migration facility of a distributed system does not allow free migration of processes from one node to another but has certain restrictions regarding which node's processes can be migrated to which other nodes of the system. What might be the reasons behind imposing such a restriction?

5. Discuss the relative advantages and disadvantages of using full-file caching and block caching models for the data-caching mechanism of a distributed file system.

6. Suppose you have to design the caching scheme of a distributed file system that has to support session semantics. Suggest a suitable solution for each of the following issues in your design: (a) Where to locate a cache? (b) What should be the unit of data caching? (c) When should modification to a cached data be propagated to its master copy? (d) When should the validation to check if a cached copy of a data is consistent with its master copy be performed?

   Give reasons for your answer.

7. What is a transaction? What are the two main factors that threaten the atomicity of transactions? Describe how atomicity is ensured for a transaction in both commit and abort.

8. Suppose the optimistic concurrency control mechanism is used in the implementation of the transaction mechanism of a file system. Give two examples of concurrently executing transactions in this system: (a) One in which the validation check is successful for all transactions and they are committed successfully (b) One in which the validation check fails and a transaction has to be aborted.

9. The password mechanism is used in a distributed system to authenticate users at login time. State the most suitable locations (according to you) for storing the login program and the password file in the following cases: (a) The distributed system is based on the workstation-server model with each workstation having a small hard disk of about 20 megabytes capacity. (b) The distributed system is based on the workstation-server model. Some of the workstations are diskless and others have a small hard disk of about 20 megabytes capacity. (c) The distributed system is based on the processor-pool model. Assume that any user is free to use any of the user terminals or workstations.

10. Is it possible to have system-specific implementations of CORBA object references while still being able to exchange references with other CORBA-based systems?