Class presentations Friday, December 8

Pandey, Toki, and Varshney 2:02-2:15

Distributed Construction of Navigation View for Radiology Workstation

Radiology Workstation, currently under development by the Medical Image Processing group of this department, is an electronic image viewing system that will provide the radiologists with quick, reliable and inexpensive accesses to patients' images.

Some of the advantages of the Radiology Workstation over conventional image viewing systems arise from its use of CRT screens instead of light boxes as the display devices. However, severly limited space available on a CRT screen makes it impossible to display more than a few images at a time. This then presents a problem of how to provide a radiologist with a convenient method of selecting particular images to be displayed.

At this time, such a method is provided by a "Navigation View," an image that can be displayed in its entirety within a CRT screen and contains miniatures of all the images of a patient currently under examination. In essence, this "Navigation View" is a map of all the images of that patient. A radiologist simply chooses a set of images from this map and the chosen images will be shown at their normal sizes on the CRT screen.

A prototype program which creates such a "Navigation View" has been developed over the past summer. However, due to the large amount of data involved (to digitally represent images), the program takes relatively long time to run. For example, even an unrealistically small test case takes approximately an hour to complete – too long to make it practical to create a "Navigation View" interactively.

The aim of our project is to distribute the process of creating such a "Navigation View" over a number of SUN-3s. This seems to be a natural extension to the current prototype program since most of operations involved (like shrinking images so that they can all fit in one screen) can be performed in parallel.

Brown, McCabe, and Katz 2:15-2:28

A Distributed Animation System: Wanda and Friends

In UNC's current virtual world research, we use the same processor to simulate the entire virtual world and maintain the information to display the world. This configuration's performance degrades significantly when there are many actors in the virtual world because all the actors and the display contend for the same CPU.

We propose to alleviate this contention by distributing the virtual world computations to many processors. We use a single processor to communicate with the Pixel-Planes display engine and a separate processor for each of the actors. These components communicate via a database server which maintains the full state of the virtual world; this server runs on yet another processor. By splitting our computations among several processors, we guarantee that each of our processes has sufficient compute resources to execute in real time, and that system performance is not significantly dependent on the number of actors in the virtual world.

Ramsisaria 2:28-2:34

NFS Statistic Collector

The aim of this project is to develop an online monitor for the file servers on the departmental network. And use this to characterize the NFS load put on the servers in terms of the RPC calls made to these servers.

Wang 2:34-2:40

Distributed Othello game

The project is an implementation of a distributed version of Othello game. Pairs of players can play their games on different machines. A central manager is responsible for keeping information about players and arranges the handshakes of the pairs of players. One important feature of the implementation is its tolerance of the failure or crash of one of the two players. Recovering players contact the central manager for the information of their opponents and connect with them, so that they can continue to play from where they were interrupted by the failure. The communication is based on TCP/IP because reliability is important.

Kupper, Peller, and Whaley 2:40-2:53

Designing a Macintosh/UNIX email gateway

The growth of the Internet has empowered those in research, government and industry to communicate rapidly through electronic mail systems. The Apple Macintosh family of computers has become more popular in the diverse Internet environment. Ethernet cards and Ethernet/AppleTalk gateways allow Macintosh systems to communicate with conventional UNIX and MVS hosts. Until recently a Macintosh could only talk to UNIX computers via applications such as NCSA Telnet. Users read mail through a vt100 emulation window. Apple recently released a TCP/IP implementation for the Macintosh that enables the Macintosh to speak, listen for and understand TCP/IP. We use the TCP/IP routines as a base for a Macintosh mail agent that can deliver and receive mail. The issues surrounding the design implementation of a reliable delivery system are explored. Our implementation takes advantage of a client/server model mail system, in which a UNIX host acts as a spool center for Macintosh clients.