APPROACH FOR QUERY IN HETEROGENEOUS DATABASES

Mohammed Qasim Yaseen[1]

[1]Iraq, South Oil Company

Abstract

Developed in the information Tec., data management and multi-database, cloud computing that introduce problem in data integration. The one of data integration problem in database is heterogeneous database query. In the end how can join and representation data from multi-database although the differences in DBMS, Data Structure, and data type.

In this research built set of algorithm to retrieval and integration data from databases such as (Oracle, SQL Server) by create virtual tables and join its to solution the heterogeneous in DBMS, Data Structure, and data type problem. system create new scheme for the tables (columns use in query) without all columns in database, after apply condition (on each table's database). This system rather than design a new database with new structure and new virtual programming for coordinated to solve integration data. the system can use also to re-fragmentation tables and re-arrangement fields in tables.

***Keywords:** -federation Databases, Distributed database, Homogenous and Heterogeneous database, Data Integration*

--***--

1. INTRODUCTION

The Distributed database (DDB) means two or more of database in different location share data between them. Database integration is a multistep process of finding similar entities in two or more databases to create a non-redundant [1]. The DDB may be homogeneity or heterogeneous, the homogeneity index to all database that share data in different locationsare symmetric in DBMS (represents in Query Processing software, storage Management software) and Database contains (Data type and structure of the table)[2]. The homogeneity one or more databases are asymmetric in DBMS or Database contains also call it if have databases asymmetric in data conceptual that save in tables.

When retrieval data from multi-database in homogeneity achieve without problem when join tables, but in heterogeneous we have problem in integration data because differences may be in table's structure, data type of join key, or DBMS.In this field was introduced many papers such as e.g:Mohammad Ghulam Ali ,2009use object-oriented approach to integrate schemas of distributed heterogeneous autonomous local component database schemas into a global schema. The resulting global schema provides a uniform interface and high level of location transparency for retrieval of data from the local component databases. A set of integration operators are defined to integrate local schemas based on the semantic relevance of their classes and to provide a model independent representation of virtual classes of the global schema [3]. Andrew, Tamas, et. al., 2008: Integrated Data Model plus links to/from ontologies to homogenize biomedical (from disease, patient and population-related) . Integrated Project that aims to develop a grid-based integrated healthcare platform for paediatrics. It is hoped that using this platform biomedical information will integrate heterogeneous data and perform epidemiological studies across Europe. [4]. Ibrahim, Naomie. 2013: This paper discusses the role of data integration and the fields that require it. paper written in this field, it discusses the techniques, strategies, approach and tools in the field of schema matching. Once matching is finished, the relations and correspondences between the various schemas have been identified. Next phase is to create the Global Schema and this is referred to as schema integration [1]

2. DATA INTEGRATION FROM HETEROGENEOUS DATABASES AND PURPOSE SYSTEM

The integration of heterogeneous data sources can be made through a variety of ways such as Federation database system, it is collection of cooperating but autonomous component database system the component DBSs are integrated to various degrees, federated database systems include uniform interface and user can retrieval data from databases with single query[5].

data Integration from databases in proposed system by apply sequence step start with open connection database until display result, some step need user to input and select data through system interface through interface user select tables that need in query, include join and select condition.

Use visual basic.net to program all algorithm, interface and create the code that use to build federation database. query apply on two step first step apply on local database second step apply on global database. Excelled system when creating virtual tables by add the fields that have been selected from the original table only and it ignores fields that don’t select, this reflected later on retrieve and display the data speed. Also before transfer the data to virtual database the system apply condition on tables and retrieval only that verify the condition.

2.1 Open the Connect &Retrieval Metadata Stage

in this stage will open the connect with first and second database to import data and metadata include tables' name, fields' name, data type. Figure 1 illustrated step for open connect with database.

The method of open the connect is different from DBMS to another, because every database has code for open the connected with DBMS through application program, Some DBMSs need input server name such as MSQL-server and another can open the connected without input sever name such as access.

This operation provided tables and fields that needed to use in query, this stage will apply on first & second databases. The field that selected will be added to virtual tables in global database after finished condition stage.

The system interface present form includes some Combo Box tools for display table name, fields name, and data type, and also include Data Grid View to view the fields and fields’ data type that need use in query operation.

2.2 Join Tables Stage

To retrieval fields from different database must be link its, In this stage join operation applied on tables.

The first step the system selected tables from first & second databases and display fields to select join key from tables1&tables2 then user select join type (INNER JOIN, LEFT JOIN..)that will be use in the query, after this step primary key must be select from first and second table then select link tool (=, <, >, <>). Figure 2represent this steps.

Second stepthe system creates the part of SQL-code that includes Join code to run on global DB after finished all stages. The code can result from this step such as "From employ INNER JOIN account on employ.ID = account.ID". Primary keys may be heterogonous in databases fields, in this state the system allow link between two different primary keys datatype.

2.3 Condition Stage

Migrationdatafrom source database to global database take a long time &we should be sure all data needed in query well be transform. Some query include condition, before transfer data must detect if query between tables include condition or no, the system apply condition on tables before transfer data to reduce data transfer time.

Some query only include condition on fields by assignment values, and other query needed apply condition between two table by assignment values from column in table1 to column in table2.

2.4 Solve Heterogonous Problem in Datatype

Before the apply transfer procedure the system apply procedure for disposed the difference in the datatype include modify fields &data types use as join key or columns use in condition.

In this step can use one of two algorithm (Vertical or Horizontal configuration). Horizontal configuration: Being testing and modifying datatype start from first row to end rowin table then transformation data. Vertical configuration is testing and modifying datatype then transformation data column by column, Vertical configuration algorithm use in this paper, Figure 5 represent flow chart for this algorithm step, Figure 6 Representation of that algorithm

2.5 Create Tables in Global Database Stage

After completed select fields and metadata of it then chose the join type and insert condition, the system will create tables in global database and transfer data after solve heterogonous problem in datatype.

The query code will have applied on global database that is federation database. to create tables must have SQL-code, to create this code the system run this procedure(1-detected the fields name for first & second database tables.2- Verify if fields that are selected to use in query included the join key, if aren't insert the system will insert it. When join tables maybe join key is different in data type, therefore the system will unify join key data type in both tables. 3- transfer data to federation database). In same time the system creates SQL-code to retrieve data from federation database.

3. CONCLUSION

The evolution in information technology and a race many of DBMSs, that are supports lot of functions. The data is very important but how can benefit and integration it from DBMS when store in different environment. One of problem in data integration is heterogeneous database query.

To solve integration problem must found A mechanism to resolve that problem. In this work introduce tool use to integrating data from heterogeneous database. Characterize our search from previous research, by allow this tool use any time to be suitable for update data and schema modifications. Also excelled by decomposition query to three division query, first apply on DB1 & second apply on DB2 and third apply on global DB, that division Reflected on the speed of data retrieval. The system provides a uniform interface and high level of location transparency for data retrieval from the local DBMS.

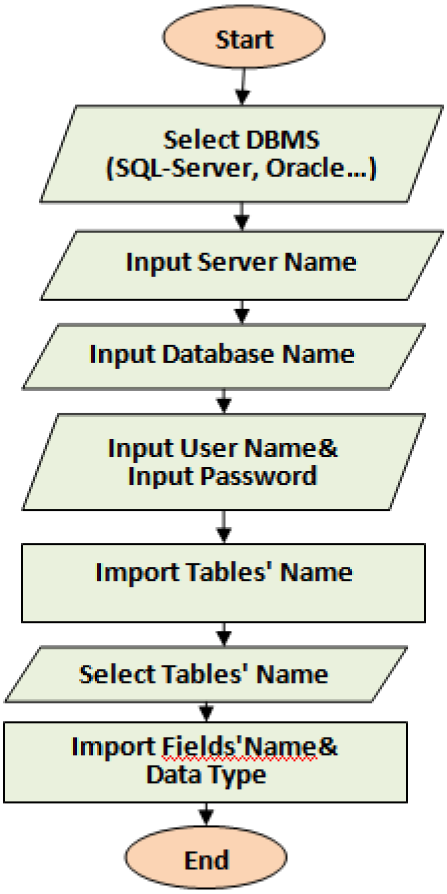

Start
Select DBMS (SQL-Server, Oracle…)
Input Server Name
Input Database Name
Input User Name& Input Password
Import Tables' Name
Select Tables' Name
Import Fields'Name& Data Type
End

Fig 1

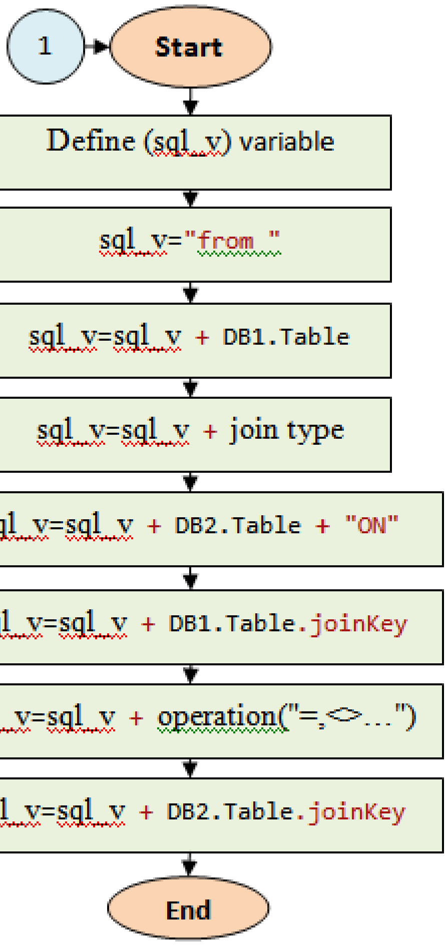

1
Start
Define (sql_v) variable
sql_v="from "
sql_v=sql_v + DB1.Table
sql_v=sql_v + join type
sql_v=sql_v + DB2.Table + "ON"
sql_v=sql_v + DB1.Table.joinKey
sql_v=sql_v + operation("=,<>…")
sql_v=sql_v + DB2.Table.joinKey
End

Fig 3

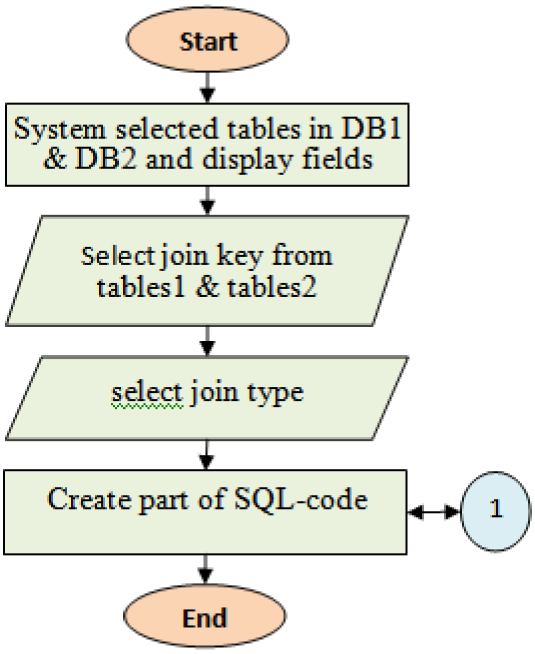

Start
System selected tables in DB1 & DB2 and display fields
Select join key from tables1 & tables2
select join type
Create part of SQL-code
1
End

Fig 2

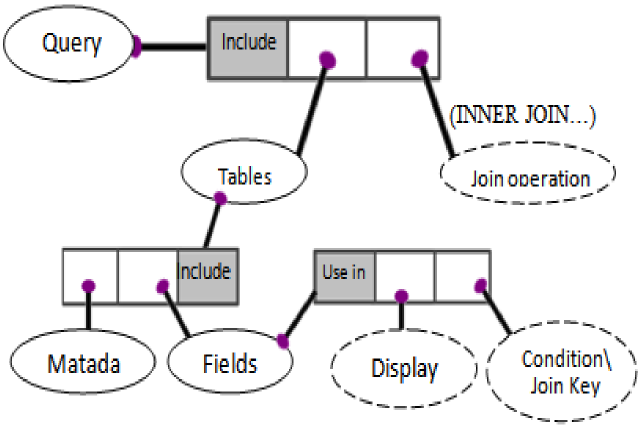


Fig 4

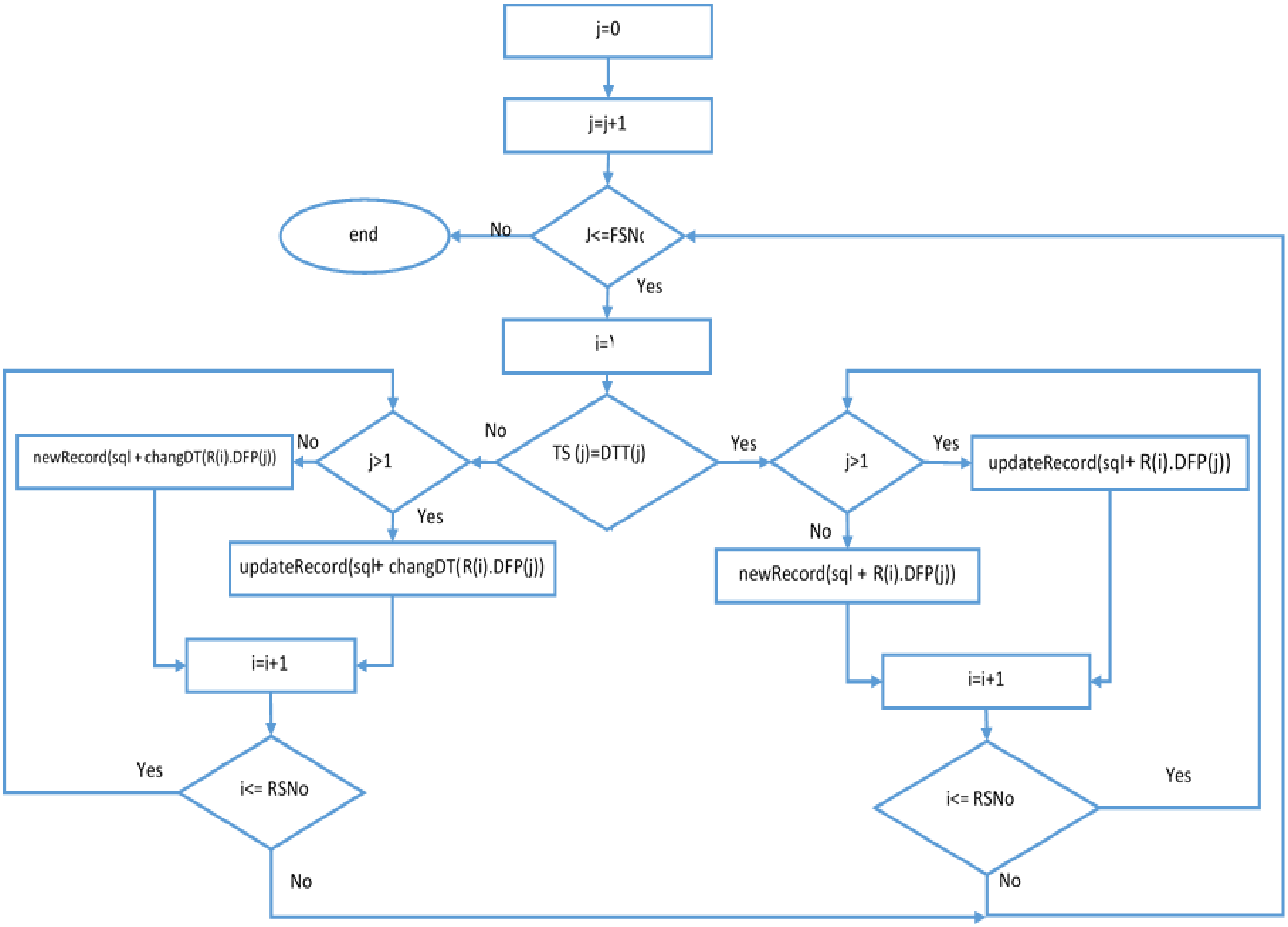

j=0
j=j+1
end
No
J<=FSNo
Yes
i=1
No
j>1
newRecord(sql + changDT(R(i).DFP(j))
No
TS (j)=DTT(j)
Yes
j>1
Yes
updateRecord(sql+ R(i).DFP(j))
Yes
updateRecord(sql+ changDT(R(i).DFP(j))
No
newRecord(sql + R(i).DFP(j))
i=i+1
i=i+1
Yes
i<= RSNo
i<= RSNo
Yes
No
No

Fig 5[6]

R : first database column location/ column need to datatype modify
DTS: array include first database datatype
DTT: array include federation database datatype/must same datatype for column second database
DFP :cell location need to data modify
FSNo: Number of columns need modify
RDNO: number of fields in column
J: count for transfer between column
I: count for transfer between column

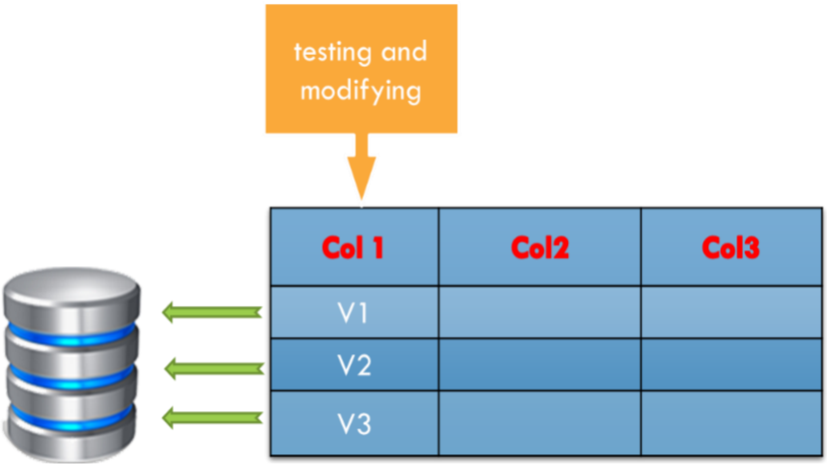
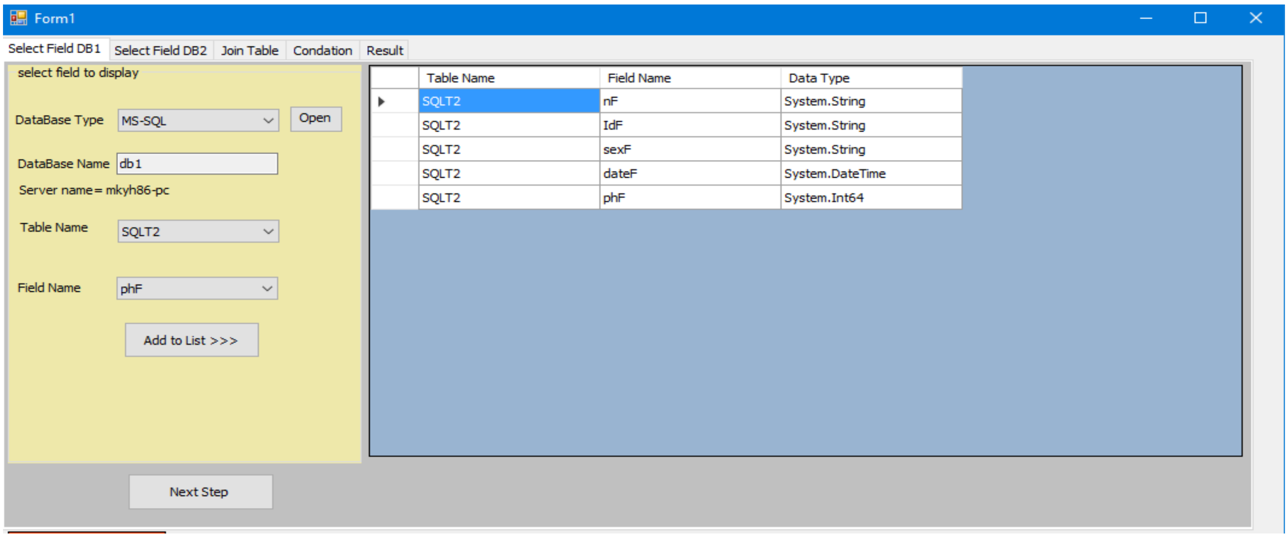


Fig 5

4. EXECUTE OF APPLICATION PROGRAM

4.1 Select Data from DB1, DB2



select field to display

DataBase Type: MS-SQL

DataBase Name: db1

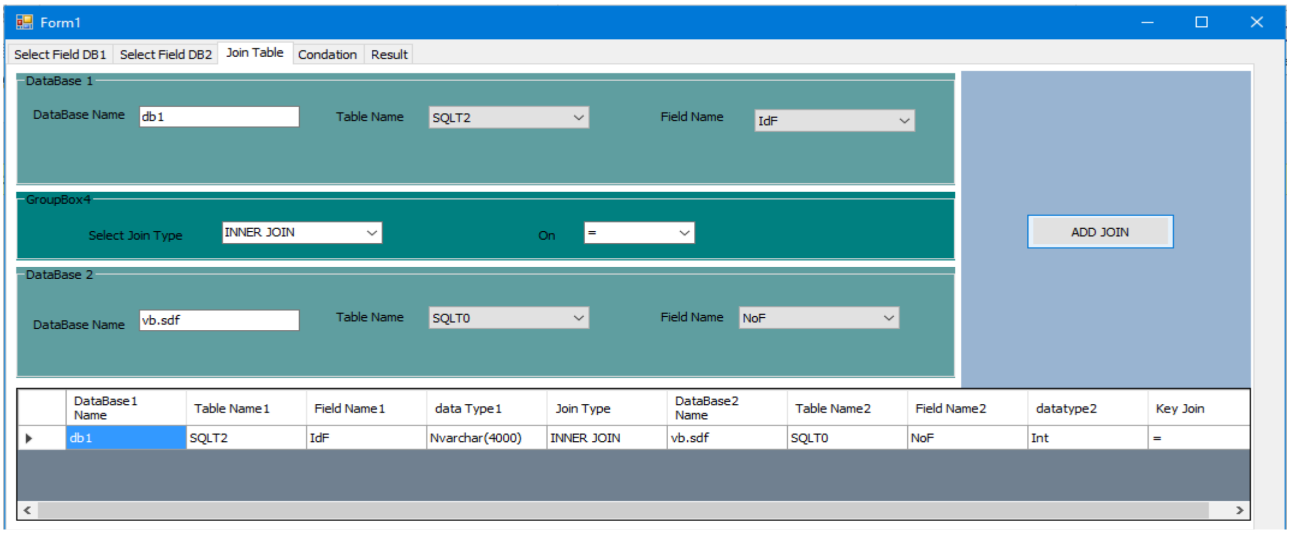
Server name= mkyh86-pc

Table Name: SQLT2

Field Name: phF

Table Name	Field Name	Data Type
SQLT2	nF	System.String
SQLT2	IdF	System.String
SQLT2	sexF	System.String
SQLT2	dateF	System.DateTime
SQLT2	phF	System.Int64

4.2 Join Tables



DataBase 1

DataBase Name: db1 Table Name: SQLT2 Field Name: IdF

GroupBox4

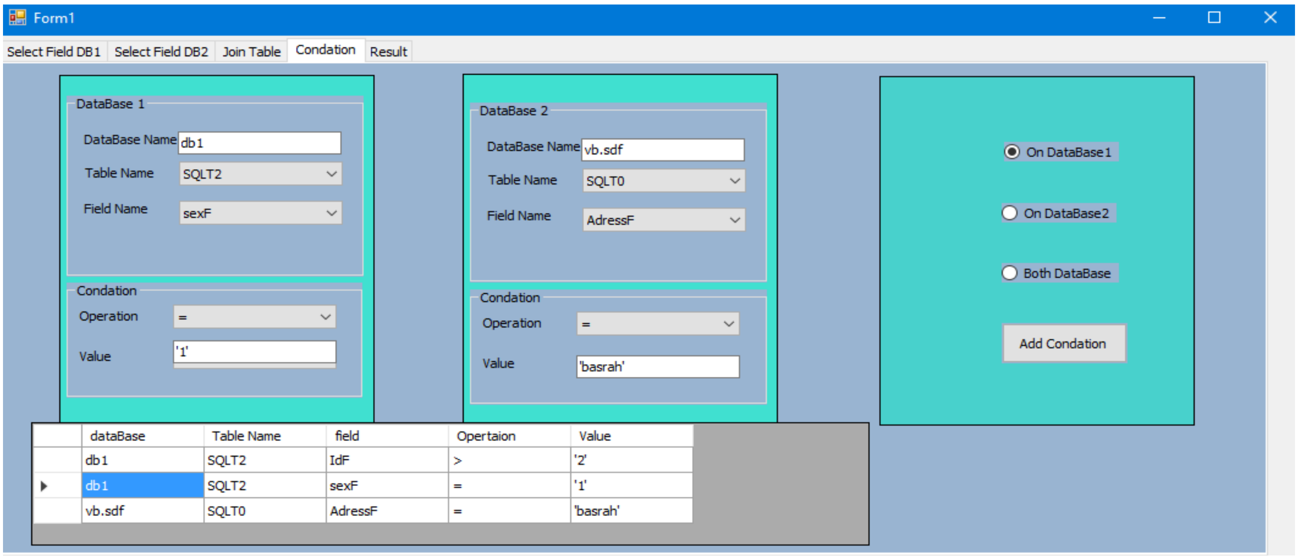
Select Join Type: INNER JOIN On: =

DataBase 2

DataBase Name: vb.sdf Table Name: SQLT0 Field Name: NoF

DataBase1 Name	Table Name1	Field Name1	data Type1	Join Type	DataBase2 Name	Table Name2	Field Name2	datatype2	Key Join
db1	SQLT2	IdF	Nvarchar(4000)	INNER JOIN	vb.sdf	SQLT0	NoF	Int	=

4.3 Apply Condition



DataBase 1

DataBase Name: db1

Table Name: SQLT2

Field Name: sexF

Condation

Operation: =

Value: '1'

DataBase 2

DataBase Name: vb.sdf

Table Name: SQLT0

Field Name: AdressF

Condation

Operation: =

Value: 'basrah'

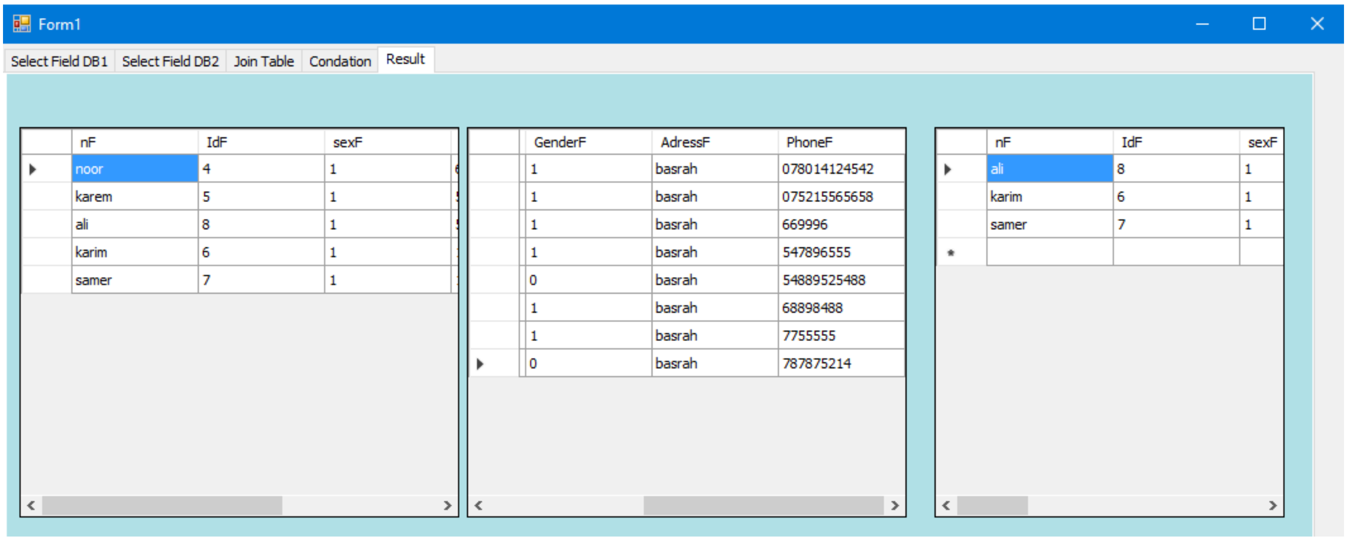
● On DataBase1

○ On DataBase2

○ Both DataBase

dataBase	Table Name	field	Opertaion	Value
db1	SQLT2	IdF	>	'2'
db1	SQLT2	sexF	=	'1'
vb.sdf	SQLT0	AdressF	=	'basrah'

4.4 Result



nF	IdF	sexF
noor	4	1
karem	5	1
ali	8	1
karim	6	1
samer	7	1

GenderF	AdressF	PhoneF
1	basrah	078014124542
1	basrah	075215565658
1	basrah	669996
1	basrah	547896555
0	basrah	54889525488
1	basrah	68898488
1	basrah	7755555
0	basrah	787875214

nF	IdF	sexF
ali	8	1
karim	6	1
samer	7	1

T1	T2
noor 4 1 6/8/1988 65655655454 karem 5 1 5/8/1970 454545458 ali 8 1 5/1/2000 1564848 karim 6 1 1/5/2010 5555 samer 7 1 1/9/2010 5558855	3 5-8-2000 1 basrah 078014124542 3 7-6-1967 1 basrah 075215565658 6 5-8-2010 1 basrah 669996 7 9-8-2010 1 basrah 547896555 8 9-8-2010 0 basrah 54889525488 9 9-8-1988 1 basrah 68898488 10 8-7-1998 1 basrah 7755555 12 4-7-1998 0 basrah 787875214
Result ali 8 1 5/1/2000 1564848 8 9-8-2010 0 basrah 54889525488 karim 6 1 1/5/2010 5555 6 5-8-2010 1 basrah 669996 samer 7 1 1/9/2010 5558855 7 9-8-2010 1 basrah 547896555	

REFERENCES

[1]. DATABASE INTEGRATION: IMPORTANCE AND APPROACHES. IBRAHIM ALMAHY, NAOMIE SALIM. Journal of Theoretical and Applied Information Technology. 2013. Vol. 54 No.1.

[2]. Database System ConceptsSixth Edition Avi SilberschatzHenry F. KorthS. Sudarshan. McGraw-Hill. 2010. http://www.db-book.com/

[3]. Object-Oriented Approach for Integration of Heterogeneous Databases in a Multidatabase System and Local Schemas Modifications Propagation. Mohammad Ghulam. International Journal of Computer Science and Information Security. 2009. Vol. 6, No. 2.

[4]. Data Model for Integrating Heterogeneous Medical Data in the Health-e-Child Project. Andrew Branson, Tamas HAUER, Richard Mcclatchey, Dmitry Rogulin and Jetendr Shamdasani. Studies in health technology and informatics. 2008. 138:13-23

[5]. Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases. Sheth and Larson. ACM Computing Surveys,1990. Vol. 22, No.3. pp. 183–236.

[6]. Data Migration Between Heterogeneous Databases (international fourth scientific conference-university of Kerbala), Talib Obeid, Mohammed Yassin. 2016