TDG-A Tool for Automated Test Data Generation

Kuldeep Arora¹ and Dr. Jaiteg Singh²
¹ Research Scholar, Punjab Technical University
Jalandhar-Kapurthala, Punjab 144601, India
² Department of Computer Applications, Chitkara University
Rajpura, Punjab 140401, India
Kuldeep.arora2003@gmail.com¹, Jaitegkhaira@yahoo.co.in²

Abstract
Software Organizations needs Test Data depending on test environment that suits most their needs for database application testing. The concept related to synthetic data generation for testing a big software project with more and more coverage. Typically Synthetic data is used for this purpose intended to be used for Performance Testing, Load Testing, Capacity Planning, etc. Organizations use many of the data formats as backends such as Mysql, Oracle etc. So the authors put efforts to create a tool that generates bulk synthetic data that support all major data sources. This paper discusses challenges faced during implementation of TDG.

Keywords: Synthetic Data, TDG, Fill Rules, Data Source, Schema

1 Introduction
In this paper we work on the tool TDG that generate synthetic data for all major database formats like Oracle, Mysql, MS SqlServer etc. It supports the generation of all categories of synthetic data & also protects the sensitive information from the misuse, with high coverage.

Benefits of using Synthetic Data

- Testing the Performance of database used in the application- such as time taken to execute the query to database.
- Testing the Data- Mining Applications.
- Secure Application Development- i.e. Privacy issues.

Testing database applications are much more challenging day by day. For Capacity Planning, Performance testing and Load testing application developer requires high volume of data in Gigabytes & Terabytes. The use of Real Data for testing is not always possible because of privacy issues. The Act Data Protection also enforces the companies and individuals to keep personal information to them. So the organizations have only the option for using the synthetic data. So it is necessary to generate the synthetic data.

Wherever it is necessary to use Production data we can use the mix of synthetic data and real data is the best way. Only appropriate amount of mix synthetic data and real data meet the requirement of the both the data security and the requirement of testing.

In all the case above manually generation of data is not possible when mass copy of data is requires. It requires automation.

Kuldeep Arora and Dr. Jaiteg Singh: TDG-A Tool for Automated Test Data Generation
2. Synthetic Data Categories

The following are the categories of synthetic data that is used for testing:

**Numeric data**: Numeric data may be list of numbers with Random digits or sequential series generation.

**String data**: String data consists of a list of strings with random or sequential series.

**Percentage base data**: This feature requires generation of data items with corresponding percentage in a list.

**Data extraction**: Data Extraction is the feature that require the generation of data from the other database table.

**Data from libraries**: Libraries are like repositories of data within their domain. Libraries are used to generate synthetic data.

3 Data Protection

Data scrambling techniques are used to protect the data confidentiality when extracting data from Production Data. Data Scrambling is a process of transferring data from the production database into a test mode database. It removes sensitivity from the sensitive information. Some of the Data Scrambling techniques are following:

- **Nulling Out**: Nulling Out the sensitive information field with NULL Value.
- **Data MASK**: Mask the data by replacing the some part of values in a field by a mask character (such as X).
- **Substitution**: Substitution is the technique consists of randomly replacing the contents of a column of data with information that looks similar but unrelated to real details.
- **Shuffle**: Shuffling is the technique that substitutes the data from the column itself.
- **Encrypt**: Encrypting the sensitive data with a key value.

4 Approach TDG

The figure showing approach that how the data is generated & puffed into the desired data source.

Data Row and Data set Data Structures are used to process the generated data. Data row represents one row with similar schema of the designated table. One by one generated data is placed into a Data Row and then full Data Row is transferred into the data set; the data set is further updated with desired data source.

5 How the Data is generated?

Now we discuss how the synthetic data is generated and Data Row is puffed up. Randomizer Modules contains the code that generates synthetic data for specific category. Randomizers needs seed values to generate data. So
the specific Randomizers is called for generating data by passing the seed values that are set by the users. Generated data is puffed into Data Row

Figure-2 explains how the particular module is called to generate the data. The challenge is to provide all the randomizers for each column of the table and internally how the generated data is puffed into data row Data Structure.

The following TDG snapshots helps in understanding the concept.

Snapshot-1: showing the randomizer ‘RandomDate’ for a column & Seed Values

Snapshot-2: showing the randomizer ‘WEIGHTEDLIST for a column & Seed Values

A customized User Control is made to store the seed values for Randomizers to generate the data. The control will be displayed for each column (or field name) of the selected table. Figure-2 depicts the approach that how the data row Data Structure is puffed with the synthetic data. Specific randomizer Module is used to generate data on the basis of rule selected in the user control.
6 Project Challenges

In this section describing the challenges experienced during building of Tool by authors. The Tool TDG (Test Data Generation) was developed to help testing team to generate high coverage data to test software. The domain of the tool is so vast it can be used for developer and testing team to test the software, and to generate the dataset for researchers and policy maker to explore the decisions.

6.1 Challenge: Connect to a data source

DTM Test Data Generator uses ODBC Driver to connect to each data Source which enforces user to install particular driver to data source. But TDG uses .net specific provider it does not requires installing driver’s specific to Data Sources.

TDG deals with relational Database Servers like Oracle, MS SqlServer, Mysql, and Data Files like MS Access, MS Excel, Xml Files, and Text Files. These are the major Data Sources that can be used as Databases in an Application. Apart from these Data Sources there is also provision to connect User DSN’s. VB.net is used to develop the tool. There is provision for connecting to remote database server also such as Oracle, MS SqlServer, and Mysql.

The choice is made to use .net Provider instead of OLE DB Provider. The benefits of using .net providers are:

- Boost the performance and scalability for accessing data
- Support data access by SQL & Direct table access
- Also support for disconnected recordset
- Gives Complete Details for referential integrity i.e. Primary/ Foreign Key
OLEDB does not provide details for Referential integrity. Different Function Modules are written using .net specific provider to connect to each type of Data Source.

<table>
<thead>
<tr>
<th>Data Source</th>
<th>Provider/ Driver</th>
</tr>
</thead>
<tbody>
<tr>
<td>MS SqlServer</td>
<td>System.Data.SqlClient</td>
</tr>
<tr>
<td>Oracle</td>
<td>System.Data.OracleClient</td>
</tr>
<tr>
<td>Mysql</td>
<td>Mysql.Data.MySqlClient</td>
</tr>
<tr>
<td>MS Access</td>
<td>Jet OleDb, Ace OleDb</td>
</tr>
<tr>
<td>MS Excel</td>
<td>Jet OleDb, Ace OleDb</td>
</tr>
<tr>
<td>Text File</td>
<td>Jet OleDb</td>
</tr>
<tr>
<td>XML</td>
<td>Used Function of System.XML XMLTextReader</td>
</tr>
</tbody>
</table>

5.2 Challenge: Schema Extraction

As TDG offers connectivity with all the major data sources such as MS Sql Server, Oracle, MySql, MS Access, and Excel etc. Each data source has different mechanism to store the schema. So the following mechanisms are used:

**Oracle**: the oracle store all table information in ALL_TAB_COLUMNS & ALLCONS_COLUMNS

To know all the column name of a table –
SELECT COLUMN_NAME FROM ALL_TAB_COLUMNS where TABLE_NAME= abc

To know the primary key of table -
SELECT COLUMN_NAME FROM ALLCONS_COLUMNS A JOIN ALL_CONSTRAINTS C ON A.CONSTRAINT_NAME=C.CONSTRAINT_NAME WHERE C.TABLE_NAME='abc' AND C.CONSTRAINT_TYPE='P'

**MS Sql Server & Mysql**: These data sources stores it’s all information’s of table in Information_schema

To know all the column name of a table –
SELECT COLUMN_NAME FROM Information_Schema.Columns where TABLE_NAME=’abc’

To know the primary key of table -
SELECT C.COLUMN_NAME FROM INFORMATION_SCHEMA.TABLE_CONSTRAINTS K,INFORMATION_SCHEMA.KEY_COLUMN_USAGE C WHERE C.TABLE_NAME=’abc’ AND CONSTRAINT_TYPE=’PRIMARY KEY’ AND k.constraint_name = c.constraint_name

**MS Access & other file formats:**

By making dataset of the table the schema is extracted of the file formats. By using the property ColumnName & DataType the schema is extracted.
5.3 Challenge: Export the data generated to other data source

The data generated must be saved somewhere, many Data generator tools provide facility to save data to excel, xml, csv and text etc. but no one provide facilitate to export to other data source.

The issue is how to convert the schema from one data source to another. Figure-3 shows the logic to export the data generated of one data source to another data source.

First of all schema is created to new data source with resemblance to connected data source after that generated data is inserted to new data source.

![Diagram](image)

**Figure 3 Logic to implement Export Data.**

5.4 Challenge: Create Database in Oracle

TDG offers user to create own Data Base for all data source formats. One major issue faced during creating the data base in oracle. The creation of data base file is done by the DBA. In Oracle the creation of database is DBCA. But here we provide the different process to create database in oracle. We just automate the manual process of creating database in oracle.

This consists of some manual steps to perform for creating database in oracle without DBCA.

1. Create initial directories where your oracle base is
2. Set `oracle_sid = <new database name>`
3. Start new oracle service using new sid and sys password
4. Copy init.ora file in pfile directories
5. Now the initial steps are performed next step is to give create database command
6. Now the desired database is created.

Our solution for automate is:

First 4 steps are performed in Batch file using DOS Commands & then CreateDatabase.sql file is called in Batch file to give the command to create database.

One more issue that occurred here is that tnsname.ora file is edited during Creation of database in oracle. It is very difficult to perform through the commands. So our solution is to use the tns entry in connection string without entering these details in tnsnames.ora file such as:

"Data Source=(DESCRIPTION =(ADDRESS_LIST =(ADDRESS = (PROTOCOL = TCP)(HOST =<servername>)(PORT = 1521)))(CONNECT_DATA =(SERVICE_NAME = "<Database name>"));User Id="<UserID>";Password="<Password>";"

5 New Features Introduced

We are encountered with the following shortcomings in the available tools such as DTM, TurboData, Spawner, GenerateData while building the TDG. Authors introduced many new features in the TDG.

- User can create their own schema in any of the data source where others do support in only csv or delimited.
- Export generated synthetic data to any of the other data source.
- Easy to use interface
- Generate data for decision making.
- Many new fill rules such as Ordinal type (generate random series of ordinance data like 3241,4231 etc)
- No need to install drivers to use particular data sources such as MySql, ODBC, ACCDB, XLSX files.

6 Conclusion

A tool TDG is built to generate the synthetic data for various types of testing the software’s and for decision making where initially datasets are not available. Our tool generates data in bulk with high coverage of code. TDG can interact to all major Data Source with generating quality data.

Many new fill rules are introduced such as user defined libraries, ordinal type (for generating data in ordinance) and many more. Apart from these features TDG also generates synthetic data for decision making.

7 References


