Assignment 1

Geoffrey Rathinapandi

CS5352 Advanced Operating Systems Design
Spring 2018

Due Date: 2/13, 8 a.m., soft copy via Blackboard.
Late submissions are accepted till 2/20, 8 a.m., with 10% penalty each day.
No submissions accepted after 2/20, 8 a.m.

Q1. (Chapter 1, Problem 2) What is the role of middleware in a distributed system?

Middleware in a distributed system is to enhance the distribution transparency that is missing in network operating systems. In other words, middleware aims at improving the single-system

Q2. (Chapter 1, Problem 4) Explain what is meant by (distribution) transparency, and give examples of different types of transparency.

Distribution transparency is the phenomenon by which distribution aspects in a system are hidden from users and applications. Examples include access transparency, location transparency, migration transparency, relocation transparency, replication transparency, concurrency transparency, failure transparency, and persistence transparency.

Q3. (Chapter 1, Problem 7) What is an open distributed system and what benefits does openness provide?

An open distributed system offers services according to clearly defined rules. An open system is capable of easily interoperating with other open systems but also allows applications to be easily ported between different implementations of the same system.

Q4. Please briefly explain what cluster computing systems, high performance computing systems, grid computing systems, and cloud computing systems are.

High-performance computing (HPC) is the use of parallel processing for running advanced application programs efficiently, reliably and quickly. The term applies especially to systems that function above a teraflop or 10^12 floating-point operations per second.

Cluster is a system comprising two or more computers or systems (called nodes) which work together to execute applications or perform other tasks, so that users who use them, have the impression that only a single system responds to them, thus creating an illusion of a single resource (virtual machine). This concept is called transparency of the system. As key features for the construction of these platforms is included elevation : reliability, load balancing and performance.

Grid computing system is a processor architecture that combines computer resources from various domains to reach a main objective. In grid computing, the computers on the network can work on a task together, thus functioning as a supercomputer.
Cloud computing system means storing and accessing data and programs over the Internet instead of your computer's hard drive

Q5. (Chapter 1, Problem 9) Scalability can be achieved by applying different techniques. Can you give three examples?

Scalability:
Scalability is the capability of a system, network, or process to handle a growing amount of work, or its potential to be enlarged to accommodate that growth.

Scaling can be achieved through
- Distribution
- Replication
- Caching.

Q6. (Chapter 2, Problem 2) What is a three-tiered client-server architecture?

Three-tier architectures are programming models that enable the distribution of application functionality across three independent systems, typically:

- Client components running on local workstations (tier one)
- Processes running on remote servers (tier two)
- A discrete collection of databases, resource managers, and mainframe applications (tier three)

Q7. (Chapter 2, Problem 4) Consider a chain of processes P_1, P_2, ..., P_n implementing a multitiered client-server architecture. Process P_i is client of process P_{i+1}, and P_i will return a reply to P_{i-1} only after receiving a reply from P_{i+1}. What are the main problems with this organization when taking a look at the request-reply performance at process P_1?

Performance can be expected to be bad for large n. The problem is that each communication between two successive layers is, in principle, between two different machines. Consequently, the performance between P1 and P2 may also be determined by n - 2 request-reply interactions between the other layers. Another problem is that if one machine in the chain performs badly or is even temporarily unreachable, then this will immediately degrade the performance at the highest level.

Q8. (Chapter 2, Problem 5) In a structured overlay network, messages are routed according to the topology of the overlay. What is an important disadvantage of this approach?

The problem is that we are dealing only with logical paths. It may very well be the case that two nodes A and B which are neighbors in the overlay network are physically placed far apart. As a consequence, the logically short path between A and B may require routing a message along a very long path in the underlying physical network.

Q9. Please describe what operations are performed when a node joins or leaves the DHT-based system.

When a node joins:

- Node starts with a random id
- Lookup returns scc(id)
- Contact succ(id) and its predecessor and insert itself in the ring (nodes need store info of predecessor)
- Transfer data items belonging to node id from succ(id)

When a node leaves:

- Node id informs departure to predecessor and successor
- Transfer data items to succ(id).

Q10. (Chapter 2, Problem 17) Sketch a solution to automatically determine the best trace length for predicting replication policies in Globule.

An origin server would need to use the traces from Ti to Ti+1 to check its prediction of policy for that period. It can simply see whether the policy that would have been chosen on the actual access patterns is the same as the one chosen based on the requests in the period Ti-1 to Ti. This would allow the server to compute the prediction error. By varying the trace length, the origin server would be able .nd the length for which the prediction is minimal. In this way, we get an automatic determination of the optimal trace length, effectively contributing to the self-managing nature of Globule.

THE END