

The Solutions to Tutorial Questions and Lab Projects of Week 1

Tutorial Questions

1. Give five types of hardware resource and five types of data or software resource that can usefully be shared. Give examples of their sharing as it occurs in distributed systems.

Answer

Hardware:

CPU: compute server (executes processor-intensive applications for clients), remote object server (executes methods on behalf of clients), worm program (shares cpu capacity of desktop machine with the local user). Most other servers, such as file servers, do some computation for their clients, hence their cpu is a shared resource.

memory: cache server (holds recently-accessed web pages in its RAM, for faster access by other local computers).

disk: file server, virtual disk server (see Chapter 8), video on demand server (see Chapter 15).

screen: network window systems, such as X-11, allow processes in remote computers to update the content of windows.

printer: networked printers accept print jobs from many computers, managing them with a queuing system.

network capacity: packet transmission enables many simultaneous communication channels (streams of data) to be transmitted on the same circuits.

Data/software:

web page: web servers enable multiple clients to share read-only page content (usually stored in a file, but sometimes generated on-the-fly).

file: file servers enable multiple clients to share read-write files. Conflicting updates may result in inconsistent results. Most useful for files that change infrequently, such as software binaries.

object: possibilities for software objects are limitless. E.g. shared whiteboard, shared diary, room booking system, etc.

database: databases are intended to record the definitive state of some related sets of data. They have been shared ever since multi-user computers appeared. They include techniques to manage concurrent updates.

newsgroup content: the *netnews* system makes read-only copies of the recently-posted news items available to clients throughout the Internet. A copy of newsgroup content is maintained at each netnews server that is an approximate replica of those at other servers. Each server makes its data available to multiple clients.

video/audio stream: servers can store entire videos on disk and deliver them at playback speed to multiple clients simultaneously.

exclusive lock: a system-level object provided by a lock server, enabling several clients to coordinate their use of a resource (such as printer that does not include a queuing scheme).

2. 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?

Answer

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.

3. What are the advantages and disadvantages of HTML, URLs and HTTP as core technologies for information browsing? Are any of these technologies suitable as a basis for client-server computing in general?

Answer

HTML is a relatively straightforward language to parse and render but it confuses presentation with the underlying data that is being presented.

URLs are efficient resource locators but they are not sufficiently rich as resource links. For example, they may point at a resource that has been relocated or destroyed; their granularity (a whole resource) is too coarse-grained for many purposes.

HTTP is a simple protocol that can be implemented with a small footprint, and which can be put to use in many types of content transfer and other types of service. Its verbosity (HTML messages tend to contain many strings) makes it inefficient for passing small amounts of data.

HTTP and URLs are acceptable as a basis for client-server computing except that (a) there is no strong type-checking (web services operate by-value type checking without compiler support), (b) there is the inefficiency that we have mentioned.

4. A search engine is a web server that responds to client requests to search in its stored indexes and (concurrently) runs several web crawler tasks to build and update the indexes. What are the requirements for synchronisation between these concurrent activities?

Answer

The crawler tasks could build partial indexes to new pages incrementally, then merge them with the active index (including deleting invalid references). This merging operation could be done on an off-line copy. Finally, the environment for processing client requests is changed to access the new index. The latter might need some concurrency control, but in principle it is just a change to one reference to the index which should be atomic.

5. The host computers used in peer-to-peer systems are often simply desktop computers in users' offices or homes. What are the implications of this for the availability and security of

any shared data objects that they hold and to what extent can any weaknesses be overcome through the use of replication?

Answer

Problems:

- People often turn their desktop computers off when not using them. Even if on most of the time, they will be off when user is away for an extended time or the computer is being moved.
- The owners of participating computers are unlikely to be known to other participants, so their trustworthiness is unknown. With current hardware and operating systems the owner of a computer has total control over the data on it and may change it or delete it at will.
- Network connections to the peer computers are exposed to attack (including denial of service).

The importance of these problems depends on the application. For the music downloading that was the original driving force for peer-to-peer it isn't very important. Users can wait until the relevant host is running to access a particular piece of music. There is little motivation for users to tamper with the music. But for more conventional applications such as file storage availability and integrity are all important.

Solutions:

Replication:

- If data replicas are sufficiently widespread and numerous, the probability that all are unavailable simultaneously can be reduced to a negligible level.
- One method for ensuring the integrity of data objects stored at multiple hosts (against tampering or accidental error) is to perform an algorithm to establish a consensus about the value of the data (e.g. by exchanging hashes of the object's value and comparing them). This is discussed in Chapter 15. But there is a simpler solution for objects whose value doesn't change (e.g. media files such as music, photographs, radio broadcasts or films).

Secure hash identifiers:

- The object's identifier is derived from its hash code. The identifier is used to address the object. When the object is received by a client, the hash code can be checked for correspondence with the identifier. The hash algorithms used must obey the properties required of a secure hash algorithm as described in Chapter 7.

6. Distinguish between buffering and caching.

Answer

Buffering: a technique for storing data transmitted from a sending process to a receiving process in local memory or secondary (disk) storage until the receiving process is ready to consume it. For example, when reading data from a file or transmitting messages through a network, it is beneficial to handle it in large blocks. The blocks are held in buffer storage in the receiving process' memory space. The buffer is released when the data has been consumed by the process.

Caching: a technique for optimizing access to remote data objects by holding a copy of them in local memory or secondary (disk) storage. Accesses to parts of the remote object are translated into accesses to the corresponding parts of the local copy. Unlike buffering, the local copy may be retained as long as there is local memory available to hold it. A cache management algorithm and a release strategy are needed to manage the use of the memory allocated to the cache. (If we interpret the word 'remote' in the sense of 'further from the processor', then this definition is valid not only for client caches in distributed systems but also for disk block caches in operating systems and processor caches in cpu chips.)

7. Describe possible occurrences of each of the main types of security threat (threats to processes, threats to communication channels, denial of service) that might occur in the Internet.

Answer

Threats to processes: without authentication of principals and servers, many threats exist. An enemy could access other user's files or mailboxes, or set up 'spoof' servers. E.g. a server could be set up to 'spoof' a bank's service and receive details of user's financial transactions.

Threats to communication channels: IP spoofing - sending requests to servers with a false source address, man-in-the-middle attacks.

Denial of service: flooding a publicly-available service with irrelevant messages.

Lab Projects

Task 1

1. Download the `WebClient1.java` from Week 1 folder of the course Moodle site.
2. Compile and run the Java program.
3. What protocols have been used for the communication between `WebClient1` and the server?

Answer

HTTPS

4. What has been downloaded by `WebClient1`? You should answer this question by reading the source of `WebClient1` and comparing its output (`page1.txt`) with the page source of <https://www.oracle.com/>.

Answer

The home page of www.oracle.com.

By checking the contents of `page1.txt` and comparing with the contents of the page (IE: View→Source; Firefox: View→Page Source), the program downloaded the page and wrote it into file `page1.txt`.

Task 2

1. Download the `WebClient2.java` from Week 1 block of the course Moodle site
2. Compile and run the Java program.
3. What protocols have been used for the communication between `WebClient2` and the server?

Answer

HTTPS

4. What have been output from `WebClient2`? Answer this question by reading the code of `WebClient2` and checking the properties of the web page.

Answer

The program output information of digital certificates from the server. The digital certificates can be checked by IE: 'right click' → Properties → certificate or Firefox: 'right click' → View Page Info → Security → View Certificate.

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder