P-3110

XML QUERY DECOMPOSITION

A PROJECT REPORT

Submitted by

S.MANOJ KUMAR

D.NARESH KUMAR

A.KARTHICK

In partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

KUMARAGURU COLLEGE OF TECHNOLOGY, COIMBATORE

ANNA UNIVERSITY, CHENNAI-600 025

APRIL 2010

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report entitled "**XML QUERY DECOMPOSITION**" is the bonafide work of Manoj Kumar.S , Naresh Kumar.D and Karthick.A, who carried out the research under my supervision. Certified also, 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.

SIGNATURE

Dr.S.Thangasamy

HEAD OF THE DEPARTMENT

Department of COMPUTER SCIENCE & ENGINEERING,

Kumaraguru College of Technology,

Coimbatore – 641006.

SIGNATURE

Mrs.R.Kalaiselvi

SENIOR LECTURER

Department of COMPUTER SCIENCE & ENGINEERING,

Kumaraguru College of Technology,

Coimbatore – 641006.

The candidates with University Register Numbers: 71206104027, 71206104030, 71206104304 were examined by us in the project viva - voice examination held on

15-04-2010.

INTERNAL EXAMINER

EXTERNAL EXAMINER

DECLARATION

DECLARATION

We,

S.Manoj Kumar , D.Naresh Kumar and A.Karthick declare that the project entitled **"XML QUERY DECOMPOSITION"**, submitted in partial fulfillment to Anna University as the project work of Bachelor of Engineering (computer science) degree, is a record of original work done by us under the supervision and guidance of Mrs.R.Kalaiselvi BE, Senior Lecturer, Department of computer science and engineering, Kumaraguru College of technology, Coimbatore.

Place: Coimbatore

Date: 15-04-2010

S.Manoj kumar
[S.MANOJ KUMAR]

D.Naresh Kumar
[D.NARESH KUMAR]

A.Karthick
[A.KARTHICK]

ACKNOWLEDGEMENT

ACKNOWLEDGEMENT

We express our sincere thanks to correspondent **Mr.M.Balasubramaniam,** Kumaraguru College of technology and the management, for providing us this opportunity to undertake this project work.

We express our profound gratitude to **Dr.S.Ramachandran** principal, Kumaraguru College of technology, for permitting us to undergo a project work.

We are greatly indebted to **Dr.S.Thangasamy**, Professor and Head of the department of computer science and engineering, for the immense support he has provided us throughout our project.

We extend our sincere thanks to our project coordinator **Mrs.P.Devaki**, Associate professor, Department of computer science and engineering, for her constant support and encouragement.

We would like to express our heartfelt thanks to our guide and class advisor, **Mrs.R.Kalaiselvi**, senior lecturer, Department of computer science and engineering, who gave the initial idea and guidance to carry out every further step and for her everlasting counseling and untiring help throughout our project.

We would also like to thank all the faculty members and the lab technicians of Department of computer science and engineering for their valuable guidance, support and engagement during the course of the project work.

ABSTRACT

ABSTRACT

In order to access data from various different data repositories, in Global-As-View approaches an input query is decomposed into several sub queries. Normally, this decomposition is based on a set of mappings, which describe the correspondence of data elements between a global schema and local ones.

However, building mappings is a difficult task, especially when the number of participating local schemas is large. In our approach, an input query is automatically decomposed into sub queries without using mappings. An algorithm is proposed to transform a global path expression (e.g. an XPath query) into local path expressions (e.g., XPath queries) executable in local schemas.

This algorithm transforms parts of a path expression from right to left. This transformation is applied from the bottom to the top of a tree and depends on structures of local schemas.

Compared to top-down approach the bottom-up approach can be more efficient. Even in the worst case, the time complexity of this bottom up algorithm can be n times better than the top down approach.

$T(n, k, h) = \min(n, h)$

Where n is the number of parts in a global query.

In the best case, for a k-ary tree of height h, the time complexity of this algorithm is

$$T(n, k, h) = n*(k^{h+1}-1)/(k-1)$$

This can reduce to a large extent the time for forming sub queries for local (e.g., XML) schemas.

TABLE OF CONTENTS

TABLE OF CONTENTS

INTRODUCTION

CHAPTER - 1

INTRODUCTION

One of the most important challenges of Web applications is the utilization of available heterogeneous web data sources to automatically share or interoperate data. This could help users, who want to get relevant data from distributed and chaotic sources, to avoid generating these data from scratch.

However, data integration requires several steps such as follows:

(i) Creating a global schema and a set of mappings for data sharing between different sources

(ii) Resolving data conflicts among different sources

(iii) Decomposing queries of users

(iv) Optimizing these queries for efficient answering

In Global-As-View (GAV) integration systems all participating data sources follow their own schemas, which typically differ from the global schema. When users pose queries based on this global schema, these queries cannot be directly employed to query local sources due to the different structures of the global schema and the local ones.

In order to access data from these sources for further processing, the input query must be decomposed into sub queries. Each sub query Conforms to the structure of a local schema, thus it can be executed to get the relevant data.

BASICS OF "XML QUERY DECOMPOSITION"

CHAPTER - 2

BASICS OF "XML QUERY DECOMPOSITION"

2.1 HETEREOGENOUS DISTRIBUTED DATABASES:

Databases are heterogeneous due to technological differences like differences in hardware, system software, differences in DBMSs and communication system which are distributed over network. Each database contains different structure or schema. An enterprise may have multiple DBMSs. Different organizations within the enterprise may have different requirements and may select different DBMSs. These databases are located in various places. In this project the databases which are in ACCESS, EXCEL, and ORACLE have been considered and distributed over different system.

2.2 DATA INTEGRATION:

Data integration involves combining data residing in different sources and providing users with a unified view of these data. This process becomes significant in a variety of situations both commercial (when two similar companies need to merge their databases) and scientific (combining research results from different bioinformatics repositories, for example). Data integration appears with increasing frequency as the volume and the need to share existing data explodes.

Issues with combining heterogeneous data sources under a single query interface have existed for some time. The rapid adoption of databases after the 1960s naturally led to the need to share or to merge existing repositories. This merging can take place at several levels in the database architecture. One popular

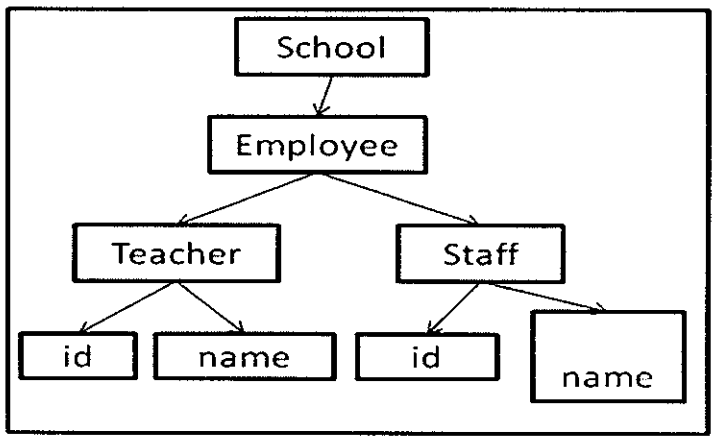
solution involves data warehousing. The warehouse system extracts, transforms, and loads data from several sources into a single queriable schema. Architecturally, this offers a tightly coupled approach because the data reside together in a single repository at query-time. Problems with tight coupling can arise with the "freshness" of data, for example when an original data source gets updated, but the warehouse still contains the older data. In order to access the updated data from distributed databases, direct query is issued to the corresponding data base from any system and it is processed in the system where the data base is located. The required data is retrieved and transmitted to the system from where the request has been received.

2.3 XML CONVERSION:

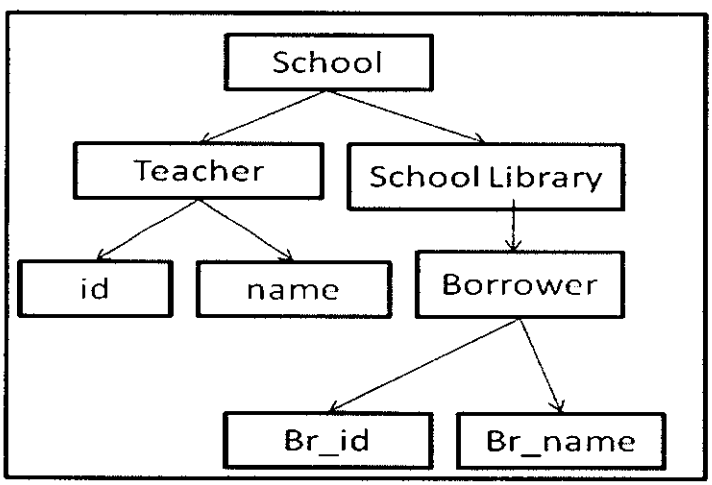
Relational databases get more and more employed in order to store the content of a web site. At the same time, XML is fast emerging as the dominant standard at the hypertext level of web site management describing pages and links between them. Thus, the integration of XML with relational database systems to enable the storage, retrieval, and update of XML documents is of major importance. Data model heterogeneity and schema heterogeneity, however, make this a challenging task.

Internally XML databases are represented as tree. Because of tree representation, searching data in a distributed database becomes easy with XML documents. So the different heterogeneous data bases are converted into XML database using the Xml converter professional edition tool. Here the methodology " XQuery" , query processing technique over distributed XML databases is used, which consists of the steps namely query decomposition, query representation, data localization, global optimization, global query execution and final result assembly.

Assume an ORACLE data base which consists of employee details who are working in a school. The tree structure or the XML document of the data base is as follows.



The tree structure of EXCEL data base which consists of school details is as follows.



2.4 SCHEMA INTEGRATION:

Data integration could not able to provide the updated data when the original data source gets updated. So schemas of several local data bases which are distributed over network are integrated and global schema is constructed. Local databases are placed in different systems. The global schema is placed in all the systems which are interconnected through some communication system. Based on the global schema the query is constructed to access data from any data base. The constructed query is called as global query. This global query has to be mapped in to query suitable to local schema. The mapped query is called as local query. This local query is executed and the corresponding data is retrieved. This query mapping process involves the construction of

- ❖ Local schema
- ❖ Global schema
- ❖ Query optimization

2.4.1 Local schema construction:

A local schema is a collection of meta-data that describes the relations in a database. A schema can be simply described as the layout of a database or the blueprint that outlines the way data is organized into tables.

The construction of local schema is done in the way representing the attributes and fields of tables in a database.

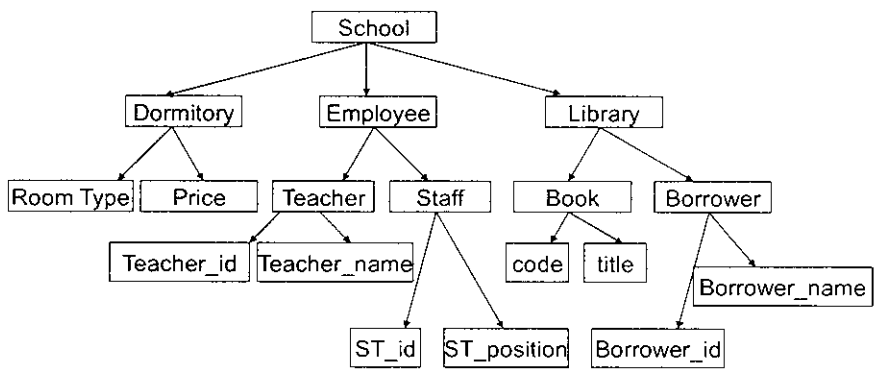
2.4.2 Global schema construction:

The global schema is defined as the global view which is formed by integrating local schemas of different data sources and the global schema (or global view) constructed to built reconcile discrepancies of different data sources.

The global schema provides a single view of all schemas which contains all metadata, so the user can confirm the availability of data and send query for required data.

The global schema of all school databases which have been already created is as follows.

Global schema



2.4 QUERY DECOMPOSITION:

The query decomposition is the process of separating elements of a query so that each decomposed queries can be processed based on the different local

databases. In bottom up approach each local schema is scanned sequentially to check whether the decomposed query can be processed based on that.

It is assumed that all conflicts are removed before converting global schema into local schema. The global query is divided into sub queries by using the query decomposition algorithm. The algorithm is as follows.

Algorithm:

Function BottomUpDecomposition(S, Qglobal)

```
Anchor: = LeftmostLeafNode;
Subquery: =";
i:=|Qglobal|;
repeat
    if Check (pi , Anchor)
    {
        if Subquery = " "
        Subquery : = pi
        else
        {
            if pi = Anchor
                Subquery : = pi + '/' +Subquery
            else
                Subquery : = pi + '//' + Subquery;
        }
    if IsRoot (pi)
        Subquery : = '/' + Subquery
```

```
else
{
if (i>1)
        % pi is not the leftmost part of Qglobal
        Anchor : = father (Pi)
else
        Subquery : = '//' + Subquery
        }
}
else
        % Pi does not exist
if (Subquery < > '') and (i=1)
        Subquery : = '//' + Subquery;
i:=i-1;
until (i=0) or (Subquery=' ') or IsRoot(pi+1);
return Subquery;
```

Global query is given as input to the algorithm. For example

"/school/employee/professor/prof_id "

Decomposed local queries are obtained as output from the algorithm. The above global query is decomposed as follows.

"/school/professor/prof_id "

"/school/employee/professor/prof_id "

ANALYSIS OF PROBLEM

CHAPTER - 3

ANALYSIS OF PROBLEM

The proposed system has to be analyzed well for its merits and demerits. Our main aim is to make use of all the merits in a particular problem and to remove the demerits of it. First the proposed system is analyzed then the merits of the system description.

The requirement specification of the proposed system should also be analyzed so that the hardware and software requirement of the system is known in order to execute our application. This helps the user in developing his/her application.

The survey of the proposed system should be made in order to find whether any software is needed to develop our application, also its been found whether there is any benefit to the system when using that particular software.

The documentation of the proposed system is made which help us to collect the details that are describing the system. And the graphical representation of the system and its activities are presented in order to make the user understand the system. Then records and description of the system elements are analyzed and maintained.

3.1 PROBLEM DEFINITION:

The purpose of this project is to provide the user required data from different databases in various locations. Normally, this decomposition is based on a set of mappings, which describe the correspondence of data elements between a global schema and local ones. However, building mappings is a difficult task, especially

when the number of participating local schemas is large. In our approach, an input query is automatically decomposed into sub queries without using mappings. An algorithm is proposed to transform a global path expression (e.g., an XPath query) into local path expressions (e.g., XPath queries) executable in local schemas. This algorithm transforms parts of a path expression from right to left. This transformation is applied from the bottom to the top of a tree and depends on structures of local schemas.

3.2 SYSTEM ANALYSIS:

The existing system has to be analyzed well for its merits and demerits. Our main aim is to make use of all its merits and to remove the demerits of it. First we must analyze the existing system and we have to find the difficulties we face by using the system. And proposed system the new system then analyze the system and find out whether it satisfies the requirements. And see the advantages of proposed system and when the fields where this particular system can be used then discuss about the merits of proposed system over the existing system. The process of analyzing a software item is to defect the differences between existing and required conditions and to evaluate the features of the software item.

3.2 .1 Existing System:

In existing approach, a user's query (e.g., an XPath query) is decomposed into sub queries with using mappings. The strategy proposed by Lausen and Marron for query decomposition with using mappings.

In the top-down strategy, the leftmost part (i.e., p1) of a global XPath query '/p1/.../pi/.../pn' is first evaluated. This evaluation is performed from the top to the bottom of the XML tree representing the local schema. This step is recursively applied to all parts of the global query from left to right (i.e., from p1 to pn).

3.2.2 Limitations in Existing system:

- The top-down query decomposition algorithm is not efficient because the worst case time complexity is more. The worst case time complexity is

  $T(n, k, h) = (k^{h+1}-1)/(k-1)$

  Where,

  n - no. of parts in global query

  h - height of the tree

  k - maximum no. of children for a node in a tree

- Duplication of Metadata exist for the database(For example metadata existed for school database which is in ACCESS is duplicated in XML).

3.2.3 Proposed system:

In a XPath query the rightmost part plays the most important role. It is the actual result, which the user wants to get from the integrated system. We need to determine whether or not a sub query exists for a specific local schema. If p_n does not exist in a local schema, we can quickly conclude that there is no sub query for this schema. Therefore, in the bottom-up strategy, we first evaluate the rightmost

part, and then sequentially proceed from the right to the left part of the input query, and from the bottom to the top of the XML tree representing the local schema. This can significantly reduce the time for searching information in XML trees.

Features of XML Query Decomposition:

- The query decomposition is performed from the bottom to the top of the XML tree representing the local schema.
- In this user given query is processed more efficiently because it takes very less time than the top down approach.
- This can be used to access any data repositories from any system connected in a network.

The proposed system is a real time solution application for access data over distributed heterogeneous database. The proposed system will reduce the time complexity to data retrieval. There are two important algorithms are implemented they are finding sub query and check existence of element collectively known as 'Query Decomposition Algorithm'.

The advantages of Bottom Up approach is

- The data can be accessed from any data source using the global query.
- This algorithm works with naming conflicts between local and global schemas, if dictionary is used.
- The data retrieval process is done in lesser time.

REQUIREMENTS

CHAPTER - 4

REQUIREMENTS

4.1 REQUIREMENT SPECIFICATIONS:

4.1.1 Hardware Requirement:

- Processor : Intel Pentium III
- CPU : 1.7 GHz.
- RAM : 1 GB
- Keyboard : Standard Keyboard with 104 Keys
- Mouse : Serial Mouse

4.1.2 Software Requirements:

- Operating System: Windows Xp, Windows7, Windows Vista.
- Front end : C# & .NET,
- Back end : Oracle, MS Office.

TESTING

CHAPTER – 5

TESTING

The testing process focuses on the logical internals of the software, ensuring that all statement have been tested, and on the functional externals; Implementation or System Testing is the stage where the theoretical Design is converted into a working System. This stage consists of the following steps.

- ❖ Making necessary changes to the system as desired by the user.
- ❖ Training the user personal. Prior to the Implementation of the stages shown below has been carried out.
- ❖ Testing the developed programs with the sample data.
- ❖ Detecting and correcting the errors.

Testing is a set of activities that can be planned in advance and conducted systematically. For this reason a template for software testing which is a set of steps into which we can place specific test design techniques and testing methods-should be defined for the software process.

A number of software testing strategies have been proposed in the literature. All provide the software developer with the template for testing and all have the following generic characteristics:

- ❖ Testing begins at the component level and works outward toward the integration of the entire computer-based system.
- ❖ Different testing techniques and appropriate at different point in time.
- ❖ The developer of the software and an independent test group conducts testing.

- ❖ Testing and debugging are different activities, but debugging must be accommodated in any testing strategy.

A strategy for software testing must accommodate low-level tests that are necessary to verify that a small source code segment has been correctly implemented as well as high-level tests validate major system functions against customer requirements. Some of the testing which is applicable for this project are tested and presented below.

5.1 UNIT TESTING:

Unit testing concentrates on each unit of the software s implemented in the source code. It focuses verification effort on the smallest unit of software design- the component or module. Using the component-level design description as a guide, important control paths are tested to uncover errors within the boundary of module.

In this testing the developed modules are tested when they are created and errors are rectified in each module. Sample data are used for testing case is taken in providing the sample data.

The module interface is tested to ensure if the information properly flows into and out of the program unit under test. The local data structure is examined to ensure that data stored temporarily maintains its integrity during all steps in an algorithm's execution. Boundary conditions are tested to ensure that the module operates at boundaries established to limit or restrict processing.

Unit testing can help during the initial development of the class or module's public interface. Unit tests force to think about how another developer will want to use the code while writing that code. This shift in focus can help to present a smaller, cleaner interface to the classes and modules. This benefit is most often

associated with test-driven development. For that in the application that is developed many modules have been used, classes for defining the methods and writing functions.

5.2 INTEGRATION TESTING:

Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing. The objective is to take unit tested components and build a program structure that has been dictated by design.

The program is constructed and tested in small increments, where errors are easier to isolate and correct, interfaces are more likely to be tested completely, and a systematic test approach may be applied. Incremental integration tests like top down integration and bottom up integration test is tested and the errors are rectified. Integration test document contains the test plan that describes the all over strategy for integration.

And here testing is divided into phases and builds that address specific functional and behavioral characteristics of the software. As module is successfully unit tested an integrated test is done to incorporate each module into overall software structure.

The integration testing done when the following modules are integrated:

- DATABASE CONSTRUCTION
- XML CONVERSION
- SCHEMA INTEGRATION
- QUERY DECOMPOSITION
- DATA RETRIEVAL

5.3 USER TESTING:

In user testing the user himself will test the software with some sample data's and check whether any errors occur and if so he will try to rectify it, and give us error free software. So the software that developed will work efficiently in all circumstances. The system group whose suggestions are incorporated top form the overall system test the developed prototype of the project.

5.4 RELIABILITY TESTING:

This is to test the reliability of the software and it's done by testing the all data sources which can work without the other data sources as an independent data source and it should produce the result of user required all other data source.

CONCLUSION AND FUTURE SCOPE

CHAPTER – 6

CONCLUSION AND FUTURE SCOPE

We have proposed a bottom-up algorithm for query decomposition without predefined mappings. The algorithm can be applied to distributed XML-based data repositories, which may contain conflicts between their respective structures. They follow a different strategy having the same motivation, but, we have proposed a more efficient query decomposition algorithm. Our contributions are as follows:

(i) a more efficient algorithm for query decomposition is proposed, that is n times better than that in the worst case and in the best case its time complexity is only $T(n,k,h)= \min(n,h)$, compared to $T(n, k, h) = n*(k^{(h+1)}-1)/ (k-1)$.

(ii) A global query is efficiently processed based on its constraints, because our algorithm can stop as soon as a local schema is found not to satisfy these constraints.

(iii) Our algorithm can work with naming conflicts between local schemas and the global one using a dictionary.

Our algorithm can also be extended to work not only with XPath queries, but also with general path expressions like those in Object- Oriented Databases. And also database fragmentation technique can be used to increase the accessing speed of the database in case of larger databases.

APPENDIX

APPENDIX 1

Coding:

Local schema construction:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Windows.Forms;
namespace WindowsFormsApplication
{
   static class Program
   {
    [STAThread]
     static void Main()
     {
      Application.EnableVisualStyles();
      Application.SetCompatibleTextRenderingDefault(false);
      Application.Run(new Form1());
     }
   }
 }


 using System;
 using System.Collections.Generic;
 using System.ComponentModel;
 using System.Data;
 using System.Drawing;
 using System.Linq;
 using System.Text;
 using System.Windows.Forms;
 using System.Xml;
 using System.IO;
 namespace WindowsFormsApplication
 {
 public partial class Form1 : Form
```

```
{
 XmlDocument xmldoc = new XmlDocument();
public Form1()
{
   InitializeComponent();
}
private void button1_Click(object sender, EventArgs e)
{
 if (!comboBox1.Items.Contains(textBox1.Text))
 {
    comboBox1.Items.Add(textBox1.Text);
 }
 String[] att = textBox2.Text.Split(new Char[] { ',' });
 XmlElement tabname = xmldoc.CreateElement(textBox1.Text);
 XmlAttribute pos = xmldoc.CreateAttribute("Position");
 pos.Value = textBox3.Text;
 tabname.SetAttributeNode(pos);
 XmlAttribute Childof = xmldoc.CreateAttribute("Childof");
 Childof.Value = comboBox1.SelectedItem.ToString();
 tabname.SetAttributeNode(Childof);
 for (int i = 0; i < att.Length; i++)
 {
    XmlElement secondelement = xmldoc.CreateElement(att[i]);
    secondelement.InnerText = "Values";
    tabname.AppendChild(secondelement);
    xmldoc.DocumentElement.InsertAfter(tabname,
    xmldoc.DocumentElement.LastChild);
 }

  FileStream fsxml = new FileStream("H:/shema4.xml", FileMode.Truncate,
 FileAccess.Write, FileShare.ReadWrite);
  xmldoc.Save(fsxml);
  textBox1.Text = "";
  textBox2.Text = "";
  textBox3.Text = "";
  comboBox1.Text = "";
}
private void Form1_Load(object sender, EventArgs e)
{
```

```
        XmlTextWriter xmlwriter = new XmlTextWriter("H:/shema4.xml",
        System.Text.Encoding.UTF8);
         xmlwriter.Formatting = Formatting.Indented;
        xmlwriter.WriteProcessingInstruction("xml", "version='1.0' encoding='UTF-
        8'");
        xmlwriter.WriteStartElement("root");
        xmlwriter.Close();
        xmldoc.Load("H:/shema4.xml");
      }
   }
}
```

Global schema construction:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Collections;
namespace xml
{
   class Dictionary
   {
      static IDictionary<String, String> Dic = new Dictionary<String, String>();
      public static void init()
      {
         Dic.Add("professor", "employee");
         Dic.Add("school_library", "Library");
         Dic.Add("student_library", "Library");
       }
      public static String getValue(String key)
      {
         Console.WriteLine("reuested String : " + key);
         String value=null;
         if (Dic.ContainsKey(key))
         {
             value= Dic[key];
         }
         return value;
```

```
    }

   }
}

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Xml;
namespace xml
{
  class XMLWriter
  {
    static  XmlTextWriter writer = new XmlTextWriter(@"H:\global.xml", null);
    public static void init()
    {
     writer.WriteStartDocument();
      writer.WriteStartElement("root");
    }
    public static void WriteStartElement(String Name)
    {
      try
      {
        writer.WriteStartElement(Name);
        Console.WriteLine("add start element  " + Name);
      }
      catch
      {
      }
    }
    public static void WriteEndElement()
    {
      try
      {
        writer.WriteEndElement();
      }
      catch
      {
```

```
            Console.WriteLine("add end element ");
        }
    }
    public static void WriteAttributes(String Name, String Value)
    {
        try
        {
            writer.WriteAttributeString(Name, Value);
        }
        catch
        {
        }
    }
    public static void Close()
    {
        writer.Close();
    }
  }
}


using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Xml;
namespace xml
{
  class Program
  {
    static int flag = 0;
    static String DictionaryFlag = null;
    static void Main(string[] args)
    {
      List<String> collection = new List<string>();
      collection.Add("root");
      String[] array = { "H:/shema1.xml", "H:/shema2.xml", "H:/shema3.xml" };
      XMLWriter.init();
      Dictionary.init();
```

```
for (int i = 0; i < array.Length; i++)
{
   String str="";
   flag = 0;
   XmlTextReader xml = new XmlTextReader(array[i]);
   while (xml.Read())
   {
      Console.WriteLine("xml name  " + xml.Name);
      switch (xml.NodeType)
      {
         case XmlNodeType.Element:
              str = xml.Name;
               if (!collection.Contains(str))
               {
                  if (str != "root")
                  {
                     if (str == "professor")
                     {
                        DictionaryFlag = str;
                        XMLWriter.WriteStartElement(str);
                        flag = 1;
                     }
                     else if (str == "school_library" || str == "student_library")
                     {
                        XMLWriter.WriteStartElement("Library");
                     }
                     else
                        XMLWriter.WriteStartElement(str);
                  }
                  xml.MoveToAttribute("Position");
                  Console.WriteLine(xml.Value);
                  if (xml.Value != "")
                  {
                     XMLWriter.WriteAttributes("Position", xml.Value);
                  }
                  xml.MoveToAttribute("Childof");
                  if (xml.Value != "")
                  {
                     if (flag == 1)
```

```
                    {
                        String child = Dictionary.getValue(DictionaryFlag);
                        Console.WriteLine("child node " + child);
                        XMLWriter.WriteAttributes("Childof", child);
                        flag = 0;
                    }
            else if (xml.Value == "school_library" || xml.Value == "student_library")
              {
                XMLWriter.WriteAttributes("Childof", "Library");
              }
            else
              XMLWriter.WriteAttributes("Childof", xml.Value);
          }
          else
           {
            Console.WriteLine(xml.Value);
           }
        Console.ReadLine();
        }
        break;
case XmlNodeType.EndElement:
                    if(xml.Name!="root")
                    XMLWriter.WriteEndElement();
                    break;
 case XmlNodeType.None:
                     break;
case XmlNodeType.XmlDeclaration:
                     break;
          }
        }
        xml.Close();
      }
XMLWriter.Close();
    }
  }
}
```

Server program :(Decomposition)

Program.cs:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Net;
using System.Net.Sockets;
using System.Xml;
using System.Xml.XPath;
namespace server
{
 class Program
 {
  static void Main(string[] args)
  {
   try
   {
    IPAddress ipAd = IPAddress.Parse("192.168.0.174");
   TcpListener myList = new TcpListener(ipAd, 8001);
   myList.Start();
   Console.WriteLine("The server is running at port 8001...");
   Console.WriteLine("The local End point is  :" + myList.LocalEndpoint);
   Console.WriteLine("Waiting for a connection.....");
   Socket s = myList.AcceptSocket();
   Console.WriteLine("Connection accepted from " + s.RemoteEndPoint);
   byte[] b = new byte[100];
   int k = s.Receive(b);
   Console.WriteLine("Recieved...");
   String recdata;
   String str = System.Text.ASCIIEncoding.ASCII.GetString(b, 0, k);
   Console.WriteLine(str.TrimEnd() + " received data");
```

```
        ASCIIEncoding asen = new ASCIIEncoding();
        String querres=Result.GetResult(str);
        s.Send(asen.GetBytes(querres));
       Console.ReadLine();
       s.Close();
       myList.Stop();
      }
       catch (Exception e)
       Console.WriteLine("Error..... " + e.StackTrace);
     }
   }
  }
}
```

Result.cs:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Xml;
using System.Xml.XPath;
namespace server
{
 class Result
 {
  public static String GetResult(String query1)
  {
   String result = null;
   String[] stringarr = query1.Split(new char[] { '/' });
   String[] array = { "c:/shema2.xml" };
   String[] file = { "c:/localschema2.xml"};
   String query = null;
   for (int i = 0; i < array.Length; i++)
   {
    query = "";
    String queryflag = "";
    int Flag = 0;
    for (int j = stringarr.Length - 1; j > 0; j--)
```

```
{
 XmlTextReader xml = new XmlTextReader(array[i]);
 while (xml.Read())
 {
  switch (xml.NodeType)
  {
  case XmlNodeType.Element:
  if (Flag == 0)
  {
   if (xml.Name.Equals(stringarr[stringarr.Length - 1]))
   {
   query = query + xml.Name + "/";
   Flag = 1;
   }
  }
  else if (Flag == 1)
  {
  if (xml.Name.Equals(stringarr[stringarr.Length - 2]))
   {
  if (xml.HasAttributes)
   {
   queryflag = xml.GetAttribute(1).ToString();
  query = query + xml.Name + "/";
   Flag = 2;
   }
  }
}
   else if (Flag == 2)
   {
   if (xml.Name.Equals(queryflag))
    {
     if (xml.HasAttributes)
     {
     Queryflag = xml.GetAttribute(1).ToString();
     query = query + xml.Name + "/";
     }
     }
  }
 break;
```

```
        }
      }
    }
    if (query != "")
    query += "root";
    String[] data = query.Split(new char[] { '/' });
    query = "";
    for (int k = data.Length - 1; k >= 0; k--)
    query = query + "/" + data[k];
    if (!query.Equals("/"))
    {
      result = GetElements(GetQuery.getXpathQuery(query), file[i]);
      Console.WriteLine("Result of query" + result);
    }
  }
  return result;
}
public static String GetElements(string expression, String fileName)
 {
     String res=null;
     XPathDocument doc = new XPathDocument(fileName);
     XPathNavigator nav = doc.CreateNavigator();
     XPathExpression expr;
     expr = nav.Compile(expression);
     XPathNodeIterator iterator = nav.Select(expr);
    try
     {
        while (iterator.MoveNext())
        {
           XPathNavigator nav2 = iterator.Current.Clone();
           res+=nav2.Value + "\n";
        }
     }
     catch (Exception ex)
     {
        Console.WriteLine(ex.Message);
     }
     return res;
}
```

```
 }
}
```

Getquery.cs:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
namespace server
{
 class GetQuery
 {
 static String str;
  public static String getXpathQuery(String query)
  {
   String[] data = query.Split(new char[] { '/' });
      str = "/" + data[1] + "/" + data[data.Length - 2] + "/" +
      data[data.Length -1];
      Console.WriteLine(str);
      return str;
  }
 }
}
```

Client Program (Decomposition):

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Net;
using System.IO;
using System.Net.Sockets;
namespace client
{
```

```
class Program
{
static void Main(string[] args)
{
try
{
   TcpClient tcpclnt = new TcpClient();
   Console.WriteLine("Connecting.....");
   tcpclnt.Connect("192.168.0.174", 8001);
   Console.WriteLine("Connected");
   Console.Write("Enter the string to be transmitted : ");
   String str = Console.ReadLine();
   Stream stm = tcpclnt.GetStream();
   ASCIIEncoding asen = new ASCIIEncoding();
   byte[] ba = asen.GetBytes(str);
   Console.WriteLine("Transmitting.....");
   stm.Write(ba, 0, ba.Length);
   byte[] bb = new byte[1024];
   int k = stm.Read(bb, 0, 1024);
   for (int i = 0; i < k; i++)
   Console.Write(Convert.ToChar(bb[i]));
   Console.ReadLine();
   tcpclnt.Close();

   TcpClient tcpclnt1 = new TcpClient();
   Console.WriteLine("Connecting.....");
  tcpclnt1.Connect("192.168.0.1", 5001);
  Console.WriteLine("Connected");
  Console.Write("Enter the string to be transmitted : ");
   Stream stm1 = tcpclnt1.GetStream();
   byte[] ba1 = asen.GetBytes(str);
   Console.WriteLine("Transmitting.....");
   stm1.Write(ba1, 0, ba1.Length);
   byte[] bb1 = new byte[1024];
    k = stm1.Read(bb1, 0, 1024);
   for (int i = 0; i < k; i++)
    Console.Write(Convert.ToChar(bb[i]));
   Console.ReadLine();
   tcpclnt1.Close();
```

```
        TcpClient tcpclnt2 = new TcpClient();
        Console.WriteLine("Connecting.....");
        tcpclnt2.Connect("169.254.247.219", 4001);
        Console.WriteLine("Connected");
        Console.Write("Enter the string to be transmitted : ");

        Stream stm2 = tcpclnt2.GetStream();
        byte[] ba2 = asen.GetBytes(str);
        Console.WriteLine("Transmitting.....");
        stm2.Write(ba2, 0, ba2.Length);
        byte[] bb2 = new byte[1024];
         k = stm2.Read(bb2, 0, 1024);
        for (int i = 0; i < k; i++)
           Console.Write(Convert.ToChar(bb2[i]));
        Console.ReadLine();
        tcpclnt2.Close();
      }
      catch (Exception e)
      {
        Console.WriteLine("Error..... " + e.StackTrace);
      }
    }
  }
}
```

APPENDIX 2

Snap shots

Access Database

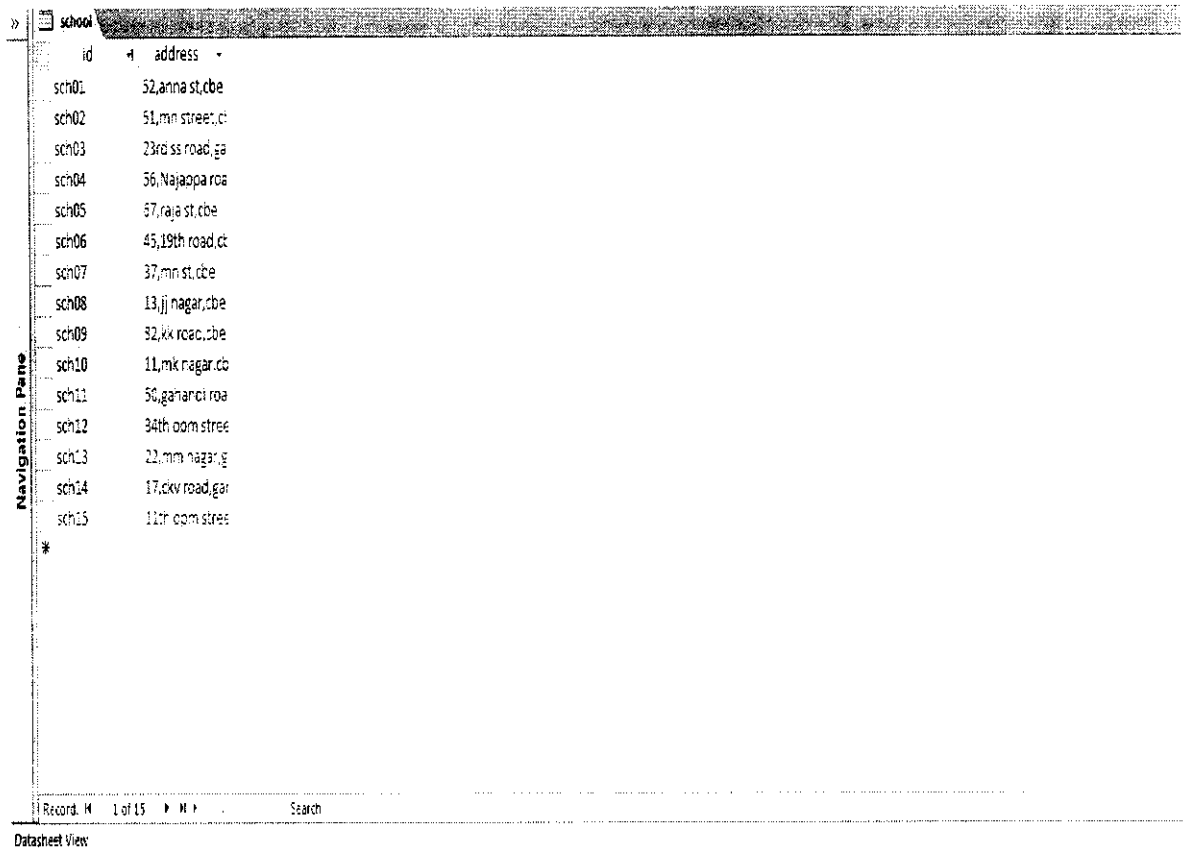


Fig:1(a): school table for Access Database

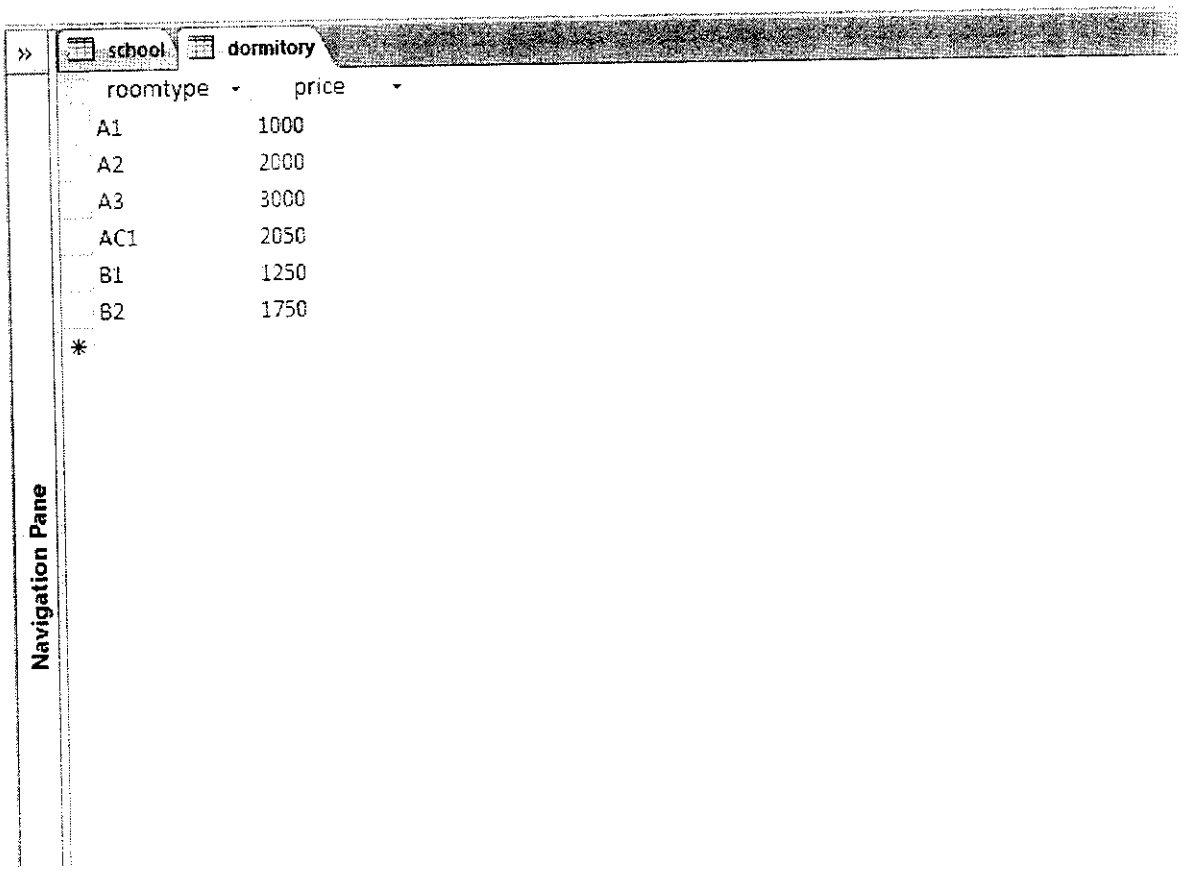


Fig:1(b): Dormitory table for Access Database

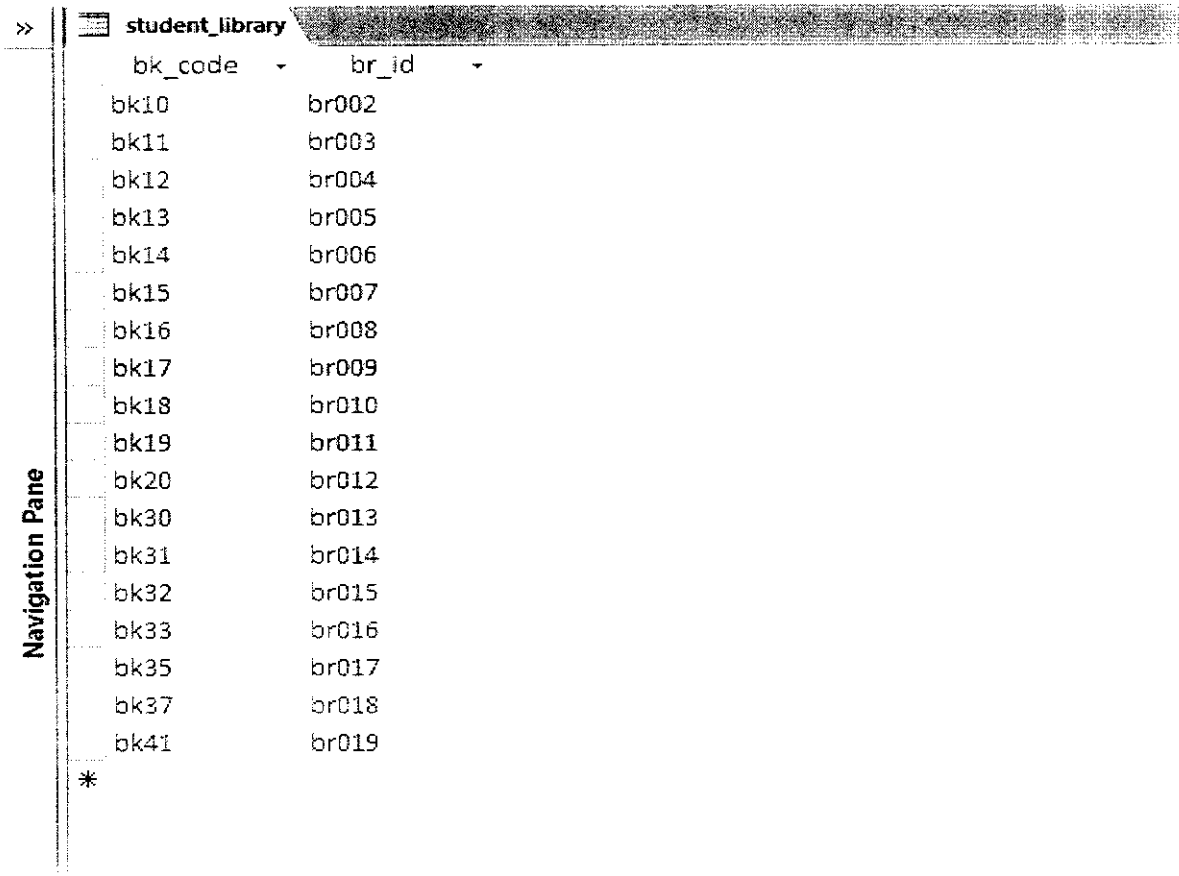


Fig:1(c): student library table for Access Database

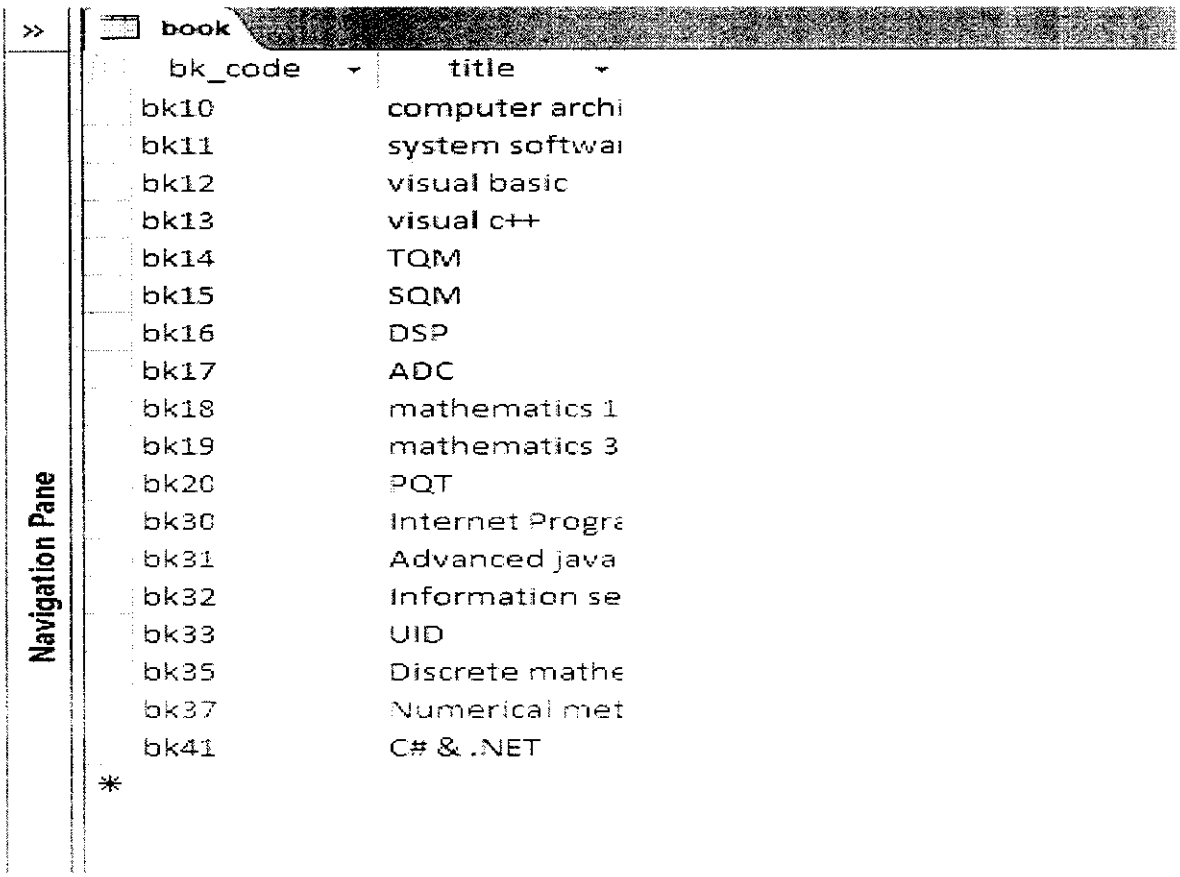
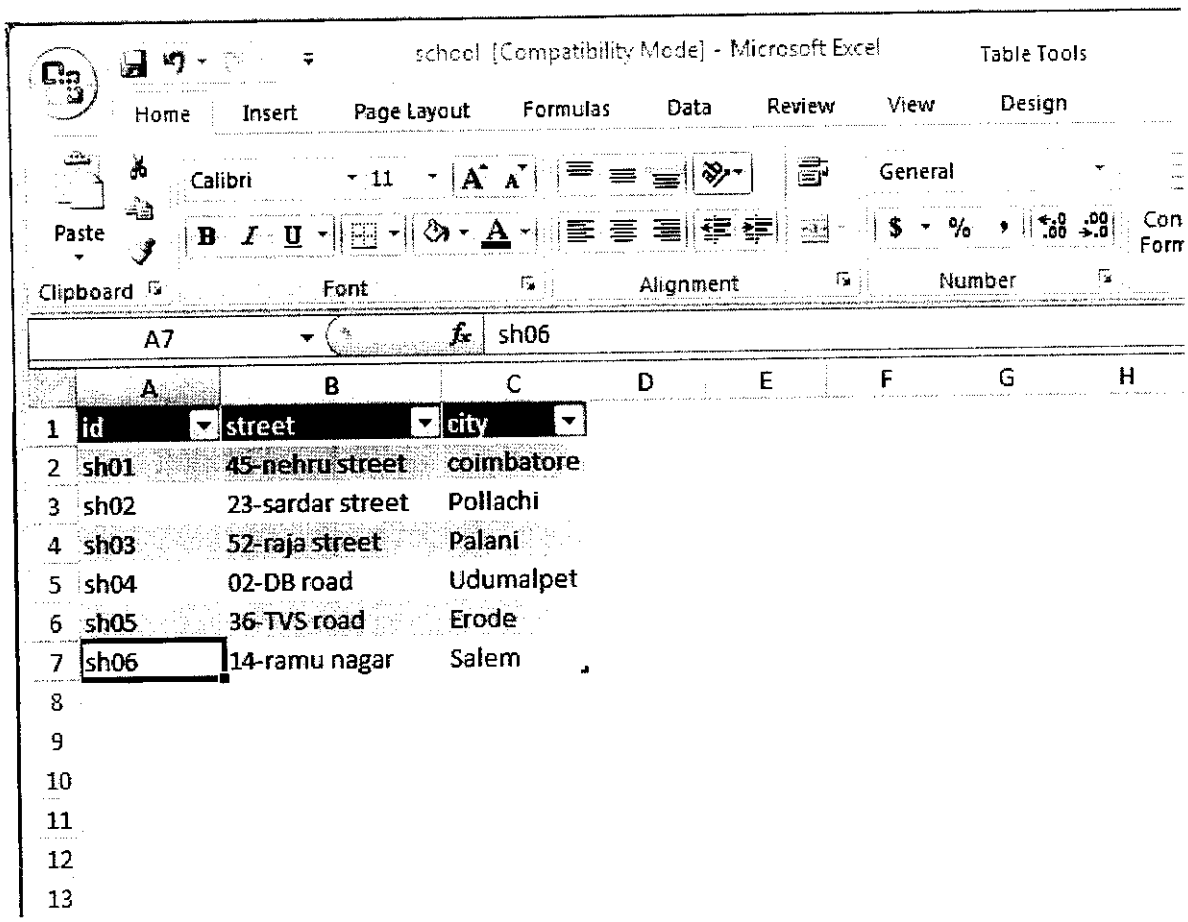


Fig:1(d): book table for Access Database

Excel Database



	A	B	C	D	E	F	G	H
1	id	street	city					
2	sh01	45-nehru street	coimbatore					
3	sh02	23-sardar street	Pollachi					
4	sh03	52-raja street	Palani					
5	sh04	02-DB road	Udumalpet					
6	sh05	36-TVS road	Erode					
7	sh06	14-ramu nagar	Salem					
8								
9								
10								
11								
12								
13								

Fig:2(a): school table for Excel Database

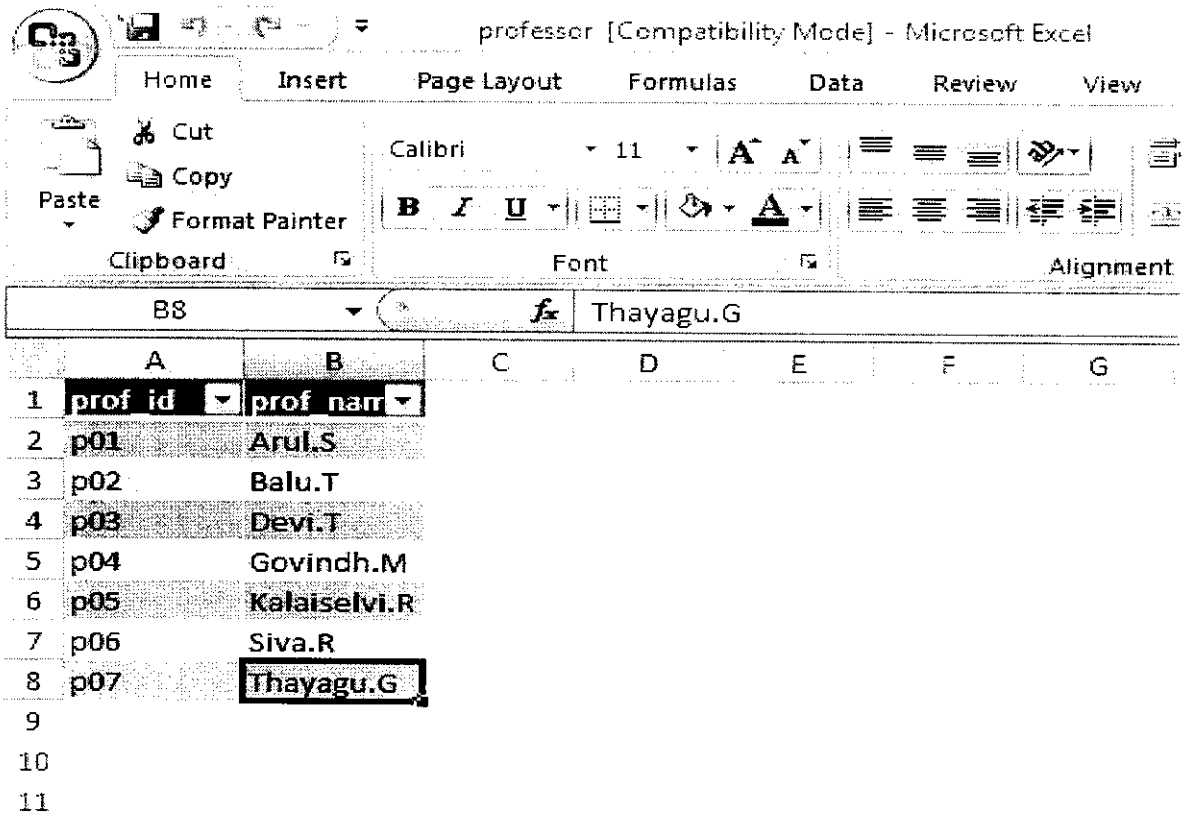


Fig:2(b): professor table for Excel Database

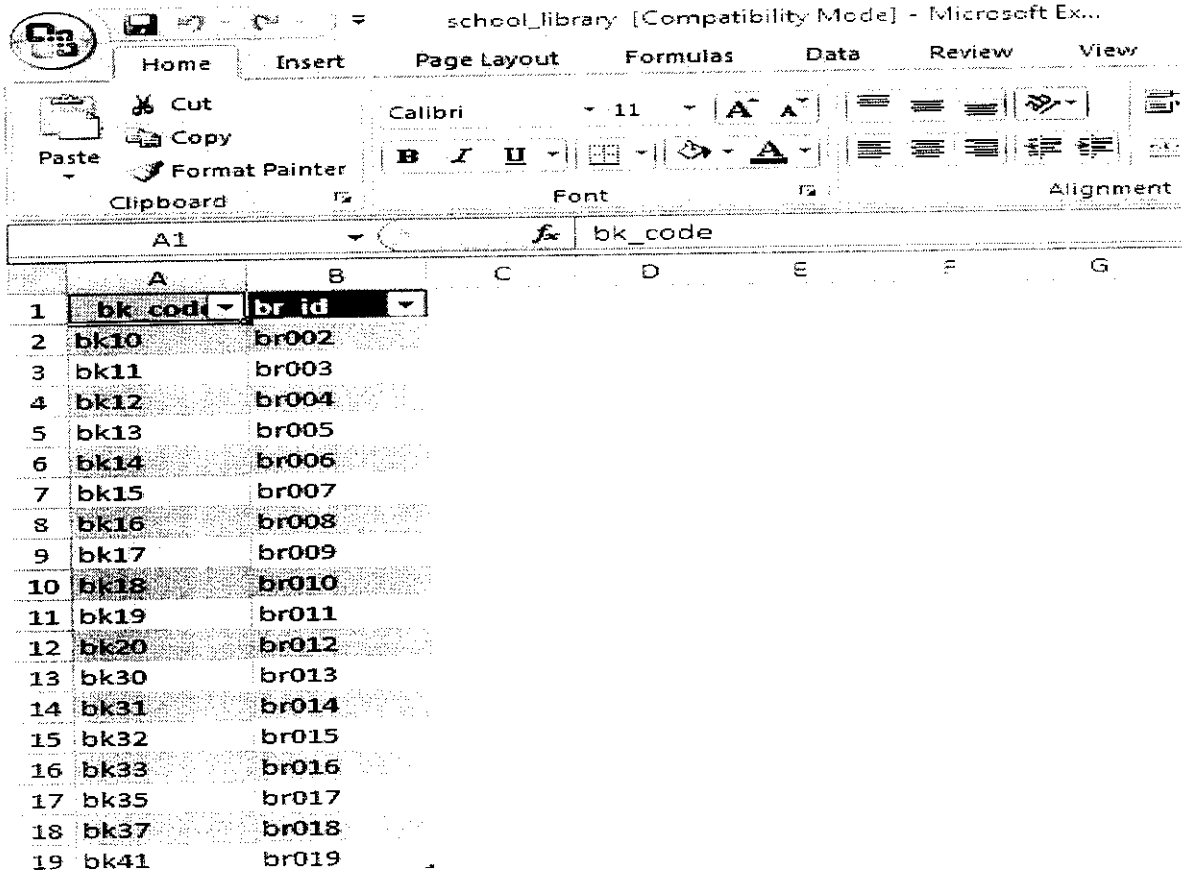


Fig:2(c): school library table for Excel Database

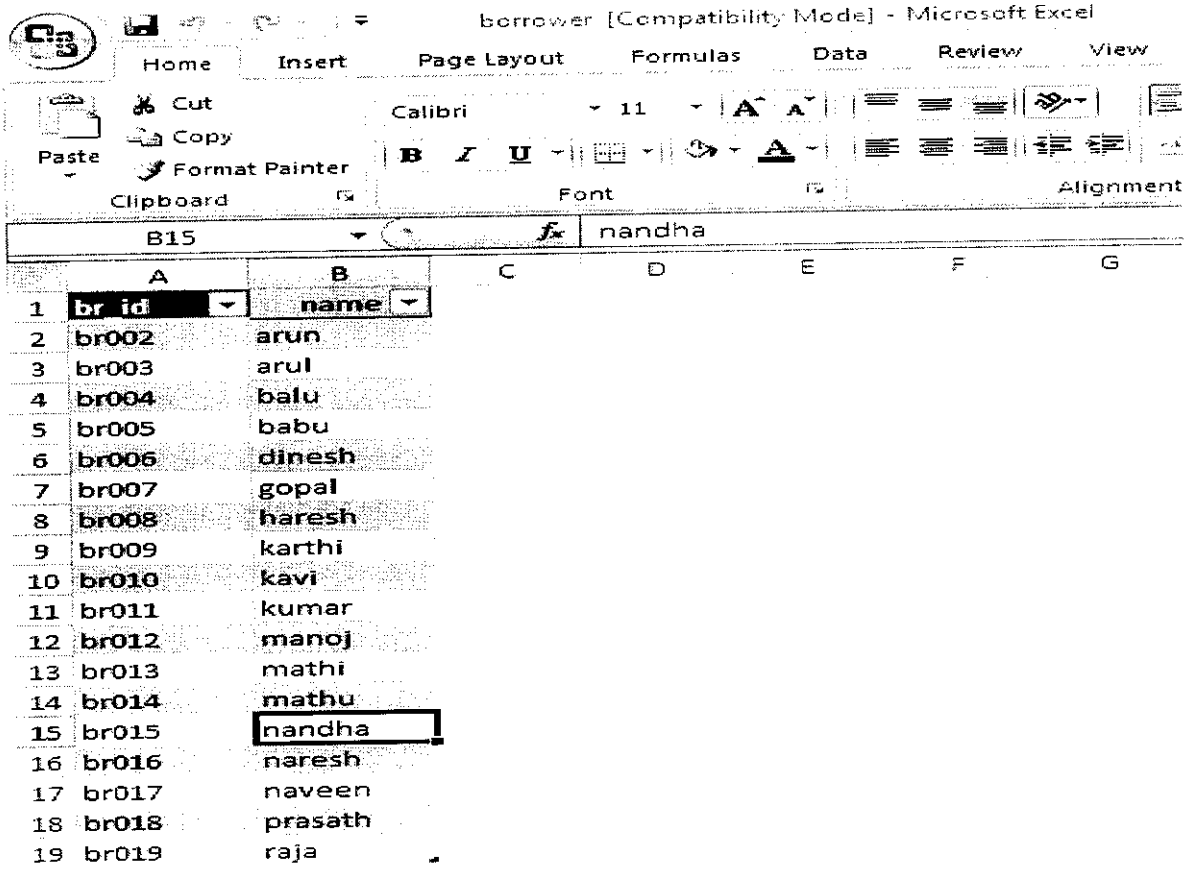


Fig:2(d): borrower table for Excel Database

Access Database to XML Database

File Edit Format View Help

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<?xml-stylesheet type="text/xsl" href="schema1.mdb.xsl"?>
<!-- This XML file is generated by RustemSoft XML Converter DEMO version -->
<table name="NewDataSet">
  <school>
    <id>sch01</id>
    <address>52,anna st,cbe</addresss>
  </school>
  <school>
    <id>sch02</id>
    <address>61,mn street,cbe</address>
  </school>
  <school>
    <id>scho3</id>
    <address>23rd ss road,ganapati,cbe</address>
  </school>
  <school>
    <id>sch04</id>
    <address>56,Najappa road,cbe</address>
  </school>
  <school>
    <id>sch05</id>
    <address>67,raja st,cbe</address>
  </school>
  <school>
    <id>sch06</id>
    <address>45,19th road,cbe</address>
  </school>
  <school>
    <id>sch07</id>
    <address>37,mn st,cbe</address>
  </school>
  <school>
    <id>sch08</id>
    <address>13,jj nagar,cbe</address>
  </school>
```

Fig:3(a): school table in XML

File Edit Format View Help

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<?xml-stylesheet type="text/xsl" href="schema1.mdb.xsl"?>
<!-- This XML file is generated by RustemSoft XML Converter DEMO version -->
<table name="NewDataSet">
   <dormitory>
    <roomtype>A1</roomtype>
    <price>1000</price>
  </dormitory>
  <dormitory>
    <roomtype>A2</roomtype>
    <price>2000</price>
  </dormitory>
  <dormitory>
    <roomtype>A3</roomtype>
    <price>3000</price>
  </dormitory>
  <dormitory>
    <roomtype>B1</roomtype>
    <price>1250</price>
  </dormitory>
  <dormitory>
    <roomtype>B2</roomtype>
    <price>1750</price>
  </dormitory>
  <dormitory>
    <roomtype>AC1</roomtype>
    <price>2050</price>
  </dormitory>
</table>
```

Fig:3(b): Dormintory table in XML

File Edit Format View Help

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<?xml-stylesheet type="text/xsl" href="schema1.mdb.xsl"?>
<!-- This XML file is generated by RustemSoft XML Converter DEMO version -->
<table name="NewDataSet">
  <student_library>
    <bk_code>bk10</bk_code>
    <br_id>br002</br_id>
  </student_library>
  <student_library>
    <bk_code>bk11</bk_code>
    <br_id>br003</br_id>
  </student_library>
  <student_library>
    <bk_code>bk12</bk_code>
    <br_id>br004</br_id>
  </student_library>
  <student_library>
    <bk_code>bk13</bk_code>
    <br_id>br005</br_id>
  </student_library>
  <student_library>
    <bk_code>bk14</bk_code>
    <br_id>br006</br_id>
  </student_library>
  <student_library>
    <bk_code>bk15</bk_code>
    <br_id>br007</br_id>
  </student_library>
  <student_library>
    <bk_code>bk16</bk_code>
    <br_id>br008</br_id>
  </student_library>
  <student_library>
    <bk_code>bk17</bk_code>
    <br_id>br009</br_id>
```

Fig:3(c): student library table in XML

File Edit Format View Help

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<?xml-stylesheet type="text/xsl" href="schema1.mdb.xsl"?>
<!-- This XML file is generated by RustemSoft XML Converter DEMO version -->
<table name="NewDataSet">
  <Book>
    <code>bk10</code>
    <title>computer architechture</title>
  </Book>
  <Book>
    <code>bk11</code>
    <title>system software</title>
  </Book>
  <Book>
    <code>bk12</code>
    <title>visual basic</title>
  </Book>
  <Book>
    <code>bk13</code>
    <title>visual c++</title>
  </Book>
  <Book>
    <code>bk14</code>
    <title>TQM</title>
  </Book>
  <Book>
    <code>bk15</code>
    <title>SQM</title>
  </Book>
  <Book>
    <code>bk16</code>
    <title>DSP</title>
  </Book>
  <Book>
    <code>bk17</code>
    <title>ADC</title>
```

Fig:3(d): Book table in XML

Excel Database to XML Database

school - Notepad

File Edit Format View Help

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<?xml-stylesheet type="text/xsl" href="school.xls.xsl"?>
<!-- This XML file is generated by RustemSoft XML Converter Evaluation version -->
<table name="NewDataSet">
  <school>
    <id>sh01</id>
    <street>45-nehru street</street>
    <city>coimbatore</city>
  </school>
  <school>
    <id>sh02</id>
    <street>23-sardar street</street>
    <city>Pollachi</city>
  </school>
  <school>
    <id>sh03</id>
    <street>52-raja street</street>
    <city>Palani</city>
  </school>
  <school>
    <id>sh04</id>
    <street>02-DB road</street>
    <city>Udumalpet</city>
  </school>
  <school>
    <id>sh05</id>
    <street>36-TVS road</street>
```

Fig:4(a): school table in XML

File Edit Format View Help

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<?xml-stylesheet type="text/xsl" href="professor.xls.xsl"?>
<!-- This XML file is generated by RustemSoft XML Converter Evaluation version -->
<table name="NewDataSet">
  <professor>
    <prof_id>p01</prof_id>
    <prof_name>Arul.S</prof_name>
  </professor>
  <professor>
    <prof_id>p02</prof_id>
    <prof_name>Balu.T</prof_name>
  </professor>
  <professor>
    <prof_id>p03</prof_id>
    <prof_name>Devi.T</prof_name>
  </professor>
  <professor>
    <prof_id>p04</prof_id>
    <prof_name>Govindh.M</prof_name>
  </professor>
  <professor>
    <prof_id>p05</prof_id>
    <prof_name>Kalaiselvi.R</prof_name>
  </professor>
  <professor>
    <prof_id>p06</prof_id>
    <prof_name>Siva.R</prof_name>
  </professor>
  <professor>
    <prof_id>p07</prof_id>
    <prof_name>Thayagu.G</prof_name>
  </professor>
</table>
```

Fig:4(b): professor table in XML

File Edit Format View Help

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<?xml-stylesheet type="text/xsl" href="school_library.xls.xsl"?>
<!-- This XML file is generated by RustemSoft XML Converter Evaluation version -->
<table name="NewDataSet">
<school_library>
    <bk_code>bk10</bk_code>
    <br_id>br002</br_id>
  </school_library>
  <school_library>
    <bk_code>bk11</bk_code>
    <br_id>br003</br_id>
  </school_library>
  <school_library>
    <bk_code>bk12</bk_code>
    <br_id>br004</br_id>
  </school_library>
  <school_library>
    <bk_code>bk13</bk_code>
    <br_id>br005</br_id>
  </school_library>
  <school_library>
    <bk_code>bk14</bk_code>
    <br_id>br006</br_id>
  </school_library>
  <school_library>
    <bk_code>bk15</bk_code>
    <br_id>br007</br_id>
  </school_library>
  <school_library>
    <bk_code>bk16</bk_code>
    <br_id>br008</br_id>
  </school_library>
  <school_library>
    <bk_code>bk17</bk_code>
    <br_id>br009</br_id>
```

Fig:4(c): school library table in XML

File Edit Format View Help

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<?xml-stylesheet type="text/xsl" href="borrower.xls.xsl"?>
<!-- This XML file is generated by RustemSoft XML Converter Evaluation version -->
<table name="NewDataSet">
  <borrower>
    <br_id>br002</br_id>
    <name>arun</name>
  </borrower>
  <borrower>
    <br_id>br003</br_id>
    <name>arul</name>
  </borrower>
  <borrower>
    <br_id>br004</br_id>
    <name>balu</name>
  </borrower>
  <borrower>
    <name>babu</name>
    <br_id>br005</br_id>
  </borrower>
  <borrower>
    <br_id>br006</br_id>
    <name>dinesh</name>
  </borrower>
  <borrower>
    <br_id>br007</br_id>
    <name>gopal</name>
  </borrower>
  <borrower>
    <br_id>br008</br_id>
    <name>haresh</name>
  </borrower>
  <borrower>
    <br_id>br009</br_id>
    <name>karthi</name>
```

Fig:4(d): borrower table in XML

Oracle Database to XML Database

school - Notepad

File Edit Format View Help

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<?xml-stylesheet type="text/xsl" href=".SYSTEM.xsl"?>
<!-- This XML file is generated by RustemSoft XML Converter Evaluation version -->
<table name="NewDataSet">
 <school>
  <id>s01</id>
  <address>45-nehru st,coimbatore</address>
 </school>
 <school>
  <id>s02</id>
  <address>25-DB road,coimbatore</address>
 </school>
 <school>
  <id>s03</id>
  <address>64-TVS road,pollachi</address>
 </school>
 <school>
  <id>s04</id>
  <address>22-raja st,udumalpet</address>
 </school>
 <school>
  <id>s05</id>
  <address>48-ansari st,pollachi</address>
 </school>
 <school>
  <id>s06</id>
  <address>68-MGR nager,coimbatore</address>
 </school>
```

Fig:5(a): school table in XML

employee - Notepad

File Edit Format View Help

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<?xml-stylesheet type="text/xsl" href=".SYSTEM.xsl"?>
<!-- This XML file is generated by RustemSoft XML Converter Evaluation version -->
<table name="NewDataSet">
 <employee>
  <prof_id>pr01</prof_id>
  <staff_id>st01</staff_id>
 </employee>
 <employee>
  <prof_id>pr02</prof_id>
  <staff_id>st02</staff_id>
 </employee>
 <employee>
  <prof_id>pr03</prof_id>
  <staff_id>st03</staff_id>
 </employee>
 <employee>
  <prof_id>pr04</prof_id>
  <staff_id>st04</staff_id>
 </employee>
 <employee>
  <prof_id>pr05</prof_id>
  <staff_id>st05</staff_id>
 </employee>
 <employee>
  <prof_id>pr06</prof_id>
  <staff_id>st06</staff_id>
 </employee>
```

Fig:5(b): employee table in XML

professor - Notepad

File Edit Format View Help

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<?xml-stylesheet type="text/xsl" href=".SYSTEM.xsl"?>
<!-- This XML file is generated by RustemSoft XML Converter Evaluation version -->
<table name="NewDataSet">
  <professor>
    <prof_id>pr01</prof_id>
    <name>k.Arul</name>
  </professor>
  <professor>
    <prof_id>pr02</prof_id>
    <name>G.Bala</name>
  </professor>
  <professor>
    <prof_id>pr03</prof_id>
    <name>S.Sujatha</name>
  </professor>
  <professor>
    <prof_id>pr04</prof_id>
    <name>K.Karthi</name>
  </professor>
  <professor>
    <prof_id>pr05</prof_id>
    <name>T.Thangu</name>
  </professor>
  <professor>
    <prof_id>pr06</prof_id>
    <name>G.Gayathri</name>
  </professor>
```

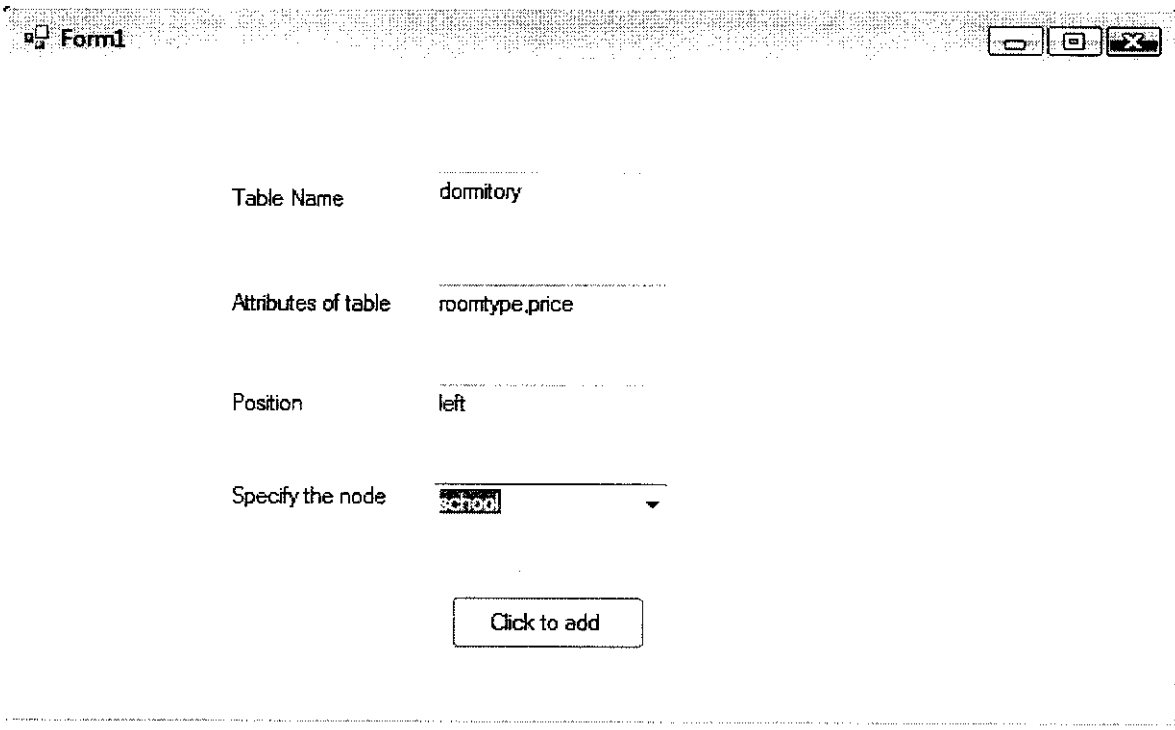
Fig:5(c): professor table in XML

File Edit Format View Help

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<?xml-stylesheet type="text/xsl" href=".SYSTEM.xsl"?>
<!-- This XML file is generated by RustemSoft XML Converter Evaluation version -->
<table name="NewDataSet">
  <staff>
    <staff_id>st01</staff_id>
    <position>Asst.Head Master</position>
  </staff>
  <staff>
    <staff_id>st02</staff_id>
    <position>Head Master</position>
  </staff>
  <staff>
    <staff_id>st03</staff_id>
    <position>Teacher</position>
  </staff>
  <staff>
    <staff_id>st04</staff_id>
    <position>Teacher</position>
  </staff>
  <staff>
    <staff_id>st05</staff_id>
    <position>Teacher</position>
  </staff>
  <staff>
    <staff_id>st06</staff_id>
    <position>Teacher</position>
  </staff>
  <staff>
    <staff_id>st07</staff_id>
    <position>Teacher</position>
  </staff>
</table>
```

Fig:5(d): staff table in XML

LOCAL SCHEMA:



GLOBAL SCHEMA:

H:\global.xml

```
<?xml version="1.0" ?>
- <root>
  - <school Position="left" Childof="root">
      <id />
      <address />
    </school>
  - <dormitory Position="left" Childof="school">
      <roomtype />
      <price />
    </dormitory>
  - <Library Position="right" Childof="school">
      <bk_code />
      <br_id />
    </Library>
  - <book Position="left" Childof="Library">
      <bk_code />
      <title />
    </book>
  - <school Position="left" Childof="root">
      <id />
      <city />
      <street />
    </school>
  - <professor Position="left" Childof="employee">
      <prof_id />
      <prof_name />
    </professor>
  - <Library Position="right" Childof="school">
      <bk_code />
```

OUTPUT:


file:///C:/Users/nanys/Desktop/new/lan/client/client/bin/Debug/client.EXE
Connecting.....
Connected
Enter the string to be transmitted : /root/select/fi
Transmitting.....
sh01
sh02
sh03
sh04
sh05
sh06
Connecting.....
Connected
Enter the string to be transmitted : Transmitting.....
s01
s02
s03
s04
s05
s06
Connecting.....
Connected
Enter the string to be transmitted : Transmitting.....
sch01
sch02
scho3
sch04
sch05
sch06
sch07
sch08

REFERENCES:

[1] Baru.C, Gupta.A, Ludaescher,B. Marciano.R, Papakonstantinou, Y. and Velikhov.P, "XML-Based Information Mediation with MIX", In Demo Session. ACM-SIGMOD'99, Philadelphia, PA, 1999.

[2] Baru, C, Ludaescher.B, Papakonstantinou.Y, Velikhov.P, and Vianu.V, "Features and Requirements for an XML View Definition Language: Lessons from XML Information Mediation", W3C's Query Language Workshop, 1998.

[3] Bi.Y and Lamb.J, "Facilitating Integration of Distributed Statistical Databases Using Metadata and XML" , Available online: http://webfarm.jrc.cec.eu.int/ETK-NTTS/Papers/final_papers/en187.pdf, 2001. Last accessed on September 8, 2006.

[4] Duong.D.D and Thuy L.T.T "Classification and Reconcilement of Conflicts between Heterogeneous XML Schemas", Proceeding of the 10th Conference on Artificial Intelligence and Applications, TAAI, 2005.

[5] Duong.D.D and Wuwongse .V, "XML Database Schema Integration Using XDD", Proceedings of Advances in Web-Age Information Management Conference, China: Lecture Notes in Computer Science, Springer Verlag, vol. 2762, 2003, pp. 92-103.

[6] Gianolli .P and Mylopoulos .J, "A semantic approach to XML based data integration", Proceedings of the 20th. International Conference on Conceptual Modeling (ER), Yokohama, Japan, 2001.

[7] Lausen .G and Marron. P.J, "Adaptive evaluation techniques for querying XML-based E-Catalogs", DBLP, 2002, pp. 19-28.

[8] Ludascher, R. Himmeroder, G. Lausen, W. May and C. Schlepphorst "Managing semistructured data with FLORID: a Deductive object- oriented perspective", Journal of Information Systems, vol. 23, no. 8, 1998, pp. 1-25.

[9] May.W, "Logic-based XML data integration: a semi-materializing approach", Journal of Applied Logic, vol. 3, No. 1, 2005, pp. 271–307.

[10] The MIX (Mediator of Information using XML). Available online at: http://db.ucsd.edu/Projects/MIX/, 1999. Last accessed on September 8, 2006.

[11] Thuy.L.T.T and Duong. D.D, "Query Decomposition Using the XML Declarative Description Language", Proceedings of International Conference of Computational Science and Its Applications, Singapore: Lecture Notes in Computer Science, Springer Verlag, vol. 3481, 2005,
pp. 1066-1075.

[12] Thuy L.T.T. and Duong D.D., "Integration of XML Databases", The Journal of Hue University – Vietnam, vol. 22, 2004, pp 45-52.

[13] Thuy L.T.T. and Wuwongse V. "Query Processing of Integrated XML Databases", Proceedings of the 5th International Conference on Information Integration and Web based Applications & Services, Jakarta, Indonesia, 2003, pp. 335-344.

[14] XML Path Language (XPath). Available online at:
http://www.w3.org/TR/xpath. Last accessed on September 8, 2006

WEBSITES:

1. **www.w3schools.com**
2. **www.c-sharpcorner.com**
3. **www.ww3.org**
4. **www.codeproject.com**
5. **www.rustemsoft.com**
6. **www.codeguru.com**