

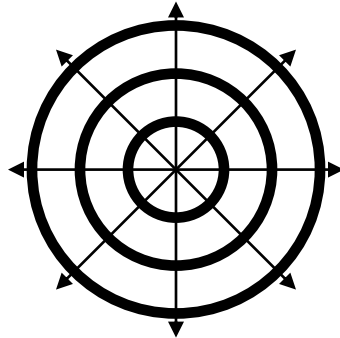
EECS 211 Spring 2010 Description of Major Project for the Term

The remaining program assignments will build an application which simulates a distributed file management system. This document describes the kinds of objects we will simulate and the general organization of the project. The objects we will implement in this project are very simplified versions of a small subset of the kinds of objects that are actually used in the internet and in distributed file systems. A sample of the commands our project will eventually be able to process is given at the end of this document.

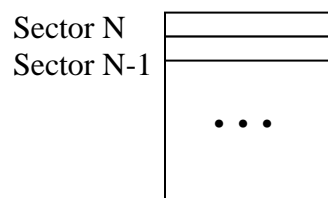
Background and Basic Objects in our Project

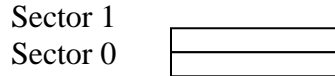
Files are stored on disk drives inside computers. In this project we will model (i.e., design classes for) disk drives and computers. Moreover, there are different kinds of computers in the real world – laptops, desktops, large storage processors, service processors like printers, etc.

A real disk drive has its storage medium divided into blocks. (The physical layout of a disk is a series of concentric circles, called tracks, and a series of radiating boundaries dividing the platter into pie-shaped sections. The result is a set of arc-shaped active areas, each of which can store a fixed number (usually 256) of bytes.)



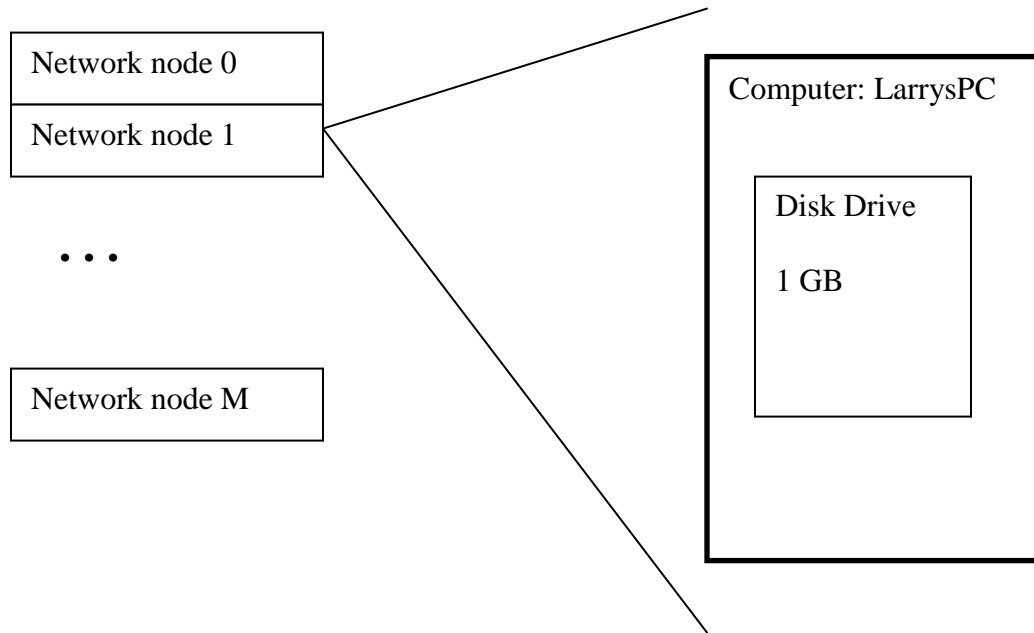
The sectors are numbered from 0 to N, where N is determined by the capacity of the drive. On formatted disk drives, usually the first sectors contain system information such as boot code (code that is automatically loaded when you restart the computer) and the root directory. In our project we will simulate this organization by providing a large amount of storage which is accessed in small chunks. We will simulate a file directory in the first sectors of our “soft” disk drive, and files themselves will be stored in a sequence of the small chunks.





A computer contains a disk drive, and of course different computers have disk drives of different sizes. Moreover, some computers are designed to be used in certain ways, even though most computers can do general computing as well. We will use the inheritance mechanism of C++ to model a general computing class, with three special kinds of computers (PC, file server, and printer server), each with its own disk drive and it's own set of special operations. For example, PC machines will be able to create new text files, while printer machines will be able to print files.

A distributed system consists of a set of computers which can send messages to each other. We will create a class corresponding to messages. We will create a small computer network by defining an array of computer machines. Each computer will have a name, and the "system" will be able to look up the location of a computer given its name. Any computer will be able to send a file to any other computer.



Tentative Sequence of Programs

We will develop our simulator in a sequence on steps. Here is the tentative sequence.

- Program 3 – the class diskDrive. We begin by developing a class that represents a disk drive. A diskDrive object will contain storage for the data blocks and a map that we will eventually use to tell which data blocks currently have data (i.e., contain parts of actual files).
- Program 4 – token parsing and recognition and display of errors. In this assignment we will write the function that reads text from a file and recognizes the next token (sequence of characters from the input file that belong together).

In later assignments this function will be called repeatedly to parse the tokens in the input file so that our project can interpret commands such as "create a new computer" or "send a file from one computer to another". We will also introduce the notion of checking for errors and displaying relevant error messages. The main C++ concepts are the processing of characters into strings and the dynamic allocation and freeing of memory.

- Program 5 – main switch. This assignment will create a main switch with a case for each command that our project will be able to simulate. When this assignment is completed, your project will be able to recognize every command that will be used in the project and branch to a corresponding case. Later assignments, then, will fill in those cases with the functions needed to actually process those commands. The main C++ concepts are the switch statement and arrays of strings.
- Program 6 – the class computer, network of computers, derived classes PC, fileServer, and printer.
- Program 7 – files on a disk drive.
- Program 8 – inputting a file to the disk drive of a computer.

Sample input file for last assignment:

```
add_network_node PC pc1 4096 Larry
add_network_node printer pr1 2048 5
add_network_node server s1 8192 TECH
add_network_node PC pc2 384 Julia
system_status
create_file pc1 f1 20
123456789
abcdefghi
create_file pc1 f2 70
111111111
222222222
333333333
444444444
555555555
666666666
777777777
create_file pc1 f3 70
aaaaaaaaa
bbbbbbbbb
cccccccc
dddddddd
eeeeeeee
fffffffff
ggggggggg
ls pc1
transfer_file pc1 pr1 f1
transfer_file pc1 pr1 f3
ls pr1
print_files pr1 f1 f3 f4
delete_files pr1 f1 f2 f3
```

```
ls pr1
transfer_file pc1 pc2 f1
transfer_file pc1 pc2 f2
transfer_file pc1 pc2 f3
transfer_file pc1 pr1 f4
ls pc2
ls pr1
system_status
halt
```