Bruce McCarl and Chengcheng Fei's GAMS Newsletter Number 50

March 2025

This Newsletter addresses the following

1 GAMS Releases 47-49

Since the last newsletter there have been new releases 47 through 49 with the formal GAMS release notes available at https://www.gams.com/latest/docs/RN_MAIN.html. As we look at these notes we note GAMS is devoting most of its new feature related efforts towards related utilities like CONNECT, TRANSFER, ENGINE, MIRO, STUDIO and the alternative GAMSPY modeling approach within Python. Across these items when we peruse these notes the things that strike us to be of broad interest are:

1.1 GAMS system

Most all of the changes in the GAMS main system that we see in the GAMS release notes are bug fixes, added features to accommodate communication with other packages and minor command tweaks. Also there have been internal moves to improve efficiency which are largely transparent to the user except in faster execution times. There is a hint that a new compiler is coming but without details.

1.2 CONNECT

GAMS has now fully released CONNECT which provides a platform independent means of exchanging data from and to file types for use with other platforms. Currently CONNECT allows data exchanges with Microsoft Excel spreadsheets, SQL databases, GDX and CSV files. The

CONNECT framework consists of a database and a set of context-specific agents that do the moving of the data. One can also manipulate items within CONNECT. These agents need to be used in an ordered set of steps that first loads data from GAMS or a file in a supported format, then applies any desired transformations, and finally unloads the data into GAMS or a file for one of the supported formats. Instructions are passed using YAML syntax and STUDIO contains a YAML editor that opens when one creates a new or edits an existing file with YAML as the file extension.

New features in the CONNECT release as compared to the Beta include:

- An option tableOfContents is available when using the ExcelWriter agent that writes a table of contents of the material being placed into the Excel workbook. However unlike what we could add using GDXXRW that currently does not include hyperlinks to the data locations within the workbook.

- An option index within the ExcelWriter and ExcelReader agents that allows one to read instructions regarding locations in the workbook directly from an index sheet within the workbook. The index spreadsheet is defined as it was when using GDXXRW and is illustrated in https://www.gams.com/latest/datalib_ml/libhtml/datalib_GDXXRWExample14.html

To use CONNECT with ExcelReader and ExcelWriter one must have access to GAMS version 46 or newer.

Using CONNECT we have run into a few issues that one needs to be aware of

1. CONNECT is based on the Python language which requires one to follow certain formatting practices. This means spacing and indentations matter within your CONNECT code.

   For example, in the code below, if "-name: c" and other "-name:" statement are not indented and spaced in the same way, errors will occur with messages generated in the log file.

```
embeddedCode Connect:
- GAMSReader:
    symbols:
        - name: c
        - name: shipment
        - name: supply_value
- ExcelWriter:
    file: results.xlsx
    clearSheet: True
    symbols: all
endEmbeddedCode
```

Here one or more spaces must be provided between a ":" or a "-" and the next symbol. For example, the statement "symbols:all" will not work, but "symbols: all" does.

We recommend you follow the spacing and indentation in the examples in the CONNECT writeup.

2. CONNECT cannot directly transfer the variables' and equations' attributes (e.g. levels, margins) into other databases. Those values need to be written into parameters first, then one needs to write the parameters into other database formats.
3. The statement "symbols: all" is a powerful statement to dump all the symbol listed in the GAMSReader/ExcelReasder to databases/GAMS. But it is only available after in GAMS version 48 and later.
4. The CONNECT error messages are not as of now specific in the link to the GAMS code in pointing out where an error arises nor are they elaborated on in the LST file. We modified the CONNECT05 example to introduce a parameter c which was not given numerical values then tried to pass it to CONNECT as below.

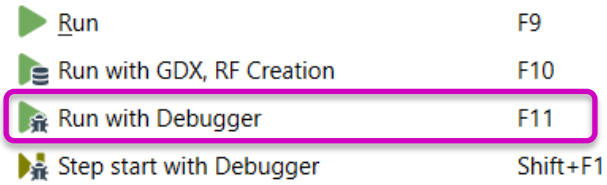
```
set i / i1*i3 /; alias (i,j,k);
parameter c;
table a(i,j) 'original matrix'
      i1     i2     i3
i1    1      2      3
i2    1      3      4
i3    1      4      3
;
$onEmbeddedCode Connect:
- GAMSReader:
    symbols:
      - name: a
      - name: i
      - name: c
      - name: k
- ExcelWriter:
    file: input.xlsx
    symbols:
      - name: a
$offEmbeddedCode
```

This resulted in the error message in the screenshot below. Note there the message points to the end of the embedded code. However note there is also a message in the LOG file saying there is a problem with the parameter c (the reporting of which we believe was just added in version 49). In recent work we had to employ a binary search over the embedded code to find the problem. We also had problems passing concatenated data back to GAMS when the size of the parameter assembled in CONNECT did not match with the GAMS declaration.

2 Studio

Studio has several new features along with repair of a few issues and one thing we hope gets better in the future. In particular

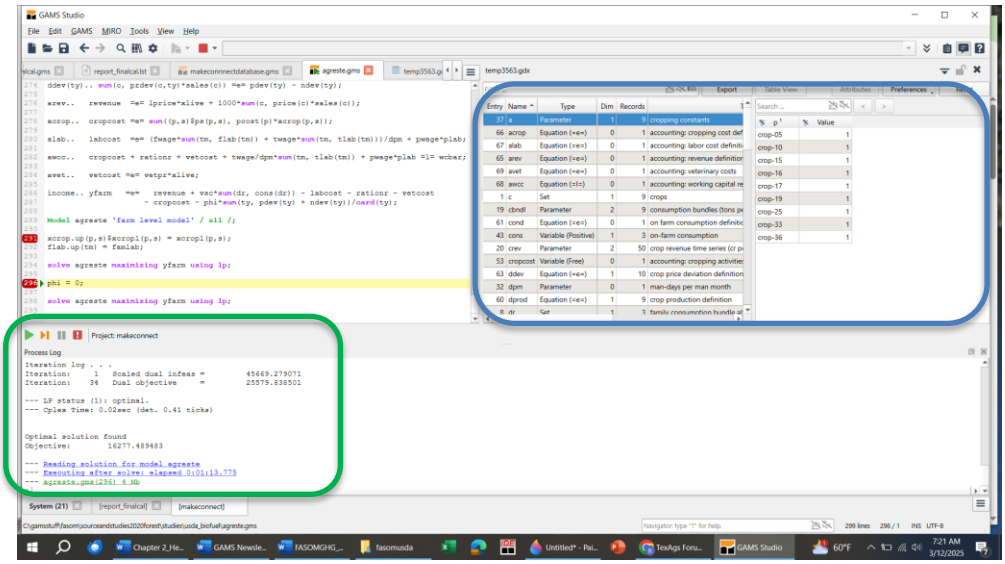
- Studio now remembers the command line parameters associated with main files.
- Studio does useful things in column insert mode (activated by Alt Shift) allowing one to type in a character that will be included in all the rows spanned by the vertical line marker. It also allows one to delete characters in all rows or identify blocks to delete or move.
- Studio now integrates a debugger that allows one to execute a job up to a point then the job pauses and allows one to examine the current data with the GDX viewer. To do this one needs to establish break points in the file (by using ctrl-click or right click on a line number and choosing to add one) and then run the file up to the first breakpoint using the STUDIO button run with debugger or the F11 key.



Execution then pauses at the first breakpoint where a GDX file is made available in a pinned window. That window access a GDX file that contains all of the current data for all declared items. One then can investigate those data and subsequently can continue execution either until to the next executable statement or the next breakpoint where the

program again pauses. At that point the GDX file is updated to reflect any further data manipulations.

The screenshot below shows the library file AGRESTE with breakpoints inserted. In that a) the line numbers with user defined breakpoints are marked in red on the left hand side, b) the GDX contents (outlined in blue) are in in the window to the right using the GDX editor that is resident in STUDIO, c) the process window which we have docked below (outlined in green) shows current status and buttons allowing execution continuation of various forms (continue to next executable statement or next breakpoint or stop execution) .



- Studio has some issues with its find routine which are on the list of things to fix but which we hope the fix of which bubbles to the top soon.
    - Use of file masks do not work well when searching over folders or multiple files. As of now the mask for files to include only seems to work for extensions (i.e. when one uses *.gms, this correctly limits the files searched to only those with a gms extension), but the mask does not seem to work for restrictions on filenames. For example, if we try to limit the search to r*.gms, this does not limit attention to gms files starting with r. However we do note the mask seems to be working in the files to exclude box (i.e. to avoid finding text in some STUDIO generated backup files we successfully use the exclusion mask ~*.gms,).
    - Studio forgets these file mask settings when it is closed down and they need to be reentered every time studio is restarted.
    - Studio always uses the last search type (i.e. in a file or a folder or a project), and for us this leads to a lot of multiple file searches in our case across large

directories when we only wanted to search in a file. Perhaps it would be good to have two buttons one for this file and one for multiple files.

3 Tools for Information Passage

Several new sets of utilities have been developed which pass things to and from GDX files and specific external file types. These are ExcelDump (this passes all the information it finds in multiple sheets in an Excel workbook into GAMS), CsvRead (Read an item from a CSV file into GAMS or GDX), CsvWrite (Write an item from GAMS or a GDX file into a CSV) and Sqlitewrite (Write an item from a GDX file into and SQLLITE database). All are used directly in a GAMS program and do not require the user to employ embedded code as would be needed is one was to use CONNECT. But those programs require creation of or reading from a GDX file and employ platform independent code to pass information when they are running. The writeup on these in the Tools manual is not very complete with missing files for at least ExcelDump and their use may require substantial dialogue with GAMS support.

4 Documentation

McCarl User Guide - GAMS has followed through with its desire to remove the McCarl Guide from the release as they think it confuses users and McCarl stopped updating it some time ago. This we feel removes a valuable option for particularly new users as it frequently contains a better or more complete explanation than does the official Guide for many topics. We did make sure the document was still available through the GAMS Forum at https://forum.gams.com/t/bruce-mccarls-documentation-collection-and-tools/3216 .

Moving data to Other Applications The tutorial originally developed by Erwin Kalvelagen on interfacing GAMS with other programs has been has been updated and renamed to Data Exchange with Other Applications. It provides guidance on how to exchange data with other applications using some of the latest recommended tools although the tools are restricted to Connect and a couple of those listed in the GAMS Tools collection. The prior document contained a lot more material on possible interface pathways. Interested users can search for the document using the keywords "**Kalvelagen Interfacing GAMS with other applications"** where some older copies are findable. Erwin may also have updates.

5 GAMSPy

Facing the growing prevalence of Python among computer users and the emergence of optimization related packages like PYOMO, the GAMS company decided to create a package that embedded the algebraic modeling features in GAMS for use within Python applications. I asked the GAMS staff to give me some material on GAMSPy and a lightly edited version of it follows.

GAMSPy is a Python package that allows users to formulate and solve optimization problems directly within Python. Essentially, it bridges the gap between the flexibility of Python and the robust optimization capabilities of GAMS.

A key strength of GAMSPy lies in its ability to facilitate the creation of set-based algebraic models directly within Python, leveraging Python classes and syntax to mirror commands and procedures embodied in the GAMS modeling language. This allows users to translate complex mathematical relationships into code that is both readable and maintainable.

The resultant version of GAMSPy allows users to define sets, parameters, variables, equations, and constraints within Python as objects and then create structured and modular models, where relationships between model components are clearly defined through Python classes.

Furthermore, GAMSPy's syntax is designed to closely resemble the mathematical set-based algebraic modeling notation that made algebraic modeling languages so successful, enabling users to write equations and constraints in a way that is both intuitive and mathematically accurate. This means that users familiar with GAMS can easily adapt to GAMSPy, while Python developers can benefit from the power of algebraic modeling without having to learn a completely new language. Data manipulation, model definition, and solution analysis can all be performed within the Python environment, streamlining the development process and enhancing productivity.

As of now, GAMSPy is in full release. GAMSPy is free to academic users without any restrictions on model size and is distributed with the ability to freely use a number of solvers (those not requiring a commercial license). Much more on capabilities and use is covered on GAMSPy in the writeup available at https://gamspy.readthedocs.io/en/latest/index.html# with more on the product page at https://www.gams.com/sales/gamspy_facts/ and licensing covered at https://gamspy.readthedocs.io/en/latest/user/installation.html . There are also GAMSPy specific release notes available.

6 GAMS Version for ARM64 Linux

GAMS is now available for the ARM64 processor in a LINUX version. GAMS staff provided us with the following on that.

The ARM processor family's energy efficiency and cost-effectiveness are driving its increasing prominence in cloud computing, particularly for optimization workloads. Within Amazon EC2, Graviton processors, based on ARM architecture, deliver significant performance-per-watt advantages. This trend is reinforced by the growing availability of optimization solvers on the ARM64 Linux platform. Leading solvers, including CPLEX, COPT, GUROBI, MOSEK, and XPRESS, have all been successfully ported, demonstrating the viability and performance of ARM for complex optimization tasks. This widespread solver support is further exemplified by GAMS's recent addition of ARM support, achieving near-complete solver availability.

This port will soon extend the benefits of Graviton machines to users of the GAMS Engine SaaS offering, making cost-effective and sustainable cloud-based optimization readily accessible.

7 Basic, Advanced and Combined courses offered soon

In July three classes will be offered online by Bruce McCarl. In particular, a **Basic** class will be offered in 4 sessions of 5 ½ hours each July 15-18 an **Advanced** class will be offered on July 17,18, 21 and 22 and a **Combined** class on July 15-18 and 21-22.

The **Basic** class is designed for users initially who know little about GAMS or are in need of a refresher but moves into more advanced application oriented topics. The **Advanced** class is best for those with some prior experience. The **Combined** course starts from the basics but goes well beyond the basics into advanced material.

7.1 What will you learn?

- The Basic Class over 4 days starts from ground zero teaching the basic GAMS syntax, covering ways of expressing models, making algebraic modeling work for you, examining your model to see if it looks right, using the GAMS integrated development environment in Studio, use of good modeling practices, enhancing model self documentation, and modeling firm problems. Then we turn to **accessing and using the documentation including the McCarl Guide, using conditionals and tuples, fixing misbehaving models, solving NLPs, doing comparative analysis, setting up a simple MCP based CGE model, linking to spreadsheets and data bases through Connect, using MIRO, improving output though report writing and information passage to other software**.
- The Advanced Class over 4 days starts with the bolded topics listed for the Basic class then spends much more time with STUDIO and fixing unbounded, infeasible and improperly solving models. The it goes on to treat pre-solution model checking, model scaling, use of advanced bases, fixing execution errors, speeding up code, managing memory use, working with solvers and a short introduction to the GAMSPY python link.
- The Combined Class over 6 days unifies the information from both the Basic and Advanced classes.

7.2 When is it taught

The classes are all offered in 5 ½ hour blocks starting in the early morning (7 am) US mountain time allowing access to those in other time zones. All materials are recorded and the recordings are made available for a year after the class. All course notes are provided in PDF form and all examples in GAMS runnable files.

Further information and other courses are listed on http://www.GAMS.com/courses.htm . Note I also give custom courses for individual groups a couple of times a year.

8 Unsubscribe or subscribe to future issues of this newsletter

Please unsubscribe through the web form available at: https://gams.us18.list-manage.com/unsubscribe?u=f1497c76718404eae593beb11&id=45ccea2ee0

Those who wish to subscribe to future issues can do this through the newsletter section of https://www.gams.com/newsletter/signup/.

This newsletter is not a product of GAMS Corporation although it is distributed with their cooperation.

March 14, 2025