Class presentations

Monday, December 4

Hamilton, Harris, and Wagner 2:02-2:15

Sophisticated talk

Our project is to have a more sophisticated version of `talk` working. This program will use UDP and will allow unlimited participants.

The user at a terminal will issue a `conf` *subject* comand, and the interface will pop-up and place a message in the lower window informing the user who was alerted about the startup of a conference on the subject. If you are on a machine and receive an beep alerting you as to the startup of a conference subject, you may type `conf` *subject* to join the session.

The X interface has two windows, one for composing and another for receiving. Four X buttons are available for manipulating the user interface.

Chatterjee, Patel, Rajanala, and Wegner 2:15-2:28

Distributed Adaptive Histogram Equalization

Adaptive histogram equalization (AHE) is a contrast enhancement method for both naturally visible and invisible (e.g. x-ray image) images. The method involves applying to each pixel the histogram equalization mapping based on the intensity proportional to its rank in the pixels surrounding it. The algorithm is inherently parallel. The parallelism is exploited by dividing the image into a number of regions and distributing the computation of histogram equalization mapping per region to the processes running on different machines. The project thus attempts to speed up computation by adding more resources.

Fredericksen, Grove, and Kiel 2:28-2:41

A distributed spanning tree algorithm simulator

Each bridge is represented by a separate process and performs the functions required to calculate the spanning tree in concert with all other reachable bridges. Each bridge can connect to two or more LANs, each of which is also a separate process. Bridges communicate by sending and receiving ethernet BPDUs through connected ports. Each LAN forwards all transmitted ethernet packets to all hosts connected to it. The entire process is presided over by the program `bigbrother`. `Bigbrother` knows how to create new LANs and bridges, connect bridges to LANs, and to order LANs and bridges to change parameters, for example, to go up, down or to pause. In addition, `bigbrother` has a lackey named `poller`. The poller, also a separate process, polls the bridges for their current state so that the dynamics of calculation of the spanning tree can be observed.

The calculated spanning tree network topology is displayed using the X11 Window System. We have built a graphics display server (yes, another separate process ...) that keeps a display list of LAN and bridge objects, as well as active bridge-lan connections.

Blakeley, Bollella, and Ruegger 2:41-2:55

Go Fish

We are attempting to exemplify a possibly new server characteristic we call transparency. The test application we chose is a card game called Go Fish. A brief description of the game is as follows. Each player is given seven cards. The players each have a chance to query another player as to whether the queried player has a particular card. The queried player responds either yes, in which case the querying player is given the card in question, or no, in which case the querying player is given a card from the cards remaining in the deck. At any time any player may remove from her hand two cards that are equal in magnitude. The game ends when either, (1) a player has no cards or (2) when the deck is exhausted. The player with the fewest cards wins.

We have constructed the server as transparently as possible. We wish players to be able to play as though they were all sitting around a table playing with actual cards. To this end we have incorporated the following abilities in our server. Players are allowed to peek at another player's cards. Players may lie to the server and each other. Players may send covert messages to other players. Players may join a game in progress and in this case they are given the status of a kibitzer. A kibitzer has at his disposal most of the commands of the true players and in addition sees the cards of many of the players without explicitly asking.