**Goals of Distributed Systems**

. Transparency

. Openness

. Reliability

. Performance

. Scalability

Transparency • How to achieve the single-system image, i.e., how to make a collection of computers appear as a single computer. • Hiding all the distribution from the users as well as the application programs can be achieved at two levels: 1) hide the distribution from users 2) at a lower level, make the system look transparent to programs. 1) and 2) requires uniform interfaces such as access to files, communication.

Openness • Make it easier to build and change • Monolithic Kernel: systems calls are trapped and executed by the kernel. All system calls are served by the kernel, e.g., UNIX. • Microkernel: provides minimal services. • IPC • some memory management • some low-level process management and scheduling • low-level i/o (E.g., Mach can support multiple file systems, multiple system interfaces.) • Standard interface, separation of policy from mechanism

Reliability • Distributed system should be more reliable than single system. – Availability: fraction of time the system is usable. Redundancy improves it. – Need to maintain consistency – Need to be secure – Fault tolerance: need to mask failures, recover from errors. • Example: 3 machines with .95 probability of being up

Performance • Without gain on this, why bother with distributed systems. • Performance loss due to communication delays: – fine-grain parallelism: high degree of interaction – coarse-grain parallelism • Performance loss due to making the system fault tolerant.

Scalability • Systems grow with time or become obsolete. • Techniques that require resources linearly in terms of the size of the system are not scalable. (e.g., broadcast based query won't work for large distributed systems.) • Examples of bottlenecks (i.e., scalability limitations) o Centralized components: a single mail server o Centralized tables/data: a single URL address book o Centralized algorithms: routing based on complete information

**Advantages of Distributed Systems**

All the nodes in the distributed system are connected to each other. So nodes can easily share data with other nodes.

More nodes can easily be added to the distributed system i.e. it can be scaled as required.

Failure of one node does not lead to the failure of the entire distributed system. Other nodes can still communicate with each other.

Resources like printers can be shared with multiple nodes rather than being restricted to just one.