Option pricing is one of the important problems in finance that demands efficient algorithms to produce accurate and fast results. In this thesis, we develop a framework for evaluating option price in a parallel and distributed environment, known as FLEET. FLEET refers to number of processors working together under one command. The focus of the work is two fold: (i) developing a client-server environment for investors where (a) a client requests for the information over the network to the server, (b) the server provides the solution as fast as possible to the client; (ii) developing a multithreaded algorithm to facilitate pricing. We assume that the clients are in a distributed heterogeneous network. The underlying framework for the client-server architecture is Common Object Request Broker Architecture (CORBA). The information requested by an investor from the server is the price of an option that is computed using the Black-Scholes (B-S) model. We have used the explicit Forward-Time Central-Space (FTCS) finite-difference scheme to solve the B-S equation and developed a multithreaded algorithm to evaluate the option price. The algorithm is implemented on an eight-node symmetric multiprocessor machine using Java OpenMP (JOMP). We have also implemented a distributed database containing the current information on the underlying asset of the option, which helps in making informed decisions. We have compared and analyzed the performance results using different scheduling technique and have achieved a speedup of about four on eight nodes.