



IMPLEMENTATION OF A FORMAT MIGRATION USING SAAS MODEL

N Sirisha

Department of Computer Science and Engineering
MLR Institute of Technology
Hyderabad, India

G Divya Jyothi

Department of Computer Science and Engineering
MLR Institute of Technology
Hyderabad, India

B Raswitha

Department of Computer Science and Engineering
MLR Institute of Technology
Hyderabad, India

B Durga Sri

Department of Computer Science and Engineering
MLR Institute of Technology
Hyderabad, India

R Karthik

Department of Electronics and Communication Engineering
MLR Institute of Technology
Hyderabad, India

Abstract : Data migration is the process of transferring data between storage types, formats, or computer systems. Data migration can be done for Server or storage, equipment Website consolidation, replacements or upgrades. It is required when organizations or individuals update computer systems, or when systems merge. Database Migration Tool Using SAAS model is a tool and Java API to migrate data from different sources (XML, DBF and JDBC) to another (DBF, JDBC, TXT). Data migration is preceded by the obsolete method of the programmer performing all the operations that included Copying the data, saving the files and Restoring the data in the database manually. Hence Data Migration in an automated programmatic approach. The conversion configuration is specified by an XML file defining source, target and Schema transformations of the data and also data on the old system is mapped to the new system by providing a design for data extraction and data loading.

Keywords : Migration; SAAS; DB; CSV;

I. INTRODUCTION

The migration tool transfers the database schemas and the data across leading databases such as Oracle, IBM DB2, MS-SQL Server, MySQL and MS Access. Format Migration tool can be connected to any Database that has a JDBC type 4 driver and generates data definition language script, insert statements, data files and related files to migrate the source database to any destination database Data Migration is essential when large organizations upgrades their servers, website consolidation, databases and systems. This transformation was mostly done by the programmer by hand. Developing the format migration tool using SAAS model implements a programmatic approach that automatically migrates the data into the required format without compromising the integrity of the data. The data migration involves many phases but typically depends on Data extraction from the old system and Data loading into the new system. The data extraction phase retrieves the relevant information from the source database in a specific pattern. The majority of data extraction comes from unstructured data sources and different data formats. These unstructured data can be in any form, such as tables and indexes. This data must be mapped to the target system [1-3]. To do so, data loading technique is used. It loads the data into the destination database as a different format than the original source format. Automated data migration minimises

the human intervention and application down time and enhances migration speed. Data migration can be done in three ways which are

- Data migration within the same database
- Data migration between different databases
- Changing the data formats.

Migration between different databases supports entire data transfer among two or more databases or criteria based migrations. If a criterion is set for a table, then its dependant tables will also be migrated based on its relationship. Changing the data formats converts the data from one format to another using CSV files. The data in these files is separated by commas.

In large organizations data has to be migrated between the systems frequently. Upgrading of servers, website consolidation, databases and systems are some of such cases. This must be done by the programmer manually. The data had to be copied, saved and restored in the target database which is a burden over the programmer. There is also an additional possibility of the data being erroneous and inaccurate after manual migration [4-7].

A. Disadvantages:

- The existing system doesn't support all database formats.
- Manual data transformation is time consuming.
- It doesn't support different formats.
- Cost ineffective and requires vast resources to implement.

B. Proposed System

Format migratory tool is a Java API to migrate data from different sources to other. The conversion configuration is specified by an XML file defining source, target and Schema transformations of the data and also data on the old system is mapped to the new system by providing a design for data extraction and data loading. It enables the user to choose the required format to which the database tables are to be converted. Some of these formats include .doc, .txt files. The entire conversion process is abstracted from the user. It takes place in the background. The user will be provided only with the interface that allows him what to convert and where to convert. The data representation in these converted formats is made using the CSV files.

C. Advantages:

- Automated migration process with complete accuracy
- Cost effective and error free project delivery.
- Higher productivity with enhanced performance.
- Supporting major database platforms.

II. PURPOSE & SCOPE

In order to transfer the schema and data across Oracle, IBM DB2, SQL Server, MySQL MySQL and MS Access databases. Migration of data from Excel and CSV (Comma Separated Value) files into databases. Migrating tabulated values from one format to another format is achieved by using intermediary CSV files. Tabulated results of one format are transformed into CSV file format which are again converted to the target format. The schema of the RDBMS must be preserved during data migration over incompatible databases. Criteria based data migrate is supported when migrating from Oracle, SQL Server, MySQL, to any database. User can set the filter criteria for a table(s) based on which, only the rows matching the criteria (subset) will be migrated. If a criterion is set for a table, then its dependant tables will also be migrated based on its relationship [8].

III. MODULES

The modules in Format Migration are:

A. User:

This module includes various operations that the user performs. The tasks involves creation of data bases, verifying whether migration was done completely. User also chooses the other modules upon which the operations would be performed. Specifying the database names, Checking the integrity of the data migrated , providing the required target extensions are the tasks that a user must perform.

B. Database Migrator:

A database migrator is the actual tool that acts as an interface between the user and the actual process of migration. It allows the user to select the source and target databases as well as the files. It abstracts the user from the actual implementation of the program and performs the migration. It is solely responsible for acknowledging the user about the success/ failure of the migration/conversion.

C. Same Database Migration :

The user migrates the database tables within same database based on dsn. User has to select the source database name with dsn, username and password. For target he has to give dsn, username and password. Here, the username and the

password are used to login to the system by the tool. The data in the source database is extracted and mapped into the target database besides preserving the schema of the source database. The user can also choose which tables are to be migrated.

D. Different Database Migration :

Here, the user can migrate database tables from one database to other database . User has to select the source database name with dsn, username and password. For target he has select target database name with dsn, username and password. Similar to the case above, the tool validates these credentials into the source and target databases and verifies the availability of the source and target databases. It also checks whether the source database consists of any data or not. Then the data is extracted and mapped to the target. After loading this mapped data into the target system it must intimate the user about the success/failure of the migration through messages. The user can even specify to migrate only specific tables into the target database.

E. Save Database file with different file extension :

In this module the user can save the database file with different format. User has to select the database name as source with dsn, username and password. The file can be converted to any format by selecting an extension, such as .doc,.txt,.csv. from the database tables as well as MS Access. User selects the target extension of the database tables that are to be converted. Then a prompt appears asking for the filename and the target location to be specified. After validating the credentials of the database, this converted file appears in the target location with all/ selected tables converted into the required format. The schema of the source and target database in all these modules is depicted by an xml file which will be abstracted from the naive user. All these modules make use of the csv files (comma separated value) to generate the target database tables. Csv files are the intermediary files generated by the tool to preserve the dependencies between the tables of the database [9].

Test Results

mysql home page:

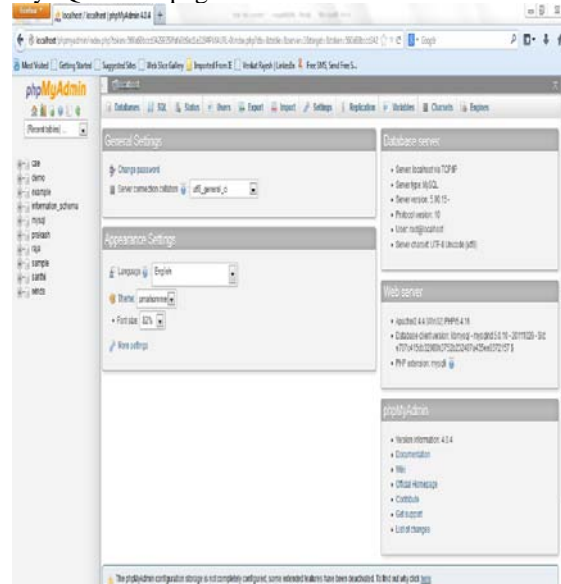


Fig 1: mysql home page

Migration into the same database

Test.No	Input	Expected behavior	Observed Behavior	Success / Failure	Status
1	Without selecting the source database .	It displays Invalid source details	Error is displayed	Success	Yes
2	Without entering the username and password of the source and target database .	Error should be displayed Source database must be selected.	Error is displayed	Success	Yes
3	Selecting invalid DSN	It displays Invalid source details	Error is displayed	Success	Yes
4	Without specifying the target.	It displays Invalid details	Error is not displayed	Failure	Yes
5	Giving invalid or non existing tables	Error should be displayed	Error is not displayed	Failure	Yes
6	Not selecting the target extension in Format conversion	Conversion shouldn't occur	No action occurred	Success	Yes
7	Not saving the file during format conversion	Conversion shouldn't occur	No action occurred	Success	Yes
8	Without specifying the				

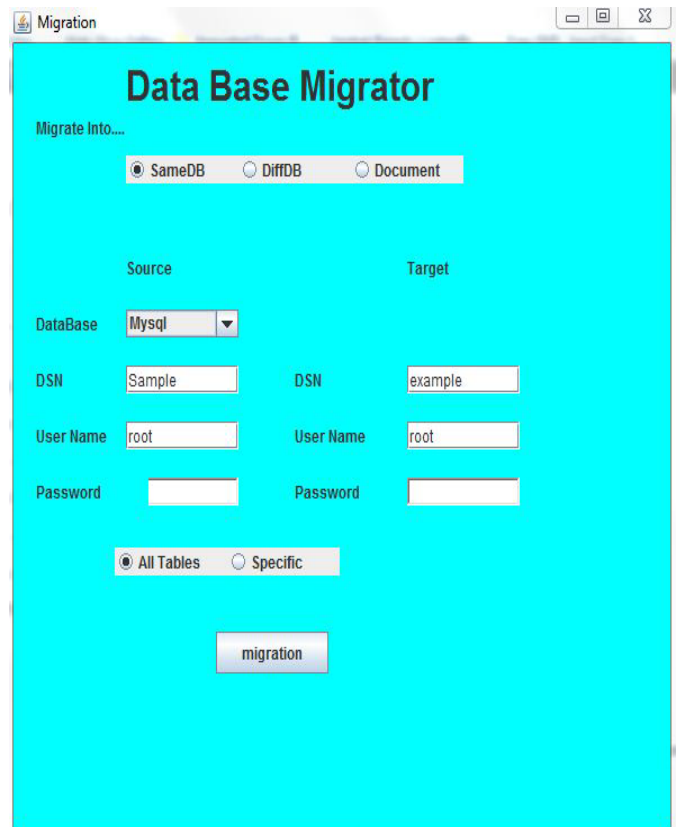


Fig 2: Migration into the same database Migration in same database:

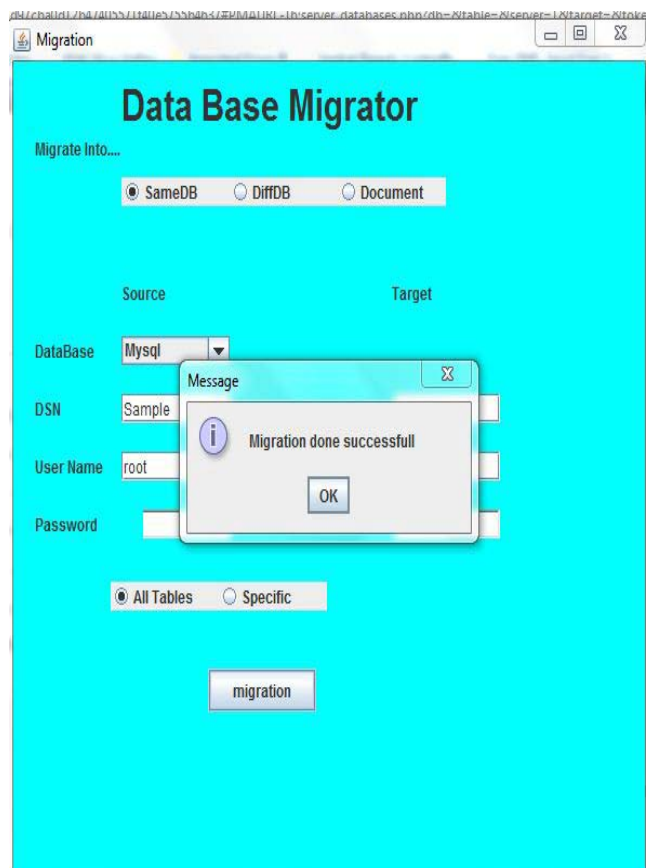


Fig 3: Migration in same database

Table name:csp								
id	username	password						
1	csp1	csp1						
2	csp2	csp2						
Table name:cspownerdetails								
id	name	age	gender	email	password	mobile		
1	Kishor	25	male	dvenkatid@customer	9.95E+09			
Table name:ttp								
id	username	password						
1	ttp1	ttp1						
2	ttp2	ttp2						
Table name:userdetails								
id	name	age	gender	email	password	mobile		
1	Kishore	25	male	customer@customer	9.95E+09			

Fig 4: Converted cvs format

IV. CONCLUSION

Automated migration of data between the databases would minimize the extent of human interaction in dealing with the databases. In fact, a programmer need not rely completely on writing sql queries for updating the data into the database. Large amount of data can be processed by using this tool in a very fast and efficient manner. Even a naive user without any programming knowledge would be able to perform the migration.

It also supports migration between major database platforms. It helps us in having immense information about the tables available in each database and further provides an option to choose what data is to be migrated particularly. It is a user friendly tool and provides the user with success or error messages. Through this we can avoid time delay, maintain accuracy and low man power.

V. FUTURE ENHANCEMENTS:

User requirements keep changing as the system is being used. Some of the future enhancements that can be done to this system are:

- As the technology emerges, it is possible to upgrade the system and can be adaptable to many more databases.
- Improving the GUI further enhances the usability and user friendliness.
- Developing a web based Format Migration tool that lets users to transfer their data over the internet.
- Enhancing the scope of the tool to work in networked devices viz, LAN, WAN etc.

REFERENCES

- [1] Van der Hoeven, Jeffrey; Bram Lohman; Remco Verdegem (2007). "Emulation for Digital Preservation in Practice: The Results". *The International Journal of Digital Curation*. 2 (2): 123–132. doi:10.2218/ijdc.v2i2.35
 - [2] Muira, Gregory, "Pushing the Boundaries of Traditional Heritage Policy: maintaining long-term access to multimedia content" (PDF). *IFLA Journal*. **33**: 323–326, (2007)..
 - [3] "Apache Tomcat - Welcome!". Apache Tomcat. The Apache Software Foundation. 2017-10-01. Retrieved 2017-10-02.
 - [4] "Apache Tomcat - Welcome!". Apache Tomcat. The Apache Software Foundation. 2017-09-30. Retrieved 2017-10-02.
 - [5] "Apache Tomcat Configuration Reference - The Realm Component". Tomcat.apache.org. Retrieved 2013-11-01.
 - [6] The Coyote HTTP/1.1 Connector, Apache
 - [7] The Coyote JK Connector, Apache
 - [8] King, Graham (2004-12-20). "High availability Tomcat - Connect Tomcat servers to Apache and to each other to keep your site running". *JavaWorld*. Retrieved 2013-02-13.
 - [9] JRemy Maucherat (2013-10-18). "Apache Tomcat 7 (7.0.47) - Additional Components". Tomcat.apache.org. Retrieved 2013-11-01.
- "Apache Tomcat Versions". Retrieved 2011-11-12.