
Towards an Agile Database

EC Wise, Inc.

info@ecwise.com

© 2001, EC Wise, Inc., All rights reserved

© 2001, EC Wise, Inc., All rights reserved

The EC Wise team has many years of applied database management and analysis experience. By leveraging this experience with our expertise in automated code generation, in software architecture and design, and in machine learning, we can provide unique value to organizations building and maintaining substantial data oriented systems. Here are some of the things we can do to accelerate your database design, development, and maintenance efforts:

In our experience, an emphasis on built-in testing and extensive automation can greatly reduce the labor required to complete a project. Clean database instances, fully tested and with sample data, often take a week using traditional methods, and the length of this cycle forces developers and testers to find workarounds. With our practices, they can have several new instances a day, or whenever needed. On a larger scale, rather than taking weeks to make several changes to the database structure, with attendant changes to referential integrity, stored procedures, and sample data, these changes can be released in a daily cycle. The increased visibility, testability, and repeatability in the database means that problems are discovered much earlier in the process, and all parties can have a greater degree of confidence in the finished product.

Design and design evaluation

Design databases that effectively support OO software

An object oriented design and development approach provides the best chance to realize a software solution optimized for adaptability and evolution. However, determining the right database design to most effectively support an object oriented design model can be tricky. Having extensive experience in just this transformation, we can quickly analyze a software design model and specify a first cut database design to support it. From there, we can iterate the design in accordance with changes in software components, and make adjustments to reflect additional information on load requirements and performance criteria that become available during the development process.

Identify and correct database design problems

We can review database design against industry standards and standard practices, and against your software architecture models, and identify any discrepancies. Even better, we can show you how to use automated script generation, sample data generation, and testing together to fix database design errors quickly, without impacting overall project schedules

Design models for effective dating / audit trail

Most database applications require an audit trail for transactions and changes in data values. Many have an additional requirement for the operator to specify an effective date for a transaction or change entered on another date. There are established patterns for addressing these requirements. Based on these patterns, we can formulate a suitable scheme, or work with an existing scheme, for effective dating and/or an audit trail. We can make this invisible to most of the middle tier code if desired.

Design for automation

We believe that computers are the most efficient resources available for most database development and maintenance tasks. They don't make scripting errors, and they can fully document all of the steps that they have performed. Building a system that is amenable to a high degree of automation requires careful consideration during the design of the database. We can help design the database to facilitate automation.

Developing database logic

Automated script generation

We've done extensive work generating stored procedures and scripts, both to save time and to provide features such as:

- Logical deletes
- Effective dating
- Auditing
- Debugging / display

Self-Testing Code

One of the most time consuming aspects of making database structure changes is testing them. While automated testing of database modifications is feasible, its not widely performed. We have a library of assertion utilities and practices that allow us to build testing into the scripts. We can change the level of display and recording of the test code as we like, for reasons of performance, unattended running, or visibility.

Sample Data

Another time consuming aspect of making database structure changes is updating the test and/or sample data to match the new databases. This is especially true where the changes result in combining or splitting up tables. In many cases, the necessity to update extensive libraries of sample data brings development projects to a halt; the specter of such changes puts a chilling effect on proposals to make changes that would drastically improve performance or flexibility. We address this problem by automating the generation of sample data. We can generate sample data for:

- Requirements testing
- Functional testing
- Load testing
- UI / middle tier testing
- Demos and prototypes

This can also be part of building personas and records of usage for the more subtle UI issues like adaptive interfaces.

Documentation

Here we can produce automated documentation, such as HTML or XML descriptions of structures, and API docs for the stored procedures, to be used by middle tier programmers.

We can also document procedures and best practices for:

- Testing
- Sample data
- Automation
- Stored procedure coding

Testing

We automate much of the testing of database logic, reducing the QA load dramatically. We can write scripts and communicate techniques for testing the existing stored procedures, views, functions, uniqueness constraints, not-null constraints, and triggers.

Performance Tuning

This is reasonably straightforward, but traditionally very time consuming and detail oriented. With our emphasis on automation, we can do a lot of this work quickly, and report on it consistently.

Analysis

EC Wise has built a library of algorithms optimized for mining and identifying patterns of information in large volumes of data. If your organization has large databases and you want to discover relationships hidden in the data, we can work with you to evaluate existing solutions, or help you develop a customized analytical program to uncover them.

Maintenance

Automated database maintenance

We have built our ability to automate database management tasks on a strong foundation of high performance code generation technology. This allows us to automate administrative tasks such as:

- Checking for consistency in the structure, such as identity fields, cross-reference tables, and naming conventions.
- Updating the structures across the board.
- Moving data into test and production databases.
- Creating meta-tables for effective dating and auditing.

Again, the net result is less time required to perform maintenance tasks, and a vastly higher confidence that they are done right the first time.