

Minutes of the Gekko steering committee meeting, September 10, 2026, at the DREAM group¹

Participants:

- Ingeborg Vind (Statistics Denmark, DST)
- Michael Osterwald-Lenum (DST)
- Jonas Dan Petersen (DST/QNA)
- Grane Høegh (DREAM)
- Patrick Kristiansen (DREAM)
- Dorte Grinderslev (DREAM)
- Philip Frost (Ministry of Finance)
- Anders Køhlert Larsen (Ministry for Economic and Interior Affairs)
- Lasse Andersen (Ministry for Economic and Interior Affairs)
- Mads Khoa-Dang Dang (Ministry of Taxation and Economic Growth)
- Emil Holst Partsch (Danmarks Nationalbank)
- Rasmus Rose Sørensen (Danish Economic Councils)
- Thomas Thomsen (Gekko editor)

Item 1. Welcome + choice of minute taker + approval of last year's minutes

Thomas was chosen as minute taker, and Dorte as moderator. The minutes from last year were approved without comment.

As mentioned in last year's minutes, this year's minutes are scaled down in scope.

Item 2. Editor's report regarding 2025-26.

The editor lined up the development for the period 2025-26.

Item 3. Status and feedback regarding Gekko 3.2 (and 3.3.x)

The Danish Ministry of Taxation and Economic Growth have had some problems getting Gekko approved by their IT department. The editor mentioned that the IT department of Danmarks Nationalbank has a rather harsh procedure for approving software, but still accepts Gekko.

¹ Links: Gekko main webpage: www.t-t.dk/gekko, organization: www.t-t.dk/gekko/organization. Gekko on GitHub: <https://github.com/thomsen67/GekkoTimeseries>.

Item 4. Prioritization of main lines regarding 2026-27

General discussion points in more or less stochastic order

Regarding Julia tendencies, it was asked why not use Python, which is less niche. The answer lies in the so-called two-language problem, among other things, but this is not a question for the Gekko steering committee to address. Could Julia be allowed to be used at for instance Statistics Denmark? Ingeborg said that given enough value for the institution, there ought to be a way. Still, a Julia package would primarily be for modeling, at least in the beginning. (See also the Gekko Roadmap 2025 paper from last year's meeting regarding Julia and other "host" environments).

Regarding the GREU (Green Reform EU) project trying out Julia. The feeling is that GAMS and gamY (preprocessor for missing features in GAMS) is a bit hard to manage and use, and GAMS Py does not solve all problems in that respect. Therefore, the hope is that Julia, together with the JuMP package + further development (to emulate gamY features and more) + an interface to the CONOPT solver could provide a single unified development environment.

There is a so-called "DUAL" project at DST, trying to modernize software use. The project is not finalized yet, but it is probably the case that offering Gekko to be able to run under Python, with the use of parquet files would be seen as positive and less "legacy".

Some users still feel that PyGekko is a bit "clunky" syntax wise, and error messages perhaps do not always propagate back to the user. But PyGekko means that it *is* possible to make Gekko part of a Python pipeline.

Porting Gekko from .NET Framework to .NET Core was discussed, postponing the migration until the last minute was advised against.

The editor would like to give seasonal adjustment (JDemetra+) a shot soon, not least to get QNA migrated to a newer Gekko version.

Gekko's DECOMP is great for book-keeping stuff, merging up contributions. For instance taxes etc., but should also work for GDP growth contributions and similar (given that the equations are correct). What is difficult is effects that ripple through large parts of the model, including through leads, though it was mentioned that Gekko can actually handle DECOMP with leads, too.

The editor would like the DECOMP engine to be as cleaned up as possible, so that the linearization, the linear algebra part, the pivoting/aggregation is well-defined.

It has been suggested that DECOMP could collapse/aggregate quarters into annual frequency, which is useful for a quarterly model like MONA. Perhaps because of the linearity of the DECOMP contributions, this is not too hard to do?

The editor would like a clickable COMPARE window, to make comparing databanks easier.

Data tracing: there are some edge cases with glitches, but overall, data tracing seems to work. A trace bug was mentioned regarding Z-variables after a SIM: for these, traces ought to survive.

Regarding the `balance()` function, for RAS-/GRAS adjustment, perhaps the Danish Economic Councils and others could use that function for input output cells rather than hand-coded balancing in Gekko (via loops). The loops can be replaced by 1 function call.

The improved `laspchain()` function often only produces 1 extra observation in the beginning of the sample period, but even that could be important for users.

Regarding the data files that the MAKRO system produces while running, it was asked if perhaps an alternative to .gdx files could be offered, because GAMS is a commercial product. Gekko can read/write .gdx files with an unpaid GAMS version in demo mode, and in that way, Gekko can act as a bridge (transforming the .gdx files into Gekko .gbk files or for instance .parquet files).²

Regarding source code documentation, it was asked what would happen if the Gekko editor suddenly stopped working on Gekko? In that case, either Gekko would be frozen in time or someone else would develop it further (since it is open source). If the Gekko source code is migrated from .NET Framework to .NET Core and perhaps cleaned up in the process, an update of the source code documentation would perhaps be logical afterwards (with the help of AI).

There was a question of how the editor's potential "Gekko time" is to be measured exactly, as he is also doing other things at DST and DREAM (this is for DST and DREAM to decide on an as-needed basis, not the Gekko steering committee).

It seems like other users than DST could have some interest in a data versioning, to handle data that must not leave the premises of the organization (if versioning is used for program files in collaboration with other users, for instance via GitHub).

Regarding MONA, it was asked if there are features to make the model replicate some reference scenario. It was suggested to look into the SIM<after> statement that addresses the question of how to replicate values for the endogenous variables (via J-terms). Besides, ENDO/EXO and SIM<fix> can be used with endogenous variables as goals and J-terms as means.

When SIM (model solving) crashes, some users felt that Gekko's error messages are worse than GAMS', while other users felt the inverse.

It would be nice and logical for the PLOT window to be able to zoom by dragging and not using the zoom drop-down. This is noted for future improvements.

Regarding interfaces to dataframes, Polars (= Rust + Apache Arrow internally) was mentioned as probably being the future state of the art rather than Pandas.

² Gekko 3.3.5 will contain a GAMS-independent gdx read/write component.

Prioritization

- Regarding (1) Julia and (2) DUAL at Statistics Denmark: these are longer term tendencies that will not be directly addressed in the coming period.
- Regarding (3) and (4) porting to .NET Core and Avalonia (cross-platform GUI framework): this will be tried out in the coming period. AI will be used for the translation/migration, and particularly (3) migrating to .NET Core ought to be done before Microsoft suddenly stops security updates for the older .NET Framework. Migrating the Gekko windows to cross-platform Avalonia will only be done if AI can help out sufficiently. With some luck, (3) and (4) will be done, and then Gekko is well suited to run after .NET Framework is phased out. As a bonus, Gekko and PyGekko will be able to run cross-platform, for instance on a Linux server.
- Regarding (5) seasonal adjustment: the editor would very much like to get this done, not least to get the Quarterly National Accounts (DST) upgraded from Gekko 3.1.14. This is not a large project.
- Regarding (6) and (7) decomposition/flowgraphs/html browsers: the editor will clean up these parts of the source code a bit more, but no major reworkings are planned.
- Regarding (8) COMPARE window: the editor feels that such a window will pay off big time regarding the source project ("kildeprojektet") and ought to be done. This is not a large project.
- Regarding (9) data tracing, this seems to work ok for now.

Item 5. Organization and choice of editor for the next period

Thomas was elected as editor for the next period. Applause was given.

Item 6. Date of the next meeting + any other business

Middle of September 2027 was suggested regarding next meeting.