dbe SOFTWARE

DBE Software, Inc.

6842 Elm Street

McLean, VA 22101

USA

E LCS@dbesoftware.com

P +1-703-847-9500

F +1-703-991-2500

Why Data Quality is not Enough

Data quality today is one of the hottest topics in any IT shop. Although a very important topic, Data Quality is far from being enough.

Information is the Key Ingredient for Decision Making

Decisions are made based on information produced by application programs that use relational database management systems (DBMS) such as Oracle, Sybase, SQL Server, DB2, etc... These database systems are comprised of the database schema and data. The schema is the core of the database, containing the data definitions, data associations and business rules.
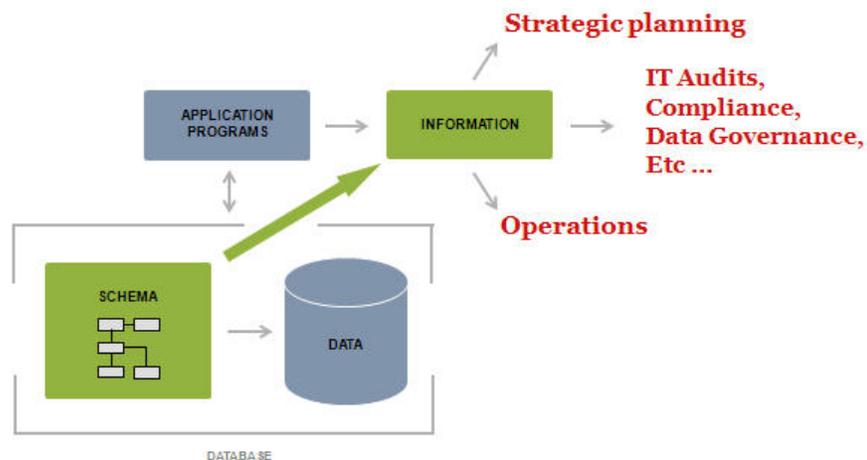

How to Obtain Quality Information

You can see that the quality of the information depends on: (a) the quality of the data stored in the database, (b) the quality of the programs that manipulate the data and (c) the quality of the database schema which controls and directs the data. For the information to have quality, the original data stored in the database must be correct, the programs must be right and finally, the schema must be flawless. If one of the components is not perfect, the resulting information will be flawed.

The Missing Piece—DBE Software's Database Examiner

Data Quality is the process of ensuring the quality of the data stored in the database. The most common procedures in use today are Data Cleansing and Data Profiling. These are important steps, but they are usually applied to the Customer data file.

Software development methodologies such as Carnegie Mellon are used to improve the quality of the application programs. As most corporations use these methodologies, this is not a major issue anymore.

The database schema must be validated for flaws. Examiner is the missing piece of the puzzle. It validates the database schema, produces 50+ diagnostics, explains the issues, provides a Teach Me facility, makes suggestions and, in the situations that do not require human intervention, generates the corrective DDL scripts.
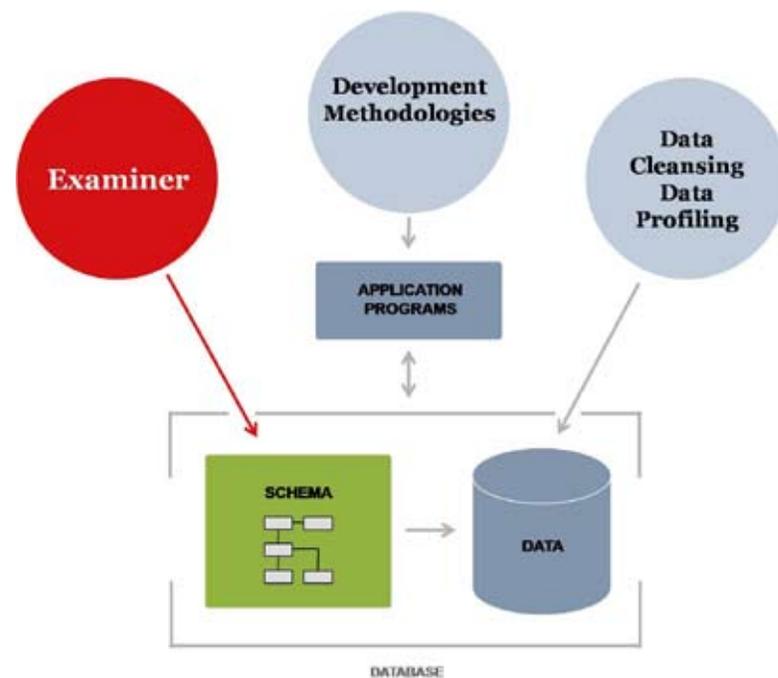
Quote from Joe Celko

Joe Celko (www.celko.com), a well know database expert and published author on relational databases, states that with a poorly designed database schema, it is very difficult to:

– Achieve application performance and
– Produce quality information.

So, the important issue is to ensure that the database schema is flawless at all times. And how we do that? With Examiner.
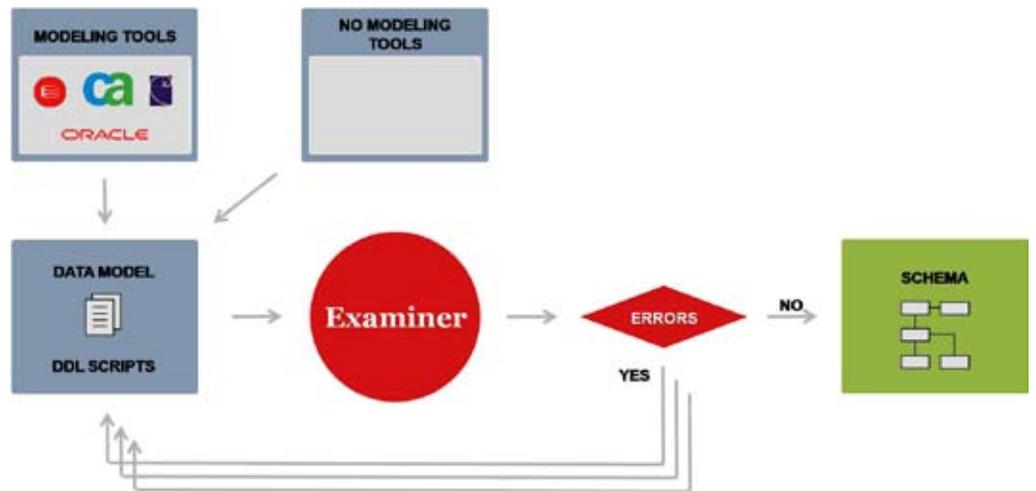
Uses of Examiner

Examiner can and should be used during the entire life-cycle of the application. Its main uses are in the development and maintenance phases of any project. Any time the schema is updated, it must be validated again. Database schemas are getting so large and complex that is basically impossible to validate them manually.

Proposed Development Process Using Examiner

What we propose is to fully "debug" the database schema before any programs are written and the database populated. This is the ideal approach that should be used for any new development. If the database already exists and/or we are in the maintenance phase, we will show in this paper how to use Examiner in a similarly efficient way.

As soon as the data model is created, it should be validated with Examiner. If a modeling tool is not being used, Examiner will validate the SQL/DDL scripts, or the database schema by reverse-engineering it.

Here Examiner is validating the data model (or scripts) and checking for errors and inconsistencies until these are eliminated. If we find any problem, we go back to the model and fix it. When the model is error free, we allow the creation of the database schema. This saves valuable time and reduces costs. All errors are detected at the data model or script level. This way the schema will be free of flaws.

Proposed Maintenance with Examiner

Another must use of Examiner is when an application needs improvements that requires changes to the database schema. The ideal situation is to evaluate the changes without making them. Examiner allows this by simulating the "merging" of the database schema with the proposed DDL scripts for the change. In this case we test the combined "schema" and check for any errors or inconsistencies. If there are problems, we then fix the proposed changes until there are no more errors and a new schema will be created. You can see that simulating the changes before imple-mentation saves time and money.
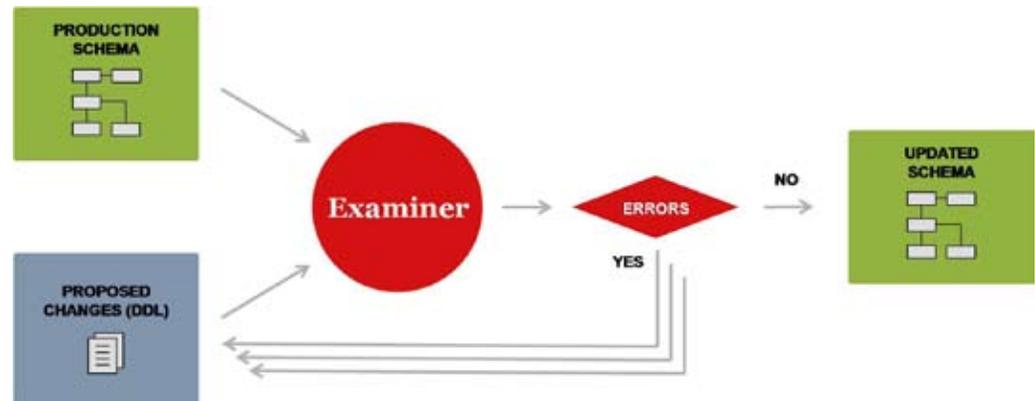
Examiner's Role

Examiner uses an extensive knowledge base about the relational model to validate the data model, SQL/DDL scripts, or the database schema. It detects the problems automatically, explains the issues and the impact, and makes suggestions on how to fix the issues. When there is no need for human intervention, Examiner generates the corrective DDL scripts.

In summary, Examiner ensures the quality of the database schema, thus minimizing the development cost and helping improve the quality of the information produced by the application programs.

Examiner provides the following benefits:
– Modelers produce better data models/schemas
– Applications are deployed earlier
– Development cost is reduced
– Programs run faster
– Data models/schemas are evaluated by a standard criteria
– Information will have quality

It is extremely difficult to measure the ROI of projects that improve the quality of the information, but it is clear that decisions based on quality information provide enormous saving.

A tangible ROI can be measured based on:
– Reduced development time
– Associated reduced development costs
– Reduced delivery time
– Reduced number of applications being scrapped

A couple of ROI examples can be found in our web site.

Prepared by **Luiz C. Siqueira**
on February, 2008