NEWS · MAY 26, 2026

TMS Data Modeler is now on GitHub — and that's a bigger deal than it sounds

After 20+ years as a closed-source commercial product, TMS Data Modeler's full source is now on GitHub. Here's what that actually means for Delphi developers and why I think it matters more than the announcement lets on.

By Dr. Holger Flick

If you've been writing Delphi for any length of time, you know TMS Data Modeler. It's the visual ER-diagram tool that quietly sits on a lot of our machines — the one you reach for when you'd rather draw a schema than hand-write `CREATE TABLE` statements across six different SQL dialects.

On May 18, TMS dropped something I genuinely did not expect: the **full source code** of Data Modeler 4.0 on GitHub. Twenty years of commercial Delphi code, in a public repo you can clone right now.

What you actually get

For the developers who've never used it: Data Modeler is the tool that lets you:

- Draw entity-relationship diagrams visually
- Generate DDL for **SQL Server, Firebird, Oracle, PostgreSQL, MySQL, and SQLite**
- Produce ALTER scripts to migrate a live database to a new model version
- Reverse-engineer an existing schema into a diagram
- Diff two versions of a model
- Convert a model from one DBMS to another (the killer feature, honestly)

That last one alone has saved me days when a client said "actually, can we move from Firebird to PostgreSQL?" mid-project.

Why this is a big deal for the Delphi crowd

Here's the part the TMS announcement undersells. For most of us, the value isn't the tool itself — we already had it, and most pros paid for it anyway. The value is everything *around* the code being available:

1. It's a real-world, large-scale Delphi application you can read. Tutorials are great, but nothing teaches you idiomatic Delphi like reading 20 years of code from a team that ships. The repo uses **Diagram Studio, Scripter, Aurelius, and VCL UI Pack** — these aren't toy components. If you've ever wanted to see how a serious Delphi app composes those pieces, your reading list just got a major addition.

2. Aurelius users, take note. Data Modeler is itself a heavy Aurelius consumer. If you're learning Aurelius or fighting it on a hairy schema, this codebase is now your reference implementation.

What I'd do with it this week

If you want to get value out of this release immediately, I'd suggest three things in order:

1. **Clone the repo and build it.** TMS rebuilt the CI/CD pipeline as part of the release, so this should actually work cleanly — which is rarer than it should be for 20-year-old codebases.
2. **Read one feature end-to-end.** Pick the DDL generator for your favorite DBMS and follow the call chain from the UI button to the emitted SQL. You'll learn more about Delphi architecture in an hour than from a week of blog posts.

The pattern I hope continues

TMS isn't the first vendor to publish source like this and I hope they aren't the last. The Delphi ecosystem has a lot of mature, deeply-engineered commercial code sitting behind closed doors. Every time some of it comes into view, the whole community gets smarter.

If you've been on the fence about Data Modeler, this is the moment. And if you've been using it for years like I have, go star the repo and start reading. There's a lot in there.

Resource	Link
Repo	github.com/tmssoftware/datamodeler
TMS announcement	tmssoftware.com/site/blog.asp?post=2476

INDEPENDENT · AD-FREE · READER-SUPPORTED

Free to read. Not free to make.

Every tutorial, article, and line of example code on Holger's Code is published without a paywall and without ads. This page is about what keeps it that way.

Professional work, published free

The content here is held to the same standard as professional client work: every tutorial is built as a real project, tested against current versions, documented step by step, and revisited when the frameworks move on. A single in-depth tutorial routinely takes days — and that's before the bills behind it: servers, storage, build infrastructure, licenses, and the hardware everything is verified on.

If an article or tutorial here has saved you an afternoon of debugging, taught you a technique, or helped you ship, it created real value for you. Supporting it isn't charity — it's paying for professional work you found useful, at whatever price you choose. And if you can't, keep reading anyway. That's what the free part means.

Ways to support

KO-FI

Buy me a coffee

A one-time tip — no account, no subscription, any amount.

BOOKS & COURSES

Buy the paid work

The paid tier of everything published free here — and you get something lasting in return.

SPREAD THE WORD

Share an article

Send a tutorial to a colleague, cite an article, subscribe to the RSS feed.

Nothing here goes behind a paywall, whether you chip in or not. But if something on this site earned its keep today, this is where to say so.

Support on Ko-fi —