

ITEE



What's New:

You should aim to catch up on previous tutorials; the last two will not introduce as much new material as before.

Learning objectives for this week:

- Understand the major concepts we've covered, including:
 - different types of communication
 - different approaches to distributing data and processing
 - networked file systems
- Apply these concepts to reasoning about distributed operating systems
- Apply these concepts to evaluating alternatives
- Understand issues in improving performance

As before, you should aim to answer all of these questions by the end of the course. Being able to answer them by the end of week 12 is a useful goal.

1. Concepts

- a. When you use the world-wide web, are you using a distributed system, or a networked system? Do you see any difference between accessing a local versus a remote file?
- b. In a UNIX lab, when you access your home directory, are you using a purely local system, a networked system or a distributed file system?
- c. It's common to use a point-to-point link in long-haul networks over very long distances, like across a big ocean. Why should this be the case?
- d. What are the advantages of stateless file servers?

2. Distributed Processing

- a. Consider each of the following cases and indicate in which cases you would consider data or process migration (neither or both may also be valid answers):
 - i. your application is small and rarely makes disk accesses, which are always small. It is running on a heavily loaded server, but could equally well run on your machine, but the files are on the server
 - ii. your application is very large and rarely makes disk accesses, which are always small. It is running on a heavily loaded server, but could equally well run on your machine, but the files are on the server
 - iii. your application is very large and makes frequent large disk accesses. It is running on a heavily loaded server, but could equally well run on your machine, but the files are on the server
- b. It takes 10ms for any network transaction, plus the transfer time. The transfer time is size of data to move / networks speed. In each of the following, calculate the time saved (or extra) resulting from migrating the process:
 - i. size of process: 1Mbyte; time to run on busy server: 1s; time to run on client: 0.1s; 100Mbit/s network
 - ii. size of process: 1Mbyte; time to run on busy server: 1s; time to run on client: 0.1s; 1000Mbit/s network
 - iii. size of process: 10Mbytes; time to run on busy server: 1s; time to run on client: 0.1s; 100 Mbit/s network
 - iv. size of process: 10Mbytes; time to run on busy server: 60s; time to run on client: 0.5s; 100Mbit/s network
 - v. size of process: 10Mbytes; time to run on busy server: 60s; time to run on client: 0.5s; 1000Mbit/s network
- c. What can you say in general about process migration, based on (b), and any other good points you can think of?
- d. Why, in general terms, is it hard to handle failures in distributed systems?

3. Distributed File Systems

- a. Which of the following file operations assume that a server must maintain client state?
 - i. open or close a file
 - ii. allow a client to cache a file
 - iii. allow a client to lock a file
 - iv. allow a client to read a file, starting from a specified location
 - v. allow a client to read a file, starting from where it left off
- b. In general terms, explain why stateful file servers are becoming the norm, despite the theoretical advantages of stateless servers.
- c. Does NFS provide location transparency or location independence? Explain.
- d. What kind of file replication is used in the world-wide web? Consider implementations of clients, servers, and other performance-enhancing services.