Distributed Systems: Concepts and Design

Edition 5

By George Coulouris, Jean Dollimore and Tim Kindberg and Gordon Blair

Chapter 1 Exercise Solutions

1.1 We define a distributed system as one in which hardware or software components located at networked computers communicate and coordinate their actions only by passing messages. What are the consequences of defining a distributed system in this manner?

1.1 Ans.

a. Concurrency

b. No global clock

c. Independent failures.

1.2 How might the clocks in two computers that are linked by a local network be synchronized without reference to an external time source? What factors limit the accuracy of the procedure you have described? How could the clocks in a large number of computers connected by the Internet be synchronized? Discuss the accuracy of that procedure.

1.2 Ans.

Several time synchronization protocols are described in Section 10.3. One of these is Cristian's protocol. Briefly, the round trip time t to send a message and a reply between computer A and computer B is measured by repeated tests; then computer A sends its clock setting T to computer B. B sets its clock to $T+t/2$. The setting can be refined by repetition. The procedure is subject to inaccuracy because of contention for the use of the local network from other computers and delays in the processing the messages in the operating systems of A and B. For a local network, the accuracy is probably within 1 ms.

For a large number of computers, one computer should be nominated to act as the time server and it should carry out Cristian's protocol with all of them. The protocol can be initiated by each in turn. Additional inaccuracies arise in the Internet because messages are delayed as they pass through switches in wider area networks. For a wide area network the accuracy is probably within 5-10 ms. These answers do not take into account the need for fault-tolerance. See Chapter 10 for further details.

1.3 Consider the implementation strategies for massively multiplayer online games as discussed in Section 1.2.2. In particular, what advantages do you see in adopting a single server approach for representing the state of the multiplayer game? What problems can you identify and how might they be resolved?

1.3 Ans.

The advantages of having a single server maintain a representation of the game are that: (1) there is a single copy and hence no need to maintain consistency of multiple copies; and (2) clients have a single place to go to discover this state. There may also be advantages to having a global view of the entire systems state.

The potential problems are that this single server may fail and may also become a bottleneck affecting the performance and scalability of the approach. To handle failure, it would be necessary to introduce replication, which in turn would require a solution to maintaining consistency across replicas. There are a number of solutions to improving performance and scalability including running the server on a cluster architecture as described in Section 1.2.2, or again using replication and load balancing within the distributed environment. Alternatively, a peer-to-peer solution can be adopted. These techniques are covered throughout the book.

1.4 A user arrives at a railway station that she has never visited before, carrying a PDA that is capable of wireless networking. Suggest how the user could be provided with information about the local services and amenities at that station, without entering the station's name or attributes. What technical challenges must be overcome?

1.4 Ans.

The user must be able to acquire the address of locally relevant information as automatically as possible. One method is for the local wireless network to provide the URL of web pages about the locality over a local wireless network.

For this to work: (1) the user must run a program on her device that listens for these URLs, and which gives the user sufficient control that she is not swamped by unwanted URLs of the places she passes through; and (2) the means of propagating the URL (e.g. infrared or an 802.11 wireless LAN) should have a reach that corresponds to the physical spread of the place itself.

1.5 Distributed systems are going through a period of significant change**,** which can be traced back to a number of influential trends. Can you identify what these might be?

1.5 Ans.

a. The emergence of pervasive networking technology.

b. The emergence of ubiquitous computing coupled with the desire to support user mobility in distributed systems.

c. The increasing demand for multimedia services.

d. The view of distributed systems as a utility.

1.6 Due to the increasing maturity of distributed systems infrastructure, organizations are moving towards viewing distributed systems as a utility. In this model, resources are provided by appropriate service suppliers and effectively rented rather than owned by the end user. Explain this model with respect to physical resources and software services. Can you give examples of some companies that support such software services?

1.6 Ans.

a. Physical resources such as storage and processing can be made available to networked computers removing the need to own such resources on their own.

b. Software services can also be made available across the global Internet using this approach.

Example of companies supporting such software services are www.google.com, www.yahoo.com.

1.7 A server program written in one language (for example C++) provides the implementation of a BLOB object that is intended to be accessed by clients that may be written in a different language (for example Java). The client and server computers may have different hardware, but all of them are attached to an internet. Describe the problems due to each of the five aspects of heterogeneity that need to be solved to make it possible for a client object to invoke a method on the server object.

1.7 Ans.

As the computers are attached to an internet, we can assume that Internet protocols deal with differences in networks.

But the computers may have different hardware - therefore we have to deal with differences of representation of data items in request and reply messages from clients to objects. A common standard will be defined for each type of data item that must be transmitted between the object and its clients.

The computers may run different operating systems, therefore we need to deal with different operations to send and receive messages or to express invocations. Thus at the Java/C++ level a common operation would be used which will be translated to the particular operation according to the operating system it runs on.

We have two different programming languages C++ and Java, they use different representations for data structures such as strings, arrays, records. A common standard will be defined for each type of data structure that must be transmitted between the object and its clients and a way of translating between that data structure and each of the languages.

We may have different implementors, e.g. one for C++ and the other for Java. They will need to agree on the common standards mentioned above and to document them.

1.8 What is client-server computing? Which of these roles is an active role, and which is a passive one? Explain remote invocation in this context.

1.8 Ans.

Client server computing refers to a running program (a process) on a networked computer that accepts requests from programs running on other computers to perform a service and responds appropriately.

Clients are active (making requests) and servers are passive.

The interaction between a client and a server, from the point when the client sends its request to when it receives the server's response, is called remote invocation.

1.9 Suppose that the operations of the BLOB object are separated into two categories – public operations that are available to all users and protected operations that are available only to certain named users. State all of the problems involved in ensuring that only the named users can use a protected operation. Supposing that access to a protected operation provides information that should not be revealed to all users, what further problems arise?

1.9 Ans.

Each request to access a protected operation must include the identity of the user making the request. The problems are:

- defining the identities of the users. Using these identities in the list of users who are allowed to access the protected operations at the implementation of the BLOB object. And in the request messages.
- ensuring that the identity supplied comes from the user it purports to be and not some other user pretending to be that user.
- preventing other users from replaying or tampering with the request messages of legitimate users.

Further problems.

- the information returned as the result of a protected operation must be hidden from unauthorised users. This means that the messages containing the information must be encrypted in case they are intercepted by unauthorised users.

1.10 The INFO service manages a potentially very large set of resources, each of which can be accessed by users throughout the Internet by means of a key (a string name). Discuss an approach to the design of the names of the resources that achieves the minimum loss of performance as the number of resources in the service increases. Suggest how the INFO service can be implemented so as to avoid performance bottlenecks when the number of users becomes very large.

1.10 Ans.

Algorithms that use hierarchic structures scale better than those that use linear structures. Therefore the solution should suggest a hierarchic naming scheme. e.g. that each resource has an name of the form 'A.B.C' etc. where the time taken is O(log n) where there are n resources in the system.

To allow for large numbers of users, the resources are partitioned amongst several servers, e.g. names starting with A at server 1, with B at server 2 and so forth. There could be more than one level of partitioning as in DNS. To avoid performance bottlenecks the algorithm for looking up a name must be decentralised. That is, the same server must not be involved in looking up every name. (A centralised solution would use a single root server that holds a location database that maps parts of the information onto particular servers). Some replication is required to avoid such centralisation. For example: i) the location database might be replicated at multiple root servers or ii) the location database might be replicated in every server. In both cases, different clients must access different servers (e.g. local ones or randomly).

1.11 A distributed system is described as scalable when it remains effective when there is a significant increase in the number of resources and the number of users. However, these systems sometimes face performance bottlenecks. How can these be avoided?

1.11 Ans.

Caching and replication may be used to improve the performance of the resources that are very heavily used in a distributed system.

1.12 A server process maintains a shared information object such as the BLOB object of Exercise 1.7. Give arguments for and against allowing the client requests to be executed concurrently by the server. In the case that they are executed concurrently, give an example of possible 'interference' that can occur between the operations of different clients. Suggest how such interference may be prevented.

1.12 Ans.

For concurrent executions - more throughput in the server (particularly if the server has to access a disk or another service)

Against - problems of interference between concurrent operations

Example:

Client A's thread reads value of variable X

Client B's thread reads value of variable X

Client A's thread adds 1 to its value and stores the result in X

Client B's thread subtracts 1 from its value and stores the result in X

Result: X := X-1; imagine that X is the balance of a bank account, and clients A and B are implementing credit and debit transactions, and you can see immediately that the result is incorrect.

To overcome interference use some form of concurrency control. For example, for a Java server use synchronized operations such as credit and debit.

1.13 The ANSA Reference Manual [ANSA 1989] identified eight forms of transparency in a distributed system. Which are the two most important transparencies among these?

1.13 Ans.

The two most important transparencies are access and location transparency; their presence or absence most strongly affects the utilization of distributed resources. They are sometimes referred to together as network transparency.

1.14 One of the main standard technological components of web is the HyperText Transfer Protocol (HTTP), which defines the ways in which browsers and other types of clients interact with web servers. What are the different features of the HTTP?

1.14 Ans.

The main features of HTTP are:

a. Request-reply interactions

b. Content types

c. One resource per request

d. Simple access control

1.15 What are the two main functions of an HTTP URL? Explain its general form. Identify the server DNS name, path name, query, and fragment for the URL http://www.google.com/search?q=sabretooth.

1.15 Ans.

An HTTP URL has two main jobs to do: to identify which web server maintains the resource, and to identify which of the resources at that server is required.

In general, HTTP URLs are of the following form:

http:// servername [:port] [/pathName] [?query][#fragment]

where items in square brackets are optional. A full HTTP URL always begins with the string 'http://' followed by a server name, expressed as a Domain Name System (DNS) name. The server's DNS name is optionally followed by the number of the 'port' on which the server listens for requests.

Then comes an optional path name of the server's resource. If this is absent then the server's default web page is required. Finally, the URL optionally ends in a query component – for example, when a user submits the entries in a form such as a search engine's query page – and/or a fragment identifier, which identifies a component of the resource.

In the URL http://www.google.com/search?q=sabretooth

The Server DNS name is www.google.com

The Path name is search

The Query is q= sabretooth

There is no Fragment.

