
DBE Software, Inc.

6842 Elm Street
McLean, VA 22101
USA

E LCS@dbesoftware.com

P +1-703-847-9500

F +1-703-991-2500

Optimizing the Development Process

The Gartner Group estimates this cost of poor data quality to US business alone to be over \$600 billion per year.

It is all About the Database Schema

All major applications 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. If the schema has flaws, the information will become corrupted, truncated, suspicious, unreachable, or just wrong.

Debugging the Schema

The schema is like a computer program and must be “debugged” to eliminate flaws. Just because the schema “compiled” and no error messages were found does not mean that the schema is without structural flaws or inconsistencies.

Programmers use “code review sessions” to make sure their programs have no flaws. Data Designers should use a schema review process to ensure the schema has no flaws. Because there is no formal process to review the schema, they are usually not checked for errors. Schemas today are so big and complex that it is impossible to validate them manually.

Enter DBE Software's Database Examiner

Most schema errors are discovered at the time the application programs are being tested. This is far too late in the development process. Examiner will identify errors in the schema long before the applications programmers start their work. This saves an enormous amount of time and resources for both the designers and the programmers.

If schema errors are found and corrected after going into production, the task to correct these errors grows exponentially. In the worst case, the data must be downloaded from the database, the schema modified, the data uploaded to different parts of the data structure, and finally, the affected programs must be altered and re-tested. This is a major flaw in the development process that results in exorbitant cost.

Schema errors must be found as soon as possible – at the data model level or when the schema was just created. At that time, there is no data to deal with or programs to modify. This is a very simple task utilizing Examiner – simply click once to validate and fix the schema. The cost of the development process will decrease by an order of magnitude and the information generated will be sound.

Today's Development Process

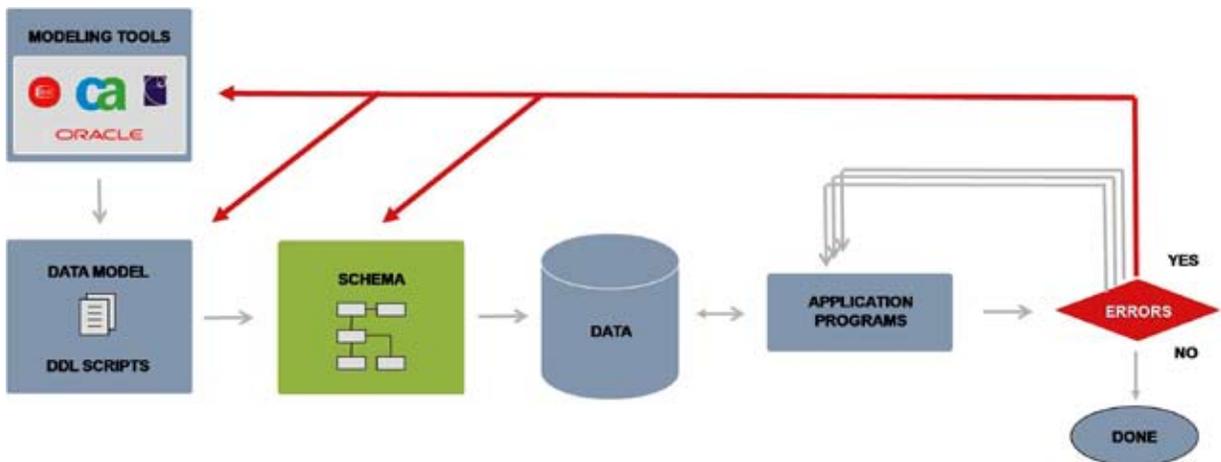
Most organizations use a modeling tool such as CA's ERwin®, Embarcadero's ER/Studio®, Oracle's Designer, Sybase's PowerDesigner and/or others to develop the data model which is

the foundation of the database schema. The modeling tool generates the SQL/DDDL scripts that are submitted to the database system that “compiles” it into the schema.

Companies are making every effort to improve the quality of their data and of their application programs, but very few understand the need for a high quality database schema. They assume that having a good data modeler using a good data modeling tool, such as ERwin®, Sybase PowerDesigner, Oracle’s Designer or Embarcadero®’s ER/Studio®, will ensure their database schemas will be sound and flawless. **This is far from reality.** Without a quality schema, the chances of generating quality information are very slim. In addition, the development process will be slower and costlier.

The Problem with Today’s Development Process

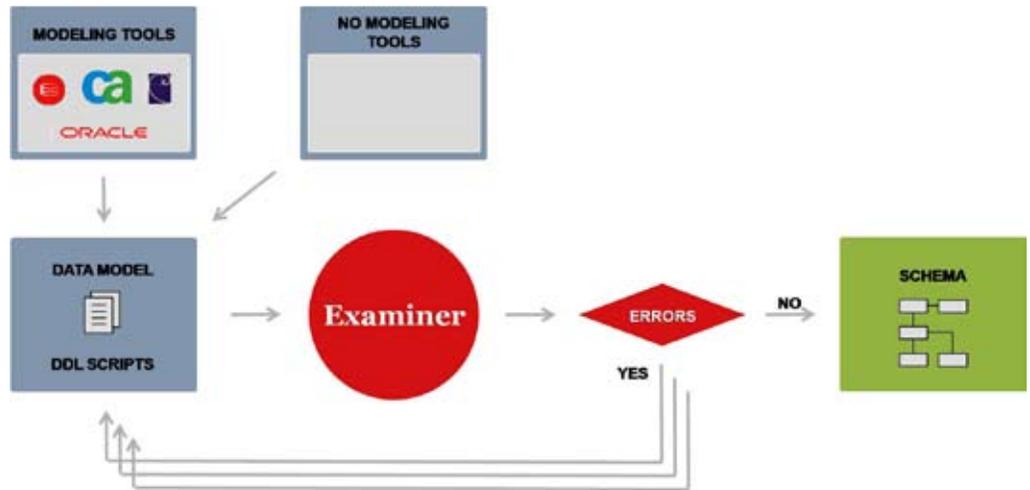
As noted, most modelers do not “debug” the schema, thus the schema will have a high probability of major flaws that may be detected during the program testing phase. If the flaws are detected during program testing, then the schema is returned for correction. This may take many iterations, making the process longer to complete. Without Examiner there is better than a 50-50 chance that major flaws are not detected and end up in the production application. This is why applications are delivered late with higher costs, and produce poor information.



Proposed Development Process Using Examiner

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.

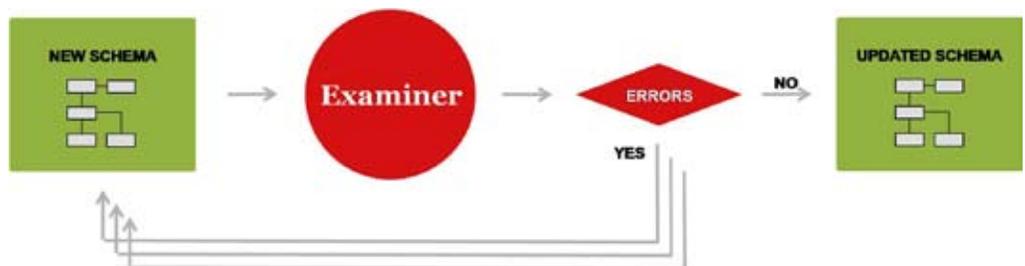
As soon as the data model is created, it should be validated with Examiner. If a modeling tool is not being used, Examiner will validate the SQL/DDDL scripts, or the database schema by reverse-engineering it.



Here Examiner is validating the data model (or scripts) and checking for errors and inconsistencies until there are none. If we find any problem, we go back to the model and fix it. When the model is error free, we then allow the creation of the database schema. This saves valuable time and reduces the costs.

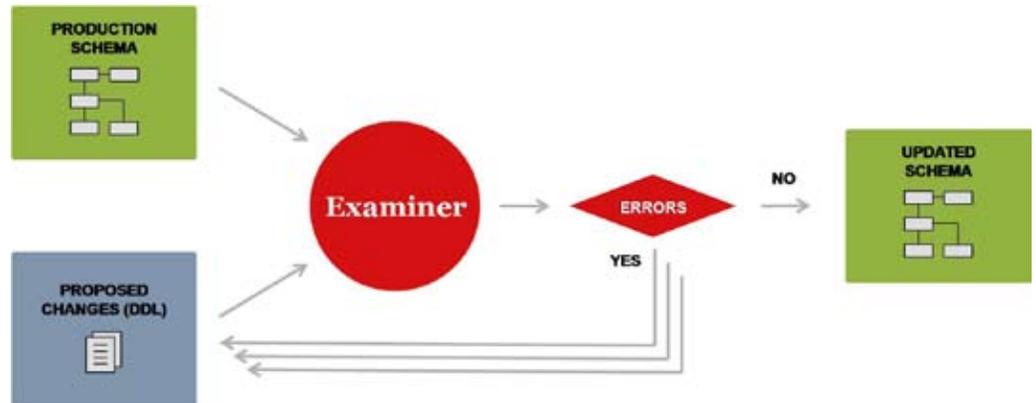
You can see that all errors are detected at the data model or script level. This way the schema will be free of flaws.

If you start with the database schema of an unpopulated database instead of the data model, the process is similar. Examiner reads the database schema, performs the validation, and detects and displays the errors. The user fixes the errors until no more errors exist. At that time, the updated schema will have no more 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 implementation saves time and money.

Examiner's Role

Examiner uses an extensive knowledge base about the relational model to validate the data model, SQL/DDDL 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.

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

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