P-2695

EFFICIENT APPROXIMATE QUERY PROCESSING IN P2P NETWORK

By

S.ASHA
Reg. No. 71206621006

Of

KUMARAGURU COLLEGE OF TECHNOLOGY, COIMBATORE

A PROJECT REPORT

Submitted to the

FACULTY OF INFORMATION AND COMMUNICATION ENGINEERING

In partial fulfillment of the requirements
For the award of the degree

Of

MASTER OF COMPUTER APPLICATION

July, 2009

BONAFIDE CERTIFICATE

KUMARAGURU COLLEGE OF TECHNOLOGY

COIMBATORE - 641006

BONAFIDE CERTIFICATE

Certified that this project report titled **"EFFICIENT APPROXIMATE QUERY PROCESSING IN P2P NETWORK"** is the bonafide work of **Ms. S.Asha** (Reg.No: **71206621006**) who carried out the research under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

29/6/09
Supervisor

Head of the Department

Submitted to Project and Viva Examination held on 06/07/2009

Internal Examiner

6/7/09
External Examiner

MEZOBLANCA
SOLUTIONS

20/05/2009

TO WHOMSOEVER IT MAY CONCERN

This is to certify that **Ms.S.ASHA (Reg.No:71206621006),** Final year **MCA** student of **KUMARAGURU COLLEGE OF TECHNOLOGY, Coimbatore** has undergone her project work titled **"EFFICIENT APPROXIMATE QUERY PROCESSING IN P2P NETWORK"** in our organization during the period from December 2008 to May 2009 and has completed the same successfully.

For Mezoblanca Solutions (India) Pvt. Ltd,

Director.

Mezoblanca Solutions (India) Pvt. Ltd. CBE - 4.

ABSTRACT

ABSTRACT

This project approaches the challenges of decision support and data analysis on P2P databases. The objective of the project can be stated as "Given an aggregate and a desired error bound at a query node peer, compute an approximate answer to the query that satisfies the error bound. Computing a high-quality random sample of the database efficiently in the P2P environment is complicated due to several factors: the data is distributed (usually in uneven quantities) across many peers, within each peer, the data is often highly correlated, and, moreover, even collecting a random sample of the peers is difficult to accomplish. To counter these problems, we have developed an adaptive two-phase sampling approach based on random walks of the P2P graph. In the first phase, we initiate a fixed-length random walk from the query node. This random walk should be long enough to ensure that the visited peers represent a close sample from theunderlying stationary distribution (the appropriate length of such a walk is determined in a preprocessing step). We then retrieve certain information from the visited peers,such as the number of tuples, the aggregate of tuples (for example, SUM, COUNT, AVG, and so forth) that satisfy the selection condition, and send this information back to thequery node. This information is then analyzed at the query node to determine the skewed nature of the data that is distributed across the network, such as the variance of theaggregates of the data at peers, the amount of correlation between tuples that exists within the same peers, the variance in the degrees of individual nodes in the P2Pgraph (recall that the degree has a bearing on the probability that a node will be sampled by the random walk), and so on. Once this data has been analyzed at thequery node, an estimation is made on how much more samples are required (and in what way should these samples be collected) so that the original query can beoptimally answered within the desired accuracy, with high probability.

ACKNOWLEDGEMENT

ACKNOWLEDGEMENT

I wish to express my deep unfathomable feeling of gratitude and indebtedness to **Dr.R.Annamalai**, Vice Principal , Kumaraguru College of Technology, Coimbatore for the successful completion of the project work.

I am very glad to express a special word of thanks to **Dr. M. Gururajan,** Professor and Head of the Department, Kumaraguru College of Technology, Coimbatore for encouraging me to do this work.

I am very much indebted to **Dr.A. Muthukumar,** Assistant Professor and Course Coordinator, Kumaraguru College of Technology, Coimbatore for his complete assistance, guidance and support given to me throughout my project.

I would express heartfelt thanks to my internal guide **Mrs. P. Parameswari**, Senior Lecturer, Kumaraguru College of Technology as without her best guidance ,support,co-operation and valuable suggestion during the course of this project.

It is my pleasure to express my profound gratitude to **Mezoblanca Solutions, Coimbatore**, for admitting me into this project. I am thankful to **Mr. Natarajan** of Mezoblanca Solutions, for his excellent guidance, timely suggestions and constant support in all my endeavors.

TABLE OF CONTENTS

TABLE OF CONTENTS

List of Tables

List of Figures

List of Abbreviations

Acronyms	Full Form
P2P	Peer to Peer
IP	Internet Protocol
TCP	Transfer Control Protocol
QSR	Query Success Rate
SQL	Standard Query Language
OLAP	Online Analytical Processing
CV	Cross Validation
RLE	Random Leader Election

INTRODUCTION

CHAPTER 1

INTRODUCTION

This chapter is organized into two parts. The first part deals with the organization profile. It provides a brief insight into the history of the organization and the products. The second part gives an introduction about the project.

1.1 ORGANIZATION PROFILE

The idea of providing quality software products at a competitive price gave birth to MEZOBLANCA SOLUTIONS India (Pvt) Ltd. MEZOBLANCA works as a part of any organization seeking cost-effective solutions. MEZOBLANCA is blessed with a team of software professionals in order to achieve its mission. MEZOBLANCANS work towards managing successful software delivery.

MEZOBLANCA SOLUTIONS India (Pvt) Ltd is a leading solution provider for software based applications. Established in 2003, The Company has been promoted by some highly experienced Professionals dedicated to provide total IT solutions under one roof.

Mezoblanca team provides top quality software solutions with an innovative view, our highly spirited Mezoblancans focus on meeting the complications of the present business with prodigious ease.

1.1.1. Mission

To help individuals enhance their creativity and build Knowledge that creates opportunities, thus deliver a result that maximizes customer's return on their existing investments, with Affordable cost High quality products in an Effective

1.1.2. Vision

To evolve as leading software Services Company by using the available resource and to provide competitive products that are best fit to latest technological innovations.

1.1.3. Products

- Trading, Accounts, Production process Soft wares
- Networking Products
- Medical Soft Solutions
- Online Training Tools
- Project For Industry, Gear Industry, Corporate sectors, Bearing Industry, Textile Industry , Educational Institutions

1.1.4. Services

- Software Development
- Project consulting
- Training

1.2 SYSTEM OVERVIEW

The project titled "Efficient Approximate Query Processing in P2P Network" offers total solution to the indefinite delay in aggregate query processing in a P2P network.

The need for this system arose when there was large network latency due to the imprecise time taken for scanning the distributed database. In a P2P environment computing a premium arbitrary sample of the database proficiently is complicated due to several factors: the data is distributed (usually in uneven quantities) across many peers, within each peer, the data is often highly correlated, and, moreover, even

collecting a random sample of the peers is difficult to accomplish. The system helps to increase the query success rate using two phase Random Walk algorithm.

The major functions of the system are

1. Graphical user interface creation and peer to peer network establishment.
2. Implementation of random walk algorithm-phase I.
3. Evaluation of base result from the algorithm.
4. Implementation of random walk algorithm-phase II.

SYSTEM STUDY

CHAPTER 2

SYSTEM STUDY

2.1 EXISTING SYSTEM

A P2P network consists of numerous peer nodes that share data and resources with other peers on an equal basis. Unlike traditional client-server models, no central coordination exists in a P2P system; thus, there is no central point of failure. P2P networks are scalable, fault tolerant, and dynamic, and nodes can join and depart the network with ease. The Traditional decision support techniques such as online analytical processing (OLAP) are commonly uses the aggregation queries at runtime from scratch by crawling and scanning the entire P2P network. The query processing across the vast P2P network is done by flooding the distributed database across the network with the query with aggregated function. This results in the indefinite delay in the time of the getting response for the query.

2.1.1 Drawbacks of the Existing System

The drawbacks of the existing systems can be summarized as below:

- The query processing involves crawling and scanning the entire P2P network and also decreases the QSR (Query Success Rate).
- This causes indefinite delay in getting the response
- The large network latency that result due to the indefinite time taken for scanning the distributed database
- The uncertainty over the result set obtaining cause huge doubt in the minds of the users

- No guarantee that the query has scanned all the records in all the peers in the network.

2.2 PROPOSED SYSTEM

The proposed system uses an Adaptive two-phase sampling approach based on random walks of the P2P graph. The random walk is implemented in two phases. In this, a small random sample of the rows of the database is drawn, the query is executed on this small sample, and the results are extrapolated to the whole database

2.2.1 Advantages of the Proposed System

The expected benefits of the Proposed System are as follows:

- It improves the search efficiency and reduces unnecessary traffic in the P2P network
- This approach helps to get the query result efficiently with limited latency.
- It has a check on the accuracy of the result of the query.
- It requires no additional message to be sent to the query node
- It gives performance speedup, with no additional time requirement for processing beyond the time required by the random walk phase.

2.3 FEASIBILITY ANALYSIS

Feasibility analysis is the measure of how beneficial or practical the development of the System will be to the project. Once the problem is explained information is gathered about the system to test whether the system is viable Technically, Financially and Operationally. Thus, feasibility study is carried out in three phases as follows.

The Key considerations involved in the feasibility analysis are

- Technical
- Operational
- Economic

2.3.1. Technical Feasibility

Technical Feasibility is the measure of practicality of a specific technical solution and the availability of technical resources and expertise. It centers on the existing computer system (hardware, software, etc.) and to what extent it can support the new addition. The level of technology is determined by factors such as the software tools available, the machine environment ,platform etc.Since the resources required for the development of the project are already available in the organization, the project is technically feasible.

The proposed system is to be developed using VB.Net and SQL server 2000 which are some of the leading technologies in the market. The technology resources are easily available and this product works effectively even for huge amount of data. Visual Studio.NET 2005 and SQL server 2000 are already available with the company. These technologies work well on Microsoft platform.

2.3.2 Operational Feasibility

Operational Feasibility asks if the system will work when it is developed and installed. It checks for the support of the management, the current business methods, user's involvement and their attitude towards the proposed system, etc.

The proposed system would be beneficial to Mezoblanca solutions satisfying the objectives when developed and installed. Also since this project reduces the latency in retrieving the data they are very much in favour of implementing the system.

2.3.3 Economic Feasibility

Economic Feasibility is the measure of the cost-effectiveness of the proposed system. The investment to be made in the proposed system must prove a good investment to the project by returning benefits equal to or exceeding the costs incurred in developing the system.

The proposed The System has been designed to work for dynamic database and it manages voluminous data. Since the effort to develop the product was found to be feasible, the development presents a good investment for the organization. Hence the above system is economically feasible.

DEVELOPMENT ENVIRONMENT

CHAPTER 3

DEVELOPMENT ENVIRONMENT

3.1 HARDWARE REQUIREMENTS

The hardware support required for deploying the application

PROCESSOR:	Pentium IV
RAM:	256 MB
HARDDISK:	Seagate 80 GB
KEYBOARD:	Logitech 104 Keys

3.2 SOFTWARE REQUIREMENTS

The software support required for deployment is:

Operating System:	Windows XP
Database:	SQL Server 2000
Software for development:	Visual Studio 2005

3.3 PROGRAMMING ENVIRONMENT

3.3.1. Visual Basic.Net

- Visual Basic . Net has flexibility , allowing one or more language to interoperate to provide the solution. This Cross Language Compatibility allows to do project at faster rate.

- Visual Basic . Net has Common Language Runtime , that allows all the component to converge into one intermediate format and then can interact.
- Visual Basic . Net has provide excellent security when your application is executed in the system
- Visual Basic .Net has flexibility, allowing us to configure the working environment to best suit our individual style. We can choose between a single and multiple document interfaces, and we can adjust the size and positioning of the various IDE elements.
- Visual Basic . Net has Intelligence feature that make the coding easy and also Dynamic help provides very less coding time.
- The working environment in Visual Basic .Net is often referred to as Integrated Development Environment because it integrates many different functions such as design, editing, compiling and debugging within a common environment. In most traditional development tools, each of separate program, each with its own interface.
- The Visual Basic .Net language is quite powerful – if we can imagine a programming task and accomplished using Visual Basic .Net.
- After creating a Visual Basic . Net application, if we want to distribute it to others we can freely distribute any application to anyone who uses Microsoft windows. We can distribute our applications on disk, on CDs, across networks, or over an intranet or the internet.
- Toolbars provide quick access to commonly used commands in the programming environment. We click a button on the toolbar once to carry out the action represented by that button. By default, the standard toolbar is displayed when we start Visual Basic. Additional toolbars for editing, form design, and debugging can be toggled on or off from the toolbars command on the view menu.
- Many parts of Visual Basic are context sensitive. Context sensitive means we can get help on these parts directly without having to go through the help menu. For example, to get help on any keyword in the Visual Basic

language, place the insertion point on that keyword in the code window and press F1.

- Visual Basic interprets our code as we enter it, catching and highlighting most syntax or spelling errors on the fly. It's almost like having an expert watching over our shoulder as we enter our code.

3.3.2 SQL Server 2000

Microsoft SQL Server 2000 is a full-featured relational database management system (RDBMS) that offers a variety of administrative tools to ease the burdens of database development, maintenance and administration. In this article, we'll cover six of the more frequently used tools: Enterprise Manager, Query Analyzer, SQL Profiler, Service Manager, Data Transformation Services and Books Online.

Enterprise Manager is the main administrative console for SQL Server installations. It provides you with a graphical "birds-eye" view of all of the SQL Server installations on your network. You can perform high-level administrative functions that affect one or more servers, schedule common maintenance tasks or create and modify the structure of individual databases.

Query Analyzer offers a quick and dirty method for performing queries against any of your SQL Server databases. It's a great way to quickly pull information out of a database in response to a user request, test queries before implementing them in other applications, create/modify stored procedures and execute administrative tasks.

SQL Profiler provides a window into the inner workings of your database. You can monitor many different event types and observe database performance in real time. SQL Profiler allows you to capture and replay system "traces" that log various activities. It's a great tool for optimizing databases with performance issues or troubleshooting.

Service Manager is used to control the MS SQL Server (the main SQL

SQLServerAgent processes. An icon for this service normally resides in the system tray of machines running SQL Server. You can use Service Manager to start, stop or pause.

Data Transformation Services (DTS) provide an extremely flexible method for importing and exporting data between a Microsoft SQL Server installation and a large variety of other formats. The most commonly used DTS application is the "Import and Export Data" wizard found in the SQL Server program group

Books Online is an often overlooked resource provided with SQL Server that contains answers to a variety of administrative, development and installation issues. It's a great resource to consult before turning to the Internet or technical support.

SYSTEM DESIGN

CHAPTER 4

SYSTEM DESIGN

4.1 ELEMENTS OF DESIGN

System Design is the most creative and challenging phase in the development of a software system. Design implies to a description of the final system and the process by which it is developed. The first step is to determine what input data is needed for the system and then to design a database that will meet the requirements of the proposed system. The next step is to determine what outputs are needed from the system and the format of the output to be produced.

During the design of the proposed system some areas where attention is required are:

- What are the inputs required and the outputs produced?
- How should the data be organized?
- What will be the processes involved in the system?
- How should the screen look?

The steps carried out in the design phase are as follows:

- Modular Design
- Input Design
- Output Design
- Database Design

4.2. Modular design

It is always difficult for any System Development team to grasp a system without breaking it into several subsystems/modules. These subsystems/modules will be a part of the original system yet they will be independent in the sense that they will incorporate within them the major functionalities of the proposed system.

A software system is always divided into several subsystems/modules which make it easier to develop and perform tests on the whole system. The subsystems are also known as the modules and the process of dividing an entire system into subsystems/modules is known as Decomposition.

The modules identified for the proposed system are as below:

The major functions of the system are

- Graphical user interface creation and peer to peer network establishment.
- Implementation of random walk algorithm-phase I.
- Evaluation of base result from the algorithm.
- Implementation of random walk algorithm-phase II.

Description:

➢ Graphical user interface creation and peer to peer network establishment:

We develop the Graphical User Interface for the easy interaction of any user who process his query in a query node. The data objects in .NET like text area, buttons and navigational tools are implemented in the user interface. Also the networks of peers are created using a common protocol- TCP/IP. The dynamic natures of the connectivity among the heterogenous peers are implemented using threading for connectivity.

- Implementation of random walk algorithm-phase I.

In order to get the base result of the query, the pre-determined numbers of peers are selected and the query search space is determined. The degree of each and every peer is determined and the entire set of edges 'E' is noted. The query is passed to a pre-determined number of peers with a jump size of say 'j', and the tuples in those peers are processed. We get the base result/record set at the end of this module.

- Evaluation of base result from the algorithm.

The accuracy of the base result is analysed by the cross validation procedure. The desired error threshold Δreq is fixed and the base result set value of the query is compared with it. The cross validation(CV) error is estimated. The value of the CV error enables us to determine the need for the phase-II of the random walk. Important pre-defined values we use for this estimation are 'M'-the total number peers in the network, 'E'- the total number of edges in the network and 'm'- the number of peers visited.

- Implementation of random walk algorithm-phase II.

Based upon the result of the first phase of the algorithm, we estimate the total number of peers 'm' to be visited more. In phase II, the number of tuples to be sampled per peer, jump size 'j' and the required accuracy Δreq. The query is processed according to the direction of the analysis we did at the end of the phase-I module. A hybrid sampling algorithm is used to sample the tuples per visited peers. The answers to the query is received at the query node which is substantially enhanced due to the sampling algorithm as recommended by the first phase of random walk.

4.3. Input Design

Input design is a part of the system design and hence must be carefully designed which otherwise lead to serious errors in the later stages of development. Inaccurate will input data is the most common cause of errors in data processing. The main objective of designing input focus on

- Controlling the amount of input required.
- Avoiding delayed responses
- Controlling errors
- Keeping errors
- Keeping process simple
- Avoiding errors

The required input is stored in the form of the tales. They may be numeric or alphanumeric values. The input screens should be user friendly, so that everyone can access the options on it without having knowledge regarding the complete system.

The user defined inputs for this project are as follows:

1. Required Accuracy(Δreq) :This parameter defines the maximum allowed error for the estimated answer.
2. Tuples Sampled per peer(t): This parameter defines the number of tuples to be sampled from each selected peer.
3. Jump Size(j): This parameter defines the number of peerss to be passed over before selecting the next peer for sampling.
4. Query(Q): Aggregate query

4.4. Output Design

The output must be provided in a format easily understandable even by a novice user. After analyzing the operations of the system, output information required for each jobs are determined .In addition to this, these outputs may be in format suitable as inputs for subsequent processing.

A major form of output is a hard copy from the printer. Printout should be designed around the output requirements of the user. An efficient output design should improve the system relationship with the end user. Output design refers to the result generated by the system. The output of a system can take many forms. The most

The output of this project is in the form of screen display. It displays the data satisfying the aggregate query with expected accuracy.

4.5. Database Design

A database is a collection of inter-related data stored with minimum redundancy to serve many users quickly and efficiently. The general objective of database design is to make the data access easy, inexpensive and flexible to the user. An elegantly designed database can play a strong foundation for the whole system.

The details about the relevant data for the system are first identified. According to their relationship, tables are designed through the following method.

- The data type for each data item in the table is decided.
- The tables are then normalized.

The tables are normalized so that they can provide better response time, have data integrity, avoid redundancy and be secure.

4.5.1 TABLE STRUCTURE

Design Conventions Used

1. Appropriate words that describe the table should be used.
2. Words used to describe table should be separated with an Underscore'_'.
3. No special character other than an underscore is used in table name.
4. No number should be used anywhere in the table name string.

Table 4.1: Client

Field	Type	Key	Null
Id	Int	Primary	No
Ip1	var char(20)		Yes
cname	var char(20)		Yes
os_name	var char(20)		Yes
os_version	var char(20)		Yes
p_memory	Bigint		Yes
client_index	Int		Yes

Table 4.2: Sub Server

Field	Type	Key	Null
id	Int	Primary	No
cname	var char(20)		Yes
port	var char(20)		Yes
q_node	var char(20)		Yes

Table 4.3: Employee Master

Field	Type	Key	Null
Id	Int	Primary	No
Age	Int		No
work_class	var char(20)		Yes
Flnwgt	var char(20)		Yes
Education	var char(20)		Yes
marital_status	var char(20)		Yes
occupation	var char(20)		Yes
relationship	var char(20)		Yes
Race	var char(20)		Yes
Sex	var char(20)		Yes
capital_gain	Int		Yes
capital_loss	Int		Yes
native_country	var char(20)		Yes
Salary	Int		Yes

Table 4.4: Salary Master

Field	Type	Key	Null
Id	Int	Primary	No
Age	Int		No
Salary	Int		Yes

Table 4.5: Population Table

Field	Type	Key	Null
Id	Int	Primary	No
country	var char(20)		Yes
state	var char(20)		Yes
city	var char(20)		Yes
city1	var char(20)		Yes
population	Int		Yes

Table 4.6: Employee Master II

Field	Type	Key	Null
id	Int	Primary	No
slno	Int		Yes
age	Int		No
work_class	var char(20)		Yes
flnwgt	var char(20)		Yes
education	var char(20)		Yes
marital_status	var char(20)		Yes
occupation	var char(20)		Yes
relationship	var char(20)		Yes
race	var char(20)		Yes
Sex	var char(20)		Yes
capital_gain	Int		Yes
capital_loss	Int		Yes
native_country	var char(20)		Yes
salary	Int		Yes

4.5.2. Database ER Diagram

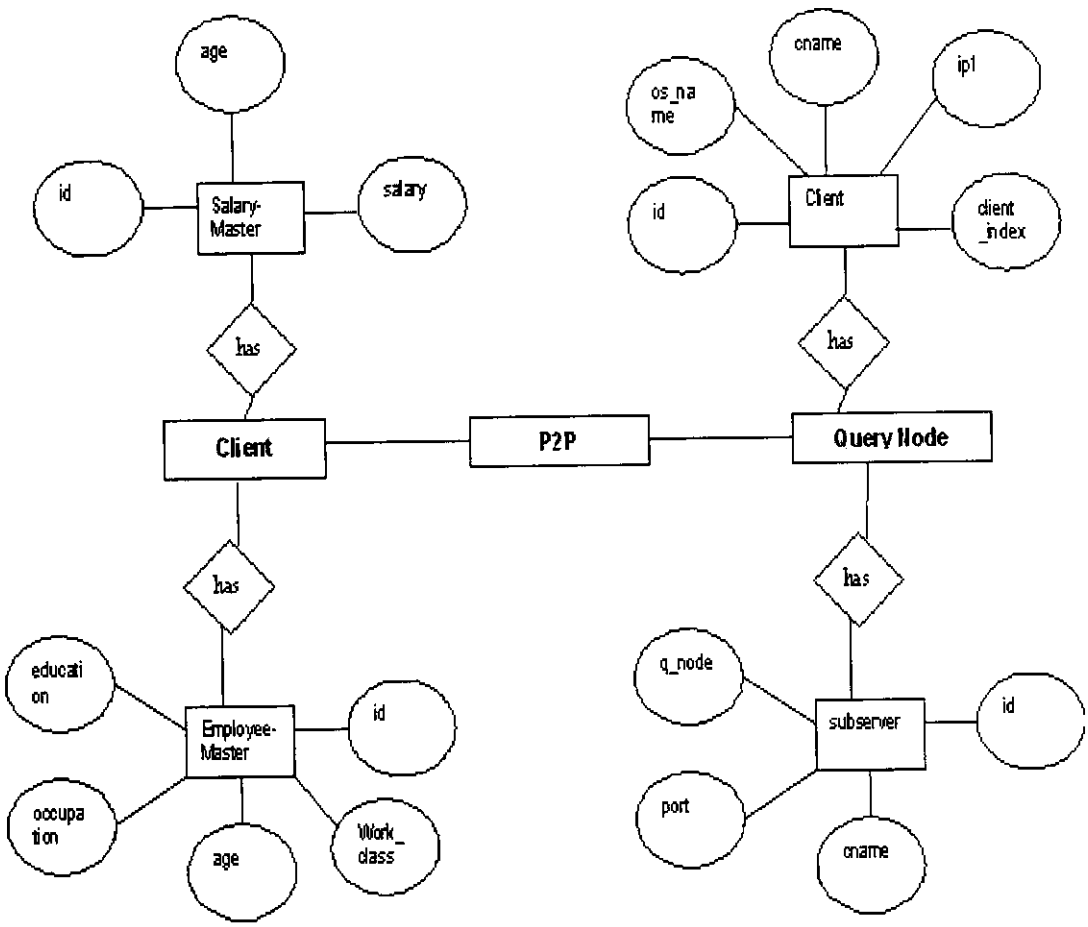


Figure 4.1

4.6 USE CASE DIAGRAMS

Use case diagrams give a picture of the different scenarios wherein users interact with the different components of the system. It gives a general idea on the requirements to be addressed by the system and the sequence of operations happening.

P2P Network Establishment

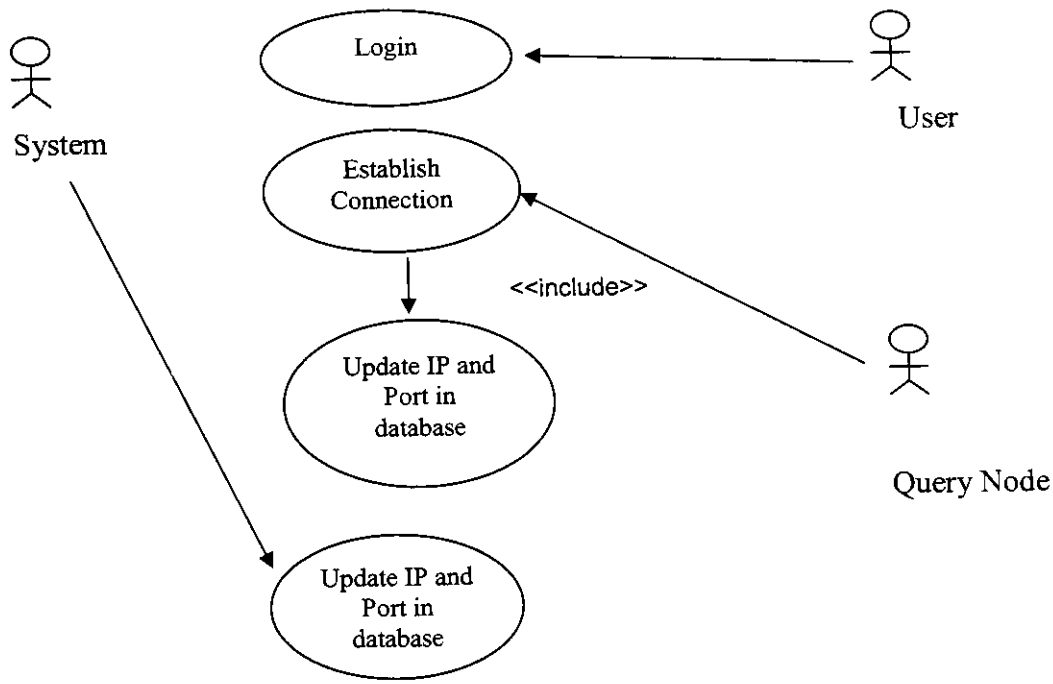


Figure 4.2

Random Walk Algorithm Phase I

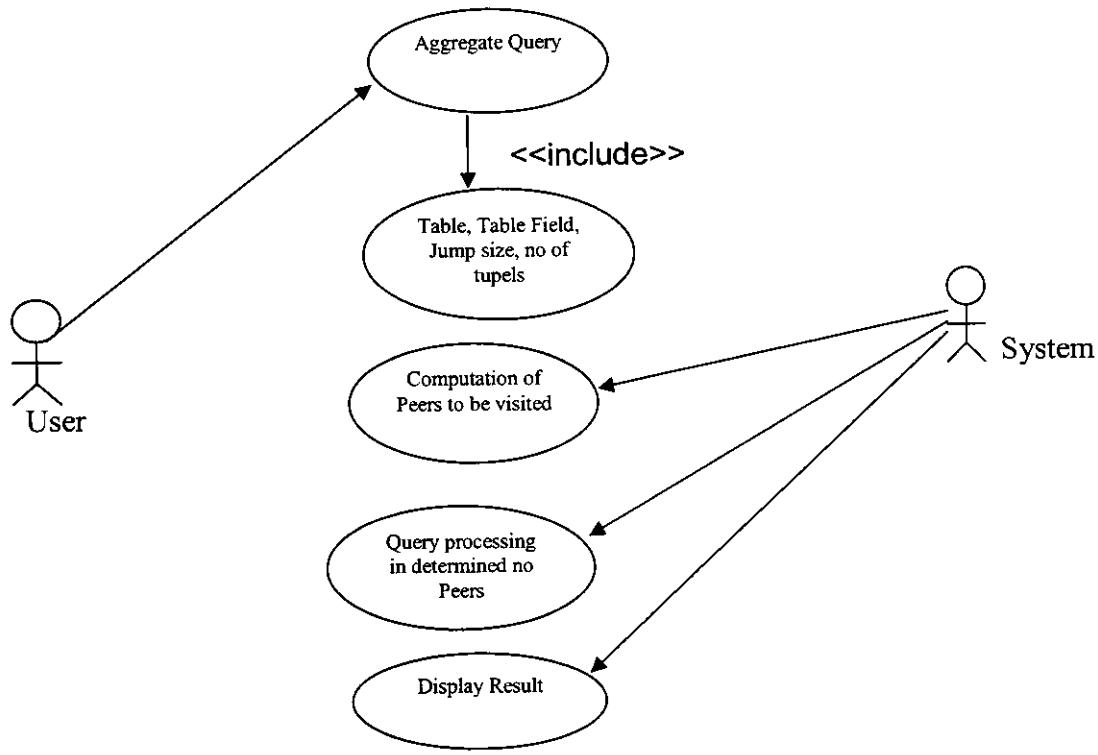


Figure 4.3

Evaluation of Base Result

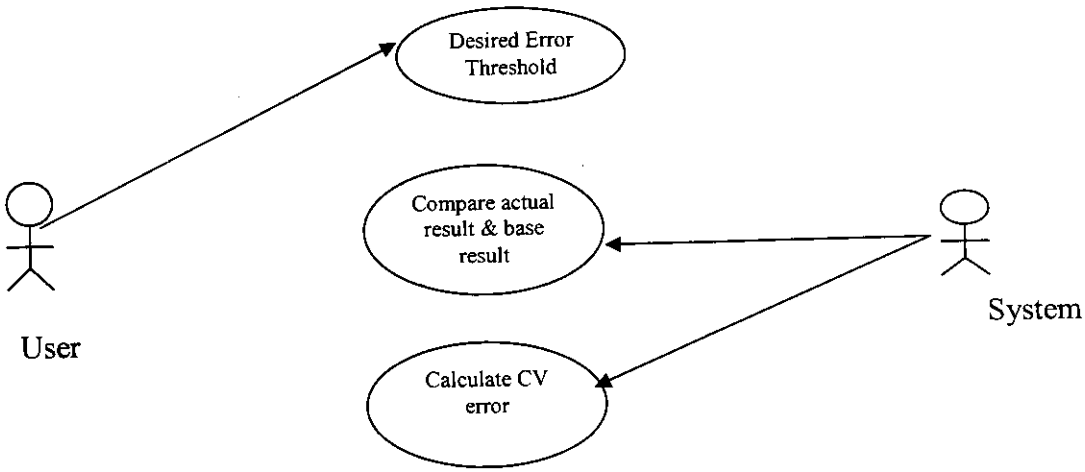
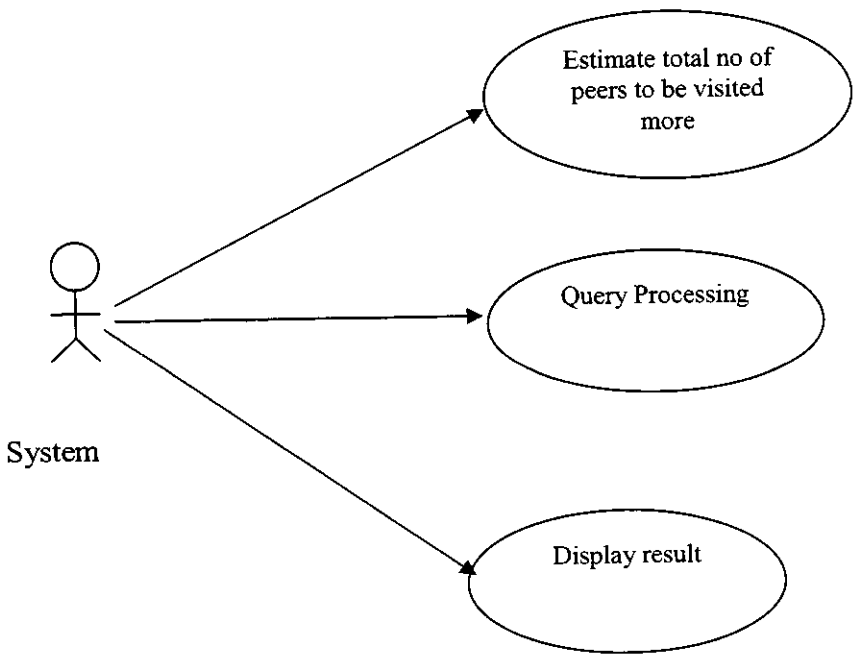
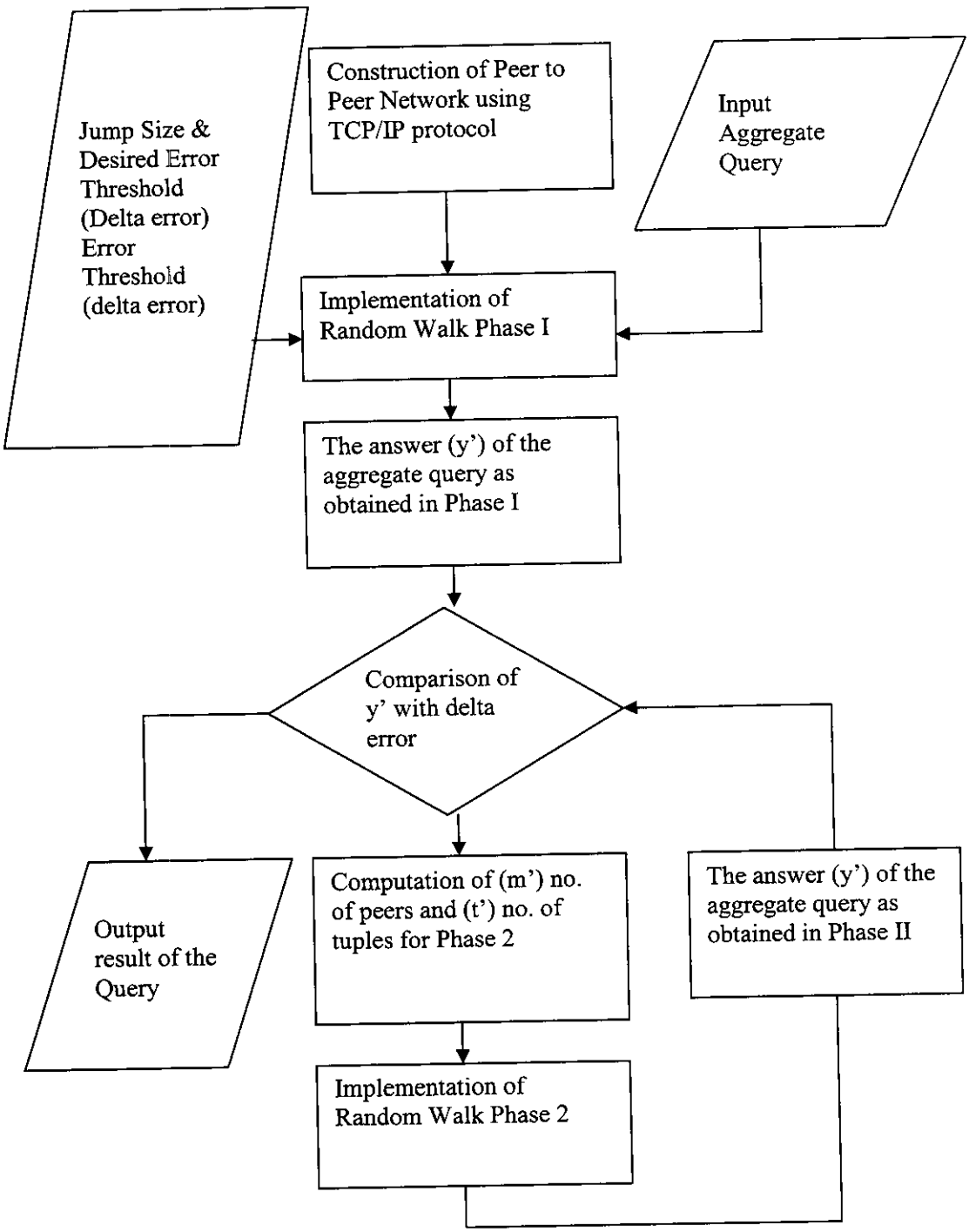


Figure 4.4

Random Walk Algorithm Phase II



4.7. System Flow Diagram



IMPLEMENTATION

CHAPTER 5

IMPLEMENTATION

System Implementation is the part of the software engineering life cycle, where, the design artifacts are converted to a working application. Coding is done in this stage using an apt framework and programming language, which would solve the specific problem the best way. Once the design is coded into a working application, it has to be verified, validated and tested in detail. The tested product if successful is deployed in the user environment.

5.1 SYSTEM VERIFICATION

System Verification System verification include the review of interim work steps and interim deliverables during a project to ensure they are acceptable.

In this project, verification determines if the system adheres to standards, uses reliable techniques and checks for consistency. The selected functions are performed according to their requirements .In data access it verifies whether the right data is being retrieved ,in terms of the right place and in the right way. In network access, it verifies whether the client is accessible.

5.2 SYSTEM VALIDATION

In this project, Validation checks whether the developer is moving towards

product that was agreed upon in the beginning. Validation also determines if the system compiles with the requirements and platforms functions for which it is intended and meets the organization's goals and user needs. It is traditional and is performed at the end of the project.

TESTING

CHAPTER 6

TESTING

Testing is a critical element of software quality and assurance and represents the ultimate review of specification design and coding. It is a vital activity that has to be enforced in the development of any system. This could be done in parallel during all the phases of system development. The feedback received from these tests can be used for further enhancement of the system under consideration. The testing phase conducts test using the Software Requirement Specification as a reference and with the goal to see whether the system satisfies the specified requirements.

The main types of tests carried out on External True System are:

- Unit Test
- Integration Test
- System Test

6.1. Unit Testing

Module or Unit Testing is the process of testing all the program units that make up a system. Unit testing focuses on an individual module thus allowing one to uncover all the errors made logically and while coding in the module.

In this System each page is tested separately as a unit. Initially the flow of control and data through that page is checked. When considering a module as a unit, the flow of data and control through the whole module is tested. The result is stored in the test plan. In a page, each control is further tested in unit testing. The process is done in all the pages of the system. Once the errors are rectified, the testing procedure is

repeated with same test cases to ensure this hasn't produced new errors. Hence this is a continuous process.

6.2 Integration Testing

Integration testing tests the process of integrating the various modules to form the completed system. Integration starts with a set of units each individually tested in isolation and ends when the entire application has been built. Integration testing verifies that the combined units function together correctly. It facilitates in finding problem that occur at interface or communication between the individual parts.

This System follows top-down integration testing. This process is continued from the page level to module level, finally to the system level. In the final stage, the whole system is taken together and tested for integration. A change in one place should be reflected through out the system. Regression testing is done after each change made into the software. This tests if the change has affected any part of the System negatively after the change was made. The whole set of test cases need to be run again to do the regression testing.

6.3 System Testing

In this project, system testing is done to fully exercise the computer-based system. This helps in verifying that all the system elements have been properly integrated and perform the allocated functions. It verifies the entire product after having integrated all software and hardware components, and validates it according to the original project requirement.

6.3.1. Stress Testing

Stress Testing executes a system in a manner that demands resources in abnormal quantity, frequency or volume. This proposed System was stress tested by simultaneously connecting from different system on network,

6.4. TEST CASES

Module Name: Connection Establishment

S.No	Test Case ID	Test Case Description	Expected Result	Actual Result	Status
1	CE_01	Enter null string in "Login Name" Textbox	"Login Name" cannot be empty. Should display message to user	Message displayed to user	Pass
2	CE_02	Check Connectivity when server is not started	"Host Unreachable" message should be displayed to the user	Message displayed to user	Pass

Module Name: Random Walk Phase I

S.No	Test Case ID	Test Case Description	Expected Result	Actual Result	Status
1	RWPI_01	Query processing request when already a query is being processed	Should display error message to user	Message displayed to user	Pass
2	RWPI_02	Enter null string in "Jump Size" Textbox	Should display error message to user	Error Message displayed to user	Pass
3	RWPI_03	Enter text in "no of tuples" Textbox	Should display error message to user	Error Message displayed to user	Pass

Module Name: Base Result Evaluation

S.No	Test Case ID	Test Case Description	Expected Result	Actual Result	Status
1	BRE_01	Enter null string in "Delta Req" Textbox	Should display error message to user	Message displayed to user	Pass
2	BRE_02	Calculated CV error greater than Delta Req.	"Advised for second phase" message should be displayed	Message displayed to user	Pass

CONCLUSION

CHAPTER 7

CONCLUSION

The project '**Efficient Approximate Query Processing in P2P Network**' has been designed and developed using VB.NET and SQL SERVER 2000.It is user friendly and deals with the approximate answering of ad hoc aggregation queries in distributed and dynamic P2P database. It computes an approximate answer to the aggregate query that satisfies the desired error bound.

The current system overcomes most of the drawbacks of existing system through the use of adaptive two-phase sampling approach based on random walks of the P2P graph. Moreover, it can be easily modified or upgraded to suit the changes in requirement or technology, arising at any point of time in the future.

The current system is very efficient because it has been repeatedly tested with the help of a variety of test cases and can therefore be implemented successfully. Since the system has been developed using standard programming codes, rules and conventions, it is easily understandable and can be reused under any similar circumstances in the future.

FUTURE ENHANCEMENT

CHAPTER 8

FUTURE ENHANCEMENT

- The efficiency of the query processing can be enhanced using Replication technique. The Replication technique is combined with Random Walk technique in order to achieve high success rate for query processing.

- The Replication technique might duplicate data more than the specified level. Therefore RLE (Randomized Leader Election) approach is used to remove multiple entries of the same data in P2P network

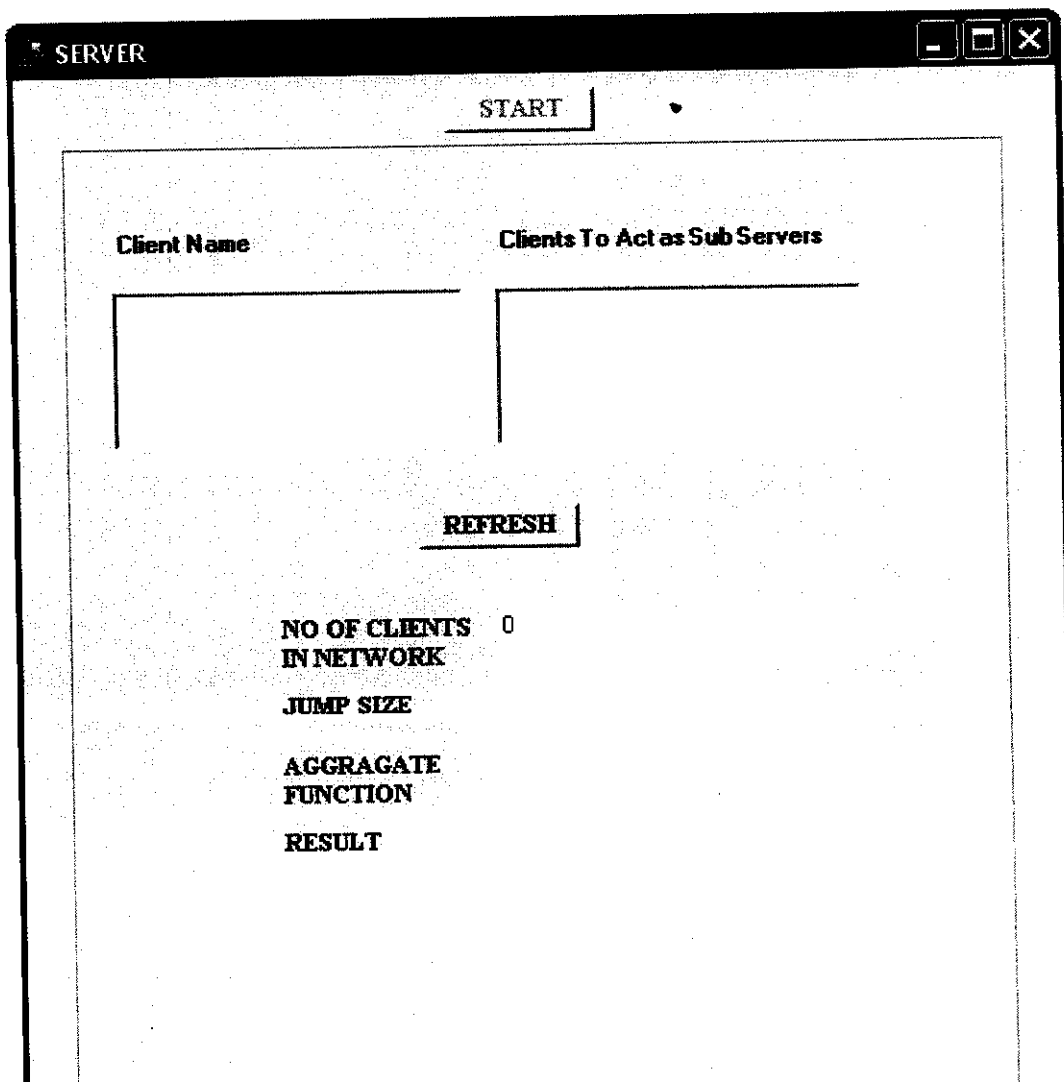
APPENDICES

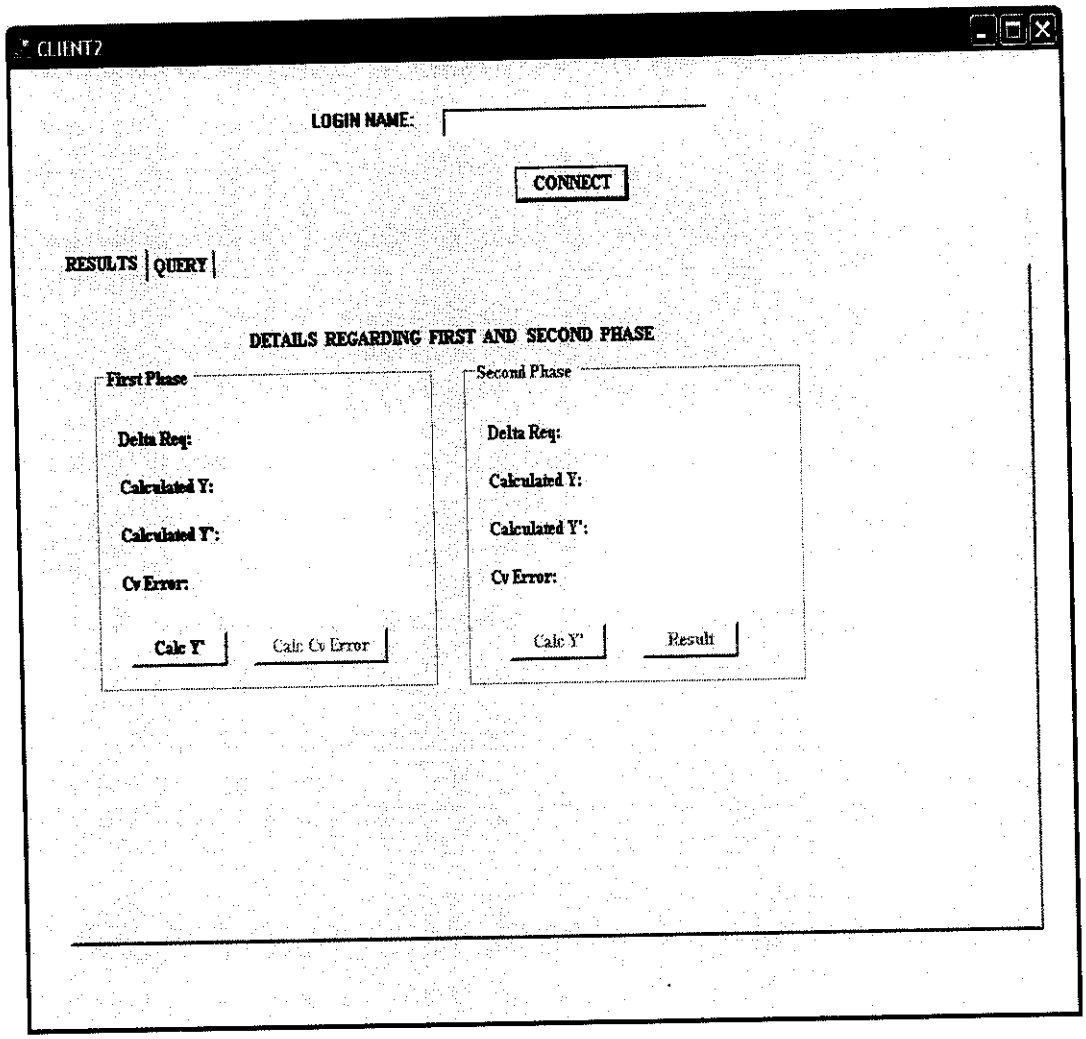
CHAPTER 9

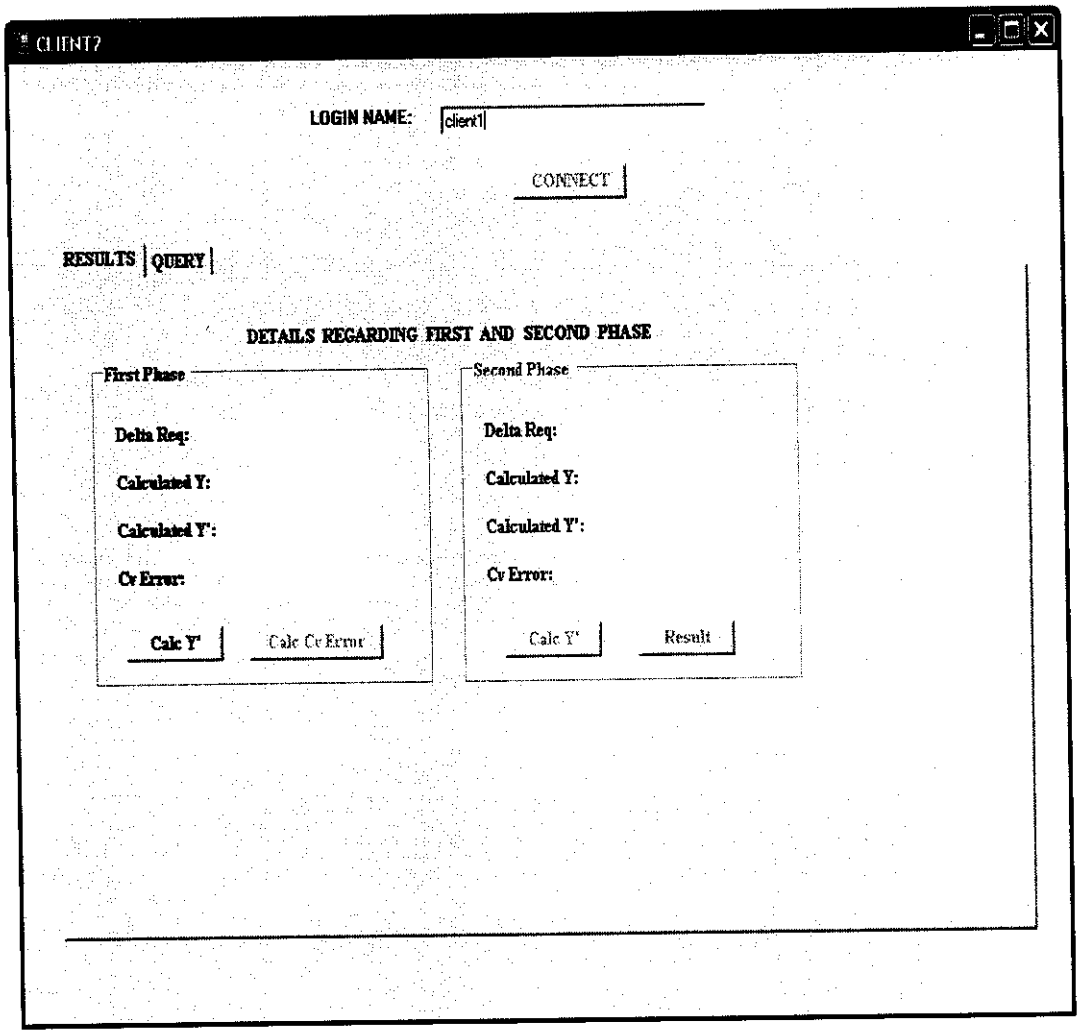
APPENDICES

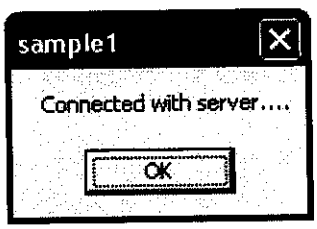
SCREEN SHOTS

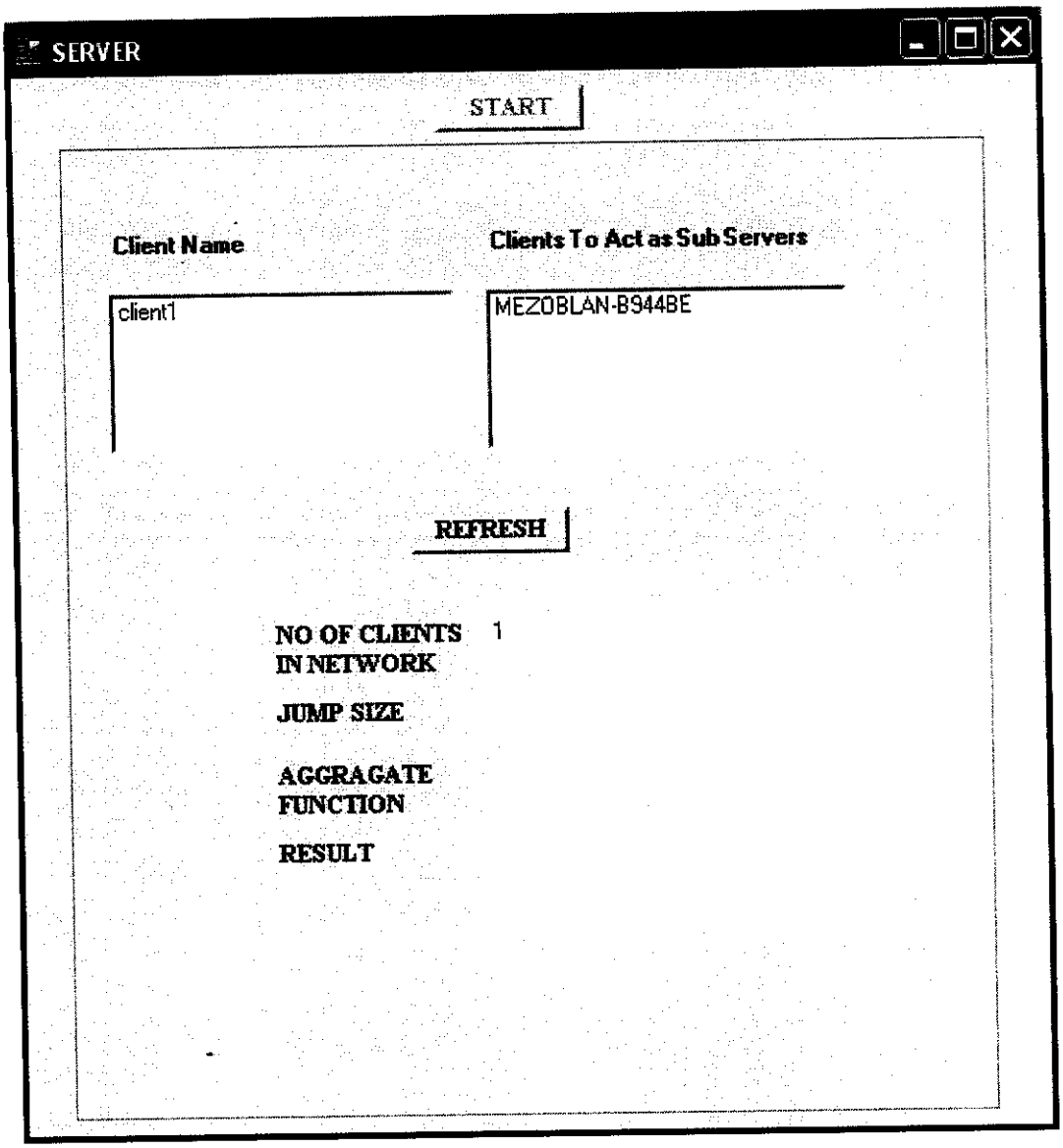
9.1 CONNECTION ESTABLISHMENT

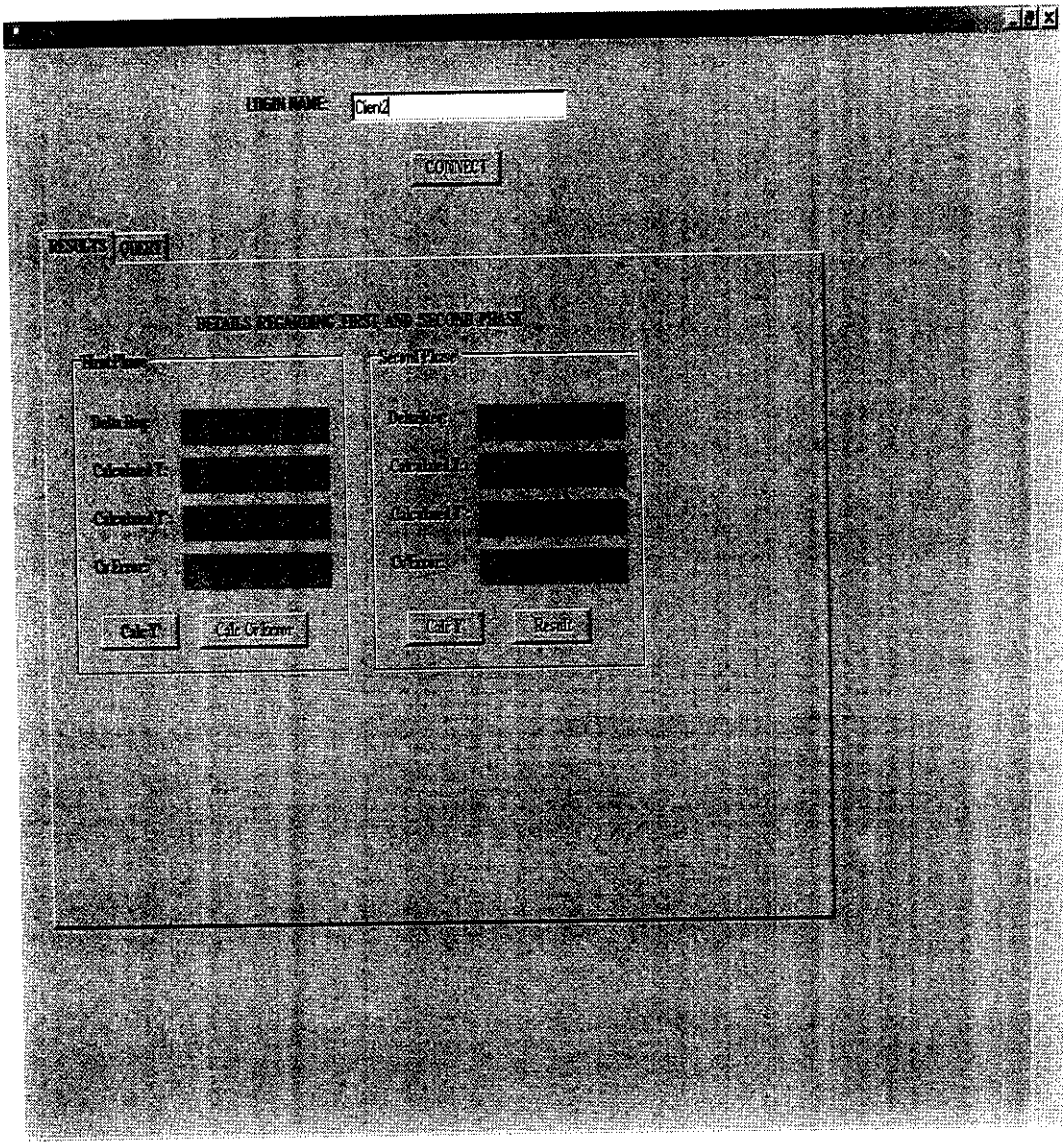


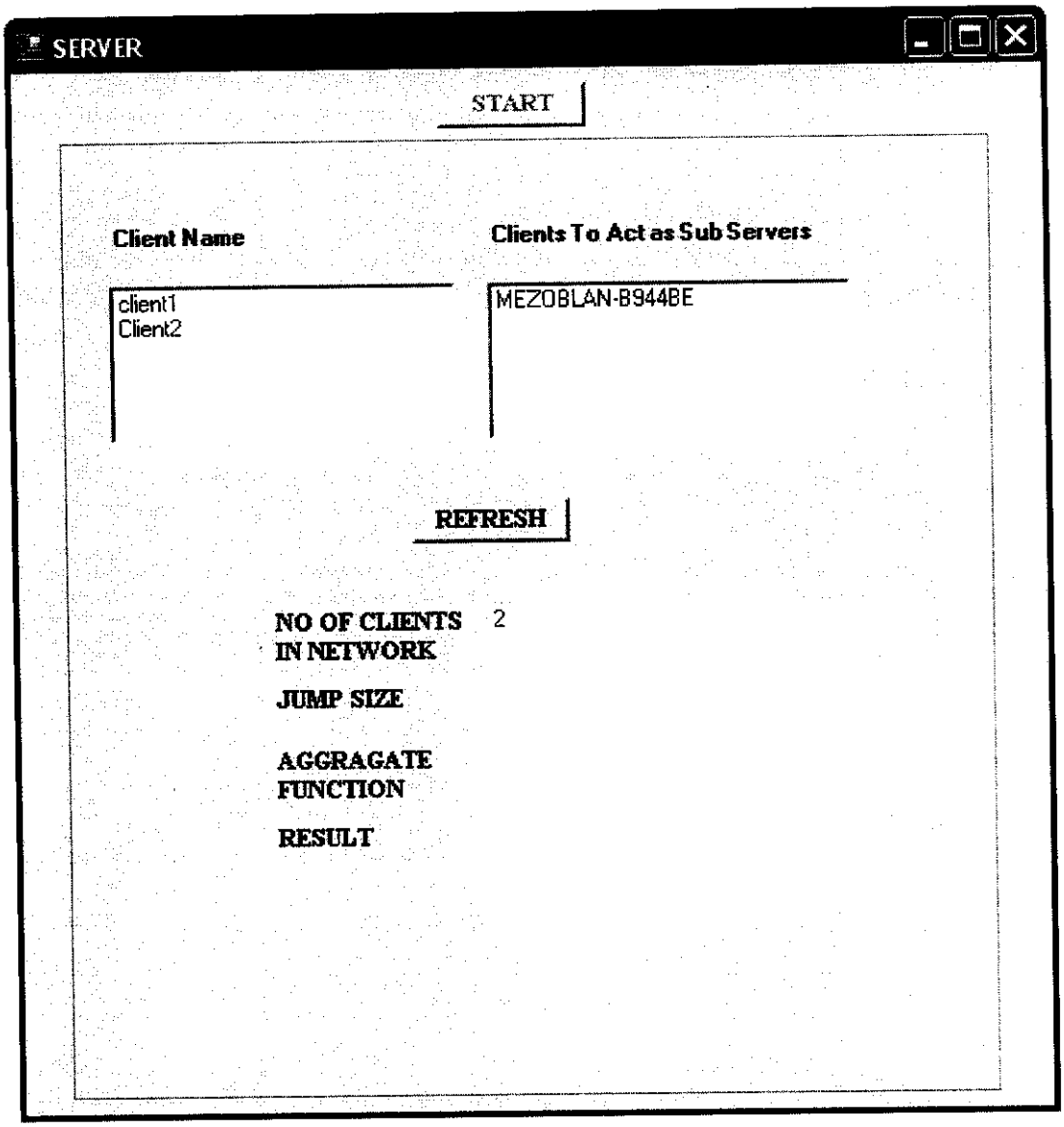

CLIENT2
LOGIN NAME:
CONNECT
RESULTS
QUERY
DETAILS REGARDING FIRST AND SECOND PHASE
First Phase
Delta Req:
Calculated Y:
Calculated Y':
Cv Error:
Calc Y'
Calc Cv Error
Second Phase
Delta Req:
Calculated Y:
Calculated Y':
Cv Error:
Calc Y'
Result


CLIENT2
LOGIN NAME:
client1
CONNECT
RESULTS
QUERY
DETAILS REGARDING FIRST AND SECOND PHASE
First Phase
Delta Req:
Calculated Y:
Calculated Y':
Cv Error:
Calc Y'
Calc Cv Error
Second Phase
Delta Req:
Calculated Y:
Calculated Y':
Cv Error:
Calc Y'
Result

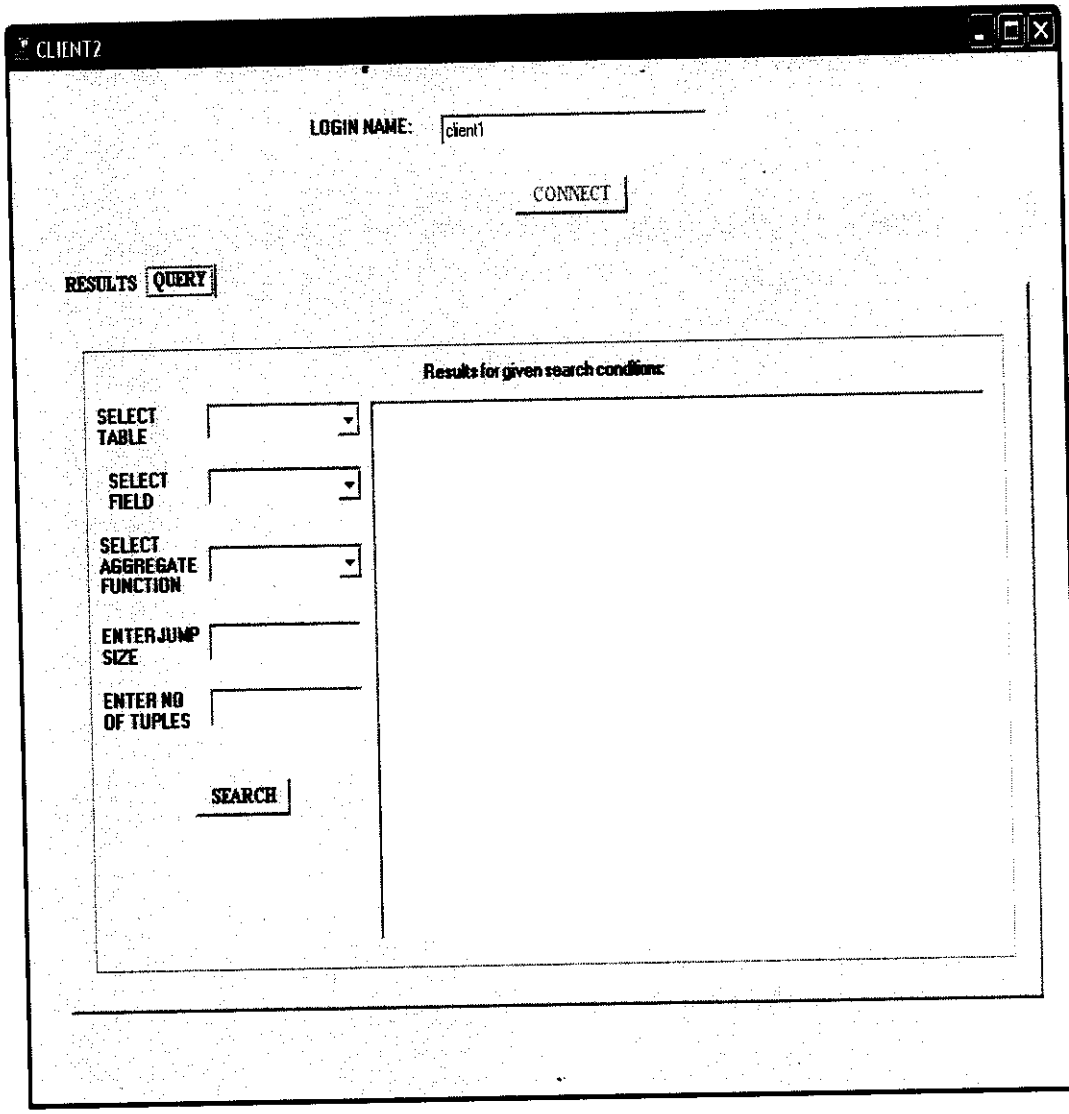

sample1
Connected with server....
OK

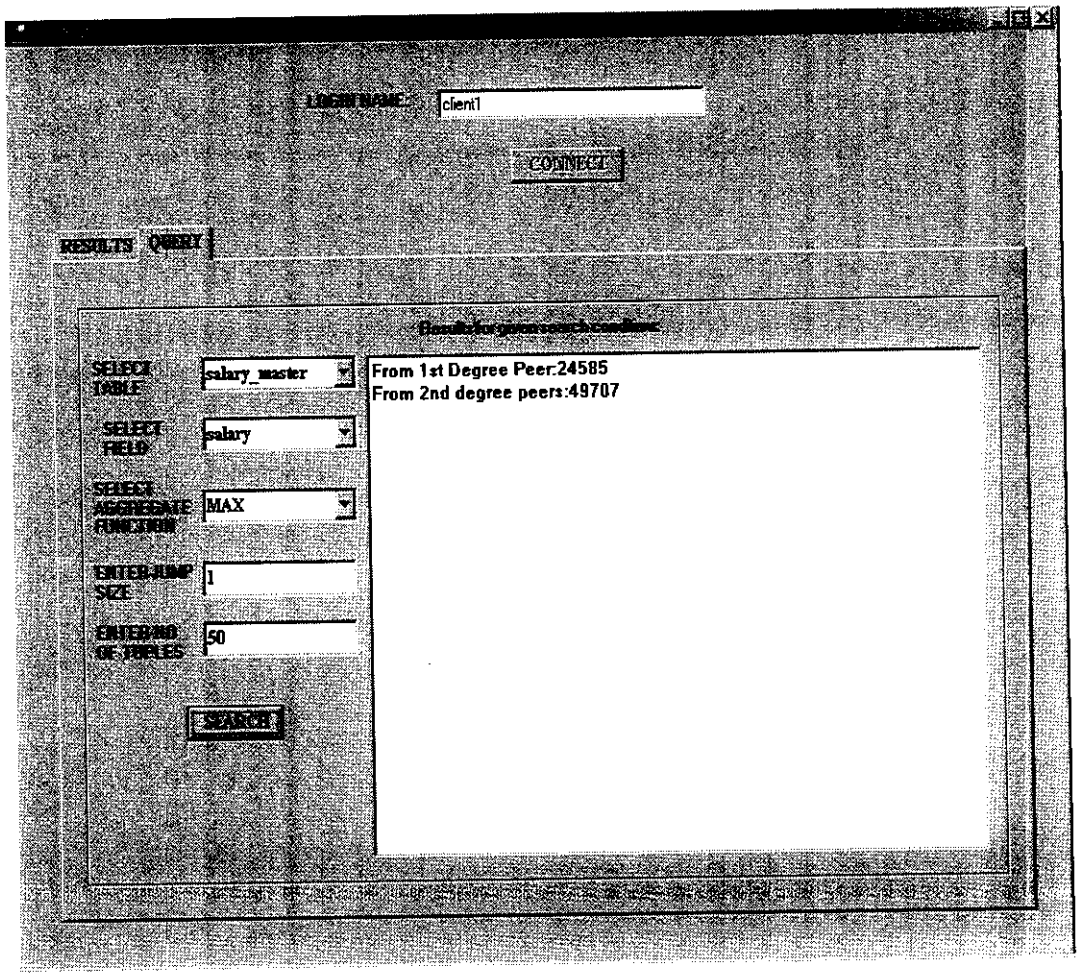

SERVER
START
Client Name
Clients To Act as Sub Servers
client1
MEZOBLAN-B944BE
REFRESH
NO OF CLIENTS IN NETWORK
1
JUMP SIZE
AGGRAGATE FUNCTION
RESULT

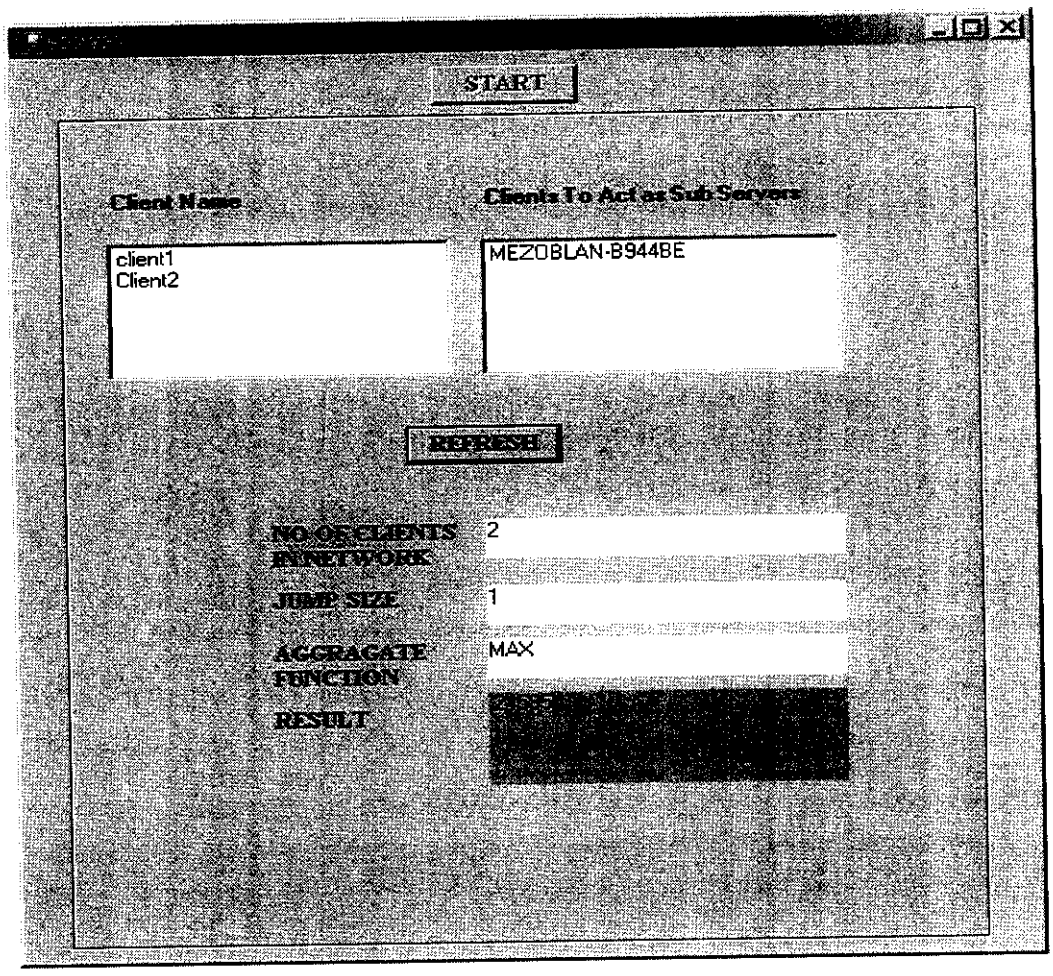

LOGIN NAME:
Client2
CONNECT
RESULTS
QUERY
DETAILS REGARDING FIRST AND SECOND PHASE
First Phase
Second Phase
Calc T
Calc Of Error
Calc T
Result

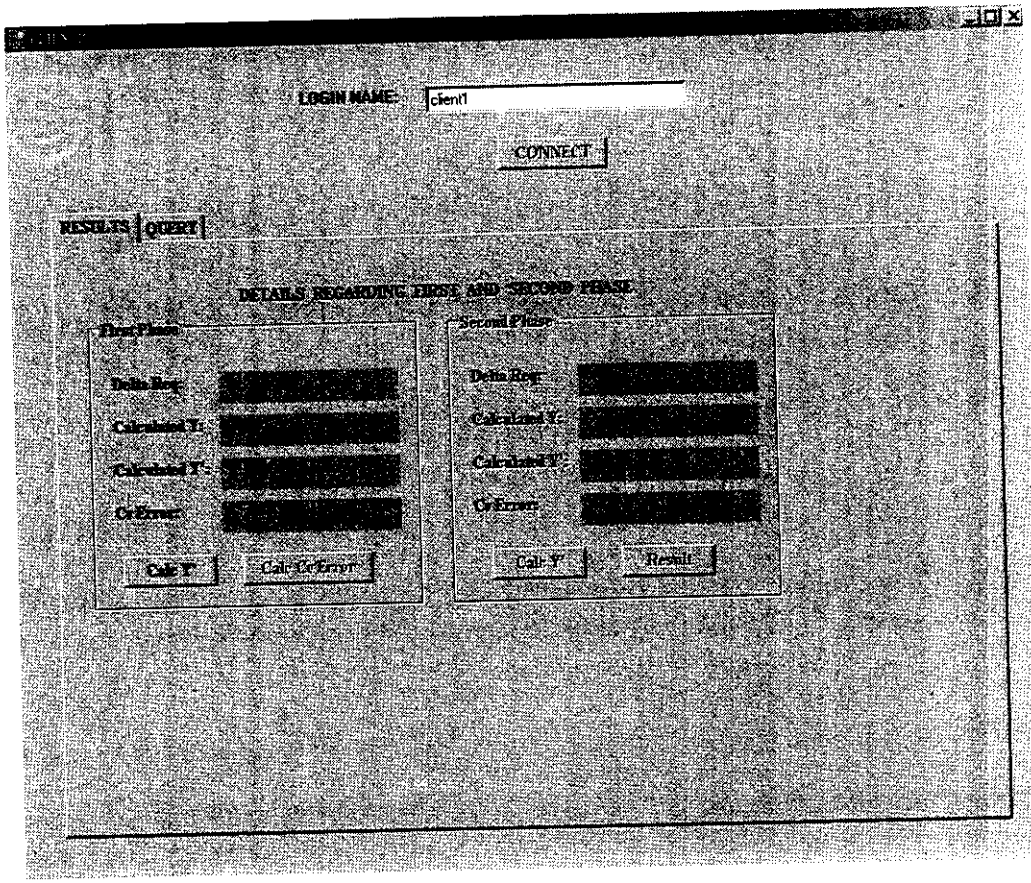

SERVER
START
Client Name
Clients To Act as Sub Servers
client1
Client2
MEZOBLAN-B944BE
REFRESH
NO OF CLIENTS IN NETWORK
2
JUMP SIZE
AGGRAGATE FUNCTION
RESULT

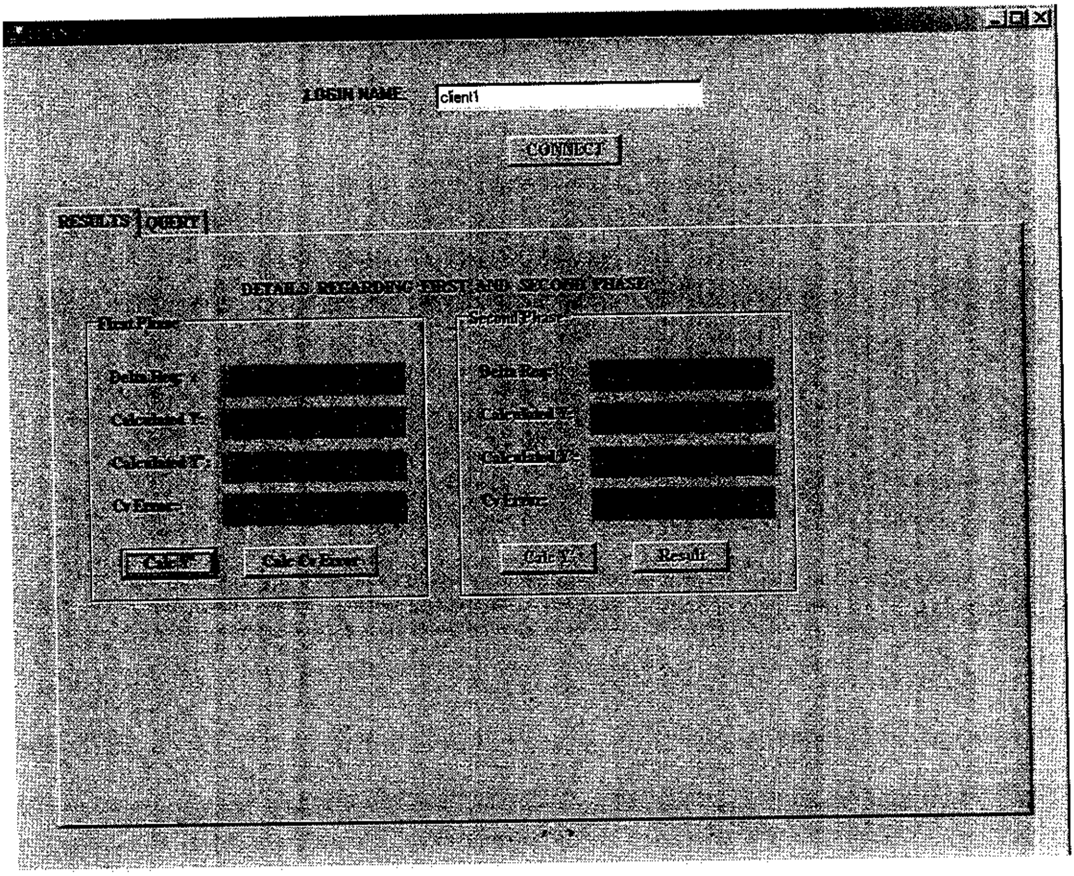
9.2 RANDOM WALK PHASE-I

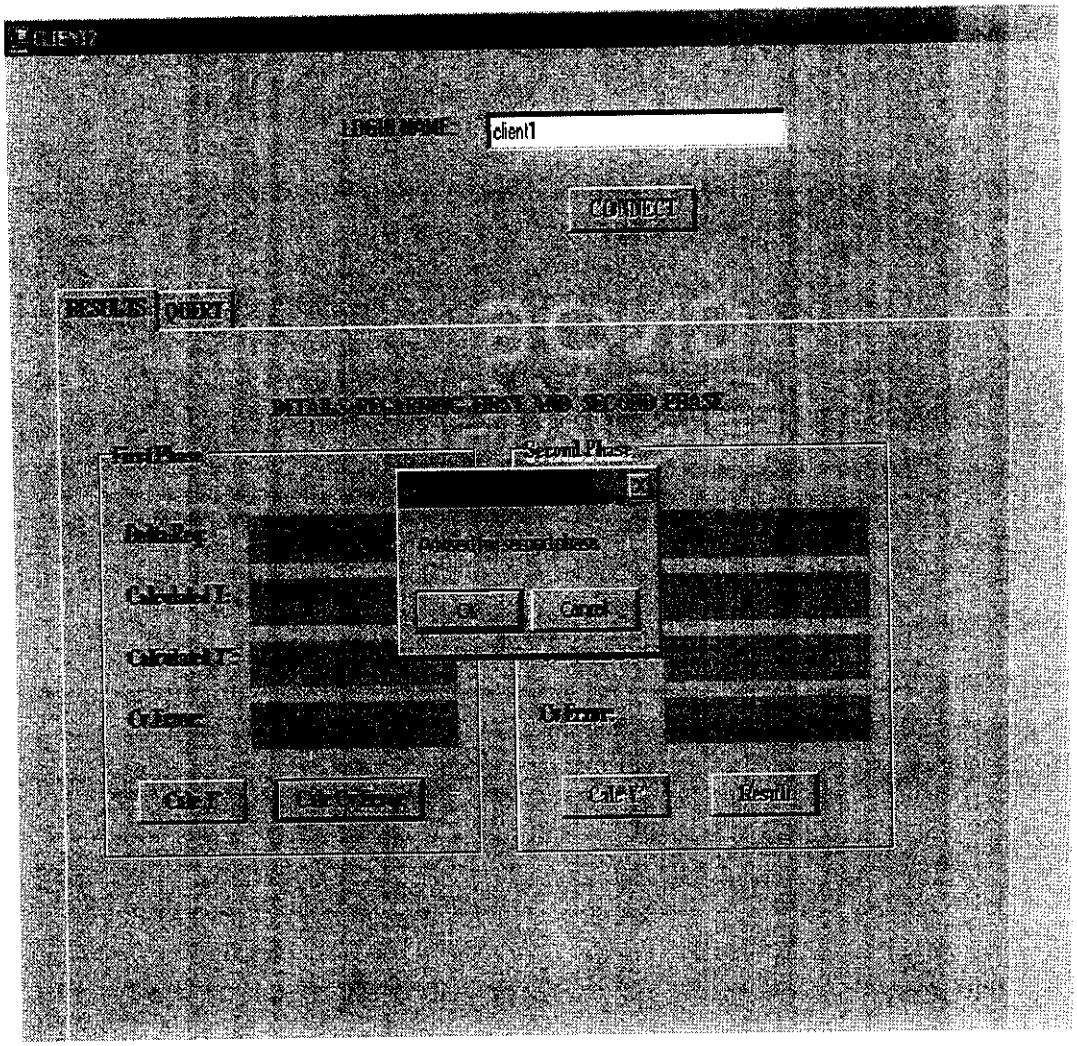



client1
CONNECT
RESULTS
QUERY
SELECT TABLE
salary_master
SELECT FIELD
salary
SELECT AGGREGATE FUNCTION
MAX
ENTER JUMP SIZE
1
ENTER NO OF TABLES
50
SEARCH
From 1st Degree Peer:24585
From 2nd degree peers:49707

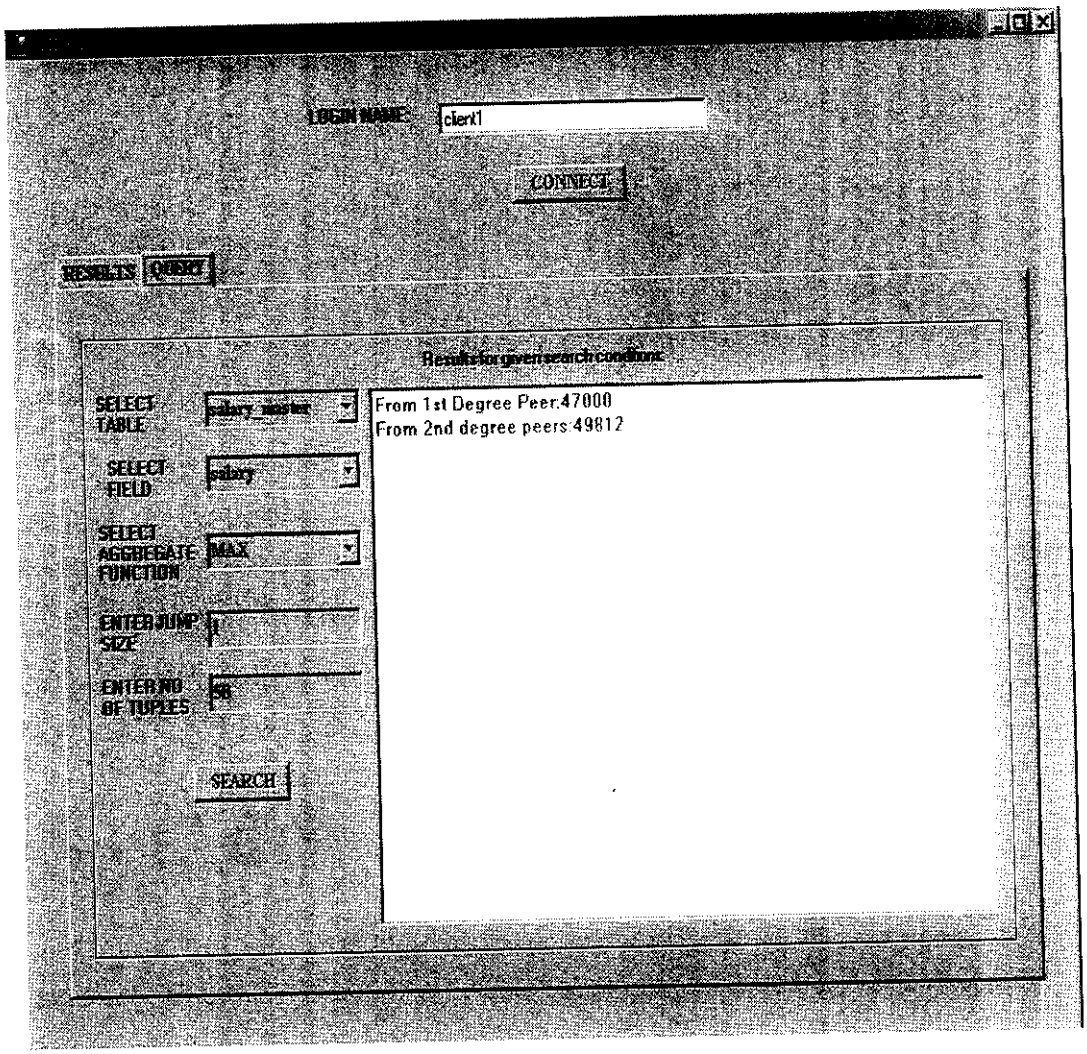

START
Client Name
Clients To Act as Sub Servers
client1
Client2
MEZOBLAN-B944BE
REFRESH
NO OF CLIENTS
NETWORK
2
JUMP SIZE
1
AGGRAGATE
FUNCTION
MAX
RESULT

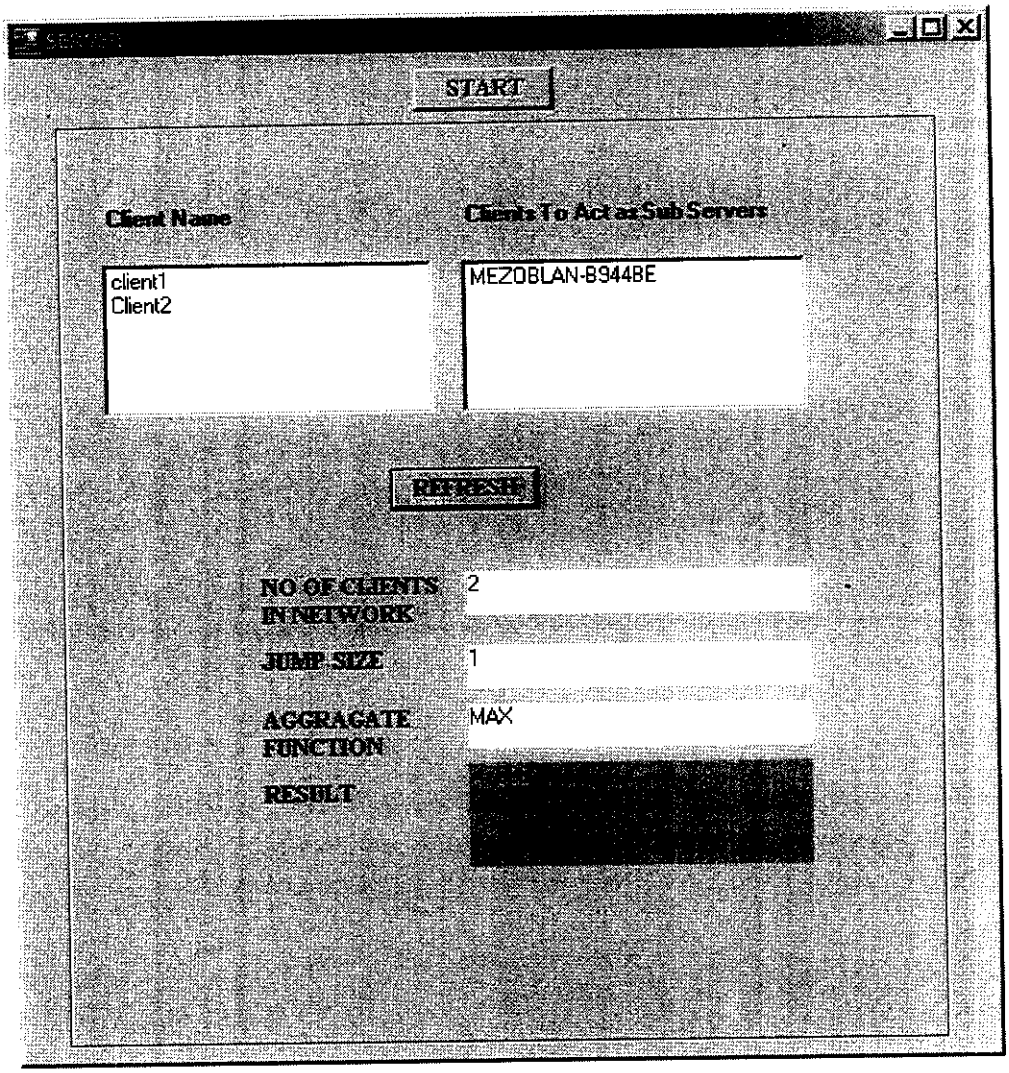

LOGIN NAME:
client1
CONNECT
RESULTS
QUERY
DETAILS REGARDING FIRST AND SECOND PHASE
First Phase
Delta Beg:
Calculated Tc:
Calculated T:
Cv Error:
Calc T
Calc Cv Error
Second Phase
Delta Beg:
Calculated Tc:
Calculated T:
Cv Error:
Calc T
Result

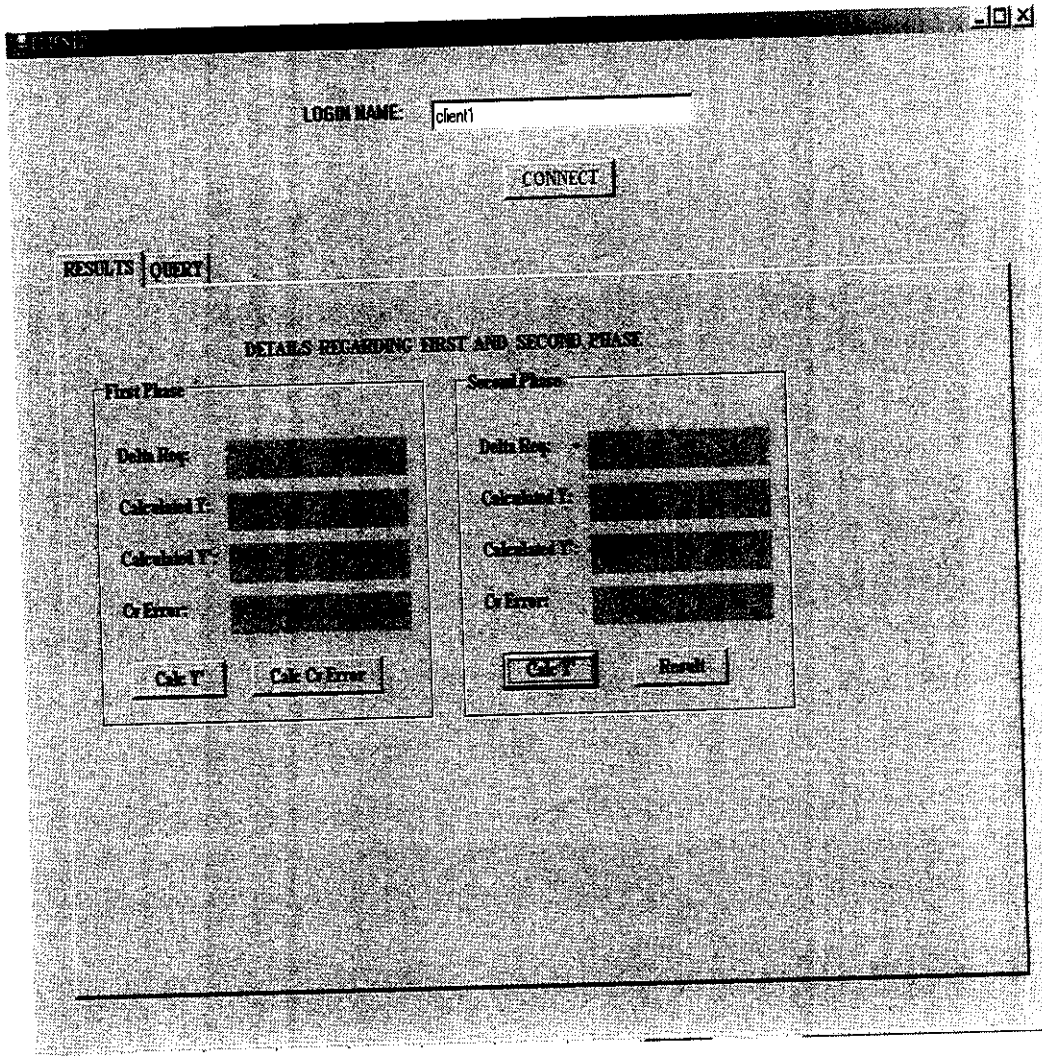

LOGIN NAME
client1
CONNECT
RESULTS
QUERY
DETAILS REGARDING FIRST AND SECOND PHASE
First Phase
Second Phase
Result

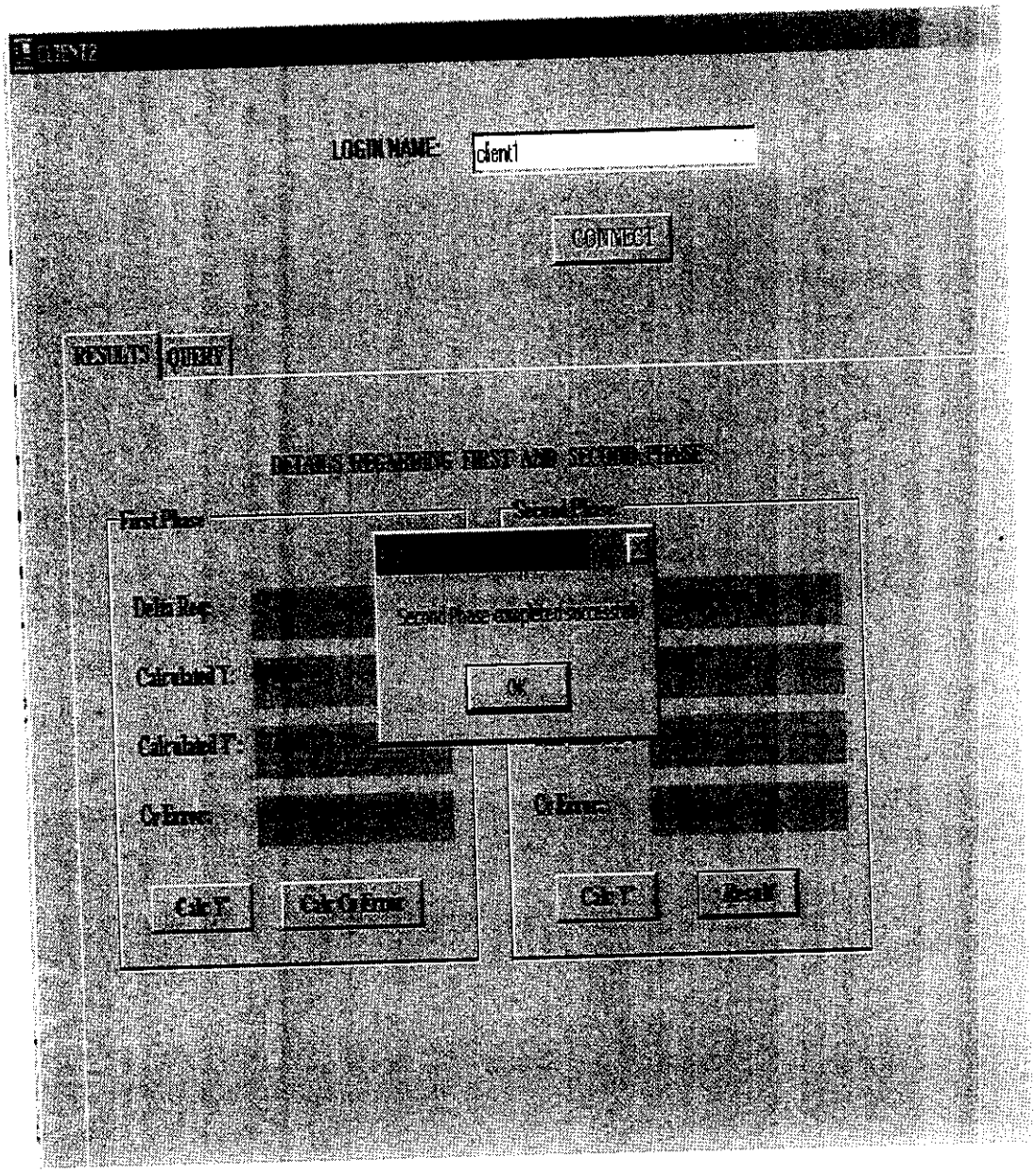

client1
CONNECT
RESULTS
QUERY
Cancel
Result

9.3 RANDOM WALK PHASE-II




START
Client Name
Clients To Act as Sub Servers
client1
Client2
MEZOBLAN-B9448E
REFRESH
NO OF CLIENTS IN NETWORK
2
JUMP SIZE
1
AGGRAGATE FUNCTION
MAX
RESULT


LOGIN NAME:
client1
CONNECT
RESULTS
QUERY
DETAILS REGARDING FIRST AND SECOND PHASE
First Phase
Delta Req:
Calculated T:
Calculated T':
Cr Error:
Calc T'
Calc Cr Error
Second Phase
Delta Req:
Calculated T:
Calculated T':
Cr Error:
Calc T'
Result


LOGIN NAME:
client1
CONNECT
RESULTS
QUERY
First Phase
Second Phase
Second Phase completed successfully
OK
Delta Temp:
Calculated T:
Calculated T:
Calc T
Calc

REFERENCES

REFERENCE

BOOKS

1. Alistair McMonnies, Object Oriented Programming in VB.Net
2. Gary Cornell and Jonathan Morrison, Programming VB .NET: A Guide for Experienced Programmers
3. Tom Barnaby, Distributed .NET Programming in VB .NET
4. Ray Rankins, Paul Jensen, Paul Bertucci, and Chris Gallelli, Microsoft SQL Server 2000 Unleashed
5. Robert Vieira, Professional SQL Server 2000 Programming (Programmer to Programmer)

REFERENCE SITES

1. www.**dotnet**-guide.com
2. www.vbdotnetheaven.com
3. www.wikipedia.com