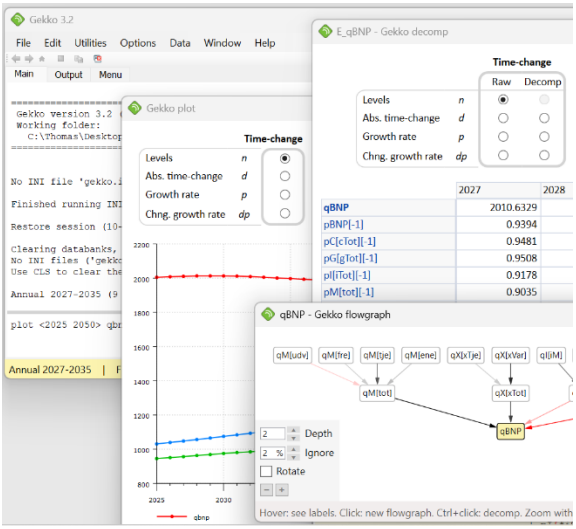
Re item 2. Editor's report regarding 2024-25[1]



First and foremost, an official Gekko 3.2 was released in August 2025. Differences between Gekko 3.0 and 3.2 can be seen here, but the release of 3.2 is actually not so much about new features (of which there are plenty relative to 3.0), but rather about wanting to get the help system re-written and updated in a comprehensive fashion. So the editor has read the help system from A to Z, has made a large number of corrections and additions, and has also erased any out-of-date parts. The new help system is present in Gekko 3.2 and any version in the coming 3.3.x series.

Since last year, there is a completely new PLOT window (Gekko 3.1.23 and on), using a layout inspired by the DECOMP window, and also using vector graphics. In the new PLOT window, it is possible to perform indexing (corresponds to new operator <i>), YoY (year-over-year, corresponds to new operator <yoy>), "All" to show first-position and reference databank values side by side (corresponds to new operator <a>), and other things. Databank swapping via the F2 window is activated again, which can be used for easier PLOT comparisons when many databanks are involved.

Also, regarding Gekko windows, there is a new flowgraph window (since Gekko 3.1.22). The flowgraph can be thought of as a visual representation of a recursive DECOMP, and DECOMP windows and flowgraph windows can be "browsed" in an integrated fashion.

Since the fall of 2024, Gekko has been able to produce a html browser (in form of a stand-alone browser showing the equations independently of Gekko) for GAMS scalar models[2] too, making it possible to show for instance MAKRO equations in a fashion similar to the use of Gekko's inbuilt DISP and DECOMP statements. The feature needs to be refined and wrapped up for general Gekko use (a html browser demo from a one year old MAKRO version and -databank can be seen here here, showing the variable qBNP).

[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.

[2] A GAMS scalar model is a version of the model where all sets, sum loops, $-expressions etc. have been removed/unrolled, leaving only the bare "atomic" equations, of which MAKRO typically has more than 1 mio. Scalar models are also what solvers like CONOPT are exposed to.

If available, scalar models are now also used for DISP, making DISP much more precise/detailed as an internal Gekko equation browser. Regarding MAKRO, the new res_... variables (naming logic) are now supported in DISP, DECOMP, etc.[3]

Regarding the INTERPOLATE statement (disaggregate timeseries from lower to higher frequencies), methods 'denton', 'cholette', and 'olsette' are now available. The last one is robust regarding relative data scaling, but 'Chow-Lin' for INTERPOLATE is still missing.

At last year's meeting in the steering committee, it was mentioned that some kinds of Gekko series statements ran slowly. This has been investigated, and in the case where composed series names like x{%i} were used, there was a bug regarding the series' meta information. A time stamp call took an unreasonable time, and slowed down the series statement a lot. This was fixed in Gekko 3.1.20: speed improvements for such statements are of an order of up to 10x.

From Gekko 3.1.23 on, there is an experimental new 'eps' concept available, inspired by GAMS. The idea is to introduce a value that can be very close to zero, but not exactly zero. This value can represent something that does not usually make sense, such as adult pupils in school classes etc. Data-wise, when a new period is added, such an eps value could be inherited from the previous period, so that it will not become a missing value per default. The ramifications are a bit philosophical, so eps is work in progress regarding for instance generating the MAKRO databank. Special rules apply regarding eps, for instance eps + eps = eps, eps – eps = eps, 5*eps = eps, etc.

Array-series are now better supported for IMPORT/EXPORT to for instance .xlsx, .csv and other data file formats, and the COPY statement supports array-series better, too.

Model format changes/enhancements for .frm have not been pursued, primarily because ADAM as a model is being phased out in any case, and other .frm users have not shown sufficient interest in enhancements. In any case, supporting GAMS- or GAMSPy-like syntax for equations and models in Gekko, rather than using the .frm syntax, would be more in line with other developments in Gekko.

Regarding the use of Gekko 2.4/2.5.x, there is now an ongoing project at Statistics Denmark about producing the MAKRO databank directly from data "sources", using Gekko 3.2 rather than Gekko 2.4. This also has to do with ADAM being phased out by the end of 2026 (after which the use of Gekko 2.4 is expected to end completely).

The Gekko webpage has been updated while releasing Gekko 3.2.

[3] These variables attempt to identify what is considered the "dependent" or "left-hand-side" variable in a given equation.

Gekko risk assessment → Gekko Roadmap 2025

The “Gekko risk assessment” paper was debated at last year’s steering committee meeting, and the paper has now been augmented/transformed/generalized from being mostly about .NET versions compatibility, to now also being about the possibility of Gekko as a package in a host language like for instance Python. Because of these augmentations, the paper has been renamed into “Gekko Roadmap 2025”. The Gekko editor has used quite a lot of time contemplating this roadmap paper, among other things also investigating different software trends, software packages, talking to users, etc.

This has provided Gekko editor with a rather clear view regarding how the Gekko project could – and perhaps ought to – progress towards a more package-based data science centric future. This process could be done in a stepwise fashion – provided there is sufficient interest in it, and the necessary funding.