



Conversion of Data along with Entire Structure of Referential Tables between SQL SERVER to MS-Excel

Chalorme Roy Mukhopadhyay¹, Prof. Dr Pranam Paul²

MCA, Narula Institute of Technology, Agarpara, Westbengal, India¹

HOD, Department of Computer Application, Narula Institute of Technology, Agarpara, Westbengal, India²

ABSTRACT: In this paper, the conversion of tables from SQL Server platform to MS Excel platform as an initial process of the project is discussed with maintaining structure of the table in intact format along with their parent child relationship. That means it will assure the conversion of the structure of original database with its original data and as well as its primary key too (if there is any primary key present in table) after data migration process is over. This whole process is called as data conversion from databases to file in my project. Finally, tables of SQL Server are converted into a MS Excel file in individually separated sheets with mentioning all proper structural formats.

KEYWORDS: Databases, Migrate, Table, Parent, Goal, SQL Server, MS Excel, Sheets, Intact, Conversion.

I. INTRODUCTION

Here, I have worked and found out how data will be migrated from SQL Server to MS Excel platform which will keep the entire structure of the database in an intact format.

As my work progressed, I found out how the details of the table along with the table structure can be fetched from SQL Server, which means SQL Server itself is allowing copying its table's structure with all details of the tables. If I was unable to collect these details and data about the structure and the tables through SQL, then I had to think about how to break the security of the DBA part of the SQL Server in an ethical manner.

II. RELATED WORKS

Now, I am discussing about my actual work to convert the database from SQL Server to MS Excel. Here in this paper, Section [1] discusses the establishing the connection. While section [2] gives the idea about the data conversion from database to an excel file. And section [3] describes an entire example of the conversion process in details.

A. ESTABLISHING THE CONNECTION BETWEEN DATABASE AND EXCEL FILE:

I have implemented my concept of converting and migrating the data from SQL Server to MS Excel by Visual Basic (6.0) to develop the software which is able to convert the whole database from SQL Server to MS Excel with maintaining all data and whole structure of the tables. Then all the necessary actions which are required for connecting the SQL Server database and MS Excel through Visual Basic were done. Lastly the SQL commands were written in VB (6.0) properly so that they could be fired in proper database in proper time and the output were kept in record sets. Here, at first, SQL commands were fired through VB (6.0), the output were temporarily stored in record sets provided by VB. From the record set, each record were taken one by one tuple wise and they were stored one by one in an MS Excel file which will be selected by us and this process has been executed by keeping the format of each fields intact.

B. REQUIRED INFORMATION ABOUT SQL SERVER DATABASE AND IMPLEMENTATION IN MS EXCEL:

2.1 Creation of table structure in MS Excel as Sheet:

To migrate the table, at first I had to get the details about the structure of the table which was to be migrated. If multiple tables were related with each other, means more than one table was joined with reference keys, and then I had

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

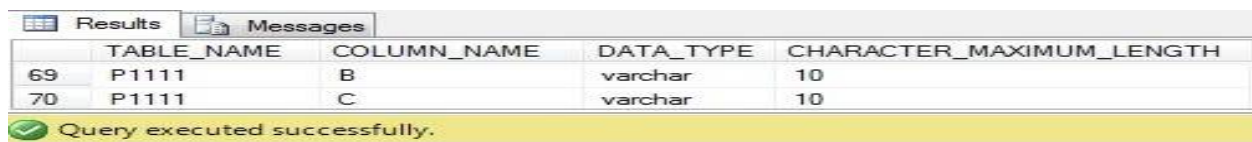
Vol. 4, Issue 9, September 2016

to get the details of every table's structure and so on. This information about the tables was used later to create the same tables with similar relations to other tables in MS Excel.

To get the actual structure of a table in SQL Server I use "sp_help <object_name>" command. But as I was working with VB and using SQL so I had to find the equivalent command of "sp_help" in SQL. I can get the structure from the table in SQL Server named "INFORMATION_SCHEMA.COLUMNS". All required fields were taken from "INFORMATION_SCHEMA.COLUMNS" using the proper SQL command.

Here, the tables are created as individual sheets in MS-Excel with the same name and same data and other information along with the constraints of the selected table for migration process have been also kept into the that particular sheet.

The required fields to be fetched from "INFORMATION_SCHEMA.COLUMNS" table were the column name, data type, character_maximum_length. This information was later used to define or create the main structure of the table in MS Excel.



	TABLE_NAME	COLUMN_NAME	DATA_TYPE	CHARACTER_MAXIMUM_LENGTH
69	P1111	B	varchar	10
70	P1111	C	varchar	10

Fig. 2.1

The output of "sp_help" equivalent SQL command for a particular table having 2 foreign keys- B, C

2.2 Getting information from table INFORMATION_SCHEMA.COLUMNS, INFORMATION_SCHEMA.TABLE_CONSTRAINTS:

In the following fig 2.2 TABLE_NAME, COLUMN_NAME, CONSTRAINT_TYPE, and CONSTRAINT_NAME from INFORMATION_SCHEMA.COLUMNS, INFORMATION_SCHEMA.TABLE_CONSTRAINTS tables are shown:



	TABLE_NAME	COLUMN_NAME	CONSTRAINT_TYPE	CONSTRAINT_NAME
2	P1111	C	FOREIGN KEY	FK_P122_I_00750D23
3	P1111	B	PRIMARY KEY	PK_P12_3BD019A8035179CE

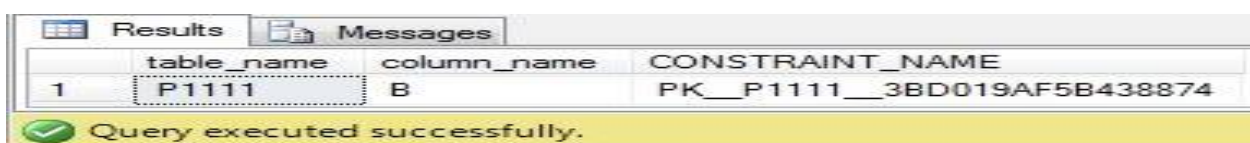
Fig. 2.2

Output of SQL command with constraint name and their type

2.3 Finding the primary key in the table:

Primary key is the important part of any table, as it can uniquely identify any tuple from other tuples. It cannot be null. It also has vital role at the time of inserting any data tuple wise in a table. So, to fetch the information about primary key of any table and store it in a variable is very important. This will be used during the creation of the sheet in MS Excel of a particular file. So, I had to use the SQL command to find the primary key in the table.

The following fig 2.3 shows the primary key present in a particular table. It provides the information about TABLE_NAME, COLUMN_NAME, and CONSTRAINT_NAME..



	table_name	column_name	CONSTRAINT_NAME
1	P1111	B	PK_P1111_3BD019AF5B438874

Fig. 2.3 Information about Primary key

In the above figure 2.3 TABLE_NAME, COLUMN_NAME, CONSTRAINT_NAME are the fields which gives the information about the primary key of a selected table and the CONSTRAINT_NAME specifies the name of the primary key.

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 9, September 2016

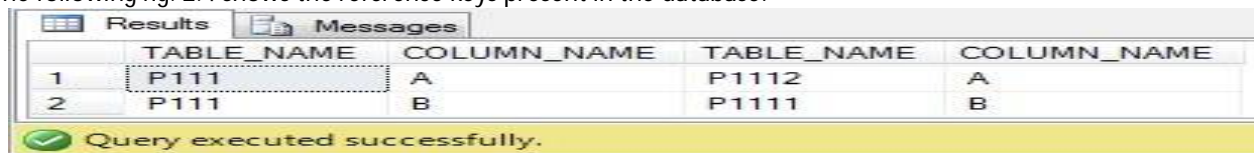
This command is also capable of finding the composite key (when more than one field together uniquely identifies the tuple, it is called composite key).

As discussed in section 2.2, that information was used to get the primary key.

2.4 Finding the foreign key between the tables:

It is also very important to find the foreign key or reference key between tables. I know, by foreign key tables are inter related to each other. And the field which is used as reference key in tables must have same data type and structure. If foreign key or its structure is altered, omitted or replaced, then the entire table structure is changed and it is not able to migrate the actual table, in fact it will show an error message. So it is very important to find the foreign keys, store in corresponding cells in the MS Excel sheet, name of which is similar to the table in SQL SERVER, keeping the entire table structure intact.

The following fig. 2.4 shows the reference keys present in the database:



	TABLE_NAME	COLUMN_NAME	TABLE_NAME	COLUMN_NAME
1	P111	A	P1112	A
2	P111	B	P1111	B

Query executed successfully.

Fig. 2.4

Foreign keys present in the tables

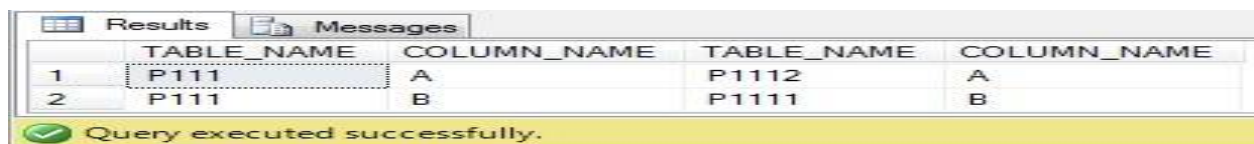
In the above fig. 2.4 the foreign key are denoted to corresponding field name and it also gives the information about the field which is referenced to it.

Similar structure is created while creating the table structure in the MS Excel and hence the main structure is kept intact

2.5 Parent- child relation between tables:

It is also important to find which table is related to each other and which one is created first. It is known that many tables could be related to each other. The table which is created first is called the parent table, while the later created table is called child table. Let it would be divided into the creation of table in levels in SQL Server database. At the time of creating the sheets in MS EXCEL, the sheets which signify the tables of first level in SQL Server must be created before the second level and so on. In similar words, every sheets which is referred as child table can only be created, if and only if all the sheets in MS EXCEL, referring the parent tables of that child table in SQL SERVER are defined first.

The following fig. 2.5 shows parent tables of a particular table selected.



	TABLE_NAME	COLUMN_NAME	TABLE_NAME	COLUMN_NAME
1	P111	A	P1112	A
2	P111	B	P1111	B

Query executed successfully.

Fig. 2.5

Parent table of a particular table

This structure is maintained while creating the database in SQL Server.

2.6 Selecting the table to be migrated:

After login in with the user name and password in the database of SQL Server, I have to select the table which will be migrated to the MS Excel. So, I have to show all the tables available under that user, so that the user can select the table. This list of tables is shown by the data grid in my software, user will select the table and the table will be migrated to the MS Excel platform keeping its main structure, relation with other table and data types of every field intact.

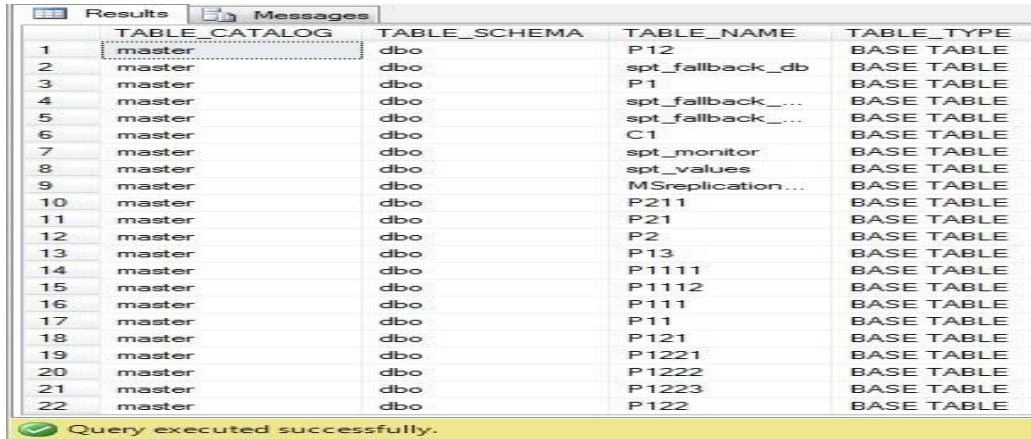
All tables available under a particular user are shown from where particular table can be chosen for migration.

In the following Fig. 2.6 I can get the list of tables available for a particular user:

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 9, September 2016



	TABLE_CATALOG	TABLE_SCHEMA	TABLE_NAME	TABLE_TYPE
1	master	dbo	P12	BASE TABLE
2	master	dbo	spt_fallback_db	BASE TABLE
3	master	dbo	P1	BASE TABLE
4	master	dbo	spt_fallback_...	BASE TABLE
5	master	dbo	spt_fallback_...	BASE TABLE
6	master	dbo	C1	BASE TABLE
7	master	dbo	spt_monitor	BASE TABLE
8	master	dbo	spt_values	BASE TABLE
9	master	dbo	MSreplication...	BASE TABLE
10	master	dbo	P211	BASE TABLE
11	master	dbo	P21	BASE TABLE
12	master	dbo	P2	BASE TABLE
13	master	dbo	P13	BASE TABLE
14	master	dbo	P1111	BASE TABLE
15	master	dbo	P1112	BASE TABLE
16	master	dbo	P111	BASE TABLE
17	master	dbo	P11	BASE TABLE
18	master	dbo	P121	BASE TABLE
19	master	dbo	P1221	BASE TABLE
20	master	dbo	P1222	BASE TABLE
21	master	dbo	P1223	BASE TABLE
22	master	dbo	P122	BASE TABLE

Fig. 2.6
Tables available under a particular user

2.7 Selecting the proper location to save the converted EXCEL file: After proceeding all of the above steps, user need to choose the location to save the Excel file (having extension “.xls”) in a particular folder or have to create a new folder in which user wants to copy the entire table with its information, data and structures and as well as its referenced tables too. After this, the migration process will be started and can view the current progress status of the conversion process that which table is migrating currently, and after the completion of the conversion process the completion message will be generated to inform that migration process is completed successfully or not.

III. SIMULATION RESULTS

3. Example:

To illustrate my work let me consider the following example given in fig. 3.A
Here, a table named **P111** has two parent tables – **P1111** and **P1112**.
I want to migrate this P111 table from SQL Server to MS Excel. As it have two parent tables,
So parent tables must be migrated first. Only after the migration of all parent tables with the data, then only the child table will be created and data will be inserted, i.e. migration will be completed.

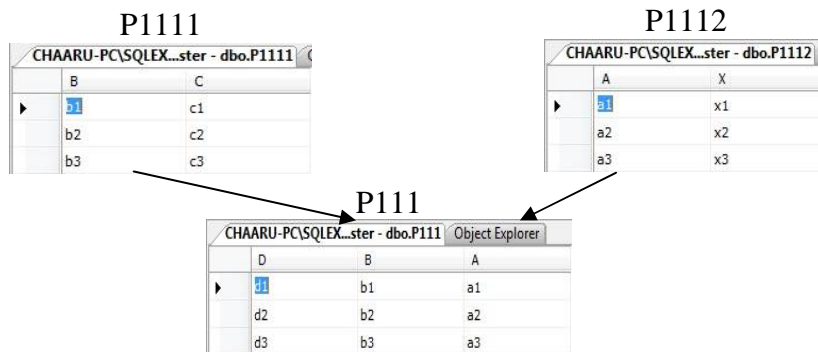


Fig. 3.A - Structure of database in SQL SERVER to be migrated in MS-EXCEL

Here, in Fig.3.A we can see in SQL SERVER, that Table P111 have 2 foreign keys named B (connects with Table P1111) and A (connects with Table P1112) which makes Table P1111 and Table P1112 as the parent tables of Table.

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 9, September 2016

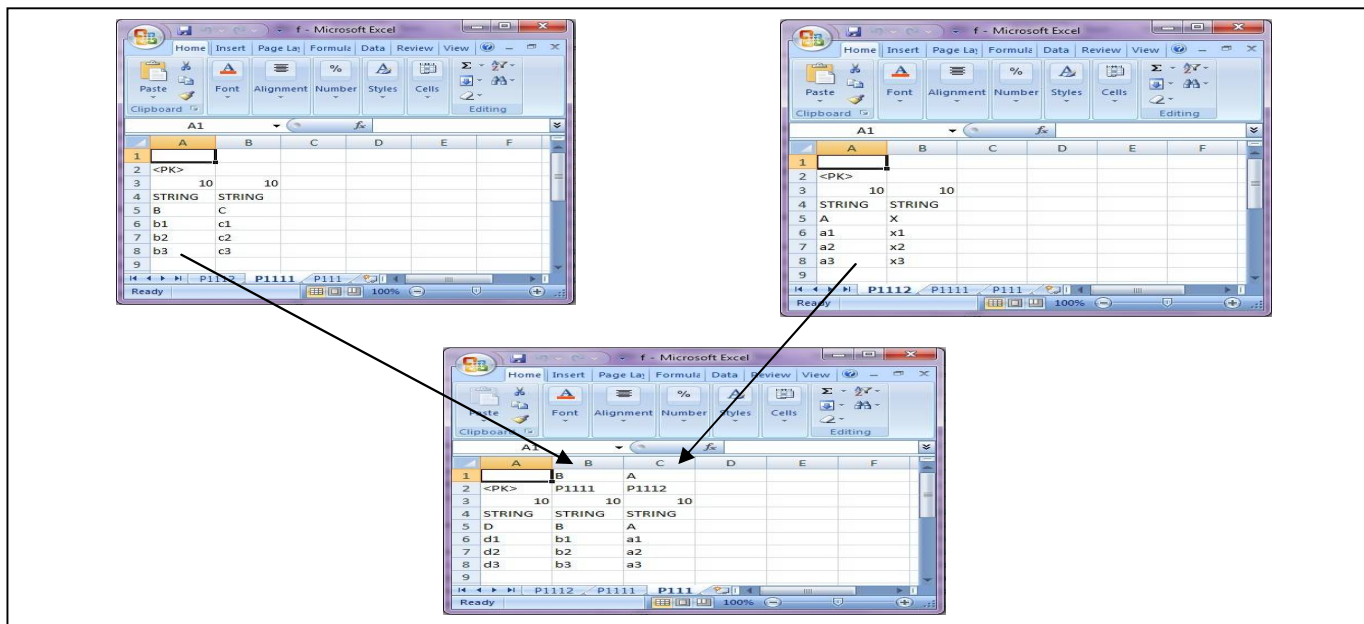


Fig.3.B -Structure of database in MS-Excel after migration from SQL SERVER

Here, in **Fig. 3.B** we can see in MS-Excel, that Table P111 converted as a excel sheet named P111 which have the information about the table such as, P111 have one Primary key named D which is represented by “<PK>” on cell no. (2, A) and also have 2 foreign keys named B [represented in cell no. (1, B)] Which connects with Table P1111 [represented in cell no. (2, B)] and A[represented in cell no. (1, C)] Which connects with Table P1112 [represented in cell no. (2, C)], so which makes Table P1111 and Table P1112 as the parent tables of Table P111.

Above figures signifies that from SQL SERVER P111 table is copied with its all data and keeping the information about the structure and Table P1111, Table P1112 are also copied in MS Excel as they are referenced table of P111.

Now, if **P1111** table is to be migrated from SQL SERVER to MS Excel.

- 1st row of the excel sheet signifies whether there is any foreign key presented in the converted and migrated table or not. If there is no foreign key in that case that row will be in empty condition.
- 2nd row of the excel sheet signifies mixed information’s about the table-

1) If there is any primary key presented in the migrated table in that case a “<PK>” [stands for Primary Key] will be placed in the top of the column name which is the primary key of that table,same procedure will be maintained if there are more than one primary key.

2) If there is any foreign key in that case in this row the foreign table which is connected with the converted table will be placed in this row according to the column name (which column of this table is connected with the table which is placed in this row)and this procedure will be maintained in case of more than one foreign keys also.

Here is the step by step explanation how these tables are migrated from SQL Server to MS Excel.

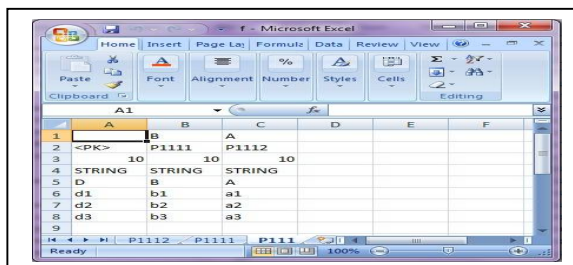
A. FINDING THE FOREIGN KEY IN CHILD TABLE:

As I select the child table (P111) and convert it, at first, it searches for the foreign key. Here child table have two foreign keys (placed in cell no. 1B & 1C). Fig. 3.1 shows the information about the foreign keys fetched.

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 9, September 2016



A	B	C
<PK>	P1111	P1112
10	10	10
STRING	STRING	STRING
D	B	A
d1	b1	a1
d2	b2	a2
d3	b3	a3

Fig 3.1

Foreign keys of child table

So before creating the structure of P111 table, it goes to P111 and P112 tables.

B. CREATION OF STRUCTURE OF FOREIGN KEY:

As it moves to P1111 table and then P1112 table, which are the parent tables of P111 table, it again searches for foreign key of P1111 and P1112 tables.

C. CREATION OF STRUCTURE AND DATA INSERTION P1111 TABLE:

As P1111 have no foreign key so it starts to construct the structure of P1111 table. Then its primary keys are to be found and collecting the tuples one by one and insert it into the corresponding excel sheet in to the corresponding cell starting from 6th row of that excel sheet.

D. CREATION OF STRUCTURE AND DATA INSERTION P1112 TABLE:

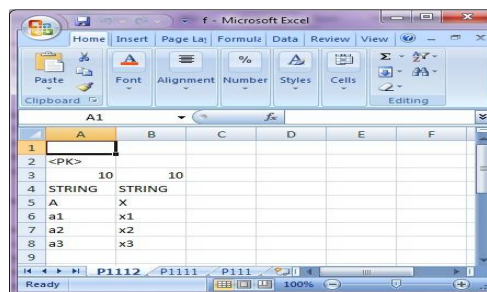
The above process is done recursion ally meaning as 3.3 creates and inserts data in table P1111, then the structure of P1112 is created, primary key is implemented and data are inserted.



A	B	C
<PK>	10	10
STRING	STRING	
B	C	
b1	c1	
b2	c2	
b3	c3	

Fig 3.3

No foreign key of P1111



A	B	C
<PK>	10	10
STRING	STRING	
A	X	
a1	x1	
a2	x2	
a3	x3	

Fig 3.4

No foreign key of P1112

Fig. 3.3 and Fig 3.4 shows it has no result for foreign key as in both parent table cell no. 1B & 1C remains blank after conversion.

E. CREATION OF STRUCTURE OF P111 TABLE AND DATA INSERTION:

As all the tables of P111 table now has been created, so it comes back to P111 table, then creates it structure and inserts all the data. And finally all tables are migrated from SQL Server to MS Excel.

IV. ANALYSIS

In this paper, I have to plan the process of conversion and migration in a special manner, as SQL Server is a database platform and MS EXCEL is a file system-which means both are different in nature, SQL Server can understand SQL queries but in case of MS EXCEL-SQL queries are not applicable for implementation, even there is no data type and primary key concept in MS EXCEL (all are considered as strings) but in SQL SERVER there is the data type concept, no primary key concept. But need to convert and migrate all data's from one platform to another and for that I had to implement such a process that will be equivalent for both platforms.



International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 9, September 2016

For these reasons, I have to plot the information's in MS EXCEL's sheet(which are individually acting as a table of SQL SERVER in MS EXCEL) after conversion and migration process from SQL SERVER in a structured format, such as-

■ 1st Row of MS EXCEL File:- If there is any foreign key in that table those will be plotted in this 1st row according to their position, that is dependent on their situated foreign table's position (discussing in next point) .

■ 2nd Row of MS EXCEL File:- This 2nd row is to hold mixed information's about the table.

If there is any primary key presented in the migrated table in that case a "<PK>" [stands for Primary Key] will be placed in the top of the column name which is the primary key of that table, same procedure will be maintained if there are more than one primary key. If there is no primary key in that case that position will be in empty position.

If there are any no. of Foreign tables connected with that table in that case those foreign table's names will be plotted in this row according to the column name (that, by which own column that table is connected with the foreign tables). If there is no Foreign Table in that case that position will be in empty position.

■ 3rd Row of MS EXCEL File:- This row is used to plot the data length of each column. Every column's data length will be represented by its upper level cell's value of 3rd Row.

■ 4th Row of MS EXCEL File:- This row is used to plot the data type of each column. Every column's data type will be represented by its upper level cell's value of 4th Row.

■ 5th Row of MS EXCEL File:- This row is used to plot the column names of each column of the table.

■ 6th Row onwards of MS EXCEL File:- From 6th row onwards the data's of the tables are being stored according to the column name.

Here, I have used recursion technique to find the foreign keys of the tables, as if the selected table for the conversion and migration process have any no. of foreign keys exists for a particular table in that case those parent tables should be copied first with its data's and all structure and after that the child table will be copied along with data's and it's structure. In this case the parent tables will be converted first in MS EXCEL and migrated in individual sheets of the excel file and the excel sheet's names will be changed according to that parent table's name, and after all parent table checking finally child table will be converted and migrated in same manner following same process.

V. CONCLUSION AND FUTURE WORK

As my initial goal was to migrating tables from one database to another different database the software developed by me is successful of migrating tables between SQL SERVER to MS Excel. Though it has some limitations but I look forward to work out and develop more accurate software.

Above said limitation of this concept can be rectified. I will work out about this limitation in future and hopefully will overcome this limitation.

Apart from this, as I am able to migrating tables from SQL Server to MS Excel, now I will concentrate on extending this concept so that it will be able to converting and migrating the tables in some other way, i.e. from some other database to some other database or some other file to some other database and vice versa. Not only that, I wish to carry forward my work and developing a software which will be able to convert and migrate tables between all major databases which follow the CODD's rule (SQL Server, My SQL, MS ACCESS) or even files.

REFERENCES

- [1] Kevin Loney and George Koch, Oracle 9i, The Complete Reference, sixteenth reprint, 2005, Oracle Press.
- [2] Silberschatz, Korth and Sudarshan, Database System Concepts, fourth edition, McGraw-Hill International Edition.
- [3] Ivan Bayross, SQL, PL/SQL: The Programming Language Of Oracle, 4th Revised Edition.
- [4] <https://www.google.co.in/>
- [5] <http://stackoverflow.com/>
- [6] <https://www.wikipedia.org/>
- [7] Yahoo.com
- [8] <https://community.oracle.com>
- [9] <https://oracle.com>
- [10] www.blocks.oracle.com
- [11] Kibria Dastagir Dafadar, Chalorme Roy Mukhopadhyay, Prof. Dr Pranam Paul; "Implementation of Data Migration between Two Different Databases for discrete table"; Paper ID: V40202104; Accepted and Published in "International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)"; ISSN No: 2320-9801 (Online), 2320-9798 (Print); Vol. 4, Issue: 1; Feb 2016; pp 2144 – 2147.



International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 9, September 2016

- [12] Chalorme Roy Mukhopadhyay, Kibria Dastagir Dafadar, Prof. Dr Pranam Paul; "Implementation of Data Migration between Two Different Databases for inter connected tables"; Paper ID: V40202106; Accepted for Publication in "International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)"; ISSN No: 2320-9801 (Online), 2320-9798 (Print); Vol. 4, Issue: 1; Feb 2016; pp 2148 – 2154.
- [13] Chalorme Roy Mukhopadhyay, Prof. Dr Pranam Paul; "Conversion of Entire Structure of Database from SQL SERVER to MS Access"; Paper ID: V40505138; Accepted for Publication in "International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)"; ISSN No: 2320-9801 (Online), 2320-9798 (Print); Vol 4, Issue: 5; May 2016; pp 9179 - 9185
- [14] Chalorme Roy Mukhopadhyay, Prof. Dr Pranam Paul; "Data Migrating with Foreign Key Concept from ORACLE to SQL SERVER"; Paper ID: V40505138A; Accepted for Publication in "International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)"; ISSN No: 2320-9801 (Online), 2320-9798 (Print); Vol 4, Issue: 5; May 2016; pp 9179 - 9185
- [15] Chalorme Roy Mukhopadhyay, Prof. Dr Pranam Paul; "Implementation of Data Migration between SQL Server to ORACLE"; Paper ID: V40505138B; Accepted for Publication in "International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)"; ISSN No: 2320-9801 (Online), 2320-9798 (Print); Vol 4, Issue: 5; May 2016; pp 9186 – 9192
- [16] Chalorme Roy Mukhopadhyay, Prof. Dr Pranam Paul; "Conversion of Data along with Entire Structure of Referential Tables between ORACLE to MS Excel"; Paper ID: V40707212; Accepted and Published in "International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)"; Impact Factor: 6.577; ISSN No: 2320-9801 (Online), 2320-9798 (Print); Vol 4, Issue: 7; July 2016; pp 13681-13688 ;
- [17] Chalorme Roy Mukhopadhyay, Prof. Dr Pranam Paul; "Implementation to convert Data along with Entire Structure of a Discrete Table between ORACLE to MS Excel"; Paper ID: V40707213; Accepted and Published in "International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)"; Impact Factor: 6.577; ISSN No: 2320-9801 (Online), 2320-9798 (Print); Vol 4, Issue: 7; July 2016; pp 13686-13691 ;

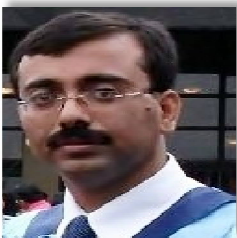
BIOGRAPHY



Chalorme Roy Mukhopadhyay, Passout Student of MCA from Narula Institute of Technology, Agarpara, Westbengal, India. She completed her BCA degree from George College (Kolkata), under the West Bengal University of Technology.

Achievements:

1. Got 431st Rank on NEST(Nationwide Education And Scholarship Test) Aptitude Test 2013,
2. Scored among the top 24 % (76 percentile) on NIIT 8th National IT Aptitude Test 2012.
3. Seven papers accepted and published in an International Journal.



Prof. Dr Pranam Paul, Assistant Professor and Departmental Head, CA Department, Narula Institute of Technology (NIT), Agarpara had completed MCA in 2005. Then my carrier had been started as an academican from MCKV Institute of Technology, Liluah. Parallely, at the same time, I continued my research work. At October, 2006, National Institute of Technology (NIT), Durgapur had agreed to enroll my name as a registered Ph.D. scholar. Then I had joined Bengal College of Engineering and Technology, Durgapur. After that Dr. B. C. Roy Engineering College hired me in the MCA department at 2007. At the age of 30, I had got Ph.D. from National Institute of Technology, Durgapur, WestBengal. I had

submitted my Ph.D. thesis only within 2 Years and 5 Months. After completing the Ph.D., I had joined Narula Institute of Technology in Computer Application Department. Parallely I continue my research work. For that, I have 39 International Journal Publications among 54 accepted papers in different areas. I am also reviewer of International Journal of Network Security (IJNS), Taiwan and International Journal of Computer Science Issue (IJCSI); Republic of Mauritius.

Achievement:

1. Accepted my name for publication in "Who's Who Science and Engineering, 2011 – 2012" published by "Marquis Who's Who", USA on 31st Dec 2010
2. Selected my name as "Top 100 Engineers' 2011", by "International Biographical Centre", Cambridge, England
3. Selected my name as "Outstanding 2000 Intellectuals of the 21st Century, 2012", by "International Biographical Centre", Cambridge, England