Cross Breed Data Migration using ACO Algorithm

Dinesh Kumar,
Research Scholar,
Banasthali University,
Banasthali, Jaipur (Raj.)

Parveen Dhyani, Ph.D.
Executive Director,
Jaipur Campus Banasthali
University, Banasthali, Jaipur
(Raj.)

Anuj Kumar Sharma,
Ph.D.
Dean Academics, OIMT,
Hissar. (Haryana.)

ABSTRACT

Amount of data in organization is going more and more as time passing. Some organization follow open source to manage data and many of them use migration process from one server to other server. There are many services available in market that are used for migration process like SQL, Oracle, DB2, and MySQL etc. any one of these is selected as target platform in which want to migrate. In this paper we are going to focus on process of migration and comparing any two approaches through which we achieve a conclusion.

Keywords

Database, Data Migration, SQL Server Integration Service (SQLSIS), SQL Stored Procedure (SQLSP).

1. INTRODUCTION

Most of organization use concept of data migration as a main activity – choose other server for better data storage and retrieval process. These organizations have to overcome challenges if want to remain in today's competition. There are some reasons behind server migration when leader of technical change and some new technology take to manage data. There are near about 40 -50 types of database available in market. It a time when organization can fulfill choice of clients which database required. When data format is changed like excel to doc in a system is also known as migration activity. Initialization of data migration activity with differences in data storage mechanism. Representation of data can be improved with the activity migration [1]. There should no redundancies in database that will be removed by database migration and better performance seeking through better data of an organization. Migration activity needs a planning to get better representation. Generally progress and risks are monitored in planning steps. Requirement of functionality includes documentation, fund and resources for database migration. Non-functional requirements include high availability, high performance and flexibility [2]. There are many types that can be data migration as following:

1. DB to Excel file or vice-versa.

2. DB to flat file or vice-versa.

3. Oracle to SQL server migration of data.

4. Up-gradation of database.

5. Excel file to flat file or vice-versa.

Data migration process access many services like migration assessment, advisory service, and architecture services etc. [3] for completion of data migration from one server to other server. Requirements for technique of data migration are shown in fig 1 as following:
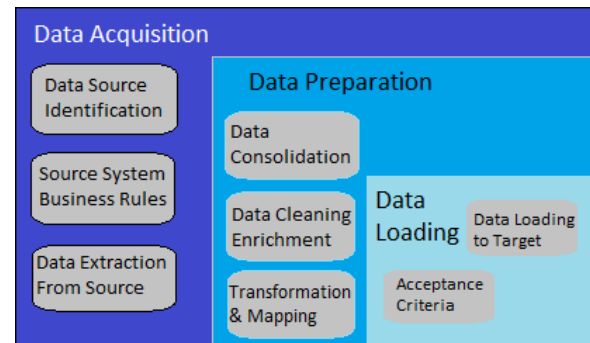

Fig. 1 Data Migration Architecture

There are three main components of data migration process.

1. **Data Acquisition:** In this step rules are extracting for partition of data and extract details regarding data like existing service, database etc.

2. **Data Preparation:** Data prepared for further step like removing redundancies and cleaning data.

3. **Data Loading:** Data loaded for further process for checking correctness.

Benefits of database migration are [4] as following:

1. Helps the enterprise choose the Database platform of choice.

2. Saving in cost from old database as compared to target database.

3. Technological superiority of the migrated database

Migration life cycle is a process that will represent step by step method [5, 16] that needs to follow data for migration as shown in fig 2.
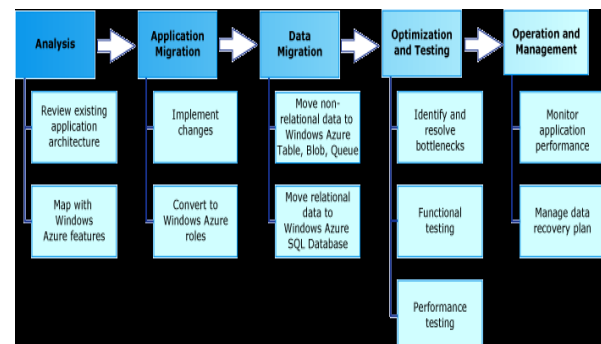

Fig. 2 Migration life cycle

The main migration steps are Analysis Phase, Application Migration Phase, Data Migration Phase, Testing and Optimization Phase, and Operation and Management Phase as shown in the diagram above.

2. MODEL DESIGN AND STRUCTURE
2.1 RDBMS

RDBMS stands for Relational database management system. A Database Management System that organizes the relational database is called as Relational Database Management System. Relational Database Management System data is prepared in form of database like tables, fields and records. All RDBMS table consists of database table rows. All database table rows consist of one or more database table fields. RDBMS accumulate the data into set of tables, which may be correlated by common fields like columns. RDBMS also give relational operator to control the data store into the database tables. A mainly RDBMS use Structure Query Language (SQL) as a database query language. The mainly well-liked RDBMS are MS SQL SERVER and DB2 server.

A Data Base Management System that is based on relational model is called as RDBMS.

2.1.1 Table: The information in RDBMS is stored in record objects called tables. The table is a set of correlated data access and it consists of column and row.

2.1.2 Fields: Every table split into small entities called fields. The fields in the CONSUMERS table consist of id, name, age, address and salary.

A field is a **column** in a table that is considered to preserve precise information concerning every record in the table.

2.1.3 Records: A records is also called as row of data is each being entry.

2.1.4 Features of RDMS:

- Supply data to be stored in tables.

- Supply ability primary key, to exclusively identify the rows

- Distribution a common column in two or more tables like primary key and foreign key.

- High Performance

- High Availability

- Data Integrity

- Robust and Secure

- Lowest cost

2.1.5 Data Models: Data model is a method of accumulating and regaining the data. There are three diverse data models. Data models be different in the way they allow users to view and control relationships between entities. Each has its have method of storing the data. The following are three different data models:
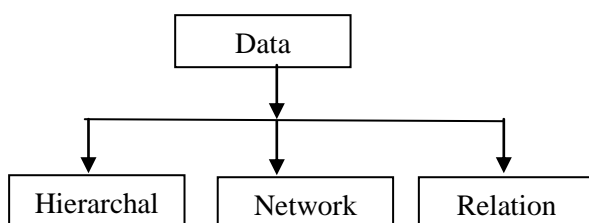
1. **Hierarchical**: In this model, data is stored in the form of a tree. The data is represent by parent child relationship. Each tree includes a single root record and one or more minor records. For case, each batch is root and students of the batch will be subordinate. This model chains only one-to-many relationship between entities.

2. **Network**: Information is stored along with pointer, which identify the relationship between entities. This model is difficult. It is hard to know both the way data is stored and the way data is change. It is able of following many-to-many relationship between entities, which hierarchical model does not.

3. **Relational:** This stores data in the type of a table. Table is a set of rows and columns.

3. APPROACHES FOR DATA MIGRATION

In this section we will discuss two approaches of data migration as following:

3.1 SQL Stored Procedure Approach

A programming language Transact-SQL (T-SQL) is used for development SQL Stored Procedure (SQLSP). Table are created by this powerful language for performing operations like updating, inserting, and retrieving from/into tables [7] for developing logic reviews. Stored procedure is called by name which is representing in set of T-SQL statement. For security purpose that are very useful – only some granted user can access the stored procedure. Unauthorized users are banned to retrieve this. Most of applications use stored procedure as data migration. But for ETL operation [8, 19] it is not viable. It provides high availability for custom logging, customization and control over the logic and data management. Some features like out-of-box memory management are not provided by this but have strong data type and operators.
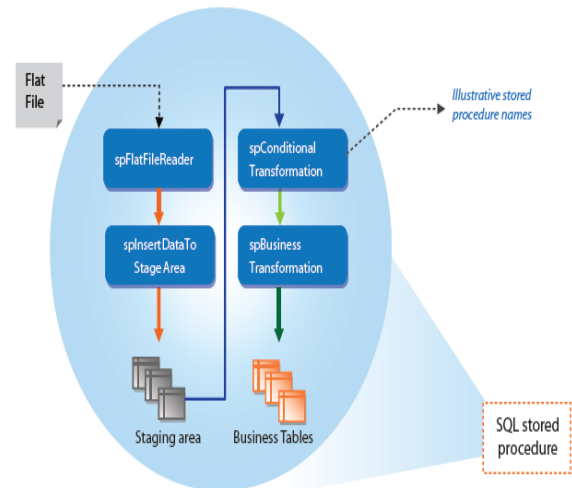

Fig. 3 SQL stored procedure process

A Transact-SQL stored procedure is a set of T-SQL code that is stored in a SQL Server database [9] and compiled when used. We create this set of code using the CREATE PROCEDURE command. We can use most Transact-SQL commands in a stored procedure; however, some commands (such as CREATE PROCEDURE, CREATE VIEW, SET SHOWPLAN_TEXT, SET SHOWPLAN_ALL, and so forth) must be the first (or only) statement in a command batch, and therefore aren't allowed in stored procedures [10]. Although

we can do most of the things a stored procedure can do with simple ad hoc Transact-SQL code, stored procedures have a number of advantages over ad hoc queries, including

- Execution plan retention and reuse
- Query auto parameterization
- Encapsulation of business rules and policies
- Application modularization
- Sharing of application logic between applications
- Access to database objects that is both secure and uniform
- Consistent, safe data modification.

A stored procedure is a procedure or a piece of SQL code that is physically stored in the database and it can be executed (like query) again and again for different purposes. It can be as simple as simple SQL selection query and as complex as doing multiple tasks (insertion, selection, joins, deletion) at the same time. Moreover a stored procedure is precompiled as well. SQL Injection prevention in database stored procedures" as proposed in [11], which describes a technique to prevent injections attacks occurring due to dynamic SQL statements in database stored procedures, which are often used in e-commerce applications.

3.2 SQL Server Integration Service (SQLSIS)

SQLSIS approach is used for data migration and ETL operations, it is better than Stored Procedure due to ETL operation preformation. This is one of the most common corporate-standard ETL tools, since a standard tool is usually forced to address enterprise scale requirements [12]. The view that ETL is simply a component of data warehousing is one that should be confined to the history books. ETL for data warehousing is merely one uses for a technology that is now being asked to grow to encompass data integration across the whole organization and to support areas such as [13]:

- Scalability
- Integration of packaged applications.
- Database consolidation.
- Connectivity; support for integration of legacy systems, relational and non-relational data sources.
- Support for web services and service orientation (SO).
- Integration with messaging middleware.
- Development collaboration.

It is combination of components that perform operations like backing up database, do merge join, SSIS package step by step execution [14, 20], loading flat & excel file, SP package execution for processing multidimensional cubes. In tool there is inbuilt task that is easy for user to use and execute and simple to develop applications in short time. Memory management from out-of-box provided by this approach and also auditing feature like error logging. Data can be migrated in SAP from other approaches [15]. The following diagram shows the control flow in SSIS:
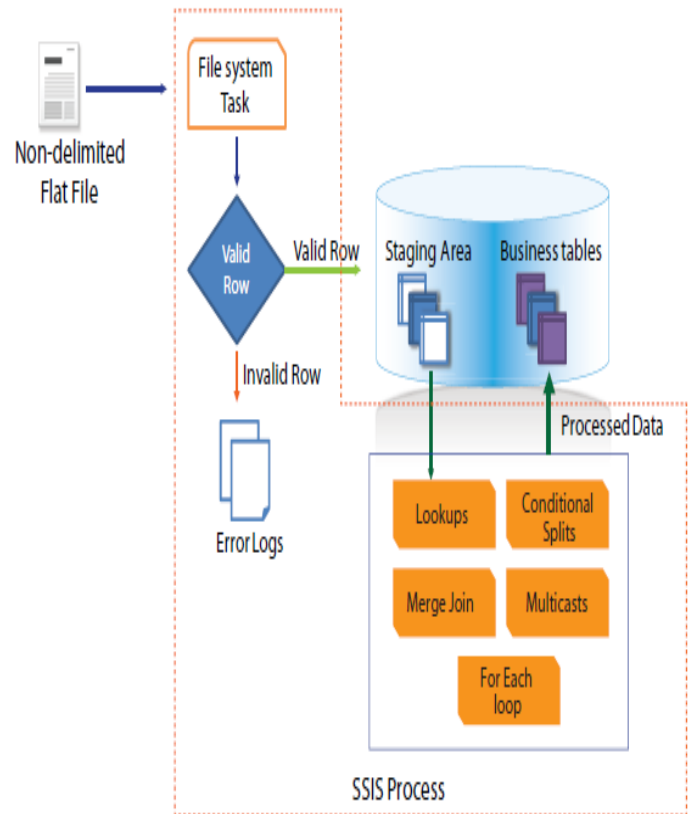

Fig. 4 SQLSIS approach

4. ADVANTAGES AND DISADVANTAGES OF SQLSIS AND SQLSP

- If everyone on our team understands SQL, it's easy to dig into the stored process. SQL is a widely known skill which may be a benefit if we have a lot of ETL writers/readers. We have to be more than a casual user of SSIS in order to understand what it's doing. The high level graphical flow is nice for documentation, but if someone needs to get into the guts, they'd better know SSIS well.

- SQL is a pain to modularize. If we use UDFs [17], we are going to incur a huge performance hit. We'll write similar code in multiple places and we'll hate our self for doing it, but often in ETL scenarios performance is king. SSIS will help us modularize and factor out our tasks.

- Don't expect to be able to easily use source control with SSIS. SQL - no problem. SSIS uses awful XML files [18] which can be checked in, but good luck differing with previous versions to see what changed and when.

- We need to think about our SPs in a modular way, even though it's hard to make them as modular as we'd like. Use temp tables to chunk up our processing. Put indexes on those temp tables before we use them. Don't try to do too much at once. Comment everything.

- If we're using cursors, we're doing it wrong. Don't be afraid to chain in some external console app we wrote in the language of our choice to do some things SQL just wasn't cut out for.

Table I represents the results in completion time to data migration.

Data Volume (In Millions)	Time Taken (In Hours)	
	SSIS	SQL Stored Procedure
1	0.5	2
5	1	4
10	2	6
20	3	8

Table I Completion Time for both approaches

5. MIGRATION ALGORITHM

Two servers minimum are involved in the data migration. The aim is to migrate the architecture system using XAML language pattern avoiding the time delay of the data migration and ensuring the security analysis of the data getting migrated. The purpose of this work is justified when the data along with the architecture is migrated to another platform. To attain the goal, a mid level XAML architecture would be drawn which would show the compatibility with both the server. In the process, the middle server would first analyze the architecture of the first server from where the data has to be migrated and would generate the XAML for it. As XAML is one of the most light weight language and it is supported by all other platforms also, it would be easier for the second server to adapt the language. The middle server would do amendments in the local XAML according to the architecture which has to be migrated to the next sever. Once the second XAML is generated, it would use the TCP IP protocol service along with the SQL Query injector to transfer the XAML from one end to another and would migrate the data completely.

The entire work will be done in phases .It has been explained below.

Step 1. Generate the XML architecture for the one cloud service using Hierarchy

Step 2. Store the xml file into the local system.

Step 3. Fetch data from the xml file to make script for it.

Step 4. Create the script for another cloud.

Step 5. Launch Script to godaddy.com using Script Wizard evaluator.

5.1 Xml Architecture
Generating XML file and filling up the data on cloud according to the XML

5.2 SAXM Meta Xml Model
Algorithm Specifications: Counting the total no of fields generated by user, generating the xml accordingly.
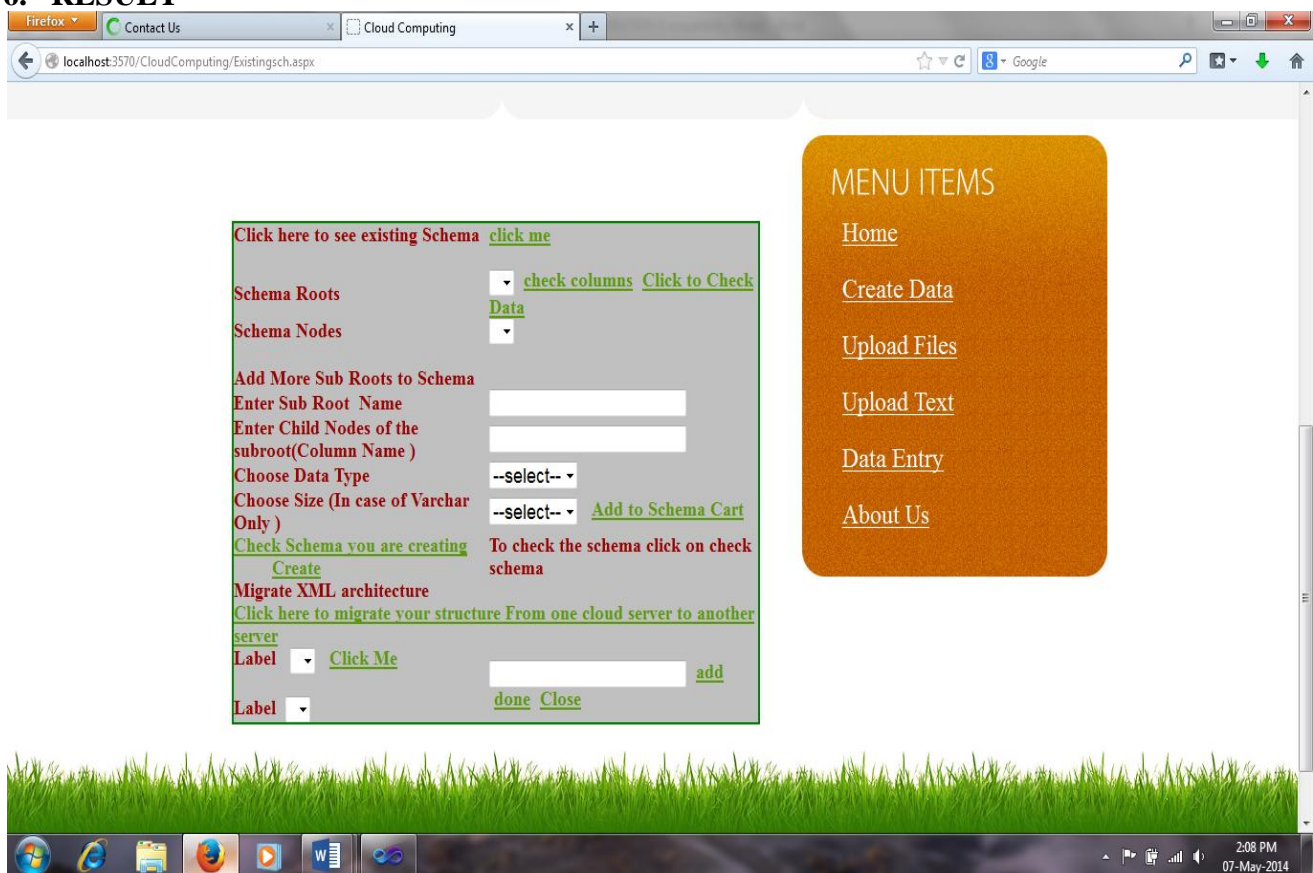
6. RESULT

Fig. 5 Original Page of Data Migration Architecture

The figure 5 shows the original page of the architecture drawing of the data elements. The description is as follows.

The result window contains a main root architecture drawing scheme and the sub root architecture drawing scheme. The migration is done on the go daddy server as follows.
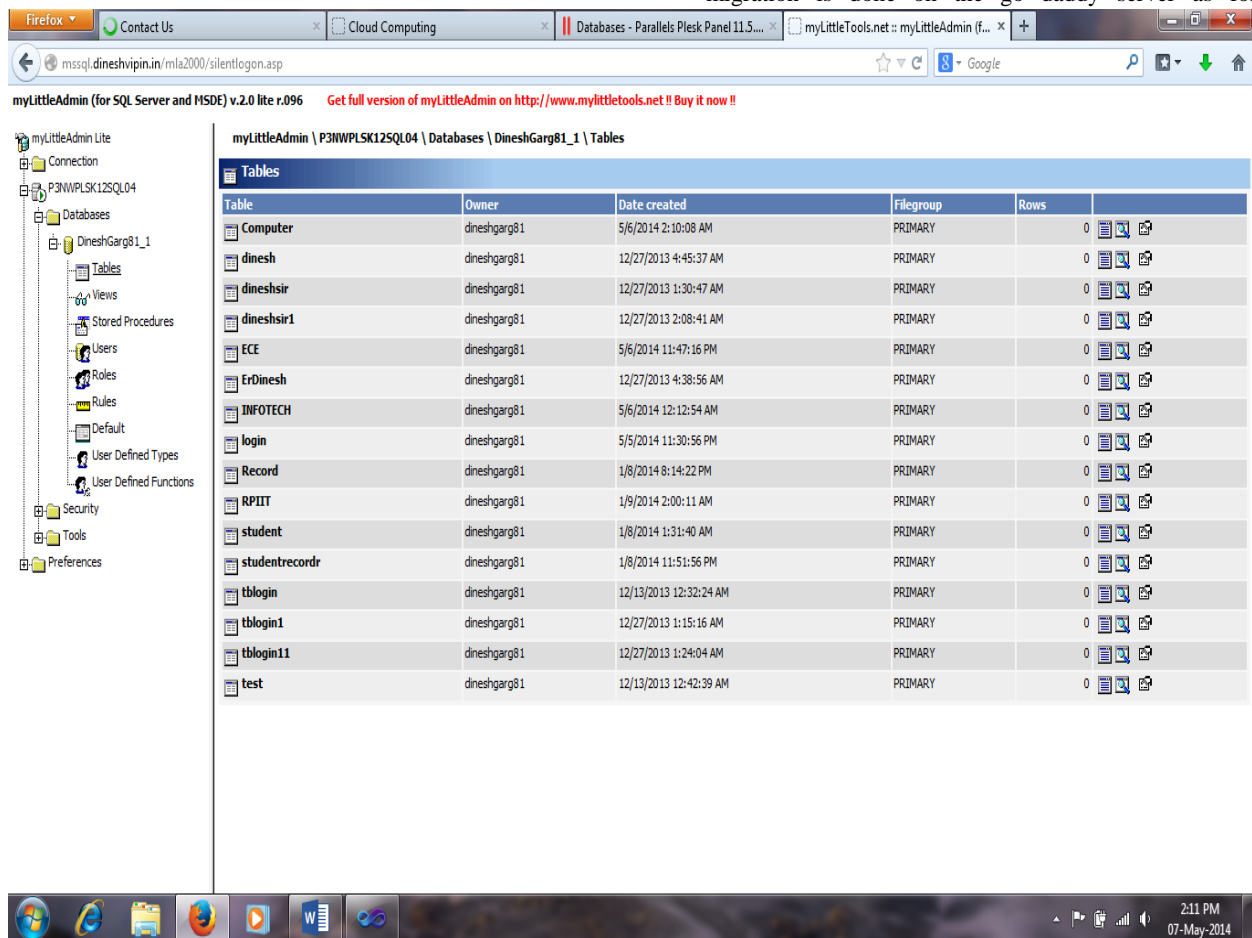

Fig. 6 Go Daddy DB Server after Migration

The above windows shows the architecture of the Go Daddy server from where they has been migrated to the another Go Daddy server.

The above xml schema is generated against the go daddy server data.

Go Daddy server is an online development and storage portal termed as cloud server of Go Daddy Operating Company where the data is to be migrated.

6. CONCLUSION AND FUTURE SCOPE

In this thesis, firstly we have reviewed the earlier work done then proposed a new algorithm based on BCO method and GODADDY server. With the work done till now, the current research work successfully migrates the server from one place to another place. The transfer accuracy is almost 90-95 percent. Though the current system is good but still it has some limitations which are listed below.

a) The current system does not evaluate any computation time for the evaluation that how much time has been elapsed in the transfer.

b) The current system does not involve any back up policy in case of any failure when the data is getting transferred.

The system is not tested with more than two servers in the migration.

7. REFERNCES

[1] Dinesh Kumar, Dr. Parveen Dhyani and Dr. Anuj Kr. Sharma "Migration of Data from One Cloud Server to Another Cloud Server" in the IJAREAS, Volume 3, No. 3, March 2014, ISSN: 2278 6252, Page. 21-29.

[2] Bergey, John, O'Brien, Liam & Smith, Dennis, "DoD Software migration planning", CMU/SEI, 2001.

[3] TietoEnator Corporation,"fenix architecture planning" for 2003-2008.

[4] Oracle expert services, "Oracle Consulting Data Migration Services".

[5] Anil Mahadev, "Database Migration: An in Depth looks!!"

[6] Kun Cheng, Selcin Turkarslan, Norberto Garcia," Migrating Data-Centric Applications to Windows Azure", Category: Guide, E-book publications date: June 2012.

[7] International Technology Group. "Status Report", October 2009.

[8] SQL server 2012 Transact SQL-DML references, online books – SQL server 2012.

[9] Sweety Patel, Piyush Patel, Saumil Patel, "Overview Of ETL Process With Its Important", International Journal

of Engineering Research and Applications IJERA, ISSN: 2248-9622, Vol. 2, Issue 2, Mar-Apr 2012, pp.906-908.

[10] "Backup and Restore of SQL Server Databases", SQL Server 2012 Books Online.

[11] "The Basics", 3831 P-01 12/3/01

[12] Ke Wei, M. Muthuprasanna and Suraj Kothari (Iowa State University). 'Preventing SQL Injection Attacks in Stored Procedures' .Software engineering conference 2006.

[13] Philip Russom, "Best Practices in Data Migration", April, 2006, Informatica the data integration company.

[14] Michael Horne, Jamie Thomson, "A Comparison of Enterprise Class Data Integration Tools", http://blogs.conchango.com, © 2005 Conchango (UK) Plc.

[15] Karthikeyan Anbarasan, "SQL Server Integration Services (SSIS) – step by step tutorial", a SSIS ebook, www.f5debug.net.

[16] ENTOTA, "Comparing 3 Common Approaches to SAP Data Migration" Briefing Paper.

[17] Oracle, "Successful Data Migration", an Oracle White Paper October 2011.

[18] "User-Defined Functions", Microsoft SQL Server 2000 Professional Skills Development Copyright © by Application Developers Training Company and AppDev Products Company, LLC.

[19] "A Standardized Approach to SSIS Migration", cognizant 20-20 insights | October 2013.

[20] Manole Velicanu, Larisa Copcea (Teohari), "Enhancing ETL Performance with Warehouse Builder", Database Systems Journal vol. IV, no. 2/2013.